

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра системного анализа и автоматического управления

**РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ ОБМЕНА СООБЩЕНИЯМИ ДЛЯ
ОПЕРАЦИОННОЙ СИСТЕМЫ ANDROID
АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ**

студента 5 курса 551 группы

направления 09.03.04 – Программная инженерия

факультета компьютерных наук и информационных технологий

Колбасича Ильи Сергеевича

Научный руководитель:

Зав. кафедрой,

к. ф.-м. н., доцент

И. Е. Тананко

Зав. кафедрой:

к. ф.-м. н., доцент

И. Е. Тананко

Саратов 2023

ВВЕДЕНИЕ

Актуальность темы. Современный мир невозможно представить без мгновенного обмена информацией между людьми. Мы живем в эпоху быстрого интернета, где взаимодействие между людьми происходит с помощью электронных устройств, а коммуникации, осуществляемые через них, становятся все более разнообразными и удобными. Одним из таких способов общения является мобильное приложение для обмена сообщениями — чат.

Создание приложения-чата для Android-платформы является важной и актуальной темой в мире современных технологий. В настоящее время все больше людей пользуется мобильными устройствами на базе операционной системы Android, и разработчики со всего мира продолжают создавать новые приложения для удовлетворения потребностей пользователей.

Приложение-чат для Android предоставляет пользователям возможность мгновенного общения с друзьями, семьей и коллегами, независимо от расстояния между ними. Такое приложение может быть использовано для работы, обучения, общения или просто для развлечения.

Важность создания приложения-чата для Android подчеркивается тем, что мобильные устройства являются нашей повседневной необходимостью, а среди них Android-устройства занимают ведущие позиции. Приложение-чат является удобным и доступным способом общения для всех категорий пользователей Android-устройств, что делает его создание актуальной задачей для разработчиков. Кроме того, создание приложения-чата для Android является одним из самых интересных и востребованных направлений в сфере мобильной разработки. Приложения для обмена сообщениями занимают лидирующие позиции в мире мобильных приложений, и конкуренция на рынке постоянно растет. Таким образом, создание качественного приложения-чата для Android — это не только возможность предоставить удобный инструмент общения для пользователей, но и реализация

креативного потенциала разработчиков, а также бизнес-возможность для создания успешного проекта.

В данном контексте создание приложения-чата для Android становится неотъемлемой частью развития современного общества. Оно помогает людям быстро и легко общаться друг с другом, оставаться в контакте с близкими и друзьями, а также делиться информацией. В то же время, такие приложения могут быть полезными для бизнеса, помогая установить контакт между продавцом и покупателем, решить вопросы по заказу и доставке товаров, а также предоставить поддержку клиентам.

Многие люди используют мессенджеры для общения, даже если они находятся в одном и том же помещении. Это вызвано тем, что обмен сообщениями через интернет может быть более удобным и эффективным, чем общение вживую.

Поэтому разработка приложений чатов для мобильных устройств, в том числе для Android, является актуальной и важной темой. В настоящее время, когда все больше людей используют смартфоны, существует огромный спрос на приложения чатов. Они позволяют людям обмениваться сообщениями и информацией с друзьями, коллегами и родственниками в любое время и в любом месте.

С учетом всех перечисленных выше факторов, создание приложения-чата для Android является важным и актуальным вызовом для разработчиков. Такое приложение может упростить коммуникацию между людьми и сделать их жизнь более удобной и эффективной.

Цель бакалаврской работы – разработка приложения для обмена сообщениями для операционной системы Android.

Поставленная цель определила **следующие задачи**:

1. Получить основные сведения об ОС Android.
2. Рассмотреть приложения-аналоги для получения представления о необходимом функционале.
3. Проанализировать инструментальные средства для создания приложения.

4. Рассмотреть основные этапы создания приложения.
5. Выбрать подходящий шаблон проектирования.
6. Разработать приложение для обмена сообщениями для операционной системы Android.

