

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ  
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных технологий

Кафедра Информационные системы и технологии

Специальность 1-98 01 03 «Программное обеспечение информационной  
безопасности мобильных систем»

Специализация Инженер-программист

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА  
К КУРСОВОМУ ПРОЕКТУ НА ТЕМУ:**

«Мобильное приложение магазина одежды с вышивкой»

Выполнил студент Валеватая Дарья Анатольевна  
(Ф.И.О.)

Руководитель проекта ассистент Уласевич Н.И.  
(учен. степень, звание, должность, Ф.И.О., подпись)

Заведующий кафедрой ст. преподаватель Блинова Е.А.  
(учен. степень, звание, должность, Ф.И.О., подпись)

Консультант: Уласевич Н.И.  
(учен. степень, звание, должность, Ф.И.О., подпись)

Курсовой проект защищен с оценкой \_\_\_\_\_

## Содержание

Введение.....	3
1. Анализ требований к программному средству .....	4
1.1 Обзор аналогичных решений .....	4
1.1.1 Аналог «Wildberries» .....	4
1.1.2 Аналог «Lamoda» .....	5
1.1.3 Аналог «Ostin» .....	6
1.2 Требования к проекту .....	7
1.3 Вывод.....	7
2. Разработка архитектуры проекта .....	8
2.1 Модель базы данных.....	8
2.2 Диаграмма вариантов использования .....	9
2.3 Архитектура приложения .....	10
2.4 Вывод.....	10
3. Реализация приложения .....	11
3.1 Технические средства разработки.....	11
3.2 Подключение базы данных .....	11
3.3 Разработка мобильного приложения .....	12
3.4 Вывод.....	14
4. Тестирование приложения .....	15
4.1 Тестирование клиентской области.....	15
4.2 Тестирование области работы администратора .....	17
4.2 Тестирование интерфейса .....	18
4.3 Вывод.....	19
5. Руководство пользователя.....	20
5.1 Руководство для пользователя .....	20
5.2 Руководство для администратора .....	26
5.3 Вывод.....	28
Заключение .....	29
Список использованных источников.....	30
ПРИЛОЖЕНИЕ А.....	31
ПРИЛОЖЕНИЕ Б .....	32

## Введение

Информационные системы и технологии все увереннее становятся неотъемлемой частью всех сфер жизнедеятельности современного человека. Мобильные приложения играют особую роль и представляют собой важную составляющую современных информационных систем. Все потому, что именно они дают возможность современному человеку жить в комфорте и получать мгновенный доступ к нужной информации.

Интернет-магазины становятся все более востребованными, так как имеют ряд преимуществ: широкий ассортимент, не ограниченный размером помещения, экономия времени, возможность сравнения цен, акций, условий доставки между конкурентами, возможность изучить отзывы для более осознанного выбора и так далее.

Основная задача разработки такого приложения состоит в создании удобного и интуитивно понятного интерфейса для пользователей, позволяющего получать актуальную информацию о совершенных заказах, каталоге товаров, акциях и отзывах.

Одной из важных задач разработки мобильного приложения для интернет-магазина является интеграция с базой данных. Для эффективного хранения и обработки информации о товарах, заказах и пользователях необходимо использовать базу данных. Одним из популярных вариантов баз данных является firebase.

Целью данного курсового проекта является разработка мобильного приложения интернет-магазина. Также будут исследованы различные аспекты такого приложения, включая функциональные возможности, архитектура и пользовательский интерфейс.

В конечном итоге, курсовой проект должен быть завершен выводами по каждому разделу работы и заключением, включающим общий вывод по проделанной работе.

## 1. Анализ требований к программному средству

### 1.1. Обзор аналогичных решений

В настоящее время каждый имеет мобильное устройство и тяжело представляет жизнь без него. Поэтому, чтобы быть востребованным, необходимо иметь достойное приложение, которое заинтересует как можно больше клиентов. На данный момент большинство предпочитает покупать онлайн, так как это доступно, удобно и экономит время.

#### 1.1.1 Аналог «Wildberries»

Первый рассмотренный аналог - это интернет-магазин «Wildberries». Мобильное приложение имеет удобный и интуитивно понятный дизайн, что облегчает навигацию и поиск нужных товаров. С помощью данного программного средства пользователь может ознакомиться со списком доступных для покупки товаров. На рисунке 1.1 отображена страница с товарами.

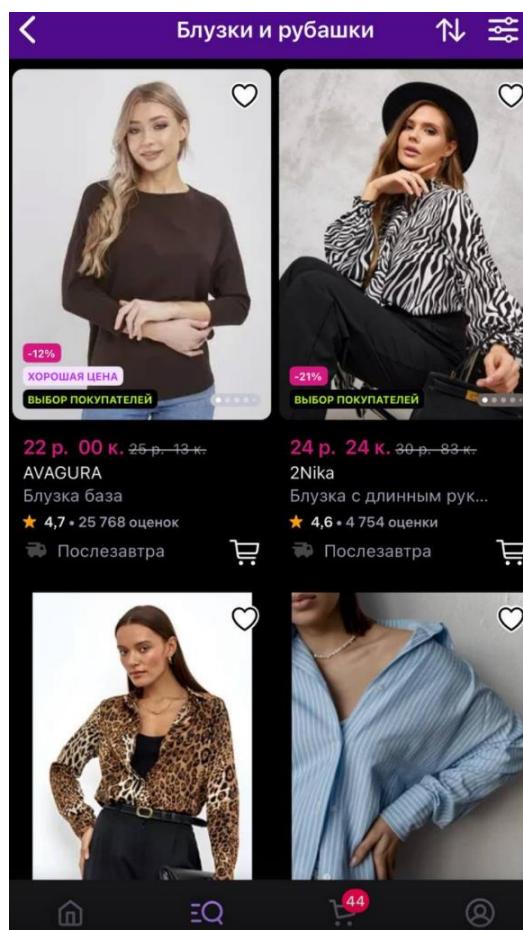


Рисунок 1.1 – Страница с товарами

Пользователь может сортировать, накладывать фильтры на товары, добавлять в избранное или корзину, просмотреть товары с акциями, прочитать отзывы и изучить рейтинг товара.

Приложение позволяет добавлять товар в корзину и оформлять заказ в считанные минуты, сохранив платежные данные и адрес доставки покупателя, первое представленно на рисунке 1.2.

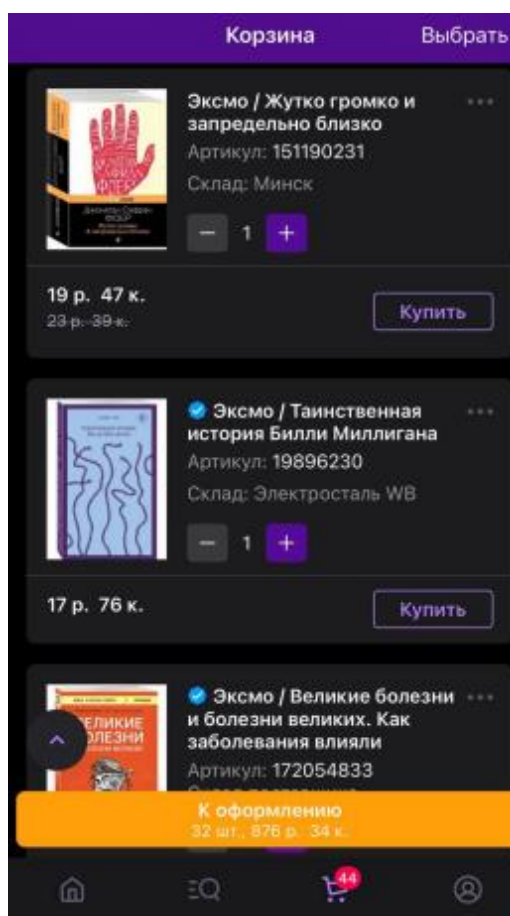


Рисунок 1.2 – Корзина

Приложение позволяет заказать все элементы в корзине сразу, нажав кнопку «К оформлению» или выбрать отдельные товары для заказа.

### 1.1.2 Аналог «Lamoda»

Следующим альтернативным решением является интернет-магазин «Lamoda». Lamoda - это ведущая российская онлайн-платформа в сфере моды и стиля. Основная специализация Lamoda - это продажа одежды, обуви и аксессуаров. Мобильное приложение Lamoda является одним из лидеров российского рынка. Оно насчитывает более 15 миллионов активных пользователей и обеспечивает более 70% всех продаж компании

У приложения широкий функционал, доступный любому пользователю: удобный поиск товара, наличие сортировки по категории, полу, размеру, возможность добавить товар в корзину и так далее. Интерфейс представлен на рисунке 1.3.

