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

«Платформа для онлайн-обучения ремеслам и хобби KnitWit»

Исполнители

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Павлов А.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Акимушкин Е.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Перепечко К.К.

Заказчик

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Тарасов В.С.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Проскуряков Е.Д.

Воронеж 2024

Содержание

[1 Терминология 4](#_Toc161240731)

[2 Общие сведения 5](#_Toc161240732)

[2.1 Полное наименование системы и название приложения 5](#_Toc161240733)

[2.2 Заказчик 5](#_Toc161240734)

[2.3 Исполнитель 5](#_Toc161240735)

[2.4 Перечень документов, на основании которых создается приложение 5](#_Toc161240736)

[2.5 Плановые сроки начала и окончания работы по созданию приложения 6](#_Toc161240737)

[2.6 Порядок оформления и предъявления заказчику результатов работ по созданию приложения 6](#_Toc161240738)

[3 Назначение и цели создания приложения 7](#_Toc161240739)

[3.1 Назначение приложения 7](#_Toc161240740)

[3.2 Цели создания приложения 7](#_Toc161240741)

[3.3 Задачи приложения 7](#_Toc161240742)

[4 Требования к приложению 8](#_Toc161240743)

[4.1 Требования к структуре приложения в целом 8](#_Toc161240744)

[4.1.1 Требования к реализации 8](#_Toc161240745)

[4.1.2 Перспективы развития, модернизации приложения 9](#_Toc161240746)

[4.2 Функциональные требования к приложению 9](#_Toc161240747)

[4.3 Требования к архитектуре приложения 10](#_Toc161240748)

[4.4 Требования к обслуживающему персоналу 11](#_Toc161240749)

[4.5 Требования к оформлению и верстке экранов 11](#_Toc161240750)

[4.5.1 Общие элементы верстки 13](#_Toc161240751)

[4.6 Общие технические требования к приложению 13](#_Toc161240752)

[4.6.1 Требования по безопасности 13](#_Toc161240753)

[4.6.2 Дополнительные требования 14](#_Toc161240754)

[5 Описание приложения 15](#_Toc161240755)

[5.1 Язык приложения 15](#_Toc161240756)

[5.2 Группы пользователей приложения 15](#_Toc161240757)

[5.2.1 Экран splash 15](#_Toc161240758)

[5.2.2 Onboarding 15](#_Toc161240759)

[5.2.3 Обучение 15](#_Toc161240760)

[5.2.4 Авторизация и регистрация 16](#_Toc161240761)

[5.2.5 Каталог 16](#_Toc161240762)

[5.2.6 Главный экран курса 16](#_Toc161240763)

[5.2.7 Экран раздела курса 16](#_Toc161240764)

[5.2.8 Профиль 17](#_Toc161240765)

[5.2.9 Экран пройденных курсов 17](#_Toc161240766)

[5.2.10 Редактор курса 17](#_Toc161240767)

[5.2.11 Настройки 17](#_Toc161240768)

[6 Состав и содержание работ по созданию приложения 18](#_Toc161240769)

[7 Порядок контроля и приемки приложения 19](#_Toc161240770)

[8 Требования к документированию 20](#_Toc161240771)

[8.1 Перечень подлежащих разработке документов 20](#_Toc161240772)

[8.2 Вид представления и количество документов 20](#_Toc161240773)

[Приложение 21](#_Toc161240774)

1. Терминология

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

**Клиентская сторона** – устройство, использующее ресурсы сервера и предоставляющий пользователю возможность взаимодействия с системой.

**Модерация** – рассмотрение заявок по размещению и редактированию курсов на платформе приложения.

**Администратор** – пользователь, имеющий доступ к расширенному функционалу по модерации курсов пользователей через панель администратора.

**REST API (REST)** – стиль архитектуры программного обеспечения для построения масштабируемых веб-приложений.

**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.

**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.

1. Общие сведения
   1. Полное наименование системы и название приложения

Полное название мобильного приложения: «Платформа для онлайн-обучения ремеслам и хобби KnitWit».

Краткое название мобильного приложения: «KnitWit».

