Министерство образования Новосибирской области

ГБПОУ НСО «Новосибирский авиационный технический колледж имени Б.С. Галущака»

**разработка** **мобильного приложения для автосервиса**

Пояснительная записка к курсовому проекту

ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

МДК.01.03 Разработка мобильных приложений

НАТКиГ.200800.010.000ПЗ

Разработал:

студент группы ПР-20.101

Зиберева А.А.

2022

**Содержание**

[Введение 3](#_Toc117112970)

[1 Исследовательский раздел 4](#_Toc117112971)

[1.1 Описание предметной области 4](#_Toc117112972)

[1.2 Образ клиента 5](#_Toc117112973)

[1.3 Сценарии 5](#_Toc117112974)

[1.4 Сбор и анализ прототипов 6](#_Toc117112975)

[2 Проектирование приложения 11](#_Toc117112976)

[2.1 UI/UX дизайн проекта 11](#_Toc117112977)

[2.2 Выбор технологии, языка и среды программирования 18](#_Toc117112978)

[3 Разработка мобильного приложения 19](#_Toc117112979)

[3.1 Разработка базы данных 19](#_Toc117112980)

[3.2 Разработка используемых плагинов 20](#_Toc117112981)

[3.3 Описание разработанных процедур и функций 21](#_Toc117112982)

[4 Тестирование 26](#_Toc117112983)

[4.1 Протокол тестирования дизайна приложения 26](#_Toc117112984)

[4.2 Протокол тестирования функционала приложения 26](#_Toc117112985)

[Заключение 29](#_Toc117112986)

[Библиография 30](#_Toc117112987)

[Приложение А 31](#_Toc117112988)

Введение

Мобильное приложение ГУ МВД Новосибирской области делает более удобным взаимодействие с органами исполнительной власти гражданина: гражданин (пользователь) имеет возможность ознакомиться с информацией об органах исполнительной власти (адреса, руководители, подразделения, структура органов), может записаться на прием в МВД, подать заявление или обращение, найти нужную разыскиваемую личность в базе данных розыска, ознакомиться с местонахождением участков полиции в городе, обратиться за помощью или консультацией. Также, приложение позволяет в любой удобный момент совершить звонок в нужную службу и узнать последние новости от ГУ МВД. Приложение позволяет также ознакомиться с правовым информированием граждан и взаимодействовать с большим набором полезных ресурсов/ссылок, которые необходимы взрослому гражданину почти каждый день.

Несмотря на то, что для ГУ МВД Новосибирской области уже существует веб-сайт – приложение поможет гораздо эффективнее и проще взаимодействовать с информацией, предоставленной органами исполнительной власти.

Целью курсового проекта является создание мобильного приложения ГУ МВД Новосибирской области.

Задачами курсового проекта в связи с указанной целью являются:

* изучение предметной области темы приложения;
* рассмотрение приложения с точки зрения пользователя для выявления необходимых функций приложения;
* создание UI/UX дизайна приложения;
* изучение вопроса целевой аудитории приложения;
* анализ существующих подобные приложения;
* определение этапов разработки мобильного приложения;
* разработка функционала приложения;
* тестирование готового приложения.

Объект исследования – приложения органов исполнительной власти (МВД).

Предмет исследования – изучение принципов функционирования и инструментов приложения.

# Исследовательский раздел

## Описание предметной области

Министерство внутренних дел (МВД) — федеральный орган исполнительной власти, осуществляющий функции по выработке и реализации государственной политики и нормативно-правовому регулированию в сфере внутренних дел, в сфере контроля за оборотом наркотических средств, психотропных веществ и их прекурсоров, в сфере миграции, а также правоприменительные функции по федеральному государственному контролю (надзору) в сфере внутренних дел. Подследственность органов внутренних дел регулирует УПК России.

Министерство внутренних дел является федеральным органом исполнительной власти, осуществляющим в пределах своих полномочий государственное управление в сфере защиты прав и свобод человека и гражданина, охраны правопорядка, обеспечения общественной безопасности и непосредственно реализующим основные направления деятельности органов внутренних дел Российской Федерации.

Основные задачи Министерства:

* разработка и принятие в пределах своей компетенции мер по защите прав и свобод человека и гражданина, защите объектов независимо от форм собственности, обеспечению общественного порядка и общественной безопасности;
* организация и осуществление мер по предупреждению и пресечению преступлений и административных правонарушений, выявлению, раскрытию и расследованию преступлений;
* обеспечение исполнения уголовных наказаний;
* руководство органами внутренних дел и внутренними войсками в целях выполнения возложенных на них задач и принятие мер по совершенствованию их деятельности;
* совершенствование нормативной правовой основы деятельности органов внутренних дел и внутренних войск, обеспечение законности в их деятельности;
* совершенствование работы с кадрами, их профессиональной подготовки, обеспечение правовой и социальной защищенности сотрудников и военнослужащих системы Министерства;
* развитие и укрепление материально-технической базы органов внутренних дел и внутренних войск.

## Образ клиента

Клиентами являются жители Новосибирской области, которые хотят иметь легкий способ связи с правоохранительными органами, соблюдать закон и изучать его, иметь возможность быстро обратиться за помощью в экстренной ситуации и т.д.

Статистика преступлений: Ленинский район – за 2021 год было зафиксировано 3 458 эпизодов. После идёт Октябрьский район – 2 324 эпизода, затем Кировский – 2 051, Заельцовский – 1 756, Центральный – 1 593, Калининский – 1 590, Дзержинский – 1 498, Советский – 1 099. На предпоследнем месте Первомайский район – 861 случай кражи. Замыкает антирейтинг Железнодорожный район с 831 случаем. Преступления с причинением тяжкого вреда здоровью чаще всего происходили также в Кировском районе – 43 эпизода. После идёт Ленинский район – 23 преступления, Заельцовский – 30, Первомайский – 17, Советский – 16, Дзержинский – 15, Калининский – 14, Центральный – 6. Самым безопасным районом стал Железнодорожный – всего 3 случая.

## Сценарии

Прежде всего, при составлении сценария выявляется цель, которой будут достигать пользователи. В первую очередь целью является простота и доступность. Например:

Гражданин, ранее служивший в армии и имеющий высшее или среднее профессиональное образование, хочет узнать информацию насчет вакансий на работу полицейским. Эту информацию он может узнать именно в приложении, т.к. оно, помимо всего прочего подразумевает информацию о вакансиях в органах исполнительной власти.