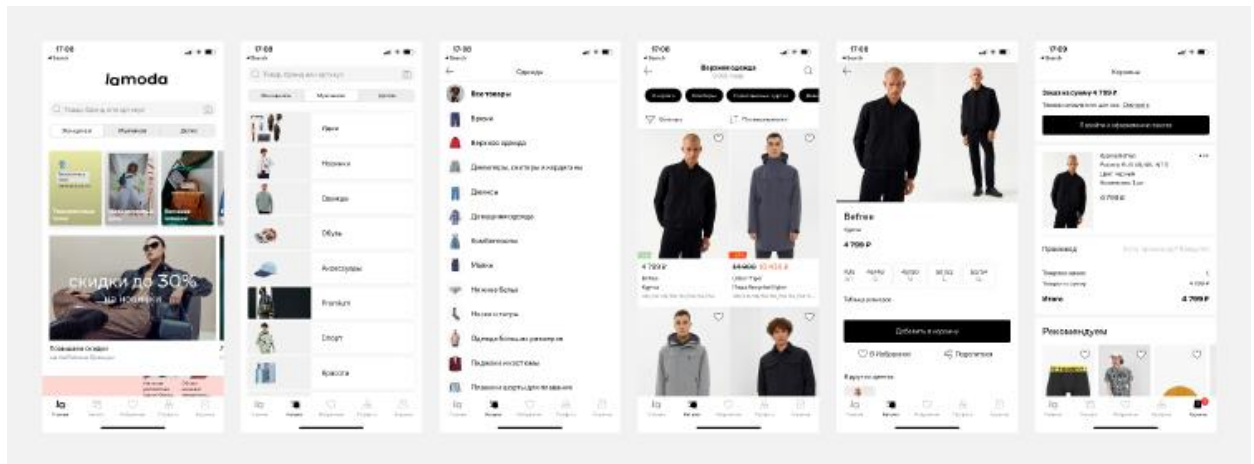


Рисунок 1.3 – Интерфейс приложения «Lamoda»

Кнопки в интерфейсе приложения отличаются заметными размерами и использованием черного цвета. Это делает навигационные элементы легко распознаваемыми и обеспечивает удобство взаимодействия с приложением.

### 1.1.3 Аналог «Ostin»

Третим рассматриваемым конкурентом является интернет-магазин «Ostin». Сейчас Ostin является одним из крупнейших российских брендов повседневной одежды и позиционирует себя как бренд «средней ценовой категории», предлагая доступные по стоимости, но качественные товары.

С помощью данного программного средства пользователь может ознакомиться с каталогом одежды, выбрать для себя подходящий по вкусу товар. Интерфейс представлен на рисунке 1.4.

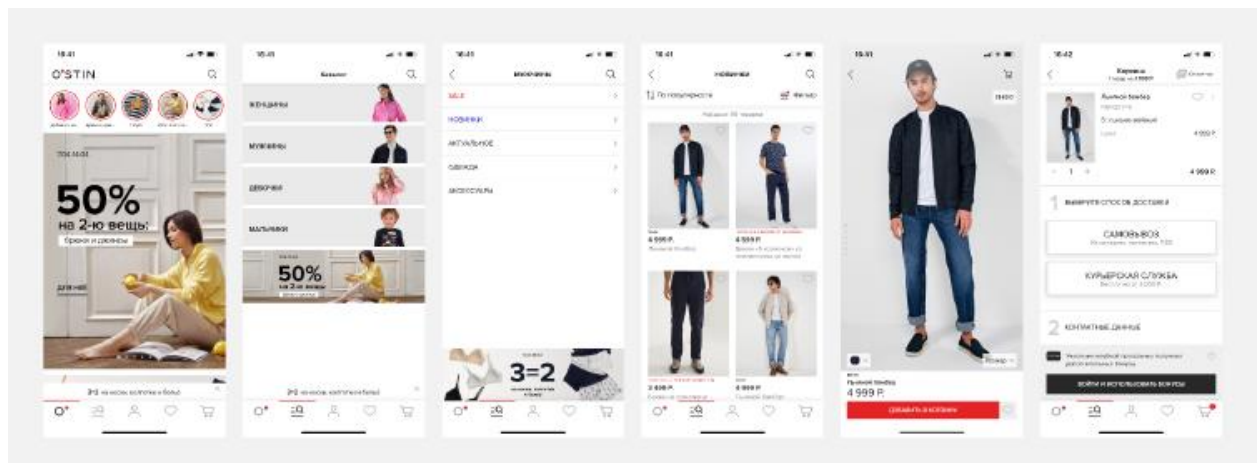


Рисунок 1.4 – Интерфейс приложения «Ostin»

Для оформления заказа нужно иметь собственный аккаунт, интерфейс страницы регистрации представлен на рисунке 1.5.



Рисунок 1.5 – Интерфейс страницы регистрации

Одним из главных плюсов является выполненный в одном стиле дизайн.

## 1.2 Требования к проекту

Программное средство должно обеспечивать возможность выполнения перечисленных ниже функций:

- управление администратором товарами;
- возможность пользователю зарегистрироваться или войти в существующую учетную запись;
- возможность просмотра каталога магазина;
- возможность использования поиска товаров;
- возможность пользователям добавления\удаления товара в корзину;
- возможность просмотра личной страницы пользователя и редактирование при необходимости;
- оформление заказа и при необходимости отмена.

## 1.3 Вывод

Были исследованы аналоги приложения, что позволило выявить ошибки и подчеркнуть успешные приемы. Приложение должно предоставлять возможность авторизации и регистрации, поиск добавление товаров в корзину и удаление из нее при необходимости, оформление заказа.

## 2. Разработка архитектуры проекта

### 2.1 Модель базы данных

Для достижения поставленных задач была создана база данных Firebase. Firebase – это облачная платформа разработки приложений, предоставляемая компанией Google. Она предлагает разработчикам комплексный набор инструментов и сервисов для создания и управления приложениями.

Структура базы данных представлена на рисунке 2.1.

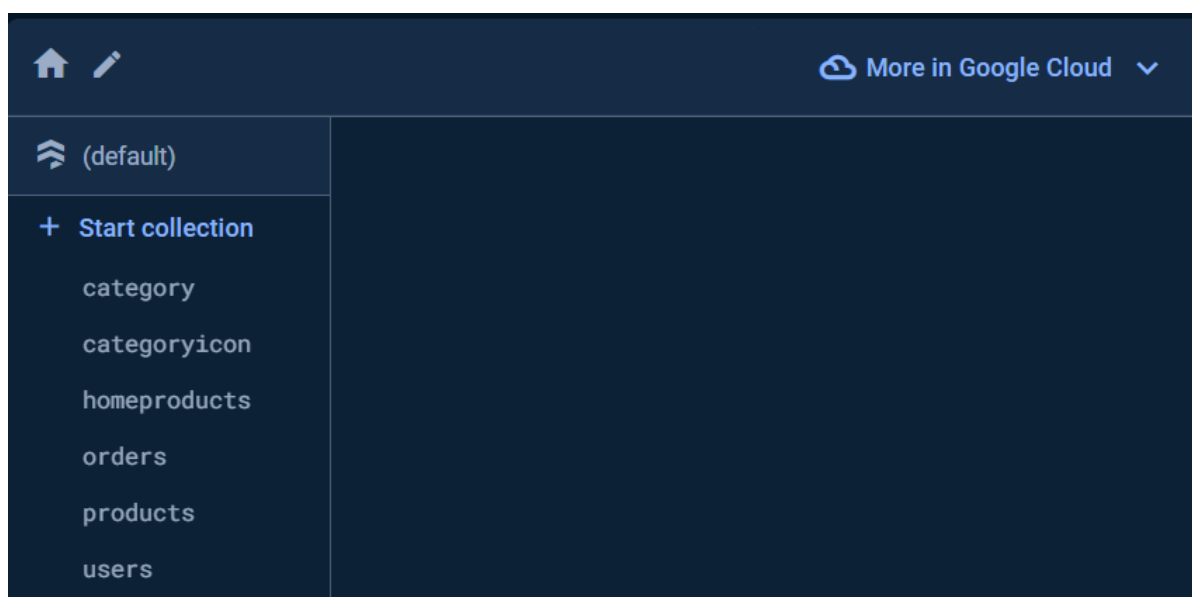


Рисунок 2.1 – Структура базы данных

Созданная база данных состоит из 6 коллекций: category, categoryIcon, homeproducts, orders, products, users.

В состав коллекции users входят следующие столбцы:

- Поле id. Хранит uid зарегистрированного пользователя.
- Поле username. Хранит имя пользователя.
- Поле email. Хранит электронную почту пользователя.
- Поле password. Хранит пароль пользователя.
- Поле phone. Хранит телефон пользователя.
- Поле address. Хранит адрес доставки пользователя.

Данная коллекция используется для хранения информации о зарегистрированных пользователях.

В состав коллекции products входят следующие столбцы:

- Поле name. Хранит название продукта.
- Поле description. Хранит описание продукта.
- Поле price. Хранит цену продукта.
- Поле image. Хранит ссылку на изображение, хранящееся в Firebase Storage.

Данная коллекция используется для хранения информации о доступных товарах.

В состав коллекции categoryIcon входят следующие столбцы:



- Поле name. Хранит название категории.
- Поле image. Хранит ссылку на изображение, хранящееся в Firebase Storage.

Данная коллекция используется для хранения изображения и названия иконок.

В состав коллекции categories входят 3 коллекции: shirts, hoodies, sweatshirts, каждая из которых содержит следующие столбцы:

- Поле name. Хранит название продукта.
- Поле description. Хранит описание продукта.
- Поле price. Хранит цену продукта.
- Поле image. Хранит ссылку на изображение, хранящееся в Firebase Storage.

Данная коллекция используется для хранения информации о продуктах по категориям.

В состав коллекции orders входят следующие столбцы:

- Поле orderItems. Хранит информацию о заказанных товарах.
- Поле date. Хранит дату оформления заказа.
- Поле isCompleted. Хранит состояние заказа.
- Поле totalPrice. Хранит общую стоимость заказа.
- Поле userEmail. Хранит почту пользователя.
- Поле userId. Хранит uid пользователя.
- Поле userName. Хранит имя пользователя.
- Поле userAddress. Хранит адрес пользователя.

