

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет

Кафедра

вычислительной техники

«Вычислительная техника»

Специальность: 09.05.01 «Применение и эксплуатация
автоматизированных систем специального
назначения»
Специализация: «Эксплуатация вычислительных машин, комплексов,
систем и сетей специального назначения»

ДИПЛОМНЫЙ ПРОЕКТ

на тему

**ПРОГРАММА ДЛЯ ПРОВЕДЕНИЯ
КВАЛИФИКАЦИОННОГО ТЕСТИРОВАНИЯ
СПЕЦИАЛИСТОВ**

Студент

(подпись, дата)

11.06.2024

Юткин Даниил Владимирович
(ФИО полностью)

Руководитель

(подпись, дата)

11.06.2024

Федюнин Р.Н.
(фамилия, инициалы)

Нормоконтролёр

(подпись, дата)

11.06.2024

Бычков А.С.
(фамилия, инициалы)

Рецензент

(должность, место работы)

Доцент ПензГТУ

(подпись, дата)

17.06.24

Петерский С.В.
(фамилия, инициалы)

Работа допущена к защите (протокол зас. кафедры от 27.05.2024 г. № 11)

Заведующий кафедрой

(подпись)

Митрохин М.А.
(фамилия, инициалы)

Работа защищена с отметкой

(протокол заседания ГЭК от 19.06.2024 № 6)

Секретарь ГЭК

(подпись)

Гурин Е.И.
(фамилия, инициалы)

Пенза 2024

Федеральное государственное бюджетное образовательное учреждение
высшего образования
ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ

Факультет

Вычислительной техники

Кафедра

Вычислительная техника

«Утверждаю»
Заведующий кафедрой
Митрохин М.А.
«16» апреля 2024 г.

ЗАДАНИЕ НА ДИПЛОМНЫЙ ПРОЕКТ

1. Студент Юткин Даниил Владимирович гр. 19ВОЭ1 факультета вычислительной техники.
2. Тема работы: «Программа для проведения квалификационного тестирования специалистов».
Тема утверждена приказом ПГУ № _____ от _____. 2024 г.
3. Руководитель работы к.т.н., доцент каф. ВТ «ПГУ» Федюнин Р.Н.
4. Задание на работу (назначение разработки, исходные данные и т.п.).

Разработать программу для тестирования специалистов по направлению «Применение и эксплуатация автоматизированных систем специального назначения». Программное обеспечение должно выполнять следующие функции:

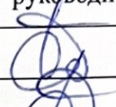
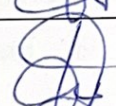
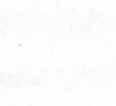
- представить многопользовательский вход в систему, предоставить возможность новому пользователю регистрации в системе,
- пользователю с правами администратора предоставить возможность:
- создавать новые тестовые задания и редактировать уже имеющиеся в системе;
- создавать новых пользователей и редактировать уже имеющихся в системе;
- пользователю с правами пользователя предоставить возможность:
- прохождения выбранных тестов;
- хранить в базе данных и отображать на экран результаты пройденных пользователями тестов; выполнять оценку результатов пройденного тестирования,
- редактировать данные собственного профиля.

Программа для тестирования должна разрабатываться с помощью среды разработки *Visual Studio Code*. База данных должна разрабатываться для реляционной БД.

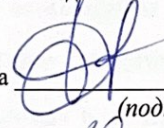
5. Перечень подлежащих разработке вопросов.

Требуется произвести анализ предметной области, определить функциональный состав программы, выбрать инструменты реализации программы и развертывания приложения в облачной среде исполнения. Необходимо разработать алгоритм работы программы, разработать *UML* диаграммы с последующим написанием программы.

6. Календарный график выполнения работы

№ п/п	Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя
1.	Анализ предметной области	10	19.04-24.04	
2.	Проектирование программы	20	25.04-30.04	
3.	Разработка базы данных	10	01.05-06.05	
4.	Отладка и тестирование	10	07.05-13.05	
5.	Разработка экономической и БЖД части	15	14.05.2024- 28.05.2024	
6.	Оформление графической части	10	29.05-01.06	
7.	Оформление выпускной работы	30	02.06-05.06	

Дата выдачи задания « » 2024 года.

Руководитель дипломного проекта  Федюнин Р.Н.
(подпись, дата)

Задание к исполнению принял студент  Юткин Д.В.
(подпись, дата)

Работу к защите допустить
Декан факультета  Фионова Л.Р.
(подпись, дата)
17.06.2024

Реферат

Пояснительная записка содержит 127 листов, 32 рисунков, 19 таблиц и 3 приложения.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, ОНЛАЙН ТЕСТИРОВАНИЕ, БАЗА ДАННЫХ, ИТ СПЕЦИАЛИСТ

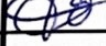
Объектом исследования является тестирование специалистов в области информационных технологий для последующей подготовки в рамках мероприятий повышения квалификации.

Цель работы – разработка программы для квалификационного тестирования специалистов.

В процессе работы необходимо выполнить анализ предметной области, сформулировать техническое задание для разрабатываемой программы, реализовать модель базы данных и программное обеспечение.

В результате проделанной работы разработана программа для интерактивного тестирования, состоящая из WEB-приложения и базы данных.

База данных спроектирована с использованием WEB-средства phpAdmin. База данных разработана для СУБД MariaDB, приложение реализовано в среде разработки Visual Studio Code.

					ПГУ 1.090501.08.001 ПЗ		
Изм.	Лист	№ докум.	Подпись	Дата			
Разраб.		Юткин Д.В.					
Провер.		Федюнин Р.Н.					
Н. контр.		Бычков А.С.					
Утв.		Митрохин М.А.					
					Программа для квалификационного тестирования специалистов		
					Пояснительная записка		
					Лит.	Лист	Листов
						2	127
					ФВТ, каф. ВТ		
					гр.19В031		

Содержание

Перечень сокращений и обозначений.	5
Введение.	6
1. Анализ предметной области.	7
1.1 Описание предметной области.	7
1.2 Анализ существующих аналогов.	10
1.3 Цели и задачи разрабатываемой программы.	21
1.4 Анализ целевой аудитории.	22
2. Техническое задание.	22
2.1 Назначение разработки.	22
2.2 Требования к программе.	22
2.3 Требования к программной документации.	23
2.4 Стадии и этапы разработки.	24
2.5 Порядок контроля и приемки.	25
3. Обоснование выбора инструментов разработки.	26
3.1 Редактор исходного кода и IDE для разработки.	26
3.2 Выбор языка программирования.	29
3.3 Иные инструменты для разработки и отладки приложения.	32
3.4 Выбор системы управления баз данных.	37
3.5 CASE инструмент моделирования данных, процессов и UML.	39
4. Разработка программы для квалификационного тестирования.	41
специалистов	
4.1 Проектирование базы данных	43
4.2 Реализация программного приложения	50
5 Технология создания базы данных программы для интерактивного. ...	65
тестирования IT специалистов	
6 Оценка затрат на разработку и внедрение программы для	69
интерактивного тестирования IT специалистов	

7 Общие принципы обеспечения безопасности на рабочем месте.	79
оператора программного приложения для интерактивного тестирования ИТ специалистов	
Заключение.	85
Список использованных источников.	86
Приложение А UML-диаграммы.	87
Приложение Б Листинг программы.	93
Приложение В Презентация.	113

Перечень сокращений и обозначений

В настоящей работе применяют следующие сокращения и обозначения:

БД – база данных

ГОСТ – государственный стандарт

ОС – операционная система

НФ – нормальная форма

ПЭВМ – персональная электронно-вычислительная машина

ЭВМ – электронно-вычислительная машина

FK (Foreign Key) – внешний ключ

PK (Primary Key) первичный ключ

SQL (Structured Query Language) – язык структурированных запросов

UML (Unified Modeling Language) – унифицированный язык моделирования

					<i>ПГУ 1.090501.08.001 ПЗ</i>	Лист
						5
Изм.	Лист	№ докум.	Подпись	Дата		

Введение

Программы для онлайн-тестирования необходимы в любом учреждении для различных целей, включая четыре основных задачи.

Оценка уровня компетенций – программы тестирования помогают HR-специалистам определить уровень профессиональной компетентности работников, что позволяет принимать обоснованные решения о продвижении, обучении или увольнении сотрудников.

Мониторинг и контроль работы сотрудников. Программы мониторинга и контроля работы сотрудников помогают руководителям отслеживать активность работников, оценить эффективность их работы и принимать меры по улучшению производительности.

Аттестация и обучение - онлайн-тестирование может быть использовано для аттестации работников, определения необходимости в дополнительном обучении или профессиональном развитии.

Управление персоналом - программы помогают HR-отделам управлять персоналом более эффективно, отслеживать результаты работы, принимать обоснованные решения о кадровых изменениях и развивать стратегии развития компании.

В целом, программы для онлайн-тестирования работников помогают улучшить эффективность работы, повысить уровень компетенций сотрудников, оптимизировать процесс управления персоналом и принимать обоснованные решения о развитии компании.

В данном дипломном проекте реализовано программное обеспечение для тестирования специалистов в области информационных технологий, с целью повышения квалификации кадров предприятия.

Реализованное программное обеспечение имеет клиент серверную архитектуру.

1 Анализ предметной области

1.1 Описание предметной области

Рынок программного обеспечения для тестирования и мониторинга сотрудников - это растущая отрасль (рисунок 1)[1], обусловленная необходимостью компаний обеспечивать качество рабочего процесса своих сотрудников, а также необходимость контролировать рабочие процессы сотрудников в рамках команд и отделов.

ДОХОД ПО СЕГМЕНТАМ

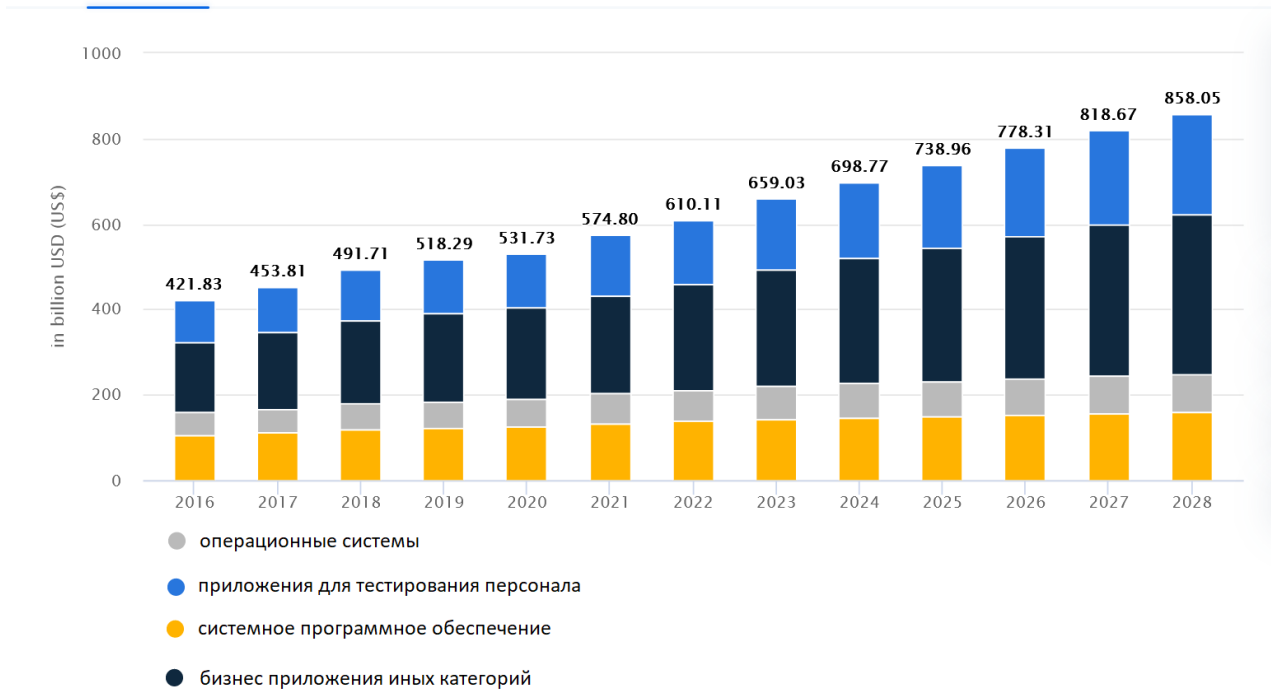


Рисунок 1 – Доли рынка мирового программного обеспечения [1]

Ниже приведены ключевые моменты о рынке программного обеспечения для тестирования персонала:

Размер рынка. Ожидается, что к 2029 году рынок программного обеспечения тестирования и мониторинга сотрудников достигнет 173 миллионов долларов США, а среднегодовой темп роста (CAGR) составит около 75% относительно в год [2].

Типы компаний, использующие программное обеспечение для тестирования и мониторинга сотрудников. Рынок включает в себя различные типы компаний, особенно компаний, занимающихся разработкой программного обеспечения, ИТ-услугами и разработкой продуктов. Эти компании используют инструменты тестирования и мониторинга сотрудников, чтобы гарантировать компетенции своих сотрудников и для контроля производительности своих сотрудников.

Драйверы роста. Рынок движим растущей потребностью компаний обеспечивать качество и безопасность своих программных продуктов и/или услуг, а также необходимостью контролировать производительность своих сотрудников [2]. Пандемия COVID-19 ускорила рост рынка, поскольку компаниям пришлось адаптироваться к удаленной работе и обеспечить компетенции своих сотрудников, работая удаленно [3].

На рынке присутствуют различные ключевые игроки, такие как VeralTest Professional, iSpring, SunRav Web Class, StartExam, все они проводят исследования и разработки в области тестирования и мониторинга персонала.

Проблемы: рынок сталкивается с такими проблемами, как нехватка отечественного свободно распространяемого программного обеспечения для тестирования ИТ персонала. Также большинство решений являются облачными продуктами, зависящими от провайдеров и вендоров ПО, это приводит к потребности в небольших локальных программных продуктах для тестирования персонала. Основная задача данного ПО - являться подручным средством для отдела кадров и технического лидера ИТ команды, чтобы быстро оценить компетенцию нового сотрудника или устроить блиц QUIZ сотрудникам на митапе.

Вывод: Рынок программного обеспечения тестирования и мониторинга сотрудников это перспективная отрасль, движимая необходимостью компаний обеспечивать должный уровень компетенций своих работников. Ожидается, что потребность в программном обеспечении и тестах продолжит расти благодаря растущему спросу на специалистов в ИТ отрасли Российской Федерации [3].

Одним из эффективных решений программы для тестирования будет реализация программы для проведения тестирования IT-специалистов в виде WEB-приложения.

WEB-приложения могут быть доступны в любое время и из любой локации с использованием любого устройства с доступом в Интернет. Это удобно для IT-специалистов, которые могут проходить тесты из офиса, дома или даже в пути, используя свой смартфон или ноутбук.

WEB-приложения могут легко масштабироваться и обрабатывать большое количество пользователей. Это позволяет эффективно тестировать множество IT-специалистов различных направлений и разного уровня подготовки.

Таким образом, WEB-приложения для проведения тестирования IT-специалистов обеспечивают удобство, эффективность и точность при проведении тестов, что делает их хорошим выбором для оценки компетенции и навыков сотрудников организации.

На рисунке 2 представлена диаграмма вариантов использования. На диаграмме вариантов использования представлены две сущности: администратор и пользователь.

В функционал администратора входит:

- авторизация;
- управление вопросами теста: создание, удаление, редактирование;
- управление пользователями: создание, удаление, редактирование;
- формирование и отправка пользователю обратной связи по тесту.

В функционал пользователя входит:

- создание аккаунта (регистрация в системе);
- авторизация в системе;
- выбор теста;
- прохождение теста;
- просмотр результата тестирования;
- возможность пройти тест вновь.

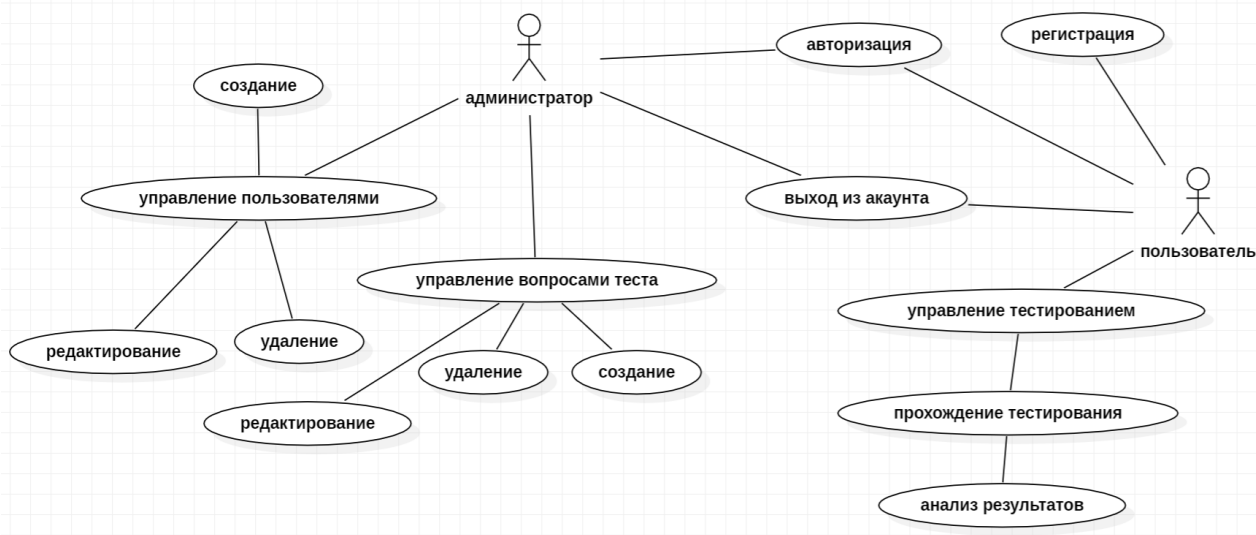


Рисунок 2 – Диаграмма вариантов использования

Пользователь на представленной диаграмме — это IT-специалист.

Администратор - сотрудник отдела кадров или технический специалист, задача которого оценить квалификацию тестируемого.

При первом использовании программы сначала необходимо осуществить регистрацию аккаунта тестируемого сотрудника, при следующей попытке входа в систему повторная регистрация тестируемого не требуется.

Когда пользователь авторизуется и входит в систему, ему предоставляется возможность выбрать тест, пройти тестирование, посмотреть свои достижения и пройти тест заново.

Когда администратор авторизуется и входит в систему, ему предоставляется возможность управления пользователями и тестами. Создание теста включает в себя оставление и добавление вопросов, а также вариантов ответов на каждый вопрос. Также администратор имеет доступ к информации о пользователях тестов и возможность обратной связи с ними.

1.2 Анализ существующих аналогов

Популярное программное обеспечение для интерактивного тестирования IT-специалистов включает в себя:

StartExam - система тестирования, которая позволяет разрабатывать тесты, проводить промежуточное и итоговое тестирование, осуществлять подготовку к экзаменам и анализировать результаты.

VeralTest Professional - комплект инструментов для создания тестов и проведения тестирования, включающий в себя редактор тестов, приложение для отображения результатов тестирования и систему администрирования.

Google Forms - инструмент для создания форм и опросов, который может быть использован для онлайн тестирования сотрудников.

Microsoft Forms - инструмент для создания форм и опросов, который может быть использован для онлайн тестирования сотрудников.

Kahoot - инструмент для создания интерактивных тестов и опросов, который может быть использован для онлайн тестирования сотрудников.

Кроме того, для тестирования также используются различные инструменты и платформы, такие как Quizlet, Quizizz, ProProfs Quiz Maker и иное программное обеспечение, которое предлагает широкий спектр функций для создания и проведения тестов IT-специалистов.

Ниже приведено подробное описание популярных решений тестирования IT-специалистов.

1.2.1 Программное обеспечение «StartExam»

StartExam - это программное обеспечение, предназначенное для тестирования сотрудников и оценки их знаний и навыков. StartExam предлагает ряд функций, которые помогут организациям масштабировать свои потребности в тестировании, включая возможность обработки большого количества сеансов тестирования, 12 встроенных отчетов и возможность создания пользовательских отчетов.

Ключевая особенность:

- Масштабируемость. StartExam может обрабатывать большое количество тестовых сессий, что делает его подходящим для организаций любого размера.

- Отчеты: Программное обеспечение предлагает 12 встроенных отчетов, и пользователи также могут запрашивать специальные отчеты для удовлетворения своих конкретных потребностей.
- Создание тестов: пользователи могут создавать тесты и сдавать экзамены для оценки знаний и навыков сотрудников.
- Отслеживание прогресса компетенций: StartExam позволяет организациям отслеживать прогресс сотрудников и определять области для улучшения их компетенций.

Преимущества:

- Простота в использовании: StartExam удобен для пользователей и администраторов, что позволяет организациям легко создавать и проводить тесты.
- Масштабируемость: программное обеспечение может обрабатывать большое количество тестовых сеансов, что делает его подходящим для организаций любого размера.
- Настраиваемость: пользователи могут запрашивать специальные отчеты в соответствии со своими конкретными потребностями.
- Экономичность: StartExam может помочь организациям сократить расходы на тестирование и обучение сотрудников.

Недостатки:

- Платное решение.
- Отсутствие отраслевых функций. Хотя StartExam является программным обеспечением для тестирования общего назначения, оно может не иметь функций, специфичных для определенных отраслей, таких как различные направления ИТ.

Вывод, StartExam - это платный инструмент с закрытым исходным кодом для организаций, желающих протестировать своих сотрудников и оценить их знания и навыки.

На рисунке 3 представлена главная страница платформы StartExam.

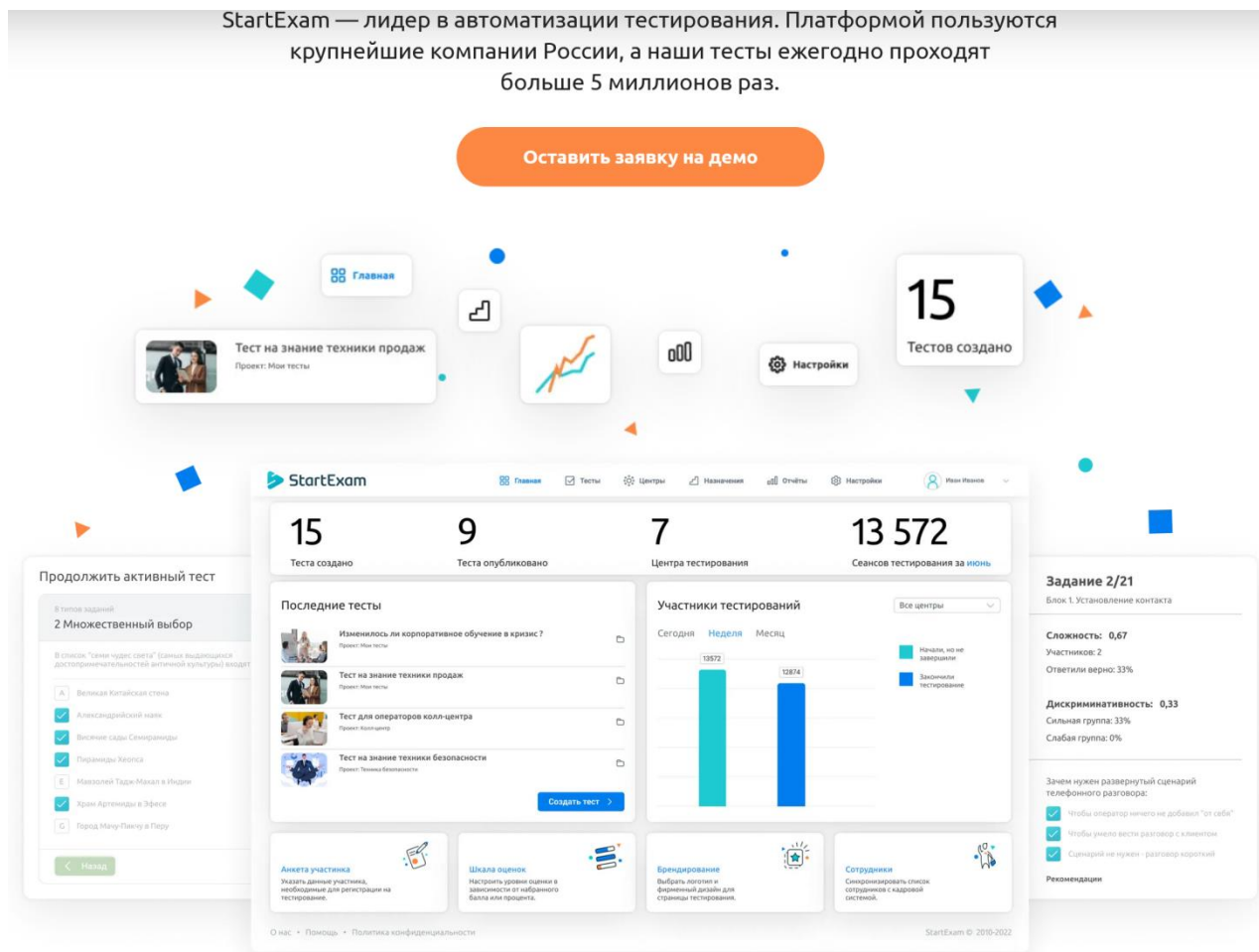


Рисунок 3 - Платформа «StartExam»

1.2.2 Программный инструмент «VeralTest Professional»

VeralTest Professional - это программный инструмент, предназначенный для помощи организациям в создании и администрировании тестов и оценок для своих сотрудников или студентов. Программное обеспечение предлагает такие функции, как:

- Создание тестов. Пользователи могут создавать тесты и викторины с несколькими типами вопросов, включая вопросы с множественным выбором, «верно/неверно» и открытые вопросы.
- Администрирование тестов: Программное обеспечение позволяет пользователям проводить тесты онлайн, что позволяет легко охватить большое количество тестируемых.

