



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО-
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЮУрГГПУ»)

ФИЗИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ
КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
МЕТОДИКИ ОБУЧЕНИЯ ИНФОРМАТИКЕ

Разработка компьютерной модели адаптивной обучающей игры
Выпускная квалификационная работа
по направлению 09.03.02 Информационные системы и технологии
Направленность программы бакалавриата
«Информационные технологии в образовании»

Проверка на объем заимствований:
72 % авторского текста

Работа рекомендована к защите
рекомендована/не рекомендована

«14» мая 2020г.
зав. кафедрой И, ИТ и МОИ

Р Рузаков А.А.

Выполнила:

Студентка группы ОФ-413-095-4-1
Валиева Диана Маратовна Валиева

Научный руководитель:

Кандидат педагогических наук, доцент
кафедры И, ИТ и МОИ

Давыдова Давыдова Н.А.

Челябинск
2020



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО-
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»**
(ФГБОУ ВО «ЮУрГГПУ»)


ФИЗИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ
**КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
МЕТОДИКИ ОБУЧЕНИЯ ИНФОРМАТИКЕ**

Разработка компьютерной модели адаптивной обучающей игры
Выпускная квалификационная работа
по направлению 09.03.02 Информационные системы и технологии
Направленность программы бакалавриата
«Информационные технологии в образовании»

Проверка на объем заимствований:
72 % авторского текста


Работа рекомендована к защите
рекомендована/не рекомендована

« 14 » мая 2020 г.
зав. кафедрой И, ИТ и МОИ


_____ Рузаков А.А.

Выполнила:
Студентка группы ОФ-413-095-4-1
Валиева Диана Маратовна

Научный руководитель:
Кандидат педагогических наук, доцент
кафедры И, ИТ и МОИ


_____ Давыдова Н.А.

Челябинск
2020

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
ГЛАВА 1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ И НАЗНАЧЕНИЕ ПРОЕКТИРУЕМОЙ АДАПТИВНОЙ ОБУЧАЮЩЕЙ ИГРЫ	5
1.1 Характеристика учебного центра дополнительного образования «Вектор знаний»	5
1.2 Описание проектируемой игры	16
1.3 Сравнение аналогичных адаптивных обучающих игр.....	18
Выводы по Главе 1	21
ГЛАВА 2. ПРОЕКТИРОВАНИЕ КОМПЬЮТЕРНОЙ МОДЕЛИ АДАПТИВНОЙ ОБУЧАЮЩЕЙ ИГРЫ.....	22
2.1 Выбор программного обеспечения для реализации проекта	22
2.2 Выбор сервера, установка на хостинг.....	26
2.3 Описание реализации основных функций.....	27
2.4 Описание компьютерной модели обучающей игры.....	36
Выводы по Главе 2	40
ГЛАВА 3. ДОКУМЕНТАЦИЯ ГОТОВОЙ ИГРЫ	41
3.1 Испытание компьютерной модели адаптивной обучающей игры	41
3.2 Руководство пользователя (учащегося) компьютерной модели адаптивной обучающей игры.....	42
3.3 Техничко-экономическое обоснование разработки компьютерной модели адаптивной обучающей игры	50
Выводы по Главе 3	53
ЗАКЛЮЧЕНИЕ	54
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	55

ВВЕДЕНИЕ

Применение приобретенных знаний на практике и способность быстро находить решения некоторых задач – очень важный этап обучения учащихся в любой сфере. Для того чтобы ученик не уставал от постоянного выполнения типичных задач, в последние несколько лет активно внедряются игровые технологии в сферу образования.

Своей нестандартностью, легкостью такие игры способны привлекать внимание школьников, показать, что обучение может быть занимательным, интересным. Ученик больше времени проводит, обучаясь по таким компьютерным моделям, а также меньше устает от выполнения однотипных заданий, улучшая свои знания и практические навыки [2].

Адаптивная обучающая игра – игра, включающая в себя комплекс необходимого теоретического материала и тестовых заданий, которая адаптируется под уровень знания учащегося по данной теме.

Создание компьютерной модели адаптивной обучающей игры позволит значительно улучшить качество освоения материала. Игра предоставляет возможность обеспечить учащегося интеллектуальной помощью на каждом этапе решения задачи и упростить работу преподавателя.

Актуальность темы обусловлена возможностью организации самостоятельного обучения и повтора материала, выполняемой учениками.

Цель работы: проектирование адаптивной обучающей игры.

Задачи работы:

1. Изучение деятельности учебного центра дополнительного образования «Вектор знаний».
2. Знакомство с техническим оборудованием и установленным программным продуктам учебного центра дополнительного образования «Вектор знаний».

3. Изучение основных бизнес-процессов предприятия.
4. Описание технико-экономического задания.
5. Анализ существующих аналогов разрабатываемой адаптивной обучающей игры.
6. Разработка удобного и интуитивно понятного интерфейса игры.
7. Проектирование и реализация компьютерной модели адаптивной обучающей игры.
8. Создание руководства пользователей адаптивной обучающей игры.
9. Внедрение разработанной компьютерной модели адаптивной обучающей игры.

Практическая значимость: разработанное приложение, обладающее адаптивными элементами, позволяет пользователям в игровой форме проверить степень усвоения учебного материала.

ГЛАВА 1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ И НАЗНАЧЕНИЕ ПРОЕКТИРУЕМОЙ АДАПТИВНОЙ ОБУЧАЮЩЕЙ ИГРЫ

1.1 Характеристика учебного центра дополнительного образования «Вектор знаний»

1.1.1 Описание учебного центра дополнительного образования «Вектор знаний»

Дата основания учебного центра дополнительного образования: 12 октября 2017 года. «Вектор знаний» лицензированный центр дополнительного образования для детей и взрослых расположен по адресу: г. Челябинск, ул. Цвиллинга, д.58-В.

Согласно лицензии, индивидуальный предприниматель имеет право оказывать образовательные услуги по реализации образовательных программ по видам образования, по уровням образования, по профессиям, специальностям, направлениям подготовки (для профессионального образования), по подвидам дополнительного образования.

В учебном центре работают преподаватели, которые осуществляют помощь в изучении школьных предметов – математика, физика, информатика, русский язык и др., олимпиадный уровень, подготовка к ЕГЭ и ОГЭ. Для творческих личностей – класс электронной музыки. Для самых маленьких работает детская школа «Знайка» (1-4 классы).

Учредитель образовательной организации: Министерство образования и науки Челябинской области.

Официальный сайт: <http://www.minobr74.ru/>.

Телефоны: +7 (919) 123-06-19; (351)223-06-19.

График работы: ежедневно с 09:00 до 19:00.

E-mail: info@vec74.ru.

Официальный сайт учебного центра: <https://www.vec74.ru/index.php>.

Группа в «ВКонтакте»: <https://vk.com/vek74>.

1.1.2 Организационная структура учебного центра дополнительного образования «Вектор знаний»

Управление учебным центром дополнительного образования осуществляется в соответствии с законодательством Российской Федерации и Челябинской области.

В учебном центре действуют следующие органы управления: административный и управленческий персонал, основной преподавательский состав, другие категории работников, а также внештатные работники. Ссылка на сайт учебного центра: <https://www.vec74.ru/>.

Педагогический совет – высший орган педагогического самоуправления, членами которого являются все учителя учебного центра, а председателем – директор. На своих заседаниях педсовет рассматривает сложные педагогические и методические вопросы, проблемы организации учебно-воспитательного процесса.

К административно-управленческому персоналу относятся: директор (руководитель), руководители структурных подразделений, главный бухгалтер.

Бухгалтерскую службу организации возглавляет главный бухгалтер, назначаемый на должность и освобождаемый от должности руководителем учебного центра. Основной задачей главного бухгалтера является организация управления функционированием и развитием системы бухгалтерского учета в структуре управления учебного центра. Главный бухгалтер подчиняется непосредственно директору(руководителю) учебного центра и несет ответственность за формирование учетной политики, учет, своевременное представление полной и достоверной финансовой отчетности.

К основному педагогическому персоналу относятся следующие сотрудники: преподаватель; репетитор; музыкальный руководитель;

учитель-организатор; социальный педагог; тренер-преподаватель (в том числе старший).

Педагогический совет – высший орган педагогического самоуправления, членами которого являются все преподаватели учебного центра, а председателем – директор. На своих заседаниях педагогический совет рассматривает сложные педагогические и методические вопросы, проблемы организации учебного и воспитательного процесса.

К учебно-вспомогательному персоналу относятся: секретарь учебной части, документовед, делопроизводитель, заведующий хозяйством, системный администратор и др.

К иному персоналу относятся: рабочий по комплексному обслуживанию здания, курьер, водитель и др. Иные категории работников устанавливаются также в соответствии с региональными нормативными актами.

Внештатный работник – работник, который оказывает образовательной организации какие-либо услуги (выполняет работы) в рамках гражданско-правового договора, договора оказания услуг (выполнения работ). Данный работник не учитывается в штате организации, поэтому в диаграммах подраздела сайта «Работники образовательной организации» информация о нём не учитывается, но при этом всю необходимую информацию можно разместить в его личной карточке.

Общая организационная структура учебного заведения, которая должна обеспечить эффективное взаимодействие всех участников образовательного процесса представлена на рисунке 1.

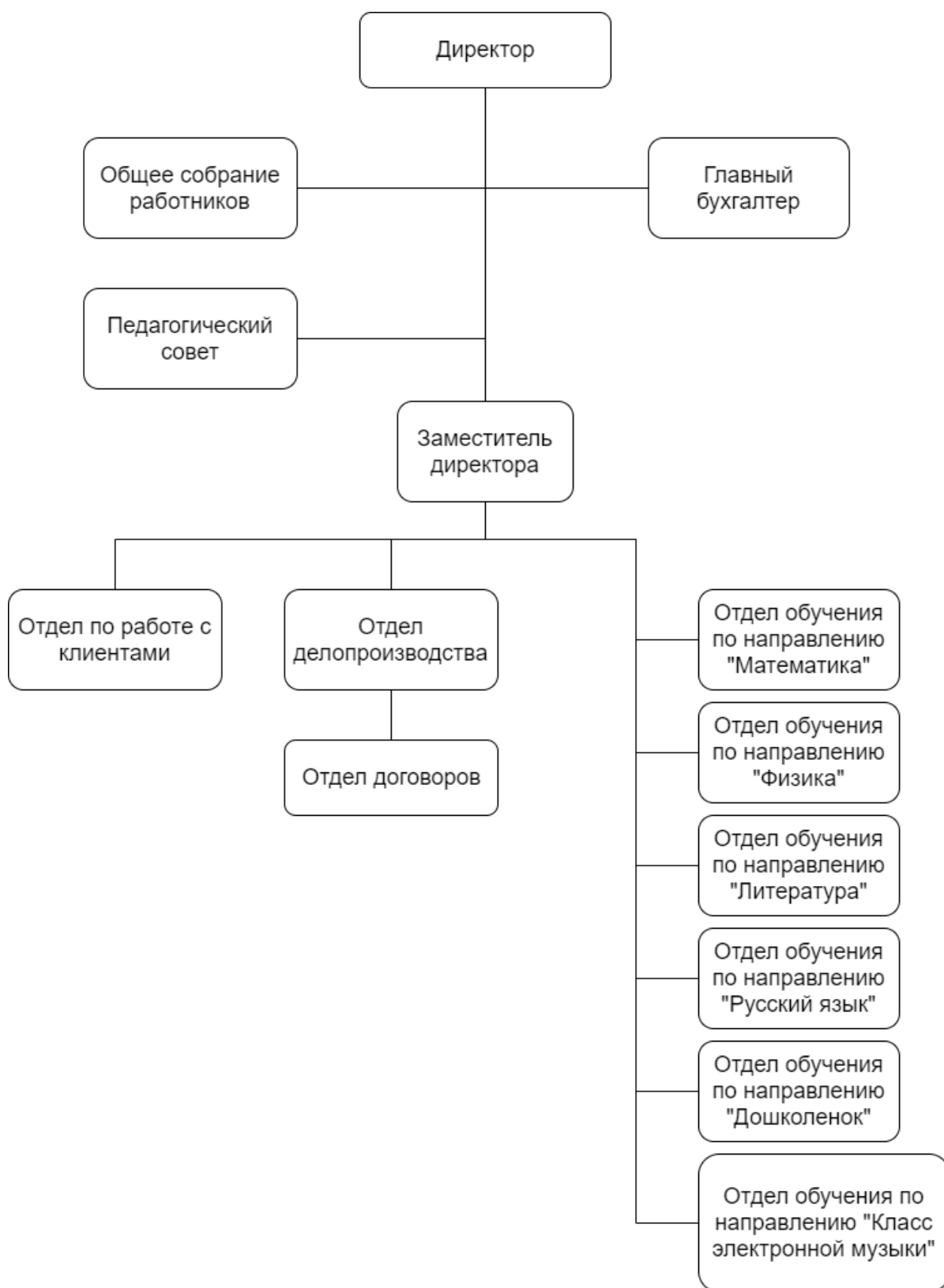


Рисунок 1 – Организационная структура учебного центра

Предметы, преподаваемые в учебном центре:

Дошколенок (подготовка детей к школе):

- обучение чтению /изучение букв, овладение способом чтения по слогам, целым словом, при возможности – бегло;
- развитие мелкой моторики руки;
- формирование графического навыка;
- овладение способом каллиграфического написания букв, цифр;
- знакомство с геометрическими формами;
- решение арифметических, геометрических и логических задач;
- формирование коммуникативных и регулятивных действий, необходимых в школе.

Математика:

- подготовка к ЕГЭ (10-11 классы);
- подготовка к ОГЭ (9 классы);
- углубленный курс по математике (5-8 классы);
- решение Вузовских олимпиадных задач по математике (10-11 классы);
- олимпиадные и развивающие задачи по математике.
- детская школа «Знайка» (1-4 классы).

Информатика и программирование:

- помощь в изучении школьного курса информатики, в том числе подготовка к ЕГЭ (10-11 классы);
- помощь в изучении школьного курса информатики, в том числе подготовка к ОГЭ (8-9 классы);
- пользовательский курс работы на компьютере, основы алгоритмизации, программирования, логики (5-7 классы);
- изучение языков программирования C++ и Python;
- изучение 3D-моделирования с помощью профессионального ПО для создания трехмерной компьютерной графики Blender;

– детская школа «Знайка» (1-4 классы): основы работы на компьютере, в интересной и доступной форме начала алгоритмизации и программирования, развитие логического мышления.

Физика:

– подготовка к ЕГЭ (10-11 классы);
– подготовка к ОГЭ (9 классы);
– углубленный курс (7-11 классы);
– экспериментальная физика (5-6 классы);
– детская школа «Знайка» (1-4 классы): основы физических явлений, проведение экспериментов и лабораторных работ, обучение правильному оформлению.

