*Министерство образования и науки Российской Федерации*

*Федеральное государственное бюджетное образовательное учреждение высшего образования*

*«Владимирский государственный университет*

*имени Александра Григорьевича и Николая Григорьевича Столетовых»*

*Кафедра информационных систем и программной инженерии*

***ПОЯСНИТЕЛЬНАЯ ЗАПИСКА***

***к курсовому проекту по дисциплине   
"Технологии разработки мобильных приложений "***

***на тему***

*«Мобильное приложение для оценки знаний техники электробезопасности»*

*Выполнили: студенты гр. ПРИ-115*

*Родионова А. В., Ханова А. А.*

*Принял: доцент Салех Х.М.*

*Владимир, 2018*

**АННОТАЦИЯ**

Тема: Мобильное приложение для оценки знаний техники электробезопасности

В данной работе представлена реализация прототипа мобильного приложения для прохождения опроса с 35 вопросами для проверки уровня знаний электробезопасности.

Курсовой проект представлен на 36 страницах, рисунков – 10, использованных источников – 9, приложений – 3.

ANNOTATION

Topic: Mobile application for assessing the knowledge of electrical safety engineering

To test the level of knowledge about electrical safety.

Course project is presented on the 36 page, pictures - 10, sources used - 9, applications - 3.

СОДЕРЖАНИЕ

[**ВВЕДЕНИЕ** 5](#_Toc533377534)

[**1 ТРЕБОВАНИЯ К РАЗРАБАТЫВАЕМОМУ ПРИЛОЖЕНИЮ** 6](#_Toc533377535)

[**1.1 Наименование программы** 6](#_Toc533377536)

[**1.2 Назначение и цели работы** 6](#_Toc533377537)

[**1.3 Определения, акронимы и сокращения** 6](#_Toc533377538)

[**1.4 Область действия** 7](#_Toc533377539)

[**1.5 Масштаб проекта** 7](#_Toc533377540)

[**1.6 Обзор аналогов** 7](#_Toc533377541)

[**1.7 Постановка задачи** 8](#_Toc533377542)

[**1.8 Сценария работы приложений** 8](#_Toc533377543)

[**1.9 Алгоритм работы приложения** 9](#_Toc533377544)

[**1.10 Функциональность продукта** 10](#_Toc533377545)

[**1.11 Классы и характеристики пользователей** 11](#_Toc533377546)

[**1.12 Среда функционирования продукта** 11](#_Toc533377547)

[**1.13 Ограничения** 11](#_Toc533377548)

[**1.14 Документация для пользователя** 11](#_Toc533377549)

[**1.15 Допущения и зависимости** 11](#_Toc533377550)

[**1.16 Функциональные требования** 11](#_Toc533377551)

[**1.17 Нефункциональные требования** 12](#_Toc533377552)

[**1.18** **Требования к внешним интерфейсам** 13](#_Toc533377553)

[**1.19 Технологии, инструменты и среда разработки** 13](#_Toc533377554)

[**2 ПРОЕКТИРОВАНИЕ СИСТЕМЫ** 14](#_Toc533377555)

[**2.1 Проектирование приложения** 14](#_Toc533377556)

[**2.2 Структура базы данных** 23](#_Toc533377557)

[**3** **РЕАЛИЗАЦИЯ СИСТЕМЫ** 25](#_Toc533377558)

[**3.1 Взаимодействие с локальной базой данных** 25](#_Toc533377559)

[**3.2** **Взаимодействие пользователей** 25](#_Toc533377560)

[**3.3 Реализация интерфейса** 25](#_Toc533377561)

[**3.4 Структура проекта** 26](#_Toc533377562)

[**3.5** **Тестирование** 27](#_Toc533377563)

[**4** **ИНСТРУКЦИЯ ПО УСТАНОВКЕ ПРИЛОЖЕНИЯ** 30](#_Toc533377564)

[4.1 Установка Web – сервиса 30](#_Toc533377565)

[4.2 Установка клиентской части 31](#_Toc533377566)

[**5.** **ОРГАНИЗАЦИЯ УПРАВЛЕНИЯ ПРОЕКТОМ** 32](#_Toc533377567)

[**5.1** **Общие принципы взаимодействия в команде** 32](#_Toc533377568)

[**5.2** **Распределение ролей в команде/зоны ответственности** 33](#_Toc533377569)

[5.3 Календарный план работ 33](#_Toc533377570)

[**ЗАКЛЮЧЕНИЕ** 35](#_Toc533377571)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 36](#_Toc533377572)

[**ПРИЛОЖЕНИЕ А ЛИСТИНГ ПРОГРАММНОЙ СИСТЕМЫ** 37](#_Toc533377573)

**ВВЕДЕНИЕ**

Знание правил техники безопасности очень важно в настоящее время, так как человеку приходится сталкиваться с различными факторами, которые могут оказать существенное влияние на его здоровье.

В настоящее время на производстве строго следят за соблюдениями техники безопасности. Правила электробезопасности не исключение. Каждый человек, работающий с соответствующим оборудованием, обязан знать технику электробезопасности и соблюдать ее. Помогает облегчить процесс запоминания этих правил специальные тесты, которые намного лучше обычной зубрежки.

Целью данного курсового проекта является разработка мобильного приложения, которое представляет из себя тест для оценки знаний по электробезопасности.

Назначение разработки данного мобильного приложения - создать удобное приложение, работающее на ОС Android, позволяющее облегчить и ускорить процесс запоминания правил, оценить свои знания по электробезопасности.

Область применения разрабатываемого мобильного приложения: помощь в прохождении инструктажа на предприятиях, самостоятельная проверка знаний.

Для реализации поставленной задачи требуется изучить предметную область, сделать анализ, спроектировать и реализовать систему.

**1 ТРЕБОВАНИЯ К РАЗРАБАТЫВАЕМОМУ ПРИЛОЖЕНИЮ**

**1.1 Наименование программы**

Наименование программы – «Проверь себя – Техника электробезопасности».

**1.2 Назначение и цели работы**

Целью данной работы является разработка мобильного приложения, помогающего людям определить уровень своих знаний по технике безопасности. Также данное приложение способно строить статистику о ранее пройденных попытках тестирования пользователей.

**1.3 Определения, акронимы и сокращения**

**Android** - портативная операционная система для коммуникаторов, планшетных компьютеров, электронных книжек, цифровых проигрывателей, наручных часов, нетбуков и смартбуков, основанная на ядре Linux.

**Мобильное приложение** – программное обеспечение, предназначенное для работы на смартфонах, планшетах и других мобильных устройствах.

**База данных** – представленная в объективной форме совокупность самостоятельных материалов, систематизированных таким образом, чтобы эти материалы могли быть найдены и обработаны с помощью электронной вычислительной машины.

**Тест** – список вопросов для оценки знаний по технике безопасности.

**Пользователь** – человек, зарегистрированный и авторизованный в системе.

**Статистика** – количественный показатель правильных ответов на вопросы теста за определенные даты.

**Профиль** – страница с данными о пользователе, доступная только ему.

**Вопрос** – форма мысли, подразумевающая получение новой информации в виде ответа.

**Ответ** – сообщение, вызванное вопросом.

**Результат теста** – проанализированная совокупность ответов респондента на вопросы теста, выраженная в форме результата, а именно количестве правильных ответов их все предложенных.

**1.4 Область действия**

Работники производств, на которых требуется знание техники безопасности.

**1.5 Масштаб проекта**

Проект подходит для России, так как тест основан на российских стандартах техники безопасности и поддерживает один язык – русский.

**1.6 Обзор аналогов**

Существует несколько аналогов разрабатываемому приложению. Самые популярные из них:

1. Техника Безопасности

Краткий обзор:

Данное приложение предназначено для персонала, обслуживающего действующие электроустановки, производящий в них оперативные переключения, выполняющий и организующий ремонтные, монтажные, наладочные работы или испытания[1].

1. Правила ТБ тепломеханического оборудования

Краткий обзор:

Приложение о правилах техники безопасности при эксплуатации тепломеханического оборудования, электростанций и тепловых сетей[2].

