

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

Факультет Компьютерных наук  
Кафедра программирования и информационных технологий

Техническое задание  
на разработку мобильного приложения  
«Приложение для контроля времени и задач TimeAhead»

Исполнители

\_\_\_\_\_ Г.А.Асатрян  
\_\_\_\_\_ М.В.Королицкий  
\_\_\_\_\_ К.А.Чуркин

Заказчик

\_\_\_\_\_ В.С. Тарасов  
\_\_\_\_\_ Е.Д. Проскуряков

Воронеж 2024

# СОДЕРЖАНИЕ

1	Терминология .....	4
2	Общие сведения .....	5
2.1	Полное наименование системы и название приложения .....	5
2.2	Наименование исполнителя и заказчика приложения .....	5
2.2.1	Наименование заказчика .....	5
2.2.2	Наименование исполнителя .....	5
2.3	Перечень документов, на основании которых создается система .....	5
2.4	Плановые сроки начала и окончания работы по созданию системы .....	6
2.5	Порядок оформления работ и предъявления заказчику результатов работ по созданию системы .....	6
3	Цели и назначение создания приложения .....	7
3.1	Назначение приложения .....	7
3.2	Цели создания приложения .....	7
3.3	Задачи приложения .....	7
4	Требования к автоматизированной системе .....	8
4.1	Требования к структуре приложения в целом .....	8
4.1.1	Требования к реализации .....	8
4.1.2	Перспективы развития, модернизации приложения .....	9
4.2	Функциональные требования к приложению .....	10
4.3	Требования к архитектуре приложения .....	11
4.4	Требования, предъявляемые к персоналу .....	11
4.5	Требование к оформлению и верстке экранов .....	11
4.5.1	Общие элементы верстки .....	15
4.6	Общие технические требования к приложению .....	15
4.6.1	Требования по безопасности .....	15
4.6.2	Дополнительные требования .....	15
5	Описание приложения .....	16
5.1	Язык приложения .....	16
5.2	Группы пользователей приложения .....	16
5.3	Описание экранов приложения .....	16
5.3.1	Экран splash .....	16
5.3.2	Onboarding .....	16

5.3.3 Экран входа .....	16
5.3.4 Активность .....	17
5.3.5 Просмотр активности .....	17
5.3.6 Создание активности .....	17
5.3.7 Привычки .....	17
5.3.8 Просмотр привычки .....	17
5.3.9 Создание привычки .....	17
5.3.10 Задачи .....	18
5.3.11 Просмотр задачи .....	18
5.3.12 Создание задачи .....	18
5.3.13 Статистика .....	18
5.3.14 Профиль .....	18
6 Состав и содержание работ по созданию автоматизированной системы .....	19
7 Порядок контроля и приемки автоматизированной системы .....	20
8 Требования к документированию .....	21
8.1 Перечень подлежащих разработке документов .....	21
8.2 Вид представления и количество документов .....	21
Приложение .....	22

## 1 Терминология

**Front-end** – клиентская часть приложения. Отвечает за получение информации с программно-аппаратной части и отображение ее на устройстве пользователя.

**Back-end** – программно-аппаратная часть приложения. Отвечает за функционирование внутренней части приложения.

**REST API** – это архитектурный стиль для создания веб-сервисов. Он позволит приложениям взаимодействовать друг с другом и обмениваться данными через интернет.

**Серверная часть** – компьютер, обслуживающий другие компьютеры (клиентов) и предоставляющий им свои ресурсы для выполнения определенных задач.

**Клиентская часть** – компьютер, использующий ресурсы сервера и предоставляющий пользователю возможность взаимодействия с системой.

**Верстка экрана** – корректное отображение интерфейса приложения на любых устройствах путем задания размеров и разрешения экрана для каждого макета.

**TabBar** – панель вкладок в нижней части экрана, позволяющая быстро переключаться между разделами приложения.

**Onboarding** – обучающая функциональность в приложении, появляющаяся при первом запуске для ознакомления пользователя с продуктом.

**Splash** – текст или лого приложения.

**Фреймворк** – готовый набор инструментов, который помогает разработчику быстро создать продукт.

## **2 Общие сведения**

### **2.1 Полное наименование системы и название приложения**

Полное название мобильного приложения: «Приложение для контроля времени и задач TimeAhead».

Условное обозначение приложения: «TimeAhead».