Литература:

– подготовка к ЕГЭ (10-11 классы).

Русский язык:

– подготовка к ЕГЭ (10-11 классы);
– подготовка к ОГЭ (9 классы);
– углубленное изучение (5-8 классы).

Класс электронной музыки:

– основы музыкальной грамоты – гармонии, сольфеджио, композиции, аранжировки (симбиоз всех этих дисциплин, специально предназначенный для начинающего заниматься электронной музыки «с нуля»);

– создание фонограмм – знакомство с секвенсором, введение в практическую аранжировку; компьютер как музыкальная студия. Курс содержит несколько разделов – по числу программ, в которых идет работа. Запись, сведение, цифровая обработка, добавление эффектов, добавление голоса, перевод звукового файла в разные форматы и т.п.

1.1.3 Описание основных бизнес-процессов учебного центра дополнительного образования

При помощи функциональной модели IDEF0 можно отразить основной бизнес-процесс деятельности учебного центра дополнительного образования (рисунок 2). Основным концептуальный принцип методологии IDEF – представление любой изучаемой системы в виде набора взаимодействующих и взаимосвязанных блоков, отображающих процессы, операции, действия, происходящие в изучаемой системе. Для начала рассматривается система в целом и ее взаимодействие с внешней средой с помощью контекстной диаграммы.

Основной работой (функцией) бизнес-процесса является деятельность учебного центра дополнительного образования. Работы (функциональные блоки) на диаграммах обозначены прямоугольниками. Каждая сторона блока имеет определенное назначение. Левая сторона блока предназначена для входов, верхняя – для управления, правая – для выходов, нижняя – для механизмов. Такое обозначение отражает определенные системные принципы: входы преобразуются в выходы, управление ограничивает или предписывает условия выполнения преобразований, механизмы показывают, что и как выполняет функция.

Взаимодействие работ с внешним миром и между собой описывается в виде стрелок, изображаемых одинарными линиями со стрелками на концах. Стрелки представляют собой некую информацию. Входящие стрелки – объекты, используемые и преобразуемые работой для получения результата (выхода) [6]. В данном бизнес-процессе – это учащиеся и данные об учащих. Исходящие стрелки выводят результат деятельности, здесь это учащиеся, освоившие образовательную программу, и отчетная документация. Управляющие стрелки являются управляющей информацией (документы, положения и руководства к системе). Механизмы – это ресурсы, выполняющие работу.

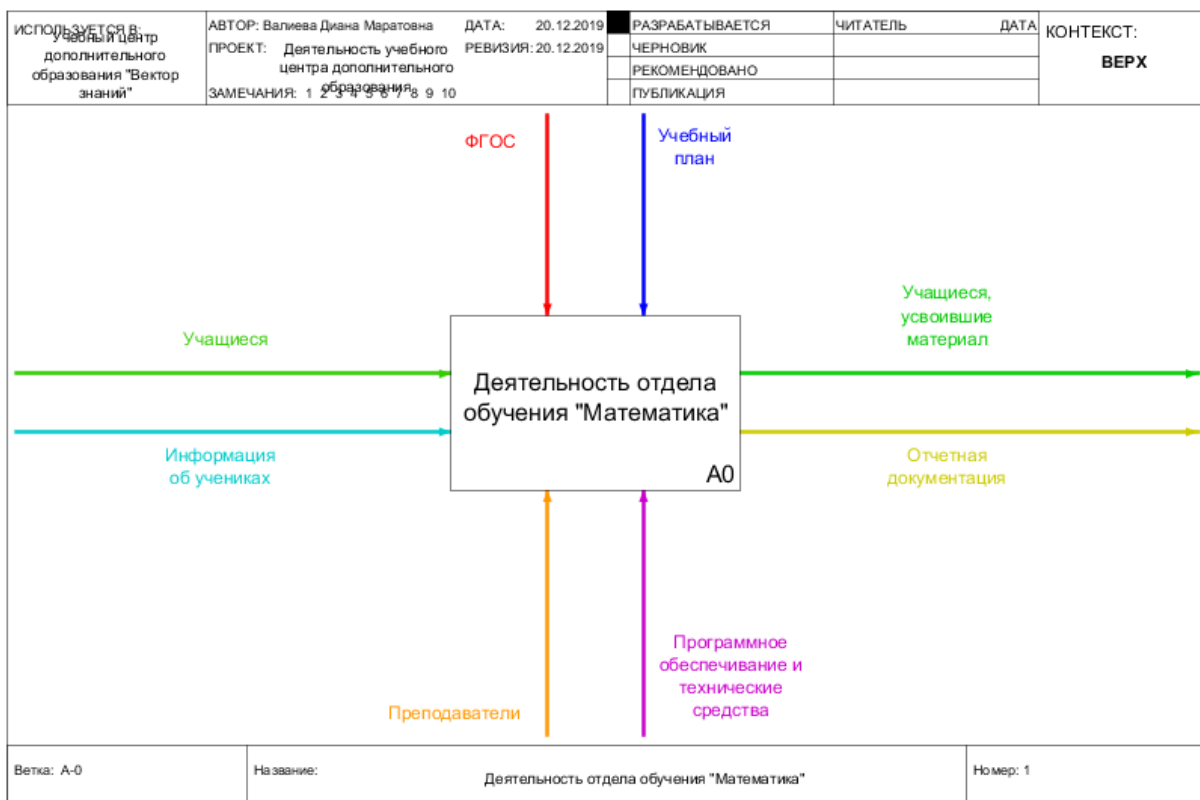


Рисунок 2 – Контекстная диаграмма организации деятельности отдела обучения по направлению «Математика»

Структурное подразделение, для которого была написана адаптивная обучающая игра – это отдел обучения по направлению «Математика».

Для представления бизнес-процессов отдела обучения по направлению «Математика» была разработана детализированная диаграмма, отражающая все микропроцессы, происходящие в рамках этой деятельности (рисунок 3).

Деятельность отдела обучения по направлению «Математика» включает в себя обеспечение подготовки к ЕГЭ, обеспечение подготовки к ОГЭ, подготовку к олимпиадам и обучение углубленному курсу.

Составляющие обучения углубленного курса наглядно отображены на рисунке 4.

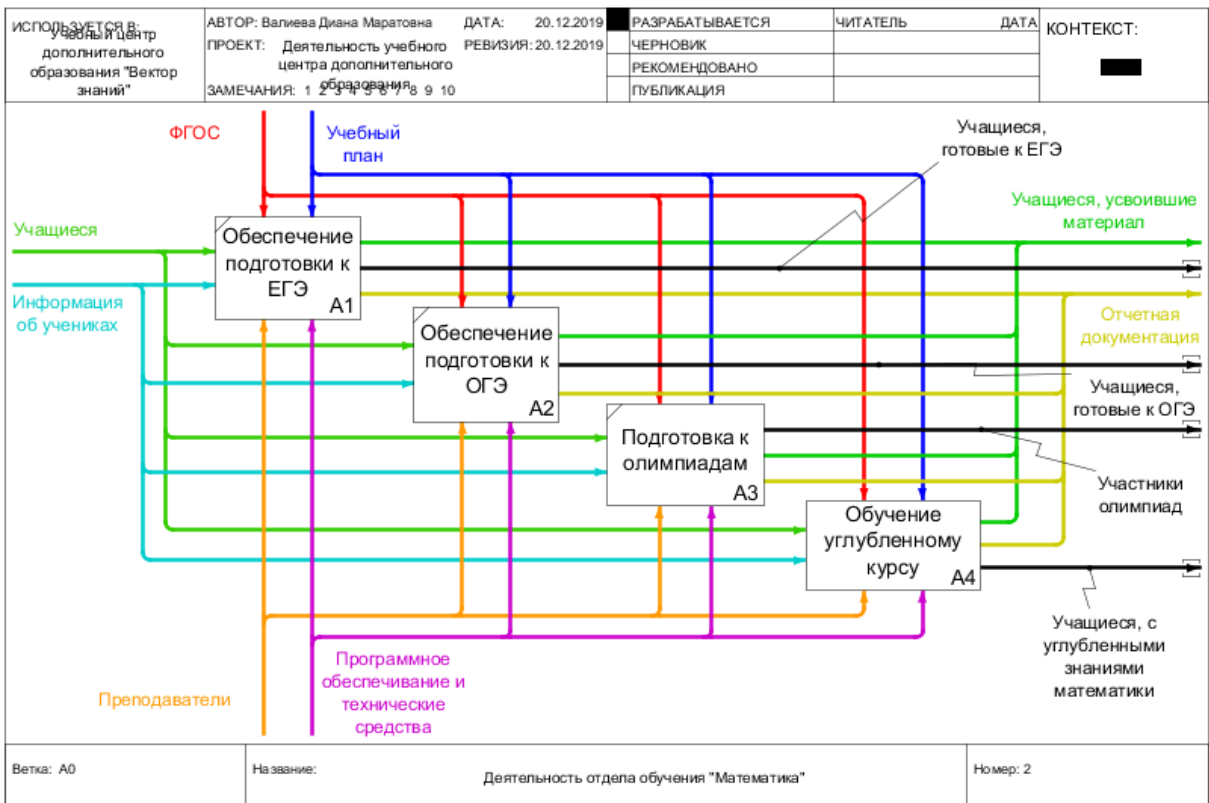


Рисунок 3 – Декомпозиция диаграммы «Деятельность отдела обучения «Математика»

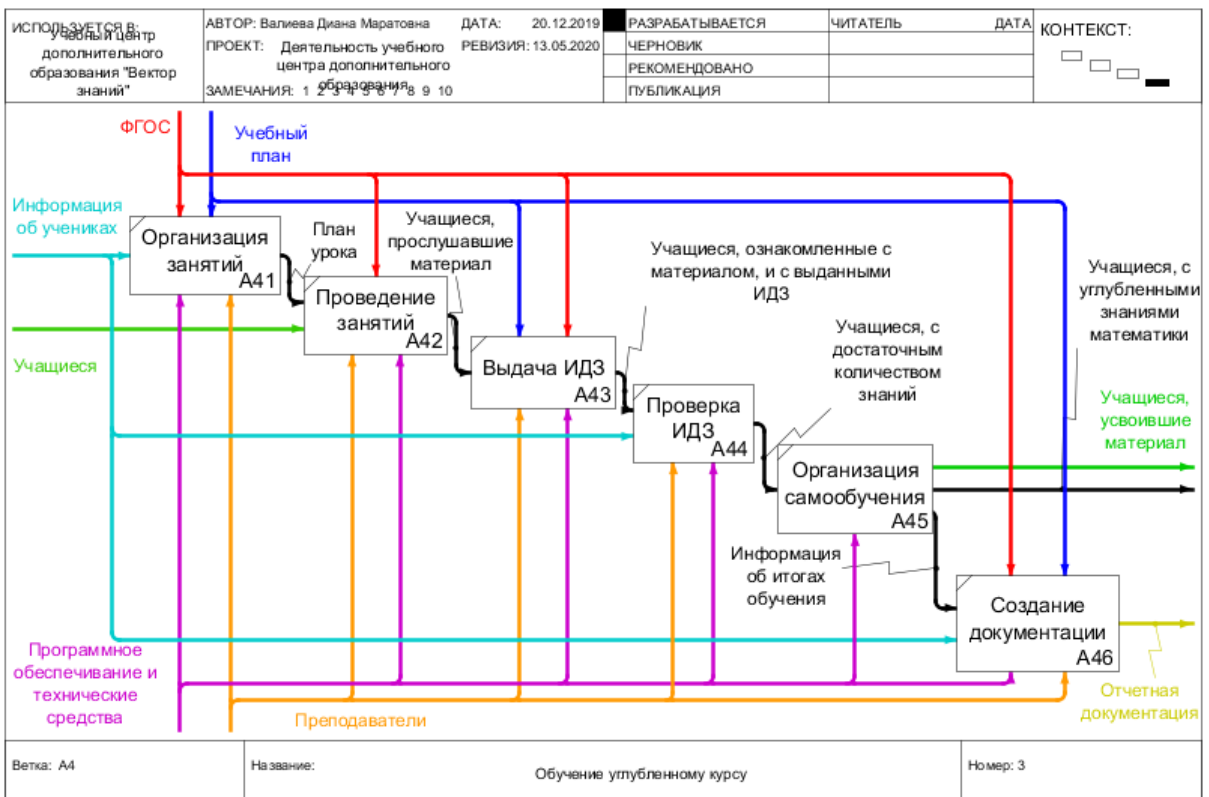


Рисунок 4 – Декомпозиция обучения углубленному курсу
Местом решаемой задачи является организация самообучения.

1.1.4 Описание средств автоматизации учебного центра дополнительного образования

В учебном центре дополнительного образования организовано три специальных учебных аудиторий, каждая из которых оснащена 11 компьютерами. Специальные учебные аудитории имеют следующие возможности:

- рабочие моноблоки, через которую учащиеся используют сеть интернет, а также внутреннюю сеть организации;
- сетевое соединение со скоростью 1 Гбит/с;
- сервер, который управляет внутренней сетью организации.

Основные характеристики моноблоков в компьютерной аудитории:

- модель моноблока: Lenovo IdeaCentre A700;
- особенности: встроенный тв-тюнер, HDMI-вход, пульт ДУ;
- диагональ экрана: 23";
- габариты (ВхШхГ): 568x430x71 мм;
- вес: 14,8 кг.

Физическая структура сети – это схема связей физических элементов сети: узлов коммутации (УК), конечных пунктов (ОП) – станций и линий передачи в их взаимном расположении с характеристиками передачи и распределения сообщений.

Обобщенная схема локальной сети представлена на рисунке 5.

