

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Институт компьютерных и инженерных наук
Кафедра информационных и управляющих систем
Направление подготовки 09.03.02 – Информационные системы и технологии

ДОПУСТИТЬ К ЗАЩИТЕ
Зав. кафедрой

_____ А.В. Бушманов
«_____» _____ 2024 г.

БАКАЛАВРСКАЯ РАБОТА

на тему: Разработка Android приложения для клиентов ООО «Кабельные системы»

Исполнитель

студент группы 0104-об

(подпись, дата)

А.С. Калашников

Руководитель

доцент, канд.техн.наук

(подпись, дата)

Т.А. Галаган

Консультант

по безопасности и
экологичности

доцент, канд.техн.наук

(подпись, дата)

А.Б. Булгаков

Нормоконтроль

инженер кафедры

(подпись, дата)

В.Н. Адаменко

Благовещенск 2024

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
профессионального образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВПО «АмГУ»)

Факультет математики и информатики
Кафедра информационных и управляющих систем

УТВЕРЖДАЮ
Зав. Кафедрой
_____ А.В. Бушманов
«___» _____

З А Д А Н И Е

К выпускной квалификационной работе студента 0104-об группы Калашникова
Андрея Сергеевича

1. Тема выпускной квалификационной работы: Разработка Android приложения
для клиентов ООО «Кабельные системы».

(утверждена приказом от 03.04.2024 № 890)

2. Срок сдачи студентом законченной работы: 14.06.2024

3. Исходные данные к выпускной квалификационной работе: предметная об-
ласть, перечень литературных источников.

4. Содержание выпускной квалификационной работы (перечень вопросов, под-
лежащих разработке): предпроектное исследование; инфологическое проектиро-
вание; разработка программного продукта.

5. Перечень материалов приложения: техническое задание

6. Консультант по выпускной квалификационной работе по безопасности и эко-
логичности – Булгаков А.Б., доцент, кандидат технических наук.

7. Дата выдачи задания: 2.10.2023

Руководитель выпускной квалификационной работы: Галаган Татьяна Алексе-
евна, доцент, кандидат технических наук.

Задание принял к исполнению: 02.10.2023 _____

РЕФЕРАТ

Выпускная квалификационная работа содержит 66 страниц, 31 рисунок, 20 источников, 1 таблица, 1 приложение.

ANDROID ПРИЛОЖЕНИЕ, БАЗА ДАННЫХ, ИНТЕГРИРОВАННАЯ СРЕДА РАЗРАБОТКИ, ДРУЖЕСТВЕННЫЙ ИНТЕРФЕЙС

В работе исследованы Android приложения, способы их проектирования, разработки и тестирования.

Цель работы: разработать удобное и функциональное Android приложение для клиентов ООО "Кабельные системы", которое позволит им быстро и удобно получать информацию о продукции организации, делать заказы, отслеживать статус выполнения заказов и взаимодействовать с представителями организации.

Исследование "Клиентоориентированные приложения для андроид" имеет практическую значимость, оно может дать разработчику знания и опыт по созданию Android приложений, а также позволит пользователям обеспечить новый взгляд на опыт использования программ для Android.

Этапами разработки приложения являются:

- изучение принципов разработки Android приложений.
- анализ существующих клиентоориентированных Android приложений;
- определение функционала приложения;
- исследование Android приложений для клиентов
- проектирование интерфейса приложения;
- разработка программного продукта при помощи IDE «Android Studio» и СУБД «Firebase Realtime Database»;
- функциональное тестирование Android приложения.

СОДЕРЖАНИЕ

Введение	6
1 Предпроектное исследование	8
1.1 Обоснование актуальности темы квалификационного исследования	12
1.2 Анализ Android приложений для клиентов	13
1.3 Обзор существующих методов решения рассматриваемой задачи	16
1.3.1 Приложение «Мой Ростелеком»	16
1.3.2 Приложение «Мой МТС»	18
1.3.3 Приложение «Yota»	19
1.4 Основные принципы разработки Android приложений	22
1.5 Формулировка задачи исследования и общей методики её решения	22
2 Проектирование Android приложения	24
2.1 Функции разрабатываемого приложения и функциональные подсистемы	24
2.2 Проектирование базы данных	27
2.3 Проектирование интерфейса приложения	30
2.4 Описание основных алгоритмов	32
3 Разработка программного продукта	34
3.1 Обоснование и выбор средств разработки	34
3.2 Написание кода и создание интерфейса	36
3.3 Тестирование программного продукта	43
4 Безопасность и экологичность	47
4.1 Безопасность	47
4.2 Экологичность	50
4.3 Чрезвычайные ситуации	51
Заключение	57
Библиографический список	58
Приложение А	60

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ

ООО – общество с ограниченной ответственностью;
ОС – операционная система;
СУБД – система управления базой данных;
БД – база данных;
ИТ – информационные технологии;
GPON – гигабитная пассивная оптическая сеть;
FTTx – оптическое волокно до точки X;
ADSL – асимметричная цифровая абонентская линия;
DOCSIS – спецификация обмена данными по коаксиальному кабелю;
ONT – терминал оптической сети;
Wi-Fi – технология беспроводной локальной сети;
ТВ – телевиденье;
ПО – программное обеспечение;
ПК – персональный компьютер;
ПАО – публичное акционерное общество;
ГОСТ – государственный стандарт;
IDE – интегрированная среда разработки;
GPS – глобальная навигационная спутниковая система.

ВВЕДЕНИЕ

Клиентоориентированные приложения для Android имеют огромную актуальность в современном мире технологий и бизнеса. С течением времени все больше людей обращаются к мобильным приложениям для удовлетворения своих потребностей и решения различных задач. Android, как самая популярная операционная система мобильных устройств, обладает огромной конкурентоспособностью и быстро растущим рынком приложений.

С увеличением числа пользователей смартфонов и планшетов, работающих на Android, увеличилось и количество загрузок приложений из «Google Play».

Ключевые моменты, характеризующие рост использования Android приложений:

- расширение функциональности, приложения становятся все более многофункциональными, предлагая пользователям широкий спектр услуг и возможностей;

- увеличение доступности, снижение стоимости устройств Android способствует увеличению их доступности в разных регионах мира, что в свою очередь ведет к росту использования приложений;

- интеграция с другими устройствами, Android приложения интегрируются с широким спектром устройств, от часов до автомобилей, что увеличивает их привлекательность и удобство использования;

- развитие технологий, прогресс в области искусственного интеллекта, машинного обучения и больших данных позволяет создавать все более умные и персонализированные приложения.

Эти факторы в совокупности способствуют увеличению количества загрузок и времени, проведенного пользователями в приложениях Android.

Целью данной работы является написание программы под ОС Android, которая сможет упростить взаимоотношение между организацией и клиентами путём создания простого и удобного интерфейса с набором функций, обеспечивающих удобный доступ ко всем услугам организации.

Для достижения поставленной цели необходимо решить следующие задачи:

- спроектировать Android приложение;
- создать простой и удобный интерфейс для пользователей;
- спроектировать БД для работы приложения;
- настроить запроса данных из БД организации по сети интернет;
- создать чат внутри приложения для общения с тех. поддержкой;
- оптимизировать приложение для работы с маломощными устройствами.

Новизной в теме разработки Android приложений для клиентов является создание нового приложения для клиентов «Кабельные системы», которого на данное время ещё нет, что позволит облегчить взаимодействие между организацией и клиентами.

Одной из новшеств этой темы является фокус на улучшении опыта пользователя в мобильных приложениях. Разработчик приложения старается создать удобные, интуитивно понятные и привлекательные интерфейсы, чтобы обеспечить гладкую и удобную работу для пользователей. Важным фактором также является персонализация приложений, чтобы пользователи могли настроить их под свои индивидуальные потребности и предпочтения.

Объект исследования: деятельность ООО «Кабельные системы» по работе с клиентами.

Предмет исследования: процесс разработки Android приложения, создание эффективного и удобного программного продукта для пользователей.

В данной выпускной квалификационной работе спроектировано и разработано приложение для клиентов ООО «Кабельные системы». Также была проведена работа по изучению услуг, предоставляемых организацией; изучен рынок Android приложений; изучен язык программирования Java и среда разработки «Android studio»; разработано и протестировано Android приложение для клиентов ООО «Кабельные системы».

1 ПРЕДПРОЕКТНОЕ ИССЛЕДОВАНИЕ

ООО «Кабельные системы» было создано в 2001 году как оператор кабельного телевидения. Благодаря качественному предоставлению услуг, оперативной работе и мягкой ценовой политике ООО «Кабельные системы» стало одним из ведущих кабельных операторов города Благовещенска. За время работы, организация сталкивалась с различными нюансами кабельного и спутникового телевидения, благодаря чему, специалисты смогли достигли высочайшего профессионального уровня. В дальнейшем они стали участвовать в строительстве и обслуживании спутниковых, кабельных и интернет-систем различных уровней сложности как бюджетных, так и коммерческих предприятий. Офис организации представлен на рисунке 1.



Рисунок 1 – Офис ООО «Кабельные системы»

Кабельные Системы использует следующие технологии для предоставления услуг связи клиентам:

– *оптический кабель GPON* – это наиболее продвинутая на сегодняшний день технология предоставления интернета конечным пользователям. Она позволяет использовать интернет на скоростях свыше 100 Мбит/с, и на любых скоростях

обеспечивает очень высокие показатели качества и стабильности соединения. Конечно, всё это зависит от провайдера. Однако, на сегодняшний день это наиболее прогрессивная технология, заметно обходящая классический FTТх, не говоря об ADSL и DOCSIS. Подключение с использованием GPON предполагает установку у клиента абонентского терминала или ONT, который может быть как отдельным от роутера устройством, так и сочетать в себе функции Wi-Fi роутера. Терминал, обычно, включает в себя медиа-конвертер, который преобразует световой сигнал с оптоволоконного кабеля в обычный электрический;

– *витая пара* – классический способ высокоскоростного соединения. В такой компьютерной сети от узла связи до определенного места доходит волоконно-оптический кабель, а далее, до абонента. Такие сети на 2024 г. являются фактическим стандартом, и большинство провайдеров предоставляют доступ в интернет в многоквартирных домах именно через эту технологию. Высокая скорость и стабильность интернета в такой сети гарантируется, если сеть качественно строится и обслуживается самим провайдером;

– спутниковое телевидение предполагает установку у абонента антенны для приёма сигнала со спутника на геостационарной орбите. Преимуществом спутникового ТВ является возможность получить большой набор каналов практически в любом месте земного шара, а недостатками являются слабая интерактивность, то есть невозможность ставить на паузу или перематывать эфир; а также, сильная зависимость качества изображения и его наличие в принципе, от погоды.

Организационная структура организации «Кабельные системы» включает в себя следующие отделы и должности (Рисунок 2):

– генеральный директор отвечает за стратегическое управление предприятием и принятие ключевых решений;

– отдел логистики отвечает за поставки товаров в организации, а также за общие логистические связи;

– финансово-экономический отдел отвечает за работу с финансовыми активами организации;

– ИТ отдел занимается работой ИС предприятия, а также общими работами

в ИТ сфере;

– отдел продаж отвечает за разработку и реализацию маркетинговых стратегий, а также за продажу услуг организации;

– отдел по работе с клиентами отвечает за работу с клиентами, занимается приёмом заявок и заключением договоров между клиентами и организацией;

– отдел кадров отвечает за процессы найма, обучения и оценки персонала, а также за поддержание хорошего рабочего климата;

– отдел производства занимается производственной частью предприятия, отвечает на заявки и выполняет технические работы.

Каждый отдел имеет свои функции и задачи, при этом сотрудники различных отделов сотрудничают и взаимодействуют для достижения общей цели предприятия.

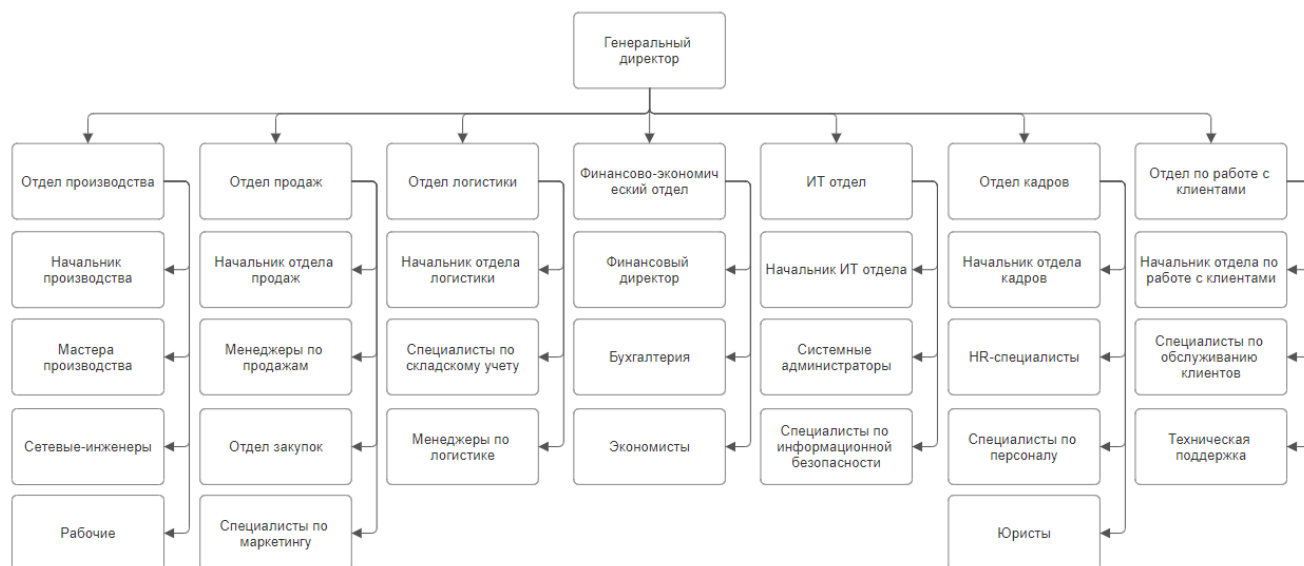


Рисунок 2 – Организационная структура ООО «Кабельные системы»

Документооборот

Процесс документооборота представляет собой перемещение документов в организации, начиная с момента их создания или получения до завершения исполнения или отправления.

Внешний документооборот ООО «Кабельные системы» представлен на рисунке 3.



Рисунок 3 – Внешний документооборот

Внутренний документооборот организации представлен на рисунке 4.

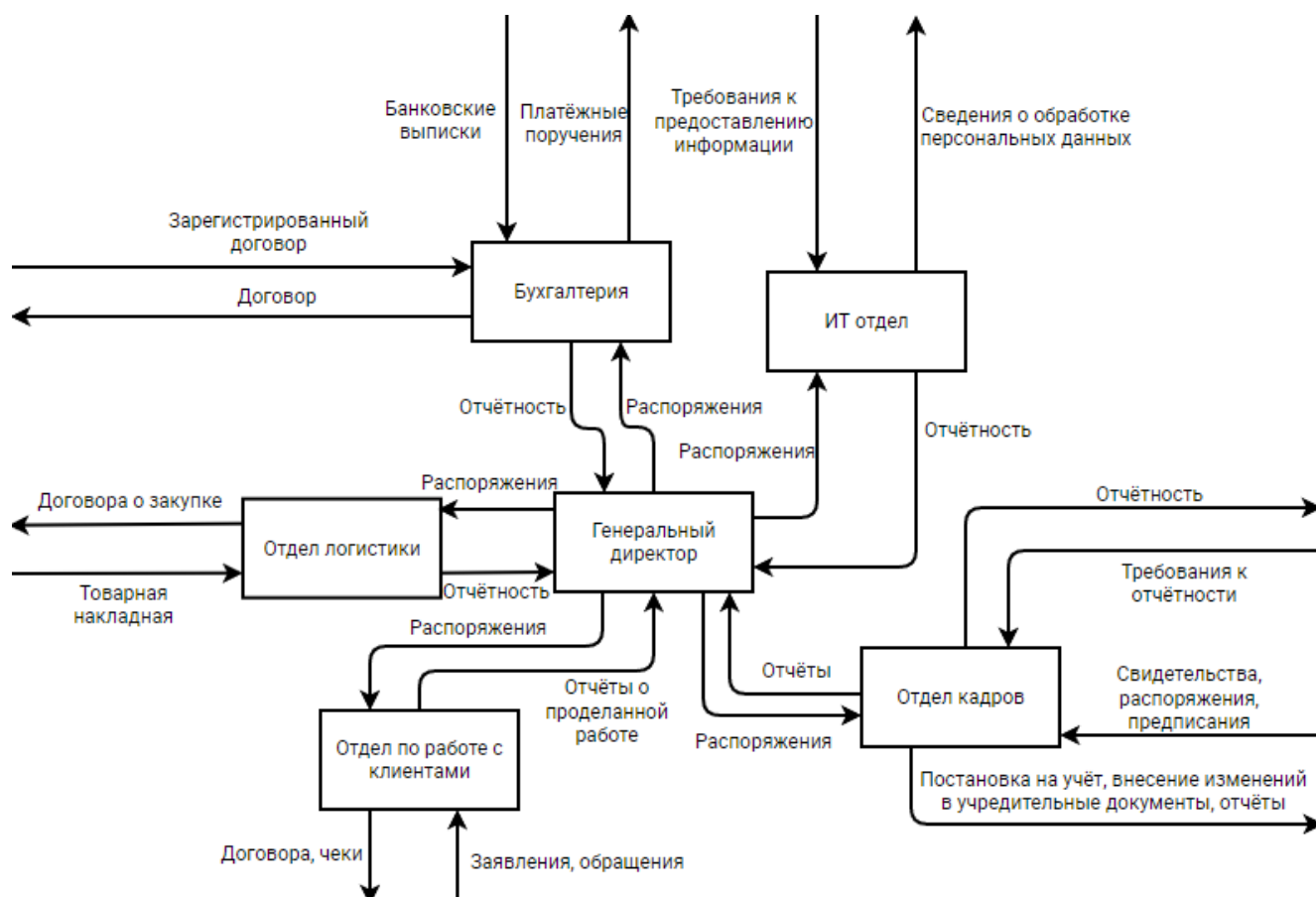


Рисунок 4 – Внутренний документооборот

Исходя из организационной структуры и внутреннего документа оборота можно сделать вывод, что весь документооборот осуществляется через главного директора, который принимает документы с одного отдела, утверждает и подписывает его, и передаёт его в другой.