- Выставление оценок и отчетность: VeralTest Professional может предлагать функции автоматического выставления оценок и создания отчетов, что позволяет легко отслеживать работу тестируемых и определять области, требующие дообучения.

Преимущества:

- Удобство: VeralTest Professional позволяет создавать и администрировать тесты в режиме онлайн, уменьшая необходимость в бумажном тестировании.
- Масштабируемость: программное обеспечение является подходящим для организаций любого размера.
- Экономия времени: функции автоматической оценки и отчетности снижают административную нагрузку при тестировании.

Недостатки:

- Платная лицензия. Для использования программы требуется приобрести платную лицензию, что может быть неудобно для некоторых пользователей.
- Ограниченность бесплатной версии. Бесплатная версия VeralTest Professional имеет ограниченный функционал по сравнению с платной версией.
- Необходимость установки. Для работы с программой требуется установка на компьютер, в отличие от WEB онлайн-инструментов тестирования.
- Требовательность к ресурсам. Во время запуска тестов VeralTest Professional потребляет достаточно большие системные ресурсы, что может сказаться на производительности компьютера.

Главная страница данного программного инструмента представлена на рисунке 4.

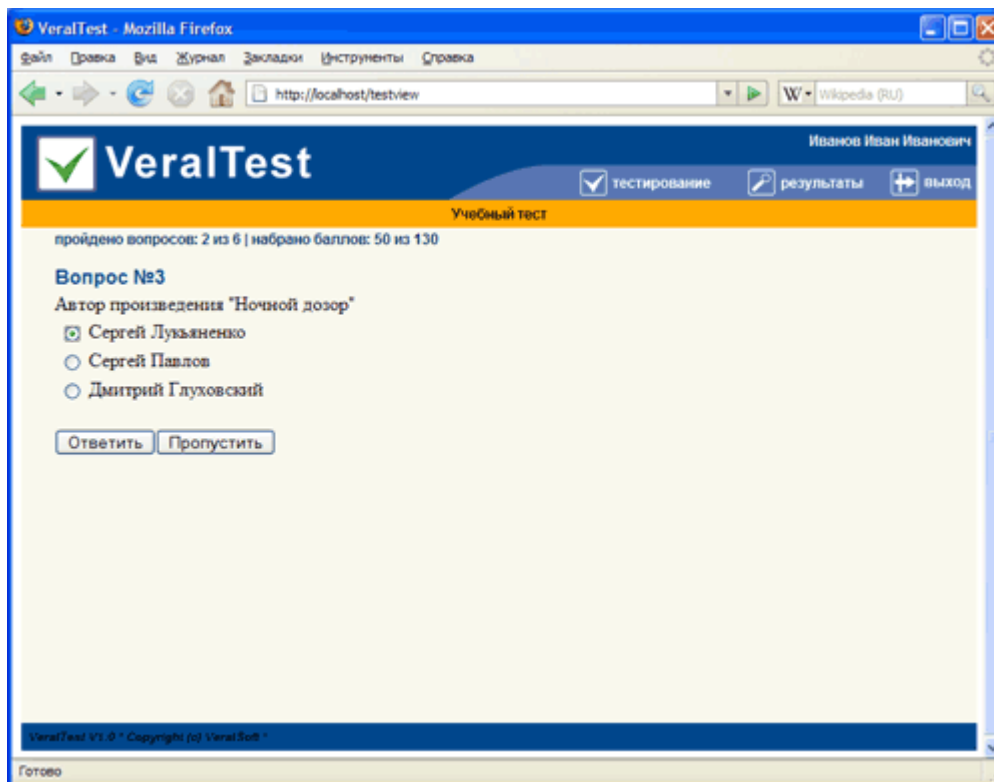


Рисунок 4 - Программный инструмент «VeralTest Professional»

1.2.3 WEB-приложение «Google Forms»

Google Forms - это бесплатное веб-приложение, предлагаемое Google в рамках службы Google Drive, которое позволяет пользователям создавать и редактировать формы, опросы и викторины онлайн.

Ключевая особенность:

- Создание форм. Пользователи могут создавать формы с различными типами вопросов, включая вопросы с множественным выбором, короткими ответами, абзацами и вопросами с флажками.
- Создание опросов и викторин. Google Forms можно использовать для создания опросов и викторин, что делает их универсальным инструментом для сбора и оценки данных.
- Совместная работа: несколько пользователей могут совместно создавать и редактировать формы, что упрощает работу с другими.
- Интеграция с Google Sheets. Ответы на формы можно автоматически сохранять в Google Sheets, что упрощает анализ и отслеживание данных.

Преимущества:

- Простота в использовании: Google Forms удобен для пользователя и прост в навигации даже для тех, у кого нет обширных технических знаний.
- Сотрудничество. Возможность совместной работы над тестами в режиме реального времени упрощает работу над формами и опросами в команде.
- Интеграция с Google Sheets. Автоматическое сохранение ответов форм в Google Sheets упрощает анализ и отслеживание данных.
- Бесплатно: Google Forms - это бесплатная служба, что делает ее доступной для частных лиц и организаций.

Недостатки:

- Ограниченная настройка. Хотя Google Forms предлагает различные типы и темы вопросов, они могут не обеспечивать тот уровень настройки, который необходим некоторым пользователям.
- Ограниченная аналитика. Хотя ответы форм можно сохранить в таблице Google, возможности аналитики ограничены по сравнению с инструментами описанными выше.
- Зависимость от сервисов Google. Google Forms является частью экосистемы Google, поэтому пользователи должны иметь учетную запись Google и понимать методы сбора данных компанией.

Главная страница данной платформы представлена на рисунке 5.

Рисунок 5 – Собеседование кандидатов с платформой «Google Forms»

1.2.4 Онлайн-платформа «Kahoot»

Kahoot - это интерактивная обучающая платформа, предлагающая ряд инструментов и функций для повышения уровня компетенций сотрудников и студентов. Ниже краткое описание Kahoot, включая его преимущества и недостатки.

Преимущества:

- Интерактивное обучение: Kahoot обеспечивает игровую среду обучения, которая привлекает учащихся и поощряет активное участие.

- Доступность: доступ к Kahoot можно получить с любого устройства, подключенного к Интернету, что делает его удобным инструментом для дистанционного обучения.
- Настройка: преподаватели могут создавать собственные викторины, опросы и обсуждения, адаптированные к их конкретной учебной программе и стилю преподавания.
- Обратная связь в реальном времени: Kahoot обеспечивает онлайн обратную связь и результаты, позволяя преподавателям оценивать понимание студентами и соответствующим образом корректировать свои инструкции.
- Сотрудничество: Kahoot позволяет студентам вместе работать над викторинами и дискуссиями, способствуя командной работе и сотрудничеству.

Недостатки:

- Ограниченное содержание: хотя Kahoot предлагает широкий выбор готовых викторин и игр, их содержание не всегда может соответствовать конкретным потребностям конкретной потребности или учебной программы.
- Зависимость от технологий: Kahoot требует стабильного подключения к Интернету и функциональных устройств, что может стать проблемой в регионах с ограниченным доступом к технологиям.
- Платная лицензия. Для использования программы требуется приобрести платную лицензию, что может быть неудобно для некоторых пользователей.

Заключение:

Kahoot — ценный инструмент для преподавателей, стремящихся вовлечь студентов в интерактивный процесс обучения.

Пример страницы платформы представлена на рисунке 6.

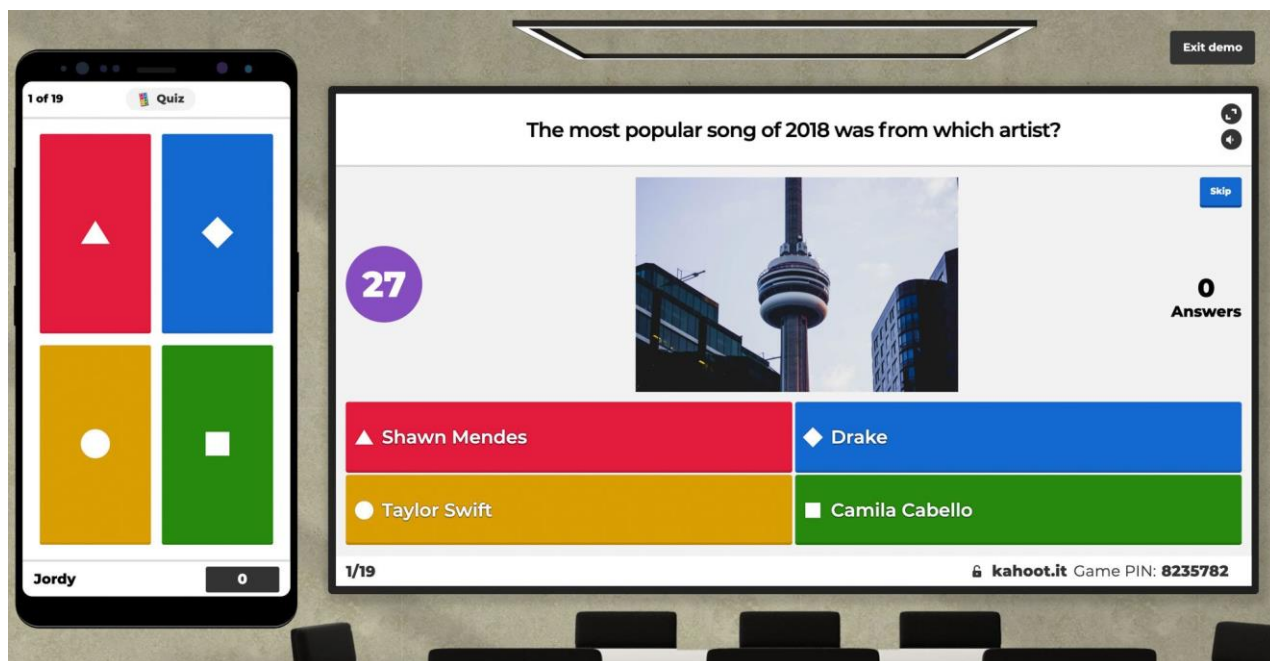


Рисунок 6 – Платформа «Kahoot»

1.2.5 Платформа «Quizlet»

Quizlet - популярная платформа онлайн-обучения, предлагающая ряд инструментов и функций, помогающих учащимся учиться и учиться более эффективно. Вот подробное описание Quizlet, включая его преимущества и недостатки.

Преимущества:

- Интерактивное обучение. Quizlet предоставляет различные интерактивные инструменты, такие как карточки, тесты и обучающие игры, которые делают обучение более увлекательным и увлекательным.
- Настройка: пользователи могут создавать свои собственные учебные наборы с учетом их конкретных потребностей и стиля обучения.
- Доступность: доступ к Quizlet можно получить с любого устройства, подключенного к Интернету, что делает его удобным инструментом для дистанционного обучения.

- Большая библиотека контента: Quizlet располагает обширной библиотекой пользовательского контента, охватывающего широкий спектр предметов и тем.
- Сотрудничество: Quizlet позволяет пользователям обмениваться информацией и учиться вместе с другими, способствуя сотрудничеству и командной работе.

Недостатки:

- Качество пользовательского контента. Несмотря на то, что Quizlet имеет большую библиотеку контента, качество стандартных учебных наборов может значительно различаться.
- Ограниченная глубина содержания. Сосредоточение внимания Quizlet на карточках и вопросах с краткими ответами может не обеспечить такой же уровень глубины и понимания, как более полные учебные материалы и мультитесты.
- Платная лицензия. Для использования программы требуется приобрести платную лицензию, что может быть неудобно для некоторых пользователей.

Пример страницы платформы представлена на рисунке 7.

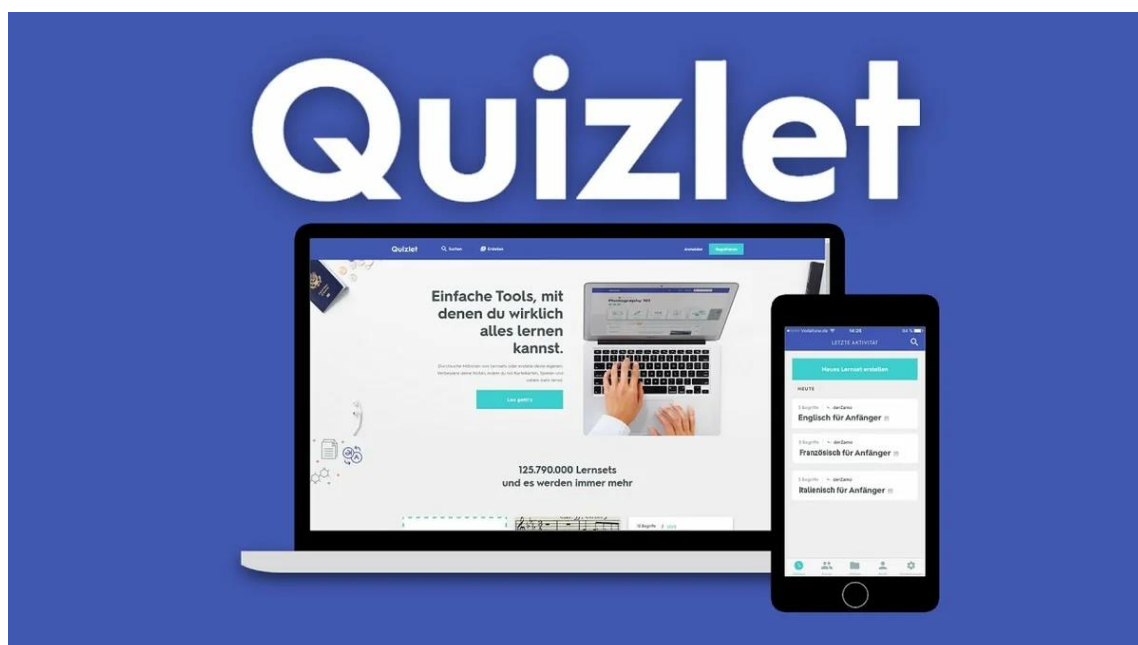


Рисунок 7 – Платформа «Quizlet»

Заключение:

Quizlet - ценный инструмент для студентов, которые хотят дополнить свое обучение и улучшить свои учебные привычки. Хотя у него есть свои ограничения, преимущества использования Quizlet, в том числе повышение вовлеченности и доступности, делают его выгодной инвестицией для многих студентов.

Программное обеспечение, реализованное в дипломном проекте, полностью в отличие от описанных выше программ является бесплатным, полностью настраиваемым со стороны потребителя, включая тесты и вопросы к ним.

1.3 Цели и задачи разрабатываемой программы

Цель – проведение объектного экспресс тестирования IT-специалистов организации на всех этапах их трудовой карьеры.

Задачами данного проекта являются:

- регистрация и авторизация пользователей в системе;
- возможность формирования тестов;
- проведение пользователям тестирования;
- фиксация результатов проведения тестирования;
- возможность просмотра результатов тестирования;
- формирование обратной связи для тестируемого.

Цели реализации программы – экспресс оценка знаний и обучение сотрудников и кандидатов IT организации. В результате внедрение по планируется достичь следующее:

- повысить скорость обучения и отбора потенциальных кандидатов IT-компании;
- улучшить и облегчить функционирование отдела кадров и профильных отделов ;
- получить список потенциальных кандидатов на повышение или занятие IT-позиции в организации.

1.4 Анализ целевой аудитории

Целевая аудитория программного обеспечения – IT организации малого и среднего уровня

Разработка данного вида программного приложения актуальна как для частных, так и для государственных компаний, у которых есть штат специалистов по информационным технологиям.

2 Техническое задание

2.1 Назначение разработки

Программа для интерактивного тестирования IT-специалистов предназначена для работы сотрудников отдела кадров и контроля квалификации сотрудников предприятия.

Цель данного проекта – разработка базы данных и WEB-приложения для тестирования IT-специалистов непосредственно на предприятии.

2.2 Требования к программе

2.2.1 Требования к функциональным характеристикам

Программа для интерактивного тестирования IT специалистов должна обеспечивать выполнение следующих функций:

Программа для интерактивного должна обеспечивать следующие функции:

- регистрация пользователей;
- прохождения и сохранение выбранного теста;
- регистрация администратора;
- создание новых тестов для пользователей;
- просмотр результатов тестирования;
- возможности редактирования и удаления существующих тестов.

2.2.2 Требования к надежности

Программа должна работать без прерываний и ошибок. В случае, если пользователь совершает неверные действия, система должна сообщить ему об этом.

Программа должна обеспечивать контроль вводимых данных. Если были введены некорректные данные или данные, являющиеся обязательными, не были введены, система должна информировать об этом пользователя.

При вводе входных данных должен осуществляться ее контроль на допустимость вводимых значений по следующим правилам:

- вводимые данные не должны противоречить заявленному типу данных;
- вводимые данные не должны противоречить законам логики.

Конечный пользователь не должен иметь доступ к исходным данным приложения. Для модификации программной части системы пользователю необходимо обратиться к разработчику данной программы.

2.2.3 Требования к параметрам технических средств

Разрабатываемое приложение предназначено для работы на персональных компьютерах специалистов по тестированию, с минимальная конфигурация технических средств:

- ОС Windows 10 и выше;
- процессор архитектуры x86 INTEL 11 поколения или его аналог;
- свободное место на жестком диске 1 ГБ и выше;
- оперативная память 8 ГБ и выше;
- монитор с диагональю не менее 27 дюймов;
- периферия: мышь, клавиатура, МФУ или принтер.

2.3 Требования к программной документации

Программа для интерактивного тестирования IT специалистов должна сопровождаться документацией, содержащей:

- техническое задание;

- описание проектирования БД;
- описание применения программы.

2.4 Стадии и этапы разработки

Стадии и этапы разработки программы для интерактивного тестирования IT специалистов представлены в таблице 1.

Таблица 1 - Стадии и этапы разработки

Стадии	Этапы	Объем работы в процентах	Срок выполнения	Исполнитель
1 Анализ предметной области	Анализ предметной области	10		Юткин Д. В.
2 Проектирование программы	Логическое и физическое проектирование БД	20		Юткин Д. В.
3 Разработка базы данных	Разработка базы данных	10		Юткин Д. В.
4 Отладка и тестирование	Отладка и тестирование	10		Юткин Д. В.
5 Разработка экономической части	Оценка затрат на разработку и внедрение программы	10		Юткин Д. В.
6 Разработка раздела БЖД	Оценка тяжести труда эксплуататора программы	5		Юткин Д. В.

Стадии	Этапы	Объем работы в процентах	Срок выполнения	Исполнитель
7 Оформление графической части	Оформление графической части	10		Юткин Д. В.
8 Оформление выпускной работы	Оформление пояснительной записки	30		Юткин Д. В.

2.5 Порядок контроля и приемки

Для проверки выполнения программой всех ее функций в соответствии с требованием технического задания необходимо разработать тестовые наборы данных. Необходимо протестировать действия программы на возможные реакции пользователя: запуск программы, заполнение полей ввода, выбор пунктов меню, нажатие кнопок, работа с базой данных. Приложение соответствует требованиям технического задания, если оно выполняет все заданные функции при различных входных данных из тестовых наборов данных.

Проверка и приемка программы должна осуществляться при ее работоспособности и в присутствии заказчика, выполнения всех возложенных на нее функциональных возможностей, а также при наличии полной документации на программу (пояснительная записка с приложениями: текстом программы и результатом текстовых примеров).

3 Обоснование выбора инструментов разработки

На основании анализа требований к характеристикам программы для интерактивного тестирования IT специалистов был выбран стек разработки.

3.1 Редактор исходного кода и IDE для разработки

В качестве редактора исходного кода для разработки настоящей программы используется среда Visual Studio Code.

Visual Studio Code (VS Code) - кроссплатформенный текстовый редактор, разработанный Microsoft. VS Code позиционируется как «легкий» редактор кода для кроссплатформенной разработки WEB-приложений. VS Code включает в себя отладчик, инструменты для работы с репозиториями и планигами, подсветку синтаксиса, IntelliSense и инструменты рефакторинга.

Он имеет широкие возможности настройки: пользовательские темы, комбинации клавиш и файлы конфигурации.

Альтернативы использования Visual Studio Code:

1. VSCodium: редактор с открытым исходным кодом без проприетарных компонентов Microsoft. Это надежный текстовый редактор, который можно настроить с помощью плагинов и который доступен для Linux, Windows и macOS.
2. Brackets: текстовый редактор и интегрированная среда разработки, предназначенные для веб-разработчиков, с поддержкой HTML, CSS, JavaScript, PHP и Python. Он имеет богатую экосистему расширений и традиционный интерфейс.
3. Che: облачная IDE, созданная для облачной разработки, без каких-либо предположений о локальной файловой системе. IDE имеет открытый исходный код и может работать как программное обеспечение как услуга (SaaS) или на экземпляре контейнера виртуализации.

4. Replit : полноценная среда разработки в браузере, включающая все от линтеров и отладчиков до сторонних пакетов, хостинга и развертывания. Предлагает совместную работу в режиме реального времени, выбор цвета, темный режим и прочие настройки для разработки программного обеспечения.
5. Textastic: расширенный редактор кода для iOS и macOS с поддержкой подсветки синтаксиса, удаленной передачи файлов и SSH.
6. Runestone : производительный, легкий текстовый редактор с подсветкой синтаксиса, номерами строк, поиском и заменой, руководством по страницам и многим другим.

Каждая из этих альтернатив предлагает уникальные функции и возможности, поэтому стоит изучить их, чтобы найти тот, который лучше всего соответствует вашим потребностям.

Таблица 2 - Сравнение функционала аналогов Visual Studio Code.

