

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

РАБОТА ПРОВЕРЕНА

ДОПУСТИТЬ К
ЗАЩИТЕ

Рецензент

Директор ЦИТ

ООО «ЧТЗ- Уралтрак»

_____ Л.К. Филатенко

« ___ » _____ 2020 г.

Заведующий кафедрой

_____ Г.И. Радченко

_____ 2020 г.

Создание корпоративного сетевого чата

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ

Руководитель работы,
к.т.н., доцент каф. ЭВМ

_____ Д.В. Топольский

« ___ » _____ 2020 г.

Автор работы,
студент группы КЭ-222

_____ В.В. Сенаторов

« ___ » _____ 2020 г.

Нормоконтроллер,
ст. преп. каф. ЭВМ

_____ С.В. Сяськов

« ___ » _____ 2020 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ
Заведующий кафедрой
ЭВМ
_____ Г.И.
Радченко
«__» _____ 2020 г.

ЗАДАНИЕ

на выполнение выпускной квалификационной работы магистра

студенту группы КЭ-222

Сенаторову Вячеславу Владимировичу

обучающемуся по направлению

09.04.01 «Информатика и вычислительная техника»

- 1. Тема работы:** «Создание корпоративного сетевого чата»
утверждена приказом ректора от «24» апреля 2020 года № 627
- 2. Срок сдачи студентом законченной работы:** 1 июня 2020 г.
- 3. Исходные данные к работе**
 - язык программирования – Java;
 - платформа разработки – Android;
 - база данных – Google Firebase.
- 4. Перечень подлежащих разработке вопросов**
 - введение и обзор литературы;

- формирование требований к системе;
- разработка мобильного приложения;
- тестирование и отладка;

5. Дата выдачи задания 1 декабря 2019 г.

Руководитель работы _____ /Д.В.Топольский

Студент _____ /В.В. Сенаторов

КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Введение и обзор литературы	01.03.2020	
Формирование требований к системе	01.04.2020	
Разработка мобильного приложения	01.05.2020	
Тестирование и отладка	15.05.2020	
Компоновка текста работы и сдача на нормоконтроль	24.05.2020	
Подготовка презентации и доклада	30.05.2020	

Руководитель работы _____ / Д.В. Топольский

Студент _____ /В.В. Сенаторов

АННОТАЦИЯ

Сенаторов В.В. Разработка и создание корпоративного
сетевого чата на Android: ФГАОУ ВО
«ЮУрГУ (НИУ)», ВШЭКН; 2020, 118 с. 25 ил.,
библиограф. список – 15 наим.

В магистерской диссертации был произведен обзор литературы, выбраны компоненты для будущего приложения, выбраны функциональные и нефункциональные требования к системе, сформирована архитектура приложения и разработан мобильный корпоративный сетевой чат. Тестирование приложение выполнено успешно.

Оглавление

Введение.....	7
1. Введение и обзор литературы.....	9
1.1 История чата.....	9
1.2 Среда разработки Android Studio	12
1.3 Язык программирования Java.....	13
1.4 Android Virtual Device (AVD).....	15
1.5 База данных Firebase Realtime	16
1.6 Сравнение отечественных и зарубежных технологий и решений.....	20
2. Формирование требований к системе.....	22
2.1 Проектирование базы данных	22
2.2 Интерфейс пользователя.....	24
2.3 Обмен сообщениями.....	25
2.4 Состояние пользователя.....	26
3. Разработка мобильного приложения	27
3.1 Создание формы авторизации	27
3.2 Создание формы регистрации	28
3.3 Создание формы регистрации с помощью мобильного телефона	29
3.4 Создание окна восстановления пароля.....	30
3.5 Создание настроек профиля	33
3.6 Создание формы сообщений	34
3.7 Создание формы контактов	35
3.8 Создание формы меню приложения.....	36
3.9 Хранение в базе данных информации	38
4. Тестирование и отладка	40
Заключение	45
Библиографический список	46
Приложение А.1	48
Приложение А.2	49
Приложение А.3	50
Приложение А.4	51
Приложение А.5	52
Приложение А.6	53
Приложение А.7	54
Приложение А.8	55
Приложение А.9	56
Приложение Б.....	57

Введение

В современных условиях динамично развивается рынок комплексных интегрированных систем автоматизации проектирование информационной подсистемы деятельности малого и крупного бизнеса. Эта организация может иметь самые различные размеры с разнообразными схемами иерархии. Такие системы предназначены для решения задач как организации в целом (управление финансовыми ресурсами, управление актуальной информацией, планирование и учет, поиск клиентов, заемщиков и кредиторов), так и уровня его подразделений и организационных отделов.

Для одной из таких организаций была поставлена задача, упростить общение между сотрудниками одной организации и обеспечить их быстрым обменом данными.

С директором организации было обговорены условия и поставлены задачи, продуман интерфейс приложения, выбрана платформа, для которой создаётся приложение.

Финальное тестирование приложение будет проходить на специально закупленных Android устройствах, которые будут выданы руководителям всех отделов организации.

В организации есть свой системный администратор, которому будет передано данное приложение для дальнейшего администрирования.

Срок сдачи готового приложения назначен на 1 сентября 2020 года.

Цели и задачи

Целью дипломной работы является проектирование и разработка средства общения между людьми одной организации.

Для решения поставленной цели необходимо решение следующих задач:

1. Изучение теоретических основ проектирования сетевых приложений;
2. Исследование методов написания сетевых приложений;
3. Выбор инструментов для разработки приложения;

4. Создание архитектуры приложения;
5. Разработка и реализация мобильного приложения;
6. Отладка и тестирование приложения.

Значимость работы состоит в том, что фирма не может использовать сторонние сетевые программы таких как: Skype, Telegram, Viber, WhatsApp, или ICQ. По мимо всего прочего сетевой чат может использоваться там, где есть необходимость сократить затраты на телефонные переговоры между сотрудниками, находящимся на разных этажах либо же в разных зданиях.

Значимость работы очень велика, в каких-то моментах сетевой чат может выступать даже как средство повышения эффективности работы, либо просто как дешевое средство коммуникации, поэтому выбор сетевого чата вместо какого-либо мессенджера — это еще и в пользу сетевой безопасности.

Актуальность

Данное приложение является актуальным, так как является заказом крупной региональной юридической компании, этот факт свидетельствует о том, что приложением будут пользоваться сотрудники одной организации, для которых оно разрабатывается.

1. Введение и обзор литературы

1.1 История чата

С конца 2000-х годов в нашей жизни прочно заняли свое место социальные сети. Однако Всемирная паутина в своем нынешнем виде существует с начала 1990-х, а общаться в глобальных сетях люди начали еще раньше.

Программы для обмена мгновенными сообщениями появились раньше, чем начал существовать современный интернет. Первая программа для обмена сообщениями в режиме реального времени называлась Talkomatic. Её использовали для переписки пользователи системы обучения PLATO.

Первый релиз Talkomatic вышел еще в 1973 году. В Talkomatic могли одновременно общаться до 5 человек – экран разделялся на одинаковые горизонтальные области, в каждой из которых появлялись сообщения, отправленные участником.

```
TALK (Channel 1)
$$$ Dr. Wool          woolley / cerl          1-20
  Okay, that's better.
  I was hoping Papa Del's is still around - it was the best
$$$ brian            brian dear / uofdel          1-25
  is garcia's still in business?
$$$ Δ                peltz / s                5-16
  I haven't eaten at delz for a while
  it's better than garcias tho.
$$$ Loren            platte / unl            1-18
  *grin* and *sigh*
$$$ Doug            dwb / nginear          1-0
  hmmm
```

Рисунок 1 – Первый чат Talkomatic

В начале 1980-х годов в Массачусетском технологическом институте также появилась система обмена мгновенными сообщениями – ею пользовались сотрудники учебного заведения в своей локальной сети. Подобные системы постепенно появлялись и в других университетах, в основном в США.

При этом еще в конце 1970-х начали появляться первые сетевые доски объявлений (bulletin board system, BBS). Сначала на подобных досках публиковались только объявления и другие отдельные сообщения, но с середины 1980-х годов IT-компании начали постепенно создавать расширения, позволяющие запускать в BBS конференции, а затем и чаты. Что представляла собой типичная BBS технически? Обычно это был компьютер с модемом, подключение происходило по телефонной линии. К BBS последовательно дозванивались пользователи и скачивали оттуда информацию.

Также в середине 1980-х годов появилась международная сеть FidoNet (об истории FidoNet мы еще напишем отдельно). Эта сеть объединила множество BBS-станций. Работа с ресурсами в FidoNet выглядела примерно так: пользователь по телефонной линии связывался со своим узлом сети, забирал и отправлял сообщения (обычно это происходило раз в сутки с помощью специальных программ). Ответы других пользователей читались уже после отключения от сети.

Первые современные чаты

В конце 1980-х годов технологии уже позволяли общаться в глобальных сетях в режиме реального времени. В 1988 году был разработан протокол IRC – одно из первых общедоступных решений для обмена мгновенными сообщениями. IRC позволяет организовывать групповые чаты и обмениваться сообщениями с конкретным пользователем. Программы для обмена сообщениями в IRC быстро набирали популярность, к 1990 году пользователи IRC появились и в СССР. Чаты, работающие по протоколу IRC, существуют до сих пор – можно скачать любую бесплатную программу-клиент и

подключиться к IRC-конференции. Правда, сейчас в IRC-чатах уже немногочисленно [3].

Первые чаты были устроены довольно просто – страница обновлялась раз в несколько секунд, и на ней появлялись новые сообщения, отправленные пользователями. Первые русскоязычные веб-чаты появились в середине 1990-х годов и быстро приобрели значительную популярность (особенно много посетителей было у chat.ru, krovatka.ru и некоторых других сайтов). Причина проста – для начала общения не требовалось скачивать и настраивать программы, достаточно было лишь зайти на определенную страницу [3].

Стало понятно, что программы для общения в IRC-чатах довольно сложны для большинства новых пользователей Интернета. Веб-чаты были проще, но в них за порядком следили суровые модераторы и блокировали отдельных пользователей. Поэтому рано или поздно должны были возникнуть альтернативные чат-клиенты. И они появились.

Во второй половине 1990-х годов стали развиваться альтернативы IRC и веб-чатам. Самой популярной на постсоветском пространстве системой обмена сообщениями стала ICQ – в простонародье «аська».

В настоящее время, чаты перешли на более высокий уровень. Появилось много новых возможностей, в отличие от более старых версий. Стало доступно обмениваться не только СМС сообщениями, но и совершать аудио и видео звонки. Не нужно как раньше ждать обновления переписки, сейчас это происходит мгновенно.

1.2 Среда разработки Android Studio

Программа Android Studio — интегрированная среда (IDE) для разработки мобильного софта в операционной системе Android. Большая часть мобильных приложений создается на ее основе.

Android Studio — бесплатный софт от Google, на котором разрабатываются приложения для следующих типов устройств: смартфоны, планшеты, умные часы. Программа стала официальной средой для ОС Android с 2014 г. Ранее использовалась платформа Eclipse.

Функционал программы:

1. С помощью Эмулятора Android просматривают этапы работы с ПК, не устанавливая приложение на смартфон;
2. Анализатор APK предназначен для контроля количества файлов и их размера;
3. Доступен режим сравнения двух пакетов, к примеру, устаревшей и обновленной версии;
4. Редактор макетов — инструмент для настройки интерфейса;
5. Профилирование в режиме реального времени отображает, сколько ресурсов потребляет приложение;
6. Новый формат публикации приложений от Android Studio Bundle, с помощью которого оптимизация проходит без изменения кода.

Во время создания приложений и утилит для операционной системы Android, пользователь программного обеспечения может наблюдать за изменениями в проекте, в режиме реального времени.

Android Studio — универсальная среда разработки, так как позволяет оптимизировать работу будущих приложения для работы не только на смартфонах, но и на планшете, портативных ПК, которые работают на основе рассматриваемой операционной системы.

В программу встроен эмулятор, позволяющий проверить корректную работу приложения на устройствах с разными экранами, с различными соотношениями сторон. Особенно актуальной эта функция стала после входа в тренды смартфонов, в которых установлены экраны с соотношением сторон 18:9.

Отличительная особенность эмулятора — просмотр приблизительных показателей производительности при запуске приложения на самых популярных устройствах.

Системные требования Android Studio

Windows, доступна для 32-битной и 64-битной ОС.

ОЗУ, требуемая для работы, начинается с 3 Гб.

Рекомендованная оперативная память на ПК — от 8 Гб.

На диске нужно более двух Гб свободного места, чтобы программа функционировала бесперебойно.

Наименьшее разрешение экрана — 1280 x 800.