Внешний документооборот показывает, что организация взаимодействует с множеством внешних сторон, такими как партнеры, клиенты, государственные органы, что соответствует законодательным и отраслевым стандартам.

1.1 Обоснование актуальности темы квалификационного исследования

В современном, постоянно развивающемся, мире существует большой спрос на приложения для работы с услугами и информацией, предоставляемых организациями клиентам. Такие приложения пришли на замену счетам по почте, смс уведомлениям и менее удобным, и эффективным интернет-ресурсам. Поэтому с ростом числа смартфонов растёт и востребованность в удобных и информативных клиентоориентированных приложениях. В наши дни большинство фирм и организаций уже обзавелись специализированными приложениями для клиентов.

Клиентоориентированные Android-приложения остаются актуальными и востребованными по нескольким причинам:

- увеличение числа пользователей смартфонов. С каждым годом все больше людей пользуются мобильными устройствами, и Android остается наиболее популярной операционной системой в этой сфере. Поэтому разработка Android-приложений позволяет достичь максимальной аудитории пользователей;

- удобство использования. Android-устройства просты в использовании, имеют понятный интерфейс и широкий спектр функций, что делает клиентоориентированные приложения привлекательными для пользователей. Мобильные приложения вошли во все сферы жизни людей. От покупок и путешествий до банковского дела и здоровья, приложения играют важную роль в повседневной жизни людей. Таким образом, клиентоориентированные Android-приложения предоставляют удобный и быстрый способ получить доступ к нужным продуктам и услугам;

- развитие электронной коммерции. С каждым годом интернет-торговля становится все популярнее, и клиенты все больше предпочитают совершать покупки

онлайн. Это делает клиентоориентированные приложения для Android неотъемлемой частью бизнес-стратегии многих организаций;

– большое количество инструментов и ресурсов разработки приложений. Android предоставляет разработчикам широкие возможности для создания качественных и интуитивно понятных приложений, Кроме того, существуют множество онлайн-курсов, форумов и сообществ, которые помогают разработчикам получать поддержку и ресурсы для разработки клиентоориентированных приложений.

Исходя из всего вышеперечисленного можно сделать вывод, что Android приложения для клиентов крайне актуальны и востребованы в наши дни. Они позволяют более эффективно взаимодействовать с приобретёнными услугами, контролировать траты, и следить за новостями и новыми предложениями.

1.2 Анализ Android приложений для клиентов

Мобильные телефоны давно перестали быть чем-то необычным и превосходно справляются со своей функцией являются средством коммуникации между людьми. При этом, недавно появившиеся, но уже прочно вошедшие в нашу жизнь смартфоны настолько функциональны, что трудно сказать, чего они не умеют: это и плеер, и фотоаппарат, и возможность использования Интернет-ресурсов, и прочее. По сути, все смартфоны стали небольшой копией компьютера, который постоянно можно иметь при себе. В наше время все больше и больше смартфонов, коммуникаторов, планшетных ПК и других видов устройств, удобных для использования как в повседневной жизни, так и в заграничных поездках, в частности, выпускаются на базе ОС Android. Во-первых, Android поддерживает большое количество устройств разных производителей. Во-вторых, Android характеризуется высокой доступностью средств разработки. Средства разработки для платформы Android бесплатны, в то время как разработка, к примеру, под iPhone требует немалых начальных финансовых вложений. Кроме всего вышеперечисленного, преимуществом ОС Android является наличие бесплатных библиотек для работы со сторонними ресурсами в то время, как для Windows Phone Mobile такие библиотеки не распространены.

Мобильное приложение – это специально разработанное под функциональные возможности гаджетов, программное обеспечение. Назначение ПО. может быть самым разнообразным: сервисы, магазины, развлечения, онлайн-помощники и другое. Эти приложения скачиваются и устанавливаются самим пользователем через мобильные маркетплейсы. Самые крупные площадки – AppStore, Google Play. Технически все приложения создаются под конкретную платформу мобильного гаджета. Наиболее популярные операционные системы – IOS, Android, Windows.

Основные преимущества приложений:

- интерфейс программы создан конкретно под работу на мобильном устройстве через сенсорный экран или кнопки;
- удобная и понятная для пользователей гаджетов навигация, мобильное меню;
- лучшее взаимодействие с пользователем через сообщения, пуш-уведомления, напоминания. Приложение может выполнять функции даже в фоновом режиме, чего нельзя сказать о сайте;
- хранение персональных данных пользователя. Эта функция расширяет возможности персонализации приложений. Например, вызывает такси на дом (прописки), записывает на прием к врачу по медицинскому полису и другие преимущества;
- гибкая обратная связь с организацией, сервисом;
- задействование больших ресурсов. Например, подключение геолокации и вызов машины в любую точку города;
- учёт биологических ритмы человека и оповещение его о необходимости следовать режиму.

Мобильными приложениями для клиентов сегодня обзаводятся и организации, работающие в массовых секторах, и те, кто сосредоточен на корпоративном рынке. Для одних выпуск мобильного приложения для клиентов – дань моде, для других – насущная необходимость, для третьих – освоение новых возможностей для бизнеса.

Android приложения для клиентов обладают рядом характеристик:

– платформа, Android приложения разработаны для операционной системы Android, что позволяет им работать на большинстве мобильных устройств, использующих эту ОС;

– интерфейс, Android приложения обычно имеют интуитивно понятный и привлекательный интерфейс, оптимизированный для мобильных устройств. Они могут включать в себя различные элементы управления, такие как кнопки, поля ввода, списки и т. д.;

– функциональность, Android приложения для клиентов разработаны для решения разных задач. Они могут включать в себя функции, такие как поиск и просмотр товаров или услуг, оформление заказов, отслеживание доставки, общение с представителями организации и т. д.;

– персонализация, Android приложения могут предлагать возможность настройки определенных параметров и предоставления персонализированной информации или рекомендаций пользователям;

– взаимодействие со сторонними сервисами, Android приложения могут использовать функциональность и данные, предоставляемые другими сервисами и платформами, такими как GPS, социальные сети, системы оплаты и т. д.;

– безопасность, Android приложения могут быть защищены паролем, шифрованием данных и другими методами безопасности для защиты личной информации и предотвращения несанкционированного доступа;

– обновления, Android приложения обычно регулярно обновляются разработчиками для исправления ошибок, улучшения функциональности и добавления новых возможностей;

– онлайн и офлайн режимы, Android приложения могут предоставлять функциональность как в режиме онлайн, так и в офлайн режиме, что позволяет пользователям использовать их даже при отсутствии интернет-соединения.

В целом, Android приложения для клиентов предлагают удобный и доступный способ взаимодействия с продуктами, услугами и информацией, предоставля-

емыми организациями. Они помогают упростить процесс покупки, заказа, коммуникации и оплаты для пользователей на мобильных устройствах.

1.3 Обзор существующих методов решения рассматриваемой задачи

В наше время существует множество приложений для Android по работе с клиентами. Такие приложения предназначены для удобного управления контактами, ведения истории взаимодействия с клиентами, планирования задач и встреч, а также для анализа и отслеживания результатов работы с клиентами. Несколько примеров из таких:

1.3.1 Приложение “Мой Ростелеком”

Мобильное приложение «Мой Ростелеком» представляет собой инновационный инструмент для клиентов ПАО «Ростелеком», позволяющий управлять услугами мобильной связи, интернетом, домашними телефонами, телевидением и другими услугами оператора. Приложение дает возможность клиентам легко и удобно проверять свой текущий счет, оплачивать счета онлайн, узнавать о новых акциях и предложениях, а также контролировать текущий тарифный план и его параметры.

Основные функции приложения включают:

- управление счетами, пользователи могут просматривать информацию о своих счетах, проверять баланс, оплачивать счета и создавать автоплатежи;
- управление услугами, клиенты имеют возможность управлять активными услугами, включая подключение и отключение услуг, изменение тарифных планов и настройку дополнительных опций;
- уведомления, приложение предоставляет клиентам уведомления о текущем состоянии счета, новых акциях и предложениях оператора;
- поддержка и обратная связь, пользователи могут обращаться в службу поддержки оператора непосредственно через приложение и отправлять обратную связь.

Преимущества приложения «Мой Ростелеком» включают:

- удобный интерфейс, приложение имеет интуитивно понятный и простой в использовании интерфейс, что делает его доступным для широкой аудитории

пользователей;

– быстрый доступ к информации, клиенты могут мгновенно получать информацию о состоянии своей учетной записи и услугах без необходимости обращения в офис или контакта со службой поддержки;

– контроль расходов, приложение позволяет пользователям легко контролировать свои расходы, оплачивать счета вовремя и изменять услуги в соответствии с ежедневными потребностями;

– персонализация, клиенты могут настраивать приложение в соответствии со своими предпочтениями, получая только необходимую информацию и уведомления.

В целом, приложение «Мой Ростелеком» является незаменимым инструментом для клиентов оператора, предоставляя им удобный способ управления услугами и контроля их расходов.



Рисунок 5 – Страница приложения «Мой Ростелеком» на Play Market

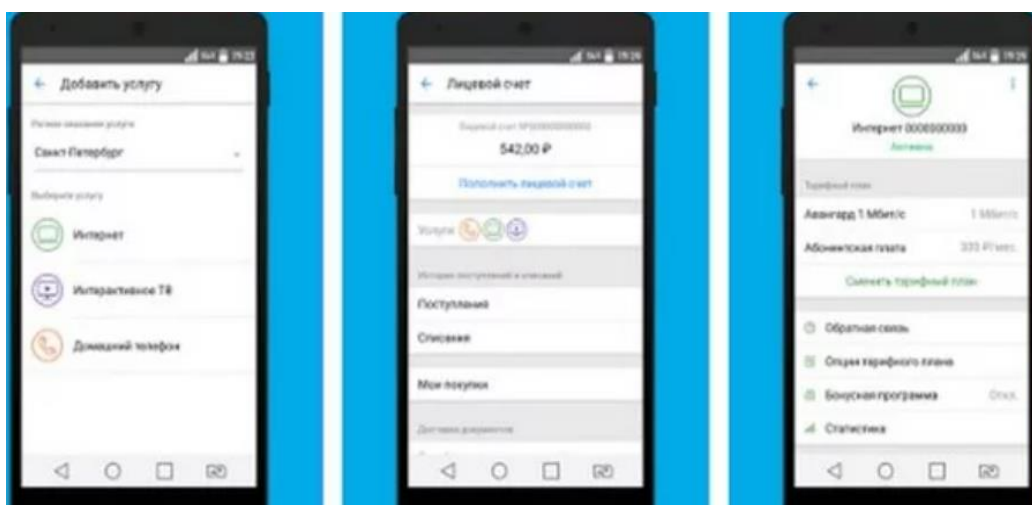


Рисунок 6 – Интерфейс приложения «Мой Ростелеком»

1.3.2 Приложение «Мой МТС»

Мобильное приложение «Мой МТС» предоставляет пользователям сети МТС удобный способ контролировать и управлять своими услугами. Оно объединяет в себе функции управления тарифным планом, проверки баланса, оплаты счетов, покупки дополнительных услуг и многое другое.

Одной из основных характеристик приложения является возможность узнать о своем текущем балансе на счету и остатке минут, сообщений и интернет-трафика. Также в приложении предоставляется доступ к истории расходов, что позволяет пользователям отслеживать свои затраты.

Приложение «Мой МТС» также предлагает удобный способ оплаты счетов и пополнения баланса через банковские карты или электронные кошельки. Пользователи также могут настраивать автоматическое пополнение баланса или устанавливать оповещения о приближающейся дате оплаты.

Дополнительно, через приложение «Мой МТС» пользователи могут изменять тарифный план, активировать и отключать услуги, подключать опции и дополнительные пакеты услуг, а также следить за своими текущими акциями и предложениями.

Пользовательский интерфейс приложения «Мой МТС» очень простой и интуитивно понятный, что делает его удобным для использования даже для тех, кто не имеет специальных технических навыков. Все необходимые функции легко доступны через навигационное меню или главный экран.

В целом, мобильное приложение «Мой МТС» является незаменимым инструментом для пользователей сети МТС, предоставляя им удобный доступ и контроль над их услугами, управление тарифным планом и возможность быстрой оплаты счетов.



Рисунок 7 – Страница приложения «Мой МТС» на Play Market



Рисунок 8 – Интерфейс приложения «Мой МТС»

1.3.3 Приложение «Yota»

Мобильное приложение «Yota» – это приложение, предназначенное для управления услугами оператора связи Yota. Оно позволяет пользователям контролировать свои тарифы и услуги, а также осуществлять настройку связи в соответствии с их потребностями. Вот некоторые функции, которые обычно предлагает такое приложение:

- управление тарифами, пользователи могут выбирать и изменять тарифные планы, а также управлять дополнительными опциями и услугами;
- просмотр расходов, приложение предоставляет информацию о текущем балансе и истории платежей, а также позволяет отслеживать расходы на связь;
- пополнение баланса, с помощью приложения можно легко пополнить баланс счета, используя различные способы оплаты;
- поддержка клиентов, приложение обычно включает функции обращения в службу поддержки и получения помощи по вопросам использования услуг.

Это приложение является удобным инструментом для клиентов Yota, позволяя им управлять своими услугами напрямую с мобильного устройства.

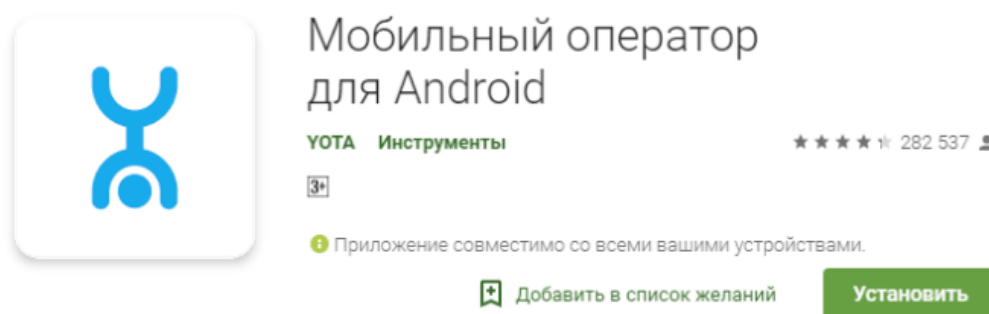


Рисунок 9 – Страница приложения «Yota» на Play Market

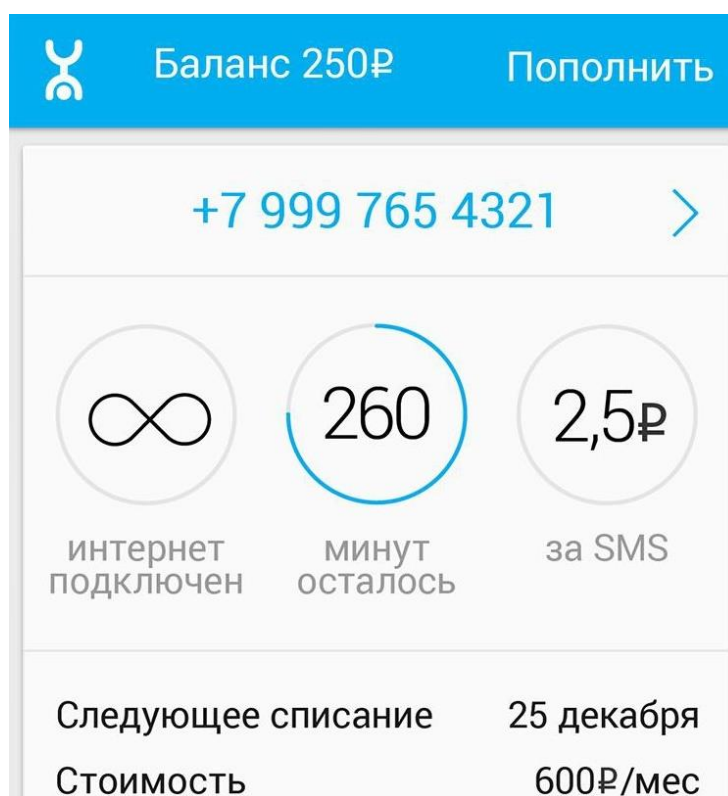


Рисунок 10 – Интерфейс приложения «Yota»

Приведём сравнительную характеристику по вышеуказанным андроид приложениям, чтобы выявить преимущества и недостатки (Таблица 1).

