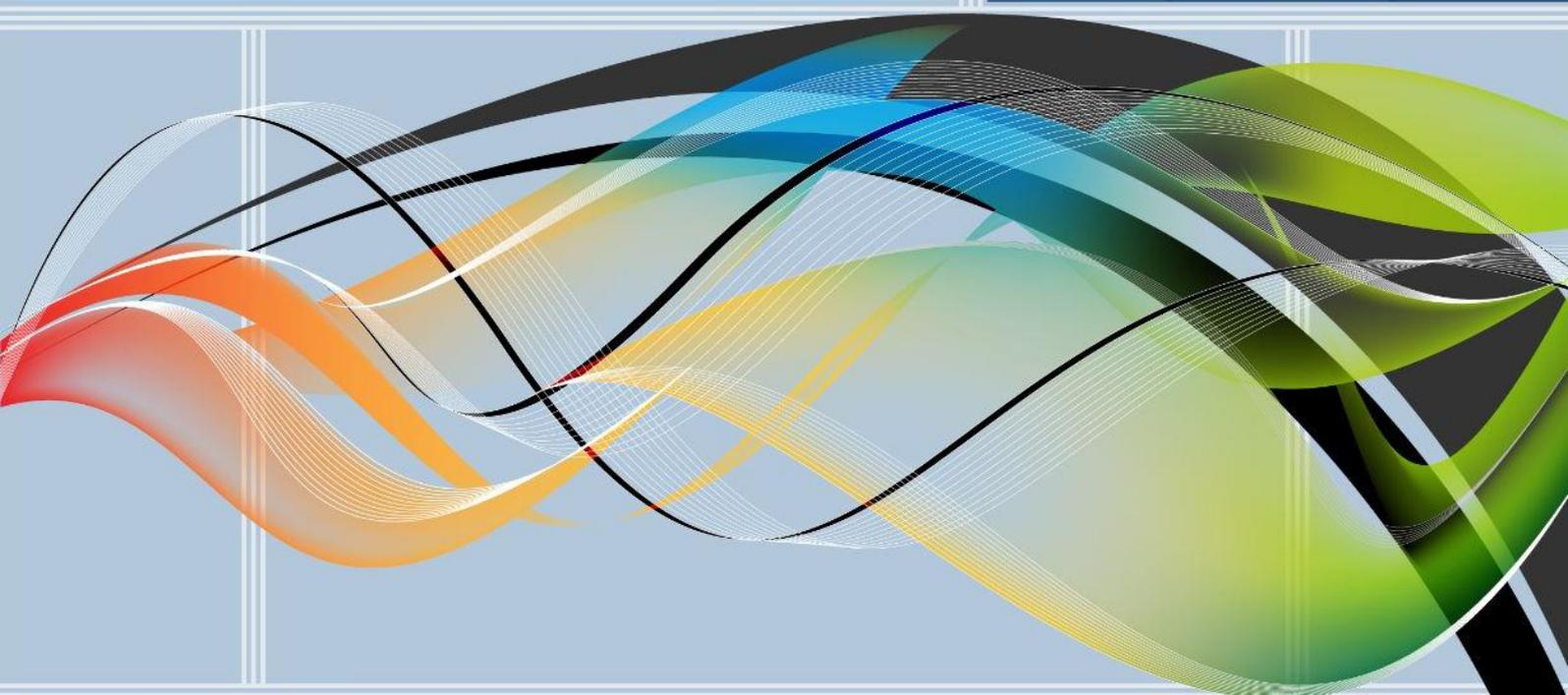


16+



XXV Всероссийская
студенческая научно-практическая
конференция Нижневартовского
государственного университета



Часть 3

Информационные технологии

Нижневартовск, 4-5 апреля 2023

Нижневартовск
NBГУ
2023

Министерство науки и высшего образования РФ
ФГБОУ ВО «Нижевартовский государственный университет»

**XXV Всероссийская студенческая
научно-практическая конференция
Нижевартовского
государственного университета**

Часть 3

Информационные технологии

*г. Нижевартовск,
4-5 апреля 2023 г.*

Нижевартовск
2023

УДК 001
ББК 72я43
Н60

16+

Печатается по решению Ученого совета
ФГБОУ ВО «Нижевартовский государственный университет»
(протокол № 2 от 31.01.2023 г.)

Н60 **XXV Всероссийская студенческая научно-практическая конференция
Нижевартовского государственного университета (г. Нижевартовск, 4-5
апреля 2023 г.) / Под общей ред. Д.А. Погоньшева. Ч. 3. Информационные
технологии. Нижевартовск: изд-во НВГУ, 2023. 206 с.**

ISBN 978-5-00047-681-9

Издание адресовано специалистам-практикам, педагогическим работникам,
научным сотрудникам, аспирантам и студентам.

ББК 72.0я43



Тип лицензии CC, поддерживаемый журналом: Attribution 4.0 International (CC BY 4.0).

ISBN 978-5-00047-681-9



9 785000 476819 >

© НВГУ, 2023

УДК 004.021

Абдразаков Д.А.
Волгоградский государственный университет
г. Волгоград, Россия

АЛГОРИТМ АВТОМАТИЗАЦИИ ПРОЦЕССА ВЫДЕЛЕНИЯ СПРЯМЛЕННЫХ УЧАСТКОВ СПИРАЛЬНОЙ СТРУКТУРЫ ГАЛАКТИК

Вереницы – полигональные рукава, образованные прямыми сегментами, являются одной из крупномасштабных особенностей в спиральной структуре галактик. По некоторым данным около половины известных галактик имеют спиральную структуру. Их исследование способствует пониманию эволюции как отдельно взятых объектов, так и всей вселенной; установлению физической природы некоторых явлений и их дальнейшее влияние на другие процессы [6].

Поиском и исследованием полигональных рукавов занимались такие ученые как: Б.А. Воронцов-Вельяминов, А.Д. Чернин и М.А. Бутенко [3; 9; 11]. Они составили каталоги со списками галактик с вереницами и их характеристиками, для чего потребовалась обработка, суммарно, свыше тридцати пяти тысяч изображений методом экспертной оценки – на глаз. Просмотр такого объема снимков требует значительных временных затрат, а также высокого уровня концентрации для уменьшения вероятности совершения ошибки. При данном подходе велика вероятность увидеть прямой сегмент там, где его нет, что естественным образом скажется на качестве исследования. Современный уровень развития информационных технологий предоставляет необходимый инструментарий для выполнения работ по автоматизации процесса выявления спиральных галактик с вереницами. Благодаря этому повысится качество и скорость исследований, направленных на понимание эволюции всей вселенной, установлению природы явлений в ней и их влияния на другие процессы и объекты [10; 5; 8].

В данной статье приводится описание алгоритма для автоматизации процесса выделения верениц в структуре спиральных галактик. На основе проводимых ранее другими исследователями работ и практического опыта был составлен и реализован алгоритм обработки снимков, представленный ниже [2; 7].

1. Адаптивная бинаризация по Гауссу. Методом рассматривается окрестность пикселя, за счет чего получается четче выделить рукава галактики.

2. Уменьшение размера изображения с сохранением пропорций. Приводит к уменьшению шумов и ускорения вычислений. Представленные ниже снимки были сжаты до двадцати процентов от исходного.

3. Пороговая бинаризация. Результатом выполнения предыдущего пункта может стать изображение в градациях серого, которое необходимо привести к черно-белому виду. Также, в зависимости от порога будет меняться детализация, что можно использовать для дополнительного уменьшения шумов.

4. Удаление пикселей, влиянием которых на конечный результат можно пренебречь, удаление шумов, шаблоны представлены на рисунке 1.

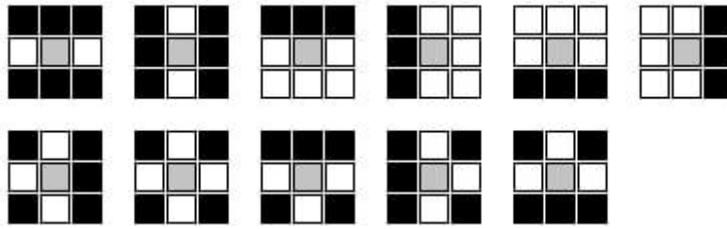


Рис. 1. Шаблоны удаляемых пикселей: серый – обрабатываемый пиксель; черный – пиксель объекта; белый – пиксель фона

5. Добавление пикселей, отсутствие которых на конечный результат приводит к неточности и разрывам, шаблоны представлены на рисунке 2.

6. Скелетизация по алгоритму Зонг-Суня. В результате этой операции изображения рукавов галактики становятся толщиной в один пиксель, увеличивая четкость спрямленных сегментов [1; 4].

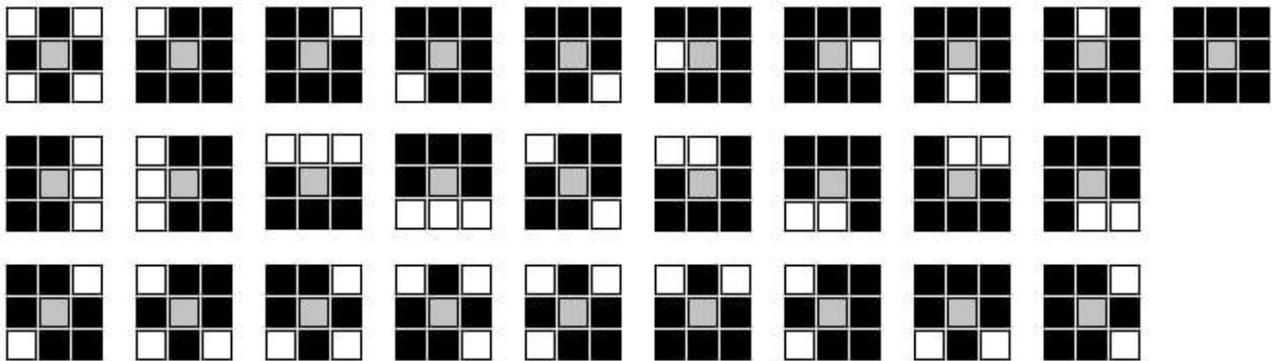


Рис. 2. Шаблоны добавляемых пикселей: серый – обрабатываемый пиксель; черный – пиксель галактики; белый – пиксель фона

Демонстрация работоспособности представленного алгоритма показывается на примере снимков галактики M 51, NGC 1300, NGC 4254 и PGC 69533 с сайтов HeperLEDA (<http://leda.univ-lyon1.fr>) и ESA/Hubble (<https://esahubble.org>), рисунки 3-6.

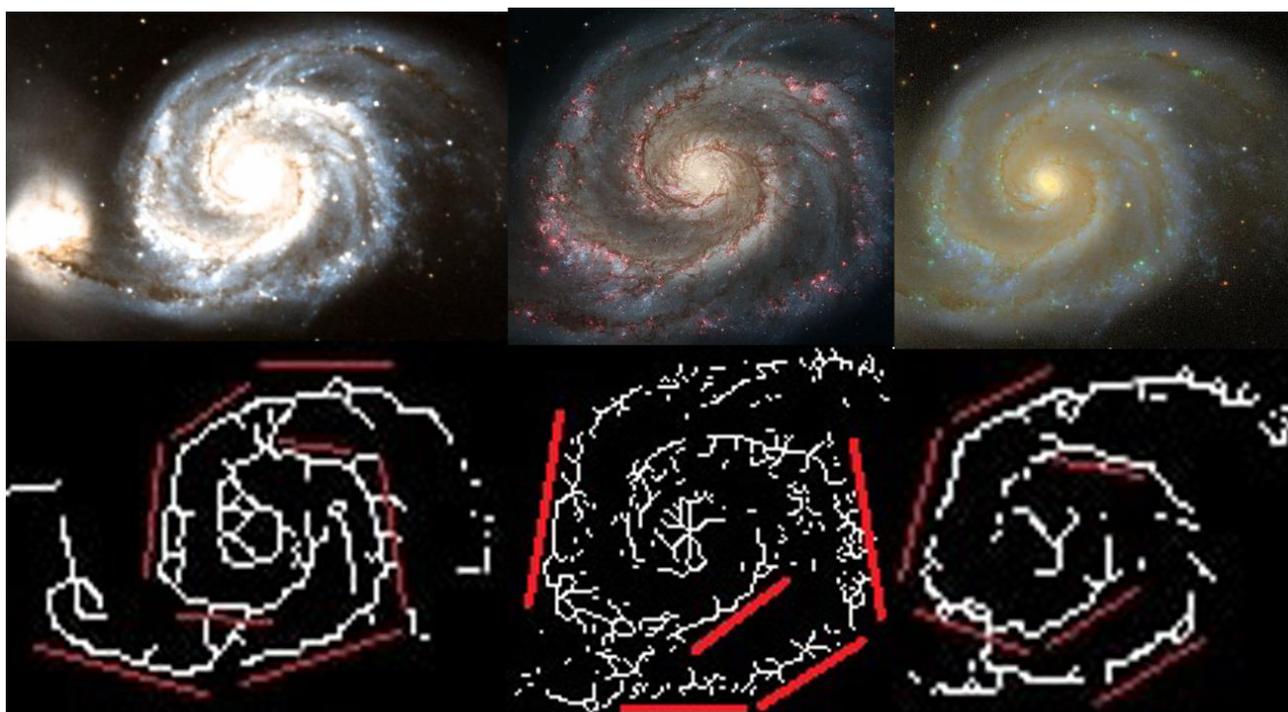


Рис. 3. В верхнем ряду представлены исходные изображения галактики М 51 до обработки: а) DSS; б) Hubble; в) SDSS9. Нижний ряд содержит те же изображения после обработки

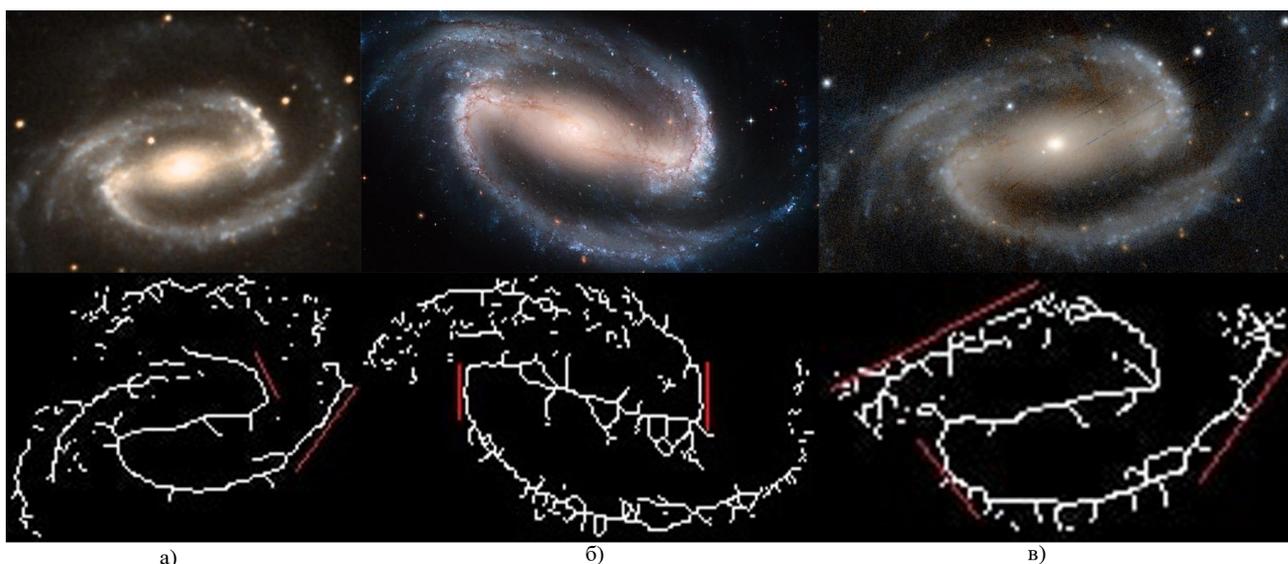


Рис. 4. В верхнем ряду представлены исходные изображения галактики NGC 1300 до обработки: а) DSS; б) Hubble; в) Pan-STARRS. Нижний ряд содержит те же изображения после обработки

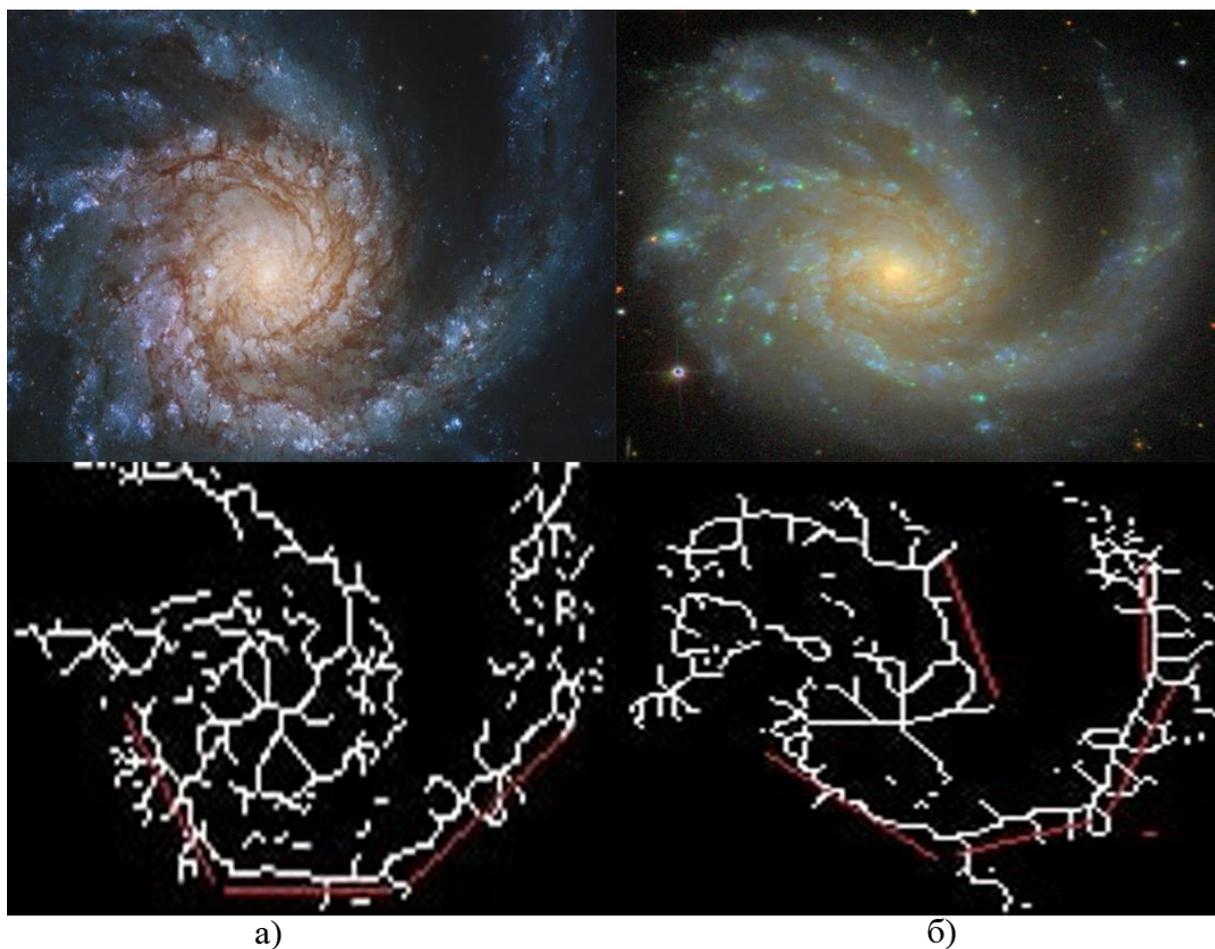


Рис. 5. В верхнем ряду представлены исходные изображения галактики NGC 4254 до обработки: а) Hubble; б) SDSS9. Нижний ряд содержит те же изображения после обработки

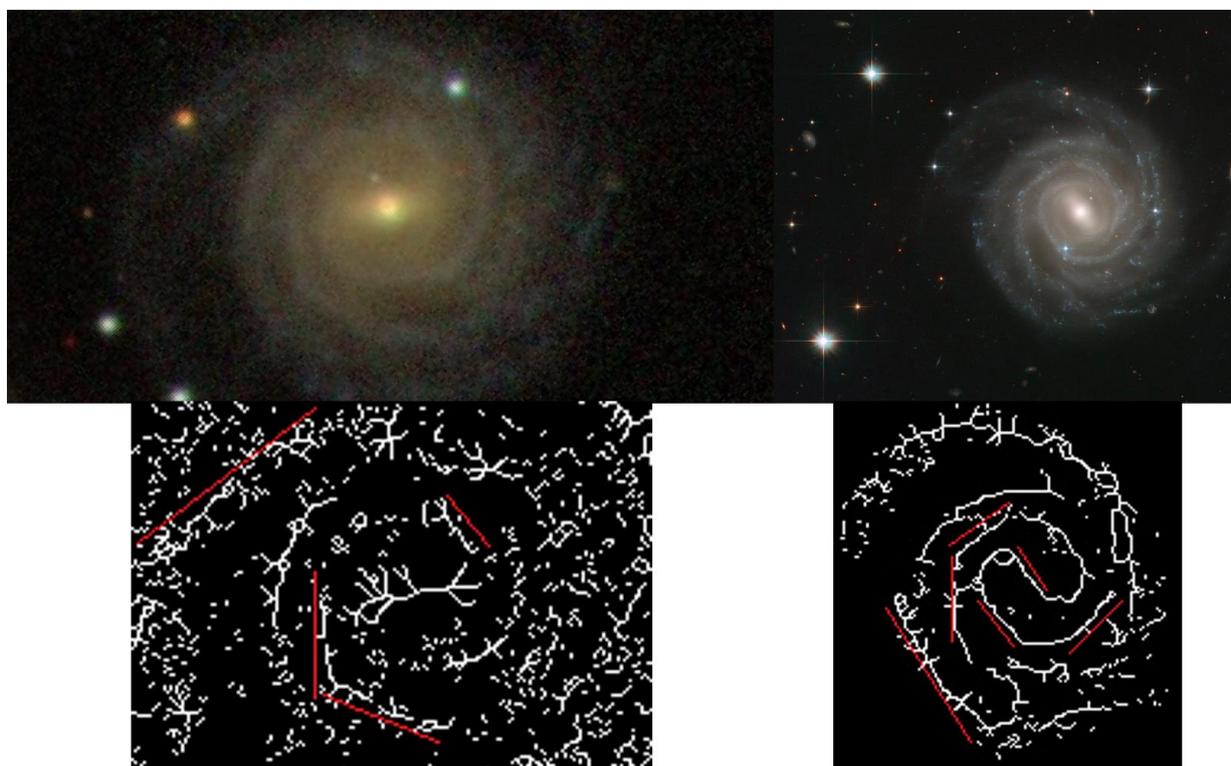


Рис. 6. В верхнем ряду представлены исходные изображения галактики PGC 69533 до обработки: а) SDSS9; б) Hubble. Нижний ряд те же изображения после обработки

После обработки нескольких изображений спиральных галактик по описанному алгоритму были сделаны следующие выводы. Во-первых, качество исходного снимка сильно влияет на результат. Множество разрывов и кривых на снимке телескопом Hubble вызвано его большим разрешением и детализацией. Во-вторых, необходим подбор параметров для методов бинаризации и скелетизации, что может потребовать дополнительных затрат времени при анализе снимков. Из-за описанной в первом пункте проблемы не представляется возможным подобрать оптимальные параметры для всех снимков. Тем не менее, описанный алгоритм пригоден для выделения прямых сегментов в структуре спиральных галактических узоров.

Литература

1. Баранов Р.П., Фаворская М.Н. Алгоритмы скелетизации объектов на изображении // Актуальные проблемы авиации и космонавтики. 2011. №7. С. 349.
2. Бутенко М.А. Статистическая обработка изображений спиральных галактик с полигональными структурами // Вестник Волгоградского государственного университета. 2015. Т. 26. №1. С. 52-60.
3. Бутенко М.А., Хоперсков А.В. Галактики с «вереницами»: новый каталог // Астрофизический бюллетень. 2017. Т. 72. №3. С. 256-275.
4. Бушенко Д.А., Сдыхов Р.Х. Технология построения алгоритмов утоньшения для скелетизации протяженных объектов // Доклады БГУИР. 2009. №7(45). С. 81-86.
5. Дацко В.С. Геометрический анализ структуры спиральных галактик // Вестник международного университета природы, общества и человека «Дубна». 2016. №3(35). С. 3-7.
6. Красильников Н.Н. Цифровая обработка 2D- и 3D-изображений. СПб.: БВХ-Петербург, 2011. 608 с.
7. Филистов Е.А. Полигональная структура спиральных галактик // Астрономический журнал. 2012. Т. 89. №1. С. 12-18.
8. Фоменко И.С., Кузьмин Н.М., Бутенко М.А. Скелетизация изображений спиральных галактик на космических снимках // Прикладная математика и информатика: современные исследования в области естественных и технических наук. 2019. С. 460-462.
9. Chernin A.D. et al. Galaxies with Rows // Astronomy Reports. 2001. Т. 45. С. 841-853.
10. Quinn J. Spiral Structure on FIRE // Bulletin of the AAS. 2022. Vol. 54. no. 6.
11. Vorontsov-Velyaminov B.A. On the nature of a new feature in the galaxy M 81 // Soviet Astronomy. 1967. Т. 10. С. 1057-1058.

© Абдразаков Д.А., 2023

УДК 004.5

Аввакумов Д.А.

Пермский национальный исследовательский политехнический университет
г. Пермь, Россия

ПРИЛОЖЕНИЕ ДЛЯ ВИЗУАЛИЗАЦИИ И АНАЛИЗА РЕЗУЛЬТАТОВ МОДЕЛИРОВАНИЯ БИОМЕХАНИЧЕСКОГО УЗЛА. ИНТЕРФЕЙС ПРИЛОЖЕНИЯ

Твердые ткани зубов (дентин и эмаль) имеют сложное строение и уникальное сочетание прочностных свойств, что позволяет им выдерживать воздействие механических нагрузок и находиться в динамическом равновесии с биохимическими процессами в течение длительного времени [2, с. 7]. Но зуб может подвергаться негативным воздействиям и разрушаться. Одной из технологий, которой располагает современная стоматология, является применение вкладки для реставрации разрушенных зубов [3, с. 39]. Совершенствование процесса лечения дефектов зуба путем повышения качества вкладок, достигается только при правильном выборе материала и технологий их изготовления. Решение данной задачи значительно упрощается с применением информационных технологий.

Компьютерное моделирование и опыт поведения различных материалов позволяют использовать их в сфере медицинских исследований [4]. Информационные технологии могут широко применяться в современной стоматологии, например для определения напряженно-деформированного состояния и нахождения благоприятной для зубов нагрузки [1, с. 201].

Наше приложение создается с целью визуализации и дальнейшего анализа поведения материалов для реставраций зуба при различном уровне внешней нагрузки. Разработанное программное обеспечение позволяет визуально спрогнозировать поведение того или иного материала при различных нагрузках и условиях. Интерфейс приложения включает в себя списки с возможностью выбора условий – материал реставрации, величина нагрузки и рассматриваемый параметр. Также интерфейс позволяет загружать рентген-изображение для выбора и сохранения необходимых параметров. На основе выбранных условий пользователь получает возможность просмотреть изображения напряжений и деформаций на модели зуба и его элементах. Описанный выше функционал поможет стоматологам сделать быстрый выбор подходящего материала для изготовления зубной вкладки с учётом физиологических особенностей конкретного пациента. Интерфейс программы при этом должен быть понятен практикующим врачам.

При разработке интерфейса программы выбраны следующие средства реализации:

- 1) язык программирования Python, данный язык легок в освоении, при этом имеет множество библиотек, в частности PyQt для создания интерфейса;
- 2) средство создания пользовательского интерфейса в связи с выбранным языком программирования – PyQt.

Важной задачей является реализация интерфейса пользователя, с помощью которого он будет настраивать и выбирать необходимые данные для последующей визуализации. Система представляет собой приложение в отдельном окне, имеющем 2 вкладки: интерфейс первой вкладки включает в себя выпадающие списки с возможностью выбора: материала вкладки,

величины нагрузки и анализируемого параметра. Данный интерфейс также содержит кнопку для визуализации данных в виде графиков, отдельные области окна, в которых будут отображаться диаграммы на основе данных и изображения моделей зуба.

При нажатии на кнопку визуализации подгружаются изображения с моделями зуба по заданным ранее условиям (величина нагрузки и материал). Изображения можно менять при нажатии отдельных кнопок переключения или при нажатии на клавишу вправо/влево. Также изображения можно увеличивать нажатием на отдельную кнопку, при этом создается новое окно, в котором можно подробнее его рассмотреть. На рисунке 1 представлено окно с увеличенным изображением.

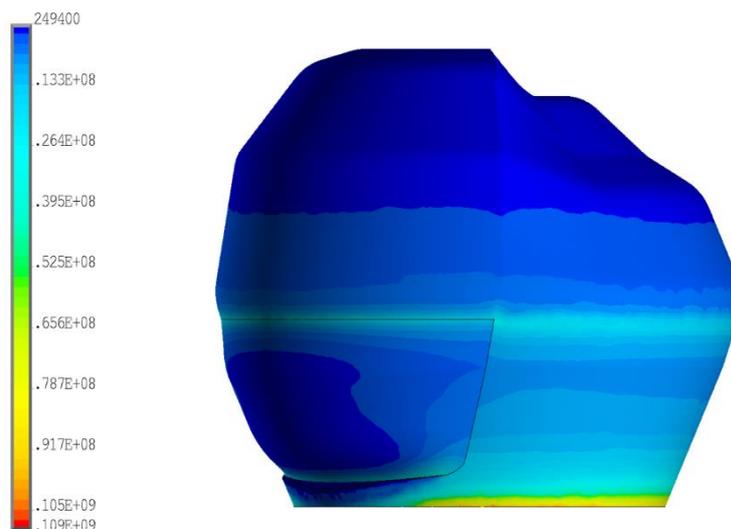


Рис. 1. Окно с увеличенным изображением

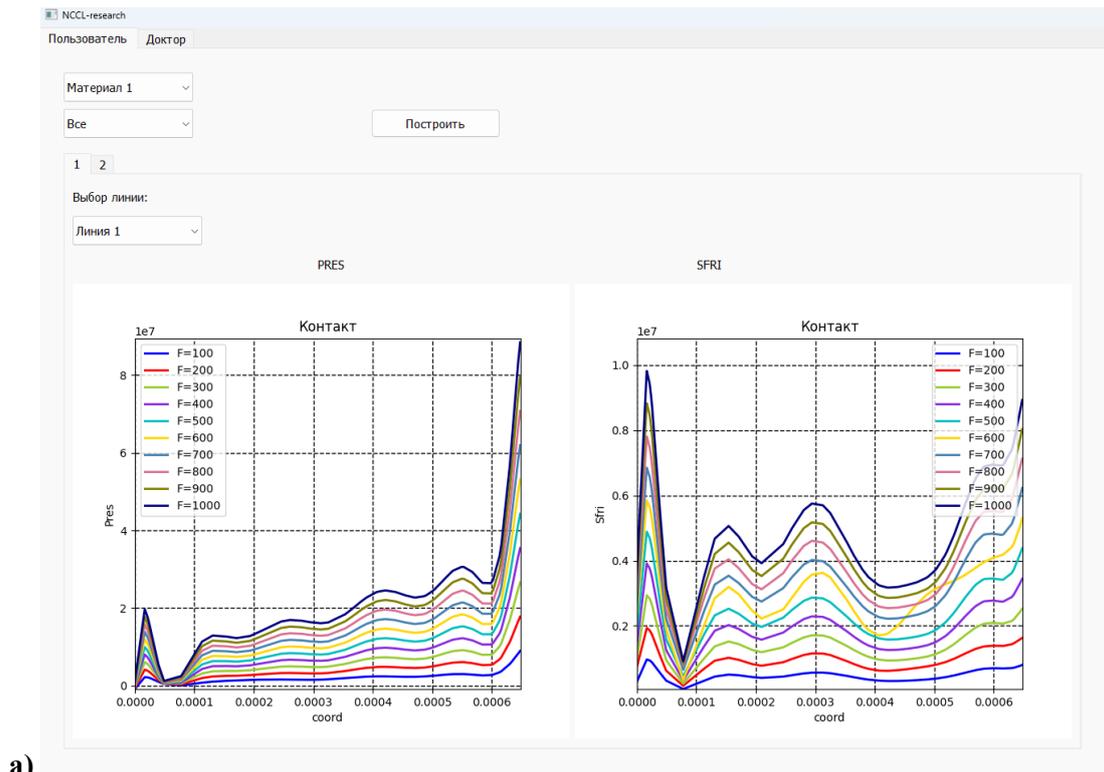
Виджет для отображения вкладок включает в себя 2 дополнительные вкладки, с помощью которых пользователь может удобно производить сравнение и анализ данных.

В первой вкладке отображаются 2 графика: контактные давление и трение. Необходимо учитывать, что значения, по которым строятся графики, зависят от выбранных пользователем данных в выпадающих списках, находящихся в верхней области интерфейса.

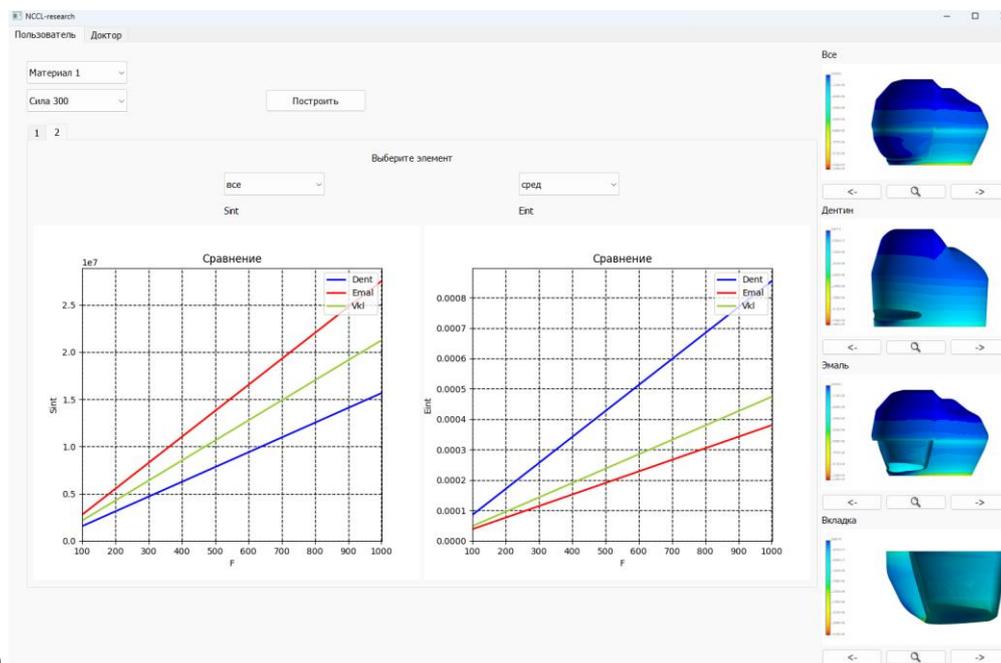
Во второй вкладке создаются дополнительные виджеты для выпадающих списков с выбором элемента: эмаль, дентин, вкладка или все. При выборе «все» создается дополнительный выпадающий список с выбором: максимальных, средних или минимальных значений параметров. В зависимости от того, что было выбрано во втором списке, стоит задача выстраивать график со всеми элементами с выбранными значениями параметров. Если в первом списке были выбраны пункты «эмаль», «дентин» или «вкладка», то график строится с минимальными, средними и максимальными значениями выбранного элемента.

В ходе работы также были рассмотрены различные методы сглаживания данных, чтобы привести графики к наиболее удобному для анализа виду, в частности метод наименьших квадратов и аппроксимация сплайнами, реализованная через специальную библиотеку. В ходе тестирования было установлено, что метод наименьших квадратов имеет наименьшую абсолютную погрешность, следовательно, графики получаются более точными, поэтому в дальнейшей работе использовался именно данный способ аппроксимации.

Примеры работы первой вкладки программы представлены на рисунке 2.



а)



б)

Рис. 2. Интерфейс первой вкладки приложения: а) окно 1, б) окно 2

Вторая вкладка данного окна создана для того, чтобы пользователь-врач мог загружать рентген-изображение зуба, на котором есть возможность выбирать необходимые параметры для вычислений. На рисунке 3 представлен общий вид интерфейса второго окна.

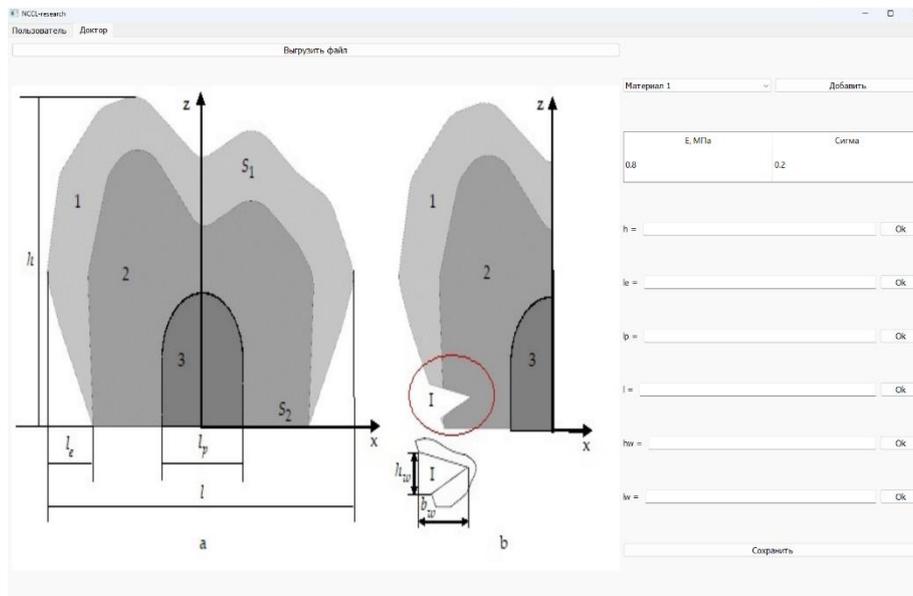


Рис. 3. Общий интерфейс вкладки

Для выбор параметров на вкладке добавлена кнопка для выгрузки изображения, нажимая на которую стоматолог может выбрать необходимое изображение среди имеющихся. Каждый параметр, необходимый для вычислений, имеет свою строку, в которую заносится его значение таким образом: врач выбирает необходимый ему параметр нажатием на строку, а дальше, щелкнув левой кнопкой мышки по картинке, задает его величину.

Также на данной влкладке добавлен виджет выпадающий список, чтобы пользователь мог выбрать необходимый материал. При выборе материала его данные, в частности модуль Юнга и коэффициент Пуассона, отображаются в специальной таблице. Сама таблица с материалами выгружается при запуске приложения из специально отведенного файла. Если материал не подходит конкретному пациенту, врач может задать свой собственный. Для этого добавлена специальная кнопка «Добавить»: при её нажатии открывается новое окно с таблицей по всем материалам и их параметрам, чтобы пользователь мог одновременно оценить все материалы. Для добавления нового материала необходимо нажать на кнопку «+», в таком случае создается еще одно окно, где врач может самостоятельно задать название материала, модуль Юнга и коэффициент Пуассона. При этом, модуль Юнга должен быть неотрицательным, а значение коэффициента Пуассона должно лежать в интервале от 0 до 0.5 неключительно. Если врач пользователь ввел неверные параметры, при попытке добавления материала высветится сообщение о том, что параметры были введены неверно и их необходимо исправить. Если все введено верно, то материал будет добавлен в общую таблицу, и его можно будет выбрать для конкретного пациента.

После выбора материала и других параметров вы можете сохранить их нажатием на кнопку «Сохранить». При нажатии на кнопку обновленная таблица загружается в уже существующий файл, а параметры выгружаются в отдельный файл для дальнейших вычислений в ANSYS Mechanical.

На рисунке 4 представлен интерфейс дополнительных всплывающих окон для задания собственного материала.

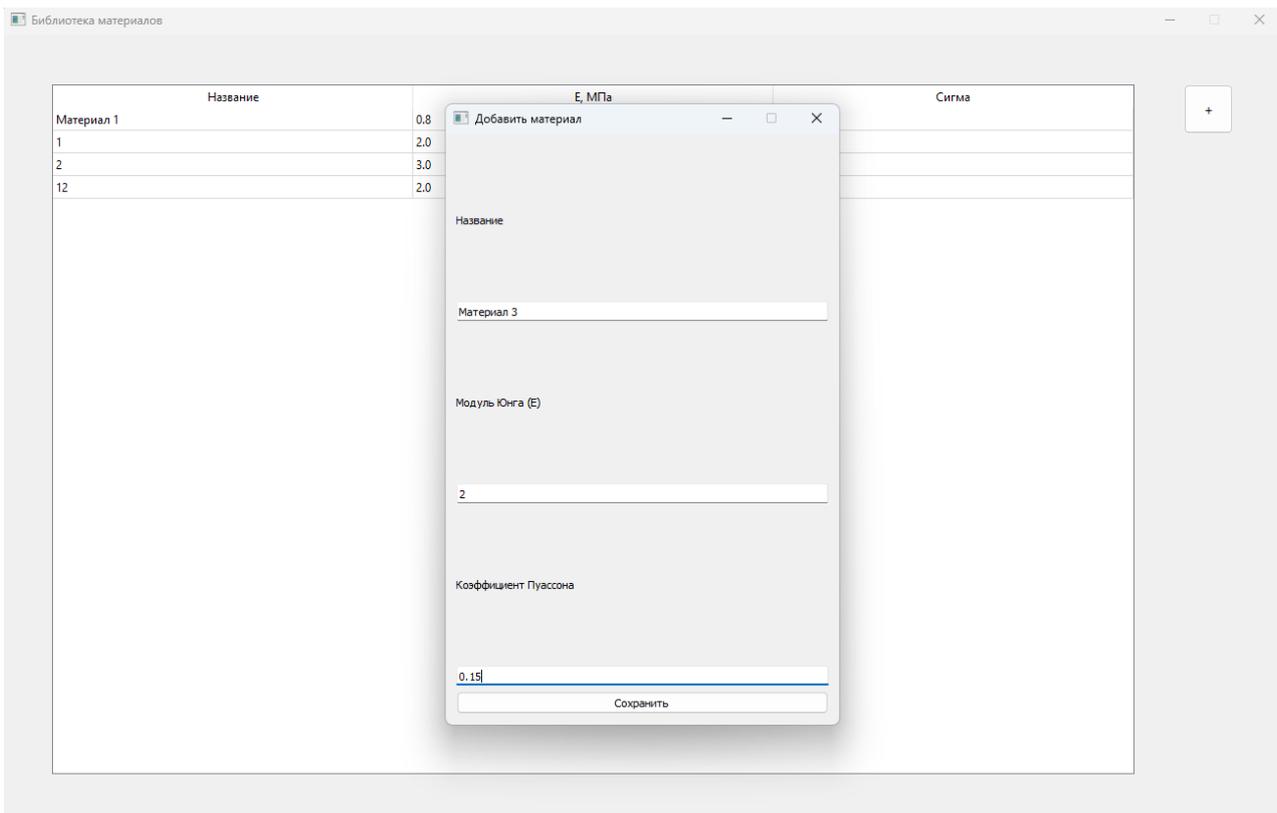


Рис. 4. Дополнительные окна для задания материала

В результате работы создан интерфейс специализированного приложения для практикующих врачей-стоматологов, которое позволяет облегчить анализ механического поведения биомеханического узла, посредством визуализации поведения материалов для вкладок зуба при различном напряжении с целью дальнейшего анализа и выбора оптимального материала.

Литература

1. Босяков С.М., Мселати А.Ф., Круподеров А.В. Математическое моделирование начальных перемещений корня зуба в форме двуполостного гиперболоида // Белорусский государственный университет. 2015. Т. 19. №2. С. 186-204.
2. Зайцев Д.В. Физические механизмы деформации и разрушения в материалах с развитой иерархической структурой. Дентин и эмаль зубов: Автореф. дис. ... доктор физ-мат. наук. Екатеринбург, 2015. 277 с.
3. Зотов П.П. Использование методов математического моделирования для совершенствования подходов к выбору технологий восстановления разрушенных зубов // Dental forum. 2012. № 3. С. 39-40.
4. Kamenskikh A.A., Sakhabutdinova L., Astashina N., Petrachev A., Nosov Y. Numerical modeling of a new type of prosthetic restoration for non-carious cervical lesions // Materials. 2022. Т. 15. С. 5102. <https://doi.org/10.3390/ma15155102>

© Аввакумов Д.А., 2023

УДК 004.93:004.77

Аксёнова Н.А., Соболев Д.Н., Сыч Д.С.

Гомельский государственный университет имени Франциска Скорины
г. Гомель, Беларусь

РАЗРАБОТКА СИСТЕМЫ МАШИННОГО ЗРЕНИЯ ДЛЯ РАСПОЗНАВАНИЯ ОСОБЫХ ТОЧЕК УГЛОВ

Основной целью машинного зрения является принятие решений о реальных физических объектах и сценах для решения конкретной задачи. Это обуславливает тесное взаимодействие машинного зрения с областью обработки изображений для решения следующего вида задач:

1. Анализ основных компонент изображения: уровней пикселей, вращение операций, контрастность, шумоподавление и т. п.
2. Отслеживание/обнаружение/распознавание объектов.
3. Выделение ключевых особенностей, например, краев изображения, углов, контуров.
4. Восстановление 3D формы объекта по 2D-изображению.

Целью настоящего исследования является поиск подходящих технологий для реализации системы машинного зрения. Основной проблемой при разработке подобного рода систем является выбор уникальных признаков, остающихся инвариантными к разного рода преобразованиям. В качестве доминирующего признака были выбраны угловые дескрипторы [1]. Определение углов часто используется при обнаружении движения, регистрации изображений, отслеживании видео, создании мозаики изображений, сшивании панорам, 3D-реконструкции и распознавании объектов.

Распознавание информации – это сложный процесс идентификации, осуществляемый на основе определенных признаков объекта, которые выводятся в выходную информацию. Эта выходная информация обычно указывает на принадлежность объекта определенному классу.

Для того чтобы успешно выделить интерпретируемую информацию из изображения, необходимо учитывать его локальные особенности. Основным инструментом для этого является детектор особых точек, который обеспечивает инвариантность распознавания объектов, несмотря на преобразования изображения.

Каждая особая (ключевая) точка имеет свой дескриптор, который является уникальным идентификатором точки, отличающим ее от остальных. Дескрипторы также должны быть инвариантными, чтобы учитывать различные преобразования изображения и обеспечивать точное соответствие между особыми точками.

Для распознавания изображений в данной работе предлагается реализовать систему машинного зрения, состоящую из камеры и программного обеспечения. Камера создает цифровой эквивалент входного изображения. Полученное изображение размещается на компьютере и распознается с помощью разработанных программных модулей. Найденные ключевые точки будут использоваться для построения 3D-моделей в редакторе 3D-графики Blender.

Чтобы выбрать оптимальный язык программирования для реализации системы компьютерного зрения, провели сравнительный анализ. Из таблицы 1 видно, что Python имеет больше всего преимуществ по сравнению с другими языками программирования. Кроме того, одно из условий проекта является поддержка выбранного языка в среде Blender для возможности подключения в качестве дополнения. Blender поддерживает только дополнения на языке программирования Python. Python имеет множество превосходных пакетов для обработки изображений, однако большинство из них основываются на представлении данных в виде многомерного массива NumPy.

Таблица 1

Сравнительный анализ языков программирования

Характеристики	Язык программирования				
	C	C++	C#	Java	Python
Поддержка ООП (объектно-ориентированного программирования)	-	+	+	+	+
Возможность компиляции в машинный код	+	+	-	-	+
Поддержка обработки исключений	-	+	+	+	+
Статическая проверка типов	+	+	-	+	-
Кроссплатформенность	+	+	+	+	+
Возможность интеграции с уже существующими инструментами на предприятии	-	+	-	-	+

Для реализации системы машинного зрения была выбрана библиотека OpenCV (Open Source Computer Vision Library). Она содержит функции, способствующие обработке изображения с помощью набора операций, не обращая внимания на то, как изображения хранятся в памяти. Примерами таких функций являются, например, получение версии изображения в градациях серого, изменение размера части изображений, фильтрация изображений по цветам, обнаружение контуров и многое другое. Использование этой библиотеки позволяет программисту с легкостью построить алгоритм компьютерного зрения [2]. Данная библиотека реализована на языке программирования C/C++, при этом разрабатывается и для других языков, например, таких как Python, Java, Ruby, Matlab, Lua. Библиотека OpenCV имеет лицензионное соглашение BSD, что позволяет свободно использовать её для научных и коммерческих целей.

Разрабатываемая библиотека должна иметь следующие функции:

1. Поддерживать большинство современных форматов графических файлов: PNG, JPEG, BMP, TIFF, GIF и SVG.
2. Идентифицировать особые точки углов.
3. Обеспечивать выбор методов распознавания углов.
4. Отображать количество и скорость распознавания особых точек углов.

Для реализации проекта была произведена декомпозиция по методологии IDEFO. Общий процесс разработки представлен на рисунке 1. На входе процесса реализации «Задачи и цели проекта», на выходе ожидается получить «Библиотеку». Механизмами разработки являются инструменты, с помощью которых ведется разработка: инженер, команда разработки,

компьютер. За управления отвечают бизнес-требования к нормализации данных, а также документации тех или иных инструментов.

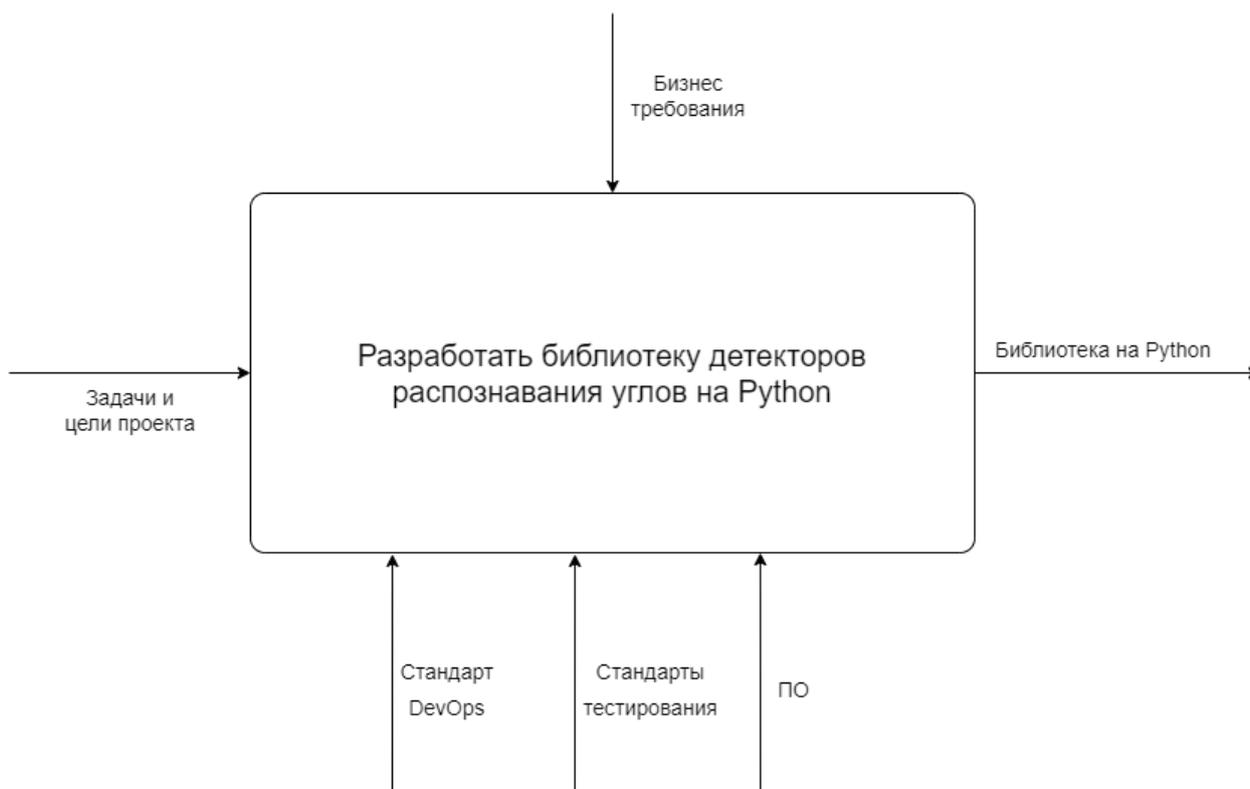


Рис. 1. IDEF0 диаграмма проекта

Далее процесс разбивается на ключевые шаги, к которым и применяются механизмы и элементы управления для упрощения дальнейшей разработки.

В данном случае выделены следующие шаги:

- разработка требований;
- выбор детекторов углов;
- реализация модуля;
- апробирование;
- тестирование.

После распределения механизмов и элементов управления получим финальный вариант диаграммы (рис. 2).

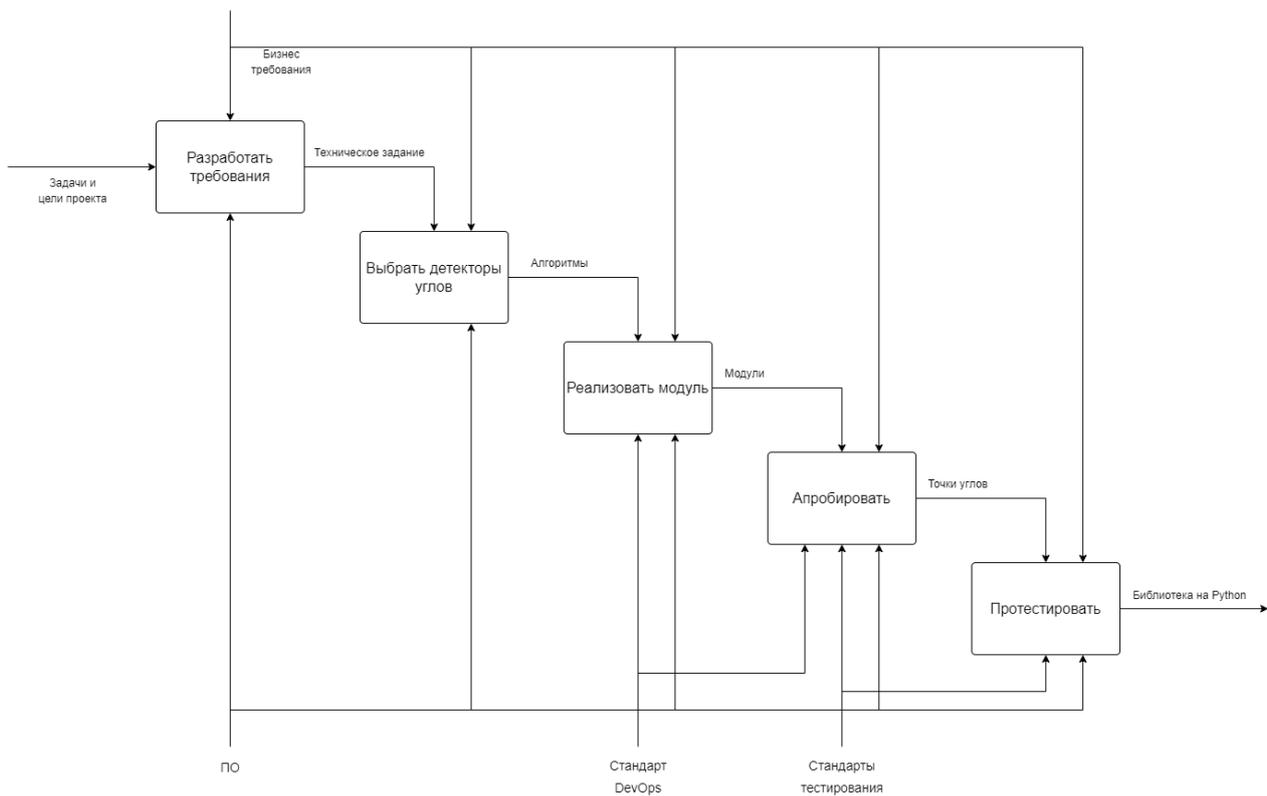


Рис. 2. Детализированная IDEF0 диаграмма

Выделяют различные типы углов в зависимости от количества пересекаемых граней: L-связанные, Y-связанные или T-связанные, X-связанные, а также выделяют стрелковидно связанные углы (рис. 3). Все из перечисленных видов углов могут быть распознаны по-разному при использовании различных угловых детекторов.

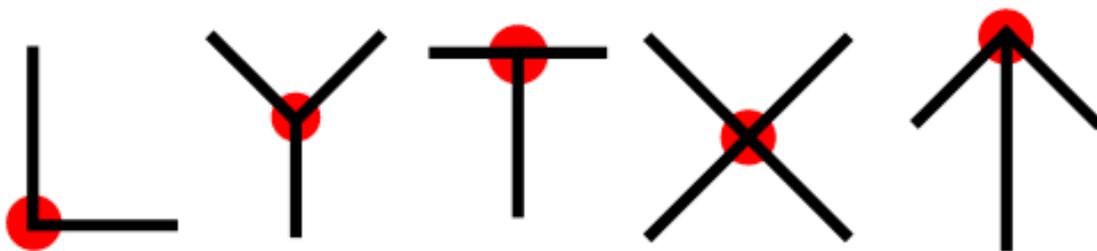


Рис. 3. Виды углов

Классификация методов определения ключевых точек проведена по трем основным направлениям:

1. Методы, основанные на интенсивности изображения, позволяют определить ключевые точки из значений интенсивности пикселей. Идентификация углов основывается в основном на областях с большим изменением интенсивности градиента во всех измерениях и направлениях.

2. Методы, использующие контуры изображения, позволяют определить контуры и места с максимальным значением кривизны, либо производить полигональную аппроксимацию контуров и определять пересечения. Однако данные методы зависят от

окрестности пересечений и могут давать неточные результаты, когда пересекаются три и более края.

3. Методы на основе модели используют модели с параметрами интенсивности, которые приспособливаются к изображениям-шаблонам до максимальной точности. Однако у данных методов имеются ограничения в применении, а их применения сводится для определения ключевых точек определенного типа, таких как L-связные углы. Кроме того, результаты работы методов на основе модели зависят от применяемых шаблонов.

На практике обширно применяются методы, базирующиеся на интенсивности изображения.

В таблице 2 приведено сопоставление детекторов углов, где для каждого метода и критерия выставлена оценка от 1 до 10.

Таблица 2

Сравнительный анализ детекторов углов

Алгоритм	Производительность	Регулярность	Приспособленность к помехам	Время выполнения	Местоположение
Deriche	6	8	4	8	8
Wang and Brady	8	8	6	8	8
Moravec	6	7	6	8	8
Kitchen and Rosenfeld	6	5	7	6	6
Forstner	8	7	7	5	9
Plessey	8	7	5	6	4
Beaudet	6	5	4	8	6
SUSAN	8	5	10	8	5
CSS	8	10	8	Зависит от используемого детектора границ	8
Trajkovic and Hedley (4-neighbours)	5	4	5	10	8
Trajkovic and Hedley (8-neighbours)	6	7	8	10	8
Zheng and Wang	8	6	7	7	5

В данной работе было проведено исследование и описана разработка системы машинного зрения для выделения ключевых характеристик изображений, в частности, особых точек углов. В статье были описаны основы и методы поиска углов. Сравнительный анализ алгоритмов распознавания углов показал, что для реализации проекта будут использоваться алгоритмы Moravec, Forstner, Plessey, CSS, SUSAN и FAST. Реализация проекта будет осуществляться на Python с применением библиотеки компьютерного зрения OpenCV. Были поставлены задачи и была разработана основная функциональность проекта.

Разрабатываемое приложение будет применяться в качестве дополнения в среде Blender для процесса моделирования трехмерных структур.

Литература

1. Demidenko O.M., Aksionova N.A. Development of a Machine Vision System for Image Recognition of Design Estimates // *Nonlinear Phenomena in Complex Systems*. 2022. Т. 25. №2. С. 159-167. <https://doi.org/10.33581/1561-4085-2022-25-2-159-167>
2. Demidenko O.M., Aksionova N.A., Varuyeu A.V., Kucharav A.I. 3D-modeling of Augmented Reality objects using Shi-Tomasi corner detection algorithms // *Journal of Physics: Conference Series*. IOP Publishing, 2021. Т. 2091. №1. С. 012058. <https://doi.org/10.1088/1742-6596/2091/1/012058>

© Аксёнова Н.А., Соболев Д.Н., Сыч Д.С., 2023

УДК 519.687

Алеников Г.С.

Нижевартовский государственный университет
г. Нижевартовск, Россия

СОЗДАНИЕ РОБОТА С РАЗНЫМИ РЕЖИМАМИ УПРАВЛЕНИЯ

Робототехника – это техническая наука, изучающая автоматизацию производственных и иных систем при помощи роботов. Она предполагает проектирование, создание и использование роботов для взаимодействия с внешней средой выполнения различных задач с частичным участием человека, или вовсе без него. Также робототехника – это одна из самых быстро развивающихся и актуальных сфер IT.

В данной статье будет представлен и детально описан четырёхколёсный робот, имеющий два режима управления – автономное и дистанционное. В автономном режиме он способен самостоятельно обходить препятствия на пути движения, а в дистанционном роботом возможно управлять с пульта.

Робот был создан на платформе Arduino. Arduino – торговая марка аппаратно-программных средств построения и прототипирования простых систем, моделей и экспериментов в области электроники, автоматики, автоматизации процессов и робототехники. Это платформа, предназначенная для «physical computing» с открытым программным кодом, построенная на простой печатной плате с современной одноимённой средой для написания программного обеспечения (<https://clck.ru/34pJLv>). Рассмотрим программную и аппаратную составляющую Arduino подробнее.

Программно Arduino представляет собой язык C++ с фреймворком Wiring. Написание кода также может происходить в собственной одноимённой IDE. Отличительной особенностью программ Arduino является обязательное наличие в коде двух основных функций: `setup()` и `loop()`. В первой функции на старте программы инициализируются нужные параметры, а во второй пишется основной код, который зацикливается, и по которому работает робот.

Аппаратно Arduino – это большой перечень плат со встроенными микроконтроллерами, а также платы расширения или shield-платы. Микроконтроллеры для Arduino отличаются наличием предварительно прошитого в них загрузчика. С помощью этого загрузчика пользователь загружает свою программу в микроконтроллер без использования традиционных отдельных аппаратных программаторов. Независимыми производителями также выпускается большая гамма всевозможных датчиков и исполнительных устройств, в той или иной степени совместимых с Arduino. Сейчас плат (как официальных, так и аналогов от сторонних производителей) в семействе Arduino больше десятка. Классические Arduino и Arduino-совместимые платы спроектированы для монтажа в стопки через штыревые разъёмы. Таким образом базовую микропроцессорную плату дополняют необходимой периферией и внешними подключениями.

За аппаратную основу робота была взята плата Iskra Neo компании Амперка – российский аналог Arduino Leonardo – одной из последних выпущенных плат в семействе

Arduino. В данной плате встроен микроконтроллер ATmega32U4 – 8-битный микроконтроллер семейства AVR. Он предоставляет в распоряжение 32 КБ флэш-памяти для хранения прошивки, 2.5 КБ оперативной памяти SRAM и 1 КБ энергонезависимой памяти EEPROM для хранения данных. Сама же плата изобилует большим количеством пинов для подключения различной периферии. Ниже представлена схема распиновки платы (рис. 1).

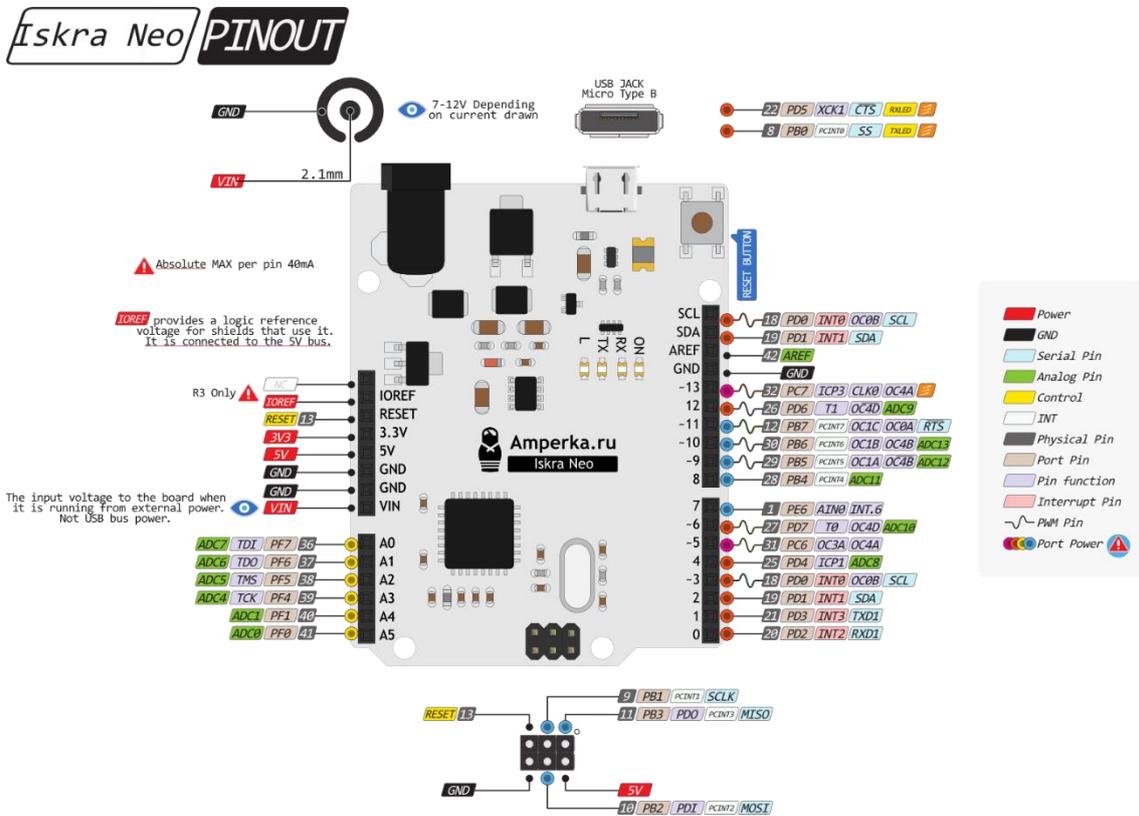


Рис. 1. Полная распиновка Iskra Neo (<https://clck.ru/JTmMx>)

Данная плата при создании робота была смонтирована на четырёхколёсную платформу производства DFRobot под названием Pirate. Платформа представляет собой металлическую основу-конструктор. Платформа Pirate включает только базовые элементы конструкции и 4 мотора для подключения к управляющей электронике.

Для подключения к Arduino четырёх моторов был использован специальный motor-shield той же компании «Амперка». Эта плата расширения насаживается сверху на основную плату, одновременно и занимая собой все пины и являясь их продолжением. Данная плата кроме подключения моторов также позволяет подключить необходимое для них внешнее питание.

Уже на данном этапе при загрузке в плату какого-либо алгоритма движения «робот» способен его исполнять. Далее будет расписан процесс автоматизации и принцип работы робота.

Автоматизация данного робота заключается в том, что он способен самостоятельно определять перед собой препятствия и обходить их. Одной из основных способностей, которые должен уметь делать автономный робот – это самостоятельно получать и обрабатывать информацию из окружающей среды. Для решения поставленной задачи в конструкции робота были использованы ультразвуковые и инфракрасные датчики.

Инфракрасный дальномер позволяет определять расстояние до объектов. Для робота была выбрана модель GP2Y0A021 компании Sharp. Сенсор определяет расстояние по отражённому лучу света в инфракрасном спектре. Дальномер может использоваться для объезда препятствий и ориентирования на местности (<https://clck.ru/34pJPB>). Выводом является аналоговый сигнал, с уровнем напряжения, зависимым от расстояния до цели в установленном направлении. Датчик подключается к управляющей электронике через 3 провода (земля, питание, выходной сигнал). Поскольку в основе работы устройства используется свет, сенсор плохо подходит для определения расстояния до светопоглощающих объектов. Дальномер не почувствует прозрачную поверхность, например, из пластика или оргстекла.

Ультразвуковой дальномер определяет расстояние до объектов точно так же, как это делают дельфины или летучие мыши. Он генерирует звуковые импульсы на частоте 40 кГц и слушает эхо. По времени распространения звуковой волны туда и обратно можно однозначно определить расстояние до объекта. В отличие от инфракрасных дальномеров, на показания ультразвукового дальномера не влияют засветки от солнца или цвет объекта. Даже прозрачная поверхность будет для него препятствием. Но могут возникнуть трудности с определением расстояния до пушистых или очень тонких предметов (<https://clck.ru/34pJPz>).

