

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Образовательная программа бакалавриата «Программная инженерия»

СОГЛАСОВАНО
Научный руководитель,
доцент департамента программной
инженерии факультета
компьютерных наук, кандидат
технических наук

_____ С.Л.Макаров
«___» _____ 2021 г.

УТВЕРЖДАЮ
Академический руководитель
образовательной программы
«Программная инженерия»
профессор департамента программной
инженерии, кандидат технических наук

_____ В.В. Шилов
«___» _____ 2021 г.

МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ ОБМЕНА ВЕЩАМИ «SWOPPER»

Пояснительная записка

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.06.02-01 81 01-1-ЛУ

Исполнитель
студент группы БПИ192
_____/ С.В.Сурова/
«___» _____ 2021 г.

| | |
|--------------|--|
| Подп. и дата | |
| Инв. № дубл. | |
| Взам. Инв. № | |
| Подп. и дата | |
| Инв. № подл. | |

УТВЕРЖДЕН
RU.17701729.06.02-01 81 01-1-ЛУ

МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ ОБМЕНА ВЕЩАМИ «SWOPPER»

Пояснительная записка

RU.17701729.06.02-01 81 01-1-ЛУ

Листов 52

| | |
|---------------------|--|
| <i>Инв. № подл</i> | |
| <i>Подп. и дата</i> | |
| <i>Взам. инв. №</i> | |
| <i>Инв. № дубл.</i> | |
| <i>Подп. и дата</i> | |

Оглавление

| | |
|--|----|
| 1. ВВЕДЕНИЕ | 3 |
| 1.1. Наименование программы | 3 |
| 1.2. Документы, на основании которых ведётся разработка | 3 |
| 2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ | 4 |
| 2.1. Назначение программы | 4 |
| 2.1.1 Функциональное назначение программы | 4 |
| 2.1.2 Эксплуатационное назначение программы | 4 |
| 2.2. Краткая характеристика области применения | 5 |
| 3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ | 6 |
| 3.1. Постановка задачи на разработку программы | 6 |
| 3.2. Описание алгоритма и функционирования программы | 6 |
| 3.2.1. Общий алгоритм работы программы | 6 |
| 3.2.2. Алгоритм регистрации нового профиля пользователя и вход в существующий профиль | 7 |
| 3.2.3. Алгоритм добавления объявления | 8 |
| 3.2.4. Алгоритм поиска объявления и фильтрации списка объявлений | 8 |
| 3.2.5. Алгоритм отображения нескольких типов сообщений в списке сообщений | 9 |
| 3.2.6. Алгоритм отображения списка сообщений в чате с пользователем | 9 |
| 3.2.7. Алгоритм осуществления звонка пользователю, опубликовавшему объявление | 10 |
| 3.3. Описание и обоснование выбора метода организации входных и выходных данных | 10 |
| 3.3.1. Описание метода организации входных данных | 10 |
| 3.3.2. Обоснование метода организации входных данных | 12 |
| 3.3.3. Описание метода организации выходных данных | 12 |
| 3.3.4. Обоснование метода организации выходных данных | 13 |
| 3.4. Описание и обоснование выбора метода выбора технических и программных средств | 13 |
| 3.4.1. Описание метода выбора технических и программных средств | 13 |
| 3.4.2. Обоснование метода выбора технических и программных средств | 13 |
| 4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ | 14 |
| 4.1. Ориентировочная экономическая эффективность | 14 |
| 4.2. Предполагаемая потребность | 14 |
| 4.3. Экономические преимущества разработки по сравнению с отечественными или зарубежными аналогами | 14 |
| СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ | 17 |
| ПРИЛОЖЕНИЕ 1 | 19 |
| ПРИЛОЖЕНИЕ 2 | 23 |
| ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ | 52 |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

1. ВВЕДЕНИЕ

1.1. Наименование программы

Наименование программы – «Мобильное приложение для обмена вещами «Swopper»» («Swopper Mobile App for Exchanging Things»).

1.2. Документы, на основании которых ведётся разработка

Основанием для разработки является учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденная академическим руководителем тема курсового проекта.

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

2.1. Назначение программы

2.1.1 Функциональное назначение программы

Функциональным назначением программы является предоставление пользователям возможности создавать объявления, которые будут содержать информацию об определенном товаре и/или об определенной услуге, осуществлять поиск объявлений других пользователей с помощью ввода текстового запроса или установки фильтров по различным параметрам. Кроме того, разрабатываемая программа выполняет организацию коммуникации пользователей в виде обмена текстовыми сообщениями и сообщениями, которые представляют изображение, а также осуществляет хранение данных о пользователях, опубликованных объявлениях и отправленных сообщениях. В общем и целом, функциональное назначение программы заключается в организации коммуникации пользователей, целью которой является обмен товарами и/или услугами.

2.1.2 Эксплуатационное назначение программы

С каждым десятилетием ограниченность природных ресурсов нашей планеты становится всё более ощутимой, как и количество отходов, вследствие чего растёт потребность в сокращении производственных темпов и расходов на производство, в повторном использовании произведённых продуктов и переработке отходов. Разрабатываемое мобильное приложение является востребованным на сегодняшний день, так как способствует удовлетворению данных потребностей.

Согласно прошлогоднему исследованию потребительского поведения, проведенного компанией «Платформа ОФД», которая ежедневно обрабатывает более сорока четырех миллионов чеков [15], на фоне пандемии, связанной с распространением коронавирусной инфекции, количество одежды, покупаемой россиянами, увеличилось почти в три раза в сравнении с 2019 годом [11]. Эти же результаты подтверждает анализ операций по картам всех банков в интернет-эквайринговой сети "Русского стандарта" [12]. В то же время, с ростом спроса на сырьё Федеральная Служба Государственной Статистики зафиксировала количество отходов, образованных за 2018 год, в размере 7,3 миллиардов тонн [10].

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Однако у большинства людей со временем появляются товары, в использовании которых они больше не нуждаются, но эти товары выполняют свои основные функции и могут быть необходимы другим людям. Хорошим примером являются детские игрушки или другие детские товары, которые теряют интерес ребёнка по мере его взросления и развития, но могут оказаться нужными для других детей младшего возраста. Данное мобильное приложение может быть использовано пользователями мобильных устройств с операционной системой Android для того, чтобы отдать даром такие товары или обменять их на другие, более востребованные. Тем самым повторно используются бывшие в употреблении товары, сокращаются сырьевые и энергетические расходы для производства новых товаров и сокращаются расходы на утилизацию, транспорт и переработку отходов.

2.2. Краткая характеристика области применения

«Мобильное приложение для обмена вещами «Swopper»» - программа для мобильных устройств с операционной системой Android, которая может быть использована для построения коммуникации с другими пользователями, для того чтобы отдать, приобрести или обменять товары и/или услуги безвозмездно, то есть без получения или траты денег, тем самым поддерживая идеи о повторном использовании вещей, о сокращении расходов на производство новых товаров и на уничтожение отходов и о сбережении не возобновляемых природных ресурсов.

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

3.1. Постановка задачи на разработку программы

Из функционального назначения программы следует, что программа должна решать поставленные задачи, описанные ниже:

- регистрация нового профиля пользователя, вход и выход из существующего профиля пользователя,
- добавление собственного объявления,
- просмотр списка объявлений, добавленных другими пользователями,
- поиск объявления по совпадению названия объявления и текстового запроса,
- фильтрация списка объявлений, добавленных другими пользователями, по типу объявления,
- фильтрация списка объявлений, добавленных другими пользователями, по категории объявления,
- фильтрация списка объявлений, добавленных другими пользователями, по местоположению объявления,
- отправка текстовых сообщений другому пользователю, добавившему объявление,
- отправка сообщений, содержащих изображение, другому пользователю, добавившему объявление,
- получение текстовых сообщений от других пользователей,
- получение сообщений, содержащих изображение, от других пользователей,
- осуществление звонка пользователю, добавившему объявление.

3.2. Описание алгоритма и функционирования программы

3.2.1. Общий алгоритм работы программы

Приложение реализует Single Activity Architecture [9]. Есть единственная главная активность приложения MainActivity, которая осуществляет управление фрагментами, представляющими отдельные экраны приложения, с помощью нижней навигационной панели BottomNavigationView [1], используя которую пользователь может перемещаться между главными экранами приложения («Объявления», «Профиль», «Добавить», «Чаты», «Настройки»), а также активность с помощью FragmentManager может выполнять замещение фрагментов при переходе пользователя в остальные экраны приложения.