Данная коллекция используется для хранения информации о заказах, которые сделал пользователь.

В состав коллекции homeproducts входят следующие столбцы:

- Поле name. Хранит название продукта.
- Поле description. Хранит описание продукта.
- Поле price. Хранит цену продукта.
- Поле image. Хранит ссылку на изображение, хранящееся в Firebase Storage.

Данная коллекция используется для хранения информации о товарах, которые отображаются на главном экране.

Таким образом были рассмотрены все сущности базы данных, используемые в разрабатываемом приложении.

## 2.2 Диаграмма вариантов использования

Диаграмма вариантов использования (Use Case Diagram) служит для визуализации функциональности системы с точки зрения ее пользователей. В рамках данного приложения выделяются две ключевые роли: администратор и пользователь, каждая из которых обладает своими характерными функциями. На рисунке 2.2 представлена диаграмма, которая показывает какие функции будет выполнять администратор и что может делать пользователь.



### 3. Реализация приложения

#### 3.1 Технические средства разработки

Для реализации курсового проекта использован язык программирования Dart и фреймворк Flutter.

Flutter - это кросс-платформенный фреймворк с открытым исходным кодом, разработанный компанией Google. Flutter предоставляет обширную библиотеку готовых виджетов, которые можно использовать для построения интерфейса приложения.

Dart - это объектно-ориентированный, открытый язык программирования, разработанный компанией Google. Он используется в качестве основного языка для разработки мобильных приложений с помощью фреймворка Flutter.

Firebase - это облачная платформа разработки приложений, предоставляемая компанией Google. Она включает в себя следующие ключевые компоненты:

Firebase Authentication: Сервис аутентификации, который обеспечивает безопасную регистрацию и аутентификацию пользователей в приложении.

Cloud Firestore: Гибкая и масштабируемая база данных в реальном времени для хранения и синхронизации данных.

Firebase Storage: Сервис хранения файлов, который позволяет загружать и скачивать файлы, такие как изображения, видео и документы.

Таким образом, данное приложение будет построено с использованием следующего стека технологий: Dart, Flutter, Firebase (Authentication, Cloud Firestore, Storage), а также других вспомогательных библиотек.

#### 3.2 Подключение базы данных

Для работы с базой данных в файл main.dart необходимо написать код, представленный на листинге 3.1.

```
Future<void> main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp( );
  runApp(const MyApp());
}
```

Листинг 3.1 – Подключение базы данных

Перед добавления вышенаписанного кода в файл pubspec.yaml. Здесь прописано необходимые библиотеки для работы с Firebase, использования коллекций и использования аутентификации.

```
firebase_auth: ^4.19.3
cloud_firestore: ^4.17.1
firebase_storage: ^11.7.4
```

Листинг 3.2 – Библиотеки в файле pubspec.yaml

При помощи данных библиотек и кода, написанного выше, была подключена база данных Firebase для разрабатываемого приложения.

### 3.3 Разработка мобильного приложения

Структура проекта представлена на рисунке 3.1.

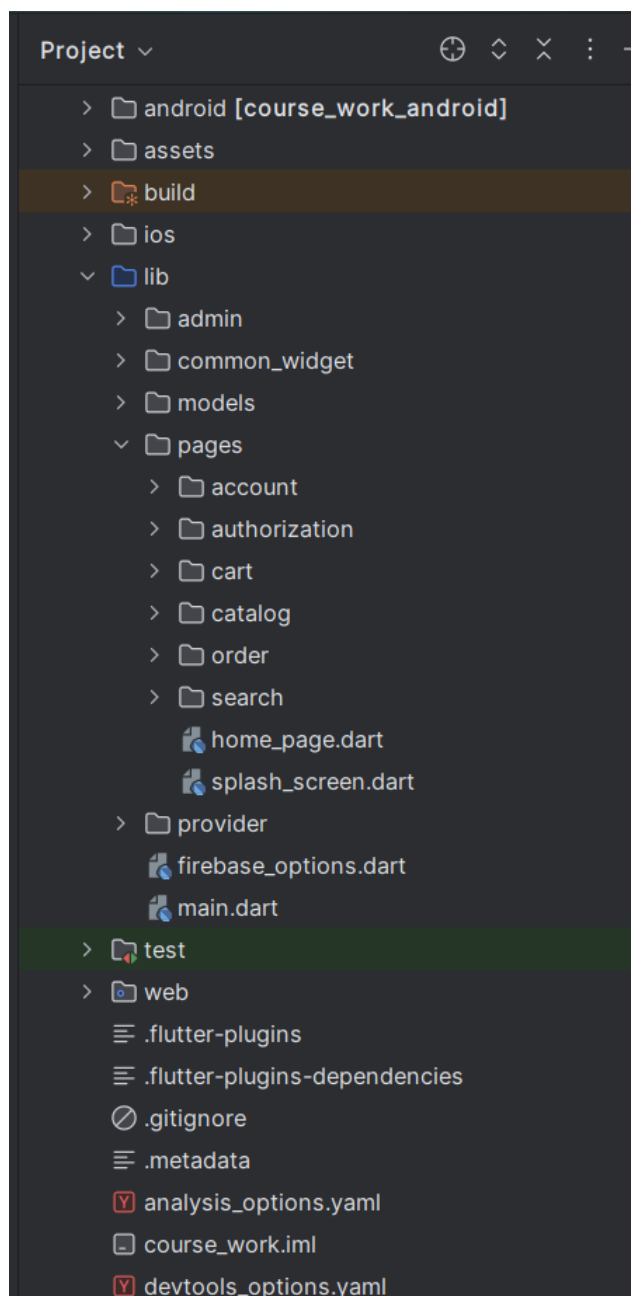


Рисунок 3.1 – Файловая структура приложения

Рассмотрим подробнее файловую структуру. Первая папка `admin` отвечает за страницы административной части, таких как добавление, редактирование и просмотр. Представления на рисунке 3.2.

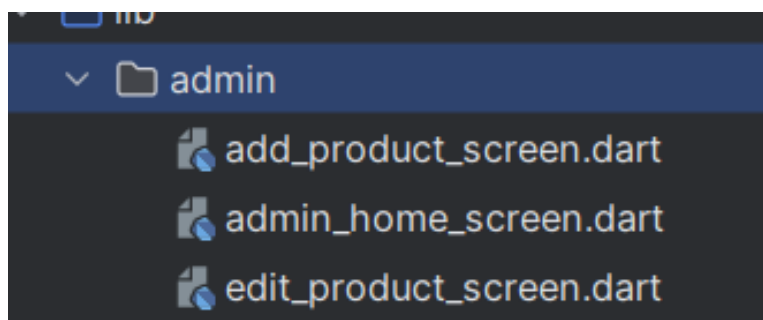


Рисунок 3.2 – Директория admin

Далее в директории `common_widgets` собраны кастомные виджеты такие, как навигационное меню, поле для ввода данных, карточка товара и так далее, которые встречаются в проекте несколько раз. Представлено на рисунке 3.3.

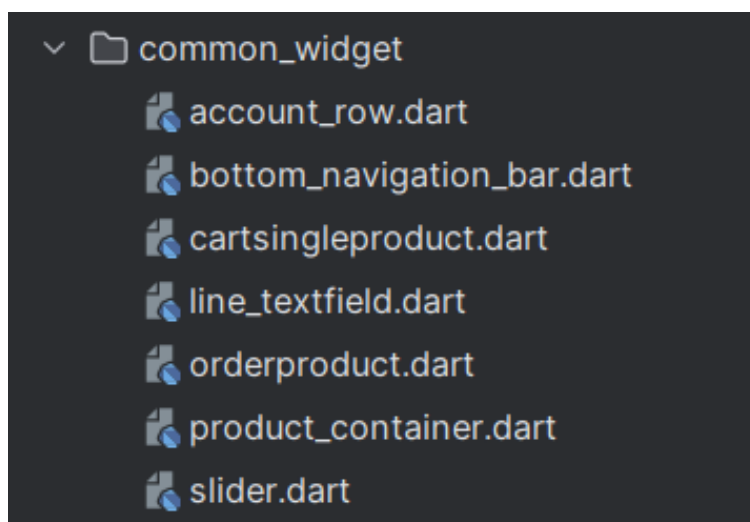


Рисунок 3.3 – Директория common\_widgets

Директория `models` содержит все необходимые модели для работы с базой данных, представлено на рисунке 3.4.

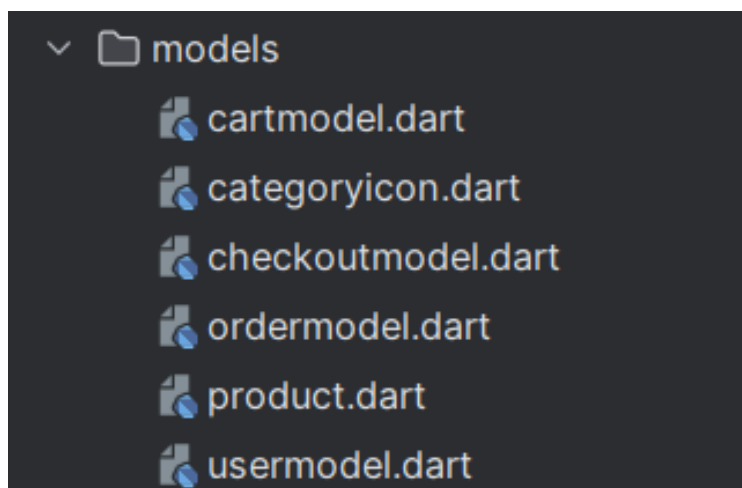


Рисунок 3.4 – Директория models

Директория pages содержит реализацию всех страниц пользовательского приложения, а именно главная страница, каталог, корзина, заказы, профиль и тд. Представлено на странице 3.5.