1. Промышленная безопасность тесты

Краткий обзор:

Назначение приложения - подготовка работников к сдаче экзамена по промышленной безопасности. Все сборники соответствуют сборникам Ростехнадзора. Приложение позволяет пройти тестирование по актуальным на данный момент категориям:

- А. Общие требования промышленной безопасности (A.1)

- Б. Специальные требования промышленной безопасности (Б.1 - Б.12)

- Г. Энергетическая безопасность (Г.1 - Г.3)

- Д. Требования безопасности гидротехнических сооружений (Д.1 - Д.4)[3]

**1.7 Постановка задачи**

Реализовать мобильное приложение на базе Android, содержащее вопросы по электробезопасности. Всего 35 вопросов. Пользователь должен ответить на все из них. Приложение, согласно логике, должно обработать данные и выдать их текущему пользователю. Управление осуществить с помощью пользовательского интерфейс.

**1.8 Сценария работы приложений**

Сценарий работы приложения:

* При входе в приложение пользователь видит главную страницу приложения, на которой находится форма авторизации и кнопка «Вход», а также кнопка «Регистрация».
* Если пользователь зарегистрирован, он может войти в приложение с помощью кнопки «Вход», если нет, пользователь должен нажать кнопку «Регистрация», после чего он попадет на форму регистрации, заполняет необходимые поля и нажимает кнопку «Зарегистрироваться». После этого пользователь может войти в приложение через форму авторизации.
* После входа пользователь попадает на страницу приветствия приложения с кнопкой «Начать тест».
* При нажатии на кнопку «Начать тест» начинается его прохождение: появляется вопрос с четырьмя вариантами ответа, в верхней части экрана высвечивается информация о том, сколько вопросов из их общего количества уже пройдено, а также идет обратный отсчет времени, отведенного на тест.
* После прохождения теста появляется всплывающее окно с информацией, что тест успешно пройден, на котором также находится кнопка, при нажатии на которую пользователь перенаправляется на форму с результатами только что пройденного теста.
* Во время прохождения теста также появляется кнопка меню, в котором есть кнопки «Пройти тест», «Профиль», «Статистика», «Выход». Кнопка «Пройти тест» перенаправляет пользователя на форму прохождения теста, «Профиль» - на профиль пользователя, «Статистика» - на форму статистики, где доступна информация о предыдущих результатах прохождения теста, «Выход» перенаправляет пользователя на форму авторизации.

**1.9 Алгоритм работы приложения**

Алгоритм работы приложения приведен на диаграмме, представленной на рисунке 1.

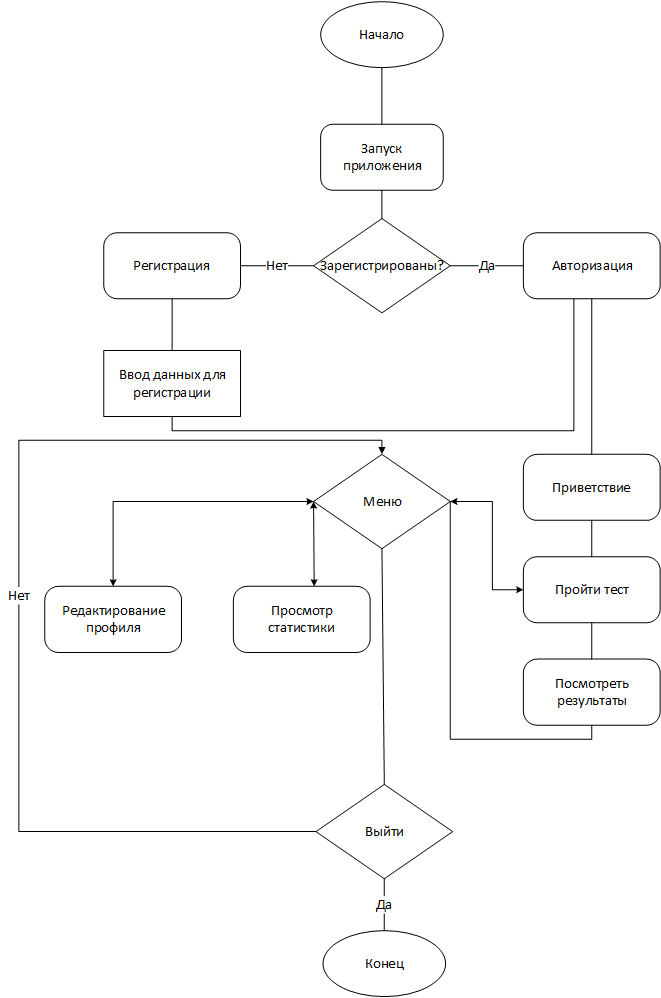


Рисунок 1. Алгоритм работы приложения.

**1.10 Функциональность продукта**

1. регистрация и авторизация в системе;
2. просмотр и редактирование профиля;
3. просмотр результатов ранее пройденных тестов;
4. возможность отвечать на вопросы теста.

**1.11 Классы и характеристики пользователей**

В программной системе не предусмотрено ролей. Есть только один тип пользователя, который проходит тест.

**1.12 Среда функционирования продукта**

Приложение будет функционировать на смартфонах с ОС Android версии 4.1 и выше.

**1.13 Ограничения**

* необходим доступ к сети Интернет;
* смартфон с ОС Android не ниже версии 4.1, API 16.

**1.14 Документация для пользователя**

Информация для пользователя будет представлена на странице приветствия приложения.

**1.15 Допущения и зависимости**

Допущений и зависимостей не имеется.

**1.16 Функциональные требования**

1. Регистрация и авторизация пользователя

* Пользователь должен зарегистрироваться и авторизоваться в системе для того, чтобы пользоваться ею.
* Без авторизации пользователю недоступен функционал системы.
* При регистрации данные нового пользователя заносятся в базу данных и, при последующем входе в систему, берутся из нее.

1. Просмотр результатов прохождения теста

* В профиле пользователь находится кнопка «Статистика», при нажатии на которую появляется страница со статистикой пользователя.
* Для того, чтобы статистика пользователя был не пустой, пользователю необходимо хотя бы один раз пройти тест.
* Статистика пользователей формируется из базы данных.

1. Ответить на вопрос теста

* Главная функция приложение – предоставление пользователю возможности отвечать на вопросы теста. При появлении вопроса, предоставляются четыре варианта ответа, один из которых правильный. Пользователь должен выбрать один из них.
* В статистике учитываются только правильные ответы пользователя.
* Вопросы и соответствующие им ответы хранятся в базе данных. Каждая полная попытка пользователя пройти тест (т. е. тест проходится до конца) заносится в базу данных и становится доступна в статистике.

1. Просмотреть профиль

* В списке меню пользователь может нажать на кнопку «Профиль», чтобы перейти к своему профилю.
* В профиле пользователь может редактировать свою личную информацию.
* Информация о пользователе хранится в базе данных.

**1.17 Нефункциональные требования**

1. Время между предъявлением системе входных данных и появлением соответствующей выходной информации – не более 5 сек.
2. Понятный интерфейс, все кнопки хорошо видны, текст разборчив
3. ОС – Android 4.1 и старше
4. Для сохранности данных на стороне мобильного приложения должно гарантироваться стабильное подключение к сети Интернет.
5. IntelliJ IDEA 2017.3.4 x64 и позднее
6. Unity 2018.2.14f1 64x и позднее
   1. **Требования к внешним интерфейсам**

Был разработан прототип пользовательского интерфейса[4]. Разработанные прототипы представлены в Приложении Б.

**1.19 Технологии, инструменты и среда разработки**

1. Программа реализована на языке C#;
2. среда разработки – Unity3D;
3. среда программирования сервера – IntelliJ IDEA;
4. язык реализации сервера – Java;
5. база данных – PostgreSQL.
6. Язык моделирования – UML
7. Среда моделирования – Enterprise Architect, Visio
8. Инструмент для тестирования API - Postmen

**2 ПРОЕКТИРОВАНИЕ СИСТЕМЫ**

**2.1 Проектирование приложения**

Функции приложения отображены на диаграмме прецедентов, представленной на рисунке 2.