Наименование	Преимущества	Недостатки
Visual Studio Code	<p>Широко используемый и легко настраиваемый редактор.</p> <p>В экосистему вносит свой вклад большое сообщество разработчиков.</p> <p>Имеет отличную поддержку широкого спектра языков программирования, включая JavaScript, PHP и многие другие.</p>	<p>Некоторые пользователи обеспокоены запатентованным характером VS Code, особенно в отношении телеметрии и возможности Microsoft внедрить проприетарные функции, которые могут не иметь открытого исходного кода.</p>

Наименование	Преимущества	Недостатки
	Имеет богатый набор расширений, которые могут улучшить его функционал	
VSCodium	Свободно распространяемая версия VS Code, которая не содержит проприетарных компонентов. Надежный текстовый редактор, который можно настроить с помощью плагинов и который доступен для Linux, Windows и macOS.	VSCodium не имеет такого же уровня поддержки сообщества, как VS Code, что влияет на его развитие и доступность расширений.
Brackets	Интегрированная среда разработки, предназначенная для веб-разработчиков, с мощной поддержкой HTML, CSS, JavaScript, PHP и Python. IDE имеет богатую экосистему расширений и традиционный интерфейс.	IDE не так широко использоваться как VS Code, что может ограничивать его функциональность и доступность документации для расширений и плагинов.
Replit	Полноценная облачная среда совместной разработки в браузере, включающая инструментарий: от линтеров и отладчиков до сторонних пакетов, хостинга и	Облачный сервис, который может вызывать опасения по поводу конфиденциальности и безопасности данных. Для эффективной работы также требуется

Наименование	Преимущества	Недостатки
	развертывания. IDE доступен для iPad и предлагает совместную работу в режиме реального времени, выбор цвета, темный режим и многое другое.	квалификация и наличие документации.

В заключение сравнительного анализа отметим, что каждый из этих редакторов кода имеет свои сильные и слабые стороны. VS Code выбран в связи с тем, что это IDE с широкими возможностями настройки, имеющий большое сообщество разработчиков, VS Code имеет собственные компоненты, которые совместимы с дебагерами, инструментами, управления версиями, и сборками веб-сервера.

3.2 Выбор языка программирование

В связи с тем что программное обеспечение, реализованное в рамках дипломного проекта является WEB-сайтом. Необходимо выбрать эффективный инструмент реализации проекта. Проведем сравнительный анализ популярных языков программирования для WEB [4]: PHP, Python и ASP.NET.

Сравним PHP, Python и ASP.NET по ключевым критериям:

1. Область применения:

- PHP хорошо подходит для разработки динамических веб-сайтов и веб-приложений.
- Python используется в более широком спектре задач: веб-разработка, аналитика данных, машинное обучение, автоматизация и т.д.
- ASP.NET предназначен для разработки веб-приложений, настольных программ и мобильных приложений в экосистеме Microsoft.

2. Скорость разработки и прототипирования:

- PHP позволяет быстро создавать простые веб-приложения.
- Python также отличается высокой скоростью разработки за счет лаконичного синтаксиса и встроенных библиотек.
- ASP.NET требует больше времени на изучение и настройку окружения, но предлагает более масштабируемые и сложные решения.

3. Производительность:

- PHP и Python в целом уступают в производительности статически типизированным языкам, таким как ASP.NET.
- ASP.NET приложения, как правило, отличаются более высокой производительностью, особенно при обработке многопоточных задач.

4. Сообщество и экосистема:

- PHP имеет очень большое и активное сообщество разработчиков, а также обширную экосистему библиотек и фреймворков.
- Python также обладает широким сообществом и множеством библиотек, особенно для научных вычислений и машинного обучения.
- ASP.NET тесно интегрирован с экосистемой Microsoft, что обеспечивает хорошую поддержку и инструментарий, но ограничено рамками этой платформы.

Таким образом на основании сравнительного анализа, PHP выбран как инструмент для быстрой разработки веб-приложения, без привязки к экосистеме Microsoft или иной технологической платформе. При этом PHP в Российской Федерации является распространенным инструментом разработки с широким рынком разработчиков и специалистов технической поддержки.

Помимо PHP, для реализации интерфейсов в рамках дипломного проекта потребуются инструменты языков HTML и CSS.

Преимущества HTML и CSS по сравнению с XHTML или Markdown [5]:

1. Разделение структуры и оформления страниц приложения:

- HTML отвечает за структуру и содержание, а CSS за внешний вид и стилевое оформление.

- связка HTML и CSS позволяет четко разделять функции и упрощает разработку и поддержку интерфейса приложения.

2. Семантичность:

- HTML-теги несут смысловую нагрузку, описывая структуру документа, что улучшает доступность, поисковую оптимизацию и читаемость кода.

3. Кроссбраузерность:

- HTML и CSS являются стандартизированными технологиями, поддерживаемыми всеми современными браузерами, что обеспечивает совместимость и единообразное отображение на различных устройствах.

4. Гибкость и адаптивность:

- CSS предоставляет широкие возможности для адаптации дизайна под разные экраны и устройства, что особенно важно в эпоху мобильного интернета.

5. Производительность:

- HTML и CSS позволяют создавать быстро загружаемые веб-страницы, в отличие от тяжелых графических решений.

Минусы HTML и CSS:

1. Ограниченные возможности:

- HTML и CSS не являются полноценными языками программирования, они предназначены для разметки и стилевого оформления.

- Для создания сложных интерактивных приложений требуются дополнительные технологии, такие как JavaScript.

2. Сложность CSS:

- Каскадные таблицы стилей со временем могут становиться громоздкими и трудными в управлении.

- Требуется определенный опыт для правильной организации CSS-кода.

3. Отсутствие логики:

- HTML и CSS не позволяют реализовывать бизнес-логику на стороне клиента, для этого необходимы серверные технологии, такие как PHP, Python, JavaScript..

4. Ограничения дизайна:

- Несмотря на гибкость CSS, существуют определенные ограничения в возможностях дизайна.
- Для реализации сложных дизайнерских решений могут потребоваться дополнительные инструменты и библиотеки.

Таким образом, HTML и CSS являются незаменимыми базовыми технологиями для WEB-разработки для создания полноценных веб-приложений в комбинации с другими языками и инструментами.

3.3 Иные инструменты для разработки и отладки приложения

Для разработки и отладки приложения требуется локальный сервер и приложение для редактирования базы данных. В качестве локального сервера выберем XAMPP (рисунок 8) или его аналог.

XAMPP - популярный стек программного обеспечения с открытым исходным кодом, который предоставляет полный набор инструментов для локальной разработки и тестирования WEB-приложений. Существует несколько альтернатив XAMPP, предлагающих аналогичную функциональность и возможности.

Вот некоторые из наиболее популярные из них:

- WampServer - среда WEB-разработки на базе ОС Windows, которая позволяет пользователям создавать WEB-приложения с использованием Apache, MySQL и PHP. Он известен своей простотой использования и является популярной альтернативой XAMPP.

- AMPPS - бесплатный пакет программного обеспечения с открытым исходным кодом, включающий Apache, MySQL, PHP и другие инструменты для WEB-разработки. Он похож на XAMPP, но предлагает сокращенный интерфейс и известен своей простотой использования.

- Denwer - локальный сервер, разработанный в России, который предлагает ряд функций, включая поддержку нескольких баз данных и компонентов. Он известен своей гибкостью и является популярной альтернативой XAMPP в России.

- OpenServer - локальный сервер, предлагающий ряд функций, включая поддержку нескольких баз данных и компонентов. Он известен своей гибкостью и является популярной альтернативой XAMPP.

- Uniform Server - бесплатный стек программного обеспечения с открытым исходным кодом, включающий Apache, MySQL, PHP и другие инструменты для WEB-разработки. Он известен своей простотой использования и является популярной альтернативой XAMPP.

- Devilbox - локальный сервер на базе Docker, предлагающий ряд функций, включая поддержку нескольких баз данных и компонентов. Он известен своей гибкостью и является популярной альтернативой XAMPP.

- Winginx - локальный сервер, предлагающий ряд функций, включая поддержку нескольких баз данных и компонентов. Он известен своей простотой использования и является популярной альтернативой XAMPP.

Каждая из этих альтернатив имеет свои сильные и слабые стороны, и выбор между ними будет зависеть от конкретных потребностей разработчика. Выбор сделан в пользу XAMPP в связи со следующими особенностями данного программного решения:

Поддержка известных платформ: XAMPP кросс платформенное решение, поддерживающее macOS, Linux и Windows

XAMPP использует расширения и плагины для включения дополнительных инструментов, таких как Tomcat, FTP-сервер FileZilla, Webalizer, Mercury Mail и т. д.

XAMPP использует MariaDB для хранения и извлечения данных, и имеет возможность перехода на MySQL.

ХАМРР поддерживает разработку на Perl и PHP, тогда как тот же WAMP поддерживает только PHP.

Поддержка SSL: ХАМРР имеет поддержку SSL, а WAMP - нет.

- Конфигурация по умолчанию: по умолчанию ХАМРР открыт для всех соединений, но этого можно защитить, настроив его так, чтобы он прослушивал только запросы с локального компьютера.

- Запуск локального сервера на виртуальной машине может обеспечить дополнительный уровень безопасности за счет изоляции сервера от хост-машины.



Рисунок 8 – Кроссплатформенная сборка WEB-сервера ХАМРР

Для администрирования баз данных выберем решение, которое работает с большинством современных баз данных в формате WEB редактора PHPMyAdmin (рисунок 9).

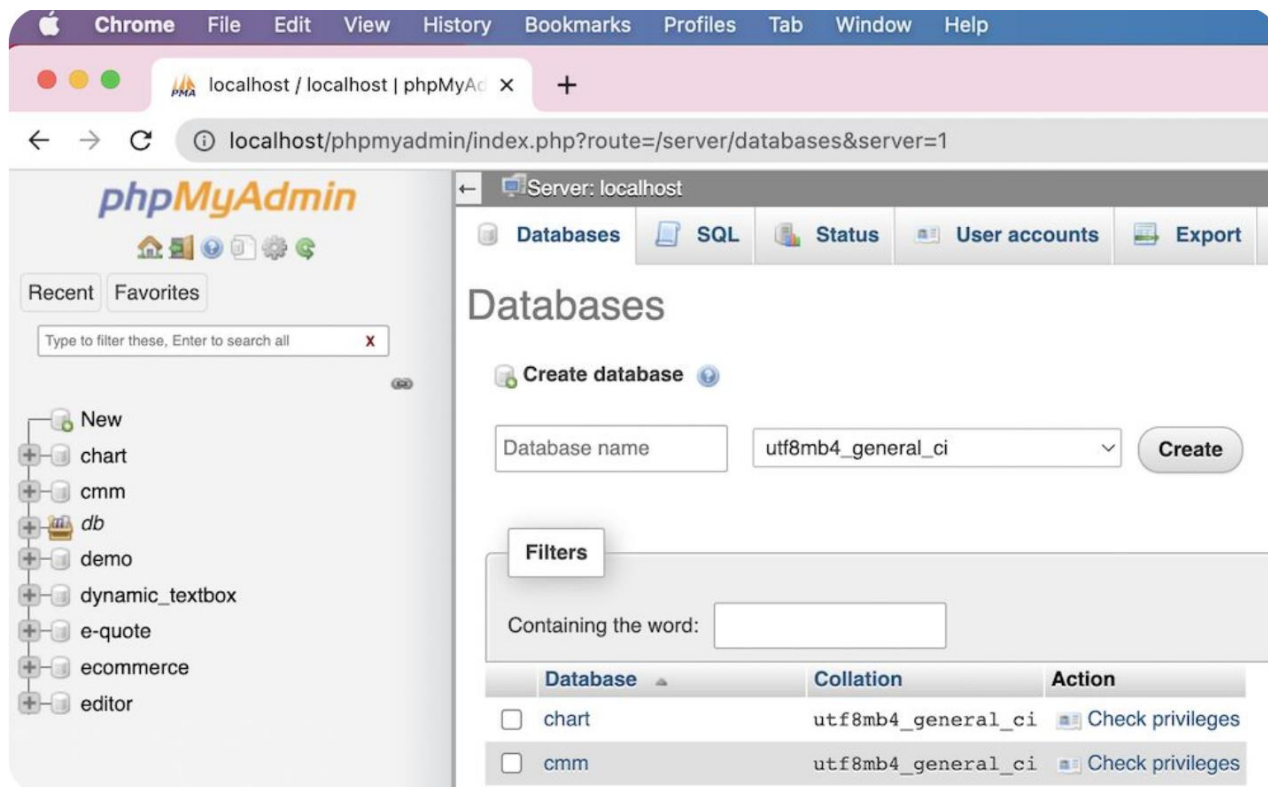


Рисунок 9 – Кроссплатформенное приложение PHPMyAdmin

PHPMyAdmin - бесплатное приложение с открытым исходным кодом, предназначенное для администрирования реляционных баз данных. Он предоставляет WEB-интерфейс для управления СУБД, выполнения команд и просмотра содержимого таблиц и баз данных через браузер. PHPMyAdmin популярен благодаря своей способности управлять базами данных без прямого ввода команд SQL, предлагая панель управления для администрирования выделенных баз данных и ее интенсивному развитию. Кроме того, его можно интегрировать в собственные разработки благодаря Стандартной общественной лицензии GNU.

Преимущества решения:

- Бесплатное программное решение с открытым исходным кодом.
- WEB-интерфейс для управления серверами и базами данных.
- Возможность управлять базами данных без прямого ввода команд SQL.
- Предлагает панель управления для администрирования выделенных баз данных.

- Интенсивное развитие и поддержка сообщества
- Может быть интегрирован в проект разработки.
- Обеспечивает высокий уровень безопасности благодаря встроенным инструментам безопасности и поддержке управления пользователями и привилегиями.

- Кроссплатформенная совместимость, работа в Linux, FreeBSD и Windows.
- Является часть популярных стеков веб-разработки, таких как LAMP и WAMP.

- Большое и развитое сообщество.

Недостатки:

- PHPMyAdmin бесплатное решение, но некоторые дополнительные плагины и расширенная техническая поддержка являются платными.

- Требует знания языка SQL и администрирования реляционных баз данных.

- Существуют угрозы безопасности, если они настроены неправильно или используются слабые пароли.

В целом, PHPMyAdmin - мощный инструмент для управления базами данных SQL, предлагающий функции, которые делают его популярным среди разработчиков и администраторов.

Существуют и альтернативные решения, предлагающие аналогичную функциональность:

- dbForge Studio - универсальный клиент СУБД SQL, предлагающий широкий спектр функций для администрирования, разработки и обслуживания баз данных (решение платное).

- Navicat: мощный инструмент администрирования баз данных, поддерживающий несколько систем баз данных, включая MySQL, PostgreSQL, Oracle и SQL Server (требуется дополнительных знаний и усилий по настройке в рамках проекта).

- HeidiSQL - бесплатный инструмент с открытым исходным кодом для управления базами данных MySQL и MariaDB, известный своей простотой

использования и многофункциональным интерфейсом (отсутствует форма поиска и фильтрации по полям таблиц БД).

- SQL Maestro - комплексный инструмент для администрирования, разработки и обслуживания баз данных, предлагающий широкий спектр функций для баз данных (профессиональное платное решение).

- EMS SQL Manager - мощный инструмент для администрирования, разработки и обслуживания баз данных, предлагающий широкий спектр функций для баз данных MySQL (профессиональное платное решение).

- SQLyog - инструмент администрирования СУБД, предлагающий широкий спектр функций для администрирования, разработки и обслуживания баз данных (профессиональное платное решение).

Эти альтернативы предлагают различную степень функциональности: от простого администрирования баз данных до комплексных инструментов разработки и обслуживания. В связи с тем, что для реализации проекта требуется простой в использовании, кросс платформенный бесплатный инструмент – PHPMysqlAdmin является оптимальным решением.

3.4 Выбор системы управления баз данных

В данном проекте в качестве СУБД используется MariaDB.

Ниже представлен сравнительный анализ популярных реляционных СУБД PostgreSQL, MySQL и MariaDB (рисунок 10).

MariaDB, MySQL и PostgreSQL - популярные СУБД с открытым исходным кодом, используемые в WEB-разработке.

У каждого есть свои сильные и слабые стороны, и выбор между ними зависит от конкретных требований проекта. Все три базы данных имеют активные сообщества, причем MySQL и PostgreSQL являются инструментами по умолчанию для создания многих приложений.

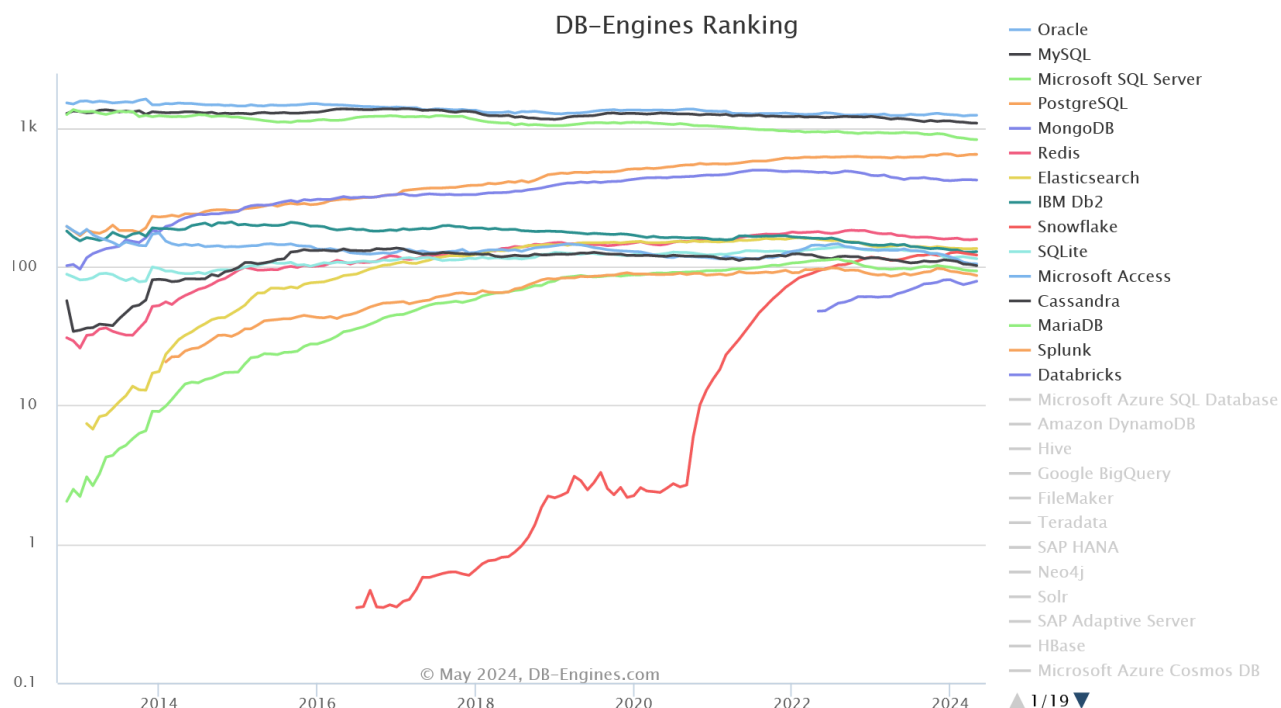


Рисунок 10 – Мировой рейтинг использования СУБД в проектах [6]

MariaDB, ответвление MySQL, набирает популярность и близка к тому, чтобы войти в десятку самых популярных СУБД в мире.

Синтаксис трех баз данных аналогичен, с небольшими различиями. Все они основаны на стандарте SQL, что делает запросы взаимозаменяемыми. Индексы PostgreSQL позволяют создавать индексы выражений, хеш-индексы и индексы В-дерева, а также частичные индексы.

MySQL и MariaDB поддерживают индексы, хранящиеся в В-деревьях, R-деревьях и хэш-индексах.

PostgreSQL считается более продвинутым и безопасным, но при этом более сложным в настройке и управлении. MySQL более широко используется и доступен, особенно для начинающих разработчиков. MariaDB - замена MySQL, предлагающая больше функций и лучшую производительность.

MySQL использует поддержку репликации «главный-подчиненный», низкие затраты на механизм хранения MyISAM, поддержку механизма хранения в памяти, кэш запросов и множество ресурсов для обучения и устранения неполадок.

MariaDB использует полный доступ ко всем функциям, полную обратную совместимость, более широкий спектр функций безопасности, больше механизмов хранения и улучшенная производительность по сравнению с MySQL.

PostgreSQL имеет улучшенную поддержку сложных запросов, мультиарендных кластеров SQL, балансировку нагрузки, кэширование SQL и мощную поддержку NoSQL, например хранилища данных JSON, а также поддержка синтаксиса XML и JSON.

В связи с тем, что XAMPP по умолчанию использует СУБД MariaDB и данная СУБД имеет весь инструментарий, для реализации проекта, в качестве БД выбрана MariaDB.

3.5 CASE инструмент моделирования данных, процессов и UML

Для разработки базы данных используются специальные CASE-средства, позволяющие разрабатывать концептуальные, логические и реляционные модели. В данном проекте используется Open ModelSphere.

Open ModelSphere - бесплатный инструмент моделирования данных, бизнес-процессов и UML с открытым исходным кодом, выпущенный под лицензией GNU General Public License (GPL). Он предоставляет комплексную платформу для моделирования различных аспектов разработки программного обеспечения, включая данные, процессы и системную архитектуру.

Преимущества решения по сравнению с аналогами:

- Бесплатность и открытый исходный код. Open ModelSphere можно бесплатно использовать, изменять и распространять, что делает ее привлекательным вариантом для частных лиц и организаций с ограниченным бюджетом.

- Инструмент поддерживает широкий спектр нотаций моделирования, включая диаграммы UML, BPMN и ER, что позволяет пользователям создавать подробные модели своих систем и процессов.

- Как инструмент с открытым исходным кодом, Open ModelSphere можно настроить в соответствии с конкретными потребностями и требованиями разработчиков.

- Поддержка сообщества инструмента с открытым исходным кодом гарантирует наличие сообщества пользователей и разработчиков, которые могут оказать поддержку и внести свой вклад в его развитие.

Недостатки решения по сравнению с аналогами:

- Open ModelSphere - это сложный инструмент, для изучения и освоения которого требуется значительное количество времени и усилий.

- Пользовательский интерфейс инструмента может быть не таким совершенным и удобным, как у некоторых коммерческих альтернатив.

- Сообщество открытого исходного кода может предоставить поддержку, она может быть не такой надежной и оперативной, как варианты коммерческой поддержки.

Open ModelSphere - мощный и гибкий инструмент для моделирования различных программного обеспечения.

4 Разработка программы для квалификационного тестирования специалистов

Программа для тестирования IT-специалистов состоит из трех взаимосвязанных частей: серверная, клиентская части и система управления базой данных.

С помощью контекстной диаграммы DFD (рисунок 11) описана система для тестирования IT-специалистов на высоком уровне абстракции. Контекстная диаграмма необходима для понимания контекста и логики процесса реализации программного обеспечения, не погружаясь в технические детали реализации.

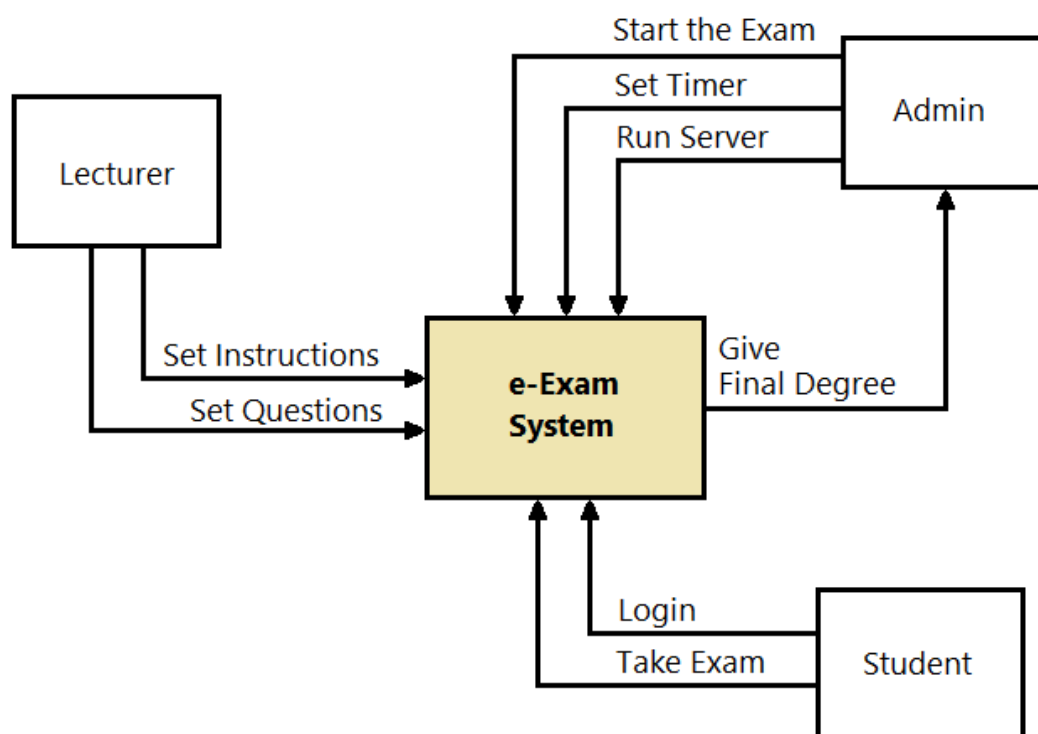


Рисунок 11 – Контекстная диаграмма программы для тестирования IT-специалистов

На основании диаграммы представленной на рисунке 10 можно сделать вывод, что для реализации целей и задач, поставленных в данном дипломном проекте п. 1.3, потребуется три роли: студент, лектор и администратор.