1.3 Язык программирования Java

На сегодняшний день язык Java является одним из самых распространенных и популярных языков программирования. Первая версия языка появилась еще в 1996 году в недрах компании Sun Microsystems, впоследствии поглощенной компанией Oracle. Java задумывался как универсальный язык программирования, который можно применять для различного рода задач. И к настоящему времени язык Java проделал большой путь, было издано множество различных версий. Текущей версией является Java 12, которая вышла в марте 2019 года. А Java превратилась из просто универсального языка в целую платформу и экосистему, которая объединяет различные технологии, используемые в целом ряде задач: от создания десктопных приложений до написания крупных веб-порталов и сервисов. Кроме того, язык Java активно применяется для создания программного

обеспечения для целого ряда устройств: обычных ПК, планшетов, смартфонов и мобильных телефонов и даже бытовой техники. Достаточно вспомнить популярность мобильной ОС Android, большинство программ для которой пишутся именно на Java[5].

Особенности Java

Ключевой особенностью языка Java является то, что его код сначала транслируется в специальный байт-код, независимый от платформы. А затем этот байт-код выполняется виртуальной машиной JVM (Java Virtual Machine). В этом плане Java отличается от стандартных интерпретируемых языков как PHP или Perl, код которых сразу же выполняется интерпретатором. В то же время Java не является и чисто компилируемым языком, как C или C++[6].

Подобная архитектура обеспечивает кроссплатформенность и аппаратную переносимость программ на Java, благодаря чему подобные программы без перекомпиляции могут выполняться на различных платформах - Windows, Linux, Mac OS и т.д. Для каждой из платформ может быть своя реализация виртуальной машины JVM, но каждая из них может выполнять один и тот же код.

Java является языком с Си-подобным синтаксисом и близок в этом отношении к C/C++ и C#. Поэтому, если вы знакомы с одним из этих языков, то овладеть Java будет легче[7].

Еще одной ключевой особенностью Java является то, что она поддерживает автоматическую сборку мусора. А это значит, что вам не надо освобождать вручную память от ранее использовавшихся объектов, как в C++, так как сборщик мусора это сделает автоматически за вас.

Java является объектно-ориентированным языком. Он поддерживает полиморфизм, наследование, статическую типизацию. Объектно-ориентированный подход позволяет решить задачи по построению крупных, но в тоже время гибких, масштабируемых и расширяемых приложений.

Установка Java

Для работы программ на языке Java на целевой машине должна быть установлена JRE (Java Runtime Environment). JRE представляет минимальную реализацию виртуальной машины, а также библиотеку классов. Поэтому, если мы хотим запускать программы, то нам надо установить JRE. Для каждой конкретной платформы имеется своя версия JRE.

Однако, так как мы собираемся не только запускать программы, но и разрабатывать их, нам потребуется специальный комплект для разработки JDK (Java Development Kit). JDK уже содержит JRE, а также включает ряд дополнительных программ и утилит, в частности компилятор Java[9].

1.4 Android Virtual Device (AVD)

Это конфигурация эмулятора, которая позволяет описать реальное устройство путем указания аппаратных и программных характеристик подлежащих эмуляции в Android Emulator.

Самый простой способ создать AVD заключается в использовании AVD Manager, который вы можете запустить из Eclipse. Также можно запустить AVD Manager из командной строки.

AVD состоит из:

Профиля оборудования: Определяет аппаратные особенности виртуального устройства. Например, можно определить, будет ли устройство оснащено камерой, использует ли оно физическую QWERTY-клавиатуру или кнопки набора номера, сколько памяти имеет, и т.д.

Указания на образ системы: определить, какая версия платформы Android будет работать на виртуальном устройстве. Можно выбрать версию стандартной платформы Android или образ системы в комплекте с дополнениями SDK.

Есть возможность указать настройки эмулятора, которые хотите использовать с AVD, размеры экрана, внешний вид, и т.д. или указать эмуляцию SD карты для использования с AVD.

Выделенной области хранения данных на вашем компьютере: пользовательские данные устройства (установленные приложения, настройки и т.д.) и данные виртуальной SD карты хранятся в этой области.

Можно создать столько AVD сколько вам нужно, в зависимости от типов устройств, которые хотите моделировать. Чтобы тщательно протестировать приложение, необходимо создать AVD для каждой конфигурации устройства (например, различные размеры экрана и версии платформы), с которым программа совместима и тестировать приложение на каждом из них.

Имейте в виду эти моменты, когда выбираете целевую систему для AVD:

Важен уровень API целевой системы, потому что приложения не смогут работать на системе чей уровень API меньше, чем необходимо для приложения, что указано в атрибуте `minSdkVersion` файла манифеста[10].

Нужно создать по крайней мере одно AVD которое использует уровень API больше, чем необходимо, поскольку это позволяет проверить совместимость приложения с последующими версиями. Такое тестирование на совместимость гарантирует, что, когда пользователи обновят свои системы, ваше приложение продолжит нормально функционировать.

Если запускать приложение на эмуляторе, создается AVD, которое включает необходимые библиотеки. Как правило, создается такое AVD с использованием дополнения для платформы AVD (например, дополнение Google API, которое содержит библиотеку Google Maps).

1.5 База данных Firebase Realtime

База данных Firebase Realtime позволяет создавать многофункциональные приложения для совместной работы, обеспечивая безопасный доступ к базе данных непосредственно из кода на стороне клиента. Данные сохраняются

локально, и даже в автономном режиме события в реальном времени продолжают срабатывать, предоставляя конечному пользователю отзывчивый опыт. Когда устройство восстанавливает соединение, база данных реального времени синхронизирует локальные изменения данных с удаленными обновлениями, которые произошли, когда клиент находился в автономном режиме, автоматически объединяя любые конфликты.

База данных предоставляет гибкий язык правил на основе выражений, называемый правилами безопасности баз данных в реальном времени Firebase, для определения того, как ваши данные должны быть структурированы и когда данные могут считываться или записываться. При интеграции с аутентификацией Firebase разработчики могут определить, кто имеет доступ к каким данным и как они могут получить к ним доступ[16].

Firebase Realtime является базой данных NoSQL и, как таковая, имеет различные оптимизации и функциональности по сравнению с реляционной базой данных. API-интерфейс Firebase Realtime позволяет выполнять только те операции, которые могут быть выполнены быстро. Это позволяет вам создать отличный опыт в реальном времени, который может обслуживать миллионы пользователей без ущерба для скорости отклика. В связи с этим важно подумать о том, как пользователи должны получать доступ к вашим данным, и затем соответствующим образом структурировать их.

Особенности базы данных Firebase Realtime

1. Реальное время: благодаря синхронизации данных, используемой в реальном времени, каждое обновление принимается устройствами / клиентами в кратчайшие сроки.

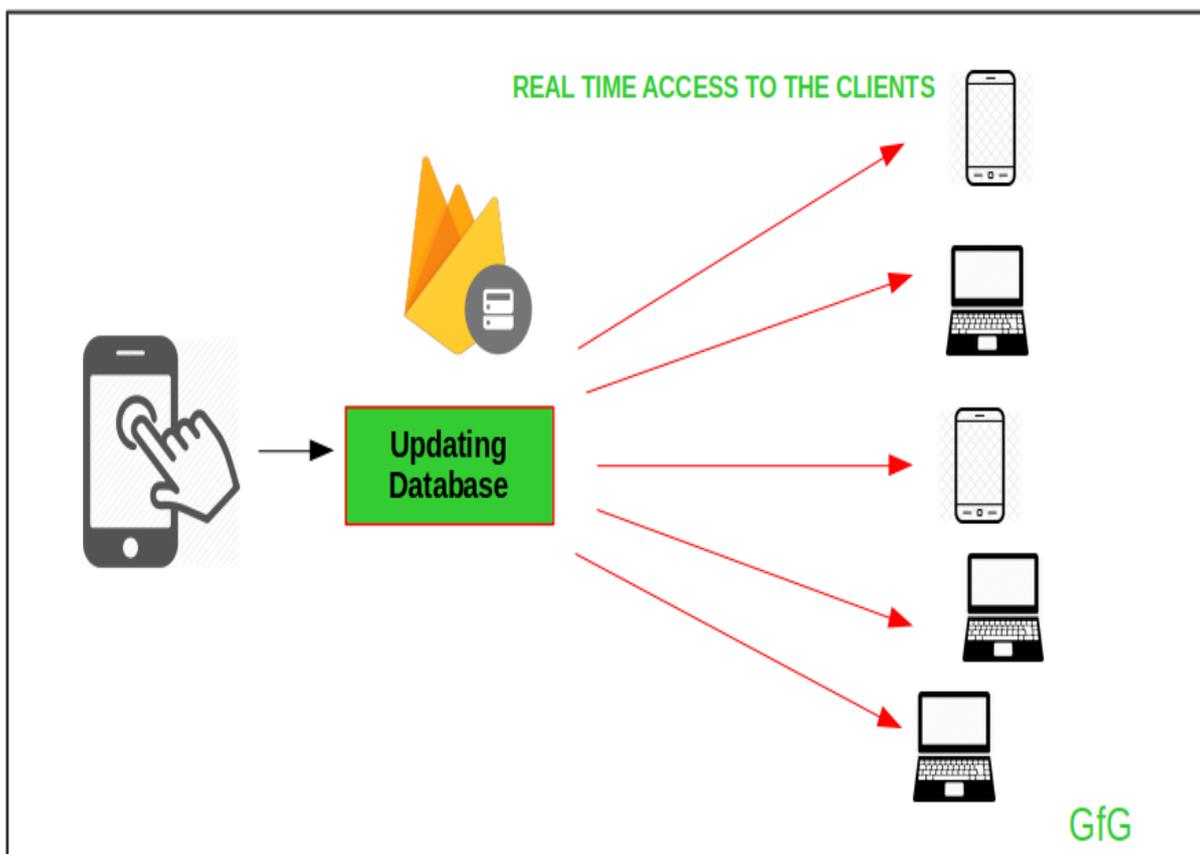


Рисунок 2 – Обновление базы данных

2. Сервер приложений не требуется. Поскольку доступ к базе данных возможен непосредственно с мобильного устройства или браузера, сервер приложений не требуется.
3. Поддержка различными языками и платформами: Unity, Android, iOS, C++, Web.

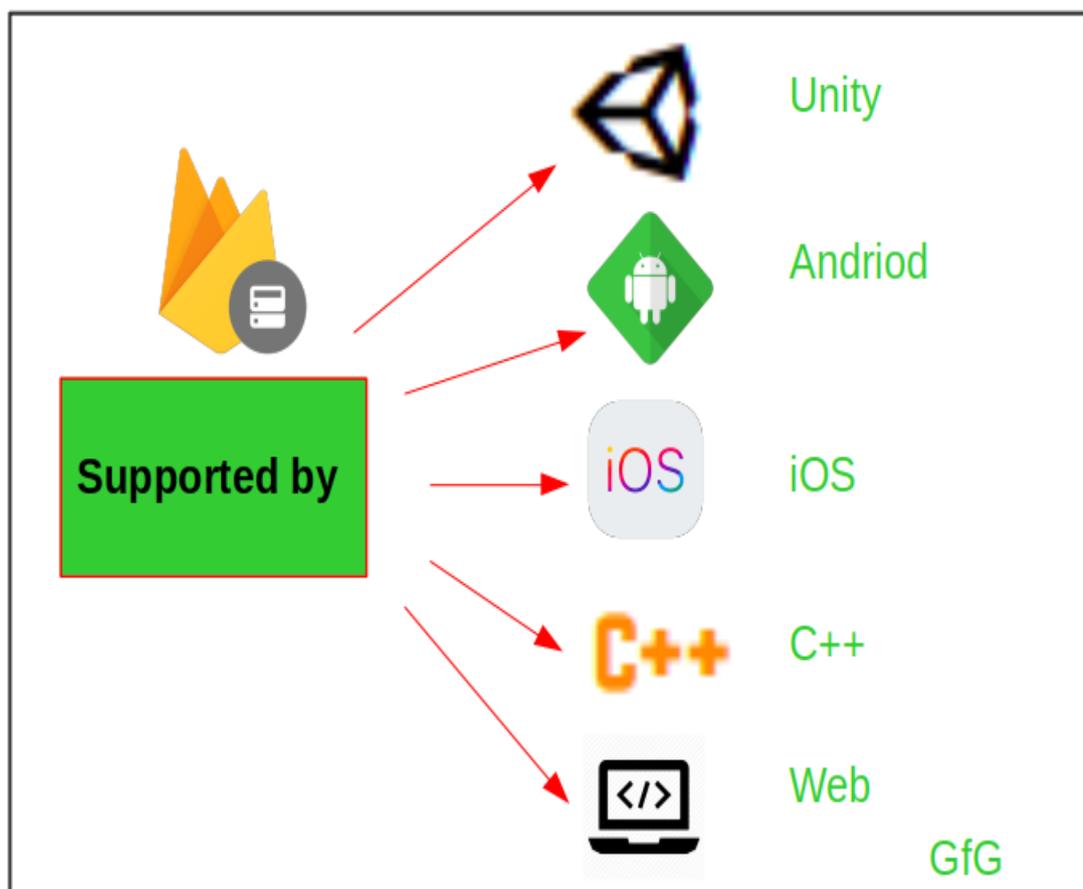


Рисунок 3 – Поддержка языков программирования

4. Разделение данных: клиент может разделить данные между несколькими экземплярами базы данных для одного и того же проекта.
5. Код на стороне клиента: динамические приложения с защищенными данными могут быть доступны непосредственно из кода на стороне клиента.