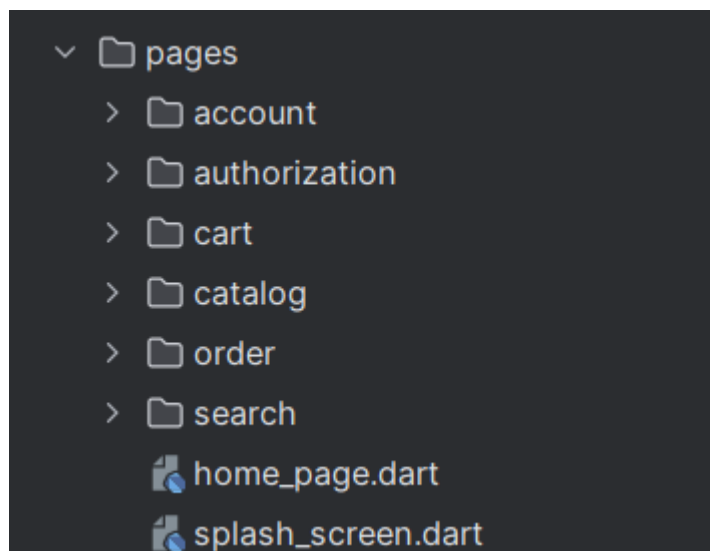


Рисунок 3.5 – Директория pages

В папке provider содержатся виджеты для работы с коллекциями базы данных Firebase, они обеспечивают получение, добавление, удаление, редактирование, поиск и сортировку элементов коллекции. Представлено на рисунке 3.6.

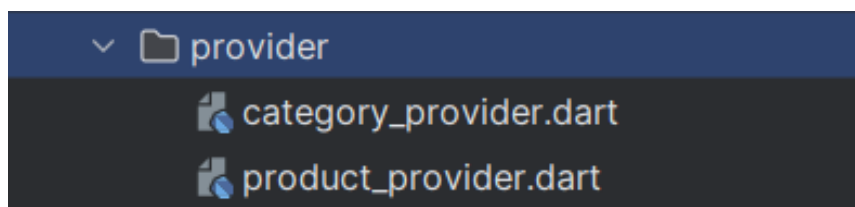


Рисунок 3.6 – Директория providers

И входной точкой приложения является файл main.dart. В этом файле содержится код, который запускает приложение и устанавливает начальную точку выполнения. Он включает в себя инициализацию основных компонентов приложения, настройку маршрутизации и вызов функции, которая запускает основной цикл событий Flutter.

### 3.4 Вывод

В данном разделе были рассмотрены аспекты разработки мобильного приложения. Были описаны технические средства, используемые для разработки.

## 4. Тестирование приложения

### 4.1 Тестирование клиентской области

На этом этапе рассмотрим, как работает приложение, если возникают исключительные ситуации при взаимодействии с клиентом.

Чтобы войти в систему, клиенту необходимо авторизоваться, то есть ввести логин и пароль. Возможно ситуация, когда пользователь вводит некорректные данные или не вводит их вовсе. Пример обработки пустого поля пароля при попытке входа в аккаунт представлен на рисунке 4.1.

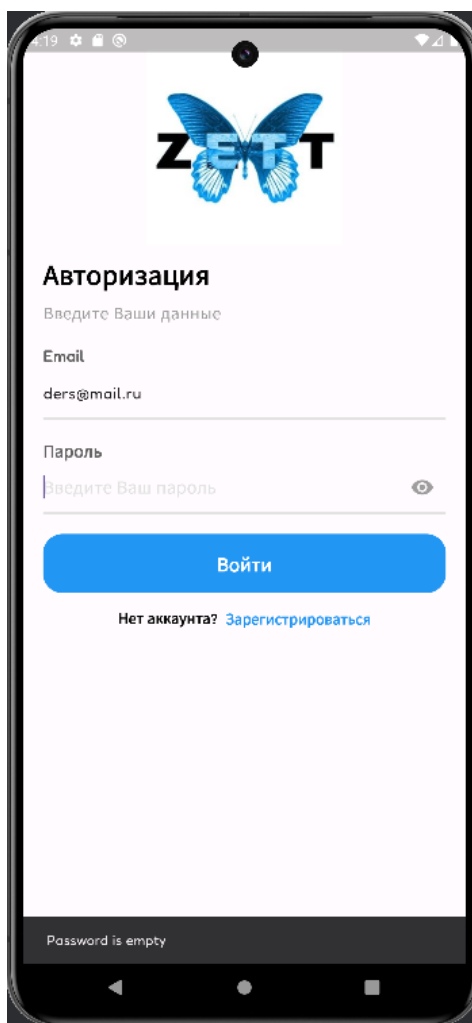


Рисунок 4.1 – Обработка пустого поля пароля

Также для валидации электронной почты используется регулярное выражение, которое проверяет соответствие адреса электронной почты определенному шаблону. Регулярное выражение обычно включает в себя проверку на наличие символа "@", проверку на допустимые символы в имени пользователя и в доменной части, а также может учитывать другие требования, такие как наличие точки в доменной части

Пример обработки ввода неверного логина или пароля при попытке входа в аккаунт представлен на рисунке 4.2. Если пользователь вводит неверные учетные данные, на экране должно отображаться сообщение об ошибке, предлагающее повторить попытку входа с корректными данными.

Это дополнение описывает, как система должна реагировать на ввод неверных учетных данных, чтобы обеспечить понятное и удобное взаимодействие с пользователем.

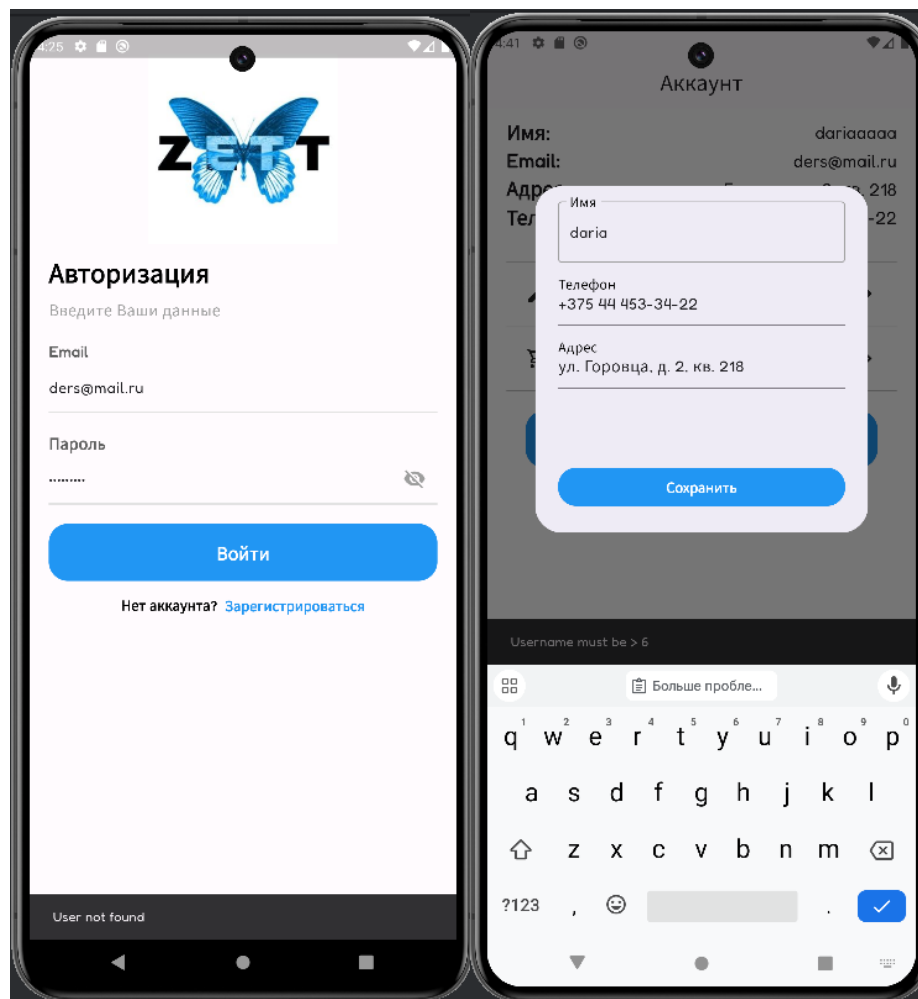


Рисунок 4.2 – Обработка неверно введенных данных

При регистрации пользователь должен заполнить все поля обязательно. Пример обработки ввода неправильного повторного пароля при регистрации представлен на рисунке 4.3. Если пароли, введенные пользователем в поля "Пароль" и "Повторите пароль" не совпадают, на экране должно отображаться сообщение об ошибке, предлагающее пользователю повторно ввести пароли, пока они не будут совпадать.