Таблица 1 – Сравнительная характеристика Android приложений

Функции	Мой Ростелеком	Мой МТС	Yota
Управление аккаунтом	Есть	Есть	Есть
Проверка баланса	Есть	Есть	Есть
Пополнение баланса	Есть	Есть	Есть
Подключение услуг	Есть	Есть	Есть
Анализ расходов	Есть	Есть	Есть
Техподдержка	Есть	Есть	Есть
Особенности	Интеграция с другими услугами Ростелекома	Система бонусов и акций	Гибкая настройка тарифов и услуг

Как можно увидеть из таблицы, все три приложения не имеют существенных функциональных недостатков, из этого следует, что данные приложения соответствуют всем стандартам современных клиентоориентированных Android приложений и способны обеспечить все потребности клиентов в необходимых функциях. Также следует отметить, что все эти приложения обладают современным дизайном и дружелюбным интерфейсом, что полностью соответствует всем современным требованиям и стандартам.

К основным преимуществам приложения «Мой Ростелеком» можно отнести интегрированность с другими услугами, предоставляемыми Ростелекомом. Преимуществом «Мой МТС» является большое количество акций, бонусов и предложений, проводимых компанией для поддержания конкурентоспособности и привлечению новых клиентов. «Yota» отличается, уникальной в своём роде, гибкой настройкой тарифов и услуг, где пользователь может детально настроить тариф под свои нужды.

1.4 Основные принципы разработки Android приложений

Для разработки современного Android приложения необходимо:

- использовать языки программирования Java или Kotlin. Эти языки являются официальными языками разработки для Android, что означает полную поддержку операционной системой и другими Android приложениями;
- соблюдать принципы «Material Design» при создании пользовательского интерфейса для Android приложений. Это поможет создать приятный и интуитивно понятный интерфейс для пользователей. Также необходимо разработать «Адаптивный дизайн». Это необходимо для корректной работы приложения на различных устройствах с разными размерами экранов и разрешениями. Например, создать адаптивные макеты интерфейса, которые будут корректно отображаться на разных устройствах;
- оптимизировать приложение, чтобы оно работало быстро и плавно на всех устройствах. Это включает в себя оптимизацию работы с памятью, использование асинхронных операций и другие методы улучшения производительности;
- обеспечить безопасность Android приложения, чтобы защитить данные пользователей и предотвратить уязвимости. Для этого необходимо соблюдать рекомендации ГОСТ по информационной безопасности и использовать соответствующие методы шифрования и аутентификации;
- провести тестирование приложения на разных версиях Android, для возможного выявления и устранения всевозможных ошибок.

1.5 Формулировка задачи исследования и общей методики её решения

Исследование, проводимое в данной работе, рассматривает такие задачи как:

- изучение основных принципов разработки Android приложений;
- анализ существующих Android приложений для клиентов;
- определение функций, необходимых для создания эффективного и удобного Android приложения для клиентов;
- исследование тенденций и популярных функций в сфере Android приложений для клиентов;
- исследование технических возможностей и ограничений при разработке

Android приложений;

- изучение пользы и преимуществ использования Android приложения для клиентов для бизнеса;

- сбор и анализ данных относительно востребованности и использования Android приложений для клиентов;

- разработка дизайна и пользовательского интерфейса для Android приложения, учитывая потребности и предпочтения клиентов;

- разработка программного продукта с использованием интегрированной среды разработки приложений «Android Studio» и СУБД «Firebase Realtime Database»;

- функциональное тестирование разработанного Android приложения по принципу «Белый ящик» и «Черный ящик».

Для решения представленных задач необходимо:

Изучить существующие Android приложения, разобраться в среде создания приложения «Android Studio» и «Firebase Realtime Database». Разработать дизайн и структуру интерфейса приложения. Проработать удобство использования и интуитивную навигацию в приложении. Разработать функциональности приложения в соответствии с заданными требованиями. Включить в приложение функции, которые позволят клиенту легко взаимодействовать с функционалом предоставляемого продукта. Провести тестирование приложения на различных устройствах, для того чтобы убедиться, что все функции работают без ошибок. Исправить любые обнаруженные ошибки или проблемы.

2 ПРОЕКТИРОВАНИЕ ANDROID ПРИЛОЖЕНИЯ

2.1 Функции разрабатываемого приложения и функциональные подсистемы

Для того чтобы спроектировать Android приложение необходимо определить функции приложения и функциональные подсистемы, которые будут включать в себя эти функции.

Необходимые функции:

- регистрация новых пользователей, позволяющая ввести свои данные для создания личного кабинета;
- авторизация пользователей, вход в свой личный кабинет по логину и паролю;
- баланс денежных средств с возможностью пополнения выводит информацию о количестве денежных средств на депозите клиента;
- возможности тарифного плана предоставляет информацию о текущем тарифном плане клиента, и всём спектре услуг, что предоставляет этот тарифный план;
- перечень возможных тарифов выводит информацию о существующих тарифах, предоставляемых организацией;
- даты платежей и сроки следующих, предоставляет информацию о совершённых платежах и сумме будущих;
- просмотр статистических данных позволяет просматривать различные данные в виде графиков, такие как, использование трафика, период активности и другие;
- справочная информация предоставляет справочную информацию о организации;
- пополнение счёта перенаправляет на страницу банка для пополнения лицевого счёта;
- техническая поддержка содержит окно с возможностью обращения в техническую поддержку в режиме реального времени.

Функции авторизации и регистрации объединим в функциональную подсистему «Вход в систему», отвечающую за ввод данных, подключение к системе и вход в личный кабинет.

Функции связанные с выводом информации пользователю объединим в функциональную подсистему «Предоставление информации», отвечающую за предоставление различной информации организацией.

Для более наглядного представления работы системы была построена диаграмма вариантов использования. На данной диаграмме видно, как участники системы «актёры» взаимодействуют между собой при помощи основных функций «прецедентов» (Рисунок 11).

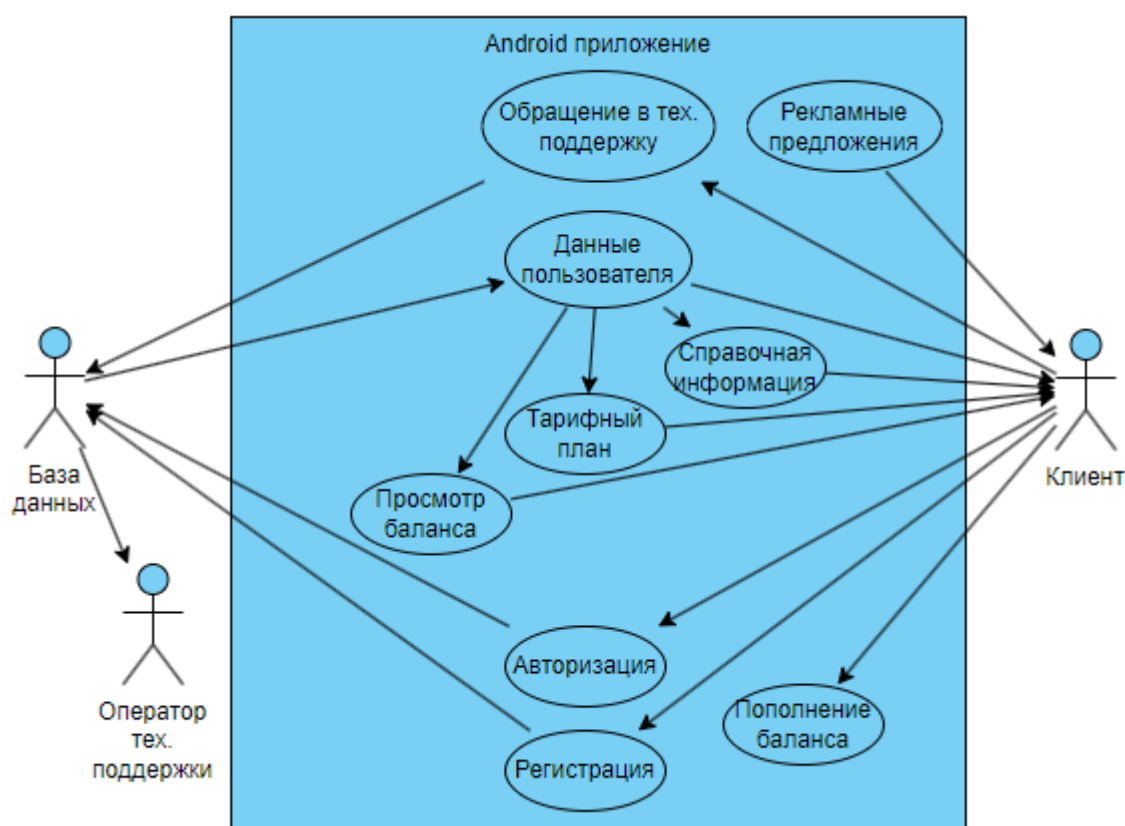


Рисунок 11 – Диаграмма вариантов использования

Также для визуализации последовательности взаимодействия между объектами в системе была построена диаграмма последовательности. На данной диаграмме можно увидеть, как объекта обмениваются сообщениями между собой за время жизненного цикла (Рисунок 12).

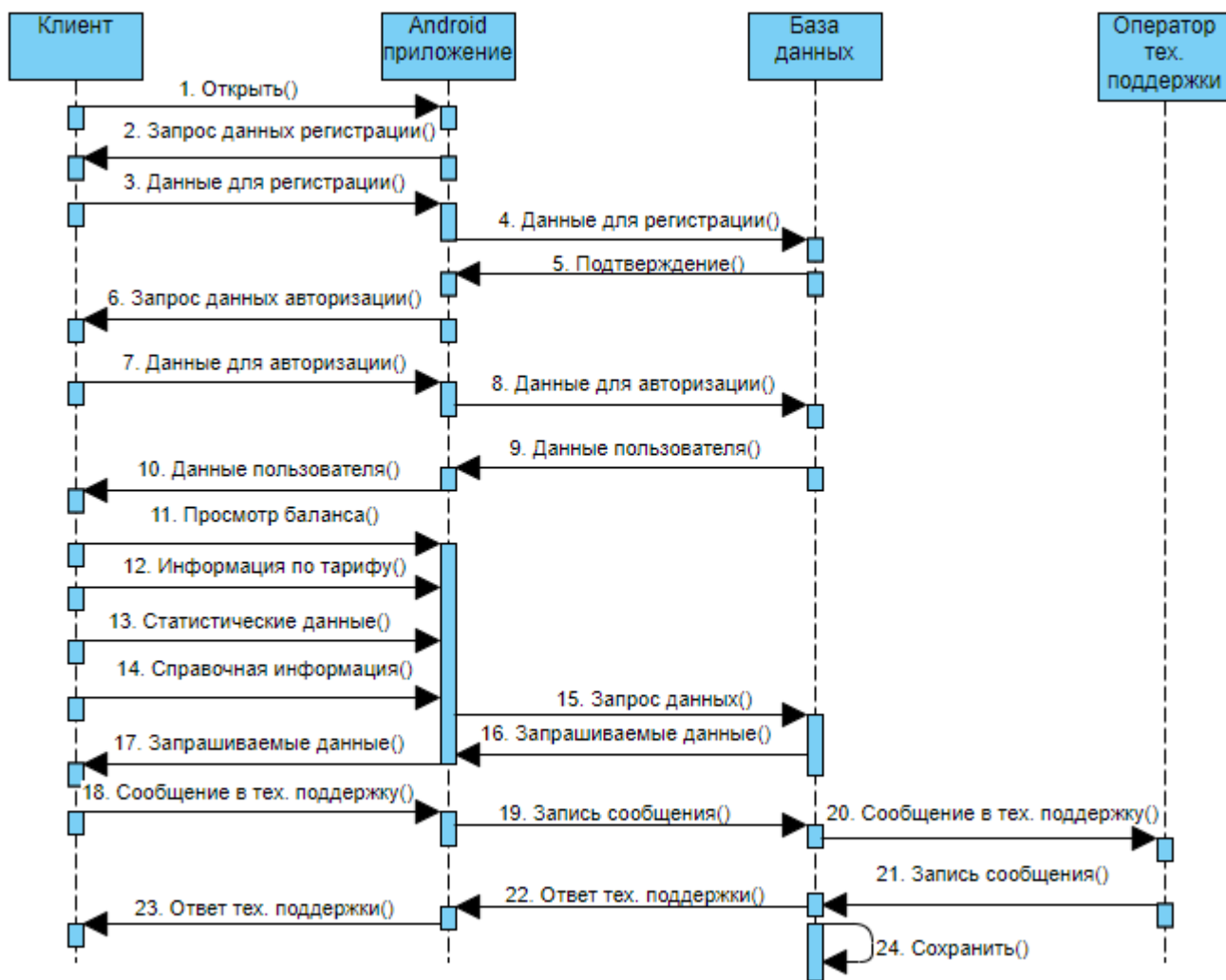


Рисунок 12 – Диаграмма последовательности

Для наглядного представления логики процедур, проводимых в программе, была построена диаграмма деятельности. На ней можно увидеть варианты выбора процессов работы программы, таких как авторизация и регистрация (Рисунок 13).

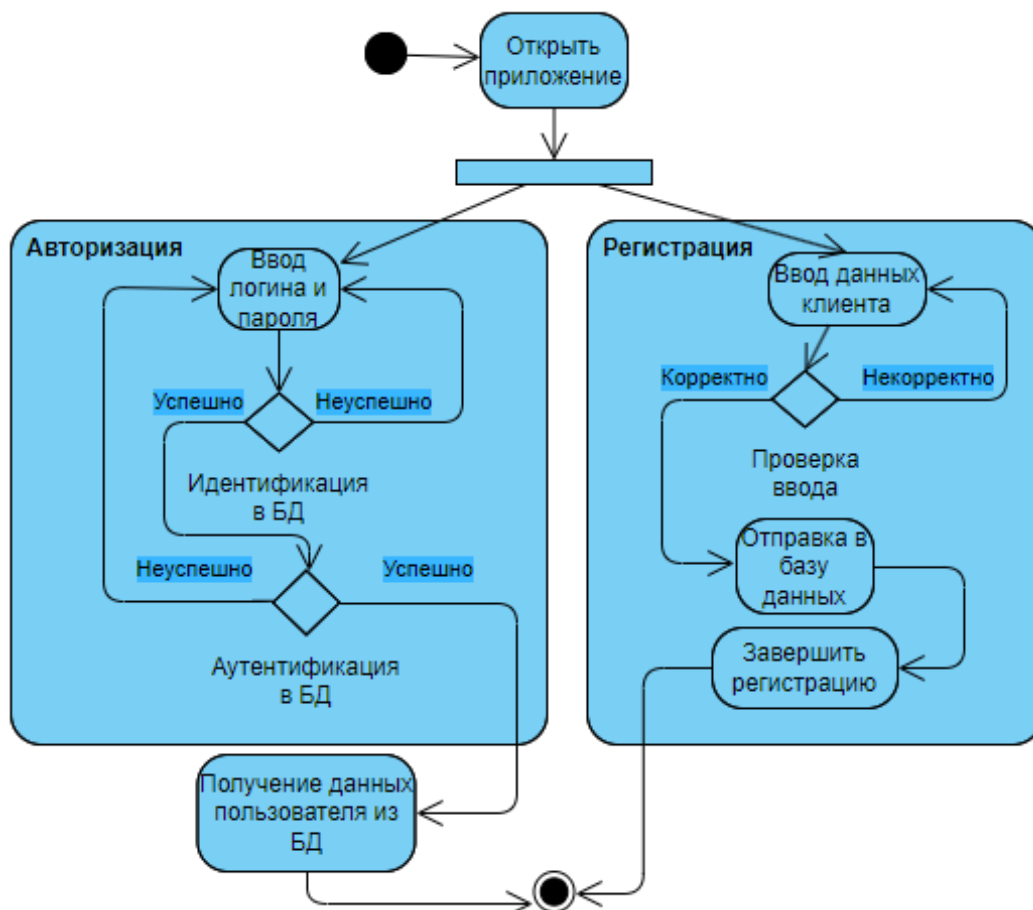


Рисунок 13 – Диаграмма деятельности

2.2 Проектирование базы данных

Для работы приложения спроектирована база данных в «Erwin data modeler» для хранения и передачи информации. В начале сделано «Логическое проектирование», продуманы необходимые сущности и связи между ними, результатом чего является логическая схема.