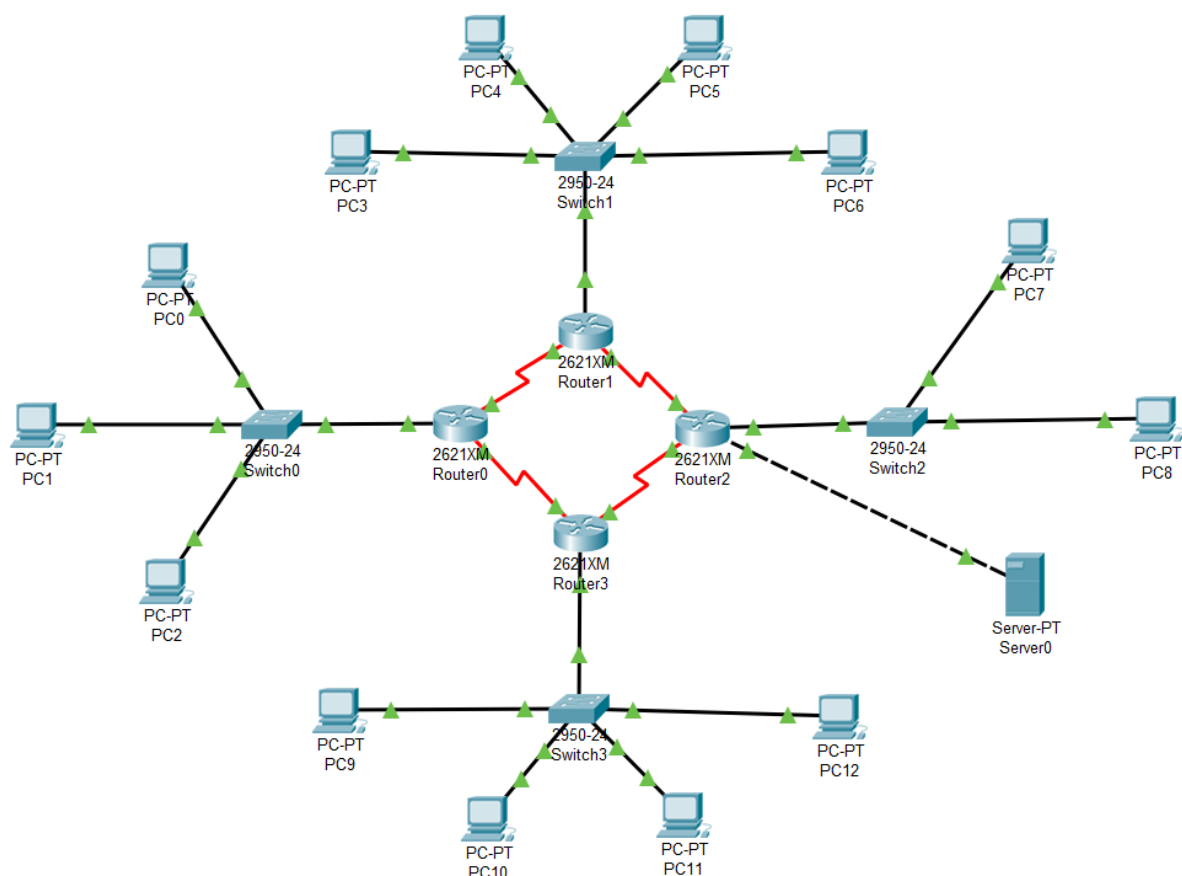


Рисунок 5 – Схема локальной сети учебного центра дополнительного образования

В учебном центре дополнительного образования имеется собственный веб-сайт, облачное хранилище данных и электронная почта.

Перечень лицензионного программного обеспечения:

- операционная система Windows 10 Enterprise 2016 LTSC;
- Microsoft Office Professional Plus 2016;
- антивирусное программное обеспечение Kaspersky Endpoint Security для бизнеса – Стандартный Russian Edition;
- WinRar или эквивалент по функциональным и техническим характеристикам.

Другие программные обеспечения, используемые в учебном центре дополнительного образования:

- компьютерная графика и дизайн – CorelDRAW Graphics Suite X3 Russian, CorelDRAW Graphics Suite X3 Russian, CorelDRAW Клипарты, CorelDRAW Фото и шрифты, Paint Net;

- вспомогательное программное обеспечение, сопровождающее образовательный процесс;
- комплекты электронных наглядных пособий и обучающих программ;
- программирование – 1С:Предприятие 8. Версия для обучения программированию, Borland Developer Studio 2006 Professional, Borland Pascal 7.0, Borland Turbo Pascal 7.0, Microsoft Visual Studio Pro 2005 Eng.

1.2 Описание проектируемой игры

1.2.1 Назначение и цели создания игры

Компьютерная модель адаптивной обучающей игры предназначена для учащихся центра дополнительного образования «Вектор знаний» для повторения и закрепления знаний и практических навыков.

Сюжет игры разворачивается в двух пространственных измерениях: объективной действительности и параллельной ей геометрической реальности, существующей в сновидении главного героя, связанных посредством объектов, находящихся одновременно в обоих пространствах.

Целями адаптивной обучающей игры являются повторение и закрепление знаний и практических навыков учащихся посредством:

1. Автоматического выявления наиболее трудных тем и задач для конкретного ученика.
 2. Сохранения результатов прохождения игры.
 3. Всплывающих подсказок в каждом уровне игры.
 4. Создания многоуровневой системы для разрабатываемой игры.
- (Для перехода на следующий уровень необходимо успешно пройти текущий).

1.2.2 Требования к структуре и функционированию игры

Разрабатываемая компьютерная модель адаптивной обучающей игры должна состоять из следующих подсистем, разделенных по функциональному признаку [4]:

1. Подсистема хранения информации (База данных с информацией о пользователях и доступных им уровнях).
2. Подсистема работы клиента с базой данных (Клиентская часть для решения задач, просмотра результатов прохождения и теоретического материала).
3. Подсистема авторизации (Предназначена для автоматизации доступа для преподавателя и учащихся по логину и паролю).
4. Адаптационный блок (Включает основной функционал игры: проверка входных знаний; проверка правильно выполненных заданий; вывод страницы с ошибкой; предоставления подсказки к текущему заданию)

Схематичное отображение подсистем и связей между ними отображено на рисунке 6.

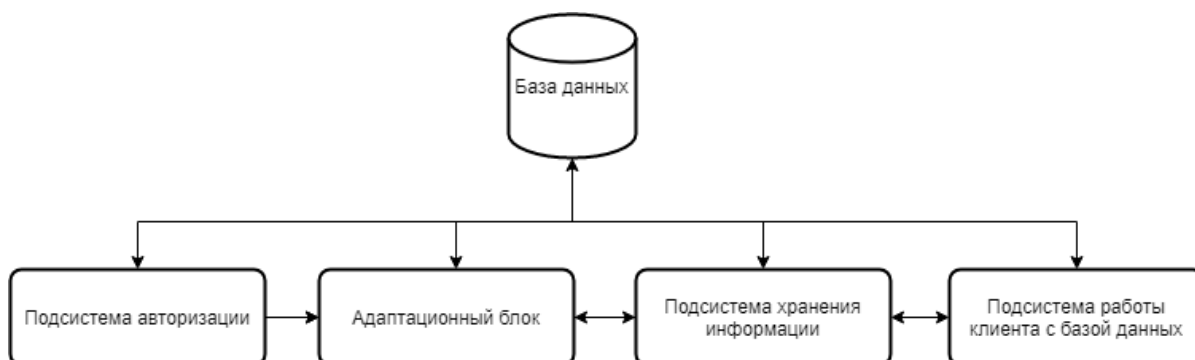


Рисунок 6 – Схема взаимодействия подсистем компьютерной модели адаптивной обучающей игры

1.3 Сравнение аналогичных адаптивных обучающих игр

1.3.1 Duolingo

Duolingo – обучающая система иностранным языкам (рисунок 7). Ссылка на веб-страницу: <https://www.duolingo.com/learn>. Приложение основано на методологии, помогающей удерживать пользователей надолго, а также на учебной программе, соответствующей международным стандартам.

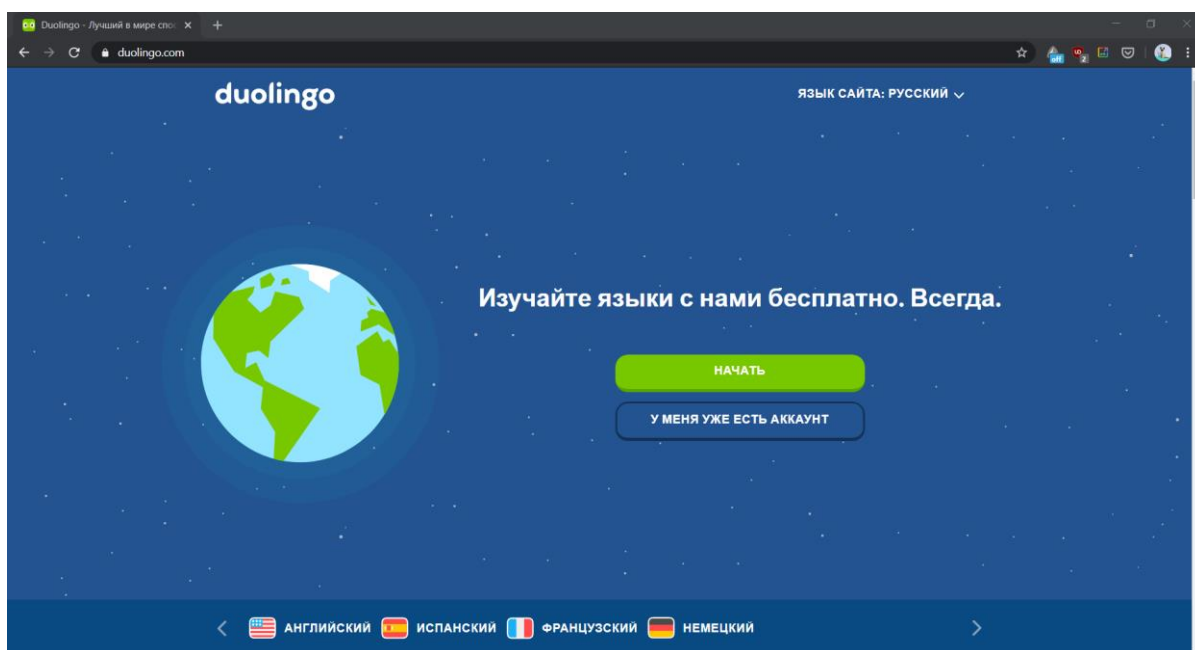


Рисунок 7 – Главная страница обучающей системы иностранным языкам Duolingo

Система представляет возможность обучения по четырем языкам:

- английский;
- немецкий;
- французский;
- испанский.

Достоинства системы:

- система адаптирует входной тест к способностям обучающегося;
- система обнаруживает опечатки в написанном тексте;

- осуществляет обзор урока;
- показывает правильное решение.

Недостатки системы:

- для скрытия рекламы необходимо оформить платную подписку;
- большое количество повторяющихся заданий;
- требуется постоянное подключение к сети интернет.

1.3.2 Проект самоподготовки к ЕГЭ Examer

Examer – это онлайн-курс для самостоятельной подготовки к ЕГЭ и ОГЭ. Главная страница онлайн-курса представлена на рисунке 8.

Система составляет индивидуальный учебный план для пользователя, позволяющий набрать желаемое количество баллов на едином государственном экзамене. На выбор предоставляются все основные предметы для сдачи ЕГЭ. Ссылка на сайт: <https://examer.ru/>.

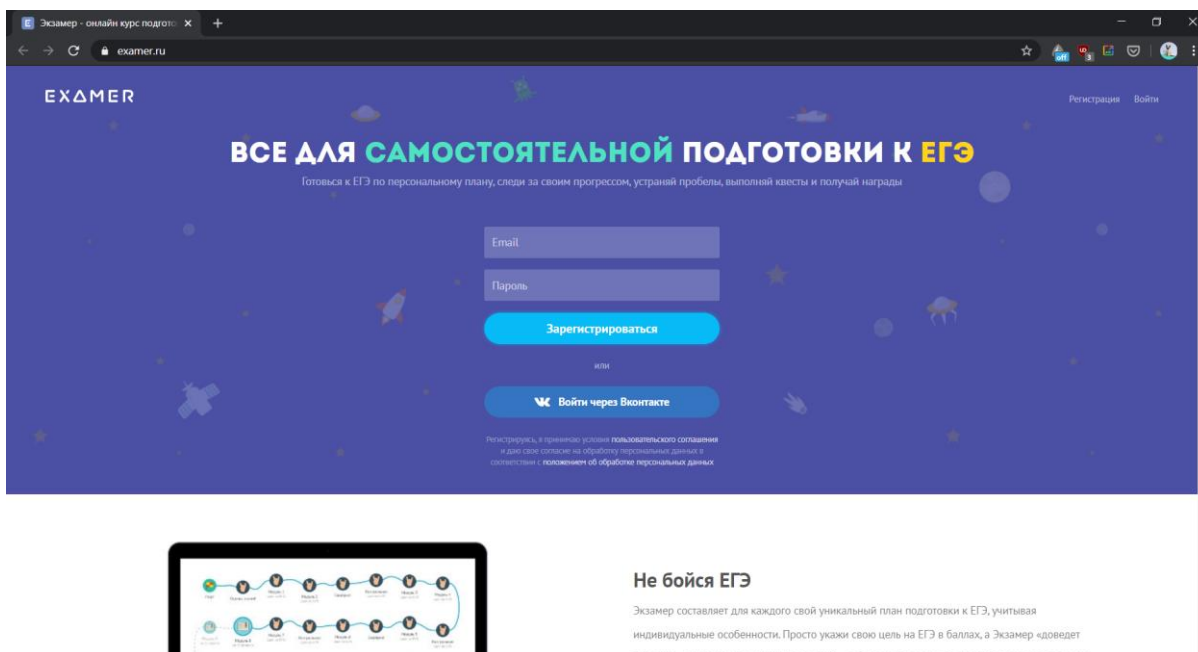


Рисунок 8 – Главная страница онлайн-курса подготовки к ОГЭ и ЕГЭ Examer

Достоинства системы:

- подбирает задачи нужной сложности в режиме «Учитель»;
- выводит решение и ответ;

- строит план подготовки к экзамену в режиме «Ученик»;
- проверяет начальный уровень подготовки обучающегося.

Недостатки:

- требуется постоянное подключение к сети интернет;
- бесплатная версия сильно ограничена. Доступны: пять вариантов задач для каждой темы предмета, два варианта ЕГЭ или ОГЭ определенного предмета, большая часть теории по предметам.

1.3.3 Проект для детей и подростков Logiclike

Используются элементы адаптивных технологий. Содержит программы для развития логического мышления Ссылка на веб-сайт: <https://logiclike.com/>.

Главная страница онлайн-ресурса представлена на рисунке 9.