Роль администратора - произвести развертывание и запуск системы, а также работа с таймингом тестирования.

Роль лектора создание, удаление и редактирование тестов и вопросов к ним.

Роль студента сдача тестирования в соответствии с правилами установленными администратором и лектором.

В связи с тем что, предполагается реализовать облегченный вариант программного обеспечения, роль администратора и лектора совмещены в одну роль. После определения общего понимания процессов и ролей для реализации проекта на основании DFD диаграммы опишем физическое взаимодействие модулей системы на хостах (рисунок 12).

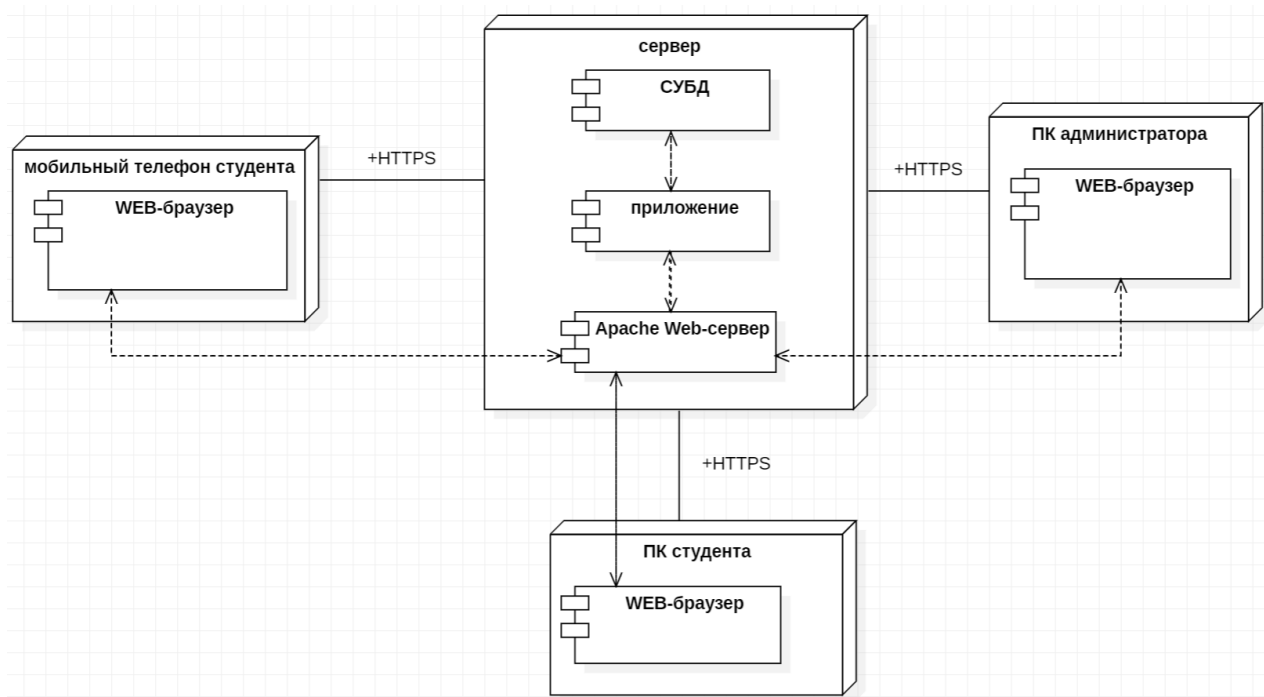


Рисунок 12 – Диаграмма развертывания программы для тестирования
IT-специалистов

На рисунке 12 показано, что ядром системы является серверная часть, в основе которой будут: WEB сервер, система БД и приложение (PHP). Так как в проекте используется XAMPP, все три модуля ядра будут развернуты на одном хосте.

В качестве клиента может выступить традиционный хост (ПК) или мобильное устройство, так как клиентская часть приложения – «тонкий клиент» в

браузере в не зависимости, какую роль требуется выбрать в данный момент времени.

Далее в разделах описывается база данных и структура и сущности программного кода реализации программы для тестирования IT-специалистов.

Так как данные и структура данных БД влияет на реализацию логики и интерфейса приложения, проектирование программы было начато с реализации СУБД.

4.1 Проектирование базы данных

Реализованная схема базы данных, представленная на рисунке 13, является реляционной БД на базе СУБД MariaDB. Реляционная БД позволяет: избежать избыточных, дублирующихся и недопустимых данных. Структуру реляционной базы данных легко изменять и поддерживать, есть возможность масштабировать базу данных в целом и таблицы в частности. Для реализации реляционной БД используют логическое и физическое представления сущностей БД.

Логическая модель данных БД - это представления данных, в которых подчеркивается структура и отношения без учета физической реализации, тогда как физические модели данных описывают, как данные будут храниться, и получать к ним доступ.

Логическая модель данных - это вариант концептуальной модели, представляющая ограничения данных, имена сущностей и отношения независимым от СУБД способом, в то время как физическая модель данных дополнительно уточняет логическую модель для реализации в конкретной технологии СУБД.

4.1.1 Список требований к данным БД

Ниже представлен список требований к базе данных на основании, которого была реализована логическая модель приложения. В списке требований изложены условия, ограничения и критерии, на которых основан проект.

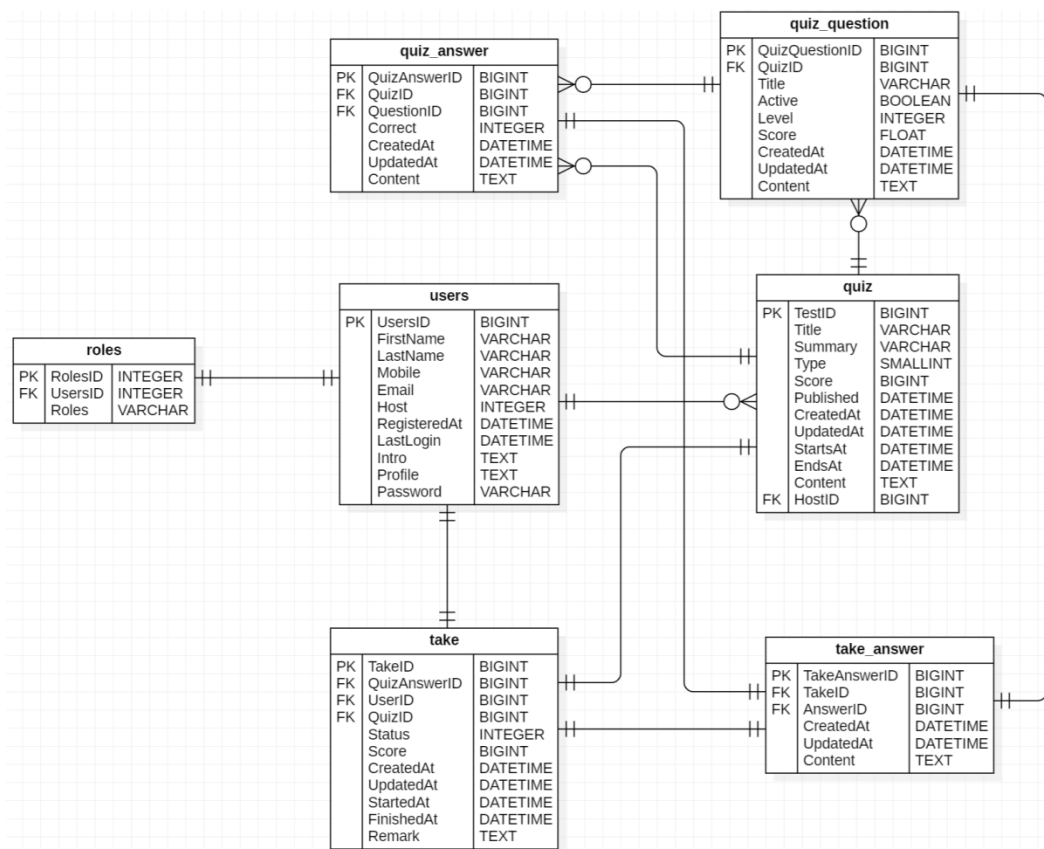


Рисунок 13 – Логическая модель данных СУБД

Для программы для тестирования IT-специалистов включают следующие требования:

- Регистрация тестируемых (студентов) должны содержать их имена и адреса электронной почты. При желании можно зарегистрировать и другие данные, такие как номер телефона.
- Регистрация администраторов (преподавателей) должна содержать их имена и адреса электронной почты. При желании можно также указать их номера телефонов.
- Администратор может тестировать более одного курса, и более одного администратора могут тестировать один курс.
- Каждый студент может записаться на один или несколько тестов. Когда студент записывается на тест, регистрируется дата зачисления.
- Запись студента на тест может быть отменена.

- Тест состоит из определенного количества вопросов, которые нумеруются последовательно.

- Прохождение теста необходимо регистрировать.

- В базе данных должна вестись запись о каждом прохождении студентом теста, сохраняя полученные баллы.

Логическая модель базы данных приведена на рисунке 12 и содержит шесть сущностей.

4.1.2 Реализация сущностей базы данных

Таблица пользователей необходима для хранения информации о пользователях (таблица 3). Данная таблица используется для связи с тестами, чтобы пользователи могли управлять своими тестами и отслеживать результаты. Ниже приведено описание всех столбцов таблицы.

Таблица 3 – Описание атрибутов сущности «Roles»

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
RolesID	Уникальный идентификатор роли.	РК	
UsersID	ID пользователя		FK
Roles	Описание роли		

Таблица 4 – Описание атрибутов сущности «Users»

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
UsersID	Уникальный идентификатор для идентификации пользователя.	РК	
FirstName	Имя пользователя		
LastName	Фамилия пользователя		

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
Mobile	Номер мобильного телефона пользователя		
Email	Электронная почта пользователя (используется для входа и регистрации)		
Password	Хэш пароля, сгенерированный соответствующим алгоритмом		
Host	Флаг, указывающий, может ли пользователь проводить тест		
RegisteredAt	Расчет времени использования пользователем с приложения		
LastLogin	Идентификация последнего входа пользователя в систему		
Intro	Краткое описание пользователя, которое будет отображаться на странице теста		
Profile	Сведения о владельце будут отображаться на странице теста		

Таблица 5 – Описание атрибутов сущности «Quiz»

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
TestID	ID идентифицирующий теста	РК	
HostID	ID хоста для идентификации хоста теста		
Title	Название теста, которое будет отображаться на странице		

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
	теста и в списках.		
Summary	Резюме с указанием ключевых моментов.		
Type	Тип, позволяющий различать типы тестов.		
Score	Общий балл за ответы		
Published	Признак является ли тест общедоступным		
CreatedAt	хранит дату и время создания теста		
UpdatedAt	хранится дата и время обновления теста		
StartsAt	хранит дату и время начала теста		
EndsAt	хранятся дата и время закрытия теста		
Content	Данные о тесте		

Таблица 6 – Описание атрибутов сущности «QuizQuestion»

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
QuizQuestionID	ID вопроса	РК	
QuizID	ID теста к которому относится вопрос		FK
Title	Название теста, которое будет отображаться на странице теста и в списках.		

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
Active	Флаг, указывающий, активен ли вопрос.		
Level	Уровень вопроса, чтобы определить, легкий ли он, средний или сложный.		
Score	Оценка отдельного вопроса. Перед публикацией теста мы должны убедиться, что одновременно активны только выборочные вопросы, а общий балл активных вопросов равен баллу теста.		
CreatedAt	Он хранит дату и время создания вопроса.		
UpdatedAt	В нем хранится дата и время обновления вопроса.		
Content	Столбец, используемый для хранения вопроса.		

Таблица 7 – Описание атрибутов сущности «QuizAnswer»

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
QuizAnswerID	ID ответа на вопрос	PK	
QuizID	ID теста		FK
QuestionID	ID вопроса теста		FK
Correct	Флаг, указывающий,		

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
	правильный ли ответ.		
CreatedAt	дату и время создания ответа.		
UpdatedAt	дату и время обновления ответа.		
Content	Столбец, для хранения ответа.		

Таблица 8 – Описание атрибутов сущности «Take»

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
QuizAnswerID	ID попытки прохождения теста	РК	
UserID	ID теста		FK
QuizID	ID вопроса теста		FK
Status	Флаг, указывающий, номер попытки прохождения теста.		
Score	Общий балл, полученный пользователем в крайней попытке.		
CreatedAt	Дата и время создания попытки пройти тест		
UpdatedAt	Дата и время обновления попытки пройти тест		
StartedAt	Дата и время начала попытки пройти тест		
FinishedAt	Дата и время завершения		

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
	попытки пройти тест		
Remark	Заметка администратора о попытке		

Таблица 9 – Описание атрибутов сущности «TakeAnswer»

Имя атрибута	Назначение	Первичный ключ	Внешний ключ
TakeAnswerID	ID попытки прохождения теста	РК	
TakeID	ID теста		FK
AnswerID	ID выбранного ответа на вопрос		FK
Created At	Дата и время ответа тест		
UpdatedAt	Дата и время обновления ответа		
Content	Поле необходимо если ответ в текстовом виде		

4.2 Реализация программного приложения

Основной функционал программы для интерактивного тестирования IT специалистов реализован в виде трех типов модулей.

Модули интерфейса – необходимы для отображения и ввода информации пользователями. Модули контроллеры отвечают за обработку запросов от модулей интерфейса и модулей моделей. Модули модели необходимы для работы с данными БД.

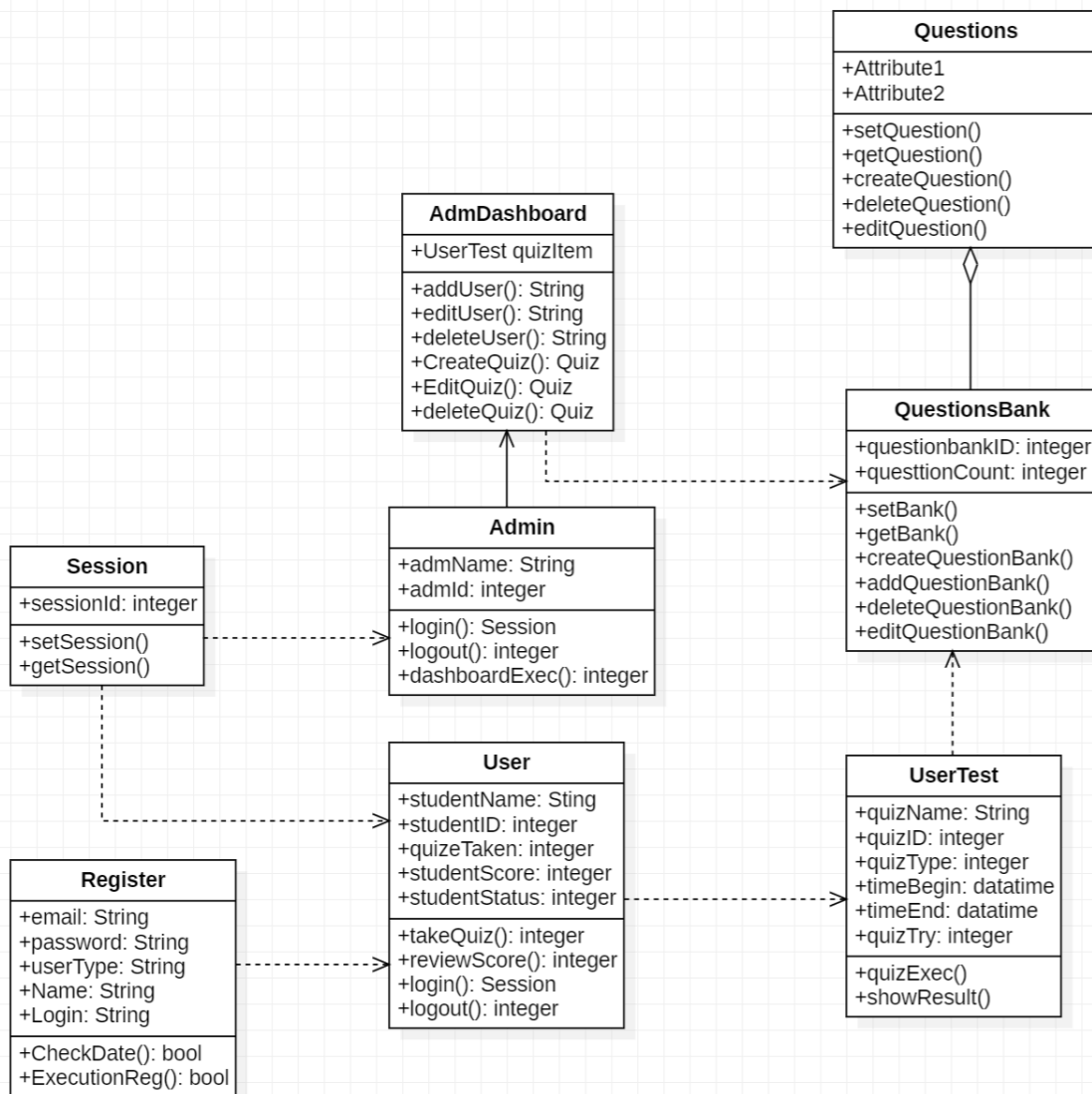


Рисунок 14 – Диаграмма классов программы для интерактивного тестирования IT специалистов

Модули интерфейса программы приведены в таблице 10.

Таблица 10 – Модули программы

Модуль	Назначение
Login	Модуль домашней страницы
Register	Модуль для авторизации или регистрации
User	Модуль для выбранного теста
UserTest	Модуль для прохождения теста

Модуль	Назначение
Results	Модуль для просмотра результатов пройденных тестов
AdminDashboard	Модуль управления тестами администратором системы

Модуль Login представляет собой домашнюю страницу, на которой пользователь может зарегистрироваться и войти в систему.

В модуле Login генерируются следующие перечисленные ниже элементы управления.

Текстовые поля ввода:

- поле ввода «Логин» для ввода логина;
- поле ввода «Пароль для входа» для ввода пароля;
- поле ввода «Код (только для администраторов)».

Кнопки:

- кнопка «Вход» для выполнения входа в систему;
- кнопка «Регистрация» для регистрации в система;

Графический интерфейс модуля Home в пустыми и заполненными полями ввода представлены на рисунке 15.

Рисунок 15 – Модуль Login. Домашняя страница

Код модуля содержит ссылки к страницам перехода на домашнюю страницу пользователя или на страницу регистрации, если пользователь еще не зарегистрирован. Модуль Login содержит объекты перехода: login, home, register. В зависимости от действий на странице и выбранной кнопки управления, будет реализован сценарий перехода на основную страницу пользователя или на страницу регистрации.

```
<body class="login">

<div class="relative flex items-top justify-center min-h-screen bg-
gray-100 dark:bg-gray-900 sm">

    @if (Route::has('login'))

        <div class="login">

            @auth <a href="{{ url('/home') }}" class="text-sm text-
gray-700 underline">Home</a>

            @else

<a href="{{ route('login') }}" class="register">Log in</a>

            @if (Route::has('register'))

                <a href="{{ route('register') }}" class="ml-4 text-sm
text-gray-700 underline">Register</a>

            @endif

        </div>

    @endif
```

В модуле Register генерируются следующие перечисленные ниже элементы управления.

Текстовые поля ввода:

– поле ввода «ФИО» для ввода логина;

- поле ввода «Логин» для ввода пароля;
- поле ввода «Электронная почта»;
- поле ввода «Пароль» для ввода пароля.

Кнопки:

- кнопка «Регистрация» для регистрации в система;
- кнопка «Вход в систему» для выполнения входа в систему;

Графический интерфейс модуля Register с полями ввода представлен на рисунке 16.

Система онлайн-экзаменов

Вход Регистрация Админ

Система онлайн-экзаменов
Полная система онлайн-экзаменов на основе MCQ

Полное имя
Введите полное имя

Введите ваше полное имя (пробелы разрешены).

Имя пользователя
Введите имя пользователя (без пробелов)

Введите только ваше имя пользователя

Email
Введите email

Мы никогда не передадим ваш адрес электронной почты кому-либо еще

Пароль
Пароль

Регистрация Уже зарегистрированы? Войдите

Рисунок 16 – Модуль Home. Домашняя страница с введенными данными

Функции данного модуля реализуются следующим кодом:

```
form method="post" action="{{route('registration')}}">
{
    @csrf
    <div class="form-group">
        <label for="Полное имя">Username</label>
        <input type="text" name="user" class="form-control"
id="exampleInputEmail1" aria-describedby="emailHelp"
placeholder="Enter Username">
        <small id="Email" class="form-text text-muted">We'll
never share your email with anyone else.</small>
```

```

</div>
<div class="form-group">
    <label for="Пароль">Password</label>
    <input type="password" name="pass" class="form-control"
id="Пароль" placeholder="Password">
</div>
<button type="submit" class="btn btn-primary">Регистрация</button>
</form>
<span>Уже зарегистрированы? Войдите <a
href="{{route('login.view')}}"> </a></span>
}

```

Для регистрации в системе пользователь должен ввести фамилию, имя, логин, адрес электронной почты и пароль в соответствующие поля ввода, затем нажать кнопку «Регистрация».

Для входа в систему пользователь должен выполнить описанные выше действия и вместо кнопки «Регистрация» нажать кнопку «Вход».

После регистрации или авторизации в системе происходит проверка правильности ввода данных и вход в систему, реализованный в модуле Entry:

Если данные введены, верно, то выполняется переход на главную страницу пользователя – модуль Home, показанную на рисунке 17.

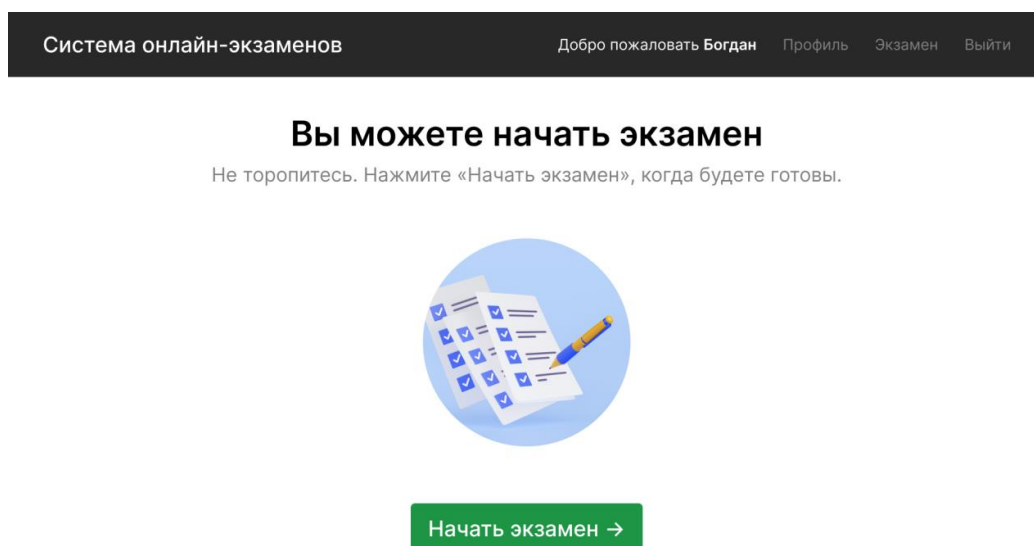


Рисунок 17 – Модуль Home. Главная страница при входе от имени пользователя

Страница представляет набор URL с динамическим переходом к модулям: Test, Profile, Login. Листинг данного модуля приведен ниже.

```
@extends('layouts.dashboard')
@section('title')
    <title>Основная страница пользователя для тестирования </title>
@endsection
@section('main')
    <tr>
        <th scope="col name ="Профиль" ">profile</th>
        <th scope="col name ="Экзамен"">testing</th>
        <th scope="col name ="Выйти"">login</th>
    </tr>
</thead>
<tbody>
    @php
        $sl=1;
    @endphp
    <tr>
        <th scope="button">< «Начать экзамен» > {{testing}}</th>
        <td><a
                                href="/give-quiz/{{ $quiz->id }}">{{ $quiz->title }}</a></td>
        <td>{{ $quiz->from_time }}</td>
        <td>{{ $quiz->to_time }}</td>
    </tr>
</tbody>
</table>
</div>
@endsection
```

В результате нажатия на кнопку «Начать экзамен» будет осуществлен переход в модуль UserTest (рисунок 18).