Или же женщина-мать очень беспокоится за жизнь своего ребенка, который каждый день отправляется в школу один. Информацию о разыскиваемых преступниках и их последнем местонахождении она может также узнать, обратившись к приложению.

Можно взять еще одну ситуацию: у девушки есть любимый писатель, который совсем недавно некорректно высказывался в отношении правительства или совершил какие-либо неправомерные действия. Она может обратиться к приложению, чтобы узнать, почему так долго не было новостей от этого писателя: она может воспользоваться поиском по базе данных розыска и, скорее всего, найдет там нужного ей человека.

## Сбор и анализ прототипов

У данного мобильного приложения не так много прототипов. Единственными прототипами являются сайт ГУ МВД Новосибирской области и приложение МВД России.

Ниже приведен рисунок (Рисунок 1) с изображением главной страницы приложения МВД России.

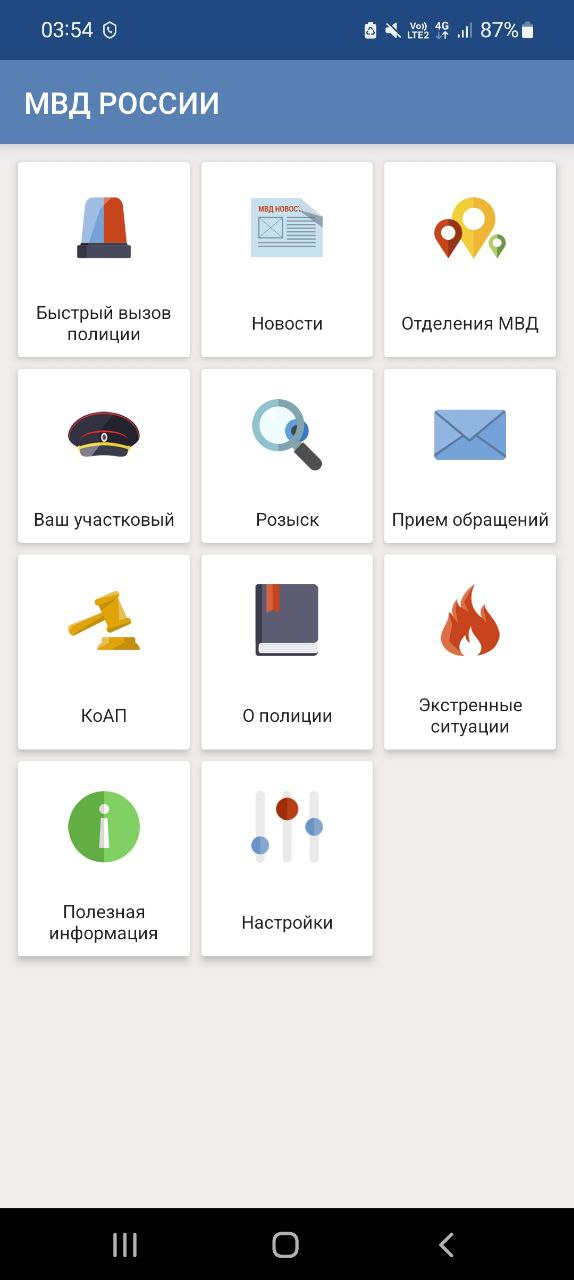


Рисунок 1 – Главная страница

У приложения есть шапка, информационная часть и навигационное меню по разделам. Шапку приложения составляет название приложения (или же официальное название органа). В навигационном меню имеются кнопки для быстрого вызова полиции, новостей, отделений МВД, раздел «Ваш участковый», розыска, приема обращений, КоАП, информация о полиции, экстренных ситуациях, полезные ссылки и ресурсы, а также настройки. Также у приложения имеется боковое меню (при переходе в какой-либо раздел в верхнем левом углу приложения появляется кнопка перехода в это меню). Там располагаются различные разделы для навигации по приложению. Данное меню представлено на рисунке 2 ниже.

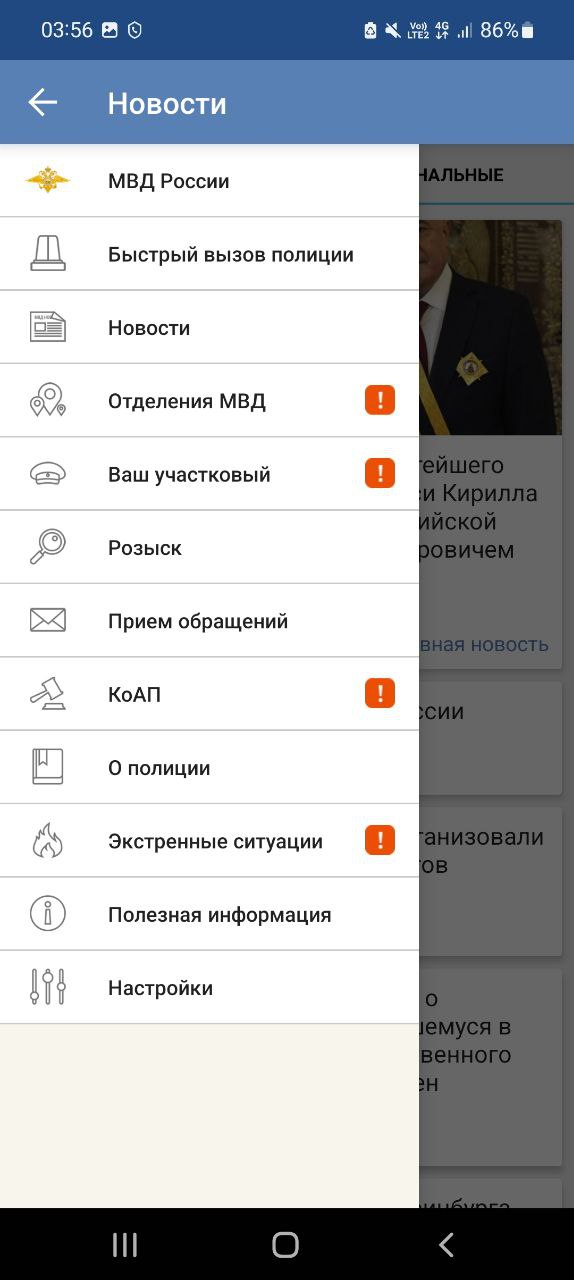


Рисунок 2 – Меню навигации

Внутри каждого раздела располагается какая-либо необходимая информация по теме раздела. К примеру, к разделе «Розыск» можно увидеть страницу с самыми разыскиваемыми преступниками.