Методологические основы разработки приложений для операционной системы Android представлены в работах Б. Филлипса, Я. Клифтона, Д. Гриффитса, Д. Скина, Я. Дарвина.

Практическая значимость бакалаврской работы. Получившееся в результате выполнения работы приложение представляет собой программный продукт, который готов к использованию на мобильных устройствах под управлением операционной системы Android. Оно также, в отличие от многих других мессенджеров, имеет низкие системные требования и может быть запущено даже на не самых производительных устройствах.

Структура и объём работы. Бакалаврская работа состоит из введения, 4 разделов, заключения, списка использованных источников и 1 приложения. Общий объём работы – 58 страниц, из них 49 страниц – основное содержание, включая 32 рисунка, цифровой носитель в качестве приложения, список использованных источников информации – 26 наименований.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Основные сведения об операционной система Android» посвящен рассмотрению первичной информации по операционной системе Android. Приводятся сведения об истории создания операционной системы, её особенностях и основных языках, используемых для разработки программного обеспечения для неё [5].

Второй раздел «Обзор существующих приложений-аналогов» посвящен обзору наиболее популярных мессенджеров для операционной системы Android, рассматриваются преимущества и недостатки каждого из них.

Третий раздел «Проектирование приложения и выбор инструментальных средств» посвящен рассмотрению архитектуры приложения, шаблона проектирования и основных инструментов, которые требуются для его создания.

Подраздел 3.1 посвящен рассмотрению клиент-серверной архитектуры в контексте её применения для разработки приложения.

Клиент-серверная архитектура широко используется в различных приложениях, в том числе и в мессенджерах для обмена сообщениями [1]. Она позволяет разделить логику работы приложения на две части: клиентскую и серверную. Клиентская часть приложения работает на устройстве пользователя и отвечает за отображение интерфейса, обработку пользовательских действий и отправку запросов на серверную часть приложения. Серверная часть же обрабатывает запросы от клиентов, хранит информацию о пользователях и сообщениях, а также отвечает за доставку сообщений до адресатов.

Использование клиент-серверной архитектуры позволяет разрабатывать масштабируемые приложения, способные обрабатывать большое количество пользователей. Кроме того, это позволяет разделить нагрузку между устройством пользователя и сервером, что улучшает производительность приложения.

Подраздел 3.2 посвящен рассмотрению шаблона проектирования MVVM, который широко применяется для разработки Android-приложений.

Данный шаблон позволяет разделить логику приложения и пользовательский интерфейс на три компонента: модель (Model), представление (View) и модель представления (ViewModel), как показано на рисунке 1.

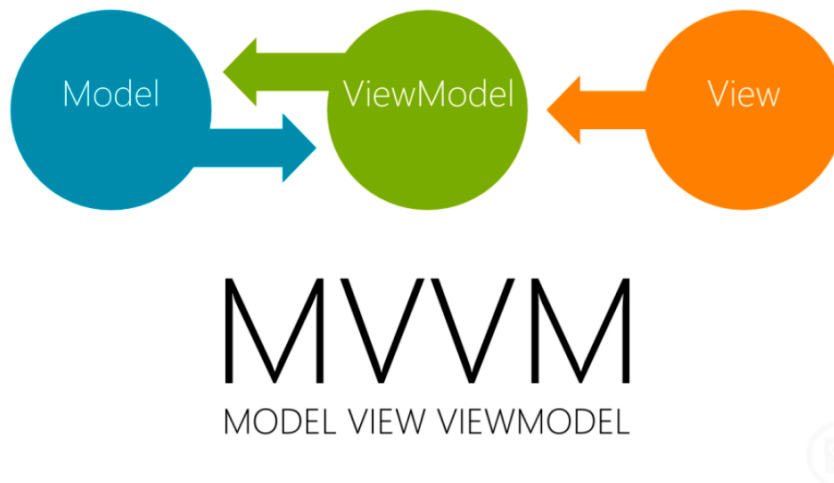


Рисунок 1 – Графическое представление шаблона проектирования MVVM

Модель (Model) — это компонент, который отвечает за обработку данных

и логику приложения. Он может содержать все данные, необходимые для работы приложения, включая данные из базы данных, файловой системы или из внешних источников. Он также может содержать методы для обработки данных и осуществления связи с другими компонентами приложения.

