

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
«Сервис для поиска и аренды оборудования и инструментов для домашнего
ремонта и строительства»

Исполнители

_____ Д.В. Бучнев

_____ В.Н. Ремезов

_____ Е.А. Клоков

Заказчик

_____ В.С. Тарасов

Воронеж 2024

СОДЕРЖАНИЕ

1	Термины, используемые в техническом задании	4
2	Общие сведения.....	8
2.1	Полное наименование системы и ее условное обозначение	8
2.2	Наименование исполнителя и заказчика приложения	8
2.3	Перечень документов, на основании которых создается приложение.....	8
2.4	Плановый срок начала и окончания работ.....	8
2.5	Состав и содержание работ по созданию системы.....	9
2.6	Порядок контроля и приемки разрабатываемой системы.....	9
3	Цели и назначение создания приложения.....	11
3.1	Цели создания приложения	11
3.2	Назначение приложения	11
4	Требования к структуре приложения в целом	12
4.1	Перечень подсистем, их назначение и основные характеристики	12
4.1.1	Пользовательский сервис.....	12
4.1.2	Сервис аренды	13
4.1.3	Сервис инструментов	13
4.1.4	Сервис авторизации и аутентификации	13
4.1.5	Требования к способам и средствам обеспечения информационного взаимодействия компонентов системы.....	14
4.2	Требования к функциям (задачам), выполняемыми приложением	14
4.2.1	Требования к сервису аренды.....	14
4.2.2	Требования к пользовательскому сервису	14
4.2.3	Требования к сервису инструментов.....	15
4.2.4	Требования к API – шлюзу	15

4.2.5 Требования к сервису авторизации и аутентификации.....	15
4.3 Требования к видам обеспечения системы	16
4.3.1 Требования к лингвистическому обеспечению системы	16
4.3.2 Требования к программному обеспечению системы	16
4.4 Общие технические требования к системе	18
4.4.1 Требования по безопасности	18
4.5 Требования к оформлению мобильного приложения	19
4.5.1 Общие требования к оформлению мобильного приложения	19
4.5.2 Требования к сплэш фрейму.....	21
4.5.3 Требования к фрейму ленты объявлений.....	22
4.5.4 Требования к фрейму со списком инструментов по заданному фильтру	23
4.5.5 Требования к фрейму карточки инструмента.....	24
4.5.6 Требования к фрейму корзины	25
4.5.7 Требования к фрейму оформление заказа.....	27
4.5.8 Требования к фрейму каталог категорий	28
4.5.9 Требования к фрейму профиля авторизованного пользователя	30
4.5.10 Требования к фрейму активных заказов	31
5 Источники разработки	33

1 Термины, используемые в техническом задании

Android – операционная система с открытым исходным кодом, созданная для мобильных устройств на основе модифицированного ядра Linux.

API – набор способов и правил, по которым различные программы общаются между собой и обмениваются данными.

API Gateway – шаблон проектирования, реализующий единую точку входа в микросервисную архитектуру.

API-шлюз – интерфейс взаимодействия, который находится перед API группы микросервисов. Облегчает отправку запросов и доставку данных сервисов.

Back-end – часть программного обеспечения, отвечающая за обработку данных и взаимодействие с сервером.

Flutter – бесплатный и открытый набор средств разработки мобильного пользовательского интерфейса, созданный компанией Google и выпущенный в мае 2017 года.

Git – распределенная система управления версиями, которая обеспечивает контроль изменений в коде, возможность ветвления и слияния кода.

GitHub – платформа для хостинга проектов на базе Git, которая обеспечивает возможность хранения кода, управления задачами, рецензирования кода и совместной работы над проектами.

HTTP — протокол передачи данных в сети Интернет, который используется для передачи информации между клиентом и сервером.

HTTPS — защищенная версия протокола HTTP, использующая шифрование для безопасной передачи данных.

Java – строго типизированный объектно-ориентированный язык программирования общего назначения, разработанный компанией Sun Microsystems.

JVM – виртуальная машина Java — основная часть исполняющей системы Java, так называемой Java Runtime Environment.

JWT – открытый стандарт для создания токенов доступа, основанный на формате JSON.