Добро пожаловать на онлайн-экзамен

Проверьте свои знания

Общее количество вопросов: **10**

Тип вопроса: Множественный выбор (MCQ)

Продолжить →

Рисунок 18 –Страница выбранного теста

На странице модуля UserTest отображается информация о тесте. По нажатию на кнопку «Продолжить» пользователь перейдет к выполнению тестирования модуля UserTest .В данной части модуля UserTest (рисунок 19) содержатся генераторы страницы вопросов и тайминг выполнения тестов. Тайминг тестирования не отображается на странице пользователя, а фиксируется в БД.

Вопрос 1 из 10

Вопрос 1: Имя переменной в PHP начинается с

- ☐ ! (Восклицательный знак)
- ☐ \$ (Доллар)
- ☐ & (Амперсанд)
- ☐ # (Хэштег)

Продолжить

Рисунок 19 – Модуль UserTest. Прохождение тестирования при входе от имени обычного пользователя

```

@section('main')
    <h1>It's a Quiz Page</h1>
    <h2>Quiz Title: {{$quiz->title}}</h2>
    <h3>Exam Time: {{$quiz->duration}} {{$quiz->duration*60}} </h3>
    <form method="post" action="{{route('store.answer')}}">
        @csrf
        <input type="hidden" name="quiz_id" value="{{$quiz->id}}" readonly
        required>
        <input id="next_question" type="hidden" name="Продолжить"
        value="{{$current_time}}" readonly required>
        @php
            @endphp
        @foreach($questions as $question)
            <select name="answer[{{$i}}]" class="form-control"
            required>
                <option selected disabled value>Question: {{$question->
                question}}</option>
                <option value="option_a">{{$question->option_a}}</option>
                <option value="option_b">{{$question->option_b}}</option>
                <option value="option_c">{{$question->option_c}}</option>
                <option value="option_d">{{$question->option_d}}</option>
            </select>
            <hr>
        @endforeach
        <button type="submit" class="btn btn-primary">Продолжить</button>
    </form>

```

Количество итераций выполнения модуля UserTest соответствует количеству вопросов в тесте. После прохождения тестирования и ответа на все вопросы модуль UserTest генерирует страницу результатов (рисунок 20).

Поздравляем! Вы только что завершили тест

Проверьте свой результат. Вы также можете проверить правильные ответы

Финальный счет: 10

[Посмотреть ответы ✓](#)[Начать тест →](#)

Рисунок 20 – Страница результата тестирования

```
@section('title')
    <title>Результаты тестирования</title>
@endsection
@section('main')
    <h1>Result List</h1>
    <div class="text-center">
        <table class="table">
            <thead>
                <tr>
                    @if(session('user_role')=='user')
                        <th scope mod=UserTest.genText>#</th>
                        <th scope name ="Финальный счет" mode=
UserTest.genScore > Финальный счет </th>
                        <th scope="button">Посмотреть ответы</th>
                        <th scope="button">Начать тест</th>
                    @endif
                    <th scope="col">Date</th>
                </tr>
            @endsection
```

По нажатию на кнопку «Посмотреть тест» (рисунок 21) будет осуществлен переход в модуль Results для детального просмотра результатов тестирования.

Все вопросы и ответы – Всего 10 вопросов

Вопрос 1: Что означает SPL в PHP?

- ☒ Стандартная PHP библиотека
- ☐ Простая PHP библиотека
- ☐ Простой PHP список
- ☐ Простой язык программирования

Вопрос 2: Что означает SPL в PHP?

- ☒ Стандартная PHP библиотека
- ☐ Простая PHP библиотека
- ☐ Простой PHP список
- ☐ Простой язык программирования

Начать экзамен →

Рисунок 21 - Модуль Results. Сообщение об ошибке

Модуль содержит, который в цикле выводит все вопросы с отмеченными ответами. В случае если ответы верные они отмечаются зеленым цветом.

```
$quiz = $quizAttempt->quiz;
$question = $quizQuestion->question;
$correct_answer = ($question->correct_options())->first()->id;
$negative_marks = self::get_negative_marks_for_question($quiz,
$quizQuestion);
if (!empty($correct_answer)) {
    foreach(count($quizQuestionAnswers)) {
        if (is_array($quizQuestionAnswers)) {
            return $quizQuestionAnswers[0]->question_option_id ==
$correct_answer ? $quizQuestion->marks : -($negative_marks); //
Return marks in case of correct answer else negative marks
        }
    }
}
```

По нажатию на ссылку «Профиль» пользователь будет направлен на страницу обновления профиля (рисунок 22).

Система онлайн-экзаменов

Добро пожаловать **Богдан** Профиль Экзамен Выйти

Обнови свой профиль

Имя

Богдан Киреев

Имя пользователя

bogdan

Email

bogdan@mail.ru

Обновить информацию

Рисунок 22 – Модуль Profile. Личный кабинет пользователя

В модуле Profile генерируются следующие перечисленные ниже элементы управления.

Текстовые поля ввода:

- поле ввода «ФИО» для ввода фамилии и имени пользователя;
- поле ввода «Логин» для ввода пароля;
- поле ввода «Электронная почта»,

Кнопки:

- кнопка «обновить информацию» для обновления информации в системе.

Ниже приведен код данной страницы.

```
<div class="form-group">
    <label for="exampleInputFIO">Имя</label>
    <input type="text" name="user" class="form-control"
id="exampleInputFIO" aria-describedby="emailHelp" placeholder="Имя">
    <small id="emailHelp" class="form-text text ">Имя
пользователя</small>
```

```

<label for="Email">Email</label>
    <input type="text">
</div>
    <button type="submit" class="btn btn-primary">Обновить
информацию</button>
</form>
<a href="{{route('profile.view')}}"></a></span>

```

При нажатии на кнопку «Выход» выполняется выход из системы и возвращение на страницу модуля Login.

При входе в систему пользователя с правами администратора будет отображаться основная страница, представленная на рисунке 22.

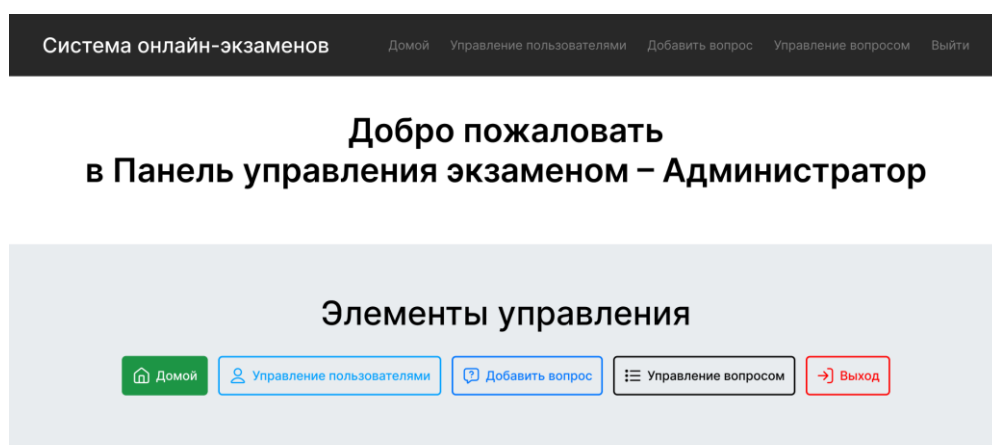


Рисунок 22 – Модуль Ноте. Личный кабинет администратора

```

<div class="container">
<div class="row">
<div class="col-lg-12 text-center">
<h1 class="mt-5">Приветствуем на панели контроля экзаменации –
Админ</h1>
<br/>
<br/>
<div class="jumbotron">
<h1>Управление</h1>
<a class="btn btn-outline-success btn-lg" href="index.php"><span
class="fa fa-home"></span> Главная</a>

```

```

<a class="btn btn-outline-info btn-lg" href="users.php"><span
class="fa fa-user-circle"></span> Управление пользователями</a>
<a class="btn btn-outline-primary btn-lg" href="quesadd.php"><span
class="fa fa-question-circle"></span> Добавить вопрос</a>
<a class="btn btn-outline-dark btn-lg" href="queslist.php"><span
class="fa fa-list"></span> Управление вопросами</a>
<a class="btn btn-outline-danger btn-lg" href="?action=logout"><span
class="fa fa-sign-out"></span> Выход</a>
</div>
</div>

```

Управление пользователями

#	Имя	Имя пользователя	Email	Действие
1	Богдан Киреев	bogdan	bogdan@mail.ru	Убрать

Рисунок 23 – Модуль Users. Личный кабинет администратора

На рисунке 23 показана работа в режиме администратора, который может посмотреть информацию о зарегистрированных пользователях.

С главной страницы модуля AdmDashboard администратор имеет доступ к банку вопросов QuestionsBank (рисунок 24) и отдельному вопросу Questions, в результате будет сформирован банк вопросов для прохождения пользователями.

Добавить вопросы

Вопрос №	:	<input type="text" value="11"/>
Вопрос	:	<input type="text" value="Введите вопрос"/>
Ответ 1	:	<input type="text" value="Введите ответ №1"/>
Ответ 2	:	<input type="text" value="Введите ответ №2"/>
Ответ 3	:	<input type="text" value="Введите ответ №3"/>
Ответ 4	:	<input type="text" value="Введите ответ №4"/>
Правильный №	:	<input type="text"/>

Подтвердить вопрос

Рисунок 24 – Модуль Question. Личный кабинет администратора

Список вопросов

#	Имя	Действие
1	Имя переменной в РНР начинается с	Убрать
2	Имя переменной в РНР начинается с	Убрать
3	Имя переменной в РНР начинается с	Убрать
4	Имя переменной в РНР начинается с	Убрать
5	Имя переменной в РНР начинается с	Убрать
6	Имя переменной в РНР начинается с	Убрать
7	Имя переменной в РНР начинается с	Убрать

Рисунок 25 – Модуль Questions. Личный кабинет администратора

На рисунке 25 показан переход по ссылке «Управление вопросом» для перехода к методу editQuestions.

5 Технология создания базы данных программы для интерактивного тестирования IT специалистов

В данном проекте в качестве инструмента работы с БД выбран MySQL Workbench. MySQL Workbench визуальный инструмент для администраторов баз данных, архитекторов баз данных и разработчиков MySQL. Этот инструмент предлагает инструменты моделирования данных, разработки SQL и администрирования для настройки и администрирования сервера.

Для работы с данным приложением, первым шагом была его инсталляция на персональный компьютер (рисунок 26).

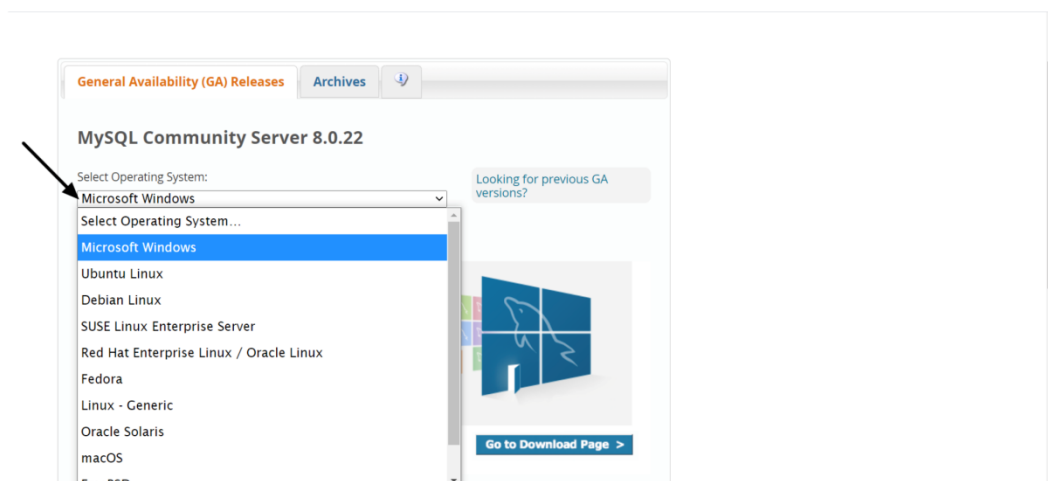


Рисунок 26 – Выбор платформы для установки MySQL Workbench

При появлении диалогового окна Wizard, необходимо выбрать программную платформу на которую будет установлено данное программное обеспечение (рисунок 26), в дипломном проекте – ОС Windows.

Далее выбираются инструменты и компоненты, с которыми будет производиться работа (рисунки 27, 28).

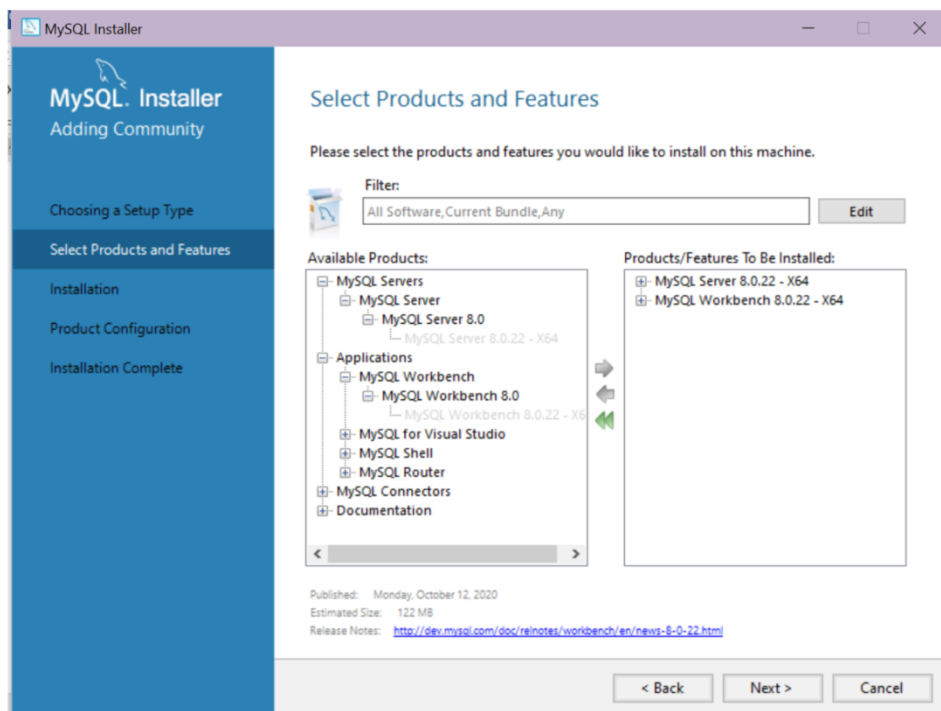


Рисунок 27 – процесс выбора компонентов необходимых для работы

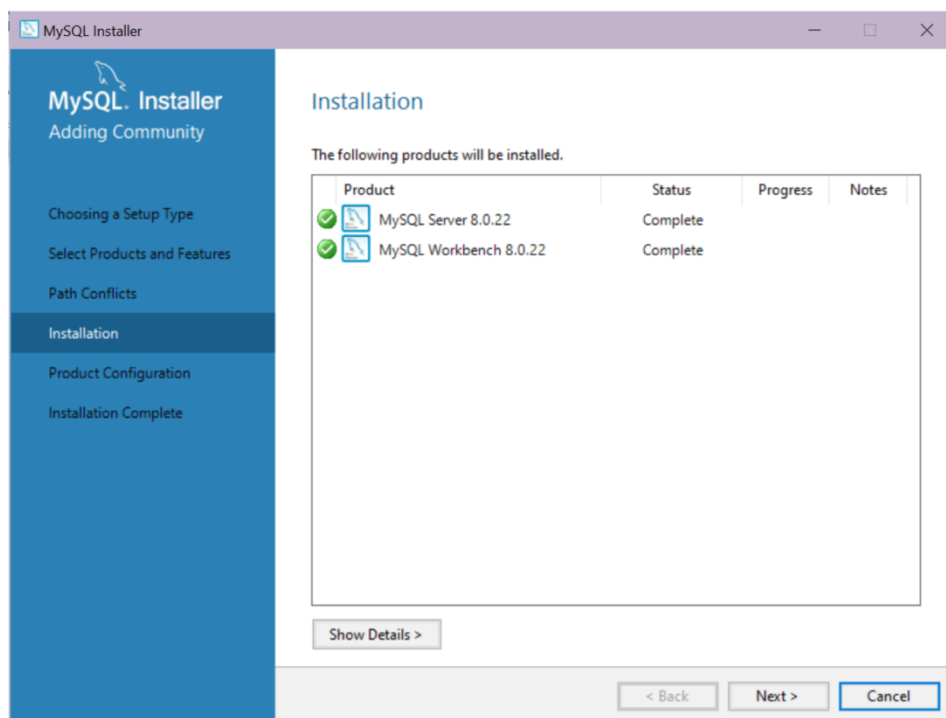


Рисунок 28 – Минимальный набор компонентов

После установки проверяются сетевые настройки между MySQL клиентом и сервером (рисунок 29). Без данной связки MySQL Workbench не сможет работать с БД и таблицами СУБД.

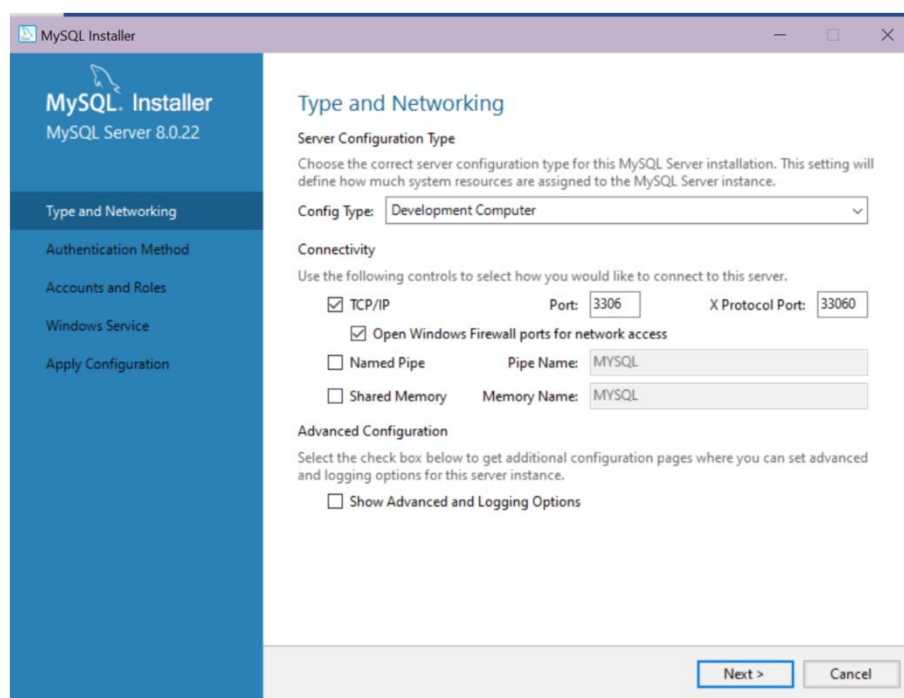


Рисунок 29 – Сетевые настройки для работы

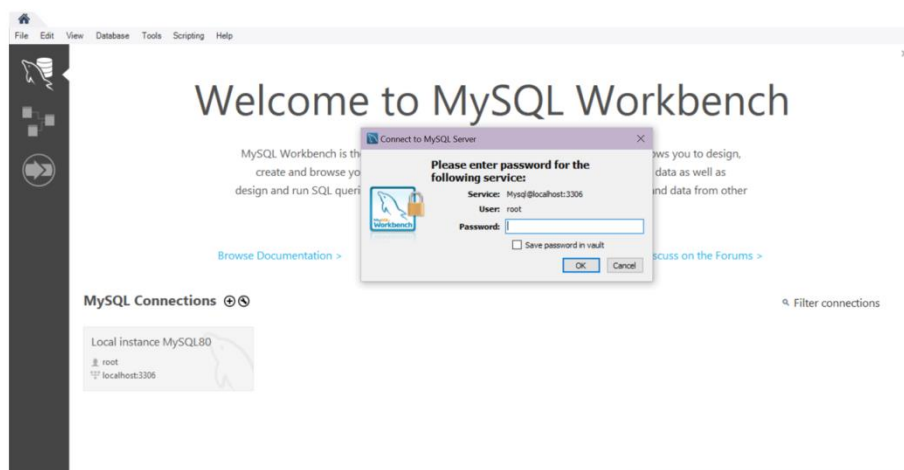


Рисунок 30 – Запуск приложения для работы

Далее следует перезагрузить приложение и с помощью клиента подключиться к СУБД MySQL (рисунок 30).

Если подключение успешно, то тестовый запрос к СУБД (рисунок 31) выдаст результат в виде созданной базы данных.

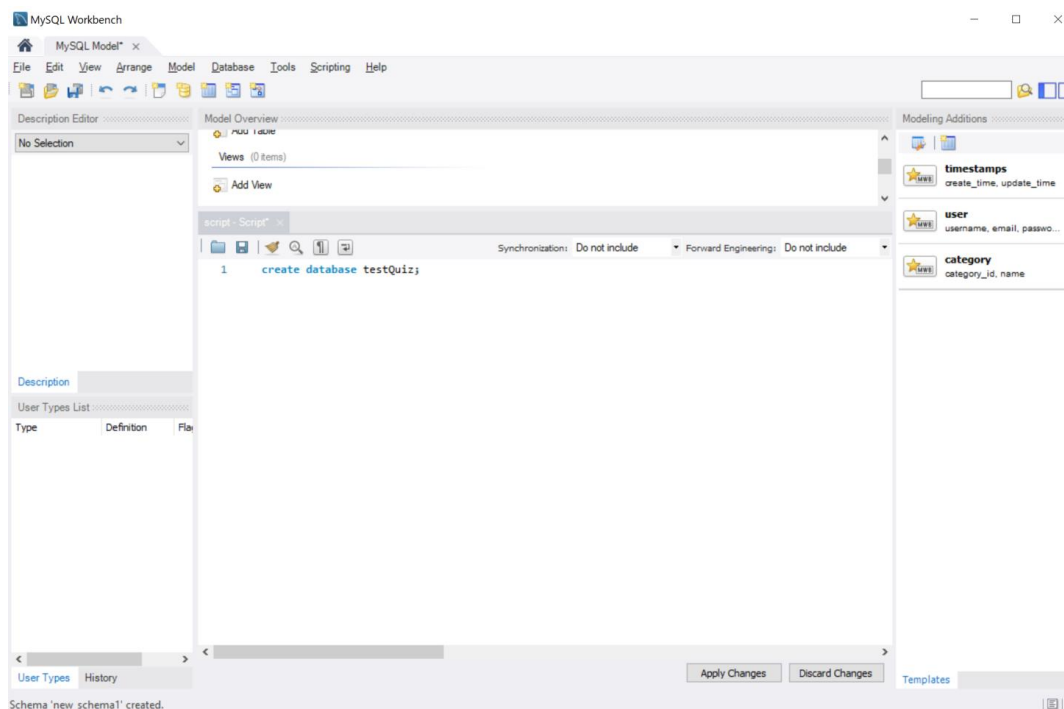


Рисунок 31 – Инструмент «Query Tool»

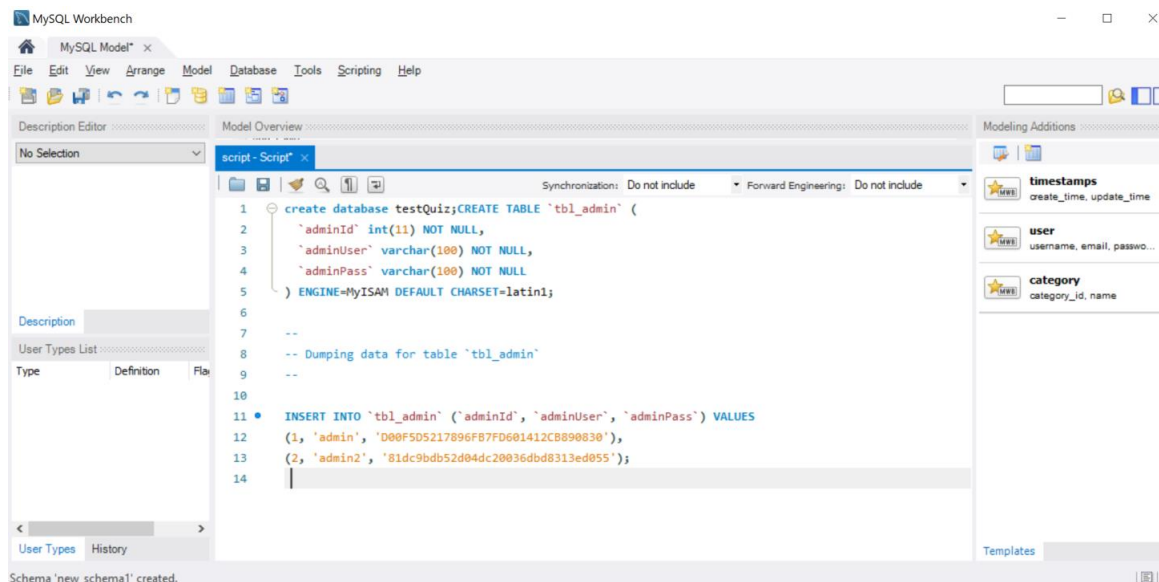


Рисунок 32 – Создание таблиц «Admins», «Users»

После создания БД, с помощью скриптов или интерактивных инструментов можно производить полный набор действий с базой данных и таблицами в ней.