Это дополнение описывает, как система должна обрабатывать ввод неправильного повторного пароля, чтобы помочь пользователю успешно пройти процесс регистрации. При регистрации пользователь должен заполнить все поля



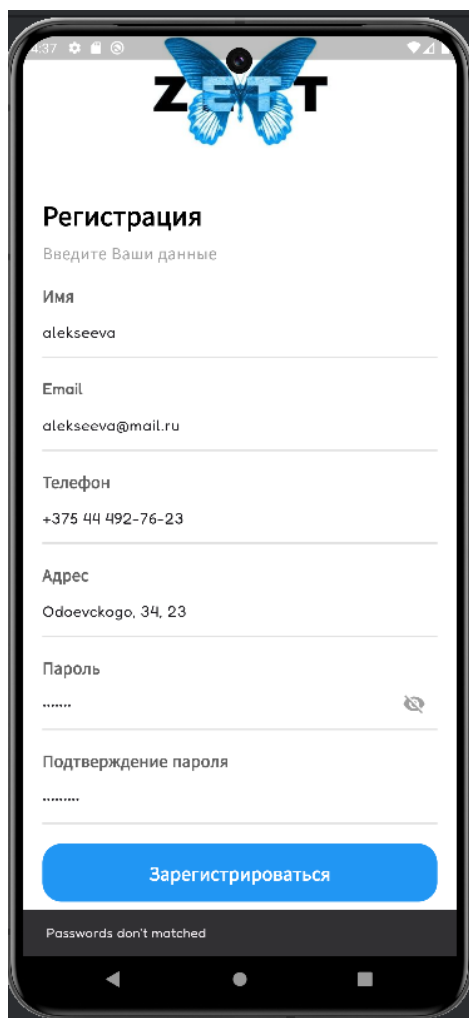


Рисунок 4.3 – Обработка регистрации при несовпадении паролей

Для отображения пароля имеется соответствующая иконка, при нажатии на которую пароль будет отображен.

Больше проблем, с которыми может столкнуться пользователь при взаимодействии с приложением, нет.

## 4.2 Тестирование области работы администратора

Администратор имеет возможность управления товарами, а именно удаление, добавления и редактирование товара, поэтому возможно возникновение исключительных ситуаций.

Если администратор пытается отправить форму с незаполненными обязательными полями, система должна отображать сообщение об ошибке, указывая, какие именно поля необходимо заполнить, чтобы успешно завершить операцию.

Цена, которую хочет ввести администратор, обязательно должна быть целым числом.

На рисунке 4.7 для наглядности показано, как реагирует приложение на данную ситуацию.

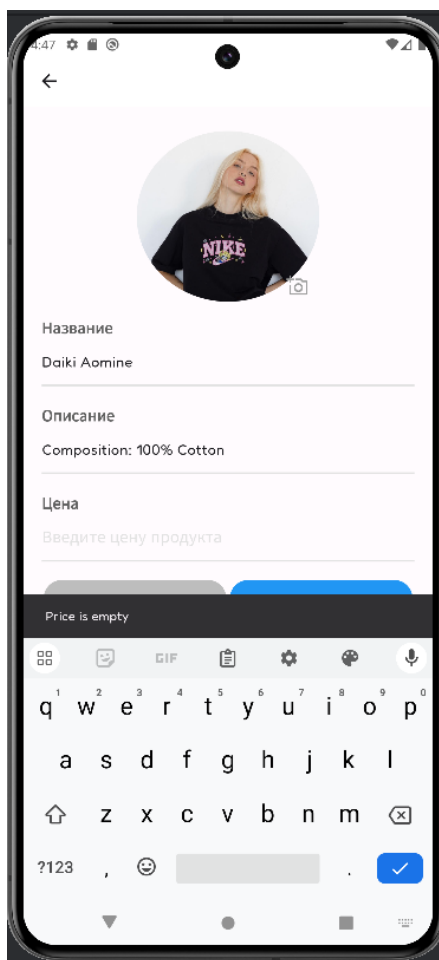


Рисунок 4.7 – Обработка исключения при не всех заполненных полях формы и при не выбранном изображении

### 4.3 Тестирование интерфейса

Протестируем адаптивность мобильного приложения. Для этого сравним то, как выглядит приложение при различных ориентациях экрана.

На рисунке 4.8 представлена страница каталога товаров при горизонтальной ориентации.

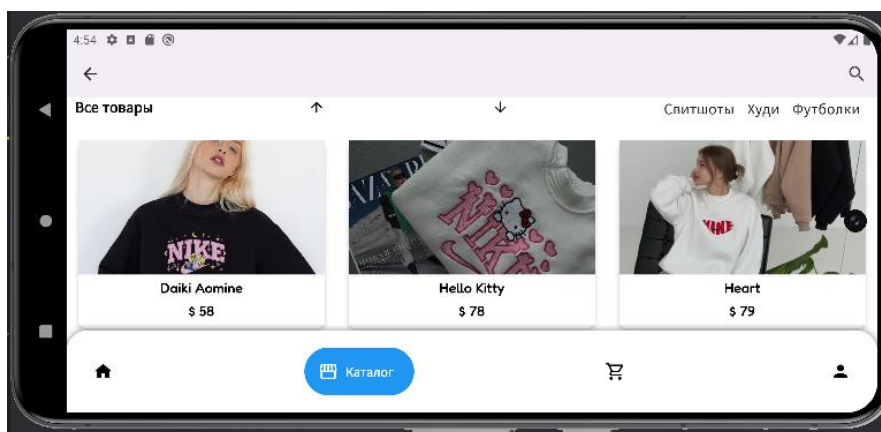


Рисунок 4.8 – Приложение при горизонтальной ориентации

На рисунке 4.9 представлена страница каталога товаров при вертикальной ориентации.

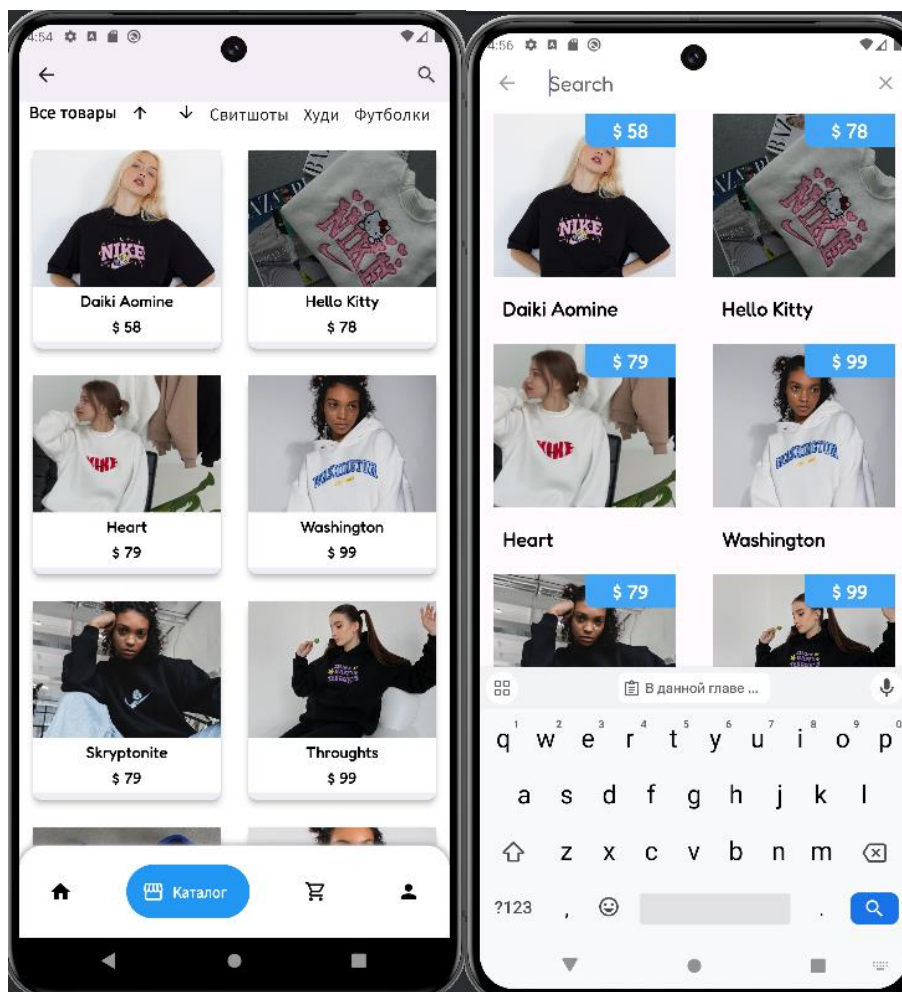


Рисунок 4.9 – Приложение при вертикальной ориентации

На представленном выше рисунке видно, что интерфейс также прекрасно адаптируется и в горизонтальной ориентации устройства

#### 4.4 Вывод

В это разделе было протестировано приложение с базой данных в исключительных ситуациях как со стороны клиента, так и администратора. В итоге приложение является полностью работоспособным.

## 5. Руководство пользователя

### 5.1 Руководство для пользователя

При входе в приложение отображаются страницы для авторизации и регистрации, выполнение представлено на рисунке 6.7.