Keycloak – современное решение для управления доступом и аутентификации, предоставляющее возможности односторонней и двусторонней аутентификации, авторизации, а также управления пользователями и ролями в приложениях и сервисах.

Kong – API-шлюз и управляемая платформа API, предназначенная для управления, масштабирования и защиты микросервисов и приложений. Он обеспечивает централизованный контроль доступа к API, мониторинг трафика, управление версиями API и другие функции для облегчения разработки и управления API.

Linux – семейство операционных систем, построенных на ядре Linux. Ядро Linux является основной частью операционной системы, управляющей ресурсами компьютера и обеспечивающей основные функции операционной системы.

PostgreSQL – объектно-реляционная система управления базами данных (СУБД), которая широко используется для хранения и обработки структурированных данных.

REST API — архитектурный стиль веб-служб, который использует протокол HTTP для передачи данных между клиентом и сервером.

Spring — фреймворк для разработки приложений на языке Java.

SpringBoot – платформа на основе Java с открытым исходным кодом, используемая для создания автономных приложений на базе Spring промышленного уровня с минимальными усилиями.

ToolBar – интерфейсный элемент программного обеспечения, представляющий собой горизонтальную или вертикальную панель с набором инструментов или команд, обычно расположенных в виде иконок или кнопок.

Ubuntu – свободная операционная система с открытым исходным кодом, основанная на ядре Linux и предназначенная для персональных компьютеров, серверов, облачных и встраиваемых систем.

Авторизованный пользователь – пользователь, который прошел процедуру аутентификации и получил доступ к определенным ресурсам, функциям или услугам в рамках системы или приложения.

Анонимный пользователь – пользователь, который не прошел процедуру аутентификации или идентификации при доступе к ресурсам, функциям или услугам в рамках системы или приложения.

Аутентификация – процесс проверки подлинности личности или учетных данных пользователя для подтверждения его идентичности.

Арендатор – лицо или организация, которая арендует или заключает договор аренды на использование недвижимого имущества или других ресурсов от владельца или арендодателя.

Арендодатель – лицо, организация или компания, которая владеет имуществом или другими ресурсами и сдаёт их в аренду другим лицам или организациям на условиях, определенных в договоре аренды.

Макет – предварительное или временное представление дизайна или композиции объекта, будь то веб-сайт, приложение, брошюра, журнал, или любой другой тип продукта.

Микросервис – подход к разработке программного обеспечения, при котором приложение строится как набор небольших, автономных, самодостаточных сервисов, каждый из которых решает отдельную бизнес-задачу или функциональность.

Токен – строка символов или чисел, которая используется для представления идентификационной информации или доступа к определенным ресурсам, функциям или услугам в информационной системе или приложении.

Триада CIA – акроним, используемый для обозначения трех основных аспектов информационной безопасности: конфиденциальности (Confidentiality), целостности (Integrity) и доступности (Availability). Эти три аспекта являются основными принципами, на которых строится система защиты информации и данных в информационной безопасности.

Фрейм – основной элемент дизайна в Фигме. Это законченный документ, который может быть страницей сайта или экраном мобильного приложения.

2 Общие сведения

2.1 Полное наименование системы и ее условное обозначение

Полное наименование системы: «Сервис для поиска и аренды оборудования и инструментов для домашнего ремонта и строительства».

Краткое наименование приложения: «RenTool».

2.2 Наименование исполнителя и заказчика приложения

Заказчик: Старший преподаватель Тарасов Вячеслав Сергеевич, Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Программирования и Информационных Технологий.

Разработчик: «2» команда группы «9».

Состав команды разработчика:

- Бучнев Дмитрий Вадимович;
- Ремезов Вадим Николаевич ;
- Клоков Евгений Александрович.

2.3 Перечень документов, на основании которых создается приложение

Данное приложение будет создаваться на основании следующих документов:

- Федерального закона от 27.07.2006 N 152-ФЗ "О персональных данных" [1].

2.4 Плановый срок начала и окончания работ

Плановый срок начала работ: 28 февраля 2024 года .

Плановый срок окончания работ: 10 июня 2024 года.