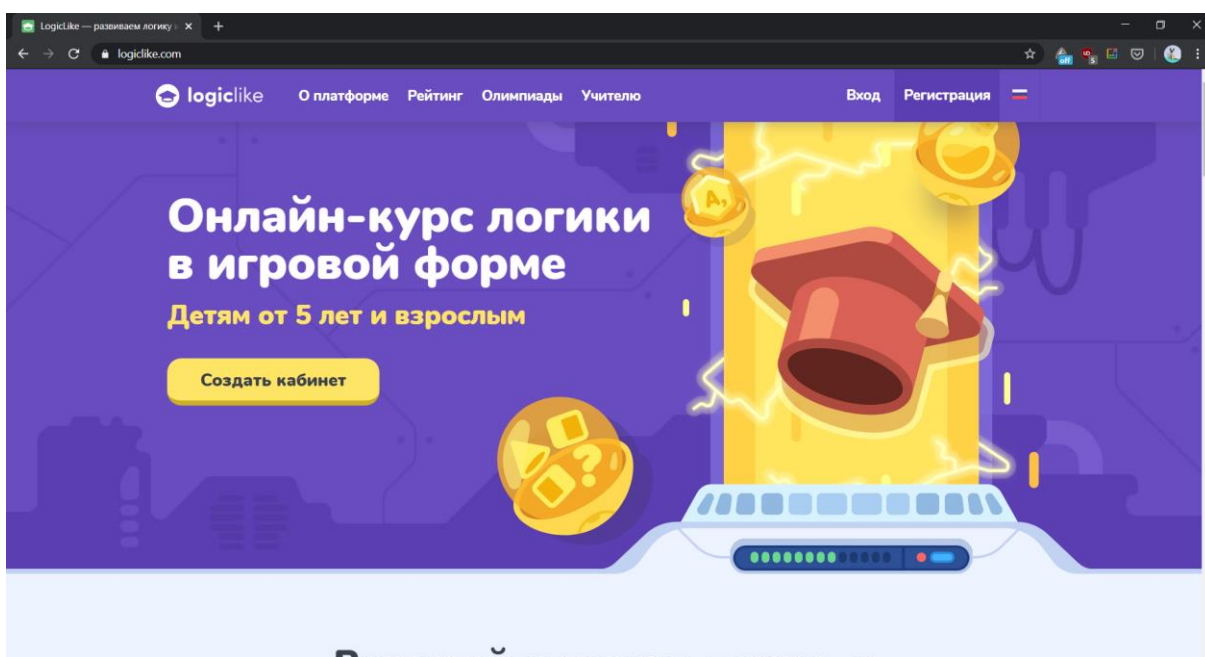


Рисунок 9 – Главная страница обучающей системы Logiclike

Достоинства системы:

- предусмотрены поветы, подсказки, возвраты в разделы;
- имеется теоретическая часть;
- большая база задач, ребусов и головоломок;

- учет наград и достижений;
- имеется комплексное обучение, также различные тематические блоки;
- три уровня сложности.

Недостатки:

- уровень «Эксперт» закрыт в бесплатной версии;
- бесплатный период использования – 48 часов или 1000 баллов;
- требуется постоянное подключение к интернету.

Выводы по Главе 1

В первой главе квалификационной работы была рассмотрена организационная структура учебного центра дополнительного образования «Вектор знаний».

Описан основной бизнес-процесс организации заказчика, проведено описание проектируемой компьютерной модели, выявлена ее цель и место в учебном процессе, сформированы требования к игре.

Выполнен сравнительный анализ схожих программных продуктов: «Logiclike», «Examer» и «Duolingo». Это позволило выявить наиболее значимые функции таких систем.

Основными пользователями данной игры будут являться ученики учебного центра. Проектируемая компьютерная модель позволит учащимся повторить и закрепить полученные знания и практические навыки.

ГЛАВА 2. ПРОЕКТИРОВАНИЕ КОМПЬЮТЕРНОЙ МОДЕЛИ АДАПТИВНОЙ ОБУЧАЮЩЕЙ ИГРЫ

2.1 Выбор программного обеспечения для реализации проекта

Среди многочисленного ПО нужно выбрать наиболее подходящие для разработки проекта. В приоритете программы со свободной лицензией, широким функционалом, удобством интерфейса и возможностью экспортировать файлы в нужное расширение.

2.1.1 Платформа для разработки игры

Платформа для разработки игр – это центральный программный компонент компьютерных и видеоигр или других интерактивных приложений с графикой, обрабатываемой в реальном времени. Он обеспечивает основные технологии, упрощает разработку.

Для сравнения были выбраны две популярные платформы для профессиональной разработки игр (таблица 1).

Таблица 1 – Сравнительная характеристика некоторых платформ

Критерий	Unity	Unreal Engine
<i>1</i>	<i>2</i>	<i>3</i>
Краткое описание	Межплатформенная среда разработки 2D и 3D компьютерных игр, которая позволяет создавать приложения, работающие под более чем 20 различными операционными системами [8]	Набор инструментов для разработки игр, имеющий широкие возможности: от создания двухмерных игр на мобильные до AAA-проектов для консолей
Стоимость	Бесплатна для персонального пользования (при выручке за игру до 100 000\$ в год). Для коммерческого использования при прибыли от игры до 200 000\$ в год оплачивается подписка 35\$ в месяц, свыше этого подписка составляет 125\$, повышение стоимости подписки открывает дополнительные возможности платформы	Unreal Engine 4 распространяется бесплатно. Однако до тех пор, пока вы не выпустите свой первый коммерческий продукт на основе UE4, далее следует платить процент от продаж Вашей игры

Продолжение таблицы 1

1	2	3
На какие платформы разрабатывается	Ios, android, windows, Windows phone, Mac, linux, webgl, ps4, psvite, xbox one, Wii U, Nintendo 3Ds, Oculus Rift, Google Cardboard Android & ios, steam, Playstation VR, Gear VR, Windows mixed reality, Daydream, Android TV, Samsung smart TV, tvos, Nintendo Switch, fireos, Facebook Gameroom, Apple arkit, Google arcore, Vuforia [12]	Windows PC, Mac, Linux, ios и Android, HTML5. Также есть встроенная поддержка Виртуальной реальности для Oculus Rift. Помимо этого, UE4 поддерживает Xbox One и playstation 4 (включая Project Morpheus)
Какого вида игры можно создавать	Игры любого жанра, симуляторы и др.	Игры (2D-3D; RTS, Action-RPG, Shooter, Racing, ММО-игры и любой другой жанр и направление), симуляторы и даже программное обеспечение. Можно использовать UE4 для архитектурной визуализации и многое другое
Языки программирования	C#, javascript, Boo (был убран из новой версии платформы)	C++ и Blueprint (собственный визуальный язык программирования)
Интерфейс	Имеет простой Drag&Drop интерфейс	Нельзя перетаскивать объекты прямо в окно редактора.
Открытость кода	Закрит	Благодаря открытому исходному коду, можно самостоятельно добавить поддержку дополнительных устройств, или же оптимизировать и дополнить уже существующие.
Обучающие ресурсы	Присутствуют на официальном сайте	Присутствуют на официальном сайте
Документация	Документация понятна, но переведена на русский язык только частично	Исчерпывающая, но только на английском языке
Плагины	Можно писать в самой платформе, так же можно выбрать из множества в магазине	Пишутся только на C++, маленькое количество доступных плагинов
Графика	Реалистичная хорошая графика	В новых версиях платформы графика тоже на высоком уровне, однако еще не дотягивает до UE

Продолжение таблицы 1

<i>1</i>	<i>2</i>	<i>3</i>
Общее впечатление	Профессиональная среда разработки игр, с удобным приятным интерфейсом	Профессиональная среда разработки игр, имеющая хорошую графику, но сложный и запутанный интерфейс

После проведенного исследования была выбрана платформа Unity 2019 для WebGL.

2.1.2 Среда разработки и язык программирования

Для написания кода игры была выбрана среда разработки Visual Studio 2019, и поддерживаемый в Unity язык программирования C#.

Visual Studio линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Silverlight [5].

Visual Studio включает в себя редактор исходного кода с поддержкой технологии IntelliSense и возможностью простейшего рефакторинга кода. Встроенный отладчик может работать как отладчик уровня исходного кода, так и отладчик машинного уровня. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения, веб-редактор, дизайнер классов и дизайнер схемы базы данных. Visual Studio позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (как, например, Subversion и Visual SourceSafe), добавление

новых наборов инструментов (например, для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования) или инструментов для прочих аспектов процесса разработки программного обеспечения (например, клиент Team Explorer для работы с Team Foundation Server) [1].

2.1.3 Графические редакторы

В данном проекте использовались следующие графические редакторы:

1. Adobe Photoshop – растровый графический редактор с мощным инструментарием. Использовался для рисования скетчей, фонов и других элементов окружения.

2. Adobe Illustrator – векторный графический редактор. Использовался для отрисовки элементов интерфейса, иллюстраций к заданиям и др.

2.1.4 Программное обеспечение для анимации

Для создания анимации движений персонажа и монстров в игре использовалась программа Dragon Bones. Dragon Bones – это бесплатная программа для создания скелетной анимации

2.1.5 Другие программы и плагины

Во время проекта использовался плагин Sprite Packer для Unity. Это бесплатный плагин с официального магазина Unity, который позволяет сохранить отдельные картинки с покадровой анимацией в единый лист спрайтов, что облегчает работу с покадровой анимацией на платформе Unity 2019 и увеличивает производительность игры.

2.2 Выбор сервера, установка на хостинг

Игра расположена на хостинге Fornex.com. Хостинг-провайдер одновременно является регистратором доменов. Отличается предельной скоростью работы, хорошим дизайном, высоким уровнем безопасности.

Выбран тариф у хостинг-провайдера «Economy». Заключён и оплачен на один год договор аренды виртуального хостинга. Получен выделенный ip-адрес хостинга (рисунок 10).

The screenshot displays the 'Тарифы' (Rates) section of the Fornex.com website. It features four vertical cards representing different hosting plans. At the top right, there is a currency selector set to 'Евро'. Each card includes the plan name, storage capacity, number of sites, bandwidth multiplier, SSL status, price per month, and a 'Заказать' (Order) button. A 'Попробовать бесплатно' (Try for free) link is also present at the bottom of each card.

Plan Name	Storage	Sites	Bandwidth	SSL	Price / month	Additional Info
ECONOMY	1ГБ	1 сайт	x1	SSL бесплатно	от 0.80 €	+ ДОМЕН .RU в ПОДАРОК при оплате за год
START	10ГБ	Полный безлимит	x2	SSL бесплатно	от 2.40 €	+ ДОМЕН .RU в ПОДАРОК при оплате за год
STANDARD	20ГБ	Полный безлимит	x4	SSL бесплатно	от 4 €	+ ДОМЕН .RU в ПОДАРОК при оплате за год
PRO	30ГБ	Полный безлимит	x8	SSL бесплатно	от 6.40 €	+ ДОМЕН .RU в ПОДАРОК при оплате за год

Рисунок 10 – Тарифы хостинга Fornex.com

Зарегистрировано доменное имя у регистратора хостинга. В личном кабинете создан домен второго уровня geometrylittledi.ru.

На хостинг с локального компьютера перенесены все файлы игры посредством ftp-клиента в директорию docs – домашнюю директорию сайта. На локальном компьютере через интерфейс RНРMyAdmin создана база данных. Через RНРMyAdmin на хостинге импортирована созданная копия базы данных. В конфигурационном файле актуализированы имя базы данных, имя пользователя и пароль, адрес сервера MySQL, так как хостинг-провайдер предлагает их другие. После проведённых операций сайт игры доступен по адресу: <https://geometrylittledi.ru/>.

2.3 Описание реализации основных функций

Компьютерная модель адаптивной обучающей игры «Геометрия 5+» представляет собой веб-игру с подключением в асинхронном режиме к серверу, где расположена база данных, при помощи php-скриптов [9]. Для автономной и наиболее быстрой работы с заданиями (не требующего подключения к сети интернет) было решено создать файл в директории игры, а не в самой базе данных, расположенной на сервере.

2.3.1 Создание и подключение базы данных с пользователями

Сервер базы данных и сервер с пользовательским интерфейсом могут находиться физически на разных компьютерах (серверах), что обеспечивает большую безопасность информационной системы [7], а также большее быстродействие приложения, либо на одном компьютере (сервере). Последний вариант упрощает физическую составляющую ИС (аппаратную часть), удешевляет её. Схема взаимодействия сервера и пользовательского интерфейса показан на рисунке 11.

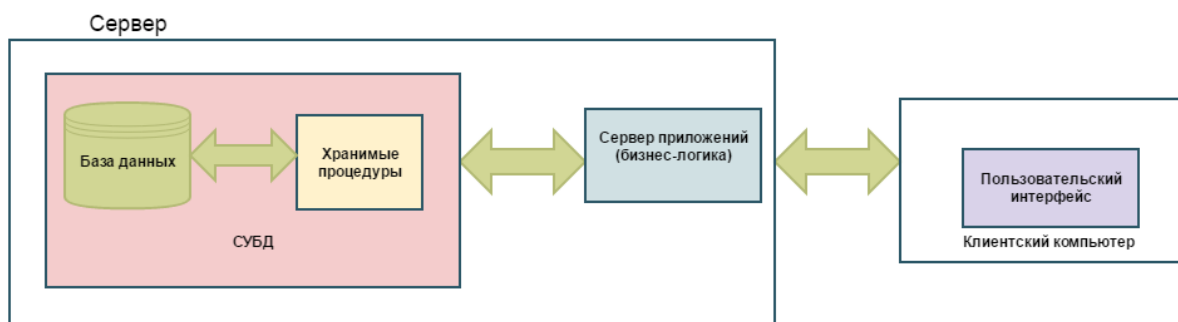


Рисунок 11 – Схема взаимодействия сервера и пользовательского интерфейса

Алгоритм авторизации и регистрации прост и выглядит следующим образом. В компьютерную модель вводятся данные о пользователе, а именно его логин и пароль, затем введенные данные передаются в соответствующие скрипты на сервере, проверяются на корректность и правильность. Если все данные были введены верно, и система не выдала

никаких предупреждений, то пользователь перенаправляется на следующую страницу, в противном случае система выдает ошибку и предлагает ввести заново данные.

2.3.2 Схема базы данных

Для реализации программного продукта потребовалось создать базу данных MySQL с одной таблицей «Users» (таблица 2) для хранения данных о пользователе (это логин, пароль и прогресс пользователя).

Таблица 2 – Users

Столбец	Тип	Null	По умолчанию
ID	int(11)	Нет	
Login	varchar(255)	Да	NULL
Password	varchar(255)	Да	NULL
ProgressLevel	int(11)	Да	NULL

2.3.3 Концепция проекта

Сюжет

Мир, в котором происходит действие игры, представлен в виде двух пространственных измерений: объективной действительности и параллельной ей геометрической реальности, существующей в сновидении главного героя, связанные посредством объектов, находящихся одновременно в обоих пространствах. Для их создания был нарисован ряд спрайтов и анимаций.

Локации

Мир объективной действительности представлен на рисунке 12.



Рисунок 12 – Мир объективной действительности

На рисунке 13 представлен мир параллельной реальности.