6. Кроссплатформенность: его можно использовать для создания серверной части для различных платформ, таких как Android, iOS, Web, iOS, а также JavaScript SDK.

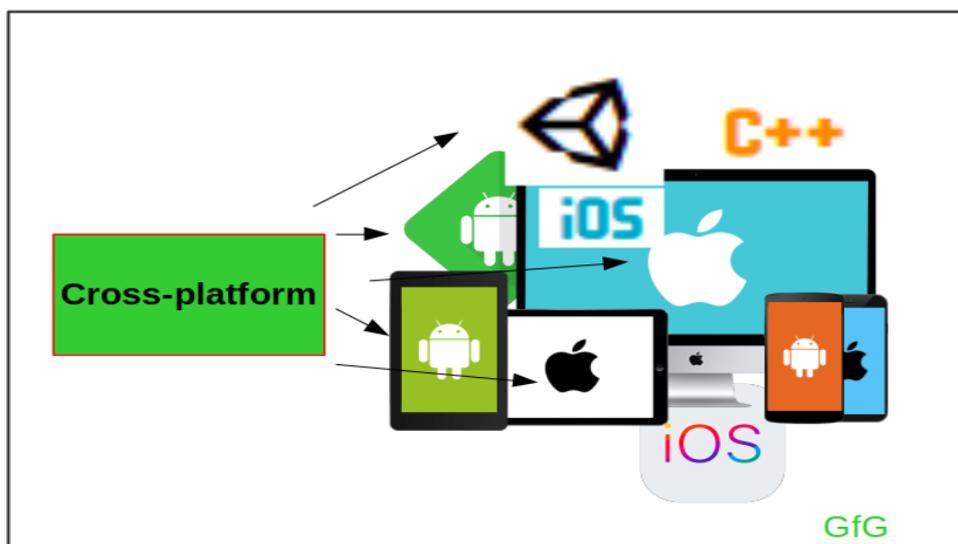


Рисунок 4 – Кроссплатформенность Firebase

1.6 Сравнение отечественных и зарубежных технологий и решений

ТамТам

ТамТам является кроссплатформенным мессенджером, позволяющим отправлять текстовые сообщения, как в личных, так и в групповых чатах.

Возможность отправлять голосовые сообщения, обмениваться фото, видео, музыкой и GIF-анимацией.

Был разработан в 2017 году компанией Mail.ru Group в качестве замены «ОК сообщения», которые принадлежат социальной сети Одноклассники.

В отличии от «ОК сообщения», ТамТам может работать без регистраций аккаунта в социальной сети.

Для шифрования данных при передаче личных сообщений ТамТам использует общепринятые проверенные решения. Это протокол TLS, который позволяет безопасно передавать все сообщения, фото и видео. Вся личная переписка в ТамТам надежно защищена и хранится на распределенной сети серверов. В открытом виде никакие данные не пересылаются.

WhatsApp

Является одним из самых популярных мессенджеров на планете, созданным компанией Facebook. Поддерживает текстовые и голосовые сообщения, с поддержкой голосовой и видеосвязи. Так же позволяет пересылать видео, аудио файлы, документы различных форматов, фото и даже отправку программных установок.

Поддерживается на таких операционных системах. как Android, iOS, Windows Phone.

Конфиденциальность и безопасность всегда были принципиально важны для компании WhatsApp, поэтому используют сквозное шифрование. Сквозное шифрование означает, что сообщения, фотографии, видео, голосовые сообщения, документы, обновления статуса и звонки защищены от попадания в чужие руки.

Сквозное шифрование в WhatsApp гарантирует, что сообщения доступны для чтения только определенному пользователю и его собеседнику и больше никому, даже WhatsApp. Пользовательские сообщения защищены уникальными замками. Только у получателя и у отправителя есть специальные ключи, которые необходимы, чтобы разблокировать и прочитать сообщения. Для дополнительной защиты каждое сообщение, которое отправляет пользователь, имеет уникальный замок и ключ. Всё это происходит автоматически: не нужно включать специальные настройки или создавать секретные чаты, чтобы обеспечить защиту своих сообщений.

В заключении можно сделать вывод, что каждый выбирает мессенджер для своей цели. Если смотреть с точки простоты использования, то мой голос идет в сторону Российского мессенджера ТамТам. А в целях безопасности, мое предпочтение падает на WhatsApp от Facebook[13].

2. Формирование тербований к системе

2.1 Проектирование базы данных

Для создания проекта сетевого чата необходима своя база данных. В данном проекте использовался облачный сервис Firebase Realtime. Включающий в себя, аутентификацию, базу данных в реальном времени, хранение файлов и сообщений.

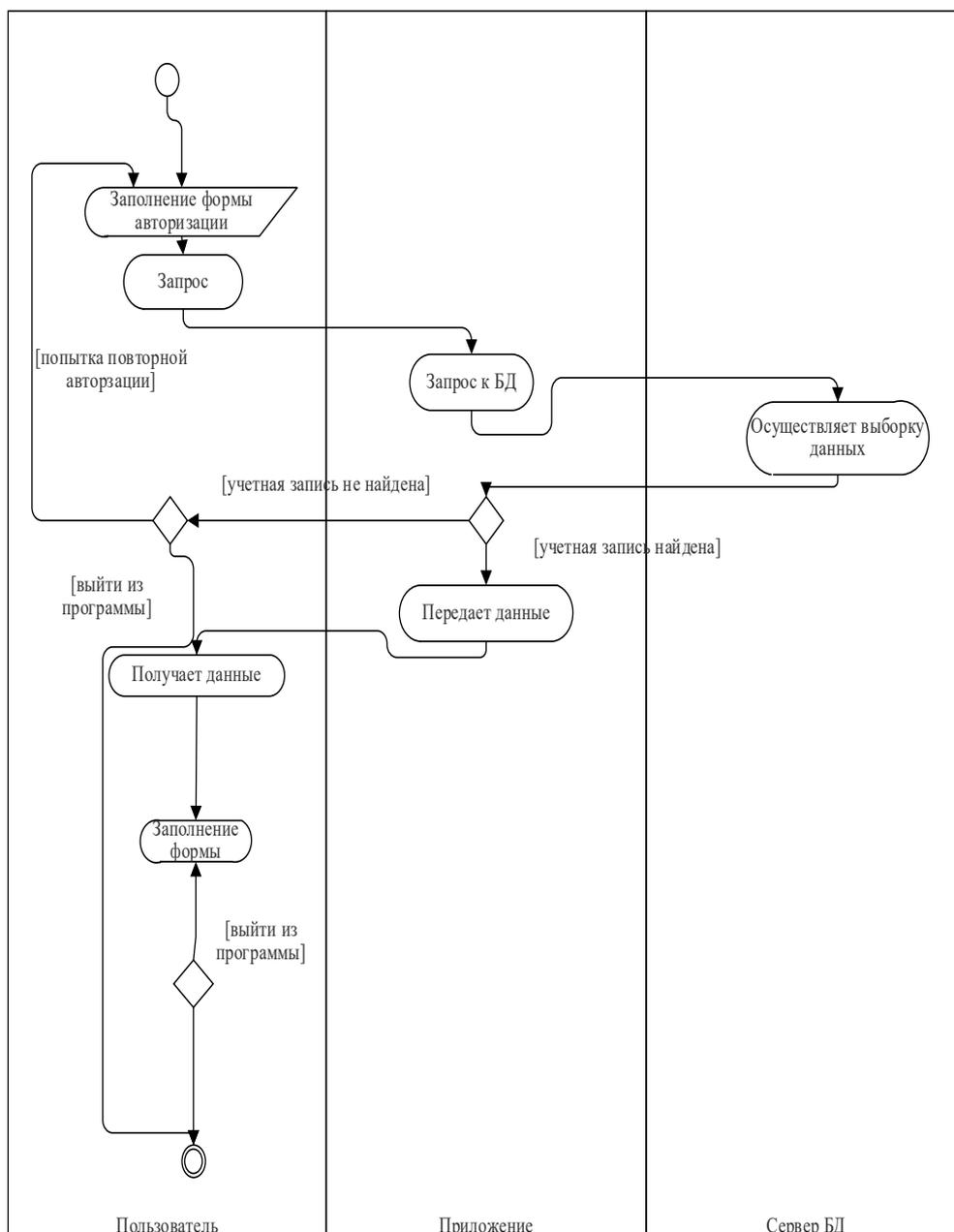


Рисунок 5 – Диаграмма деятельности для пользователя

Есть много способов регистрации пользователей. Для создания этого приложения, использовалась регистрация с помощью электронной почты и пароля, а также регистрация при помощи мобильного номера телефона.

При регистрации с помощью электронной почты и пароля, пользователь вводит данные почты и пароля, после чего, данные отправляются на сервер и записываются в базу данных. Пароль, придуманный пользователем, шифруются с помощью ключа.

Для регистрации с помощью мобильного телефона, открывается специальное окно, где предлагается ввести номер телефона и придумать пароль. После ввода данных, на указанный номер телефона приходит СМС, сформированное правилами безопасности базы данных, которое содержит шестизначный код. Пользователь должен ввести код в форму для проверки. После чего, телефон, используемый, как логин для входа в приложение, записывается в базу данных, а пароль шифруется с помощью ключа.

Если пользователь забыл пароль, он может перейти в окно «Забыли пароль» и восстановить его с помощью электронной почты, на которую придёт письмо, содержащее ссылку на восстановление пароля от своей учетной записи.

Если была забыта почта, то можно обратиться к администратору приложения за помощью.

Администрированием базы данных занимается только администратор приложения, этим обеспечивается безопасность хранения данных.

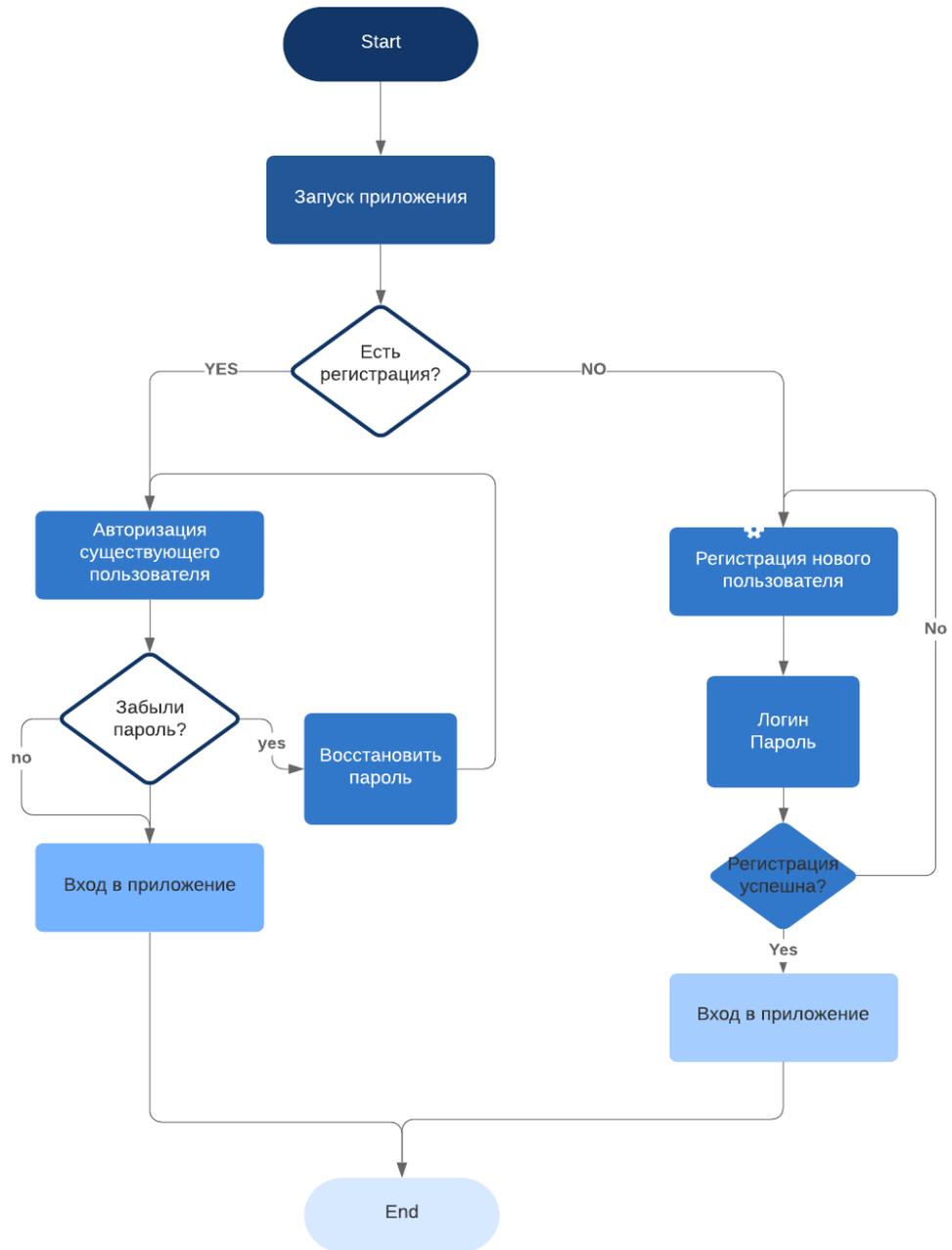


Рисунок 6 – Блок схема регистрации пользователя

2.2 Интерфейс пользователя

Для удобства пользователя, нужно разработать такой интерфейс, в котором разберется даже неопытный пользователь. Все должно быть легко и понятно каждому.