6 Оценка затрат на разработку и внедрение программы для интерактивного тестирования IT специалистов

6.1 Организация производства разработки программы для интерактивного тестирования IT специалистов

Разработка программного приложения осуществлена на базе собственной площади общей площадью 30 кв.м. Содержание помещения в месяц составляет 15000 рублей.

Для разработки программного продукта приобретено оборудование (вычислительная техника), лицензионное программное обеспечение, дополнительные материалы.

Данные по потребности в оборудовании и лицензионном программном обеспечении представлены соответственно в таблицах 11 и 12.

Таблица 11 – Потребность в оборудовании

Оборудование	Источник	Количество	Ориентировочная цена за единицу, рублей	Потребляемая мощность, Ватт
Компьютер Lime 410	ООО «ДНС Ритейл»	5	51000	150
Сервер Lime S510	ООО «ДНС Ритейл»	1	172000	500
Клавиатура и мышь	ООО «ДНС Ритейл»	5	900	2,5
Монитор HP H2718	ООО «ДНС Ритейл»	5	15500	18

Таблица 12 – Потребность в лицензионном программном обеспечении

Лицензионное программное обеспечение	Источник	Цена, рублей	Срок действия лицензии, дней
Microsoft Windows 10	Интернет-магазин ООО «1Бит»	29000	Бессрочная
Лицензионное программное обеспечение	Источник	Цена, рублей	Срок действия лицензии, дней
Антивирус Kaspersky Plus 2023	Интернет-магазин Kaspersky URL: https://www.kaspersky.ru/	1999	365
VS Code	Интернет-магазин VS Code URL: https://code.visualstudio.com/download	-	Бессрочная
СУБД PostgreSQL	Интернет-магазин PostgreSQL URL: https://www.postgresql.org/download/	-	Бессрочная
OpenModelSphere 3.2	Интернет-магазин Open Model Sphere URL: http://www.modelsphere.com/org/download.html	-	Бессрочная

Перечень дополнительных материалов, источники их приобретения, объем, и ориентировочные цены приведены в таблице 13.

Таблица 13 – Потребность в дополнительных материалах

Материал	Источник	Количество, штук	Ориентировочная цена за единицу, рублей
Стол рабочий с надстройкой	ООО «Мебельер»	1	2500
Настольная лампа	ООО «Мебельер»	5	1200
Стол компьютерный	ООО «Мебельер»	4	5000
Кресло	ООО «Мебельер»	4	5000
Бумага	Гипермаркет «Канцлер»	5	300

Материал	Источник	Количество, штук	Ориентировочная цена за единицу, рублей
Канцтовары	Гипермаркет «Канцлер»	5	800

Разработка программы для интерактивного тестирования IT специалистов осуществлялась командой разработчиков, представленных в таблице 14.

Данные по основной заработной плате разработчиков программного приложения и их количественному составу приведены в таблице 15.

Таблица 14 – Состав коллектива разработчиков

Специалист	Количество, человек	Описание занятости
Руководитель проекта	1	<ul style="list-style-type: none"> - управление проектом - набор команды исполнителей - управление ресурсами проекта - контроль соблюдения требований заказчика - приемка и внедрение проекта - анализ предметной области
Программист	1	<ul style="list-style-type: none"> - создание базы данных - написание, отладка и первичное тестирование программного кода - разработка и утверждение программной документации - приемка и внедрение программного кода
Тестировщик	1	- тестирование программы

Таблица 15 – Сведения о заработной плате

Специалист	Количество, человек	Сумма заработной платы, рублей
Руководитель проекта	1	80000
Программист	1	60000
Тестировщик	1	40000

6.2 Расчет себестоимости программного обеспечения для интерактивного тестирования IT специалистов

Себестоимость разработки программного приложения включает следующие расходы:

- на приобретение оборудования;
- на приобретение лицензионного программного обеспечения;
- затраты на дополнительные материалы;
- затраты на заработную плату;
- затраты на обслуживание программы.

Затраты на оборудование в период разработки составят 509000 рублей и представлены в таблице 11.

Затраты на программное обеспечение в период разработки составят 30999 рублей и представлены в таблице 12.

Затраты на дополнительные материалы в период разработки составят 54000 рублей и представлены в таблице 23.

Затраты на заработную плату включают затраты на основную заработную плату разработчиков программного приложения и отчисления на страховые взносы.

С учетом выходных дней при 5-дневной рабочей неделе количество рабочих дней в месяце в среднем составляет 21 день. На разработку всего проекта необходимо 3 месяца.

Таким образом, затраты на заработную плату разработчиков составят 180000 рублей в месяц, 509000 рублей за весь период создания проекта.

Отчисления на страховые взносы – обязательные отчисления по нормам, установленным законодательством государственного социального страхования, в Пенсионный фонд РФ, фонд обязательного медицинского страхования, фонд социального страхования РФ от затрат на оплату труда работников, включаемых в

себестоимость продукции (программного обеспечения), по элементу «Затраты на оплату труда» (кроме тех видов оплаты, на которые страховые взносы не начисляются).

Отчисления на страховые взносы производится согласно «Налоговому кодексу Российской Федерации» от 14.07.2002 № 239-ФЗ.

С января 2023 года введена единая базовая ставка по страховым взносам. Действующие тарифы на страховые взносы работодателей установлены статьей 425 НК РФ и составляют 30%.

Так как компания осуществляет деятельность в области ИТ, то она может претендовать на льготную ставку (подп. 3 п. 1 ст. 427 НК РФ) равную 7,6%.

Таким образом, с выплат в пределах МРОТ (минимального размера оплаты труда) уплачиваю взносы – 30%, с суммы свыше МРОТ уплачиваются взносы – 7,6%. МРОТ на 2023 год составляет 16242 рубля.

Данные по суммам отчислений на страховые взносы приведены в таблице 16.

Таблица 16 – Отчисления на страховые взносы

Сотрудник	Заработная плата, рублей	Взнос по единой ставке, рублей	Сумма, подлежащая пониженной ставке по взносу, рублей	Взнос по пониженной ставке, рублей	Общая сумма взноса с сотрудника за месяц, рублей
Руководитель проекта	80000	5872,6	44758	4325,6	9198,2
Программист	60000	5872,6	39758	3945,6	8818,2
Тестирующий	40000	5872,6	6758	1285,6	6158,2

Затраты на заработную плату и отчисления на страховые взносы представлены в таблице 17.

Таблица 17 – Затраты на заработную плату

Сотрудник	Заработная плата, рублей	Общая сумма отчислений, рублей	Всего за 1 месяц, рублей	Всего за 3 месяца, рублей
Руководитель проекта	80000	9198,2	89198,2	267 594,6
Программист	60000	8818,2	68818,2	206 454,6
Тестирующий	40000	6158,2	46158,2	138 474,6
Итого:	180000	24 174,6	204174,4	612 523,2

Затраты на заработную плату команды проекта по созданию программного приложения составят 612 523,2 рублей из которых общая сумма отчислений на страховые взносы разработчиков составит 24 174,6 рублей, а заработная плата составит 636 697,8 рублей.

Затраты на обслуживание включают следующие статьи:

- амортизация оборудования, составляющая 20% от стоимости оборудования;
- обслуживание оборудования, составляющее 15% от амортизационных отчислений;
- затраты на электроэнергию, включающие затраты на электроэнергию персональных компьютеров, оргтехники и бытовых приборов;
- расходы на содержание помещения;
- накладные расходы.

Амортизация оборудования составляет 20% от стоимости оборудования. Было приобретено 5 персональных компьютеров по 51000 рублей. Таким образом затраты на амортизацию оборудования в месяц составят:

$$\frac{(5 * 51000) * 0,2}{12} = 4\,250 \text{ руб.}$$

Обслуживание оборудования составляет 15% от амортизационных отчислений.

Таким образом затраты на обслуживание оборудования составят:

$$4\,250 * 0,15 = 637,5 \text{ руб.}$$

Предположительно в день по 8 часов будут работать 3 ПК:

$$150 * 8 * 63 * 5 = 378 \text{ кВт.}$$

Таким образом, за 63 дня разработки программного приложения будет использовано 378 кВт электроэнергии на питание ноутбуков, а также 2 кВт на прочие приборы:

$$378 + 2 = 380 \text{ кВт.}$$

Накладные расходы составляют 75% от заработной платы. Таким образом, затраты на накладные расходы составляют:

$$612\,523,2 * 0,75 = 459\,392,4 \text{ руб.}$$

Результат расчетов по статьям затрат на обслуживание приведены в таблице 25.

Себестоимость разработки определяется как сумма затрат, включающая все статьи затрат на реализацию проекта, представленное в таблице 18.

Таблица 18 – Затраты на обслуживание

Статья затрат	Показатель отчислений	Сумма отчислений в месяц, рублей	Итоговая сумма отчислений за период разработки, рублей
Амортизация оборудования	20%	4250	12750
Обслуживание оборудования	15%	637,5	1912,5
Статья затрат	Показатель	Сумма	Итоговая сумма отчислений

Статья затрат	Показатель отчислений	Сумма отчислений в месяц, рублей	Итоговая сумма отчислений за период разработки, рублей
	отчислений	отчислений в месяц, рублей	за период разработки, рублей
Затраты на электроэнергию	2,97 на 1 кВт/ч	1128,6	3385,8
Накладные расходы	75%	153130,8	459392,4
Аренда офиса	-	15000	45000
Итого, рублей		174146,9	522440,7

Таблица 19 – Общая сумма затрат

Статья затрат	Сумма отчислений, рублей
Приобретение оборудования	509000
Приобретение лицензионного программного обеспечения	30999
Затраты на заработную плату	612 523,2
Затраты на дополнительные материалы	54000
Затраты на обслуживание	522440,7
Итого:	1728962,9

Таким образом, сумма затрат на разработку проекта составит 1728962,9 рублей и будет соответствовать себестоимости программного приложения контроля выполнения задач.

6.3 Оценка эффективности программы для интерактивного тестирования IT специалистов

Необходимость и целесообразность реализации инвестиционных проектов осуществляется путем оценки их экономической эффективности.

Экономическая эффективность инвестиционных проектов характеризуется системой показателей, отражающих соотношение результатов (притока, поступлений, дохода) и затрат (оттока, выплат) от операционной, инвестиционной и финансовой деятельности.

За счет внедрения и использования программы для интерактивного тестирования IT специалистов также ожидаются следующие неисчисляемые эффекты:

- увеличение скорости проведения собеседований;
- минимизация ошибок, вызванных человеческим фактором;
- повышение эффективности оценки кандидатов;
- упрощение проведения собеседований;
- снижение нагрузки на сотрудников.

6.4 Анализ рисков при разработке и внедрении программы для интерактивного тестирования IT специалистов

При разработке и внедрении программы для интерактивного тестирования IT специалистов необходимо учитывать и анализировать некоторые риски.

Технические риски связаны с проблемами совместимости программы с различными операционными системами и версиями браузеров, а также с необходимыми зависимостями и инструментами разработки. Также возможны ошибки и несовершенства в программе, которые могут привести к неправильным результатам или сбоям.

К рискам в вопросах безопасности можно отнести тот факт, что программа может стать объектом хакерской атаки или взлома, особенно если она содержит конфиденциальную или чувствительную информацию. Необходимо предусмотреть меры защиты данных и проверки на уязвимости.

Если программа не будет правильно протестирована на различных сценариях использования, уровень надежности и точности результатов может

быть недостаточным. Необходимо провести полное и детальное тестирование программы.

Недостаточное понимание потребностей пользователей также может стать проблемой. Если программа не будет удовлетворять потребностям ИТ специалистов или отраслевым стандартам, она может быть непригодной к использованию. Необходимо провести анализ требований и консультации с пользователями.

Разработка и внедрение программы требуют времени, денег и ресурсов. Если не будет достаточного финансирования или не будет выделены достаточные ресурсы, разработка может быть затянута или качество программы может быть недостаточным.

Также ИТ отрасль быстро развивается, и новые технологии и требования могут возникнуть по ходу разработки программы. Необходимо быть готовым к изменениям и адаптироваться к новым требованиям и технологиям.

Существует также проблема масштабируемости: если программу будут использовать большое количество пользователей или она будет использоваться на разных инфраструктурах, возможны проблемы с производительностью и масштабируемостью. Необходимо учесть это при разработке и тестировании программы.

Недостаточное количество документации о программе или недостаточное обучение пользователей могут стать причиной низкой эффективности использования программы. Необходимо предоставить достаточную документацию и обучение пользователям.

При предпроектном анализе должны быть раскрыты все риски, а также пути их минимизации. В конце должен быть составлен обобщающий отчет по всем рискам.

7 Общие принципы обеспечения безопасности на рабочем месте оператора программного приложения для интерактивного тестирования IT специалистов

7.1 Актуальность вопроса безопасности человека на рабочем месте

Вопросы безопасности на рабочем месте являются крайне актуальными и важными, поскольку небезопасные условия труда могут привести к серьезным последствиям для работников и организации в целом. Несчастные случаи на производстве или офисе могут вызвать травмы, инвалидность или даже смерть сотрудников.

От уровня безопасности на рабочем также может зависеть на производительность и эффективность работы. Если сотрудники заботятся о своей безопасности и не сталкиваются с рисками, они могут работать более продуктивно и качественно. Вместе с тем, соблюдение правил и мер безопасности на рабочем месте может снизить затраты на несчастные случаи, медицинское обслуживание и правовую ответственность.

Безопасность на рабочем месте включают в себя такие аспекты, как предотвращение пожаров и взрывов, защита от химических, биологических и радиационных опасностей, предотвращение падений, употребление наркотиков и алкоголя на рабочем месте, эргономика и безопасность работы с инструментами и оборудованием, а также обеспечение безопасности информации и цифровых технологий.

В современном мире с постоянными изменениями в технологиях и рабочих процессах, актуальность вопросов безопасности на рабочем месте становится еще более значимой. Организации должны применять передовые методы и технологии для обеспечения безопасной и здоровой рабочей среды, а сотрудники должны

быть обучены и осведомлены о правилах и процедурах безопасности для минимизации рисков и улучшения своего благополучия и производительности.

7.2 Требования СанПиН при работе с компьютером

СанПиН (Санитарные правила и нормы) – это документ, устанавливающий санитарные требования к условиям труда работников в различных сферах деятельности, включая работу с компьютером.

Гигиенические требования к ЭВМ и работе с ними устанавливаются СанПиН 2.2.2/2.4.1340-03 от 30.05.2003 [6]. Статьи 107 и 109 Трудового кодекса Российской Федерации [7] предписывают выделять время на отдых сотрудникам в течение рабочего дня.

Некоторые из основных требований СанПиН при работе с компьютером включают:

- оптимальное освещение рабочего места: рекомендуется использовать освещение соответствующей интенсивности и равномерности, чтобы избежать несоответствия яркости экрана и окружающей среды;

- рациональное размещение рабочего места: высота стола, стула и монитора должна быть настроена таким образом, чтобы соблюдать правильное положение тела и предотвращать напряжение спины, шеи и глаз;

- правильная организация экрана: рекомендуется настраивать яркость, контрастность и цветовую гамму монитора в соответствии с требованиями пользователя для более комфортного зрения;

- разумные паузы и физические упражнения: рекомендуется делать перерывы через определенные промежутки времени для отдыха глаз и мышц, а также для выполнения физических упражнений, чтобы снизить мускульное напряжение и предотвратить сухость глаз;

- организация работы: необходимо установить режим работы, включающий перерывы на отдых и физические упражнения, а также предусмотреть

возможность выполнения разнообразных задач для снижения монотонности работы с компьютером;

– обучение и консультации: работодатели должны предоставлять своим сотрудникам обучение и консультации по соблюдению санитарных требований при работе с компьютером.

7.3 Требования безопасности при работе с электроприборами

При работе с электроприборами необходимо соблюдать следующие требования безопасности:

– электроприбор должен иметь надлежащую изоляцию и не поврежденные провода или штепсельную вилку, нельзя использовать электроприборы, у которых обнаружены повреждения;

– устройства, которые требуют заземления, должны быть правильно подключены к специальным заземляющим проводам;

– влага может представлять угрозу для безопасности и привести к поражению электрическим током, поэтому необходимо избегать использования электроприборов во влажной среде или рядом с водой;

– нельзя перегружать электроприборы, подсоединяя их к слишком мощным источникам электроэнергии или допуская перегрузку сети, так как это может привести к перегреву и повреждению прибора, а также стать причиной пожара;

– перед подключением или отключением электроприбора нужно убедиться, что он выключен и отсоединен от источника электроэнергии, а также запрещается трогать провода или штепсельные гнезда мокрыми или грязными руками;

– электроприборы должны храниться в безопасном и сухом месте, в недоступности для детей;

– электроприборы должны регулярно проходить техническое обслуживание для обнаружения и устранения возможных неисправностей или повреждений;

– перед использованием электроприбора, ознакомьтесь с инструкциями по его эксплуатации, следуйте инструкциям и не используйте приборы для работы, для которых они не предназначены.

Соблюдение этих требований безопасности поможет избежать аварийных ситуаций, повреждений электроприборов и предотвратить возможность получения электрического удара или пожара.

7.4 Влияние ПЭВМ на физиологию человека

ПЭВМ (персональные электронные вычислительные машины) имеют определенное влияние на физиологию человека. Ниже приведены некоторые основные аспекты.

– визуальная нагрузка. Работа за компьютером может привести к усталости глаз и неправильной позиции тела, вызывая напряжение мышц шеи, спины и рук. Длительное использование монитора также может вызвать сухость глаз и другие проблемы со зрением;

– сидячий образ жизни. Частое работа за компьютером может привести к сидячему образу жизни, который связан со множеством здоровотворных проблем, включая ожирение, слабые мышцы и проблемы с сердечно-сосудистой системой;

– повышенный стресс. Работа под давлением, сроки и требования могут вызывать у людей чрезмерный стресс. Это может привести к повышенному уровню тревожности, расстройству сна и другим психологическим проблемам;

– экранные времена. Значительное время, проведенное за компьютером, может привести к увеличению времени, проводимому перед экранами. Это может негативно сказаться на снах, социальном взаимодействии и общем физическом здоровье;

– электромагнитные излучения. Некоторые люди могут быть чувствительны к электромагнитным излучениям, которые испускают компьютеры и другая

электроника. Это может вызывать головные боли, раздражительность и другие симптомы;

Для уменьшения негативного воздействия ПЭВМ на физиологию человека, важно принимать следующие меры:

- правильно настроить рабочее место с удобным стулом, подлокотниками и подставкой для ног;
- правильно распределить время и делать перерывы, чтобы избежать чрезмерной усталости и напряжения глаз;
- постоянно следить за позицией тела и правильно пользоваться клавиатурой и мышью для минимизации нагрузки на шею, спину и руки;
- устанавливать фильтры или использовать специальные очки для снижения напряжения глаз и сухости;
- регулярно заниматься физическими упражнениями, чтобы компенсировать сидячий образ жизни и укрепить мышцы;
- обратиться к специалистам по эргономике или здоровью на работе для получения советов по улучшению рабочих условий.

Важно соблюдать меры предосторожности и заботиться о своем физическом и психологическом здоровье при работе с ПЭВМ, чтобы минимизировать негативное воздействие на организм.

7.5 Техника безопасности при работе с персональным компьютером

При работе с персональным компьютером важно соблюдать определенные техники безопасности, чтобы минимизировать риск возможных травм и проблем со здоровьем. Существует ряд основных правил по БЖД при работе с ПК.

- позиция тела. Необходимо соблюдать правильную осанку. Стул должен быть удобным и обеспечивать поддержку спины. Ноги должны быть полностью

на полу или на подставке для ног. Руки должны быть удобно размещены на клавиатуре и мыши, не вызывая напряжения в руках и плечах;

– расположение монитора. Монитор должен быть расположен на уровне глаз или немного ниже, чтобы уменьшить нагрузку на глаза и шею. Яркость и контрастность должны быть отрегулированы так, чтоб чтение было комфортным;

– отведение глаз. Перерывы и упражнение для глаз необходимы, чтоб снять напряжение и предотвратить усталость зрения;

– перерывы. Согласно типовой инструкции по охране труда при работе на ПК (персональном компьютере) ТОИ Р-45-084-01 [8], которая была утверждена Приказом Минсвязи Российской Федерации №162 от 02.07.2001, деятельность на ЭВМ делится на 3 группы: группа А – чтение информации с монитора подразумевает два перерыва по 15 минут, без учета обеденного перерыва; группа Б – набор текста подразумевает перерывы по 10 минут раз в час; группа В – творческий труд подразумевает перерывы по 15 минут раз в час;

– клавиатура и мышь. Клавиатуру и мышь нужно использовать правильно, не нажимая слишком сильно и не напрягая пальцы. Расположение клавиатуры и мыши должно быть комфортным, не вызывая перенапряжение запястий или плеч;

– организация рабочего места. Рабочая область свободна от беспорядка и ненужных предметов, чтобы иметь достаточно места для работы;

– регулировка освещения. Обеспечьте хорошее освещение рабочего места. Избегайте слишком яркого или слишком тусклого света, чтобы избежать усталости глаз.

Соблюдение этих правил безопасности поможет предотвратить некоторые проблемы, связанные с работой за компьютером, и поддерживать здоровье во время использования ПК.

Заключение

В ходе выполнения дипломного проекта была изучена предметная область, а именно работа специалистов по тестированию. На основе ее анализа было сформулировано техническое задание для разработки программы для интерактивного тестирования IT специалистов. Затем были выбраны средства разработки, благодаря которым было разработано программное приложение.

Программа для интерактивного тестирования IT специалистов реализована в соответствии с техническим заданием.

В разделе экономики была произведена оценка затрат на разработку и внедрение программы.

В разделе безопасность жизнедеятельности были проанализированы и описаны общие принципы обеспечения безопасности на рабочем месте оператора программы для интерактивного тестирования IT специалистов.