Рисунок 3 – Раздел «Розыск»

В этом же разделе мы можем совершить запрос к базе данных розыска, чтобы найти нужно нам человека, который находится в розыске по стране, городу, области или находится в международном розыске. Ниже на рисунок (Рисунок 4) представлена страница заполнения формы на зарос в базе данных розыска.

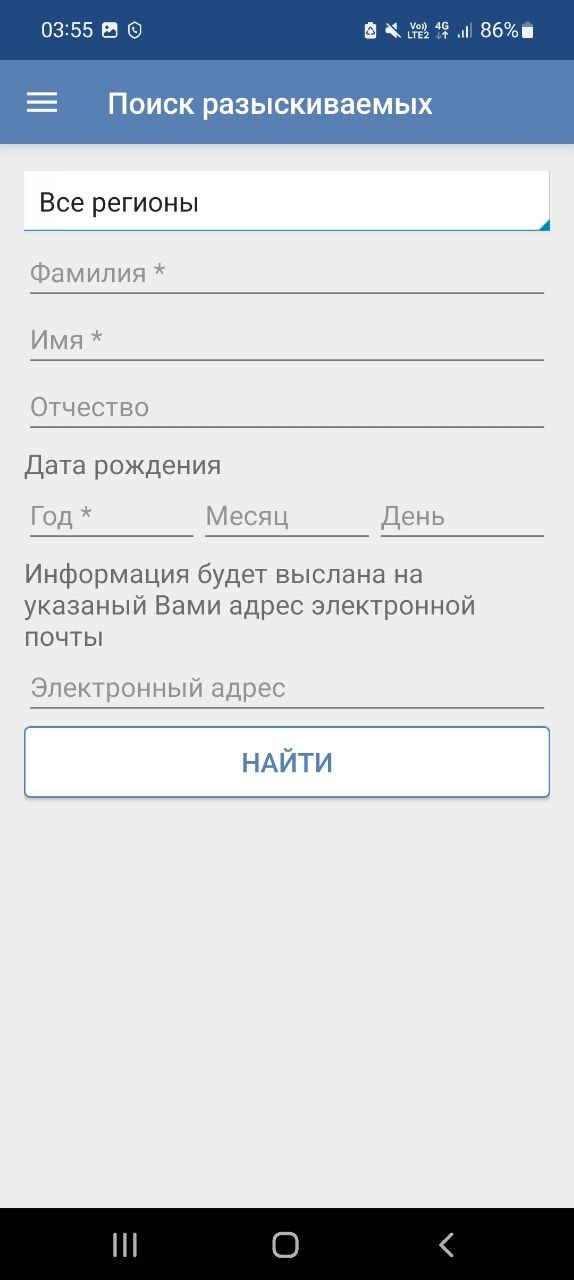


Рисунок 4 – Страница формы запроса

Также есть раздел отделений МВД. Данная страница обрабатывает местоположение пользователя и показывает список ближайших отделений МВД. Данный раздел продемонстрирован на рисунке 5 ниже.

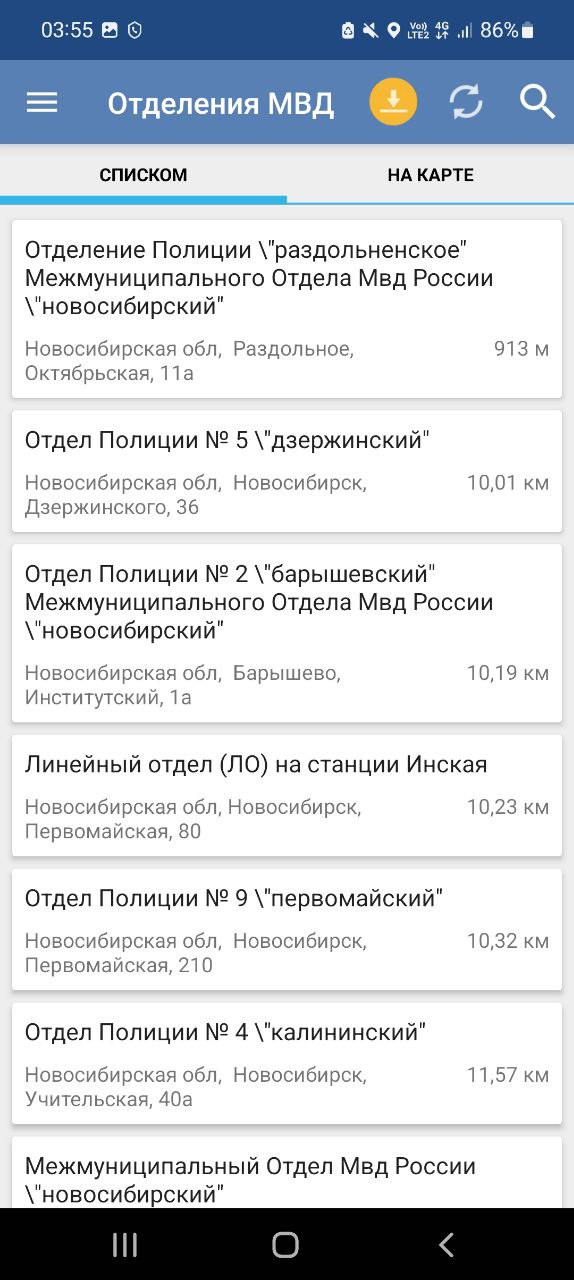


Рисунок 5 – Страница с отделениями

Приложение предусматривает также раздел для новостного блока, где располагаются самые свежие новости от МВД России. На рисунке 6 представлен скриншот данного раздела. Каждая новость располагается в качестве заголовка и его описания в отдельном блоке, где еще указано время и дата публикации новости. Сами новости разгруппированы по типу (федеральные и региональные).