Логическое проектирование – это этап разработки информационной системы, на котором создается логическая модель базы данных на основе требований и бизнес-логики предприятия. На этом этапе определяются сущности, атрибуты и отношения между ними, которые будут представлены в базе данных. Логическое проектирование позволяет определить структуру данных, их связи и способы доступа к ним. Этот этап разработки помогает установить общую концепцию системы, определить ее основные функциональные возможности и обеспечить эффективность дальнейшей работы над проектом.

Опишем необходимые сущности:

- сущность «Сотрудник» – содержит информацию о сотруднике организации, его персональные данные и идентификационный номер;
- сущность «Услуга» – содержит информацию о предоставляемой услуги сотрудником;
- сущность «Клиент» – содержит информацию о клиенте организации;
- сущность «Личный кабинет» – содержит информацию о текущем тарифном плане, балансе счёта и платежах;
- сущность «Тариф» – содержит информацию о тарифе;
- сущности «Данные», «Трафик», «Период активности» – содержат данные о использовании услуг клиентом.

Разработаем связи для логического взаимодействия между сущностями:

«Услуга» и «Сотрудник» имеют связь «один ко многим». Один сотрудник может предоставлять несколько услуг, но одна услуга предоставляется одним сотрудником.

«Услуга» и «Клиент» имеют связь «многие ко многим». Один клиент может использовать несколько услуг, и одна услуга может использоваться несколькими клиентами. Для упрощения взаимодействия сущностей в будущей БД, разобьём связь «многие ко многим» между Услуга и Клиент на более простые «один ко многим», путём добавления переходной сущности «Услуга / Клиент».

«Клиент» и «Личный кабинет» имеют связь «один к один». У одного клиента может быть только один личный кабинет.

«Личный кабинет» и «Тариф» имеют связь «один ко многим». К личному кабинету может быть подключен только один тариф, один тариф может быть у множества личных кабинетов.

«Личный кабинет» и «Данные» имеют связь «один ко многим». К личному кабинету относятся только одни, определённые, данные, данные могут быть у множества кабинетов.

«Данные» и «Трафик» имеют связь «один ко многим». В одной таблице Данные хранятся множество данных о трафике, данные о трафике хранятся в одной определённой таблице.

«Данные» и «Период активности» имеют связь «один ко многим». В одной таблице Данные хранятся множество данных о периоде активности, данные о периоде активности хранятся в одной определённой таблице.

Все связи сущностей, за исключением «Данные», «Трафик» и «Период активности», имеют «идентифицирующую связь», то есть их первичный ключ входит в состав первичного ключа исходной сущности. Связи между «Данные», «Трафик» и «Период активности» имеют «не идентифицирующую связь», поскольку их первичные ключи входят в состав атрибутов главной сущности. В итоге была получена логическая схема (Рисунок 14).

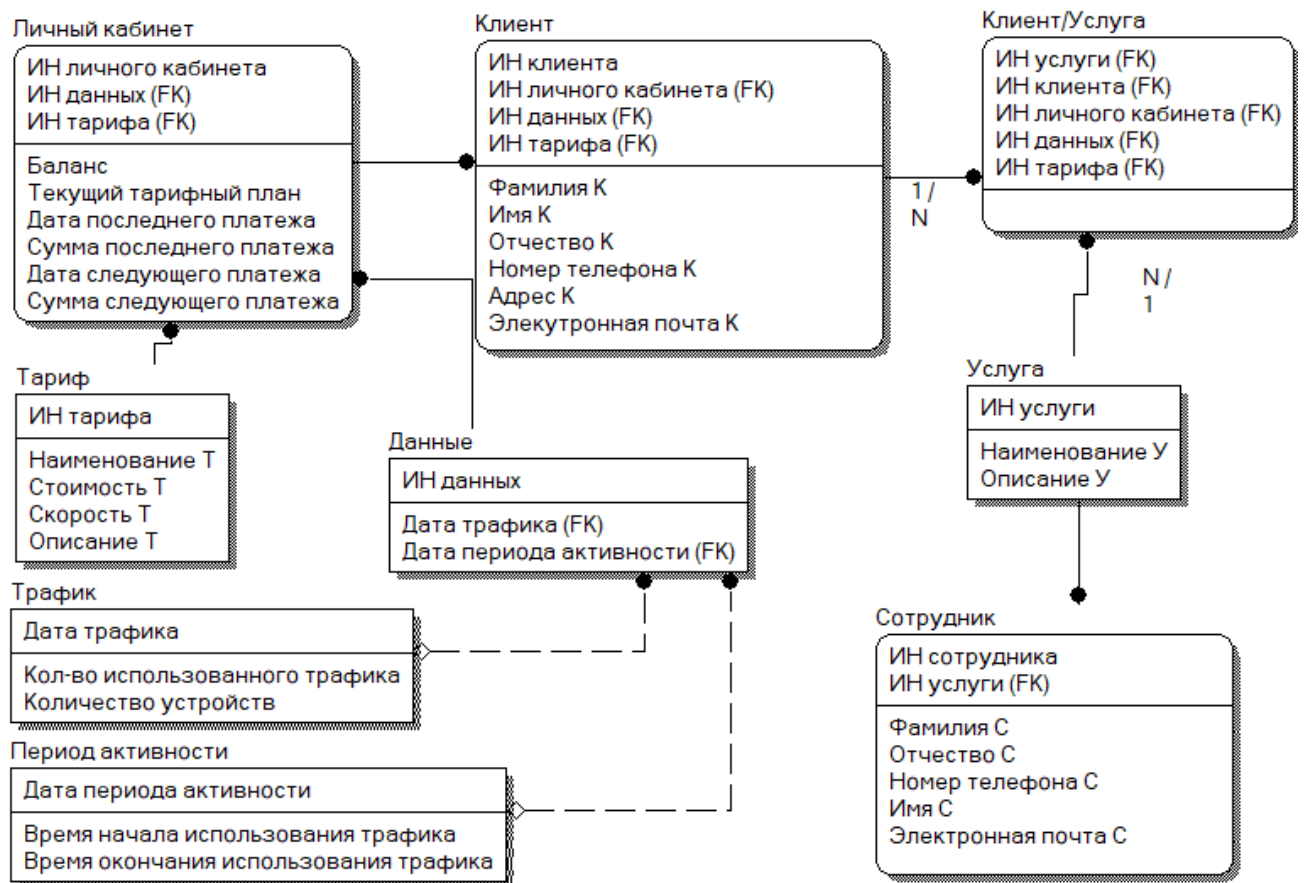


Рисунок 14 – Логическая схема БД

Далее проведено «Физическое проектирование», продуманы и присвоены типы данных для каждой сущности. Физическое проектирование базы данных включает в себя определение конкретной структуры и характеристик для хранения данных в базе данных, а также играет важную роль в создании эффективной и надежной базы данных, которая хранит и обрабатывает данные. На данном этапе

определены, какие типы данных будут использоваться для хранения различных видов информации в базе данных.

Проработаны типы данных для всех атрибутов в каждой сущности:

Идентификационным номерам присвоен тип «Integer», т.е. целочисленный. Также «Integer» присвоен для атрибутов: баланс, стоимость, сумма последнего и следующего платежа, количеству сотрудников. Большинству остальных атрибутов присвоен тип «Text», символьный тип данных, с указанием необходимой допустимой длины. Например атрибут «Номер телефона» сотрудника имеет тип «Text(11)», т. к. номера телефонов состоят из 11 цифр. Атрибутам, где необходимо указание даты или времени присвоен тип «Date/Time», который хранит информацию о дате и времени. Получена готовая физическая схема (Рисунок 15).

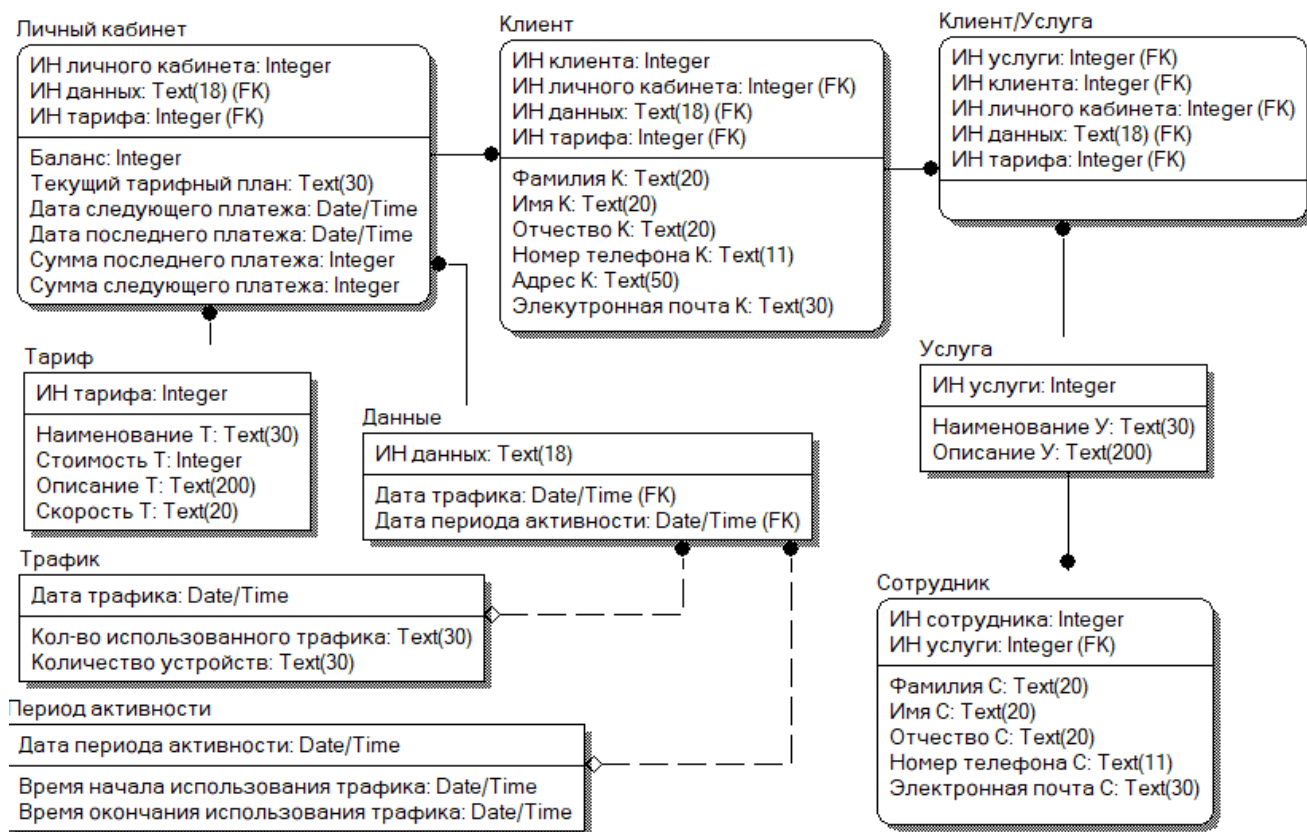


Рисунок 15 – Физическая схема БД

2.3 Проектирование интерфейса

Для разрабатываемого приложения спроектирован макет интерфейса для более удобной разработки. Спроектирована главная форма авторизации, которая встречает пользователя при входе в приложение. В верхней части расположен логотип с названием организации. Под логотипом расположены две кнопки: «Вход»

и «Регистрация». Эти кнопки осуществляют функцию переключателя, кнопка «Регистрация» меняет форму авторизации, то есть заменяет поля логина и пароля на поля регистрации. Кнопка «Вход» возвращает обратно форму авторизации. Под кнопками расположены поля ввода логина и пароля. Под полями ввода расположена кнопка подтверждения данных. И на конец, внизу формы, расположено текстовое поле для вывода текстовой информации. Макет интерфейса спроектирован при помощи графического редактора. Размеры кнопок, полей и окон носят примерный характер, так как, вероятно, в ходе разработки дизайна их будет необходимо менять. В итоге получен готовый макет интерфейса страницы авторизации (Рисунок 16).

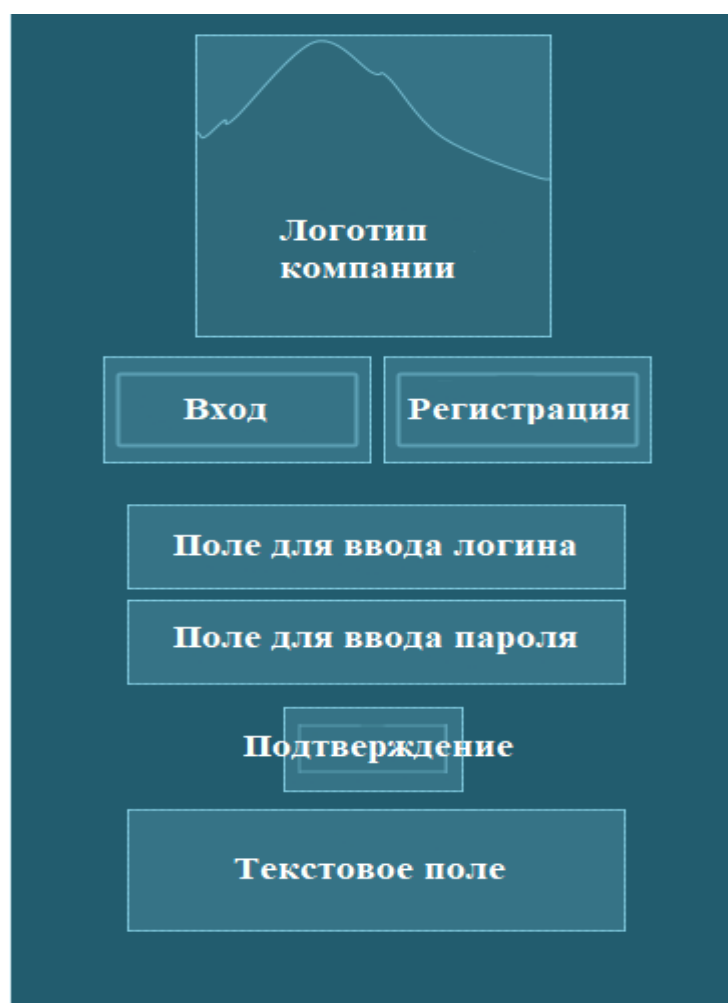


Рисунок 16 – Макет интерфейса авторизации

Далее спроектирован интерфейс личного кабинета, в который попадает пользователь после успешной авторизации.

В верхней части расположен «Слайдер» с рекламой, акциями и предложениями. Ниже расположен «Баланс лицевого счёта» клиента и текущий тарифный

план. Далее расположен «Слайдер» с возможными тарифными планами и краткой информацией о них. Под слайдером с тарифами расположен чат с технической поддержкой и кнопку отправки сообщения. И, наконец, в самом низу расположено навигационное меню с остальными функциями программного продукта. В итоге получен макет личного кабинета (Рисунок 17).