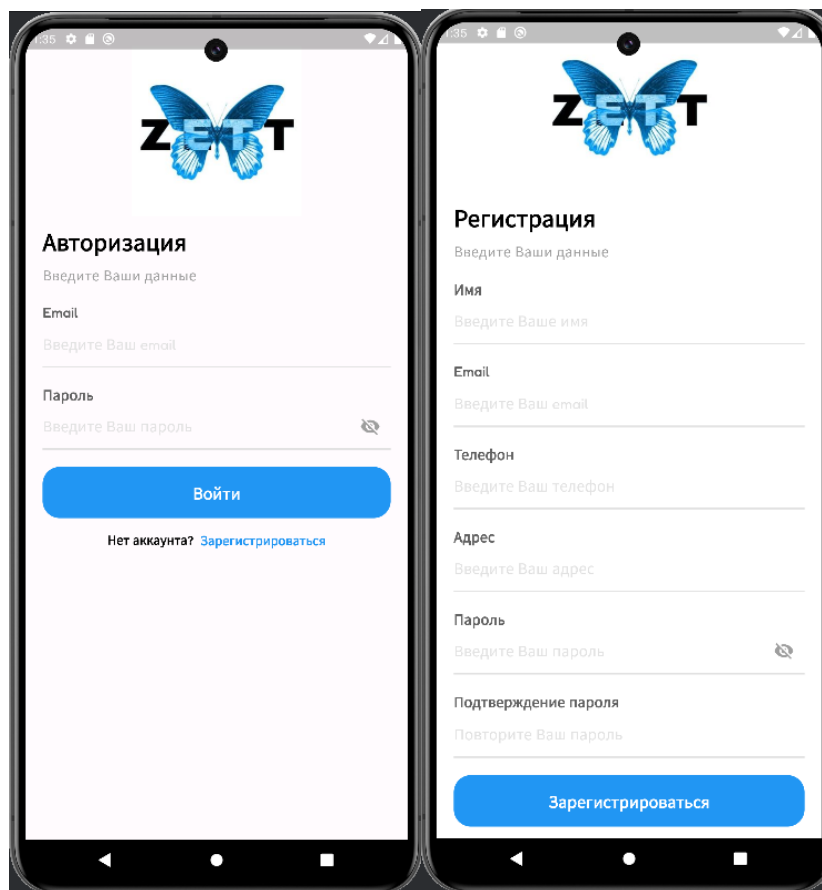


Рисунок 5.1 – Авторизация и регистрация пользователя

После чего пользователь переходит на страницу «Главная», представленной на рисунке 5.2, где присутствует слайдер с актуальными акциями, есть возможность перейти к товарам по категориям или в каталог все товаров, нажав на кнопку «Каталог», выделенную синим цветом на экране или с помощью меню, каталог и категории представлены на рисунке 5.3.

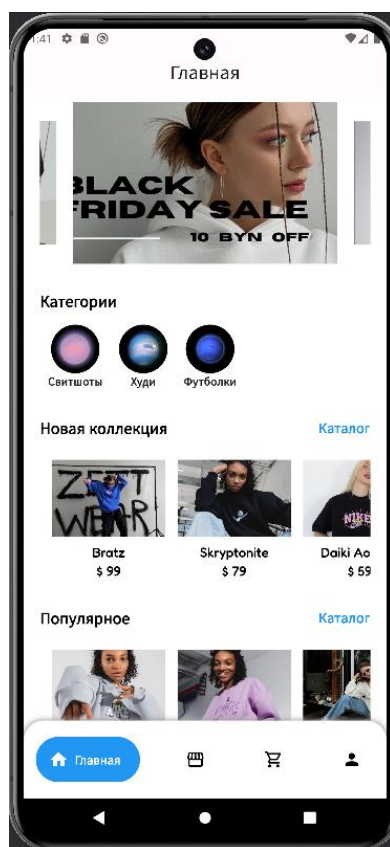


Рисунок 5.2 – Страница «Главная»

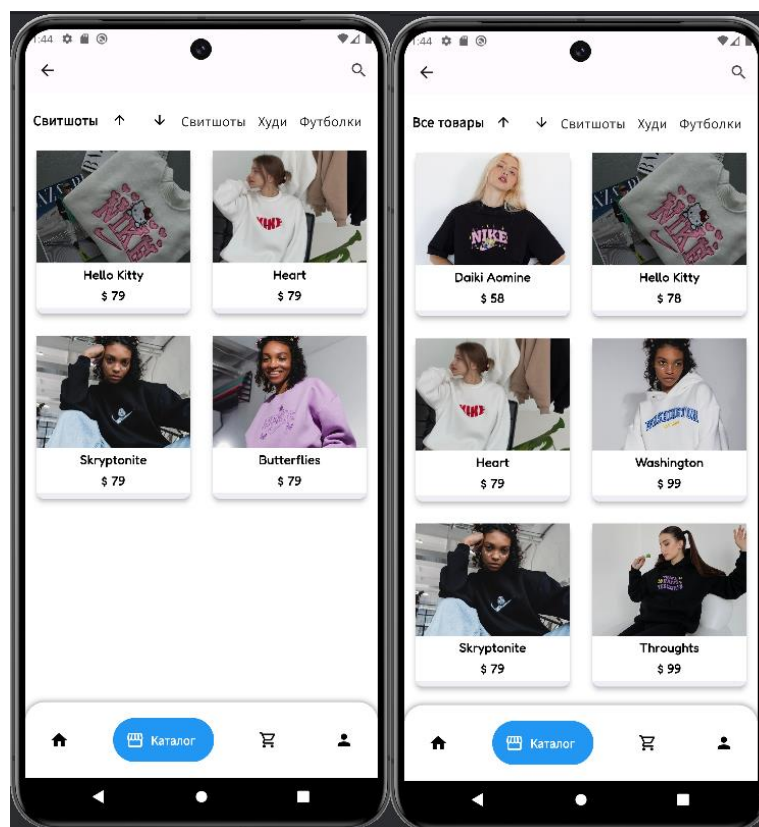


Рисунок 5.3 – Страница «Каталог»

На страницах присутствует возможность сортировки по цене и фильтрации по категориям. А также присутствует поиск как всего ассортимента, так и по определенной категории, представлено на рисунке 5.4.

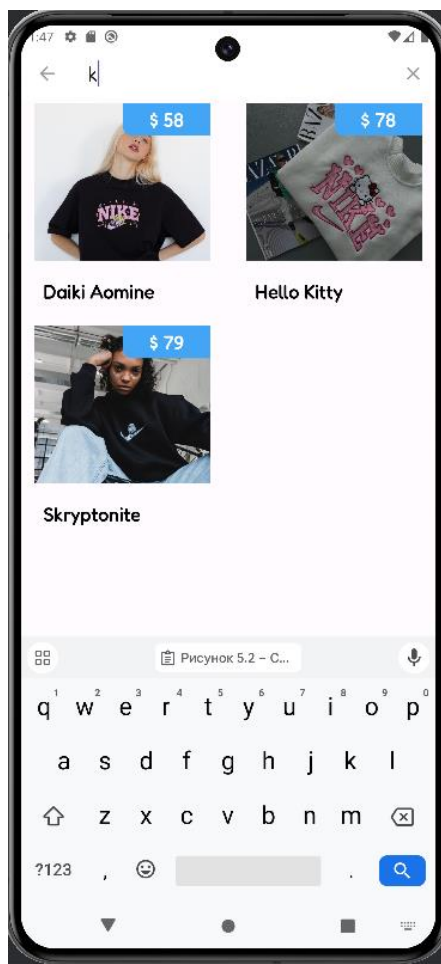


Рисунок 5.4 – Результат поиска

При нажатии на товар пользователь переходит на страницу с более подробным описанием продукта, где может указать размер одежды и желаемое количество для заказа, представлено на рисунке 5.5.

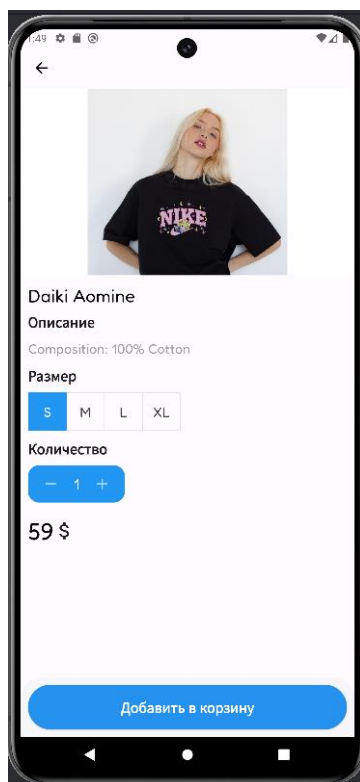


Рисунок 5.5 – Страница товара

Далее для добавления товара в корзину необходимо нажать соответствующую кнопку, после чего пользователь переходит на страницу «Корзина», где может регулировать размер одежды.

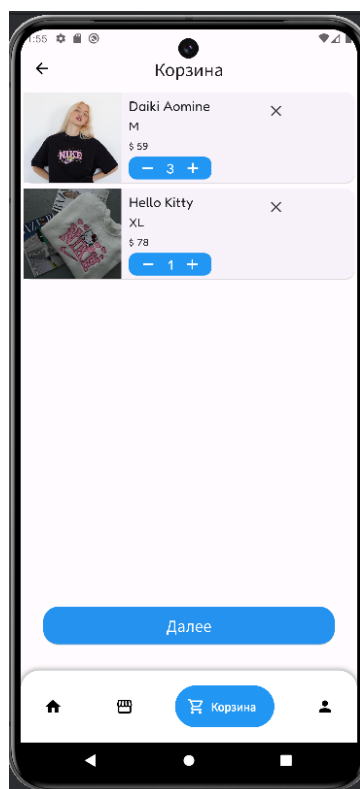


Рисунок 5.6 – Страница «Корзина»

При необходимости пользователь может удалить товар из корзины, нажав соответствующую иконку для удаления, или перейти к следующему этапу оформления заказа, нажав на кнопку «Далее». Страницы «Корзина» представлена на рисунке 5.6.