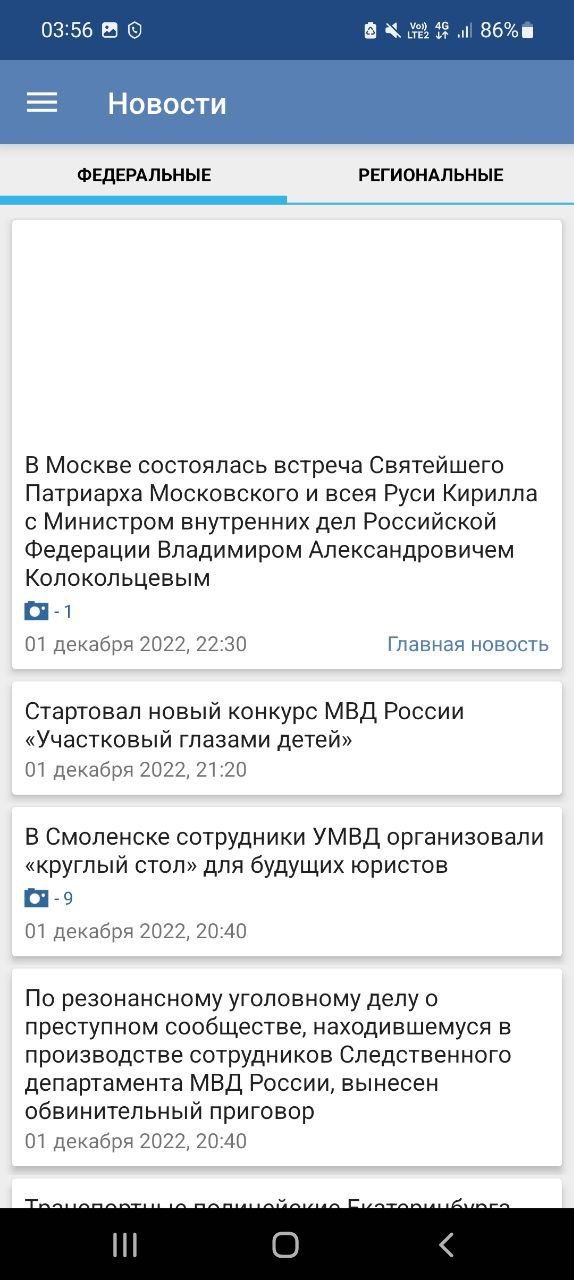


Рисунок 6 – Страница новостей

Исходя из рассмотрения данного приложения можно сделать вывод:

Плюсы приложения:

* оснащенность главными функциями;
* цветовая гамма имеет преимущественно три основных цвета;
* приложение достаточно удобно в использовании и навигации.

Минусы:

* навигация, несмотря на свое удобство, осуществлена немного странно (главное меню с иконками, по сути, используется только единожды, а в основном навигация осуществляется из бокового меню, о существовании которого может не догадываться не особо осведомленный пользователь, поэтому ему придется несколько раз возвращаться назад, в главное меню, чтобы перейти к другому разделу).

В заключении трендами в разработке являются: быстрого вызова полиции и других служб, возможность поиска по базе данных розыска, полезная информация, карта и др.

# Проектирование приложения

## UI/UX дизайн проекта

Дизайн проекта разработан в программе Figma.

Для проекта были определены основные экраны:

* стартовый экран;
* экран главного меню;
* экран для новостного блока и новости;
* экраны для перехода между разделами с информацией (ГУ МВД, для граждан, деятельность, контакты);
* экран с картой;
* экраны для розыска и заполнения формы запроса;
* экран для заполнения формы обращений;
* экран для просмотра графика приема граждан;… (это еще не все)

Цветовая гамма была выбрана посредством изучения сайта ГУ МВД и приложения МВД России. Были выявлены основные цвета, который более знакомо репрезентуют МВД, с какими цветами данный орган исполнительной власти ассоциируется у людей, и какие цвета будут лучше всего подходить для такого официального и формального приложения.

Потому было принято решение в разработке макета отдать предпочтение спокойным цветам: синий, черный, белый и серый. Красным цветом решено было отмечать такую информацию, как номера телефонов или предупреждения.

Ниже на рисунке (Рисунок 6) представлен логотип автосервиса.



Рисунок 6 – Логотип приложения

На логотипе изображен герб ГУ МВД Новосибирской области.

На рисунке (Рисунок 7) ниже представлена главная страница приложения.

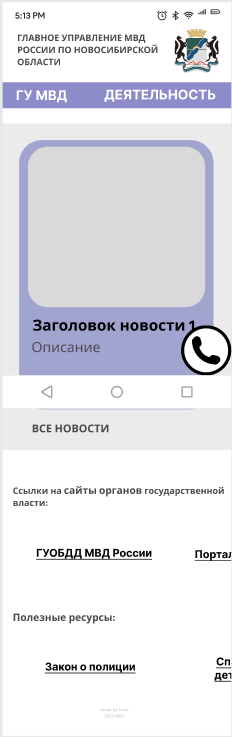


Рисунок 7 – Страница «Главное меню»

На данной странице мобильного приложения предоставлена возможность навигации по разделам приложения (прокручивающееся верхнее меню), использование полезных ссылок и просмотр самых последних новостей.

На рисунке (Рисунок 8) ниже представлена страница новостного блока и внешний вид страницы новости.

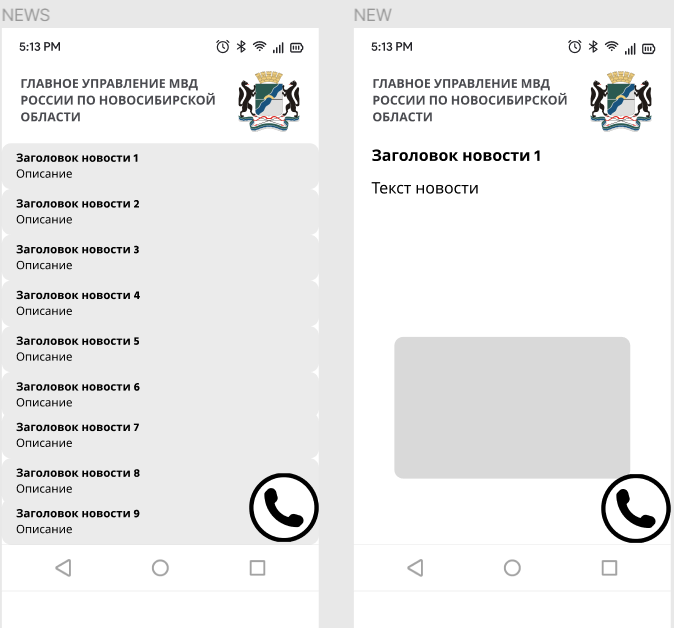


Рисунок 8 – Страницы «Новости» и «Новость»

На данной странице «Новости» имеется возможность перехода между заголовками на страницы новостей.