2.5 Состав и содержание работ по созданию системы

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

- сбор необходимой информации, постановка целей приложения, которые в будущем должны быть достигнуты;
- анализ предметной области, анализ прямых и косвенных конкурентов, выделение интересующих категорий исследования, оценка показателей качества продуктов;
- разработка ТЗ, создание презентации;
- построение предметной модели программы, описание спецификаций данных, определение связей между сущностями, построение концептуальной модели БД, построение логической модели БД;
- разработка рабочего проекта, состоящего из написания, отладки и корректировки кода программы;
- проведение тестирования и доработка продукта по замечаниям и предложениям.

2.6 Порядок контроля и приемки разрабатываемой системы

Предварительные отчёты по работе будут проводиться в следующие сроки:

- 1 аттестация (середина марта 2024) – создан репозиторий проекта на GitHub, создан проект Miro с общей логикой системы, написана документация по проекту и готовое техническое задание;

—2 аттестация (конец апреля 2024) – написана основополагающая часть кода приложения, реализована БД и ее взаимодействие с сервером, проведена отладка и доработка кода;

—3 аттестация (конец мая 2024) – проведено тестирование приложения, разработан курсовой проект, выполнены завершающие работы по доработке приложения, предоставлена готовая система.

3 Цели и назначение создания приложения

3.1 Цели создания приложения

Целями выполнения работ по созданию приложения «RenTool» является:

- реализация возможности арендовать инструмент;
- сокращение простоя оборудования;
- расширение клиентской базы.

3.2 Назначение приложения

Мобильное приложение «RenTool» предоставляет возможность физическому лицу арендовать представленное оборудование. Пользователи (арендаторы) могут просматривать доступные предложения, выбирать необходимое оборудование и инструменты, а также оформлять заявки на аренду.

4 Требования к структуре приложения в целом

Работа системы будет представлять из себя клиент серверное взаимодействие. Клиентом будет выступать мобильное приложение или веб-браузер. Сервер посредством REST API возвращает пользователю необходимые данные.

Серверная часть приложения будет спроектирована согласно шаблону проектирования API Gateway. Общая архитектура системы представлен на Рисунке 1.

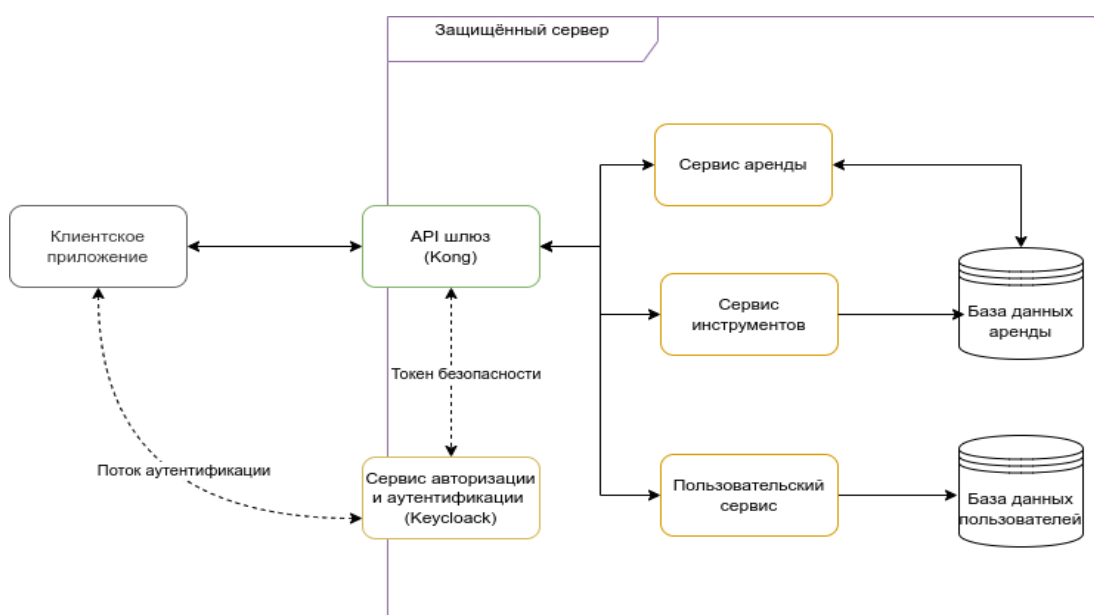


Рисунок 1 - Общая архитектура системы

4.1 Перечень подсистем, их назначение и основные характеристики

4.1.1 Пользовательский сервис

Пользовательский сервис является разрабатываемым и предназначен для предоставления пользователям системы (арендаторам) получать информацию о статусе своих заказов, а так же базовую информацию о своем профиле.