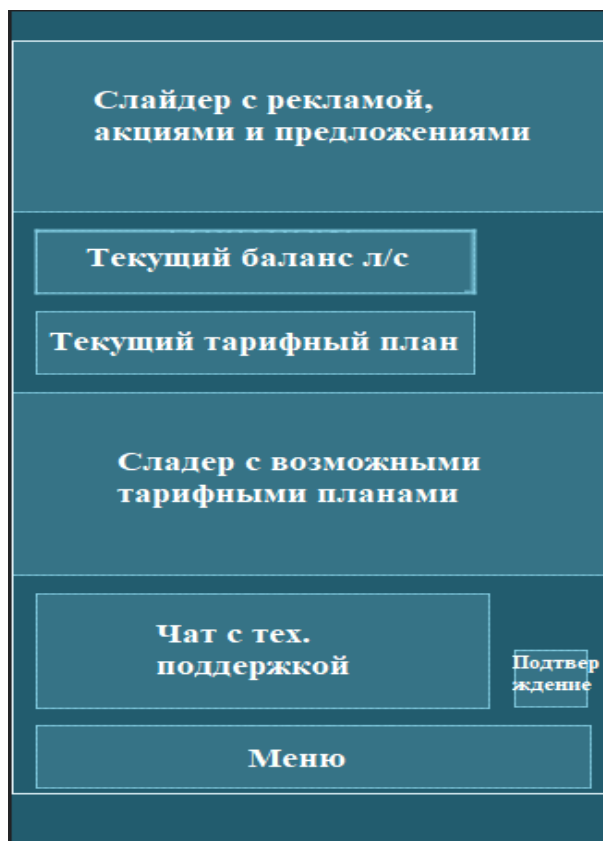


Рисунок 17 – Макет интерфейса личного кабинета

2.4 Описание основных алгоритмов

Опишем основные алгоритмы, которые будут использованы при работе приложения.

Первый необходимый алгоритм – это авторизация пользователей. Приложение получает входные данные от клиента: логин и пароль. Далее проводит процесс идентификации, отправляет логин в базу данных, которая сравнивает его со всеми имеющимися и в случае нахождения сходства начинает процесс аутентификации, в ином случае возвращает сообщение об отсутствии логина в базе данных, которое приложение выводит пользователю, при аутентификации приложение отправляет пароль в базу данных, которая сравнивает его на соответствие с логином. В случае

соответствия база данных отправляет сообщение об успешном входе и предоставляет приложению доступ к имеющимся в системе данным, принадлежащим к данному пользователю, в ином случае отправляет ошибку об несоответствии логина и пароля, приложение выводит пользователю сообщение о неправильном логине или пароле. После успешной аутентификации приложение открывает страницу личного кабинета, запрашивает и выводит на экран данные пользователя, к которым приложение получило доступ.

Второй алгоритм – это регистрация новых пользователей. Пользователь запускает приложение и попадает на стартовое окно авторизации, для перехода на страницу регистрации нажимает на соответствующую кнопку и приложение загружает страницу ввода данных. Приложение получает на входе регистрационные данные от клиента, обрабатывает их на наличие некорректного ввода, и в случае, если обработчик не выявил ошибок, отправляет их в базу данных, в ином случае выводит ошибку ввода.

Третий алгоритм – чат с технической поддержкой. Пользователь в меню личного кабинета нажимает на кнопку «Техническая поддержка», приложение загружает страницу с диалоговым окном, текстовым полем для ввода данных и кнопкой подтверждения. Приложение выводит приветственное сообщение с предложением о помощи. Пользователь вводит текстовое сообщение и нажимает кнопку отправки, приложение отправляет его в базу данных, которая, в свою очередь, создаёт новую подтаблицу. Оператору видит новое сообщение и отвечает своим, которое также отправляет в таблицу. Приложение синхронизирует и выводит все сообщения из созданной подтаблицы, то есть выводит новое сообщение от оператора. Пользователь обменивается сообщениями с работником технической поддержки до тех пор, пока проблема не решена, после чего закрывает подтаблицу, таким образом при повторном обращении клиента в техническую поддержку создастся новая подтаблица с обращением.

3 РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА

3.1 Обоснование и выбор средств разработки

Для создания программного продукта была выбрана интегрированная среда разработки Android приложений – «Android Studio».

«Android Studio» – это интегрированная среда разработки (IDE) для создания приложений под платформу Android. Она предоставляет разработчикам удобную среду для написания кода, отладки, тестирования и развертывания приложений на устройствах под управлением Android.

«Android Studio» основана на платформе «IntelliJ IDEA» и предоставляет большой набор инструментов для удобной и эффективной разработки. В «IDE» есть встроенные средства для создания пользовательского интерфейса, управления проектами, анализа кода, производительности и других функций, необходимых для создания качественных приложений для Android.

«Android Studio» также поддерживает языки программирования Kotlin и Java, а также множество фреймворков и библиотек для Android разработки. Имеется возможность интеграции с Android эмуляторами или физическими устройствами для тестирования приложений в реальных условиях перед их запуском в магазине приложений. Интерфейс «IDE» представлен на рисунке 18.

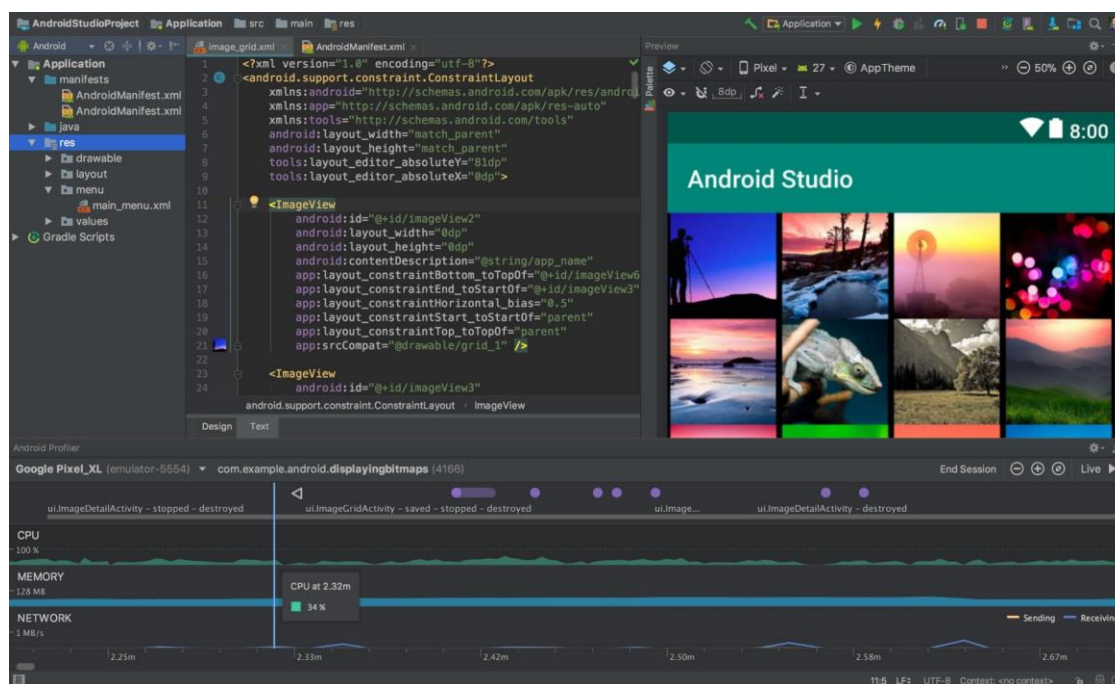


Рисунок 18 – Интерфейс Android Studio

Выбор данной среды разработки обусловлен эффективностью, удобством и широким спектром функций. На данное время это самая распространённая «IDE» для Android в мире, которая содержит широкий инструментарий по созданию Android приложений, большое количество библиотек для реализации различного функционала и множество учебного материала.

Также был выбран «Firebase» для создания и настройки базы данных.

«Firebase» – это платформа разработки приложений от Google, которая предоставляет инструменты и инфраструктуру для упрощения процесса создания, улучшения и роста мобильных и веб-приложений. Она включает в себя множество сервисов и API, которые помогают разработчикам в различных аспектах, включая аутентификацию пользователей, хранение данных, работу с базами данных, аналитику, доставку уведомлений и многое другое.

Одной из ключевых компонентов «Firebase» является «Firebase Realtime Database», которая представляет собой облачную базу данных NoSQL, позволяющую хранить и синхронизировать данные между пользователями в реальном времени. Это означает, что каждый раз, когда данные изменяются, все клиенты, подключенные к базе данных, автоматически получают эти обновления.

Основные преимущества использования Firebase для работы с базами данных:

- синхронизация в реальном времени, данные обновляются в реальном времени у всех пользователей, что идеально подходит для интерактивных приложений;

- масштабируемость, Firebase предлагает гибкость в масштабировании, что позволяет приложениям расти вместе с увеличением пользовательской базы;

- безопасность, платформа предоставляет мощные инструменты для управления доступом и безопасности данных;

- интеграция с другими сервисами Google, Firebase легко интегрируется с другими сервисами Google, такими как Google Analytics, что позволяет разработчикам получать детальную аналитику по своим приложениям.

Интерфейс «Firebase Realtime Database» представлен на рисунке 19.

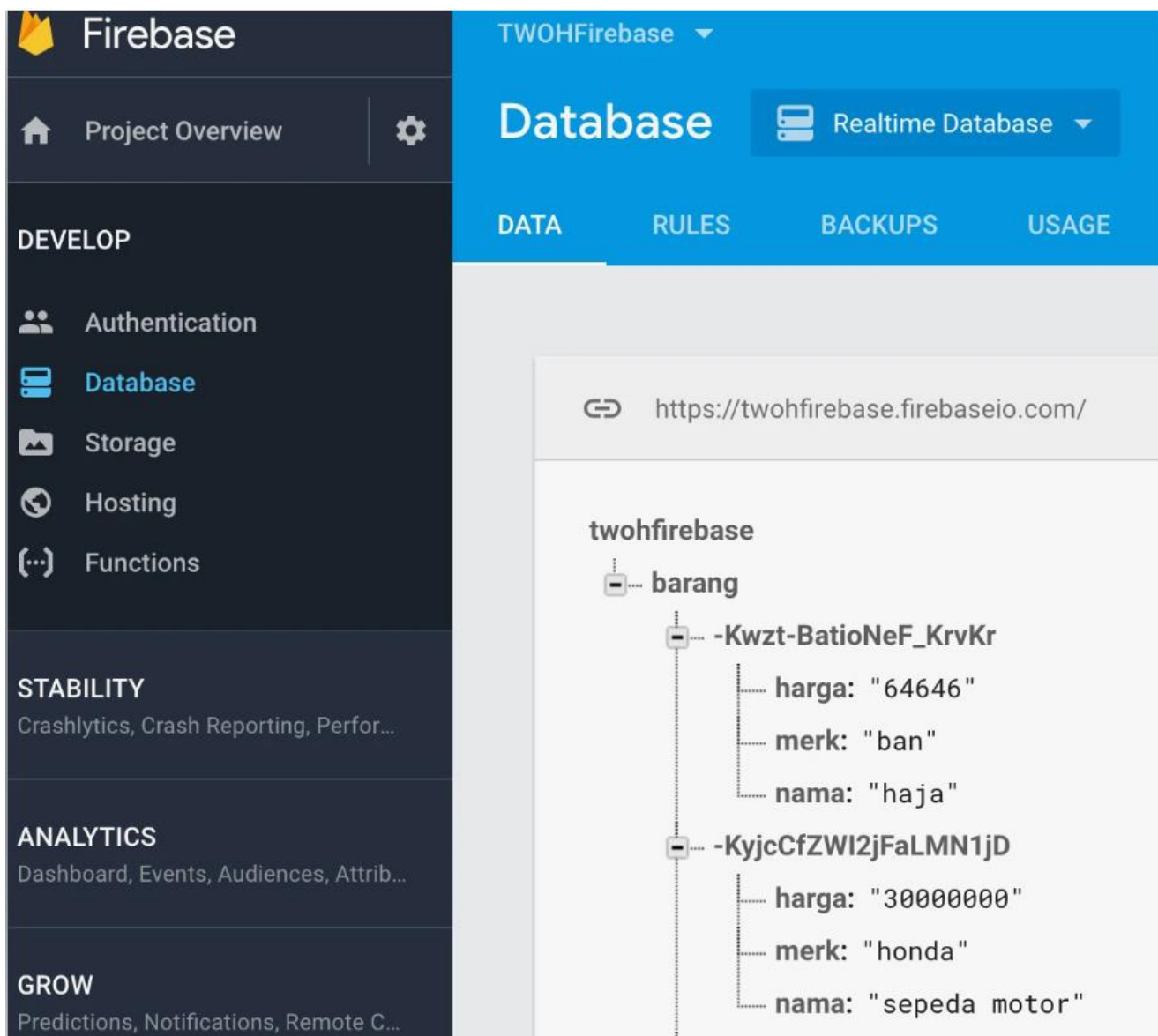


Рисунок 19 – Интерфейс «Firebase Realtime Database»

3.2 Написание кода и создание интерфейса

Разработанный код и созданный интерфейс программы.

Для начала создан новый проект в Android Studio, выбраны стартовые настройки для удобного начала работы. Новый проект назван «Kabel_system». Создана первая страница авторизации, следуя ранее разработанному макету. Установлен логотип организации сверху по центру в виде «шапки» страницы (Рисунок 20).



Рисунок 20 – Верхняя часть страницы

Далее под логотипом установлены две кнопки «Вход» и «регистрация». В кодовой части установлены отступы, размеры, а также в качестве цвета выбран один из основных цветов организации: «Морозное небо».

Под этими кнопками размещены два поля для ввода текста, один для ввода логина, второй для пароля. Кнопка «Вход» запускает алгоритм авторизации и, в случае успеха, перенаправляет на страницу личного кабинета. Кнопка «Регистрация» открывает новую страницу для регистрации (Рисунок 21).

Вход	Регистрация
Ведите логин/email	
Введите пароль	

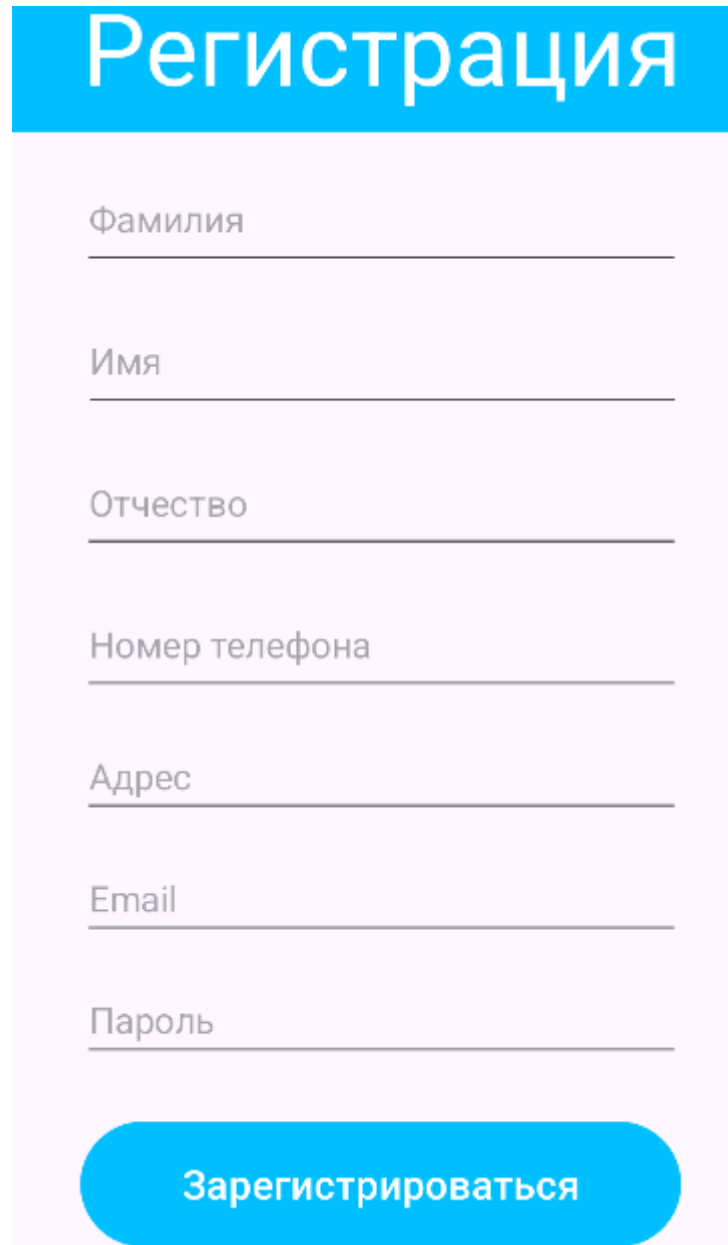
Рисунок 21 – Интерфейс авторизации

В нижней части страницы «шапке» добавлено пустое текстовое поле для вывода текстовых сообщений пользователю, необходимые в качестве подсказки в случае некорректного ввода пользователем, а также выводящие сообщения об ошибках. Также в коде прописан необходимые параметры для всех элементов.

Сделаны последние настройки, добавлена масштабируемость объектам. Это

позволит объектам корректно отображаться на различных устройствах, что позволит избежать многих ошибок на этапе тестирования. Получена готовая заглавная страница авторизации.

После создания страницы авторизации была создана страница регистрации (Рисунок 22).