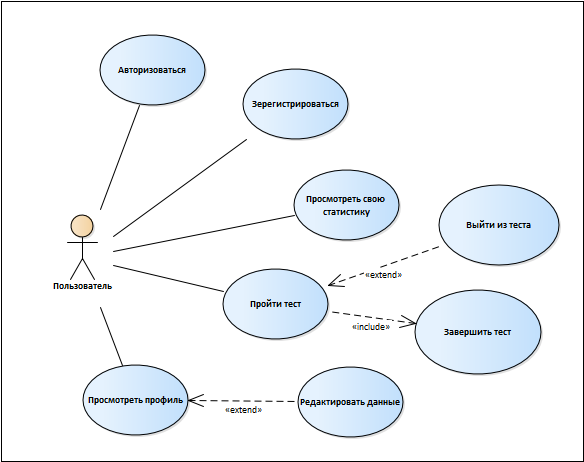


Рисунок 2. Диаграмма прецедентов.

* Спецификация варианта использования «Зарегистрироваться»

**Название:** Зарегистрироваться

**Краткое описание:** Действующее лицо вводит данные для регистрации, и, в случае успеха, создается регистрационная запись, и действующее лицо получает возможность зайти в систему.

**Действующее лицо:** Пользователь

**Предусловие:** Нет

**Основной поток:**

1. Действующее лицо нажимает на кнопку «Регистрация» на главном экране системы;
2. Система перенаправляет действующее лицо на форму ввода данных для регистрации;
3. Действующее лицо вводит свои данные;
4. Действующее лицо нажимает кнопку «Регистрация»;
5. Система проверяет правильность введенных данных;
6. Система отсылает запрос на сервис;
7. Система получает и обрабатывает ответ от сервиса;
8. Система перенаправляет действующее лицо на страницу входа.

**Постусловие:** Создана регистрационная запись о действующем лице

**Альтернативный поток 1:**

1. Альтернативный поток начинается на шаге 5, если введенные данные не правильны;
2. Система сообщает действующему лицу, что данные введены некорректно;
3. Поток возвращается на шаг 3.

**Альтернативный поток 2:**

1. Альтернативный поток начинается на шаге 7, если сервис не отвечает на запрос;
2. Система показывает действующему лицу форму с ошибкой.

* Спецификация варианта использования «Авторизоваться»

**Название:** Авторизоваться

**Краткое описание:** Действующее лицо вводит данные для аутентификации, и, в случае успеха, система предоставляет ему доступ к закрытым функциям.

**Действующее лицо:** Пользователь

**Предусловие:** Действующее лицо зарегистрировано в системе

**Основной поток:**

1. Действующее лицо вводит свои данные для аутентификации;
2. Действующее лицо нажимает кнопку «Вход»;
3. Система проверяет правильность введенных данных;
4. Система отсылает запрос на сервис;
5. Система получает и обрабатывает ответ от сервиса;
6. Система перенаправляет действующее лицо на страницу приветствия.

**Постусловие:** Действующее лицо авторизовано в системе

**Альтернативный поток 1:**

1. Альтернативный поток начинается на шаге 5, если введенные данные не правильны;
2. Система сообщает действующему лицу, в каких полях ошибка;
3. Поток возвращается на шаг 3.

**Альтернативный поток 2:**

1. Альтернативный поток начинается на шаге 7, если сервис не отвечает на запрос;
2. Система показывает действующему лицу форму с ошибкой.

* Спецификация варианта использования «Просмотреть свою статистику»

**Название:** Просмотреть свою статистику

**Краткое описание:** Пользователь переходит на страницу, где содержится информация обо всех попытках прохождения теста.

**Действующее лицо:** Пользователь

**Предусловие:** Действующее лицо авторизовано в системе

**Основной поток:**

1. Действующее лицо переходит в свой профиль;
2. Действующее лицо нажимает кнопку «Статистика»;
3. Система перенаправляет действующее лицо на страницу со статистикой.

**Постусловие:** -

**Альтернативный поток 1:**

1. Альтернативный поток начинается на шаге 1, если сервис не отвечает на запрос;
2. Система показывает пользователю форму с ошибкой.

* Спецификация варианта использования «Пройти тест»

**Название:** Пройти тест

**Краткое описание:** Пользователь отвечает на вопросы теста.

**Действующее лицо:** Пользователь

**Предусловие:** Действующее лицо авторизовано в системе

**Основной поток:**

1. Действующее лицо нажимает на кнопку «Пройти тест»;
2. Система отсылает запрос на сервис о получении данных вопросов;
3. Система получает и обрабатывает ответ от сервиса.

**Постусловие:** Результат прохождения теста сохраняется в БД

**Альтернативный поток 1:**

1. Альтернативный поток начинается на шаге 1, если сервис не отвечает на запрос;
2. Система показывает пользователю форму с ошибкой.

* Спецификация варианта использования «Выйти из теста»

**Название:** Выйти из теста

**Краткое описание:** Пользователь во время прохождения теста принудительно завершает его, не закончив.

**Действующее лицо:** Пользователь

**Предусловие:** Действующее лицо авторизовано в системе

**Основной поток:**

1. Действующее лицо нажимает на кнопку выхода из теста (кнопка стрелка назад в нижнем левом углу);
2. Появляется диалоговое окно подтверждения завершения теста;
3. Система не сохраняет текущий результат пользователя, если пользователь принудительно завершил тест.

**Постусловие:** Результат прохождения теста не сохраняется в БД

**Альтернативный поток 1:**

1. Альтернативный поток начинается на шаге 2, если действующее лицо не соглашается с завершением теста;
2. Пользователь продолжает прохождение теста.

**Альтернативный поток 2:**

1. Альтернативный поток начинается на шаге 3, если сервис не отвечает на запрос;
2. Система показывает пользователю форму с ошибкой.

* Спецификация варианта использования «Завершить тест»

**Название:** Завершить тест

**Краткое описание:** Пользователь завершает тест после того, как полностью его прошел.

**Действующее лицо:** Пользователь

**Предусловие:**

1. Действующее лицо авторизовано в системе
2. Действующее лицо полностью прошло тест

**Основной поток:**

1. Действующее лицо ответило на все вопросы, и система перенаправила его на форму с информацией о том, что тест пройден. Пользователь нажимает на кнопку «Результат»
2. Появляется диалоговое окно с результатом прохождения теста и кнопкой меню.
3. После нажатия кнопки меню диалоговое окно закрывается, пользователь попадают в меню приложения;
4. Система сохраняет текущий результат пользователя.

**Постусловие:** Результат прохождения теста сохраняется в БД

**Альтернативный поток 1:**

1. Альтернативный поток начинается на шаге 2, если сервис не отвечает на запрос;
2. Система показывает пользователю форму с ошибкой.

* Спецификация варианта использования «Просмотреть профиль»

**Название:** Просмотреть профиль

**Краткое описание:** Пользователь переходит на страницу профиля, где содержится информация, доступная только ему.

**Действующее лицо:** Пользователь

**Предусловие:** Действующее лицо авторизовано в системе

**Основной поток:**

1. Действующее лицо нажимает на кнопку меню, которая расположена в верхнем левом углу;
2. В меню Действующее лицо нажимает на кнопку «Профиль»
3. После нажатия кнопки «Профиль» система перенаправляет пользователя на страницу профиля.

**Постусловие:** -

**Альтернативный поток 1:**

1. Альтернативный поток начинается на шаге 2, если сервис не отвечает на запрос;
2. Система показывает пользователю форму с ошибкой.

* Спецификация варианта использования «Редактировать данные»

**Название:** Редактировать данные

**Краткое описание:** Пользователь изменяет свои данные на странице профиля.

**Действующее лицо:** Пользователь

**Предусловие:**

1. Действующее лицо авторизовано в системе
2. Действующее лицо находится на странице профиля

**Основной поток:**

1. Действующее лицо редактирует поля с данными;
2. Действующее лицо нажимает на кнопку «Сохранить»
3. Система сохраняет измененные данные.