### **2.2 Наименование исполнителя и заказчика приложения**

#### **2.2.1 Наименование заказчика**

Заказчик: Старший Преподаватель Тарасов Вячеслав Сергеевич, Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Заказчик: Ассистент Проскуряков Егор Дмитриевич, Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

#### **2.2.2 Наименование исполнителя**

Разработчик: команда №6 группы №9.

Состав команды разработчика:

- Асатрян Гагик Арманович;
- Королицкий Михаил Витальевич;
- Чуркин Кирилл Александрович.

### **2.3 Перечень документов, на основании которых создается система**

Данное мобильное приложение разрабатывается на основании следующих документов:

- Закон РФ от 07.02.1992 N 2300-1 (ред. от 11.06.2021) «О защите прав потребителей»;

— Федеральный закон «О персональных данных» от 27.07.2006 N 152-ФЗ.

#### **2.4 Плановые сроки начала и окончания работы по созданию системы**

Начало работы по созданию приложения: 21.02.2024.

Окончание работы: 10.06.2024.

#### **2.5 Порядок оформления работ и предъявления заказчику результатов работ по созданию системы**

К результатам относятся:

- мобильное приложение, разработанное в соответствии с ТЗ;
- техническое задание;
- курсовая работа по проекту;
- видео с презентацией проекта и демонстрацией работы;
- UML диаграммы.

Результаты передаются заказчику частями по завершении каждой стадии работы по созданию приложения. Документация – в электронном виде, в формате PDF.

Проектная документация должна быть разработана в соответствии с ГОСТ 34.602-20.

### **3 Цели и назначение создания приложения**

#### **3.1 Назначение приложения**

Приложение «TimeAhead» помогает пользователям эффективно распределять свою деятельность, улучшать управление временем и достигать своих целей.

#### **3.2 Цели создания приложения**

К целям создания приложения «TimeAhead» относятся:

- создание удобного и интуитивно понятного приложения для эффективного управления временем и задачами;
- предоставление подробной информации о том, как пользователь использует свое время, чтобы помочь ему в анализе и улучшении своих рабочих процессов и эффективности;
- расширение клиентской базы для привлечения рекламодателей;
- получение прибыли путем интеграции рекламы.

#### **3.3 Задачи приложения**

Разрабатываемый проект должен решать следующие задачи:

- ведение и просмотр записей о использовании времени. Пользователь может создавать записи о деятельности, указывать время начала и конца, категорию и описание;
- планирование задач. Пользователь может создавать задачи, указывая сроки, и отмечать их как выполненные;
- отслеживание привычек. Пользователь может создавать новые привычки, указывая периодичность;
- анализ использования времени. Пользователь может просмотреть статистику использования своего времени.

## **4 Требования к автоматизированной системе**

### **4.1 Требования к структуре приложения в целом**

#### **4.1.1 Требования к реализации**

Система должна иметь архитектуру, соответствующую модели клиент-серверного взаимодействия. Клиентом будет выступать мобильное приложение. Сервер посредством REST API возвращает пользователю необходимые данные.

Для реализации серверной части будут использоваться следующие средства:

- язык программирования Java 17;
- фреймворк SpringBoot 3;
- PostgreSQL;
- Flyway;
- Keycloak.

Язык программирования Java выбран, так как он независим от платформы. То есть, приложение, написанное на Java, можно скомпилировать и запустить на любой системе, поддерживающей виртуальную машину Java (JVM).

SpringBoot был выбран, потому что предоставляет легкую и быструю разработку приложений, а также интеграцию с различными базами данных. Он имеет встроенную интеграцию с экосистемой Spring.

Использование базы данных PostgreSQL обусловлено ее гибкостью и надежностью. Так же она поддерживает пользовательские объекты и их поведение, включая типы данных, функции, операции и другое.

FlyWay позволяет удобно управлять базой данных. Основным преимуществом является поддержка инструментов для миграций баз данных.



Keycloak является бесплатным и надежным сервисом для аутентификации, регистрации и управления доступом. Выбран, потому что он универсален, масштабируем и прост в интеграции с другим сервисами.

Для реализации клиентской части мобильного приложения и сервисного веб-приложения будут использоваться следующие средства:

- язык программирования Dart;
- фреймворк Flutter.

Dart – универсальный язык программирования с простым синтаксисом. Он позволяет создавать приложения для разных платформ, включая веб, мобильные устройства и даже встроенные системы.