* 1. Заказчик

Заказчик: Старший преподаватель Тарасов Вячеслав Сергеевич, Воронежский государственный университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

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

* 1. Исполнитель

Исполнитель: студент Павлов Александр Викторович, Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Исполнитель: студент Акимушкин Евгений Владимирович, Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Исполнитель: студент Перепечко Константин Кириллович, Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

* 1. Перечень документов, на основании которых создается приложение

Данное мобильное приложение будет разрабатываться на основе данного технического задания и должно удовлетворять всем требованиям, указанным в нем, и на основании следующих документов:

* Федеральный закон «О персональных данных» от 27.07.2006 N 152-ФЗ;
* Закон РФ от 07.02.1992 N 2300-1 (ред. от 14.07.2022) «О защите прав потребителей».
  1. Плановые сроки начала и окончания работы по созданию приложения

Начало работы по созданию приложения: 16.02.2024.

Окончание работы: 10.06.2024.

* 1. Порядок оформления и предъявления заказчику результатов работ по созданию приложения

К результатам будут относиться:

* мобильное приложение, разработанное в соответствии с ТЗ;
* панель администратора в web-браузере, разработанная в соответствии с ТЗ;
* техническое задание;
* курсовая работа по проекту;
* видео с презентацией проекта и демонстрацией его исправной работы;
* UML диаграммы.

Результаты будут передаваться заказчику частями по завершении каждой стадии работы по созданию приложения. Документация – в электронном виде в формате MS PDF.

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

1. Назначение и цели создания приложения
   1. Назначение приложения

Мобильное приложение «KnitWit» будет онлайн платформой для обучения ремеслу и хобби. Оно будет предоставлять доступ к видео-урокам и инструкциям для развития навыков в различных областях, обеспечивая удобство обучения и вдохновляя пользователей на творчество.

* 1. Цели создания приложения

К целям создания приложения относятся:

* создание удобной площадки для онлайн-обучения пользователей;
* предоставление доступа к обучающим материалам для развития навыков;
* стимулирование интереса пользователей к творчеству и саморазвитию.
  1. Задачи приложения

Разрабатываемый проект будет решать следующие задачи:

* приобретение знаний без траты собственного времени на поиск информации;
* отслеживание прогресса при прохождении курса;
* предоставление обширной базы курсов для разных классов пользователей.

1. Требования к приложению
   1. Требования к структуре приложения в целом
      1. Требования к реализации

Система должна будет состоять из сервера приложения, реляционной базы данных, клиентской части.

Основной стек технологий:

Back-end (серверная часть):

* Java 17;
* Spring Framework;
* PostgreSQL;
* система сборки Maven.

Язык Java был выбран, так как он не зависит от платформы. Можно создать Java-приложение на Windows, скомпилировать его в байт-код и запустить на любой другой платформе, поддерживающей JVM – виртуальную машину Java.

Основным преимуществом Spring Framework является большое количество реализованных внутренних библиотек, позволяющих быстро и качественно писать код.

В качестве базы данных была выбрана Postgres, т.к. поддерживает пользовательские объекты и их поведение, включая типы данных, функции, операции и другое. Это делает Postgres невероятно гибким и надежным. Среди прочего, он умеет создавать, хранить и извлекать сложные структуры данных.

Front-end (клиентская часть):

* Dart;
* Flutter.

Dart – универсальный язык программирования с простым синтаксисом.

Главное преимущество Flutter – скорость разработки. Готовые решения данного фреймворка позволяют писать меньше кода, что значительно упрощает процесс создания приложений и существенно экономит время разработчиков.

* + 1. Перспективы развития, модернизации приложения

В долгосрочной перспективе возможны следующие направления модернизации мобильного приложения:

* добавления подписки на курсы на платной основе;
* интеграция ненавязчивой рекламы;
* добавление системы отзывов с комментариями от пользователей.
  1. Функциональные требования к приложению

К разрабатываемому приложению выдвигаются следующие функциональные требования:

Авторизованный пользователь будет обладать следующими возможностями:

* получение доступа к выбранному курсу по подписке;
* фильтрация и сортировка каталога курсов;
* выход из аккаунта;
* создание собственного курса и его редактирование;
* оставлять и просматривать общую пользовательскую оценку курса;
* просмотр списков курсов;
* возможность включения и отключения уведомлений;
* редактирование профиля пользователя.

Неавторизованный пользователь будет обладать следующими возможностями:

* просмотр списка курсов;
* просмотреть информацию о курсе;
* авторизация;
* регистрация.