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Иллюстрация навигационного графа приложения представлена ниже, где зелёным цветом выделены главные экраны приложения.

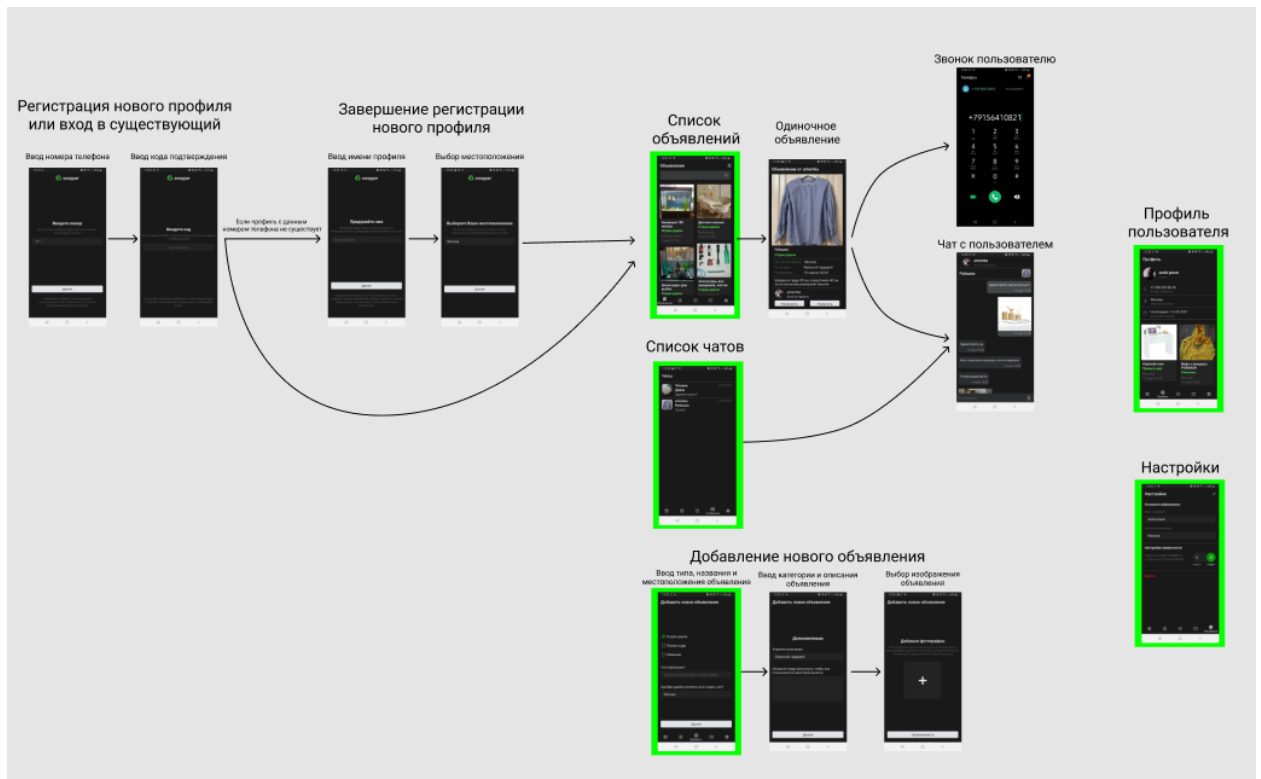


Рисунок 1. Иллюстрация навигационного графа приложения

3.2.2. Алгоритм регистрации нового профиля пользователя и вход в существующий профиль

При запуске приложения осуществляется проверка текущего пользователя на null с помощью свойства currentUser класса FirebaseAuth [4]. Если данное свойство равно null, то происходит процесс регистрации нового профиля пользователя или вход в существующий профиль с помощью кода подтверждения, который приходит в сообщении на номер телефона. Если данного номера телефона нет в базе данных, то для завершения регистрации пользователю необходимо ввести непустое имя профиля, не содержащее нецензурных слов и выбрать местоположение из предложенного списка городов России с численностью населения более 100 тысяч человек на 2021 год.

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

3.2.3. Алгоритм добавления объявления

После нажатия на пункт «Добавить» на нижней навигационной панели приложения отображается экран приложения, в котором пользователю, чтобы завершить добавление объявления необходимо:

- выбрать тип объявления из списка «Отдам даром», «Приму в дар», «Обменяю»,
- ввести непустое название объявления, не содержащее нецензурных слов,
- выбрать местоположение объявления из списка городов России с численностью населения более 100 тысяч человек на 2021 год,
- выбрать категорию объявления из списка «Мужской гардероб», «Женский гардероб», «Детский гардероб», «Электроника», «Автотовары», «Для дома и дачи», «Спорт», «Хобби и развлечения», «Красота и здоровье», «Детские товары», «Продукты питания», «Животные», «Услуги», «Другое»
- ввести непустое описание объявления, не содержащее нецензурных слов,
- выбрать изображение объявления из существующих изображений на устройстве.

3.2.4. Алгоритм поиска объявления и фильтрации списка объявлений

Для поиска объявления по текстовому запросу пользователю необходимо ввести текстовый запрос в поисковую строку, и после нажатия на кнопку, завершающую ввод, поле `mQueryFilter` будет проинициализировано введенным значением (или значением по умолчанию при сбросе поиска). После чего будет запущен метод `filter` класса `AdvertsFragment`.

Для фильтрации списка объявлений по типу, категории и/или местоположению объявления пользователю необходимо выбрать соответствующие значения в каждом из фильтров (или установить значение по умолчанию для фильтрации лишь по некоторым параметрам, или сбросить фильтры до значений по умолчанию при отмене фильтрации). После завершения выбора значений фильтров поля `mTypeFilter`, `mCategoryFilter`, `mLocationFilter` будут проинициализированы введенными значениями и будет запущен метод `filter` класса `AdvertsFragment`.

Метод `filter` осуществляет поиск объявлений,

1. название которых содержит значение поля `mQueryFilter` (если оно не проинициализировано значением по умолчанию), с помощью метода `bitap` в классе `ObsceneFilter`, который реализует одноименный алгоритм `Bitap` для

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

нечеткого поиска подстроки в строке с пороговым значением ошибок (в данном случае оно равно округленному вверх значению $0,2 * \text{длина текстового запроса}$),

2. тип, категория и местоположение которых совпадает со значениями полей mTypeFilter, mCategoryFilter, mLocationFilter соответственно (если они не проинициализированы значениями по умолчанию).

3.2.5. Алгоритм отображения нескольких типов сообщений в списке сообщений

Для отображения списков в программе используется инструмент RecyclerView [2] с соответствующими классами Adapter, который расширяет макеты представлений элемента списка и привязывает данные к их представлениям, Holder, который хранит ссылки на представления в файле макета элемента списка и View, который содержит данные представления в файле макета элемента списка. Для отображения двух типов сообщений в RecyclerView используется интерфейс MessageView, от которого наследуется представление каждого типа сообщения TextMessageView и ImageMessageView, интерфейс MessageHolder, от которого наследуются классы для каждого типа сообщения TextMessageHolder и ImageMessageHolder, а для создания объекта класса MessageView и MessageHolder соответствующего типа сообщения используются методы getView в классе ViewFactory и getHolder в классе HolderFactory.

3.2.6. Алгоритм отображения списка сообщений в чате с пользователем

При переходе на экран приложения, который отображает чат с пользователем, загружаются только несколько последних сообщений с наибольшим значением отметки времени, когда сообщение было отправлено. Количество загружаемых сообщений определяется значением переменной mCountMessages = 10 в классе ChatFragment. Если пользователь прокручивает список к более старым сообщениям, то есть к сообщениям с меньшим значением отметки времени, и позиция сообщения, которое отображено на экране первым сверху, в списке сообщений меньше или равна 3, то осуществляется дополнительная загрузка следующих mCountMessages сообщений, которые добавляются в начало списка с помощью функции расширения addItemToTop класса ChatAdapter.