4.1.2 Сервис аренды

Сервис аренды является разрабатываемым и предназначен для предоставления пользователям системы (арендаторам) записи заявки на аренду.

4.1.3 Сервис инструментов

Сервис инструментов является разрабатываемым и предназначен для получения информации о инструменте, осуществления поиска и фильтрации инструментов.

4.1.4 Сервис авторизации и аутентификации

Сервис авторизации и аутентификации является интеграционным сервисом, предназначенным для обеспечения входа и регистрации пользователей в системе. Он генерирует токены авторизации для предоставления доступа к API – шлюзу и другим сервисам системы "RenTool". Этот сервис обеспечивает безопасность и защиту личных данных пользователей, а также управление их учетными записями.

4.1.5 Требования к способам и средствам обеспечения информационного взаимодействия компонентов системы

Информационное взаимодействие между клиентом, которым может выступать как мобильное приложение, так и веб-браузер и серверной части системы должно осуществляться путём использования протокола HTTPS.

4.2 Требования к функциям (задачам), выполняемыми приложением

4.2.1 Требования к сервису аренды

К разрабатываемой подсистеме сервиса аренды выдвигаются следующие функциональные требования:

- запись заявки на оформление аренды;
- изменение статуса заказа;
- предварительный отказ от арендного оборудования, при условии, что пользователь зарегистрирован;
- продление арендного оборудования, при условии, что пользователь зарегистрирован;
- информирование пользователя о статусе принятия заявки на оформление аренды.

4.2.2 Требования к пользовательскому сервису

К разрабатываемой подсистеме пользовательского сервиса выдвигаются следующие функциональные требования:

- предоставление информации профиля пользователя при условии, что пользователь зарегистрирован;

- обновление пользовательского профиля, при условии, что пользователь зарегистрирован.

4.2.3 Требования к сервису инструментов

К разрабатываемой подсистеме сервиса инструментов выдвигаются следующие функциональные требования:

- добавление нового инструмента в базу данных;
- предоставление информации о конкретном инструменте;
- поиск инструментов;
- фильтрация инструментов;
- удаление инструмента из базы данных;
- обновление информации инструмента;
- изменение статуса инструмента.

4.2.4 Требования к API – шлюзу

К API – шлюзу выдвигаются следующие функциональные требования:

- валидация токена авторизации;
- перенаправление запросов клиента к службам системы.

4.2.5 Требования к сервису авторизации и аутентификации

К интеграционной подсистеме авторизации и аутентификации выдвигаются следующие функциональные требования:

- авторизация и аутентификация пользователей;
- регистрация пользователей;
- предоставление JWT.

4.3 Требования к видам обеспечения системы

4.3.1 Требования к лингвистическому обеспечению системы

Система должна поддерживать интерфейс на русском языке.

4.3.2 Требования к программному обеспечению системы

Требования к программному обеспечению клиента мобильного приложения:

—приложение должно устанавливаться и работать на мобильных устройствах под управлением операционной системы Android версии 9 и выше;

—для реализации должен быть использован фреймворк Flutter из-за своей гибкости, мультиплатформенной поддержки и скорости разработки.