**Постусловие:** -

**Альтернативный поток 1:**

1. Альтернативный поток начинается на шаге 2, если сервис не отвечает на запрос, возникает ошибка сохранения данных;
2. Система сообщает пользователю о некорректности введенных данных или показывает пользователю форму с ошибкой.

Диаграмма последовательностей процесса «Пройти тест» представлена на рисунке 3

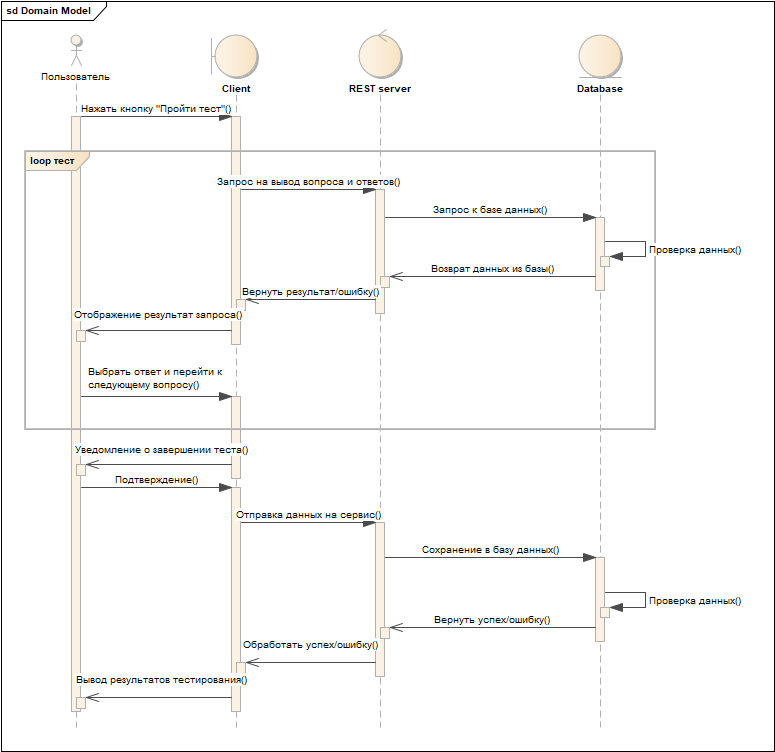


Рисунок 3. Диаграмма последовательности

Здесь представлен процесс прохождения теста пользователем. Пользователь выбирает варианты ответа, клиентская часть приложения запоминает его результат. После подтверждения завершения теста, пользователь видит результат только что пройденного тестирования, а результат сохраняется в базе данных.

Диаграмма состояний объекта «Тест» представлена на рисунке 4.

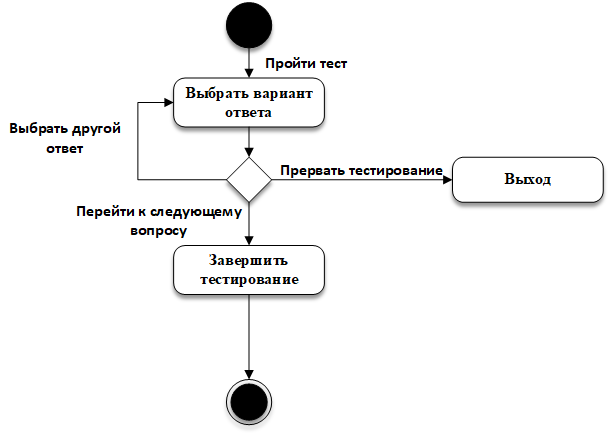


Рисунок 4. Диаграмма состояний объекта «Тест».

На диаграмме состояний показаны состояния объета «Тест» во время прохождения тестирования пользователем.

**2.2 Структура базы данных**

Структура базы данных представлена на диаграмме, которая приведена на рисунке 5.

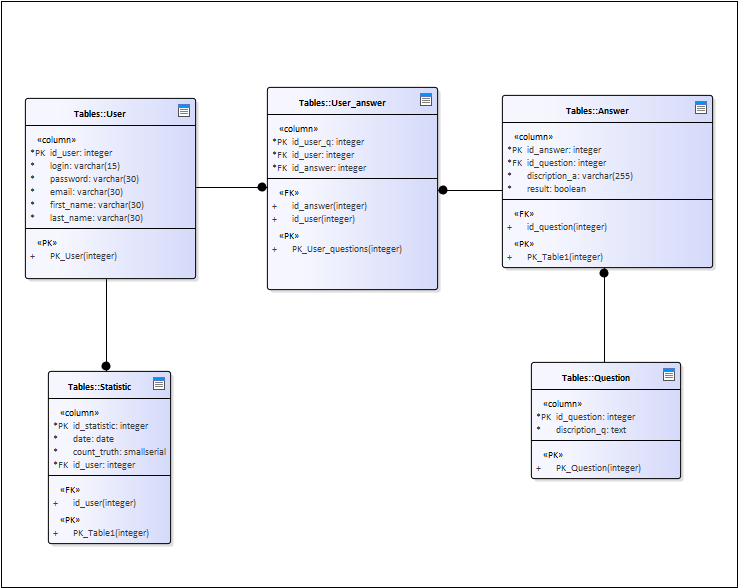


Рисунок 5. Структура базы данных

* Таблица Users содержит информацию о профиле локального пользователя. Данные представляются в виде 6 полей: id пользователя, логин, пароль, электронный адрес, имя и фамилия.
* Таблица Statistic хранит в себе информацию об уже пройденных тестах. Содержит 4 поля: id, дату прохождения, количество правильных ответов, id пользователя
* Таблица Questions хранит в себе вопросы теста, содержит 2 поля: id и сам вопрос.
* Таблица Answer хранит в себе варианты ответов на вопросы, содержит 4 поля: id, id вопроса, сам вопрос и является он верным или нет.
* Таблица UserAnswer содержит информацию об ответах пользователя.

1. **РЕАЛИЗАЦИЯ СИСТЕМЫ**

**3.1 Взаимодействие с локальной базой данных**

Локальная база данных в проекте не предусмотрена.

* 1. **Взаимодействие пользователей**

Взаимодействие устройства пользователя и приложения наглядно видно на диаграмме развертывания (рисунок 6).

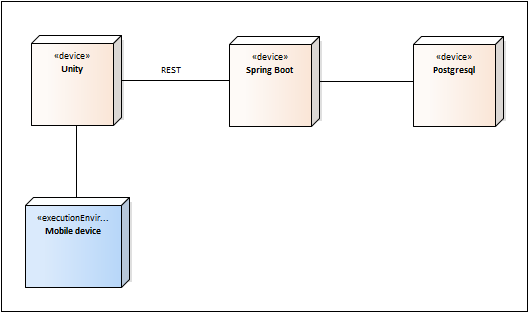


Рисунок 6. Диаграмма развертывания

**3.3 Реализация интерфейса**

В качестве интерфейса используются сцены среды Unity.

Для интерфейса была выбрана цветовая схема из следующих основных цвета: белый и оранжевый. Белый является нейтральным цветом, оранжевый – цвет энергии. Он возбуждает нервную систему, но не так сильно как красный. Такого действия вполне достаточно, чтобы повысить концентрацию внимания, добавить энергии. Это помогает также снять легкое нервное напряжение, ведь многие могут волноваться перед прохождением сложного теста.

Скриншоты с интерфейсом программы представлены в Приложении В.

**3.4 Структура проекта**

Сервис имеет следующую структуру:

* Сущности (каталог domain), которые соответствуют сущностям в базе данных Users, Question, Statistic, Answer, UserAnswer.
* Для каждой сущности был создан интерфейс (каталог repos), содержащий CRUD операции. Они соответствуют каждой сущности исходя из первой части своего названия: UserRepos – Users, QuestionRepos – Question, AnswerRepos – Answer, StatisticRepos – Statistic, UserAnswerRepos – UserAnswer.
* Также для каждой сущности были созданы контроллеры (каталог controller) для реализации операций с базой данных, которые также соответствуют по названиям: UserControlle – Users, QuestionController – Question, AnswerController – Answer, StatisticController – Statistic.
* В каталоге service хранятся классы для обеспечения безопасного использования приложения: классы формирования токена.