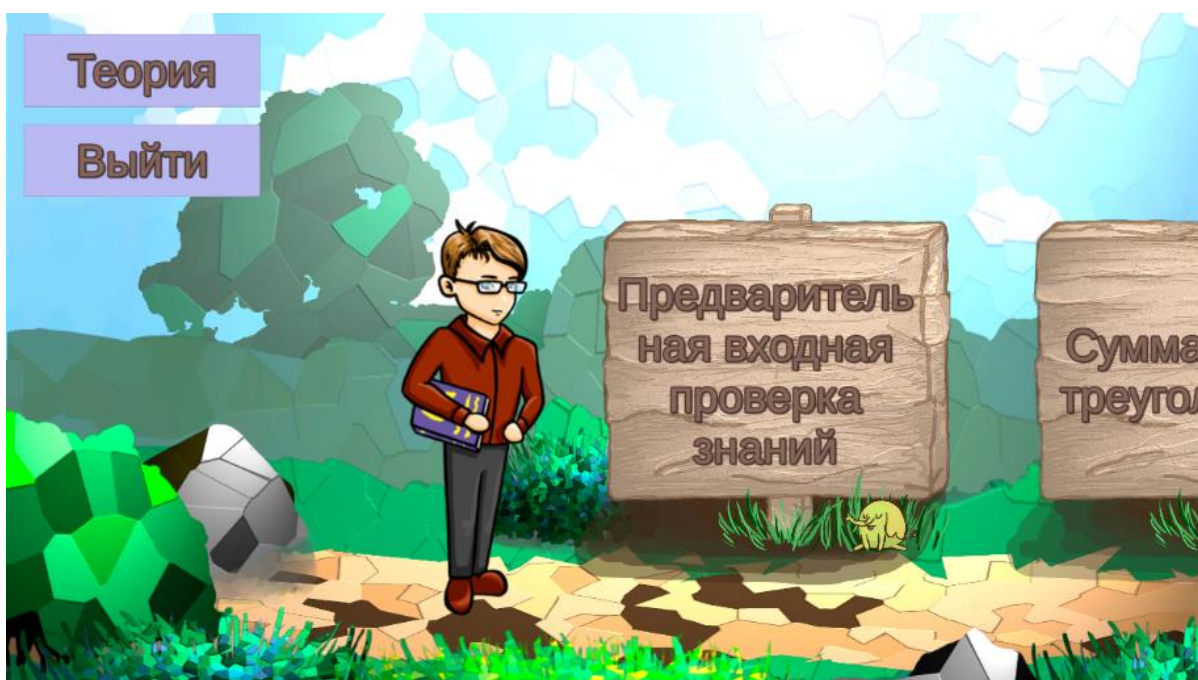


Рисунок 13 – Мир параллельной геометрической реальности

Главный герой

Для отражения пространственного измерения объективной действительности персонаж и анимации рисовались в растровом графическом редакторе (рисунок 14).

то, что встроенные компоненты Unity могут быть очень разносторонними, может оказаться, что их возможностей недостаточно, чтобы реализовать особенности геймплея. Чтобы исправить это, Unity позволяет создавать свои компоненты, используя скрипты. Они позволяют активировать игровые события, изменять параметры компонентов, и отвечать на ввод пользователя каким угодно способом [8]. Unity изначально поддерживает два языка программирования: C# и UnityScript. По программе обучения мною был изучен язык программирования C#, именно поэтому был выбран именно этот язык программирования для написания кода.

Наиболее сложная часть реализации компьютерной модели адаптивной обучающей игры связана с адаптацией процесса контроля прохождения заданий под каждого пользователя. Для того чтобы игра была адаптивной, нами были добавлены следующие функции:

1. Функция проверки входных знаний. После регистрации пользователь может пройти входную проверку знаний. Алгоритм выбирает из каждого уровня каждой темы случайное задание и выводит на экран. Затем определяется количество правильных заданий. Таким образом, после прохождения проверки, пользователю будут доступны уровни, на которые он дал правильные ответы.

2. Функция проверки правильно выполненных заданий. Если все задания пройдены верно, то прогресс пользователя в базе данных обновляется.

3. Функция вывода страницы, если пользователь допустил две ошибки подряд, позволяющей сделать выбор: пройти уровень заново, вернуться к повторению теории или продолжить (в таком случае, уровень не будет пройден).

4. Функция вывода подсказки к текущему заданию, если пользователь в течение определенного количества времени не выбрал ответ.

Скрипт с кодом, позволяющий вычислять данные функции показан в Листинге 1.

Листинг 1 – Код вывода и анализа заданий

```
public TextAsset ex;
public Font fontStyle;

public GUIStyle customButton;
public int rndOpt;
public int peremProgrExQuest = CurrentUser.peremProgrEx; //Переменная ПРОЙДЕННЫХ
заданий В уровне
public int progrQuest=0; //Переменная ПРАВИЛЬНО пройденных заданий в уровне
public int wrongAnsw=0; //Переменная ОШИБОЧНО пройденных заданий в уровне
public GameObject wrong;

/ TIMER
[Header("Text")]
[SerializeField] private TextMeshProUGUI textTimer;
[SerializeField] private char characterSqliter = ':';

[Header("Logic")]
private float timer;
private bool isActive;
public bool pt;

public void Update()
{
    if (isActive)
    {
        timer += Time.deltaTime;
        UpdateText();
    }
}

private void UpdateText()
{
    float seconds = (timer % 60);
    float minutes = ((int)(timer / 60) % 60);
    float hours = (int)(timer / 3600);

    textTimer.text = hours.ToString("00") + characterSqliter
+ minutes.ToString("00") + characterSqliter
+ seconds.ToString("00");
}

public void StartTime(float second)
{
    isActive = true;
    timer = second;
    UpdateText();
}

public void ResetTimer()
{
    timer = 0;
    UpdateText();
}

public float GetTimeSinceStart()
{
    return timer;
}
```

Продолжение листинга 1

```
// Start is called before the first frame update
void Start()
{
    int[] a = { 0, 1, 2};
    rndOpt = a[new System.Random().Next(0, a.Length)];
    StartTime (0);
}

// Вернуться к теории
public void ChangeQuestTheory()
{
    SceneManager.LoadScene(4);
}

// Перепройти уровень
public void ChangeWrongQuest()
{
    progrQuest=0;
    peremProgrExQuest=0;
    wrongAnsw = 0;
    Start();
    pt = false;
    ResetTimer();
    wrong.SetActive(false);
}

//Продолжить проходить уровень, несмотря на ошибки
public void ContinueWrongQuest()
{
    ExerciseForXMLClass forExer = ExerciseForXMLClass.Load(ex);
    int curTheme = ThemesUI.currentTheme;
    int currentLev = CurrentUser.exNum;
    if (peremProgrExQuest ==
forExer.themes[curTheme].levels[currentLev].exercises.Length)
    {
        SceneManager.LoadScene(1);
    }
    else
    {
        wrong.SetActive(false);
        ResetTimer();
    }
}

// OnGUI for Questions
private void OnGUI()
{
    int curTheme = ThemesUI.currentTheme;
    int currentLev = CurrentUser.exNum;

    GUIStyle myButtonStyle = new GUIStyle(GUI.skin.button);
    myButtonStyle.fontSize = 45;

    // Load and set Font
    Font myFont = fontStyle;
    myButtonStyle.font = myFont;

    GUIStyle guiStyle = new GUIStyle();
    guiStyle.fontSize = 40; //change the font size
    guiStyle.font = myFont;
```

Продолжение листинга 1

```
guiStyle.wordWrap = true;

ExerciseForXMLClass forExer = ExerciseForXMLClass.Load(ex);

// Если прошло пять минут, вывести ПОДСКАЗКУ
if (textTimer.text == "00:00:02")
    pt = true;

// Если панелька с ошибкой НЕ активна
if (wrong.activeSelf == false)
{
    //Если количество пройденных заданий меньше чем в уровне
    if (peremProgrExQuest <
forExer.themes[curTheme].levels[currentLev].exercises.Length) //Переменная, отвечающая
за количество пройденных заданий в уровне
    {
        //Load a Texture (Assets/Resources/Textures/texture01.png)
        var texture = Resources.Load<Texture2D>("Textures/" +
forExer.themes[curTheme].levels[currentLev].exercises[peremProgrExQuest].options[rndOp
t].picture);
        GUI.DrawTexture(position: new Rect(-200, 100, Screen.width - 400,
Screen.height - 400), texture, ScaleMode.ScaleToFit);

        GUI.Label(new Rect(Screen.width - 850, 100, Screen.width - 900,
Screen.height - 350),
forExer.themes[curTheme].levels[currentLev].exercises[peremProgrExQuest].options[rndOp
t].textOpt, guiStyle);

        if (pt == true)
        {
            // Make a button using a custom GUIContent parameter to pass in
the tooltip.
            GUI.Button(new Rect(Screen.width - 518, Screen.width - 900, 50,
50), new GUIContent("?",
forExer.themes[curTheme].levels[currentLev].exercises[peremProgrExQuest].exHelp));
            GUI.Label(new Rect(Screen.width - 850, Screen.width - 840, 400,
40), GUI.tooltip, guiStyle);
        }

        GUILayout.BeginArea(new Rect(80, 440, 320, Screen.height - 300));

        //Обходим все элементы EX в LEVELS
        for (int j = 0; j <
forExer.themes[curTheme].levels[currentLev].exercises[peremProgrExQuest].options[rndOp
t].answers.Length; j++)
        {
            //Если кнопка ответов нажата
            if
(GUILayout.Button(forExer.themes[curTheme].levels[currentLev].exercises[peremProgrExQu
est].options[rndOpt].answers[j].answerText, customButton))
            {
                ResetTimer();
                //Если нажата кнопка с правильным ответом
                if
(forExer.themes[curTheme].levels[currentLev].exercises[peremProgrExQuest].options[rndO
pt].answers[j].answerNum == "true")
                {
                    //Переменная ПРАВИЛЬНО пройденных заданий в уровне
увеличивается на 1
                    progrQuest++;
                    peremProgrExQuest++;
                    wrongAnsw = 0;
                }
            }
        }
    }
}
```

Продолжение листинга 1

```
        Start();
        pt = false;
        break;
    }
    // Если нажта кнопка с неправильным ответом:
    else
    {
        //Переменная пройденных заданий В уровне увеличивается на
1
        peremProgrExQuest++;
        wrongAnsw++;

        // Если переменная пройденных заданий РАВНА
        // Количеству заданий в уровне
        if (peremProgrExQuest ==
forExer.themes[curTheme].levels[currentLev].exercises.Length)
        {
            SceneManager.LoadScene(1);
        }

        // Иначе, если количество последних двух ответов ошибочны
        else if (wrongAnsw == 2)
        {
            wrong.SetActive(true);
        }
        Start();
        pt = false;
        break;
    }
    }
}
GUILayout.EndArea();
}

//Если количество пройденных заданий РАВНО количеству пройденных заданий в
уровне
else
{
    // Если количество решенных заданий РАВНО
    // Количеству заданий в уровне
    if (peremProgrExQuest ==
forExer.themes[curTheme].levels[currentLev].exercises.Length)
    {
        // Если количество ПРАВИЛЬНО решенных заданий РАВНО
        // Количеству заданий в уровне
        if (progrQuest ==
forExer.themes[curTheme].levels[currentLev].exercises.Length)
        {
            //Если текущая тема РАВНА последней теме юзера ИИИ
            //Если текущий уровень РАВЕН последнему уровню юзера
            if (forExer.themes[curTheme].numberTheme == Xml111.progT - 1 &&
forExer.themes[curTheme].levels[currentLev].talonExer == Xml111.progU - 1)
            {
                //Если текущий уровень РАВЕН
                //количеству уровней в теме
                if (forExer.themes[curTheme].levels[currentLev].talonExer
+ 1 == forExer.themes[curTheme].levels.Length)
                {
                    Xml111.progT++;
                    Xml111.progU = 1;
                }
            }
        }
    }
}
```

Продолжение листинга 1

```
        else
        {
            Xml111.progU++;
        }
    }
}
SceneManager.LoadScene(1);
}
}
```

2.4 Описание компьютерной модели обучающей игры

Подход к интеграции процесса обучения в игру зависит от внутреннего представления (модели) предметной области, изучаемой в игре. По способу представления предметной области можно выделить игры на основе имитационного моделирования, на основе ситуационного моделирования и на основе формально-логической модели.

Имитационные модели реализуются в играх-симуляторах (simulator), которые моделируют реальные условия профессиональной деятельности специалиста в некоторой области знаний. К этому классу игр относятся симуляторы управления различными транспортными средствами, медицинские, тактические, социальные и бизнес-симуляторы. Разработка симуляторов включает разработку реалистичной модели процесса в соответствующей предметной области, составляющей предмет изучения в игре [10]. Например, разработка авиа-симулятора (Microsoft® Flight Simulator X, X-Plane и др.) требует создания физической (симуляция параметров самолёта и его взаимодействия со средой) и графической (виртуальная или реальная машина пилота) моделей. Игровая составляющая в таких играх реализуется с использованием богатых графических и звуковых возможностей современных средств вычислительной техники для компьютерной интерпретации реальных процессов.

Ситуационные модели используются в играх, в которых реализовано обучение на примерах (case study-подход) [2] с использованием

возможностей компьютерных игр (организация диалогов, визуализация персонажей, событий и среды обитания и т.д.). Эти игры, как правило, реализуются в ролевом (role-playing) или приключенческом (adventure) жанре. Игрок помещается в виртуальную среду, созданную на основе описаний ситуаций из действительности, действует в соответствии с заложенным в игру сценарием (планом действий), ведет предварительно предусмотренные в игре диалоги, выбирает свои решения из большого количества заданных наборов и в итоге обучается на примерах правильных решений. Процесс проектирования таких игр включает разработку описания, игрового сюжета на основе сценария, разработку виртуальной среды и способа ее отображения. Основной задачей разработчиков таких обучающих игр является выбор и реализация способов наиболее привлекательного с игровой точки зрения воспроизведения сценария.