На рисунке (Рисунок 9) ниже представлены страницы разделов из меню навигации Главной Страницы. Их всего четыре и каждая из них имеет свои разделы.

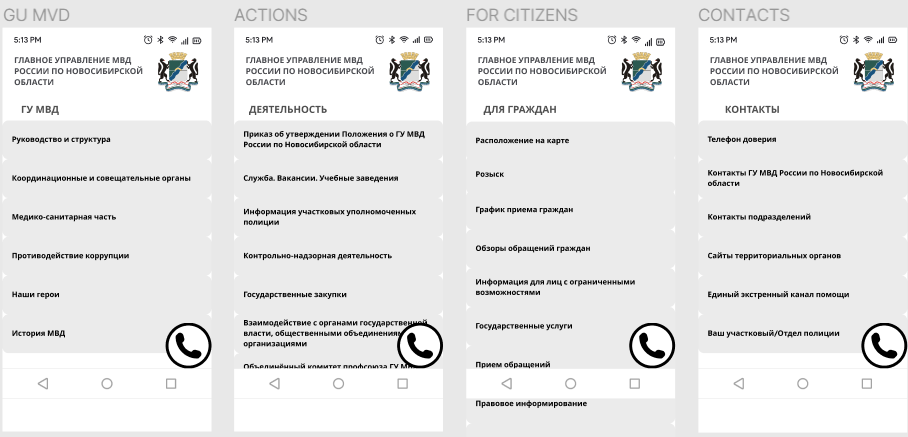


Рисунок 9 – Страницы с разделами меню навигации

На рисунке (Рисунок 10) ниже представлен экран с разделом графика прием граждан.



Рисунок 10 – Страница «График приема граждан»

На данной странице пользователь может ознакомиться с графиком приема граждан в ГУ МВД по НСК и какие сотрудники этот прием осуществляют.

На рисунке (Рисунок 11) ниже представлена страница заполнения формы для запроса к базе данных розыска и сама страница с самыми разыскиваемыми лицами.

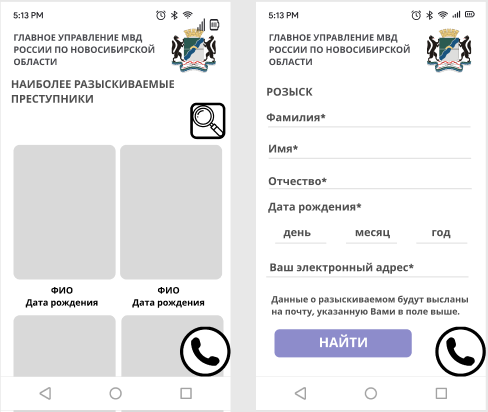


Рисунок 11 – Страницы «Розыск» и запрос

На данных страницах пользователь может просмотреть профили преступников, оформить свой запрос на поиск преступника в базе данных.

На рисунке (Рисунок 12) ниже представлена страница с формой для оформления обращения в органы исполнительной власти.

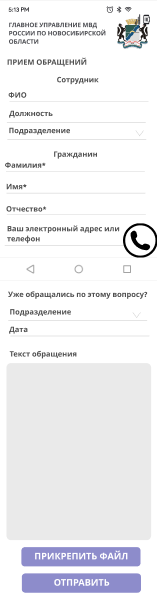


Рисунок 12 – Страница «Прием обращения»

На данной странице пользователь имеет возможность заполнить форму обращения, которое будет отправлено в МВД, после чего сотрудники его изучат и отправят пользователю на него ответ.

На рисунке ниже (Рисунок 13) показан экран с картой.

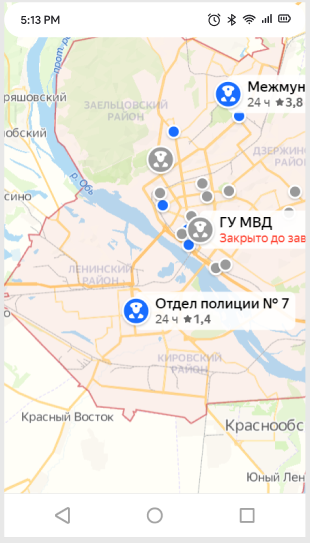


Рисунок 13 – Страница «Карта»

Данный экран позволяет пользователю просматривать отмеченные на ней отделения полиции МВД в ближайшей доступности.

На рисунке ниже (рисунок 14) показан экран с контактными номерами МВД. Каждый номер подсвечен красным цветом. При нажатии на какой-либо из них приложение автоматически набирает этот номер в приложении «Телефон» на устройстве пользователя, что позволяет осуществлять быстрый набор нужных номеров.

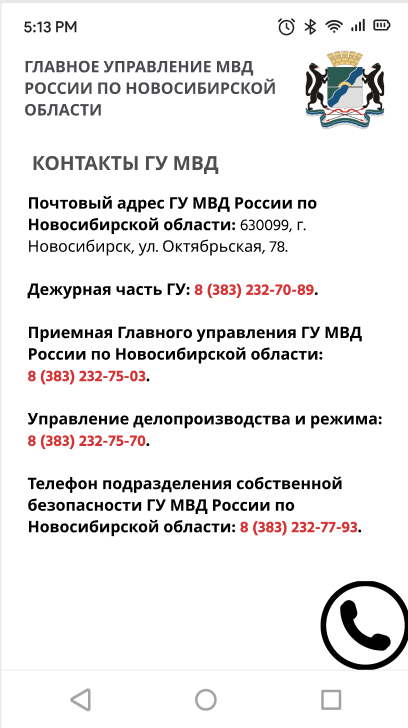


Рисунок 14 – Страница «Контакты»

Также, на каждом из экранов расположена иконка с трубкой телефона – она нужна для того, чтобы быстро вызвать полицию. При нажатии на нее происходит тоже самое, что и при нажатии на номер телефона в контактах: устройство само набирает номер для быстрого вызова. Данная иконка расположена на каждом экране для того, чтобы пользователь мог в любой момент вызвать полицию, когда это необходимо.