The image shows a registration form with a blue header containing the word "Регистрация" in white. Below the header, there are seven input fields stacked vertically, each with a light gray label and a horizontal line for text entry. The labels are: "Фамилия", "Имя", "Отчество", "Номер телефона", "Адрес", "Email", and "Пароль". At the bottom of the form is a blue rounded rectangular button with the white text "Зарегистрироваться".

Рисунок 22 – Интерфейс регистрации

Сверху по центру расположено название страницы, ниже расположены семь полей для ввода текста, кнопка и, в конце, поле для вывода текста из приложения. Все элементы расположены столбиком по центру экрана. Семь полей отвечают за вводимые данные: фамилия, имя, отчество, номер телефона, адрес, электронная

почта, пароль. Кнопка отвечает за подтверждение и отправку данных для регистрации. Последнее поле выводит текст с информацией о успешной регистрации, либо о неудаче с указанием на ошибку. В итоге получена форма регистрации.

После создания формы регистрации была создана страница личного кабинета. В «шапке» страницы расположено приветственное сообщение с указанием имени и отчества пользователя для понимания, какой пользователь в данное время авторизовался в приложении, ниже по плану расположен «слайдер», который показывает различную рекламную информацию о акциях и предложениях организации. Для создания слайдера импортирована сторонняя библиотека «Image Slider», которая добавляет функции для создания таких объектов. Создан экземпляр класса, заданы необходимые параметры, такие как отступы, размеры и другое. С электронного ресурса организации ООО «Кабельные системы», было загружено несколько рекламных баннеров, которые были импортированы в приложение. В итоге получен готовый слайдер (Рисунок 23).



Рисунок 23 – Верхняя часть личного кабинета

Далее, под слайдером, расположены несколько текстовых полей которые будут выводить информацию о балансе лицевого счёта клиента и текущем тарифном плане с краткой информацией услуг, предоставляемых этим планом (Рисунок 24).

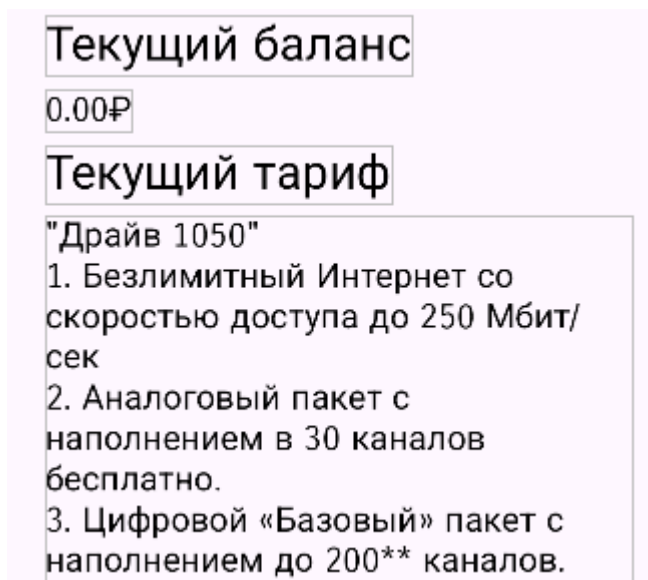


Рисунок 24 – Информация о тарифе

Под информацией о текущем тарифном плане сделан перечень всех тарифов, предоставляемых организацией с их характеристиками и возможностью перехода на новый тариф. Реализовано это так же при помощи слайдера. Для этого создан новый экземпляр класса «ImageSlider», загружены с официального сайта организации изображения с предоставляемыми тарифами и добавлены в приложение. Также создана новая функция всплывающего окна. Закодировано, что при нажатии на изображение с тарифом открывается окно с краткой информацией о нём. В этом окне расположены две кнопки: «Закрыть» и «Перейти на новый тариф». Первая кнопка закрывает окно с информацией о тарифе, а вторая создаёт заявку на смену тарифа. Закончив с настройкой, окна получен готовый слайдер с возможными тарифами (Рисунок 25).

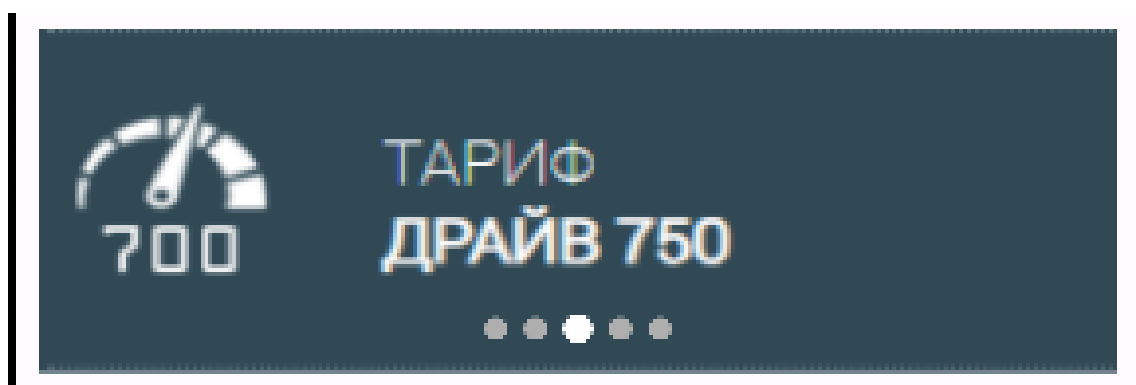


Рисунок 25 – Слайдер с тарифами

И, наконец, в низу страницы расположено меню приложения. Для этого создан объект «LinearLayout» для удобного и корректного расположения кнопок

меню на нём. Созданы пять кнопок, отвечающих за переход между страницами.

Первая кнопка «Справочная информация» открывает страницу с перечнем справочной информации. Для этого создан новый объект «фрагмент» для расположения всех страниц меню в пределах одной формы. Это позволило оптимизировать и ускорить работу приложения, так как вместо постоянного открытия новой страницы вся информация будет обновляться во фрагменте в пределах одной формы. Добавлены разделы с информацией предоставляемой организацией и текстовые поля, где эта информация выводится.

Вторая кнопка «Статистика» выводит информацию о использовании трафика пользователем и строит график для удобного просмотра. Добавлен новый «фрагмент», на нём размещены текстовые поля для вывода информации и объект для создания графиков. Для получения доступа к этому объекту импортирована библиотека «MPAndroidChart», которая позволяет строить графики и рисовать диаграммы. Настроены оси графиков для правильного отображения информации, заданы настройки расположения и отображения данных.

Третья кнопка «Главная страница» возвращает на ранее созданную основную страницу кабинета. Для этого создан новый фрагмент, на который перенесена ранее созданная страница.

Четвёртая кнопка «Чат с технической поддержкой» открывает форму для обращения в техническую поддержку. Для этого создан новый фрагмент, в котором расположены два текстовых поля, в первом отображаются все сообщения между сотрудником и клиентом, а второе необходимо для ввода сообщения пользователем. Также создана кнопка, которая будет отвечать за отправку сообщений. Для компактности поле ввода и кнопку отправки расположено ниже первого текстового поля, на одном уровне. Получен готовый чат с технической поддержкой (Рисунок 26).

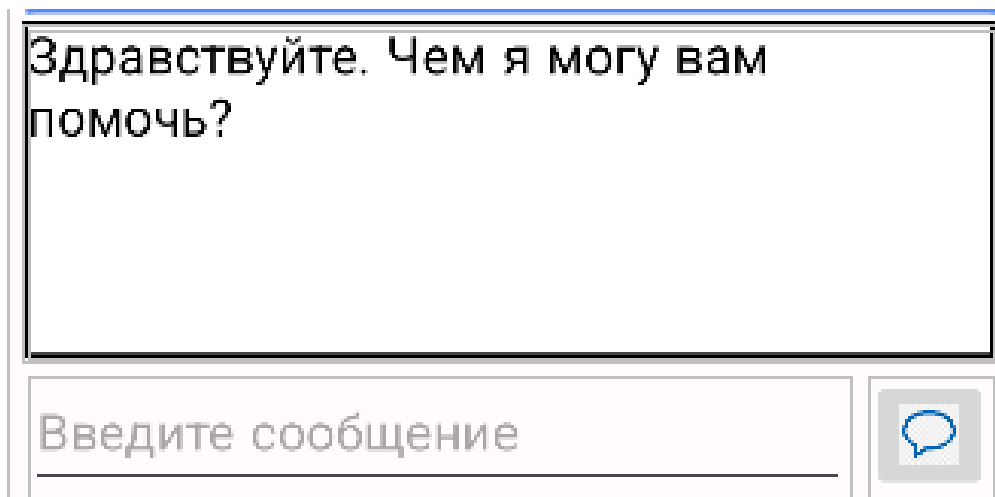


Рисунок 26 – Чат с технической поддержкой

Пятая кнопка «Настройки» открывает страницу с настройками приложения. Создан новый «фрагмент», на котором расположены несколько текстовых полей и кнопок. Сверху расположена информация о учётной записи. Ниже расположена кнопка для смены персональных данных, например, адреса и другие. Настроен переход между «фрагментами» по кнопкам и получено готовое меню приложения (Рисунок 27).

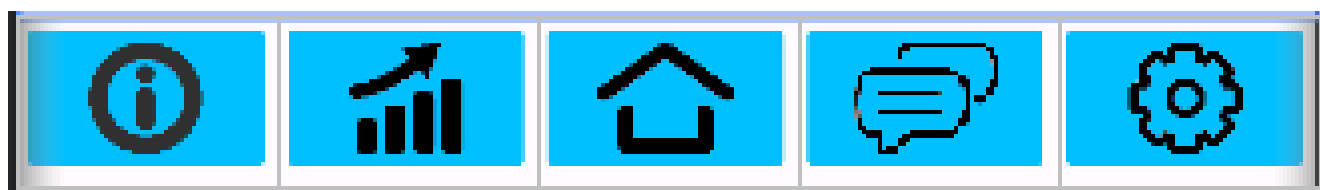


Рисунок 27 – Меню приложения

После созданного интерфейса настроен процесс перехода между страницами авторизации и личного кабинета. Для этого импортирована новая библиотека для подключения ранее созданной базы данных. Настроена передача данных, после нажатия «Войти» данные с информацией о логине и пароле передаются базе данных, там происходит «идентификация», поиск по логину на наличие пользователя в системе. В случае нахождения идентификатора начинается «аутентификация», то есть проверка правильности пароля, в обратном случае выводится ошибка «Пользователь не найден». В случае если логин верен, а пароль нет – выводится ошибка «Не верный пароль». В итоге программирования было создано множество файлов программы, которые можно увидеть в браузере IDE «Android Studio» (Рисунок 28).

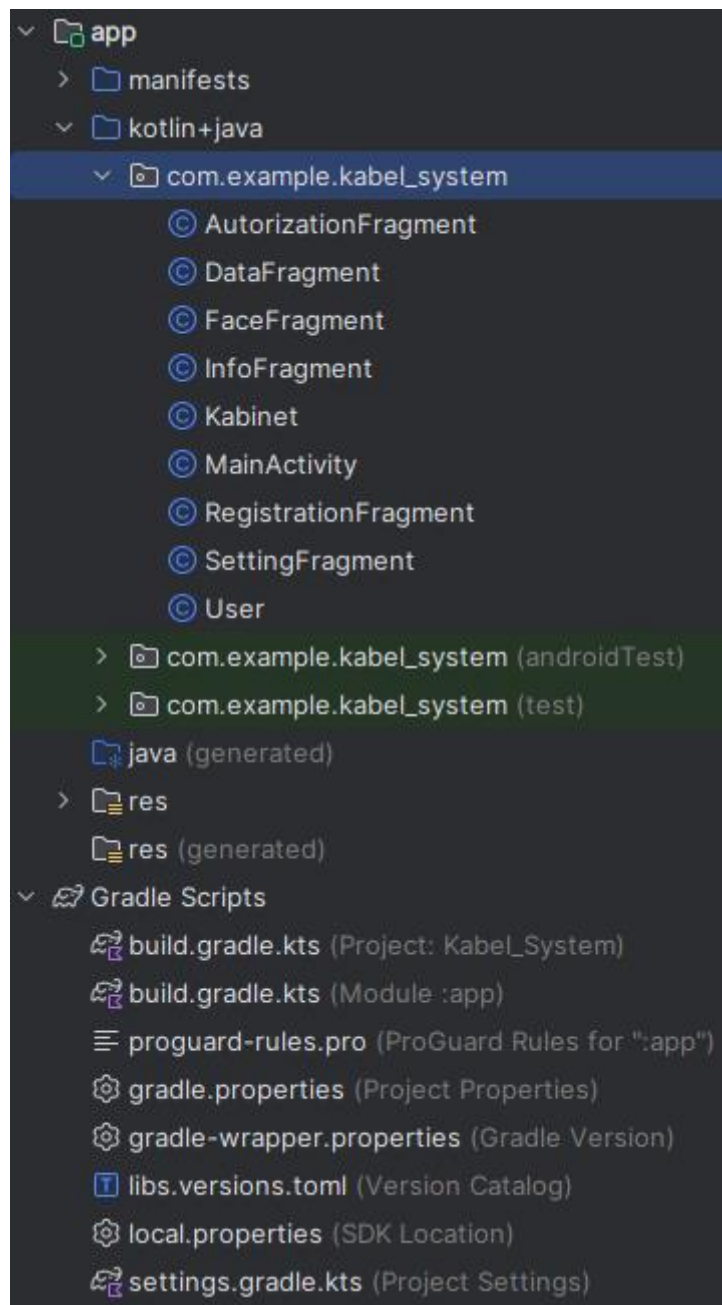


Рисунок 28 – Файлы программы

На этом разработка приложения и создание интерфейса была закончена. Осуществлён переход к тестированию ПО, для выявления возможных ошибок в работе.

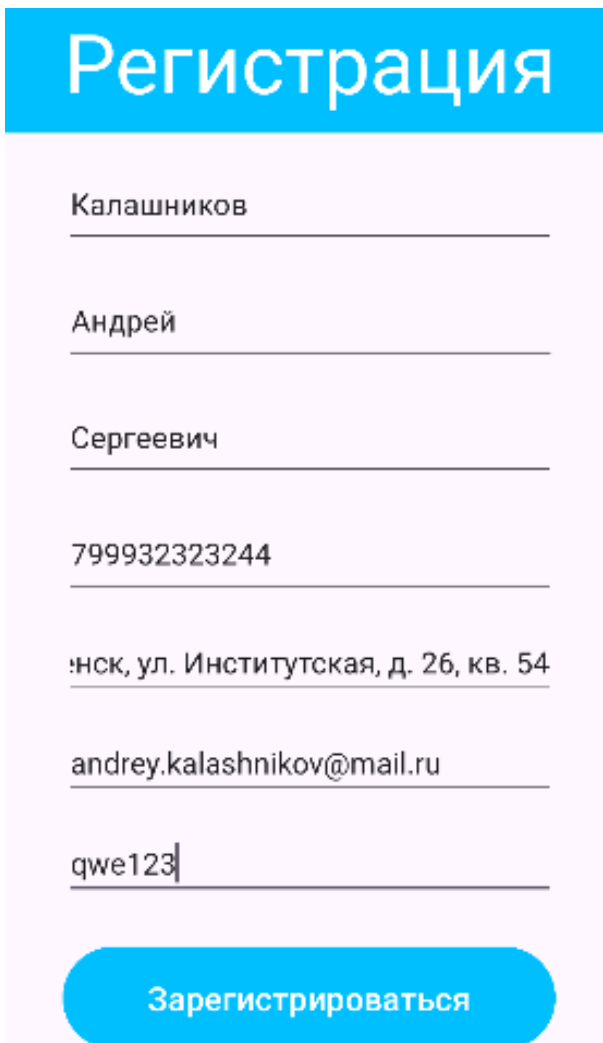
3.3 Тестирование программного продукта

Для того, чтобы убедиться в правильности работы приложения проведено полное тестирование всего функционала программы.

Тестирование ПО – это процесс проверки программного обеспечения для обнаружения ошибок, дефектов и недоработок, а также для оценки его работоспособности и соответствия заявленным требованиям.

Тестирование является важным этапом разработки программного обеспечения, поскольку позволяет выявить и устранить ошибки до выпуска продукта в эксплуатацию, что способствует повышению качества и стабильности работы программы.

Для начала тестирования запущено приложение. Была открыта начальная страница авторизации, нажата кнопка «Регистрация» и создан новый аккаунт. Для этого заполнены все нужные поля и нажата кнопка «Зарегистрироваться». Получено сообщение об успешной регистрации (Рисунок 29).