Представление (View) — это компонент, который отвечает за отображение

пользовательского интерфейса. Он может содержать все элементы пользовательского интерфейса, включая кнопки, текстовые поля, изображения и другие элементы. Он также может содержать методы для обработки пользовательских событий, таких как нажатие на кнопку.

Модель представления (ViewModel) — это компонент, который отвечает за связь между моделью и представлением. Он может содержать данные, необходимые для отображения пользовательского интерфейса и обработки пользовательских действий, таких как выбор элемента в списке [2]. Он также может содержать методы для обработки пользовательских событий, таких как нажатие на кнопку.

Подраздел 3.3 посвящен рассмотрению основных инструментальных средств, используемых для разработки Android-приложений, а именно: языка программирования Kotlin, IDE Android Studio и облачного сервиса Firebase.

Подраздел 3.4 посвящен построению use-case диаграммы, содержащей основной функционал приложения.

Use-case диаграммы обычно используются для определения требований к системе и для общения между разработчиками и заказчиками. Они также могут быть использованы в качестве основы для проектирования архитектуры системы и ее компонентов.

В use-case диаграммах обычно выделяют следующие виды взаимодействия:

- Обобщения (Generalizations) — это отношения, которые позволяют описывать отношения наследования между Use-Case'ами.
- Связи (Associations) — это отношения между акторами и прецедентами, которые описывают, как акторы используют прецеденты.
- Включения (Inclusions) — это отношения между прецедентами, которые описывают, как один прецедент может включать в себя другой прецедент.
- Расширения (Extensions) — это отношения между прецедентами, которые описывают, как один прецедент может расширять другой прецедент.

Для построения диаграммы был определён актёр, в данном случае реальный человек, то есть, некий пользователь, который взаимодействует с приложением. Далее определены варианты использования и обозначены их на схеме овалами. Затем установлены отношения между ними и пользователем, как показано на рисунке 2.