На клиентской части была создана следующая структура:

* Папка Assets – главная папка, содержит сцены, анимацию, файлы настройки, остальные папки проекта.
* Папка Prefabs - папка, содержащая префабы.
* Папка Scripts - папка, содержащая скрипты, реализующие связь с сервисом:
* Authorization – скрипт, реализующий авторизацию пользователя.
* Registration – скрипт, реализующий регистрацию нового пользователя.
* ProfileScript – скрипт, реализующий работу с профилем пользователя.
* ScrollViewAdapter – скрипт, реализующий вывод статистики пользователя.
* QuestionScript – скрипт, реализующий логику работы с тестом.
* Transition – скрипт переходов между сценами.
* Timer – скрипт таймера.
* FadeInOut – скрипт для анимации.
* Папка Sprites - папка, содержащая картинки.
  1. **Тестирование**

Для проверки корректности работы мобильного приложения был выбран метод функционального тестирования.

Тестирование проводилось по функциональным требованиям системы. Были сымитированы ситуации реального использования приложений и проверены все требуемые функции разрабатываемой системы. Также производилось тестирование интерфейса на четырех различных устройствах с различной конфигурацией и размером дисплеев:

* Sumsung Galaxy S8, экран 6.2 дюйма
* Meizu Pro 7, экран 5.2 дюйма
* Xiaomi Redmi 4X, экран 5 дюймов
* Samsung Galaxy Grand Prime, экран 5 дюймов
* Sony Xperia S, экран 4.3 дюйма

В ходе тестирования проверялась корректность выполняемых функций, а также правильность и удобство работы пользовательского интерфейса.

Все функции работали корректно, все страницы отображались как следует, а время отклика системы было минимально.

Также для тестирования REST сервиса был использован такой инструмент тестирования API как Postman. Было произведено тестирование всего функционала сервиса. Пример тестирования сущности Users представлено на рисунках 7 – 9.

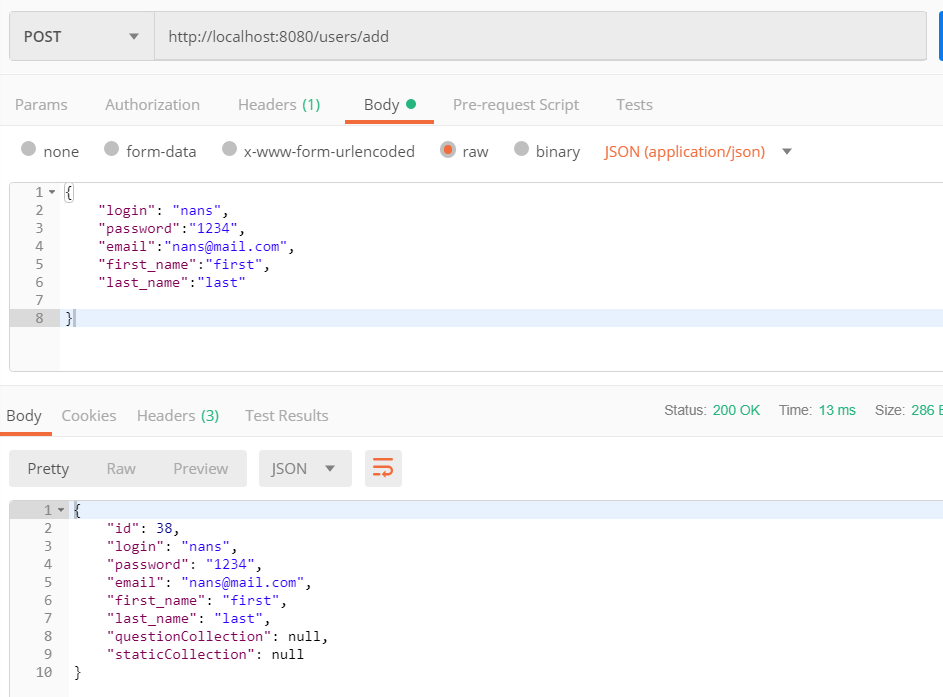
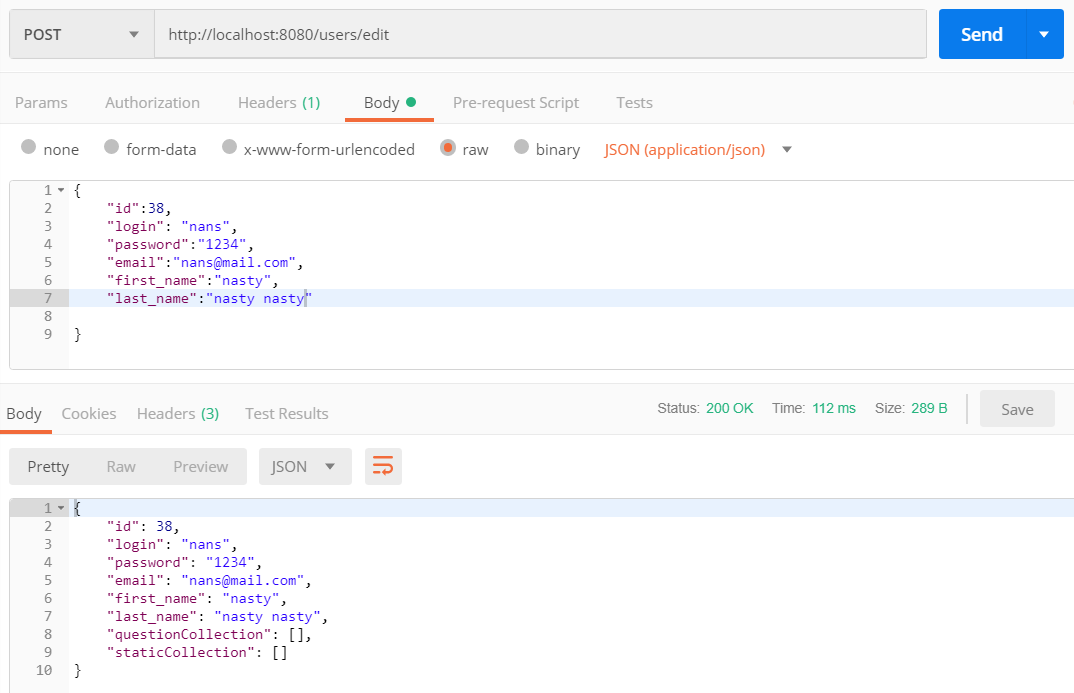


Рисунок 7. Добавление (регистрация) пользователя.