Flutter был выбран, так как имеет кроссплатформенность и содержит готовые решения, которые позволяют писать меньше кода, что значительно ускоряет разработку.

#### **4.1.2 Перспективы развития, модернизации приложения**

В долгосрочной перспективе возможны следующие направления модернизации мобильного приложения:

- добавление подписки для разблокировки дополнительных функций: анализ активности за день, анализ выполненных задач;
- добавление ненавязчивой рекламы;
- добавление рекомендаций по перераспределению времени по категориям активностей, созданных на основе статистики потраченного времени.

## 4.2 Функциональные требования к приложению

К разрабатываемому приложению выдвигаются следующие функциональные требования:

Авторизированный пользователь обладает следующими возможностями:

- добавление с указанием времени начала и конца, названия, описания и категории (тип), редактирование и удаление записи об активности;
- просмотр созданных активностей по дням;
- добавление с указанием срока выполнения, то есть напоминания, названия, описания и категории (тип), редактирование и удаление записи о поставленной задаче;
- просмотр созданных задач с возможностью сортировки по времени и по категориям;
- добавление с указанием дней и времени напоминания, названия и описания, редактирование и удаление записи о привычке;
- просмотр всех созданных привычек и просмотр по дням;
- создание своих категорий;
- просмотр статистики показывает уровень сформированности привычки.

Неавторизированный пользователь обладает следующими возможностями:

- добавление с указанием временем начала и конца, названия, описания и категории активности (тип), редактирование и удаление записи об активности;

— регистрация или авторизация в аккаунт.

Администратор обладает следующими возможностями:

- просмотр активности пользователей, то есть количество запросов на сервер;
- блокировка подозрительных пользователей.

### **4.3 Требования к архитектуре приложения**

Разрабатываемый проект должен удовлетворять следующим основным требованиям:

- корректная работа на устройствах с операционной системой Android 9.0 и новее;
- корректная работа сервиса для администратора на браузерах Google Chrome версии 122.0.6261.112 и выше, а также в Yandex Browser версии 24.1.3.809 и выше;
- соответствие модели клиент-серверного взаимодействия. Клиентом будет выступать мобильное приложение или браузер.

### **4.4 Требования, предъявляемые к персоналу**

Для корректной работы приложения на стороне заказчика требуется персонал, а именно администратор, который будет обладать компетенцией по использованию функционала админ-сервиса.

### **4.5 Требование к оформлению и верстке экранов**

Экраны мобильного приложения должны быть оформлены в едином стиле с использованием шрифта Roboto.

Необходимо корректное и одинаковое отображение экранов мобильного приложения на устройствах с операционной системой Android 9.0 и выше.

На рисунках 1-5 показаны макеты основных экранов мобильного приложения:

The image displays two side-by-side wireframe screens for a mobile application, separated by a vertical gray line. The left screen is titled 'frame - Авторизация' (Authorization) and features a dark gray header with the word 'Авторизация'. Below the header are two light gray input fields labeled 'почта' (email) and 'пароль' (password). At the bottom of the screen is a dark gray button labeled 'Войдите' (Log in), followed by the text 'или зарегистрируйтесь здесь' (or register here) and 'Войти как гость' (Log in as guest). The right screen is titled 'frame - Регистрация' (Registration) and features a dark gray header with the word 'Регистрация'. Below the header are three light gray input fields labeled 'имя' (name), 'почта' (email), and 'пароль' (password). At the bottom of the screen is a dark gray button labeled 'Зарегистрироваться' (Register), followed by the text 'или войдите здесь' (or log in here).