Когда пользователь отправляет текстовое сообщение или сообщение, содержащее изображение, то оно добавляется в конец списка с помощью метода addItemToBottom класса ChatAdapter и запускается прокрутка списка до последнего элемента с помощью метода smoothScrollToPosition класса RecyclerView. [6]

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

3.2.7. Алгоритм осуществления звонка пользователю, опубликовавшему объявление

При просмотре объявления, опубликованного другим профилем, пользователю доступна кнопка «Позвонить», если в настройках профиля, который опубликовал данное объявление выбран соответствующий параметр видимости номера телефона другим пользователям. После нажатия на данную кнопку запускается новая активность `Intent.ACTION_DIAL` [5], которая открывает приложение для звонков на устройстве пользователя и вставляет в поле ввода номера телефона номер телефона пользователя, опубликовавшего объявление. После чего пользователь, подтверждая своё действие, может выбрать кнопку вызова и осуществить звонок.

3.3. Описание и обоснование выбора метода организации входных и выходных данных

3.3.1. Описание метода организации входных данных

При регистрации нового профиля пользователя программа должна обрабатывать:

- введенный пользователем номер телефона,
- введенный пользователем код подтверждения,
- введенное пользователем имя профиля,
- выбранное пользователем местоположение из предложенного списка городов России с численностью населения более 100 тысяч человек на 2021 год.

При входе пользователя в существующий профиль программа должна обрабатывать:

- введенный пользователем номер телефона,
- введенный пользователем код подтверждения.

При редактировании профиля пользователя программа должна обрабатывать:

- введенное пользователем имя профиля,
- выбранное пользователем местоположение из предложенного списка городов России с численностью населения более 100 тысяч человек на 2021 год,
- выбранное пользователем изображение профиля из существующих на устройстве.

При выборе настроек приложения программа должна обрабатывать выбранную пользователем видимость номера телефона для других пользователей из предложенных значений «скрыт» и «виден».

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

При добавлении собственного объявления программа должна обрабатывать:

- выбранный пользователем тип объявления из предложенного списка: «Отдам даром», «Приму в дар» и «Обменяю»,
- введенное пользователем название объявления,
- выбранное пользователем местоположение из предложенного списка городов России с численностью населения более 100 тысяч человек на 2021 год,
- выбранную пользователем категорию объявления из предложенного списка: «Мужской гардероб», «Женский гардероб», «Детский гардероб», «Электроника», «Автотовары», «Для дома и дачи», «Спорт», «Хобби и развлечения», «Красота и здоровье», «Детские товары», «Продукты питания», «Животные», «Услуги», «Другое»,
- введенное пользователем описание объявления,
- выбранное пользователем изображение объявления из существующих на устройстве.

При поиске объявлений по названию в списке объявлений программа должна обрабатывать текстовый запрос, введенный пользователем.

При фильтрации списка объявлений, добавленных другими пользователями, по типу объявления программа должна обрабатывать выбранный пользователем тип объявления из предложенного списка: «Не важно», «Отдам даром», «Приму в дар», «Обменяю».

При фильтрации списка объявлений, добавленных другими пользователями, по категории объявления программа должна обрабатывать выбранную пользователем категорию объявления из предложенного списка: «Не важно», «Мужской гардероб», «Женский гардероб», «Детский гардероб», «Электроника», «Автотовары», «Для дома и дачи», «Спорт», «Хобби и развлечения», «Красота и здоровье», «Детские товары», «Продукты питания», «Животные», «Услуги», «Другое».

При фильтрации списка объявлений, добавленных другими пользователями, по местоположению объявления программа должна обрабатывать выбранное пользователем местоположение из предложенного списка городов России с численностью населения более 100 тысяч на 2021 год и значения «Не важно».

При отправке текстовых сообщений другому пользователю, добавившему объявление, программа должна обрабатывать введенный пользователем текст сообщения.

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

При отправке сообщений, содержащих изображение, другому пользователю, добавившему объявление, программа должна обрабатывать выбранное пользователем изображение из существующих на устройстве.

3.3.2. Обоснование метода организация входных данных

Приведённая организация входных данных полностью отвечает требованиям технического задания и является одной из наиболее простых.

3.3.3. Описание метода организации выходных данных

При просмотре собственного профиля пользователя программа должна выводить:

- имя профиля пользователя,
- номер телефона пользователя,
- местоположение пользователя,
- дату регистрации пользователя в формате ДД.ММ.ГГГГ,
- изображение профиля пользователя,
- список объявлений, добавленных пользователем (изображение, название, тип, местоположение и дату публикации объявления).

При просмотре списка объявлений, добавленных другими пользователями, программа должна выводить список объявлений (изображение, название, тип, местоположение и дату публикации объявления), название которых содержит текстовый запрос, а тип, категория и местоположение соответствуют выбранным фильтрам.

При просмотре объявления, добавленного другим пользователем, программа должна выводить изображение, название, тип, местоположение, категорию, дату публикации и описание объявления, а также изображение и имя профиля пользователя, добавившего объявление.

При просмотре собственного объявления, добавленного текущим пользователем, программа должна выводить изображение, название, тип, местоположение, категорию, дату публикации и описание объявления.

При просмотре текстового сообщения, отправленного другим пользователем текущему пользователю, программа должна выводить текст и дату отправки сообщения.

При просмотре сообщения, содержащего изображение и отправленного другим пользователем текущему пользователю, программа должна выводить изображение и дату отправки сообщения.

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

При просмотре списка чатов с другими пользователями, которые содержат хотя бы одно сообщение, программа должна выводить список чатов (изображение и название объявления, имя профиля собеседника, текст и дату отправки последнего сообщения).

3.3.4. Обоснование метода организации выходных данных

Приведённая организация выходных данных полностью отвечает требованиям технического задания и является одной из наиболее простых.

3.4. Описание и обоснование выбора метода выбора технических и программных средств

3.4.1. Описание метода выбора технических и программных средств

Для бесперебойной работы программного продукта требуется мобильное устройство с:

- сенсорным экраном диагональю не меньше 4,7 дюйма,
- операционной системой Android не ниже версии 5.0, то есть с минимальной версией SDK 21,
- объемом свободной встроенной памяти не меньше 100 МБ,
- объёмом оперативной памяти не меньше 1 ГБ.

3.4.2. Обоснование метода выбора технических и программных средств

Количество мобильных устройств с версией операционной системы Android выше 5.0 составляет 94,1% от всех мобильных устройств с данной операционной системой [3].

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

4.1. Ориентировочная экономическая эффективность

В рамках данной работы расчет экономической эффективности не предусмотрен.

4.2. Предполагаемая потребность

Данную программу могут использовать все пользователи мобильных устройств с операционной системой Android, кто хочет отдать, приобрести или обменять товары и/или услуги безвозмездно, тем самым поддерживая идеи о повторном использовании вещей, о сокращении расходов на производство новых товаров и на уничтожение отходов, и о сбережении не возобновляемых природных ресурсов. Вышеперечисленные проблемы и их решение являются востребованными в областях, связанных с влиянием потребительского поведения на экологию и экономику.

4.3. Экономические преимущества разработки по сравнению с отечественными или зарубежными аналогами

На момент создания программы наиболее продвинутыми аналогами в области приложений для обмена вещами являются: «Отдам даром в РФ. Барахолка по всей России.» [14], «Отдам даром» [13], «Swapbox - обмен товарами и услугами» [7], «Swop.it – обмен товарами рядом» [8] и другие.

«Отдам даром в РФ. Барахолка по всей России.» содержит рекламу поверх содержимого страницы вверху и внизу каждой страницы, а также рекламу с музыкой при переходе между страницами приложения. У данного аналога отсутствуют следующие функции:

- редактирование собственного профиля пользователя,
- отображение фотографии объявления,
- выбор таких настроек приватности, как видимость номера телефона другим пользователям,
- фильтрация по категории объявления,
- нечёткий (допускающий опечатки) нечувствительный к регистру поиск объявлений по текстовому запросу,
- звонок пользователю, опубликовавшему объявление.

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Кроме того, коммуникация пользователей осуществляется однонаправленно через отправку комментария с указанием имени, email, номера телефона профиля и текста сообщения, и приложение завершается некорректно при переходе в фрагмент «Настройки».

«Отдам даром» содержит рекламу поверх содержимого страницы приложения. У данного аналога отсутствуют следующие функции:

- просмотр объявлений других городов,
- отображение фотографии объявления,
- ввод названия объявления,
- фильтрация по местоположению объявления,
- нечёткий (допускающий опечатки) поиск объявлений по текстовому запросу,
- звонок пользователю, опубликовавшему объявление.