Рисунок 8. Изменение пользователя

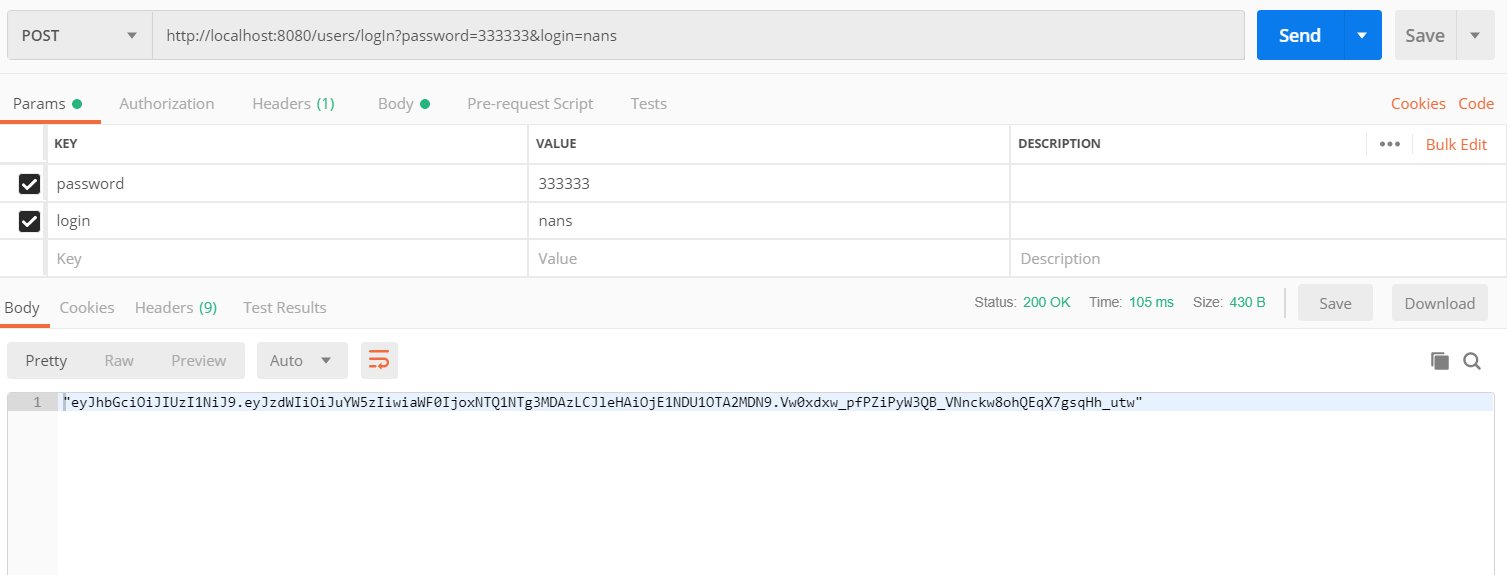


Рисунок 9. Вход пользователя. Метод возвращает токен.

1. **ИНСТРУКЦИЯ ПО УСТАНОВКЕ ПРИЛОЖЕНИЯ**

## 4.1 Установка Web – сервиса

Для использования функционала мобильного приложения необходимо поднять Web-сервис.

Шаг 1: Убедитесь, что у вас установлены обязательные компоненты такие как:

* IntelliJ IDEA 2017.3.4 x64 и позднее (например учебная версия)
* СУБД Posqgresql

Шаг 2: Скачайте web-сервис (доступен по ссылке: <https://github.com/nans1996/Victorina_development/tree/master/mobile_service> ).

Шаг 3: Открытие проекта.

* При запуске IntelliJ IDEA выбираем пункт Import Project
* В проекте сервиса выбираем pom.properties
* Подгруздка зависимостей. Устанавливаем галочку на против пункта: Import Maven project automaticall

Шаг 4: Создание подключения

* Заходим в Posqresql и создаем базу данных. Для этого можно использовать командную строку или средство самой базы данных PgAdmin.
* В проекте web-сервиса выбираем вкладку Database после чего выполняем добавление нового подключения (зеленый плюсик) и выберем пункт создать подключение к PostqreSQL.
* Для успешного подключения необходимо заполнить все поля: host, на котором располагается БД, название базы данных, пользователя и его пароль

Шаг 5: Также необходимо отредактировать файл настройки application.properties в соответствие с этапом, выполненным ранее

Шаг 6: Сборка и разворачивание проекта

* На вкладке Maven Projects выполним команду собрать. После чего запустим наш проект.

Если все пункты выполнены верно, проект соберется без ошибок и наш сервис готов к использованию.

## 4.2 Установка клиентской части

Перейдем к настройке клиентской части приложения.

Шаг 1: Убедитесь, что у вас установлены обязательные компоненты такие как:

• Unity 2018.2.14f1 64x и позднее

Шаг 2: Скачайте и разархивируйте проект приложения (доступен по ссылке: <https://github.com/nans1996/Victorina_development/tree/master/El_Protecting> ).

Шаг 3: Открытие проекта.

• При запуске Unity появляется окно с ранее созданными проектами. Нужно нажать на кнопку Open

• После нажатия кнопки Open нужно выбрать папку с проектом. После этого проект появится в списке ранее созданных.

• Нажмите на название проекта, после этого начнется его открытие

Шаг 4: Запуск проекта

• После запуска среды откроется окно с интерфейсом приложения.

• Чтобы запустить приложение, необходимо нажать на кнопку запуска (кнопка Play в верхней части окна среды).

• Для работы приложения необходимо, чтобы был развернуть сервис.

Если все пункты выполнены верно, приложение запуститься и с ним можно работать.

1. **ОРГАНИЗАЦИЯ УПРАВЛЕНИЯ ПРОЕКТОМ**
   1. **Общие принципы взаимодействия в команде**

Для того чтобы добиться определенной эффективности коллективных взаимодействий, необходимо создать команду.

Для организации эффективной работы в нашей команде мы придерживались некоторых принципов:

• Позитивное отношение, ясность и общие цели для нас куда более сильные мотиваторы, чем дедлайны и бесконечные списки дел. Нет ничего плохого в стрессе как таковом, но без положительных эмоций он приносит лишь психологические страдания и снижает продуктивность.

• Мы ценим удобство правил и предписаний, но воображение и возможность меняться для нас куда важнее. Структурированная командная работа, правила, сам процесс — эффективны только при условии их гибкости. Структура является инструментом и должна меняться по ходу того, как развивается рабочий процесс.

• Мы ценим индивидуальные качества каждого, но способность работать в команде и разделять общие цели для нас куда важнее. Хорошая идея приходит к кому-то одному, но воплощает в жизнь её вся команда. Только находясь вместе, общаясь напрямую и оценивая работу друг друга, мы создали мобильное приложение для проверки знаний техники электробезопасности.

Взаимодействие и общение команды происходит посредством общей конференции в социальной сети Вконтакте.

Благодаря принципам работы в команде, которых мы придерживались, наша работа была более четкой и эффективной.

* 1. **Распределение ролей в команде/зоны ответственности**

Можно говорить о некоторых признаках организации хорошей команды. В ней участники считают себя частью рабочей группы. Соблюдается баланс между эффективной командной работой и сотрудничеством. Члены группы чувствуют себя компетентными, выполняют задания самостоятельно и несут ответственность за их выполнение. Между ними идет непрерывная дискуссия для улучшения сотрудничества и повышения эффективности командной работы. Каждый человек свободно предлагает свои идеи и критикует других.

Члены команды знают о заданиях друг друга и имеют представление о талантах и способностях каждого, что означает интерес и уважение друг к другу. Поэтому роли в нашей команде распределились следующим образом:

*Родионова Анастасия* - лидер команды, организатор разработки проекта, отвечает за разработку Web – сервиса, обеспечение безопасного подключения пользователя приложения (работа пользователя с использованием токена), реализация связи приложения и web-сервиса, тестирование Web – сервиса, тестирование связи приложения и web-сервиса.

*Ханова Анастасия* - Front-end разработчик, отвечает за аналитическую составляющую проекта, построение диаграмм, проектирование UX/UI интерфейса, тестирование UX/UI интерфейса, создание анимации, создание логотипа приложения, ответственная за юзабилити проекта.

## Календарный план работ

Календарный план работ представлен на рисунке 10.