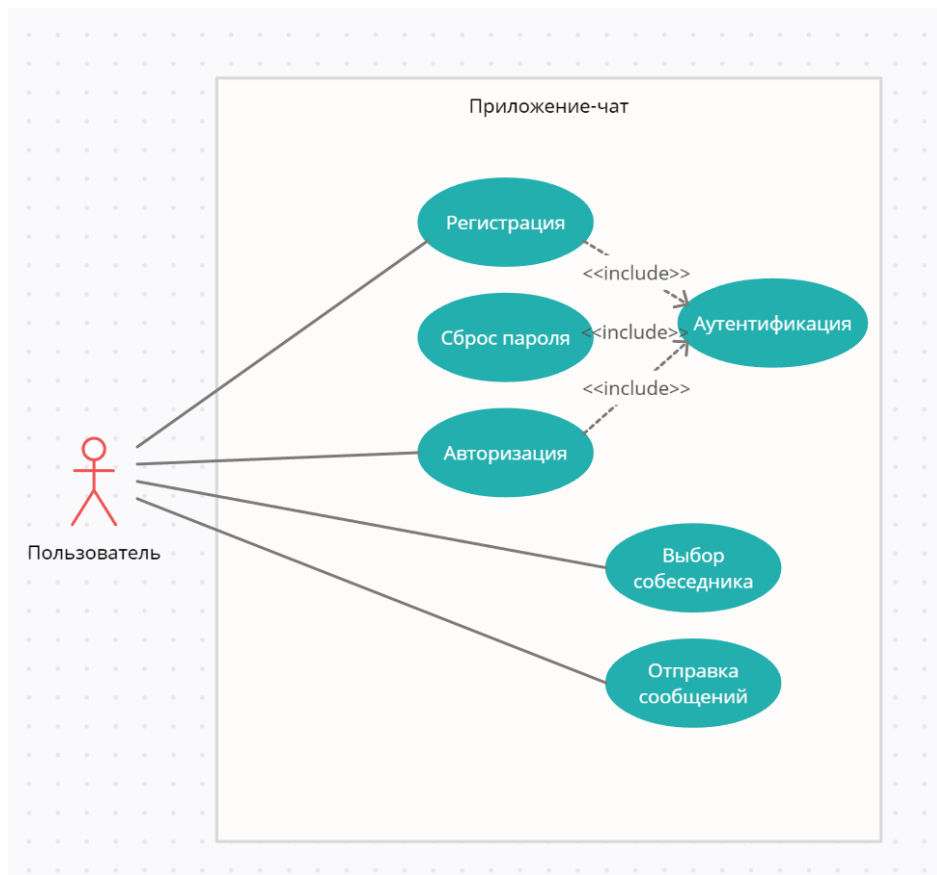


Рисунок 2 – Use-case диаграмма приложения

Четвертый раздел «Разработка приложения» посвящен рассмотрению всех основных этапов разработки, описанию структуры приложения и реализации приложения.

Подраздел 4.1 посвящен рассмотрению основных этапов разработки. Были определены такие этапы как:

- Подготовительный этап, определяющий требования к создаваемому приложению.
- Этап реализации программной части.
- Заключительный этап, на котором происходит тестирование приложения.

Подраздел 4.2 посвящен обзору основных компонентов каждого Android-приложения, а именно:

- Activity.
- Intent.

- Service.
- Broadcast Receiver.
- Content Provider.

Особое внимание уделено рассмотрению жизненного цикла Activity и объекту Intent, который выступает связующим звеном между всеми Activity приложения [3].

Приведены сведения об основных экранах приложения, которые реализуются в следующем подразделе, а именно:

- Экран логина, где пользователя будет возможность произвести вход в приложение под своим аккаунтом.
- Экран восстановления пароля, где можно будет восстановить забытый пароль, введя свою почту.
- Экран регистрации.
- Экран со списком пользователей, куда происходит переход после успешного логина.
- Экран чата с выбранным пользователем.

Подраздел 4.3 посвящен реализации программного кода приложения.

В качестве первого этапа здесь рассматривается создание xml-разметки для всех экранов приложения с описанием всех используемых при этом компонентов и особенностей взаимодействия с ними, а внешний вид получившихся экранов представлен в графическом виде.

Далее производится добавление облачного сервиса Firebase в проект, включая модули Authentication и Realtime Database. Описываются основные этапы взаимодействия с данным сервисом, создание в Firebase-проекте модулей Authentication и Realtime Database, производится их настройка и подключение Firebase к проекту в Android Studio для дальнейшей работы.

На следующем этапе происходит реализация логики приложения. Это включает в себя построение навигации внутри приложения, создание ViewModel для всех представленных экранов, создание адаптеров для

контейнера RecyclerView [4], реализуется взаимодействие с модулями Firebase.

На заключительном этапе происходит тестирование всего функционала получившегося приложения. Тестирование сопровождается графическим представлением экранов приложения и их функций, как, например, показано на рисунке 3.