Требования к программному обеспечению серверной части:

—серверная часть приложения будет реализована согласно API Gateway подходу с использованием микросервисной архитектуры. Это поможет упростить управление и масштабирование приложения;

—в качестве основной операционной системы на сервере, должна быть Ubuntu, поскольку она предлагает: стабильность и надёжность, широкую поддержку производителя, безопасность, поддержку облачных вычислений;

—СУБД для разрабатываемых сервисов должна использоваться PostgreSQL версии 14. Она была выбрана из-за надёжности, стабильности и скорости работы;

—язык программирования для написания микросервисов должен быть Java версии 21. Так как он не зависит от платформы. Можно создать

Java-приложение на Linux, скомпилировать его в байт-код и запустить на любой другой платформе, поддерживающей JVM – виртуальную машину Java;

—основной фреймворк, который необходимо использовать SpringBoot. SpringBoot предлагает ряд преимуществ, которые делают его отличным выбором для разработки веб-приложений Java, а именно: упрощение разработки, интеграция с экосистемой Spring, автоматическая настройка, поддержка тестирования, производительность, поддержка микросервисов;

—API шлюзом будет выступать готовый сервис Kong так как он оптимизирован для работы с микросервисами и распределенными архитектурами, а так же обеспечивает надежную информационную безопасность;

—в качестве сервиса авторизации, регистрации, а так же аутентификации будет использован Keycloak, в связи с тем, что он обладает следующим рядом преимуществ: универсальность, масштабируемость, простая интеграция с другим сервисами, хорошо написанная документация.

4.4 Общие технические требования к системе

4.4.1 Требования по безопасности

Система должна соответствовать триаде CIA, а именно:

- конфиденциальности;
- целостности;
- доступности.

Конфиденциальность представляет из себя то, что доступ к информации должен быть ограничен и предоставлен только тем, кто имеет на это право. Это будет достигнуто путем внедрения в систему процедуры аутентификации.

Целостность представляет из себя то, что информация остаётся неповрежденной и актуальной. Это будет достигнуто путем внедрения в систему алгоритмов шифрования.

Доступность представляет из себя то, что информация и ресурсы всегда доступны для тех, кто должен ими пользоваться. Это будет достигнуто путем внедрения резервного копирования данных и защиты от DDoS-атак.

4.5 Требования к оформлению мобильного приложения

4.5.1 Общие требования к оформлению мобильного приложения

Мобильное приложение должно быть оформлен в единой цветовой палитре. У фреймов приложения должен быть единый стиль. В мобильном приложении должен присутствовать разработанный логотип. Основной шрифт используемый в мобильном приложении – Montseratt. На Рисунке 2 представлены основные материалы используемые при составлении макетов, а именно:

- основной логотип приложения;
- основной шрифт;
- основная палитра цветов;
- шаблоны цветов используемые при создании макетов;
- иконки используемые внутри приложения;
- ToolBar выступающий основным элементов навигации по приложению.

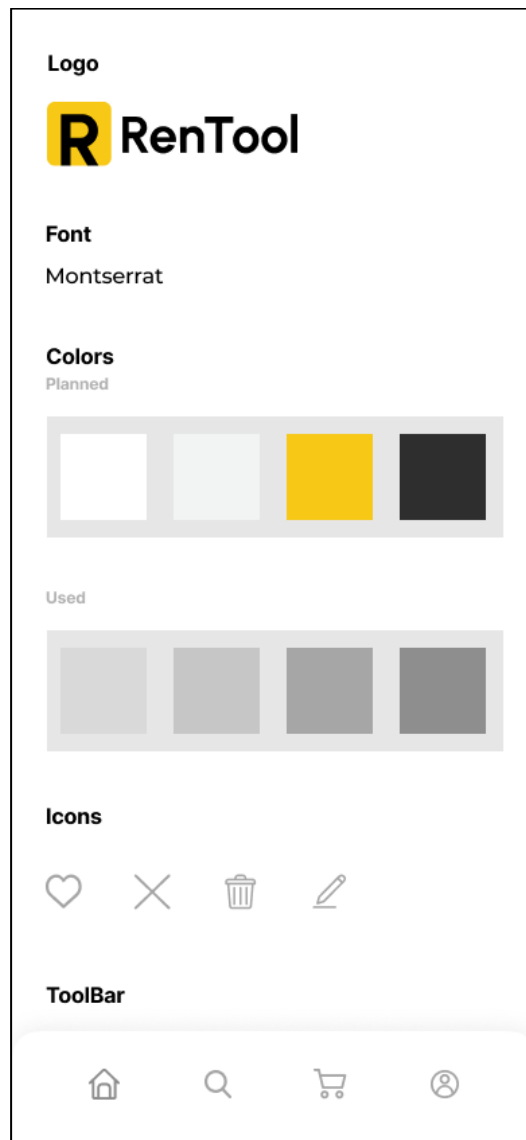


Рисунок 2 - Материалы использованные при создании макета

4.5.2 Требования к сплэш фрейму

Пользователи которые могут увидеть данный фрейм: анонимный, зарегистрированный.