Рисунок 1 – Экран авторизации

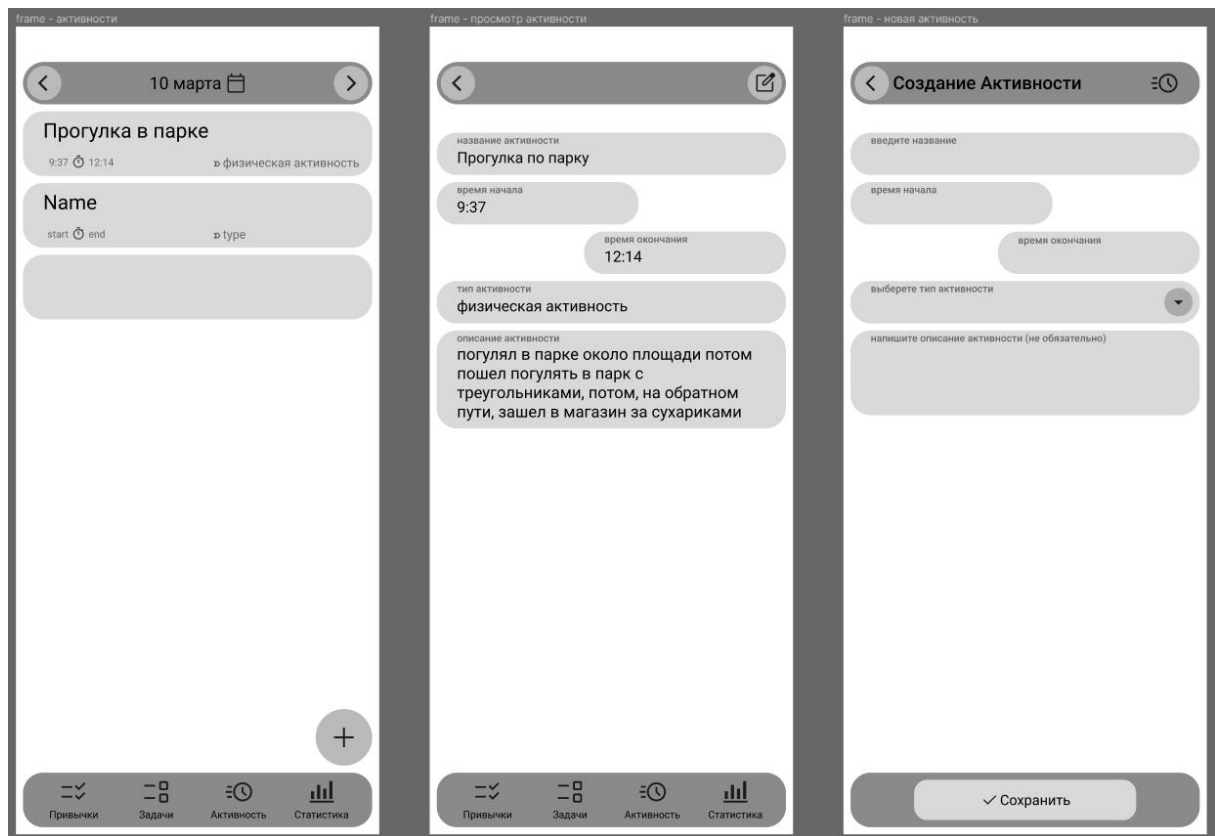


Рисунок 2 – Экран «Активность»



Рисунок 3 – Экран «Задачи»