Администратор будет обладать следующими возможностями:

* редактирование каталога курсов;
* создание курса и его редактирование;
* выход из аккаунта;
* возможность включения и отключения уведомлений;
* модерация курсов.
  1. Требования к архитектуре приложения

Разрабатываемый проект должен будет удовлетворять следующим основным требованиям:

* корректная работа на устройствах с операционной системой Android 9.0 и новее;
* реализация основных задач, стоящих перед данным проектом, функционала, описанного в данном техническом задании;
* соответствие шаблону Клиент-Серверного приложения с применением «Clean architecture» и разделением на два слоя: слой представления (presentation layer) – front-end, слой доступа к данным (data layer) – back-end и связью между ними по средству Rest API.
  1. Требования к обслуживающему персоналу

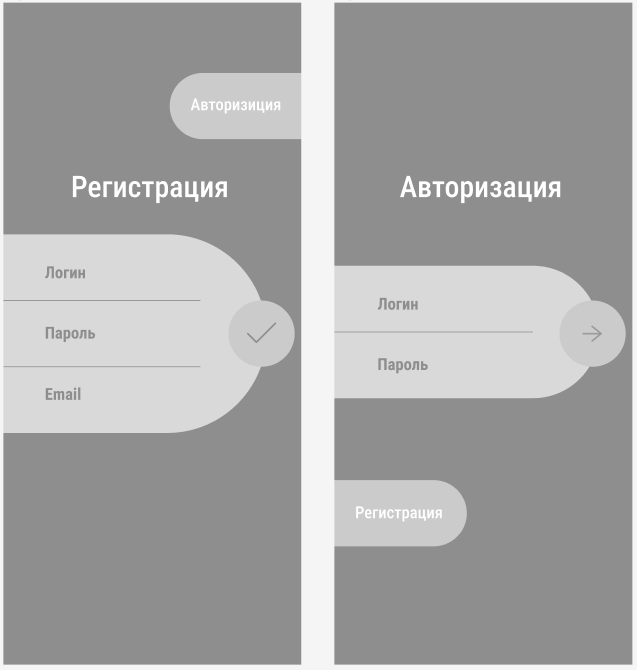
Обслуживанием приложения будет заниматься команда поддержки, которая обеспечит актуальность и стабильную работоспособность приложения. Администрированием баз данных курсов и модерацией будет заниматься команда администрации.

* 1. Требования к оформлению и верстке экранов

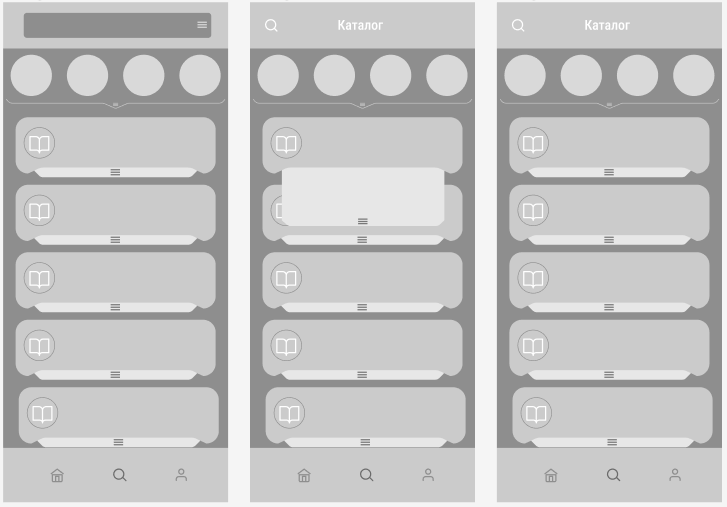
Экраны мобильного приложения будут оформлены в едином стиле с использованием шрифта Roboto.

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

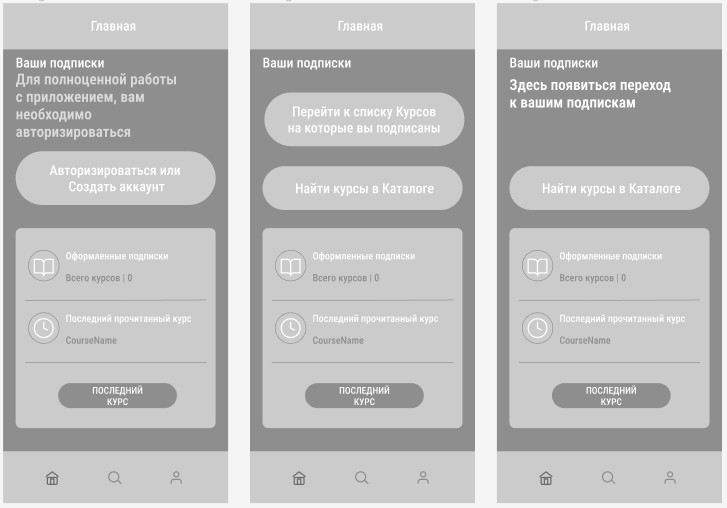
На рисунках 1-4 показаны прототипы основных типов экранов мобильного приложения:



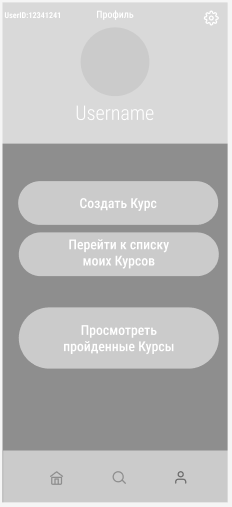
1. – Экран авторизации



1. – Экран вкладки каталога со списком курсов



1. – Экран вкладки обучения



1. – Экран вкладки профиля

Сервисная страница системы должна будет корректно работать в следующих браузерах:

* Google Chrome 122.0.6261.112 и выше;
* Yandex Browser 24.1.3.809 и выше.
  + 1. Общие элементы верстки

TabBar будет содержать кнопки перехода на экраны обучения, каталога, профиля.

* 1. Общие технические требования к приложению
     1. Требования по безопасности
* Обмен данных между клиентом и сервером должен осуществлять по протоколу https;
* Пароли пользователей должны хранится в базе данных в хешированном виде; для хеширования должен использоваться алгоритм bcrypt.
  + 1. Дополнительные требования

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

1. Описание приложения
   1. Язык приложения

Данное приложение будет поддерживать только русский язык.

* 1. Группы пользователей приложения

Пользователи приложения делятся на следующие группы:

* неавторизованный пользователь;
* авторизованный пользователь;
* администратор.

Функционал для данных ролей описан в пункте 4.2.

* + 1. Экран splash

Входная точка в приложение для любых пользователей. На данном экране будут запрашиваться системные разрешения при первом входе в приложение. А именно разрешение уведомлений и доступа к файловой системе устройства.

* + 1. Onboarding

При первом запуске приложения пользователем будет показываться данная группа экранов. Данная группа экранов ознакомит пользователя с ключевыми функциями приложения. Переключение экранов будет осуществляться с помощью свайпа влево или с помощью нажатия на кнопку «Далее». На последнем экране кнопка «Далее» заменится на кнопку «Начать». Onboarding можно будет пропустить, нажав на иконку крестика в правом верхнем углу экрана.

* + 1. Обучение

При запуске приложения пользователь окажется на экране обучения.

Здесь авторизованный пользователь сможет перейти к списку курсов, на которые он подписан, и перейти к последнему прочитанному им курсу. Также перейти к экрану каталога.

Для неавторизованного пользователя будет предложено войти в аккаунт или зарегистрироваться.

* + 1. Авторизация и регистрация

На экране регистрации пользователю будет предложено ввести свои логин, пароль и почту для создания аккаунта.

На экране авторизации пользователю будет необходимо ввести свои логин и пароль для входа в аккаунт.

* + 1. Каталог

На экране каталога пользователь будет иметь возможность находить курсы по фильтру через заданные теги или в строке поиска. После этого выведется список отфильтрованных курсов для просмотра и дальнейшего взаимодействия. Пользователь сможет перейти на главную страницу курса или же открыть всплывающую «шторку» с кратким описанием курса.

* + 1. Главный экран курса

На главном экране курса будет отображаться его название, описание, автор, общая оценка, список разделов и кнопка оформления подписки. Авторизованный пользователь, который не подписан на данный курс, сможет просмотреть демо-материалы и подписаться на курс. Авторизованный пользователь с подпиской на этот курс будет иметь возможность изучать материал разделов и поставить оценку курсу.

* + 1. Экран раздела курса

Здесь пользователь сможет увидеть название раздела и его содержимое. На этом же экране будут находиться переходы к списку разделов, следующему и предыдущему разделам, а также кнопка отметки прогресса.