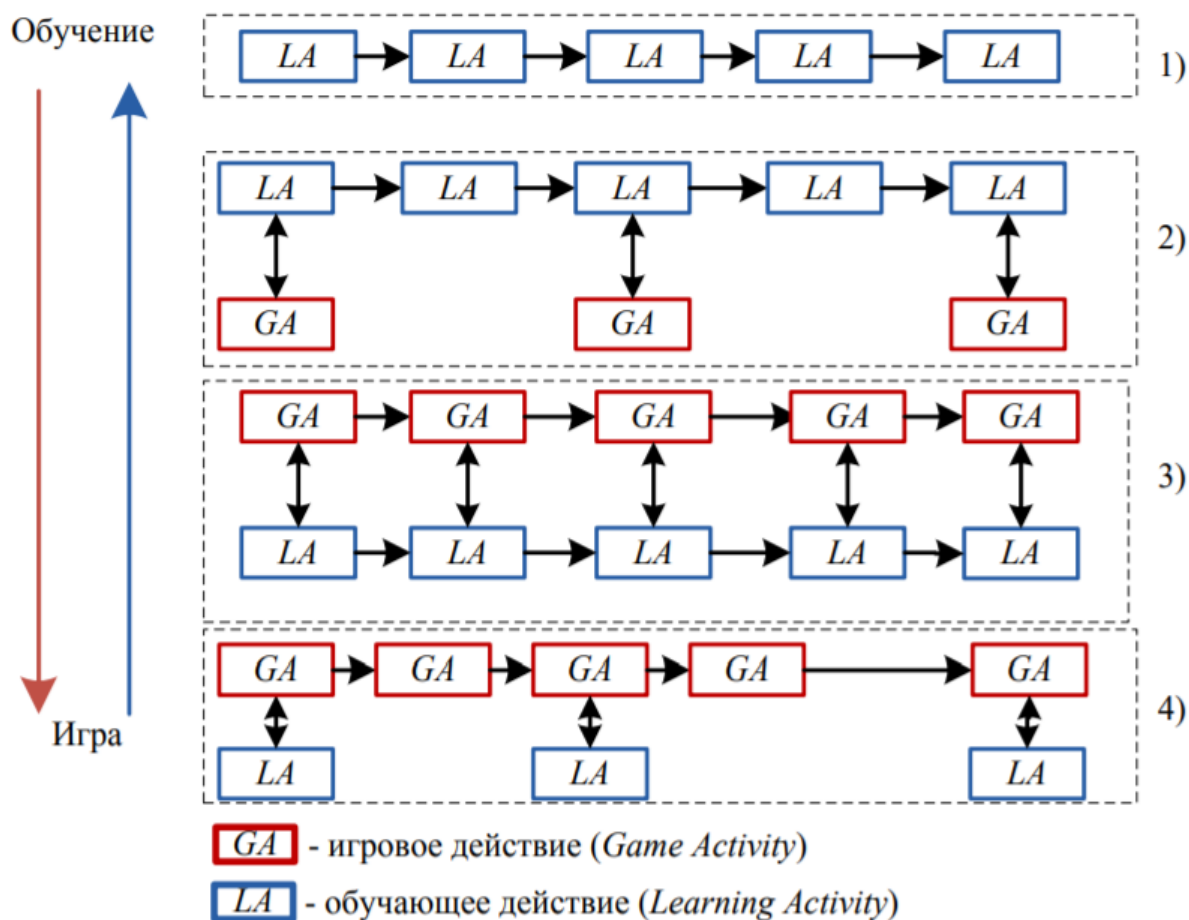
В таких играх, как паззлы (puzzle games), игры на сопоставление объектов (matching games), различные варианты игр на тренировку памяти (memory games, brain training games) используется формально-логическая модель предметной области. Такие игры основываются на проверке соответствия вводимых игроком данных заданным в системе паттернам.

На рисунке 16 показаны возможности различных моделей предметной области с точки зрения взаимодействия игровой и обучающей компонент. Ситуационные и имитационные являются менее формализованными и предоставляют разработчику игры больше возможностей для организации геймплея. Однако такие модели применимы для случаев, когда изучаемая предметная область имеет реальную интерпретацию.



Рисунок 16 – Возможности различных моделей предметной области с точки зрения организации взаимодействия игровой и обучающей компонент

Анализ подходов к интеграции обучающих и игровых компонентов позволяет выделить четыре способа организации сценариев в обучающих играх. Под сценарием обучающей игры понимают набор взаимосвязанных элементов сценария, представляющих игровую и обучающую компоненты. Доминирование в игре обучающей или игровой компоненты определяет тип сценария, реализованного в игре (рисунок 17).



- 1) обучающий сценарий
- 2) обучающий сценарий с элементами игры
- 3) независимые игровой и обучающий сценарии
- 4) игровой сценарий с элементами обучения

Рисунок 17 – Модели сценариев в обучающих играх

Типичными примерами игр, в которых реализован только обучающий сценарий, являются игры-симуляторы. Иногда такие игры выделяют в отдельную категорию обучающих систем и не относят к категории игр [11]. Применение в обучающих системах различных игровых элементов определяет принадлежность таких систем к обучающим играм второй группы. Интеграция отдельных обучающих действий в игровой сценарий позволяет усилить обучающий эффект, однако цель обучения в явном виде не определяется.

Нами был выбран третий способ организации сценария, основанный на разработке обучающих игр, в которых существуют два

непересекающихся сценария, и, соответственно, две непересекающиеся цели – цель обучения и игровая цель. В нашем случае, для достижения игровой цели необходимо достижение обучающей цели.

Выводы по Главе 2

Во второй главе были рассмотрены инструментальные средства реализации автоматизированной системы: в качестве платформы для разработки игр была выбрана Unity 2019 и язык программирования C#. Выбран хостинг и зарегистрирован домен. Для работы с базой данных на стороне сервера был использован язык программирования PHP. В качестве базы данных используется MySQL. Рассмотрены графические редакторы для рисования скетчей, деталей интерфейса, фона, создания бесшовных текстур. Рассмотрено взаимодействие компьютерной модели с сервером, сохранение данных о пользователе в базу данных.

Выделены основные функции, касающиеся адаптивности игры: проверка входных знаний; проверка правильно выполненных заданий; вывод страницы с ошибкой; предоставления подсказки к текущему заданию.

ГЛАВА 3. ДОКУМЕНТАЦИЯ ГОТОВОЙ ИГРЫ

3.1 Испытание компьютерной модели адаптивной обучающей игры

Целью данного проекта является проверка работоспособности игры и соответствие результатов ее работы с требованиями корректности, надежности и правильности.

Тестирование компьютерной модели необходимо разбить на два блока.

Блок первый – регистрация и авторизация пользователей в игре. В данном блоке проверяется работоспособность кода, отвечающего за подключение к серверу и базе данных, расположенной в нем. При вводе неверных данных (пустых или уже имеющих в БД) в окне регистрации, пользователю показывается соответствующая ошибка. При правильном вводе данных будет доступна возможность пройти этап аутентификации. Если пользователь ввел корректные данные в окне с авторизацией, то игра считает прогресс данного пользователя и перенаправляет на страницу с темами, доступными ему. В противном случае игра выведет оповещение с ошибкой авторизации. После каждого пройденного уровня сохраняется прогресс пользователя в базе данных.

Блок второй – проверяем игру на возможные ошибки в работе основной страницы с заданиями. Проверка вывода и определение правильно выполненных заданий, результатов работы кода, отвечающего за функции адаптивности игры.

3.2 Руководство пользователя (учащегося) компьютерной модели адаптивной обучающей игры

Для запуска игры необходимо ввести в адресную строку вашего браузера <https://geometrylittledi.ru/>. Откроется начальная страница игры (рисунок 18).



Рисунок 18 – Начальная страница игры

При клике на кнопку «Регистрация» появится окно регистрации, в котором необходимо ввести логин и пароль (рисунок 19).

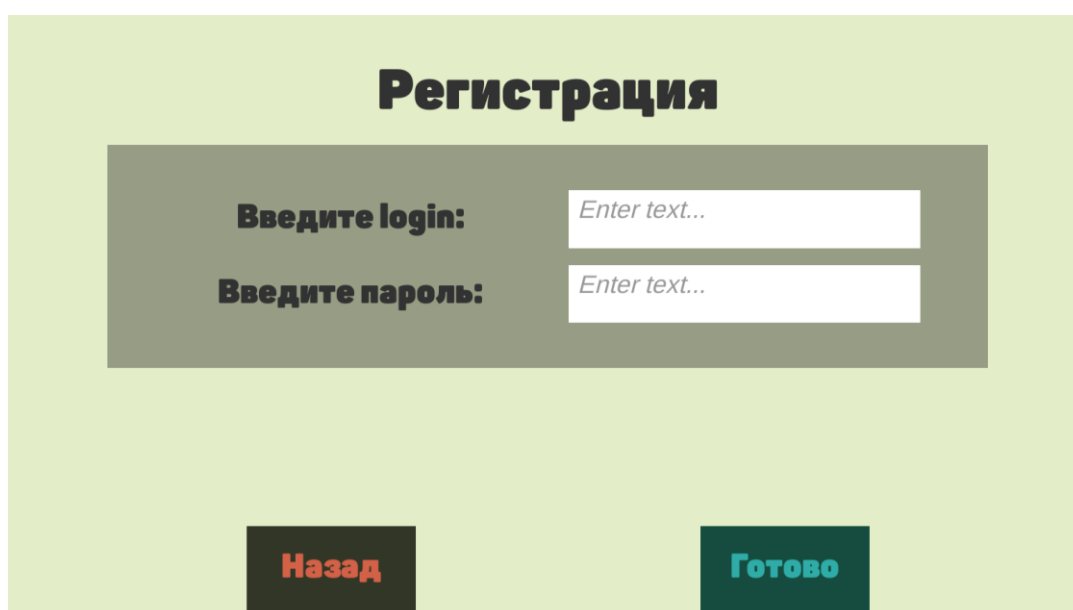
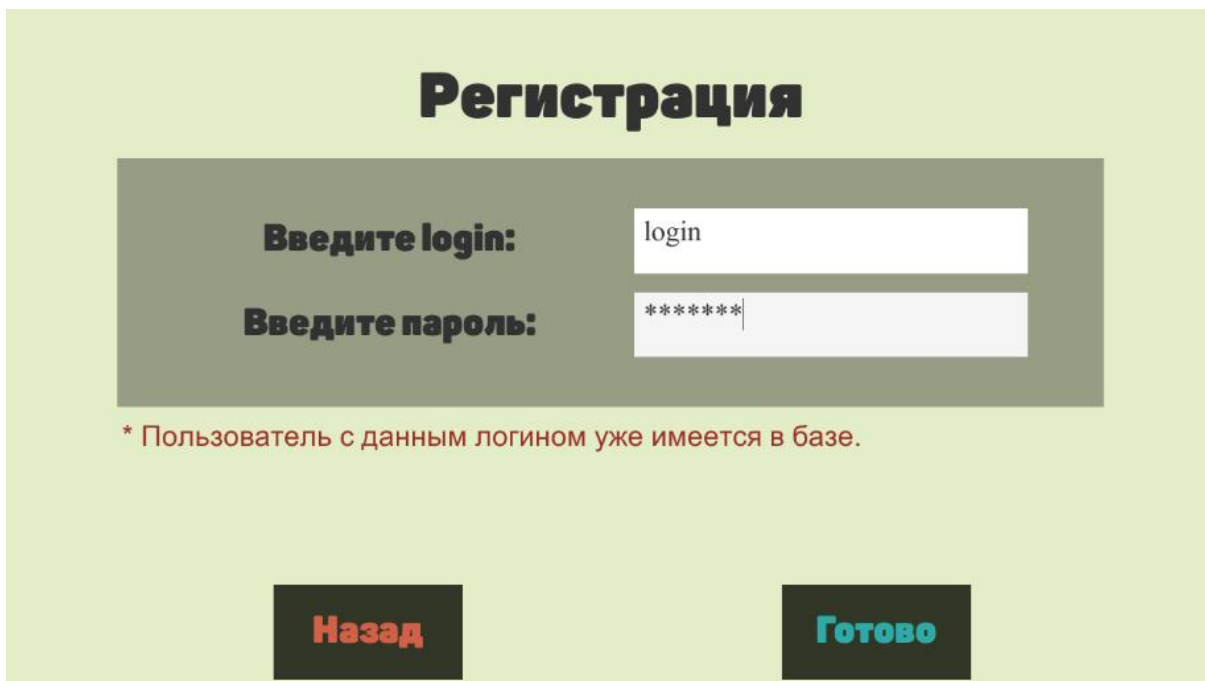


Рисунок 19 – Страница с регистрацией

Если в базе данных имеется пользователь с таким логином, то игра показывает соответствующее сообщение (рисунок 20).



The screenshot shows a registration page titled "Регистрация" (Registration) on a light green background. A dark grey rectangular box contains the registration form. On the left, there are two labels: "Введите login:" (Enter login) and "Введите пароль:" (Enter password). To the right of these labels are two input fields. The first field contains the text "login", and the second field contains "*****" with a cursor at the end. Below the input fields, a red error message reads: "* Пользователь с данным логином уже имеется в базе." (User with this login already exists in the database). At the bottom of the dark grey box, there are two buttons: "Назад" (Back) in red text on a black background, and "Готово" (Done) in cyan text on a black background.

Рисунок 20 – Страница с сообщением об ошибке регистрации пользователя

После успешной регистрации пользователю необходимо авторизоваться, нажав на кнопку «Войти» (рисунок 21).



The screenshot shows the same registration page as in Figure 20, but with a success message. The dark grey box now contains the text: "Пользователь с ником okfok успешно зарегистрирован!" (User with nickname okfok successfully registered!). Below this message is a single button labeled "Войти" (Log in) in cyan text on a black background. At the bottom of the page, there are two buttons: "Назад" (Back) in red text on a black background, and "Готово" (Done) in cyan text on a black background.

Рисунок 21 – Окно с сообщением об успешной регистрации

После открытия окна авторизации, в котором необходимо ввести зарегистрированные логин и пароль пользователя (рисунок 22).

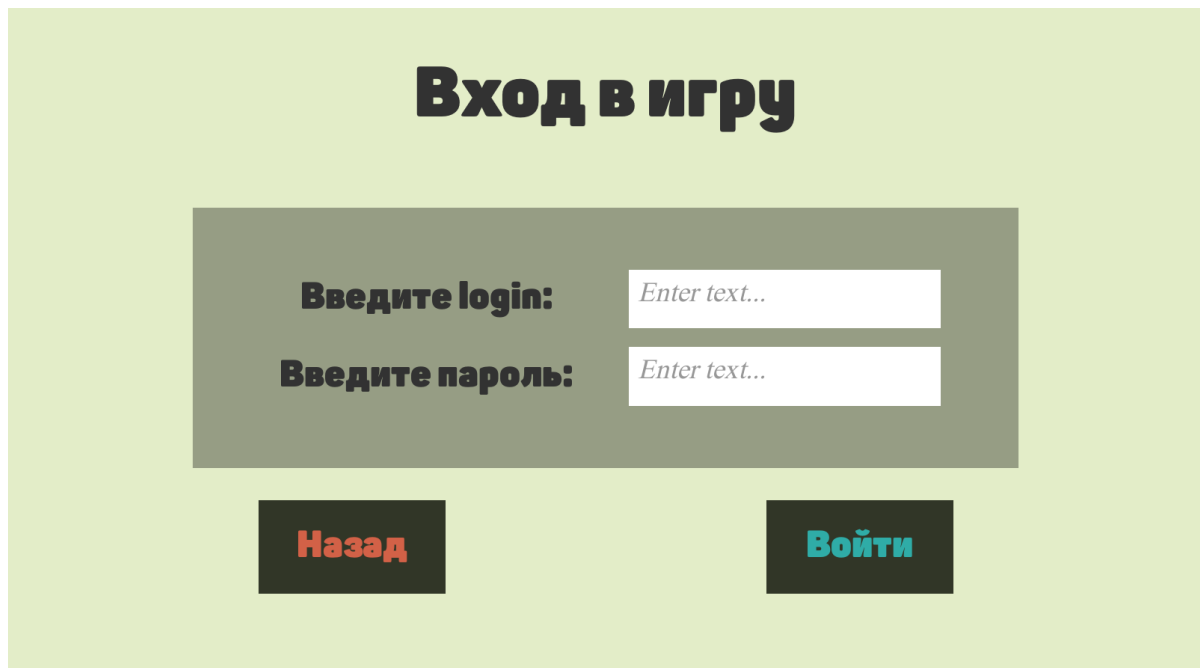


Рисунок 22 – Страница авторизации пользователя

После успешной авторизации открывается сцена с отражением объективной реальности (рисунок 23).