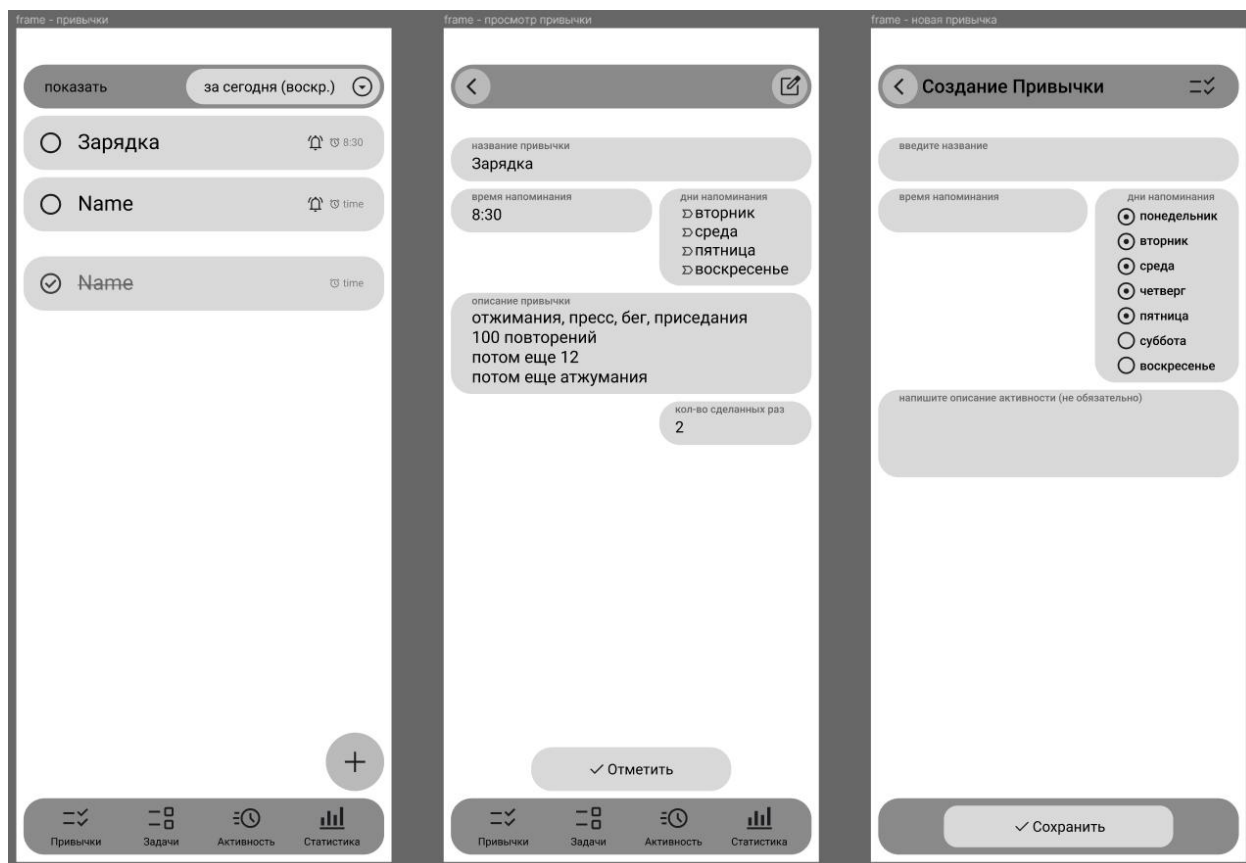


Рисунок 4 – Экран «Привычки»

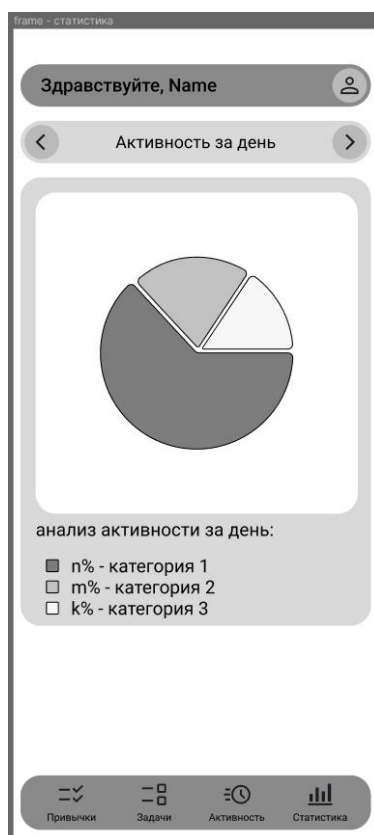


Рисунок 5 – Экран «Статистика»

Сервисная страница системы, то есть сервис администрирования, должен корректно отображаться в следующих браузерах:

— Google Chrome 122.0.6261.112 и выше;

— Yandex Browser 24.1.3.809 и выше.

#### **4.5.1 Общие элементы верстки**

TabBar содержит кнопки перехода на экраны привычек, задач, активности и статистики.

### **4.6 Общие технические требования к приложению**

#### **4.6.1 Требования по безопасности**

Связь между клиентом и сервером должна осуществляться посредством использования https протокола.

Данные пользователей, такие как пароли, должны храниться в базе данных в хешированном виде. Хеширование будет производиться с помощью хеш-функции bcrypt.

#### **4.6.2 Дополнительные требования**

При первом запуске мобильного приложения пользователь должен иметь возможность ознакомиться с основными функциями приложения.

## **5 Описание приложения**

### **5.1 Язык приложения**

Данное приложение поддерживает только русский язык.

### **5.2 Группы пользователей приложения**

Пользователи приложения делятся на следующие группы:

- неавторизованный пользователь;
- авторизованный пользователь;
- администратор.

Функционал для данных ролей описан в пункте 4.2.

### **5.3 Описание экранов приложения**

#### **5.3.1 Экран splash**

Входная точка в приложение для любых пользователей. На данном экране запрашиваются системные разрешения при первом входе в приложение. Например, разрешение уведомлений.