* + 1. Профиль

На данном экране неавторизованному пользователю будет предложено войти в аккаунт или зарегистрироваться.

Для авторизованного пользователя на экране будут отображаться его аватар и логин.

Здесь пользователь сможет перейти в настройки, к списку всех пройденных курсов и к списку своих собственных курсов, а также в редактор и создать свой собственный курс.

* + 1. Экран пройденных курсов

Здесь пользователь сможет увидеть список всех пройденных им курсов и перейти на главный экран конкретного курса.

* + 1. Редактор курса

Здесь пользователю представится возможность создать новый или отредактировать уже существующий курс, если он является автором или администратором.

* + 1. Настройки

Здесь авторизованный пользователь сможет отредактировать личную информацию, включить или выключить уведомления и выйти из аккаунта.

1. Состав и содержание работ по созданию приложения

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

* сбор необходимой информации, постановка целей, задач приложения, которые в будущем должны быть реализованы;
* анализ предметной области, анализ конкурентов и построение структуры требований, ведущих к решению поставленных задач и целей;
* построение модели программы, описание спецификаций данных, определение связей между сущностями, разработка модели БД;
* разработка рабочего проекта, состоящего из написания кода, отладки и корректировки кода программы;
* проведение тестирования программного обеспечения.

1. Порядок контроля и приемки приложения

Предварительные отчеты по работе будут проводиться во время рубежных аттестаций:

* 1 аттестация (середина марта 2024) – создан репозиторий проекта на GitHub, распределены задачи проекта в таск-менеджере YouTrack, создан проект Miro с общей логикой системы, предоставлены промежуточные результаты по курсовому проекту, готовое техническое задание, презентация проекта, видео-презентация проекта и сопроводительное письмо преподавателю практики;
* 2 аттестация (середина апреля 2024) – написана основополагающая часть кода приложения, реализована БД и ее взаимодействие с сервером, проведена отладка и доработка кода, проведено тестирование по работе приложения;
* 3 аттестация (конец мая 2024) – разработан курсовой проект, выполнены завершающие работы по доработке приложения, предоставлено готовое приложение.

1. Требования к документированию
   1. Перечень подлежащих разработке документов

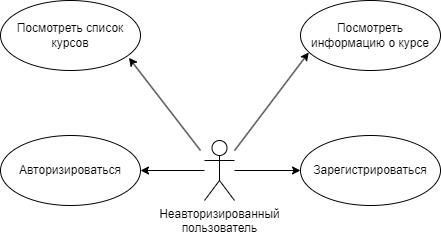
* Техническое задание;
* Курсовой проект;
* Сопроводительное письмо.
  1. Вид представления и количество документов

Документы должны быть представлены в электронном виде и опубликованы на сайте github.com в репозитории команды разработчика, а также в печатном виде. Проектная документация должна быть разработана в соответствии с ГОСТ 7.32-2001.