Задание на дипломное проектирование выполнено в полном объеме.

					<i>ПГУ 1.090501.08.001 ПЗ</i>	Лист
						85
Изм.	Лист	№ докум.	Подпись	Дата		

Список использованных источников

1. <https://www.statista.com/outlook/tmo/software/worldwide#revenue>
2. Официальный сайт Visual Studio Code [Электронный ресурс] <https://code.visualstudio.com/> (дата 01.03.2023).
3. Фримен Элизабет, Фриман Эрик – «Изучаем программирование на HTML5». – 2013. – 592с.
4. Макфарланд Д. – «Новая большая книга CSS». – 2016. -720с.
5. Еременко А.В. – «Проектирование баз данных с помощью OpenModelSphere». – 2011. – 73с.
6. Карпова И.П. – «Базы данных. Курс лекций и материалы для практических заданий». – Учебное пособие. – 2013. – 240с.
7. СанПиН 2.2.2/2.4.1340-03. Библиографическая запись. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы. Изд-во стандартов, 2003. – 41 с.
8. Трудовой кодекс Российской Федерации об утверждении перечней вредных и (или) опасных производственных факторов и работ № 302. – 2011.
9. Типовая инструкция по охране труда при работе на персональном компьютере. – 2001. – 3 с.

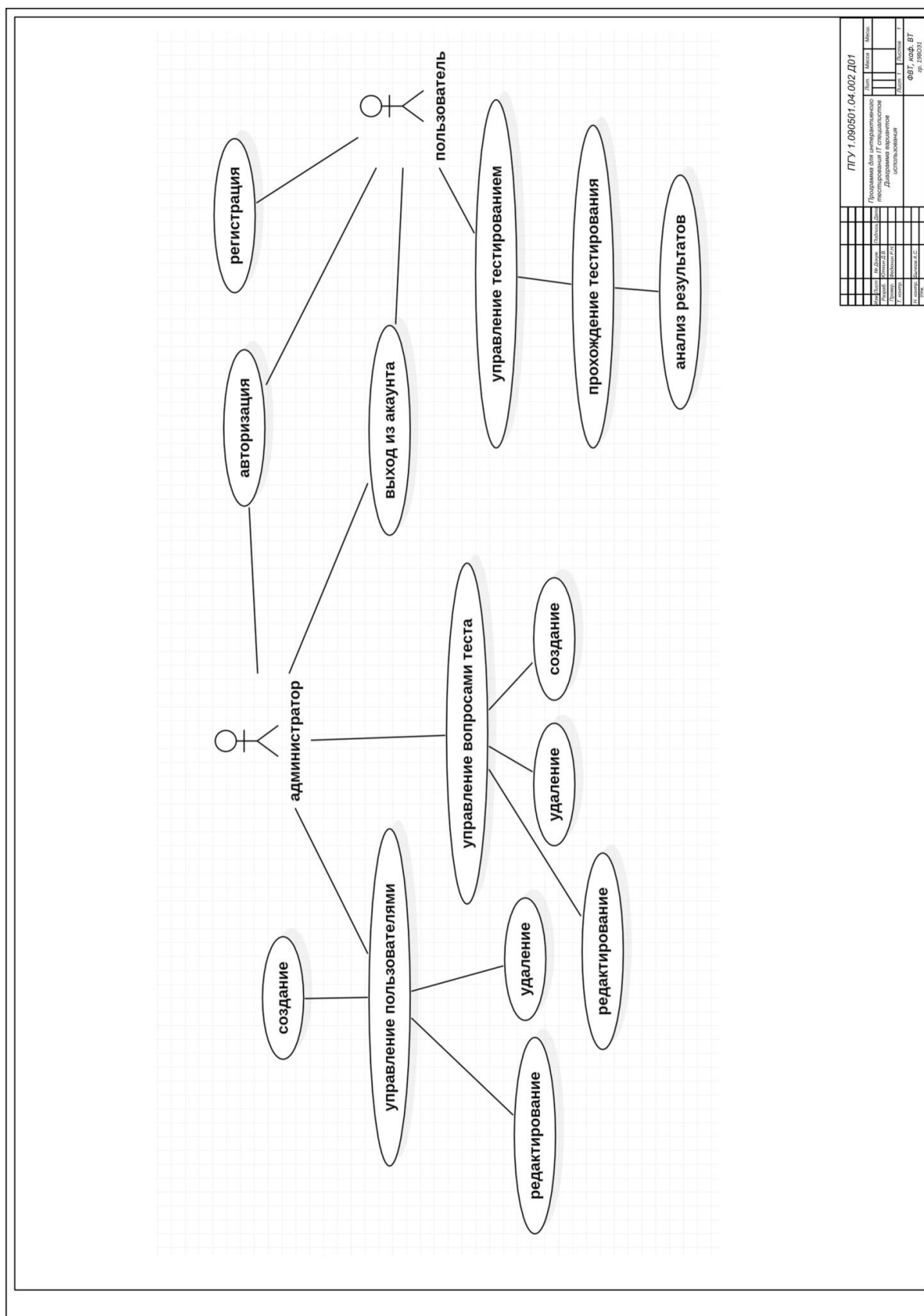
Приложение А

(обязательное)

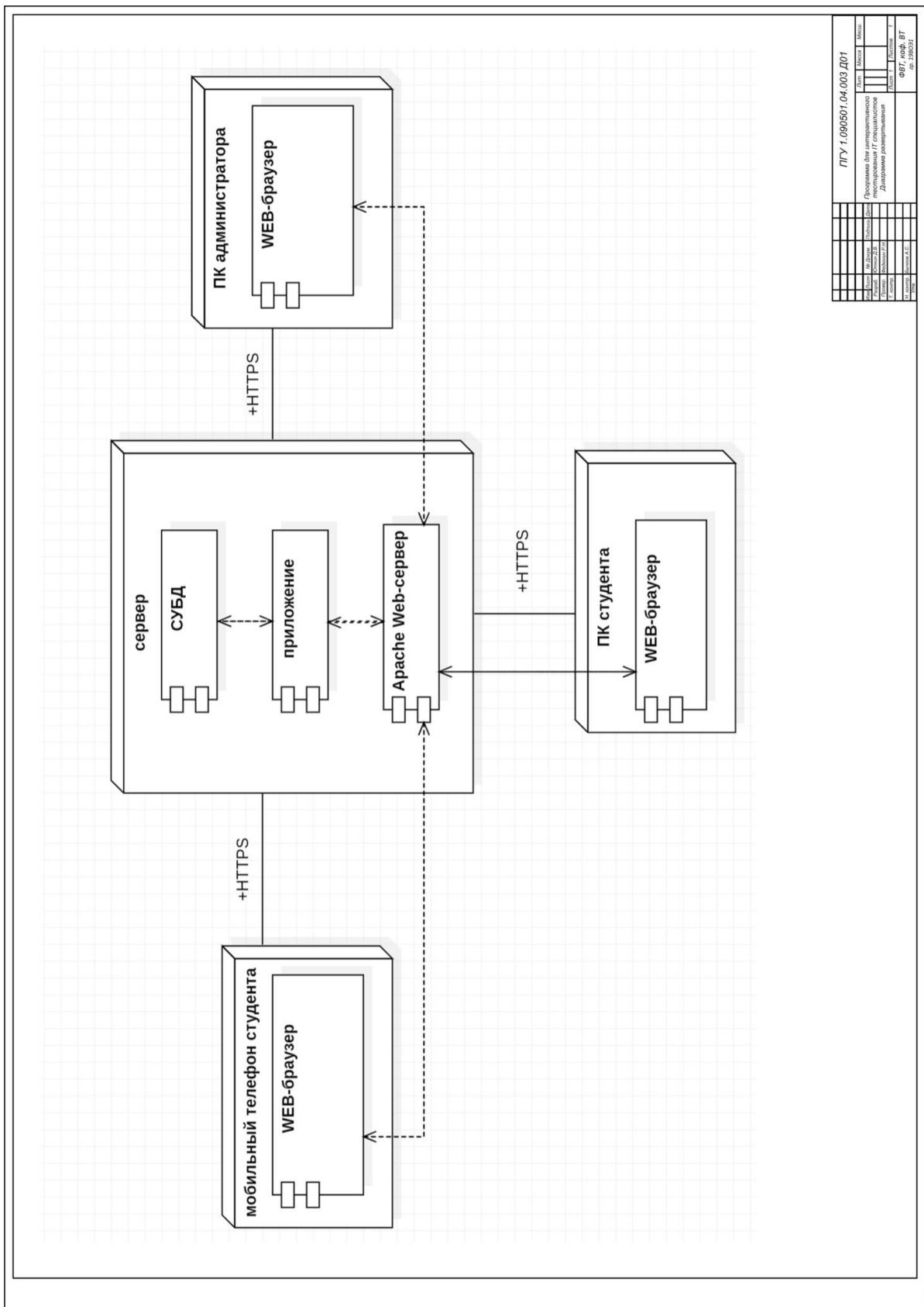
Приложение для квалификационного тестирования специалистов
UML-диаграммы

					ПГУ 1.090501.08.001 ПЗ	Лист
						87
Изм.	Лист	№ докум.	Подпись	Дата		

Приложение А.1 – Диаграмма вариантов использования



Приложение А.2 – Диаграмма развертывания



ПГУ 1.090501.04.003 Д01									
Имя	Ф.И.О.	Подпись	Дата	Лист	Масштаб	Масштаб	Масштаб	Масштаб	Масштаб
Имя	Ф.И.О.	Подпись	Дата	Лист	Масштаб	Масштаб	Масштаб	Масштаб	Масштаб
Имя	Ф.И.О.	Подпись	Дата	Лист	Масштаб	Масштаб	Масштаб	Масштаб	Масштаб
Имя	Ф.И.О.	Подпись	Дата	Лист	Масштаб	Масштаб	Масштаб	Масштаб	Масштаб
Имя	Ф.И.О.	Подпись	Дата	Лист	Масштаб	Масштаб	Масштаб	Масштаб	Масштаб
Имя	Ф.И.О.	Подпись	Дата	Лист	Масштаб	Масштаб	Масштаб	Масштаб	Масштаб
Имя	Ф.И.О.	Подпись	Дата	Лист	Масштаб	Масштаб	Масштаб	Масштаб	Масштаб
Имя	Ф.И.О.	Подпись	Дата	Лист	Масштаб	Масштаб	Масштаб	Масштаб	Масштаб
Имя	Ф.И.О.	Подпись	Дата	Лист	Масштаб	Масштаб	Масштаб	Масштаб	Масштаб

Изм.	Лист	№ докум.	Подпись	Дата

```

    erDiagram
        roles ||--o{ users : "has"
        users ||--o{ quiz_question : "has"
        users ||--o{ quiz_answer : "has"
        users ||--o{ quiz : "has"
        users ||--o{ take_answer : "has"
        quiz_question ||--o{ quiz_answer : "has"
        quiz ||--o{ take_answer : "has"
  
```

roles

PK	RolesID	INTEGER
FK	UsersID	INTEGER
	Roles	VARCHAR

users

PK	UsersID	BIGINT
	FirstName	VARCHAR
	LastName	VARCHAR
	Mobile	VARCHAR
	Email	VARCHAR
	Host	INTEGER
	RegisteredAt	DATETIME
	LastLogin	DATETIME
	Intro	TEXT
	Profile	TEXT
	Password	VARCHAR

quiz_question

PK	QuizQuestionID	BIGINT
FK	QuizID	BIGINT
	Title	VARCHAR
	Active	BOOLEAN
	Level	INTEGER
	Score	INTEGER
	CreatedAt	DATETIME
	UpdatedAt	DATETIME
	Content	TEXT

quiz_answer

PK	QuizAnswerID	BIGINT
FK	QuizID	BIGINT
FK	QuestionID	BIGINT
	Correct	BOOLEAN
	CreatedAt	DATETIME
	UpdatedAt	DATETIME
	Content	TEXT

quiz

PK	TestID	BIGINT
	Title	VARCHAR
	Summary	VARCHAR
	Type	SMALLINT
	Score	BIGINT
	Published	BOOLEAN
	CreatedAt	DATETIME
	UpdatedAt	DATETIME
	StartsAt	DATETIME
	EndsAt	DATETIME
	Content	TEXT
FK	HostID	BIGINT

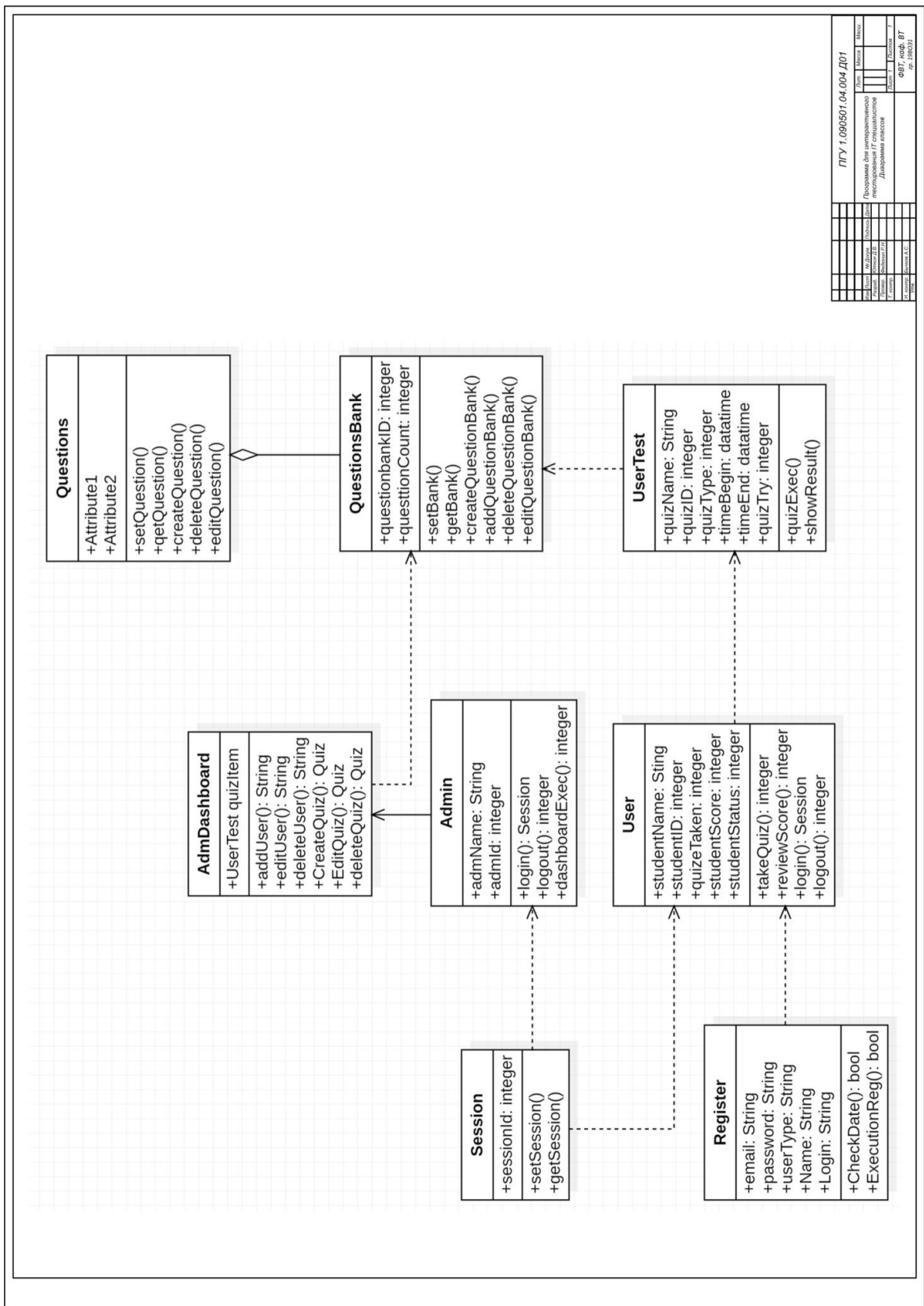
take_answer

PK	TakeAnswerID	BIGINT
FK	TakeID	BIGINT
FK	AnswerID	BIGINT
	CreatedAt	DATETIME
	UpdatedAt	DATETIME
	Content	TEXT

take

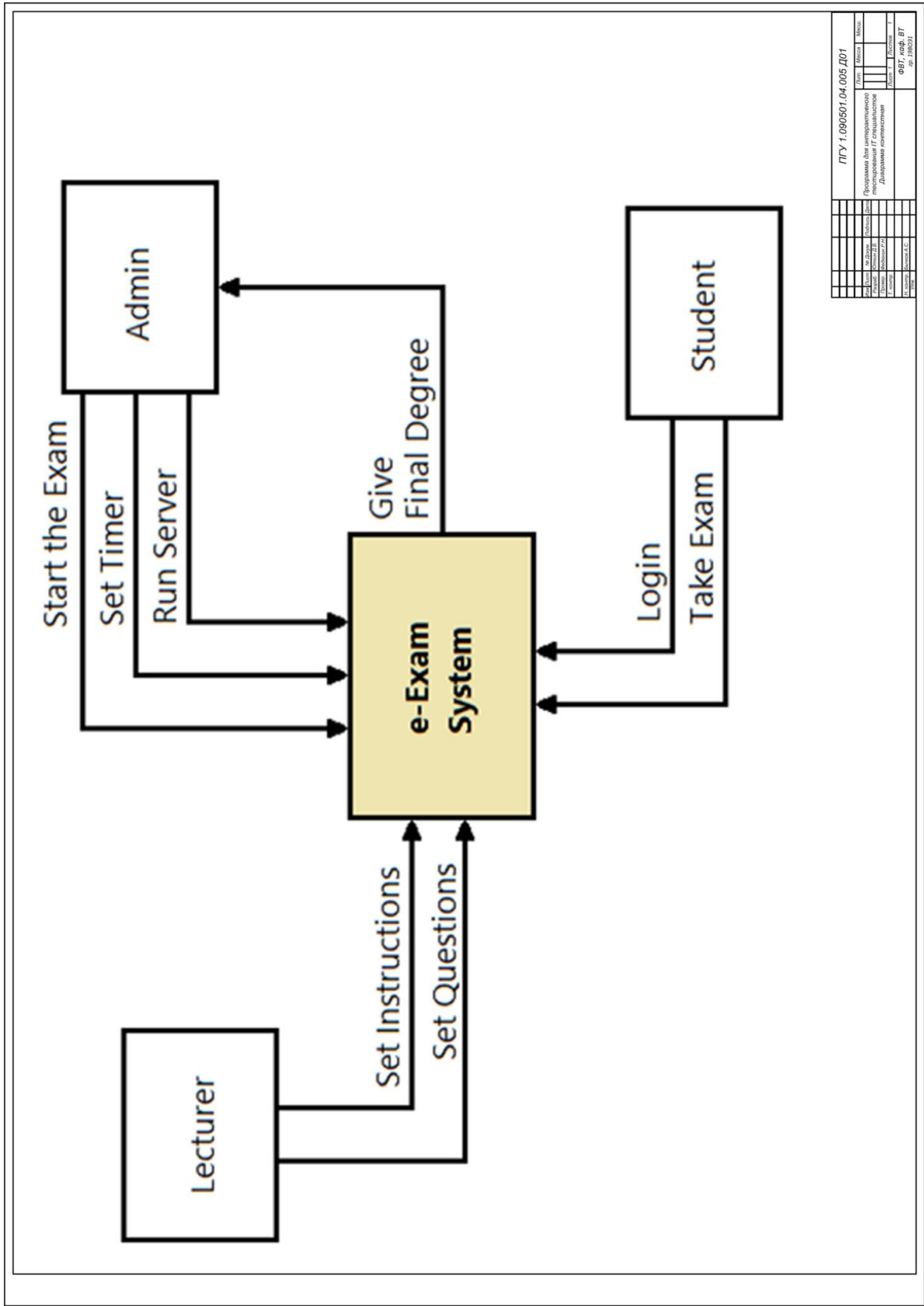
PK	TakeID	BIGINT
FK	QuizAnswerID	BIGINT
FK	UserID	BIGINT
FK	QuizID	BIGINT
FK	Status	INTEGER
	Score	BIGINT
	CreatedAt	DATETIME
	UpdatedAt	DATETIME
	StartedAt	DATETIME
	FinishedAt	DATETIME
	Remark	TEXT

Приложение А.4 – Диаграмма классов



																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					</
--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	----

Приложение А.4 – Диаграмма контекстная



Приложение Б

(обязательное)

Приложение для квалификационного тестирования специалистов

Листинг программы

					ПГУ 1.090501.08.001 ПЗ	Лист
						93
Изм.	Лист	№ докум.	Подпись	Дата		

Модуль Login.php

```
<?php

require 'config.php';

if(isset($_SESSION['login_id'])) {

    header('Location: http://localhost:3000/');

    exit;

}

require 'vendor/autoload.php';

$client = new Google_Client();

$client->setClientId('584734149097-
7aleubruh4qc5g154h56b0mb9abv73tt.apps.googleusercontent.com');

$client->setClientSecret('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');

$client->setRedirectUri('http://localhost/quiz/login.php');

$client->addScope("email");

$client->addScope("profile");

if(isset($_GET['code'])):

    $token = $client->fetchAccessTokenWithAuthCode($_GET['code']);

    if(!isset($token["error"])){

        $client->setAccessToken($token['access_token']);

        // getting profile information

        $google_oauth = new Google_Service_Oauth2($client);

        //echo "<pre>"; print_r($google_oauth);die;
```

					ПГУ 1.090501.08.001 ПЗ	Лист
						94
Изм.	Лист	№ докум.	Подпись	Дата		

```

        $google_account_info = $google_oauth->userinfo->get();

        $_SESSION['login_id']
=mysqli_real_escape_string($db_connection,      $google_account_info-
>id);

        header('Location: http://localhost:3000/');

        exit;

    }

    else{

        header('Location: login.php');

        exit;

    }

else:

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-
scale=1.0">

    <meta http-equiv="X-UA-Compatible" content="ie=edge">

    <title>Login</title>

    <link rel="stylesheet" href="index.css">

```

```

</head>

<body>

    <div class="_container">

        <h2 class="heading">Login</h2>

    </div>

    <div class="_container btn">

        <a type="button" class="login-with-google-btn" href="<?php
echo $client->createAuthUrl(); ?>">

    </a>

</div>

</body>

</html>

<?php endif; ?>

```

Модуль Logout.php

```

<?php

session_name("logout")

$id = $_SESSION['login_id'];

if (ini_get("session.use_cookies")) {

    $params = session_get_cookie_params();

```

```

        setcookie(session_name(), '', time() - 42000,

            $params["path"], $params["domain"],

            $params["secure"], $params["httponly"]

        );
    }

    $_SESSION = array();

    session_destroy();

    header("Location: login.php");

    exit;

?>

```

Модуль TestCore.js

```

import React, { useState } from "react";

import axios from "axios";

// default question object define

const quiz = {

    topic: "Sports",

    totalQuestions: 10,

    perQuestionScore: 1,

    questions: [

```

```

    {
      question: "",
      choices: [],
      correctAnswer: 5,
    },
  ],
};

// base API url

const baseUrl = "http://localhost/quiz/ ";

// component

const Quiz = () => {
  const [activeQuestion, setActiveQuestion] = useState(0);
  const [selectedAnswer, setSelectedAnswer] = useState("");
  const [showResult, setShowResult] = useState(false);
  const [selectedAnswerIndex, setSelectedAnswerIndex] =
    useState(null);

  const [selectedTopic, setSelectedTopic] = useState(0);
  const [selectedTopicquiz, setSelectedTopicquiz] = useState(quiz);
  const [result, setResult] = useState({
    score: 0,
    correctAnswers: 0,
    wrongAnswers: 0,
  });

```

```

});

const [topics, settopics] = useState([]);

const [Answered] = useState([]);

const { questions } = selectedTopicquiz;

const { question, choices, correctAnswer } =
questions[activeQuestion];

// fetch topics from database

React.useEffect(() => {

    axios.get(baseUrl + "topic/read.php").then((response) => {

        settopics(response.data.records);

    });

}, []);

const getQuestions = async (selectedTopic) => {

    const post = { topic: selectedTopic };

    try {

        const res = await axios.post(baseUrl + "questions/read.php",
post);

        setSelectedTopicquiz(res.data.records);

    } catch (e) {

        alert(e);

    }

};