Регистрация

Калашников

Андрей

Сергеевич

799932323244

nsk, ул. Институтская, д. 26, кв. 54

andrey.kalashnikov@mail.ru

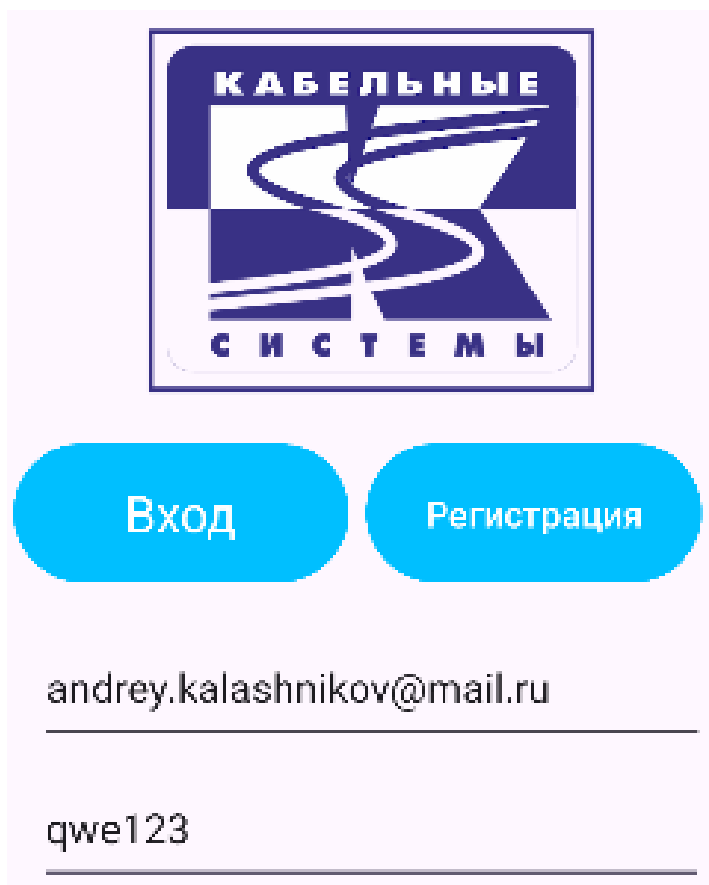
qwe123

Зарегистрироваться

Рисунок 29 – Регистрация нового пользователя

Перед переходом к авторизации, приложение протестировано на вывод ошибок. При заполнении некоторые поля оставлены пустыми и проведена попытка регистрации, в итоге получено сообщение о неудачной регистрации с указанием на пустую строку, также проведена проверка на несовпадение повтора пароля, полу-

чено соответствующее сообщение о некорректном вводе паролей. Блок ввода данных регистрации прошел тестирование, переход к тестированию авторизации. Для перехода на страницу авторизации нажата кнопка «Вход». Для входа введены данные пользователя (Рисунок 30).



The image shows a login interface for 'КАБЕЛЬНЫЕ СИСТЕМЫ'. At the top is a logo with a stylized blue and white graphic of a cable or fiber optic path, with the text 'КАБЕЛЬНЫЕ' above and 'СИСТЕМЫ' below. Below the logo are two blue buttons: 'Вход' (Login) and 'Регистрация' (Registration). Under the 'Вход' button, there is a text input field containing the email address 'andrey.kalashnikov@mail.ru'. Below that is another text input field containing the password 'qwe123'.

Рисунок 30 – Вход в личный кабинет

Всё прошло успешно и после небольшой прогрузки, был загружен личный кабинет. Сверху отображены рекламные предложения, «слайдер» работает и его содержимое указывается корректно. Ниже расположена информация о текущем тарифе и балансе лицевого счёта. Под ними расположен ещё один слайдер с перечнем доступных тарифов. При нажатии на картинку открывается окно с информацией о тарифе при нажатии «Закрыть» – закрывается. Внизу страницы расположено меню со справочной информацией, статистикой, главным экраном, технической поддержке и настройками. Страницы переключаются корректно. Вся информация, расположенная на них, отображается верно (Рисунок 31).

Здравствуйте, Андрей Сергеевич!



Текущий баланс

1000.00 ₺

Текущий тариф

"Драйв 1050"

1. Безлимитный Интернет со скоростью доступа до 250 Мбит/сек
2. Аналоговый пакет с наполнением в 30 каналов бесплатно.
3. Цифровой «Базовый» пакет с наполнением до 200** каналов.

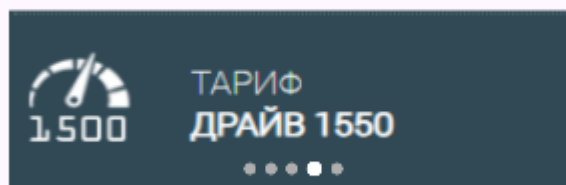


Рисунок 31 – Личный кабинет

Все функции прошли проверку, приложение протестировано и готово к использованию.

4 БЕЗОПАСНОСТЬ И ЭКОЛОГИЧНОСТЬ

4.1 Безопасность

В разработке Android приложения для клиентов ООО "Кабельные системы" одним из основных приоритетов является обеспечение безопасности пользователей. Данное приложение должно быть защищено от возможных угроз и атак со стороны злоумышленников. Для этого в приложении будут использоваться современные методы шифрования данных, аутентификации пользователей и механизмы защиты от вредоносного ПО.

Android приложения могут столкнуться с различными угрозами, включая вредоносные программы, перехват данных, атаки на пользовательский интерфейс и другие виды атак. Некоторые из наиболее распространенных угроз и рекомендации по их защите представлены в пунктах (а-г):

а) вредоносные программы – это вредоносные приложения, которые могут украсть конфиденциальные данные, повредить устройство или причинить другие виды вреда. Вредоносные программы для Android могут проникнуть на устройство через загрузки из неофициальных источников, фишинговые ссылки, поддельные приложения или даже через вредоносные объявления. Эти программы могут причинить ущерб вашим личным данным, вымогать деньги или причинить другой вид вреда. Чтобы защитить устройство от вредоносных программ необходимо следовать рекомендациям:

1) устанавливать приложения только из официальных магазинов приложений. Это поможет избежать установки поддельных приложений, содержащих вредоносный код;

2) обновлять операционную систему и приложения регулярно. Обновления содержат исправления уязвимостей, которые могут быть использованы вредоносными программами;

3) устанавливать антивирусное приложение на свое устройство. Это поможет обнаруживать и удалять вредоносные программы, прежде чем они нанесут вред вашему устройству;

4) быть осторожным при нажатии на ссылки в сообщениях, электронной почте или на сайтах. Убедитесь, что источник доверенный, прежде чем взаимодействовать с содержимым;

5) избегать подключения к общедоступным Wi-Fi сетям, особенно тех, которые не требуют пароля. Хакеры могут использовать такие сети для перехвата вашего трафика и кражи личных данных;

б) перехват данных – это атаки, при которых злоумышленники могут получить доступ к конфиденциальным данным пользователей, таким как пароли, данные кредитных карт и личная информация. Перехват данных в приложениях Android может происходить через различные методы, включая незащищенное соединение с сервером, использование небезопасных библиотек или недостаточную обработку данных. Это может привести к утечке конфиденциальной информации, такой как личные данные, пароли, финансовая информация и другие чувствительные данные пользователей. Для защиты от перехвата данных в приложениях Android рекомендуется применять следующие методы:

1) для защиты данных в пути между устройством и сервером следует использовать защищенное соединение HTTPS. Это поможет предотвратить перехват данных посредством шифрования;

2) важно проводить обработку и хранение конфиденциальных данных на стороне сервера, чтобы предотвратить их утечку через уязвимости в приложении;

3) разработчики приложений должны использовать проверенные и безопасные библиотеки и API для обработки и передачи данных, чтобы предотвратить уязвимости, которые могут привести к перехвату данных;

4) для защиты данных на устройстве следует использовать шифрование, особенно для хранения конфиденциальной информации, такой как пароли и личные данные;

5) проведение регулярного тестирования на уязвимости поможет выявить потенциальные проблемы с безопасностью данных в приложении;

б) важно обучать команду разработчиков и тестировщиков основам

безопасности приложений и методам защиты данных;

в) атаки на пользовательский интерфейс – это тип атак, который может позволить злоумышленнику изменять поведение приложения или отображать фальшивые диалоговые окна для обмана пользователей. Для защиты от таких угроз, разработчики должны использовать безопасные практики разработки, такие как проверку подписей цифровых сертификатов и проведение тестирования на уязвимости. Данный тип атак может включать в себя различные виды мошенничества, вредоносных программ и других угроз безопасности. Некоторые из наиболее распространенных атак на пользовательский интерфейс Android включают в себя:

1) фишинг – это атака, при которой злоумышленники создают поддельные веб-сайты или приложения, имитирующие легитимные сервисы, чтобы получить конфиденциальные данные пользователей, такие как пароли, номера кредитных карт и личная информация;

2) манипуляция с помощью социальной инженерии, при которой злоумышленники могут использовать манипуляцию, чтобы убедить пользователей в установке вредоносных приложений или предоставлении конфиденциальной информации;

3) вредоносные приложения, при помощи которых злоумышленники могут создавать вредоносные приложения, которые выдают себя за легитимные, чтобы получить доступ к данным пользователя или устройству;

4) межсайтовый скриптинг (XSS) – это атака, при которой злоумышленники внедряют вредоносные скрипты в веб-страницы или приложения, чтобы получить доступ к данным пользователей.

Для защиты от этих атак на пользовательский интерфейс необходимы: Регулярные обновления операционной системы и приложений, которые помогут закрывать уязвимости и обеспечивать безопасность устройства. Использование антивирусных программ, помогающее обнаруживать и удалять вредоносные приложения и программы. Установка приложений только из официальных и проверенных источников, таких как RuStore, и проявление осторожности при предоставлении разрешений приложениям. Обучение пользователя основам безопасности в

сети и распознавание мошеннических схем для предотвращения атак на пользовательский интерфейс. Включение двухфакторной аутентификации для важных аккаунтов, что повышает уровень безопасности;

г) Атаки на серверы, атаки на API, атаки на библиотеки и другие. Для защиты от таких угроз рекомендуется использовать безопасные методы аутентификации, шифрование данных при передаче и использование безопасных библиотек.

Для обеспечения безопасности Android приложений необходимо следовать советам по безопасности разработки приложений, регулярно обновлять и тестировать приложения на наличие уязвимостей, использовать аутентификацию, шифрование и другие методы защиты данных, а также следить за безопасностью своего устройства и сети.

4.2 Экологичность

При разработке Android приложения для клиентов ООО "Кабельные системы" уделяется внимание экологичности и устойчивому развитию. В приложении будут учитываться принципы энергосбережения и оптимизации производительности устройств, чтобы снизить нагрузку на батарею и ресурсы устройства. Также будет поддерживаться возможность использования приложения на различных устройствах с разной производительностью и мощностью.

Оптимизация Android приложения для энергоэффективной работы является важным аспектом, который помогает уменьшить потребление энергии и продлить время автономной работы устройства. Некоторые способы оптимизации:

– оптимизация работы сети. Использование эффективных способов передачи данных, такие как использование HTTP/2 для загрузки контента, сокращение количества запросов к серверу, кэширование данных и минимизация объема передаваемых данных;

– управление процессами и памятью. Оптимизация работы с процессами приложения, во избежания излишнего использования процессора и памяти. Разумное использование фоновых задач и ограничение их, чтобы не загружать устройство;

– оптимизация визуализации и анимации. Избегание излишнего использования видео и анимаций, которые могут потреблять много ресурсов. Использование легких и оптимизированных компонентов интерфейса пользователя;

– управление энергопитанием. Использование функций Android для управления энергопитанием, таких как JobScheduler для планирования фоновых задач или WorkManager для выполнения задач при наличии доступных ресурсов;

– оптимизация работы с мультимедийным контентом. При работе с аудио и видео контентом необходимо использовать кодеки с высокой эффективностью сжатия, чтобы уменьшить размер передаваемых данных и процессорных нагрузок;

– тестирование и профилирование. Необходимо проводить тестирование приложения на реальных устройствах с разными характеристиками, а также профилировать его, чтобы выявить узкие места в потреблении ресурсов и энергии.

С учетом всех этих аспектов и методов оптимизации, можно значительно улучшить энергоэффективность Android приложения, что положительно сказывается на пользовательском опыте и на экологическом следе приложения.

4.3 Чрезвычайные ситуации

Для обеспечения готовности к чрезвычайным ситуациям в разработанном Android приложении будут предусмотрены механизмы аварийного отключения и автоматического восстановления работы приложения. Также будут разработаны инструкции и рекомендации для пользователей по действиям в случае возникновения проблем с приложением.

Для оперативного реагирования на чрезвычайные ситуации будет организована круглосуточная служба поддержки и мониторинга, которая будет следить за работоспособностью приложения и предпринимать необходимые меры по его восстановлению в случае необходимости.

Перечень возможных чрезвычайных ситуаций, которые могут возникнуть при работе Android приложения предоставлены в пунктах (а-е):

а) сбой в работе сервера. Если сервер, к которому приложение обращается для получения данных, перегружен или не работает, пользователи могут столкнуться с проблемами в использовании приложения. Сбои в работе сервера Android

приложения могут быть вызваны различными причинами, такими как проблемы с сетью, неправильная конфигурация сервера, ошибки в коде приложения и т.д. Для решения сбоев в работе сервера Android приложения можно предпринять следующие меры:

- 1) проверить сетевое соединение. Необходимо убедиться, что сервер имеет доступ к интернету и что сетевые настройки на устройстве, на котором запущено приложение, корректно настроены;

- 2) проверить логи сервера. Необходимо просмотреть логи работы сервера, чтобы выявить возможные ошибки или проблемы, которые могут привести к сбоям. Также проверить логи Android приложения на наличие ошибок и предупреждений;

- 3) обновить приложение. Убедиться, что установлена последняя версия Android приложения, чтобы избежать известных ошибок и уязвимостей;

- 4) перезапустить сервер. Перезапустить сервер, чтобы устранить возможные проблемы с его работой.

В целом, для успешного решения сбоев в работе сервера Android приложения важно провести анализ причин сбоя, диагностировать их и принять соответствующие меры для исправления проблемы;

б) утечка персональной информации. Если данные пользователей хранятся не безопасно или приложение имеет уязвимости в системе безопасности, это может привести к утечке конфиденциальных данных. Утечка персональной информации в Android приложении может произойти по различным причинам, таким как неправильная настройка прав доступа к данным, несанкционированный доступ к базе данных или слабые меры безопасности при передаче данных через сеть. Для предотвращения утечки персональной информации в Android приложении можно применить следующие меры:

- 1) шифрование данных. Важно зашифровать чувствительные данные перед их хранением или передачей через сеть;

- 2) ограничение доступа. Необходимо ограничить доступ к персональной информации только необходимым пользователям и предоставить доступ

только через защищенные механизмы аутентификации;

3) безопасная передача данных. Для защиты данных при передаче через сеть следует использовать протоколы шифрования, такие как HTTPS;

4) аудит безопасности. Необходимо регулярно проверять приложение на наличие уязвимостей, а также мониторить несанкционированные попытки доступа к данным;

5) обновление системы безопасности. Важно регулярно обновлять систему безопасности приложения и операционную систему, чтобы устранить известные уязвимости;

6) соблюдение законодательства. При разработке приложения следует соблюдать законы и правила, касающиеся защиты персональной информации пользователей.

Приложение должно быть построено с соблюдением всех вышеперечисленных мер безопасности, чтобы предотвратить утечку персональной информации и обеспечить конфиденциальность пользователей;

в) недостаточное тестирование. Если приложение не было тщательно протестировано на различных устройствах и операционных системах, пользователи могут столкнуться с ошибками, зависаниями или неправильной работой функций. Недостаточное тестирование Android приложения может привести к появлению ошибок и сбоев, что негативно отразится на пользовательском опыте. Для решения этой проблемы следует проводить комплексное тестирование приложения на всех этапах его разработки. Вот несколько способов улучшить тестирование Android приложения:

1) используйте автоматизированные тесты: автоматизированные тесты помогут повысить скорость тестирования и обеспечить более полное покрытие функциональности приложения;

2) регулярно проводите ручное тестирование: ручное тестирование позволяет выявить проблемы, которые могут быть упущены при автоматизированном тестировании;

3) проводите тестирование на различных устройствах: убедитесь, что

приложение работает корректно на различных версиях Android и разных устройствах;

4) используйте инструменты для отслеживания ошибок: используйте специализированные инструменты для отслеживания и исправления ошибок приложения;

5) привлекайте пользователей для бета-тестирования: предоставьте доступ к бета-версии приложения небольшой группе пользователей, чтобы получить обратную связь и выявить проблемы до релиза;

г) атаки хакеров. Злоумышленники могут попытаться провести атаку на приложение, нарушить его работу или украсть данные пользователей. Атаки хакеров на Android приложения могут происходить различными способами, включая следующие:

1) внедрение вредоносного кода в приложение или его обновление. Это может привести к краже личной информации пользователей, установке вредоносных программ на устройство и даже управлению им;

2) атака на незащищенную сеть, через которую отправляется и получается информация от приложения. Хакеры могут перехватывать данные, включая логины, пароли, банковские данные и другую конфиденциальную информацию;

3) использование уязвимостей в коде приложения для выполнения SQL инъекций или других типов атак. Это может привести к удалению данных, повреждению приложения или краже информации.