Кроме того, коммуникация пользователей осуществляется однонаправленно через аутентификацию в сервисе «Вконтакте», которая проходит дважды, предоставление разрешения данному приложению и репост соответствующего объявления.

У «Swarbox - обмен товарами и услугами» отсутствуют следующие функции:

- фильтрация по местоположению объявления,
- нечёткий (допускающий опечатки) поиск объявлений по текстовому запросу.

Более того, при выборе местоположения при регистрации профиля приложение работает некорректно, вследствие чего невозможно завершить регистрацию, войти в профиль, добавить собственное объявление и откликнуться на существующее объявление другого пользователя.

«Swor.it – обмен товарами рядом» содержит рекламу поверх содержимого страницы приложения. У данного аналога отсутствуют следующие функции:

- просмотр списка объявлений (необходимо пролистывать каждое объявление),
- просмотр собственного объявления,
- фильтрация по местоположению объявления,
- фильтрация по категории объявления,
- нечёткий (допускающий опечатки) поиск объявлений по текстовому запросу,

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

- звонок пользователю, опубликовавшему объявление.

Кроме того, в приложении реализована только возможность обмена опубликованного товара на внутреннюю валюту приложения, которую можно приобрести за настоящую валюту, а также коммуникация происходит не посредством чата, а отметкой объявления как понравившегося с одной стороны и выбор пользователем, опубликовавшим объявление, кандидата на обмен среди данных отметок.

Данное приложение:

- распространяется бесплатно;
- не содержит рекламу;
- предоставляет полностью бесплатный контент, то есть не требует вложения никаких денежных средств во время использования и не содержит платные версии и функции;
- имеет функцию выбора настроек приватности, таких как видимость номера телефона другим пользователям;
- имеет нечёткий (допускающий опечатки) поиск объявлений по текстовому запросу;
- имеет функцию фильтрации по категориям и местоположению;
- имеет функцию звонка пользователю, опубликовавшему объявление;
- имеет внутренний чат для обмена текстовыми сообщениями и сообщениями, которые представляют изображение.

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

- 1) BottomNavigationView [Электронный ресурс] / Android Developers. Режим доступа: <https://developer.android.com/reference/com/google/android/material/bottomnavigation/BottomNavigationView?hl=ru-RU>, свободный. (дата обращения: 15.05.2021)
- 2) Create dynamic lists with RecyclerView [Электронный ресурс] / Android Developers. Режим доступа: <https://developer.android.com/guide/topics/ui/layout/recyclerview?hl=ru>, свободный. (дата обращения: 15.05.2021)
- 3) Distribution dashboard [Электронный ресурс] / Android Developers. Режим доступа: <https://developer.android.com/about/dashboards/index.html>, свободный. (дата обращения: 15.05.2021)
- 4) Firebase.Auth.FirebaseAuth Class Reference [Электронный ресурс] / Firebase. Режим доступа: <https://firebase.google.com/docs/reference/unity/class/firebase/auth/firebase-auth?hl=ru>, свободный. (дата обращения: 15.05.2021)
- 5) Intent [Электронный ресурс] / Android Developers. Режим доступа: <https://developer.android.com/reference/android/content/Intent?hl=ru>, свободный. (дата обращения: 15.05.2021)
- 6) RecyclerView [Электронный ресурс] / Android Developers. Режим доступа: <https://developer.android.com/reference/androidx/recyclerview/widget/RecyclerView?hl=he>, свободный. (дата обращения: 15.05.2021)
- 7) Swapbox - обмен товарами и услугами [Электронный ресурс] / Leonid Пыаев. Режим доступа: <https://play.google.com/store/apps/details?id=su.swapbox&hl=ru&gl=US>, свободный. (дата обращения: 15.05.2021)
- 8) Swop.it – обмен товарами рядом [Электронный ресурс] / Pointfor Services. Режим доступа: <https://play.google.com/store/apps/details?id=it.swop.android&hl=ru&gl=US>, свободный. (дата обращения: 15.05.2021)
- 9) Use Android Jetpack to Accelerate Your App Development [Электронный ресурс] / Android Developers. Режим доступа: <https://android-developers.googleblog.com/2018/05/use-android-jetpack-to-accelerate-your.html?m=1>, свободный. (дата обращения: 15.05.2021)
- 10) БГД – Охрана окружающей среды в России [Электронный ресурс] / Федеральная служба государственной статистики. Режим доступа: https://www.gks.ru/bgd/regl/b18_54/Main.htm, свободный. (дата обращения: 15.05.2021)
- 11) Исследование: продажи одежды в России в июне выросли втрое [Электронный ресурс] / ТАСС. Режим доступа: <https://tass.ru/ekonomika/8920501>, свободный. (дата обращения: 15.05.2021)
- 12) Исследование: россияне стали в три раза больше покупать в интернете [Электронный ресурс] / ТАСС. Режим доступа: <https://tass.ru/ekonomika/9207319>, свободный. (дата обращения: 15.05.2021)
- 13) Отдам даром [Электронный ресурс] / DimICE. Режим доступа: <https://play.google.com/store/apps/details?id=ru.dimice.darom>, свободный. (дата обращения: 15.05.2021)

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

14) Отдам даром в РФ. Барахолка по всей России [Электронный ресурс] / InterestingAppTeam. Режим доступа: <https://play.google.com/store/apps/details?id=ru.interapp.fleamarket&hl=ru&gl=US>, свободный. (дата обращения: 15.05.2021)

15) Чек Индекс: как меняется потребительское поведение в разных товарных категориях [Электронный ресурс] / Платформа ОФД. Режим доступа: <https://platformaofd.ru/news/checkindex-kak-menyaetsya-potrebitelskoe-povedenie/>, свободный. (дата обращения: 15.05.2021)

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

ПРИЛОЖЕНИЕ 1

ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ

Таблица 1.1 – Описание и функциональное назначение классов в пакете com.example.swopper

| Класс/файл | Назначение |
|--------------|--|
| MainActivity | Класс главной активности, с которой начинается выполнения приложения Android |

Таблица 1.2 – Описание и функциональное назначение файлов в пакете com.example.swopper.database

| Класс/файл | Назначение |
|------------|---|
| values | Файл, который содержит константные значения для работы с базой данных |
| functions | Файл, который содержит функции, выполняющие обращение к базе данных |

Таблица 1.3 – Описание и функциональное назначение классов и файлов в пакете com.example.swopper.utils

| Класс/файл | Назначение |
|-----------------------|---|
| AppChildEventListener | Класс, который является модификацией базового класса ChildEventListener |
| AppTextWatcher | Класс, который является модификацией базового класса TextWatcher |
| AppValueEventListener | Класс, который является модификацией базового класса ValueEventListener |
| ObsceneFilter | Класс фильтра нецензурных слов |
| UserStatus | Класс перечислений текущего статуса пользователя |
| utils | Файл, который содержит утилиты – вспомогательные методы и константные значения для работы программы |

Таблица 1.4 – Описание и функциональное назначение классов в пакете com.example.swopper.models

| Класс/файл | Назначение |
|--------------|--|
| AdvertModel | Класс данных объявления |
| CommonModel | Класс данных общей сущности, содержащей значения объявления, пользователя и сообщения для чата |
| MessageModel | Класс данных сообщения |
| UserModel | Класс данных пользователя |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 1.5 – Описание и функциональное назначение классов в пакете com.example.swopper.ui.add_advert

| Класс/файл | Назначение |
|----------------------------------|--|
| AddAdditionalInformationFragment | Класс фрагмента, в котором пользователь осуществляет выбор категории и ввод описания объявления при добавлении нового объявления |
| AddImageFragment | Класс фрагмента, в котором пользователь осуществляет выбор изображения объявления при добавлении нового объявления |
| AddMainInformationFragment | Класс фрагмента, в котором пользователь осуществляет ввод названия и выбор типа и местоположения объявления при добавлении нового объявления |
| BasicFragment | Класс базового фрагмента при добавлении нового объявления |

Таблица 1.6 – Описание и функциональное назначение классов в пакете com.example.swopper.ui.adverts

| Класс/файл | Назначение |
|------------------------------|--|
| AdvertFragment | Класс фрагмента объявления |
| AdvertsAdapter | Класс адаптера списка объявлений, который расширяет макеты объявлений и привязывает данные к их представлениям |
| AdvertsAdapter.AdvertsHolder | Класс, который осуществляет хранение ссылок на представления в файле макета объявления |
| AdvertsFragment | Класс фрагмента списка объявлений |
| BasicFragment | Класс базового фрагмента при отображении списка объявлений |