```

```

const addQuestions = async (Answers) => {

    const post = { allAnswer: Answers };

    try {

        await axios.post(baseUrl + "questions/add.php", post);

    } catch (e) {

        alert(e);

    }

};

const onTopicSelected = (topic, index) => {

    if (selectedAnswer === "") {

        setSelectedTopic(index);

        getQuestions(index);

    }

};

const onClickNext = () => {

    setSelectedAnswerIndex(null);

    Answered.push(selectedAnswerIndex);

    if (activeQuestion === questions.length - 1) {

        addQuestions(Answered);

    }

    setResult((prev) =>

        selectedAnswer

```

```

    ? {

        ...prev,

        score: prev.score + 1,

        correctAnswers: prev.correctAnswers + 1,

    }

    : { ...prev, wrongAnswers: prev.wrongAnswers + 1 }

);

if (activeQuestion !== questions.length - 1) {

    setActiveQuestion((prev) => prev + 1);

} else {

    setActiveQuestion(0);

    setShowResult(true);

}

};

const onAnswerSelected = (answer, index) => {

    setSelectedAnswerIndex(answer.AnsID);

    if (answer.AnsID === correctAnswer) {

        setSelectedAnswer(true);

    } else {

        setSelectedAnswer(false);

    }

};

```

```

    const addLeadingZero = (number) => (number > 9 ? number :
`0${number}`);

    const addLevel = (number) =>

        number <= 3 ? `Beginners` : number <= 7 ? `Intermediate` :
`professional`;

    return (

        <div className="quiz-container">

            {selectedTopic > 0 ? (

                !showResult ? (

                    <div>

                        {

                            <div>

                                <div>

                                    <span className="active-question-no">

                                        {addLeadingZero(activeQuestion + 1)}

                                    </span>

                                    <span className="total-question">

                                        /{addLeadingZero(questions.length)}

                                    </span>

                                    <span className="active-question-no">

                                        -{addLevel(activeQuestion + 1)}

                                    </span>

                                </div>

                            </div>

                        }

                    )

                )

            }

        </div>

```

```

<h2>{question}</h2>

<ul>

  {choices.length &&

    choices.map((answer, index) => (

      <li

        onClick={() => onAnswerSelected(answer,
index) }

        key={answer.ans}

        className={

          selectedAnswerIndex === answer.AnsID

            ? "selected-answer"

            : null

          }

      >

        {answer.ans}

      </li>

    ) ) }

</ul>

<div className="flex-right">

  <button

    onClick={onClickNext}

    disabled={selectedAnswerIndex === null}

```

```

        >

        {activeQuestion === questions.length - 1

        ? "Finish"

        : "Next"}

    </button>

</div>

</div>

}

</div>

) : (

<div className="result">

    <h3>Result</h3>

    <p>

        Total Question: <span>{questions.length}</span>

    </p>

    <p>

        Total Score:<span> {result.score}</span>

    </p>

    <p>

        Correct Answers:<span> {result.correctAnswers}</span>

    </p>

    <p>

```

```

Wrong Answers:<span> {result.wrongAnswers}</span>

</p>

</div>

)

) : (

<div>

<ul id="menu">

{topics.length &&

topics.map((topic, ind) => (

<li

onClick={ () =>

onTopicSelected(topic.topic_name, topic.topicID)

}

key={topic.topicID}

disabled={selectedAnswerIndex === null}

className={

selectedTopic === topic.topicID ? "selected-
topic" : null

}

>

{topic.topic_name}

</li>

)}}

```

```

        </ul>

    </div>

    ) }

</div>

);

};

export default Quizze;

```

Модуль student.php

```

class Student extends Model

{

    protected $table = 'students';

    protected $fillable = [

        'name',

        'surname',

        'email',

        'is_blocked',

    ];

    public function paginate(int $items, array $filter):
LengthAwarePaginator

    {

        $studentQuery = Student::query();

        if (isset($filter['name'])) {

```

```

        $studentQuery = $studentQuery

        ->where('name', 'like', '%'.$filter['name'].'%')

        ->orWhere('surname', 'like',
'%'.$filter['name'].'%');    }

    return $studentQuery

    ->orderBy('id')

    ->paginate($items);

}

public function getById(int $id)

{

    return Student::query()

        ->whereKey($id)

        ->first();

}

public function result(): HasOne

{

    return $this->hasOne(Result::class, 'student_id', 'id')->orderByDesc('id');

}

public function getId(): int

{

    return $this->id;

}

```

```

public function getName(): string

{

    return $this->name;

}

public function getSurname(): string

{

    return $this->surname;

}


public function getEmail(): string

{

    return $this->email;

}


public function isBlocked(): bool

{

    return $this->is_blocked;

}


public function getCreatedAt(): string

{

    return $this->created_at;

}

```

```

    }

    public function getUpdatedAt(): string

    {

        return $this->updated_at;

    }

    public static function getInstance(): Student

    {

        return new static();

    }

}

```

Модуль administrator.php

```

class Admin extends Model

{

    protected $table = 'users';

    protected $fillable = [

        'name',

        'surname',

        'email',

        'password',

        'token',

        'status',

    ]

```

```

];

protected $hidden = [

    'password',

    'token',

];

public static function query(): Builder

{

    return parent::query()

        ->where('status', '=', true);

}

public function getUserByToken(string $token)

{

    return User::query()

        ->where('token', '=', $token)

        ->first();

}

public function getUser(string $email)

{

    return User::query()

        ->where('email', '=', $email)

        ->first();

}

```

```

public function create(array $items)

{

    return User::query()->create($items);

}

public function getId(): int

{

    return $this->id;

}

public function getName(): string

{

    return $this->name;

}

public function getSurname(): string

{

    return $this->surname;

}

public function getEmail(): string

{

    return $this->email;

}

public function getPassword(): string

{

```

```

        return $this->password;

    }

    public function getToken(): string

    {

        return $this->token;

    }


    public function getCreatedAt(): string

    {

        return $this->created_at;

    }

    public function getUpdatedAt(): string

    {

        return $this->updated_at;

    }

    public static function getInstance(): User

    {

        return new static();

    }

}

```

Приложение В

(обязательное)

Приложение для квалификационного тестирования специалистов

Презентация

					ПГУ 1.090501.08.001 ПЗ	Лист
						113
Изм.	Лист	№ докум.	Подпись	Дата		

ПРОГРАММА ДЛЯ ПРОВЕДЕНИЯ
КВАЛИФИКАЦИОННОГО
ТЕСТИРОВАНИЯ СПЕЦИАЛИСТОВ

Автор работы:

студент гр. 19ВОЭ1 Юткин Даниил Владимирович

Научный руководитель:

к.т.н., доцент Федюнин Роман Николаевич

Пенза, 2024

Изм.	Лист	№ докум.	Подпись	Дата

Цель работы и объекты

Цель работы – реализовать программное приложение предназначенное для тестирования специалистов по направлению «Применение и эксплуатация автоматизированных систем специального назначения».

Приложение необходимо для двух категорий пользователей:

- Преподаватели – пользователи с правом формирования и загрузки тестов;
- Студенты – пользователи с правом использования загруженных тестовых заданий.

Проект является клиент-серверным приложением, выполняющим следующие функции:

- Формирование тестовых заданий, прохождение тестовых заданий, анализ результатов индивидуального тестирования, мультимедийный формат работы.

Задачи приложения

предоставить пользователю вне зависимости от статуса право авторизации и аутентификации.

предоставить администратору (преподавателю):

возможность размещать готовые тесты для студентов.

возможность анализа процесса тестирования студентов.

предоставить авторизованному студенту:

возможность выбрать и пройти тестовое задание.

возможность просмотра результата пройденного тестирования.

Изм.	Лист	№ докум.	Подпись	Дата

Диаграмма вариантов использования

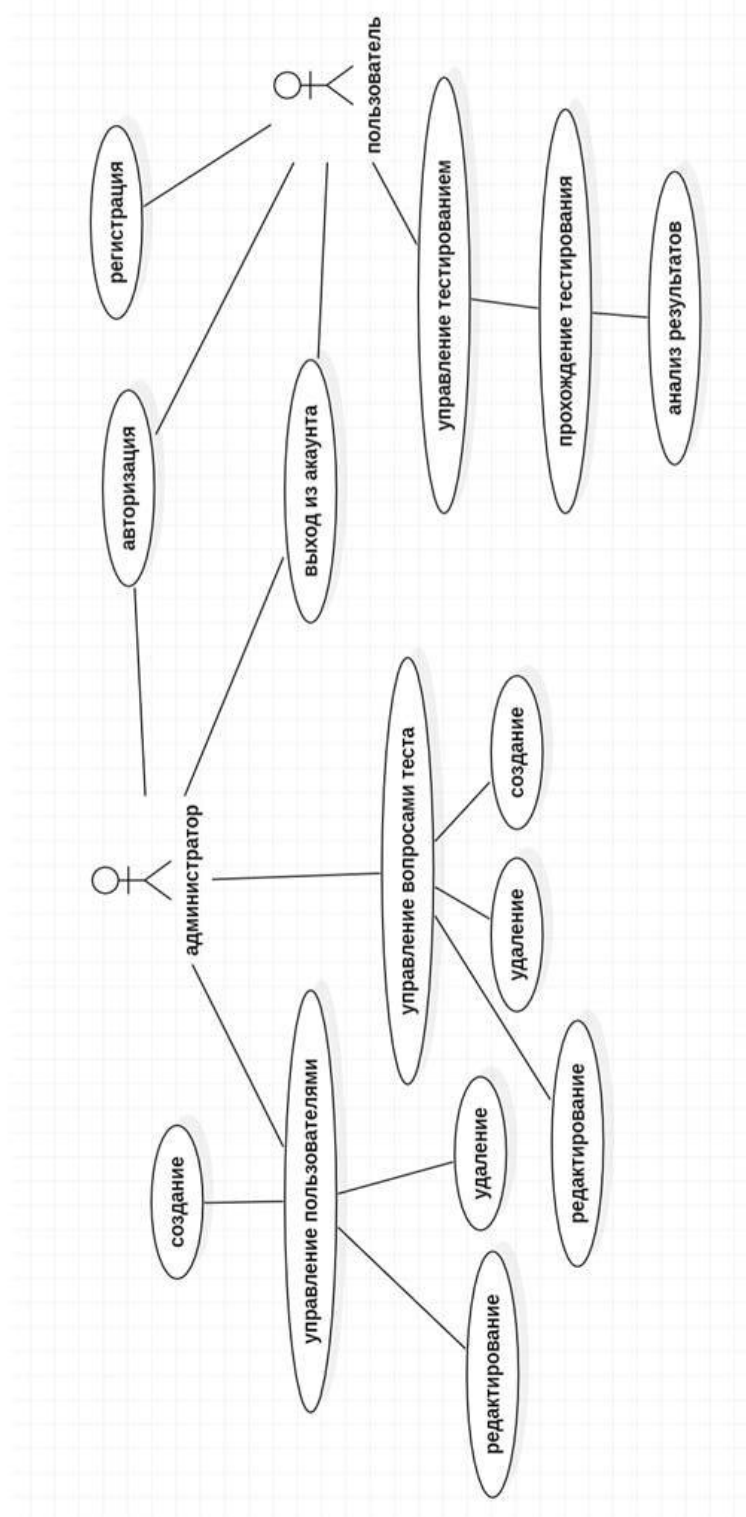
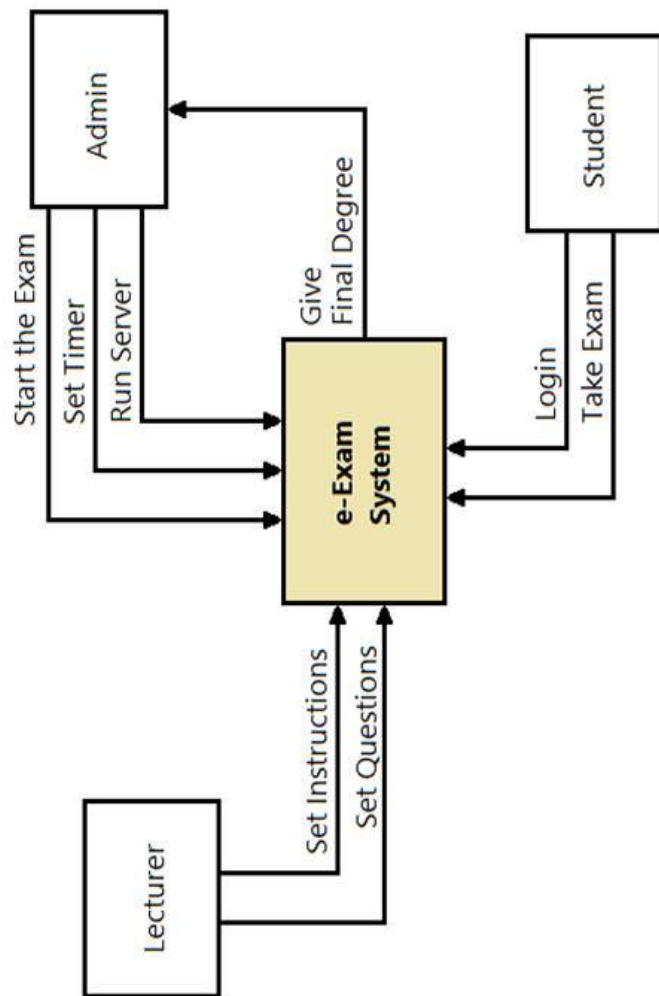


Диаграмма контекстная



Сравнение аналогов

Quizlet, Quizizz, ProProfs Quiz Maker и иное программное обеспечение, которое предлагает широкий спектр функций для создания и проведения тестов IT-специалистов.

1. *StartExam*
2. *Google Forms*
3. *Kahoot*

Программное обеспечение, реализованное в дипломном проекте, полностью в отличие от описанных выше программ является бесплатным, полностью настраиваемым со стороны потребителя, включая тесты и вопросы к ним.

Стек технологий

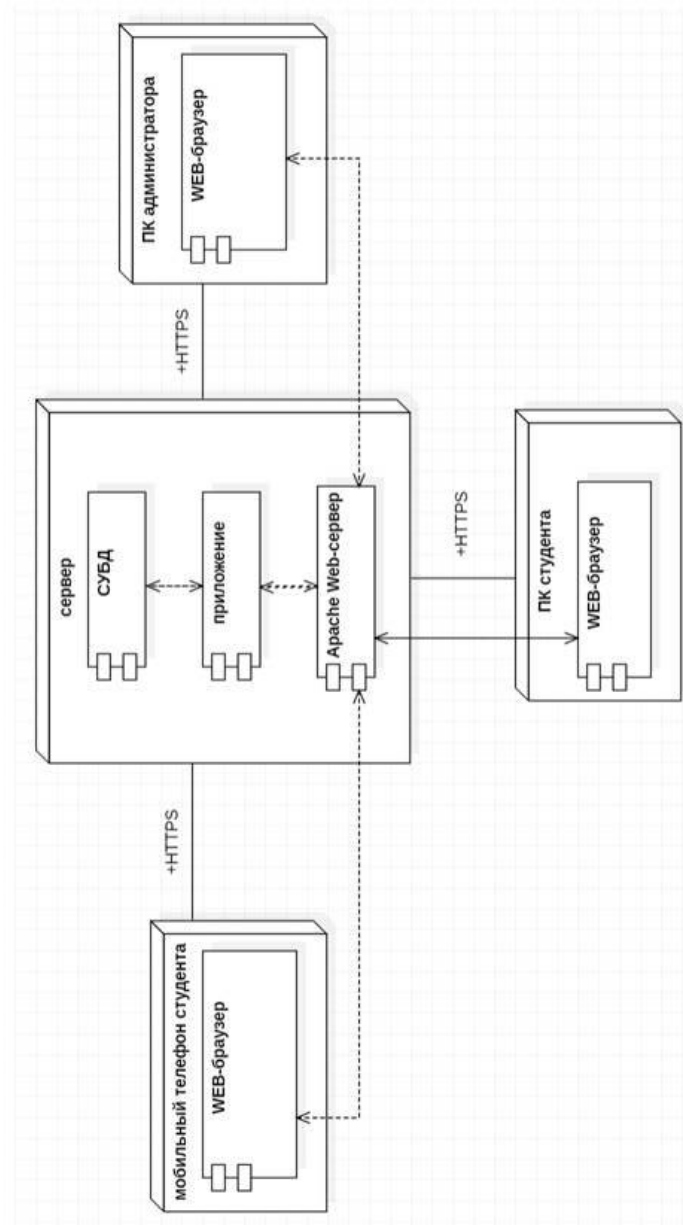


Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090501.08.001 ПЗ

Лист
120

Архитектура приложения



Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090501.08.001 ПЗ

Структура базы данных

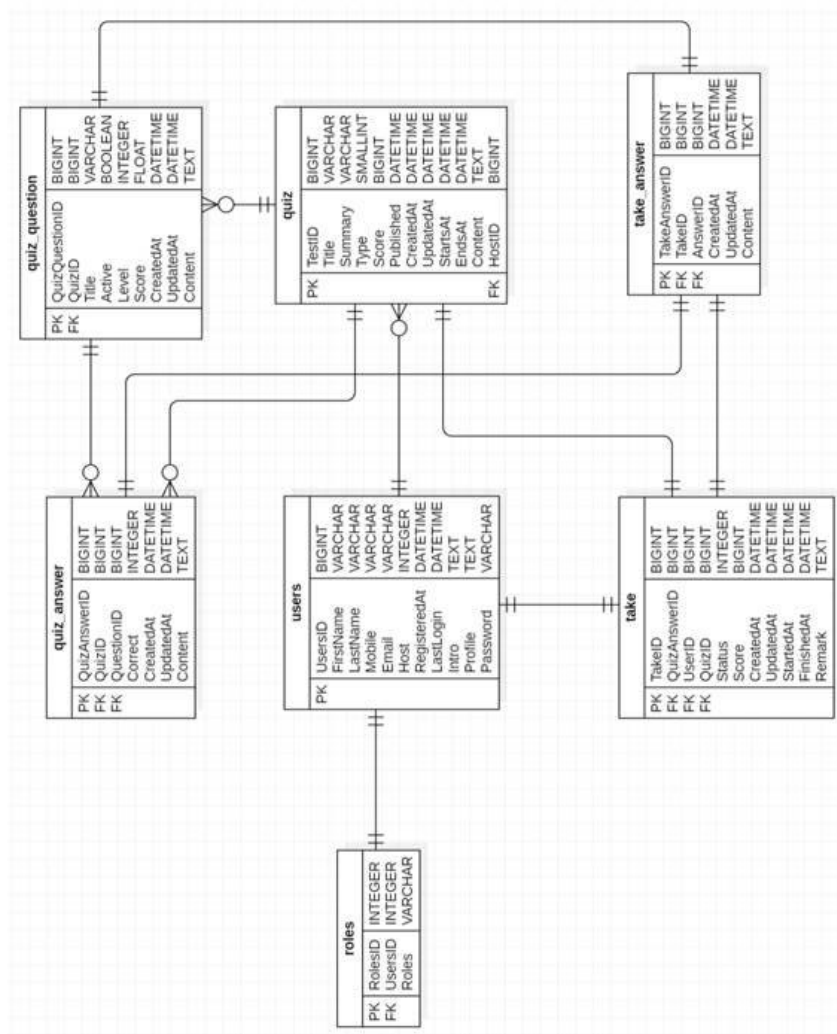
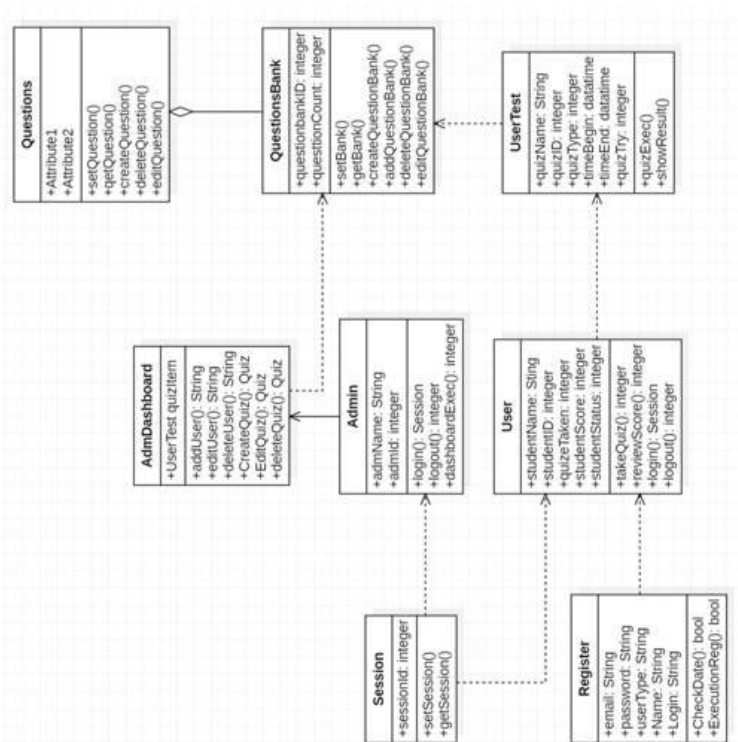


Диаграмма классов



Скриншоты запущенного приложения (студент)

Система онлайн-экзаменов

Вход Регистрация Адаптер

Система онлайн-экзаменов

Полная система онлайн-экзаменов на основе MSQ

Полное имя
Введите полное имя

Введите ваше полное имя (пробелы разрешены)

Имя пользователя
Введите имя пользователя (без пробелов)

Введите полное имя или идентификатор

Email
Введите email

Номер для идентификации пользователя: уникальный номер, не 0000

Система онлайн-экзаменов

Добро пожаловать Богдан Профиль Экзамен Выйти

Поздравляем! Вы только что завершили тест

Проверьте свой результат. Вы также можете проверить правильные ответы

Финальный счет: 10

Посмотреть ответы ✓

Начать тест →

Система онлайн-экзаменов

Добро пожаловать Богдан Профиль Экзамен Выйти

Вы можете начать экзамен

На территории. Нажмите «Начать экзамен», когда будете готовы.



Начать экзамен →

Система онлайн-экзаменов

Добро пожаловать Богдан Профиль Экзамен Выйти

Добро пожаловать на онлайн-экзамен

Проверьте свои знания

Общее количество вопросов: 10

Тип вопроса: Множественный выбор (MCQ)

Продолжить →

Скриншоты запущенного приложения (администратор/преподаватель)

Система онлайн-экзаменов

Домой Управление пользователями Добавить вопрос Управление вопросами

Вход – Панель администратора

Введите email

admin

Пароль

Вход

Система онлайн-экзаменов

Домой Управление пользователями Добавить вопрос Управление вопросами

Добавить вопросы

Вопрос №

:

11

Вопрос

:

Введите вопрос

Ответ 1

:

Введите ответ №1

Ответ 2

:

Введите ответ №2

Ответ 3

:

Введите ответ №3

Ответ 4

:

Введите ответ №4

Правильный №

:

Подтвердить вопрос

Добро пожаловать
в Панель управления экзаменом – Администратор

Список вопросов

#	Имя	Действие
1	Имя переменной в PHP начинается с	Удалить
2	Имя переменной в PHP начинается с	Удалить
3	Имя переменной в PHP начинается с	Удалить
4	Имя переменной в PHP начинается с	Удалить
5	Имя переменной в PHP начинается с	Удалить
6	Имя переменной в PHP начинается с	Удалить
7	Имя переменной в PHP начинается с	Удалить

Элементы управления

Домой

Управление пользователями

Добавить вопрос

Управление вопросами

Выход

Результаты работы

- Разработано приложение полностью соответствующее техническому заданию;
- Локально настроена техническая среда для исполнения приложения;
- Проведено функциональное тестирование (успешно);
- Намечены задачи по развитию функциональных возможностей программы.

Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090501.08.001 ПЗ

Спасибо за внимание!

					<i>ПГУ 1.090501.08.001 ПЗ</i>	Лист
						127
Изм.	Лист	№ докум.	Подпись	Дата		