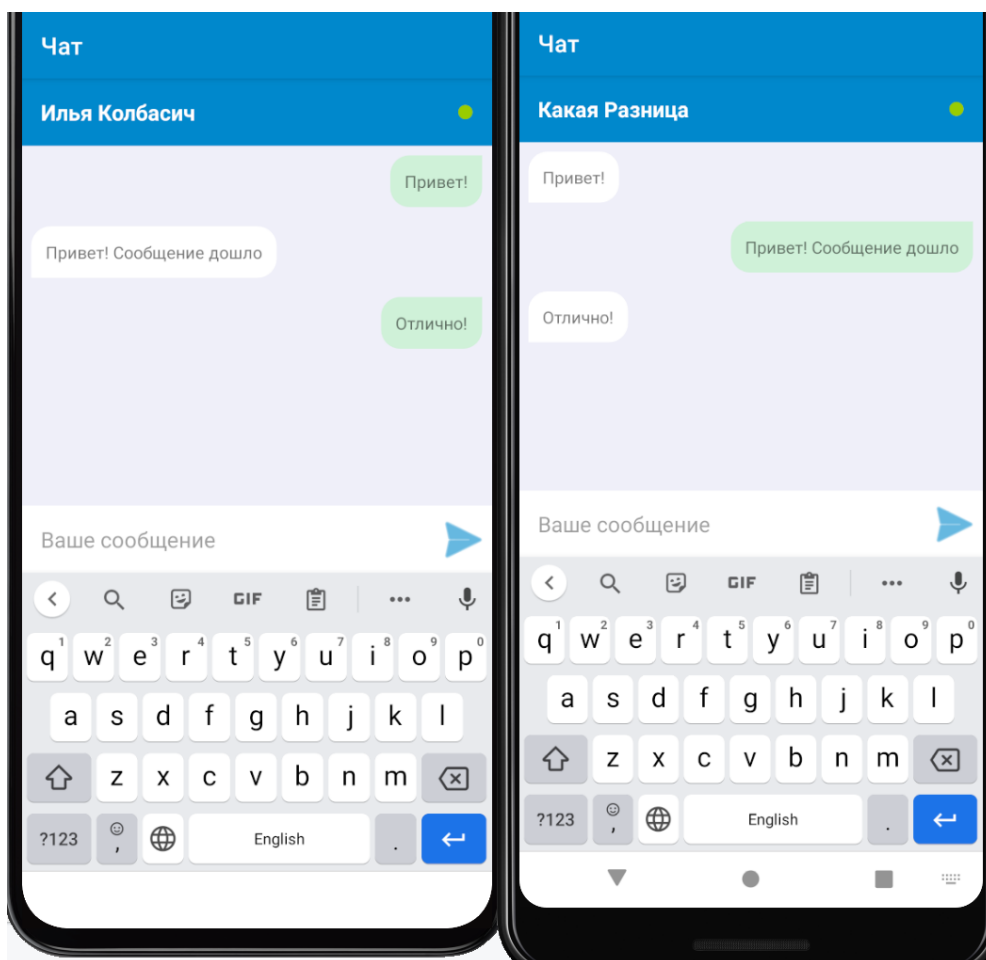


Рисунок 3 – Обмен сообщениями между пользователями

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы было разработано приложение для обмена сообщениями для операционной системы Android.

В процессе выполнения поставленных задач были получены основные сведения об операционной системе Android, рассмотрены приложения-аналоги, проанализированы инструментальные средства для создания приложения, рассмотрены основные этапы создания приложения, выбран шаблон проектирования и осуществлена разработка приложения.

В ходе выполнения работы были использованы на практике и расширены теоретические знания в области разработки приложений для операционной системы Android с использованием языка Kotlin, среды разработки Android Studio и сервиса Firebase.

Получившееся приложение было протестировано на работоспособность основного функционала.

Таким образом, можно сделать вывод, что цель выпускной квалификационной работы достигнута.

Основные источники информации:

1. Дарвин, Я. Android. Сборник рецептов. / Я. Дарвин. — Москва: Диалектика, 2019. — 768 с.
2. Клифтон, Я. Проектирование пользовательского интерфейса в Android / Я. Клифтон. — Москва: ДМК Пресс, 2017. — 452 с.
3. Филлипс, Б. Android. Программирование для профессионалов / Б. Филлипс. — Санкт-Петербург: Питер, 2021. — 704 с.
4. Скин, Д. Kotlin. Программирование для профессионалов / Д. Скин. — Санкт-Петербург: Питер, 2017. — 556 с.
5. Гриффитс, Д. Head First. Kotlin / Д. Гриффитс. — Санкт-Петербург: Питер, 2022. — 464 с.