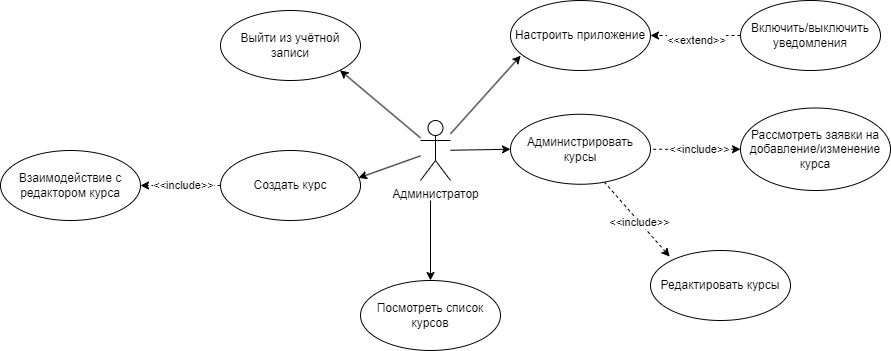
Приложение



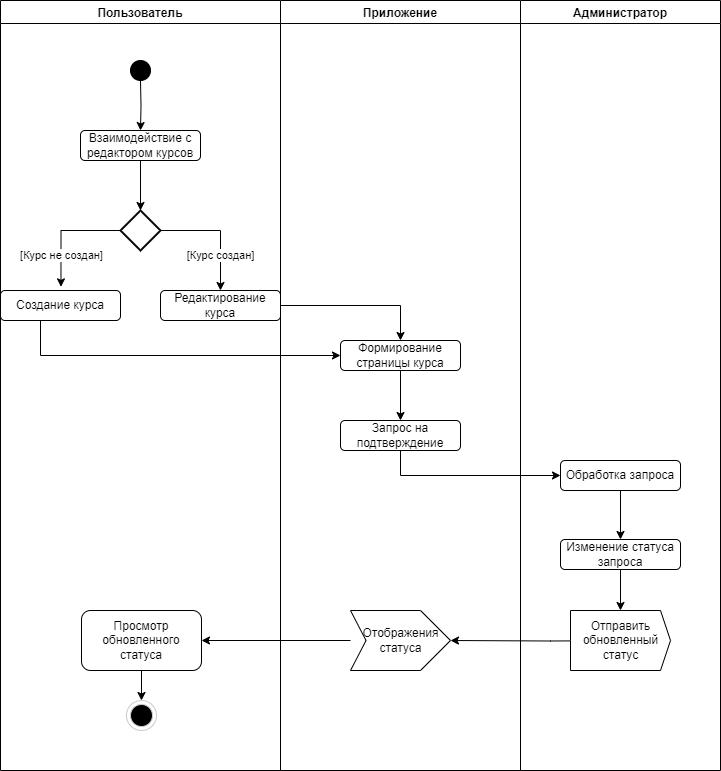
1. – Диаграмма прецедентов



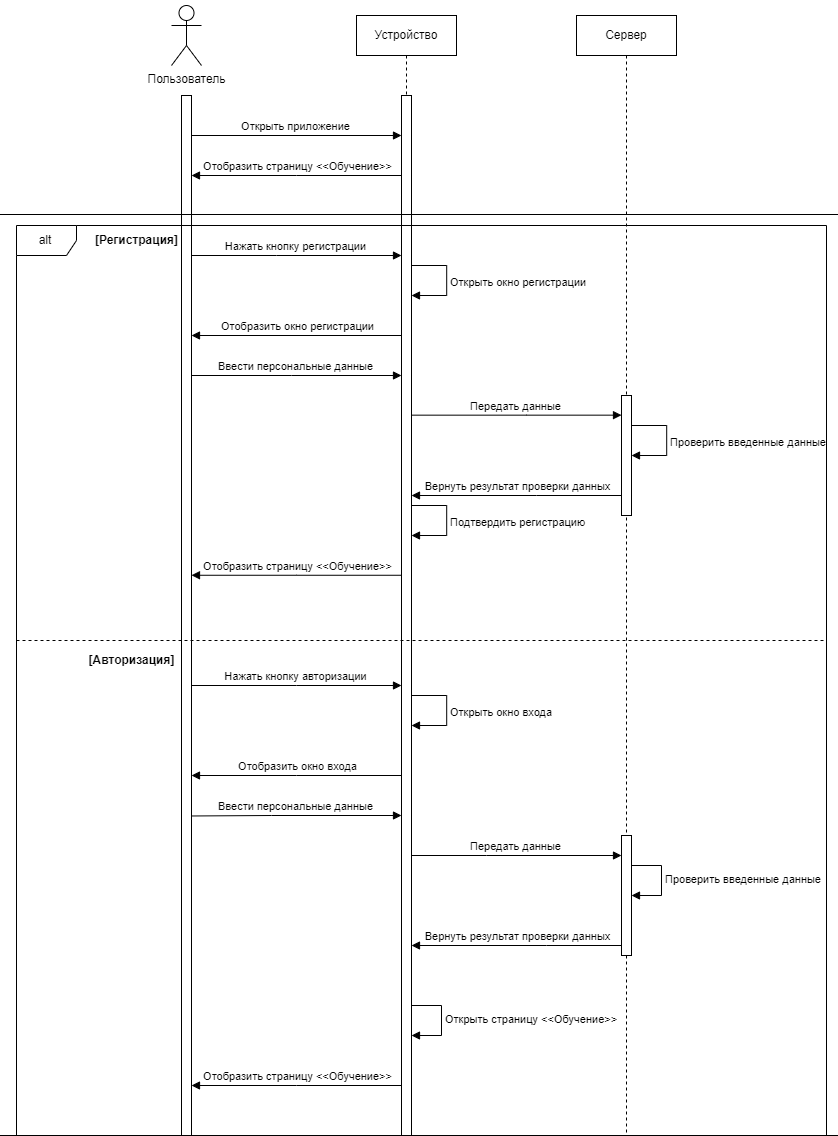
1. – Диаграмма прецедентов



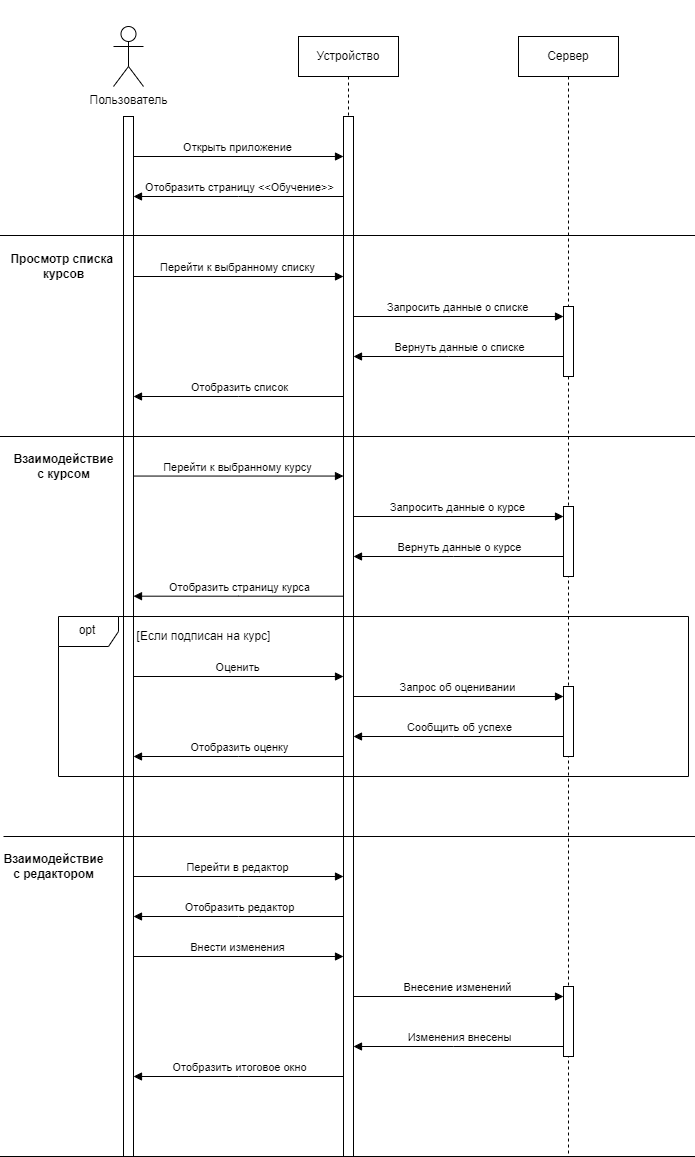
1. – Диаграмма прецедентов