Каждый новый пользователь при первом запуске приложения должен пройти регистрацию. В последующие разы, регистрацию не нужно повторять, а только проходить аутентификацию. После успешной регистрации пользователю присваивается уникальный идентификатор, который хранится в базе данных и является ключом в таблице пользователей.

После входа в приложение, пользователя должно открыться окно профиля, в котором он должен ввести некоторые данные о себе. А именно имя, которое будет отображаться всем пользователям, статус – любая информация о себе или свои мысли и фотография профиля. Все эти данные имя, статус и фотография хранятся в базе данных.

Для пользователя доступно небольшое меню. Меню содержит функцию поиска друзей, создания группы, настройки профиля и выход из учетной записи.

2.3 Обмен сообщениями

Одной из основных функций чата является обмен сообщениями. В организации для которой создается данное приложение, нужен не только обмен сообщениями, но и отправка фото и документов.

Отправленные файлы и сообщения хранятся в базе данных, файлы и фотографии хранятся в отдельной папке storage.

Обмен сообщениями происходит как в диалоге с одним человеком, так и с группой людей одновременно.

При отправке сообщения фиксируется имя пользователя, отправившего сообщение, время и дата отправки. Все эти данные хранятся в базе данных, в таблице message, там же хранится уникальный идентификатор отправителя, получателя, время и дата сообщения и тип сообщения.

2.4 Состояние пользователя

Для каждого пользователя существует два состояния. «Online» и «Offline». Когда пользователь заходит в приложение, его состояние меняется на «Online», а при выходе «Offline». Оба состояния хранятся в базе данных в таблице Users. Помимо надписи состояния, активность пользователя определяется зеленым кружком возле имени пользователя, это означает, что пользователь сейчас находится в сети и готов к общению.

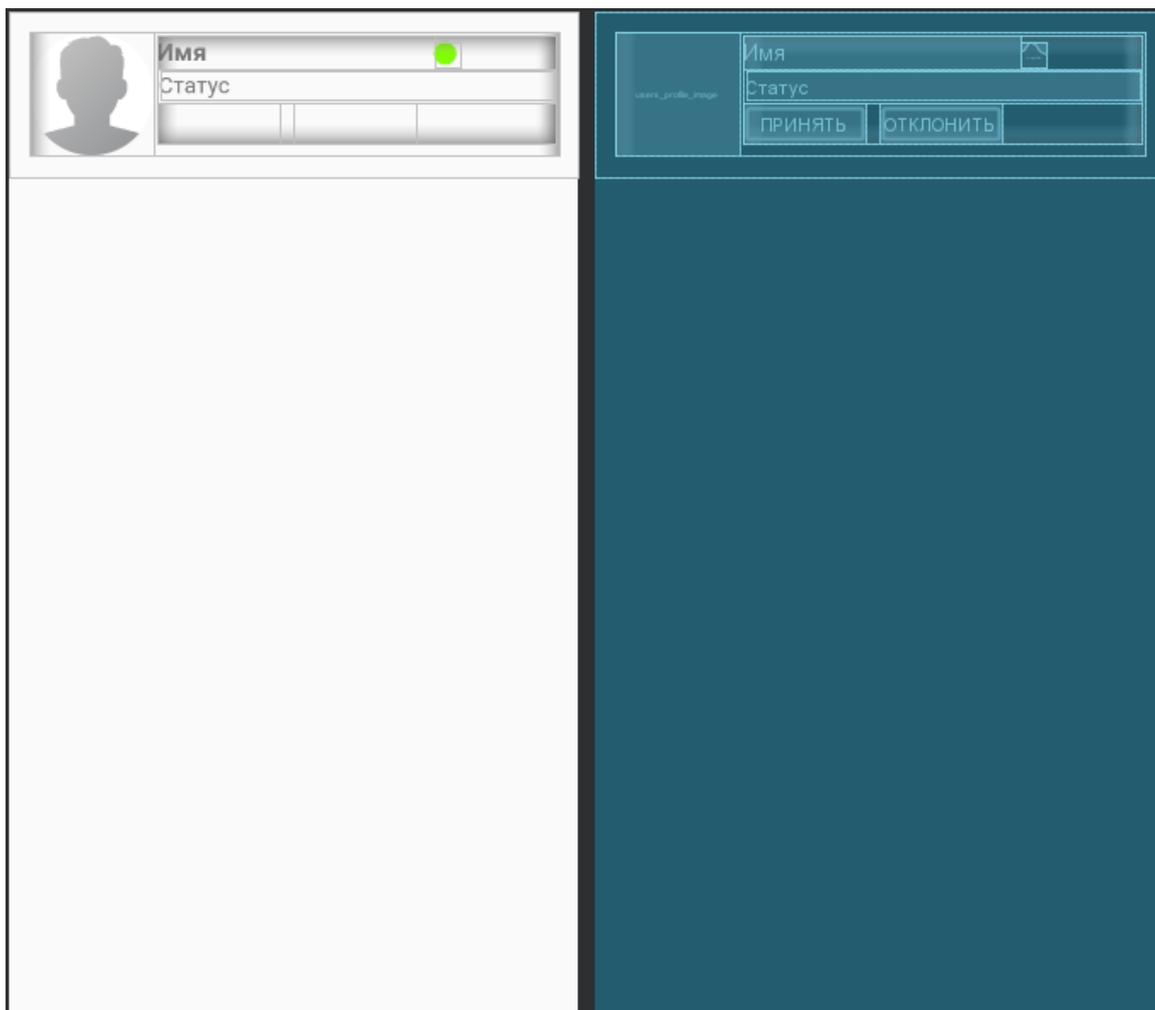


Рисунок 7 – Состояние пользователя

3. Разработка мобильного приложения

3.1 Создание формы авторизации

Для входа в приложение, необходимо пройти регистрацию пользователя. При запуске приложение открывается форма авторизации пользователя (см. рисунок 7), если у пользователя нет учетной записи, ему предлагается завести новую учетную запись. Переходя в другую форму, пользователь должен указать электронную почту и пароль в специальные поля, после чего подтвердить введенные данные. После успешной регистрации, пользователь попадает в форму настройки профиля, где должен указать имя, статус и фотографию.

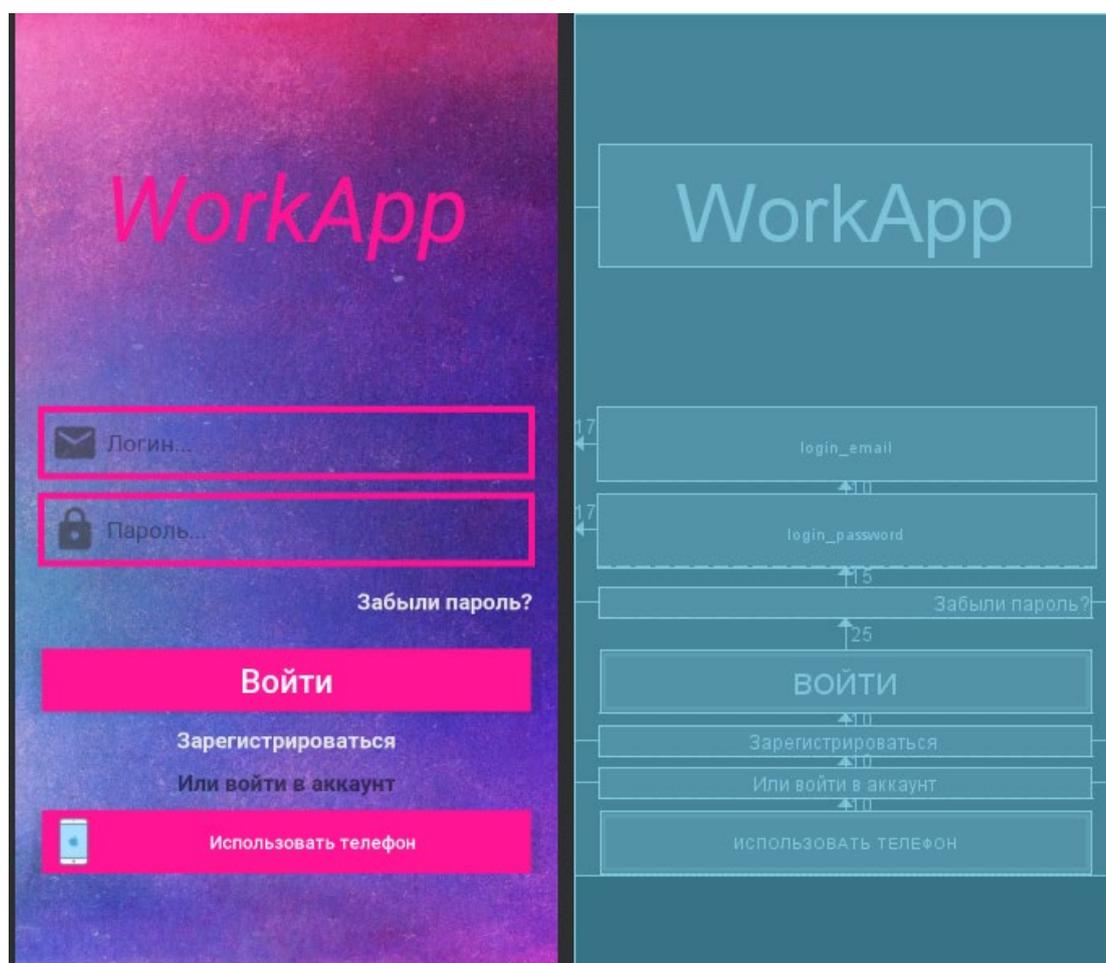


Рисунок 8 – Форма авторизации

Листинг программы, который отвечает за работу кнопки «Войти» приведен в приложении А.1.