На робота датчики устанавливаются следующим образом: инфракрасные дальномеры – под углом в стороны на передней части робота так, чтобы в их поле зрения попадали препятствия по бокам от робота; ультразвуковой дальномер размещается посередине, чтобы фиксировать тонкие препятствия, не попадающие в поля зрения инфракрасных дальномеров. На рисунке 2 продемонстрировано размещение датчиков на роботе.



Рис. 2. Принцип установки датчиков и их области видимости

«Видя» разными датчиками препятствия, робот определяет, с какой оно находится стороны, и соответственно отворачивает в другую сторону. Ниже представлена таблица алгоритмов работы робота, то есть его реакции на различные условия среды.

Таблица

Таблица алгоритмов работы робота и реакции на условия среды

Препятствие	Датчики	Действие
Слева	1	Отвернуть вправо
Справа	2	Отвернуть влево
Посередине	3	Развернуться
Справа и слева	1 и 3	Развернуться

Такое расположение датчиков и алгоритм работы позволяют роботу «уворачиваться» практически от любого встречаемого препятствия и выходить из глухих углов.

В дальнейшем в код и конструкцию робота был добавлен простой переключатель, позволяющий переводить устройство в режим ручного управления. Данный режим реализован дистанционно через 4-канальный 433 МГц ресивер и трансмиттер (приемник и передатчик). Приёмник подключается к плате Arduino и монтируется на робота, а с передатчика в руках оператора подаются управляющие сигналы.

Для использования в роботе был выбран 4-канальный 433МГц передатчик от китайского производителя в виде готового решения четырёхкнопочного пульта. На сигналы от этих 4-х кнопок были запрограммированы простые команды – ехать вперёд, ехать назад, поворачивать направо, поворачивать налево.

На рисунке 3 продемонстрирована схема подключения электроники. Как уже упоминалось, motor-shield насаживается на основную плату Arduino, полностью её перекрывая, но для наглядности схемы подключения всех элементов идут к плате напрямую.

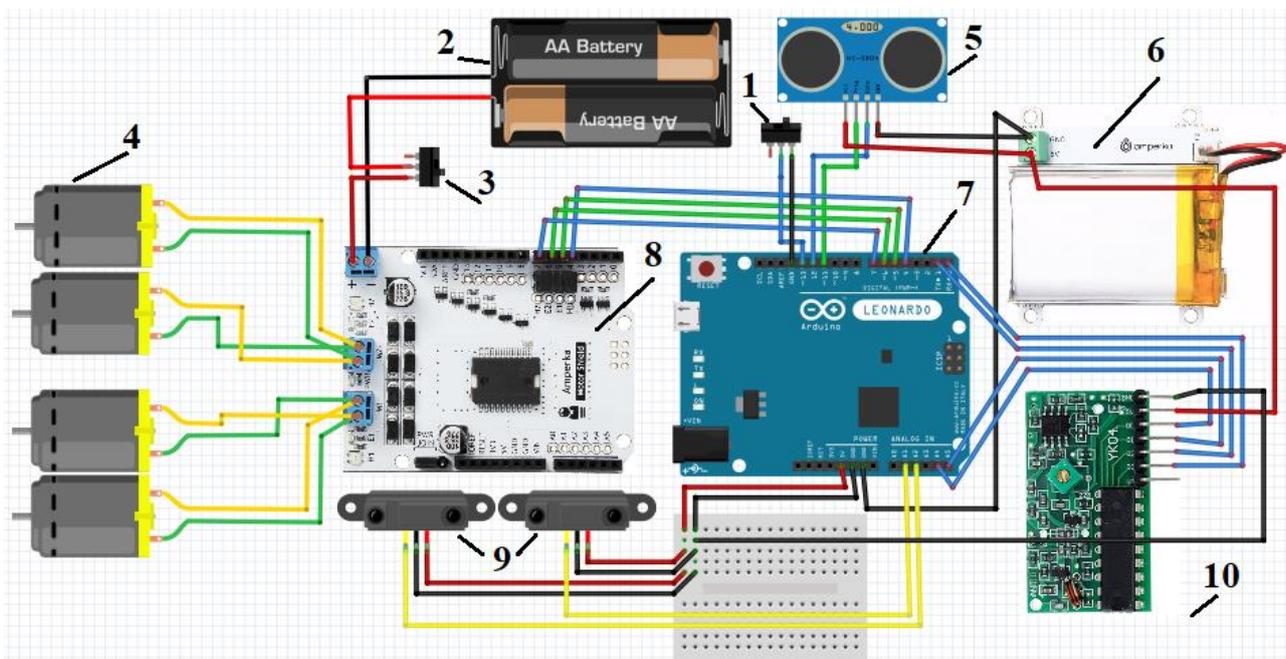


Рис. 3. Схема подключения электроники: 1 – переключатель режима радиоуправление/автономная работа; 2 – батарейный блок моторов; 3 – переключатель включения моторов; 4 – моторы; 5 – ультразвуковой датчик; 6 – power bank, питающий ультразвуковой датчик и приёмник; 7 – плата Arduino leonardo; 8 – motor shield; 9 – инфракрасные датчики; 10 – 433 Mhz приёмник сигнала

В процессе разработки и конструирования робота был выявлен ряд проблем, которые можно изложить в этой статье подробнее, а также рассказать о способах их решения.

Проблема «барахлящих» датчиков. По мере разработки робота и добавления всё новых и новых элементов электроники в конструкцию, «поломался» ультразвуковой датчик. Он начал выдавать неправильные значения, замена датчика не исправляла проблему, а в отдельном тестировании на пустой плате он показывал, что работает исправно.

Причиной данной проблемы оказалась недостаточность питания. Сама плата Arduino способна питать электронику с напряжением в 5 V, но в итоге плата перестала справляться с питанием датчиков, когда их стало три (два инфракрасных и один ультразвуковой). Решением стало добавление power-банка для отдельного питания этого датчика. При проектировании важно уделять особое внимание достаточному питанию различных элементов электроники.

Вторая проблема заключалась в 433 МГц передатчике и приёмнике. Их дальность работы оказалась равна буквально метру-полтора.

Причиной этой проблемы являются эти элементы сами по себе. Дешевые радиодетали, что были выбраны для реализации радиоуправления, оказались просто-напросто неспособны выдавать сильный и стабильный сигнал на длинное расстояние. Решить данную проблему можно заменой данных деталей на более дорогие, качественные и мощные образцы, или же полностью изменить этот тип радиоуправления на работу через bluetooth с подключением к предварительно разработанному мобильному приложению для управления роботом. Выбор электроники – не менее важный этап разработки, чем само конструирование и программирование готового продукта. При конструировании рекомендуется предварительно тестировать электронику в отдельности, чтобы точно осознавать её характеристики и особенности работы.

Робототехника – особая часть ИТ, в равной степени сочетающая в себе высокие требования как к навыкам написания кода, так и к пониманию техники и электронных устройств. Даже разработка примитивнейшего автоматизированного/радиоуправляемого робота показала безграничный потенциал и глубину данной области.

© Алеников Г.С., 2023

УДК 004.04

Браженец М.И.

Низневартовский государственный университет
г. Нижневартовск, Россия

АНАЛИЗ И СРАВНЕНИЕ СРЕДСТВ РАЗРАБОТКИ WEB-ПРИЛОЖЕНИЯ ЭЛЕКТРОННОГО ЖУРНАЛА

Создание единого информационного пространства является необходимым условием эффективной организации деятельности любого предприятия или организации [2]. Данная работа посвящена проведению анализ и сравнение средств разработки web-приложения электронного журнала.

В условиях информатизации устаревает так называемый «бумажный контроль». Его главная проблема – это невозможность учащегося посмотреть свои успехи и самостоятельно оценить свои возможности в освоении учебной программы [1]. Для сбора информации и составления статистики по успеваемости придётся обработать и сортировать огромное количество бумажной документации, которая отнимает невероятное количество времени.

Отсутствие электронного журнала учёта посещаемости и успеваемости в университете является актуальной проблемой. Для создания приложения необходимо выбрать средства разработки, которые будут влиять на время его создания и качество конечного продукта. Под качеством web-приложения будем подразумевать следующее: скорость обучения работы в приложении, подверженность различным ошибкам и сбоям, эргономичность интерфейса, что определяет заданные требования к средствам разработки, необходимым для реализации web-ресурса. Кроме того, приложение должно соответствовать требованиям, предъявляемым журналам учёта.

Для сравнения были выбраны две существующих автоматизированных системы учёта «ЭлЖур» и «Дневник.ру». «Дневник.ру» – социальная сеть с возможностью ведения Электронного журнала и Электронного дневника. «ЭлЖур» – система для автоматизированного учета и анализа успеваемости и посещаемости (с построением различных отчетов и графиков), а также автоматизации различных школьных процессов.

При проведении обзора был составлен перечень требований:

1. Разбиение групп по подгруппам даёт возможность разделять большие классы. Аналогично происходит и с группами с большим количеством учеников в университетах.
2. Объединение групп для поточного обучения. Актуально для потоковых лекций на факультетах.
3. Настраиваемая система оценивания поможет скорректировать систему оценок индивидуально под требования каждого преподавателя.
4. Формирование выборки по работам позволит оценить успехи и трудности по определённым видам работ.
5. Настройка прав редактирования по ролям обеспечит или ограничит функционал в зависимости от полномочий.
6. Отметки посещаемости нужны для статистики посещаемости и отчётности.

7. Полное описание задания, даты размещения/сдачи и сопоставление с темами лекций поможет студентам, которые по какой-либо причине пропустили занятия или хотят достать недостающие работы.

8. Проверка заданий и выставление оценок дают возможность студентам оценить свой уровень подготовки по предмету.

9. Автоматическое формирование отчёта по успеваемости и посещаемости сокращает время, затраченное на составление, и обеспечивает доступ к документации в любое время.

10. Возможность экспорта отчётов в форматах PDF бумажной версии стандартного вида и экспорт в Excel. Формирует документацию для организаций без лишних временных затрат и исключает механические ошибки при заполнении.

11. Составление автоматического индивидуального рейтинга для отслеживания успеваемости. Полезная функция для студентов, помогает самоорганизовываться.

Поскольку бизнес за последние годы стремительно переходит в онлайн-формат веб-приложения становятся все более популярными. Приложения можно разделить на кастомные (написанные кодом) и ноукод (собранные в конструкторах).

Для написания кастомных приложений разработчики используют IDE и текстовые редакторы. Интегрированная среда разработки (IDE) – это программное приложение, которое помогает программистам эффективно разрабатывать программный код. Оно повышает производительность разработчиков, объединяя такие возможности, как редактирование, создание, тестирование и упаковка программного обеспечения в простом для использования приложении (<https://clck.ru/VCGyL>). Текстовый редактор – самостоятельная компьютерная программа или компонент программного комплекса (например, редактор исходного кода интегрированной среды разработки или окно ввода в браузере), предназначенная для создания и изменения текстовых данных. Текстовые редакторы предназначены для работы с текстовыми файлами в интерактивном режиме.

В таблице 1 приведены сравнения характеристик интегрированных сред разработки.

Таблица 1

Таблица сравнения IDE

	Visual Studio Code	WebStorm	Geany
Автодополнение	Поддерживает	Поддерживает	Поддерживает
Поддержка фреймворков	Поддерживает	JavaScript-фреймворки	Поддерживает
Сниппеты	Поддерживает	Поддерживает	Поддерживает
Интерактивный просмотр	Поддерживает	Поддерживает	Недоступно
Плагины	Поддерживает	Поддерживает	Поддерживает
Фолдинг	Поддерживает	Поддерживает	Недоступно
Рефакторинг	Недоступно	Поддерживает	Недоступно
Отладка	Поддерживает	Поддерживает	Поддерживает (требуется установка GeanyGDB)
Поддерживаемые языки	JavaScript, TypeScript, CSS и HTML	PHP, HTML и JavaScript	64 языка программирования
Операционные системы	Кроссплатформенный	Кроссплатформенный	Кроссплатформенный
Макросы	Поддерживает	Поддерживает	Поддерживает

Автосохранение	Поддерживает	Поддерживает	Недоступно
Поддержка баз данных	Поддерживает	Поддерживает	Поддерживает
Быстрый просмотр документации	Поддерживает	Поддерживает	Недоступно

Visual Studio имеет огромную коллекцию всевозможных расширений, которая постоянно пополняется. Из недостатков можно выделить тяжеловесность этой IDE. Для выполнения даже небольших правок могут потребоваться значительные ресурсы, поэтому если нужно выполнить какую-то простую и быструю задачу, удобнее использовать более легкий редактор (<https://clck.ru/Vi8uH>).

Главное удобство WebStorm в том, что рабочая среда уже полностью настроена – все, что вам остается просто открыть нужный проект. Эта IDE пользуется популярностью среди разработчиков благодаря хорошей поддержке языков (в том числе языков бэкенда), отличному встроенному функционалу и простоте использования (<https://clck.ru/34pKJi>).

Одним из самых приятных моментов в Geany является поддержка большого количества типов файлов. Для своей работы программа не требует множества зависимостей. Интерфейс простой и удобный (<http://geany.ru>). Однако, невозможно создать проект на основе шаблона. Более того, было бы большим преимуществом иметь некую систему шаблонов, чтобы можно было создавать проект не с пустого места, а с какой-то начальной структуры.

Самой оптимальной среди интегрированных сред разработки является WebStorm. Обосную свой выбор тем, что в данной IDE есть уже готовый интерфейс, который работает без необходимости устанавливания расширений, рефакторинг работает надёжно, хорошо работает даже с большими базами. Также есть встроенный отладчик, эмулятор терминала, интеграция с инструментами тестирования – это означает, что большинство задач можно выполнить в WebStorm, не прибегая к сторонним ресурсам.

Если перед нами стоит задача создать ресурс с простой структурой, то конструкторы сайтов — это то, что нам нужно. Конструкторы сайтов – это сервисы, в которых можно собрать сайт без знаний вёрстки и дизайна. Конструкторы подходят, чтобы протестировать идею, сделать блог, портфолио, персональный сайт или небольшой интернет-магазин. Они не требуют серьёзных вложений времени и денег, а сайты получаются красивыми и удобными. Рассмотрим два наиболее популярных конструктора Tilda и WebFlow.

В таблице 2 представлены характеристики конструкторов сайтов.

Таблица 2

Таблица сравнения конструкторов сайта

	Tilda	Webflow
Экспорт кода	HTML, CSS, JS	HTML/JS
Уровень освоения	легкий	тяжелый
Каталог готовых шаблонов	Более 210, загружать внешние нельзя	Более 100 шаблонов, 40 бесплатных
Адаптация под разные устройства	Поддерживает	Поддерживает
Управление базами данных	Поддерживает	Недоступно
SEO-инструменты	Поддерживает	Поддерживает

Интеграции	45 интеграций, среди которых Битрикс 1-С, WordPress или через API	Google Домены MailChimp, API, Zapier
Доступные языки	Русский, English	English
Командная работа	Поддерживает	Поддерживает
Безлимитные бэкапы	Поддерживает	Поддерживает
CMS-функционал	Zero Block	Поддерживает
Безопасность	Сертификат LetsEncrypt	Встроенный SSL-сертификат
Сеть доставки контента CDN	Поддерживает	Поддерживает
Бесплатный хостинг	Поддерживает	Недоступно
Загрузка своих шрифтов	Поддерживает	Недоступно

Положительные стороны конструктора Tilda:

1. Возможность создать уникальный дизайн. Наличие Zero Block – редактора позволит создать любой блок или элемент на странице с нуля.
2. Легкий и доступный интерфейс.
3. Интеграция с другими популярными сервисами. Возможность интеграции с чатами и мессенджерами, платежными системами и рассылками.
4. Экспорт статичного кода html и все необходимые файлы (изображения, css, js).
5. Для других пользователей Тильды можно предоставить полный или ограниченный доступ к редактированию вашего проекта.
6. Автоматическая адаптация сайта под любой тип устройства.
7. Изображения хранятся в сети доставки (и дистрибуции) контента (CDN), что сокращает время загрузки за счет использования географически распределенных серверов.
8. Наличие бесплатного сертификат Let's Encrypt для безопасности.
9. Добавление сайт в поисковые системы Яндекс и Google для нахождения ошибок, влияющих на индексацию.
10. Управление базой данных клиентов;
11. Интерфейс и документация на русском языке.

Из отрицательных сторон можно выделить не полный спектр возможностей для создания многостраничного сайта (ограничение до 500 страниц), однотипность шаблонов, также сервис не имеет возможности собирать, обрабатывать и обмениваться большим количеством данных. Больше подойдет для низконагруженных сайтов (<https://clck.ru/34pKFv>).

Положительные стороны конструктора Webflow:

Возможность экспорта кода и работы в команде; встроенный SSL-сертификат включен для обеспечения соответствия текущим требованиям онлайн-безопасности без дополнительных затрат, а интегрированная сеть доставки контента (CDN) обеспечивает скорость и безопасность вашего веб-сайта; настраиваемый CMS-функционал; примеры сайтов в сообществе; бесплатные безлимитные бэкапы; генерация SEO-заголовков (<https://clck.ru/34pKBs>).

Существующие ограничения:

1. Много профессиональных платных шаблонов.
2. Нельзя обратно загрузить экспортированный код, вставка реализована только на уровне отдельных блоков.

3. Документация и интерфейс на английском языке.

4. Слабо адаптируется для российского рынка, так как имеющиеся интеграции и поддерживаемые платёжные решения не совместимы с популярными в РФ.

5. Сложная настройка мобильной версии.

6. В бесплатной версии нельзя запустить рабочий сайт.

7. Отсутствует управление базами данных.

8. Нельзя загружать свои шрифты.

9. Тяжелый интерфейс для обучения.

Наиболее оптимальным конструктором является Tilda, в нем множество готовых блоков и шаблонов сайтов, которые гораздо проще переделать под себя, чем начинать всё с нуля.

Анализ средств разработки позволил сделать вывод, что оптимальной средой разработки интерфейса будущего web-приложения является конструктор сайтов Tilda, а интегрированная среда разработки WebStorm пригодится для написания серверной части.

В заключении необходимо отметить, что для разработки полноценного продукта, в котором будет реализованы все необходимые функции, необходимо использовать несколько средств разработки в совокупности.

Литература

1. Тараканова А.П. Современные проблемы контроля успеваемости и посещаемости студентов ВУЗа // Горный информационно-аналитический бюллетень (научно-технический журнал). 2014. №3. С. 396-398.

2. Plaksina I.G., Kirsanova A.A., Topolsky D.V., Topolskiy N.D., Yumagulov N.I., Zvereva E.A. Automation of Project Publishing Process in Corporate Information System // Global Smart Industry Conference (GloSIC). IEEE. 2018. С. 1-7. <https://doi.org/10.1109/GloSIC.2018.8570069>

© Браженец М.И., 2023

БАЗА ЗНАНИЙ НА ОСНОВЕ ОНТОЛОГИИ ДЛЯ ОБУЧЕНИЯ ТЕХНИЧЕСКОГО ПЕРСОНАЛА ПРИЕМНОЙ КОМИССИИ

Современное общество характеризуется постоянным развитием и нововведениями и это сказывается на повышении требований работодателя к компетенциям своих сотрудников, поэтому вопрос об обучении своего персонала на данный момент является актуальным. Внедрение новых либо неизвестных в данном конкретном коллективе информационных технологий чаще всего ознаменовано необходимостью к обучению использования их в рабочем процессе [3, с. 387]. Многие организации начинают использовать для самообучения базы знаний, которые помогают решать такие проблемы, как расширение или замена персонала, утрата квалификации сотрудника

Основная цель любой базы знаний – сокращение времени на поиск информации и получение ответа на типовой вопрос в любой момент. Поскольку полноценные базы знаний содержат в себе не только фактическую информацию, но и правила вывода, позволяющие делать автоматические умозаключения об уже имеющихся или вновь вводимых фактах и тем самым производить семантическую (осмысленную) обработку информации [2, с. 95].

На данный момент разработаны различные модели представления знаний практически для любой предметной области. К ним относятся: продукционные модели; семантические сети; фреймы; логические модели; нейронные сети [1, с. 33].

Следует отметить, что в последнее время наибольшую популярность для представления знаний приобретает – онтология, которая, по мнению многих ученых, способна решить все вышеперечисленные проблемы. Несмотря на то, что в 1980-х годах сообщество искусственного интеллекта начало использовать термин «*онтология*» для обозначения, как теории моделируемого мира, так и компонента систем, основанных на знаниях. взаимодействия систем баз знаний друг с другом, онтологией можно считать интеллектуальные средства поиска, представления и обработки знаний и запросов, Онтология (информатика) (<https://clck.ru/34pcU8>).

Широкое практическое применение онтологий, прежде всего, обусловлено таким преимуществом, как эффективное описание семантики данных для выбранной предметной области, которое позволяет при обращении к поисковой системе получать в ответ ресурсы, семантически релевантные запросу.

Приемная комиссия – это подразделение высшего или среднего учебного заведения, осуществляющее приемную кампанию на будущий учебный год. Главная цель создания приемной комиссии – организация приемной кампании: прием документов от лиц, поступающих в вуз и обеспечение зачисления в образовательное учреждение, знакомит абитуриентов с правилами и распорядком учебного учреждения, документацией по аккредитации направлений и специальностей, критериями поступления. Также проводит

консультации по вопросам ведения обучения и, самое главное, выносит решение о поступлении или не поступлении абитуриента в данное учебное заведение.

В настоящее время приемная комиссия является структурированной и состоит из определенных единиц, выполняющих установленные обязанности.

Анализируя состав приемной комиссии различных вузов, установили, что в основном она состоит из председателя приемной комиссии; заместителя председателя комиссии; заведующих кафедрами; председателей предметных экзаменационных комиссий; секретариата; технического персонала.

Технический персонал принимает документы, консультирует по составу пакета, необходимого при поступлении; производит запись данных об абитуриенте в регистрационный журнал; оформляет документы личных дел абитуриентов; отвечает на вопросы поступающих в форме звонков, электронных писем, вопросов в устной форме абитуриентов, пришедших за консультацией по вопросам, связанным с организацией учебной деятельности, зачисления, Приемная комиссия.

В настоящее время наблюдается тенденция на увеличение объема знаний и соответственно требования руководства к уровню подготовки персонала, таким образом, благодаря использованию обучающей онтологии, возможно, решить данную проблему и адаптироваться к данным изменениям.

Результаты данной работы, прежде всего, предназначены для технического персонала, поскольку от правильности и полноты заполненной ими информации (уровень предыдущего образования, перечень предметов, необходимых для поступления) зависят рейтинговые списки абитуриентов, приказы о зачислении. Обучающая онтология «Абитуриент» позволит обеспечить лёгкость и быстроту обучения технического персонала правилам приема в вуз.

Алгоритм действия технического персонала при обучении заключается в следующем: воображаемый абитуриент предоставляет информацию о документе, об образовании, имеющихся результатах экзаменов, а технический персонал вносит все эти данные в соответствующие вкладки в редакторе онтологии. Для разработки обучающей онтологии использовали редактор Protege 4.3. После того, как внесена вся необходимая информация, включается в работу машина вывода Reasoner для проверки согласованности и автоматического извлечения знания о возможности «быть рекомендованным к зачислению» (согласно документу об образовании и имеющимся результатам экзаменов), либо же абитуриент будет отнесен к подклассу «Не прошел контроль».

На рисунке 1 представлен фрагмент обучающей онтологии. На данном этапе проектирования определили базовый класс «Абитуриент» и его подклассы – специальности (уровень бакалавриата/специалитета: гуманитарные, технические, экономические; и «поступающий в магистратуру»), сформулированы свойства объектов (topObjectProperty) и созданы несколько примеров экземпляров. Экземпляры взаимодействуют друг с другом через свойства объектов или Object Property. В разрабатываемой онтологии свойства экземпляров задаются с помощью раздела Data Properties.

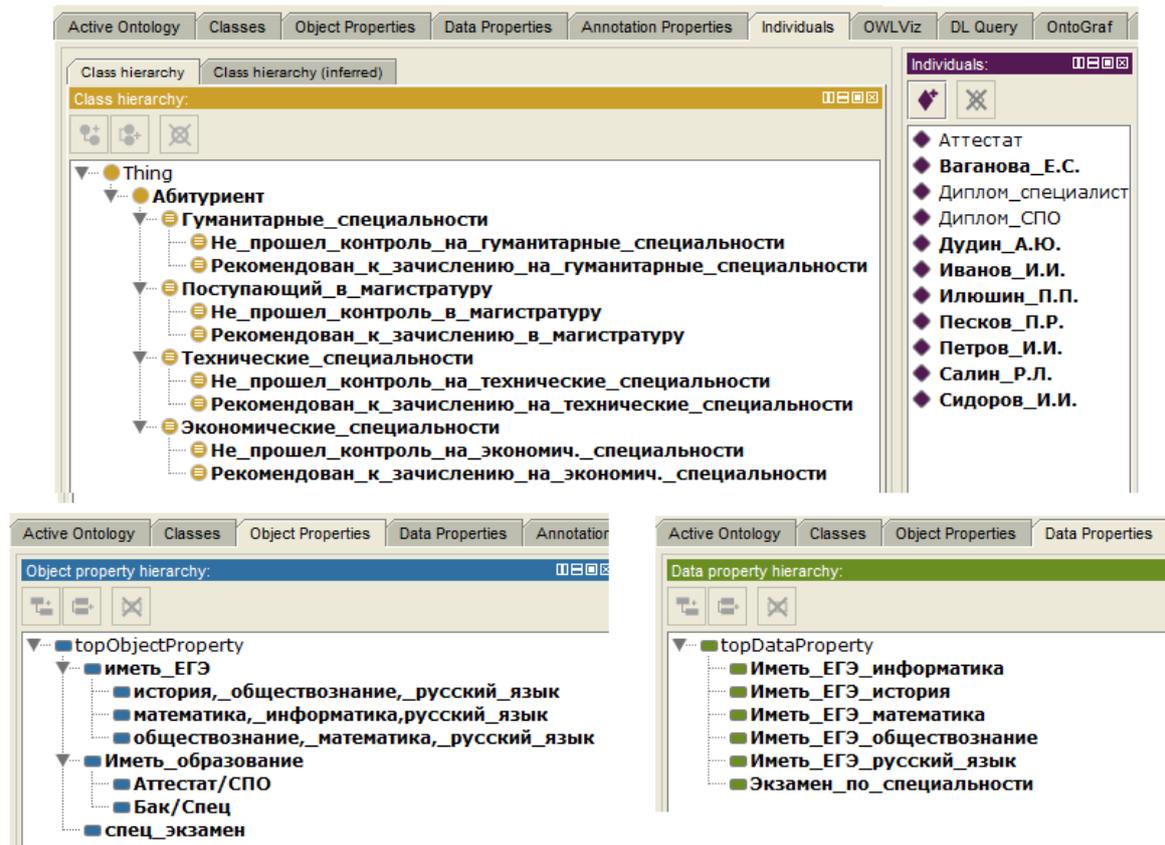


Рис. 1. Фрагмент учебной онтологии «Абитуриент»

На рисунке 2 представлен фрагмент онтологии, а именно - подкласс «гуманитарные специальности» с его характеристиками и ограничениями, представленными в разделе «Description». Указаны условия (необходимо иметь результаты экзаменов по истории, обществознанию и русскому языку) при которых абитуриент может рассматриваться в качестве индивида данного подкласса (в нашем случае это Иванов И.И., Илюшин П.П.), желающих поступать на данные направления и ряд ограничений (не пересечений с подклассами).

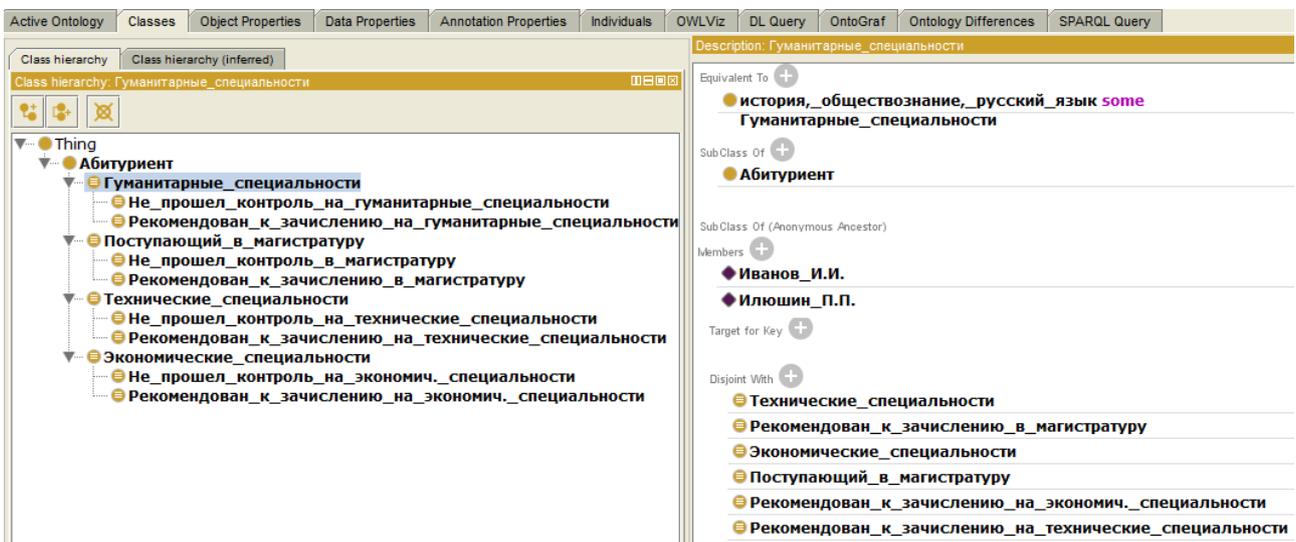


Рис. 2. Подкласс «Гуманитарные специальности» с его характеристиками и ограничениями

На рисунках 3 и 4 представлены результаты вывода Reasoner о том, какой абитуриент относится к подклассу «Не прошел контроль» (рис. 3), и кто может быть «Рекомендован к зачислению» (рис. 4). Так мы видим, что абитуриент Илюшин П.П. отнесен к подклассу «Не прошел контроль на гуманитарные специальности» по причине того, что не преодолел минимальные баллы по необходимым предметам, а именно по истории – менее 36 баллов. С абитуриентом Ивановым И.И. (рис. 4) наблюдается обратная ситуация, в силу того, что его результаты ЕГЭ превышают пороговые значения, то результатом работы Reasoner является отнесение данного абитуриента к подклассу «Рекомендован к зачислению на гуманитарные специальности».

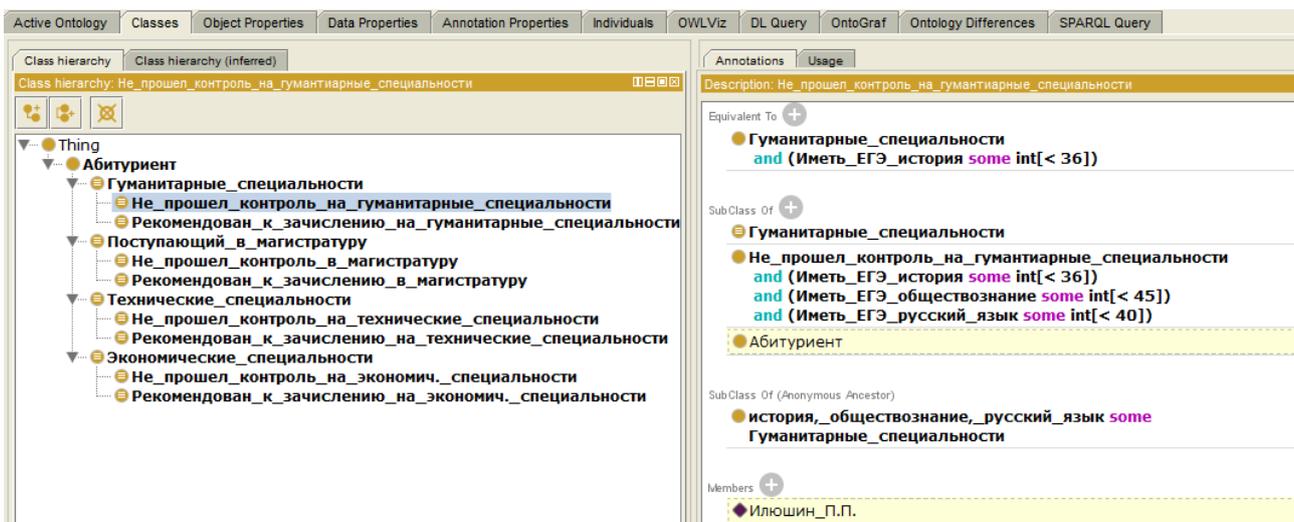


Рис. 3. Вывод Reasoner об абитуриенте, который «Не прошел контроль на гуманитарные специальности»

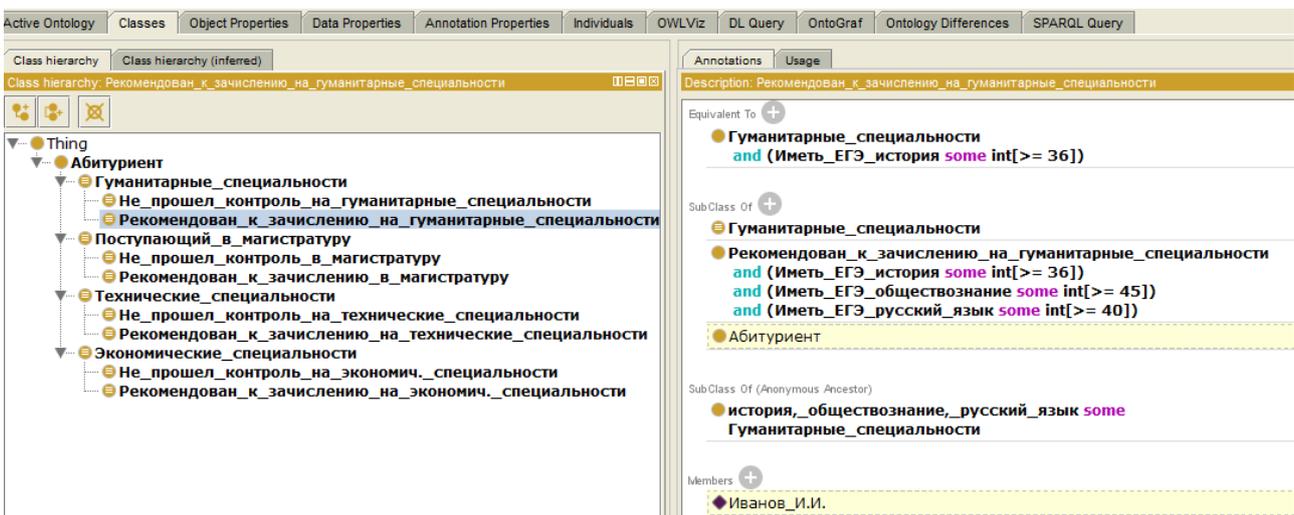


Рис. 4. Вывод Reasoner об абитуриенте, который «Рекомендован к зачислению на гуманитарные специальности»

Следует отметить, что онтологии относят к рекомендательным экспертным системам, поскольку могут использоваться в базах знаний для обнаружения новых фактов, выявления взаимосвязей между элементами, Приемная комиссия. Таким образом, содержание

разрабатываемой обучающей онтологии предназначено для того, чтобы технический персонал мог использовать автоматизированную систему для рекомендации подачи документов для поступления на ту или иную специальность в вузе.

Далее, в рамках данной работы, предстоит расширять онтологическую базу знаний введением дополнительных понятий и отношений (сроки получения документа об образовании, детализация специальностей), проверять онтологию на противоречивость при получении списков рекомендованных абитуриентов к зачислению и внедрять её в процесс обучения технического персонала приемной комиссии.

Литература

1. Верхотурова Ю.С. Онтология как модель представления знаний // Вестник Бурятского государственного университета. 2012. № 5. С. 32-37.
2. Гатаулин Т.М., Малыхин В.И., Гончаров Л.Л. Знания их количество операция над ними // Вестник Университета. 2015. №11. С. 94-99.
3. Джафаров А.И., Гахова Н.Н. Разработка информационного обеспечения для интеграции статистических данных на примере переписи населения // Теория и практика современной науки. 2022. № 6(84). С. 387-393.

© Ваганова Е.С., Горбачев И.В., 2023

УДК 004.45

Гребеникова С.А., Сметанникова Т.А.

Липецкий государственный технический университет
г. Липецк, Россия

ИСПОЛЬЗОВАНИЕ ПРОГРАММ ТРЕХМЕРНОГО МОДЕЛИРОВАНИЯ В СОЗДАНИИ ЦИФРОВОЙ СКУЛЬПТУРЫ

Рэймонд Курцвейл в своем эссе «Закон ускоренной отдачи» писал: «Анализ истории технологий показывает, что технологические изменения экспоненциальны, в отличие от здравого смысла «интуитивно-линейного» взгляда. Таким образом, мы не переживем 100 лет прогресса в 21 веке – это будет больше похоже на 20 000 лет прогресса по сегодняшнему курсу.» [2, с. 1]. Думаю, все образованные люди современности согласятся с ним. Прогресс движется вперед - музыка, искусство, архитектура давно переходят в цифровой формат и крепко утверждаются в нем. В наши дни для создания книги не нужны кипы бумаги, для создания картины не нужны холсты и кисти, а для создания скульптуры не всегда нужна глина или мрамор. 3D-модель, созданная профессиональным дизайнером, во много раз превосходит в точности построения модель, вымеренную и начерченную на бумаге с помощью линейки и карандаша. Помимо этого, трехмерную компьютерную модель можно будет рассмотреть во всех нужных ракурсах в объеме, цвете и подходящей текстуре, внося при необходимости изменения и убирая недостатки, если таковые имеются [1, с. 4].

Цифровые художники получают мировую известность, продают свои работы за тысячи долларов, при этом их затраты составляет только плата за обучение и подписку на официальные пакеты программ. В мире непрерывно растущего качества изображений и моделей незаменимыми становятся «цифровые скульпторы», специалисты, способные создать точно проработанные модели предметов, персонажей, зданий, целых миров. И разумеется, ведущие компании-производители операционных систем и программного обеспечения стремятся предложить рынку самые качественные платформы для создания таких цифровых миров. В данной статье я рассматриваю две популярные программы для скульптинга, оцениваю их возможности и привожу примеры их профессионального использования.

Прототипы программ для 3D моделирования появились еще в середине прошлого века. Это были простейшие редакторы, заложившие основу для создания пользовательского графического интерфейса. Первой такой программой была Sketchpad, продукт докторской диссертации Айвена Сазерленда. Работа в Sketchpad происходила при помощи манипуляций светового пера на X-Y координатном CRT плоттер дисплее. Sketchpad считается предком современных САПР программ и крупным прорывом в компьютерной графике в целом (<https://www.cam.ac.uk>).

Подобные программы продолжали развиваться и совершенствоваться, но их возможности были ограничены, слабые процессоры компьютеров не позволяли создавать достаточно совершенные модели для кино и игровой индустрии. Тем не менее, в то время уже создавались первые программы 3D-моделирования, первые алгоритмы трассировки лучей для рендеринга трехмерной сцены и активно развивался полигональный метод моделирования

трехмерных объектов, который сейчас является основным. (<https://clck.ru/33aXUr>) Первая масштабная массовая работа, выполненная с помощью 3D графики – это мультфильм «История игрушек» от студии Pixar совместно с компанией Walt Disney. Первый полнометражный анимационный фильм, полностью смоделированный на компьютере, стал огромным прорывом в киноиндустрии (<https://clck.ru/34pci6>).

Существует два основных вида 3D-моделирования: полигональное и параметрическое. Работа над объектом с помощью полигонального моделирования заключается в манипуляциях над поверхностью в пространстве. Такая поверхность разделена на секции – полигоны, которые можно изменять и перемещать. Линии, образующие секции, называются рёбрами, а точки их пересечения – вершинами. С помощью предоставляемых программой функций художник смещает полигоны относительно друг друга, работает с их вершинами, увеличивает или уменьшает их количество. Количество полигонов в профессиональных программах может быть огромным. Чем оно больше – тем детальнее становится моделируемый объект, что приближает художника к желаемому результату.

При параметрическом моделировании происходит создание простой модели и дальнейшая работа с ней посредством усложнения и изменения. За основу берётся математическая модель с нужными параметрами, изменяя которые можно создавать различные фигуры. Параметрическое 3D-моделирование появилось первым по сравнению с полигональным и явилось улучшением традиционной инженерной графики, оно упрощает понимание чертежей и обеспечивает зрительное восприятие проектируемых деталей и механизмов. Так как в данной статье рассматривается высокополигональная цифровая скульптура, тема параметрического технического моделирования подробно раскрываться не будет.

Первой программой для рассмотрения я выбрала ZBrush. Это графическое приложение, призванное упростить создание компьютерной графики. ZBrush привлекает широкую аудиторию, предоставляя мощные инструменты, которые дают пользователям интуитивно понятный и визуальный контроль над создаваемыми ими цифровыми изображениями (<https://clck.ru/34pchn>). ZBrush является одной из популярных программ для цифровой скульптуры, сочетающий в себе 3D и 2.5D моделирование, текстурирование и рисование. Отличительной особенностью этого софта является имитация процесса «лепки» трёхмерной скульптуры. Функция рендеринга в реальном времени позволяет увеличить наглядность и заметно упростить создание трехмерного объекта. ПО позволяет не только моделировать трехмерный объект, но и накладывать на него изображения, текстуры, шумы.

Интерфейс ZBrush достаточно дружелюбен, хоть и может сначала показаться сложным, но такое впечатление производят почти все профессиональные графические редакторы. Рассмотрим основные элементы интерфейса. В центре экрана стандартно расположена область документа в которой отображается результат работы. Слева и справа расположены вместительные Palett (палитра). По периметру рабочей области расположены так называемые «полки» инструментов: кисти, настройка цвета, основные действия. Справа сверху находится

строка быстрых функций для манипуляций с окнами, строками, готовыми скриптами и изменения палитр.

Функции программы достаточно обширны, можно рассмотреть основные для любой работы над сложными объемными объектами. Программа предлагает несколько десятков стандартных кистей, с помощью которых можно производить первоначальные универсальные манипуляции. Пользователь имеет возможность создавать кисти самостоятельно или загружать уже готовые из интернета, что позволяет каждому художнику подстроить цифровые «художественные принадлежности» под себя. У каждой кисти есть размер, интенсивность, жесткость, можно настроить текстуру или узор. В программе можно создавать уникальные 2D-рисунки, чтобы затем наложить их на модель, или же рисовать непосредственно на поверхности моделей, что позволяет прорабатывать не только детали объемов, но и принты на одежде, потертости, родинки или пигментные пятна на теле персонажа. Как и все программы для скульптинга ZBrush дает возможность текстурирования поверхности модели путем накладывания на нее рельефов кожи, ткани, дерева и т. д. Программа предоставляет удобную функцию «transpose», которая позволяет изменить положение какой-либо части модели без риггинга, то есть без необходимости простраивать скелет для механики модели. Инструмент «fibermesh» позволяет создавать типовые волокна и быстро «выращивать» их на модели без необходимости моделировать все их по отдельности. С помощью этого инструмента можно в несколько раз ускорить процесс создания прядей волос, травы, шипов и т. п.

Однако, у программы есть и минусы, о которых не всегда говорится. Во-первых, это платная лицензия, которая может стать проблемой для многих пользователей. В декабре 2021 года компания Maxon купила Pixologic, создателя ZBrush, и сменила политику использования. Обладатели бессрочной подписки не лишились своего приобретения, но бесплатно получать актуальные обновления в реальном времени они больше не могут, что приводит к частым ошибкам (<https://clck.ru/34pcgE>). Возможность работы с «цифровой глиной» также может являться минусом. Работа с примитивами может облегчить процесс «лепки», но с проработкой мелких анатомических особенностей возникнут проблемы. Создание персонажей с нуля требует знания анатомии и физики, без этого получить реалистичного персонажа будет сложно.

Существуют тысячи примеров удивительных работ, выполненных в программе ZBrush. Художник из Великобритании Damien Guimoneau создает кропотливо проработанные модели животных и антропоморфных существ. В числе его работ модели из нашумевшего фильма «Король лев» 2019 года студии Disney. Любой может ознакомиться с его работами на сайте ArtStation и на официальных сайтах студий (<https://clck.ru/34pcfy>).

Когда вы используете ZBrush для создания профессиональных иллюстраций или анимации, в итоге вам будет необходимо показать или передать вашу работу другим. ZBrush делает это возможным благодаря мощной системе рендеринга под названием Best Preview Render. Рендер BPR обеспечивает рендеринг модели в режиме редактирования с использованием высококачественного сглаживания при полном размере документа. BPR

предоставляет несколько новых опций рендеринга, включая трассировку лучей, текстуру, рассеивание света и эффекты волокон.



Рис. 1. Рафики, Шрам и Симба из фильма «Король лев»

Второй из обозреваемых программ для создания объемных персонажей и сложных предметов стала Blender. Это профессиональное свободное и открытое программное обеспечение для создания трёхмерной компьютерной графики, включающее в себя средства моделирования, скульптинга, анимации, симуляции, рендеринга, постобработки и монтажа видео со звуком, компоновки с помощью «узлов», а также создания 2D-анимаций. В настоящее время Blender пользуется большой популярностью среди бесплатных 3D-редакторов в связи с его быстрым стабильным развитием и технической поддержкой.

Blender с момента выпуска программы считался сложным для освоения. Каждая функция могла вызываться сочетанием горячих клавиш, при том что функционал программы был невероятно велик. Некоторые функции могли вызываться исключительно с клавиатуры, такое количество сочетаний клавиш было невозможно запомнить. Когда Blender стал проектом с открытым исходным кодом, были добавлены полные контекстные меню ко всем функциям, а использование инструментов сделано более логичным и гибким. С последующим улучшением пользовательского интерфейса были введены цветовые схемы, прозрачные плавающие элементы, а также новая система просмотра дерева объектов и другие различные мелкие изменения.

Интерфейс пользователя обладает как общими функциями всех программ для скульптинга, так и особенными командами и окнами. Режимы редактирования включают в себя два основных режима: Объектный режим (Object mode) и Режим редактирования (Edit mode), которые переключаются клавишей Tab. Объектный режим удобнее использовать для прямых визуальных манипуляций с отдельными объектами, в то время как режим редактирования – для работы с числовыми параметрами объекта. К примеру, для работы над

моделью в объектном режиме мы можем перемещать, изменять размер и вращать модель целиком или отдельные ее части, а режим редактирования используется для манипуляций над отдельными вершинами конкретной модели, как при параметрическом моделировании. Графический интерфейс Blender состоит из нескольких окон, которые в свою очередь могут быть разделены на разделы и индивидуально настроены. Иконки каждой секции могут настраиваться и изменяться с помощью тех же инструментов, которые применяются для работы в 3D-пространстве; возможно уменьшать и увеличивать кнопки инструментов тем же путём, что и в 3D-просмотре. Пользователь имеет возможность организовывать свое рабочее пространство удобным для него образом, что оптимизирует работу над уникальными задачами (<https://clck.ru/345cd5>).

В связи с доступностью и универсальностью программы существует огромное количество примеров продуктов, выполненных с помощью Blender. Sean Kennedy, обладатель двух премий «Оскар» в номинации лучшие визуальные эффекты за фильмы «Золотой компас», «Жизнь Пи» рассказал на Blender Conference 2017 о сериалах в которых он использовал Blender. Среди них: «Смертельное оружие», «Подача», «Мир дикого запада», «Орвилл», «Одарённые» (<https://clck.ru/34pseo>). Мультфильм «Друзья следующего поколения» был полностью создан в Blender развивающейся канадской студией Tangent Animation для Netflix.



Рис. 2. Постер мультфильма «Next Gen» (Друзья следующего поколения)

Почти все профессиональные программы для скульптинга включают в себя встроенные рендеры. Так, в поставку Blender входят несколько рендеров разных версий. Blender Render (Blender Internal) – оригинальный движок рендеринга Blender, исходный код которого был написан ещё в 90-х. Это смесь новых и старых технологий рендеринга, включающих в себя трассировку лучей, подповерхностное рассеивание, глянцевые отражения и даже примитивная система глобального освещения. Удалён из Blender в ветке 2.8.

Clay Render – Гипсовый рендер, применяет материал глины ко всем объектам сцены, без изменения их материалов. Входит в поставку Blender с версии 2.79.

EEVEE – Назван в честь покемона Иви, позже придумана расшифровка Extra Easy Virtual Environment Engine. Представляет собой полнофункциональный PBR движок для визуализации в реальном времени. Доступен в Blender с версии 2.8 (<https://clck.ru/34pcfP>).

За последние десятки лет было создано еще множество достойных программ для цифрового скульптинга, которые имеют собственные достоинства и недостатки. Рассмотрение всех их займет больше времени, но каждый художник найдет для себя подходящий софт. Данная статья может помочь с выбором между двумя представленными программами. ZBrush показывает себя как более адаптированная под художественные нужды программа, обладающая интуитивно понятным интерфейсом и широкими возможностями, но она является платной и может допускать незначительные ошибки. Blender является более сложной программой, имеющей строгий интерфейс, которая потребует достаточно времени для изучения. Но Blender имеет чуть более обширные функции, его можно получить бесплатно, что не потребует вложений для начала работы. Современный мир развивается с огромной скоростью, поэтому каждому художнику, желающему выйти на высокий уровень, необходимо будет изучать цифровую скульптуру.

Литература

1. Сметанникова Т.А., Ананьева Т.В. Трехмерное моделирование наградных изделий // Архитектура и дизайн в цифровую эпоху (г. Москва, 23-24 апреля 2021г.). Москва, 2021. С. 399-405.
2. Kurzweil R. The Law of Accelerating Returns. 2001. <https://clck.ru/34pccw>

© Гребеникова С.А., Сметанникова Т.А., 2023

УДК 004.056.5

Енин В.М.

Нижевартовский государственный университет
г. Нижневартовск, Россия

ЗАЩИТА И РЕЗЕРВАЦИЯ ДАННЫХ СИСТЕМЫ

Резервация данных – залог качественной и бесперебойной работы систем предприятия, а также защита от вирусов-шифровальщиков.

Если данные системы будут стёрты или повреждены (любым способом), то при помощи резервной копии можно восстановиться до функциональной версии, теряя минимум информации.

Обратимся к статистике. Эксперты «Информзащиты» сообщают, что в январе-июне 2022 года количество киберинцидентов с участием вредоносных программ-вымогателей, которые блокируют работу ИТ-инфраструктуры организации, на российском рынке выросло почти в два раза. Аналитики «Group-IB» зарегистрировали четырёхкратный рост по сравнению с тем же периодом 2021 года. Рост числа атак на предприятия зарегистрировали и в «Лаборатории Касперского» [3].

Знаковым в этом вопросе является событие 2018 года, а конкретно – появление вируса-шифровальщика «WannaCry».

WannaCry (также WannaCrypt или Wana) – вирус-вымогатель, получивший широкое распространение в 2017 году.

После успешного взлома компьютера, WannaCry пытается распространяться по локальной сети на другие компьютеры, как червь. Он сканирует другие компьютеры на предмет наличия той самой уязвимости, которую можно эксплуатировать с помощью EternalBlue, и если находит, то атакует и шифрует и их тоже.

Получается, что, попав на один компьютер, WannaCry может заразить всю локальную сеть и зашифровать все компьютеры, присутствующие в ней. Именно поэтому серьезнее всего от WannaCry досталось крупным компаниям, чем больше компьютеров в сети, тем больше ущерб.

Существует множество способов защитить данные резервной копией. Остановимся на некоторых из них.

Физическая резервация данных

Представляет собой жёсткий диск, разбитый на разделы для сохранения образа системы Windows, специальными программами или же стандартными настройками Windows.

Данный способ не требует подписки или вложения средств, кроме покупки винчестера.

Из плюсов можно отметить высокую скорость восстановления и записи файлов в Backup, а также то, что диск всегда будет под рукой для восстановления системы в случае угрозы или уже случившейся атаки шифровальщиком.

Распределение киберинцидентов по метрикам (объекты атак, методы, последствия) и внутри категорий	Категории жертв										
	Госучреждения	Промышленность	Медицинские учреждения	IT-компании	Наука и образование	СМИ	Сфера услуг	Другие	Без привязки к отрасли	Частные лица	
Всего атак	100	49	69	23	32	31	25	140	140	105	
Объект	Компьютеры, серверы и сетевое оборудование	51	45	61	19	28	15	18	105	119	34
	Веб-ресурсы	44	4	4	5	3	15	3	41	14	2
	Люди	39	21	37	4	15	7	13	43	69	95
	Мобильные устройства	1					1		1		18
	Другие	2		1	1				10	3	1
Метод	Использование ВПО	38	30	29	12	18	6	9	56	79	48
	Социальная инженерия	39	21	37	4	15	7	13	43	69	95
	Эксплуатация уязвимостей	34	25	23	14	12	18	12	61	55	6
	Компрометация учетных данных	6	3	10	5	4	2	1	10	8	3
	Компрометация цепочки поставок	10	1	4					2	3	2
	Другие	17	1	1	2	1	5		32	16	
Последствие	Утечка конфиденциальной информации	39	23	59	12	18	5	18	56	42	58
	Нарушение основной деятельности	40	17	12	6	13	16	3	51	22	1
	Прямые финансовые потери		2	5	1	1	1		20	10	26
	Ущерб интересам государства	26					18		6	3	2
	Использование ресурсов компании или частных лиц для проведения атак	5	1	2	5	2	1	5	7	12	6
	Другое		1		1	1			1	3	3
	Неизвестно	28	20	3	4	7	1	2	26	66	24

© Positive Technologies 0% 10% 20% 30% 40% 100%

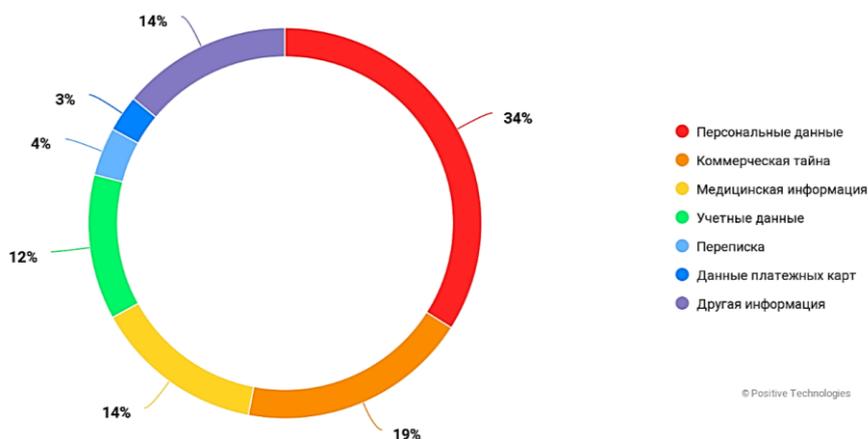


Рис. Статистика по киберинцидентам и типам атакуемых данных [2]

Сетевая резервация данных

Резервация данных на сервер предприятия при помощи локальной сети. На сервере ставится специальная программа для резервации данных системы (или всего образа). На компьютерах сотрудников будет стоять аналогичная программа, с зашифрованным подключением для обращения к серверу и передачей на него своих данных.

Главным плюсом данной резервации является практически полная безопасность резервных копий, так как они будут находиться только в серверной, как и диск резерва. Даже если вирус попадёт на компьютеры пользователей организации, то он не затронет сервер. Зашифровать данные на диске сервера будет практически невозможно.

Разница между созданием резервной копии и созданием образа системы

Существует два метода резервного копирования данных. Первый метод заключается в резервном копировании только необходимых данных, таких, как документы, видео и фотографии, а второй метод заключается в резервном копировании всех данных внутри компьютера, включая операционную систему [1].

Метод резервного копирования всех данных также называется «созданием образа системы». Если пользователь создаёт образ системы, он может восстановить его состояние на момент создания в маловероятном случае возникновения проблемы с компьютером. Однако при таком способе обновлённые данные после создания образа системы не будут отражаться, и если с самого начала возникла проблема с компьютером, то образ системы будет создан с учётом проблемы.

Можете более эффективно создавать резервные копии компьютера, используя резервное копирование только необходимых данных и создание образа системы.

Процедура резервного копирования компьютера

Что должен предпринять пользователь для резервного копирования данных на своём компьютере? Представим процедуру резервного копирования для операционной системы Windows 10.

Есть несколько шагов для резервного копирования данных в этой системе. В любом случае заранее необходимо подключить целевое устройство резервного копирования.

Как сделать резервную копию некоторых данных:

- щёлкнуть правой кнопкой мыши по «Пуск», и выбрать «Настройки»;
- щёлкнуть «Обновление и безопасность» – «Резервное копирование»;
- нажать «Добавить диск» и выбрать носитель для резервного копирования.

В правой части экрана будет отображаться «Автоматическое резервное копирование файлов», поэтому включим его.

Если нужно создать резервную копию немедленно или установить интервал резервного копирования и папку, то нужно нажать «Дополнительные параметры» и выполните настройки.

Как создать образ системы:

- ввести «Панель управления» в окне поиска в левом нижнем углу экрана, чтобы открыть экран панели управления;
- нажать «Система и безопасность» – «Резервное копирование и восстановление (Windows 7)»;
- нажать «Создать образ системы» слева;
- после выбора места назначения резервного копирования, нажать «Далее», проверить данные, для которых требуется создать резервную копию, и начать резервное копирование.

В заключении рассмотрим примеры сторонних программных утилит для резервного копирования, указав их достоинства и недостатки.

Macrium Reflect

Достоинства:

1. бесплатная версия имеет широкий функционал и имеет все те же функции, что и платная версия, но имеет ограничение по резервации компьютеров (до трёх);
2. имеется возможность создать «аварийную флешку» или интегрировать Windows PE в Windows Boot Manager, что облегчает восстановление системы, даже при полном уничтожении загрузочного файла boot.ini или части операционной системы;
3. хранить резервные копии можно на FTP-сервере или в общих папках. Утилита сама просканирует файловые директивы и выдаст на выбор файлы восстановления;
4. может создать полный образ системы, а также снимок экрана, даже при работе за компьютером.

Недостатки:

1. для работы с сервером Windows Server (2008, 2008 R2, 2012, 2022) необходима покупка программного продукта типа Server;
2. не имеет собственного облачного хранилища данных по логину и паролю.

Acronis

Достоинства:

1. имеет бесплатную версию на один компьютер, что подойдёт для использования компьютера в личных целях;
2. имеется функция аварийной флешки;
3. есть возможность хранить резервные копии образа системы в специальном арендованном облачном хранилище;
4. может создать образ диска, не зависимо от типа операционной системы (Windows, Linux, MAC).

Недостатки:

1. программа является платной, также дополнительная плата потребуется и на аренду облачного хранилища;
2. не имеет функции резервации образа системы без отключения компьютера.

Литература

1. Мийзамов А.А., Енин В.М., Матющенко И.А. Актуальные вопросы кибербезопасности // International Journal of Advanced Studies in Computer Engineering. 2021. №1. С. 17-21. 3
2. Обзор киберугроз за I квартал 2022 года. <https://clck.ru/34pckq>
3. Самойдюк А. Безопасность и конфиденциальность: в чём разница? <https://clck.ru/34pckW>

© Енин В.М., 2023

УДК 519.687.7

Зубаль А.М.

Нижевартовский государственный университет
г. Нижевартовск, Россия

РАЗРАБОТКА АВТОНОМНОГО РОБОТА

В данной статье рассматривается разработка автономного робота на платформе Arduino, использующего два сонара для ориентирования в пространстве, 4 мотора для передвижения и аккумулятор.

За основу был выбран контроллер на основе Arduino – Iskra Neo (см. рис. 1), работающий на частоте 16 МГц и обладающий 32 КБ флеш-памяти для хранения прошивки, 2,5 КБ оперативной памяти и 1 КБ энергонезависимой памяти для хранения данных при обесточивании (<https://clck.ru/34pdBn>).

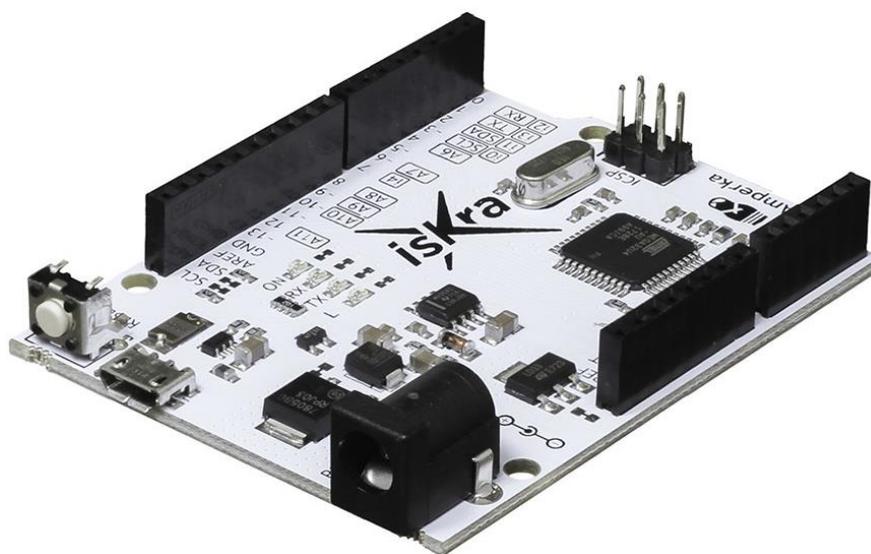


Рис. 1. Контроллер Iskra Neo

В качестве сонара выбран ультразвуковой датчик Arduino HC-SR04 (см. рис. 2) – датчик, генерирующий ультразвуковые сигналы на частоте 40кГц и получающий обратно отражённый сигнал. В отличие от инфракрасных датчиков, на показания ультразвукового датчика не влияет цвет объекта. Встроенный таймер измеряет время, прошедшее между отправкой сигнала и его приёмом, за счёт чего и высчитывается расстояние до объекта (<https://clck.ru/34pJPz>).

В качестве моторов используются моторы-редукторы с пластиковыми шестернями с рабочим напряжением от 3 до 12 В (см. рис. 3).

Для управления моторами используется Motor Shield (см. рис. 4) – плата расширения, предназначенная для двухканального управления скоростью и направлением вращения коллекторных двигателей постоянного тока, напряжением 5-24 В и максимальным током до двух ампер на канал (<https://clck.ru/TiR3Z>).

В качестве корпуса для робота была выбрана платформа Arduino для 4-х колёсного робота 4 WD Mobile Robot Platform Smart Car Chassis Arduino Compatible (см. рис. 5).



Рис. 2. Ультразвуковой датчик



Рис. 3. Мотор-редуктор

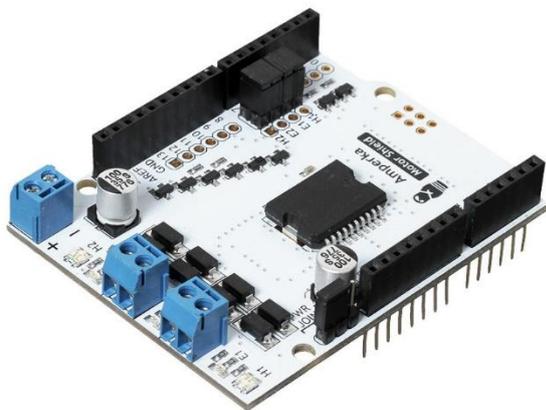


Рис. 4. Плата Motor Shield



Рис. 5. Корпус для робота

Автономного робота при соответствующих доработках можно использовать как основу для робота пылесоса, дрона для разведки местности или робота для доставки.

В первую очередь при разработке автономного робота нужен был принцип работы и его приблизительное устройство, которое в будущем будет модифицироваться и дорабатываться, в зависимости от полученных во время испытаний данных. Основным принципом для построения робота был принцип прохождения лабиринта через правило “одной руки” – чтобы робот постоянно продолжал путь, мы просто будем смотреть, нет ли впереди него препятствия и в случае чего всегда поворачивать в одну выбранную сторону.

Взяв этот алгоритм за основу осталось лишь собрать робота, написать исходный код, провести испытания и доредактировать исходный код в зависимости от полученных на испытаниях результатов.

Сама конструкция робота очень простая (см. рис. 6) – четыре мотора, осуществляющих передвижение робота за счёт надетых на них колёс, два сонара, для проверки на наличие препятствий перед роботом сверху и снизу, для обнаружения препятствий на разных высотах, закреплённый на корпусе источник питания для моторов и основа всего – плата Arduino с прикреплёнными к ней платой Motor Shield и аккумулятором.

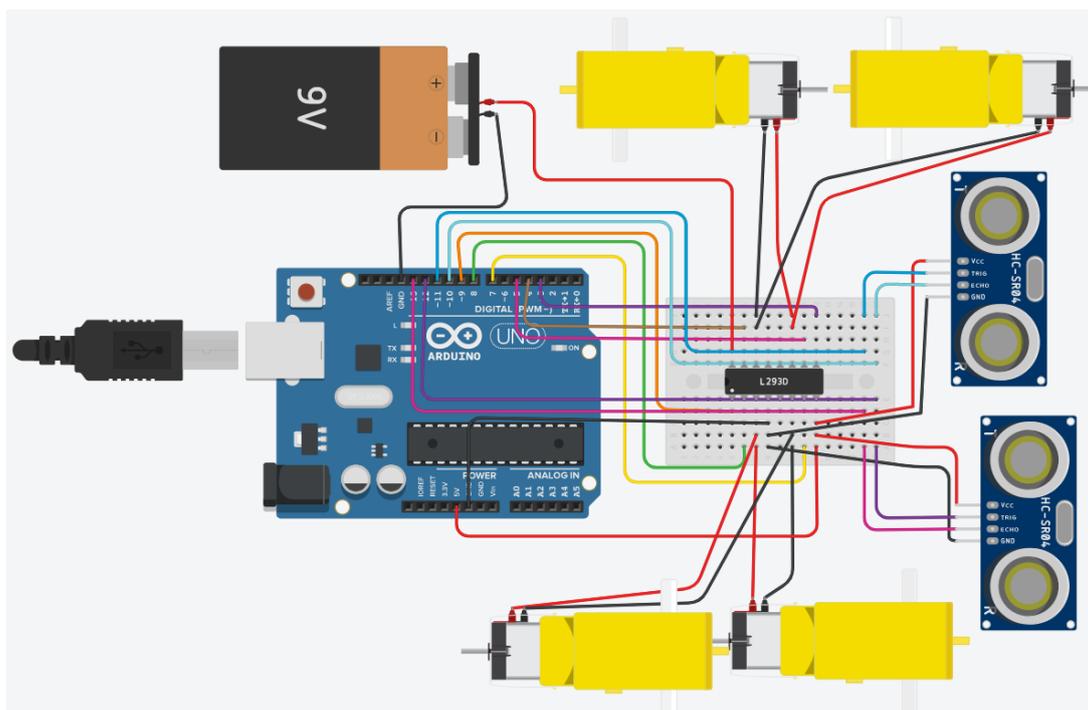


Рис. 6. Общая схема устройства робота

В написании кода в первую очередь для более оптимальной работы сонаров, а следовательно, и более оптимальной работы всего робота, нам потребуется библиотека `<NewPing.h>`.

После этого надо реализовать функции для движения робота вперед и назад и функцию для поворота, выбрать приблизительное расстояние до препятствия, при котором робот будет поворачивать.

После всего вышеперечисленного и базовых тестов проводится корректировка кода в зависимости от результатов тестов. Например, из-за невозможности сонаров уловить боковые препятствия добавляется функция, при которой робот отъезжает назад и поворачивает в том случае, если показания датчиков долго не меняются (что означает, что робот застрял).

© Зубаль А.М., 2023

УДК 378:372.8

Королюк Я.А., Куликова Н.Ю.

Волгоградский государственный социально-педагогический университет
г. Волгоград, Россия

ОПЫТ РАЗРАБОТКИ И ИСПОЛЬЗОВАНИЯ ИНТЕРАКТИВНЫХ ОБРАЗОВАТЕЛЬНЫХ РЕСУРСОВ НА ОСНОВЕ HTML5 ДЛЯ ОНЛАЙН-ОБУЧЕНИЯ ШКОЛЬНИКОВ ИНФОРМАТИКЕ

Происходящие в обществе процессы, связанные с развитием сети Интернет, появлением различных сетевых технологий, повлекли за собой изменения в сфере образования. Стали появляться качественно новые отличительные модели образования, в которых изменились роли субъектов образовательного процесса, которые становятся направленными на самообразование и развитие с раскрытием внутренних потребностей к учению [6]. На первый план выходят ключевые навыки для жизни в информационном обществе, среди которых особо важным, по мнению многих зарубежных и Российских исследователей, является «умение учиться» [11; 14; 15]. При этом базовым элементом является образовательная коммуникация, подразумевающая взаимодействие участников в интерактивном режиме, направленное на понимание и ценностно-смысловое согласование содержания обучения для его дальнейшего осмысления и присвоения [16].