#### **5.3.2 Onboarding**

Если пользователь первый раз запустил приложение, то показывается данная группа экранов. Она знакомит пользователя с ключевыми функциями приложения. Переключение экранов осуществляется с помощью свайпа влево или с помощью нажатия на кнопку «Далее». На последнем экране кнопка «Далее» заменяется на кнопку «Начать». Onboarding можно пропустить, нажав на иконку крестика, находящийся в правом верхнем углу экрана.

#### **5.3.3 Экран входа**

На этом экране пользователю будет предложено войти в аккаунт или зарегистрироваться в приложении. Также имеется возможность войти в качестве гостя.



При входе в аккаунт необходимо будет ввести почту и пароль.

При регистрации нового аккаунта следует ввести имя, почту и пароль.

#### **5.3.4 Активность**

На данном экране пользователь может просматривать список активностей по выбранным дням, перейти к созданию новой активности и перейти к просмотру отдельной активности.

#### **5.3.5 Просмотр активности**

Здесь отображается информация о выбранной активности. Пользователь также имеет возможность отредактировать информацию об активности, кроме дня.

#### **5.3.6 Создание активности**

Здесь пользователю предоставляется возможность создать новую активность и заполнить информационные поля.

#### **5.3.7 Привычки**

На данном экране пользователь может просматривать список своих привычек в определенный день недели, напоминания о которых срабатывают в указанное время. Также пользователь имеет возможность создать новую привычку. Путем нажатия на одну из привычек пользователь может перейти на экран просмотра конкретной привычки.

#### **5.3.8 Просмотр привычки**

Здесь отображается информация о выбранной привычке. Пользователь может отметить ее как выполненную, а также имеет возможность отредактировать информацию о привычке.

#### **5.3.9 Создание привычки**

Здесь пользователю предоставляется возможность создать новую привычку, выбрать дни и заполнить информационные поля.

### **5.3.10 Задачи**

На данном экране пользователь может просматривать список невыполненных и выполненных задач с возможностью фильтрации, отметить задачу как выполненную, перейти к просмотру выбранной задачи или созданию новой задачи.

### **5.3.11 Просмотр задачи**

Здесь отображается информация о выбранной задаче. Пользователь также имеет возможность отредактировать информацию о задаче.

### **5.3.12 Создание задачи**

Здесь пользователю предоставляется возможность создать новую задачу и заполнить информационные поля.

### **5.3.13 Статистика**

Здесь отображается статистика пользователя.

Представлен прогресс формирования привычки.

Также пользователь может перейти в свой профиль путем нажатия на соответствующую иконку.

### **5.3.14 Профиль**

Здесь отображается информация о пользователе, имеется возможность отредактировать данные и выйти из аккаунта.

## **6 Состав и содержание работ по созданию автоматизированной системы**

Состав и содержание работ по созданию мобильного приложения включают в себя следующие этапы:

- сбор необходимой информации, постановка целей, которые в будущем должны быть реализованы;
- анализ предметной области и конкурентов. Выявление требований, предъявляемых к мобильному приложению, которые необходимо реализовать, для решения поставленных задач. Подготовка и проработка ТЗ;
- построение модели программы, описание спецификаций данных, определение связей между сущностями, разработка модели БД;
- разработка рабочего проекта, состоящая из написания кода, отладки и корректировки кода программы;
- проведение тестирования программного обеспечения.

## **7 Порядок контроля и приемки автоматизированной системы**

Предварительные отчеты по работе будут проводиться во время рубежных аттестаций:

- 1 аттестация (конец марта 2024) – создан репозиторий проекта на GitHub, распределены задачи проекта в таск-менеджере YouTrack, создан проект Miro с общей логикой системы, написано техническое задание;
- 2 аттестация (конец апреля 2024) – написана основополагающая часть кода приложения, реализована БД и ее взаимодействие с сервером, проведена отладка и доработка кода, проведено тестирование по работе системы;
- 3 аттестация (конец мая 2024) – разработан курсовой проект, выполнены завершающие работы по доработке приложения, предоставлена готовая система.

## **8 Требования к документированию**

### **8.1 Перечень подлежащих разработке документов**

- Техническое задание;
- Курсовой проект;
- Сопроводительное письмо.

### **8.2 Вид представления и количество документов**

Документы должны быть представлены в электронном виде и опубликованы на сайте [github.com](https://github.com) в репозитории команды разработчика, а также в печатном виде.

Проектная документация должна быть разработана и оформлена в соответствии с ГОСТ 7.32-2001.