Таблица 1.7 – Описание и функциональное назначение классов в пакете com.example.swopper.ui.chat

| Класс/файл | Назначение |
|--------------|----------------------|
| ChatFragment | Класс фрагмента чата |

Таблица 1.8 – Описание и функциональное назначение классов в пакете com.example.swopper.ui.chat.message_recycler_view

| Класс/файл | Назначение |
|-------------|--|
| ChatAdapter | Класс адаптера списка сообщений, который расширяет макеты сообщений и привязывает данные к их представлениям |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 1.9 – Описание и функциональное назначение классов в пакете
com.example.swopper.ui.chat.message_recycler_view.holders

| Класс/файл | Назначение |
|--------------------|---|
| HolderFactory | Класс, который создаёт объект одного из классов, которые хранят ссылки на представления в файле макета сообщения, в зависимости от типа сообщения |
| ImageMessageHolder | Класс, который осуществляет хранение ссылок на представления в файле макета сообщения, которое представляет изображение |
| MessageHolder | Интерфейс, содержащий объявления методов для классов, которые осуществляют хранение ссылок на представления в файле макета сообщения |
| TextMessageHolder | Класс, который осуществляет хранение ссылок на представления в файле макета текстового сообщения |

Таблица 1.10 – Описание и функциональное назначение классов в пакете
com.example.swopper.ui.chat.message_recycler_view.views

| Класс/файл | Назначение |
|------------------|--|
| ViewFactory | Класс, который создаёт объект одного из классов, которые содержат данные представлений в файле макета сообщения, в зависимости от типа сообщения |
| ImageMessageView | Класс, который содержит данные представлений в файле макета сообщения, которое представляет изображение |
| MessageView | Интерфейс, содержащий объявления полей для классов, содержащих данные представлений в файле макета сообщения |
| TextMessageView | Класс, который содержит данные представлений в файле макета текстового сообщения |

Таблица 1.11 – Описание и функциональное назначение классов в пакете
com.example.swopper.ui.chats

| Класс/файл | Назначение |
|---------------|--|
| ChatsAdapter | Класс адаптера списка чатов, который расширяет макеты чатов и привязывает данные к их представлениям |
| ChatsFragment | Класс фрагмента списка чатов |
| BasicFragment | Класс базового фрагмента при отображении списка чатов |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 1.12 – Описание и функциональное назначение классов в пакете
com.example.swopper.ui.profile

| Класс/файл | Назначение |
|-----------------|---|
| ProfileFragment | Класс фрагмента профиля пользователя |
| BasicFragment | Класс базового фрагмента при отображении профиля пользователя |

Таблица 1.13 – Описание и функциональное назначение классов в пакете
com.example.swopper.ui.registration

| Класс/файл | Назначение |
|-----------------------|--|
| EnterCodeFragment | Класс фрагмента, в котором пользователь осуществляет ввод кода верификации при регистрации нового профиля или при входе в существующий профиль |
| EnterLocationFragment | Класс фрагмента, в котором пользователь осуществляет выбор местоположения при регистрации нового профиля |
| EnterPhoneFragment | Класс фрагмента, в котором пользователь осуществляет ввод номера телефона при регистрации нового профиля или при входе в существующий профиль |
| EnterUsernameFragment | Класс фрагмента, в котором пользователь осуществляет ввод имени профиля при регистрации нового профиля |

Таблица 1.14 – Описание и функциональное назначение классов в пакете
com.example.swopper.ui.settings

| Класс/файл | Назначение |
|------------------|-------------------------------------|
| SettingsFragment | Класс фрагмента настроек приложения |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

ПРИЛОЖЕНИЕ 2

ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ, МЕТОДОВ И СВОЙСТВ

Таблица 2.1.1 – Описание полей и свойств класса MainActivity

| Имя | Модификатор доступа | Тип | Назначение |
|-----------------------|---------------------|----------------------|---|
| mBinding | private | ActivityMainBinding | Объект класса, сгенерированного библиотекой DataBinding, которая позволяет привязывать представления в файле макета к источникам данных |
| mBottomNavigationView | public | BottomNavigationView | Нижняя навигационная панель |
| mActionBarToolbar | public | Toolbar | Верхняя панель приложения |

Таблица 2.1.2 – Описание методов класса MainActivity

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------------|---------------------|------|-----------|---|
| initNavigationBar | private | Unit | - | Инициализация нижней навигационной панели и верхней панели приложения |
| setStartFragment | private | Unit | - | Переход на начальный фрагмент в зависимости от того, зарегистрирован пользователь или нет |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.2.1 – Описание полей и свойств файла values

| Имя | Модификатор доступа | Тип | Назначение |
|----------------------|---------------------|-------------------|--|
| AUTH | public | FirebaseAuth | Объект, осуществляющий работу с Firebase authentication API и управляющий учетными записями и учетными данными пользователей |
| CURRENT_UID | public | String | Уникальный идентификатор текущего пользователя |
| USER | public | UserModel | Текущий пользователь |
| REF_DATABASE_ROOT | public | DatabaseReference | Ссылка на базу данных для доступа к чтению и записи данных |
| REF_STORAGE_ROOT | public | StorageReference | Ссылка на хранилище файлов для доступа к загрузке и получению файлов |
| FOLDER_PROFILE_IMAGE | public | String | Название директории в хранилище с изображениями профилей пользователей |
| FOLDER_ADVERT_IMAGE | public | String | Название директории в хранилище с изображениями объявлений |
| FOLDER_MESSAGE_IMAGE | public | String | Название директории в хранилище с изображениями сообщений |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.2.1

| Имя | Модификатор доступа | Тип | Назначение |
|--------------------|---------------------|--------|---|
| TYPE_MESSAGE_TEXT | public | String | Название типа сообщений для текстовых сообщений в базе данных |
| TYPE_MESSAGE_IMAGE | public | String | Название типа сообщений для сообщений, представляющих изображение, в базе данных |
| NODE_USERS | public | String | Название узла базы данных, в котором хранятся данные профилей пользователей |
| NODE_ADVERTS | public | String | Название узла базы данных, в котором хранятся данные объявлений пользователей |
| NODE_MESSAGES | public | String | Название узла базы данных, в котором хранятся данные сообщений пользователей |
| NODE_CHATS | public | String | Название узла базы данных, в котором хранятся данные чатов пользователей |
| CHILD_ID | public | String | Название ключа для значения идентификатора пользователя, объявления, сообщения или чата |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.2.1

| Имя | Модификатор доступа | Тип | Назначение |
|-------------------|---------------------|--------|---|
| CHILD_USERNAME | public | String | Название ключа для значения имени профиля пользователя |
| CHILD_PHONE | public | String | Название ключа для значения номера телефона пользователя |
| CHILD_HIDDEN | public | String | Название ключа для значения скрыт ли номер телефон пользователя от других пользователей |
| CHILD_PHOTO_URL | public | String | Название ключа для значения адреса ресурса, представляющего изображение профиля пользователя или объявления |
| CHILD_STATUS | public | String | Название ключа для значения статуса профиля |
| CHILD_REGISTRATED | public | String | Название ключа для значения отметки времени, когда был зарегистрирован профиль |
| CHILD_LOCATION | public | String | Название ключа для значения местоположения профиля или объявления |
| CHILD_NAME | public | String | Название ключа для значения названия объявления |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.2.1

| Имя | Модификатор доступа | Тип | Назначение |
|-------------------|---------------------|--------|---|
| CHILD_DESCRIPTION | public | String | Название ключа для значения описания объявления |
| CHILD_TYPE | public | String | Название ключа для значения типа объявления или сообщения |
| CHILD_CATEGORY | public | String | Название ключа для значения категории объявления |
| CHILD_POSTED | public | String | Название ключа для значения отметки времени, когда было добавлено объявление |
| CHILD_OWNER | public | String | Название ключа для значения идентификатора пользователя, который добавил объявление |
| CHILD_TEXT | public | String | Название ключа для значения текста сообщения |
| CHILD_FROM | public | String | Название ключа для значения идентификатора пользователя, отправившего сообщение |
| CHILD_SENDED | public | String | Название ключа для значения отметки времени, когда было отправлено сообщение |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.2.1

| Имя | Модификатор доступа | Тип | Назначение |
|----------------|---------------------|--------|--|
| CHILD_FILE_URL | public | String | Название ключа для значения адреса ресурса, представляющего файл, отправленный в сообщении |
| CHILD_USER | public | String | Название ключа для значения идентификатора пользователя-собеседника в чате |
| CHILD_ADVERT | public | String | Название ключа для значения идентификатора объявления, которое закреплено в чате |