Особенно сильно изменения затрагивают сам школьный курс информатики. Интенсивное развитие информатики как науки приводит к непрерывному обновлению и усложнению содержания обучения, увеличению его объемов, быстрой потере актуальности учебной информации в учебниках и электронных образовательных ресурсах [3; 5]. В таких условиях, где происходит стремительное развитие не только подходов к обучению, но и самого содержания предмета, находится учитель информатики, деятельность которого направлена на постоянное самосовершенствование в своей области знаний, а также на непрерывный поиск инновационных средств и методов обучения [4; 9].

Для реализации образовательных коммуникаций в условиях сетевого взаимодействия с обучающимися учитель может использовать большое количество сетевых сервисов и площадок для создания интерактивных заданий, видео, плакатов, презентаций, призванных облегчить работу учителя при организации и построении своих уроков. Существующие ресурсы разнообразны, однако они не всегда могут соответствовать необходимым требованиям изучаемой на уроке темы, поэтому многие учителя находятся в поиске альтернативных способов создания собственных ресурсов, которыми они смогут пользоваться неоднократно. Одним из современных способов организации образовательного пространства для обучающихся является создание веб-ресурсов учителя информатики с интегрированными в них интерактивными образовательными ресурсами. Самой популярной на сегодняшний день является технология HTML5, которая является не просто продолжателем языка разметки HTML, а новой открытой платформой, предназначенной для создания различных веб-приложений, интерактивных образовательных ресурсов, мультимедийного контента, с использованием аудио, видео, графики, анимации и др. [10].

Разберём основные понятия более подробно. Е.Л. Батакова и Н.В. Батакова считают, что интерактивность является главным педагогическим инструментом электронно-образовательных ресурсов, так как интерактивность – это возможность активного влияния на содержание, внешний вид электронного ресурса, манипулирование его объектами [2]. Вслед за Н.Ю. Куликовой под *интерактивностью* будем понимать взаимодействие участников образовательного процесса непосредственно или опосредованно средствами обучения с реализацией принципов обратной связи для обеспечения учебного диалога [11]. Под *интерактивным образовательным ресурсом* будем понимать образовательный ресурс, который представлен в электронной форме, содержит систематизированную информацию по какой-то учебной теме и позволяет реализовать интерактивное взаимодействие с обучающимися [12].

Для онлайн-обучения школьников удобно создавать веб-ресурс в виде сайта, на котором размещаются: структурированная информация, интерактивные задания, тесты, ссылки и др. Такой ресурс можно дополнять, изменять в зависимости от изменения содержания теоретической части или с появлением новых технологий, новых сервисов, что является очень удобным для такого предмета как информатика с его особенностями, описанными выше.

Для успешного создания веб-ресурса необходимо качественно разработать базовый образовательный контент. Для этого необходимо оценить исходный материал, отобрать и структурировать его. Здесь учителю информатики понадобятся его хорошие предметные знания в области информатики как науки.

Согласно Н.А. Пластининой базовыми принципами разработки образовательного контента являются соответствие содержания и сложности контента учебным целям, структурированность объёма, визуализация информативности и простота управления ресурсом [13].

Помимо предметных знаний, учитель должен обладать современными пользовательскими представлениями о дизайне, навигации, удобстве использования ресурса и комфортности работы с ним. Основные моменты, на которые стоит обратить внимание учителю при создании собственного веб-ресурса, это внутренняя и внешняя навигация (Где я сейчас нахожусь? К чему и как я перейду дальше? Как попасть на главную страницу, с которой всё началось?), графика (читаемость и системность представления данных), цветовая стилистика и порционное представление текста (комфортная, единая для всех страниц цветовая палитра, единый стиль оформления, сбалансированное размещение в тексте объектов графики) [13].

Учитель информатики имеет достаточно компетенций, чтобы разрабатывать собственные образовательные ресурсы с использованием не только сетевых сервисов, но и языка HTML5, CSS и JavaScript.

С.С. Бакулевская в своей работе обосновала преимущества данной технологии для учителей и выделяет следующие достоинства образовательных ресурсов, созданных с помощью HTML5 [1, с. 32]:

– Эффективность кода: разделение контента (HTML), информации о стиле его оформления (CSS) и поведения (JavaScript) позволяет использовать код повторно.

– Легкость сопровождения: разделение контента (HTML), информации о стиле его оформления (CSS) и поведения (JavaScript) позволяет вносить изменения в код с помощью классов и функций.

– Доступность: использование стандарта HTML5 позволяет без дополнительных усилий сделать веб-приложения доступными для людей с ограниченными возможностями.

– Совместимость с устройствами: веб-приложение будет работать не только на различных платформах (Windows, Mac, Linux), но также на альтернативных устройствах просмотра, таких как смартфоны, планшеты, цифровые панели и игровые консоли.

В Волгоградском государственном социально-педагогическом университете на факультете математики, информатики и физики ведется активная работа по подготовке будущих учителей информатики, математики и физики к созданию и использованию интерактивных образовательных ресурсов при онлайн-обучении школьников. На базе образовательного портала Волгоградского государственного социально-педагогического университета (<http://edu.vspu.ru>), частью которого является портал электронного обучения ВГСПУ (<http://lms.vspu.ru>) организованы сетевые сообщества студентов, учителей и школьников [6]. Студенты факультета математики, информатики и физики разрабатывают интерактивные образовательные ресурсы и встраивают их в онлайн-курсы, которые в дальнейшем апробируются на школьниках в рамках педагогической практики.

Далее приведем пример идеи разработки авторских интерактивных образовательных ресурсов для обучения школьников информатике по теме «Компьютер как универсальное устройство обработки информации». Данная тема, согласно тематическому планированию, состоит из пяти подтем (рис. 1).

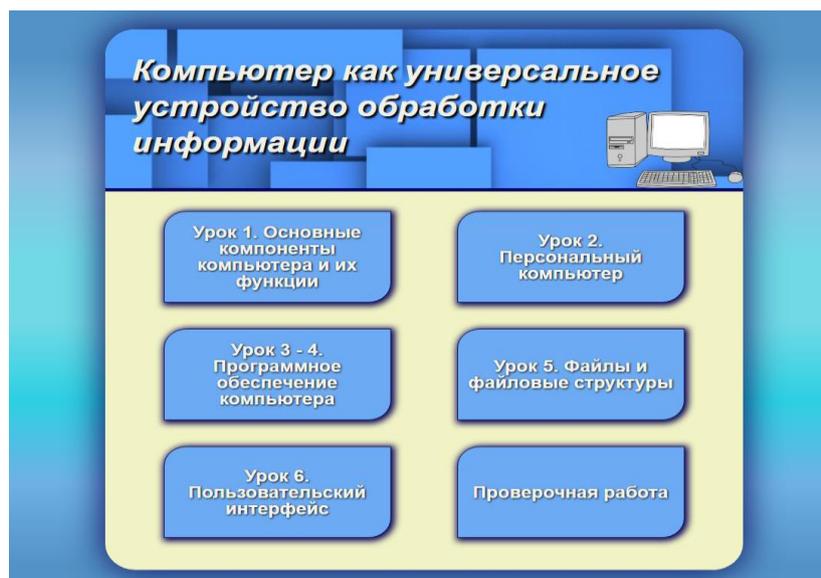


Рис. 1. Пример главной страницы

Каждая подтема имеет собственную страницу с теоретическим и практическим материалом (интерактивные задания, самоконтроль). Для реализации обратной связи были разработаны средствами HTML5, CSS и JavaScript и встроены в каждую подтему викторины.

На рисунке 2 показан пример для темы «Персональный компьютер». Такая викторина содержит: определенное количество вопросов; номер текущего вопроса; сам вопрос; четыре варианта ответа на текущий вопрос (один из ответов является верным); кнопку «Далее»; прогресс-бар для перехода между вопросами в виде кругов, которые окрашиваются либо в зелёный цвет, либо в красный, в зависимости от правильности выбранного ответа; финальное окно, показывающее количество верных ответов и имеющее кнопку повторного прохождения викторины.

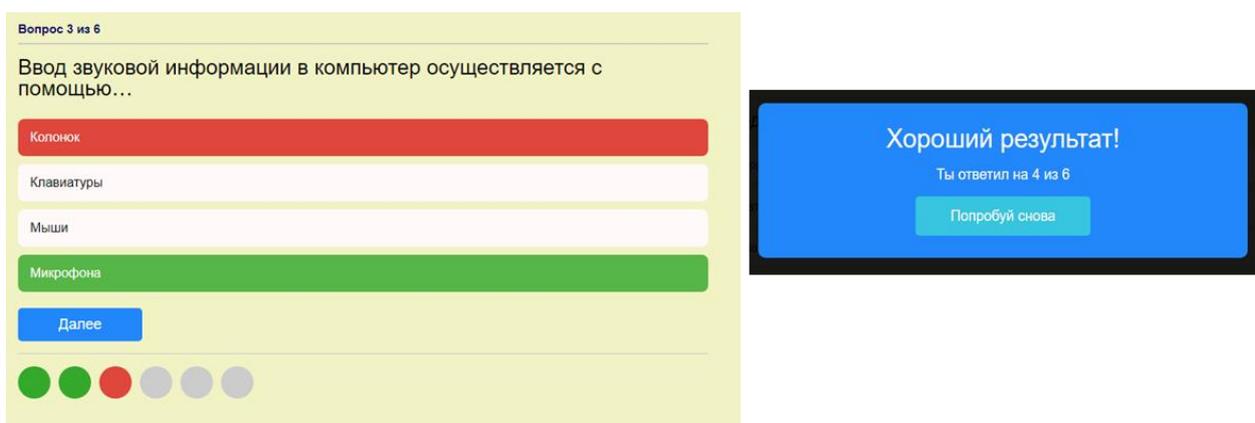


Рис. 2. Пример викторины для темы урока «Персональный компьютер»

На рисунке 3 представлен интерактивный ресурс по теме «Программное обеспечение компьютера» с использованием технологии drag-and-drop, на основе использования которого обучающимся необходимо произвести классификацию программного обеспечения компьютера. Данный ресурс имеет возможность проверки правильности выполнения классификации. Технология drag-and-drop позволяет работать с браузером простым перетаскиванием. Такая функция была добавлена в новый стандарт HTML5 и её использование значительно упростилось, теперь не нужно использовать много JavaScript кода [8]. Перетаскивание объектов мышью используется достаточно давно веб-программистами, для этого использовались JavaScript-библиотеки, такие как jQuery, ExtJS или ручное манипулирование DOM-объектами [17]. В стандарт HTML5 использование drag-and-drop изначально включено с реализацией новым атрибутом «draggable» (с событиями по каждому этапу действий по перемещению объектов: dragstart, drag, dragenter, dragleave, dragover, drop, dragend) [17].



Рис. 3. Пример задания на классификацию для темы урока «Программное обеспечение компьютера»

На рисунке 4 представлен интерактивный плакат, разработанный студентами, с четкой структурой, включающей в себя различные блоки (первого плана с начальным представлением об общем содержании учебной темы, представленной в интерактивном плакате и подчиненные ему блоки с развернутой информацией и встроенными интерактивными тренажерами, интерактивным видео, заданиями, тестами и др.). Навигацию по всем блокам интерактивного плаката позволяют реализовать встроенные управляющие элементы [11].



Рис. 4. Пример интерактивного плаката по теме «Информация и информационные процессы» в динамике онлайн-работы с ним

Интерактивные плакаты относят к визуальным дидактическим регулятивам логико-смыслового типа (Штейнберг В.Э., Манько Н.Н., Мустаев А.Ф. и др.) и являются эффективными инструментами учителя для решения поставленных им задач комплексно [11; 18; 19]. В интерактивных плакатах учитель может интегрировать электронные мультимедийные учебные материалы (теоретические ресурсы с интерактивным видео и мультимедийной информацией, интерактивные анимации и тренажеры для закрепления учебного материала и отработки получаемых навыков, тестовые и проверочные задания для реализации оперативного и итогового контроля) в одно педагогическое средство с учетом специфики онлайн-обучения и обеспечения многоуровневой работы с учебной информацией с ее максимальной визуализацией на всех этапах обучения [7; 11].

Анализ опыта создания и использования авторских интерактивных образовательных ресурсов показывает следующие возможности их использования для учителя при онлайн-обучении школьников:

1. Возможность восполнять постоянную нехватку готовых интерактивных образовательных ресурсов, связанную с непрерывным развитием информатики как науки и быстрым устареванием имеющихся ресурсов. Анализ педагогического опыта показывает, что учителя активно создают собственные веб-ресурсы с использованием как языка HTML5, так и сетевых сервисов Интернета.

2. Повысить качество обучения с помощью размещения на различных онлайн-платформах интерактивных образовательных ресурсов, например, интерактивных плакатов. При использовании онлайн интерактивных плакатов повышается эффективность обучения за счет использования в них: максимальной визуализации; организации четкой структуры; использования интерактивных возможностей для разворачивания и свертывания учебной информации; использования возможностей детализации учебного контента; реализации взаимодействия с интерактивным ресурсом как при очном, так и при дистанционном обучении; организации активной работы школьников с учебной информацией независимо от места и времени занятий.

3. Эффективно организовывать образовательную деятельность обучающихся на основе онлайн-сообществ школьников, позволяющих реализовать их всестороннюю социализацию, развитие и воспитание. Создавать онлайн-сообщества школьников учитель может на основе: сетевых и социальных сервисов интернета, своего личного образовательного сайта или базе своей образовательной организации создавать портал в Интернете.

4. Положительно влиять не только на образовательный процесс у обучающихся, но и повышать интерес к предмету, развивает самостоятельность, мотивацию.

5. Самостоятельно развиваться самому учителю, постоянно находиться в поиске новых открытий и полезных источников информации, строить собственные траектории обучения, совершенствовать свои педагогические навыки.

В заключении отметим, что созданные интерактивные образовательные ресурсы помогут учителю в дальнейшем экономить время и силы, а также будут полезными для обучающихся, которые по какой-либо причине не могут посещать учебное заведение.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 19-29-14064 «Теоретико-методологические основы и технологическое обеспечение реализации образовательной деятельности в онлайн-сообществах учащихся школ».

Литература

1. Бакулевская С.С. Подготовка педагогических работников к использованию технологий HTML5 для разработки электронных образовательных ресурсов // Информатика и образование. 2019. №5(304). С. 32-40.
2. Батакова Е.Л., Батакова Н.В. Интерактивные средства обучения как часть электронно-образовательных ресурсов // Вестник Томского государственного педагогического университета. 2016. №1(166). С. 105-109.
3. Бешенков С.А., Шутикова М.И., Рыжова Н.И. Формирование содержания курса информатики в контексте обеспечения информационной безопасности личности // Вестник РУДН. Серия: Информатизация образования. 2019. Т. 16. №2. С. 128-137.
4. Борисова Н.В., Бычкова Д.Д., Пантелеймонова А.В., Белова М.А. Информационно-образовательная среда в системе подготовки учителя информатики // Информатика и образование. 2020. №7 (316). С. 55-63.
5. Босова Л.Л. Современные тенденции развития школьной информатики в России и за рубежом // Информатика и образование. 2019. №1 (300). С. 22-32.
6. Данильчук Е.В., Куликова Н.Ю. Подготовка будущих учителей информатики к созданию и использованию виртуальных образовательных площадок в обучении школьников // Известия Волгоградского государственного педагогического университета. 2020. №10(153). С. 9-16.
7. Диков А.В. Интерактивные видеоуроки в социальных сетях Интернета // Школьные технологии. 2018. №1. С. 65-70.
8. Кизянов А. О. Использование технологии drag-and-drop в HTML5 // Постулат. 2018. №1(27). С. 10.
9. Ковчур С.А. Педагогическая обратная связь как показатель эффективности дистанционного обучения // Вестник Минского государственного лингвистического университета. Серия 2: Педагогика, психология, методика преподавания иностранных языков. 2020. №2(38). С. 32-39.
10. Конева А. С. Применение возможностей HTML5 для разработки образовательного медиаконтента в сети интернет // Электронное обучение в непрерывном образовании. 2016. №1. С. 259-265.
11. Куликова Н. Ю., Данильчук Е.В., Сергеев А.Н. Онлайн-обучение школьников информатике на основе веб-платформы с интерактивными плакатами: теория и опыт реализации // Информатика и образование. 2021. №6(325). С. 29-37. <https://doi.org/10.32517/0234-0453-2021-36-6-29-37>.
12. Кутепов М.М., Лебедева А.А., Максимова К.А. Дидактические возможности интерактивных электронных образовательных ресурсов // Балтийский гуманитарный журнал. 2020. №3(32). С. 128-130. <https://doi.org/10.26140/bgj3-2020-0903-0028>.

13. Пластинина Н.А., Григорьева Е.С. Создание базового образовательного контента для дистанционного обучения // Вестник Нижневартовского государственного университета. 2021. №1(53). С. 48-55. <https://doi.org/10.36906/2311-4444/20-4/07>.

14. Рамазанов Р.Г., Гриншкун В.В. Влияние сетевых сообществ и перехода к Web 3.0 на смену подходов к получению образования // Вестник РУДН. Серия: Информатизация образования. 2018. Т. 15. №1. С. 89-100.

15. Семенова Н.И., Солодейников М. А. Образовательная коммуникация в виртуальном сообществе как компонент сетевой учебной стратегии // Экстернат.РФ. 2021. №3(14). С. 105-110.

16. Сергеев А.Н. Инструменты дистанционного и электронного обучения в сообществах учащихся и педагогов: состав, особенности использования и предпочтения // Вестник Российского университета дружбы народов. Серия: Информатизация образования. 2020. Т. 17. №4. С. 323-336.

17. Сухов К. HTML5 – технология будущего. Продолжаем погружение // Системный администратор. 2012. №3(112). С. 74-81.

18. Штейнберг В.Э., Манько Н.Н. Современный дидактический регулятив: теория и технология // Отечественная и зарубежная педагогика. 2019. Т. 1. №5(62). С. 160-183.

19. Штейнберг В.Э., Мустаев А.Ф. Основания графической реализации логико-смыслового моделирования в дидактике // Образование и наука. 2017. Т. 19. №3. С. 46-76.

© Королук Я.А., Куликова Н.Ю., 2023

РАБОТА С RFID МЕТКАМИ СТАНДАРТА ISO/IEC 15693 НА ПРИМЕРЕ ПРИЛОЖЕНИЯ ДЛЯ АНДРОИД

RFID является технологией идентификации, которая открывает возможности для выполнения многих поставленных задач. К этой технологии относятся как метки (теги), так и считыватели (ридеры). Им находят применение во многих сферах. Радиочастотная идентификация (RFID) – использует электромагнитные поля для отслеживания меток, которые прикреплены к объектам и предметам, а также для их идентификации. В метках (или тегах) содержится информация в электронном виде.

У пассивных меток нет встроенного источника тока для питания. Пассивные теги собирают благодаря КМОП-чипу энергии посредством запроса радиоволн от рядом расположенного RFID - считывателя.

У активных меток присутствует локальный источник питания (батарея). Они могут работать на расстоянии сотни метров непосредственно от считывателя RFID. Метка может не быть расположенной в пределах прямой видимости устройства, что отличает её от штрих кода. В связи с этим она может быть встроена в отслеживаемый объект (<https://clck.ru/34pdkG>).

Где используются RFID метки.

RFID-метки (теги) могут использоваться во многих сферах. Такая метка может быть прикреплена к автомобилю на производстве или на стройке. Также может применяться для оптимизации логистики, автоматизации процессов, быстрой инвентаризации, складского учёта, идентификации животных, а фармакологическая продукция с такими тегами может быть отслежена.

Так как такие метки могут прикрепляться к наличным деньгам, личной собственности, элементам одежды и даже вживляться в человека и животных, считывание личной информации без согласия человека часто может быть серьёзным нарушением конфиденциальности.

Эти проблемы привели к разработке стандартных спецификаций для решения вопросов конфиденциальности и безопасности. Теперь при регистрации человека на закрытом производстве или строительстве, в случае получения сотрудником травмы, пострадавшего легче опознать, оказать первую помощь и вызвать скорую, например, при поражении электрическим током.

Метки разработаны и созданы с защитой против различных ударов, от света, дождя, масла, пыли, влаги и других жестких условий. Таким меткам обычно не нужны источники питания, т.к. они пассивны в работе и могут работать круглосуточно без риска потери энергии. Эти виды бирок для тяжелых условий эксплуатации часто прикрепляются к разного рода транспортным средствам для управления автопарком, отслеживания контейнеров для снабжения и идентификации транспортных средств.

Виды протоколов и стандартов RFID.

Все транспондеры, которые используются в системах RFID, работают на определенных частотах по регламентируемым протоколам – в соответствии с едиными международными стандартами, содержащими их описание.

ISO/IEC 18000

Наиболее популярные в последнее время стандарты, охватывающие частотный диапазон от 125 кГц до 2,45 ГГц. Поддерживаются как пассивными, так и активными – с миниатюрной батарейкой – радиометками, транслирующими сигнал на дистанции от 20 см до нескольких десятков метров.

ISO 11784/11785

Международные стандарты, которые регулируют теги, используемые в животноводстве. Такие радиометки часто имплантируются. Также транспондеры с микрочипами могут закрепляться непосредственно на теле животного.

ISO/IEC 14443

Международный стандарт, который описывает параметры идентификационных карт, работающих на частоте 13,56 МГц, правила и порядок их использования. Делится на два типа – Type A и Type B. Различаются схемами модуляции сигналов, принципами шифрования данных и способами предотвращения конфликтов, возникающих при работе с радиочастотными устройствами. Регламентированная дистанция информационного обмена – до 10 см.

ISO/IEC 15693

Популярный High RFID-стандарт, который используется при изготовлении меток, функционирующих на частоте 13,56 МГц, и чипов для бесконтактных банковских карт. Совместим с ISO 18000-3 и обеспечивает информационный обмен на больших дистанциях (<https://clck.ru/34pdkf>).

Область применения ISO/IEC 15693

Широко применяется в сферах:

- производства – для автоматизированной идентификации сырья и готовой продукции;
- медицинского обслуживания – для работы со смарт-картами, полисами, амбулаторными картами пациентов, документами врачей;
- транспорта – для управления парковками;
- безопасности – для контроля доступа в здании;
- финансов – при управлении активами;
- торговли и логистики – для маркировки ТМЦ;
- в библиотеках – для учета книг на полках.

Плюсы и минусы меток стандарта ISO/IEC 15693

Одним из главных плюсов метки по стандарту ISO/IEC 15693 является ее диапазон для срабатывания. Диапазон расстояния в пассивном режиме составляет до 1 метра, в активном режиме работы (с подключенным питанием) может достигать 100 метров.

Сигнал проходит сквозь большинство материалов за исключением металлов – происходит некоторое затухание.

Большой диапазон температур, при котором работа выполняется корректно. В среднем от -20 °С до +65 °С, в зависимости от модели устройства.

Так как метка может работать как в режиме с подачей энергии, так и без нее, у некоторых меток есть ограничения на то, какие функции они могут выполнять. Данная особенность является как плюсом, из-за своей универсальности и возможности считать первоначальные данные в экстренной ситуации, так и минусом, ведь без питания доступ к основным данным может быть ограничен. Так же без знания того, что на метку подается дополнительное питание, могут возникнуть проблемы со сроками работы на предприятии после аварии, замедлить разработку устройств и приложений, которые зависят от метки, а так же потерять эти данные в принципе.

Трудности могут возникнуть так же в манипуляциях с данными, т. к. принимать и отдавать их метка может в формате от одного до четырех блоков из четырех шестнадцатеричных чисел. Соответственно, для корректной работы необходимо строго соблюдать правила отправки сообщения для чтения или записи данных с устройства.

RFID и смартфоны

Сейчас большинство смартфонов поддерживают работу с NFC, что позволяет взаимодействовать с пассивной высокочастотной RFID, к которой могут относиться метки стандарта ISO/IEC 15693. У них так же есть ограничение на расстояние, с которого будет происходить взаимодействие с устройством. Обычно оно не превышает нескольких сантиметров. Для большего радиуса действия или любого другого типа RFID / активной RFID необходимо использовать внешний считыватель для работы с ними с мобильными устройствами.

Так же следует учитывать модель смартфона и версию его операционной системы. Так например модуль NFC у компании Apple присутствует начиная с модели Iphone 6, однако для взаимодействия с RFID метками необходима версия операционной системы IOS 11.

В смартфонах на операционной системе Android для этого есть специальные классы: NfcA, NfcB, NfcF, NfcV, NDEF. Данные классы были добавлены в 2010 году и до сих пор являются актуальными. Для наших целей подходит класс NfcV, с ним мы работать и будем.

Подготовка приложения к работе

Для начала, в файле манифеста необходимо написать разрешение на использование nfc .
<uses-permission android:name="android.permission.NFC"/>

Так же необходимо прописать intent-filter для нашей активности, который будет отвечать за обнаружение nfc-тэга поблизости.

```
<intent-filter>  
  <action android:name="android.nfc.action.NDEF_DISCOVERED" />  
  <category android:name="android.intent.category.DEFAULT" />  
</intent-filter>
```

```
<intent-filter>
  <action android:name="android.nfc.action.TAG_DISCOVERED" />
  <category android:name="android.intent.category.DEFAULT" />
</intent-filter>
<intent-filter>
  <action android:name="android.nfc.action.TECH_DISCOVERED" />
  <category android:name="android.intent.category.DEFAULT" />
</intent-filter>
```

Листинг 1. Файл manifest

Затем переходим в класс активности, в которой мы и будем работать с nfc. Там необходимо создать экземпляр класса NfcAdapter, определить его и проверить, есть ли на устройстве поддержка nfc.

```
nfcAdapter = NfcAdapter.getDefaultAdapter(context);
if(nfcAdapter == null){
    Toast.makeText(context, "No nfc support on this device", Toast.LENGTH_SHORT).show();
    finish();
}
```

Листинг 2. Определение nfc адаптера

После этого необходимо переопределить функцию onNewIntent, для обнаружения nfc - метки, определения тэга и дальнейшего работы с ним.

```
@Override
protected void onNewIntent(Intent intent) {
    super.onNewIntent(intent);
    // setIntent(intent);
    if(NfcAdapter.ACTION_TAG_DISCOVERED.equals(intent.getAction())
        || NfcAdapter.ACTION_NDEF_DISCOVERED.equals(intent.getAction())
        || NfcAdapter.ACTION_TECH_DISCOVERED.equals(intent.getAction())){
        if(myTag == null) {
            myTag = intent.getParcelableExtra(NfcAdapter.EXTRA_TAG);
        }
    }
}
```

Листинг 3. Обнаружение nfc тэга

После успешного определения метки, необходимо получить экземпляр класса NfcV из тэга.

```
if(nfcV == null){
    nfcV = NfcV.get(myTag);
}
```

Листинг 4. Определение nfcV экземпляра

После определения NfcV объекта можем приступить к работе с ним.

Команды для работы с NfcV

Для работы с nfcV определено 4 вида команд: обязательные, дополнительные, пользовательские, закрытые.

Обязательные команды расположены в диапазоне от '01' до '1F'. Коды дополнительных команд расположены в диапазоне от '20' до '9F'. Коды пользовательских команд находятся в диапазоне от 'A0' до 'DF'. Коды закрытых команд находятся в диапазоне от 'E0' до 'FF'. Список команд представлен на рисунке.

Код команды	Тип команды	Функция
'01'	Обязательная	Инвентаризация
'02'	Обязательная	Состояние покоя
'03' – '1F'	Обязательная	RFU
'20'	Дополнительная	Считать единичный блок
'21'	Дополнительная	Записать единичный блок
'22'	Дополнительная	Блокировать блок
'23'	Дополнительная	Считать набор блоков
'24'	Дополнительная	Записать набор блоков
'25'	Дополнительная	Выбор
'26'	Дополнительная	Возврат в состояние готовности
'27'	Дополнительная	Записать идентификатор семейства приложений AFI
'28'	Дополнительная	Блокировать идентификатор семейства приложений AFI
'29'	Дополнительная	Записать идентификатор формата хранения данных DSFID
'2A'	Дополнительная	Блокировать идентификатор формата хранения данных DSFID
'2B'	Дополнительная	Получить системную информацию
'2C'	Дополнительная	Получить состояние безопасности нескольких блоков
'2D' – '9F'	Дополнительная	RFU
'A0 – 'DF'	Пользовательская	В зависимости от производителя ИС
'E0 – 'FF'	Закрытая	В зависимости от производителя ИС

Рис. Список команд для работы с NfcV

В данной статье мы рассмотрим команды чтения/записи, т.к они используются чаще всего.

Для выполнения команд необходимо отправить массив байтов, который состоит из:

- флага(от '00' до '60');
- номера команды;
- UID метки;

- номера блока(целевого либо начального);
- данных(если мы используем команды для записи);
- количество блоков(если используем функции с чтением/записью более чем одного блока).

Для того, чтобы передать данные для чтения и записи с тэга `nfcV` используется команда `transceive`. На вход она получает заранее подготовленный массив байтов, состоящий из флага, кода команды, `id` тэга, данных либо номеров начального, конечного или целевого блока.

Команда для чтения единичного блока

Сначала формируем массив байтов, открываем соединение с тэгом, после начинаем отправку их в наш тэг, по окончании необходимо закрыть соединение. Каждое действие с `nfcV` должно происходить в отдельном от основного потока, иначе приложение сломается.

```
byte[] id = myTag.getId();//получаем id тэга
byte[] read = new byte[3 + id.length];//создаем массив байтов, который будем
формировать
byte blockNo = 0;//номер блока, который будем считывать
read[0] = 0x60;//флаг
read[1] = 0x20;//код команды
System.arraycopy(id,0,read,2,id.length);
read[2+id.length] = blockNo;
new Thread(new Runnable() {
    @Override
    public void run() {
        try {
            if(!nfcV.isConnected()){
                nfcV.connect();
            }
            byte[] result = nfcV.transceive(read);

        } catch (IOException e) {
            e.printStackTrace();
        } finally {
            if(nfcV.isConnected()){
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        Toast.makeText(context, "NfcV is connected",
Toast.LENGTH_SHORT).show();
                    }
                });
            }
        }
    }
});
```

```
    }else{
        runOnUiThread(new Runnable() {
            @Override
            public void run() {
                Toast.makeText(context, "NfcV is not connected",
Toast.LENGTH_SHORT).show();
            }
        });
    }
}
}).start();
```

Листинг 5. Пример кода для чтения единичного блока

Так как код для работы со следующими функциями идентичен, ограничусь только формированием массива байтов, для последующих команд.

Команда для считывания набора блоков

```
byte[] id = myTag.getId();
byte[] read = new byte[4 + id.length];
byte blockNo = 0;//номер начального блока
byte blockCount = 50;//количество блоков для считывания
System.out.println("block count : " + (nfcV.getMaxTransceiveLength()+1)/8);
read[0] = 0x60;
read[1] = 0x23;
System.arraycopy(id,0,read,2,id.length);
read[2+id.length] = blockNo;
read[3+id.length] = blockCount ;
```

Листинг 6. Пример кода для чтения набора блока

Команда для записи единичного блока

```
String s = "Home";
byte[] content = s.getBytes(StandardCharsets.US_ASCII);
StringBuilder sbd = new StringBuilder();
for (byte b : content) {
    sbd.append(String.format("%02X ", b));
}
byte[] data = new byte[3+content.length];
data[0] = 0x00;
data[1] = 0x21;
```

```
data[2] = 1;  
System.arraycopy(content,0,data,3,content.length);
```

Листинг 7. Пример кода для записи единичного блока

Команда для записи набора блоков

```
String s = "Our home";  
byte[] content = s.getBytes(StandardCharsets.US_ASCII);  
StringBuilder sbd = new StringBuilder();  
for (byte b : content) {  
    sbd.append(String.format("%02X ", b));  
}  
byte[] data = new byte[4+content.length];  
byte blockCount = 1;  
data[0] = 0x00;  
data[1] = 0x24;  
data[2] = 1;  
data[3] = blockCount;  
System.arraycopy(content,0,data,4,content.length);
```

Листинг 8. Пример кода для записи набора блоков

Важное замечание, счет количества блоков для записи начинается с 0. Следовательно значение 0 соответствует 1 блоку для записи, значение 1 соответствует 2 блокам для записи.

RFID метки стандарта ISO/IEC 15693 являются крайне универсальным средством идентификации объектов в разных сферах. Они применяются практически везде. Обладают очень неплохими характеристиками, а так же довольно небольшой стоимостью на рынке.

В работе они показывают себя стабильно хорошо, мгновенно откликаются на направленный на них считыватель, достаточно быстро выполняют команды и обмениваются данными. Имеют большое количество форм-факторов, стойкость к температурам, активный и пассивный режим работы. Отлично подойдут для стандартных нужд по типу пропуска на место работы, так и специфичных, по типу сбора данных с отслеживающего оборудования.

В статье представлены результаты изучения стандарта ISO/IEC 15693-3:2009, а также разобран алгоритм работы с метками RFID на примере Android-приложения.

© Косовцов Н.А., 2023

УДК 004

Костюченко П.А., Корнаухова М.А.
Волгоградский государственный университет
г. Волгоград, Россия

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ МОНИТОРИНГА УСПЕВАЕМОСТИ СТУДЕНТА

В данной работе рассмотрен процесс проектирования и создания клиент-серверного мобильного приложения для мониторинга успеваемости студента на базе фреймворка React Native, языка программирования Java Script, а также с помощью асинхронного хранилища данных Async Storage и облачного хранилища данных Cloud Firestore, используемого как сервер.

Разработанное мобильное приложение необходимо для того, чтобы обеспечить каждому студенту простой и удобный способ получения актуальных данных об успеваемости. В основном эти данные можно получить лишь на сайте учебного заведения, что не всегда возможно. Загрузка сайта требует больших вычислительных мощностей и наличие постоянного интернет соединения. Данные недостатки порождают необходимость поиска современных и актуальных способов распространения информации. Одним из таких средств является мобильное приложение для мониторинга данных об успеваемости студента.

В настоящий момент информационные системы широко применяются в различных областях деятельности [4; 5]. Одним из возможных вариантов использования является применение этих систем для контроля успеваемости. Далее представлены некоторые работы в этой области.

В работе [1] авторами рассматривается информационная система для контроля за успеваемостью. Данная программа разработана при помощи стека технологий Java Spring MVC + Web-сервис + Xamarin-приложение. В работе подробно описаны все преимущества и недостатки каждой технологии для разработки подобного рода проектов. Также автором описываются возможные паттерны поведения пользователя. Программа состоит из сервера, на котором непосредственно хранятся данные, и клиента, который имеет функции считывания и хранения данных. В качестве данных в данной программе выступают некие учебные материалы, которые может разместить пользователь с ролью преподаватель, а выполнить и разместить выбранный материал пользователь с ролью ученик.

Передача данных между базой данных и пользователем передается по протоколу HTTPS, что является не вполне стандартным решением, так как в случае с учебными материалами, коими в том числе могут являться не только текстовая информация, но и также аудио или видео материалы. Материалы такого рода обычно передаются по протоколам UDP, так как передача по HTTPS, протоколу хоть и является более надежной, но считается избыточной, и в то же время снижает скорость передачи таких объемных видов информации.

В работе [3] рассматривается программа, которая представляет собой электронный журнал учета успеваемости учащихся школы. В ней очень подробно описан анализ языков

программирования, в том числе таких как C, C++, Python, которые традиционно не применяются для разработки такого рода программ.

Помимо этого, данная работа подробно описывает как программные, так и аппаратные средства разработки, что также не несёт практической пользы для раскрытия темы создания электронного хранилища данных для сведений об успеваемости. В конечном итоге автором был выбран язык программирования JavaScript, а также язык разметки документов PHP и система управление базами данных MySQL. Приложение представляет собой небольшой веб-сайт, на котором в виде таблицы выводится информация об успеваемости определённой группы учеников. При этом в работе, не указано осуществлялись ли средства разработки по предотвращению простейших атак, таких как, например SQL инъекции. Также в схеме логического устройства базы данных представлено соотношение сущности и типа данных значения этой сущности. В большинстве случаев значения, которые являются ключевыми значениями базы данных формируются случайным образом, а не формируются посредством хэш функций или других инструментов. Такой способ задания ключевых значений не является надёжным и может хоть и с маленькой вероятностью, но привести к возникновению коллизий.

В целом работа предоставляет простой и понятный интерфейс, который подходит для решения задачи фиксации в электронном виде данных об успеваемости конкретного учебного заведения, однако система слабо поддаётся масштабированию или какому-то нетривиальному изменению заложенного в ней функционала.

В данной работе представлено приложение для мониторинга успеваемости студентов. В качестве стека разработки был выбран фреймворк языка JavaScript ReactNative, который позволяет вести кроссплатформенную разработку приложений с помощью специального языка XML.

Разработанное приложение представляет собой клиентскую часть, которая в свою очередь состоит из логики доступа к данным, а также локального асинхронного хранилища, которое предоставляет фреймворк. Второй частью приложения является облачное хранилище данных, в котором содержится набор так называемых каталогов, каждый из которых содержит в себе данные об успеваемости определённой группы за какой-то семестр. Сам каталог представляет собой набор документов, имя каждого из документов содержит номер зачетки определенного студента, а внутри каждого документа размещены структура типа ключ-значение, которые легко интегрируются в фреймворк и позволяют обновлять данные без лишних манипуляций.

На этапе проектирования приложения необходимо с учётом выбранных для разработки программных средств спроектировать архитектуру приложения, а также его основные функциональные и структурные элементы. Для разработки приложения была выбрана архитектура клиент-сервер, однако с большим количеством реализации логики на стороне клиента. Данное решение обусловлено тем, что функционал приложения не подразумевает необходимости использования слишком больших вычислительных мощностей, и современные мобильные устройства вполне могут обладать гораздо большим их запасом. Кроме того, для реализации условия масштабируемости приложения необходимо чтобы при

распространении приложения для его реализации в других учебных заведениях, не приходилось вносить сколько-нибудь значительные изменения в структуру данных об успеваемости студентов данного учебного заведения. То есть взаимодействие с информационными ресурсами ограничивается тривиальными запросами: put, push, delete. Такие запросы поддерживают абсолютное большинство баз данных, кроме того в качестве формата передаваемых данных используется язык JSON, помимо своей наибольшей популярности также легко переводится в любой другой формат данных.

Также на этапе проектирования были выбраны основные структурные и функциональные элементы приложения.

1. Форма регистрации, а также форме авторизации, для определения студента по в базе данных.
2. Таблица для вывода данных об успеваемости.
3. Формы для ввода данных об успеваемости преподавателями.
4. А также чекбоксы выпадающие списки и поля ввода для определения семестра, из которого получаются или в котором изменяются данные об успеваемости, предметов или групп.

Как уже было сказано ранее, для разработки приложения был выбран фреймворк языка JavaScript – React Native. Фреймворк React Native позволяет создавать стандартные компоненты, которые позднее могут использоваться в структурно совпадающих элементах. Данный функционал как правило значительно облегчает дальнейшее масштабирование приложения, за счёт использования уже написанных структурных элементов, и последующей передачи в них новых параметров, появившихся при масштабировании приложения.

Кроме того, теми же программными инструментами реализовано написания собственной библиотеки UI элементов. Данный приём позволяет создавать собственные библиотеки графических элементов для приложения. Каркас данных элементов прописывается единожды, однако стилистика может быть переменной в зависимости от параметров, которые в них передаются. Таким образом можно адаптировать экземпляр приложения под цветовую и стилистическую гамму, соответствующую другим информационным ресурсам учебного заведения. А также легко менять данную стилистику в случае тематических событий. Что касается программной реализации, то данный функционал, улучшает качество кода, избегая его дублирования и громоздкость [2].

Одним из требований к данному приложению является наличие офлайн доступа, чтобы расширить функционал изначального веб-ресурса - сайта образовательной организации. Для этого к приложению подключается библиотека AsyncStorage - некий аналог базы данных, который позволяет обращаться к себе без доступа к интернету. При каждом запросе в базу данных происходит копирование данных об успеваемости конкретного студента на его устройство и в дальнейшем студент может обращаться к ним. В случае с изменениями, которые вносят преподаватели в офлайн режиме, то они сохраняются на устройстве, и в дальнейшем при обращении к базе данных, сначала происходит сравнение данных с данными на устройстве и сначала к ним применяются изменения, которые отличают информацию на

устройстве от информации в базе данных, а потом применяет изменения, вносимые преподавателем. На рисунке 1 представлен пример базы данных учебного заведения, реализованный с помощью облачного хранилища данных Cloudstore FireBase. Как видно из рисунка структура базы данных представляет собой стандартную структуры базы данных, в который за отметку студента по конкретному предмету отвечают такие параметры как группа, семестр, и собственно сам предмет. Данная структура стандартная и общая для большинства учебных заведений, и при масштабировании приложения в уже прописанных запросах, необходимо будет лишь поменять названия параметров, или внести в них незначительные изменения.



Рис. 1. Пример используемой базы данных

Основной информацией, которую должно будет отображать приложение, будут сведения об успеваемости студента, а именно количество баллов, набранных за конкретную дисциплину в конкретный промежуток времени. Таким образом самым логичным способом отображения данной информации будет вывод на экран таблицы со сведениями об успеваемости студента. Также навигационная панель должна включать в себя возможность изменять интервал времени, за который отображается успеваемость студента. Интерфейс приложения представлен на рисунке 2.

При входе в приложение пользователь вводит номер своей зачётной книжки, данный номер служит названием каталога, в котором содержатся данные об успеваемости студента. При каждом обращении пользователя к серверу, данные в локальном хранилище автоматически обновляются, вне зависимости от того были они изменены или нет.

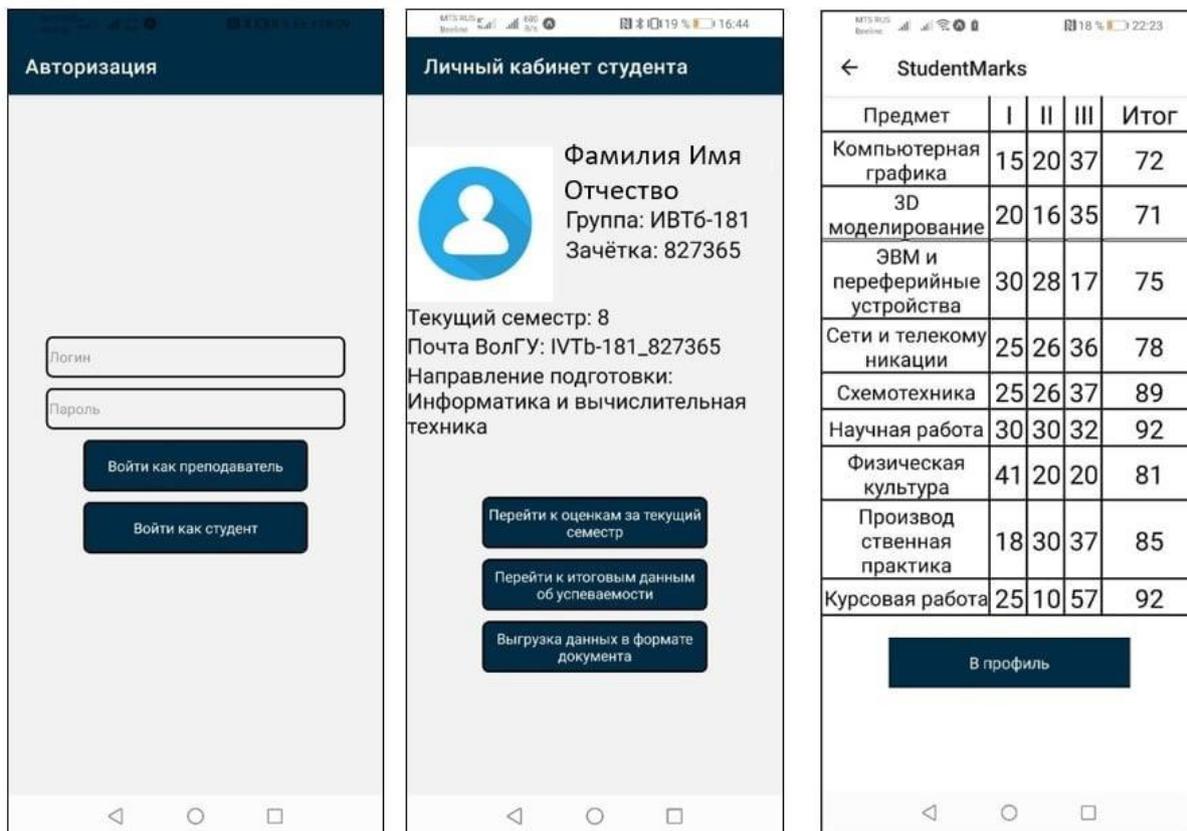


Рис. 2. Интерфейс мобильного приложения

Литература

1. Барабаш Г.И. Возможности программы ISPRING SUITE для подготовки урока информатики, контроля знаний и отслеживания успеваемости учащихся // Информатика: проблемы, методология, технологии. 2017. №5. С 81-84.
2. Завражин А.М. React Native. Разработка кроссплатформенного приложения // Аллея науки. 2018. №6(22). С. 1040-1047.
3. Захаров О.О., Бирюков Д.Н., Дудкин А.С., Алиев А.О. Программный модуль учета активности обучающихся в ходе дистанционного обучения: пат. 2021614275. заявл. 09.03.21; опубл. 22.03.21.
4. Корнаухова М.А., Основин С.С., Баскаков Е.В., Полусмакова Н.С., Савин Е.С. Информационная система помощи выбора специальности для абитуриентов на основе нейронной сети // Инженерный вестник Дона. 2022. № 11(95). С. 73-81.
5. Стефейкин Э.Н. Разработка модели комплекса корпоративных информационных систем и его интерфейсного комплекса для образовательного учреждения среднего профессионального образования // Студенческий. 2022. № 37-1(207). С. 29-38.

© Костюченко П.А., Корнаухова М.А., 2023

УДК 004.946

Куприянова Т.В, Поспелова Н.В.

Нижегородский государственный технический университет им. Р.Е. Алексеева
г. Нижний Новгород, Россия

О РАЗРАБОТКЕ AR ПРИЛОЖЕНИЯ ДЛЯ КИБЕР-МУЗЕЯ

В настоящее время дополненная реальность (Augmented Reality, AR) затрагивает многих людей. Проникновение AR в различные сферы жизни неизбежно. Использование этой технологии позволяет обучаться и взаимодействовать с клиентами, а также повышает социальную эффективность.

Компания Statista (<https://clck.ru/J8KjS>), которая занимается аналитикой и обработкой пользовательских данных, прогнозирует, что популярность AR приложений будет увеличиваться. Такой вывод был сделан в результате обработанных компанией Statista данных об объеме мирового рынка приложений дополненной реальности для смартфонов. По прогнозам, к 2024 году в мире будет 1,73 миллиарда пользователей мобильной AR.

По данным Министерства Культуры Российской Федерации (<https://clck.ru/34pfPt>), по всей России существует более 3700 музеев, и раньше им уделялось мало внимания. Поэтому главной задачей в веке информационных технологий является повышение уровня культуры и качества жизни человека, привлечение его внимания к культурным ценностям как в мегаполисах, так и в деревнях. Для этого нужны новые подходы и методы.

Будущее музеев в цифровое время однозначно будет связано с развитием информационных технологий. Использование дополненной реальности в стенах музея позволит увеличить приток новых посетителей, погрузиться в новую атмосферу и повысить вовлеченность в музейную экспозицию.

Существует разрыв реального и цифрового мира, из-за которого пользователю сложно извлечь максимум пользы из доступных данных. Такую проблему ограниченности информации можно решить с помощью внедрения технологий дополненной реальности. Использование AR объектов в реальном мире поможет пользователю быстрее получать и обрабатывать информацию.

Целью проводимой научно-исследовательской работы является проведение подготовительных мероприятий для разработки приложения с технологиями дополненной реальности для Кибер-музея. Для достижения поставленной цели планируется решение следующих задач:

1. Изучение возможностей и перспектив применения технологий дополненной реальности в музеях;
2. Выявление роли AR приложения непосредственно в Кибер-музее;
3. Анализ и сравнение существующих приложений-аналогов с дополненной реальностью;
4. Выбор программных инструментов для разработки AR приложения;
5. Построение архитектуры системы приложения с дополненной реальностью для Кибер-музея.

Дополненная реальность – одно из перспективных направлений будущего. В отличие от виртуальной реальности, AR в некоторой степени дополняет реальность, добавляет в среду элементы виртуальной реальности, но ни в коем случае не меняет ее полностью [2]. Это стирает грань между реальностью и созданной компьютером информацией. При этом для того, чтобы увидеть и взаимодействовать с виртуальными предметами, достаточно воспользоваться камерой смартфона. Повсеместное использование мобильных устройств предоставило общественности широкие возможности для ознакомления с AR приложениями в различных сферах.

Цифровизация повсеместно набирает обороты и, благодаря развитию Интернета вещей в России, создаются новые возможности для продвижения технологий дополненной реальности. Внедрение AR технологий во все сферы жизни способствуют росту рынка и извлечению прибыли. Уже сейчас в России дополненная реальность успешно применяется в таких областях, как здравоохранение, строительство, образование, логистика, индустрия развлечений и других.

Большинство музеев нашего времени в привычном для нас виде считаются устаревшими. Появилась мода на современные музеи, которые становятся мультимедийными центрами и увлекают посетителей интерактивными форматами взаимодействия с экспонатами. Благодаря современным информационным технологиям музеи преобразуются, а у посетителей создаются новые положительные впечатления.

Использование AR технологий в музее позволяет накладывать разнообразный медиаконтент (графика, анимация и видео) на реальный мир, что позволяет музеям оживлять коллекции. С помощью мобильных приложений предоставляется дополнительная информация об экспонате или о самом культурном учреждении, а также предлагается возможность использования персонального мобильного гида по музейной коллекции или галерейным пространствам.

В дальнейшем планируется использовать AR технологии при разработке AR приложения для Кибер-музея, который расположен в по адресу: Владимирская обл., г. Муром, площадь Крестьянина, д. 6В. Кибер-музей был открыт в 2012 году и стал первой экспозицией подобного рода в России. В интерактивном научно-техническом Кибер-музее представлено более 3000 экспонатов, рассказывающих об отечественной и мировой истории вычислительной техники: от первых разработок до их расцвета в 21 веке. При этом Кибер-музей будет интересен посетителям любого возраста, так как каждый найдет для себя занимательные экспонаты. На рисунке 1 представлены фотографии с официального сайта Кибер-музея (<https://clck.ru/34pfPU>), где посетители взаимодействуют с экспонатами.



Рис. 1. Взаимодействие посетителей с экспонатами Кибер-музея

Внедрение AR технологий положительно отразится на всей деятельности Кибер-музея, а именно:

1. Это поможет увеличить количество постоянных посетителей и привлечь новых;

2. Дополненная реальность создаст эмоциональную связь между Кибер-музеем и гостем.

Это может происходить благодаря «оживлению» экспонатов в привычном окружении, например, в доме посетителя;

3. Это культурное просвещение нового уровня. Без вмешательства экскурсовода, посетитель может узнать для себя что-то новое об экспонатах. Для получения новых знаний необходимо лишь отсканировать соответствующую картинку.

Для использования технологий дополненной реальности для Кибер-музея планируется разработать AR приложение. Создание собственного AR приложения необходимо по нескольким причинам:

1. Существующие приложения-аналоги были спроектированы для конкретного музея и его экспозиции. Переделывать их под другой музей - сложная и трудоемкая работа;

2. Существующие AR приложения имеют либо переизбыток функций (это осложняет пользователю взаимодействие с приложением), либо недостаток функций (из-за этого будет низкое вовлечение посетителя);

3. Некоторые существующие приложения-аналоги предполагают отрисовку слишком больших 3D-сцен. Это трудозатратная работа, которая на начальном этапе разработки AR приложения не принесет пользы.

Для разработки AR приложения для Кибер-музея необходимо обратиться к существующим аналогам, проанализировать их и выявить те функции, которые можно реализовать в новом продукте. Всего было выявлено более 30 современных музеев со всего мира, в которых используются мобильные AR приложения. Из них были выделены 14 музеев, наиболее популярных и с успешно реализованными разработками. Выбранные музеи сравнивались по следующим критериям и соответствующей нумерацией: возможность интерактивности (1), наличие больших 3D сцен, которые были специально созданы (2), наличие цифрового стенда (3), необходимость использования VR-гарнитуры (4), возможность использования виртуального гида (5), поддержка многофункциональности (6) и уровень

сложности реализации подобного AR приложения (7). На основе данных о каждом AR приложении для музеев, была разработана таблица 1, в которой представлена вся необходимая информация для анализа существующих AR приложений. Символом галочки отмечены пункты, в которых подчеркивается наличие той или иной характеристики, а символ крестика, наоборот, показывает отсутствие. Символ со знаком вопроса обозначает, что информации о данной характеристике не имеется.

Таблица

Сравнение существующих AR приложений для музеев мира

Наименование музея (или AR приложения для музея) и ссылка на официальный сайт	1	2	3	4	5	6	7
Кливлендский художественный музей (https://clck.ru/34pfVE)	✓	×	✓	×	×	×	High
Археологический Парк Карнунтум (https://www.carnuntum.at/de)	×	✓	×	×	✓	×	High
Музей кельтского наследия (https://clck.ru/34pfVc)	✓	×	×	×	✓	✓	Medium
Центр современного искусства в Сиэтле (https://clck.ru/33Usb4)	×	✓	×	×	×	×	High
Национальный музей Республики Марий Эл им. Т.Евсеева (https://www.fumus.ru)	✓	✓	×	×	✓	✓	High
Музей Антверпена (http://www.kmska.be)	✓	×	×	✓	✓	×	High
Третьяковская галерея (https://clck.ru/FzEoo)	✓	✓	×	✓	✓	✓	High
Виртуальный музей частной коллекции семьи Кремер (https://clck.ru/34pfWB)	×	✓	×	✓	✓	×	High
Google Art Project (https://clck.ru/JFu7s)	×	✓	×	×	×	×	Low
Музей-усадьба В.И. Сурикова (https://clck.ru/34pfXZ)	×	✓	×	×	×	×	Low
Детройтский институт искусств, платформа дополненной реальности Tango от Google (http://www.dia.org)	✓	✓	×	?	✓	✓	High
Национальный музей естественной истории в Вашингтоне, AR система «Broadcast AR» (https://clck.ru/34pfXs)	×	✓	×	?	×	×	High
Приложение Artefact (https://ar.culture.ru/)	✓	×	×	×	✓	✓	Medium
Государственный Дарвинский музей (https://clck.ru/34muvo)	✓	×	×	×	✓	✓	Medium

При сравнительном анализе существующих AR приложений для музеев выделено несколько решений, на которые можно будет ориентироваться при разработке AR приложения для Кибер-музея. К ним относятся: Музей кельтского наследия в г. Зальцбург, приложение Artefact и Государственный Дарвинский музей с AR квестом «Код жизни». На основе взятых в пример реализованных AR приложений были выявлены главные критерии успешной разработки AR приложения. К ним относятся:

Таким образом, исходя из проанализированной таблицы 1, были выявлены основные критерии успешной разработки AR приложения. К ним относятся:

1. Возможность интерактивности внутри приложения с 3D объектами (игры, квесты и т. д.);
2. Наличие виртуального гида по музею;
3. Многофункциональность приложения (поддержка нескольких видов взаимодействий с экспонатами);
4. Отсутствие необходимости разработки больших 3D сцен и использования цифрового стенда и VR-гарнитуры.

Важнейшим шагом перед разработкой AR приложения является выбор платформы, движка и инструментов для реализации проекта.

Говоря о выборе движка для настройки и компиляции приложения, был выбран межплатформенный движок Unity 3D [1], в котором будет прописываться код функционала на языке программирования C#. Этот выбор объясняется тем, что по сравнению с другими популярными программами, движок Unity наиболее простой, удобный и универсальный по сравнению с другими платформами, а также имеет самое большое профессиональное сообщество и наиболее подробную документацию.

В AR приложении будут использоваться 3D объекты, поэтому их нужно создавать под будущее приложение для Кибер-музея. Для решения этой задачи среди различных 3D программ был выбран Blender из-за его простоты в освоении, возможности бесплатного использования, наличия инструментов для моделирования ЭВМ и поддержка анимации.

При создании AR приложения также необходимо обратиться к дополнительному ПО, а именно – к плагинам (AR фреймворкам), самыми распространенными являются: Google ARCore, AR Foundation, Vuforia, EasyAR и Wikitude. Среди них был выбран плагин Vuforia SDK, поскольку он является мощным инструментом, обладает всеми необходимыми функциями и хорошо работает в кроссплатформенном движке Unity. Несмотря на недостаточно полную документацию, на просторах интернета существует множество видеуроков по решению различных задач.

Определившись с платформой, движком и инструментами, можно приступать к разработке архитектуры проекта. Но перед этим более подробно остановимся на значении AR приложения для Кибер-музея. При внедрении AR приложения планируется реализация цифровых музейных продуктов, направленных на взаимодействие с посетителями музея. С помощью нового формата предполагается увеличение количества посетителей, а также обеспечение интересного пребывания в музее. При этом часть цифровых продуктов будет монетизирована, что принесет дополнительный доход Кибер-музею. Разнообразный характер монетизируемых музейных цифровых продуктов (например, уникальные игровые AR карточки) и онлайн-событий (например, квесты, розыгрыши) повысит вовлеченность посетителей к музейной экспозиции и запустит процесс «сарафанного радио».

Применение AR технологий в пространстве Кибер-музея поможет решить проблему недоступности информации об экспонате, поскольку некоторые посетители при посещении музея не пользуются услугами экскурсовода и самостоятельно изучают музейные экспонаты. В таком случае большинство информации остается без внимания. А дополненная реальность позволяет «прикрепить» к экспонату информацию в виде текста, аудио или видео [3].

Мобильным приложением с дополненной реальностью для Кибер-музея можно будет пользоваться не только в стенах Кибер-музея, но и в доме посетителя. При запуске приложения будет возможно два варианта развития событий:

1. Если пользователь находится в Кибер-музее, он сканирует QR код, предоставленный в стенах Кибер-музея.

В этом случае пользователю доступно несколько функций: во-первых, это получение информации (в виде пояснительной таблички/статьи/видеообзора) об экспонате после сканирования QR-кода рядом с соответствующим объектом; во-вторых, это прохождение AR

квеста; и в-третьих, это виртуальный запуск ЭВМ (например, демонстрация того, как включается и работает компьютер ДВК-1, при условии, что реальный экспонат, находящийся в Кибер-музее в нерабочем состоянии). Кибер-музей позиционируется как интерактивный научно-технический музей, а также отличается от других музеев своими увлекательными загадками и квестами. Поэтому ожидается, что AR квест в Кибер-музее привлечет как новых, так и старых посетителей. А виртуальный запуск ЭВМ, их «оживление», порадует и удивит посетителей, относящихся к IT-сфере.

2. Если пользователь запускает приложение у себя дома, сканируется коллекционная карточка, приобретенная в Кибер-музее.

Для выполнения этого условия предварительно приобретается физическая коллекционная карточка в Кибер-музее. И поскольку это игра для двоих, требуется наличие оппонента с другой коллекционной карточкой из Кибер-музея. С точки зрения реализации для начала создается их дизайн (лицо и рубашка), а также в том же количестве 3D персонажи, смоделированные в стилистике Кибер-музея. Рубашки у карточек одинаковые, а на лицах изображены портреты привязанного к этой карточке героя. Все персонажи уникальны и имеют особые силовые характеристики. При сканировании физической карточки на ней появляется герой, его имя и боевые способности. В начале появляется всплывающее окно для чтения правил, а затем кнопка «начать игру». Все игровые действия происходят в рамках одного мобильного AR приложения Кибер-музея. Правила этой игры будут написаны и адаптированы, основываясь на правилах карточной игры «Бакуганы», широко известной среди детей нулевых годов 21 века.

В разработанном в будущем мобильном приложении с дополненной реальностью для Кибер-музея планируется реализация описанного функционала. В настоящее время ведется работа по проектированию и с логикой приложения. В дальнейшем будет полная реализация мобильного приложения и официальный релиз.

Ознакомившись со значением AR приложения для Кибер-музея, его концепцией и функциональными возможностями, которые планируется реализовать, можно приступать к разработке архитектуры мобильного приложения. Архитектура системы необходима при проектировании, так как она отображает ключевые элементы и их взаимодействия друг с другом. Составляющими архитектуры системы в проекте с AR приложением являются: WiFi роутер, смартфон с установленным мобильным приложением, движок Unity и SDK Vuforia. Архитектура системы приведена на рисунке 2.

Архитектура системы связывает смартфон и установленное в нем мобильное AR приложение с SDK Vuforia, которая, в свою очередь, взаимодействует с движком Unity. Между ними происходит обмен информацией по сети Internet. В результате все элементы системы передают данные и взаимодействуют друг с другом.

Таким образом, в рамках данной работы были проведены подготовительные мероприятия для разработки приложения с дополненной реальностью для Кибер-музея. Были изучены возможности применения AR технологии в Кибер-музее, проведен сравнительный анализ существующих AR приложений для музеев. На основе этого анализа были выявлены

основные критерии и решения для успешной разработки собственного AR приложения, описаны его функции и возможности, которые будут реализованы в дальнейшем. А также были выбраны необходимые инструменты для разработки приложения с дополненной реальностью и построена архитектура системы.

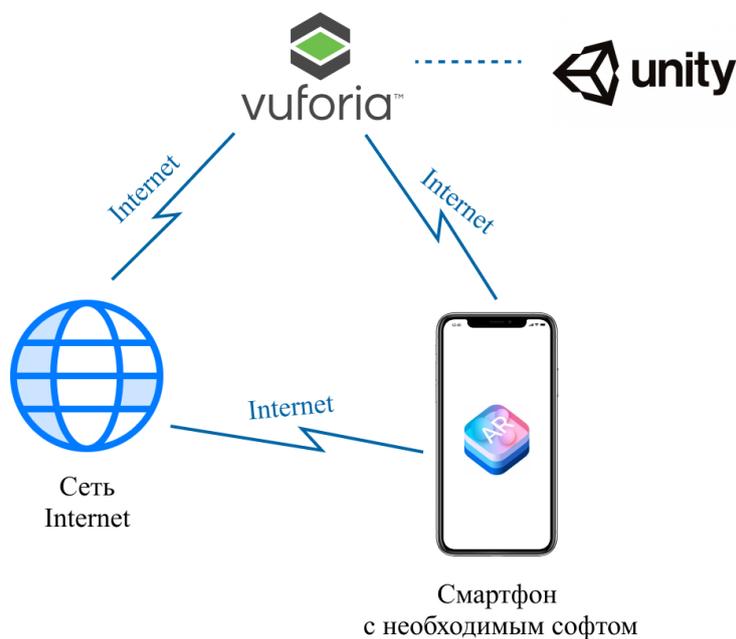


Рис. 2. Архитектура системы проекта с AR приложением для Кибер-музея

В дальнейшем предстоит еще много работы: планируется более детальное продумывание структуры приложения, написание сценария для AR квеста, правил для карточной коллекционной игры, создание Mind map, User Flow, разработка 3D моделей в программе Blender, сборка и компилирование проекта на движке Unity, работающего с плагином Vuforia, тестирование и отладка, и в результате запуск AR приложения для Кибер-музея.

Литература

1. Мэннинг Д., Батфилд-Эддисон П. Unity для разработчика. Мобильные мультиплатформенные игры. СПб: Питер, 2018. 304 с.
2. Перфилова Е.Р., Багдасарян Р.Х. Информационные технологии в музее: внедрение VR, AR и 3D технологий в современную музейную среду // Культура и время перемен. 2021. №4 (35).
3. Петрухина О.В. Дополненная реальность в пространстве музея // Месмахеровские чтения – 2019: Мат-лы международной научно-практической конференции (г. Санкт-Петербург, 21-22 марта 2019 г.). СПб.: СПГХПА им. А. Л. Штиглица, 2019. С. 122-125.

© Куприянова Т.В, Поспелова Н.В., 2023

УДК 004.056

Лазорин Д.С.

Российский государственный университет нефти и газа
(национальный исследовательский университет) им. И.М. Губкина
г. Москва, Россия

О ПОДХОДАХ К ОБЕСПЕЧЕНИЮ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ЦИФРОВЫХ ДВОЙНИКОВ

Процессы цифровизации всё больше оказывают влияние на развитие современных технологий. Происходит автоматизация: старое оборудование заменяется на более усовершенствованное. Это благотворно сказывается на экономической составляющей компаниях. Цифровой двойник является неотъемлемой частью каждого будущего производства. Его ключевая особенность заключается в непрерывном получении, обновлении и обработке данных, получаемых с реального объекта. Такой подход позволяет вносить различные корректировки, тем самым обеспечивая бесперебойную и безопасную работу целого технологического процесса. Получение информации происходит благодаря датчикам и сенсорам, которые устанавливаются на реальный объект. Передача информации выполняется по защищённому от взлома каналу связи в систему цифрового двойника, который в дальнейшем использует полученные данные в работе. Для обеспечения информационной безопасности требуются новые подходы и методы, которые необходимо совершенствовать не только с технологической стороны, но и с нормативной. Сегодня на производстве интегрированы автоматизированные системы управления технологическим процессом (АСУ ТП) [2]. Это комплексные системы, которые осуществляют автоматический сбор и обработку данных, тем самым оптимизируя управление технологическим объектом. Цифровой двойник является частью автоматизированной системы управления. Заводя разговор про нефтегазовую отрасль, большинство месторождений, действующих или находящихся на завершающем этапе эксплуатации, не оснащены даже АСУ ТП. Это тем или иным образом замедляет процессы добычи. Однако всё меняется: подход к добыче на новых месторождениях. Происходит внедрение не только АСУ ТП, но и цифровых двойников. Происходит обмен данными, следовательно, чрезвычайно важно обеспечить информационную безопасность технологической системы на производстве.

Нельзя отрицать, что цифровые двойники становятся всё более популярной технологией нового поколения на пути к информационной трансформации. Это происходит в результате сочетания физического оборудования и цифровой платформы, которая обеспечивает сбор данных датчиков и обработку в режиме реального времени, тем самым повышая эффективность и стратегическую важность компаний. Рассматривая топливно-энергетический комплекс России, цифровой двойник помогает выбирать наиболее продуктивные сценарии работы и проводить эксперименты в цифровой (виртуальной) среде, которые в реальной жизни могут быть связаны с тем или иным риском, например, повреждение оборудования, угроза жизни сотрудникам. Своевременное развитие данных технологий сможет привести к оптимизации процессов в существующих индустриальных

системах и инфраструктуре. Подобная тенденция приобретает интерес и спрос у существующих компаний. Привлечение новых инвестиций в технологию цифровых двойников поможет корпорациям быть более конкурентоспособными и завершить комплексным образом процессы цифровизации.

Цифровые технологии включают в себя различные составляющие. Первой составляющей является углубленная аналитика данных, включающая прогнозную аналитику, большие данные, а также интеллектуальный анализ данных на основе машинного обучения и искусственного интеллекта. Говоря про нефтегазовую отрасль, то в сфере разведки и добычи ископаемого топлива развитие новых цифровых технологий позволит реализации новых методов разведки нефти и газа. Вторая составляющая включает в себя виртуальную или дополненную реальность. Благодаря такому подходу компании смогут реализовывать комплексные экспертные системы, интерактивные электронные технические руководства, которые помогут отображать информацию о режимах работы оборудования и не только. Внедрение подобных технологий безусловно повысит производительность труда за счёт сокращения времени на выполнение операций, времени на их подготовку, оптимизирует работу персонала. Для этого чрезвычайно важным образом важны вычислительные мощности. Третья составляющая посвящена бизнес-процессам, их цифровизация позволит оптимально перераспределить рабочий коллектив по проектам, снизить количество ошибок и критических ситуаций, обеспечить прозрачность коммерческих решений. Четвёртая составляющая: облачные вычисления. Организация сетевого доступа по запросам к основному фонду конфигурируемых вычислительных ресурсов значительно упростит работу. К ним относятся серверы, базы данных, программное обеспечение. Пятая неотъемлемая составляющая, кибербезопасность. Обеспечение информационной безопасности каждой компоненты целостной системы важная задача для которой требуются современные подходы и методы. Это совокупность методов для защиты от атак злоумышленников на программное обеспечение, компьютеры, серверы, серверы, сети, системы. Целенаправленные атаки носят различный характер. Они могут быть направлены на получение доступа к конфиденциальным данным, нарушение функционирования инфраструктуры, изменение бизнес-процессов. В современном мире атаки на оборудование строятся таким образом, что управление системой остаётся возможным, но не оптимальным, что приводит к серьёзным экономическим потерям [3].