## ПРИЛОЖЕНИЕ

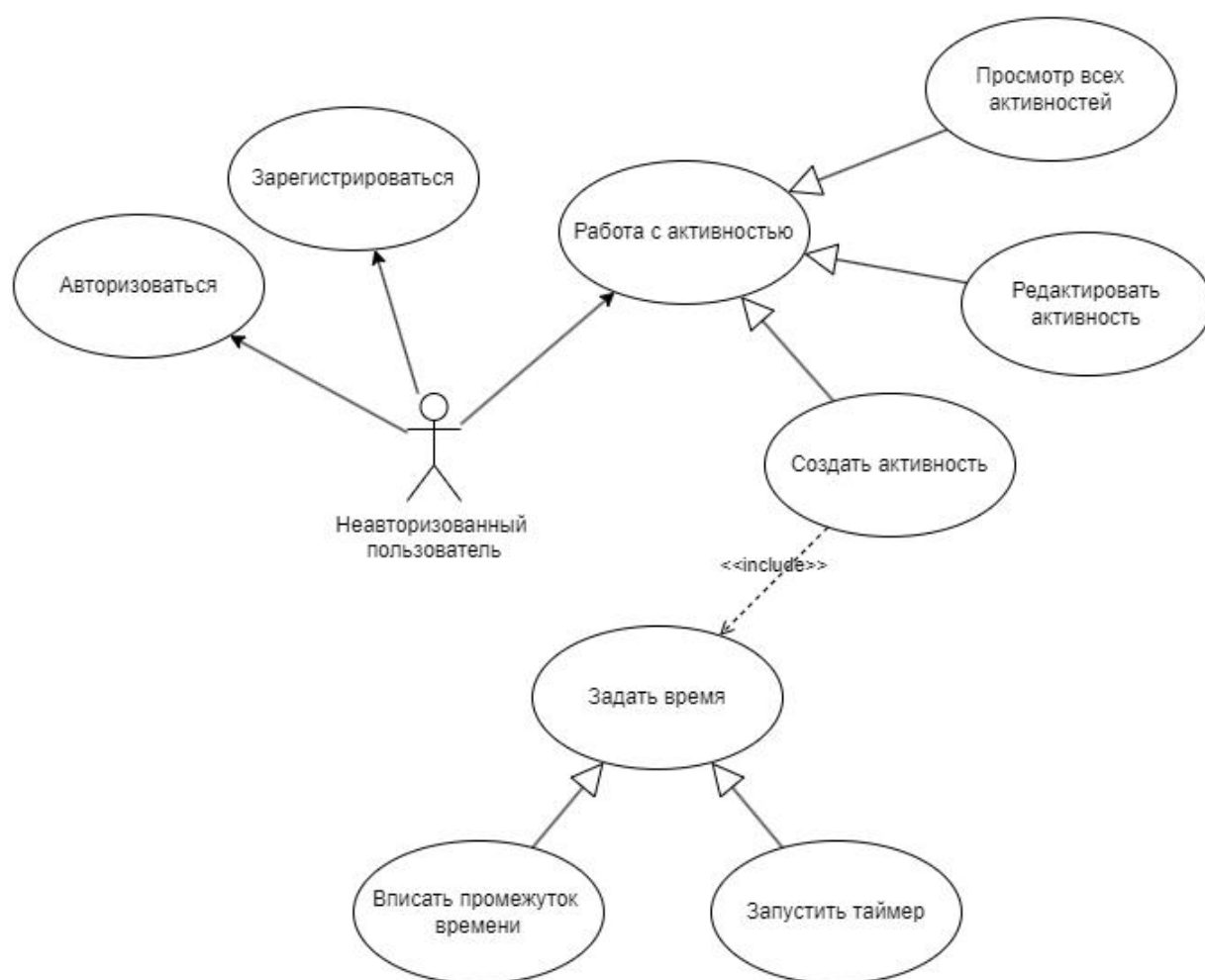


Рисунок 6 – Диаграмма прецедентов Неавторизованного пользователя

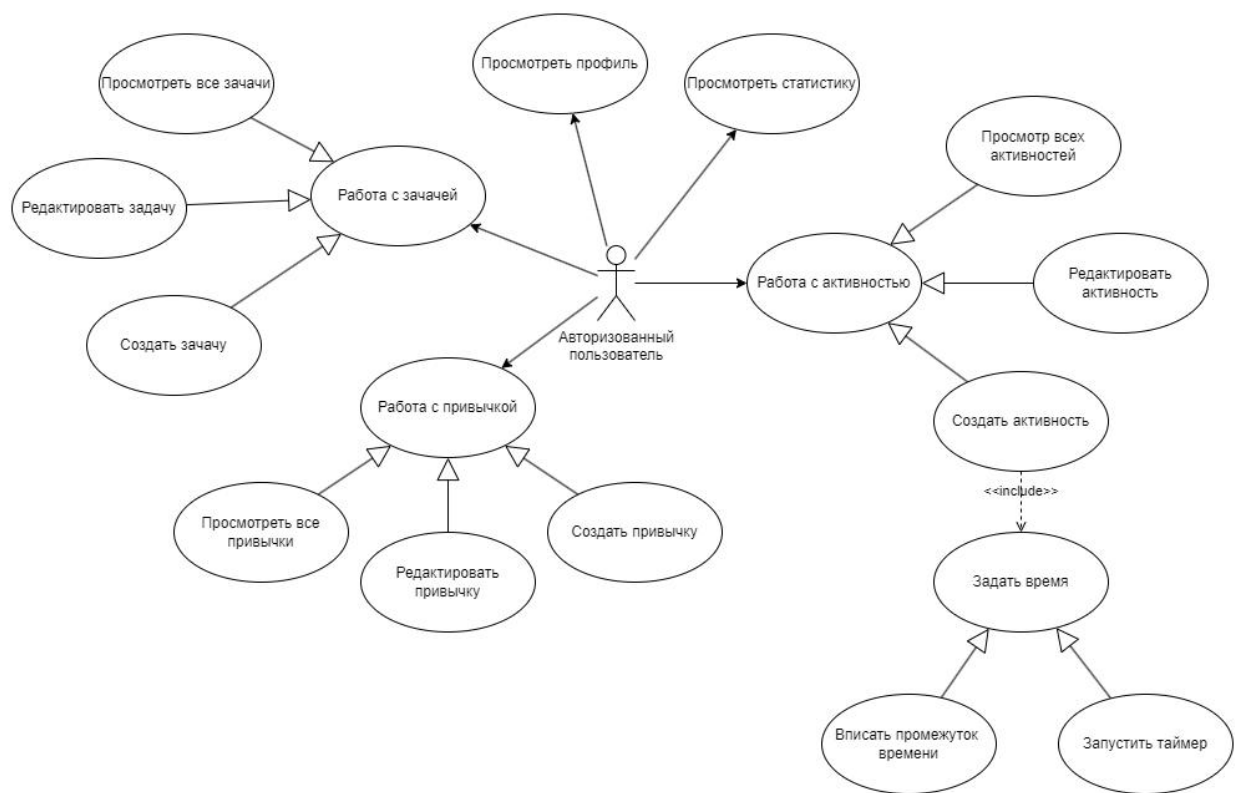