Таблица 2.2.2 – Описание методов файла functions

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-----------------------|---------------------|------|--|--|
| initFirebase | public | Unit | - | Инициализация всех значений, связанных с работой с базой данных и хранилищем: AUTH, CURRENT_UID, USER, REF_DATABASE_ROOT, REF_STORAGE_ROOT |
| initUser | public | Unit | - | Инициализация текущего пользователя из базы данных |
| setChangesTo Database | public | Unit | username: String, location: String, hidden: String | Обновление значений имени, местоположения профиля пользователя и видимость номера телефона пользователя для других пользователей в базе данных |
| setImageTo Storage | public | Unit | uri: Uri, path: StorageReference, crossinline function: () -> Unit | Функция высшего порядка, которая загружает изображение в хранилище |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.2.2

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|----------------------|---------------------|------|--|---|
| getUrlFromStorage | public | Unit | path: StorageReference, crossinline function: (url: String) -> Unit | Функция высшего порядка, которая получает адрес изображения |
| setUrlToDatabase | public | Unit | ref: DatabaseReference, url: String, crossinline function: () -> Unit | Функция высшего порядка, которая отправляет значение адреса ресурса по указанному адресу в базу данных |
| sendTextMessage | public | Unit | message: String, recipient: String, function: () -> Unit | Функция высшего порядка, которая отправляет данные текстового сообщения со сгенерированным идентификатором, с отправителем в виде текущего пользователя, с указанным текстом и получателем в базу данных |
| uploadImageToStorage | public | Unit | uri: Uri, messageKey: String, recipient: String | Отправка сообщения, которое представляет собой изображение, которое включает в себя загрузку изображения в хранилище, получение адреса данного изображения и отправка данных сообщения с отправителем в виде текущего пользователя, с указанным идентификатором и получателем в базу данных |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.2.2

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|----------------------|---------------------|--------|---|--|
| sendImage Message | public | Unit | url: String, messageKey: String, recipient: String | Отправка данных сообщения, представляющего изображение, с отправителем в виде текущего пользователя, с указанным идентификатором, получателем и адресом ресурса в базу данных |
| getId | public | String | first: String, second: String | Функция, которая преобразует указанные строки по следующему правилу: { first.length } x { second.length } x { first } { second } Используется для создания идентификатора чата по идентификатору собеседника и закрепленного объявления |
| saveToChatList | public | Unit | userId: String, advertId: String | Отправка данных чата со сгенерированным идентификатором, с указанными идентификаторами собеседника и закрепленного объявления в базу данных |

Таблица 2.3.1.1 – Описание полей и свойств класса ObsceneFilter

| Имя | Модификатор доступа | Тип | Назначение |
|-----------|---------------------|------------------------------|---|
| forbidden | public | List<String> | Список запрещенных слов, поиск которых будет выполняться фильтром |
| replaces | public | Map<String, List<String>> | Список замен, которые могут использовать пользователи для маскировки запрещенного слова |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.3.1.2 – Описание методов класса ObsceneFilter

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------|---------------------|---------|--|---|
| min | private | Int | a: Int, b: Int, c: Int | Поиск минимума из трех целых числе |
| levenshtein | private | Boolean | input: String, pattern: String, limit: Int | Поиск заданной подстроки в строке с указанным максимальным количеством ошибок с помощью расстояния Левенштейна |
| bitap | public | Boolean | input: String, pattern: String, limit: Int | Поиск заданной подстроки в строке с указанным максимальным количеством ошибок с помощью алгоритма Bitap |
| search | public | Boolean | phrase: String, filter: Filters | Неточный поиск слов из списка forbidden с учетом возможных замен из словаря replaces в заданной строке с помощью указанного алгоритма для поиска (Bitap по умолчанию) |

Таблица 2.3.2 – Описание методов класса UserStatus

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|--------|---------------------|------|-----------------------|--|
| update | public | Unit | newStatus: UserStatus | Обновление значения статуса текущего пользователя на указанное |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.3.3.1 – Описание полей и свойств файла utils

| Имя | Модификатор доступа | Тип | Назначение |
|--------------|---------------------|--------------|-------------------------------|
| APP_ACTIVITY | public | MainActivity | Главная активность приложения |

Таблица 2.3.3.2 – Описание методов файла utils

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------------|---------------------|--------|--|---|
| showToast | public | Unit | message: String | Отображение Toast с указанным сообщением |
| restartActivity | public | Unit | - | Перезапуск главной активности приложения |
| replaceFragment | public | Unit | fragment: Fragment, addStack: Boolean | Замещение фрагмента в главном контейнере приложения заданным фрагментом |
| replaceFragment | public | Unit | fragment: Fragment, id: Int, addStack: Boolean | Замещение фрагмента в контейнере с указанным id заданным фрагментом |
| hideKeyboard | public | Unit | - | Функция, которая убирает клавиатуру с экрана |
| String.asDate | public | String | - | Функция расширения класса String, которая выполняет преобразование отметки времени в формате "dd.MM.yyyy" |
| String.asDateTime | public | String | - | Функция расширения класса String, которая выполняет преобразование отметки времени в формате "d MMMM HH:mm" |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.3.3.2

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|--------------------|---------------------|--------|---------------------------|--|
| String.asTime | public | String | - | Функция расширения класса String, которая выполняет преобразование отметки времени в формате "HH:mm" |
| ImageView.setImage | public | Unit | url: String, default: Int | Функция расширения класса ImageView, которая устанавливает изображение, расположенное по указанному адресу |

Таблица 2.4.1.1 – Описание полей и свойств класса AdvertModel

| Имя | Модификатор доступа | Тип | Назначение |
|-------------|---------------------|--------|--|
| id | public | String | Идентификатор объявления |
| name | public | String | Название объявления |
| description | public | String | Описание объявления |
| type | public | String | Тип объявления |
| category | public | String | Категория объявления |
| photoUrl | public | String | Адрес изображения объявления |
| location | public | String | Местоположение объявления |
| posted | public | Any | Отметка времени, когда было добавлено объявление |
| owner | public | String | Идентификатор пользователя, который добавил объявление |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.4.1.2 – Описание методов класса AdvertModel

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|--------|---------------------|---------|-------------|---|
| equals | public | Boolean | other: Any? | Переопределенный метод, который сравнивает объявления по идентификатору |

Таблица 2.4.2.1 – Описание полей и свойств класса CommonModel

| Имя | Модификатор доступа | Тип | Назначение |
|----------|---------------------|--------|---|
| id | public | String | Идентификатор чата |
| user | public | String | Идентификатор пользователя, который является собеседником |
| username | public | String | Имя профиля пользователя, который является собеседником |
| advert | public | String | Идентификатор объявления, которое закреплено в чате |
| name | public | String | Название объявления, которое закреплено в чате |
| photoUrl | public | String | Адрес изображения объявления, которое закреплено в чате |
| text | public | String | Текст последнего сообщения в чате |
| sended | public | Any | Отметка времени, когда было отправлено последнее сообщение в чате |

Таблица 2.4.2.2 – Описание методов класса CommonModel

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|--------|---------------------|---------|-------------|---|
| equals | public | Boolean | other: Any? | Переопределенный метод, который сравнивает чаты по идентификатору |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.4.3.1 – Описание полей и свойств класса MessageModel

| Имя | Модификатор доступа | Тип | Назначение |
|---------|---------------------|--------|--|
| id | public | String | Идентификатор сообщения |
| type | public | String | Тип сообщения |
| text | public | String | Текст сообщения |
| from | public | String | Идентификатор пользователя, который отправил сообщение |
| sended | public | Any | Отметка времени, когда было отправлено сообщение |
| fileUrl | public | String | Адрес файла, содержащегося в сообщении |

Таблица 2.4.3.2 – Описание методов класса MessageModel

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|--------|---------------------|---------|-------------|--|
| equals | public | Boolean | other: Any? | Переопределенный метод, который сравнивает сообщения по идентификатору |