Примеры применения цифрового двойника на производстве. Это «цифровое месторождение» и «цифровой нефтеперерабатывающий завод». Реализация цифровых двойников основывается на передаче данных с определённых устройств в специализированное пространство (облачное) с дальнейшей их обработкой. Большинство облачных технологий являются импортными (<https://clck.ru/34pfqv>). Нынешние тенденции в Российской Федерации направлены на технологическую независимость, что подтверждает актуальность данной научной работы.

Структура цифрового двойника. Цифровой двойник может реализовываться на различных уровнях: компонентов, активов, систем и процессов. Его использование не только

в нефтегазовом секторе, но и других отраслях привело к существованию различий в типах цифровых двойников. Конкретно это зависит от сферы применения продукта. Существуют двойники компонентов, которые представляют копию конкретной части самого актива, что позволяет контролировать основные компоненты, напрямую влияющие на производительность системы. Близнецы активов. Актив состоит из множества компонентов, которые функционируют совместным образом. Он представляет информацию для исследования системы, улучшению отдельных её частей. Блок-близнецы. Особый тип цифрового двойника, охватывающий системный блок производственной цепочки, состоящий из нескольких активов. Это улучшает контроль и функциональность системы в целом. Двойники процессов. Реализуют целостное производственное предприятие, выявляют системные интеграции, которые влияют на общую производительность.

Моделирование цифрового двойника происходит в единой информационной среде (с общим доступом для сотрудников) и все данные взаимосвязаны. Например, процессы компании связаны с организационной структурой, процессами и информационной архитектурой (поскольку большая часть процессов выполняется с использованием существующих информационных систем). Системы управления информацией выдают текущую информацию для анализа и получают данные о доходах от цифровых двойников. На основе связки организационной структуры и бизнес-процессов система понимает, в каких процессах задействован сотрудник, какие функции на него возложены. Среднее производственное время выполнения функций, время простоя помогают системе рассчитать стоимость процесса. Сочетание систем управления компании позволяет автоматизировать обновление информации для всего процесса и создать совместную расширенную платформу. Любые изменения в компании и ее цифровая копия позволяют следить за всей нормативной документацией организации и подстраиваться под изменения стратегии компании. Связывание информационной архитектуры с другими цифровыми объектами дает возможность обмениваться информацией между программными продуктами, используемыми в компании, и сервисами связи.

Одним из ключевых элементов физического объекта является то, что способность восприятия для восприятия физического объекта должна быть улучшена из-за требований во многих сценариях приложений. В процессе восприятия физического объекта возникают некоторые проблемы, такие как неоднородные нескольких источников. В таких областях применения, как производство, крайне необходимы всеобъемлющие и спонтанные срабатывания. Датчики интегрируют необработанные данные в полевых условиях и в реальном времени в цифровой двойник. Он играет решающую роль в измерении различных физических параметров и обеспечении возможностей мониторинга, управления и поддержки принятия решений.

Важно обратить внимание, что нет опубликованного стандарта для физических объектов цифрового двойника, многие стандарты можно использовать повторно или ссылаться на них, существующие профили полевой шины, сопутствующие спецификации и другие спецификации, определяющие свойства устройств и компонентов, должны быть перенесены

в стандартизированные словари, а характеристики концептуальных активов, такие как документы планирования, должны быть включены в стандартизированные словари.

У цифровых двойников есть несколько возможных случаев, некоторые проблемы необходимо компенсировать для достижения целей и развертывания рассматриваемой технологии. Необходимо преодолеть несколько проблем. Правильность данных. Требуются подробные сведения о неисправностях машины. Для прогнозирования отказов необходимо насытить цифрового двойника данными о режимах отказов оборудования. Такие данные следует собирать за длительный период для наблюдения за процессом износа машин. Гибкость. Каждый раз при изменении конфигурации оборудования или состояния его элементов необходимо заново создавать цифрового двойника. Любая модификация, влияющая на работу оборудования, требует изменения его модели и лежащих в ее основе алгоритмов. Такие модификации на уровне машины (замена оригинальных деталей изготовленными по индивидуальному заказу) или на уровне завода (изменения политики эксплуатации) не всегда отражаются в заводских спецификациях и поэтому не могут быть точно смоделированы, что увеличивает риск ошибок. Квалификация специалистов. Для работы с современными технологиями должны быть сотрудники, хорошо разбирающиеся в системном моделировании, и аналитики, способные проводить грамотный анализ. Кроме того, корректная модель должна точно отражать все свойства физического объекта, включая параметры давления, температуры, а также электрические характеристики. Это требует участия руководителей предприятий, инженеров-технологов, инженеров-электриков, производителей оборудования, что усложняет развертывание.

Нефтегазовая отрасль характеризуется достаточно высокой степенью консерватизма. Культура и устоявшиеся подходы к управлению, сформировавшиеся за последние десятилетия в отрасли, определяют очень осторожное отношение к использованию новых технологий и инструментов. Особенно это касается инноваций, требующих необходимых изменений в системе управления предприятием.

Сложность поступающих решений приводит к высоким неопределенностям, уровню затратных рисков, опасности и сложности производства, а также человеческому фактору. Однако очевидно, что изменения в отрасли неизбежны, и чем раньше компании изменят свои подходы к стратегическому управлению, навыкам и технологиям, тем успешнее они будут на этом высококонкурентном рынке.

Теперь ключом к успеху является способность адаптироваться и увеличивать производство тех ресурсов, которые создают наибольшую ценность. Лидерами являются те компании, которые владеют цифровыми технологиями для эффективной непрерывной цепочки создания ценности. Этого нельзя отрицать.

Переход к разработке месторождений с особо сложными геологическими условиями залегания запасов, который часто имеет место, подталкивает к поиску и принятию новых решений при проектировании и строительстве технических сооружений. Обеспечение бесперебойной работы оборудования – одна из важнейших задач любого промышленного предприятия. Даже небольшая внеплановая остановка производственного комплекса,

вызванная выходом из строя оборудования, может привести к значительным финансовым затратам и экологическим проблемам. Сокращение потерь от простоя оборудования помогает компаниям оставаться конкурентоспособными на рынке. Однако важно не только устранить отказы оборудования, парализующие работу всего месторождения, но и создать организационную основу компании, способную стать основой для дальнейшего развития в сложных внешних условиях.

Сложность работы на удаленных объектах, дорогостоящие простои и поломки оборудования могут быть решены с помощью цифровых двойников. Сервисные аспекты, в частности нефтегазового сектора, делают тему исследования актуальной.

Основная ценность бизнеса заключается в обслуживании оборудования в значимых секторах, таких как энергетика и крупномасштабное производство, где повреждение небольшой детали приводит к полной остановке, например, производственной линии или всего насосного агрегата. Следовательно, это влечет за собой большие потери не только выпускаемых элементов, но и денежные потери.

В ранее проведенной научной работе нами были рассмотрены и проанализированы существующие подходы к обеспечению информационной безопасности цифровых двойников [1]. Выяснено, что использование современных технологий защиты информации в рамках актуальных для цифровых двойников моделей угроз и нарушителя нацелено на решение частных задач. Были проанализированы отечественные решения. Компания Infotecs – лидирующая Российская компания, реализующая решения в сфере информационной безопасности. Она предлагает решения VipNet Industrial Security: линейка продуктов, которая конструирует систему защиты АСУ ТП и систем межмашинного взаимодействия (<https://clck.ru/34pfs3>). Особенность заключается в том, что существуют решения как для промышленных протоколов, которые обеспечивают защиту каналов связи и сетевое экранирование, так и для Windows и Linux с специализированным программным обеспечением.

Таким образом, цифровые двойники – это новое решение для технологических систем. Необходимо обеспечить комплексный подход к обеспечению информационной безопасности цифровых двойников. Стоит обратить особое внимание на концепцию многоуровневой платформы, так как она включает в себя реализацию комплексного подхода, состоящего из низкоуровневых способов защиты канала связи при помощи криптографических алгоритмов до высокоуровневых задач.

Был реализован веб-сайт (<https://securedt.ru>) с полноценной и актуализирующийся информацией о научной работе, а также контактами для дальнейшего взаимодействия.

Литература

1. Лазорин Д.С., Правиков Д.И., Щербаков А.Ю. О подходах к обеспечению информационной безопасности цифровых двойников // Вестник современных цифровых технологий. 2022. № 11. С. 48-55.



2. Линьков В.А. Централизованные АСУ ТП. Состав и структура АСУ ТП // Международный студенческий научный вестник. 2019. № 6. С. 12. EDN ВНQBVO
3. Levaniuk D. Application of digital twins for maintenance in the gas and oil sector. Lappeenranta–Lahti University of Technology LUT, 2022. С. 7-37.

© Лазорин Д.С., 2023

МУЛЬТИЗАГРУЗОЧНЫЙ РЕАНИМАТОР И ЕГО УТИЛИТЫ

Мультizaгpyзочная флешка представляет собой набор дистрибутивов и утилит, с которой можно установить Windows или Linux, восстановить систему и сделать много других полезных вещей (<https://clck.ru/34pkKk>) [1; 2].

Программы для создания мультizaгpyзочной флешки

WinSetupFromUSB – одна из лучших утилит для быстрого создания мультizaгpyзочной USB-флешки с возможностью установки операционных систем Windows и Linux, создания встроенной виртуальной машины QEMU и различных ISO-образов. Простая и бесплатная, программа характеризуется стабильной работой и низкими требованиями к ресурсам компьютера (<https://clck.ru/34pkLn>).

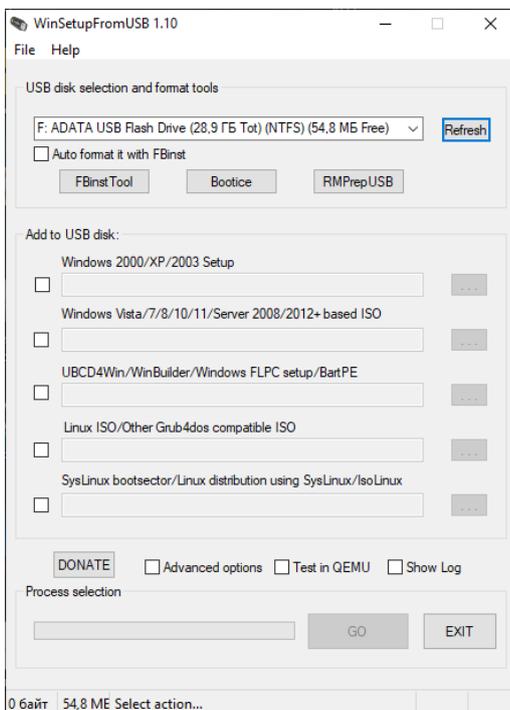


Рис. 1. Интерфейс программы WinSetupFromUSB

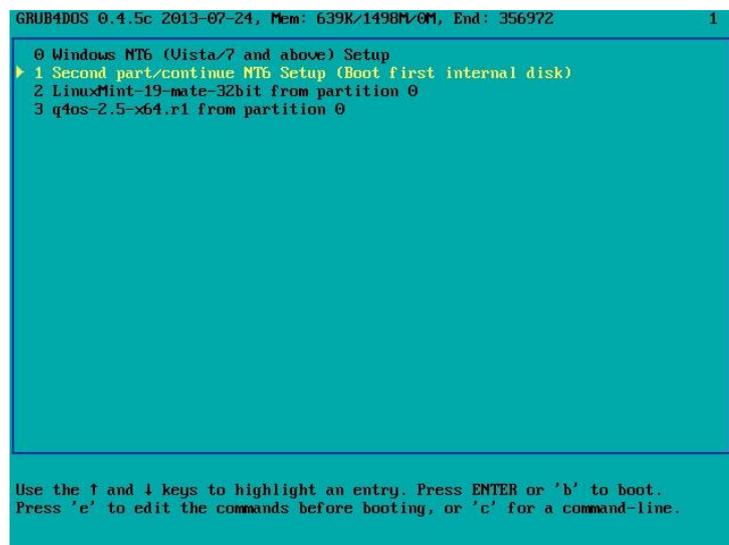


Рис. 2. Интерфейс меню GRUB4DOS

Ventoy – бесплатная и простая портативная утилита с открытым исходным кодом для создания мультizaгpyзочных USB-накопителей. Присутствует поддержка большинства операционных систем, протестирована работа более 620 образов загрузочных образов.

После создания флешки, можно скопировать на неё файлы различных загрузочных образов, без необходимости их распаковывать или как-либо подготавливать, и при загрузке компьютера с этой флешки выбрать из списка необходимый образ (<https://clck.ru/34pkMw>).

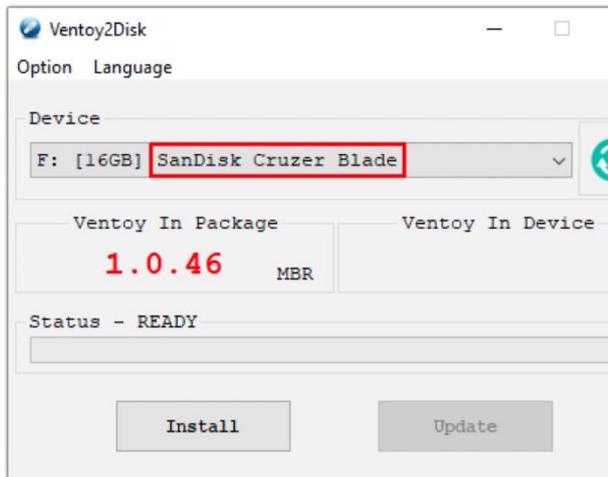


Рис. 3. Интерфейс программы Ventoy

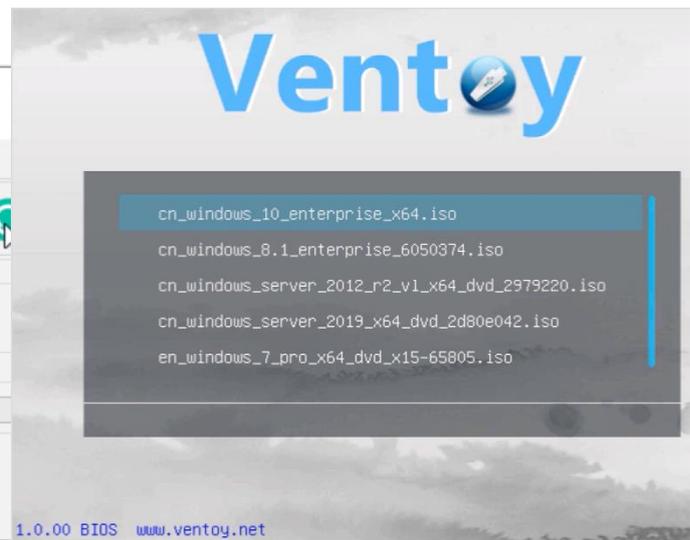


Рис. 4. Интерфейс меню Ventoy

Easy2Boot – инструмент для создания мультизагрузочных флешек, которые могут содержать в себе любой дистрибутив, LiveCD или образ дискеты. Easy2Boot работает в тесной связке с приложением для создания загрузочных накопителей RMPrepUSB, которое уже должно быть установлено на Вашем компьютере (<https://clck.ru/34pkNd>).

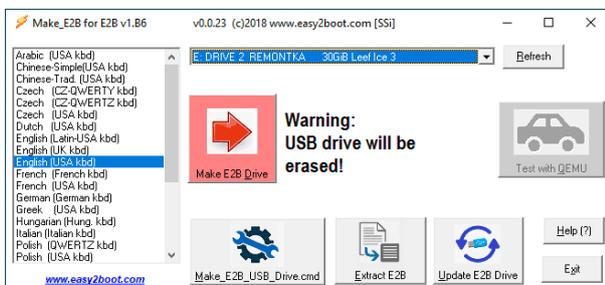


Рис. 5. Интерфейс программы Easy2Boot

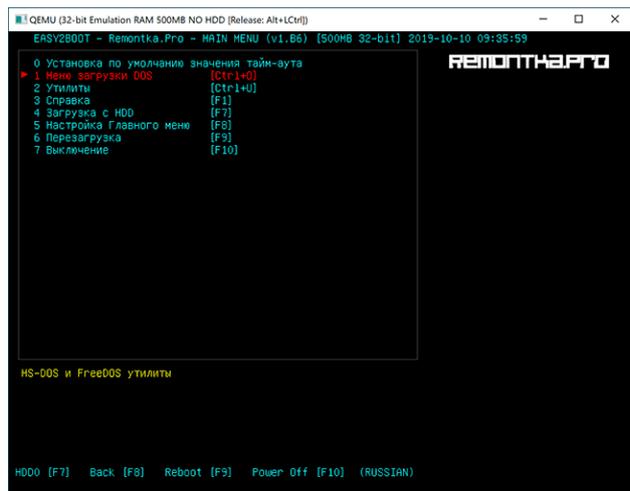


Рис. 6. Интерфейс меню Easy2Boot

Утилиты

Загрузочная флешка **MSDaRT** (Microsoft Diagnostic and Recovery Toolset) – это мощнейший инструмент для восстановления системы, с основными компонентами которой справится любой продвинутый пользователь персонального компьютера (<https://clck.ru/34pkPW>).

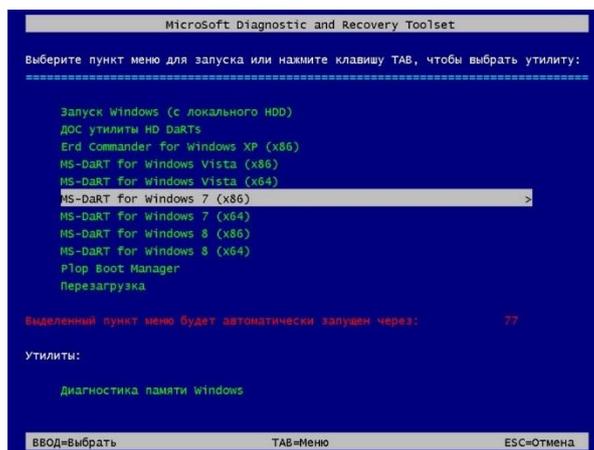


Рис. 7. Интерфейс меню MSDaRT

AdminPE – аварийный Live-диск на базе WinPE (урезанной версии обычной Windows), ориентированный на опытных пользователей и системных администраторов. Это, по сути, работающая с флешки или DVD-диска портативная операционная система с возможностями штатного функционала Windows и подборкой стороннего ПО для решения разного рода (<https://clck.ru/34pkQS>).

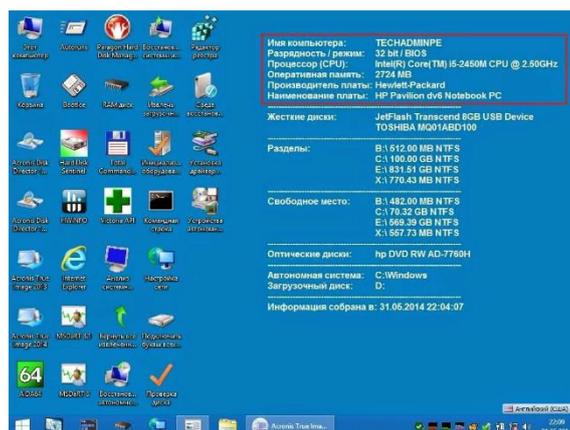


Рис. 8. Интерфейс меню AdminPE

WinPE 10-8 Sergei Strelec – аварийный Live-диск для случаев программного и аппаратного сбоя работы Windows-компьютеров.

Live-диск WinPE 10-8 Sergei Strelec – результат плодотворной работы харьковчанина Сергея Стрельца. Энтузиаст создал универсальное средство для диагностики, обслуживания компьютеров на базе Windows и их реанимации после программного или аппаратного сбоя (<https://clck.ru/34pkRF>).

Литература

1. Мийзамов А.А., Енин В.М., Матющенко И.А. Актуальные вопросы кибербезопасности // International Journal of Advanced Studies in Computer Engineering. 2021. №1. С. 17-21.
2. Самойдюк А. Безопасность и конфиденциальность: в чём разница?
<https://clck.ru/34pckW>

© Ларин Д.Е., 2023

УДК 004.031.42:004.77

Логвинец В.А.

Гомельский государственный университет им. Франциска Скорины
г. Гомель, Беларусь

РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ КОНТРОЛЯ ЗНАНИЙ

На сегодняшний день в сети Интернет существует множество различных специализированных инструментов, позволяющих строить эффективные системы управления как самими образовательными учреждениями, так и учебными материалами, что позволяет в разы увеличить качество современного образования. Более того, такие системы открывают целый ряд совершенно новых возможностей для своих пользователей, как, например, возможность дистанционного образования и контроля знаний, что является обоснованием актуальности создания подобных систем.

Основное внимание в автоматизированной системе контроля знаний (далее – АСКЗ) уделяется учреждениям, обеспечивающим получение общего среднего образования в Республике Беларусь. Это обусловлено обязательным характером общего среднего образования, наивысшим уровнем стандартизации условий организации и проведения образовательного процесса.

Целями создания АСКЗ являются:

1. Обеспечение прозрачности образовательного процесса и повышение объективности оценки обучающихся.
2. Частичное делегирование обязанностей педагогов автоматизированной системе.
3. Повышение качества образования за счёт его индивидуализации.
4. Поддержка принятия управленческих решений по совершенствованию образовательного процесса объективными и статистически значимыми данными.
5. Повышение эффективности управления в сфере образования за счет внедрения современных цифровых аналитических инструментов.
6. Облегчение образовательного процесса для всех его участников, путём проведения целого ряда автоматизаций.

Для создания системы были поставлены следующие задачи:

1. Анализ альтернативных решений из сферы онлайн-образования в Беларуси.
2. Формирование общей характеристики целевой аудитории и отдельных её групп.
3. Постановка задачи и формирование базовых функциональных возможностей.
4. Выбор технологического стека.
5. Декомпозиция общего процесса разработки.
6. Структурирование данных и выбор подходящей базы данных.
7. UX/UI-проектирование.

При разработке АСКЗ должна быть обеспечена сохранность информации, защита её от несанкционированного доступа. Внедрение системы в информационную структуру учреждений общего среднего образования позволит совершенствовать процесс обучения,

выстраивать индивидуальную образовательную траекторию учащихся и оптимизировать работу педагогов и администрации.

Среди основных альтернативных решений, действующих на территории Республики Беларусь, можно выделить систему электронного обучения Moodle и онлайн-сервис Schools.by. Moodle или Modular Object-Oriented Dynamic Learning Environment – международная бесплатная система электронного обучения, через которую можно дистанционно развивать и тестировать учащихся со всего мира.

На сегодняшний день Moodle является одной из самых популярных систем электронного обучения. Она переведена более чем на 100 языков. В основном, ей пользуются учреждения высшего образования (<https://www.ispring.ru/elearning-insights/moodle>). В свою очередь, Schools.by – единая образовательная платформа в Республике Беларусь, предлагающая бесплатный сервис электронных журналов и дневников, выводя отношения между школой и семьёй на высокотехнологичный уровень. У Schools.by более 2 миллионов участников образовательного процесса из 3990 учреждений по всей стране, 1 место по посещаемости среди образовательных порталов и топ 10 Интернет-ресурсов Беларуси по версии рейтинга Alexa Rank (<https://clck.ru/34pm4W>).

Несмотря на то, что Moodle и Schools.by являются решениями в области образования, они достаточно сильно между собой различаются. Например, система Moodle не реализует возможность ведения электронных журналов и дневников, отправки SMS-оповещений о пропущенных занятиях, информирования о предстоящей контрольной или самостоятельной работе, создания страницы учреждения образования в сети Интернет с последующей поддержкой ведения новостной ленты и пр. Несмотря на это, Moodle, в отличие от Schools.by, позволяет удобно создавать и проводить онлайн-тесты и -курсы. Такие различия обусловлены в первую очередь тем, что система Moodle имеет более широкий спектр применения, абстрагируясь от территориальной или структурной зависимости какого-либо образовательного учреждения, в то время как сервис Schools.by сконцентрирован исключительно на учреждениях среднего образования. При разработке АСКЗ следует учесть весь накопившийся позитивный и негативный опыт упомянутых альтернативных решений.

Чтобы найти востребованную уникальность проекта на рынке и определить потенциальных пользователей, необходимо составить общую характеристику целевой аудитории, а также описать отдельные её группы. Проект АСКЗ сконцентрирован на белорусской системе среднего образования и может быть внедрён в учреждения только этого уровня, такие как, например, школы, гимназии и лицеи. Разработка платформы должна вестись с учётом интересов и пользовательского опыта всех потенциальных групп целевой аудитории. Проект насчитывает четыре потенциальных групп: учащиеся (около 40%, мальчики и девочки до 18 лет); учителя (около 30%, преимущественно женщины от 20 до 50 лет); родители (около 20%, преимущественно женщины от 20 до 50 лет); администрация (около 10%, мужчины и женщины от 25 до 50 лет).

Постановка задачи.

Постановка задачи и формирование базовых функциональных возможностей выполняется на основании составленной характеристики целевой аудитории и интересов отдельных её групп. Так как проект ориентирован сразу на четыре группы пользователей, должны быть реализованы роли с различными уровнями доступа, при чём у одного пользователя может быть сразу несколько ролей. Таким образом, должно быть разработано интерактивное Full-Stack Web-приложение с авторизацией, различными уровнями доступа, реализующее следующие базовые функциональные возможности:

1. Создание и проведение онлайн-тестов, викторин, презентаций.
2. Сбор и предоставление сводной аналитической информации.

Процесс выбора технологического стека должен опираться как на интересы групп целевой аудитории, так и на базовые функциональные возможности разрабатываемого приложения. В проекте АСКЗ Front-End сервер будет разработан с использованием Web-фреймворка Next.js на базе библиотеки React.js. React обладает собственным виртуальным DOM, который управляет фактическим DOM браузера и, так как он намного быстрее, чем DOM браузера, значительно повышает производительность [1, с. 25]. React «из коробки» использует рендеринг на стороне клиента (CSR), что увеличивает время первоначальной загрузки сайта и негативно влияет на SEO. Проблема начальной скорости загрузки в CSR может быть решена с помощью рендеринга на стороне сервера (SSR), при котором сервер извлекает информацию из базы данных и отправляет на страницу подготовленный HTML-файл [3, с. 72]. За счёт SSR Next.js значительно увеличивает скорость отклика Front-End сервера и даёт возможность создавать адаптированные под индексацию поисковыми роботами (SEO-friendly) страницы, что позволит легче находить ресурсы АСКЗ в поисковой выдаче браузера.

Для максимального охвата пользователей будет также использована технология PWA (Progressive Web Application), позволяющая устанавливать Web-приложение из браузера как нативное мобильное или десктопное приложение непосредственно на устройство пользователя. С использованием PWA открывается возможность создания для приложения специального скрипта Service Worker. Ключевой особенностью технологии Service Worker является, то, что этот скрипт может перехватывать и модифицировать запросы навигации и ресурсов, гибко кэшировать ресурсы для обеспечения полного контроля над поведением приложения в определенных ситуациях (например, когда сеть не доступна) [2, с. 622].

В свою очередь Back-End приложения будет использовать серверный Web-фреймворк Nest.js на базе Express.js и Node.js. Данный фреймворк позволяет эффективно организовать серверный код, в отличие от более низкоуровневого фреймворка Express.js, на котором он основывается. Клиент-серверная архитектура АСКЗ также предполагает реализацию сервером универсального REST API с соблюдением архитектурного стиля CRUD (акроним от Create Read Update Delete), способного принимать POST, GET, PUT и DELETE запросы. В свою очередь, ответ, ожидаемый от сервера по любому из запросов, всегда должен быть в формате JSON, за исключением случаев запроса медиа-ресурсов (картинок, видео и пр.). Некоторый

функционал приложения нуждается в соединении с сервером в режиме реального времени, как, например, онлайн-тесты, -викторины и –презентации. В отличие от обычного REST API, работающего по HTTP, в таких случаях лучше всего использовать технологию WebSocket, протокол связи поверх TCP-соединения, предназначенный для обмена сообщениями между браузером и Web-сервером в режиме реального времени, используя постоянное соединение. И Front-End, и Back-End приложения будут создаваться с использованием технологии TypeScript, строго типизированного языка программирования, основанного на JavaScript. В крупных проектах использование строгой типизации является практически жизненной необходимостью, так как такие решения позволяют выявить огромное количество потенциальных проблем ещё на этапе разработки.

Разработка проекта.

Для отображения процесса разработки удобно воспользоваться построением графической нотации IDEF0, методологии функционального моделирования, предназначенной для формализации и описания бизнес-процессов. В случае с АСКЗ (рис. 1), управляющими элементами всего процесса разработки являются принципы систематизации информации, документация Nest.js, примеры UI-дизайна из Dribbble и Behance, документация Next.js и React.js, механизмами же являются локальный сервер, Nest.js, Figma, Next.js и React.js, на вход поступают цели и задачи проекта, а на выходе ожидается готовое Web-приложение.

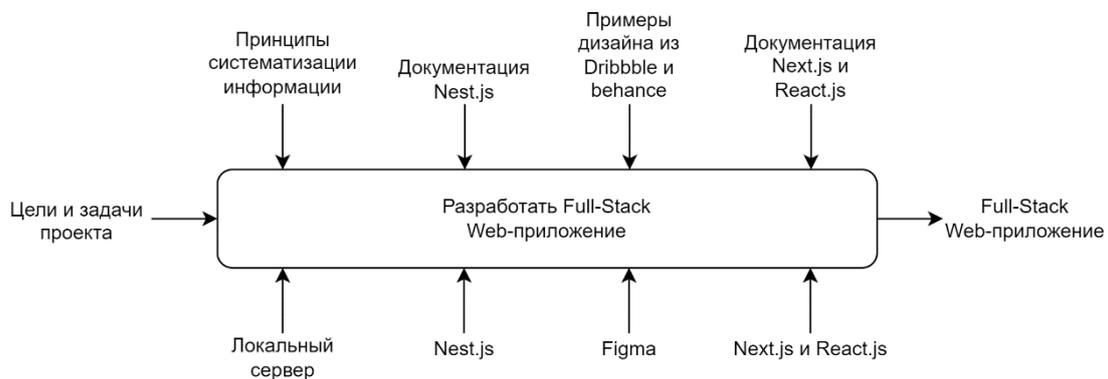


Рис. 1. Общий процесс разработки

Проведя декомпозицию общего процесса разработки получим 5 подпроцессов (рис. 2): разработка структуры базы данных; разработка Back-End сервера; разработка прототипа UI; разработка Front-End сервера; компоновка и тестирование.

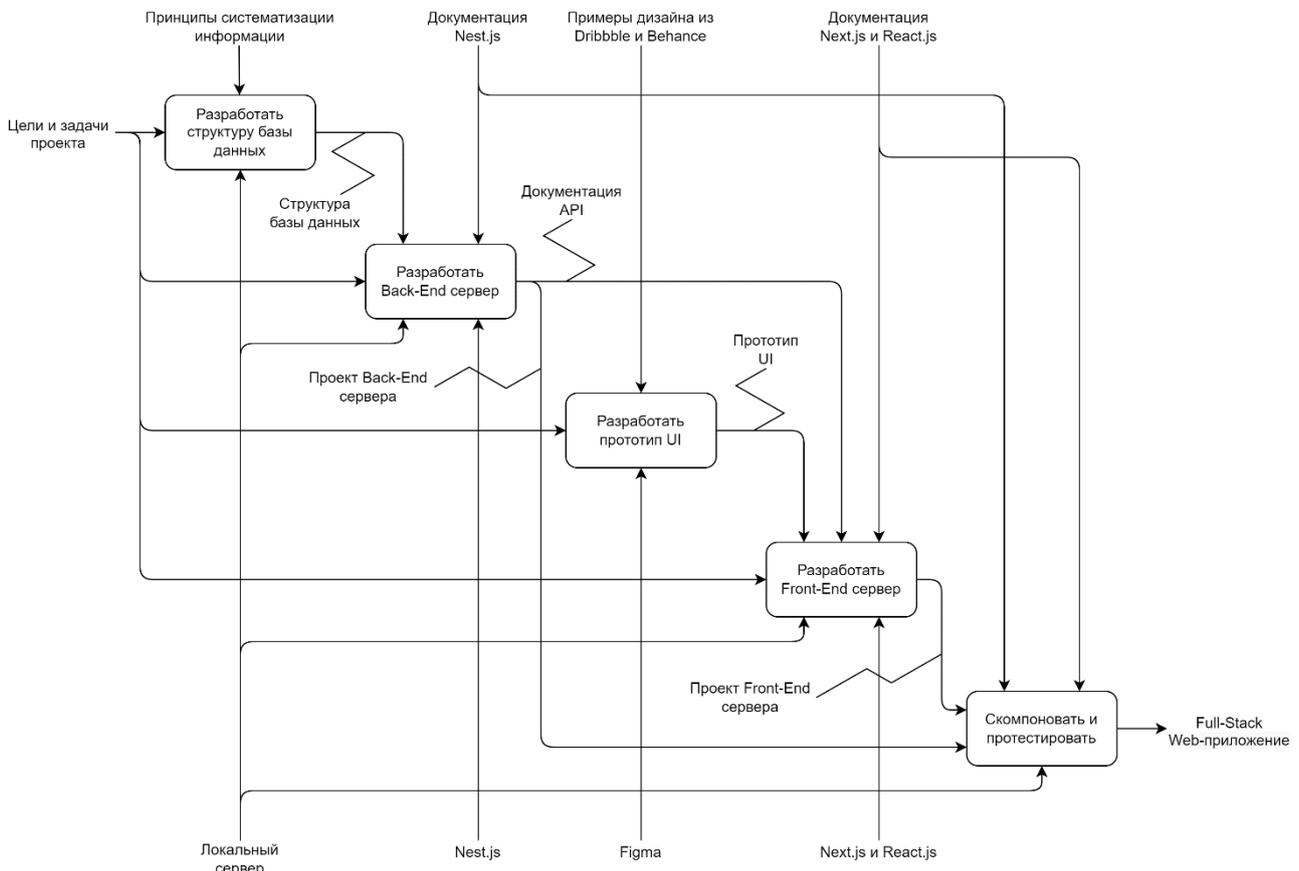


Рис. 2. Декомпозиция процесса разработки

Так как в АСКЗ несколько клиентов могут иметь доступ к одним и тем же актуальным данным, их хранение должно быть вынесено в единую базу данных на стороне сервера. Неплохим решением для подобного рода проектов может послужить Firebase, платформа для разработки приложений на Unity, Android, iOS и Web-приложений от компании Google, в которой есть самые современные функции для разработки, перекомпоновки, тестирования, аналитики и улучшения качества приложений. Фактически, Firebase выступает в роле уже готового Back-End сервера для решения большинства типичных задач. В АСКЗ будет активно использоваться сервис Firebase Realtime Database, бесплатно предоставляющий документно-ориентированную NoSQL базу данных размером в 1 ГБ. Подобные базы данных специально созданы для определения максимально гибких моделей данных. В связи с простотой разработки, функциональностью и производительностью при любых масштабах базы данных NoSQL получили широкое распространение и имеют обширное применение. В АСКЗ предполагается наличие множества различных сущностей. Среди них, одними из ключевых являются Resident (учреждение образования) и User (пользователь). На их примере можно рассмотреть общий принцип работы и взаимодействия сущностей в базе данных. В СУБД Firebase Realtime Database все сущности представлены в формате JSON. Однако, для лучшего понимания, их можно привести к SQL-представлению, выделив сущности не как ключи иерархического дерева, а как отдельные таблицы (рис. 3).

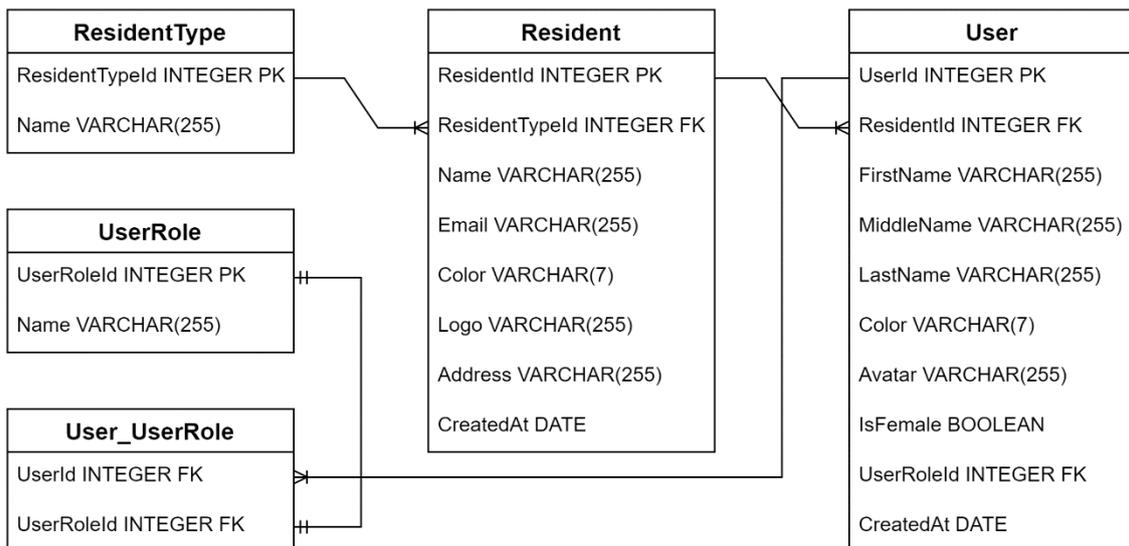


Рис. 3. SQL-представление базы данных

В проекте АСКЗ будет задействовано три базовых шаблона проектирования: Singleton, Builder и Observable. Паттерн Singleton будет использоваться при написании класса «Firebase», так как этот класс будет отвечать за взаимодействие бекенд сервера и базы данных Firebase. Таким образом, программа не будет создавать дополнительные объекты в памяти и создаст единую точку доступа к удалённой базе данных во всём программном комплексе.

```

import ...
export default class Firebase {
private static instance: Firebase;
public static getInstance(): Firebase {
if(Firebase.instance) return Firebase.instance;
return Firebase.instance = new Firebase();
}
...
}
    
```

Шаблон Builder (Строитель) – это порождающий паттерн проектирования, который позволяет создавать сложные объекты пошагово. Строитель даёт возможность использовать один и тот же код строительства для получения разных представлений объектов. Отличительной чертой класса на базе этого паттерна является то, что методы этого класса возвращают ссылку на экземпляр этого же класса, что в дальнейшем позволяет последовательно вызывать эти методы в коде. С использованием данного паттерна открывается возможность удобно создавать экземпляр класса «Dialog» нужной «комлектации», используя цепочку вызова методов.

```

import ...
type Body = JSX | string;
export default class Dialog {
private title: string = "";
    
```

```
private body: Body = null;  
public setTitle(title: string): Dialog { this.title = title; }  
public setBody(body: Body): Dialog { this.body = body; }  
...  
}
```

Шаблон Observable (Наблюдатель) – это поведенческий паттерн проектирования, который создаёт механизм подписки, позволяющий одним объектам следить и реагировать на события, происходящие в других объектах. Класс на базе такого шаблона ведёт учёт «подписчиков» на какое-либо событие. В случае, если это событие произойдёт, класс-обозреватель уведомит об этом всех своих подписчиков. У подписчиков есть возможность отписаться от обозревателя в любое время. На базе этого шаблона проектирования можно реализовать класс «Theme» на клиентской стороне. Любой компонент в React сможет подписаться на событие изменения темы в приложении (например, смена тёмной темы на светлую), чтобы предпринять какие-либо действия.

```
import ...  
type Subscriber = (newThemeMode: string) => void;  
export default class Theme {  
private static subscribers: Array<Subscriber> = [];  
private static broadcast(themeMode: string): void {  
Theme.subscribers.forEach(sub => sub(themeMode));  
}  
public static subscribe(fn: Subscriber) {  
Theme.subscribers.push(fn);  
}  
public static unsubscribe(fn: Subscriber) {  
Theme.subscribers = Theme.subscribers.filter(sub => sub !== fn);  
}  
...  
}
```

В данной работе было проведено исследование и описана разработка универсального интерактивного образовательного Web-приложения. Данное приложение позволит автоматизировать образовательный процесс в рамках дистанционного образования, внесёт дополнительные возможности для проведения очных занятий. Формат Web-приложения позволяет добиться максимального уровня кроссплатформенности и единого пользовательского опыта у всех групп целевой аудитории. Для реализации использованы такие веб-технологии, как Next.js, React.js, Nest.js, Express.js, Node.js, а также облачные сервисы Firebase, Google Cloud и Vercel. Кроссплатформенность достигнута посредством выбора исключительно веб-технологий, работающих по протоколу HTTP. Полный или частичный офлайн доступ достигнут за счёт использования проксирующих веб-технологий,

таких как Progressive Web Application и Service Worker. Для хранения и обработки данных в реальном времени выбран сервис Firebase и база данных Firebase Realtime Database. Асинхронная обработка запросов бекенд-сервером достигнута за счёт использования в проекте серверного веб-фреймворка Nest.js на базе Node.js, который не блокирует поток за счёт реализованного Event Loop под капотом.

Литература

1. Байдыбенко А.А. Современные фреймворки для разработки web-приложений // Интеллектуальные технологии на транспорте. 2020. №4. С. 23-28.
2. Газизуллин Н.И. Разработка прогрессивного веб-приложения с помощью технологии PWA // Научно-образовательный журнал для студентов и преподавателей «StudNet». 2020. №8. С. 622-625.
3. Князев И.В. Анализ работы приложения с использованием Server-Side Rendering: миграция, настройка и развёртывание приложения Next.js // Sciences of Europe. 2021. №76. С. 71-74.

© Логвинец В.А., 2023

УДК 004.031.42:004.77

Мироненко Е.В.

Гомельский государственный университет им. Франциска Скорины
г. Гомель, Беларусь

РАЗРАБОТКА ОНЛАЙН-ПЛАТФОРМЫ ДЛЯ ОБМЕНА ИЗОБРАЖЕНИЯМИ

Создание и поддержка социальных сетей, в частности онлайн-галерей, является актуальной и востребованной задачей на рынке веб-приложений. Несомненно, виртуальный мир быстро развивается, приобретая множество функций, которые заменяют функции реального мира. Эта тенденция коснулась и области изобразительного искусства, вектор развития которого стремительно переходит в «оффлайновое пространство».

Существует множество вариантов платформ социальных сетей, а количество творческих приложений для художников и дизайнеров растет с каждым днем. Приложения, предназначенные специально для художников и дизайнеров, упрощают представление художественных работ нужной аудитории и, в конечном счете, позволяют продавать больше произведений искусства в Интернете.

Данный интернет-портал будет позволять пользователям добавлять изображения в режиме онлайн, осуществлять поиск, классифицировать их по тематическим коллекциям, а также предоставлять возможность делиться ими с другими пользователями.

На сегодняшний день на территории Республики Беларусь для подобных целей наибольшей популярностью пользуются зарубежные сервисы ArtStation, DeviantArt и Pinterest. Каждая из платформ имеет свои преимущества и ориентирована на различные целевые аудитории.

ArtStation пользуется популярностью у профессиональных цифровых художников, является признанной площадкой для размещения портфолио. Учитывая ориентированность на определенную целевую аудиторию, трафик данного сервиса не так высок.

DeviantArt более популярен, т. к. имеет более низкий порог вхождения и нацелен на более юную аудиторию любителей, рисующих на вольную тему.

Pinterest имеет наиболее упрощенный функционал и удобный пользовательский интерфейс. Сервис крайне прост в использовании, что обуславливает наибольшую популярность. Главной особенностью является использование ИИ в механизме поиска по изображениям, что обуславливает достаточно гибкую ленту предлагаемого контента.

ArtStation имеет преимущество перед DeviantArt благодаря своему профессиональному пользовательскому интерфейсу, который делает его удобным как для художников, так и для обычных любителей искусства. Также, в ArtStation есть возможность создать свое собственное портфолио, подписываться на других пользователей даже без предварительного создания учетной записи. В DeviantArt и Pinterest необходимо создать учетную запись, прежде чем подписываться на кого-либо.

В рамках разработки данной онлайн-платформы были учтены ключевые преимущества каждой из платформ, как на уровне архитектуры.

Для реализации проекта определены следующие задачи:

1. Исследовать актуальность разработки проекта, провести обзор выбранной целевой аудитории и анализ сферы информационных технологий на предмет альтернативных решений.

2. Выбрать функционал и определить техническую составляющую задачи веб-приложения.

3. Провести декомпозицию процесса разработки.

4. Выбрать стек используемых технологий веб-приложения.

5. Структурировать базу данных веб-приложения.

6. Разработать дизайн и элементы бренда для пользовательского интерфейса.

Также были выделены основные функции для реализации веб-приложения:

- аутентификация пользователя;
- возможность базовой настройки аккаунта пользователя;
- просмотр публикуемых изображений в общей ленте;
- публикация изображений с заданными параметрами;
- добавление изображений в пользовательские коллекции;
- сортировка изображений в коллекциях;
- возможность добавления комментариев;
- поиск изображений по ключевым словам;
- возможность редактирования своих публикаций.

В рамках проекта были выделены три основные целевые аудитории: художники/дизайнеры, обыватели, рекрутеры. Целевая аудитория художников является основополагающей, но не преобладающей (порядка 30%). Основной же процент трафика занимает группа обычных пользователей: потребителей контента (около 60%), арт-рекрутеры и HR-менеджеры (порядка 10%).

Данный интернет-портал будет позволять пользователям добавлять изображения в режиме онлайн, осуществлять поиск по ним, классифицировать их по тематическим коллекциям, а также предоставлять возможность делиться ими с другими пользователями.

Для выполнения подобных задач веб-приложение будет построено по принципу клиент-серверной архитектуры. Она описывает, как происходит работа с пользователями, где хранятся данные и как обеспечивается их защита.

В клиент-серверной архитектуре используется три компонента:

1. Клиент – программа на стороне виртуального пользователя, которая запрашивает сервер о предоставлении данных или возможности выполнить определенные действия системы. Ее назначение – дать пользователю удобный способ взаимодействия с сервером;

2. Сервер – компьютер, на котором хранится сайт или приложение. Основная задача сервера – бесперебойная работа и возможность обрабатывать множество запросов от пользователей. Сервер предназначен для решения ряда задач в ходе выполнения программного кода, выполняет определенные действия в ответ на запросы клиентов, обеспечивает пользователям доступ к определенным данным и базам данных;

3. База данных – программа, в которой хранятся все данные приложения.

Если сервер выполняет функции приложения и базы данных, то такая архитектура называется двухуровневой. Такой подход используют для небольших приложений, где нет большого количества клиентов. Хоть такой способ и проще, но его надежность небольшая.

Для решения этой проблемы безопасности в моем веб-приложении была использована база данных. Она хранится отдельно от сервера. Сервер в этом случае выполняет роль логической машины, которая обрабатывает данные, но не хранит их.

В данном проекте будут использованы фронтенд и бэкенд-сервер. Для фронтенд использован web-фреймворк Next.JS, который является надстройкой над React.JS. React.JS обеспечивает выдающееся быстродействие благодаря своей виртуальной модели DOM и алгоритму интеллектуального согласования (smart-reconciliation algorithm). Одно из побочных преимуществ – возможность тестирования без запуска браузера с отсутствующим графическим интерфейсом [3].

Для бэкенд-сервера используется web-фреймворк Express.JS. Обе составляющие будут реализованы при помощи строго типизированного языка программирования TypeScript, основанном на JavaScript (рис. 1). JavaScript применяет «утиную» типизацию, а TypeScript ее моделирует при помощи структурной типизации. В связи с этим, значения, присваиваемые интерфейсам, могут иметь свойства, не указанные в объявленных типах [2].

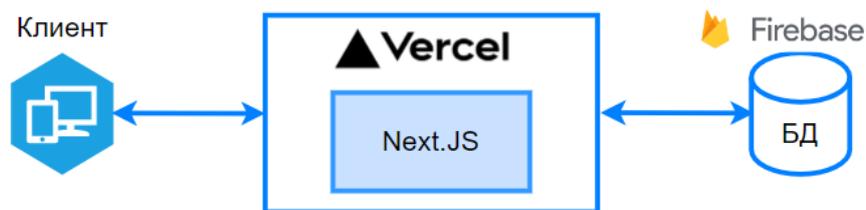


Рис. 1. Схема клиент-серверной архитектуры приложения

Декомпозиция процесса разработки будет проведена по принципу методологии IDEF0. Функциональная модель IDEF0 – это блоки, каждый из которых имеет несколько входов и выходов. В каждом блоке есть управление и механизмы, детализирующиеся до необходимого уровня. В левом верхнем углу расположена самая важная функция. Она соединяется с остальными стрелками и описаниями функциональных блоков. У каждой стрелки или активности есть свое значение. Благодаря этому такая модель используется для описания любых административных и организационных процессов.

На рисунке 2 приводится общая IDEF0 диаграмма. Главная цель проекта – веб-приложение. Для этого определяем главный блок «Разработать веб-приложения», его входные данные (Техническое задание) и выходные данные (Веб-приложение).

Далее были определены этапы реализации проекта, представленные на рисунке 3. Главный блок подразделяется на подпроцессы: «Разработать структуру базы данных», «Разработать бэкенд-сервер», «Разработать прототип интерфейса», «Разработать фронтенд-сервер» и «Скомпоновать и протестировать».

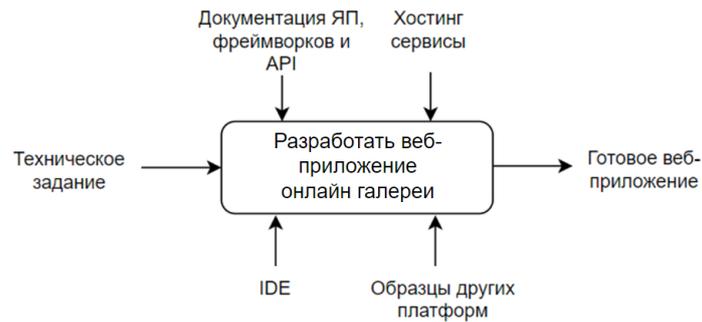


Рис. 2. IDEF0 схема проекта

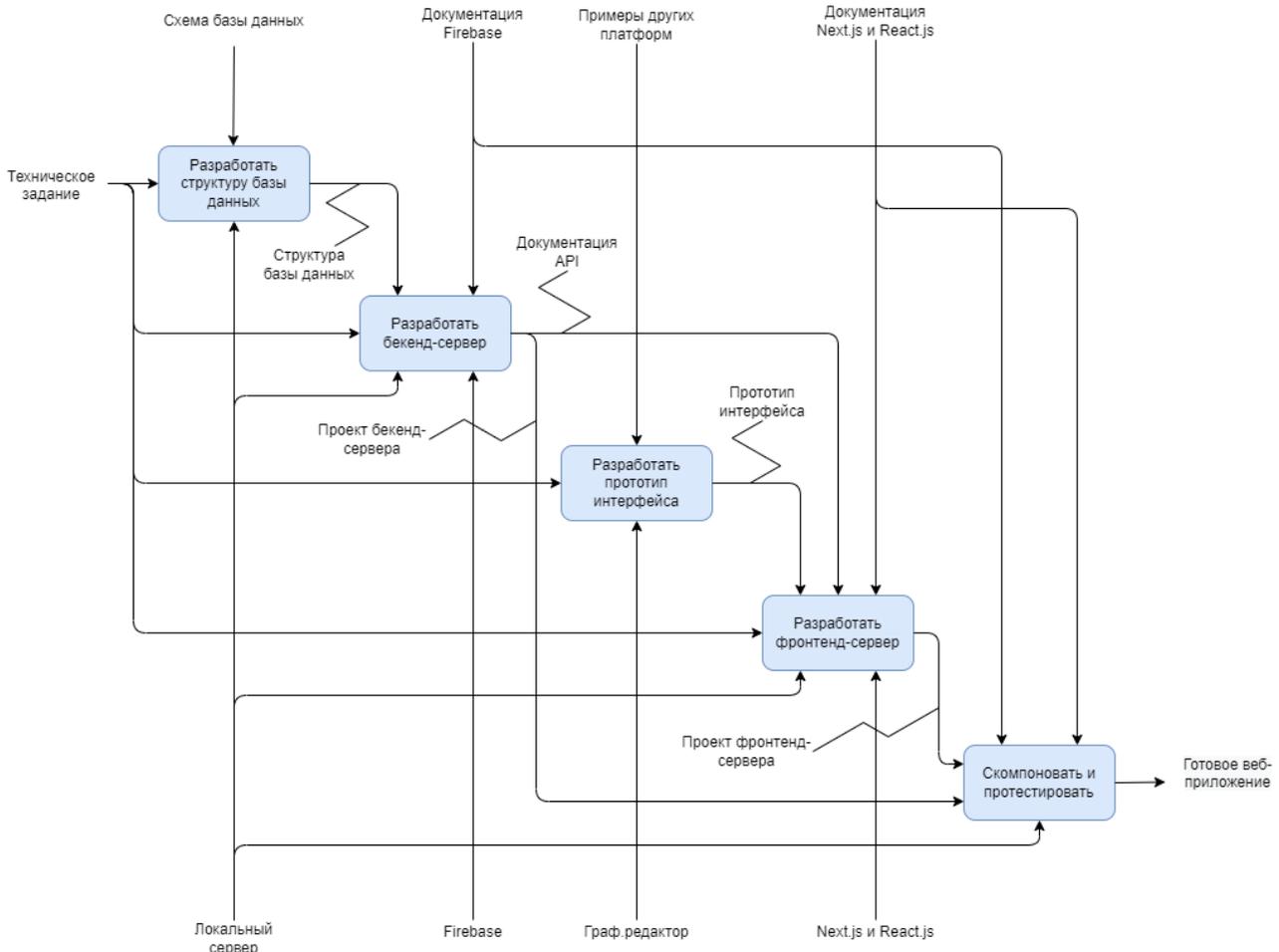


Рис. 3. Детализированная IDEF0 схема проекта

Блок «Разработать структуру базы данных» включает создание непосредственной структуры базы данных, которая будет функционировать на сервисе Firebase, определение сущностей, типов данных, а также общей структуры базы данных проекта. Входными данными блока является схема базы данных проекта, локальный сервер, а выходным – структура базы данных.

Блок «Разработать бэкенд-сервер» содержит разработку бэкенд-сервера проекта, являющийся связующим звеном базы данных и фронтенд-сервера. Входными данными блока является документация Firebase, структура базы данных, локальный сервер, техническое задание, а выходными – документация API и проект бэкенд-сервера.

Блок «Разработать прототип интерфейса» предполагает разработку UX/UI проекта, опираясь на образцы платформ схожей тематики и назначения. Входными данными блока являются примеры других платформ, графический редактор, техническое задание, а выходным – прототип интерфейса.

Блок «Разработать фронтенд-сервер» включает разработку сервера, обслуживающего фронтенд составляющую проекта, для написания которого будут использованы фреймворки React.JS и Next.JS. Входными данными блока являются документация React.JS и Next.JS., локальный сервер, техническое задание, документация API, а также прототип интерфейса, выходными – проект фронтенд-сервера.

Заключительный блок «Скомпоновать и протестировать» содержит в себя финальные отладку, доводку и тестирование разработанного комплекса приложений. Входными данными блока являются проект фронтенд-сервера, документация Firebase, документация React.JS и Next.JS., локальный сервер, проект бекенд-сервера, а выходным – готовое веб-приложение.

Для хранения данных приложения будет использоваться сервис Firebase. Firebase предлагает два облачных решения для баз данных, доступных для клиентов, которые поддерживают синхронизацию данных в реальном времени.

Cloud Firestore – это новейшая база данных Firebase для разработки мобильных приложений. Он основан на успехах базы данных реального времени с новой, более интуитивно понятной моделью данных. Cloud Firestore также предлагает более богатые, быстрые запросы и масштабируется дальше, чем база данных реального времени.

Realtime Database – это исходная база данных Firebase. Это эффективное решение с малой задержкой для мобильных приложений, которым требуется синхронизация состояний между клиентами в режиме реального времени. Firestore и Firebase Realtime Database предлагают NoSQL и имеют следующие возможности:

- 1 Пакеты SDK, ориентированные на клиента, без серверов для развертывания и обслуживания.
- 2 Обновления в реальном времени.
- 3 Бесплатный тариф, а далее оплата того, что используется.

В силу простоты использования и развертывания для реализации проекта будет использоваться Realtime Database. База данных хранится в формате JSON.

Следующий этап разработки представляет из себя разработку UX/UI-интерфейсов. Проектирование пользовательского интерфейса – это создание тестовой версии приложения. Это начальный этап разработки пользовательского интерфейса, когда распределяются функции приложения по экранам, определяются макеты экранов, содержимое, элементы управления и их поведение [1, с. 4].

UX/UI-дизайн – это проектирование любых пользовательских интерфейсов, в которых удобство использования так же важно, как и внешний вид. UX и UI неразрывно связаны между собой, и являются неотъемлемой частью успешного цифрового продукта. Визуально привлекательный и удобный пользовательский интерфейс – ключевой показатель качества сайта. В сочетании с грамотной структурой и логичной навигацией по разделам ресурса, он

привлекает посетителей и улучшает функциональность сайта. Главная задача в такой работе, как проектирование web-интерфейсов, – максимально упростить жизнь пользователю, сделать так, чтобы он достигал желаемый результат, затрачивая минимум усилий [3, с. 5].

В качестве основного цвета веб-приложения был выбран синий цвет с использованием контрастирующих тонов, темного и светлого. При этом семантически связаны общей фигурой, полигональной текстурой фона, цветами и стилем.

Проект получил название «Focal Point», что дословно можно перевести как «Центр кристаллизации», «Точка координации». Это соотносится со назначением и функционалом проекта, подобно кристаллу, приложение позволяет объединять в одном месте разнообразные художественные предпочтения пользователя. Также это можно интерпретировать как многогранность пользователей и самого сайта. Самая близкая к зрителю пустая грань олицетворяет потенциальное место для творчества пользователя. Логотип проекта изображён на рисунке 4 в виде лаконичного черного многогранника с одной светлой гранью.



Рис. 4. Название и логотип проекта

В ходе разработки дизайна и бренда, были подобраны первичные тематика, стиль и оформление веб-проекта. Полигональные формы, а также контраст дополнительных черного и белого цветов зададут общую строгость, концептуальность и современность проекта. Яркий синий цвет символизирует идеи лояльности, стабильности, безопасности и авторитета. Многие компании и бренды используют оттенки синего в своих логотипах и мессенджерах именно из-за этих ассоциаций.

В проекте было задействовано два шаблона проектирования: Singleton и Factory. Несмотря на их простоту, они часто используются в самых разнообразных проектах. В качестве шаблонов, которые не были использованы, выступают Singleton и Builder.

В шаблоне Factory создается объект, но при этом клиенту не открывается логика его создания. К вновь созданному объекту можно обратиться с помощью общего. Данный шаблон проектирования используется повсеместно на всех этапах создания проекта.

Singleton – порождающий шаблон проектирования, гарантирующий, что в однопоточном приложении будет единственный экземпляр некоторого класса, и предоставляющий глобальную точку доступа к этому экземпляру.

В данном проекте этот шаблон реализует класс пакета Firebase, а именно FirebaseDatabase, который является входной точкой в базу данных. Отличительной чертой такого класса является то, что он создаёт в себе статический экземпляр своего же класса, делает конструктор приватным (недоступным из вне) и реализует метод getInstance() для получения этого статического экземпляра.

В данной статье была описана разработка веб-приложения онлайн-галереи с элементами социальной сети. Проанализирована предметная область, выделены основные информационные объекты, которые нашли своё отражение в формировании общей структуры данных базы данных, определены все взаимодействия и отношения между ними. Был изучен рынок схожих проектов и платформ.

Также, была составлена декомпозиция проекта по методологии IDEF0 и спроектирована схема базы данных данного веб-приложения. Приведен список использованных шаблонов проектирования. Описан процесс реализации UX/UI дизайна, разработка общей визуальной составляющей бренда и общего стиля.

Программные и технические средства, которые применяются для разработки проекта: язык программирования TypeScript, фреймворки Node.JS, React.JS, Express.JS, Next.JS, а также метаязык SCSS. В качестве базы данных будет использован сервис Firebase.

Литература

1. Брусенцова Т.П. Проектирование интерфейсов пользователя. Минск: БГТУ, 2019. 170 с.
2. Вандеркам Д. Эффективный TypeScript: 62 способа улучшить код. СПб.: Питер, 2020. 288 с.
3. Мардан А. React быстро. Веб-приложения на React, JSX, Redux и GraphQL. СПб.: Питер, 2019. 560 с.

© Мироненко Е.В., 2023

УДК 004.41

Мордвинов А.В.

Новосибирский национальный исследовательский государственный университет
г. Новосибирск, Россия

СОЗДАНИЕ ПРОТОТИПА ПЛАТФОРМЫ ДЛЯ РАЗРАБОТКИ ЭЛЕКТРОННЫХ УСТРОЙСТВ

Сегодня создание пользовательских электронных устройств разной степени сложности доступно для широкой аудитории. Область их применения поистине огромна. Это могут быть как простые устройства наподобие часов и метеостанций, так и автоматизированные системы управления профессионального уровня [1; 2]. Платформа Arduino (<https://clck.ru/ZGa2G>) в последнее десятилетие значительно упростила разработку таких устройств и, как следствие, снизила порог вхождения. Большое сообщество и база знаний открывают дорогу людям, далеким от программирования и электроники.

Однако эта платформа все еще имеет ряд возможностей для дальнейшего упрощения. Например, языки C и C++ обладают особенностями, которые сложны для освоения неподготовленными пользователями. Вместе с тем, эти особенности редко относятся непосредственно к решаемой задаче. Подключение электронных датчиков к устройству также не всегда интуитивный процесс – отсутствие стандартизации библиотек усложняет инициализацию и использование программных компонентов.

Цель данной работы – создание прототипа платформы для разработки электронных устройств, которая абстрагирует все несущественные для решения задач детали, такие как выбор библиотек, их настройка и подключение, инициализация компонентов. Такая платформа упростит подход к разработке и вовлечет еще большую аудиторию в область.

Первым этапом был проведен анализ существующих решений, преследующих ту же цель – упрощение процесса разработки. Ниже будут отмечены два наиболее значимых из них.

Xod.io (<https://clck.ru/33eeak>) содержит обширную библиотеку компонентов и открыт к расширению. На первый взгляд, такая схема выглядит многообещающе (рис. 1). Графический редактор открывает возможность создавать алгоритмы каждому, даже не имеющему опыта в программировании. Однако при детальном рассмотрении становится понятно, что на больших проектах такую схему трудно читать, отлаживать и редактировать. Таким образом, теряется то преимущество, которое должен давать графический язык – простота и наглядность.

Другая система ESPHome (<https://clck.ru/33eeoP>) также содержит обширную библиотеку компонентов и активное сообщество. Вместо языка для скриптов, авторами был выбран простой текстовый конфигурактор на языке разметки YAML. Главный недостаток системы состоит в том, что заточена она в первую очередь на домашнюю автоматизацию. Задачи, выходящие за эти рамки, решить трудно и часто невозможно.

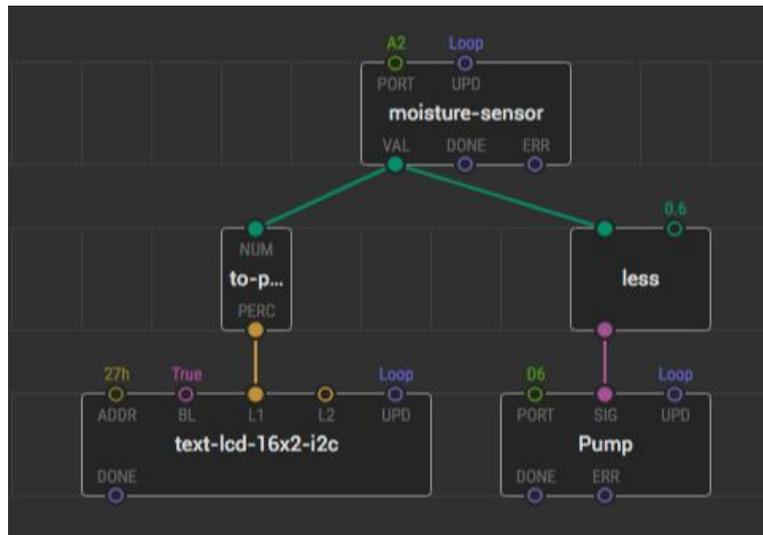


Рис. 1. Интерфейс `hod.io`

После проведенного анализа были сформированы требования к системе. Стало понятно, что графический язык программирования упрощает разработку только для простых проектов. `hod.io`, несмотря на впечатляющую проработку пользовательского интерфейса и высокое качество сервиса, оказывается неудобным для решения задач, например, уровня автоматизации теплицы – схема становится слишком большой и нечитаемой.

Резюмируя: императивные низкоуровневые языки C++ и C сложны в освоении, графические языки показывают себя неприменимым на практике, поэтому наиболее разумным остается «усредненный» вариант: стандартный язык программирования, но упрощенный, содержащий минимальный набор описательных конструкций. Данное решение ограничивает инструментарий пользователя, при этом оставаясь «реалистичным», в том смысле, что пользователь, научившись работать с упрощенным языком, при необходимости с легкостью перейдет на C или C++.

Для выбранной структуры языка программирования подходит подмножество языка C с некоторыми изменениями и ограничениями применения его синтаксиса. Наиболее частые абстракции, с которыми приходится работать при написании программ для электронных девайсов, были вынесены в отдельные языковые примитивы. Примитивы настраиваются через элементы пользовательского интерфейса, тем самым, уменьшается размер исходного кода и минимизируется возможность ошибок инициализации, настройки и использования.

Таких примитива системы было выделено три:

1. `Components` – периферия. Электронные компоненты, такие как кнопки, датчики температуры, влажности, освещенности, устройства отображения (дисплеи, лампочки), устройства управления (реле, электродвигатели).
2. `Timers` – таймеры, срабатывающие в определенный интервал времени. Таймер принимает программный код на исполнение в качестве лямбда-выражения.
3. `State` – переменные глобального состояния приложения, доступные в программном коде.

Для разработки пользовательского интерфейса был выбран следующий стек технологий:

- ReactJS – библиотека для создания пользовательских интерфейсов;
- Zustand – простой и эффективный state manager приложения;
- Monaco editor – библиотека, помогающая создать редактор кода в браузере. Открыта к расширению для кастомных языков программирования.

Пользовательский интерфейс был реализован в виде Web приложения. Левая часть отведена под примитивы системы, для каждого примитива разработана отдельная вкладка. В левом окне отображено меню для работы со полями примитивов. Для каждого типа доступно добавление и удаление полей. Правая часть отведена под редактор кода проекта. Для него реализован функционал подсветки синтаксиса, автодополнения и подсветки ошибок.

Первая вкладка отведена для примитива Components (рис. 2).

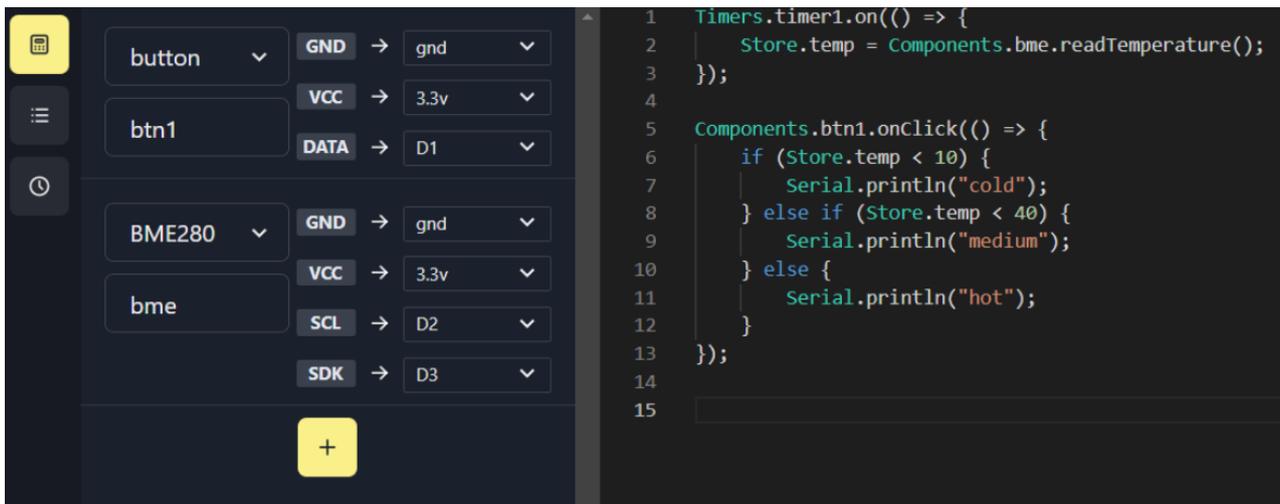


Рис. 2. Вкладка для выбора компонентов

Пользователь выбирает компонент из списка доступных. На данный момент система имеет поддержку наиболее распространенных компонентов в области хобби-проектов, в дальнейшем список планируется расширять. Каждый компонент обладает уникальной схемой подключения и набором пинов. Для каждого пина пользователю предлагается выбрать удобный ему разъем на плате контроллера (из доступных для данного типа пинов). При наличии только одного осмысленного варианта подключения он будет выбран автоматически – например, для пина GND существует единственный вариант (GND на плате). Для некоторых компонентов (например, работающих по SPI шине), после компиляции и запуска система может сообщить о неправильном их подключении.