Для защиты от таких атак необходимо использовать шифрование данных при передаче и хранении на устройстве, также регулярно обновлять приложение для исправления уязвимостей и исправления ошибок безопасности, проводить аудит безопасности приложения для обнаружения потенциальных уязвимостей и устранения их, ограничивать доступ к конфиденциальной информации, например, через аутентификацию и авторизацию пользователей, использовать антивирусное программное обеспечение на устройстве для обнаружения и удаления вредонос-

ного ПО и обучать сотрудников и пользователей безопасным практикам использования приложения, например, не отвечать на подозрительные запросы или не нажимать на подозрительные ссылки;

д) недостаточное обновление. Если разработчики не выпускают регулярные обновления приложения, это может привести к несовместимости с новыми версиями операционных систем или уязвимостям безопасности, которые не были устранены.

Проблема с недостаточными обновлениями разработчиков Android приложений заключается в том, что приложения могут становиться уязвимыми для кибератак, ошибок и неполадок. Не усовершенствованные приложения могут работать нестабильно, терять функциональность или перестать поддерживаться новыми версиями операционной системы Android.

Кроме того, отсутствие обновлений может привести к ухудшению производительности приложений, несовместимости с новыми устройствами или функциями платформы Android. Пользователи могут столкнуться с проблемами загрузки данных, замедлением работы приложений, вылетами и другими неприятностями.

Для разработчиков важно регулярно выпускать обновления, чтобы исправлять ошибки, обновлять функциональность, следить за изменениями в требованиях платформы и обеспечивать безопасность приложения. Регулярные обновления также помогут удерживать пользователей, улучшать репутацию и конкурентоспособность приложения на рынке;

е) потеря связи. Если устройство пользователя теряет соединение с интернетом во время использования приложения, это может привести к потере данных или невозможности завершить операции. Потеря связи в Android приложении может проявиться в различных формах и иметь различные причины. Вот некоторые из наиболее распространенных сценариев потери связи:

1) отсутствие интернет-соединения. Если устройство потеряло подключение к сети Wi-Fi или мобильным данным, приложение может не иметь доступа к интернету и, следовательно, не сможет обновлять данные с сервера или загружать новый контент;

2) проблемы с сервером. Иногда проблема может быть на стороне сервера, с которым приложение взаимодействует. Если сервер недоступен или работает нестабильно, приложение может не получать необходимую информацию;

3) ошибки в коде приложения. Некорректно написанный код приложения может привести к сбоям в работе, включая потерю связи. Например, неправильная обработка сетевых запросов или отсутствие обработки ошибок сети может вызвать проблемы;

4) проблемы с настройками устройства. Также проблема может быть связана с настройками самого устройства, такими как ограничения на использование данных для определенных приложений или блокировка сети.

Чтобы решить проблему потери связи в Android приложении необходимо проверить подключение к интернету на устройстве, перезапустите приложение, если этого будет недостаточно то, перезагрузить устройство и проверить настройки сети на устройстве.

ЗАКЛЮЧЕНИЕ

В заключении отметим, что все поставленные цели и задачи были выполнены. Была проведена большая работа по изучению нового материала: работа с интегрированной средой разработки приложений под ОС Android, проектирование, создание и настройка базы данных, тестирование ПО.

За время выполнения выпускной квалификационной работы был проанализирован рынок мобильных приложений на базе Android, в связи с чем были выявлены основные требования к разработке таких приложений, спроектирована и настроена база данных «Firebase Realtime Database» для хранения и обработки данных клиентов и организации.

Также было спроектировано и разработано приложение для клиентов ООО «Кабельные системы», которое позволит значительно улучшить качество обслуживания клиентов организации, упростить процесс заказа услуг, а также повысить уровень коммуникации между клиентами и представителями организации. Полученное приложение обладает широкими функциональными возможностями, простым и интуитивно понятным интерфейсом, что делает его удобным и эффективным инструментом для взаимодействия с клиентами. Данное приложение представляет собой важный шаг в развитии информационных технологий организации, что способствует увеличению конкурентоспособности на рынке.

В дальнейшем есть несколько путей развития данного приложения, например, добавление дополнительных функциональных возможностей, функций, которые могли бы улучшить опыт пользователей, такие как уведомления о статусе заказа, интеграция с каталогом товаров или возможность оформления заказа через приложение. При развитии приложение возможно продолжение совершенствования интерфейса и дизайна, для того чтобы следовать будущим нормам и стандартам в дизайн коде. Также можно улучшить безопасность данных пользователей, путём добавления дополнительных «Двухэтапных аутентификаций», которые помогут дополнительно защитить данные пользователей от хищения злоумышленниками.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1 Волк, В. К. Базы данных. Проектирование, программирование, управление и администрирование / В. К. Волк. – Изд-во Лань, 2021. – 242 с.

2 Годин, В. В. Базы данных. Проектирование баз данных. Учебник для академического бакалавриата / В. В. Годин, Н. П. Стружкин. – Изд-во ЮРАЙТ, 2020. – 477 с.

3 Годин, В. В. Базы данных: проектирование. Практикум. Учебное пособие для СПО / В. В. Годин, Н. П. Стружкин. – Изд-во ЮРАЙТ, 2018. – 291 с.

4 Джеквони, К. Идеальный тестировщик. Концепции, навыки и стратегии высококачественного тестирования / К. Джеквони. – Изд-во БХВ, 2024. – 416 с.

5 Золкин, А. Л. Технологии разработки и проектирования информационных систем. Часть 1. Учебное пособие / А. Л. Золкин, Г. Я. Ратушняк. – Изд-во КноРус, 2023. – 201 с.

6 Золкин, А. Л. Технологии разработки и проектирования информационных систем. Часть 2. Учебное пособие / А. Л. Золкин, Г. Я. Ратушняк. – Изд-во КноРус, 2024. – 349 с.

7 Колисниченко, Д. Н. Программирование для Android / Д. Н. Колисниченко. – Изд-во БХВ, 2021. – 288 с.

8 Кугаевских, А. В. Проектирование информационных систем. Системная и бизнес-аналитика / А. В. Кугаевских. – Изд-во НГТУ, 2019. – 256 с.

9 Мартишин, С. А. Базы данных. Практическое применение СУБД SQL и NoSQL-типа для применения проектирования информационных систем / С. А. Мартишин, М. В. Храпченко, В. Л. Симонов. – Изд-во Форум, 2018. – 368 с.

10 Назина, О. К. Что такое тестирование. Курс молодого бойца / О. К. Назина. – Изд-во БХВ, 2022. – 592 с.

11 Осипов, Д. Л. Технологии проектирования баз данных / Д. Л. Осипов. – Изд-во ДМК Пресс, 2019. – 498 с.

12 Прамодкумар, Д. С. NoSQL. Методология разработки нереляционных баз данных / Д. С. Прамодкумар. – Изд-во Диалектика, 2020. – 192 с.

- 13 Ретабоуил, С. Android NDK. Руководство для начинающих / С. Ретабоуил. – Изд-во ДМК Пресс, 2016. – 518 с.
- 14 Самойлов, Д. Краткий справочник тестировщика / Д. Самойлов. – Изд-во Ridero, 2023. – 116 с.
- 15 Стасышин В. М. Проектирование информационных систем и баз данных / В. М. Стасышин. – Изд-во НГТУ, 2018. – 100 с.
- 16 Стюарт, К. Android. Программирование для профессионалов / К. Стюарт, К. Марсикано, Б. Гарднер, Б. Филлипс. – Изд-во Питер, 2021. – 704 с.
- 17 Федотенко, М. А. Разработка мобильных приложений. Первые шаги / М. А. Федотенко. – Изд-во Лабораторная знаний, 2019. – 335 с.
- 18 Фрайман, З. М. Создание приложений для смартфонов и планшетов под ОС Android / З. М. Фрайман. – Изд-во Едиториал УРСС, 2020. – 504 с.
- 19 Чхави, Р. Д. Сам себе тестировщик. Пошаговое руководство по тестированию ПО / Р. Д. Чхави. – Изд-во Питер, 2024. – 240 с.
- 20 Ян, Д. Ф. Android. Сборник рецептов. Задачи и решения для разработчиков приложений / Д. Ф. Ян. – Изд-во Диалектика, 2019. – 768 с.

ПРИЛОЖЕНИЕ А

Техническое задание.

1 Введение

1.1 Наименование программы

Наименование программы – Android приложение для клиентов ООО «Кабельные системы».

1.2 Краткая характеристика области применения

Android приложение для клиентов ООО «Кабельные системы» предназначена для удобного взаимодействия между клиентами и организацией. Программное обеспечение имеет простой и понятный интерфейс, обладает простотой использования и предназначен для устройств на базе Android.

2 Основания для разработки

Основанием для разработки является необходимость создания ПО для ОС Android берущее информацию из БД клиентов организации, для предоставления актуальной информации и удобного взаимодействия между клиентами и поставщиками услуг.

3 Назначение разработки

Программа будет использоваться клиентами организации «Кабельные системы», на личных Android устройствах клиентов. Предполагается свободное распространение ПО в официальных магазинах приложений.

3.1 Функциональное назначение

Программа предоставляет доступ к следующим функциям:

- авторизация,
- баланс денежных средств,
- возможности текущего тарифного плана,
- перечень возможных тарифов,
- даты платежей и сроки следующего,
- просмотр статистических данных (использование трафика, период активности),

Продолжение ПРИЛОЖЕНИЯ А

- справочная информация,
- чат с технической поддержкой,
- оплата счёта,
- предоставление информации о рекламе и предложениях.

3.2 Эксплуатационное назначение

Программа должна эксплуатироваться на личном устройстве клиента на базе ОС Android с обязательным доступом к сети Интернет.

4 Требования к программе или программному изделию

4.1 Требования к функциональным характеристикам

4.1.1 Требования к составу выполняемых функций

После входа и авторизации в программу перед клиентом открывается информация и предоставляется возможность выбора из нескольких функций:

- На главном экране отображается: текущий баланс, тарифный план, модуль с рекламными предложениями.

- Кнопка с онлайн-чатом. После нажатия открывается чат с сотрудником тех. поддержки.

- Кнопка с тарифами. Открывает все возможные предложения организации. При нажатии и подтверждении возможен переход на другой тарифный план.

- Кнопка оплаты. Открывает окно с выбором способа оплаты. После выбора перенаправляет на страницу платёжной системы.

- Справочная информация. Открывает справочную информацию по разделам.

4.1.2 Требования к организации входных и выходных данных

Данные о клиентах хранятся в базе данных на серверах организации. СУБД обеспечивает защиту информации от посторонних лиц, предоставляет ограниченную информацию непосредственно клиенту. Права доступа к СУБД есть только у уполномоченных сотрудников организации.

4.1.3 Требования к временным характеристикам

После изменения данных, новая информация отображается не позднее, чем

Продолжение ПРИЛОЖЕНИЯ А

через 10 секунд.

4.2 Требования к надежности

Вероятность безотказной работы системы должна составлять не менее 99.99% при условии исправности сети (связи приложений оператора и посетителя с базой данных).

4.2.1 Требования к обеспечению надежного (устойчивого) функционирования программы

В связи с тем, что в базе данных хранятся данные о клиентах (конфиденциальная информация) – базу данных стоит резервировать (резервирование замещением). Надежное (устойчивое) функционирование программы должно быть обеспечено выполнением организацией совокупности организационно-технических мероприятий, перечень которых приведен ниже:

- организацией бесперебойного питания технических средств;
- использованием лицензионного программного обеспечения;
- регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998 г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;
- регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

4.2.2 Время восстановления после отказа

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем (не крахом) операционной системы, не должно превышать 10 минут при условии соблюдения условий эксплуатации технических и программных средств. Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

Продолжение ПРИЛОЖЕНИЯ А

4.2.3 Отказы из-за некорректных действий оператора

Отказы программы возможны вследствие некорректных действий оператора (пользователя) при взаимодействии с операционной системой. Во избежание возникновения отказов программы по указанной выше причине следует обеспечить работу пользователя без предоставления ему административных привилегий.

4.3 Условия эксплуатации

Программа запускается и работает на устройстве клиента. База данных находится на серверах организации. Должна существовать устойчивая связь по сети между клиентом и базой данных. Программа открывается на весь экран, не должно быть возможности запустить любое стороннее программное обеспечение способное помешать работе программы. Запуск программы должен осуществляться после старта операционной системы.

4.3.1 Климатические условия эксплуатации

Специальные условия не требуются.

4.3.2 Требования к видам обслуживания

Программа не требует проведения каких-либо видов обслуживания.

4.3.3 Требования к численности и квалификации персонала

При установке и настройке ПО не требуется специальной квалификации.

4.4 Требования к составу и параметрам технических средств

Состав технических средств:

- Android устройство клиента, включающее в себя:
- процессор x86 с тактовой частотой, не менее 1 ГГц;
- оперативную память объемом, не менее 1 Гб.

4.5 Требования к информационной и программной совместимости Приложения клиента запрашивает данные с СУБД по сети Интернет, при этом используется протокол HTTP.

4.6 Требование к маркировке и упаковке

Программное изделие передается по сети Internet в виде архива – загружается с официального сайта производителя или магазина приложений.

Продолжение ПРИЛОЖЕНИЯ А

Специальных требований к маркировке не предъявляется. Для проверки подлинности программного обеспечения рекомендуется проверять контрольные суммы загруженных файлов со значениями, указанными на официальном сайте.

4.7 Требования к транспортированию и хранению

Специальных требований не предъявляется.

4.8 Специальные требования

Программа должна обеспечивать взаимодействие с пользователем посредством графического пользовательского интерфейса, разработанного согласно рекомендациям организации-производителя операционной системы.

5 Требования к программной документации

Предварительный состав программной документации:

- техническое задание (включает описание применения);
- программа и методика испытаний;
- руководство системного программиста;
- руководство оператора;
- руководство программиста;
- ведомость эксплуатационных документов;
- формуляр.

6 Техничко-экономические показатели

Программа «Android приложение для клиентов Кабельные системы» используется клиентами организации. За счёт удобного взаимодействия с клиентом предполагается повышение привлекательности организации на фоне конкурентов, и как следствие увеличение прибыли.

7 Стадии и этапы разработки

Разработка должна быть проведена в три стадии:

- техническое задание;
- технический (и рабочий) проекты;
- внедрение.

На стадии «Техническое задание» должен быть выполнен этап разработки,

Продолжение ПРИЛОЖЕНИЯ А

согласования и утверждения настоящего технического задания. На стадии «Технический (и рабочий) проект» должны быть выполнены перечисленные ниже этапы работ:

- разработка программы;
- разработка программной документации;
- испытания программы.

На стадии «Внедрение» должен быть выполнен этап разработки «Подготовка и передача программы».

Содержание работ по этапам:

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

- постановка задачи;
- определение и уточнение требований к техническим средствам;
- определение требований к программе;
- определение стадий, этапов и сроков разработки программы и документации на нее;
- согласование и утверждение технического задания.

На этапе разработки программы должна быть выполнена работа по программированию (кодированию) и отладке программы. На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями ГОСТ 19.101-77. На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:

- разработка, согласование и утверждение порядка и методики испытаний;
- проведение приемо-сдаточных испытаний;
- корректировка программы и программной документации по результатам испытаний.

На этапе подготовки и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию на объектах заказчика.

Продолжение ПРИЛОЖЕНИЯ А

8 Порядок контроля и приемки

Приемосдаточные испытания программы должны проводиться согласно разработанной исполнителем и согласованной заказчиком «Программы и методики испытаний». Ход проведения приемо-сдаточных испытаний заказчик и исполнитель документируют в протоколе испытаний. На основании протокола испытаний исполнитель совместно с заказчиком подписывают акт приемки-сдачи программы в эксплуатацию.

Список используемой литературы

1. ГОСТ 19.201-78 Единая система программной документации. Техническое задание. Требования к содержанию и оформлению. 1978. Режим доступа: <http://protect.gost.ru/document.aspx?control=7&id=155153>

2. ГОСТ 24.701-86. Единая система стандартов автоматизированных систем управления. Надежность автоматизированных систем управления. Основные положения. М.: Издательство стандартов, 1987. – 17 с. 3. Создание проекта форм интерфейса и карты диалоговых окон в PLANTUML [Электронный ресурс]. Режим доступа: <https://habr.com/ru/post/279373/> (27.09.2020)