При холодном запуске приложения необходимо показывать фрейм на котором будет изображена иконка логотипа. На Рисунке 3 представлен пример как это должно выглядеть.

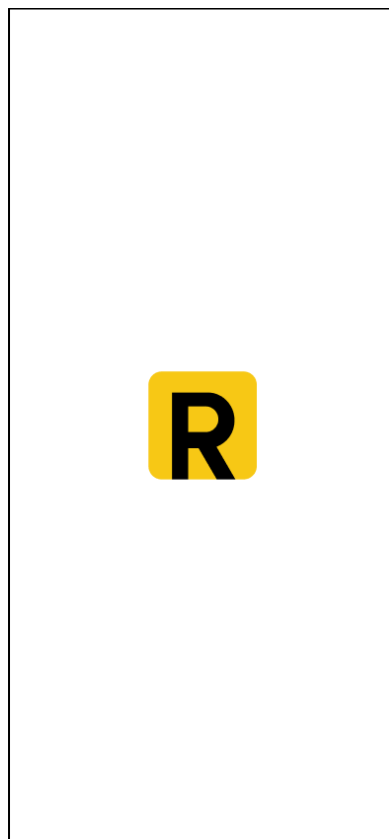


Рисунок 3 - Макет сплэш фрейма

4.5.3 Требования к фрейму ленты объявлений

Пользователи которые могут увидеть данный фрейм: анонимный, зарегистрированный.

Данный фрейм представляет из себя текстово-графические объявления. Основные элементы расположенные на этом фрейме:

- строка поиска, позволяющая задать фильтр поиска;
- рекламный постер;
- раздел «Избранное», где будет расположен список карточек избранных пользователем инструментов;
- ToolBar выступающий основным элементов навигации по приложению.

На Рисунке 4 представлен макет ленты объявлений.



Рисунок 4 - Макет ленты объявлений

4.5.4 Требования к фрейму со списком инструментов по заданному фильтру

Пользователи которые могут увидеть данный фрейм: анонимный, зарегистрированный.

На этом фрейме располагаются инструменты, которые были заданы в строке поиска из ленты объявлений. Основные элементы, которые расположены на этом фрейме:

- строка поиска;
- список инструментов представляющий из себя карточки товара;
- ToolBar выступающий основным элементов навигации по приложению.

На Рисунке 5 представлен макет со списком инструментов по заданному фильтру.



Рисунок 5 - Макет со списком инструментов по заданному фильтру

4.5.5 Требования к фрейму карточки инструмента

Пользователи которые могут увидеть данный фрейм: анонимный, зарегистрированный.

Этот фрейм представляет из себя карточку с фотографией, названием, описанием инструмента. Здесь пользователь может задать количество добавляемых инструментов в корзину заказа. Если анонимный пользователь попытается добавить инструмент в корзину, то ему будет предложено войти или зарегистрироваться в приложении.

На этой карточке располагается иконка, позволяющая добавить инструмент в избранное. Так же здесь расположена иконка, позволяющая закрыть карточку и вернуться к просмотру списка инструментов.

Основные элементы, которые расположены на этом фрейме:

- иконка добавления в «Избранное»;
- иконка позволяющая закрыть карточку;
- фото инструмента;
- название инструмента;
- описание инструмента;
- кнопка добавления инструмента в корзину заказа;
- кнопки позволяющие регулировать количество добавляемых в корзину заказа инструментов;
- ToolBar выступающий основным элементов навигации по приложению.

На Рисунке 6 представлен макет карточки инструмента.