Таблица 2.4.4.1 – Описание полей и свойств класса UserModel

| Имя | Модификатор доступа | Тип | Назначение |
|----------|---------------------|--------|---|
| id | public | String | Идентификатор профиля пользователя |
| username | public | String | Имя профиля пользователя |
| phone | public | String | Номер телефона пользователя |
| hidden | public | String | Видимость номера телефона пользователя другим пользователям |
| photoUrl | public | String | Адрес изображения профиля пользователя |
| status | public | String | Статус пользователя |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.4.4.1

| Имя | Модификатор доступа | Тип | Назначение |
|-------------|---------------------|--------|---|
| registrated | public | Any | Отметка времени, когда пользователь зарегистрировался |
| location | public | String | Местоположение пользователя |

Таблица 2.4.4.2 – Описание методов класса UserModel

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|--------|---------------------|---------|-------------|--|
| equals | public | Boolean | other: Any? | Переопределенный метод, который сравнивает профили пользователей по идентификатору |

Таблица 2.5.1 – Описание методов класса AddAdditionalInformationFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------|---------------------|------|--|--|
| initSpinner | private | Unit | - | Установка адаптера и элементов в выпадающий список для выбора категории объявления |
| next | private | Unit | description: String, category: String | Валидация значений, введенных пользователем и переход к следующему фрагменту |

Таблица 2.5.2 – Описание методов класса AddImageFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|----------|---------------------|------|-----------|---|
| addPhoto | private | Unit | - | Запуск активности, в которой осуществляется выбор фотографии из приложений пользователя |
| complete | private | Unit | - | Завершение добавления нового объявления |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.5.3 – Описание методов класса AddMainInformationFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------|---------------------|------|--|---|
| initSpinner | private | Unit | - | Установка адаптера и элементов в выпадающий список для выбора местоположения объявления |
| next | private | Unit | type: String, name: String, location: String | Валидация значений, введенных пользователем и переход к следующему фрагменту |

Таблица 2.6.1.1 – Описание полей и свойств класса AdvertFragment

| Имя | Модификатор доступа | Тип | Назначение |
|-------|---------------------|-----------|---|
| owner | private | UserModel | Пользователь, который добавил данное объявление |

Таблица 2.6.1.2 – Описание методов класса AdvertFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|------------|---------------------|------|----------------------------------|---|
| initAdvert | private | Unit | crossinline function: () -> Unit | Инициализация данными представлений, которые относятся к содержимому объявления, в файле макета одиночного объявления |
| initOwner | private | Unit | crossinline function: () -> Unit | Инициализация данными представлений, которые относятся к автору объявления, в файле макета одиночного объявления |
| call | private | Unit | - | Запуск активности для звонка пользователю, добавившему объявление |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.6.2.1 – Описание полей и свойств класса AdvertsFragment

| Имя | Модификатор доступа | Тип | Назначение |
|------------------|---------------------|--------------------------|---|
| mRecyclerView | private | RecyclerView | Элемент RecyclerView для отображения списка объявлений |
| mAdapter | private | AdvertsAdapter | Адаптер для mRecyclerView |
| mAdvertsListener | private | AppValueEventListener | Слушатель изменений значений в списке объявлений |
| mAdverts | private | MutableList<AdvertModel> | Список объявлений |
| mQueryFilter | private | String | Текст запроса, введенного пользователем в поисковую строку (по умолчанию “”) |
| mTypeFilter | private | String | Тип объявления, выбранный пользователем в фильтрах (по умолчанию “Не важно”) |
| mCategoryFilter | private | String | Категория объявления, выбранная пользователем в фильтрах (по умолчанию “Не важно”) |
| mLocationFilter | private | String | Местоположения объявления, выбранное пользователем в фильтрах (по умолчанию “Не важно”) |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.6.2.2 – Описание методов класса AdvertsFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|------------------|---------------------|---------|-----------------------------|---|
| initRecyclerView | private | Unit | - | Инициализация инструмента RecyclerView для отображения списка объявлений |
| initSearchBar | private | Unit | - | Инициализация поисковой строки |
| filter | private | Unit | - | Фильтрация списка объявлений по параметрам mTypeFilter, mCategoryFilter, mLocationFilter и нечеткий поиск по параметру mQueryFilter |
| search | private | Boolean | name: String, query: String | Неточный поиск запроса в названии объявления с помощью алгоритма Bitap |
| chooseFilters | private | Unit | - | Сброс или установка пользователем значений фильтров и фильтрация списка объявлений по соответствующим значениям |

Таблица 2.7.1 – Описание полей и свойств класса ChatFragment

| Имя | Модификатор доступа | Тип | Назначение |
|----------------------|---------------------|-----------------------|--|
| mChatToolbar | private | View | Верхняя панель чата |
| mUserToolbarListener | private | AppValueEventListener | Слушатель изменений данных профиля пользователя, который является собеседником для отображения в верхней панели чата |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.7.1

| Имя | Модификатор доступа | Тип | Назначение |
|-------------------------|---------------------|-----------------------|--|
| mAdvertToolbarListener | private | AppValueEventListener | Слушатель изменений данных объявления для отображения в верхней панели чата |
| mSwipeRefreshLayout | private | SwipeRefreshLayout | Элемент SwipeRefreshLayout для обновления элементов в списке сообщений |
| mRecyclerView | private | RecyclerView | Элемент RecyclerView для отображения списка сообщений |
| mLayoutManager | private | LinearLayoutManager | Элемент LinearLayout Manager для разметки макетов элементов списка сообщений |
| mAdapter | private | ChatAdapter | Адаптер для mRecyclerView |
| mMessagesListener | private | AppChildEventListener | Слушатель изменений значений в списке сообщений |
| mCountMessages | private | Int | Количество сообщений, загружаемых из базы данных |
| mIsScrolling | private | Boolean | Прокручивает ли в настоящее время пользователь список сообщений |
| mSmoothScrollToPosition | private | Boolean | Перемещать ли позицию отображаемого сообщения на последнюю |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.7.2 – Описание методов класса ChatFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|------------------|---------------------|------|-----------|---|
| init | private | Unit | - | Инициализация полей класса |
| attachImage | private | Unit | - | Запуск активности, в которой осуществляется выбор фотографии из приложений пользователя |
| initToolbar | private | Unit | - | Инициализация верхней панели чата |
| initRecyclerView | private | Unit | - | Инициализация инструмента RecyclerView для отображения списка сообщений |
| updateData | private | Unit | - | Дозагрузка более старых сообщений при прокручивании пользователем списка сообщений вниз |

Таблица 2.8.1 – Описание полей и свойств класса ChatAdapter

| Имя | Модификатор доступа | Тип | Назначение |
|----------------|---------------------|----------------------------|---|
| mMessagesCache | private | MutableList<MessageView> | Список объектов класса, которые содержат данные представлений в файле макета сообщения для mCountMessages сообщений, загруженных с базы данных |
| mHolders | private | MutableList<MessageHolder> | Список объектов класса, которые хранят ссылки на представления в файле макета сообщения для mCountMessages сообщений, загруженных с базы данных |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.8.2 – Описание методов класса ChatAdapter

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|--------------------|---------------------|------|--|--|
| attachItemToBottom | public | Unit | item: MessageView, onSuccess: () -> Unit | Функция расширения, которая добавляет новое загруженное сообщение в список сообщений в конец при отправке нового сообщения |
| attachItemToTop | public | Unit | item: MessageView, onSuccess: () -> Unit | Функция расширения, которая добавляет новое загруженное сообщение в список сообщений в начало при прокрутке списка сообщений вверх |

Таблица 2.9.1 – Описание методов класса HolderFactory

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-----------|---------------------|-------------------------|----------------------------------|--|
| getHolder | public | RecyclerView.ViewHolder | parent: ViewGroup, viewType: Int | Создает объект одного из классов, которые хранят ссылки на представления в файле макета сообщения, в зависимости от типа сообщения |

Таблица 2.9.2.1 – Описание полей и свойств класса ImageMessageHolder

| Имя | Модификатор доступа | Тип | Назначение |
|-------------------------|---------------------|------------------|---|
| inboundMessageContainer | private | ConstraintLayout | Контейнер исходящего сообщения |
| inboundMessage | private | ImageView | Элемент ImageView для отображения изображения в исходящем сообщении |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.9.2.1

| Имя | Модификатор доступа | Тип | Назначение |
|--------------------------|---------------------|------------------|---|
| inboundMessageDateTime | private | TextView | Элемент TextView для отображения времени и даты отправки сообщения для исходящего сообщения |
| outboundMessageContainer | private | ConstraintLayout | Контейнер входящего сообщения |
| outboundMessage | private | ImageView | Элемент ImageView для отображения изображения во входящем сообщении |
| outboundMessageDateTime | private | TextView | Элемент TextView для отображения времени и даты отправки сообщения для входящего сообщения |