Для выбранных устройств генерируется структура Components, доступная пользователям в программном коде. Редактор поддерживает автодополнение для созданных пользователем уникальных имен компонентов, а также для набора функций согласно набору функций компонента.

Доступные функции делятся на три типа:

1. Считывание значения с компонента. На строке 2 (рис. 2) происходит чтение значения компонента bme, и сохранение его в поле temp (о примитиве Store будет рассказано далее).

2. Задание программного кода на исполнение для интерактивных компонентов (такие как кнопки и потенциометры). На строках 5-13 (рис. 2) описывается функция, которая будет вызвана при физическом нажатии на кнопку.

3. Задания значения для актуаторов и компонентов отображения (LED, дисплеи).

Вторая вкладка отведена под примитив State (рис. 3).

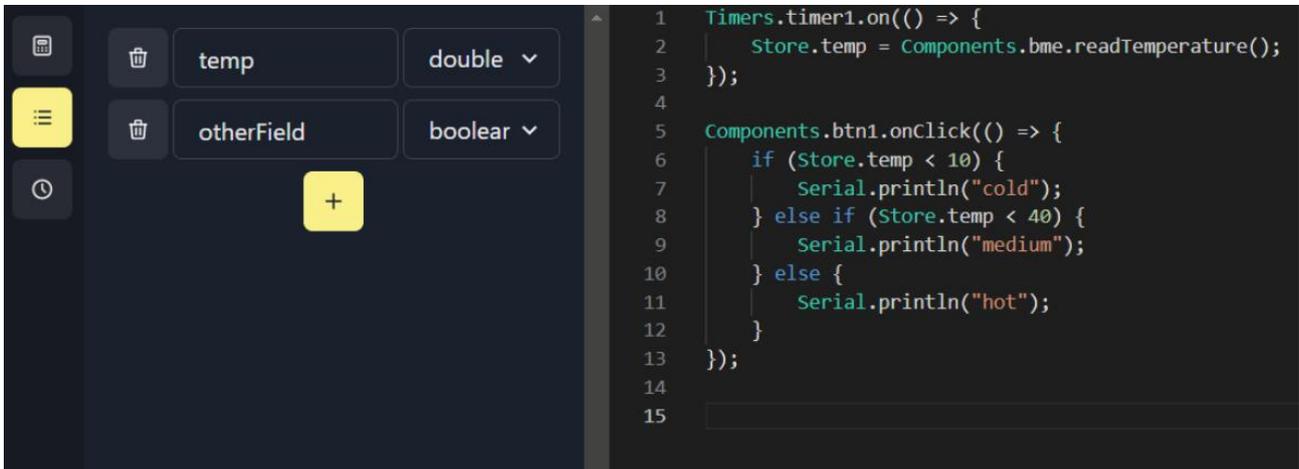


Рис. 3. Вкладка для задания глобального состояния приложения

Здесь задается глобальное состояние приложения. Пользователь добавляет поля, для каждого задает имя и тип. После чего все поля доступны в программном коде через структуру State. Так же, как и в предыдущем случае, работает автодополнение. Значение поля можно задавать – на строке 2 (рис. 3) значение прочитанное из компонента сохраняется в поле temp, и читать – на строка 6 (рис. 3) значение temp сравнивается с константой.

Третья вкладка отведена под примитив Timers (рис. 4).

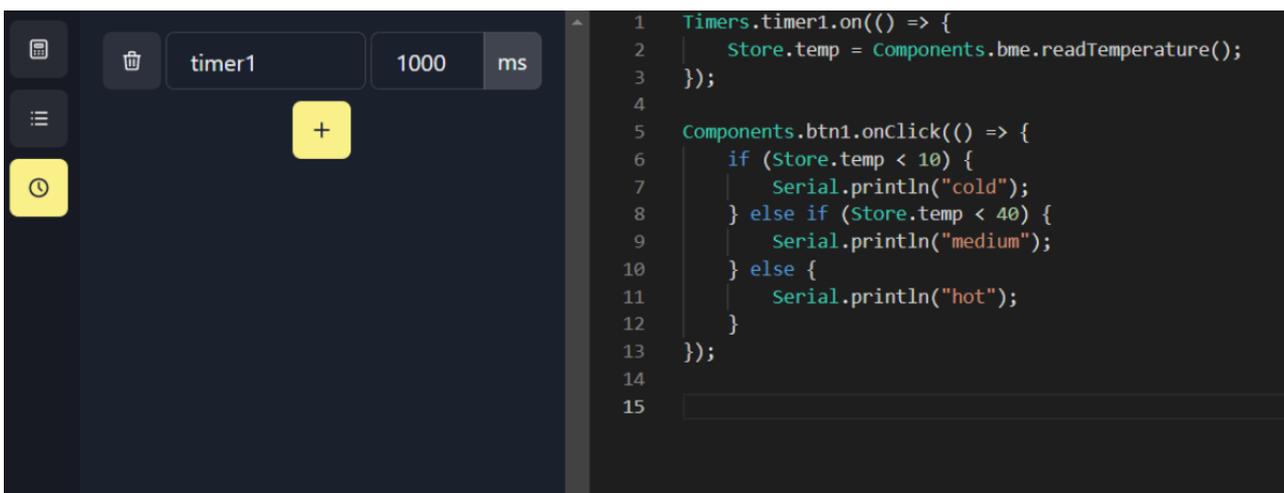


Рис. 4 Вкладка для создания таймера

Здесь происходит создание таймеров. Для каждого таймера задается имя и интервал в миллисекундах. В дальнейшем планируется добавление продвинутых интервалов (например, «по будням в 17:05», «каждый день на рассвете»). Таймеры доступны в программном коде через структуру Timers. Для структуры поддерживается автодополнение. Каждый таймер

имеет метод оп который принимает лямбда-выражение с программным кодом. Код будет исполняться в момент срабатывания таймера (в соответствии с заданным интервалом).

В результате данной работы был разработан прототип платформы, который включает в себя упрощенный предметно-ориентированный язык, Web IDE для работы с примитивами системы и написания скриптов. Ключевым принципом разработанной платформы является выделение трех ключевых аспектов разработки в качестве примитивов языка. Эти примитивы задаются через пользовательский интерфейс. Таким образом от пользователя скрываются механизмы установки и подключения библиотек, инициализации компонентов, подключения и инициализации таймеров. Такой подход позволяет пользователю сразу приступить к решению задачи, не углубляясь в документацию. Предполагается, что данного подхода достаточно для решения большей части задач в рамках хобби-проектов по созданию электроники, что и является целью проекта. Недостатки неизбежно будут выявлены в ходе тестирования на реальных задачах. В этом случае функционал примитивов может быть расширен, могут быть добавлены новые примитивы, а также изменения может претерпеть и сам язык.

Литература

1. McLean K.M., Pasulka A.L., Bockmon E.E. A low-cost, accessible, and high-performing Arduino-based seawater pH control system for biological applications // HardwareX. 2021. Vol. 10. <https://doi.org/10.1016/j.ohx.2021.e00247>
2. Turner B., David J.H., Carlyle-Moses D.E., Rahman M. Low-cost high-resolution stemflow sensing // Journal of hydrology. 2019. № 570. P. 62-68. <https://doi.org/10.1016/j.jhydrol.2018.12.072>

© Мордвинов А.В., 2023

РЕДУКЦИЯ БОЛЬШИХ ДАННЫХ В РАСПРЕДЕЛЕННЫХ СИСТЕМАХ НА ОСНОВЕ ФУНКЦИОНАЛЬНОГО ПОДХОДА

Неотъемлемой частью мировой экономики являются территориально распределенные организации и компании (далее - ТРК), распространившие свою деятельность на большую часть земного шара. Под ТРК понимается иерархия, состоящая из большого количества природных и искусственных территориально распределенных гетерогенных акторов, прямо или косвенно участвующих в процессе достижения общей стратегической цели [1]. К типичным ТРК, в частности, относятся компании eBay, Walmart, Amazon, Alibaba и до. К общими свойствами ТРК можно отнести: наличие центрального офиса и большого количества территориально распределенных подразделений; применение автоматизированных систем поддержки бизнес-процессов (CRM) на всех уровнях иерархии; применение датчиков для фиксации результатов деятельности, использование интернет-коммуникаций для передачи отчетности на высший уровень для принятия стратегических решений. Быстрое развитие ТРК обусловлено внедрением новых методов ведения глобального бизнеса на основе инфраструктуры Интернет и массовым применением компьютеров и смартфонов [2].

Глобализация бизнеса привела к лавинообразному росту объема данных, необходимых для принятия решений на всех уровнях ТРК. В результате потоки данных с нижних уровней иерархии в центр стали перегружать каналы связи, что увеличило к запаздыванию управляющих решений, снижению их качества и потере конкурентных преимуществ ТРК.

Для решения проблем больших данных было предложено несколько подходов, включая теорию сетей, сжатие больших данных, уменьшение размерности (редукция), устранение избыточности, интеллектуальный анализ данных, машинное обучение и др. Практическое применение этих подходов затруднено специфическими свойствами ТРК: акторы на каждом уровне для поддержки собственных бизнес-процессов используют различные типы и структуры данных, оригинальное программное обеспечение (ПО) и формы отчетности. Более того, существующие системы поддержки принятия решений (СППР) в ТРК ориентированы на плановые задачи, относительно стабильные глобальные условия и низкий уровень неопределенности. В условиях пандемии COVID-19, вызвавшей рост глобальной неопределенности, компании стали менять структуру, резко увеличилось количество распределенных акторов (особенно в области ИТ и торговли), появилось множество новых диагностических показателей для оценки их деятельности и соответствующее локальное ПО. Традиционные СППР в новых условиях оказались неэффективны, что привело к поиску новых подходов, включая “small data” [4] и “data-driven decision-making” [3], которые используют разные методы редукции больших данных [5]. Под редукцией в СППР понимается сокращение исходного объема данных до уровня, достаточного для принятия решений.

В данной работе предлагается функциональный подход к редукции больших данных, инвариантный типу акторов и позволяющий сократить время синтеза решений на минимальном объеме данных.

Базовые положения и модели. В основу постановки задачи положим следующие утверждения, отражающие реалии современных ТРК и условия решения оперативных задач:

1. Акторы ТРК, независимо от природы, снабжены датчиками, фиксирующими текущие значения диагностических параметров акторов.

2. Результаты работы любого алгоритма для оценки состояния акторов любого типа можно отобразить на единую шкалу.

3. Шкала может содержать диапазоны значений, соответствующие текущему состоянию актора и релевантному управляющему решению.

4. Предметная область для оперативных задач, решаемых в условиях неопределенности, формируется экспертом, знающим специфику среды, ТРК и решаемой задачи.

На основании положений (1)-(4) построим комплекс концептуальных моделей, формирующий общий взгляд на ТРК. Для простоты изложения ограничимся рассмотрением трехуровневыми ТРК.

ТРК в целом представим кортежем:

$$TDC = (A, B, C, com) \quad (1)$$

где: A, B, C – акторы высшего, среднего и нижнего уровней; com – коммуникации.

Модели акторов опишем по унифицированной схеме:

$$mA = (adrA, crmA, xA, vxA, rA, com) \quad (2)$$

$$mB = (adrB, crmB, xB, vxB, rB, com) \quad (3)$$

$$mC = (adrC, crmC, xC, vxC, rC, com) \quad (4)$$

где: x_C, x_B, x_A – данные от датчиков; vxC, vxB – объем передаваемых данных; r_C, r_B, r_A – ресурсы для обеспечения выполнения решений.

Датчики понимаются в широком смысле слова – как любые источники достоверных данных, характеризующих объект наблюдения.

На основании утверждений (1)-(4) и моделей (1-4) сформулируем задачу.

Пусть имеется ТРК, состоящая из центра (A) и m территориально распределенных компонентов (B), включающих $n \rightarrow \infty$ гетерогенных объектов (C). Акторы уровней A, B, C имеют системы поддержки бизнес-процессов $crmA, crmB, crmC$, объединенные каналами связи com. На основе отчетных локальных алгоритмов и данных x_A, x_B объемом $dx_A \rightarrow \infty, dx_B \rightarrow \infty$ crm -системы формируют и отчетные данные, которые отсылаются в центр A, где на их основе принимаются стратегические решения за время $t \rightarrow \infty$.

Требуется разработать унифицированную методику, обеспечивающую минимизацию объема передаваемых данных ($vxC, vxB \rightarrow 0$) и времени принятия оперативных решений ($t \rightarrow \infty$) в актором A независимо от количества и типа акторов C.

Основное требование к решению – не менять алгоритмы и объем потоков данных в системах CRM для решения плановых задач.

Для разработки методики решения используем парадигму high level architecture [7], предполагающую построение внешних по отношению к CRM функций f_A , f_B , f_C , наблюдающих за состоянием акторов A , B , C на основе анализа дополнительных параметров dx_A, dx_B, dx_C . Под дополнительными параметрами понимаются данные, полученные на основе дополнительного анализа x_C, x_B, x_A или из источников, не использованных в CRM.

$$f_A = (x_A/dx_A, alg_A, ogr_A, V_A, U_A, com) \quad (5)$$

$$f_B = (x_B/dx_B, alg_B, org_B, V_B, U_B, com) \quad (6)$$

$$f_C = (x_C/dx_C, alg_C, ogr_C, V_C, U_C, com) \quad (7)$$

где: V_A, V_B, V_C – множество возможных состояний акторов на уровнях A, B, C ; U_A, U_B, U_C – множество управляющих решений, релевантных состояниям.

Соответственно целостный взгляд на ТРК формируется тремя кортежами:

$$m_A = (adr_A, crm_A, f_A, x_A, vx_A, com) \quad (8)$$

$$m_B = (adr_B, crm_B, f_B, x_B, vx_B, com) \quad (9)$$

$$m_C = (adr_C, crm_C, f_C, x_C, vx_C, com) \quad (10)$$

Реализация моделей (8)-(10) включает следующие шаги.

Шаг 01. Уяснение топ-менеджером ТРК новой оперативной задачи.

Шаг 02. Разработка функций f_A, f_B, f_C для решения, включая формирование alg, V, U, ogr .

Шаг 03. Интеграция f_A, f_B, f_C в жизненный цикл СППР.

Шаг 04. Регулярный «опрос» датчиков и применение алгоритмов alg_C для синтеза состояний $V_j = f_C(x_C/dx_C)$ для всех акторов C . Период опроса устанавливается в зависимости от важности задачи и, соответственно, масштаба последствий «упущенной» ситуации.

Шаг 05. Оправка от акторов C пакета $\langle adr_Cj, V_j, U_j \rangle$ акторам B или A в зависимости от степени нарушения ограничений ogr_C .

Шаг 06. Принятие решений и выполнение решений U_j на уровнях C or B or A .

Шаг 07. Переход на шаг 04.

В результате негативные ситуации на нижнем уровне фиксируются на раннем этапе их развития и нормализуются решениями на уровне C, B, A в зависимости от сложности ситуации и наличия ресурсов. Соответственно, разгружаются каналы связи, так как передается только та информация, которая необходима для оперативного принятия решений при соблюдении гомеостаза процессов решения плановых задач системами CRM.

Для практического применения разработанной методики возможны различные подходы. Для крупных торговых компаний, размещающих в разных странах множество объектов, целесообразно использовать многоагентный подход. Под агентом понимается автономная программа (микросервис), участвующая в реализации проекта. Конечное множество агентов, достаточное для реализации проекта, называется многоагентной системой (МАС).

В основу построения МАС положим системный принцип: множество агентов должно решать проблему системно, с учетом вклада состояния всех объектов. Для этого агенты синтеза состояния на уровне объектов должны формировать интегральный показатель,

отображаемый на единую шкалу, независимо от рода деятельности актора. В качестве такой шкалы взят континуум 0.00..1.00 и функции принадлежности, предложенные Д. Заде [6].

На основе вышесказанного предлагается архитектура, включающая семь агентов, реализующих полный цикл решения от получения сигналов от датчиков для визуализации результатов принятия решений в текстовом или графическом виде (рис. 1).

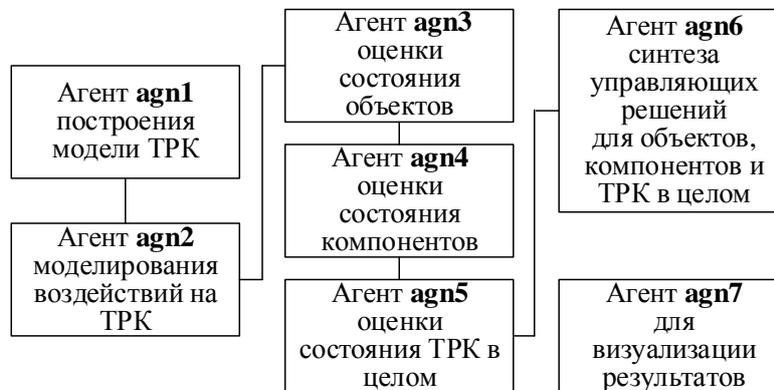


Рис. 1. Многоагентная архитектура

Важным преимуществом данной архитектуры является возможность коррекции кода любого агента без изменения других агентов. Соответственно, появляется возможность применения унифицированной общей схемы при гибком выборе алгоритмов синтеза управляющих решений для различных по природе акторов (роботов, людей, автомобилей).

Архитектура реализована на языке Python. Для хранения данных, сформированных экспертом во время построения модели ТРК, использовалась СУБД MySQL.

В качестве апробации МАС рассмотрим решение следующей актуальной задачи.

Требуется построить систему для решения различных задач поддержки принятия управляющих решений для администрации прибрежного города, который в результате изменения климата оказался в зоне повышенного риска затопления приливными волнами цунами. В городе имеется 1900 зданий, размещенных в пяти районах. В качестве первой задачи выделим определение количества разрушенных зданий в каждом районе. При решении будем использовать справочную информацию (адреса зданий, ресурсы для восстановления).

Ниже представлен вариант решения, включающий диалог с экспертом в процессе построения модели Города, и сообщения агентов о работе без сбоев. Для моделирования результатов воздействия цунами (фактически показаний датчиков, установленных в здания) использовался вектор из 1900 случайных чисел в диапазоне 0.00...1.00 (рис. 2).

Аналогичным образом решаются задачи категоризации домов для распределения ресурсов для устранения негативных последствий, направления бригад МЧС в наиболее пострадавшие районы и решения многих других задач.

```
активен агент agn1
Имя ТРК: Город
Названия компонентов: Район1, Район2, Район3, Район4, Район5
Количество объектов в компонентах: 400, 300, 500, 100, 600
Возможные состояния объектов: норма, поврежден, разрушен
Управляющие решения: наблюдать, отремонтировать, снести
Алгоритм для синтеза состояний: algMAX
База с ресурсами ТРК: dbres.frm
База с адресами объектов: dbadr.frm
Карта КМС: mapfile.jpg
agn1: модель ТРК "Город" построена
активен агент agn2
[0.35, 0.04, 0.75, 0.71, 0.61, 0.33, 0.3 ...
agn2: вектор воздействия построен.
активен агент agn3
agn3: Оценка объектов выполнена.
активен агент agn4
agn4: оценка компонентов выполнена
активен агент agn5
agn5: оценка ТРК выполнена.
активен агент agn6
agn6: синтез управляющих решений выполнен.
активен агент agn7
agn7: визуализация результатов выполнена.
```

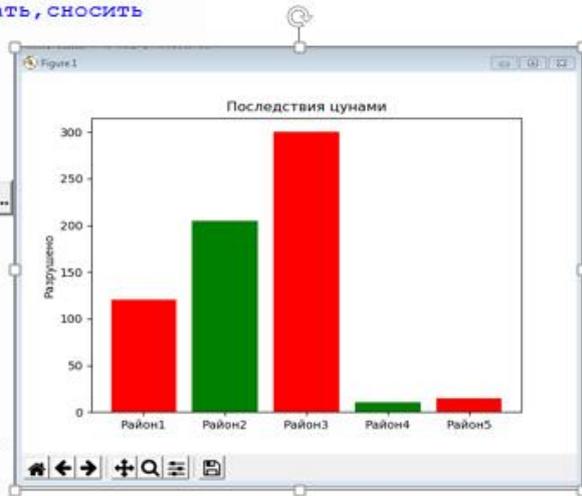


Рис. 2. Диалог агента с экспертом и вывод результата

На основе результатов апробации можно считать, что функциональный подход достаточно эффективен для оперативного принятия решений на основе редукции больших данных в территориально распределенных компаниях и организациях. Дополнение плановой части (CRM) расширяемым “оперативным” модулем позволяет синтезировать управляющие решения для устранения негативных ситуаций на ранней стадии их возникновения, не используя для пересылки в центральный офис базы данных локальных акторов, что существенно разгружает каналы связи без потери качества решения. Наибольший эффект методика принесет для разработки систем мониторинга ТРК в потенциально опасных районах.

Литература

1. Виссия Х., Краснопрошин В.В., Вальвачев А.Н. Принятие решений в информационном обществе. СПб: ЛАНЬ, 2019. 227 с.
2. Управление развитием крупномасштабных систем (MLSD'2022). Труды XV-й междунар. конф, 26-28 сент. 2022 г., Москва. М.: ИПУ РАН, 2022. 1500 с.
3. Elgendy N. DECAS: a modern data-driven decision theory for big data and analytics // Journal of Decision Systems. 2022. Т. 31. № 4. С. 337-373.
4. Ghopra A. Handbook of Normative Multiagent Systems. College Publication, 2018. 424 с.
5. Hekler E. et al. Why we need a small data paradigm // BMC Medicine. 2019. Т. 17. С.1-9.
6. Zadeh L., Klir G., Yuan B. Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems. World Scientific Pub. Co. Inc., 1996. 840 p.
7. 1516.1-2000 IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA). Federate Interface Specification 2000. IEEE, 2000. 480 p.

УДК 519.682

Мурина А.В., Воронин В.В.

Тихоокеанский государственный университет
г. Хабаровск, Россия

АНАЛИЗ ЭФФЕКТИВНОСТИ РАЗРАБОТКИ ВЕБ-САЙТОВ *BOOTSTRAP*

Каждый профессиональный разработчик программного обеспечения заинтересован в повышении своей производительности труда, именно поэтому большинство веб-разработчиков предпочитают использовать в качестве инструментальных средств CSS-фреймворки, самым распространенным из которых является *Bootstrap*.

Bootstrap представляет собой свободный фреймворк интерфейсов для быстрой и простой Web разработки. *Bootstrap* предоставляет шаблоны дизайна, основанные на HTML и CSS для разметки, форм, кнопок, таблиц, навигации, диалоговых окон, каруселей изображений и многого другого, а также дополнительные плагины JavaScript. На основе *Bootstrap* можно легко создавать сайты с «отзывчивым дизайном» – сайты, которые одинаково хорошо выглядят на всех типах устройств, от небольших телефонов до настольных компьютеров [3].

В данной работе приведены результаты анализа основных особенностей данного инструментального средства и оценки его эффективности.

Обзор основных особенностей *Bootstrap*

Основная особенность *Bootstrap* заключается в том, что он содержит в себе два фундаментальных подхода, а именно: *OOCSS* и *Atomic CSS*.

Подход *OOCSS* соответствует методологии объектно-ориентированного CSS, так как он использует объекты или структуры данных, состоящие из полей данных и методов для взаимодействия с разработанным прикладным приложением или подпрограммами [4].

Два основных принципа *OOCSS*, а именно: отделение структуры от оболочки и разделение контейнера и содержимого.

Отделение структуры от оболочки означает отделение стиля макета от стиля дизайна. Например, для создания кнопки используется два класса, один из которых отвечает за структуру кнопки, а другой за оформление.

Методология *OOCSS* базируется на двух основных вещах:

- стили структуры;
- стили оформления.

К структуре относится всё, что непосредственно влияет на расположение контента внутри. К стилям структуры, например, можно отнести:

- высоту;
- ширину;
- внутренние и внешние отступы.

К стилям оформления, например, относятся:

- фон;
- шрифты;
- тени и цвет границ.

Второй подход, который используется в *Bootstrap* – это применение класса *Atomic CSS*. На данном классе построены утилиты во фреймворке. По существу, *Atomic CSS* характеризуется таким подходом, при котором один класс использует одно свойство. Это способствует сокращению избыточности или дублирования кода.

Компоненты и утилиты позволяют быстро создать прикладной проект различной сложности и конфигурации. Единственное ограничение - это тип дизайна. Выбранный компонент будет выглядеть в точности так, как в документации. Однако *Bootstrap* является современным фреймворком и имеет целый спектр привлекательных дизайнов.

Оценка эффективности *Bootstrap*

Данная задача решалась относительно возможностей конкретного приложения. С помощью фреймворка *Bootstrap* пятой версии разработан оригинальный дизайн интернет-магазина по продаже садовой мебели «*GreenHome*».

Оценим эффективность работы *Bootstrap* на примере каталога товаров этого интернет-магазина. Основные функциональные возможности каталога иллюстрируются на рисунке 1.

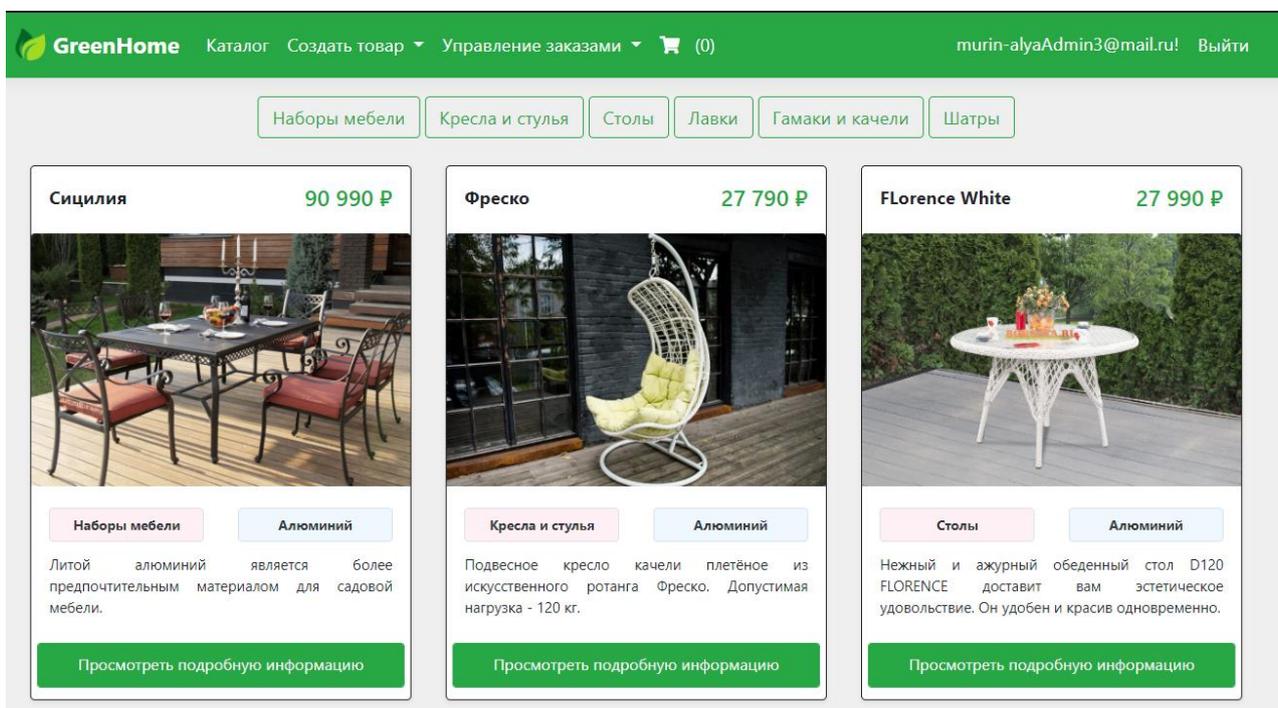


Рис. 1. Каталог интернет-магазина «GreenHome»

Каталог интернет-магазина состоит из отдельных карточек товаров, выпиленных с помощью *Bootstrap 5* (на рис. 1 представлены три карточки).

В качестве примера на рисунке 2 приведен адаптированный код карточки товара интернет-магазина «*GreenHome*». В шаблоне «Скелет» *Bootstrap* в процессе разработки изменен цвет кнопки, добавлены плашки с описанием категории товара и типа материала, появилась строка с указанием названия и цены товара.

```
<div class="card bg-white rounded shadow-sm" style="height: 30rem; border: 1px solid #222">
  <div class="card-body px-3 pt-3 pb-1 row">
    <div class="col-7"><label class="float-left">@Model.Name</label></div>
    <div class="col-5"><label class="float-right"><span class="text-info h5">@string.Format("{0:c0}", Model.Price)</span>
    </label> </div>
  </div>
  
  <div class="card-body p-1 px-3 row">
    <div class="col-6">
      <span class="badge p-2 border w-100 text-dark" style="background-color: #lavenderblush">@Model.Category.Name</span>
    </div>
    <div class="col-6 border-0">
      <span class="badge p-2 border w-100 text-dark" style="background-color: #aliceblue">@Model.ApplicationType.Name</span>
    </div>
    <div class="col-12 pt-2" style="font-size: 13px; text-align: justify">
      <p>@Model.ShortDesc</p>
    </div>
    <div class="col-12 p-1">
      <a asp-action="Details" asp-route-id="@Model.Id" class="btn btn-dark form-control btn-sm p-2" style="height: 40px">
        Просмотреть подробную информацию</a>
      </div>
    </div>
  </div>
</div>
```

Рис. 2. Адаптированный код карточки товара

На официальном сайте *Bootstrap* (рис. 3) существует большое число разнообразных карточек и *html* кода к ним (<https://clck.ru/34pnh>).

Рис. 3. Официальная страница Bootstrap

Как следствие, разработчику не требуется «с нуля» разрабатывать дизайн и верстать страницу вручную (выравнивать картинку, подбирать размер кнопки и выбирать шрифт), необходимо только скопировать «скелет» кода и оформить его под свой проект.

Таким образом, как результат, предлагается акцентировать внимание на следующих эффективных преимуществах использования инструментария *Bootstrap*.

1. *Bootstrap* упрощает и существенно ускоряет процесс разработки. Этот процесс сопровождается непрерывной работой по добавлению или удалению функционала на текущей странице.

Готовые компоненты и утилиты *Bootstrap* позволяет разработчику быстро и качественно сверстать данную страницу.

2. В настоящее время существует большое число разнообразных поисковых систем, каждая из которых по-своему обрабатывает *HTML* и *CSS*. Компоненты *Bootstrap* способны адаптироваться под любую поисковую систему и корректно отображать веб-приложения на различных устройствах [5].

3. Наличие большого числа стильных и продуманных компонентов.

4. Фундаментом фреймворка *Bootstrap* является собственная сеточная система. В составе *Bootstrap* имеются компоненты, позволяющие выстроить структуру веб-сайта на основе адаптивной сеточной компоновки.

Сеточная система *Bootstrap* дает возможность определить набора строк, каждая из которых содержит набор независимых столбов, что позволяет создать надежную сеточную компоновку веб-страницы. Кроме того, каждый столбец может иметь различные размеры, чтобы идеально соответствовать шаблону [2].

5. Для создания страниц не требуется экспертного знания *HTML* и *CSS*. Данная особенность позволяет создавать новые блоки не только профессиональному верстальщику, но и любому желающему, который знает базовые основы вёрстки.

6. Наличие большого количества сообществ и обучающих материалов, которые помогут начинающему разработчику разобраться в особенностях работы с *Bootstrap* (<https://clck.ru/32aTkx>).

В результате использования инструментария *Bootstrap* выявлены, по нашему мнению, два следующих негативных момента.

1. Фреймворк *Bootstrap* даёт не только набор классов, но и интерактивные компоненты. Для их работы используется несколько *JavaScript* библиотек, которые увеличивают вес подключаемых файлов на странице. Система *Bootstrap 5* вместе с библиотеками *JavaScript* весит ориентировочно около 300 Кб. Это может стать критичной точкой при разработке мобильных приложений, ориентированных на использование при слабом интернет-соединении [1].

2. Стандартные компоненты инструментального средства накладывают ограничения на дизайн. Модификация компонентов потребует знаний в вёрстке, так как даже незначительные изменения необходимо проверить в браузерах и на разных разрешениях. При этом дизайн стандартных компонентов *Bootstrap* лаконичен и является примером для дизайнера. При своей

простоте они функциональны и помогают правильно подать информацию. Даже в таких ограничениях создаются красивые и удобные сайты за счёт грамотной работы с *UI* (<https://clck.ru/34pnbx>).

Bootstrap – это эффективный фреймворк, используемый для разработки веб-приложений. Его библиотека интерфейсных компонентов широко используется для создания интерактивных и адаптивных веб-приложений и веб-сайтов, к которым мы привыкли сегодня. Фреймворк имеет лицензию *MIT* и, следовательно, его можно бесплатно использовать как в открытых, так и в коммерческих проектах.

Превосходство *Bootstrap* основывается на трех аспектах. Первый – таблица стилей, содержащая базовую CSS-разметку для каждого HTML-элемента, обеспечивающую их единообразный привлекательный вид. Второй аспект – компоненты. Их можно использовать многократно простым копированием и вставкой кода. Третий аспект – подключаемые модули и плагины на JavaScript.

Данный фреймворк содержит большое число разнообразных утилит и компонентов, с помощью которых можно быстро и качественно создать проекты любых размеров от элементарных лендингов до сложных веб-приложений. Использование данного инструментария позволит существенно сократить время реализации проекта.

Литература

1. Мишин Т. Быстрое создание современных сайтов. Издательские решения, 2016. 265 с.
2. Морето С. Bootstrap в примерах / Пер. с англ. Рагимов Р.Н. М.: ДМК Пресс, 2017. 314 с.
3. Перепелица Ф.А. Эффективная разработка веб-сайтов. Bootstrap. СПб: Университет ИТМО, 2015. 71 с.
4. Прохоренко Н.А. Bootstrap и CSS-препроцессор Sass. Самое необходимое. СПб.: БХВ-Петербург, 2021. 497 с.
5. Jorg Krause. Introducing Bootstrap 4. Berlin, Germany, 2016. 207 с.

© Мурина А.В., Воронин В.В., 2023

УДК 004

Мысягин В.В.

Нижегородский государственный технический университет им. Р.Е.Алексеева
г. Нижний Новгород, Россия

ЦИФРОВЫЕ ДВОЙНИКИ И ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ В ПРОИЗВОДСТВЕ. ОТЛИЧИЯ И ПРАКТИКИ

В современном мире, при обилии современных технических терминов, некоторые специалисты путают или подменяют понятия «цифровой двойник» и «имитационное моделирование». Давайте разберемся, в чем принципиальное отличие этих понятий и рассмотрим практику внедрения современными предприятиями данных технологий.

Рассматривая процесс перехода предприятия на автоматизированное цифровое производство, называемый «цифровизация», следует дать различия между понятиями «цифровизация» и «автоматизация». Цель автоматизации – освободить человека от рутинных и опасных для жизни и здоровья операций, оставляя ему больше времени на анализ и принятие решений, творческий подход. Цифровизация предполагает следующий шаг – анализ и принятие решений машинами, сужение до предела участия человека в производстве и минимизацию человеческого фактора. Автоматизация – одна из целей, задач цифровизации или начальная стадия цифровизации, составляющая нижнего уровня.

Почему вообще современный высокотехнологичный сектор уделяет такое внимание вопросу цифровой трансформации?

Переход к цифровым технологиям дает возможность радикально изменить подход к производству. Он лежит в основе Промышленного Ренессанса, который предполагает автоматизацию процессов, использование Интернета вещей (IoT), искусственного интеллекта, больших данных и облачных вычислений. Внедрение цифровых технологий также способствует изменению стратегии поведения производителей в отношении к инновациям и удовлетворению запросов потребителей [1].

Цифровая трансформация промышленности и развитие цифровой экономики являются одними из главных инструментов повышения промышленной конкурентоспособности, устойчивости на рынке и сокращения издержек производства.

В секторах экономики, где потребители хотят получать более персонализированные товары, или где контрагенты специализируются на индивидуальных технологических и инженерных решениях, производители, которые широко используют цифровые технологии, повышают шанс перейти от традиционной модели массового производства к модели массовой персонализации.

Все стадии жизненного цикла продукции при цифровой трансформации предприятия повышают уровень сотрудничества друг с другом, повышается качество обмена информации о ходе реализации продукции за счет применения принципа цифровой непрерывности данных - от создания идеи, анализа рыночной ситуации, проектирования и производства до дальнейшего обслуживания в процессе эксплуатации и утилизации. Этот глобальный тренд набирает популярность по мере перехода производителей от массового производства к новой

модели, основанной на обмене цифровой информацией. Организации - лидеры будут выделяться за счет инновационной продукции, повышенной производительности и исключительного обслуживания. Они будут более гибкими и смогут удовлетворять потребности клиентов в индивидуализированных товарах, а также имеют возможность успешно и быстро адаптироваться в условиях турбулентной среды.

Цифровизация - это средство, метод получения желаемого результата, а именно гибкого производства, приносящего клиентам отличный результат в виде качественного продукта, а владельцам – более высокую прибыль. Цифровая трансформация – это процесс перевода предприятия из текущего состояния в «гибкое».

За счет комплексного подхода во внедрении и использовании новых принципов проектирования и цифровых технологий ведущие производители могут разумно координировать все аспекты своей производственной деятельности и интегрировать цепочку создания стоимости, которая проходит от поставщиков до конечных потребителей.

Так что такое цифровые двойники?

Цифровые двойники – не технология и не отдельный продукт, *это концепция*, в то время как 3D-моделирование, интернет вещей, 4G/5G, большие данные, искусственный интеллект, технологии блокчейна, граничные и облачные вычисления, – это технологии, которые используются при создании этой концепции (<https://clck.ru/33dwUM>). Принцип состоит в том, чтобы спроектировать эквиваленты, «двойники» физических объектов или активов в цифровом пространстве.

Цифровой двойник – цифровая копия живого или искусственного объекта. Этот термин относится к цифровой копии потенциальных и реальных физических активов (физический двойник), процессов, людей, мест, систем и устройств. Цифровые двойники создаются с целью облегчать средства контроля, понимания и оптимизации функций всех физических активов, обеспечивая беспрепятственную передачу данных между физическим и виртуальным мирами.

Чтобы чуть глубже разобраться в вопросе, немного затронем истоки - основой для цифрового производства послужили такие идеи, как:

1. Концепция «Интернет вещей» – оснащение предметов на производстве встроенными технологиями для взаимодействия между собой и внешней средой.

2. Концепция «Большие данные» – совокупность инструментов и методов обработки больших объемов данных для извлечения из них ценной информации для анализа и принятия решений.

3. Концепция «Киберфизические системы» - взаимодействие датчиков, оборудования и информационных систем между собой для прогнозирования, самонастройки и адаптации к изменениям, происходящим в производственном процессе.

Под цифровым производством понимается использование технологий цифрового моделирования всех стадий жизненного цикла изделия: изготовление опытного образца, производство изделий, эксплуатация и утилизация их. По сути, речь идет о создании цифровых двойников продукта и процессов его производства, включая оборудование и персонал предприятия [2].

Имитационное моделирование является составляющей любого цифрового двойника. Следовательно, напрашивается предположение: средства для разработки имитационных моделей могут быть применимы и для проектирования цифровых двойников.

Условно цифровые двойники можно разложить на два элемента.

Модели поведения являются первым элементом при создании цифрового двойника. Такие модели описывают поведенческие структуры и механизмы реагирования на изменяющуюся внешнюю среду. Основой дискретно-событийного моделирования как раз и является точное описание поведения в форме перехода между состояниями. Вторым элементом цифрового двойника являются данные. Особенность и сложность проектирования и внедрения цифрового двойника заключается в том, что данные не могут быть упрощены. В то время как в системах имитационного моделирования эта проблема решаема – данные принимают упрощенный вид. Таким образом, для некоторого множества моделей дискретно-событийного моделирования моделирование наступления событий на основе статистического распределения является типичным. Данные поступают в модель как от физических объектов, так и зарождаются уже самой моделью, поступают из разных серверов. В свою очередь сами данные могут быть или статическими, или динамическими. Таким образом, цифровым двойникам характерна поддержка множества соединений.

В упрощенном виде основное отличие цифрового двойника от имитационной модели может быть сформулировано такими тезисами: имитационное моделирование – это анализ «что будет если...», в то время как цифровой двойник – это анализ «что происходит сейчас и что будет если...» [3].

Цифровой двойник можно назвать моделированием в реальном времени. Классическое имитационное моделирование производится в виртуальных средах, которые могут представлять физические среды, вместе с тем они не объединяют данные в реальном времени. Моделирование в реальном времени происходит посредством обмена информации между цифровым двойником и соответствующей цифровому двойнику физической средой. Это дает возможность наиболее точно характеризовать и давать прогнозы аналитических моделей, а также повышается качество управления и мониторинга системы. Имитационные модели не требуют, в отличие от цифровых двойников поддержания большого количества данных.

Для общего понимания сути имитационного моделирования следует указать, что при внедрении этого метода исследуемая система заменяется моделью, которая с высокой точностью описывает и характеризует эту реальную систему, и уже на основе модели проводятся исследования и эксперименты с целью извлечения и анализа информации системы. Имитационное моделирование дает возможность имитировать поведение реальной системы во времени. Так, например, эксперименты ставятся над дорогими, опасными или невозможными объектами системы в имитационной среде.

Имитационное моделирование позволяет создавать приближенные к реальности ситуации, учитывая всевозможные особенности и нюансы системы, сводя к минимуму финансовые и производственные риски. Это дает преимущество предприятиям в их инновационной деятельности по срокам, трудозатратам и финансовым издержкам.

Очень часто для анализа работы и экспериментов необходима детальная информация об элементах, системах. В этом случае функционирование системы описывается в виде последовательности операций на ЭВМ, а программа для ЭВМ составляется на основе алгоритма поведения системы. Имитация реализуется на основе арифметических и логических операций в последовательности, которая соответствует моделируемому процессу.

Имитационное моделирование, как правило, используется на этапе проектирования и в ряде случаев для оптимизации, в то время как цифровые двойники применимы на всех этапах жизненного цикла продукции в реальном времени.

Предположим, планируется провести очередную диагностику станка на производстве. Но мастер начинает диагностику не по классическому пути - проверка текущего состояния оборудования и систем станка, проверка системы ЧПУ с применением средств диагностики. А вместо этого он сокращает путь в несколько раз. Ему представляются все данные о системах и станке до текущего момента, включая замену деталей в прошлом, данные технических показателей, информацию о записях предыдущих осмотров и диагностических мероприятий, и даже о потенциальных проблемах по итогам собранной и проанализированной информации, становится очевидна экономия времени и экономическая эффективность.

Во всем мире предприятия начинают свой путь цифровой трансформации с создания имитационных моделей критически важных активов, которые позволяют разрабатывать сценарии «что если?». Но в условиях реального производства активы, переходя к приработке новой продукции, новыми людьми и в новых режимах возникают непрогнозируемые и нежелательные типы поведения системы. В связи с этим в компании Gartner, например, утверждают, что эволюция цифровых двойников выходит за рамки активов и включает всю организацию целиком, вместе с людьми, процессами и типами поведения (<https://clck.ru/33dwUM>).

Цифровые двойники способствуют пониманию взаимодействий и корреляций между событиями и факторами, вызвавшими эти события, и вследствие интеграции их в имитационные модели.

Концепция цифровых двойников только начинает распространяться в производственном секторе. Организации используют разные подходы для построения имитационных моделей, но цифровые двойники – это то, к чему они придут. Например, в компании Tesla у каждого из произведенного автомобиля есть свой цифровой двойник, и вся информация через цифровой двойник поступает на имитационную модель на фабрике.

Имитационное моделирование позволяет так же находить узкие места в производстве. Так, компании, добившиеся успехов в расширении, перед привлечением окончательных инвестиций проверяют свой проект по масштабированию производства с помощью имитационного моделирования. Это позволяет своевременно обнаружить проблему на стадии проектирования и найти наиболее оптимальное и приемлемое с точки зрения рентабельности решение.

Например, компания AGCO запланировала расширить свое производство дизельных двигателей. Планировалось производить около 30000 дизельных двигателей в год, в основном

для сельскохозяйственной техники. Инвестиции крупные, поэтому необходимо все просчитать и учесть всевозможные факторы. Компания провела моделирование запланированной системы механической обработки компонентов, что позволило проверить, все ли в плане работает так, как ожидалось, и дало понимание какие компоненты и детали необходимо заменить.

Благодаря моделированию компания AGCO смогла оптимизировать свои планы, и, например, при обнаружении узких мест, вызванных нехваткой оборудования, их можно устранить, добавив дополнительное оборудование. Или наоборот, некоторые единицы оборудования могут быть локально устранены из плана, если имитационная модель подсказывает, что для эффективной работы необходимо и достаточно меньшее количество оборудования. Изменяя план в одном месте, могут возникать простои оборудования или узкие места в других местах. Моделирование позволяет наиболее быстро и точно подобрать наиболее верную комбинацию.

«Результат моделирования очень порадовал. Например, мы смогли провести исследования достижимости некоторого оборудования, которые раньше не могли делать без остановки производства. И мы смогли реально проанализировать время цикла, которое раньше можно было только оценить» - добавляет Норберт Прицль, AGCO (<https://clck.ru/33dx9T>).

Интересное применение имитационному моделированию нашли четыре предпринимателя из Нидерландов, которые основали компанию MondMaskerFabriek. Это первое предприятие по производству лицевых масок в Нидерландах, чтобы удовлетворить высокий спрос в условиях резкого дефицита (в начале пандемии COVID-19 в Нидерландах отсутствовало местное производство масок для лица, так же дефицит увеличивался сбоем в мировом производстве и логистике) (<https://clck.ru/33dxF8>).

Их инициатива была осложнена тем фактом, что для защиты своих работников от коронавируса их предприятие должно было постоянно предпринимать меры по социальному дистанцированию. Так же необходимо было открыть производство в кратчайшие сроки.

Был разработан макет, включающий в себя рабочие станции ручной сборки для сшивания и упаковки, а также для выполнения анализа пропускной способности.

Далее, используя планировку в качестве схемы логистики и обучения операторов, предприниматели использовали статистику для проверки расстояния между работниками, а также их производительности. Так же для поддержания безопасного и здорового рабочего места была разработана функция в программе имитационного моделирования, которая вычисляет расстояние между работниками и активирует оповещение каждый раз, когда дистанция в 1,5 метра нарушается. Это позволило в дальнейшем расширить производство в два раза и быстро спроектировать производственную линию, которая так же не нарушала бы требования дистанцирования. В то время как остальному миру пришлось закрыть фабрики из-за отсутствия дистанцирования, технология моделирования помогла построить фабрику за несколько недель, одновременно обеспечив безопасную рабочую среду.

Подводя итоги вышесказанному, будет справедливо заметить, что в России данные концепции находятся только на стадии развития, но уже сейчас можно судить о том, что

имитационное моделирование и цифровые двойники предприятия позволяют в разы увеличить эффективность работы, прибыль предприятия, безопасность работников и стабильно высокое качество принимаемых решений. И, несмотря на четкие правила и методики внедрения данных концепций, здесь есть место творческому подходу, ведь эти системы внедряются индивидуально в каждом предприятии, так как у всех предприятий есть свои особенности, возможности и требования.

Литература

1. Барсегян Н.В. Влияние цифровизации экономики на развитие промышленности // Вызовы цифровой экономики: итоги и новые тренды: сб. статей II Всероссийской научно-практической конференции (г. Брянск, 07 июня 2019 г.). Брянск, 2019. С. 71-75.
2. Коновалова Г.И. Методологические основы оперативного управления цифровым производством // Вызовы цифровой экономики: итоги и новые тренды: сб. статей II Всероссийской научно-практической конференции (г. Брянск, 07 июня 2019 г.). Брянск, 2019. С. 246-252.
3. Намиот Д.Е., Покусаев О.Н., Куприяновский В.П., Жабицкий М.Г. Цифровые двойники и системы дискретно-событийного моделирования // International Journal of Open Information Technologies. 2021. Т. 9. №2. С. 70-75.

© Мысягин В.В., 2023

УДК 004.056.55

Орлов Д.Ю.

Нижевартовский государственный университет
г. Нижевартовск, Россия

ЦИФРОВАЯ СТЕГАНОГРАФИЯ И ЕЁ ПРИМЕНЕНИЕ

Задача защиты информации от несанкционированного доступа решалась во все времена на протяжении истории человечества. Уже в древнем мире выделилось два основных направления решения этой задачи, существующие и по сегодняшний день: криптография и стеганография. Целью криптографии является скрытие содержимого сообщений за счет их шифрования. В отличие от этого, при стеганографии скрывается сам факт существования тайного сообщения [1, с. 3].

Слово «стеганография» имеет греческие корни и буквально означает «тайнопись». Исторически это направление появилось первым, но затем во многом было вытеснено криптографией. Тайнопись осуществляется самыми различными способами. Общей чертой этих способов является то, что скрываемое сообщение встраивается в некоторый безобидный, не привлекающий внимание объект. Затем этот объект открыто транспортируется адресату. При криптографии наличие зашифрованного сообщения само по себе привлекает внимание противников, при стеганографии же наличие скрытой связи остается незаметным [1, с. 3].

Сегодня же мы поговорим о современном виде этой науки – цифровой стеганографии. Но перед этим рассмотрим основные понятия, используемые далее.

Стеганографической (скрытой) передачей информации именуют процессы, реализующие способы передачи информации, при которых вероятна передача добавочной информации в структуре данных, представленных в цифровом виде и используемых в качестве контейнера, особенно из-за их избыточности.

Контейнером (скрывающим объектом) называют такие цифровые данные, применение избыточности коих позволяет передавать добавочную информацию, не обличая прецедента передачи. Контейнер, не имеющий добавочной информации, именуют пустым, в противном случае – стего-контейнером. Системы управления, связи и безопасности, сумма технологий и методов встраивания и извлечения добавочной информации без обнаружения нарушения целостности контейнера пользователем, позволяет утверждать о формировании стеганографического канала передачи информации.

Стеганографической системой (стегосистемой) именуют комплекс способов и методов передачи и приема пустого контейнера, функционирующих взаимосвязано со средствами и методами, используемыми для создания тайного канала передачи информации.

Стеганографическая стойкость (стойкость) определяется вероятными усилиями (атаками) нарушителя в отношении стегосистемы. В зависимости от целей системы стеговложения под скрытностью подразумевается надежность такого к прецеденту обнаружения, либо надежность к попытке удаления или уничтожения стеговложения, если факт его существования не является секретом для нарушителя

Итак, рассмотрим способы скрытой передачи данных самыми различными способами. На рисунке 1, в виде дерева, обозначены и классифицированы по типу стего-контейнера методы сокрытия данных в разных видах информации. Будь это аудио файлы или текстовая информация, или видео файлы. Всё это может быть контейнером для использования в стеганографической системе.



Рис. 1. Классификация цифровой стеганографии (<https://clck.ru/34po8f>)

Одни из методов на рисунке 1 был использован при выполнении курсовой работы. Это был метод сокрытия в наименьших значащих битах (НЗБ). С помощью него можно скрыть данные например, в изображении, что и было показано в курсовой работе. При этом сам контейнер никак не будет отличаться от исходного, но иметь внутри скрытую информацию и его можно будет передать в любом не вызывающем подозрения виде. Но у этого метода есть и минусы. Если специально не усложнять алгоритм сокрытия, то НЗБ легко подвергается стего-атаке, для выяснения наличия в изображении скрытых данных. Также при сжатии контейнера можно потерять данные внутри.

Далее давайте на примерах узнаем, как используют методы цифровой стеганографии. Начнём с примера стего-водяного знака (СВЗ) для фото. Примером будет служить алгоритм аутентификации подлинности изображений Canon Original Data Security в цифровых камерах Canon. Камеры Canon делали запись информации Original Decision Data (ODD), которая заносилась в блок EXIF файлов JPG и содержала зашифрованные данные о дате и координатах съёмки. У компании было устройство OSK-E3 (специализированный софт и смарт-карта), где проверялась подлинность фотографий (рис. 2).



Рис. 2. Устройство OSK-E3 (<https://clck.ru/34po7g>)

Оно широко использовалось юристами, криминалистами и для подготовки улик к предъявлению в суде. Использовалось это устройство до обнаружения уязвимости компанией Elcomsoft и её работником Дмитрием Скляровым. Компания Elcomsoft объявила 30 ноября 2010 года, что система шифрования, используемая камерами Canon для защиты изображений и метаданных exif от подделки, несовершенна и не может быть исправлена.

Это поставило под сомнение, например, улики в суде, которые предоставляли, как подлинно проверенные с помощью OSK-E3. Ведь до того, как уязвимость была публично объявлена, неизвестно сколько ещё людей о ней знали и могли использовать для своей выгоды.

В качестве примера цифровой стеганографии текстовой информации приведём пример одного сайта. Это сайт, посвященный разбору треков рэп-исполнителей, применял два вида апострофов в тексте для того, чтобы использовать азбуку Морзе. Апострофами в текстах была зашифрована простая фраза: Red handed («с поличным»). Таким образом редакторы защищали от копирования уникальный контент ресурса.

Правда стеганографическая стойкость такого метода сомнительна. Ведь если текст копируется с изменением стиля, апострофы могут измениться или уйти. И ещё, могут возникнуть и другие факторы, которые повлияют на форматирование исходного текста.

А также, некоторые современные цветные принтеры применяют технологии стеганографии ради затруднения использования их для печати валюты. На каждой отпечатанной странице дополнительно в особом порядке размещается группа точек, диаметром в доли миллиметра. Эта информация включает модель принтера и его серийный номер (рис 3.). Таким образом, впоследствии возможно выявить, на каком оборудовании была отпечатана конкретная купюра и выявить его владельца. К примеру, принтеры HP и Xerox применяют для этой цели группу точек бледно-желтого цвета, заметные лишь на синем фоне с применением оптического увеличения (<https://clck.ru/34poCh>).

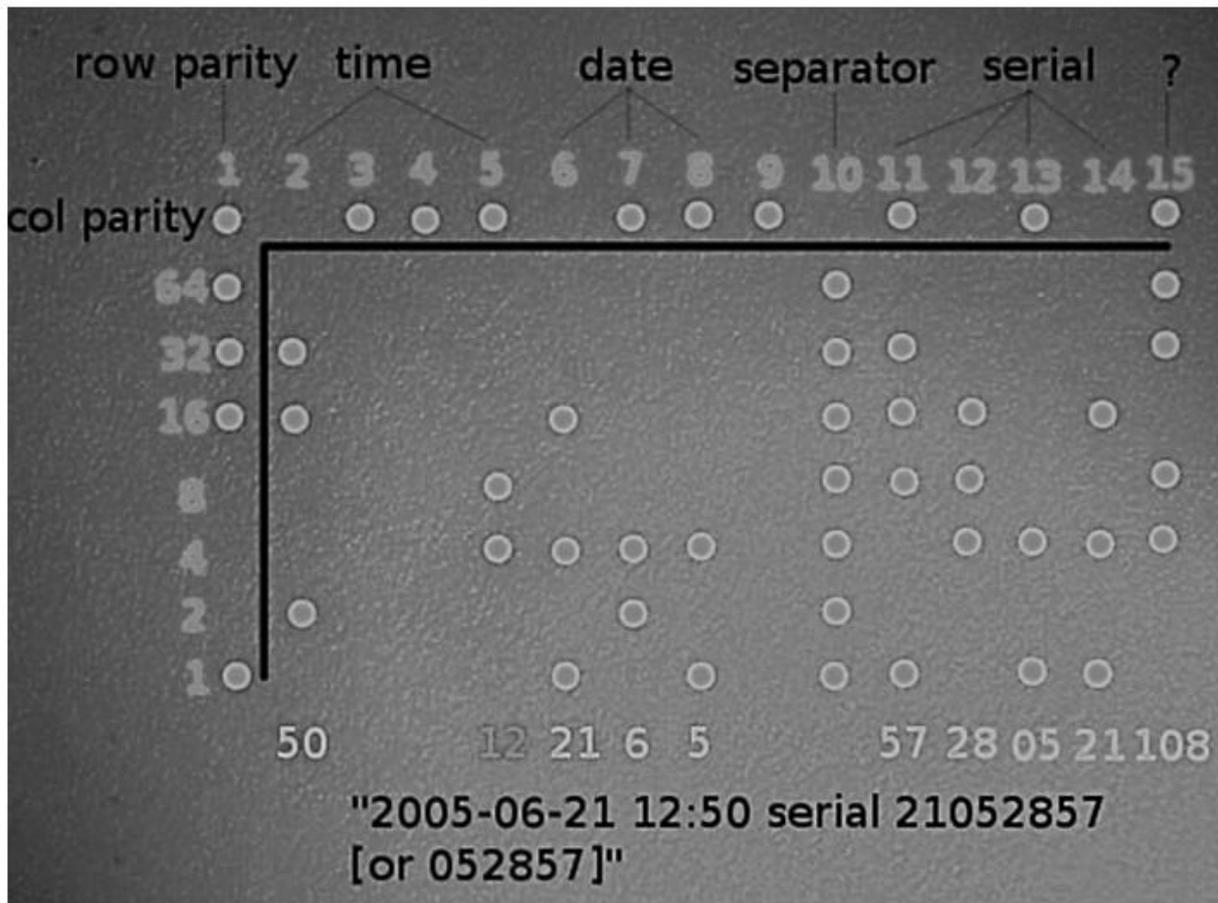


Рис. 3. Положение точек показывает время и дату распечатки, а также серийный номер устройства (<https://clck.ru/34po6T>)

Теперь немного про искусственный интеллект (ИИ). В последние пару лет эта область сделала гигантский скачок в развитии. Появилось множество самых различных, по назначению ИИ. Генерация уникального текста, изображений из предложения, создание 3D-моделей и т.д. Сначала ИИ показывал средний результат, но в результате длительного обучения встал на один уровень с человеческими работами. Из-за этого возник вопрос: как определить, где работа человека, а где работа алгоритма?

Стеганография может решить этот вопрос. Уже сегодня существует электронная цифровая подпись, в которой используются способы стеганографии. Проверка электронной цифровой подписи осуществляется при помощи программного обеспечения (средства ЭЦП) и сертификата открытого ключа автора. С помощью средства ЭЦП можно подписать любой электронный файл, после чего он станет полноценным электронным документом. Если в такой документ в процессе пересылки будут внесены какие-либо правки, пусть даже совсем незначительные, то подмена будет обнаружена. Сертификат открытого ключа содержит персональную информацию о владельце, что позволяет однозначно идентифицировать автора документа.

Но ЭЦП подходит больше к документам. Нас же интересуют изображения, модели, музыка и т.д. Тут нам и пригодится цифровая стеганография. Ведь именно эта область

стеганографии сосредоточена на работе с нужным нам типом данных. Обширный круг способов и методов этой области поможет нам в решении вопроса авторства.

Можно встраивать уникальный ключ, дату и время, и информацию о ИИ на этапе генерации. Или прятать QR-код с ссылкой на ИИ в изображения или видео. После чего написать небольшой сервис по проверке авторства файлов. Это поможет отделить контент ИИ от контента, созданного человеком. Стойкость в таком случае будет определяться типом данных контейнера, в который мы зашиваем информацию.

В заключение хотелось бы сказать, что с развитием технологий будет развиваться и стеганография. Особенно в век компьютерных технологий, когда прогресс не стоит на месте. Цифровая стеганография, как мы видим из примеров, успешно применяется в наше время. И это мы знаем только успешно вскрытые случаи применения. Сколько же на самом деле скрывается информации с помощью стеганографии, нам остается только догадываться.

Будут появляться новые методы по сокрытию цифровых данных, которые будут более стойкими к изменениям. Но также будут и совершенствоваться способы атаки на контейнер, для обнаружения факта сокрытия информации или извлечения самих данных из контейнера.

Литература

1. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография. М.: Солон-Пресс, 2009. 272 с.

© Орлов Д.Ю., 2023

УДК 001:378.1

Полякова А.И., Дробахина А.Н.
Кемеровский государственный университет
г. Новокузнецк, Россия

ФОРМИРОВАНИЕ ЦИФРОВЫХ КОМПЕТЕНЦИЙ ПРЕПОДАВАТЕЛЕЙ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ В ПРОЦЕССЕ РАЗРАБОТКИ ЦИФРОВЫХ ПОРТФОЛИО

Сегодня педагог становится проводником новых цифровых образовательных технологий и формирует метапредметные навыки не только у обучающихся, но и у коллег. Поэтому сформированность компетентности самого педагога, его способность осваивать и применять цифровые технологии и конкретные сервисы в решении педагогических задач, приобретает особое значение.

Результаты исследования цифровых компетенций педагогических работников, полученные исследовательской группой университета Иннополис в 2021-2022 году, показали, что лучше всего сформирована компетенция «Применение цифровых продуктов и цифровых образовательных ресурсов» – педагоги на высоком уровне овладели умениями выбирать цифровые образовательные ресурсы и эффективно их использовать [2].

В то же время исследование показало, что компетенция «Оценка и учебная аналитика» имеет самый низкий результат, а каждый шестой участник исследования имеет нулевой уровень владения этой компетенцией. Исследователи утверждают, что данный навык плохо проработан у преподавателей в части анализа и оценки образовательных результатов [2].

Все вышесказанное позволяет сформулировать проблему исследования, которая заключается в необходимости рассмотрения цифрового портфолио как инструмента личностно-профессионального развития преподавателей профессионального образования, а также как необходимого элемента в формировании цифровой компетенции в области анализа и оценки результатов своего профессионального профиля.

Создание цифрового портфолио отвечает современным требованиям к педагогу, поскольку он должен не только знать свой предмет и владеть методикой его преподавания, но и демонстрировать результаты, достигнутые в обучающей, воспитательной, творческой, самообразовательной деятельности.

В этой связи можно вспомнить слова А.С. Gingerich, что «магия» портфолио педагога заключается в том, что оно позволяет продемонстрировать сложность процесса обучения и профессионального роста» [6, с. 2].

Рассмотрим дефиницию «цифровое портфолио педагога».

Barrett Н.С., Hartell E., Bastidas J.A. отмечают, что первые определения цифрового портфолио появились в середине 1990-х. [4].

Цифровое портфолио берет свое начало от бумажного, однако имеет ряд отличительных особенностей.

В широком смысле «портфолио» – это коллекция документов и материалов, иллюстрирующих теорию, развитие и достижения педагога в результате непрерывного процесса самообразования и самооценки [5, с. 24].

В работах отечественных авторов (Е.А. Безызвестных [1], О.Г. Смоляниновой [3]) цифровое портфолио рассматривается как совокупность результатов профессиональной деятельности, предназначенное для последующего анализа и комплексного оценивания личностного и профессионального развития, созданное с использованием средств информационно-коммуникационных технологий, ресурсов и сервисов телекоммуникационных сетей.

Портфолио педагога должно демонстрировать рост, развитие и эффективность преподавания, это в свою очередь ориентирует на использование интерактивности и создания мультимодального и гипертекстового текста, которым можно поделиться в сети [8, с. 7].

Цифровые портфолио представляют собой инструмент, который позволяет архивировать, систематизировать, отбирать информацию и, кроме того, позволяют размышлять над ней в той мере, в какой педагоги решают, какие документы представлять, поэтому это можно понимать, как составные части цифровой компетентности.

Аргументы в пользу разработки педагогом своего цифрового портфолио:

- цифровое портфолио может педагогу организовать свою работу, улучшает управление временем;
- цифровое портфолио позволяет делиться опытом между коллегами;
- публикации цифровых портфолио поощряют других педагогов разрабатывать свои материалы в формате портфолио;
- если педагоги смогут продемонстрировать рост, основанный на их собственной самооценке и результатах их записей и документов в портфолио, они будут готовы осуществлять контроль над своей деятельностью, поддерживать престиж педагога и требовать соответствующего признания своей профессии обществом.

Разработка портфолио – непростая задача.

Сложность разработки цифрового портфолио связана с большим объемом работ, выполняемых на каждом этапе, тщательным планированием, подбором содержания, продумыванию дизайна, выбором среды разработки [7, с. 15].

Этапы разработки цифрового портфолио представлены на рисунке.

Структура портфолио с точки зрения технической реализации должна удовлетворять следующим показателям:

- реализация различных цифровых материалов (видео, аудио, инфографика) демонстрирующих профессиональный опыт и знания;
- соответствие визуальных элементов портфолио профессиональному контенту педагога;
- согласованность цвета, шрифтов и других вариантов оформления на всех страницах портфолио;
- наличие навигации внутри страниц;

- функционирование гиперссылок, кнопок, фотогалереи и других технических функций;
- качество фото и видео контента;
- возможность отображения портфолио в разных браузерах и на разных устройствах [8, с. 42].

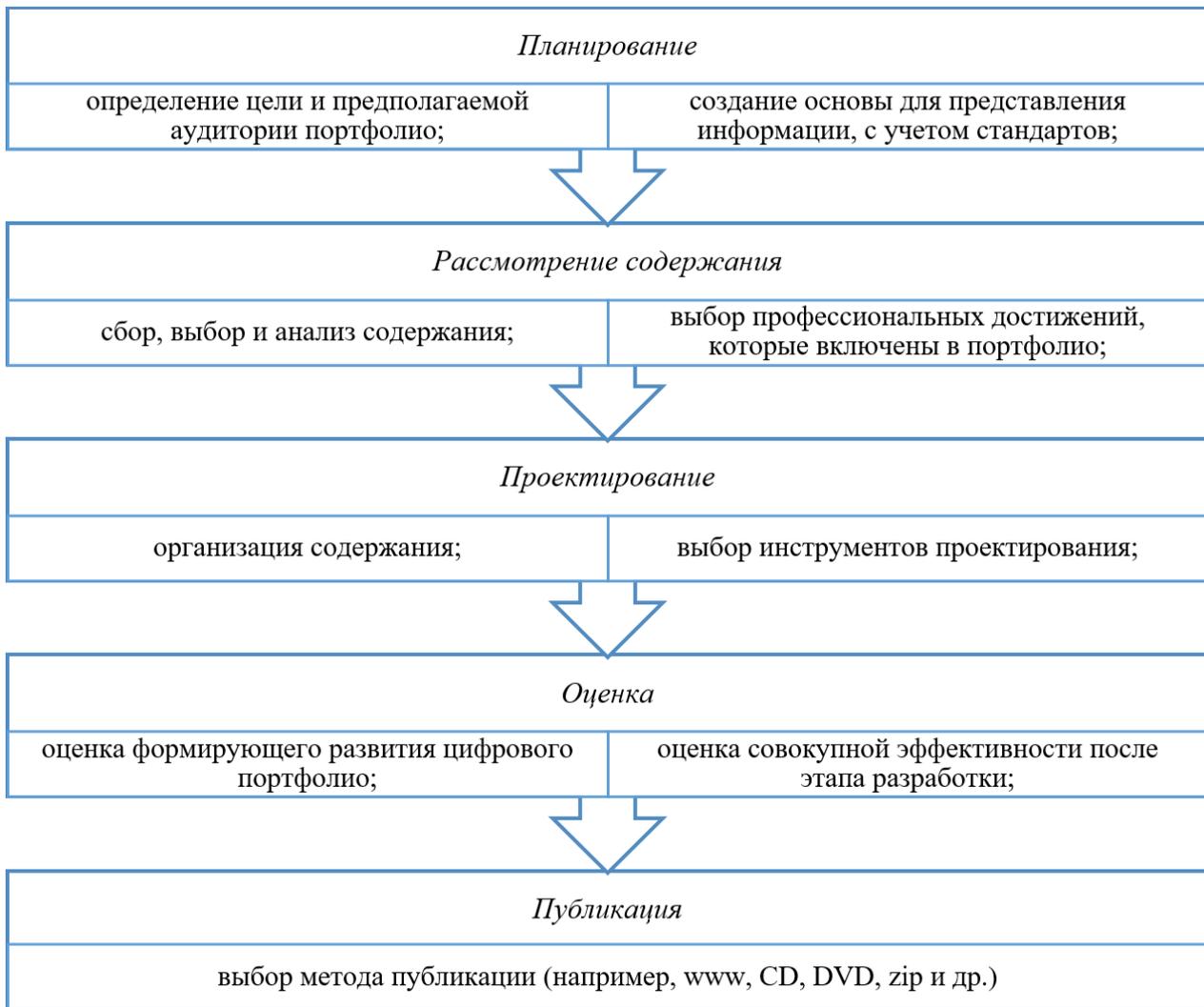


Рис. Этапы разработки цифрового портфолио

На основании результатов конкурса «Дорога к успеху», организованного ГАПОУ «Кузбасский колледж архитектуры, строительства и цифровых технологий» (г. Новокузнецк) в 2022 и 2023 гг., можно сделать вывод, что некоторые педагоги неверно представляют себе содержание портфолио, недостаточно владеют технологиями для разработки портфолио, используют лишь базовые функциональные возможности выбранной среды.