После перехода к следующему этапу оформления заказа пользователь переходит на страницу, представленную на рисунке 5.7, где видит описание товара, который он хочет приобрести, итоговую сумму заказа и при необходимости может удалить товар, нажав на крестик. Для оформления заказа нужно нажать на соответствующую кнопку.

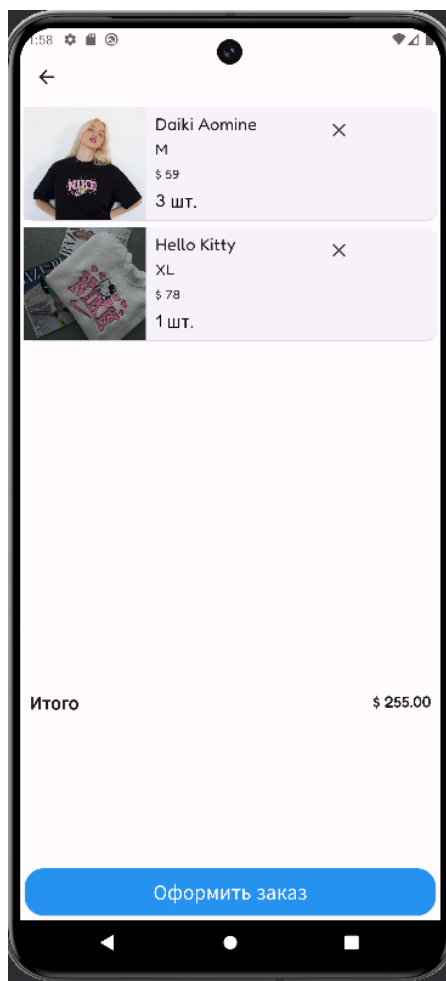


Рисунок 5.7 – Проверка заказа

После нажатия на кнопку «Оформить заказ» у пользователя запросить подтверждения действия, в случае если его ответ отрицателен, то вернуться на страницу назад, а если ответ будет положителен, то заказ будет оформлен.

После оформления заказа пользователь может зайти на страницу «Мои заказы» через профиль аккаунта и проверить свой заказ и при необходимости отменить заказ. Страница представлена на рисунке 5.8.



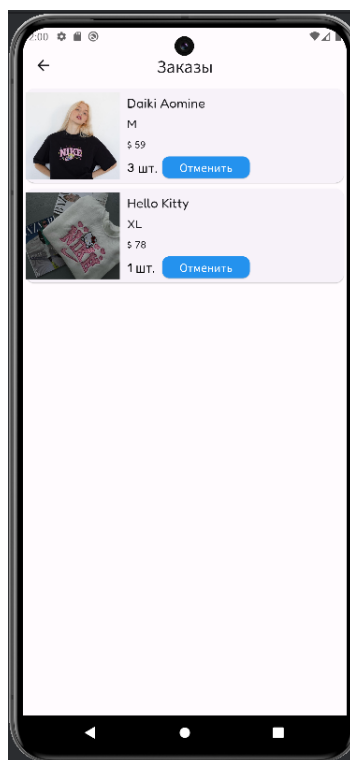


Рисунок 5.8 – Страница «Мои заказы»

У пользователя есть профиль аккаунта, представленный на рисунке 5.9, где выводится вся информация о нем, при необходимости пользователь может редактировать свои данные, нажав на виджет «Редактировать».

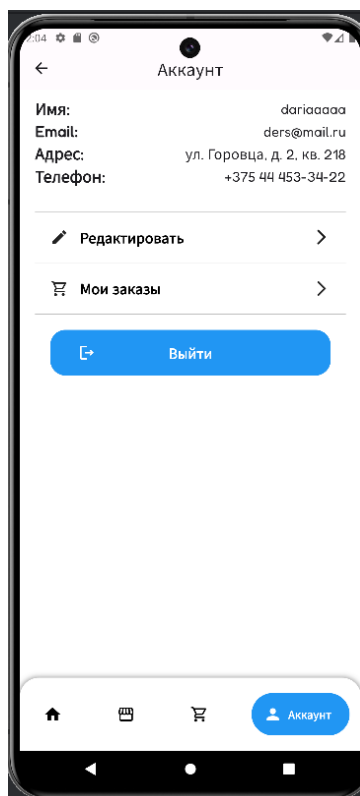


Рисунок 5.9 – Страница «Аккаунт»

При нажатии на кнопку «Выйти» пользователь выходит из своего аккаунта и переход на страницу авторизации.

## 5.2 Руководство для администратора

Задача администратора в данном курсовом проекте – управление товарами, поэтому при авторизации в приложении администратор переходит на страницу с товарами, представленную на рисунке 5.10.

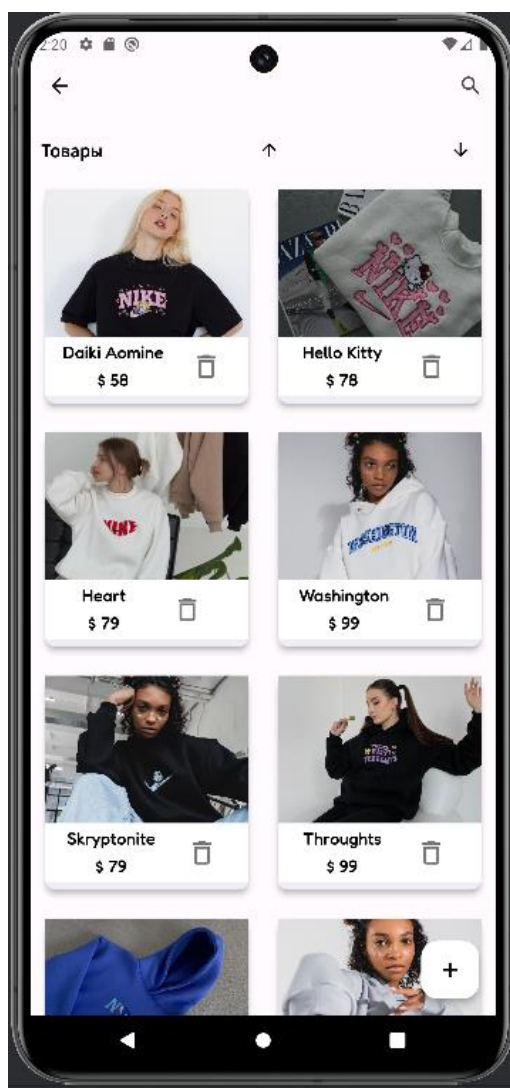


Рисунок 5.10 – Страница администратора

У администратора есть возможность осуществлять поиск товара, удалять товар, нажав на соответствующую иконку удаления, а также добавление товара нажав на кнопку «+» в нижней части экрана. При нажатии на «+» администратор переходит на страницу, представленную на рисунке 5.11.

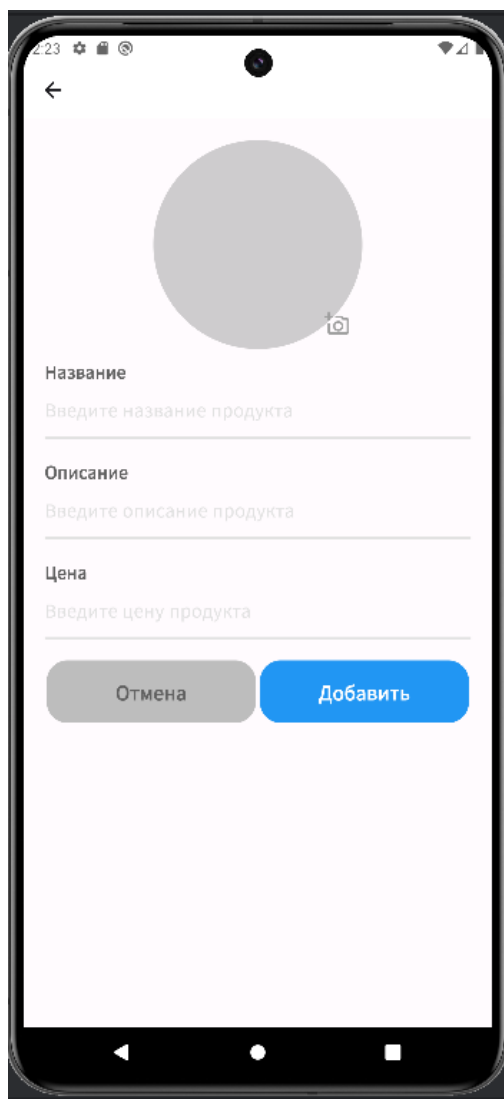


Рисунок 5.11 – Страница добавления товара

Администратор должен заполнить все поля и добавить фото из галереи. После чего нажать кнопку «Добавить» и новый товар отобразится в каталоге товаров. При нажатии на кнопку «Отмена» товар не будет добавлен и администратор перейдет на страницу каталога товаров.

В каталоге товаров при нажатии на определенный товар администратор имеет возможность редактировать товар(поменять фото, описание и цену). Страница редактирования представленная на рисунке 5.12.