Рисунок 7 – Диаграмма прецедентов Авторизованного пользователя

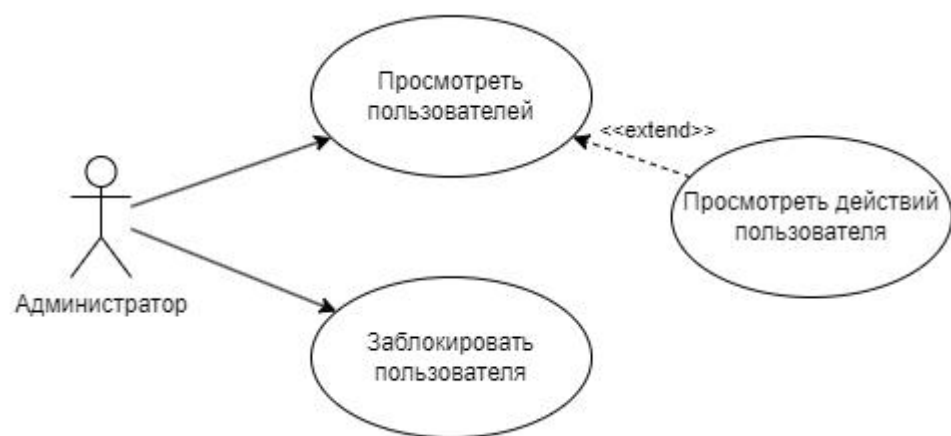


Рисунок 8 – Диаграмма прецедентов Администратора