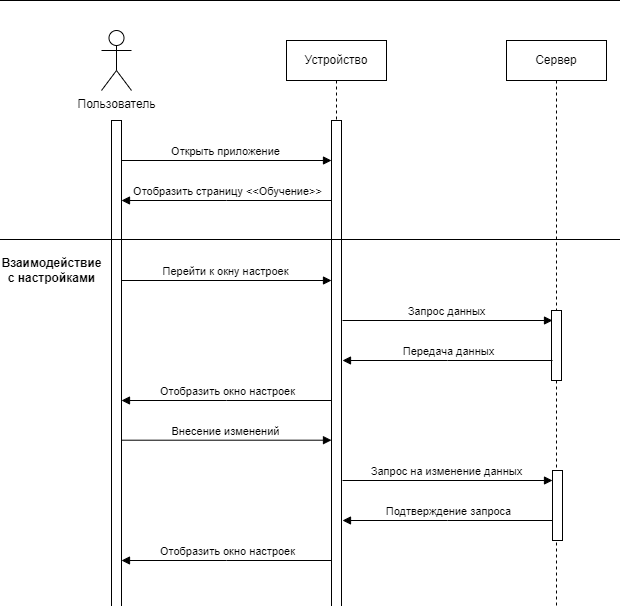
1. – Диаграмма активности (взаимодействие с редактором курса)



1. – Диаграмма последовательности



1. – Диаграмма последовательности (продолжение)



1. – Диаграмма последовательности (продолжение)