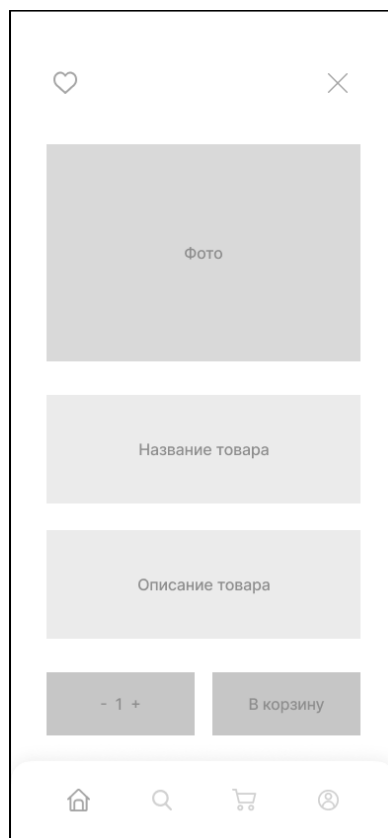


Рисунок 6 - Макет карточки инструмента

4.5.6 Требования к фрейму корзины

Пользователи которые могут увидеть данный фрейм: зарегистрированный.

На фрейме корзины располагается список добавленных инструментов для дальнейшего оформления заказа. Здесь пользователь может удостовериться в том, что он выбрал, ознакомиться с итоговой ценой заказа.

Пользователь может уменьшить или увеличить количество заказываемых инструментов. Если пользователь хочет убрать определённый инструмент из списка полностью, то он может нажать кнопку уменьшения количества инструментов, когда количество станет ноль, то инструмент уберётся из корзины автоматически.

В верху экрана справа расположена кнопка позволяющая очистить полностью корзину заказов. Снизу располагается кнопка к переходу оформления заказа.

Основные элементы, которые расположены на этом фрейме:

- название экрана;
- иконка корзины;
- список инструментов;
- кнопка перехода к оформлению заказа;
- ToolBar выступающий основным элементов навигации по приложению.

На Рисунке 7 представлен макет карточки инструмента.



Рисунок 7 - Макет корзины

4.5.7 Требования к фрейму оформление заказа

Пользователи которые могут увидеть данный фрейм: зарегистрированный.

На фрейме оформление заказа пользователь может выбрать способ доставки, дату начала и конца аренды, способ оплаты. Здесь располагается итоговая сумма заказа.

Основные элементы, которые расположены на этом фрейме:

- кнопки выбора доставки на дом или самовывоз;
- выбор даты начала и конца аренды;
- способ оплаты;
- итоговая сумма;
- кнопка заказать;

На Рисунке 8 представлен макет оформления заказа.

The wireframe shows a vertical sequence of elements within a rectangular frame. At the top, there are two side-by-side buttons labeled 'Доставка' (Delivery) and 'Самовывоз' (Self-pickup). Below these are four stacked rectangular boxes, each containing a label: 'Выбор даты начала и конца аренды' (Select start and end date of rental), 'Выбор времени начала и конца аренды' (Select start and end time of rental), 'Способ оплаты' (Payment method), and 'Итого (сумма)' (Total (sum)). At the bottom of the frame is a wide button labeled 'Заказать' (Order).

Рисунок 8 - Макет оформления заказа

4.5.8 Требования к фрейму каталог категорий

Пользователи которые могут увидеть данный фрейм: анонимный, зарегистрированный.

На этом фрейме расположена информация о категориях, на которые пользователь может нажать и они зададут фильтр для поиска.

Основные элементы, которые расположены на этом фрейме:

- поисковая строка;
- кнопка с названием категории;
- ToolBar выступающий основным элементов навигации по приложению.

На Рисунке 9 представлен макет каталога категорий.