## Выбор технологии, языка и среды программирования

Средой программирования выбрана программа Android Studio версии. Языком программирования является Kotlin.

Используемый сервис для базы данных и аутентификации – Firebase.

!!!!ДОПИСАТЬ

# Разработка мобильного приложения

## Разработка базы данных

К проекту был подключён сервис Firebase посредством помощника Firebase в программе Android Studio и прописывания кода в файлах проекта.

С помощью Authentication Firebase была реализована аутентификация пользователей (Рисунок 13), условием которой является наличие пароля и электронной почты у пользователя. Так же при заполнении полей регистрации происходит проверка заполнения: валидация пароля и электронной почты.

Рисунок 13 – Аутентификация пользователей

В ходе анализа были выявлены сущности: запись на приём, чат, пользователи, услуги.

На рисунке (Рисунок 14) ниже представлена таблица «Пользователи», которая включает в себя контакт пользователя и VIN его автомобиля.



Рисунок 14 – Таблица «Пользователи»

На рисунке (Рисунок 15) ниже представлена таблица «Запись на приём», которая включает в себя название услуги, дату записи и VIN обслуживаемого автомобиля.

Рисунок 15 – Таблица «Запись на приём»

## Разработка используемых плагинов

Ниже приведен листинг (Листинг 1), на котором изображено название эмулятора для запуска приложения в Android Studio.



Листинг 1 – Эмулятор

Так же были использованы, как встраиваемые плагины, так и дополнительные. Ниже приведен листинг, (Листинг 2) используемых плагинов.



Листинг 2 – Плагины

Добавленные плагины предназначаются для подключения Firebase, RealTime Database, Auth и навигационного меню.

## Описание разработанных процедур и функций

В приложении разработаны следующие функции:

* регистрация и авторизация;
* переход между страницами с помощью навигационного меню;
* запись на приём;
* просмотр услуг автосервиса;
* появление календаря по нажатию на после с выбором даты;
* чат.

Ниже представлен листинг (Листинг 4) кода, который отвечает за регистрацию пользователя в системе.

public class MainActivity extends AppCompatActivity {  
  
 EditText inputemail, inputpassword, cconformpassword;  
 Button registerBtn, vxodBtn;  
 String emailPattern = "[a-zA-z0-9.\_-]+@[a-z]+\\.+[a-z]+";  
 ProgressDialog progressDialog;  
  
 FirebaseAuth mAuth;  
 FirebaseUser mUser;  
  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 vxodBtn = findViewById(R.id.*vxodBtn*);  
 inputemail = findViewById(R.id. *inputemail*);  
 inputpassword = findViewById(R.id. *inputpassword*);  
 cconformpassword = findViewById(R.id.*cconformpassword*);  
 registerBtn = findViewById(R.id. *registerBtn*);  
 progressDialog = new ProgressDialog(this);  
  
 mAuth = FirebaseAuth.*getInstance*();  
 mUser = mAuth.getCurrentUser();  
  
 vxodBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 startActivity(new Intent(MainActivity.this, login.class));  
 }  
 });  
  
  
 registerBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 PerforAuth();  
 }  
 });  
 }  
  
  
  
 private void PerforAuth() {  
 String email = inputemail.getText().toString();  
 String password = inputpassword.getText().toString();  
 String conformpassword = cconformpassword.getText().toString();  
  
 if(password.isEmpty() || password.length()<6)  
 {  
 inputpassword.setError("Введите пароль, не менее 6 символов");  
 } else if(!password.equals(conformpassword))  
 {  
 cconformpassword.setError("Пароли не совпадают");  
 }  
 else{  
 progressDialog.setMessage("Пожалуйста, подождите");  
 progressDialog.setTitle("Регистрация");  
 progressDialog.setCanceledOnTouchOutside(false);  
 progressDialog.show();  
  
  
  
 mAuth.createUserWithEmailAndPassword(email, password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {  
 @Override  
 public void onComplete(@NonNull Task<AuthResult> task) {  
 if (task.isSuccessful())  
 {  
 progressDialog.dismiss();  
 sendUserToNextActivity();  
 Toast.*makeText*(MainActivity.this, "Регистрация прошла успешно", Toast.*LENGTH\_LONG*).show();  
  
 }  
 else {  
 progressDialog.dismiss();  
 Toast.*makeText*(MainActivity.this, ""+task.getException(), Toast.*LENGTH\_LONG*).show();  
  
 }  
 }  
 });  
 }  
 }  
  
 private void sendUserToNextActivity() {  
 Intent intent = new Intent(MainActivity.this, Home.class);  
 intent.setFlags(Intent.*FLAG\_ACTIVITY\_CLEAR\_TASK*| Intent.*FLAG\_ACTIVITY\_NEW\_TASK*);  
 startActivity(intent);  
 }  
  
  
}

Листинг 4 – Регистрация пользователя

Во время регистрации происходит проверка пароля пользователя на соответствие: пароль состоит не менее из 6 символов и совпадает с повторным ведением пароля. А также, если пользователь с данным адресом электронной почты уже зарегистрирован, пользователь получит предупреждение об этом.