Отметим, что выбор конкретной среды для разработки цифровых портфолио является отдельной, непростой задачей.

На основании результатов вышеупомянутого конкурса можно сделать вывод, что самой популярной средой для разработки портфолио педагога по-прежнему остается PowerPoint. На втором месте – онлайн сервис Google Sites, а на третьем – WIX.

Таблица 1

Выбор сред для разработки портфолио

Название среды для создания портфолио	Количество человек	Доля, %
PowerPoint	129	83,25%
Google Sites	12	7,74%
WIX	7	4,52%
Portfolio42	3	1,94%
Tilda	2	1,29%
Blogger	1	0,65%
Итого	155	100%

Сравним популярные у педагогов среды для разработки портфолио по пяти категориям: простота использования, настройка, объем хранилища, поддержка пользователей.

Таблица 2

Возможности платформ, используемых для представления цифрового портфолио педагога

ПРОСТОТА ИСПОЛЬЗОВАНИЯ	
Google Sites	Имеет интуитивный и простой интерфейс для перетаскивания различных компонентов для создания страниц. Два типа меню: вертикальное и горизонтальное. Имеет интерфейс перетаскивания, но для некоторых пользователей может с первого раза показаться сложной для применения. Предлагаются шаблоны по умолчанию, которые можно использовать для редактирования самой страницы. После того, как элемент встроен, меню самого элемента предоставляют различные возможности для редактирования или адаптации к странице.
PowerPoint	Программа предоставляет возможность легко и быстро редактировать размещаемый материал. Привычный для многих педагогов инструмент разработки презентаций. PowerPoint включает в себя разнообразные информационные объекты: текст, аудио-визуальный материал, которые могут быть наиболее эффектно представлены за счет разнообразных возможностей управления ходом показа слайдов и анимации за счет триггеров.
WIX	Интерфейс представляет собой перетаскивание. Шаблоны предлагают элементы по умолчанию, которые можно использовать для редактирования. После того, как элемент встроен в страницу, меню самого элемента предоставляют различные возможности для редактирования или адаптации к странице.
НАСТРОЙКА	
Google Sites	Допускают ограниченное количество вариантов оформления страниц; организация проста. Возможно встраивать видео и изображения.
PowerPoint	Программа позволяет как работать с нуля, так и при редактировании шаблонов. Сохранение в нескольких форматах: Презентация PowerPoint (ppt), Демонстрация PowerPoint (pps), Текстовый документ (pdf), Фильм (wmv), Картинки (jpeg).
WIX	Предлагает большое количество шаблонов, классифицированных по типологии страниц, большому разнообразию шрифтов и большому спектру цветов.
ОБЪЕМ ХРАНИЛИЩА	
Google Sites	Объем 100 Мб. Для образовательного домена до 100 Гб.
PowerPoint	Документы педагога встраиваются как изображения, объемные документы можно встраивать как гиперссылки (необходимо использовать облачное хранилище).
WIX	Объем 500 Мб, не требует покупки для дополнительного объема. Предлагают галерею для просмотра всех загруженных личных документов.
ТЕХНИЧЕСКАЯ ПОДДЕРЖКА	
Google Sites	Учебное пособие и форум для обсуждения проблем.

PowerPoint	Служба поддержки Microsoft позволяет обратиться к практическим статьям, видео и учебным материалам по разработке презентаций.
WIX	Справочный центр с большим количеством вопросов и форумов, предлагаются интерактивные видеоуроки.
ОФОРМЛЕНИЕ ЦИФРОВОГО ПОРТФОЛИО ПЕДАГОГА	
Google Sites	Веб-дизайн для нескольких устройств. Профессиональное развитие можно задокументировать по мере заполнения новыми материалами. Открытый доступ в сети Интернет. Можно выстроить обратную связь.
PowerPoint	Не размещается автоматически в браузерах. Доступно портфолио в ограниченного доступе. Невозможно выстроить обратную связь. Отсутствие возможности ознакомиться с большими документальными материалами (разработки уроков, научные статьи, сценарии внеурочных мероприятий).
WIX	Веб-дизайн для нескольких устройств. Профессиональное развитие можно задокументировать по мере заполнения новыми материалами. Открытый доступ в сети Интернет. Можно выстроить обратную связь.

Подводя итоги, отметим, что существует необходимость повышения уровня владения цифровыми технологиями преподавателей в контексте создания и оформления цифрового портфолио, знакомства с возможностями сред для их разработки.

Литература

1. Безызвестных Е.А. Электронный портфолио как средство формирования ИКТ-компетентности будущих педагогов-тьюторов: Дис. ...канд. пед. наук. Красноярск, 2019. 289 с.
2. Отчет о результатах оценки цифровых компетенций педагогических работников // Университет Иннополис. 2021. 35 с.
3. Смолянинова О.Г. Электронный портфолио – технология обучения в течение всей жизни и ресурс профессионального развития: контекст профессионального стандарта педагога // Образование через всю жизнь: непрерывное образование в интересах устойчивого развития. 2015. №13. С. 87-90.
4. Barrett H.C. Researching electronic portfolios and learner engagement: The REFLECT initiative // Journal of Adolescent and Adult Literacy. 2007. №50 (6). С. 436-449.
5. Bastidas J.A. The Teaching Portfolio: A Tool to Become a Reflective Teacher // English Teaching Forum. 1996. №3-4, July/October. С. 20-28.
6. Gingerich A.C. The Digital Teaching Portfolio Handbook // Scholarship and professional work. LAS, 2006. С. 1-3.
7. Hopper T., Hong F., Sanford K., Monk D. What Is a Digital Electronic Portfolio in Teacher Education? A Case Study of Instructors' and Students' Enabling Insights on the Electronic Portfolio Process // CJLT/RCAT. Т. 44(2). С. 1-20.
8. Pujolà J.T. El portafolio digital en la docencia universitaria // Primera edición: noviembre de. 2019. 146 с.

© Полякова А.И., Дробахина А.Н., 2023

УДК 004.92(075.8)

Пуличев М.П., Вайнер Л.Г.

Тихоокеанский государственный университет
г. Хабаровск, Россия

СРАВНИТЕЛЬНЫЙ АНАЛИЗ СПОСОБОВ ПОСТРОЕНИЯ ЭВОЛЬВЕНТЫ ПРИ СОЗДАНИИ ГРАФИЧЕСКОЙ МОДЕЛИ ЗУБЧАТОГО КОЛЕСА

В машиностроении самое широкое применение получили передаточные механизмы с зубчатыми колесами в качестве ведущего и ведомого звеньев. Зубчатое колесо представляет собой диск с нарезанными на его внешней конической или цилиндрической поверхности зубьями. За счет зацепления зубьев смежных зубчатых колес происходит передача вращательного движения.

Наибольшее распространение в настоящее время имеют зубчатые колеса, в которых поверхность зуба имеет эвольвентный профиль [1]. Эвольвента – это кривая, образованная траекторией движения точки, принадлежащей прямой, перекатывающейся без скольжения по окружности. Данная прямая называется производящей прямой, а окружность, по которой она перекатывается – основной окружностью. Точное изображение эвольвенты сопряжено с известными трудностями, существует несколько способов ее построения [2]. Каждый из способов построения эвольвенты характеризуется определенными погрешностями.

В данной работе дан сравнительный анализ двух способов построения эвольвентной поверхности зуба при 3D моделировании зубчатой передачи: упрощенный – с применением заменяющих дуг окружности и точный – с использованием принципов и алгоритма построения эвольвентной кривой. 2D и 3D построения и моделирование производились в графической среде nanoCAD (<https://clck.ru/34pqmR>).

Для обоих способов заданными являются следующие параметры зубчатого колеса: D – диаметр делительной окружности; d_1 – диаметр вершин зубьев, d_2 – диаметр впадин зубьев, D_b – диаметр основной окружности, m – модуль, S – делительная окружная толщина зуба, φ – угол наклона линии зуба на делительном цилиндре.

Порядок построения эвольвенты в графическом редакторе nanoCAD упрощенным способом:

1. Изобразим в рабочем пространстве окружности диаметрами D , d_1 , d_2 (рис. 1).
2. Построим основную окружность. Основная окружность – это окружность, по которой «перекатывается» прямая линия, своим концом вычерчивая эвольвенту.
3. Отметим точку A на окружности вершин зубьев.
4. На прямой, соединяющей точки A и O , отметим точку B , находящуюся на основной окружности.
5. Разделим расстояние AB на 4 части и отметим точкой C значение равное $1/4 AB$ (рис. 1).
6. От точки C проведем касательную к основной окружности и в точке касания отметим точку D .

7. Разделим расстояние DC на четыре части и отметим точкой E полученное значение равное 1/4 DC от точки D в сторону точки C (рис. 2, а).



Рис. 1. Параметрические окружности зубчатого колеса с указанием характерных точек построений: 1 – окружность вершин зубьев; 2 – делительная окружность; 3 – основная окружность; 4 – окружность впадин

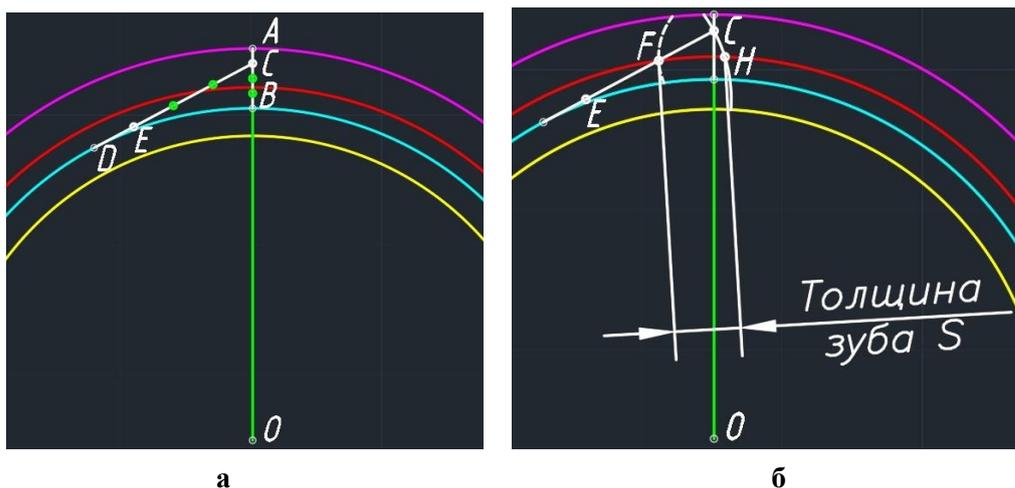


Рис. 2. Предварительные построения (а) и определение части бокового контура зуба (б)

1. Проведем дугу окружности с центром в точке E, проходящую через точку C. Место пересечения с делительным диаметром - точка H. Это будет часть одной стороны зуба.

2. Проведем дугу окружности с центром в точке H радиусом, равным толщине зуба S. Толщина зуба вычисляется по формуле

$$S = \frac{\pi m}{2}.$$

Место пересечения с делительным диаметром отметим точкой F. Эта точка находится на другой стороне зуба (рис. 2, б).

3. Проведем ось симметрии через центр O и середину расстояния FH (рис. 3, а).

4. Линия профиля зуба, отображённая зеркально относительно этой оси и будет второй стороной зуба (рис. 3, а).

5. Построим переходную кривую, сопрягающую окружность впадин и эвольвенту. Упрощенно, это будет дуга окружности с радиусом $R = 0.4m$ (рис. 3, б).

Полный профиль зуба, построенный с помощью упрощенного способа с использованием заменяющих дуг окружностей, приведен на рис. 3, б.

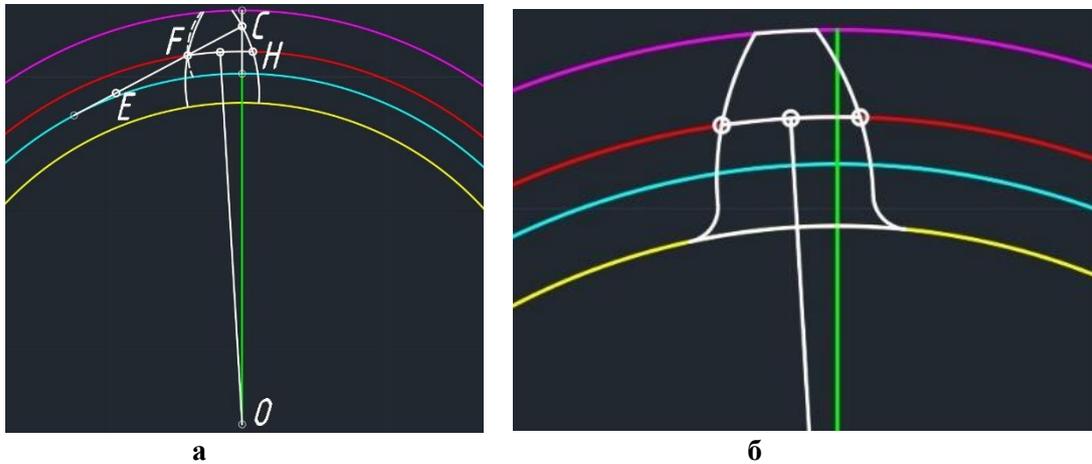


Рис. 3. Построение кривых боковых контуров (а) и профиля с переходными кривыми (б)

Рассмотрим второй, наиболее точный способ построения профиля зуба.

Последовательность точного построения эвольвенты в графическом редакторе nanoCAD классическим способом:

1. Проведем окружности диаметрами D , d_1 , d_2 , построим основную окружность, проведем отрезок AO (рис. 1).

2. Из точки A проведем касательную к основной окружности и точку касания обозначим M . Построим отрезок OM (рис. 4, а).

3. Повернем отрезок AO вокруг точки O на угол $inv(\alpha_a)$ радиан, отметим отрезок OM_0 . Длина отрезка AM и длина дуги основной окружности MM_0 должны быть равны. $inv(\alpha_a) = tga - a$ [рад];

4. Поделим отрезок AM и дугу MM_0 на одинаковое количество равных частей (на рис. 4, а объекты поделены на 5 частей). Точки деления обозначим цифрами.

5. Командой *ПОВЕРНУТЬ* точку 1 на отрезке AM переместим на эвольвенту. Параметры команды: *Объекты* – указать точку 1 на отрезке AM ; *Базовая точка* – указать точку O ; *Опорный угол* – указать последовательно точку O , затем – точку M ; *Угол поворота* – указать точку 1 на дуге MM_0 .

Результат выполнения команды показан на рисунке 4, а.

6. Повторим пункт 5 для остальных точек отрезка AM . В параметрах команды будут меняться *Объекты* и *Угол поворота*.

7. Соединим точки $0, 1', 2', 3', 4'$ плавной кривой линией. Направление линии в точке $4'$ должно указывать на точку O , как показано на рис. 4, а.

8. Построим переходную кривую, сопрягающую собой окружность впадин и эвольвенту. Упрощенно, это будет окружность с радиусом $R = 0.4 * m$;

9. Проведем отрезок OB из центра O в точку пересечения эвольвенты и делительной окружности. Повернем отрезок OB вокруг точки O на угол $\gamma=S/R$, где S – делительная окружная толщина зуба. Проведем ось симметрии OC через середину отрезка BB' .

10. Командой *ЗЕРКАЛО* отобразим левую сторону зуба (рис. 4, б).

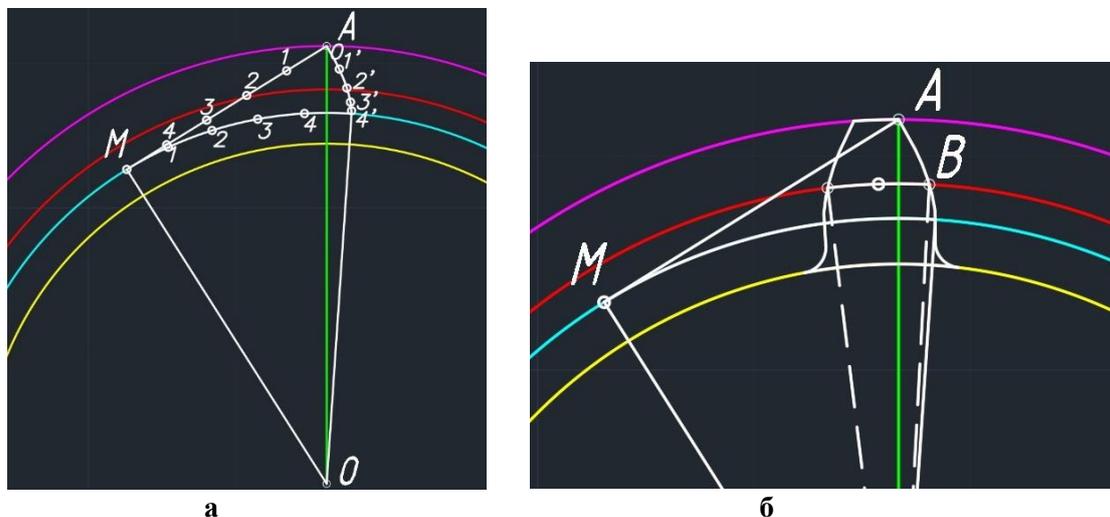


Рис. 4. Точное построение кривых боковых контуров (а) и профиля с переходными кривыми (б)

Полный профиль зуба, полученный с помощью точного способа построения эвольвентных кривых, приведен на рисунке 4, б.

Сопоставим на одном изображении профили зубьев, полученные рассмотренными выше способами (рис. 5).

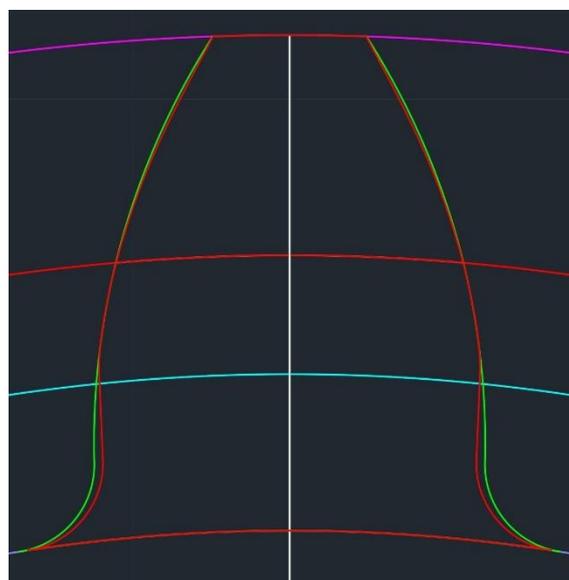


Рис. 5. Сопоставление профилей зубьев, построенных двумя способами

В результате сравнительного анализа можно сделать следующие выводы:

- профили головки зуба практически совпадают;
- основные отличия можно отметить в нижней части профиля зуба – зуб, построенный точным способом, имеет меньшую толщину и больший радиус скругления у основания.

Таким образом, можно отметить, что для построения зубчатого колеса, которое будет использоваться в геометрической модели механизма, рациональнее будет использовать упрощенный метод, так как построения таким методом выполняются быстрее, а геометрические погрешности метода визуально незаметны.

На рисунке 6 в качестве примера изображена построенная в графическом редакторе nanoCAD 3D модель зубчатого колеса с профилем зубьев, построенным упрощенным способом.

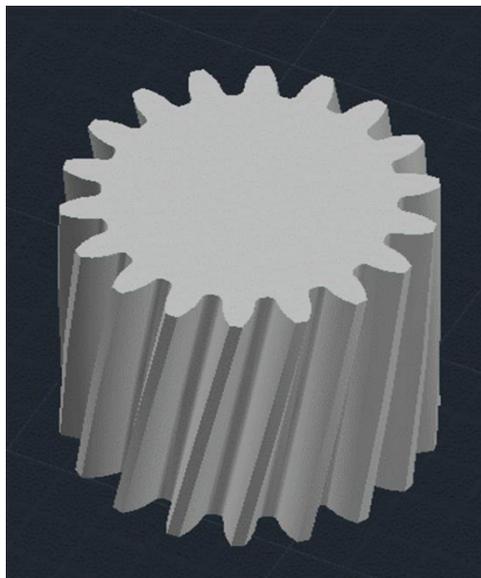


Рис. 6. 3D модель зубчатого колеса с профилем зубьев упрощенной геометрии

При инженерном анализе зубчатого зацепления, решении контактных задач и при углубленном сравнительном анализе геометрической формы зубьев целесообразно использовать метод точного построения.

Литература

1. Артоболевский И.И. Теория механизмов и машин. М.: Альянс, 2008. 640 с.
2. Левитская О.Н., Левитский Н.И. Курс теории механизмов и машин. М.: Высш. шк., 1985. 279 с.

© Пуличев М.П., Вайнер Л.Г., 2023

ПРИМЕНЕНИЕ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ЗАДАЧИ КЛАССИФИКАЦИИ УРАВНЕНИЙ В ОБЩЕМ ВИДЕ

Использование сверточных нейронных сетей (ИНС) для распознавания образов на изображениях является в настоящее время современным трендом. В последнее время ИНС всё лучше и лучше справляются с задачей классификации. Об этом можно судить по результатам работ [1; 2; 4]. Лысцов Н.А., Мартышкин А.И. в своей работе описывают применение искусственных нейронных сетей (ИНС) для определения кошек и собак на изображениях [4]. Друки А.А. использует тот же подход для выделения и распознавания автомобильных номеров на изображениях со сложным фоном [3].

Подготовка обучающих данных.

Для обучения ИНС было сформировано 29 классов графиков, ключевым моментом при составлении классов было отличие графиков друг от друга в таблице. Представлены список классов функций, которые ИНС обучалась определять.

Таблица

Класс для ИНС	Формула для генерации графиков	Значения коэффициентов
0	$y = kx + b$	$k, b \in R$
1	$y = ax^2 + bx + c$	$a > 0; b, c \in R$
2	$y = ax^2 + bx + c$	$a < 0; b, c \in R$
3	$y = ax^3 + bx^2 + cx + d$	$a < 0; b, c, d \in R$
4	$y = ax^3 + bx^2 + cx + d$	$a > 0; b, c, d \in R$
5	$y = ax^4 + bx^3 + cx^2 + dx + e$	$a > 0; b, c, d, e \in R$
6	$y = ax^4 + bx^3 + cx^2 + dx + e$	$a < 0; b, c, d, e \in R$
7	$y = \frac{k}{x-a} + b$	$a, b \in R, k > 0$
8	$y = \frac{k}{x-a} + b$	$a, b \in R, k < 0$
9	$y = c * x - k + b$	$k, b \in R$ $c > 0$
10	$y = c * x - k + b$	$k, b \in R$ $c < 0$
11	$y = b * x - a $	$a > 0, b > 0$
12	$y = b * x - a $	$a > 0, b < 0$
13	$y = \frac{k}{ x-a - c} + b$	$a, b \in R, k > 0, c > 0$
14	$y = \frac{k}{ x-a - c} + b$	$a, b \in R, k > 0, c < 0$
15	$y = \frac{k}{ x-a - c} + b$	$a, b \in R, k < 0, c > 0$
16	$y = \frac{k}{ x-a - c} + b$	$a, b \in R, k < 0, c < 0$
17	$y = \frac{k}{ x-a } + b$	$a, b \in R, k > 0$

18	$y = \frac{k}{ x - a } + b$	$a, b \in R, k < 0$
19	$y = \left \frac{k}{x} - a \right $	$a \in R, k > 0$
20	$y = \left \frac{k}{x} - a \right $	$a \in R, k < 0$
21	$y = k x - b + ax + c$	$k > 0, a > 0$ $b, c \in R$
22	$y = k x - b + ax + c$	$k > 0, a < 0$ $b, c \in R$
23	$y = k x - b + ax + c$	$k < 0, a > 0$ $b, c \in R$
24	$y = k x - b + ax + c$	$k < 0, a < 0$ $b, c \in R$
25	$y = k * x * x - a + b$	$b \in R, k > 0, a > 0$
26	$y = k * x * x - a + b$	$b \in R, k < 0, a > 0$
27	$y = \frac{k x - b}{x}$	$b \in R, k > 0$
28	$y = \frac{k x - b}{x}$	$b \in R, k < 0$

Для обучающих и тестовых данных были следующие требования. При обучении было нужно на каждый класс по 1000 тренировочных изображений, тестовых и контрольных графиков по 100. Формат изображений растровый, а размер составлял 640 на 480 пикселей. График должен был построен по точкам с равномерным шагом. На изображении не должно быть ничего лишнего, например, осей координат, легенды и прочего, то есть только белый фон и чёрные точки графика. Наконец, в пределах одного класса все экземпляры графиков должны быть уникальными.

На данный момент не существует какой-либо базы данных, содержащей наборы изображений графиков функций всех нужных классов. Также нету средств, генерирующих множество различных графиков, получающихся из одной формулы (при разных значениях параметров). Поэтому было решено создать собственный генератор обучающих изображений.

При создании программы-генератора использовались следующие технологии. Код программы написан на языке программирования **Си**, а само создание изображения графика выполняется с помощью свободно распространяемой утилиты **gnuplot**. В генераторе используются системные вызовы и утилиты операционной системы **Linux**. Далее на рисунке 1 приведена общая, схема устройства программы-генератора.

Исходный код генератора разделён на два модуля. При запуске программы пользователь передаёт конфигурационный файл и имя директории, куда нужно сохранить результат работы. Поэтому один из модулей отвечает за обработку пользовательского запроса, а во втором модуле реализованы 29 функций, каждая из которых отвечает за создание изображения графика соответствующего класса. Конфигурационный файл пользователя обрабатывается построчно. В очередной строке указывается номер класса для построения, а вторым числом количество экземпляров этого класса. Далее в цикле нужное число раз вызывается функция для построения изображений.

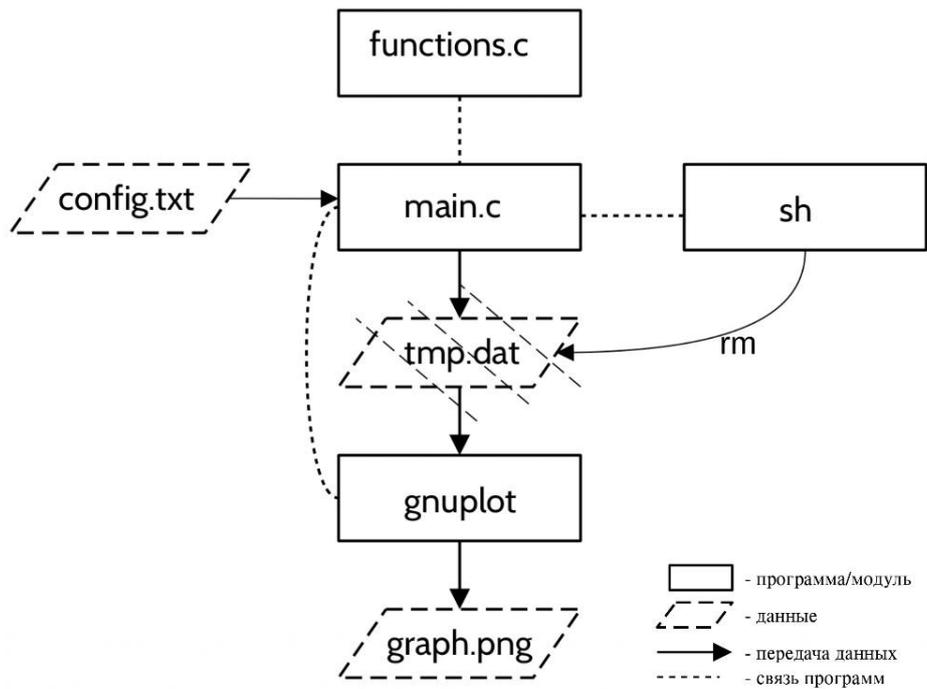


Рис. 1. Общая схема генератора графиков. Генератор состоит из двух модулей: **main**, **functions**

Процесс формирования изображения графика происходит так. В каждой процедуре построения сначала задаётся видимая область графика – диапазон отображаемых значений по координатам X и Y, определяются коэффициенты, которые фигурируют формуле, задающей функцию (см. табл.). Каждый коэффициент принимает случайное значение из заранее заданного промежутка. Для вычисления случайных значений используется формула (1), обеспечивающая равномерное распределение на промежутке (a,b), где **rand()** стандартная функция языка Си, а **RAND_MAX** константа – максимальное возможное псевдослучайное число.

$$R = a + (b - a) \left(\frac{\text{rand}}{\text{RAND_MAX}} \right), \quad (1)$$

Далее в зависимости от текущего выбранного шага построения вычисляется число точек для построения графика на данной области определения. После этого заполняются массивы точек $X[i]$, $Y[i]$ графика функции, которые потом записываются во временный файл в текстовом формате. Наконец, на стандартный поток ввода утилиты **gnuplot** передаётся команда для построения графика по данным из только что сформированного файла. Так получается одно изображение. Данный цикл повторяется нужное число раз, а затем считывается следующая строка из пользовательского файла. После того, как построены все изображения данного класса выполняется вызов утилиты ОС **Linux rm** для очистки директории от более не нужных текстовых файлов.

Нужно отметить, что с самого начала создаётся директория для сохранения результатов. Уже в ней для каждого класса своя поддиректория, в которой содержатся пронумерованные изображения графиков этого класса.

Примеры обучающих данных.

Представлены несколько изображений, взятых из одной из обучающих выборок. На рисунках 2 и 3 приведены изображения графика функции класса 2 из таблицы. Видно, что при одном том же шаге построения все экземпляры графиков в рамках данного класса получают уникальными. Это достигается за счёт наличия случайных коэффициентов в выражениях для функций, коэффициенты вычисляются по формуле (1).

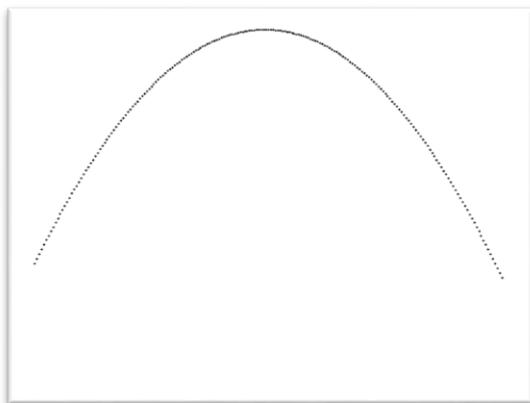


Рис. 2. Класс 12, шаг построения: 0.1

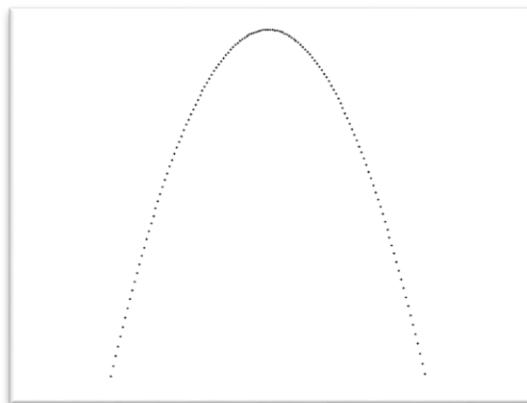


Рис. 3. Класс 12, шаг построения: 0.1

Далее показаны экземпляры графиков разрывной функции, которые относятся к классам 17, 18 таблицы. Рисунок 4 это график функции класса 17, а рисунок 5 функции класса 18.

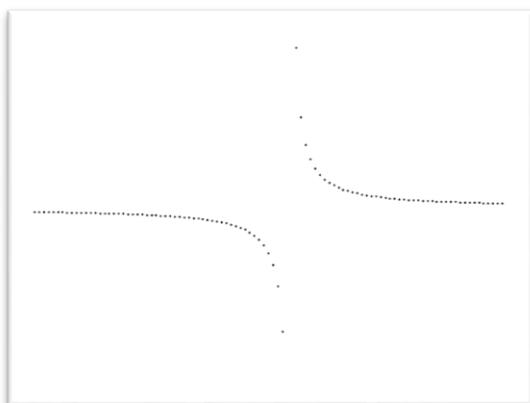


Рис. 4. Класс 17, шаг построения: 0.1

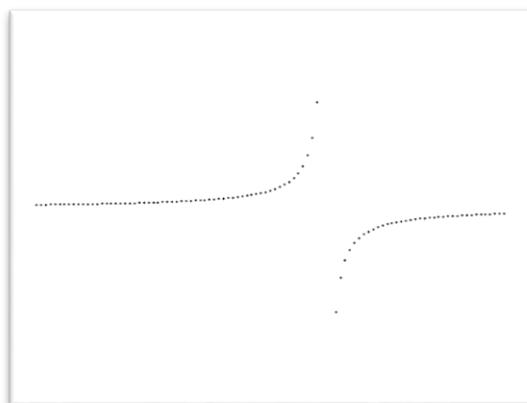


Рис. 5. Класс 18, шаг построения: 0.1

После формирования обучающего набора данных. Была сконструирована и обучена ИНС с использованием библиотеки keras. Структура сети состояла из 5 сверточных слоев, с размером окна 3x3. После каждого сверточного слоя стоял слой max-pooling – на котором выбирается максимальное значение из соседних. После сверточных слоев шёл слой Flatten – который связывает сверточные слои с полносвязными. В завершении было 2 полносвязных слоя, на крайнем полносвязном слое было 29 нейронов, по количеству классов функций.

Для подбора весовых коэффициентов ИНС были установлены следующие параметры обучения:

1. В качестве функции активации использовалась функция softmax;
2. В качестве функции потерь была выбрана: categorical crossentropy;
3. В качестве функции оптимизации RMSprop.

Скорость обучения была адаптивной, и начиналась $10e-3$ до $10e-4$. Обучение продолжалось в течении 13 эпох.

Результаты обучения и тестирования ИНС.

Обучение проходило в течении 13 эпох, с двумя изменениями скорости обучения. На крайней эпохе потери составили $3.2885e-11$ на этапе обучения и 0.0150 на этап валидации для контрольного набора данных.

После обучения ИНС была использована для проверки точности обучения на тестовой выборке. Итого из 2900 данных используемых для проверки эффективности обучения ИНС неверно классифицировала только 3 изображения, графики этих функций представлены на рисунке 6.

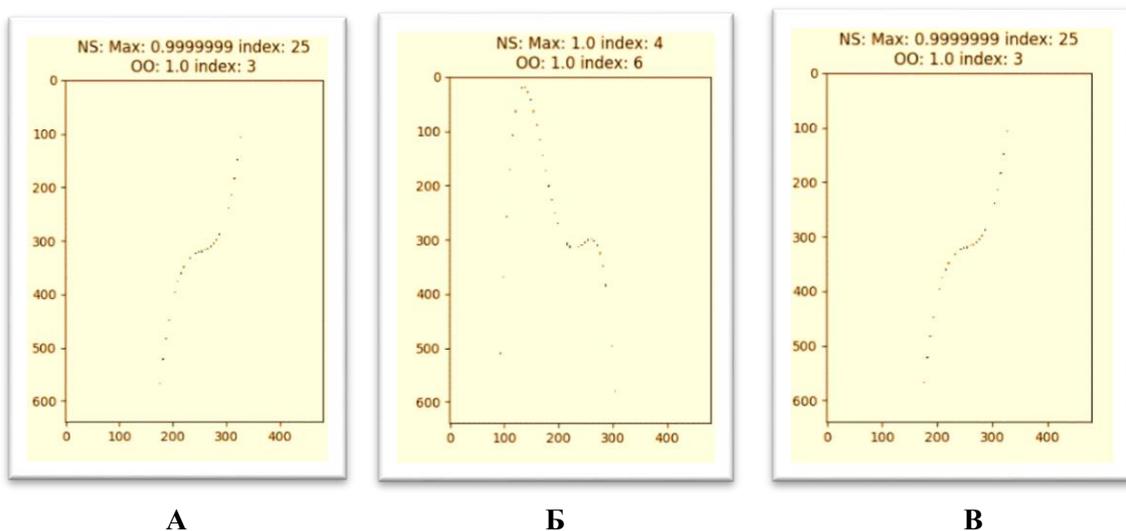


Рис. 6. Изображения которые не верно были определены ИНС для 25 и 3 класса функций

Как видно из результатов проверки обученной ИНС, сеть способна с точностью 99.9% верно определять класс функции по ее графическому образу.

Результаты обучения ИНС для идентификации уравнений по их графическим образам демонстрируют хорошие показатели, которые позволяют предположить, что использовать сверточные нейронные сети можно для определения не только каких-то реальных физических объектов. Но и для определения абстрактных образов – математических объектов. Такой подход можно использовать для первичного определения типа функции для задач аппроксимации или интерполяции.

Литература

1. Андриенко И.С. Классификация изображений с помощью нейронных сетей // Постулат. 2023. №1.

2. Деменкова Т.А., Шпиева Е.В. Построение моделей свёрточной нейронной сети для задач распознавания образов // Научно-технический вестник Поволжья. 2020. №5. С. 76-79.

3. Друки А.А. Применение сверточных нейронных сетей для выделения и распознавания автомобильных номерных знаков на изображениях со сложным фоном // Известия ТПУ. 2014. №5. С. 85-91.

4. Лысцов Н.А., Мартышкин А.И. Нейронные сети: применение и перспективы // Студенческий научный форум 2019: Мат-лы XI международной студенческой научной конференции. 2019. № 3-2. С. 35-38.

5. Пальмина К.С. Распознавание образов с использованием Tensorflow // Современная наука: актуальные вопросы, достижения и инновации: XV Международная научно-практическая конференция (г. Пенза, 05 сентября 2020 г.). Пенза, 2020. С. 81-85.

© Смородинов А.Д., Рассадин А.А., 2023

УДК 519.688

Сорокин А.О., Слива М.В.

Нижевартовский государственный университет,
г. Нижневартовск, Россия

РАЗРАБОТКА ЛИЧНОГО КАБИНЕТА ПОЛЬЗОВАТЕЛЯ ИНТЕРНЕТ-ПРОВАЙДЕРА НА ПЛАТФОРМЕ ANDROID

Современный пользователь мобильного интернета ожидает от своего интернет-провайдера максимальной удобства и функциональности в работе с услугами. Личный кабинет интернет-провайдера для мобильных устройств является одним из основных инструментов, позволяющих получить доступ к нужным услугам и контролировать их использование.

Личный кабинет пользователя интернет-провайдера – это онлайн-сервис, который предоставляется клиентам провайдера для управления своими услугами, проверки счетов, оплаты, изменения тарифов, отслеживания состояния сети и других функций. Через личный кабинет пользователь может получить доступ к своим персональным данным, а также получить помощь и поддержку со стороны провайдера. Личный кабинет обычно защищен паролем или логином и паролем, чтобы гарантировать безопасность данных пользователя. Личный кабинет является важным элементом взаимодействия между пользователем и провайдером, обеспечивая удобство и простоту использования интернет-услуг.

Веб версия личного кабинета состоит из двух частей: верхняя – содержит в себе элементы навигации (услуги, платежи, помощь, настройки, выход) и центральная – содержит в себе переключаемый контейнер (рис. 1)

Навигация в мобильном приложении состоит из четырех фрагментов, комбинирующих элементы меню.

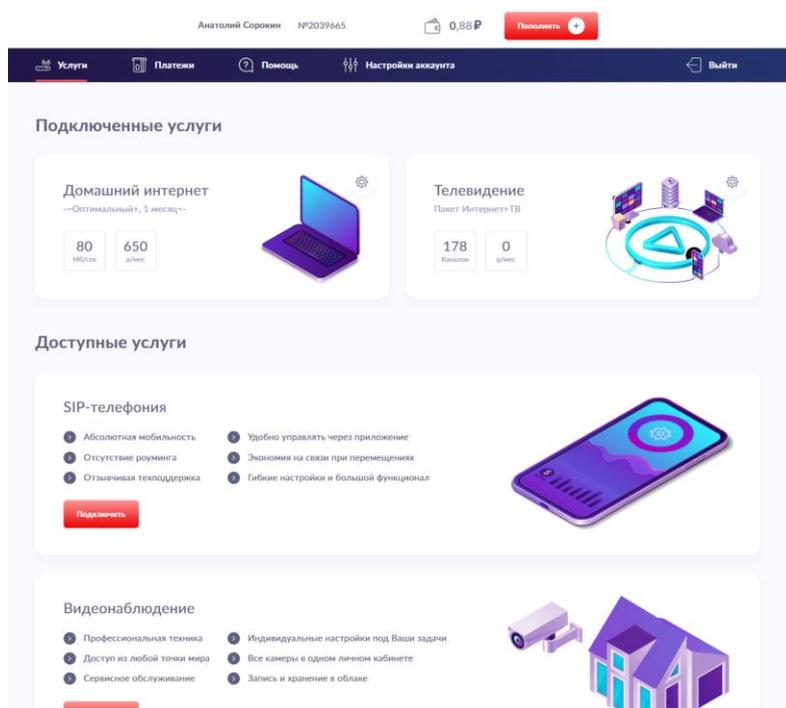


Рис. 1. Веб версия личного кабинета

В главном фрагменте отображается основная информация о подключенных услугах пользователя с возможностью перехода к ней для редактирования, например, при выборе услуги домашний интернет появляется возможность сменить тарифный план, приостановить интернет, подключить/отключить статический ip-адрес, и доступные услуги с возможностью оставить заявку на подключение (рис. 2).

В фрагменте «платежи» отображается текущий баланс, рекомендуемый платеж для пополнения, история пополнений и списаний с информацией об услуге, стоимости и дате операции (рис. 3).



Рис. 2. Главный (домашний) фрагмент



Рис. 3. Фрагмент «Платежи»

В фрагменте «настройки» отображается номер договора, номер телефона, адрес электронной почты пользователя и сам договор (рис. 4). Есть возможность смены номера телефона и почты.

В фрагменте «помощь» отображается номер телефона службы технической поддержки, номера телефона и адреса абонентских отделов компании в разных городах.

В мобильной платформе Android визуальные компоненты представления сильно зависят от состояния приложения и регулируются его жизненным циклом (рис. 5), который в данный момент представлен пользователю (<https://clck.ru/34pvs2>).

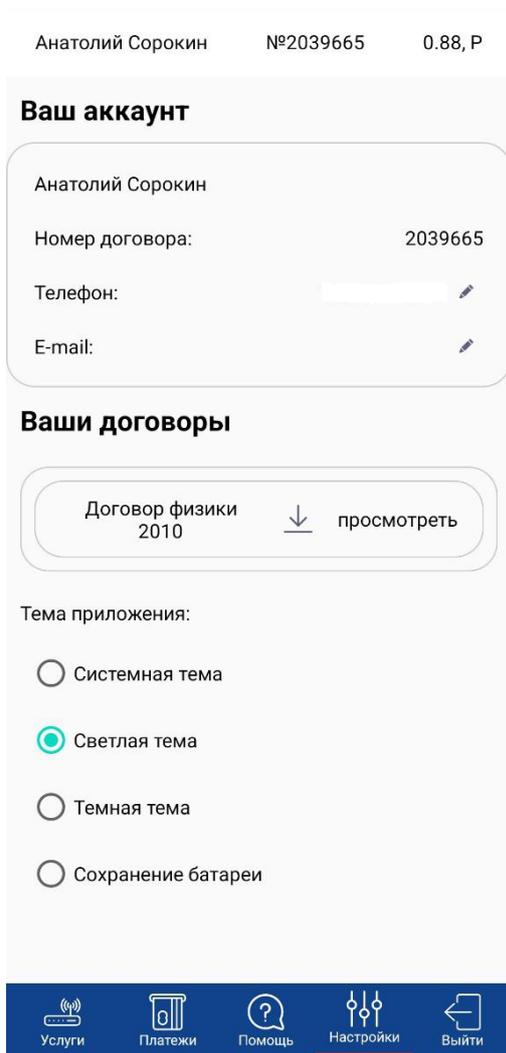


Рис. 4. Фрагмент «Настройки»

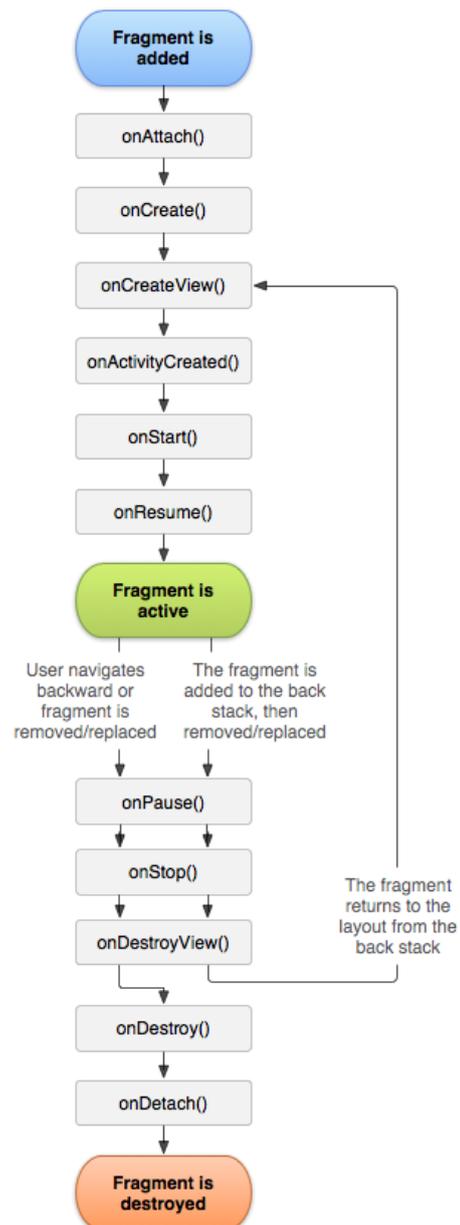


Рис. 5. Жизненный цикл фрагмента

При смене фрагментов система вызывает метод `onDestroy()` у отображаемого представления и `onCreate()` следующего. При этом фрагменты не сохраняют свое состояние, так как каждый раз создается новый экземпляр. Как следствие вся информация теряется, а разделы загружаются заново, что приводит к повышенной нагрузке устройства.

Для решения этой проблемы было принято решение о разделении интерфейса и логики, для этого использовалась библиотека Моху. MVP (Model-View-Presenter) – это паттерн программирования графических интерфейсов. В нём приложение делится на три компонента (рис. 6) (<https://clck.ru/34pvtld>):

- Model (Модель) работает с данными, проводит вычисления и руководит всеми бизнес-процессами;
- View (Вид или представление) показывает пользователю интерфейс и данные из модели;

- Presenter (Представитель) выступает связующим звеном между моделью и видом (<https://clck.ru/34pvwz>).

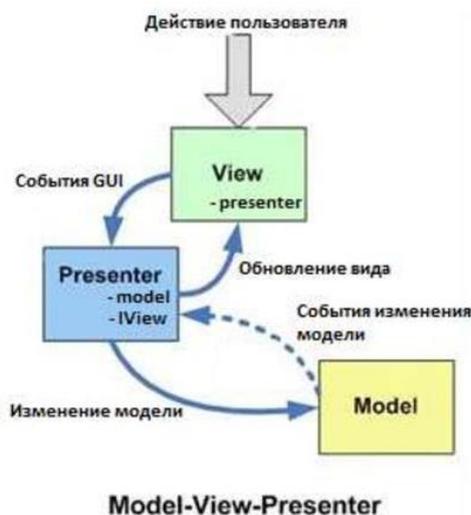


Рис. 6. Структура паттерна MVP

Несмотря на некоторые ограничения, связанные с доступностью интернета и возможностями личных кабинетов для мобильных устройств, они являются важной частью работы интернет-провайдеров. В будущем, вероятно, эти кабинеты станут еще более удобными и функциональными, и будут продолжать служить удобным инструментом для пользователей мобильного интернета.

Применение паттерна MVP приводит к разбиению приложения на три компонента - модель, представление и представитель, что упрощает процесс разработки и тестирования, а также облегчает модификацию и поддержку приложения в будущем. Кроме того, разделение приложения на три компонента позволяет более четко определить обязанности каждого компонента, что способствует более эффективному взаимодействию между ними. Например, модель отвечает за логику приложения, данные и их обработку, представление – за отображение данных и взаимодействие с пользователем, а представитель – за управление взаимодействием между моделью и представлением. Это также облегчает распределение задач между участниками команды разработки и ускоряет процесс разработки приложения в целом.

© Сорокин А.О., Слива М.В., 2023

МОДЕЛИРОВАНИЕ СИСТЕМ ДЛЯ ПРОАКТИВНОГО УПРАВЛЕНИЯ ВЫЧИСЛИТЕЛЬНЫМИ КОМПЛЕКСАМИ

Поддержка вычислительных комплексов в критических инфраструктурах (банковских, телекоммуникационных, промышленных) в рабочем состоянии (с гарантированным объемом вычислительных ресурсов) является актуальной проблемой современного цифрового общества. Деградация производительности, отказ или сбой оборудования не позволяют своевременно обрабатывать информацию, оперативно выполнять банковские и другие операции, что может привести к серьезным последствиям (финансовым потерям, крупным авариям и т. п.).

Одним из перспективных подходов к решению проблемы является использование интеллектуальных систем, реализующих схему: сбор данных – построение прогностической модели – проактивное принятие решений – выполнение действий. Рядом авторов были разработаны системы проактивного управления, способствующие эффективному принятию оперативных решений [1-4].

Однако, в лабораторных условиях трудно производить разработку соответствующих систем (без взаимодействия с самой критической инфраструктурой). Поэтому актуальной является проблема создания модельной системы, которая позволяла бы исследователям использовать её для разработки систем проактивного управления.

В работе предлагается один из возможных вариантов решения указанной проблемы.

Концептуальное моделирование критической информационной системы.

Рассмотрим концептуальную модель критической информационной системы, которая описывает её базовые элементы и значимые их параметры. Она может быть описана в виде следующего кортежа:

$$System = (Clients \cup Balancers \cup APPs \cup DBs, Links), \quad (1)$$

где $Clients = \{Client_i\}$ – множество клиентов системы. Это могут быть, как пользовательские устройства, так и другие внешние системы;

$Balancers = \{Balancers_j\}$ – множество балансировщиков, которые распределяют запросы клиентов и внешних систем по сервисам. Сервисы внутри системы так же могут общаться между собой через балансировщики;

$APPs = \{APP_k\}$ – множество прикладных сервисов системы;

$DBs = \{DB_l\}$ – базы данных системы (могут быть любых типов).

$Links = \{Link_m\}$ – множество двусторонних временных связей, возникающих во время общения между элементами системы.

Элементы $Client_i, Balancers_j, APP_k, DB_l, Link_m$ в свою очередь также могут быть представлены кортежами:

$Client_i = (Protocol_i, Profile_i)$, где соответственно $Protocol_i$ – определяет спецификацию, а $Profile_i$ – профиль взаимодействия (запросы, их параметризация, частота и т. д.) для i -го клиента;

$Balancers_j = (Protocol_j, RPS_j, Throughput_j)$, где RPS_j – количество запросов в секунду, а $Throughput_j$ – пропускная способность (Mbps) для j -го балансировщика. Предполагается, что балансировщик поддерживает все протоколы взаимодействий и задержка на балансировщике не значительна по сравнению с временем формирования ответа на запрос.

$APP_k = (Compute_Modules_k, Soft_Service_k)$ – набор инстансов прикладного ПО, где $Compute_Modules_k$ – множество вычислительных модулей, а $Soft_Service_k$ – тип прикладного сервиса системы, который используется на вычислительных модулях $Compute_Modules_k$;

$DB_l = (Compute_Modules_l, Soft_Service_DB_l)$ – описывает набор инстансов баз данных, обслуживающих один тип инстансов прикладного сервиса, где $Compute_Modules_l$ – множество вычислительных модулей, $Soft_Service_DB_l$ – тип программного обеспечения баз данных, который используется в вычислительных модулях $Compute_Modules_l$. Это могут быть любые (не обязательно реляционные) базы данных;

$Link_m = ((P_{mn}, P_{mo}))$ – множество двухсторонних временных связей между элементами системы, осуществляемых общение через конкретные открытые порты.

$Compute_Modules = \{Compute_Module_c\}$ – множество вычислительных модулей системы, в котором каждый вычислительный модуль также можно описать соответствующим кортежем:

$Compute_Module = (CM_type, CM_cpu_limit, CM_cpu_perf, CM_ram_limit, CM_ram_perf, CM_Storages, CM_net_throughput)$, где $CM_type \in \{ "physical\ server", "logical\ or\ hw\ partition", "virtual\ machine", "container" \}$ – тип модуля, CM_cpu_limit – количество, доступное модулю процессоров (CPU или vCPU). CM_cpu_perf – максимальная производительность одного процессора модуля, – доступный объем ОЗУ модуля (Gib), – максимальная производительность ОЗУ модуля (определяется пропускной способностью и временем доступа к ячейкам памяти), $CM_net_throughput$ – максимальная пропускная способность модуля (Gbps);

$CM_Storages = \{CM_Storage_p\}$ – модули хранения, доступные вычислительному модулю. Каждый модуль хранения в свою очередь определяется кортежем:

$CM_Storage = (CM_Storage_type, CM_Storage_capacity, CM_Storage_perf)$, где $CM_Storage_type \in \{ "local", "external" \}$ – определяет тип подключения дискового модуля к вычислительному модулю.

При этом вариант *local* описывает локальные дисковые ресурсы (HDD, SSD). А компонента *external* – внешние по отношению к вычислительному модулю системы ресурсы хранения данных. Последние могут быть подключены с помощью различных устройств ввода-вывода, поддерживающих различные протоколы блочного (iSCSI – SCSI over Ethernet, Fiber Channel Protocol – SCSI over Fiber Channel, NVMe-oF – NVMe over Fiber Channel), файлового (CIFS, NFS) и объектного (S3) доступа;

$CM_Storage_capacity$ – объем дискового модуля (Gib);

$CM_Storage_perf$ – производительность модуля можно определить, как кол-во операций ввода-вывода в секунду IOPS (при минимальном времени отклика) или пропускной способностью (GBps), для различных профилей нагрузки.

Профиль нагрузки определяется процентом операций чтения (%Read), размером блоков данных (Kib) и распределением, которое показывает какие размеры блоков присутствуют в запросах, какова их доля и задержки по ним в суммарном потоке запросов.

$Soft_Service = \{Soft_Service_k\}$ – типы прикладного сервиса системы, которые используются в вычислительных модулях системы;

$Soft_Service_DB = \{Soft_Service_DB_l\}$ – типы программного обеспечения баз данных, которые используются в вычислительных модулях;

В информационных системах для интеграции различных сервисов часто используются такие элементы, как брокеры сообщений (н-р, RabbitMQ, IBM WebSphere MQ, ActiveMQ Artemis, Apache Kafka и др.). В нашем случае эти сервисы не выделены в отдельный класс, так как по сути их можно отнести или к множеству сущностей APPs или DBs, если данный функционал реализован на уровне баз данных (н-р, Oracle Advanced Queuing).

В структуру модели явно не включены: телекоммуникационное оборудование (Switches, Routers), системы управления ресурсами и различные системы контроля периметра информационной системы. Все это может ограничивать доступ к системе из вне и между компонентами по портам и протоколам (Firewalls). А также провести к более глубокой инспекции обмена на прикладном уровне (WAF – Web Application Firewalls) и необходимости выполнять анализ обмена на уровнях OSI Model 3 и 4 (IPS – Intrusion Prevention System), отслеживая аномалии и делая проверки для известных уязвимостей и векторов атак на основе базы сигнатур и установленных политик.

В структуру так же явно не включены, системы мониторинга и логирования, антивирусной защиты, шифрования, а так же иные технологические сервисы и системы (LDAP, DNS, сервис резервного копирования и восстановления, CI/CD системы, сервис синхронизации времени и т. д.). Предполагается, что данные сервисы корректно настроены и их работа не сказывается на работоспособности рассматриваемой системы.

Постановка задачи.

С учетом описанной модели ставится следующая задача.

Разработать систему концептуально соответствующую, описанной ранее модели критической информационной системы (1), удовлетворяющей следующим критериям (2):

1. Система может функционировать на рабочей станции, ноутбуке или виртуальной машине с ресурсами не более 4 ядер CPU и 8 GiB ОЗУ;
2. Множество APPs представлено компактным прикладным web приложением, которое способно обрабатывать внешние запросы;
3. Инстанс прикладного web сервиса функционирует в вычислительном модуле $Compute_Module$, который обладает определенным лимитом по ресурсам CPU (другие лимиты так же возможны, но не обязательны);

4. Система имеет возможность масштабирования в рамках определенных выше ресурсов, т. е. в процессе работы под нагрузкой можно менять количество экземпляров прикладного web приложения за счет увеличения или уменьшения количества вычислительных модулей. Управление масштабированием доступно из программного кода и в ручном режиме;

5. По каждому вычислительному модулю системы *Compute_Module* собирается метрика утилизации %CPU, частота сбора метрик $\sim 2 \text{ с}^{-1}$. Данные сохраняются в файлы формата csv с указанной частотой. При добавлении вычислительных модулей, по ним автоматически добавляется сбор статистики;

6. В системе должен быть модуль реактивного управления масштабированием системы, который функционирует по следующему алгоритму: после каждого получения метрик утилизации %CPU по работающим вычислительным модулям *Compute_Module*, на которых запущен прикладной web сервис, рассчитывается процент средней утилизации $\overline{\%CPU}_N$ по количеству (N) работающих модулей, if $\overline{\%CPU}_N > 50\%$, то система автоматически добавляет еще один вычислительный модуль *Compute_Module* с прикладным web сервисом, if $\overline{\%CPU}_N < 20\%$, то система автоматически выключает один вычислительный модуль *Compute_Module* с прикладным web сервисом, if $\overline{\%CPU}_N$ находится в интервале от 20% до 50%, то система действий по изменению конфигурации не производит;

7. После каждого изменения состояния системы (при масштабировании) должен включаться стабилизационный режим, при котором система реактивного управления не выполняет действий по изменению конфигурации прикладного web сервиса в течении заданного промежутка времени;

8. Множество *Balancers* представлено сервисом балансировки запросов между экземплярами прикладного web приложения;

9. При добавлении нового экземпляра web приложения последний автоматически добавляется в балансировку нагрузки, а при выключении экземпляра web приложения - автоматически исключается из балансировки нагрузки;

10. Множество *Clients* представлено нагрузочной системой, в которой можно задать профиль нагрузки для прикладной системы по запросам и различные профили пользователя. При этом, профиль нагрузки можно задавать с помощью функции или заранее подготовленных данных, а так же посредством работы в web и cli клиентах. Нагрузочная система должна в online отображать статистику по проводимому тестированию и иметь возможность сохранения отчетов со статистикой по конкретным запросам в виде RPS, Response Time, ошибкам при выполнении запросов и количеству пользователей;

11. Нагрузочная система функционирует в вычислительном модуле *Compute_Module* и может поддерживать распределенную конфигурацию экземпляров, работающих одновременно в нескольких вычислительных модулях *Compute_Module*;

12. При нагрузочном тесте с максимальной нагрузкой потребление ресурсов CPU вычислительным модулем *Compute_Module*, в котором функционирует нагрузочная система, не должно быть больше 1 ядра CPU;

13. Наличие экземпляра базы данных возможно, но не обязательно;

14. Система поддерживает IaC (Infrastructure as a Code) модель;

Реализация системы.

Для решения поставленной задачи использовался алгоритм, включающий следующие основные этапы:

1. Определение ключевых функциональных блоков системы. Подготовка верхнеуровневой схемы (необязательно);
2. Определение возможных вариантов реализации каждого блока с учетом требований к системе;
3. Анализ возможных реализаций с учетом следующих критериев (наличие готовых компонентов, которые можно использовать для построения функционального блока, простота реализации, время реализации) и выбрать вариант реализации прототипа системы;
4. Создание прототипа системы;
5. Качественная оценка соответствия прототипа критериям задачи в процессе тестирования;
6. Если критерии не выполняются, то вернуться к шагу 2;
7. Выполнить необходимое число итераций (шаги 2-6), пока прототип не будет соответствовать критериям задачи;

Далее были определены функциональные блоки системы и построена верхнеуровневая схема прототипа (рис. 1):

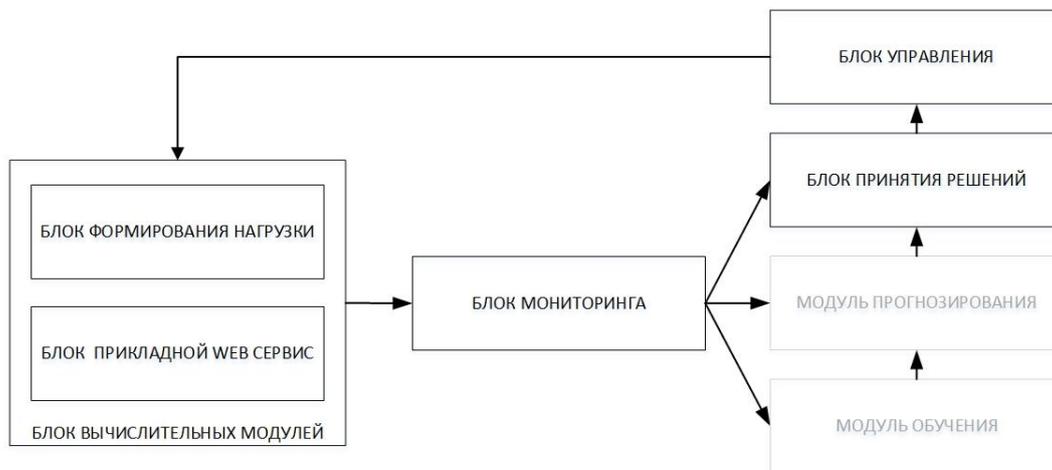


Рис. 1. Функциональная схема прототипа системы

Модули прогнозирования и обучения добавлены опционально, чтобы показать один из вариантов дальнейшего развития системы с учетом реализации проактивного управления в будущем.

Блок вычислительных модулей. Данный функционал является базовым, т.к. на его основе строится то или иное решение. В работах [1-4] решение задач проактивного управления основывалось на виртуальных машинах, которые разворачивались в различных публичных облаках (Amazon, Google и др.) или частных облаках или центрах обработки данных. При этом для управления виртуальными машинами (создание, запуск, остановка, удаление) использовались возможности облачного сервиса или систем управления центра обработки

данных. Виртуальные машины - хорошее решение в части изоляции приложений, но они требуют больше вычислительных ресурсов, т. к. для каждой виртуальной машины нужны ресурсы для операционной системы. Кроме того нужно использовать модуль, который бы повторял функционал облачного сервиса по управлению виртуальными машинами (создание, запуск, остановка, удаление), а так же выполнял настройку операционных систем в виртуальных машинах, установку, настройку и управление прикладным web сервисом и поддерживал модель IaC.

В результате исследований было принято решение в качестве вычислительных модулей использовать контейнеры. Они являются менее ресурсозатратными, более простыми в реализации, соответствуют базовым критериям (2).

Рассмотрено несколько решений: Docker Compose (<https://clck.ru/34pwGu>), Kubernetes Cluster (<https://clck.ru/34pwHr>). Вариант с Docker Compose оказался более простым, при этом использование Kubernetes Cluster в варианте с одной worker node (н-р, Minikube (<https://clck.ru/34pwL7>)) так же возможно.

В итоге было принято решение использовать в прототипе системы Docker Compose для блока вычислительных модулей. Данное решение позволяет создавать вычислительные модули (Docker контейнеры), управлять ими, собирать метрики утилизации контейнеров. Есть готовые библиотеки, (н-р, Docker SDK for Python (<https://clck.ru/34pwLJ>)) для работы с Docker API Engine.

Балансировка нагрузки по контейнерам с прикладным web приложением осуществляется встроенными в Docker Compose средствами по имени сервиса.

Конфигурация сервисов и ресурсов описывается в YAML файле.

Блок прикладной web сервис. В виду экономии ресурсов, стабильности работы, простоты настройки и эксплуатации в качестве прикладного web приложения было решено реализовать микросервис, основанный на популярном высокопроизводительном минималистичном web фреймворке echo.labstack (<https://clck.ru/34pwLe>), написанном на высокоуровневом языке Go (<https://go.dev>).

Блок формирования нагрузки. В качестве генераторов нагрузки рассматривалось 2 варианта программного обеспечения: Apache JMeter (<https://jmeter.apache.org>), написанный на Java и Locust (<https://locust.io>), написанный на Python. Apache JMeter является более функциональным, сложным в настройке и более требовательным к ресурсам, при этом Locust обладает меньшим функционалом, прост в настройке, позволяет описывать профили нагрузки в виде классов на Python, менее требователен к ресурсам, поддерживает распределенную конфигурацию инстансов, позволяет задавать профиль пользовательской нагрузки в виде функции или заранее подготовленных данных. В качестве генератора нагрузки для прототипа был выбран Locust.

В процессе отладки решения было обнаружено, что в случае большого количества легких запросов, утилизация контейнера с locust по %CPU заметно превышает утилизацию %CPU контейнеров, в которых располагаются прикладные web приложения. Потребовалось на уровне web приложения добавить дополнительный endpoint (/load), который вызывает код на

Go, с помощью которого, используя параллельные goroutines, достигается требуемое повышение утилизации процессора %CPU. Добавление данного запроса, более ресурсоемкого по CPU, позволило изменить баланс между утилизацией нагружающей и нагружаемой системами. Экспериментально было определено, что прототип системы с лимитом ресурсов, указанных в критериях (2) и сформированным профилем нагрузки, выдерживает до 700-800 активных пользователей. Дальнейшее повышение нагрузки, приводит к пределу по утилизации виртуальной машины, на которой развернут Docker Compose.

Блок мониторинга, принятия решений, управления. Данные блоки были реализованы в виде нескольких модулей на Python. Отдельно стоит отметить, что в работе библиотеки с Docker API Engine есть особенность, которая связана с тем, что API не возвращает в одном запросе статистику по всем контейнерам, как это делает консольная команда `docker stats`. Кроме того, сам запрос выполняется ~ 1 с, т. к. системному демону `docker` необходим определенный интервал времени, чтобы рассчитать соответствующие средние значения метрик. Поэтому для корректного сбора метрик была использована библиотека `joblib` (<https://clck.ru/34pwJv>), которая позволяет реализовать параллельные запросы. С помощью Python был реализован функционал по реактивному масштабированию прикладной web системы при изменении нагрузки и другие требования (см. 2). При этом, при каждом изменении состояния системы, сохраняется статистика в виде csv файлов по состояниям для каждого вычислительного модуля.

Результаты экспериментов.

Стабильность работы системы была проверена на основе тестов, различных по длительности и характеру пользовательской нагрузки.