3.2 Создание формы регистрации

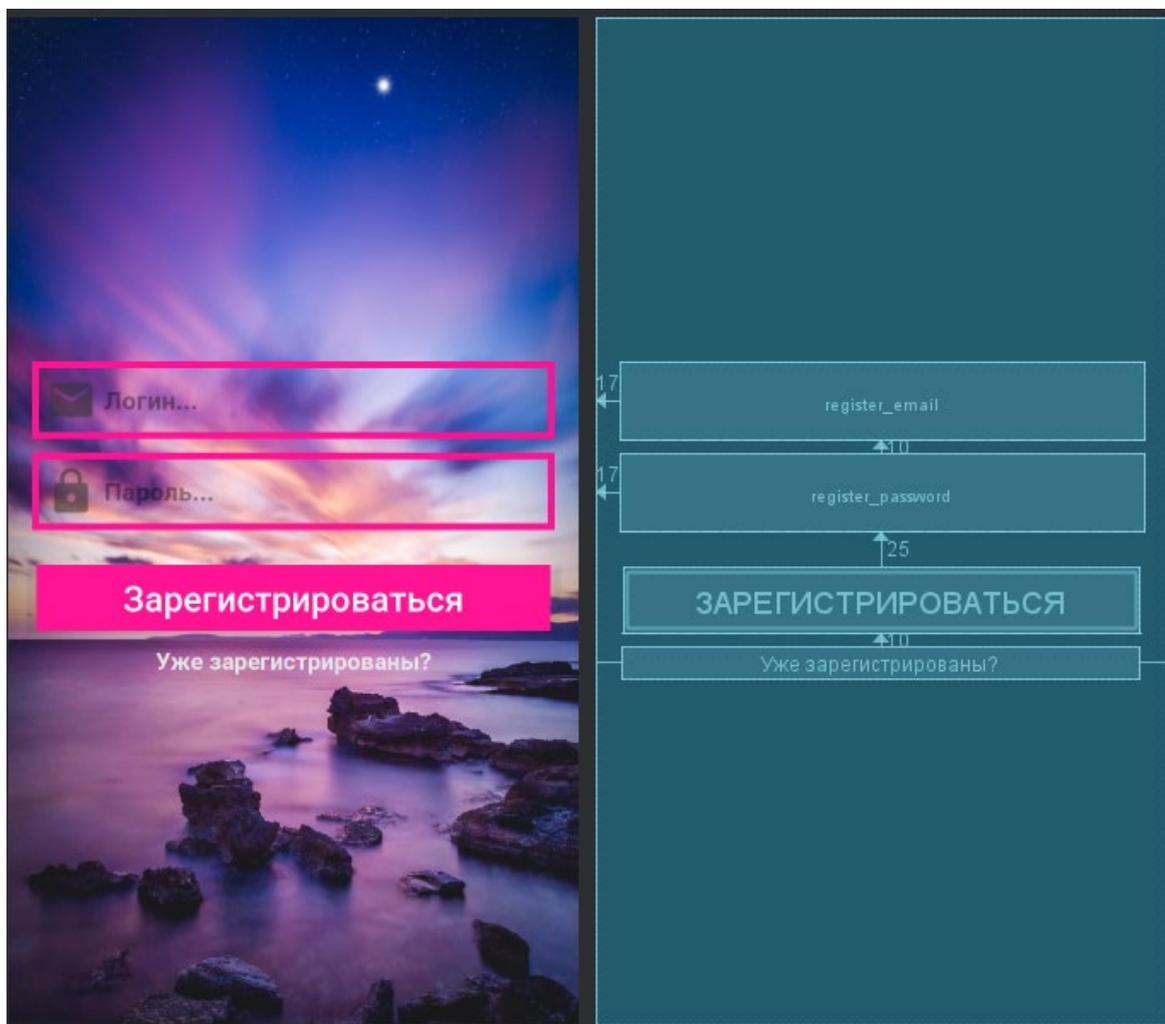


Рисунок 9 – Форма регистрации

Как показано на рисунке 8, если у пользователя нет ранее созданной учетной записи в приложении, ему предлагается создать новую. Пользователь должен заполнить поля почтой и придумать пароль, после чего подтвердить введенные им данной кнопкой «Зарегистрироваться».

Если пользователь уже имеет учетную запись, он может нажать на «Уже зарегистрированы?» и приложение переведет его в форму авторизации, где нужно просто ввести учетные данные в поля и войти в приложение.

Листинг программы, который отвечает за работу кнопки «Зарегистрироваться» приведен в приложении А.2.

3.3 Создание формы регистрации с помощью мобильного телефона

Помимо регистрации с помощью электронной почты пользователю доступна регистрация с помощью мобильного номера. Пользователь должен ввести номер телефона в поле, но обязательно с кодом страны, в нашем случае «+7». После нажатия на кнопку «Отправить код проверки», пользователю придет шестизначный код, который нужно ввести в специальное поле, для подтверждения номера.

После регистрации по номеру телефона, пользователь попадет в меню настройки профиля, а данные запишутся в базу данных

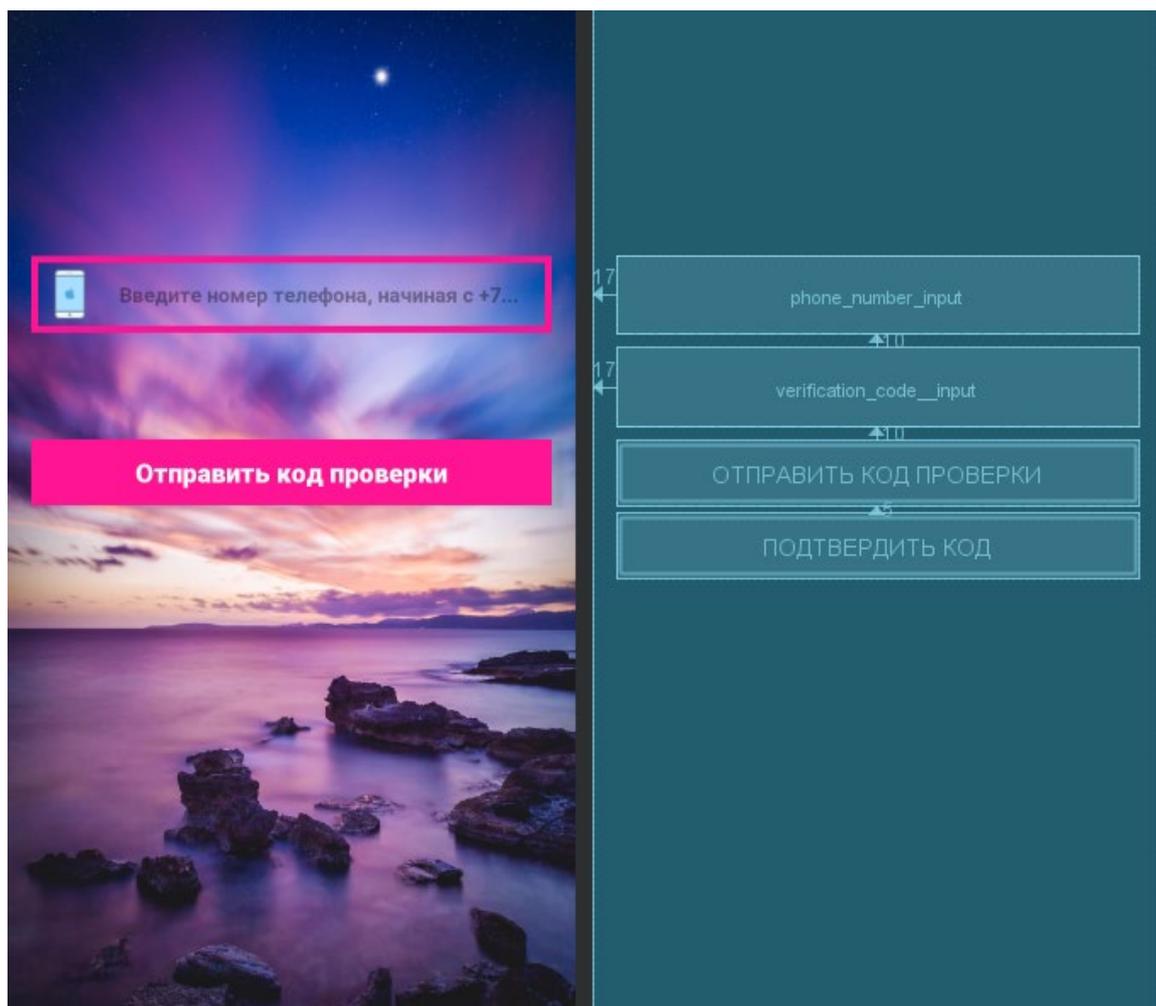


Рисунок 10 – Регистрация по номеру телефона

Листинг программы, который отвечает за работу кнопки «Отправить код проверки» приведен в приложении А.3.

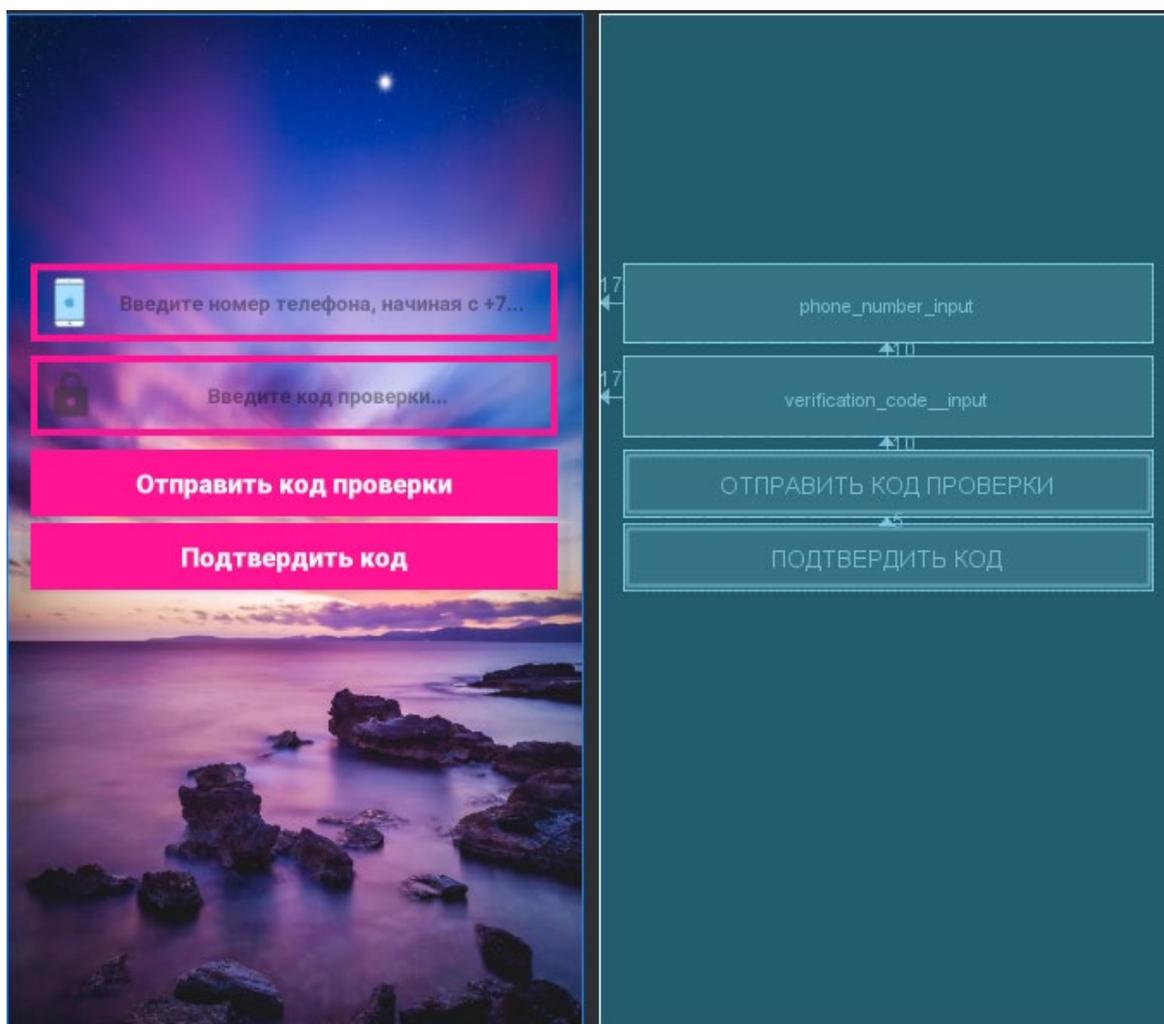


Рисунок 11 – Подтверждение кода

Листинг программы, который отвечает за работу кнопки «Подтвердить код» приведен в приложении А.4.

3.4 Создание окна восстановления пароля

Если у пользователя есть учетная запись, но он забыл свой пароль, то можно нажать на надпись: «Забыли пароль?» и приложение откроет форму восстановления пароля.