Ниже на листинге (Листинг 5) кода представлен метод перехода на другую активность.

vxodBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 startActivity(new Intent(MainActivity.this, login.class));  
 }  
 });  
  
  
 registerBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 PerforAuth();  
 }  
 });  
}

Листинг 5 – Переход на другие активности

Ниже на листинге (Листинг 6) представлен код, отвечающий за авторизацию пользователя в системе.

public class login extends AppCompatActivity {  
  
 EditText inputemail, inputpassword;  
 Button loginBtn;  
 Button createNewAccount;  
 String emailPattern = "[a-zA-z0-9.\_-]+@[a-z]+\\.+[a-z]+";  
 ProgressDialog progressDialog;  
  
 FirebaseAuth mAuth;  
 FirebaseUser mUser;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_login*);  
  
 inputemail = findViewById(R.id.*inputemail*);  
 inputpassword = findViewById(R.id.*inputpassword*);  
 loginBtn = findViewById(R.id.*loginBtn*);  
 createNewAccount = findViewById(R.id.*createNewAccount*);  
  
 progressDialog = new ProgressDialog(this);  
  
 mAuth = FirebaseAuth.*getInstance*();  
 mUser = mAuth.getCurrentUser();  
  
  
 createNewAccount.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 startActivity(new Intent(login.this, MainActivity.class));  
 }  
 });  
  
 loginBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 perforLogin();  
 }  
 });  
  
  
 }  
  
 private void perforLogin() {  
  
 String email = inputemail.getText().toString();  
 String password = inputpassword.getText().toString();  
  
 if (password.isEmpty() || password.length() < 6) {  
 inputpassword.setError("Введите пароль, не менее 6 символов");  
 } else {  
 progressDialog.setMessage("Пожалуйста, подождите");  
 progressDialog.setTitle("Вход");  
 progressDialog.setCanceledOnTouchOutside(false);  
 progressDialog.show();  
  
 mAuth.signInWithEmailAndPassword(email, password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {  
 @Override  
 public void onComplete(@NonNull Task<AuthResult> task) {  
 if (task.isSuccessful()) {  
 progressDialog.dismiss();  
 Toast.*makeText*(login.this, "Вход выполнен", Toast.*LENGTH\_LONG*).show();  
 }else  
 {  
 progressDialog.dismiss();  
 Toast.*makeText*(login.this, ""+task.getException(), Toast.*LENGTH\_LONG*).show();  
  
 }  
 }  
 });  
 }  
  
 }  
  
 private void sendUserToNextActivity() {  
 Intent intent = new Intent(login.this, Home.class);  
 intent.setFlags(Intent.*FLAG\_ACTIVITY\_CLEAR\_TASK*| Intent.*FLAG\_ACTIVITY\_NEW\_TASK*);  
 startActivity(intent);  
 }  
  
}

Листинг 6 – Авторизация пользователя

Авторизация была выполнена с помощью сервиса Firebase Auth.

# Тестирование

## Протокол тестирования дизайна приложения

Для проведения тестирования дизайна приложение было выбрано устройство Honor 10i c разрешение экрана 2340×1080.

Проверка была проведена на:

* оптимальный размер кнопок;
* читабельный размер шрифта;
* приятное глазу цветовое сочетание;
* корректное расположение элементов;

## Протокол тестирования функционала приложения

Для проверки правильности ввода пароля при регистрации были проведены UnitTests, которые изображены ниже на листинге (Листинг 3):

public class PasswordUtils {  
  
 private static final String *VALID\_PASSWORD\_REGEX* = "((?=.\*\\d)(?=.\*[a-z])(?=.\*[A-Z])(?=.\*[@#$%=:\\?]).{8,12})";  
  
 @Test  
 public void validatePassword\_EmptyString() {  
 *// setup* String password = "";  
  
 *// execute* boolean actual = PasswordUtils.*validatePassword*(password);  
  
 *// assert  
 assertFalse*(actual);  
 }  
  
 @Test  
 public void validatePassword\_Null() {  
 *// setup* String password = null;  
  
 *// execute* boolean actual = PasswordUtils.*validatePassword*(password);  
  
 *// assert  
 assertFalse*(actual);  
 }  
  
 @Test  
 public void validatePassword\_Missing\_OneNumber() {  
 *// setup* String password = "Abcdefg#";  
  
 *// execute* boolean actual = PasswordUtils.*validatePassword*(password);  
  
 *// assert  
 assertFalse*(actual);  
 }  
  
 @Test  
 public void validatePassword\_LengthTooLong() {  
 *// setup* String password = "Abcdefg5#abcdefgabcd";  
  
 *// execute* boolean actual = PasswordUtils.*validatePassword*(password);  
  
 *// assert  
 assertFalse*(actual);  
 }  
  
 @Test  
 public void validatePassword\_Missing\_OneUpperCaseLetter() {  
 *// setup* String password = "abcdefg5#";  
  
 *// execute* boolean actual = PasswordUtils.*validatePassword*(password);  
  
 *// assert  
 assertFalse*(actual);  
 }  
   
  
  
 public static boolean validatePassword(String password) {  
 return (password != null && password.matches(*VALID\_PASSWORD\_REGEX*));  
 }  
}

Листинг 3 – UnitTests

Так же для тестирования функционала были разработаны TestCase.

Test-Case №1

Название тест кейса: Регистрация

Предусловия тест кейса: приложение открыто02

Шаги тест кейса:

* 1. шаг тест кейс №1:открыть страницу регистрации;
  2. шаг тест кейса №2: заполнить все поля;
  3. шаг тест кейса №3: нажать на кнопку «регистрация».

Ожидаемый результат тест кейса:

Пользователь зарегистрирован в системе.

Статус кейса: выполнено.

Test-Case №2

Название тест кейса: Авторизация

Предусловия тест кейса: пользователь зарегистрирован

Шаги тест кейса:

* 1. шаг тест кейс №1: открыть страницу авторизации;
  2. шаг тест кейса №2: заполнить все поля;
  3. шаг тест кейса №3: нажать на кнопку «Вход».

Ожидаемый результат тест кейса:

Пользователь войдёт в систему.

Статус кейса: выполнено.

Test-Case №3

Название тест кейса: Переход на другую активность

Предусловия тест кейса: пользователь вошёл в систему

Шаги тест кейса:

1. шаг тест кейс №1: открыть страницу с навигацией;
2. шаг тест кейса №2: нажать на желаемую кнопку.

Ожидаемый результат тест кейса:

Пользователь перешёл на необходимую ему страницу.

Статус кейса: выполнено.

Заключение

В результате работы было разработано клиентское мобильное приложение для автосервиса.

Во время разработки возникли трудности с подключением сервиса Firebase к проекту, проблема была решена изменением Android на AndroidX. AndroidX – это переработанная библиотека, чтобы сделать имена пакетов более понятными.

В дальнейшем планируется совершенствовать приложение и добавить уже к имеющемуся функционал.