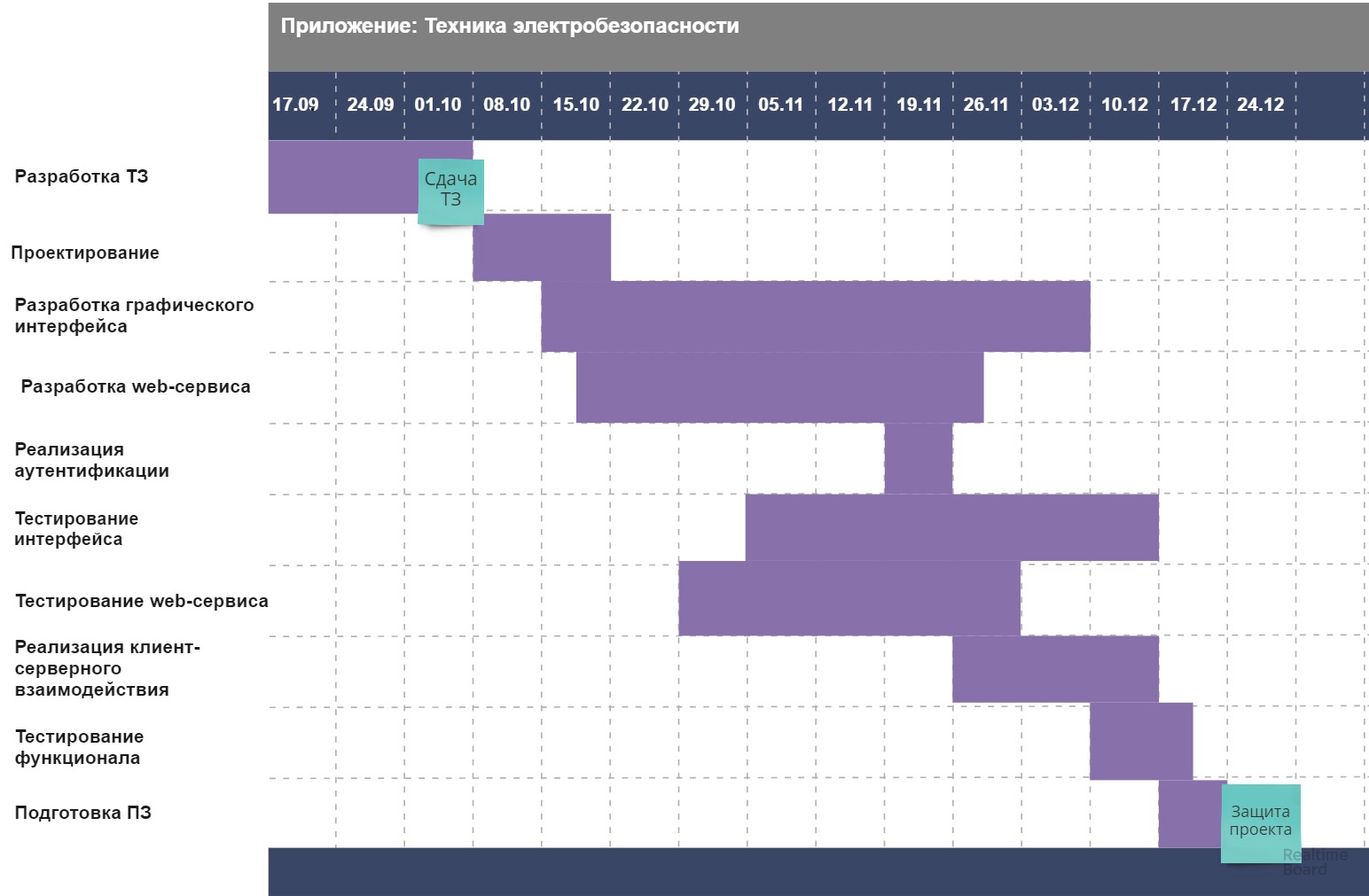


Рисунок 10. Календарный план работ.

Таким образом, были распределены основные роли и зоны ответственности в команде, построен план работы над проектом.

**ЗАКЛЮЧЕНИЕ**

В ходе работы было реализовано мобильное приложение для удобного прохождения тестов по электробезопасности. Была достигнута главная цель – создать приложение, которое будет оценивать уровень знаний у проходящего тест.

В приложении было реализовано:

* Регистрация и авторизация
* Просмотр и редактирование личных данных
* Возможность прохождения теста в ограниченное количество времени (30 минут)
* Просмотр результата завершенного текущего теста
* Просмотр результатов прохождения теста за все время

Разработанное мобильное приложение является актуальным на сегодняшний день и имеет большую значимость, так как техника электробезопасности в современное время является очень важной вещью и знания ее необходимы при работе на многих предприятиях. Область применения достаточно широка: помощь в прохождении и сдаче инструктажа по технике электробезопасности.

В процессе разработки были реализованы все основные функции для успешной работы приложения.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Приложение «Техника безопасности» [Электронный ресурс] – Режим доступа: <https://play.google.com/store/apps/details?id=ru.tbelectric>.
2. Приложение «Правила ТБ тепломеханического оборудования» [Электронный ресурс] – Режим доступа: <https://play.google.com/store/apps/details?id=moskovchenko.ptbetoet>.
3. Приложение «Промышленная безопасность тесты» [Электронный ресурс] – Режим доступа: <https://play.google.com/store/apps/details?id=ponasenkov.vitaly.promtests>
4. Создание веб и мобильных прототипов [Электронный ресурс] – Режим доступа: <https://www.fluidui.com/>
5. Unity. Документация [Электронный ресурс] – Режим доступа: <https://docs.unity3d.com/ru/current/Manual/index.html>
6. Стиллмен Э., Грин Д. – Изучаем C#, 3-е изд. – Спб.: Питер, 2014. – 816 с.
7. Буч Г., Рамбо Д., Якобсон И. – Язык UML. Руководство пользователя. 2-е изд.: Пер. с англ. Мухин Н. – М.: ДМК Пресс, 2006. – 496 с.: ил.
8. Хокинг Д. – Unity в действии. Мультиплатформенная разработка на C#: изд. «Mining», 2015. – 333 с.
9. Калберстон Р., Браун К., Кобб Г. Быстрое тестирование М.: изд. Дом «Вильямс», 2002.

**ПРИЛОЖЕНИЕ А  
ЛИСТИНГ ПРОГРАММНОЙ СИСТЕМЫ**

Полный код приложения доступен по ссылке <https://github.com/nans1996/Victorina_development>

Клиентская часть приложения: <https://github.com/nans1996/Victorina_development/tree/master/El_Protecting>

Web - сервис: <https://github.com/nans1996/Victorina_development/tree/master/mobile_service>

**ПРИЛОЖЕНИЕ Б**

Пользовательский интерфейс

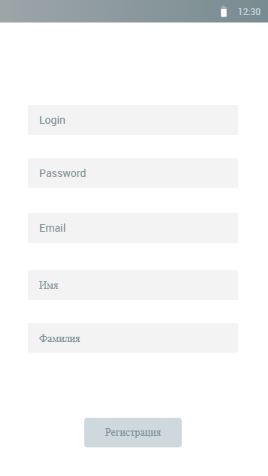
 

Рис. Б1. Форма авторизации Рис. Б2. Форма регистрации

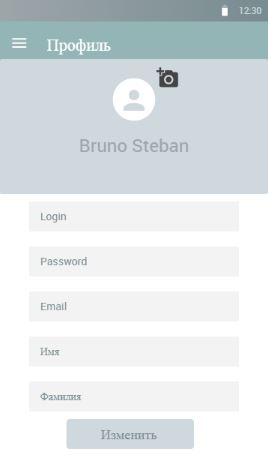
 

Рис. Б3. Профиль пользователя Рис. Б4. Стартовое окно викторины

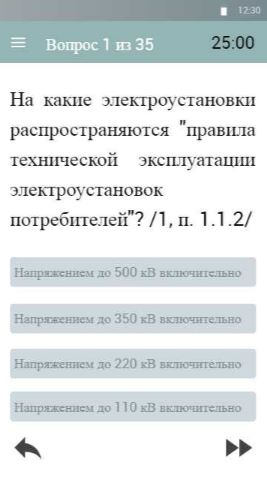
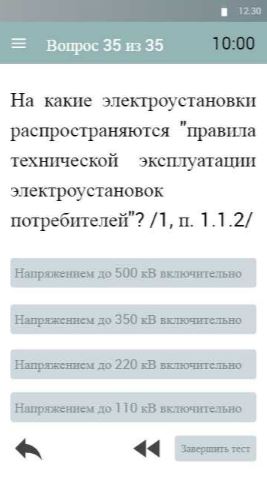
 