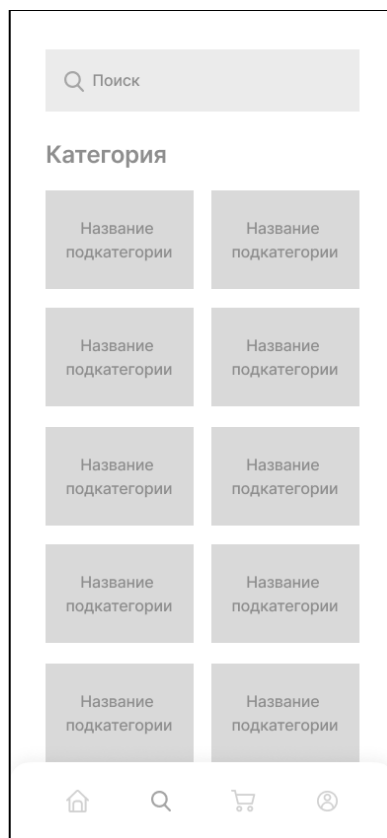


Рисунок 9 - Макет каталога категорий

4.5.9 Требования к фрейму профиля авторизованного пользователя

Пользователи которые могут увидеть данный фрейм: зарегистрированный.

На этом фрейме расположена основная пользовательская информация. Здесь можно узнать о количестве активных заказов. Узнать историю заказов.

Основные элементы, которые расположены на этом фрейме:

- имя пользователя;
- номер телефона;
- кнопка активных заказов;
- кнопка изменить пароль;
- кнопка выхода из аккаунта;
- ToolBar выступающий основным элементов навигации по приложению.

На Рисунке 10 представлен макет авторизованного профиля пользователя.

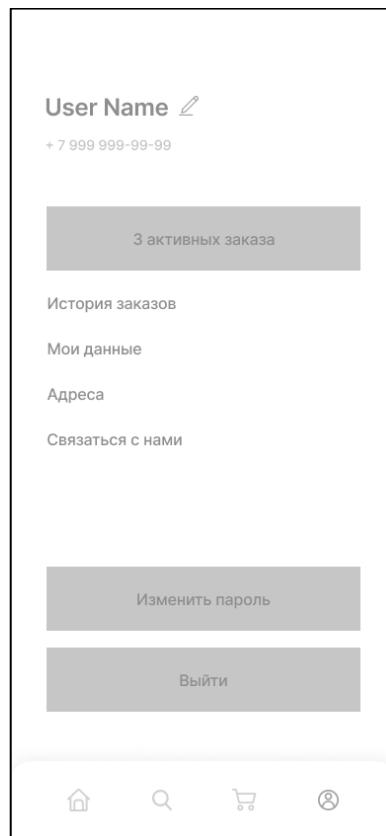


Рисунок 10 - Макет профиля пользователя

4.5.10 Требования к фрейму активных заказов

Пользователи которые могут увидеть данный фрейм: зарегистрированный.

На этом фрейме пользователь может узнать о состоянии активные заказы

Основные элементы, которые расположены на этом фрейме:

- список карточек инструментов;
- кнопка продлении аренды;
- ToolBar выступающий основным элементов навигации по приложению.

На Рисунке 11 представлен макет активных заказов.

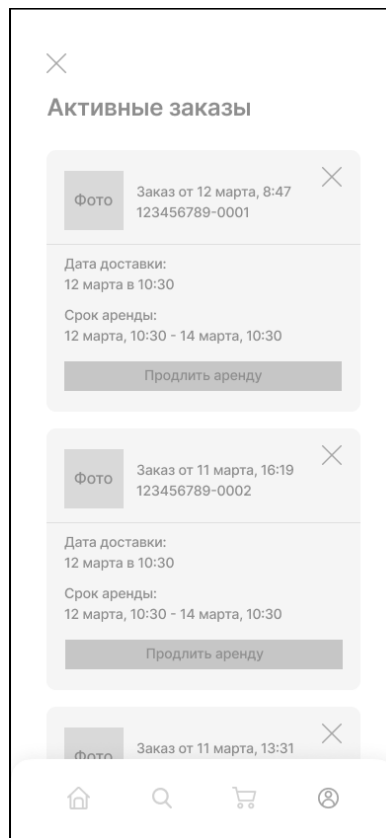


Рисунок 11 - Макет активных заказов

5 Источники разработки

1. ФЗ "О персональных данных" от 27.07.2006 N 152-ФЗ [В Интернете].
Доступно: https://www.consultant.ru/document/cons_doc_LAW_61801/