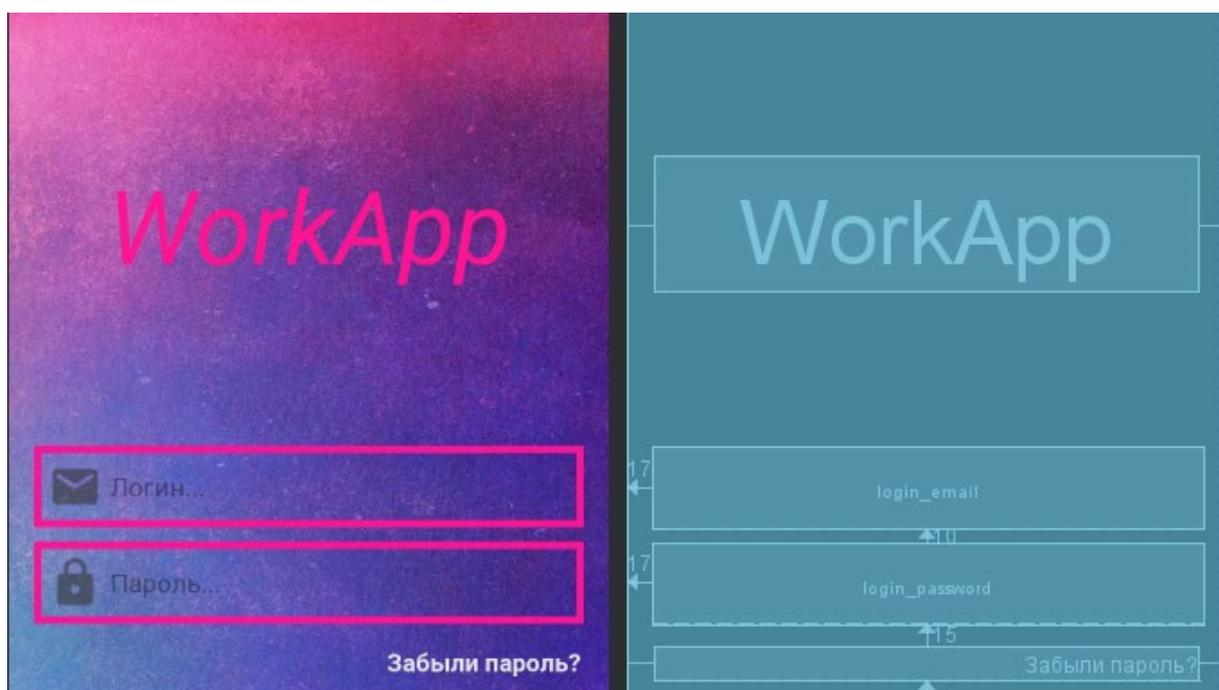


Рисунок 12 – Забыли пароль

После открытия формы восстановления пароля, пользователю нужно ввести свою почту в поле и подтвердить, нажав кнопку сбросить пароль. После чего на указанную им почту, будет выслано сообщение с ссылкой на восстановление пароля, переходя по которой, пользователю нужно придумать новый пароль и ввести его.

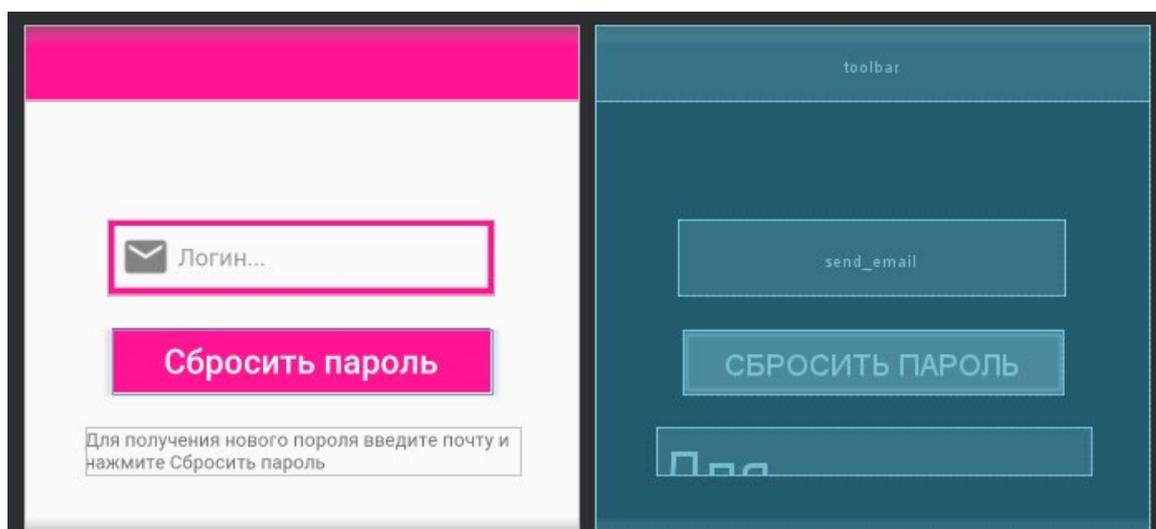


Рисунок 13 – Форма восстановления пароля

Высланное письмо на почту имеет вид:

Сброс пароля для приложения "WorkApp"

 **WorkApp** noreply@workapp-46155.firebaseio.com
Вам:  slav.senatorov@yandex.ru ^
Папка: Входящие

12 мая в 13:55

Здравствуйте!

Чтобы сбросить пароль аккаунта slav.senatorov@yandex.ru для входа в приложение WorkApp, перейдите по ссылке ниже.

https://workapp-46155.firebaseio.com/_/auth/action?mode=resetPassword&oobCode=_2s8NY7XD1Dnw7HJ4mcxchBpELXteGi5gp_mJvrn1H4AAAFyCBqxKw&apiKey=AlzaSyB5DYPx_JRLy7xwzBJCQcK8DIREj5rW2o4&lang=ru

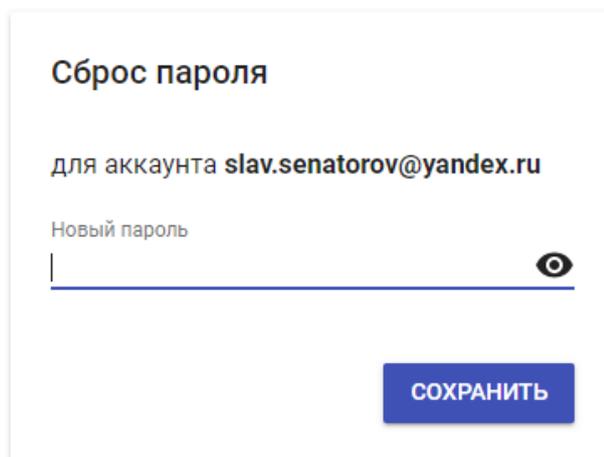
Если Вы не отправляли запрос на сброс пароля, просто проигнорируйте это письмо.

С уважением,

команда WorkApp

Рисунок 14 – Письмо восстановления пароля

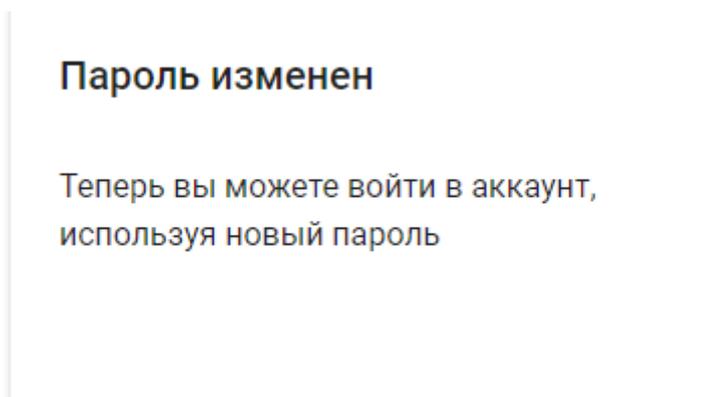
При переходе по ссылке откроется следующее окно:



The screenshot shows a web form titled "Сброс пароля" (Reset Password). Below the title, it says "для аккаунта **slav.senatorov@yandex.ru**". There is a text input field labeled "Новый пароль" (New password) with a blue underline and a toggle icon (an eye) to its right. At the bottom right of the form is a blue button with the text "СОХРАНИТЬ" (SAVE).

Рисунок 15 – Сброс пароля

После введения нового пароля, открывается окно:



The screenshot shows a confirmation message. At the top, it says "Пароль изменен" (Password changed). Below that, it says "Теперь вы можете войти в аккаунт, используя новый пароль" (Now you can log in to your account using the new password).

Рисунок 16 – Пароль изменен

Листинг программы, который отвечает за работу кнопки «Сбросить пароль» приведен в приложении А.5.

3.5 Создание настроек профиля

После создание учетной записи открывается окно настройки профиля. Здесь пользователю нужно заполнить поля: имя и статус, и по желанию загрузить фотографию профиля. Нажав на кнопку «обновить», данные введенные пользователем сохраняются в базу данных и ему присвоится уникальный идентификатор, который будет занесен в базу.

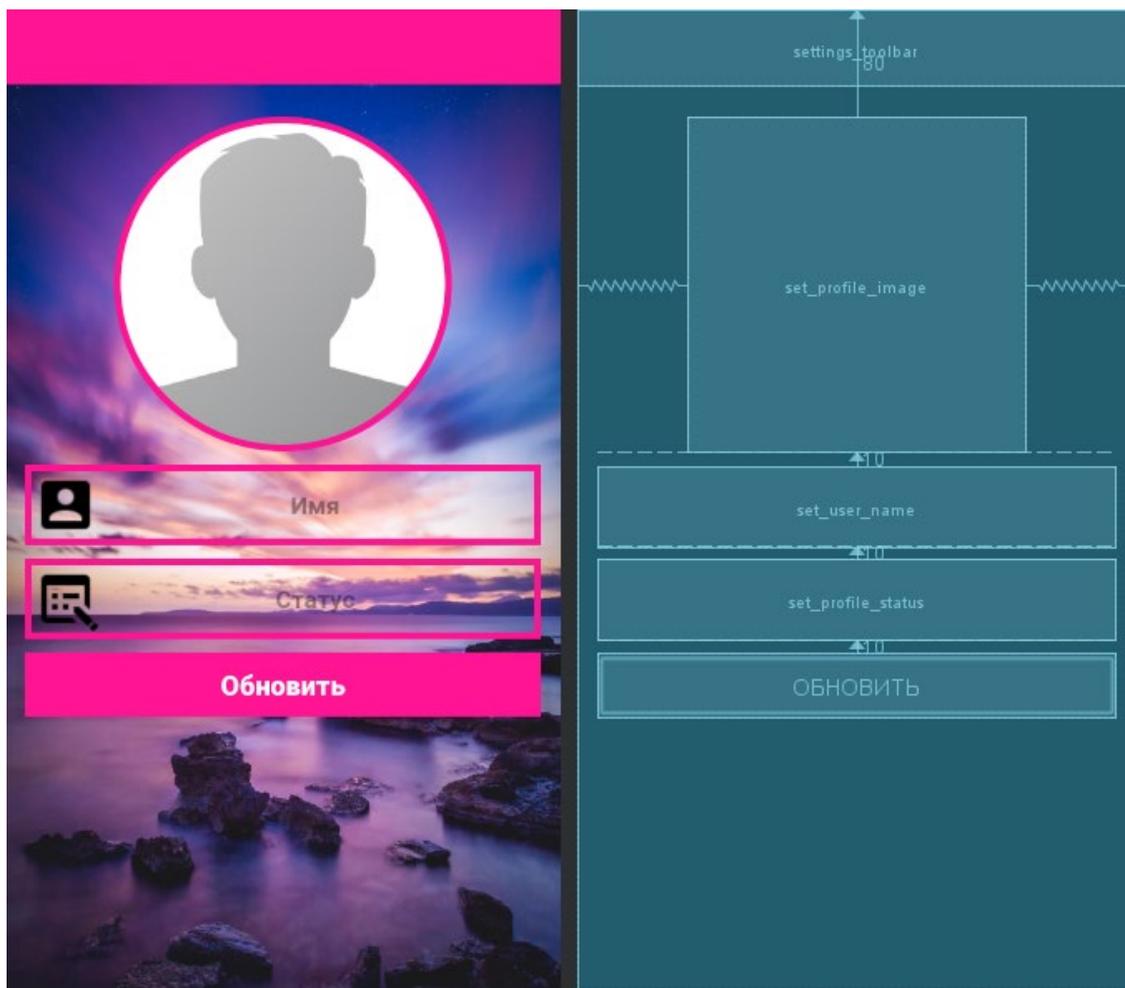


Рисунок 17 – Настройки профиля пользователя

Листинг программы, который отвечает за работу кнопки «Обновить» приведен в приложении А.6.

3.6 Создание формы сообщений

Главной функцией чата является обмен сообщениями. Обмен сообщениями может проходить как в чате с одним пользователем, так и с группой пользователей сразу, то есть групповой чат.

Кроме обычных сообщений, предусмотрено обмен файлами в формате pdf и docx и фотографиями в формате jpeg. Файлы и фото хранятся в отдельных папках в базе данных.

В базе данных хранится следующая информация о сообщениях:

- дата сообщения;
- отправитель сообщения;
- сообщение;
- ID сообщения;
- время отправки сообщения;
- получатель сообщения;
- формат сообщения.