Рис. Б5. Форма с 1 вопросом Рис. Б6. Форма с последним вопросом

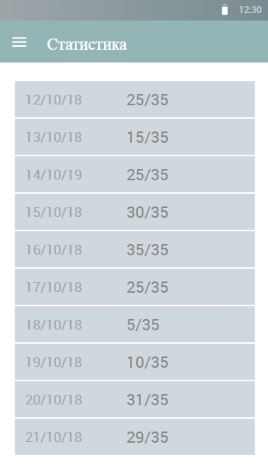
 

Рис. Б7. Статистика пользователей Рис.Б8. Меню приложения

**ПРИЛОЖЕНИЕ В**

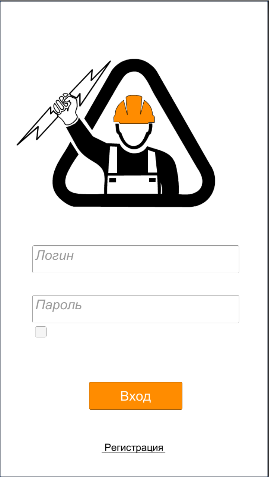
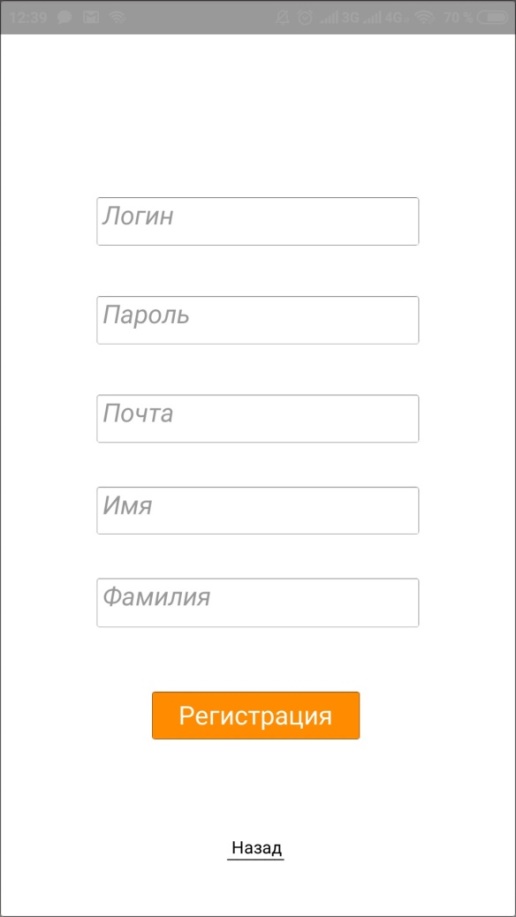
 

Рисунок В1. Окно авторизации Рисунок В2. Окно регистрации.

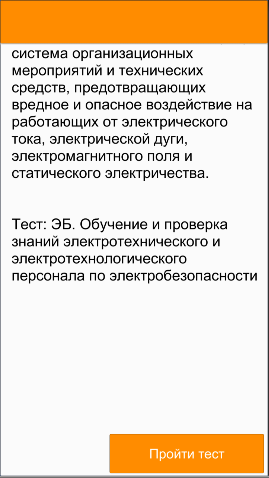
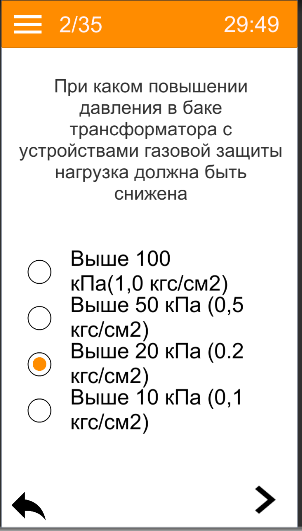
 

Рисунок В3. Приветствие пользователя. Рисунок В4. Форма вопроса.

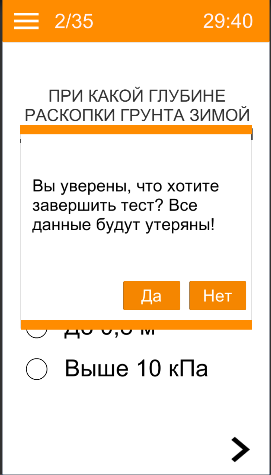
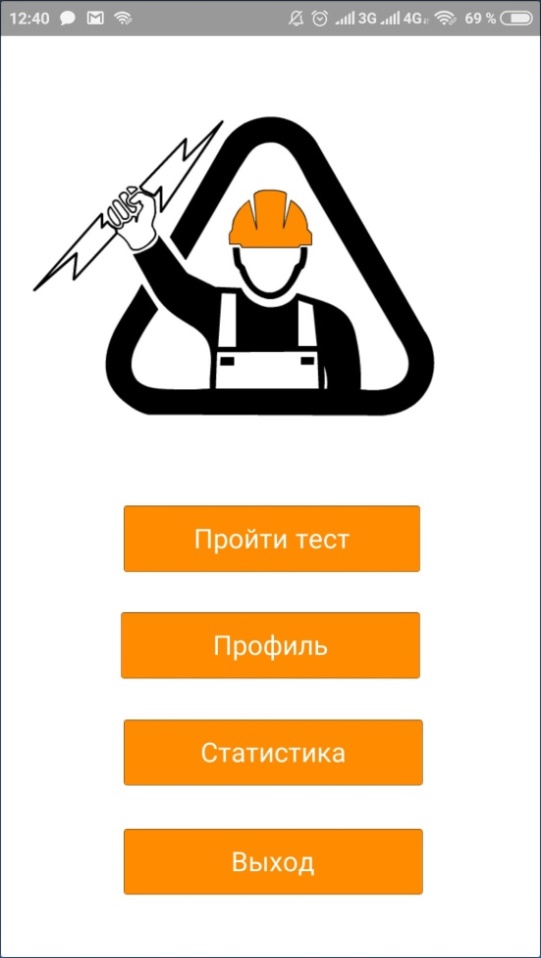
 

Рисунок В5. Всплывающее окно. Рисунок В6. Меню.

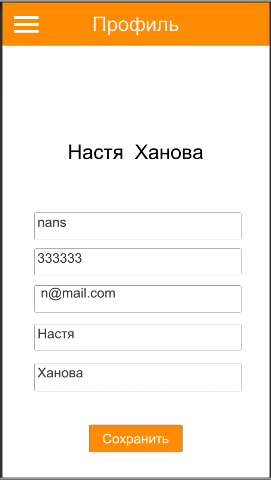
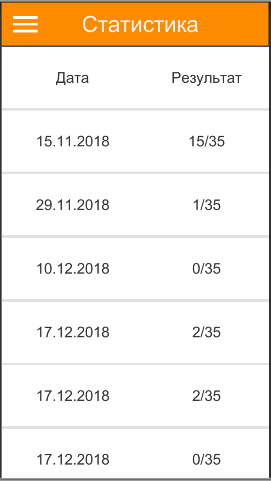
 

Рисунок В7. Профиль. Рисунок В8. Статистика.

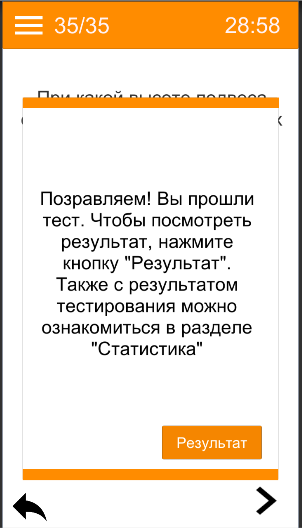
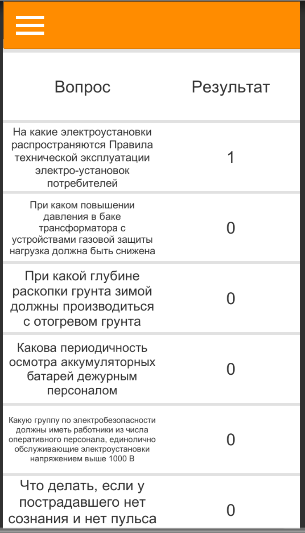
 

Рисунок В9. Завершение теста. Рисунок В10. Результат теста.



Рисунок В11. Форма ошибки.