На рисунке 2 представлен результат работы системы под нагрузкой, когда зависимость количества активных пользователей от времени задавалась аналитической функцией (3), к которой был добавлен случайный шум:

$$y(x) = 100 + 100e^{-\left(\frac{x}{200}-3\right)^2} + 200e^{-\left(\frac{x}{200}-9\right)^2} + 160e^{-\left(\frac{x}{200}-15\right)^2} + 200e^{-\left(\frac{x}{200}-21\right)^2} + 100e^{-\left(\frac{x}{200}-27\right)^2} \quad (3)$$

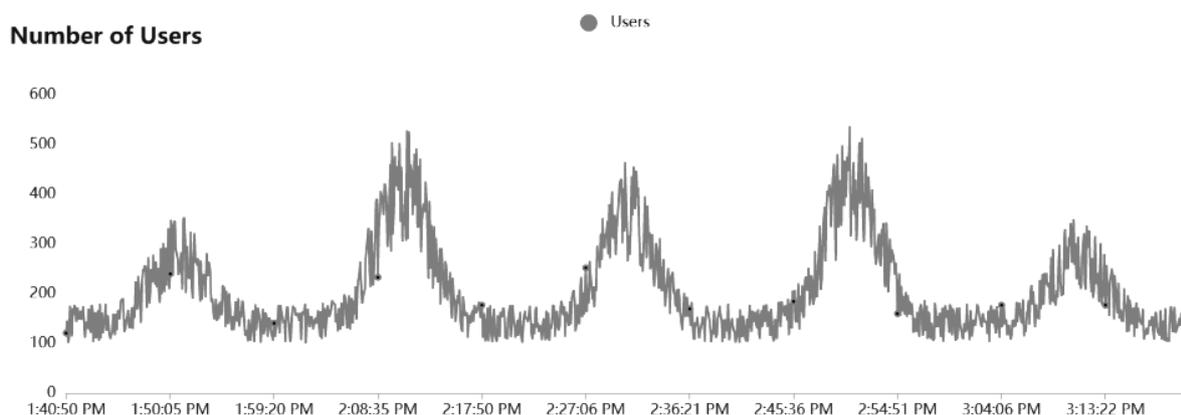


Рис. 2. Нагрузка, заданная аналитической функцией со случайным шумом

На рисунке 3 представлены примеры графиков утилизации %CPU вычислительных модулей для разных состояний прикладной системы под нагрузкой (горизонтальная ось – номер отсчета по сбору метрики):

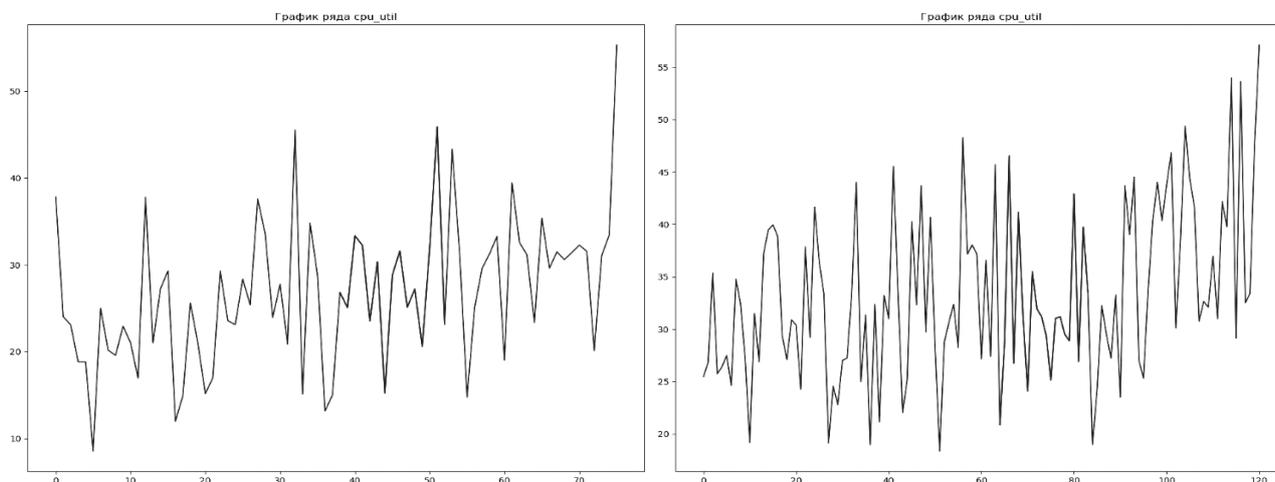


Рис. 3. Утилизация %CPU вычислительных модулей

Таким образом, в результате проведенных исследований была разработана прототип системы, удовлетворяющий критериям задачи (2). Прототип был построен на основе модели критической информационной системы, которую можно использовать для разработки и тестирования интеллектуальной проактивной системы управления. Система позволяет моделировать внешнюю нагрузку и получать метрики вычислительных модулей прикладных систем в виде рядов данных, которые можно использовать в прогностических моделях. Предложенная система легко масштабируется и может быть перенесена практически без изменений в Kubernetes Cluster.

Литература

1. Khan N., Elizondo D.A., Deka L., Molina-Cabello M.A. Fuzzy Logic applied to System Monitors // IEEE Access. 2021. Vol. 9. P. 56523-56538. <https://doi.org/10.1109/ACCESS.2021.3072239>
2. Lakew E.B., Birke R., Pérez J.F., Elmroth E., Chen L.Y. Smalltail: Scaling cores and probabilistic cloning requests for web systems // 2018 IEEE International Conference on Autonomic Computing (ICAC). IEEE, 2018. С. 31-40. <https://doi.org/10.1109/ICAC.2018.00013>
3. Persico V., Grimaldi D., Pescape A., Salvi A., Santini S. A fuzzy approach based on heterogeneous metrics for scaling out public clouds // IEEE Transactions on Parallel and Distributed Systems. 2017. Vol. 28. №8. P. 2117-2130. <https://doi.org/10.1109/TPDS.2017.2651810>
4. Tran D., Tran N., Nguyen G., Nguyen B.M. A proactive cloud scaling model based on fuzzy time series and SLA awareness // Procedia Computer Science. 2017. Vol. 108. P. 365-374. <https://doi.org/10.1016/j.procs.2017.05.121>

РАСПОЗНАВАНИЕ ТЕКСТОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

На сегодняшний день существуют программы, способные распознавать печатный текст с точностью в 99,9%. Но все еще существуют проблемы с определением данных рукописного текста. Связаны данные трудности с тем, что символы рукописного в отличие от печатного имеют более динамичную природу. Одинаковые символы могут иметь различный размер, форму, наклон и т. д. Также сложности при распознавании могут создавать различные декоративные элементы, используемые при написании символа, лишние пересечения букв, а также шумы на носителе текста (пятна на бумаге).

Проблему разнообразия почерков решает тот факт, что вне зависимости от почерка различия между буквами более значительны по сравнению с различными написаниями одного символа.

Наиболее распространенным способом распознавания рукописных текстов является применение технологии нейронных сетей. Нейронная сеть – это модель, представляющая собой сеть искусственных нейронов, выполняющих математические операции. Каждый нейрон получает на вход определенные данные, которые он обрабатывает и передает на следующие нейроны [3].

Общий алгоритм системы распознавания текста приведен на рисунке 1.



Рис. 1. Общий алгоритм распознавания текста

На этапе предобработки изображение приводится к форме, удобной для выделения текста. Для снижения уровня шума используется фильтрация Гаусса.

С помощью адаптивной бинаризации производится отделение текста от фона и шумов на нем. Затем производится выравнивание текста.

Сегментация строк и слов происходит с помощью дилатации, все пиксели в каждой строке объединяются в единую черную полосу. Предполагая, что в межстрочных интервалах количество черных пикселей значительно меньше, чем в самих строках, можно найти порог для строки. Рассчитаем порог по следующей формуле:

$$I = 0.1 * Str_{\max},$$

где Str_{\max} - максимальное число черных пикселей в строке.

Найденный порог позволяет разделить изображение на строки. Работа алгоритма представляет собой последовательный просмотр массива, который содержит число черных пикселей каждой строки и сравнение их с минимальным количеством. После нахождения множества пар индексов строк, соответствующих границам строк.

Аналогичным образом можно разделить строки на слова, только слова отделяются при нахождении определенного количества белых пикселей в строке.

Проблемы сегментации начинаются при определении отдельных символов внутри слов, поскольку слова представляют единую конструкцию слитно написанных букв. Здесь нам помогает двухпроходный метод связанных компонент [2]. При обходе по изображению создается двумерная матрица, в которой хранятся данные о метках пикселей.

Во время первого прохода происходит сканирование соседей слева и сверху от текущего пикселя слева направо и сверху вниз. Каждому найденному черному пикселю дается метка в зависимости от значения пикселей соседей. Если все соседи в матрице имеют значения фона, то пиксель становится промежуточным. Если в матрице только один сосед относится к объекту, то пиксель становится соседом. В случае нахождения среди соседей нескольких объектов, выбирается метка с наименьшим значением и пиксель получает значение выбранной метки-представителя.

Второй проход происходит в обратном направлении и рассматриваются все соседи. Происходит поиск связи между различными полученными метками. При нахождении связи все метки меняются в зависимости от значения меток соседей. Результат сегментации представлен на рисунке 2.

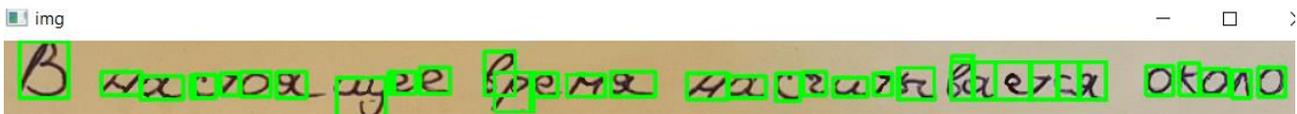


Рис. 2. Результат сегментации символов

Поскольку при обработке используются изображения больших размеров, для распознавания текстов используются сверточные и рекуррентные нейронные сети. Они позволяют быстро и эффективно работать с текстами [1].

Архитектура сверточных нейронных сетей является предпочтительной для распознавания рукописных текстов, так как она наиболее устойчива к разнице между различными почерками. Она позволяет правильно находить символы вне зависимости от изменений размеров, наклонов и других искажений, получаемых при сканировании рукописных текстов.

Сверточная нейронная сеть представляет собой два вида слоев: сверточные и подвыборочные. Сверточные слои отвечают за поиск и создание карт признаков символов. Карты признаков получаются слишком большими и излишне подробными. Подвыборочные слои отвечают за уменьшение объема данных в картах признаков. Для улучшения эффективности распознавания необходимо использовать несколько сверточных и подвыборочных слоев, которые должны чередоваться [4].

Значимым достоинством сверточных нейронных сетей является тот факт, что слои нейронов не являются полносвязными, что позволяет распараллеливать работу в них. Это позволяет значительно ускорить обучение модели.

На практике была создана сверточная нейронная сеть с тремя сверточными и подвыборочными слоями. Выходы нейронов третьего подвыборочного слоя соединены с многослойным персептроном, который разделяет полученные карты признаков в зависимости от распознаваемого символа. Схема созданной нейронной сети представлена на рисунке 3.

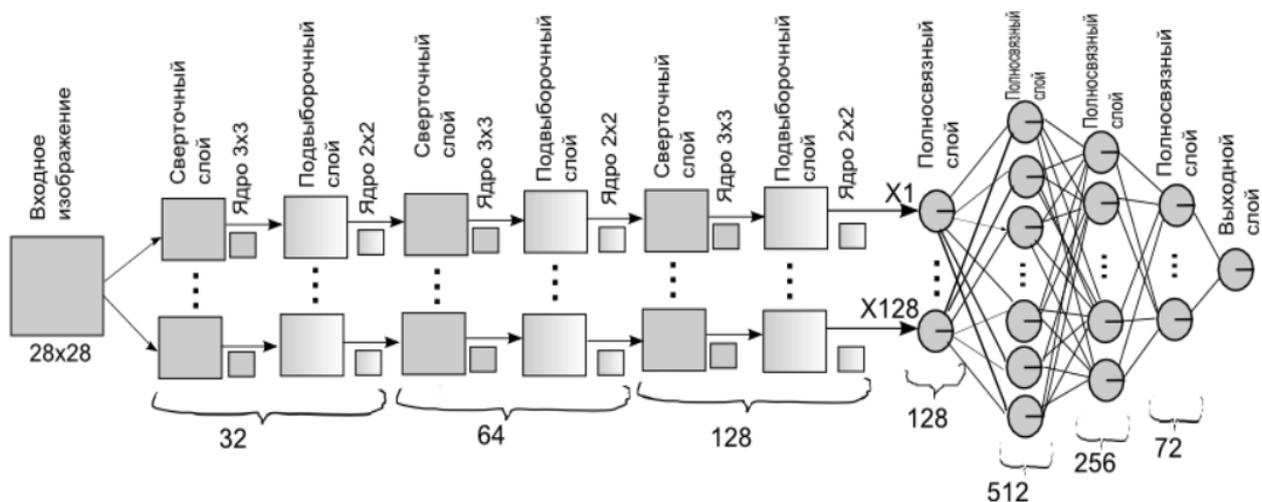


Рис. 3. Схема нейронной сети

На выходе мы получаем модель, обученную на 72 классах символов, в которые входят буквы алфавита русского языка в нижнем и верхнем регистрах, а также цифры. Нейронная сеть обучалась на датасете, полученном из набора MNIST (содержащий изображения цифр от 0 до 9) и базы данных рукописных кириллических символов. Данные датасета сбалансированы, и каждый из 72 классов объектов содержал 5250 элементов.

После определения принадлежности символов текста к той или иной категории, возможна дальнейшая проверка на правильность результатов. Для этого существует модуль постобработки, который работает непосредственно с текстовыми данными полученными после распознавания. Полученные слова проверяются на орфографию с помощью словарей. При отсутствии слов в словаре, в нем ищутся похожие, и предлагается вариант исправления.

Максимальная точность распознавания русского рукописного текста, программами, найденными в открытых источниках, составляет от 55 до 70%.

Нейронная сеть, созданная в рамках данной работы, позволяет достигать точности до 65%. Модуль постобработки позволяет улучшить данный результат на 10-15%.

Готовая программа имеет графический интерфейс и позволяет на обученной модели производить распознавание текстов. Графический интерфейс представлен на рисунке 4.

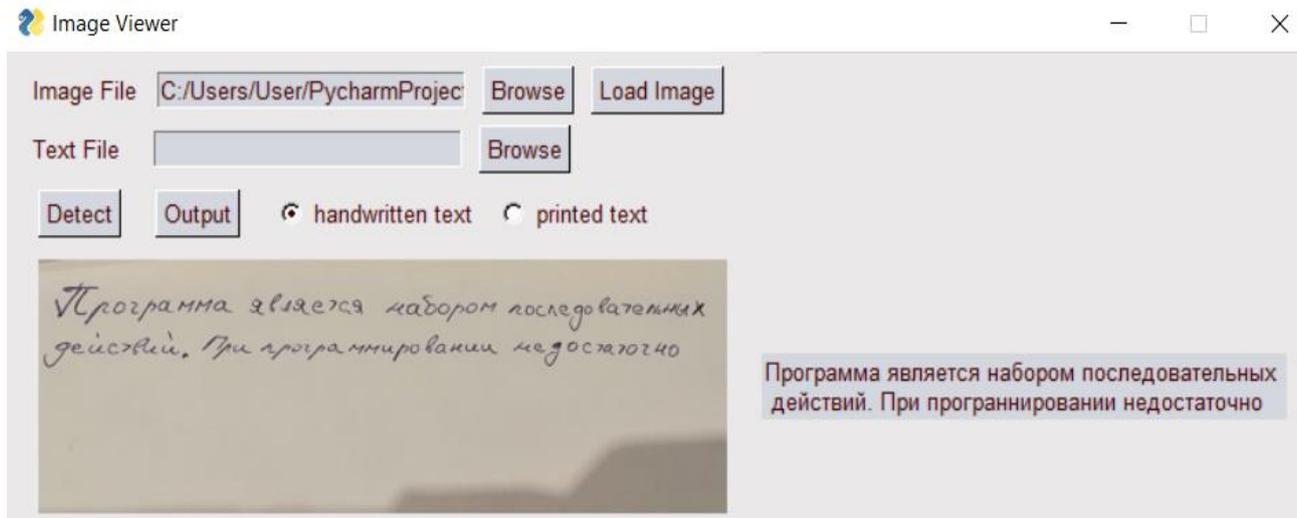


Рис. 4. Интерфейс программы

Для написания программы использовался язык Python, поскольку он имеет большое количество библиотек компьютерного зрения, которые позволяют достаточно легко создавать нейронные сети различных предназначений и архитектур. Были использованы библиотека машинного зрения Tensorflow и библиотека глубокого обучения Keras.

Важной частью программ для распознавания рукописного текста является модуль его сегментации. Попытка различить символы в словах требует не меньше внимания, чем сам процесс распознавания. В данной работе для распознавания символов использовался один из наиболее эффективных методов – двухпроходный метод связанных компонент, который является не менее эффективным в сравнении с другими методами.

На данный момент лучшим вариантом для распознавания рукописного текста является использование сверточных нейронных сетей, поскольку они позволяют работать с данными, имеющими различные искажения.

Если сравнивать с другими методами машинного обучения (такими как SVM, Extra Tree Classifier, MLP, KNN), то архитектура сверточных нейронных сетей является самой точной. Достигается это за счет увеличения сложности используемых нейронных сетей глубокого обучения.

На точность сверточных нейронных сетей влияет выбор алгоритма оптимизации. По результатам тестирования программы с различными оптимизаторами становится понятно, что для данной задачи наиболее точные результаты получаются при использовании оптимизатора

Adadelta, который на 1-1.5% побеждает своих главных оппонентов (оптимизаторы Adam и AdaGrad) и на 10% - SGD.

Сверточные нейронные сети не только позволяют с высокой точностью распознавать рукописные тексты, но и обучение модели происходит на порядок быстрее за счет распараллеливания этого процесса с использованием архитектуры CUDA [5].

Проведенное сравнение результатов работы разработанной программы с найденными аналогами позволяет сделать вывод об эффективности и корректности работы программы. Кроме того, важным моментом является то, что дальнейшее улучшение программы возможно при помощи некоторой оптимизации алгоритма сегментации символов, а также нахождении более приемлемого датасета. Также возможно введение рекуррентной нейронной сети на этап постобработки для улучшения ее точности.

Несмотря на высокую точность при распознавании большинства рукописных текстов, программа может не справиться с распознаванием, если текст написан сильно неразборчивым почерком. Данная проблема является актуальной для большинства методов распознавания.

Литература

1. Исафилов Х.С. Применение нейронных сетей в распознавании рукописного текста // Молодой ученый. 2016. №29. С. 24-27.
2. Попова Е.С., Спицын В.Г., Болотова Ю.А. Программа и алгоритм сегментации и распознавания рукопечатных символов с помощью сверточных нейронных сетей // GraphiCon 2018: труды 28-й Междунар. конф. по компьютерной графике и машинному зрению (г. Томск, 24-27 сентября 2018). Томск, 2018. С. 230-233.
3. Скрыпников А.В., Денисенко В.В., Хитров Е.Г., Евтеева К.С., Савченко И.И. Распознавание рукописного текста с использованием нейронных сетей // Современные наукоемкие технологии. 2021. №6-1. С. 91-95.
4. Хаустов П.А. Алгоритм сегментации рукописного текста на основе построения структурных моделей // Фундаментальные исследования. 2017. №4-1. С. 88-93
5. Чичкарев Е.А., Сергиенко А.В., Балалаев Е.Ю. Использование моделей машинного обучения и сетей глубокого обучения для распознавания рукописных чисел и букв русского и латинского алфавитов // InterConf. 2021. №87. С. 363-380.

© Суворин А.В., 2023

РАЗРАБОТКА КОРПОРАТИВНОГО САЙТА

В последние годы количество веб-ресурсов значительно увеличилось, как и их влияние на успешность и конкурентоспособность компании. Из-за обыденности наличия своего веб-ресурса, корпоративный сайт перестает быть уникальным авторским творением, вместо этого он становится типовым программным продуктом, создаваемым на базе шаблонов.

Проблемой корпоративных сайтов является отсутствие уникальности по наполнению контента. На сегодняшний день существует множество различных CMS и конструкторов для разработки сайтов, но самым продуктивным и удовлетворяющим требования конкретного заказчика является самостоятельная разработка сайта «с нуля».

Такой подход к написанию сайта или веб-приложения требует от разработчика серьезных знаний не только языков программирования, но и понимания архитектуры информационных систем, логики бизнес-процессов клиента и многого другого. В то же время создание сайта «с нуля» гарантирует его полную клиент-ориентированность и отсутствие лишних временных затрат на выполнение ненужных процессов [1].

Преимущества выбора самостоятельной разработки:

1) Свобода выбора

Возможность реализации задач любой сложности, полное соответствие функционала техническому заданию, отсутствие необходимости адаптации какого-либо шаблона.

2) Широкие возможности продвижения

В отличие от сайта, разработанного с помощью CMS и конструкторов, продвинуть в поиске самостоятельно разработанный сайт намного легче.

3) Индивидуальный дизайн

Только самостоятельная разработка позволяет создать тот продукт, который необходимо получить согласно индивидуальным требованиям заказчика [3].

Метод разработки.

Плоский дизайн

Плоский или Flat дизайн представляет собой «упрощенное» оформление сайта, которое отличается минимализмом, отсутствием лишних деталей, простотой. Такой дизайн называют плоским, потому что все объекты расположены в одной плоскости и не имеют объема [4].

Принципы плоского дизайна:

1) Двумерная графика.

Упрощение дизайна касается не только форм и кнопок, но и самой графики. Стиль плоского дизайна – двумерные иллюстрации, которые используются как полноценные изображения в галереях, на слайдерах и других областях сайта. Особенность таких изображений заключается также в отсутствии каких-то лишних выделяющихся элементов и соблюдении единого стиля в оформлении.

2) Читаемая типографика.

Стиль Flat убрал все лишнее из дизайна и вывел на первый план контент на сайте. Поэтому для пользователя стало особенно важно, какая типографика используется на ресурсе. Отсутствие теней и других декоративных деталей привело к тому, что тексты стали восприниматься по-иному. В основном в плоском дизайне используются шрифты без засечек для лучшего восприятия пользователем. Иногда типографика играет роль навигационного элемента. Например, меню на сайте оформляют в минималистичном стиле и оставляют только названия разделов.

3) Цветовые акценты.

Еще одна особенность Flat-дизайна – используемая цветовая схема. Преимущественно это белый фон, один главный цвет для элементов и различные цветовые акценты. Обычно палитра включает в себя 2-3 цвета.

4) Белое пространство.

Чтобы создать ощущение легкости, дизайнеры используют не только чистые цвета, но и правильно распределяют пространство на сайте. Элементы располагаются с учетом пробелов и воздуха между ними [5].

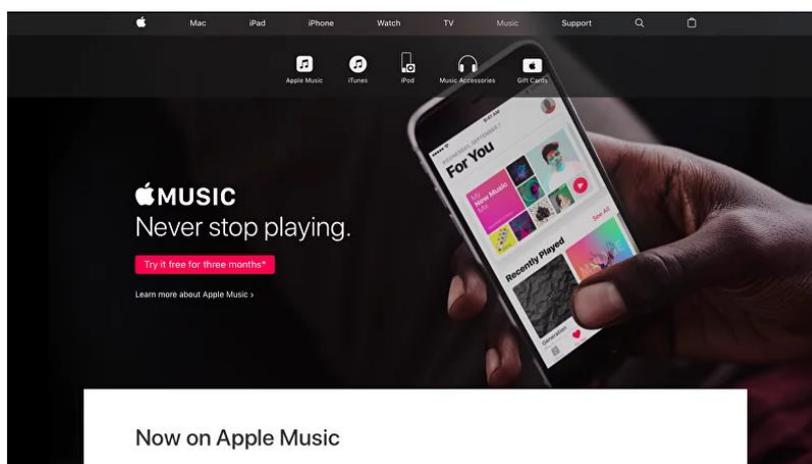


Рис. 1. Пример плоского дизайна

Преимущества выбора плоского дизайна:

- Простота информационной архитектуры. Данный дизайн хорошо подходит для сайтов с небольшим количеством текста и объемом 1-10 страниц.
- Гибкая структура. В плоском дизайне преимущественно используются геометрические формы. Такой тип расположения элементов на странице позволяет пользователям легко и быстро найти то, что им нужно.
- Оформление текста. В плоском дизайне лаконичность и простота также присутствует и в форматировании текста, так как отсутствие теней и прочих эффектов заметно упрощает чтение [2].

Рассмотрим этапы разработки сайта на примере сайта, предлагающего услуги профессионального фотографа.

1. Создание UseCase-диаграммы.

Диаграмма вариантов использования показывает, какой функционал и кем может быть использован на сайте. Диаграмма иллюстрируется на рисунке 2.

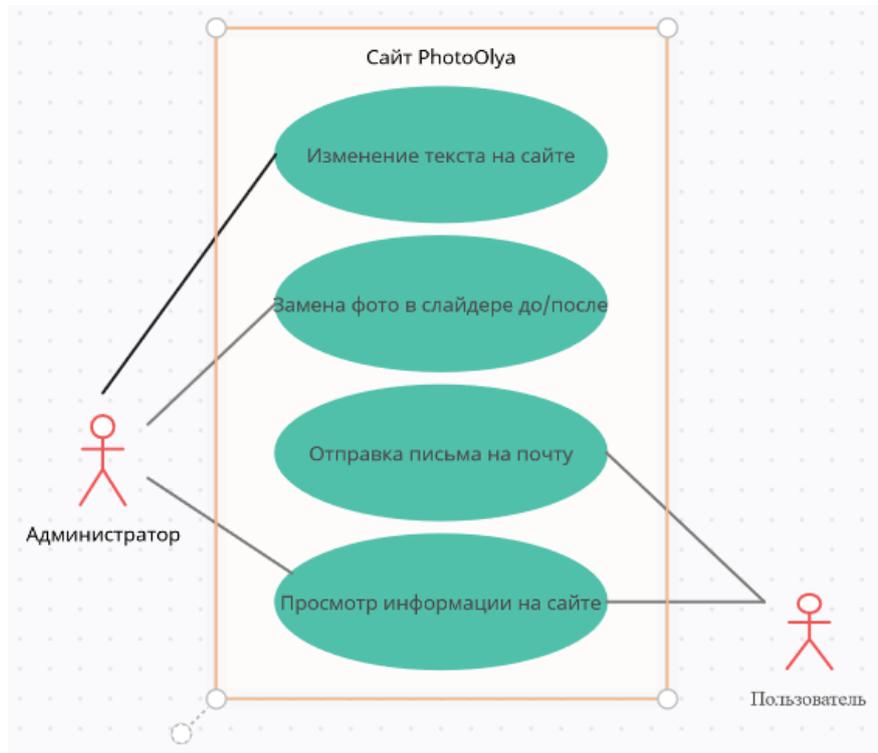


Рис. 2. UseCase-диаграмма

2. Разделы сайта.

Сайт выполнен в стиле Лендинг, содержит модальное окно для отправки заявки на фотосессию (рис. 3), краткую информацию о самом фотографe (рис. 4), раздел с работами (рис. 5, 6) и контактную информацию (рис. 8).

Также есть раздел «До и после обработки» (рис. 7), который демонстрирует клиенту навык фотографа в ретуши фотографий. Как показал анализ аналогичных сайтов, данный раздел является уникальным составляющим контента представленной разработки.

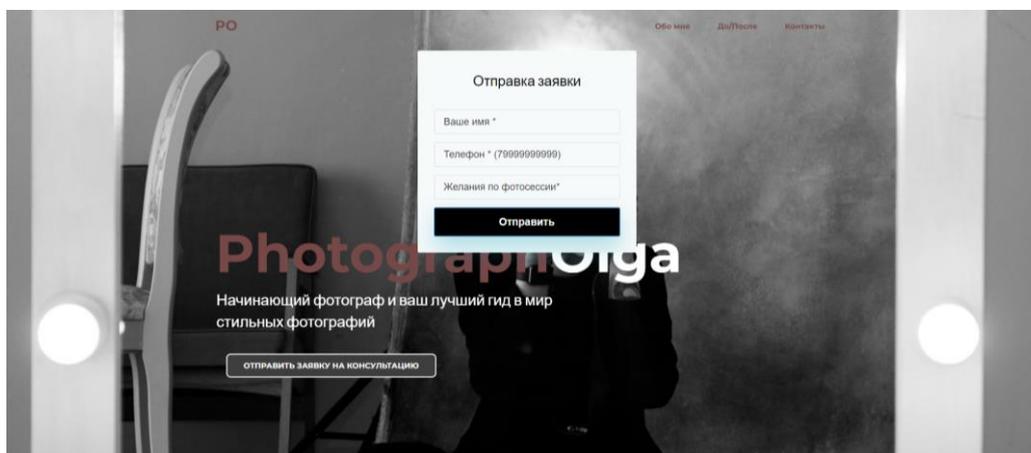


Рис. 3. Форма обратной связи

Кто я такая?

Меня зовут Оля и я начинающий фотограф. Занимаюсь фотографией уже 3 года и имею обширное портфолио из различных фотосетов в различных стилях и вот несколько преимуществ работы со мной:

♥ **Общительная и открыта для критики**

Я всегда рада пообщаться с людьми, и воспринимаю критику не как оскорбление, а повод стать лучше!:)

🕒 **Расстановка приоритетов**

Для меня очень важно чтобы после работы со мной вы получили кучу положительных эмоций от фотосессии и сохранили важные моменты своей жизни в восхитительных фотографиях!

⚙️ **Дисциплинированность и ответственность**

Я стараюсь делать свою работу максимально эффективно. Благодаря этому все фото будут обработаны и переданы вам в лучшем виде!

Рис. 4. Раздел «Обо мне»



Рис. 5. Раздел «Мои работы»



Рис. 6. Полноэкранный просмотр в разделе «Мои работы»



Рис. 7. Раздел «До и после обработки»

Контакты

Я надеюсь, что смогла заинтересовать вас.
Ниже представлены сервисы для контакта со мной и дальнейшего сотрудничества!

Обратная связь!



Рис. 8. Раздел «Контакты»

Таким образом, выбранный подход к разработке сайта позволил не только отобразить весь необходимый контент с учетом требований к эргономике сайта, но и продемонстрировать профессионализм фотографа с самой выгодной для него позиции.

Литература

1. Алаудинов Б.Р., Шахбазова М.С. Разработка веб-сайта, основные этапы разработки // Развитие современной науки и технологий в условиях трансформационных процессов: сборник статей международной научно-практической конференции (г. Москва, 07 декабря 2022). Грозный, 2022. С. 18-22.
2. Абилдаева Г.Б., Шакирова Ю.К., Савченко Н.К., Зайцева С.В. Плоский дизайн. Преимущества использования // Молодой ученый. 2015. №6. С. 43-45.
3. Махкамов Ф.И., Соседко В.В. Особенности и проблемы разработки корпоративного сайта // Информационные технологии в науке и производстве: сборник статей всероссийской молодежно-технической конференции (г. Омск, 08-09 февраля 2019). Омск, 2019. С. 169-174.
4. Молодецкая С.Ф. Современные стили дизайна веб-страниц // Информационные технологии: состояние и перспективы развития: сборник статей международной научно-практической конференции (г. Москва, 17 апреля 2019). Екатеринбург, 2019. С. 105-113.
5. Ничипорович М.О., Улюмджиева Б.Н. Почему плоский дизайн называют «плоским» // Роль и место информационных технологий в современной науке: сборник статей научно-практической конференции (г. Магнитогорск, 16 января 2018). Уфа, 2018. С. 24-30.

© Тарнавская О.А, 2023

УДК 519.687

Томшин Н.А.

Нижевартовский государственный университет
г. Нижневартовск, Россия

МАНИПУЛЯТОР ДЛЯ СБОРКИ КУБИКА РУБИКА

Кубик Рубика – популярная механическая головоломка, классический вариант которого собрать неподготовленному человеку без подсказок практически невозможно. Цель проекта – создать робота, который самостоятельно бы собирал любой стандартный кубик Рубика.

Для сборки кубика робот должен иметь возможность совершать над ним все возможные операции, для этого ему нужен специальный механизм-манипулятор.

Для начала разберём, что из себя представляет кубик Рубика с точки зрения механики. Это 26 подвижных частей снаружи, в исходном положении образующих форму куба, но проще представить это как куб, который поделили на 3 слоя по его 3 перпендикулярным рёбрам, таким образом, что любые слои могут начать вращаться относительно параллельных им слоёв. Одновременно вращаться могут только параллельные слои; пока они одновременно не выровняются, перпендикулярные слои разобраны и их взаимное вращение механически заблокировано.

Размер кубика Рубика – число слоёв, на которое поделён куб по каждой оси, обозначается как $N \times N \times N$, или короче $N \times N$, где N -число слоёв. Стандартный кубик – это 3×3 с длиной ребра 56мм. Все возможные операции над кубиком Рубика – это все возможные повороты любых отдельных слоёв относительно других, после которых слои оказываются выравнены (угол поворота кратен 90°). Для стандартного кубика определение всех операций можно сузить до всех возможных поворотов 6 внешних сторон. Тем не менее для этого всё равно необходимо фиксировать один слой и вращать смежный. Для конструктивного элемента, способного это сделать, вводится понятие «рамка». В простейшем исполнении – это 2 плоские параллельные рамы, расположенные на небольшом расстоянии друг от друга, в которые может войти куб и не прокручиваясь. Расположив границу слоёв в зазоре между рамами, вращая их, можно вращать слои, которых они касаются, причём под действием сил трения, слои, расположенные дальше тех, над которыми выполняется действие, будут вращаться вместе с ними.

В общем случае рамка – абстрактный механизм, поворачивающий друг относительно друга 2 смежные группы слоёв. Группы слоёв зависят от расположения куба в рамке. В частном случае, когда куб целиком располагается в подвижной рамке, он будет вращаться целиком, и это можно использовать при ограниченном числе рамок, чтобы, вращая куб, подставлять нужную сторону.

Другой абстрактный механизм – опрокидыватель. Он предназначен исключительно для поворота куба целиком по оси, перпендикулярной оси вращения рамки, давая возможность приводить куб в любую ориентацию, имея только одну рамку.

Были рассмотрены несколько несложных схем, обеспечивающих полноту операций (табл. 1).

Примеры схем

Название, описание	Картинка	Степени свободы (и число моторов)	Плюсы и минусы
Лифт-рамка Куб перемещается между рамкой и опрокидывателем с помощью лифта		3-2: 1. лифт 2. поворот рамки 3. опрокидыватель, может быть связан с лифтом	-медленная работа; +простота конструкции
Барабаны Куб перемещается между двумя перпендикулярными рамками.		2: 1. синхронный поворот рамок 2. перемещение кубика	-кубик испытывает много трения; -сложный механизм перемещения куба; +минимум моторов
Стол-рамка Поворотный стол-подвижная рама, а неподвижная рама совмещает включает функцию опрокидывателя. Также она может освободить куб для вертикального вращения.		2: 1. поворот стола 2. движение рамки	-сложный механизм опрокидывания; +минимум моторов
Клешни Клешни хватают куб таким образом, что могут работать как 2 перпендикулярные рамки. Если одна отпускает, куб может вращаться целиком.		4: 1. контроль захвата 1 2. поворот клешни 1 3. контроль захвата 2 4. поворот клешни 2	-сложная конструкция модулей; +модули взаимозаменяемы; +быстрота операций; +внешний вид

Была выбрана схема «Клешни», ради потенциальной скорости работы с кубиком и для придания роботу более привлекательного внешнего вида (клешни напоминают руки). Вообще клешни и подобные схемы - популярное решение в роботах, собирающих подобные головоломки.

Преимущество в скорости обусловлено двумя вещами:

- в теории поверхность куба не будет испытывать износа трением, что снимает скоростные ограничения, связанные с этим фактором;
- в процессе работы кубу не нужно между вращениями совершать линейные перемещения, на чём экономится время.

Схему с клешнями разумно исполнить, сделав сами клешни одинаковыми модулями - легко заменяемыми сборочными единицами.

Созданию этого модуля манипулятора и посвящена эта статья.

Технические требования к манипулятору:

- Клешня должна быть способна удержать куб на весу.
- Длина ребра куба 56мм.
- Куб должен захватываться нормально к его грани на определённом расстоянии.
- Клешня должна уметь вращаться по нормали к захваченной стороне без угловых ограничений с кубиком и без.
- Клешня должна быть способна останавливаться под прямыми углами, чтобы корректно совершать операции над кубом.
- Клешни должны быть способны совместно выполнять операции над кубиком.
- Все вышеизложенные требования должны выполняться при любой ориентации в пространстве.
- Предусмотрены элементы крепления для установки на каркас робота.

На рисунке 1 представлена окончательная схема, которая пошла в изготовление рабочих экземпляров.

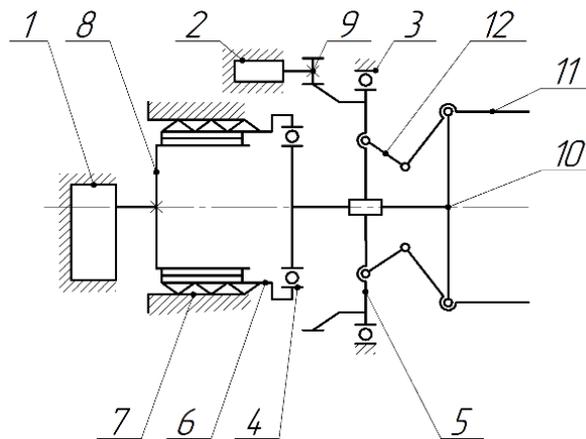


Рис. 1. Принципиальная схема:

- 1 – сервомотор; 2 – мотор постоянного тока; 2 – подшипник шариковый основания;**
3 – подшипник шариковый межвальный; 4 – основание клешни; 5 – винт; 6 – резьба опорная;
7 – диск шлицевой; 8 – ведущее зубчатое колесо; 9 – вилка; 10 – палец; 11 – звено механизма

Как и было обозначено ранее, у клешни есть 2 степени свободы: вращение и захват. Основание клешни крепится на шариковом подшипнике в корпусе. Вращение основанию сообщается через зубчатое зацепление ведущего колеса, закреплённого на оси мотора с зубчатым венцом на самом основании. Положение захвата контролируется сообщением поступательного движения вилке по оси вращения, таким образом обе степени свободы могут контролироваться независимо. Поступательное движение сообщается через поворот винта, находящегося в резьбовом соединении с корпусом. Межвальный шариковый подшипник связывает винт с вилкой, давая им свободно вращаться, не ограничивая поворот клешни определённым диапазоном. Сам винт вращается сервомотором через шлицевое соединение, позволяющее винту перемещаться вдоль оси вращения, относительно стационарного

сервомотора. При движении вилки вдоль оси вращения рычажный механизм заставит пальцы захвата сжиматься или разжиматься.

Для корректной работы манипулятора необходимо сделать надёжный механизм захвата, который в теории должен сжать пальцы до размера куба в определённом положении. Также необходимо учесть, что когда клешня не держит куб, она не должна мешать кубу вращаться по любой оси, а когда клешня держит куб, смежная клешня должна свободно уметь хватать и вращать боковые стороны.

Синтез механизма проводился в бесплатной программе Linkage 3.16.14

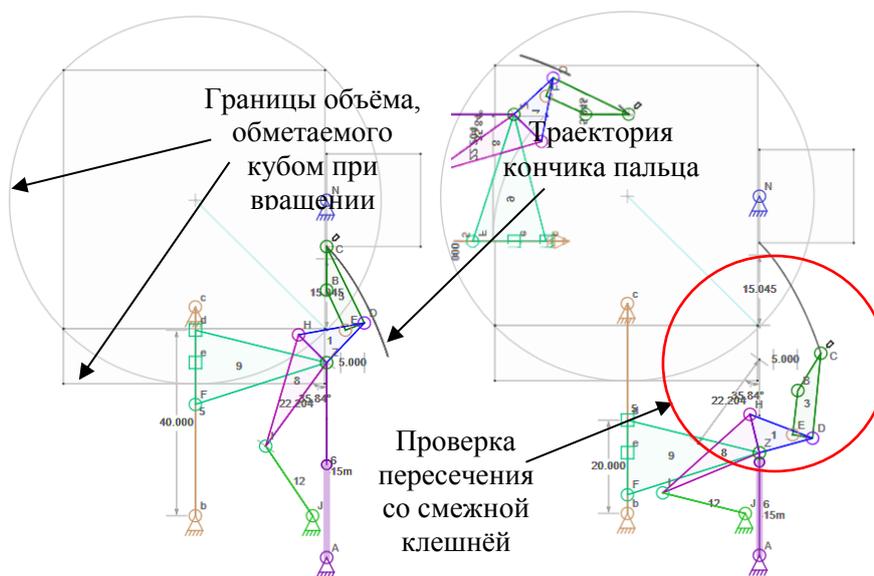


Рис. 2. Модель плоского механизма в двух крайних положениях

На рисунке 2 представлена модель, на основе которой были выбраны размеры для звеньев механизма захвата. Программа позволяет достаточно точно считать расстояния и углы в выбранных единицах измерения (мм, градусы), чтобы использовать в чертежах (моделях) деталей.

Важнейшей задачей было предотвращение столкновений клешни с кубом или другой клешней, когда этого не нужно. Задаётся, что рабочая плоскость – осевая плоскость, параллельная боковой стороне куба. В этой плоскости происходит максимальное сближение деталей при вращении по принадлежащим ей осям. Было обозначено сечение куба (квадратом), затем окружностью, сечение, которое куб обметает при вращении по нормальной оси, и прямоугольником, касающимся этой окружности, обозначен нижний край сечения, которое обметает куб при вращении по горизонтальной оси. При отпуске все детали клешни не должны пересекать общий контур этих сечений. Были подобраны форма и расстояние перемещения, при котором между звеньями не возникают критические углы, приводящие к заклиниванию, детали захвата покидают опасную зону с запасом расстояния, а также при захвате позволяют смежной (перпендикулярной) клешне свободно вращаться в отпущенном состоянии. Запас расстояния необходим для исключения трения, столкновений

из-за неточностей изготовления и из-за существенной толщины реальных деталей, которые при вращении обметут расширенное, относительно оси вращения, сечение.

Кроме технических требований, возникли и другие ограничения, связанные с готовыми изделиями, использованными в модуле – это электронные компоненты, подшипники и корпуса.

Таблица 2

Готовые изделия

Наименование, характеристики	Изображение	Комментарий
Подшипник 6816 шариковый Внешний диаметр 100мм Внутренний диаметр 80мм		Были специально приобретены, для использования в проекте до окончательной его проработки, что привело к трудностям, с размещением деталей в совокупности с корпусом.
Подшипник 6812 шариковый Внешний диаметр 78мм Внутренний диаметр 60мм		
Сервомотор S06NF STD* Две штуки Рабочее напряжение 5В Диапазон поворотов 180° Максимальный момент сил 13кг•см		Сервомотор подходит для контроля вращения в определённом диапазоне углов. Встроенный контролер сам управляет двигателем для достижения желаемого угла поворота.
Мотор постоянного тока название модели стёрто Две штуки Рабочее напряжение 12В		Простые щёточные моторы постоянного тока. Сила которых на ощупь была определена как достаточная для вращения клешни. Несмотря на одинаковый формфактор, у выбранной пары немного отличаются размеры и характеристики.
Энкодер** MiniQ Robot Chassis Encoder 2 канала от инфракрасных датчиков		Т.к. вращение клешни управляется простым электромотором, нужно использовать отдельный датчик для контроля поворота клешни. Энкодер будет считывать информацию с цилиндрической поверхности с чёрно-белой разметкой, по которой можно будет определить ориентацию клешни в данный момент.
Корпус (герметичный) Две штуки Габариты: 120×120×60		До достаточного освоения 3д печати было решено в качестве корпуса использовать модифицированную пару готовых герметичных корпусов из запасов лаборатории. Размещение вышеперечисленных деталей получалось лишь впритык. Несмотря на все неудобства, корпуса надёжно защищают внутренности от механических повреждений и имеют удобные места для крепления

Примечание: *Технические характеристики сервомотора S06NF STD (<https://dlmh9ip6v2uc.cloudfront.net/datasheets/Robotics/S06NFSTD.pdf>); **Энкодер для шасси miniQ (<https://clck.ru/34qDri>).

Кроме того в сборке использовались стандартные шурупы нескольких размеров.

Было решено разместить все электронные компоненты внутри составного корпуса в форме куба с ребром 120мм. Пришлось выпилить отверстия в стенках корпусов, чтобы весь механизм поместился внутрь, а клешня была расположена снаружи.

Все новые детали было решено изготавливать на лабораторном 3д принтере. PLA пластик, которым печатает принтер, вполне подходит для нагрузок, которые будут испытывать детали манипулятора – это было предположение, которое подтвердилось на практике. Хотя 3д печать позволяет создавать детали сложной формы, у неё есть недостатки: долгое время печати и дополнительные штрафы по времени и использованному материалу, если у детали отсутствует плоская поверхность или угол между стенкой детали и горизонтом слишком мал. Также чтобы изготавливать детали определённого качества, нужно подобрать подходящие для данного материала настройки. Самые беспроблемные материалы были PLA+, они хорошо печатались и не отходили от стола. Чуть хуже печатался PLA, но детали получались, тем не менее, рабочими. С помощью PETg и тем более ABS на момент написания статьи не удалось напечатать хороших деталей.

Проектирование деталей и сборки велось в учебной версии отечественной программы КОМПАС-3D 21.0.2.1174.

При проектировании создавались параметрические модели с внешними связями. Таким образом, при необходимости изменить некоторые размеры, это можно было сделать в исходной детали, а все зависящие перестроятся автоматически, если это изменение не нарушит саму связь.

Изначально создавались модели компоновочные модели деталей и сборки, в общем виде описывающие форму изделий, но зачастую не обладающие всеми подробностями форм или деления на отдельные детали. Затем, когда было проверено, что компоновочная модель может собираться, изготавливаться и выполнять необходимые перемещения без пересечения деталей, начиналась работа над финальными моделями деталей, по которым будут изготавливаться детали. Это позволило сэкономить немного времени, сначала создав общую композицию, а потом проработав мелкие детали.

По окончании моделирования, финальные версии детали были экспортированы в stl-файлы и использованы в программе Ultimaker Cura 5.2.1 для создания проектов печати деталей, по отдельности или нескольких. На основе проектов генерируется g-код – инструкции для 3д принтера. В течение пары недель удалось распечатать набор деталей для одного модуля. Такое длительное время связано с ограниченным временем работы лаборатории.

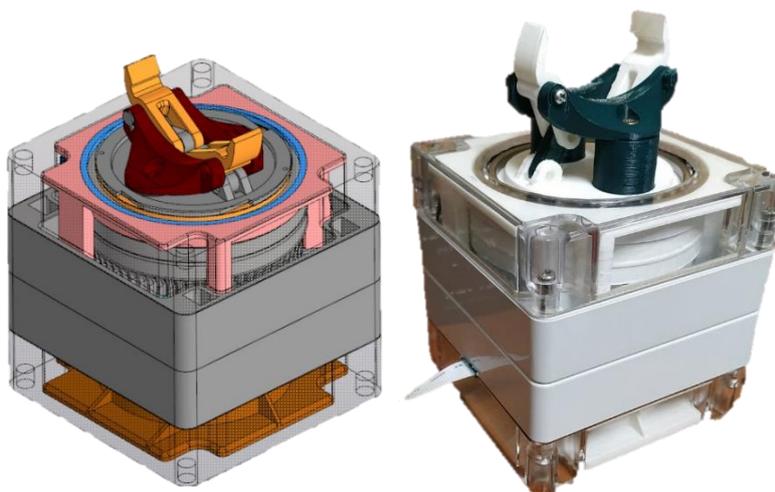


Рис. 3. Модель и реальный образец

После сборки были проведены испытания на удержание куба в воздухе, вращение клешни и возможность поворота удерживаемой стороны. В течении нескольких итераций испытаний были обнаружены и решены критические проблемы.

Таблица 3

Работа над ошибками

Проблема	Решение
Зубчатое колесо недостаточно плотно сидит на валу электродвигателя	Двухкомпонентный эпоксидный клей надёжно зафиксировал пластик на металле
Один из электродвигателей больше другого по диаметру и при установке деформирует детали в сборке за пределы нормы	Спроектировано исполнение деталей с изменённым размером для модуля с этим двигателем
В действительности диапазона работы сервомоторов не хватает для движения винта на требуемую длину	В модели увеличен ход резьбы, чтобы поворот на 180 градусов давал смещение значительно больше необходимого
Механизм захвата не захватывает куб в крайнем положении	На поверхности, зажимающие куб, нанесён дополнительный слой силиконового двустороннего скотча

В результате исправлений, технические требования были исполнены. Затем по обновлённой модели были распечатаны детали второго модуля, который после сборки сразу успешно прошёл испытания.

© Томшин Н.А., 2023

УДК 004.6

Топорков А.И.

Пермский национальный исследовательский политехнический университет
г. Пермь, Россия

**ПРИЛОЖЕНИЕ ДЛЯ ВИЗУАЛИЗАЦИИ И АНАЛИЗА РЕЗУЛЬТАТОВ
МОДЕЛИРОВАНИЯ БИОМЕХАНИЧЕСКОГО УЗЛА.
ПОДГОТОВКА И ОБРАБОТКА ДАННЫХ**

В процессе формирования нового метода лечения некариозных дефектов зубов врачам-стоматологам необходимо подтвердить применимость разработанного подхода, в том числе проводя исследования напряженно-деформированного состояния зуба с дефектами и новым видом реставрации (либо новым вариантом материала) [3].

Одной из основных задач при анализе показателей является специализированная обработка полученных данных к наиболее подходящему для визуализации виду с помощью алгоритмов сортировки, сглаживания и т. д. Алгоритм обработки данных должен быть универсальным, быстрым и экономичным по отношению к вычислительной мощности. В связи с тем, что естественнонаучные эксперименты характеризуются большими потоками информации, для визуализации и анализа полученных данных необходимо прибегать к возможностям ЭВМ. Наиболее распространенным подходом к решению статистических задач является метод наименьших квадратов (МНК) в его различных модификациях. МНК демонстрирует высокую эффективность для линейных моделей, но на практике встречаются ситуации, когда связь искомого параметра с измеряемой величиной нелинейная. В таком случае применяют нелинейный МНК или другие методы обработки [1].

Чаще всего помочь пользователю сориентироваться в данных помогают методы визуализации информации. Сегодня концепция отображения информации ассоциируется со средством усиления ментальных процессов человека. Визуализация является связующим звеном между зрительной системой и компьютером, помогая извлекать идеи из огромных массивов данных, что способствует научному исследованию и прогнозированию. Научная визуализация помогает ученым и инженерам более эффективно наблюдать физические явления, скрытые в больших объемах информации. В первом случае эта информация часто представлена в численном виде с заранее определенными переменными. В настоящее время применяют методы анализа и разведки данных для обработки крупных массивов с последующей визуализацией, чтобы существенно облегчить обнаружение общих шаблонов в данных.

В нашем случае визуализация данных будет представлена в виде графиков, которые позволят наглядно сравнить деформационные характеристики биомеханического узла при различной нагрузке, чтобы врач-практик мог уверенно сделать выбор в пользу оптимального материала. Визуализация данных должна полноценно отражать различные зависимости параметров, чтобы практикующий врач не испытывал трудности при их анализе. Средства реализации должны быть свободно распространяемыми на территории РФ. Методы реализации следует выбирать с учётом того, чтобы была возможность выполнять расчёты,

обработку и хранение данных на пользовательском ПК (без приобретения специальной вычислительной техники) за относительно небольшой период времени.

При написании алгоритма парсинга данных и последующей визуализации были выбраны следующие средства реализации:

1) язык программирования Python 3, данный язык легок в освоении и удобен для решения алгоритмических задач, поскольку имеет множество свободно распространяемых библиотек, в частности Matplotlib для визуализации данных;

2) средство визуализации данных в связи с выбранным языком программирования – Matplotlib. Библиотека имеет хороший функционал и проста для понимания начинающим.

В качестве исходных данных для обработки и визуализации используются результаты вычислительных экспериментов по прогнозированию НДС системы зуб-реставрация, полученные в САЕ-пакете ANSYS Mechanical APDL. Первым шагом необходимо выгрузить исходные данные, сохранённые в формате текстовых файлов со смешанными данными. Для этого выделены основные пункты по извлечению и обработке данных:

1) полученные исходные данные должны быть записаны во внутренние массивы данных (директорию текстовых файлов либо в базу данных).

2) запись данных из файла в массив возможно производить с помощью функции, которая сначала находит нужный столбец с данными (если таковых несколько), после чего построчно записывает данные для последующей обработки.

Для разработки функции обработки файлов с исходными данными составлена функциональная модель (рис. 1):

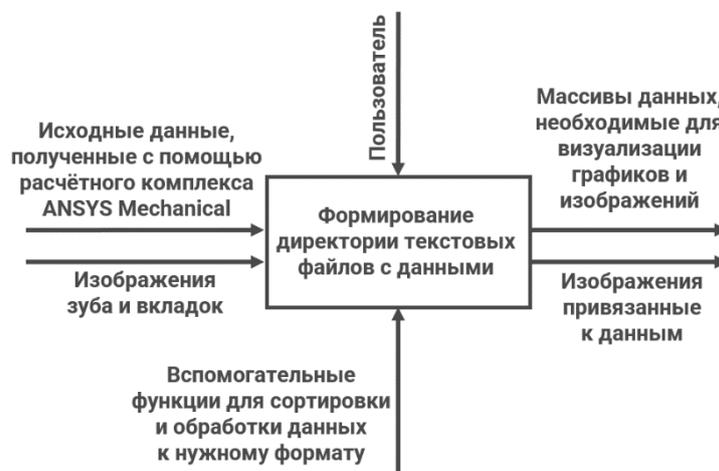


Рис. 1. Функциональная модель

Основная сложность обработки исходных файлов заключается в необходимости игнорирования побочной информации. На рисунке 2 показан вид исходных данных. Для записи в базу данных необходимых значений реализован алгоритм, который сначала отыскивает название нужного столбца, затем записывает значения из него.

В дальнейшем данные выбираются по индексу столбца, т. е. необходимо отыскать название столба и записать его индекс, а затем построчно занести соответствующие данные.

На рисунке 3 продемонстрированы сформированные директории (рис. 3а) и вид отсортированных данных (рис. 3б).

```

**** POST1 NODAL ELASTIC STRAIN LISTING ****
PowerGraphics Is Currently Enabled

LOAD STEP=      1 SUBSTEP=      7
TIME=      1.0000 LOAD CASE=      0
NODAL RESULTS ARE FOR MATERIAL      2

  NODE      EPEL1      EPEL2      EPEL3      EPELINT      EPELEQV
 3915  0.71489E-004 -0.11283E-004 -0.35619E-003  0.42768E-003  0.40191E-003
 3916  0.55205E-004  0.34680E-005 -0.22507E-004  0.77711E-004  0.57806E-004
 3917  0.12301E-003 -0.10593E-004 -0.28834E-003  0.41136E-003  0.36352E-003
 3918  0.13025E-003 -0.52700E-005 -0.21088E-003  0.34114E-003  0.26349E-003
 3919  0.64604E-004 -0.41621E-005 -0.15570E-003  0.22031E-003  0.16752E-003
 3920  0.37779E-004 -0.42363E-005 -0.96722E-004  0.13450E-003  0.10450E-003
 3921  0.41343E-004 -0.37532E-005 -0.79615E-004  0.12096E-003  0.91402E-004
 3922  0.22061E-004 -0.19025E-005 -0.49707E-004  0.71767E-004  0.65661E-004
 3923  0.20048E-005 -0.61615E-006 -0.19220E-004  0.21225E-004  0.32346E-004
 3924  0.98726E-005  0.35281E-005 -0.26540E-004  0.36412E-004  0.34535E-004
 3925  0.77891E-005  0.30654E-005 -0.31169E-004  0.38958E-004  0.30066E-004
  
```

Рис. 2. Исходные данные

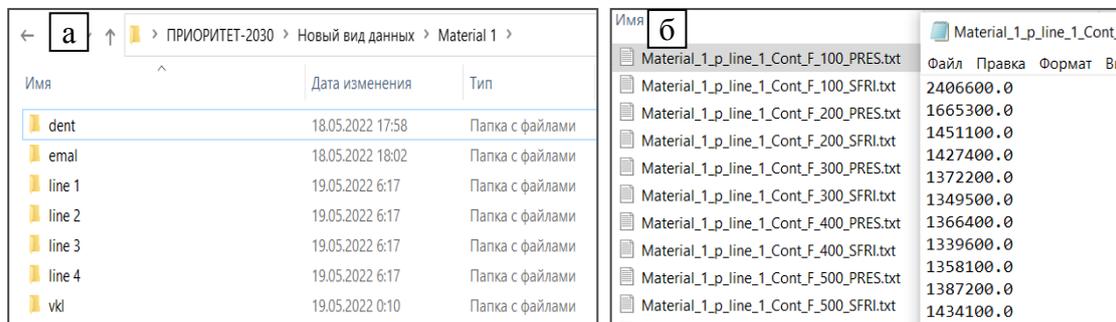


Рис. 3. Обработанные данные:

а) директории с данными; б) структурированные и сортированные данные

Вторым шагом реализован алгоритм, позволяющий осуществлять сглаживание результатов численного эксперимента (в частности, параметры контактного взаимодействия на границе зуб-реставрация) с помощью интерполяционных методов. В рамках работы были рассмотрены два основных метода интерполяции: метод наименьших квадратов и метод сплайнов. При реализации сглаживания с помощью метода наименьших квадратов, строится классическая система линейных уравнений:

$$\begin{cases} a_0 n + a_1 \sum_{i=0}^n x_i + \dots + a_k \sum_{i=0}^n x_i^k = \sum_{i=0}^n y_i \\ a_0 \sum_{i=0}^n x_i + a_1 \sum_{i=0}^n x_i^2 + \dots + a_k \sum_{i=0}^n x_i^{k+1} = \sum_{i=0}^n y_i x_i \\ \dots \\ a_0 \sum_{i=0}^n x_i^k + a_1 \sum_{i=0}^n x_i^{k+1} + \dots + a_k \sum_{i=0}^n x_i^{2k} = \sum_{i=0}^n y_i x_i^k \end{cases}$$

где x_i, y_i – координаты точек, полученные в результате обработки исходный файлов. Методами нелинейного программирования подбирается степень полинома. Затем происходит решение системы для отыскания коэффициентов a_k и последующее построение полинома.

Изображение простых, но в то же время красивых графиков помогает предоставить необходимую информацию наиболее доступно для пользователя. Библиотека matplotlib, для проектов на языке программирование Python 3, обладает широкими возможностями настройки

отображения данных, но предварительно необходимо разобраться с методами библиотеки и функционалом. Применение различных параметров позволяет гибко подстраивать визуализацию данных, приводя график к наиболее удобочитаемому варианту. Знание параметров и основных методов библиотеки существенно облегчает задачу аналитики данных, позволяя за короткие промежутки времени представлять и анализировать большие объёмы данных. Такие приёмы экономят время и ресурсы специалистов, позволяя сконцентрироваться на других аспектах своей деятельности [2].

На рисунке 4 представлены результаты, полученные при сглаживании данных с помощью интерполяции сплайнами (рис. 4а) и с помощью метода наименьших квадратов (рис. 4б).

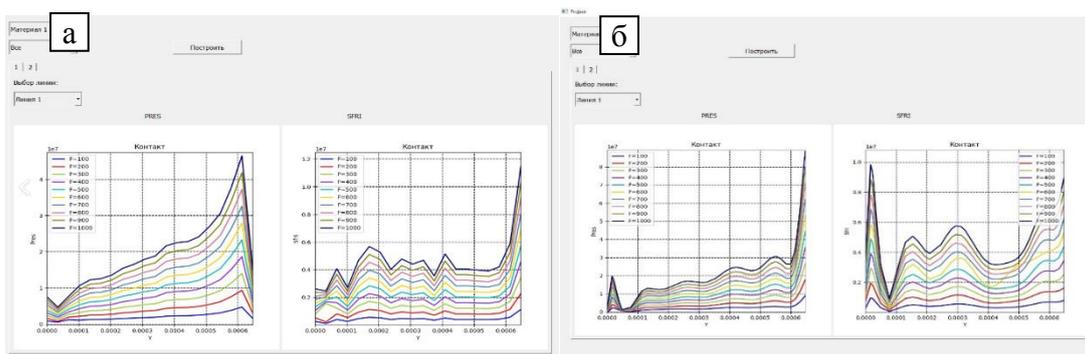


Рис. 4. Внешний вид графиков:

а) интерполирование результатов сплайнами; б) интерполирование с помощью МНК

Отмечено, что при интерполяции с помощью метода наименьших квадратов абсолютная погрешность получается минимальной, поскольку специальная библиотека, реализующая интерполяцию с помощью сплайнов, существенно обрезает начальные значения.

В результате разработаны алгоритмы поиска, обработки и подготовки данных для использования при построении виджетов с графиками в интерфейсной части разрабатываемого приложения.

Литература

1. Мальцев Д.П. Выбор метода для функции интерполяции // Международный студенческий научный вестник. 2020. №3. С. 3.
2. Пылов П.А. Аналитика возможностей визуализации данных в разнообразных темах оформления на основе библиотек matplotlib и seaborn // Россия молодая. 2020. С. 21115.1-21115.5.
3. Kamenskikh A.A., Sakhabutdinova L., Astashina N., Petrachev A., Nosov Y. Numerical Modeling of a New Type of Prosthetic Restoration for Non-Carious Cervical Lesions. Materials 2022, 15, 5102. <https://doi.org/10.3390/ma15155102>

УДК 371.31

Федулов С.И., Виштак О.В.

Балаковский инженерно-технологический институт – филиал НИЯУ МИФИ
г. Балаково, Россия

ОБЗОР ТРЕНАЖЁРОВ ДЛЯ ФОРМИРОВАНИЯ «КЛАВИАТУРНОЙ ГРАМОТНОСТИ»

Развитие компьютерных технологий увеличивает степень вовлечённости населения в работу с информационными системами. Работа по подготовке людей к подобному взаимодействию должна осуществляться в школе, в неё должно входить, в частности, обучение слепому вводу текста. В статье определяются критерии и производится оценка существующих решений для обучению машинописному вводу текста с точки зрения использования в центре дополнительного образования.

Компьютеры давно стали частью обыкновенной реальности для современного человек, но их степень влияния на жизнь человека не постоянна. С каждым годом появляются новые технологии, коренным образом меняющие возможности человека и сразу же вливающиеся в ряды привычных и используемых технологий. К таким технологиям можно отнести блокчейн, нейронные сети, облачные технологии, технологии обработки больших данных, современные аддитивные технологии, робототехнику [2; 3; 5-8].

Приоритетным направлением для Российской Федерации является развитие цифровой экономики, которая основывается на достижениях в области информационных технологий [4]. К факторам, которые положительно влияют на развитие цифровой экономики, в первую очередь относят свободный доступ в интернет, позволяющий расширять рынки сбыта и привлекать новых покупателей отечественным производителям [1].

Особенно важными для людей являются навыки, позволяющие использовать информационные технологии с учётом развития как цифровой экономики в частности, так и информационных технологий в целом. Современное образование может поспособствовать получению данных навыков для поколений, которым придётся существовать в мире с ещё большим уровнем внедрения информационных технологий.

Одним из основных способов повышения скорости работы с цифровой информацией, доступный даже для школьников младших классов, является слепой метод набора текста, также называемый машинописью [9]. Машинопись позволяет не только повысить скорость работы с текстом, но также уменьшает нагрузку на глаза и шею человека, убирая необходимость поднимать и опускать глаза с монитора на клавиатуру и обратно.

До развития информационных технологий обучение слепому методу набора осуществлялось по бумажным самоучителям, в настоящее же время задача значительно упростилась. На выбор пользователя представлено множество решений, со своими особенностями и уникальными возможностями. В данной статье будет приведён обзор существующих рыночных решений, при этом приоритетом будет наличие возможностей, упрощающих использование конкретного решения в качестве инструмента для обучения в рамках школьной образовательной программы.

Для сравнения существующих продуктов будут использоваться следующие критерии:

1. Бесплатная лицензия. Поиск и выделение средств на приобретение какого-либо программного обеспечения, применимость которого ограничивается малой частью одной дисциплины является весьма маловероятным сценарием для государственных образовательных учреждений. Бесплатная лицензия является преимуществом.

2. Доступность русского языка. Обусловлено необходимостью изучения этого языка в рамках школьной программы.

3. Доступность английского языка. Обусловлено необходимостью изучения этого языка в рамках школьной программы.

4. Локализация интерфейса. Локализация необходима для работы с тренажёром учеников начальных классов и учеников, изучающих в рамках программы любой другой иностранный язык вместо английского.

5. Отсутствие необходимости установки. Необходимость установки относится к недостаткам, так как оптимальная модель не требующая от организации никаких вмешательств, это, например, предоставление интерфейса к системе как для учеников, так и для преподавателей в удалённом режиме.

6. Отечественный разработчик. Местоположение разработчика является одним из критериев, обеспечивающих преимущество, так как это гарантирует попадание как продукта, так и разработчика под юрисдикцию Российской Федерации. Можно увидеть результат отсутствия возможности правового воздействия на примере тренажёра RataType, который с недавнего времени располагает на своём сайте политические лозунги, что определённо является фактором, препятствующим использованию данного веб-приложения государственными образовательными учреждениями.

7. Наличие базового курса. Наличие базового курса означает разбитие процесса обучения на уроки, доступ к которым возможен из меню. Упрощает отслеживание прогресса, гарантирует полное покрытие всех клавиш клавиатуры.

8. Возможность для работы с группой. Возможность работы с группой включает в себя любой дополнительный функционал для организации процесса обучения (кроме создания профилей или учётных записей).

9. Отсутствие рекламы. Отсутствие рекламы как в платной, так и в бесплатной версии.

Ниже приведён обзор некоторых решений, потенциально пригодных для использования в качестве инструмента при реализации образовательных программ.

1. **Stamina.** URL: https://stamina.ru/keyboard_trainer/download

В российском сегменте данная программа появилась одной из первых и получила широкую популярность. На данный момент основная версия программы датирована 2007 годом, последними дополнениями стали уроки, доступные с 2015 года. Интерфейс программы разбит на две зоны и может считаться эталонным для подобного вида программ: сверху находится бегущая строка с текстом для набора, снизу расположена виртуальная клавиатура, нужная для упрощения работы начинающим.

При помощи настроек возможна глубокая кастомизация программы, включая изменение размера текста, ширины строки, относительного размера виртуальной клавиатуры, цвета отдельных клавиш, схемы расположения пальцев (схемы зон покрытия) т. д. Для пользователей доступно несколько сценариев работы: изучение комбинаций букв, фраз, прохождение уроков, работа над символами и печать текстов, скопированных из любого источника. После прохождения любого текста доступно окно со статистикой, на которой можно увидеть зависимость скорости печати от времени, а также процент ошибок и среднюю скорость.

Благодаря наличию профилей несколько человек может использовать одну программу отдельно, что открывает некоторые возможности для её использования в образовательных учреждениях. Для упрощения изучения интерфейса программы доступна справка, написанная на русском языке и покрывающая все разделы программы. Не смотря на преимущественно положительное отношение пользователей к шуткам, интегрированным в программу и сопровождающих пользователя на всем пути обучения, их можно было бы отнести к нежелательному функционалу, снижающему концентрацию. Отметим, что существует возможность их отключения во время установки программы.

2. **Touch Typing Study.** URL: <https://www.typingstudy.com/ru>

Данный сайт предоставляет доступ к курсам для 108 языков, курс для русского языка разбит на 15 занятий. С самого начала доступ открыт ко всем частям курса, перед началом прохождения курса можно проверить свои навыки в специальном тесте скорости набора текста. Кроме языков есть возможность для работы с цифровой клавиатурой и развивающими играми. Для использования сервиса регистрация не требуется, но она необходима только для отслеживания прогресса. Поддержка проекта осуществляется посредством донатов, для спонсоров сервис скрывает рекламу.

Во время прохождения задания отображается статистика, показывающая скорость набора (слов в минуту), количество введённых знаков, прогресс выполнения теста, количество ошибок, показатель точности и время прохождения теста. Статистика после прохождения теста показывает скорость набора для каждого символа отдельно, такая функция может оказаться полезной для самостоятельной работы над ошибками.

При прохождении любого задания в нижней части экрана отображается виртуальная клавиатура, на которой зелёным цветом подсвечивается клавиша, которую требуется нажать в данный момент.

3. **Typing Club.** URL: <https://www.typingclub.com/russkiy>

Typing Club – сайт, который ориентирован на использование в образовательных учреждениях. Для организаций доступны две лицензии: бесплатная и платная. Обе версии позволяют проходить курс обучения полностью, но платная лицензия подразумевает наличие большего количества инструментов для аналитики. Сайт отличается современным дизайном, интуитивно понятным интерфейсом и богатым выбором курсов на английском языке. Для работы без сохранения прогресса регистрация не требуется.

Обучающимся доступны не только базовые курсы, но и множество курсов для альтернативных раскладок и курсов для проработки сложных слов и терминов из различных областей. Есть специальные курсы, в которых обучающиеся должны набирать текст какой-либо истории на фоне иллюстративной анимации, такое решение выглядит особенно привлекательным для увеличения заинтересованности школьников. Однако большинство таких дополнительных курсов созданы для английского языка.

Для преподавателей существует возможность создания курсов из отдельных уроков. Запись учащихся в учебную группу осуществляется при помощи кода учебной группы, после вхождения в которую преподаватель может видеть статистику как по классу, так и по каждому обучающемуся в отдельности. Доступна бесплатная лицензия и для преподавателей, и для обучающихся.

4. **Fast Keyboard Typing.** URL: <http://fastkeyboardtyping.com>

Fast Keyboard Typing предоставляет доступ к 189 заданиям, разбитым по 28 категориям. Для прохождения заданий и теста на скорость установлен лимит в 5 ошибок, в упражнениях лимит ошибок не установлен. По окончании теста выводится статистика - скорость набора, затраченное время и данные о введённых символах. Для тренировки доступно три языка.

По собственным данным ресурса в системе на данный момент зарегистрировано 38 544 пользователей, при этом проект позиционируется как некоммерческий, а сам тренажёр – как полностью бесплатный.

5. **Klavarog.** URL: <https://klava.org>

Отличительная особенность данного сайта заключается в использовании индивидуальных словарей, создаваемых пользователями из наиболее употребляемых ими слов. Кроме этого, возможна работа с тренажёрами по предустановленным словарям с различными скоростными режимами.