При первом входе в игру необходимо пройти к лавочке и кликнуть по ней.



Рисунок 23 – Игровой мир объективной действительности

Далее персонаж попадает в параллельную геометрическую реальность. Пользователь может пройти предварительную проверку знаний, для этого необходимо кликнуть по соответствующей табличке (рисунок 24).

Также на сцене имеются кнопки «Выйти» и «Теория».

При клике на кнопку «Выйти» открывается начальное окно (рисунок 18).

При клике на кнопку «Теория» открывается окно с теоретическим материалом, необходимых для решения задач (рисунок 30).

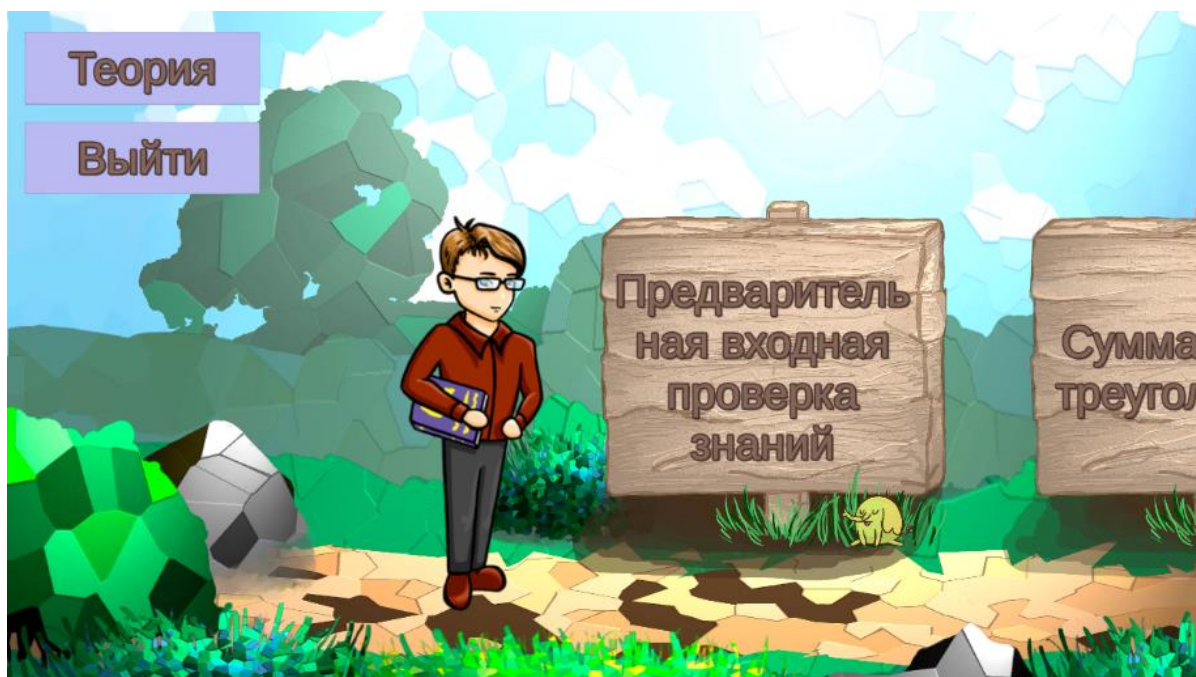
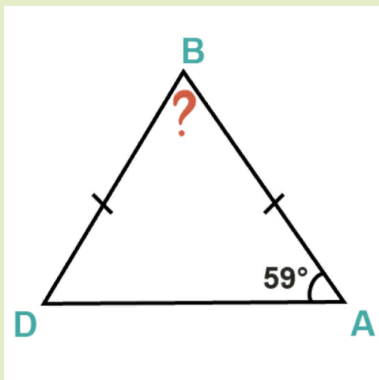


Рисунок 24 – Игровой мир параллельной геометрической реальности

Далее открывается окно с заданиями входной предварительной проверки знаний (рисунок 26).

Задание 1 / 7

Найдите угол В треугольника $\triangle ABC$, если угол $A = 59^\circ$.



- 45
- 68
- 83
- 62



Рисунок 25 – Страница с заданиями входной проверки знаний

После прохождения предварительной проверки пользователь может ознакомиться с результатом (рисунок 26). Результат автоматически сохраняется в базу данных. Игроку будут доступны те уровни, которые он готов выполнить по результатам предварительного тестирования.

**Поздравляем, вам
доступны уровни: 3**

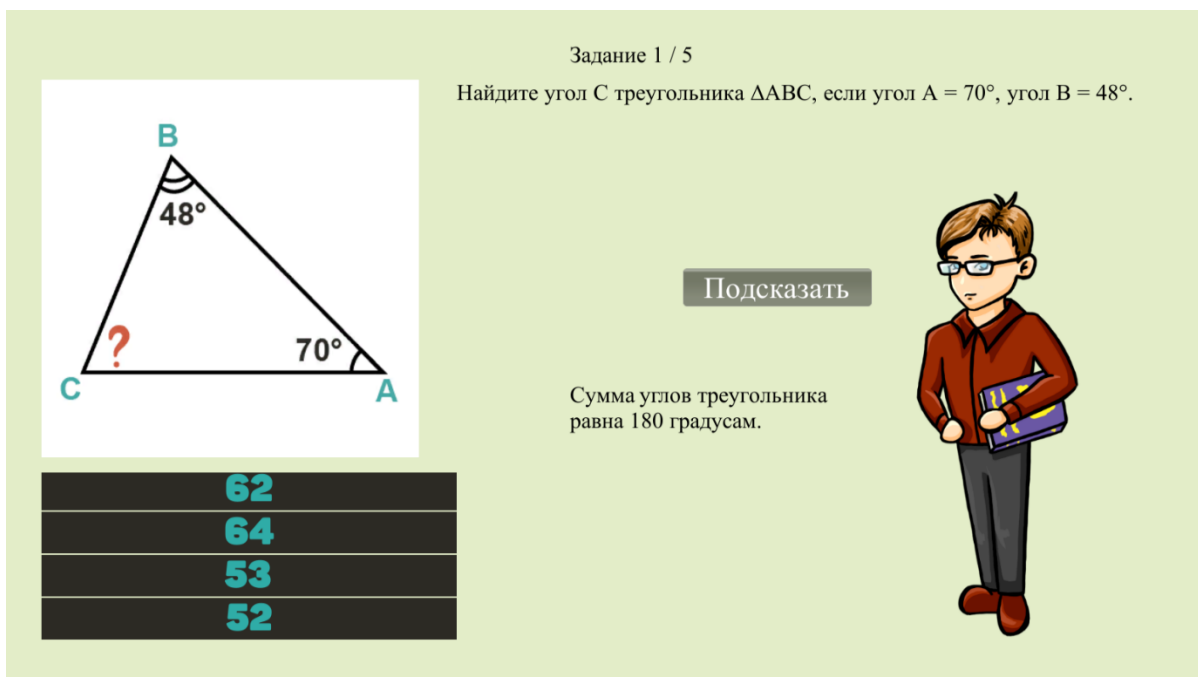
Выбрать уровень



Рисунок 26 – Окно результатов предварительной входной проверки знаний

Для дальнейшего прохождения игры необходимо выбрать последнюю видимую таблицу и кликнуть по ней. Откроется окно с заданиями в данном уровне (рисунок 27).

По истечении пяти секунд появляется кнопка с подсказкой. Для отображения подсказок необходимо навести мышку на появившуюся кнопку (рисунок 27).



The screenshot shows a game interface with a light green background. On the left, there is a diagram of a triangle with vertices labeled A, B, and C. Angle B is marked as 48 degrees, angle A is marked as 70 degrees, and angle C is marked with a red question mark. Below the diagram is a vertical list of four options: 62, 64, 53, and 52. To the right of the diagram, the text reads 'Задание 1 / 5' and 'Найдите угол C треугольника $\triangle ABC$, если угол $A = 70^\circ$, угол $B = 48^\circ$ '. Below this text is a grey button labeled 'Подсказать'. Further down, a hint is displayed: 'Сумма углов треугольника равна 180 градусам.' To the right of the hint is a cartoon illustration of a boy with glasses, wearing a red shirt and grey pants, carrying a blue and yellow backpack.

Рисунок 27 – Страница с заданиями в уровнях

После успешного прохождения всех заданий пользователь возвращается к сцене геометрической реальности (рисунок 24), результат автоматически сохраняется в базу данных и при повторном входе в игру данный уровень будет доступен.

Если пользователь допустил ошибки во всех заданиях подряд появляется окно с соответствующим предупреждением (рисунок 28).

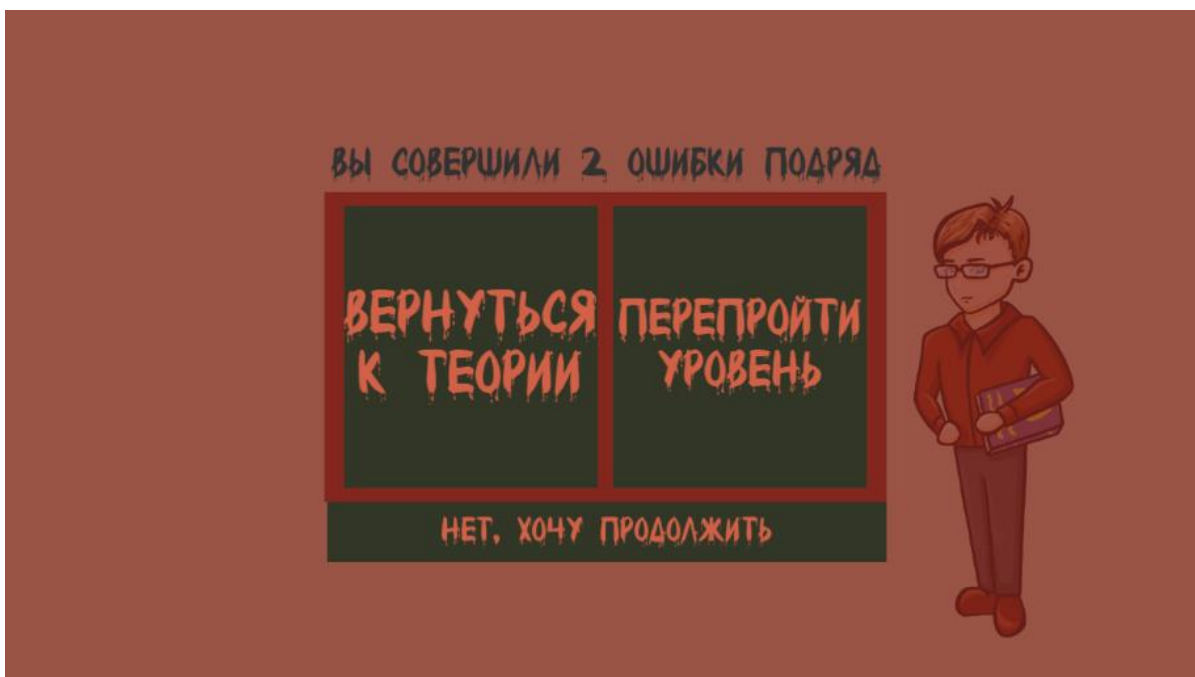


Рисунок 28 – Окно с предупреждением о совершении двух ошибок подряд

Если нажата кнопка «Пройти уровень заново» открывается окно с заданиями (рисунок 27) из этого же уровня. Результат предыдущего прохождения сбрасывается.

При клике на кнопку «Нет, хочу продолжить» закрывается окно с предупреждением о совершении двух ошибок подряд, дальнейшие ответы не учитываются, пользователь может ознакомиться с другими заданиями этого уровня.

При нажатии на кнопку «Вернуться к теории» открывается окно с названиями тем теоретического материала (рисунок 29).

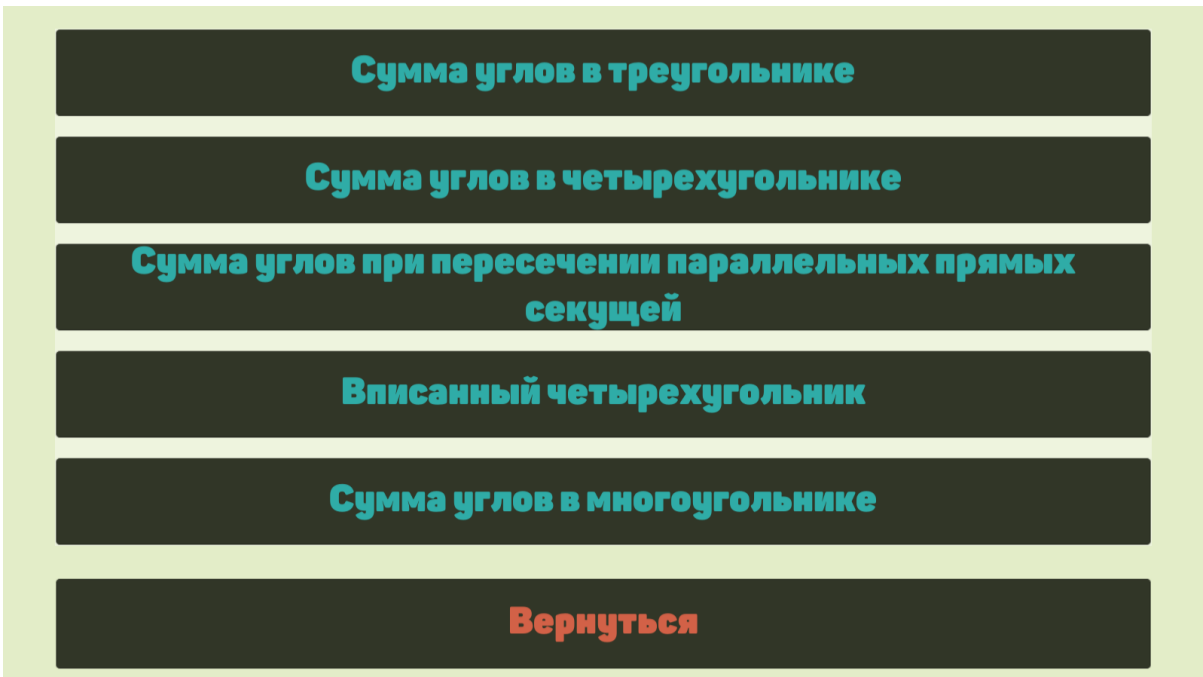


Рисунок 29 – Страница с темами теоретического материала игры
 После выбора темы открывается соответствующая страница с теорией (рисунок 30).