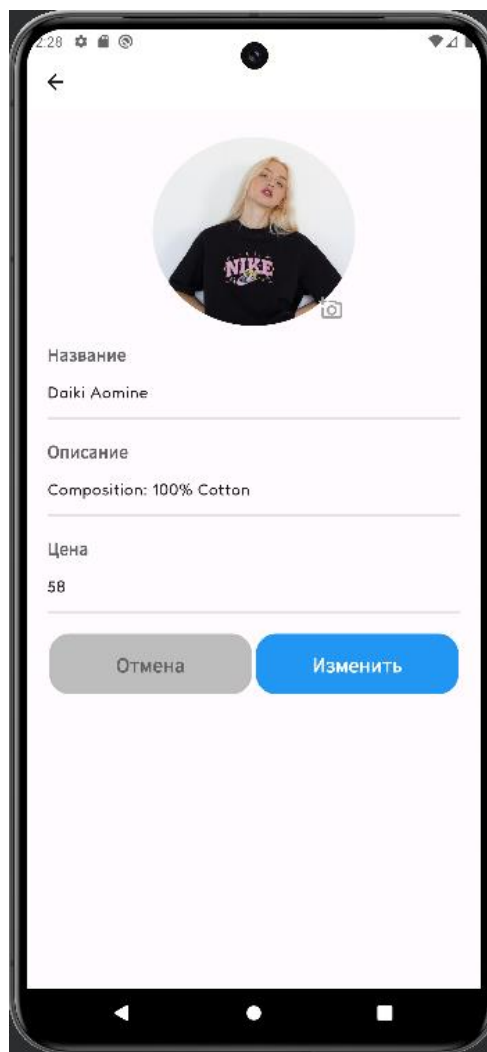


Рисунок 5.12 – Страница редактирования товара

После чего нажать кнопку «Изменить» и обновленный товар отобразится в каталоге товаров. При нажатии на кнопку «Отмена» товар не будет изменен и администратор перейдет на страницу каталога товаров.

### 5.3 Вывод

В данном разделе было представлено подробное руководство пользователя. Все компоненты работают исправно. В ходе выполнения курсового проекта были выполнены основные задачи и требования.

## Заключение

Результатом выполнения курсового проекта стало Flutter приложение для магазина одежды с вышивкой. В ходе курсовой работы была разработана база данных Firebase», состоящая из коллекций, связанных с темой проекта. Приложение является средством взаимодействия клиента с базой данных.

Были реализованы основные требования, указанные в листе задания вместе со следующими пунктами:

- регистрация и авторизация пользователей;
- возможность редактирования профиля;
- поиск, сортировка, фильтрация товаров;
- добавление в корзину и удаление из нее;
- оформление и отмена заказа;

Проект не может окончательно претендовать на полноценный коммерческий продукт, поскольку для демонстрации технологии и работы с базой данных были опущены некоторые атрибуты, реализация которых не относится к основной теме курсового проекта.

Протестировав все компоненты приложения, можно прийти к заключению, что все основные требования выполнены и приложение работает исправно.

### **Список использованных источников**

1. Разработка приложения на Flutter с нуля [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/articles/594963/> – Дата доступа: 02.03.2024.
2. Документация по компонентам MaterialUI [Электронный ресурс]. – Режим доступа: <https://m3.material.io/components>. – Дата доступа: 02.04.2024.
3. Документация о Firebase Authentication [Электронный ресурс]. – Режим доступа: <https://firebase.google.com>. – Дата доступа: 16.03.2024.
4. Статья о использовании Firestore [Электронный ресурс]. – Режим доступа: <https://medium.com/firebase-tips-tricks>. – Дата доступа: 30.03.2024.
5. Flutter Documentation [Электронный ресурс]. – Режим доступа: <https://docs.flutter.dev/>. – Дата доступа: 01.04.2024.
6. Firebase Documentation [Электронный ресурс]. – Режим доступа: <https://firebase.google.com/>. – Дата доступа: 29.04.2024.

## ПРИЛОЖЕНИЕ А

## Диаграммы вариантов использования

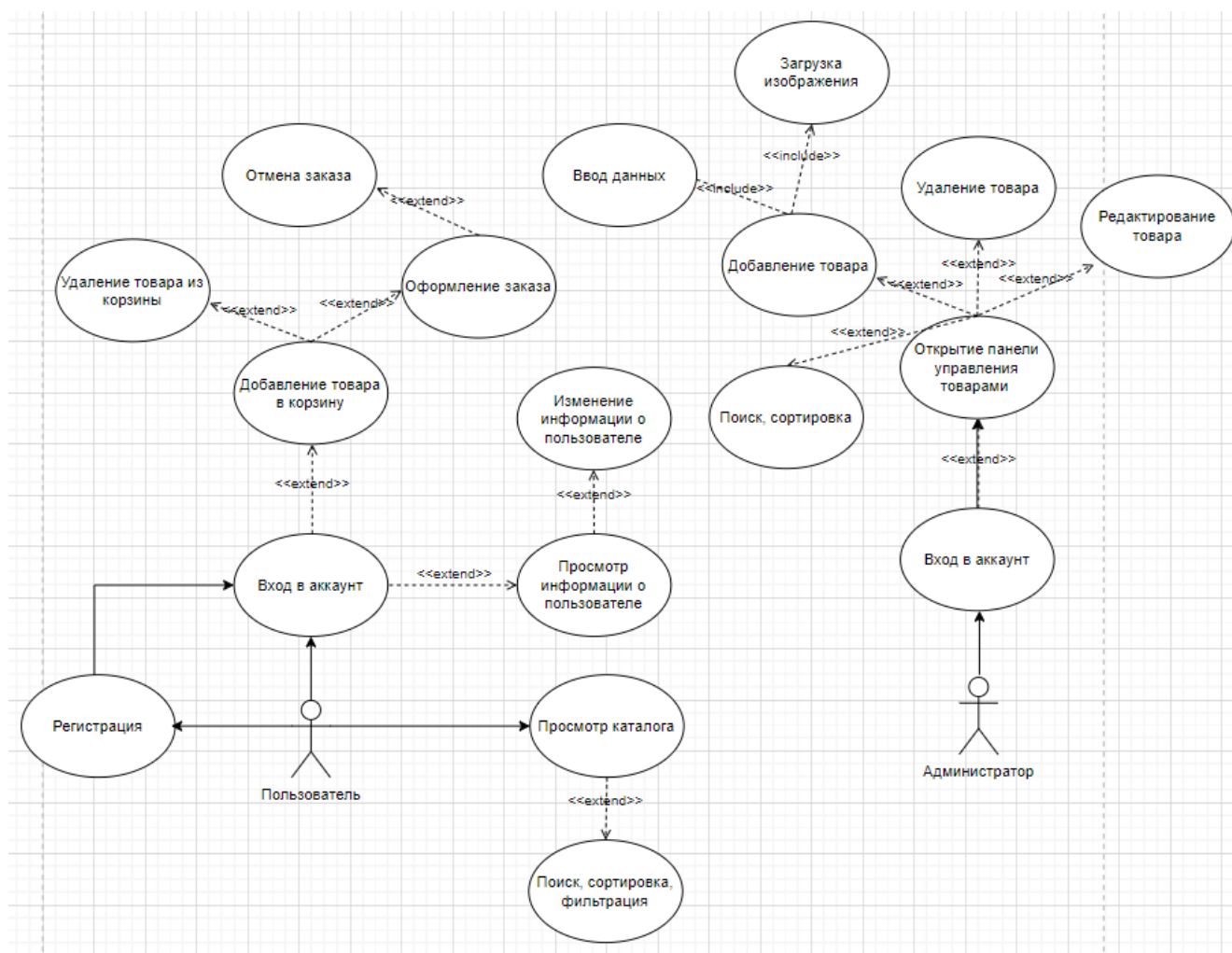


Рисунок 1 – Диаграмма вариантов использования

## ПРИЛОЖЕНИЕ Б

### Диаграмма компонентов

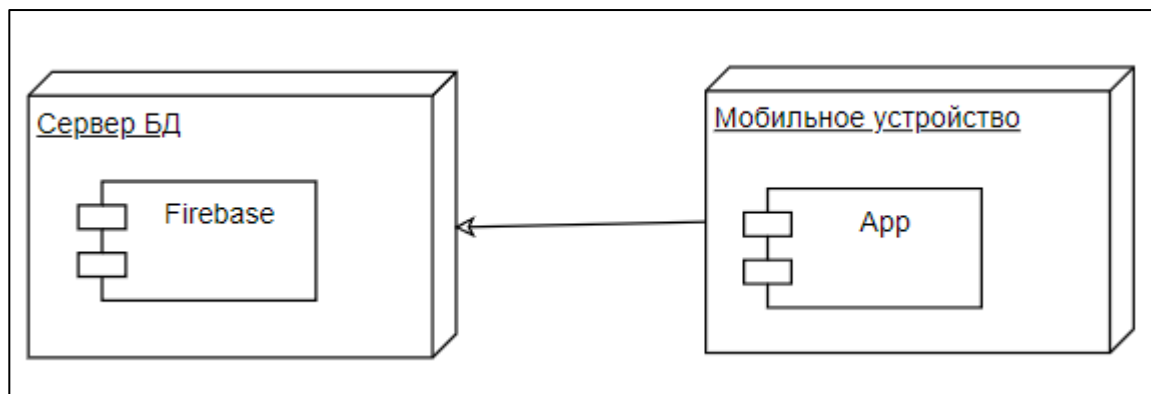


Рисунок 1 – Архитектура приложения (диаграмма компонентов)