Таблица 2.9.2.2 – Описание методов класса ImageMessageHolder

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------|---------------------|------|-------------------|--|
| drawMessage | public | Unit | view: MessageView | Инициализация представлений в файле макета сообщения, которое представляет собой изображение |

Таблица 2.9.3.1 – Описание полей и свойств класса TextMessageHolder

| Имя | Модификатор доступа | Тип | Назначение |
|-------------------------|---------------------|------------------|--|
| inboundMessageContainer | private | ConstraintLayout | Контейнер исходящего сообщения |
| inboundMessage | private | TextView | Элемент TextView для отображения текста исходящего сообщения |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.9.3.1

| Имя | Модификатор доступа | Тип | Назначение |
|--------------------------|---------------------|------------------|---|
| inboundMessageDateTime | private | TextView | Элемент TextView для отображения времени и даты отправки сообщения для исходящего сообщения |
| outboundMessageContainer | private | ConstraintLayout | Контейнер входящего сообщения |
| outboundMessage | private | TextView | Элемент TextView для отображения текста входящего сообщения |
| outboundMessageDateTime | private | TextView | Элемент TextView для отображения времени и даты отправки сообщения для входящего сообщения |

Таблица 2.9.3.2 – Описание методов класса TextMessageHolder

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------|---------------------|------|-------------------|---|
| drawMessage | public | Unit | view: MessageView | Инициализация представлений в файле макета текстового сообщения |

Таблица 2.10.1.1 – Описание полей и свойств класса ImageMessageView

| Имя | Модификатор доступа | Тип | Назначение |
|------|---------------------|--------|---|
| id | public | String | Идентификатор сообщения |
| text | public | String | Текст сообщения (для сообщения, представляющего изображение, инициализировано “”) |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.10.1.1

| Имя | Модификатор доступа | Тип | Назначение |
|---------|---------------------|--------|--|
| from | public | String | Идентификатор пользователя, который отправил сообщение |
| sended | public | Any | Отметка времени, когда было отправлено сообщение |
| fileUrl | public | String | Адрес изображения, содержащегося в сообщении |

Таблица 2.10.1.2 – Описание методов класса ImageMessageView

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------|---------------------|---------|-------------|--|
| getTypeView | public | Int | - | Возвращает целочисленную константу, определяющую тип сообщения как сообщение, которое представляет изображение |
| equals | public | Boolean | other: Any? | Переопределенный метод, который сравнивает сообщения по идентификатору |

Таблица 2.10.2.1 – Описание полей и свойств класса TextMessageView

| Имя | Модификатор доступа | Тип | Назначение |
|--------|---------------------|--------|--|
| id | public | String | Идентификатор сообщения |
| text | public | String | Текст сообщения |
| from | public | String | Идентификатор пользователя, который отправил сообщение |
| sended | public | Any | Отметка времени, когда было отправлено сообщение |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.10.2.1

| Имя | Модификатор доступа | Тип | Назначение |
|---------|---------------------|--------|--|
| fileUrl | public | String | Адрес изображения, содержащегося в сообщении (для текстового сообщения инициализирован "") |

Таблица 2.10.2.2 – Описание методов класса TextMessageView

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------|---------------------|---------|-------------|--|
| getTypeView | public | Int | - | Возвращает целочисленную константу, определяющую тип сообщения как текстовое сообщение |
| equals | public | Boolean | other: Any? | Переопределенный метод, который сравнивает сообщения по идентификатору |

Таблица 2.10.3 – Описание методов класса ViewFactory

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|---------|---------------------|-------------|-----------------------|---|
| getView | public | MessageView | message: MessageModel | Создает объект одного из классов, которые содержат данные представлений в файле макета сообщения, в зависимости от типа сообщения |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.11.1 – Описание полей и свойств класса ChatsFragment

| Имя | Модификатор доступа | Тип | Назначение |
|---------------|---------------------|-------------------|---|
| mRecyclerView | private | RecyclerView | Элемент RecyclerView для отображения списка чатов |
| mAdapter | private | ChatsAdapter | Адаптер для mRecyclerView |
| mChats | private | List<CommonModel> | Список чатов |

Таблица 2.11.2 – Описание методов класса ChatsFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|------------------|---------------------|------|-----------|---|
| initRecyclerView | private | Unit | - | Инициализация инструмента RecyclerView для отображения списка чатов |

Таблица 2.12.1 – Описание полей и свойств класса ProfileFragment

| Имя | Модификатор доступа | Тип | Назначение |
|---------------|---------------------|---|---|
| mRecyclerView | private | RecyclerView | Элемент RecyclerView для отображения списка собственных объявлений профиля пользователя |
| mAdapter | private | FirestoreRecyclerViewAdapter <AdvertModel, AdvertHolder> | Адаптер для mRecyclerView |

Таблица 2.12.2 – Описание методов класса ProfileFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|------------------|---------------------|------|-----------|---|
| initRecyclerView | private | Unit | - | Инициализация инструмента RecyclerView для отображения списка собственных объявлений профиля пользователя |
| initProfile | private | Unit | - | Инициализация данными представлений в файле макета профиля пользователя |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.12.2

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-----------------|---------------------|------|-----------|---|
| changeUserPhoto | private | Unit | - | Запуск активности, в которой осуществляется выбор фотографии из приложений пользователя для установки нового изображения профиля пользователя |

Таблица 2.13.1 – Описание методов класса EnterCodeFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|------------|---------------------|------|--------------|---|
| verifyCode | private | Unit | code: String | Верификация кода для успешного завершения аутентификации пользователя с помощью Firebase authentication API |

Таблица 2.13.2 – Описание методов класса EnterLocationFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|----------------------|---------------------|------|------------------|---|
| initSpinner | private | Unit | - | Установка адаптера и элементов в выпадающий список для выбора местоположения пользователя |
| completeRegistration | private | Unit | location: String | Завершение регистрации нового профиля пользователя |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.13.3.1 – Описание полей и свойств класса EnterPhoneFragment

| Имя | Модификатор доступа | Тип | Назначение |
|-----------|---------------------|---|---|
| mPhone | private | String | Номер телефона, введенный пользователем |
| mCallback | private | PhoneAuthProvider. OnVerificationStateChangedCallbacks | Инструмент Firebase authentication API для отправки и верификации кода по номеру телефона |

Таблица 2.13.3.2 – Описание методов класса EnterPhoneFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|------------------|---------------------|------|-----------|--|
| sendCode | private | Unit | - | Проверка введенного номера на количество символов, при успешном результате проверки – отправка кода по номеру телефона |
| authenticateUser | private | Unit | - | Отправка и верификация кода по номеру телефона mPhone с помощью mCallback |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Таблица 2.13.4 – Описание методов класса EnterUsernameFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|----------------------|---------------------|------|---------------------|--|
| validateUsername | private | Unit | username: String | Проверка введенного имени профиля на не пустоту и отсутствие нецензурных слов, при успешном результате проверки – переход к следующему фрагменту для продолжения регистрации нового профиля пользователя |
| completeRegistration | private | Unit | location: String | Завершение регистрации нового профиля пользователя |

Таблица 2.14 – Описание методов класса SettingsFragment

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|-------------|---------------------|------|--|---|
| initSpinner | private | Unit | - | Установка адаптера и элементов в выпадающий список для выбора местоположения пользователя |
| change | private | Unit | username: String, location: String, hidden: Boolean | Проверка введенного имени профиля на не пустоту и отсутствие нецензурных слов, при успешном результате проверки – сохранение изменений имени, местоположения профиля и видимости номера телефона для других пользователей в базе данных |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

Продолжение таблицы 2.14

| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
|------------|----------------------------|------------|------------------|-------------------------------|
| exit | private | Unit | - | Выход из профиля пользователя |

| | | | | |
|-------------------------|--------------|--------------|--------------|--------------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |
| RU.17701729.06.02-01 81 | | | | |
| Инв. № подл. | Подп. и дата | Взам. Инв. № | Инв. № дубл. | Подп. и дата |

[illegible]