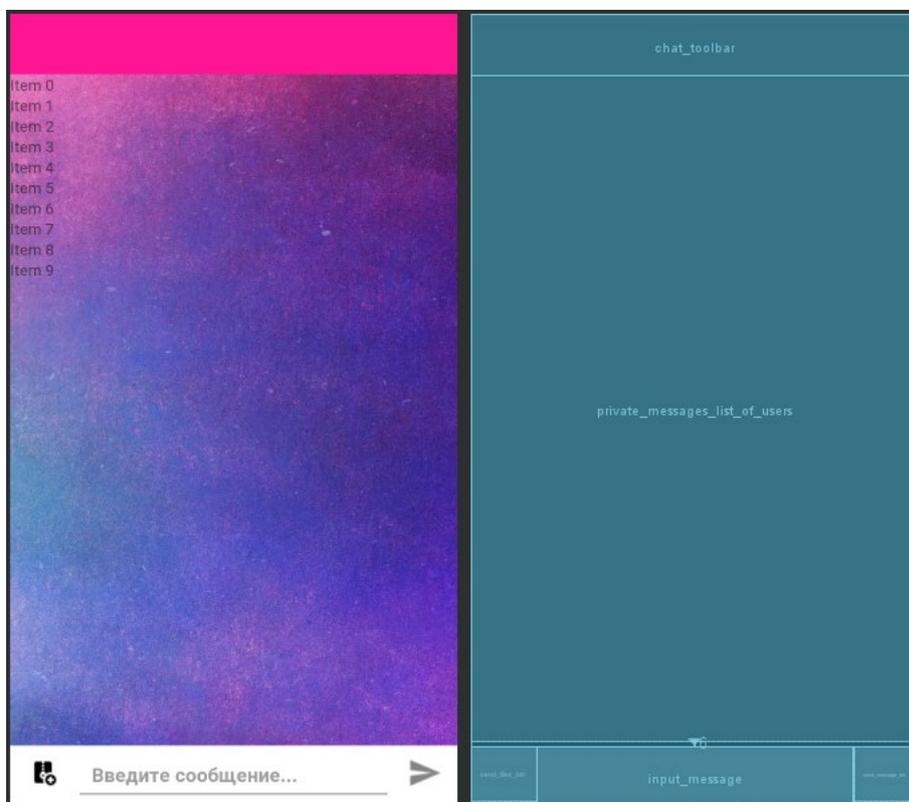


Рисунок 18 – Форма сообщений

На рисунке 17 показано окно чата. Здесь пользователю отображаются входящие и исходящие сообщения. В левом нижнем углу значок выбора файлов, которые можно отправить, фото или документ на выбор.

Нажатием на отправленное сообщение открывается меню с действиями:

- удалить сообщение у меня;
- скачать и открыть этот документ;
- открыть изображение;
- удалить сообщение у всех;
- назад.

При «удалении сообщения у меня», удаляются сообщения только у пользователя, выбравший это действие, а у остальных оно будет видно.

При выборе «Скачать и открыть этот документ» отправленный документ от собеседника скачивается на телефон и автоматически открывается через программу, выбранной операционной системой Android.

«Открыть изображение» позволяет открыть фото, не скачивая его на телефон.

«Удалить сообщение у всех» выбрав этот пункт, пользователь удаляет сообщение не только у себя, но и у собеседника.

Листинг программы, который отвечает за работу кнопки «Отправки сообщения» приведен в приложении А.7.

3.7 Создание формы контактов

Форма контактов содержит информацию о контактах, которые были добавлены в список друзей. Список всех контактов можно найти, выбрав в меню пункт «Найти друзей», найдя нужного собеседника нужно зайти к нему в профиль и отправить запрос на создания нового чата с ним. Пользователю придет оповещение с информацией, что другой пользователь хочет начать с ним диалог. Остается только выбрать, принять или отклонить запрос. Если

запрос был принят, то оба пользователя будут отображаться в списке контактов.

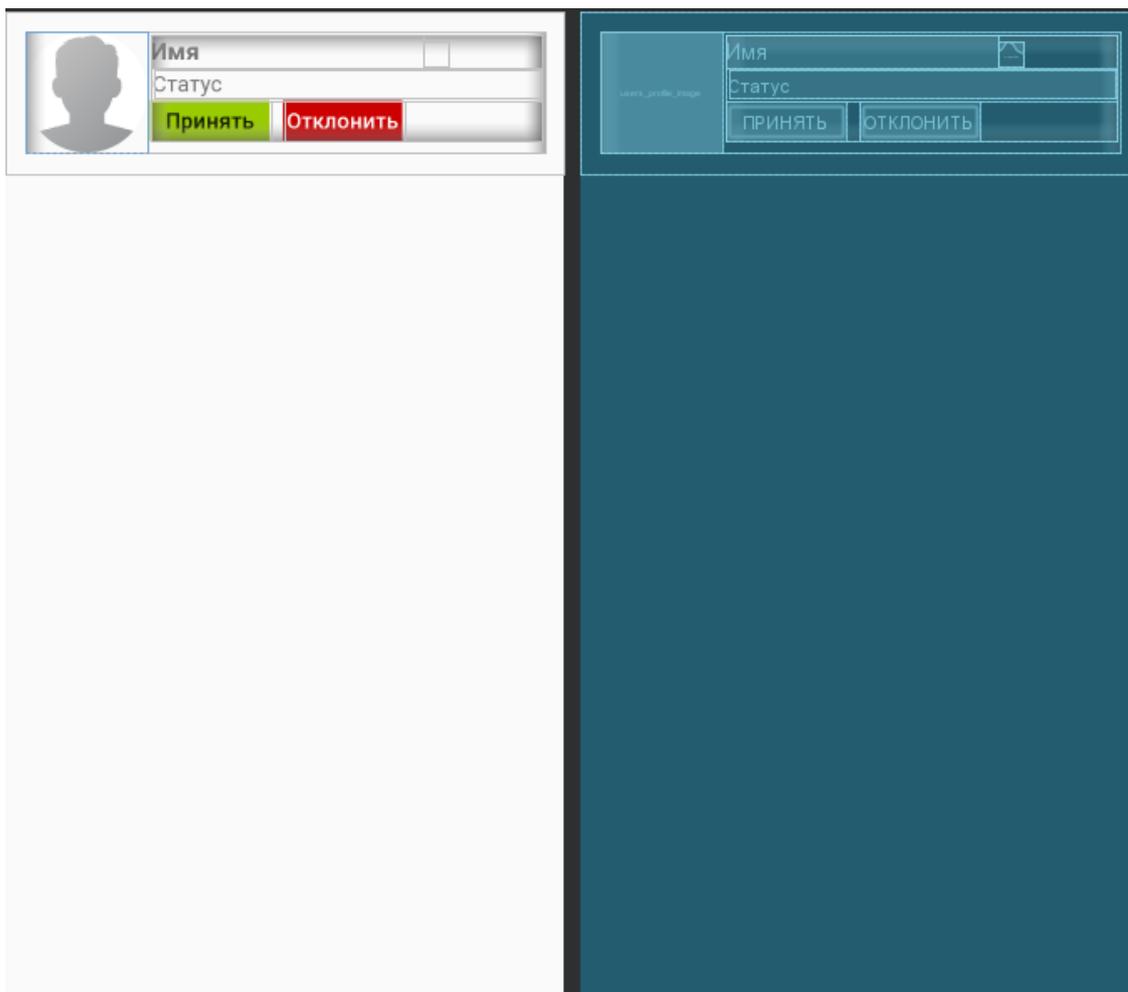


Рисунок 19 – Запрос на создание нового чата

Листинг программы, который отвечает за работу кнопки «Принять» приведен в приложении А.8.

3.8 Создание формы меню приложения

В меню можно зайти, нажав в приложении на значок в правом верхнем углу. Откроются 4 пункта меню:

- найти друзей;
- добавить группу;
- настройки;
- ВЫХОД.

Выбрав пункт «Найти друзей» приложение откроет форму всех пользователей этого приложения, где уже сам пользователь решает кого добавить в друзья, а кого нет.

Пункт «Добавить группу» позволяет пользователю создать новую группу, в которой можно переписываться сразу с несколькими пользователями.

«Настройки» приложение переведет вас в форму профиля, где можно будет изменить свой статус или фотографию профиля.

Выход из учетной записи можно выполнить, нажав на пункт «Выход»

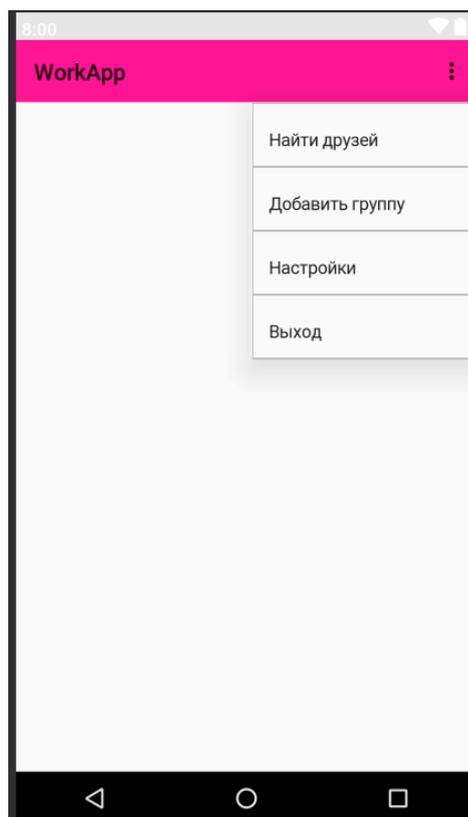


Рисунок 20 – Меню приложения

Листинг программы, который отвечает за работу кнопки «Меню» приведен в приложении А.9.

3.9 Хранение в базе данных информации

В базе данных храниться вся информация о пользователях, сообщениях, группах и отправленных файлах. Firebase database позволяет просматривать аналитику в разных областях. Можно смотреть аналитику активности пользователей за разные периоды. Просматривать состояние занятой и свободной памяти в хранилище.

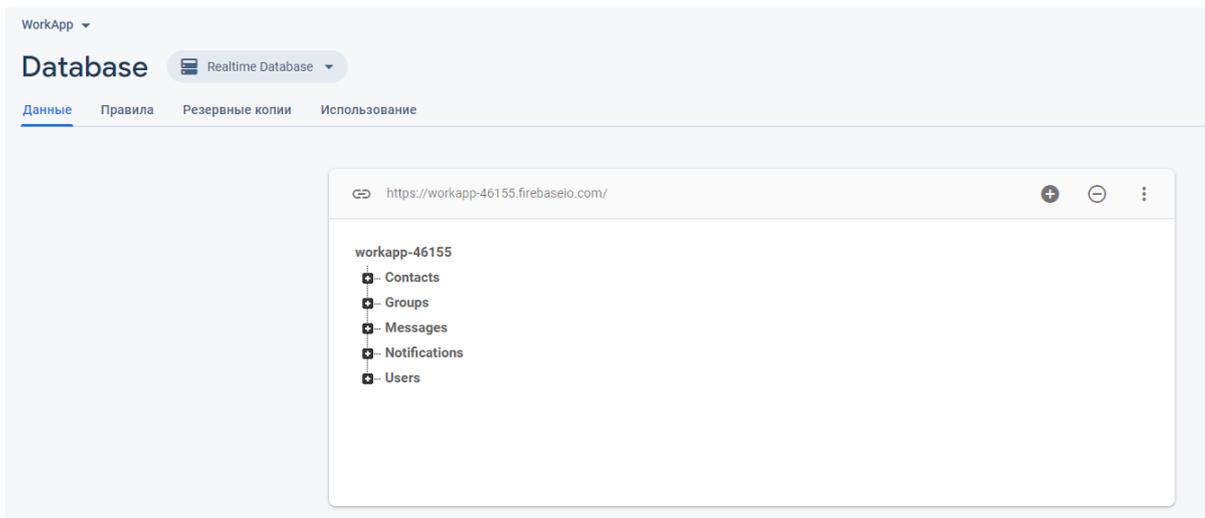


Рисунок 21 – Структура базы данных

База данных состоит из 6 таблиц. Контакты, группы, запросы, сообщения, оповещения, пользователи.

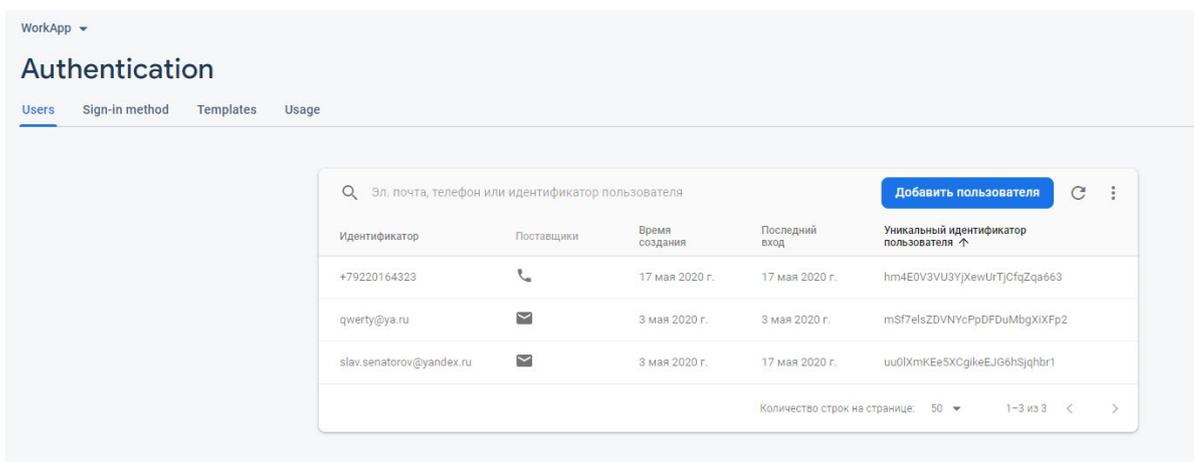


Рисунок 22 – Пользователи, зарегистрированные в базе данных

В базе хранится информация о зарегистрированных пользователях через электронную почту и через номер телефона.