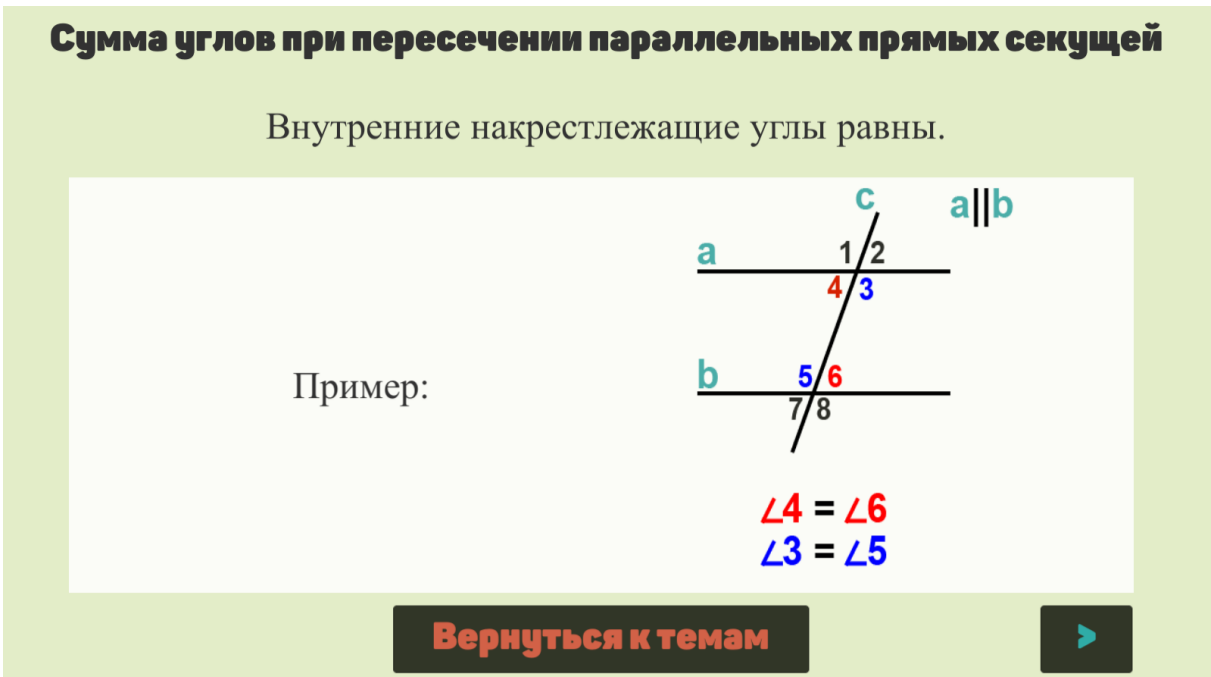


Рисунок 30 – Страница с теорией
 После прохождения всех уровней игрок возвращается в мир объективной действительности (рисунок 31).

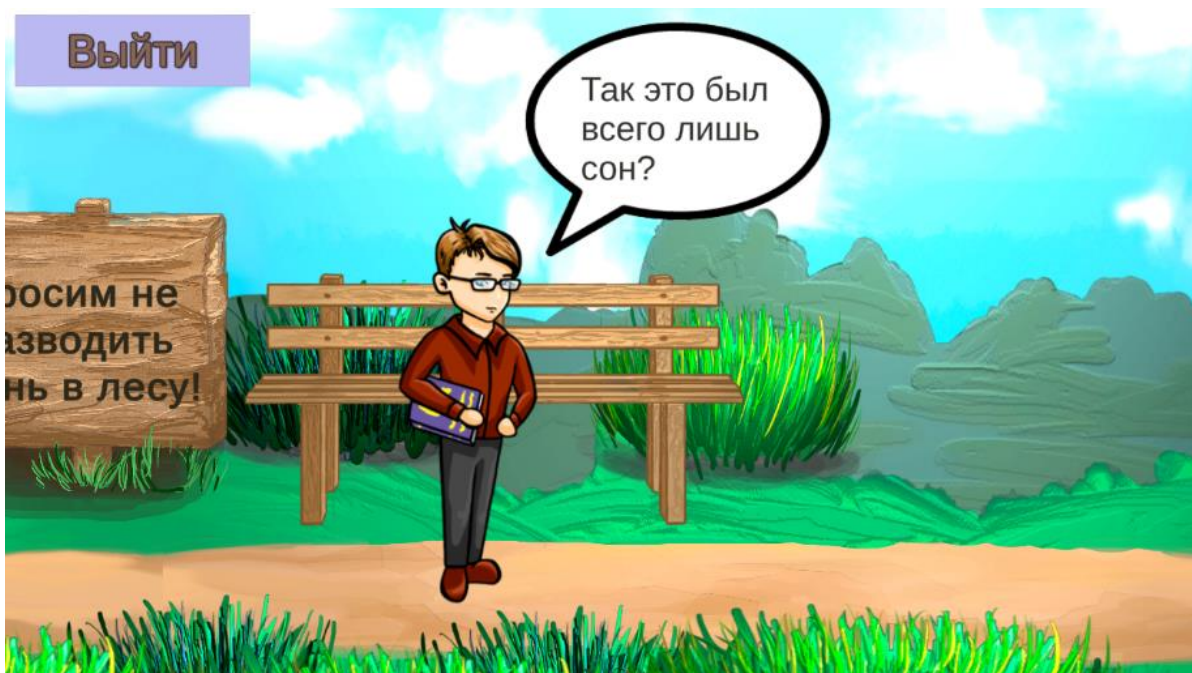


Рисунок 31 – Конец игры

3.3 Технико-экономическое обоснование разработки компьютерной модели адаптивной обучающей игры

Технико-экономическое обоснование (ТЭО) – документ, в котором представлена информация, из которой выводится целесообразность (или нецелесообразность) создания продукта или услуги. ТЭО содержит анализ затрат и результатов какого-либо проекта.

При разработке программного продукта могут потребовать трудовые затраты: использование вычислительной техники, в нашем случае аренда хостинга, а также и другие затраты целесообразность которых можно обосновать, выполняя пункт технико-экономического обоснования настоящей выпускной квалификационной работы.

Поэтому приступая к описанию технико-экономического обоснования в рамках выпускной работы, необходимо затронуть следующие вопросы:

- расчет затрат на оплату труда;
- расчет на затраты для материалов;
- определение величины прочих прямых расходов;

– расчет полной себестоимости разрабатываемой системы.

Расчет затрат на оплату труда

Выполненная работа может определяться как оплата труда исполнителей и руководителей проектной группы, а также оплата труда рабочих дней. Расчет затрат на оплату труда представлен в таблице 3.

Таблица 3 – Расчет затрат на оплату труда

№	Показатель	Единица измерения	Время/стоимость выполняемой работы	Оклад руб./мес.	Сумма затрат
1	Затраченное время исполнителя на разработку	День	60	-	60
2	Ставка дневной заработной платы исполнителя	Руб.	400	24000	24000
3	Страховые взносы на обязательное пенсионное страхование	Руб.	-	22%	5280
4	Страховой взнос на обязательное соц. страхование на случай временной нетрудоспособности	Руб.	-	2,9%	153
5	Страховой взнос на обязательное медицинское страхование	Руб.	-	5,1%	269
Итого (руб.)					24000
Итого с учетом страховых взносов (руб.)					29702

Расчет на затраты для материалов

В процессе реализации программного продукта могут быть задействованы материалы необходимые для разработки или материалы требуемые на предмет разработки, позволяющие определить эффективность и качество выполнения разработки проекта. В перечень затрат на материалы могут входить затраты: расходы на канцелярские товары, расходы на заправку картриджей, а также другие расходы на материалы. Перечень расходов и их расчет представлен в таблице 4.

Таблица 4 – Перечень затрат на материалы

№	Наименование материала	Единицы измерения	Количество	Цена ед. (руб.)	Сумма затрат (руб.)
1	Бумага для печати на принтере	Уп.	1	200	200
2	Папка-скоросшиватель	Шт.	1	57	57
Итого (руб.)					257

Определение величины прочих прямых расходов

К определению величины прямых расходов можно отнести устройства приборы, а также технические средства, с помощью которых разрабатывается программный продукт. В перечень можно так же отнести расходы за оказания услуг, сопутствующих разработке, транспортные расходы и другие внешние расходы. Перечень представлен в таблице 5.

Таблица 5 – Перечень прямых расходов

№	Показатель	Ед. измерения	Количество	Стоимость ед. (руб.)	Сумма затрат мес. (руб.)
1	Аренда хостинга	Месяц	12	67	804
2	Покупка домена	Год	1	270	270
3	Проезд	Поездки	19	25	475
Итого (руб.):					1549

Расчет полной себестоимости разрабатываемой системы

представлен в таблице 6.

Таблица 6 – Расчет себестоимости разрабатываемой системы

№	Показатель	Единица измерения	Сумма затрат
1	Суммарный расчет на оплату труда с учетом страховых взносов	Руб.	29702
2	Расходы на материалы	Руб.	257
3	Перечень прямых затрат	Руб.	1549
Итого (руб.):			31508
Средний уровень рентабельности проектов		%	25
Планируемая прибыль		Руб.	7877
Цена разработки информационной системы		Руб.	39385

Выводы по Главе 3

В третьей главе были описаны и представлены этапы испытания разработанной системы, а также её настройка и тестирование.

Испытания производились на территории учебного центра дополнительного образования «Вектор знаний». Все файлы программного продукта были переданы в структурное подразделение вместе с руководством пользователя и инструкцией по эксплуатации для дальнейшего внедрения сотрудниками учебного центра.

В процессе тестирования разработанный продукт работал без сбоев и ошибок.

Разработанное руководство к системе позволит упростить взаимодействие пользователя с системой.

Для расчета рентабельности проекта было осуществлено технико-экономическое обоснование информационной системы, цена разрабатываемой системы составляет 39385 рублей.

ЗАКЛЮЧЕНИЕ

В ходе написания квалификационной работы была спроектирована и реализована компьютерная модель адаптивной обучающей игры, предназначенная для использования учащимися учебного центра дополнительного образования «Вектор знаний».

В качестве средств разработки была выбрана платформа Unity 2019, язык программирования C#, для работы с базой данных на стороне сервера был использован язык программирования PHP, в качестве СУБД была выбрана MySQL, для рисования иллюстраций были выбраны графические редакторы Adobe Photoshop и Adobe Illustrator.

Произведено сравнение существующих программных продуктов, аналогичных разрабатываемой ИС, позволяющее выявить наиболее значимые функции, а также определить главные достоинства и недостатки подобных систем.

Программный продукт реализован в полном объеме в соответствии с указанными в техническом задании требованиями. Компьютерная модель адаптивной обучающей игры успешно прошла тестирование.

В ходе выполнения работы по разработке компьютерной модели адаптивной обучающей игры были достигнуты все цели и выполнены требования, поставленные заказчиком, осуществлена автоматизация процесса организации самостоятельного обучения в учебном центре дополнительного образования «Вектор знаний».

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Васильев А. С# Объектно-ориентированное программирование : учебное пособие / А. Васильев. – Санкт-Петербург : Питер, 2012. – 320 с. – ISBN 978-5-459-01238-5.
2. Виртуальная реальность современного образования: идеи, результаты, оценки. Материалы VII Международной научно-практической интернет-конференции «Виртуальная реальность современного образования. VRME 2017»: сборник статей и тезисов; г. Москва, 2–6 октября 2017 г. / Московский педагогический государственный университет. Институт физики, технологии и информационных систем: под общ. ред. М. Е. Вайндорф-Сысоевой. – Москва: МПГУ, 2017. – 165с. – ISBN 978-5-4263-0554-0.
3. ГОСТ 19.503-79 – 1980. Издания. Единая система программной документации. Руководство системного программиста. Требования к содержанию и оформлению. – Москва: Стандартинформ, 2010. – 4 с.
4. Грекул В. И. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. – Москва : Издательство Юрайт, 2019. – 385 с. – ISBN 978-5-9916-8764-5.
5. Марченко А. Л. Основы программирования на С# : учебное пособие / А. Л. Марченко. – Москва : Бинوم. Лаборатория знаний, 2014. – URL: <http://www.iprbookshop.ru/67382> (дата обращения 7.11.2019). – Режим доступа: ЭБС «IPRbooks», по паролю.
6. Методология функционального моделирования IDEF0 : Руководящий документ. – Москва : ИПК Издательство стандартов, 2000. – 75 с.
7. Селетков С.Н. Мировые информационные ресурсы : учебное пособие / С. Н. Селетков, Н. В. Днепровская. – Москва : Евразийский

открытый институт, 2010. – URL: <http://www.iprbookshop.ru/10894> (дата обращения 23.03.2020). – Режим доступа: ЭБС «IPRbooks», по паролю.

8. Хокинг Дж. Unity в действии. Мультиплатформенная разработка на C# / Дж. Хокинг. – Санкт Петербург : Питер, 2019. – 352 с. – ISBN 978-5-4461-0816-9.

9. Что такое php : [сайт]. – 2020. – URL: <https://www.php.net> (дата обращения: 15.03.2020).

10. Nutaro J. J. Building Software for Simulation Theory and Algorithms, with Applications in C++ / J. Nutaro, New Jersey: John Wiley & Sons, 2011 . – 347 p. – ISBN 978-0-470-41469-9.

11. Sokolowski J. A. Modeling and simulation fundamentals : theoretical underpinnings and practical domains / J. A. Sokolowski, C. M. Banks. – New Jersey : John Wiley & Sons, 2009. – 437 p. – ISBN 978-0-470-48674-0.

12. Unity – Manual: Unity User Manual (2019.3) : [сайт]. – URL: <https://docs.unity3d.com/Manual/index.html> (дата обращения 05.04.18).