Библиография

1. Android Developers [электронный ресурс]: Документация Android Stusio – https://developer.android.com/docs/
2. Help Center [электронный ресурс]: Документация Java: https://docs.oracle.com/en/java/
3. Автостат [электронный ресурс]: Структура российских автовладельцев по возрасту и маркам https://www.autostat.ru/infographics/40710/
4. Firebase [электронный ресурс]: Документация Firebase - https://firebase.google.com/docs?hl=uk
5. Material Design [электронный ресурс]: Bottom Navigation: https://material.io/components/bottom-navigation/android
6. Git [электронный ресурс]: Документация Git: https://git-scm.com/
7. Figma [электронный ресурс]: Прототипирование проектов https://www.figma.com/

Приложение А

Техническое задание

Министерство образования Новосибирской области

ГБПОУ НСО «Новосибирский авиационный технический колледж

имени Б.С. Галущака»

**разработка мобильного приложения для автосервиса**

НАТКиГ.202100.010.000ПЗ

Выполнил:

студент группы

ПР-20.106

Петрищева А.В.

2022

**Содержание**

Введение 34

1 Назначения разработки 35

2 Требования к мобильному приложению 36

2.1 Требования к функциональным характеристикам 36

2.2 Требования к надёжности 36

2.3 Условия эксплуатации 37

2.4 Требования к составу и параметрам технических средств 2337

2.5 Требования к информационной и программной совместимости 2337

2.6 Требования к защите информации 38

2.7 Требования к маркировке и упаковке 2338

3 Требования к программной документации 2338

4 Технико-экономические показатели 2338

5 Стадии и этапы разработки 23638

6 Порядок контроля и приёмки 2339

**Введение**

Настоящее техническое задание распространяется на разработку информационной системы «Разработка мобильного приложения для автосервиса», используемого для записи на приём, просмотра услуг, чата с оператором.

Наименование приложения: «Garage54».

Краткая характеристика области применения: приложение для автосервиса – полезное оснащение телефона автовладельца. При необходимости пользователь может просмотреть имеющиеся в автосервисе услуги, записаться на приём и обратиться за помощью к оператору.

Условные обозначения и сокращения:

БД – База данных;

ИС – Информационная система;

ОС – Операционная система;

СТО – Станция технического обслуживания.

Основанием для проведения разработки является Протокол №6 от 21 февраля 2022 года.

Наименование темы разработки – «Разработка мобильного приложения для автосервиса».

Условное обозначение темы разработки – «Автосервис – Garage54».

**1** **Назначение разработки**

Основное назначение мобильного приложения заключается в:

* осведомлении клиентов об услугах;
* возможности клиента записи на приём.

Лица, которые могут работать с данной Системой:

пользователь – просматривает услуги, регистрируется, авторизуется, оформляет запись;

администратор – добавляет и редактирует услуги, просматривает записи.

**2 Требования к мобильному приложению**

2.1 Требования к функциональным характеристикам

Требования к составу выполняемых функций:

* регистрация и авторизация пользователей;
* просмотр имеющихся услуг;
* запись на приём;
* чат с оператором.

Входные данные должны быть организованы в виде вводимого, в специальную форму, текста, соответствующего определённому шаблону. Данные, вводимые вручную, проверяются на корректность.

**2.2 Требования к надёжности**

Обеспечение устойчивого функционирования должно выполняться несколькими действиями:

* организация бесперебойного питания оборудования пользователя;
* использование лицензионного программного обеспечения;
* организация стабильного интернет-соединения.

Приложение должно контролировать входную информацию:

* соблюдение типов данных при заполнении полей;
* операции изменения, удаления и сохранения.

Время восстановления после отказа, вызванного сбоем электропитания технических средств, не фатальным сбоем ОС или файловой системы, не должно превышать 15 минут при соблюдении условий эксплуатации технических и программных средств и правильной настройки операционной системы.

**2.3 Условия эксплуатации**

Обслуживание ИС включает в себя:

1. информационное обслуживание – ввод и редактирование информации БД;
2. системное администрирование БД ИС.

**2.4 Требования к составу и параметрам технических средств**

Для работы приложения на устройстве требуется: объем свободной памяти не менее 3 ГБ, версия устройства Android 10 и выше.

**2.5 Требования к информационной и программной совместимости**

Проектирование взаимодействия с файловой системой должно быть выполнено в рамках разработки курсового проекта. При разработке взаимодействия с файловой системой должен быть использован язык программирования Java.

**2.6 Требования к защите информации**

Доступ к информации БД предоставляется только администратору базы данных.

**2.7** **Требования к маркировке и упаковке**

Требования к маркировке и упаковке не предъявляются.

**3 Требования к программной документации**

Состав программной документации должен включать в себя:

* техническое задание;
* проектную документацию.

**4 Технико-экономические показатели**

Экономические преимущества разработки и ориентировочная экономическая эффективность не рассчитывается.

**5 Стадии и этапы разработки**

Таблица 1 – Стадии разработки

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название этапа | Срок, даты | Отчётность |
| 1 | Исследование предметной области | 25.03.2022-28.03.2022 | Пояснительная записка |
| 2 | Выбор моделей, описывающих предметную область | 01.04.2022-03.04.2022 | Пояснительная записка |
| 3 | Разработка технического задания | 03.04.2022-05.04.2022 | Техническое задание |
| 4 | Изучение Java и Firebase | 01.04.2022-  18.04.2022 | Пояснительная записка |
| 5 | Анализ требований и уточнение спецификаций | 14.04.2022 | Спецификации программного обеспечения |
| 6 | Проектирование структуры мобильного приложения, проектирование компонентов (технический проект) | 15.04.2022-29.04.2022 | Схема структурная системы и спецификации компонентов |
| 8 | Кодирование клиентской части | 01.05.2022-20.06.2022 | Программный  Продукт |
| 9 | Тестирование компонентов.  Сборка и комплексное тестирование | 20.06.2022 | Тексты программных компонентов |
| 10 | Разработка программной  документации | 24.06.2022 – 26.06.2022 | Программная  документация |
| 12 | Защита | 27.06.2022 |  |

**6 Порядок контроля и приёмки**

Виды испытаний – защита курсового проекта.

Общее требования к приёмке:

* техническое задание;
* пояснительная записка;
* программный продукт;
* презентация.