По умолчанию в каждой сессии показывается средняя скорость набора и процент ошибок. После регистрации пользователям становится доступен раздел со статистикой, содержащий данные из каждой сессии.

На сайте приведена теоретическая справка для работы со всеми разделами. Тренажёры размещены отдельно, для их использования не требуется регистрация. Для проработки доступны различные словари, в том числе словари для нескольких языков программирования, наборы для цифрового блока и частотные словари.

Дополнительно сервис включает функционал для изучения иностранных языков и развития навыка скорочтения.

6. **Соло на клавиатуре.** URL: <https://solo.nabiraem.ru>

Данный тренажёр изначально выпускался в виде скачиваемой программы, но затем был переработан и теперь позиционируется как подписочный сервис. Но возможность разового приобретения осталась, стоимость при этом составит 5000 рублей, минимальная цена за 1 неделю – 300 рублей. Для оценки своих возможностей предусмотрен тест скорости печати, по результатам которого можно определить степень необходимости приобретения подписки. На сайте представлены курсы для 9 языков и для цифрового блока.

Доступна как глобальная статистика, так и статистика во время выполнения – количество ошибок, скорость набора, средний балл. Для выполнения заданий установлено ограничение на количество ошибок, что по мнению некоторых пользователей портит впечатление от работы с тренажёром.

К недостаткам можно отнести невозможность перейти на любое занятие, для пользователей с любым уровнем печати доступен один и тот же курс с необходимой вводной частью.

7. **Keybr.** URL: <https://www.keybr.com>

Сайт отличается строгим современным дизайном, полезными и простыми настройками. На данный момент имеется поддержка семи языков а также возможность добавления текста, сравнимая с Klavaorg и Stamina.

Программа использует интеллектуальный алгоритм для подстраивания дальнейших уроков под особенности набора пользователя. Из результатов прохождения тестов формируется профиль, включающий следующие данные: общее затраченное время, общее количество занятий, максимальная скорость, средняя скорость, скорость набора каждой буквы, диаграмма сравнения предыдущих результатов и т. д.

В качестве недостатка можно указать отсутствие локализованного интерфейса.

8. **Typing Master.** URL: <https://www.typingmaster.com>

Typing Master является скачиваемой программой, доступны версии как для индивидуального использования, так и для образовательных организаций. Версия для школ позволяет учителю управлять профилями учеников, создавать и проверять тесты, настраивать общие параметры и давать обучающимся обратную связь.

Программа для индивидуального обучения включает работу с предложениями, фразами, словами, включает модуль для работы с цифровым блоком и тренировки по специальным символам. Typing Master оснащена алгоритмом для учёта ошибок, способным создавать индивидуальные задания для наиболее сложных комбинаций.

К серьёзным недостаткам можно отнести отсутствие русского языка и высокую цену.

9. **Rapid Typing.** URL: <https://www.rapidtyping.com/ru/online-typing-test.html>

Программа предназначена для использования в образовательных организациях. Скачанный файл должен устанавливаться на компьютер преподавателя, ученики должны работать с программой через общую папку, учётные записи учеников создаются учителем. Для оценки работы учащихся доступна панель со статистикой каждого ученика.

Rapid Typing поддерживает возможность работы с текстами в формате rtx и txt, создавая из них уроки. Программа поддерживает 21 язык и при этом полностью бесплатна.

Результат оценки всех перечисленных программ приведён в виде таблицы, а также в виде многоуровневой диаграммы на рисунке. Во второй колонке таблицы указаны веса, которые означают ценность каждого из критериев. Данная корректировка была введена для того, чтобы бесплатный продукт с рекламой оказывался лучше платного без рекламы.

Таблица

Сводная таблица характеристик тренажёров

	Бесплатная лицензия	Доступность русского языка	Доступность английского языка	Локализация интерфейса	Необходимость установки	Отечественный разработчик	Наличие базового курса	Возможность работы с группой	Отсутствие рекламы	Итоговая оценка
Бес	2	1	1	1	1	1	1	1	1	
Rapid Typing	+	+	+	+	-	-	+	+	+	8
Stamina	+	+	+	+	-	+	+	-	+	8
Klavarog	+	+	+	+	+	+	-	-	+	8
Fast Keyboard Typing	+	+	+	+	+	-	+	-	+	8
Typing Club	+	+	+	+	+	-	+	+	-	8
Соло на клавиатуре	-	+	+	+	+	+	+	-	+	7
Touch Typing Study	+	+	+	+	+	-	+	-	-	7
Keybr	+	+	+	-	+	-	-	-	+	6
Type Master	-	-	+	-	-	-	+	+	+	4

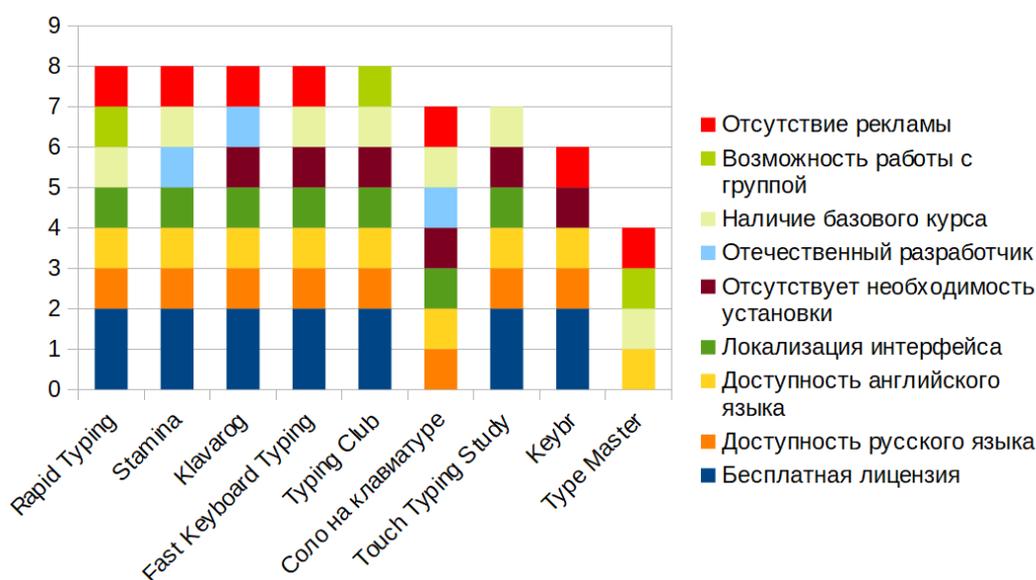


Рис. Распределение продуктов в соответствии с критериями

Исходя из собранной информации, можно сказать, что оптимальная рыночная стратегия для создателей тренажёров – ориентация на образовательные учреждения. Данная стратегия нацелена на большую платёжеспособность образовательных учреждений и большую специфичность программного обеспечения. Под специфичностью подразумевается функционал, реализующий наличие в системе преподавателя. Такой функционал не интересен

индивидуальным пользователям, в следствие чего они не готовы за него платить. Отсутствие интереса индивидуальных пользователей и усложнение процесса разработки приводит к меньшему количеству рыночных предложений и меньшему количеству игроков, поэтому некоторые производители предъявляют действительно высокие цены на свои продукты.

Наибольшее количество баллов в таких условиях получили не те решения, которые ориентированы на предоставление услуг образовательным организациям, а те продукты, которые предлагают наименьшую цену. Как можно видеть на рисунке 1, первое и второе место занимают некоммерческие приложения, выпущенные достаточно давно.

Литература

1. Вертакова Ю.В., Плахотникова М.А., Бабкин А.В. Тенденции развития цифровой экономики в России // Инновационные кластеры цифровой экономики: теория и практика. Санкт-Петербург, 2018. С. 290-315. <https://doi.org/10.18720/IEP/2018.4/13>.

2. Виштак Н.М., Ефремов Р.В., Беляева В.В. Облачные технологии в образовании // Актуальные проблемы и пути развития энергетики, техники и технологий: VII Международная научно-практическая конференция (г. Балаково, 23 апреля 2021 г.). Балаково, 2021. С. 155-161.

3. Виштак Н.М., Забродин М.Д. Облачные технологии в образовании // Современные технологии и автоматизация в технике, управлении и образовании: IV Международная научно-практическая конференция (г. Балаково, 16 декабря 2021 г.). Балаково, 2022. С. 230-235.

4. Захаров Д.В. Цифровизация экономики: проблемы и перспективы // Развитие науки, национальной инновационной системы и технологий: Международная научно-практическая конференция (г. Белгород, 13 мая 2020 г.). Белгород, 2020. С. 102-107.

5. Очкур Г.В., Денисов И.В. Обзор машинных методов обучения // Актуальные проблемы и пути развития энергетики, техники и технологий: VIII Международная научно-практическая конференция (г. Балаково, 20 апреля 2022 г.). Балаково, 2022. С. 304-308.

6. Рябенков Р.А., Михеев И.В. Блокчейн технология // Актуальные проблемы и пути развития энергетики, техники и технологий: VI Международная научно-практическая конференция (г. Балаково, 23 апреля 2020 г.). Балаково: 2020. С. 225-229.

7. Рябенков Р.А., Михеев И.В. Искусственный интеллект и его потенциал // Актуальные проблемы и пути развития энергетики, техники и технологий: VI Международная научно-практическая конференция (г. Балаково, 23 апреля 2020 г.). Балаково: 2020. С. 229-233.

8. Рябенков Р.А., Михеев И.В. Машинное обучение // Современные технологии и автоматизация в технике, управлении и образовании: IV Международная научно-практическая конференция (г. Балаково, 18 декабря 2019 г.). Балаково: 2020. С. 224-228.

9. Gillespie C., Leader L. We can... can they? Touch Typing for First Graders. Diss. Ph. D. thesis, Citeseer, 2005.

УДК 004.4

Харитонов Л.С., Устюжанин И.П., Бочкарева М.Д., Эварт Т.Е.
Арзамасский политехнический институт (филиал)
Нижегородского государственного технического университета
г. Арзамас, Россия

РАЗРАБОТКА КРОССПЛАТФОРМЕННОГО ЭЛЕКТРОННОГО УЧЕБНОГО ПОСОБИЯ ПО ПРОГРАММИРОВАНИЮ НА ЯЗЫКЕ C++ С ПОМОЩЬЮ XAMARIN.FORMS/C#

Информация является важнейшим ресурсом в современном обществе, в связи с чем информационные технологии проникли во все сферы человеческой деятельности, и, несомненно, заняли там очень прочные позиции. Исключением не является и сфера образования, где электронные средства всё чаще рассматриваются как неотъемлемая часть целостного образовательного процесса, призванная качественно повысить его эффективность [4]. И они вполне оправдывают своё призвание, ведь информатизация учебного процесса даёт отличные результаты, а цифровые пособия имеют массу преимуществ перед «классическими» бумажными. Информация в них представляется в интерактивном, доступном, визуально приятном и понятном виде, её проще редактировать, обновлять и т. д. Эти преимущества особенно актуальны при изучении программирования. Электронное учебное пособие (ЭУП) можно использовать как на занятиях, так и для самостоятельной работы, без которой невозможно представить полноценное изучение программирования. На занятиях же можно посвятить время изучению необъятных и постоянно пополняющихся сложных концепций и парадигм, в которых начинающему разработчику сложно разобраться самостоятельно. Неотъемлемый компонент ЭУП, – автоматизированное тестирование, – помогает экономить время как обучаемого, так и обучающего, исключает ошибку при проверке и наиболее быстро и объективно оценивает результаты [2].

Существует множество путей создания обучающих программ. Чаще всего ЭУП создаются в веб-формате. Это можно обосновать относительной простотой разработки и отсутствием многих ограничений, связанных с платформозависимостью. Однако зачастую пользователи предпочитают иметь отдельное самостоятельное приложение, что особенно актуально для мобильных платформ. Нативные приложения имеют высокую производительность, но возникает сложность – они могут быть недоступны на устройствах разного типа.

В связи с вышеизложенным, задача разработки кроссплатформенного электронного учебного пособия по основам программирования на языке C++ является актуальной. Также, разработанное приложение может быть рекомендовано обучающимся в Арзамасском политехническом институте по направлениям подготовки «Прикладная математика» и «Информационные системы и технологии».

Постановка задачи. Требуется разработать кроссплатформенное приложение для изучения программирования на C++. На главной странице должны располагаться кнопки для выбора занятий по разделам. После выбора конкретного раздела должна появляться страница с вкладками, содержащая следующие элементы – лекцию, практическое занятие и

тестирование по теме. Лекции должны быть разделены по параграфам, а практические работы – по заданиям. В десктопной версии приложения необходимо также добавить встроенные компилятор и редактор блок-схем для того, чтобы можно было выполнять практические и лабораторные работы без использования какого-либо дополнительного программного обеспечения.

Как уже было отмечено, конечному пользователю важно, чтобы приложение было доступно на платформах и устройствах разного типа. При разработке под разные платформы заметны существенные различия в программных интерфейсах, в подходах к построению графического интерфейса, и программированию – используются разные среды и языки, что заставляет заниматься «переводом» исходного кода, что, в свою очередь, не сказывается положительно на сроках разработки.

Xamarin представляет собой единый инструмент, позволяющий создавать приложения сразу для множества платформ, пользуясь средой разработки Visual Studio и языком C# (или F#). Текущая версия Xamarin 5.0 (вышла 5.01.2021) позволяет разрабатывать приложения под ОС Android, iOS, UWP (Universal Windows Platform), Tizen, WPF, GTK# (Linux), MacOS, Watch OS и TV OS. Таким образом, на современном этапе Xamarin охватывает все самые распространенные платформы. При этом на Xamarin создается единый код для всех платформ, есть прямой доступ к нативным API каждой платформы и можно программировать, используя .NET и производительный и ясный C# [5; 6].

Xamarin работает поверх фреймворка Mono, который представляет собой воплощение системы .NET Framework на базе свободного ПО. Mono, в свою очередь, может работать поверх разных платформ – Windows, Android, Linux, и т. д. На уровне каждой конкретной платформы Xamarin полагается на ряд субплатформ (Xamarin.Android, Xamarin.iOS, и т. д.). Через эти субплатформы приложения могут направлять запросы к прикладным интерфейсам на устройствах под управлением различных ОС. В итоге благодаря этим платформам возможно отдельно разрабатывать приложения для Android, iOS, и т. д. Но наиболее интересной особенностью Xamarin является возможность создавать кроссплатформенные приложения. Данная возможность представлена технологией Xamarin.Forms, которая работает уровнем выше субплатформ. Xamarin.Forms с помощью рендереров (объектов для связи контролов на XAML/C# с нативными контролами) транслируют визуальные компоненты Xamarin.Forms в графический интерфейс, специфичный для каждой платформы [6].

При создании приложения в папке решения формируются несколько проектов: главный, описывающий общую, кроссплатформенную логику, а также платформозависимые проекты (.Android, .iOS и .UWP). Главный проект со всеми файлами компилируется в библиотеку dll, а остальные три проекта содержат ссылки на него. Кроме этого, они также содержат, в частности, ссылки на свои специфические библиотеки (Xamarin.Forms.Platform.Android, Xamarin.Forms.Platform.iOS, Universal Windows), которые определяют для своей мобильной платформы статический метод Xamarin.Forms.Init(), выполняющий инициализацию системы Xamarin.Forms. Разработка в основном ведётся в главном проекте.

В качестве основы приложения была взята оболочка Xamarin.Forms Shell, нацеленная на упрощение создания кроссплатформенных мобильных приложений, и включающая в себя следующий функционал: удобный пользовательский интерфейс навигации с боковым меню и вкладками, а также схему навигации на основе URI и встроенный обработчик поиска.

Оболочка Shell прекрасно сочетается с использованием архитектурного паттерна MVVM (Model-View-ViewModel), основанного на привязке данных. Привязка является одним из ключевых моментов разработки на Xamarin. Она частично упрощает понимание кода, уменьшает связность между файлами, делает код короче и в общем случае убирает необходимость в использовании обработчиков событий. Паттерн проектирования MVVM предусматривает разделение проекта на три части – Models (модели – используемые приложением данные), Views (представления – визуальный интерфейс) и ViewModels (модели представлений – логика по получению данных из модели, которые затем передаются представлению) [5]. При этом скрипты из Views и ViewModels не содержат прямых ссылок друг на друга и взаимодействуют только с помощью механизма привязки.

Дизайн приложения можно разрабатывать как с помощью XAML-разметки, так и напрямую на языке C#, однако удобнее разделять построение дизайна от логики приложения, поэтому создавать его лучше с помощью XAML, а в C#-коде определять только логику.

На главной странице расположены кнопки для выбора занятий по темам. Нажатие на кнопку переводит пользователя на специальную навигационную страницу с вкладками, на которой отображается содержимое соответствующего раздела (рис. 1).

В мобильной версии приложения доступны нижние вкладки «Лекция», «Практика» и «Тест». Лекционные занятия разделены по параграфам, а практические – по заданиям. Навигация между параграфами и заданиями осуществляется с помощью верхних вкладок. В десктопной версии приложения есть еще две вкладки – «C++ компилятор» и «Редактор блок-схем». По умолчанию после выбора занятия загружается страница «Лекция».

В основу ЭУП было положено учебное пособие [1], разработанное сотрудниками кафедры «Прикладная математика» Арзамасского политехнического института.

Приложение взаимодействует с внедренной базой данных (БД) SQLite, в которой хранится весь учебный контент, с помощью Entity Framework Core (EF Core). EF Core предоставляет объектно-ориентированный способ доступа к реляционным и нереляционным БД в среде .NET. Таким образом, EF Core можно рассматривать как инструмент объектно-реляционного отображения (ORM), использующий язык LINQ для определения запросов к БД и возврата данных в связанные экземпляры классов .NET [1, с. 61]. Тексты лекционных и практических занятий хранятся в БД в виде html-строк и выводятся в элемент WebView (рис. 1).

Вывод информации из БД на экран реализован с помощью механизма привязки. Для установки привязки у объекта-цели определяется свойство BindingContext. В качестве значения оно принимает источник привязки (в данном случае, объект-запись БД), который устанавливается в коде C# либо с помощью XAML-разметки.

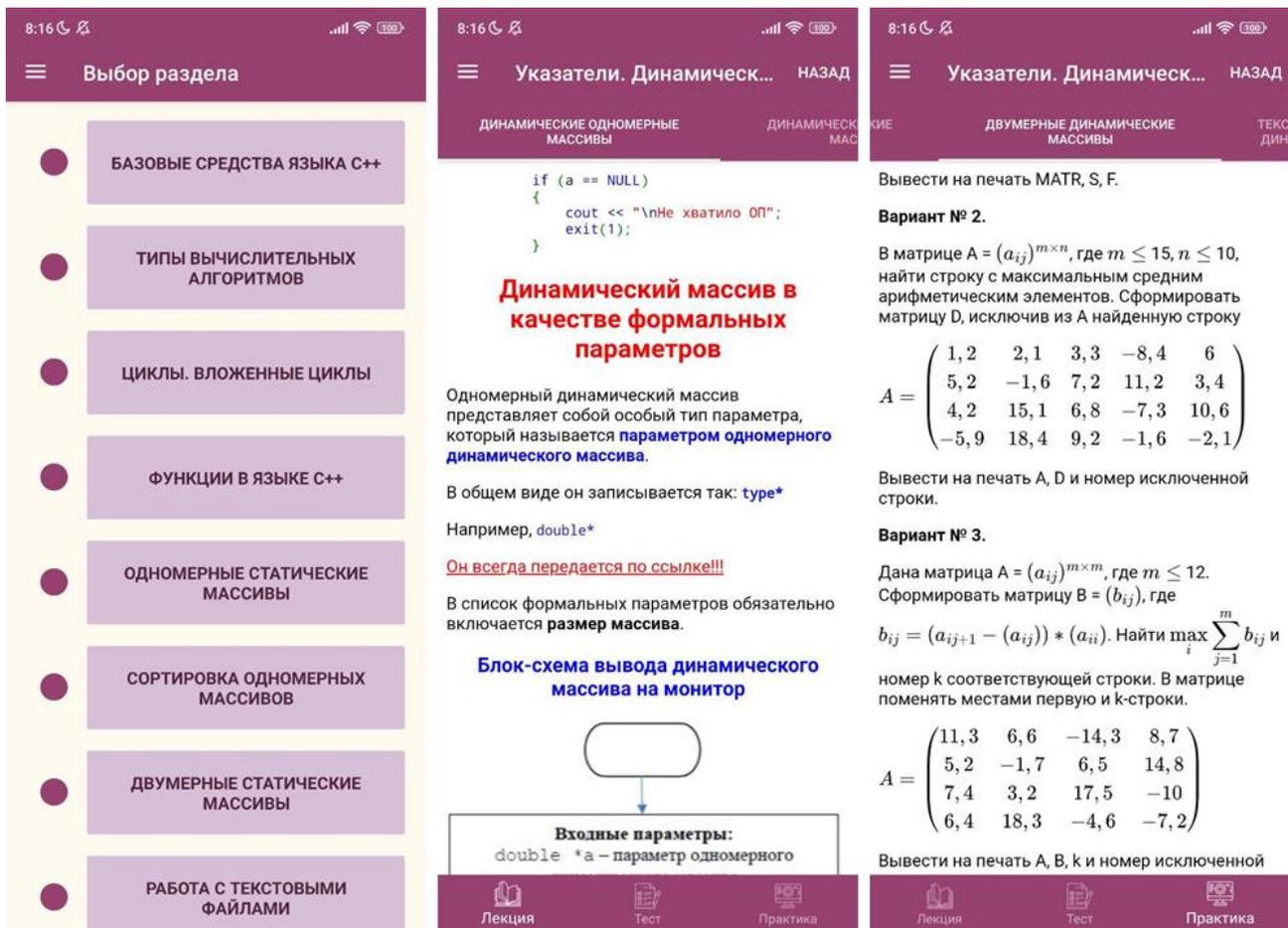


Рис. 1. Скриншоты (слева направо) главной страницы, лекции и практического занятия

Страница тестирования заслуживает особого внимания. При переходе на вкладку «Тест» появляется всплывающее сообщение, требующее от пользователя подтверждения для начала тестирования (рис. 2). При отрицательном ответе приложение возвращается назад к лекционной странице. При положительном ответе появляется ещё одно сообщение, напоминающее пользователю количество времени, которое отведено на прохождение теста. После его принятия начинается обратный отсчет и предоставляется возможность отвечать на вопросы. Вопросы появляются в случайном порядке. На экране демонстрируется оставшееся время, номер вопроса, сам вопрос (в текстовом виде и с картинкой, если необходимо) и кнопки с вариантами ответов на него. При выборе варианта ответа кнопка подсвечивается зелёным или красным цветом (в зависимости от того, правильный ли ответ был выбран), и совершается переход к следующему вопросу (рис. 2). После ответа на последний вопрос, по окончании времени или в любое другое время по желанию тестируемого щелчком по кнопке «Завершить сейчас!» можно подвести итоги теста. Тогда программа подсчитает число правильных ответов, процент верного выполнения теста и выведет на экран сообщение с этими данными и некоторыми комментариями (рис. 2), после принятия которого приложение вернётся к лекционной странице.

Элементы «Компилятор» и «Редактор блок-схем» представляют собой онлайн-сервисы, загружаемые в элемент WebView (рис. 3).

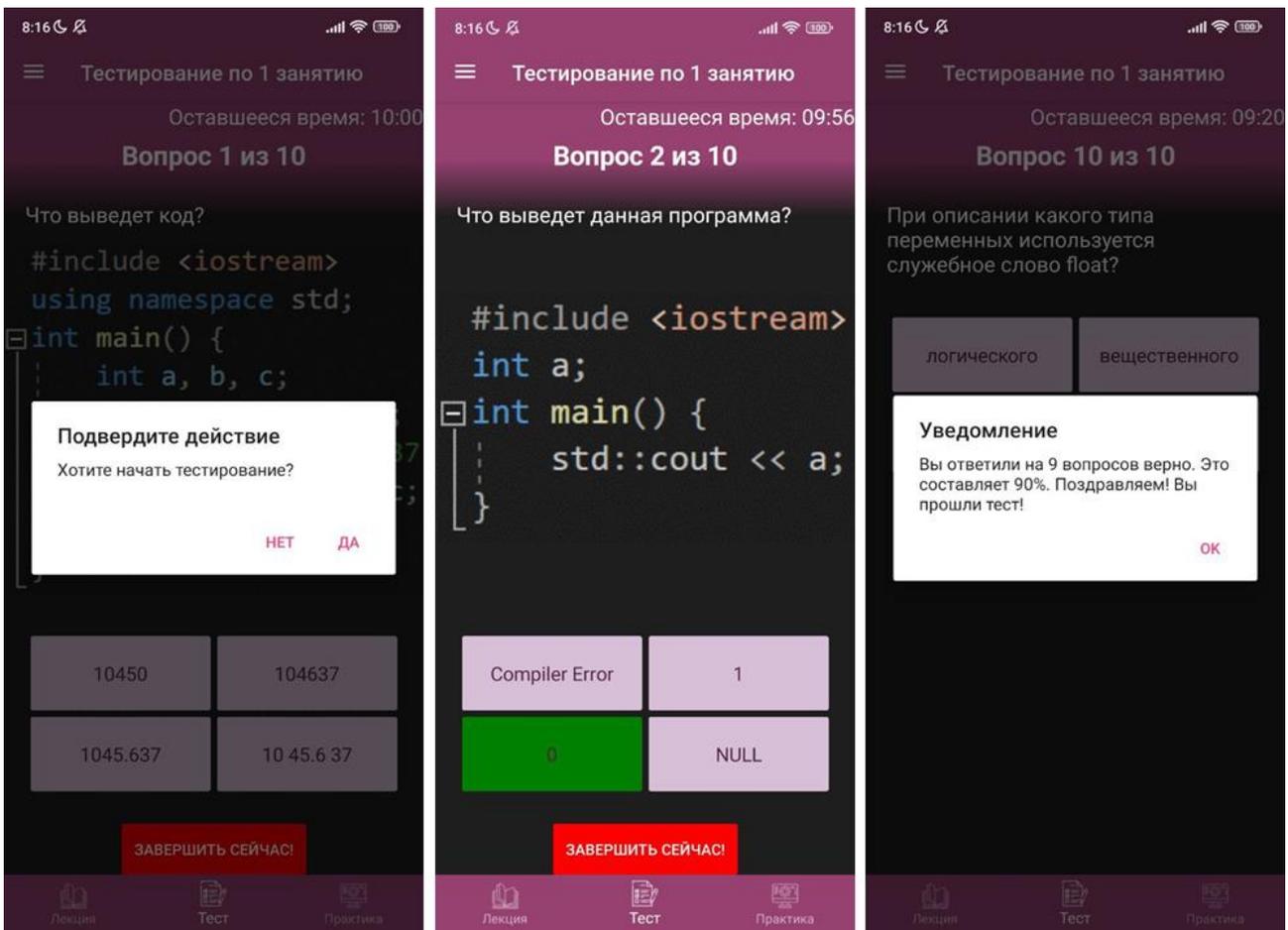


Рис. 2. Скриншоты страницы тестирования

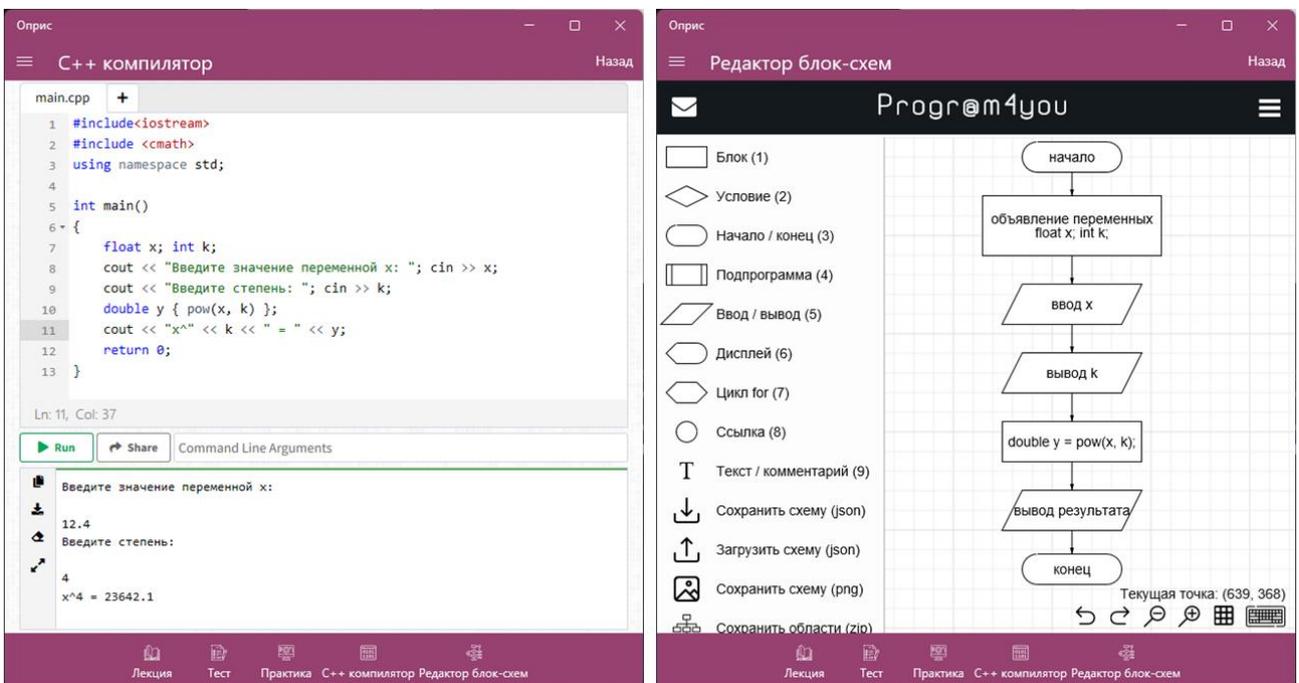


Рис. 3. Скриншоты страниц «С++ компилятор» (слева) и «Редактор блок-схем» (справа)

Таким образом, было разработано кроссплатформенное приложение, представляющее собой электронное учебное пособие по программированию на языке С++.

Разработанный программный продукт успешно прошёл первичную апробацию со студентами Арзамасского политехнического института, обучающимися по направлениям подготовки «Прикладная математика» и «Информационные системы и технологии». Учащиеся смогли установить приложение и пользоваться им на устройствах различного типа.

Установочные файлы приложения с подробно описанными шагами для установки, а также с решениями часто возникающих проблем опубликованы на веб-сервисе для хостинга IT-проектов GitHub (URL: <https://github.com/LeoKhariton/Opric>).

В будущем планируется добавление личных кабинетов для студентов и преподавателей с возможностью ведения статистики, а также перенос базы данных приложения на сервер.

Литература

1. Лазарева А.Б., Троицкий А.В., Жилина Т.Е. Использование языка программирования C++ для решения задач высшей математики. Н. Новгород, 2012. 334 с.
2. Ромазанова О.В., Богачева Н.В. Достоинства и недостатки компьютерного тестирования // Казанский вестник молодых учёных. 2018. Т. 2. № 5(8). С. 49-51.
3. Смит Дж. П. Entity Framework Core в действии / пер. с англ. Д.А. Беликова. М.: ДМК Пресс, 2022. 690 с.
4. Сулла Р.В., Красовская Л.В. Информационные технологии в школьном образовании // Сборник статей Международной научно-практической конференции. Белгород, 2017. С. 521-523.
5. Bennett J. Xamarin in Action. Creating native cross-platform mobile apps // Manning, 1st edition, 2018. 608 p.
6. Petzold Ch. Creating Mobile Apps with Xamarin.Forms. Cross-platform C# programming for iOS, Android and Windows Phone. – Microsoft Press, Xamarin, Inc., 2016. 1187 p.

© Харитонов Л.С., Устюжанин И.П., Бочкарева М.Д., Эварт Т.Е., 2023

ПРОЕКТНАЯ ДЕЯТЕЛЬНОСТЬ ПО ИНФОРМАТИКЕ КАК УСЛОВИЕ РАЗВИТИЯ КРЕАТИВНОГО МЫШЛЕНИЯ

Процесс цифровизации ставит перед системой образования задачу воспитать личность, обладающую такими умениями, как коммуникация, кооперация, критическое мышление и креативность. Для достижения заявленных результатов педагогу необходимо использовать творческий подход в реализации новых методов и технологий обучения, которые способствуют формированию самостоятельно находить нужную информацию, выдвигать предположения и делать выводы, аргументировать свои мысли, развитию креативного мышления обучающихся.

Креативное мышление – это «компонент функциональной грамотности, под которым понимают умение человека использовать свое воображение для выработки и совершенствования идей, формирования нового знания, решения задач, с которыми он не сталкивался раньше» [1]. По версии PISA креативное мышление – это «критическое осмысливание своих разработок, совершенствование их» [3]. Также креативное мышление трактуется как «способность нестандартно воспринимать окружающий мир и генерировать принципиально новые идеи» [1].

Выделяют следующие компоненты креативного мышления (по Д.П. Гилфорду) [2], представленные на рисунке.



Рис. Компоненты креативного мышления

Таким образом, креативное мышление является необходимым инструментом, позволяющим обучающемуся состояться в жизни, как неординарной личности, способной мыслить и действовать «не как все». В связи с вышесказанным актуальность развития креативного мышления обучающихся в современных условиях является неоспоримой.

Учителя для решения поставленной задачи используют различные методы, формы и средства обучения. Проектная деятельность, как вид творческой деятельности, видится нами одним из эффективных средств развития креативного мышления обучающихся.

Проектную деятельность исследователи определяют, как «совместную учебно-познавательную, творческую или игровую деятельность обучающихся, имеющую общую цель, согласованные методы, способы деятельности, направленную на достижение общего результата деятельности» [4]. Непременным условием проектной деятельности является наличие заранее выработанных представлений о конечном продукте деятельности, этапов проектирования (выработка концепции, определение целей и задач проекта, доступных и оптимальных ресурсов деятельности, создание плана, программ и организация деятельности по реализации проекта) и реализации проекта, включая его осмысление и рефлексию результатов деятельности. При работе над проектом обучающийся развивает в себе умения самостоятельного планирования своей деятельности, анализа, синтеза, которые в свою очередь способствуют развитию креативного мышления.

Приведем примеры развития креативного мышления на каждом этапе проекта:

1. Погружение в проблему. Организовать процесс развития креативного мышления обучающегося на данном этапе можно, предложив ему самостоятельно определить что он хочет исследовать, т. е. выявить проблему. В данном случае проявляются такие компоненты креативного мышления, как оригинальность и беглость обучающегося.

2. Планирование деятельности. Процесс развития креативного мышления на данном этапе происходит посредством планирования своей деятельности, направленной на формирование осознанного выбора темы проекта. Ход проекта, этапы исследования, предмет и методы исследования ученик также определяет сам, что развивает такие компоненты креативного мышления, как оригинальность, гибкость и точность.

3. Осуществление деятельности. На этом этапе креативное мышление можно развивать, предложив обучающимся самостоятельно подобрать материал, информационные источники. Отталкиваясь уже от выбранного материала, они могут выстраивать свою работу, брать ключевую идею из материала, менять ее под себя и т. д. Единственным условием выбора материала является адекватность и целесообразность его использования. Этот этап формирует такие компоненты креативного мышления, как беглость и гибкость.

4. Оформление результатов. В ходе работы над проектом, ученик сталкивается с этапом, где нужно разработать продукт проектной деятельности. И этот этап дает обучающемуся в полной мере раскрыть себя с творческой стороны и продемонстрировать свое креативное мышление. Продукт можно представить в различном виде. Например, в виде плаката, мультфильма, игры и т. д. Этот этап позволяет обучающемуся включить свою фантазию и воображение, это здесь приветствуется, а также сформировать такой компонент креативного мышления, как оригинальность.

5. Представление результатов. Креативное мышление на данном этапе происходит за счет осуществления организации представления результатов проделанной работы. Результат работы можно представить в виде классной мини-конференции, публичной защиты, ролевой

игры и т. д. Это тот самый этап, где можно в полной мере включить свое креативное мышление и представить результат проделанной работы в интересном и творческом виде. На данном этапе формируются такие компоненты креативного мышления, как оригинальность и точность.

б. Оценка результатов. Организовать процесс развития креативного мышления на данном этапе можно, предложив обучающимся самим разработать либо внести некоторые изменения в критерии оценивания проекта. Данный этап позволяет сформировать такие компоненты креативного мышления, как беглость, гибкость и оригинальность.

С точки зрения потенциала школьных дисциплин для развития креативного мышления информатика является более перспективной, так как направлена на развитие у обучающихся навыков работы с информацией, умения ориентироваться в информации, работать с цифровыми технологиями.

В качестве примера проектной деятельности, рассмотрим проект по информатике «Одна семья в истории России», имеющий межпредметные связи с историей. Данный проект разработан по классификации Полат Е.С. [5].

Таблица

Пример проекта

Компоненты проекта	Описание проекта
Тема проекта	«Одна семья в истории России»
Класс	7
Объект исследования	Династия Романовых
Цель проекта	Для учителя: привлечение внимания обучающихся к истории России и значению в ней династии Романовых, воспитание гражданственности и патриотизма. Для ученика: систематизация теоретического материала о династии Романовых.
Задачи проекта	1. Изучить историю становления династии Романовых. 2. Выделить основные события правления династии Романовых. Создать сайт о членах семьи Романовых и основных событиях. 3. Разработать стратегию «Спаси Романовых».
Проблема	Как правление династии Романовых влияет на историю современной России?
Этапы реализации проекта (включая исследовательский), краткое их описание	1 этап. Верховный тайный совет. (Подготовительный этап) Группы: 1. Боярская дума (изучает теоретическую часть, связанную с династией Романовых, выделяет основные моменты, которые необходимо представить на сайте) 2. Земский собор (анализируют конструкторы для сайтов, создают сайт и древо семьи Романовых (https://clck.ru/34qFK4)) 3. Воевода (создают стратегию по данной теме, и проводят эксперимент (https://clck.ru/34qFKL)) 2 этап. Уклад жизни царской семьи. (Исследование) 1. Деление на подгруппы. 2. Изучение литературы по истории династии Романовых. 3. Анализ различных платформ для создания сайта. 4. Обобщение и систематизация собранного материала. 3 этап. Падение дома Романовых. (Отчет и защита проекта) 1. Создание сайта и стратегии.

	<p>2. Проведение эксперимента. 3. Отчет и защита проекта.</p>
Исследовательский этап проекта (полное описание)	<p>Цель исследования: создание стратегической игры по спасению семьи Романовых. Методы исследования: теоретический, экспериментальный. Оборудование для проведения исследования: ПК, конструктор сайтов. Ход исследования: 1. Разделить обучающихся на группы. 2. Определить задачи для каждой группы: 1 группа находит весь теоретический материал. 2 группа анализирует различные платформы и создает сайт. 3 группа разрабатывает стратегию. 3. Каждая группа выполняет задание. 4. Все группы делают выводы о значимости династии Романовых на историю России.</p>
Продукт проекта	Продуктом проекта является игра-стратегия, ее смысл заключается в том, что дети должны подойти творчески к данной задаче, и разработать план стратегии, который поможет им спасти Романовых. А также сайт «Династия Романовых».
Перспектива проекта	Заключается в том, что в дальнейшем можно представить различные исторические события в совершенно разных вариантах, например таких как веб-квест, создание собственного фильма.
Классификация проекта по Е.С. Полат	Доминирующая в проекте деятельность: исследовательский проект.
	Предметно-содержательная область: межпредметный проект (информатика, история).
	Способ общения в процессе проектной деятельности: непосредственное общение
	Характер координации проекта: проект с явной координацией.
	Характер контактов: внутришкольный проект.
	Количество участников проекта: групповой.
	Продолжительность проекта: краткосрочный (4 занятия).

В ходе работы над проектом учитель выполняет роль консультанта, помогает, направляет деятельность обучающихся в методически нужное русло. Разрабатывая собственный проект, у обучающихся развиваются навыки работы в команде, коммуникативность, креативное мышление и самостоятельность.

Таким образом, использование проектной деятельности по информатике обладает высоким потенциалом для развития креативного мышления обучающихся, так как в ходе работы над проектом у обучающихся развиваются и формируются умение работать с информацией, в том числе представить информацию в доступном, наглядном, эстетичном и творческом виде; опыт публичной демонстрации и презентации своего проекта, способность изложить и высказывать свою точку зрения, защищать и отстаивать свои интересы и идеи; навыки командной, групповой и самостоятельной работы; умение делать выбор и принимать самостоятельные решения.

Литература

1. Величко Ю.В. О соотношении понятий «креативность», «творческие способности», «творческое воображение», «творческое мышление» в исследованиях отечественных и зарубежных психологов. 2014. №2(3). С. 629-631.
2. Дружинин В.Н. Психология общих способностей. 2-е изд. Санкт-Петербург: Питер, 1999. 368 с.
4. Михалкина Е.В., Никитаева А.Ю., Косолапова Н.А. Организация проектной деятельности. Р/на-Дону: Изд-во ЮФУ. 2016. 146 с.
3. Наумова А.П. Креативное мышление в цикле исследований PISA-2022 // Молодой ученый. 2022. №1(396). С. 249-251.
5. Полат Е.С. Метод проектов. Минск: БГУ, 2003. 244 с.

© Шахмирзоева М.И., 2023

УДК 004.9

Шкуратова Ю.А.

Волгоградский государственный социально-педагогический университет
г. Волгоград, Россия

КОМПЬЮТЕРНОЕ ТЕСТИРОВАНИЕ В СИСТЕМЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ СТУДЕНТОВ

Любая человеческая деятельность нуждается в оценке и проверке. Причем как качественно, так и количественно. То же относится и к знаниям. Однако всегда возникает вопрос в целесообразности, адекватности, объективности и валидности результатов оценки и проверки. На фоне возросшей актуальности дистанционного обучения и практического применения информационных технологий в системе образования данный вопрос становится проблемным и предельно актуальным. При этом вслед за В.С. Аванесовым отметим, что сегодня есть необходимость перехода от классно-урочных форм массового обучения к индивидуальным формам по образованию каждого, что особенно актуально в условиях дистанционного обучения [1, с. 132-140]. Но переход к индивидуальным формам обучения возможен при создании адаптивных образовательных технологий, использующих автоматизированный учёт реальных учебных достижений обучающихся, который невозможен без использования качественных тестов и соответствующих для них программ для автоматизации тестирования в условиях дистанционного обучения [1, с. 132-140; 7, с. 29-37].

В свете всего вышеописанного контроль уровня знаний – это неотъемлемая составная часть процесса обучения, причем предельно важная. Посредством контроля обеспечивается реализация обратной связи по принципу «обучающийся – преподаватель». В работах В.С. Аванесова и В.А. Углева выделяются более часто используемые на практике подходы к тестированию обучающихся: текущий и итоговый контроль, где наибольший потенциал заложен в текущий контроль, в следствии его обучающих возможностей, воздействия на качество обучения и оперативность [1, с. 132-140; 13, с. 19]. Функции контроля знаний: контролирующая, обучающая, воспитательная, диагностирующая и мотивирующая.

Ныне педагог имеет в своем распоряжении огромное множество специализированного программного обеспечения, предназначенного для проведения контроля уровня знаний обучающихся. В распоряжении современного преподавателя масса мультимедийных продуктов и инструментов для создания также и своих авторских средств контроля знаний. Причем инструментарий в наличии как иностранного, так и отечественного происхождения. Разработанные с использованием информационных технологий такие инструменты прекрасно адаптивны, открыты и стандартизированы. И главное – это наличие возможности, взяв готовое решение, расширить или же нарастить до того уровня, что необходимо именно под какие-то конкретные условия. Современные средства успешно позволяют проводить индивидуальный и групповой контроль знаний, причем также и в удаленном, дистанционном формате [3, с. 71-75].

Современные системы контроля знаний достаточно универсальны и автоматизированы, позволяют проводить не просто контроль, но и самоконтроль, а это позволяет смело их считать также и тренажерами при подготовке к итоговым контролям и экзаменам.

Обзор иностранной и отечественной литературы. В 1864 году Дж. Фишер, британский исследователь, впервые официально применил тест для проверки знаний учащихся. Несколько позже, в 1883 году, Ф. Гальтон – английский психолог – разработал теоретические основы тестирования. Стоит подчеркнуть тот факт, что развитие тестирования положило начало активного внедрения математических методов в педагогику и психологию.

В отечественном образовательном процессе составление и применение тестов впервые находит свое отражение в педагогике и психологии в 1920-х годах. Именно в 1926 году была опубликован первый набор тестов для школ (<http://www.effecton.ru/199.html>). Вопросы тестовой теории отражены в трудах В.С. Аванесова, Б. Блума, С.И. Григорьева, Беспалько В.П., А.Н. Майорова, П. И. Пидкасистого, И.П. Подласого, А.О. Татур и многих других.

Педагогу чрезвычайно важно обладать актуальными сведениями о том, в какой степени учащиеся воспринимают и усваивают учебный материал. Однако, традиционный контроль является длительной, трудоемкой и не всегда в достаточной мере объективной частью рутины педагога. Использование информационных технологий в значительной мере облегчает и систематизирует процесс осуществления контроля. Проблема реализации связанных с контролем функций по мнению ряда исследователей, таких как А.П. Карпенко, А.С. Домников, В.В. Белоус, подразделяется на три основных составляющих: подготовка к контролю, проведение контроля, обеспечение обратной связи [4, с. 56-59].

Под тестом как правило понимают систему заданий, позволяющую измерять уровень усвоения знаний обучающихся, степень развития их определенных качеств, способностей или особенностей личности. Тестовый контроль отличается объективностью измерений результатов обучения, так как они ориентированы не на субъективное мнение человека, оценивающего знания обучающихся, а на объективные эмпирические критерии [10, с. 38-44].

Тестирование обладает рядом преимуществ, в сравнении с иными, а именно: высокая научная обоснованность, технологичность, точность, стандартизованность, объективность оценивания результатов, отличная сочетаемость с современными образовательными технологиями.

Можно выделить следующие этапы в эволюции контроля знаний [11, с. 433–440]:

1. *Традиционный контроль.* Характеризуется использованием предварительно заготовленных вариантов заданий, а затем проверяется педагогом для оценивания результатов учащихся.

2. *Контроль с использованием не компьютерных средств.* Характеризуется предварительно заготовленными бланками или карточками, в которых содержатся контрольные задания и тесты. Требуется заполнить эти бланки или выполнить задания по карточкам. После учитель выполнит проверку работ, используя специальные шаблоны и таблицы с готовыми ответами, сопоставляя их с ответами учащихся.

3. *Контроль с использованием технических устройств.* Каждый учащийся, индивидуально, получает набор определенных тестовых заданий. Он выполняет эти задания, поочередно вводя свои ответы и решения в техническое устройство, где уже оно автоматически сверяет введенное с правильными вариантами. После технического устройство само рассчитывает оценку. В некоторых случаях учащемуся демонстрируются его ошибки. А иногда, если это заложено в программу устройства, есть возможность вернуться к неправильно выполненным заданиям и повторить попытку.

4. *Компьютерный контроль.* Этот вид контроля осуществляется с использованием специального программного обеспечения (ПО) на компьютерах. Довольно часто такое ПО позволяет сформировать индивидуальный набор контрольных тестовых материалов. Демонстрация заданий происходит на экране монитора. Учащийся посредством графического интерфейса пользователя (GUI) взаимодействует с программным обеспечением, выполняя задания. После выполнения представленного набора контрольных материалов ПО средствами аппаратной части компьютера производит проверку и анализ ответов и решений, а после выполняет расчет оценки, как правило, с демонстрацией учащемуся результатов. Компьютер также обеспечивает долговременное хранение результатов контроля.

5. *Удаленный контроль.* Со все большим внедрением сети Интернет в образовательный процесс и на фоне разнообразных социальных и экономических явлений возрастает роль дистанционного обучения. При таком обучении преподаватель и учащийся могут находиться на немыслимом расстоянии друг от друга. Однако проблема контроля знаний по-прежнему особенно актуальна. Данный вид контроля отличается от всех прочих еще и тем, что у учащихся появляется возможность свободно выбирать место и время проведения тестирования.

Проведение тестирования представляет из себя один из видов педагогического контроля, также как рейтинг и мониторинг.

Большинство авторов рассматривает тестовые технологии в образовательном процессе как элемент контроля. Таким образом, компьютерное тестирование – это элемент контроля, позволяющий с минимальными временными затратами объективно проверить знание значительного количества обучаемых. В исследованиях Кофтан Ю.Р., Остапенко В.А. и др., обучающие компьютерные тесты, определяются как система взаимосвязанных тестовых заданий, предоставляющая возможность проводить оценивание соответствия знаний обучающегося некоторой экспертной модели знаний в определенной предметной области [6]. Компьютерные тесты вызывают положительную оценку самих тестируемых. Заметными преимуществами компьютерных тестов являются: автоматическая проверка результатов и исключение или минимизирование человеческого фактора. Следует отметить и психологический аспект – в процессе проверки знаний обучаемыми преподаватель рассматривается не как оппонент, а как союзник.

Постановка задачи исследования. Задача теоретического исследования, описываемого в данной работе, сводится к *анализу особенностей компьютерного тестирования в системе дистанционного обучения.*

Описание исследования. Проведение компьютерного тестирования может выполняться как во время урочных занятий, так и во внеурочное время, выступая своеобразной разновидностью самостоятельной работы.

Вариант во время урочных занятий хорошо подходит для дисциплин, проходящих в компьютерных аудиториях. Однако, предпочтительнее проведение во внеурочное время по процедуре представленной на рисунке 1.

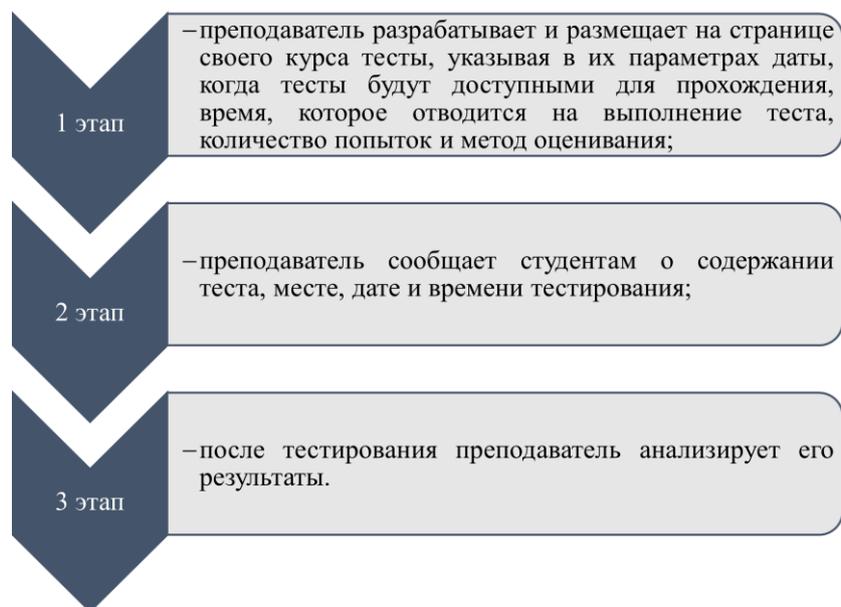


Рис. 1. Процедура проведения тестирования во внеурочное время

При этом данная процедура не обязательно предполагает физическое присутствие преподавателя в компьютерном классе в это время. Для проведения непосредственно самого тестирования в университете выделяются компьютерные классы, оборудованные всем необходимым.

Диагностические тесты обладают также рядом важнейших критерием (рис. 2).

действенность теста:

- полнота, всесторонность проверки, пропорциональность представления всех элементов изучаемых знаний, умений;

надежность теста:

- характеризуется стабильностью, устойчивостью показателей при повторных измерениях с помощью того же теста или его равноценного заменителя,
- значительно зависит от трудности их выполнения, которая определяется соотношением правильных и неправильных ответов на тестовые вопросы;

дифференцированность теста:

- способность теста отделить тех, кто усвоил материал на необходимом уровне, от тех, кто заданного уровня не достиг.

Рис. 2. Важнейшие критерии диагностических тестов

На рисунке 3 представлен набор правил, которым необходимо руководствоваться при составлении тестов [14, с. 432].

однозначность заданий:

– тестовые задания не должны допускать произвольного толкования;

однозначность ответов:

– должна быть исключена возможность формулирования многозначных ответов;

соответствие изученному:

– нельзя включать ответы, неправильность которых на момент тестирования не может быть обоснована учащимися;

подбор дистракторов:

– неправильные ответы должны конструироваться на основе типичных ошибок и должны быть правдоподобными;

уникальность:

– вопросы не должны повторять формулировок учебника.

Рис. 3. Набор правил при составлении тестов

После того как тест составлен, необходимо выполнить апробацию. Однако, на процессе апробации не стоит останавливаться. Требуется постоянно мониторить качество тестовых заданий, уделяя внимание вновь добавленным [9, с. 432 с].

Для анализа тестовых заданий могут использоваться следующие статистические параметры:

среднеквадратичное отклонение (СКО):

– измеряет разброс баллов, полученных испытуемыми при ответе на конкретное задание теста;

индекс дифференциации (ИД):

– является грубым индикатором способности конкретного тестового задания отделить более успешных испытуемых от менее успешных;

коэффициент дифференциации (КД):

– другой способ измерения способности конкретного задания разделять сильных и слабых испытуемых.

Рис. 4. Статистические параметры

Коэффициент дифференциации – это коэффициент корреляции между множеством значений ответов, полученных испытуемыми при выполнении конкретного задания, с результатами выполнения ими теста в целом [5, с. 214].

Классическим примером теста обученности является любой тест с одной попыткой, после его выполнения тестируемый получает итоговую оценку. Все варианты ответов при

этом демонстрируются, выделяясь разным цветом как правильные и неправильные. Тестируемый получает возможность проанализировать свой ответ и понять, где он ошибся или, наоборот, ответил верно. Это показатель того, что учащийся думает и анализирует и, как следствие, он обучается [12, с. 76-81].

Тест можно настроить также и иначе для того, чтобы тестируемые смогли обдумать и проанализировать ход выполнения теста и исправить возникающие ошибки. Суть заключается в том, чтобы не демонстрировать правильные и неправильные ответы, вывести результат и баллы, но при этом позволить пройти заново некоторое число раз, перекомбинируя вопросы и варианты ответов.

Исходя из вышеописанного, *обучающий тест* можно определить как совокупность заданий, ориентированных на определение уровня усвоения небольших по объему аспектов содержания обучения, которые предполагают предоставление студенту возможности анализа и, возможно, исправления своих ошибок [2, с. 390-400; 8, с. 296]. Тестируемые могут несколько раз проходить тест, не учитывая оценки за результат работы на данном этапе. В случае *контрольно-обучающих тестов* студенту уже демонстрируется оценка его деятельности.

Разработка обучающих тестов связана с большим объемом работы, поэтому важно чтобы у учителя были инструменты, облегчающие их создание. К таким инструментам можно отнести сетевые сервисы для создания тестов и викторин. В настоящий момент существует множество сетевых сервисов для разработки и использования обучающих компьютерных тестов (Classmarker, Madtest, Yandex Forms, Google Forms и др.), позволяющих повысить мотивацию обучающихся за счет визуализации и интерактивности, добавления элементов соревнования, что выделяет их на фоне стандартных технологий тестирования [15, с. 447-450].

Заключение. Результаты проведенного теоретического исследования показывают, что существует значительное множество характеристик компьютерных тестов, где особым потенциалом в рамках дистанционного обучения обладает обучающее тестирование. Было выявлено, что особенностью обучающего теста является то, что он предназначен для активной работы обучающихся с предложенными заданиями, при этом ему обеспечивают интерактивную пошаговую поддержку в виде небольших подсказок в виде консультаций, чтобы помочь осмыслить свои действия по выбору и получению ответов на предложенные задания. Выявлено, что в большинстве случаев компьютерное обучающее тестирование знаний обучающихся с использованием информационных технологий достаточно эффективно, может существенно повысить мотивацию обучающихся, которая помогает более эффективному построению их самостоятельной работы, снижают субъективизм при контроле знаний, помогают индивидуализировать темп работы, уровень усвоения и закрепления учебного материала.

Литература

1. Аванесов В.С. Проблема соединения тестирования с обучением // Народное образование. 2016. №7-8(1458). С. 132-140.

2. Гейн А.Г., Некрасов В.П. Компьютерные обучающие тесты как средство диагностики развития метакогнитивных компетенций // Новые образовательные технологии в вузе: Сб. тезисов докладов участников конференции (г. Екатеринбург, 18-20 февраля 2014 г.). Екатеринбург, 2014. С. 390-400.
3. Каменная Е.Ш., Ягафарова Х.Н. Функции текущего контроля знаний студентов технического профиля в педагогических условиях индивидуализации обучения // Известия ВГПУ. 2019. №7 (140). С. 71-75.
4. Карпенко А.П., Домников А.С., Белоус В.В. Тестовый метод контроля качества обучения и критерии качества образовательных тестов. Обзор // Машиностроение и компьютерные технологии. 2011. №4. С. 56-59.
5. Ким В.С. Тестирование учебных достижений. Уссурийск: Изд-во УГПИ, 2007. 214 с.
6. Кофтан Ю.Р., Остапенко В.А. Методические аспекты разработки обучающих и тестирующих курсов дистанционного обучения // Серия материалов Всероссийской школы-семинара «Информационные технологии в управлении качеством образования и развитии образовательного пространства». М.: Исследовательский центр проблем качества подготовки специалистов, 2000. С. 170-185.
7. Куликова Н.Ю. Опыт использования мобильных технологий для осуществления систематического и оперативного контроля знаний обучающихся // Дистанционное и виртуальное обучение. 2016. № 4(106). С. 29-37.
8. Майоров А.Н. Теория и практика создания тестов для системы образования. М.: Интеллект-центр, 2001. 296 с.
9. Михайлычев Е.А. Дидактическая тестология. М.: Народное образование, 2001. 432 с.
10. Одинаев Д.Ш. Тестовая форма контроля знаний обучающихся // Современное образование (Узбекистан). 2014. №8. С. 38-44.
11. Прокофьева Н.О. Вопросы организации компьютерного контроля знаний // Educational Technology & Society. 2006. №9(1). С. 433-440.
12. Смагина И.Л. Обучение студентов стратегиям самостоятельной учебной деятельности // Известия ВГПУ. 2019. №7 (140). С. 76-81.
13. Углев В.А. Модели и методы построения систем обучающего компьютерного тестирования на основе экспертных систем с элементами нечёткой логики: автореф. ... канд. тех. наук. Красноярск, 2009. 19 с.
14. Чельшкова М.Б. Теория и практика конструирования педагогических тестов. М.: Логос, 2002. 432 с.
15. Шкуратова Ю.А. Анализ сетевых сервисов для создания обучающих компьютерных тестов по информатике // Наука молодых – будущее России. Международная научная конференция перспективных разработок молодых ученых (г. Курск, 12-13 декабря 2022 г.). Курск, 2022. С. 447-450.

ОРГКОМИТЕТ

Председатель оргкомитета – Горлов Сергей Иванович, д-р физ.-мат. наук, профессор, ректор;

Заместитель председателя – Погоньшев Денис Александрович, канд. биол. наук, доцент, первый проректор, проректор по научной работе.

Члены оргкомитета

Шульгин Олег Валерьевич – канд. экон. наук, доцент, начальник управления научных исследований;

Давыдова Светлана Александровна – канд. пед. наук, доцент, декан факультета физической культуры и спорта;

Долгина Екатерина Станиславовна – канд. культурологии, доцент, декан гуманитарного факультета;

Иванов Вячеслав Борисович – канд. пед. наук, доцент, декан факультета экологии и инжиниринга;

Истрофилова Олеся Ивановна – канд. пед. наук, доцент, декан факультета педагогики и психологии;

Павловская Анастасия Анатольевна – канд. пед. наук, доцент, декан факультета искусств и дизайна;

Пащенко Оксана Ивановна – канд. пед. наук, доцент, декан факультета информационных технологий и математики.

СОДЕРЖАНИЕ

Абдразаков Д.А. АЛГОРИТМ АВТОМАТИЗАЦИИ ПРОЦЕССА ВЫДЕЛЕНИЯ СПРЯМЛЕННЫХ УЧАСТКОВ СПИРАЛЬНОЙ СТРУКТУРЫ ГАЛАКТИК	4
Аввакумов Д.А. ПРИЛОЖЕНИЕ ДЛЯ ВИЗУАЛИЗАЦИИ И АНАЛИЗА РЕЗУЛЬТАТОВ МОДЕЛИРОВАНИЯ БИОМЕХАНИЧЕСКОГО УЗЛА. ИНТЕРФЕЙС ПРИЛОЖЕНИЯ	9
Аксёнова Н.А., Соболев Д.Н., Сыч Д.С. РАЗРАБОТКА СИСТЕМЫ МАШИННОГО ЗРЕНИЯ ДЛЯ РАСПОЗНАВАНИЯ ОСОБЫХ ТОЧЕК УГЛОВ	14
Алеников Г.С. СОЗДАНИЕ РОБОТА С РАЗНЫМИ РЕЖИМАМИ УПРАВЛЕНИЯ	20
Браженец М.И. АНАЛИЗ И СРАВНЕНИЕ СРЕДСТВ РАЗРАБОТКИ WEB-ПРИЛОЖЕНИЯ ЭЛЕКТРОННОГО ЖУРНАЛА	25
Ваганова Е.С., Горбачев И.В. БАЗА ЗНАНИЙ НА ОСНОВЕ ОНТОЛОГИИ ДЛЯ ОБУЧЕНИЯ ТЕХНИЧЕСКОГО ПЕРСОНАЛА ПРИЕМНОЙ КОМИССИИ	30
Гребеникова С.А., Сметанникова Т.А. ИСПОЛЬЗОВАНИЕ ПРОГРАММ ТРЕХМЕРНОГО МОДЕЛИРОВАНИЯ В СОЗДАНИИ ЦИФРОВОЙ СКУЛЬПТУРЫ	35
Енин В.М. ЗАЩИТА И РЕЗЕРВАЦИЯ ДАННЫХ СИСТЕМЫ	41
Зубаль А.М. РАЗРАБОТКА АВТОНОМНОГО РОБОТА	45
Королук Я.А., Куликова Н.Ю. ОПЫТ РАЗРАБОТКИ И ИСПОЛЬЗОВАНИЯ ИНТЕРАКТИВНЫХ ОБРАЗОВАТЕЛЬНЫХ РЕСУРСОВ НА ОСНОВЕ HTML5 ДЛЯ ОНЛАЙН-ОБУЧЕНИЯ ШКОЛЬНИКОВ ИНФОРМАТИКЕ	48
Косовцов Н.А. РАБОТА С RFID МЕТКАМИ СТАНДАРТА ISO/IEC 15693 НА ПРИМЕРЕ ПРИЛОЖЕНИЯ ДЛЯ АНДРОИД	56
Костюченко П.А., Корнаухова М.А. РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ МОНИТОРИНГА УСПЕВАЕМОСТИ СТУДЕНТА	64
Куприянова Т.В., Поспелова Н.В. О РАЗРАБОТКЕ AR ПРИЛОЖЕНИЯ ДЛЯ КИБЕР-МУЗЕЯ	69
Лазорин Д.С. О ПОДХОДАХ К ОБЕСПЕЧЕНИЮ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ЦИФРОВЫХ ДВОЙНИКОВ	76

Ларин Д.Е.	МУЛЬТИЗАГРУЗОЧНЫЙ РЕАНИМАТОР И ЕГО УТИЛИТЫ	82
Логвинец В.А.	РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ КОНТРОЛЯ ЗНАНИЙ	87
Мироненко Е.В.	РАЗРАБОТКА ОНЛАЙН-ПЛАТФОРМЫ ДЛЯ ОБМЕНА ИЗОБРАЖЕНИЯМИ	95
Мордвинов А.В.	СОЗДАНИЕ ПРОТОТИПА ПЛАТФОРМЫ ДЛЯ РАЗРАБОТКИ ЭЛЕКТРОННЫХ УСТРОЙСТВ	102
Мороз А.С.	РЕДУКЦИЯ БОЛЬШИХ ДАННЫХ В РАСПРЕДЕЛЕННЫХ СИСТЕМАХ НА ОСНОВЕ ФУНКЦИОНАЛЬНОГО ПОДХОДА	107
Мурина А.В., Воронин В.В.	АНАЛИЗ ЭФФЕКТИВНОСТИ РАЗРАБОТКИ ВЕБ-САЙТОВ BOOTSTRAP	112
Мысягин В.В.	ЦИФРОВЫЕ ДВОЙНИКИ И ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ В ПРОИЗВОДСТВЕ. ОТЛИЧИЯ И ПРАКТИКИ	117
Орлов Д.Ю.	ЦИФРОВАЯ СТЕГАНОГРАФИЯ И ЕЁ ПРИМЕНЕНИЕ	123
Полякова А.И., Дробахина А.Н.	ФОРМИРОВАНИЕ ЦИФРОВЫХ КОМПЕТЕНЦИЙ ПРЕПОДАВАТЕЛЕЙ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ В ПРОЦЕССЕ РАЗРАБОТКИ ЦИФРОВЫХ ПОРТФОЛИО	128
Пуличев М.П., Вайнер Л.Г.	СРАВНИТЕЛЬНЫЙ АНАЛИЗ СПОСОБОВ ПОСТРОЕНИЯ ЭВОЛВЕНТЫ ПРИ СОЗДАНИИ ГРАФИЧЕСКОЙ МОДЕЛИ ЗУБЧАТОГО КОЛЕСА	133
Смородинов А.Д., Рассадин А.А.	ПРИМЕНЕНИЕ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ЗАДАЧИ КЛАССИФИКАЦИИ УРАВНЕНИЙ В ОБЩЕМ ВИДЕ	138
Сорокин А.О., Слива М.В.	РАЗРАБОТКА ЛИЧНОГО КАБИНЕТА ПОЛЬЗОВАТЕЛЯ ИНТЕРНЕТ-ПРОВАЙДЕРА НА ПЛАТФОРМЕ ANDROID	144
Старовойтов А.А.	МОДЕЛИРОВАНИЕ СИСТЕМ ДЛЯ ПРОАКТИВНОГО УПРАВЛЕНИЯ ВЫЧИСЛИТЕЛЬНЫМИ КОМПЛЕКСАМИ	148
Суворин А.В.	РАСПОЗНАВАНИЕ ТЕКСТОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ	156
Тарнавская О.А.	РАЗРАБОТКА КОРПОРАТИВНОГО САЙТА	161
Томшин Н.А.	МАНИПУЛЯТОР ДЛЯ СБОРКИ КУБИКА РУБИКА	166

Топорков А.И. ПРИЛОЖЕНИЕ ДЛЯ ВИЗУАЛИЗАЦИИ И АНАЛИЗА РЕЗУЛЬТАТОВ МОДЕЛИРОВАНИЯ БИОМЕХАНИЧЕСКОГО УЗЛА. ПОДГОТОВКА И ОБРАБОТКА ДАННЫХ	173
Федулов С.И., Виштак О.В. ОБЗОР ТРЕНАЖЁРОВ ДЛЯ ФОРМИРОВАНИЯ «КЛАВИАТУРНОЙ ГРАМОТНОСТИ»	177
Харитонов Л.С., Устюжанин И.П., Бочкарева М.Д., Эварт Т.Е. РАЗРАБОТКА КРОССПЛАТФОРМЕННОГО ЭЛЕКТРОННОГО УЧЕБНОГО ПОСОБИЯ ПО ПРОГРАММИРОВАНИЮ НА ЯЗЫКЕ C++ С ПОМОЩЬЮ XAMARIN.FORMS/C#	184
Шахмирзоева М.И. ПРОЕКТНАЯ ДЕЯТЕЛЬНОСТЬ ПО ИНФОРМАТИКЕ КАК УСЛОВИЕ РАЗВИТИЯ КРЕАТИВНОГО МЫШЛЕНИЯ	190
Шкуратова Ю.А. КОМПЬЮТЕРНОЕ ТЕСТИРОВАНИЕ В СИСТЕМЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ СТУДЕНТОВ	195
ОРГКОМИТЕТ	202

Научное издание

**XXV Всероссийская студенческая
научно-практическая конференция
Нижевартовского государственного университета**

Часть 3

Информационные технологии

Нижевартовск, 4-5 апреля 2023 г.

ISBN 978-5-00047-681-9



Под общей редакцией: *Д.А. Погоньшева*
Редактор: *И.С. Анцева*
Технический редактор: *Д.В. Вилявин*
Обложка: *Д.В. Вилявин*

Подписано в печать 28.06.2023
Формат 60×84/8
Гарнитура Times New Roman. Усл. печ. листов 9,88
Электронное издание. Объем 13,88 МБ. Заказ 2281

Издательство НВГУ
628615, Тюменская область, г. Нижневартовск, ул. Маршала Жукова, 4
Тел./факс: (3466) 24-50-51, E-mail: red@nvsu.ru, izdatelstvo@nggu.ru