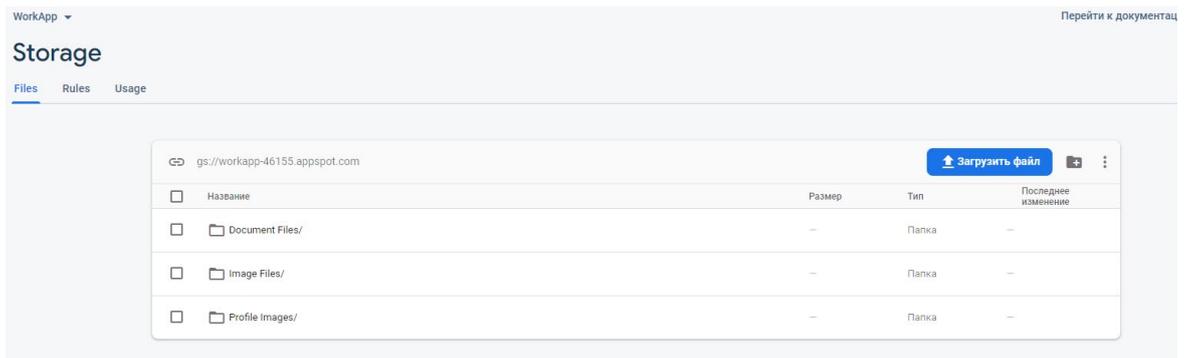


Рисунок 23 – Хранилище файлов

Файлы, отправленные пользователями, хранятся в разных папках. Для отправленных фотографий своя папка, для документов и фотографий профиля так же свои папки.

4. Тестирование и отладка

Функциональное тестирование – это тестирование программного обеспечения в целях проверки реализуемости функциональных требований, то есть способности программного обеспечения в определенных условиях решать задачи, нужные пользователям [10]. Функциональные требования показывают, что именно делает программное обеспечение, как оно себя ведет и какие задачи оно выполняет. Используя метод функционального тестирования, произведем проверку работы мобильного приложения [14].

Тест № 1. Корректное отображение профиля пользователя.

Цель: проверить корректность отображения профиля пользователя после регистрации.

Ход проведения:

1. Создать учетную запись пользователя;
2. Проверить, что форма профиля открывается корректно;
3. Нажать на фотографию в профиле;
4. Выбрать новую фотографию из галереи;
5. Загрузить выбранную фотографию в профиль;
6. Закрыть приложение;
7. Открыть приложение заново и убедиться, что фотография загружена;
8. Проверить базу данных и убедиться, что фотография сохранена в БД.

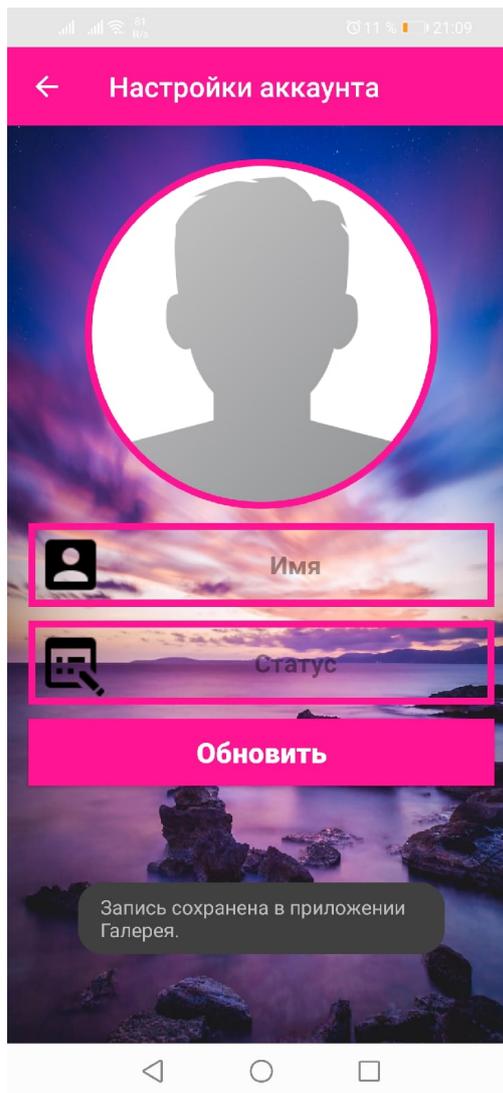


Рисунок 24 – профиль пользователя

Результат: тест пройден.

Тест № 2. Корректность работы отправки запросов на добавление в контакты.

Цель: проверить корректность работы отправки запросов на добавление в контакты. Прием и отклонение запросов.

Ход проведения:

1. Открыть мобильное приложение;
2. Перейти на экран поиска друзей;
3. Найти нужного пользователя;
4. Перейти в профиль пользователя;
5. Нажать на кнопку добавление в список контактов.

Со стороны получателя:

1. Получить уведомление о запросе на добавления в список контактов от другого пользователя;
2. Выбрать пункт добавления в список контактов;
3. Проверить список контактов;

Результат: тест пройден.

Тест № 3. Корректность работы экрана сообщений.

Цель: проверить корректность отображения и обновления сообщений пользователей.

Ход проведения:

1. Открыть мобильное приложение;
2. Перейти в диалог с пользователем;
3. Отправить сообщение текстового формата;
4. Отправить файл, pdf или docx;
5. Отправить фотографию формата jpeg;
6. Открыть приложение у другого пользователя и убедиться, что выше перечисленные сообщения получены;
7. Перейти в базу данных и убедиться, что сообщения сохранены;
8. Перейти в базе данных в хранилище и убедиться, что файлы сохранены в базе.

Результат: тест пройден.

Тест № 4. Отображение действий с сообщениями.

Цель: проверить корректность работы меню сообщений.

Ход проведения:

1. открыть мобильное приложение;
2. перейти в диалог с пользователем;
3. нажать на уже отправленное сообщение;
4. выбрать пункты «Удаление сообщения у меня»;
5. проверить, удалилось ли сообщение;

1. Выбрать пункт «Удаление сообщения у всех» (если сообщение, отправляли вы);
2. Проверить, удалилось ли сообщение у всех пользователей;
3. Если было отправлено изображение, выбрать пункты «Открыть изображение»
4. Проверить открылось ли изображение.
5. Если был отправлен документ, выбрать пункт «Скачать и открыть документ»
6. Проверить, открылся ли документ после скачивания.

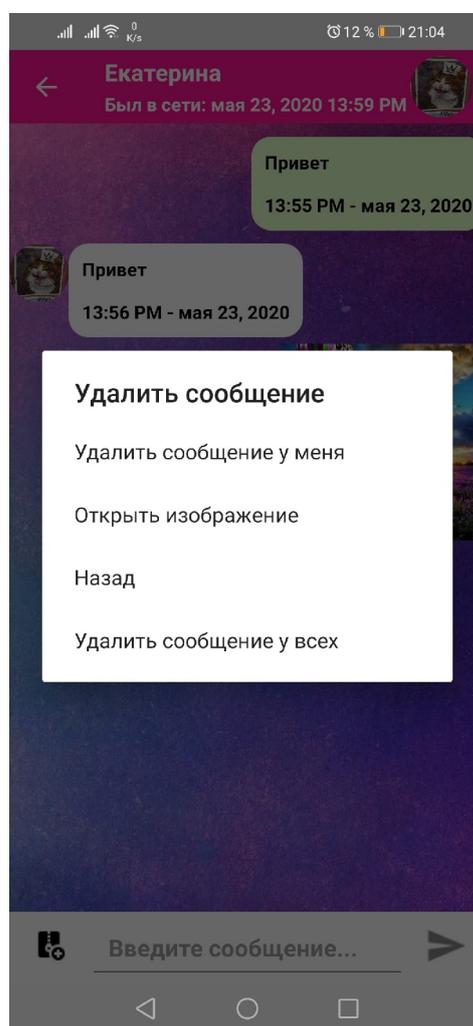


Рисунок 25 – меню сообщений

Результат: тест пройден.

Интеграционное тестирование – полная проверка программного продукта после его сборки с целью выявления ошибок, возникающих в процессе интеграции программных модулей или компонентов [10].

Мобильное приложение было проверено на таких версиях операционных систем и разрешениях экранов, как: Honor 8x (Android 10.0) - 2340x1080 пикселей, Samsung j7(Android 9.0) - 1920x1080 пикселей. Все элементы и экраны мобильного приложения были отображены без ошибок на различных телефонах и их разрешениях.

В ходе проверки на разных версиях операционных систем ошибок также не найдено.

Заключение

Роль мобильных приложений в повседневной жизни растет. Многие люди постоянно пользуются несколькими десятками приложений. Почти каждый день выходят новые, а также выходят и обновления на уже установленные приложения. Скачивание приложений не требует долгого времени и особых навыков, установка также проста и понятна [11].

Целью работы является разработка корпоративного сетевого чата для операционной системы Android. Для достижения данной цели были поставлены и решены следующие задачи:

1. Произведена постановка задачи;
2. Произведен обзор существующих аналогов;
3. Изучены современные платформы и средства разработки для тестировной системы Android;
4. Реализовано приложение;
5. Протестировано приложение;

Все поставленные задачи были решены.

Разработанное приложение имеет перспективы дальнейшего развития. В связи с усложнениями бизнес-процессов и ростом требований к мобильным приложениям, у онлайн чатов, возникает потребность в расширении возможностей и функционала приложения.

В перспективе планируется реализовать следующие возможности:

- при отправке сообщения реализовать оповещения, о принятом сообщении;
- добавить возможность видео и аудио звонков;
- подключить тариф от Google Firebase, для большей защиты информации;
- сделать поддерживание данного приложения не только для Android, но и для iOS.

Библиографический список

1. Голощапов Алексей Google Android / Голощапов Алексей - Программирование для мобильных устройств; БХВ-Петербург - Москва, 2012. - 448 с.
2. Берд, Барри Java для чайников / Барри Берд. - М.: Диалектика / Вильямс, 2013. - 521 с.
3. История создания сетевых чатов. - <https://habr.com/ru/post/91639> (дата обращения: 22.04.2020).
4. СТО ЮУрГУ 04-2008 Стандарт организации. Курсовое и дипломное проектирование. Общие требования к содержанию и оформлению. Введен 01.09.2008 – Челябинск.: Изд. ЮУрГУ, 2008. – 56 с.
5. Эккель, Брюс Философия Java / Брюс Эккель. - М.: Питер, 2016. - 809 с.
6. Монахов, В. Язык программирования Java и среда NetBeans (+ CD-ROM) / В. Монахов. - М.: БХВ-Петербург, 2012. - 720 с.
7. Kaner Cem, Falk Jack, Nguyen Hung Quoc. Testing Computer Software. – USA: Wiley Computer Publishing, 1999. – 479 p.
8. Для чего нужны мобильные приложения? - http://forum.ru/publ/igry_i_soft/dlja_chego_nuzhny_mobilnye_prilozhenija/4-1-0-5 (дата обращения: 03.04.2020).
9. Коматинени С., Маклин Д. Android для профессионалов. / Коматинени С., Маклин Д. – Создание приложений для планшетных компьютеров и смартфонов [пер. с англ.] – М.: Издательский дом «Вильямс», 2012. – 880 с.
10. Варакин М.В. Разработка мобильных приложений под Android. / Варакин, М.В М.: УЦ «Специалист», 2012. – 128 с.
11. Content Providers | Android Developers – <https://developer.android.com> (дата обращения: 15.04.2020)
12. Что такое APK-файлы на Android и зачем они нужны? | AndroidLime. – <https://androidlime.ru/apk-files>. – Дата обращения: 10.05.2019.

13. Выбор SMS-мессенжера для Android устройств. – <https://overclockers.ru/lab/show/84944/vybiraem-sms-messendzher-dlyaandroid-ustrojstv-pulse-sms-handcent-next-sms-a-takzhe-itogirassmotreniya-pyati-prilozhenij#2>. – Дата обращения: 12.05.2020.
14. Ручное тестирование. - <https://www.appline.ru/services/testing/ruchnoetestirovanie>. – Дата обращения: 18.05.2019.
15. Обзор существующих баз данных. – <https://devacademy.ru/article/sqlite-vs-mysql-vs-postgresql>. – Дата обращения: 19.05.2020.