Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых» (ВлГУ)

Кафедра ВТиСУ

**КУРСОВАЯ РАБОТА**

по дисциплине «Разработка мобильных приложений»

на тему «Разработка мобильных приложений на платформе Android. Часть 15. Разработка мобильного приложения «Кулинарные рецепты»»

Студент: Свяжин М.В.

Институт: ИИТЭ

Направление: 09.03.03

Группа: ПИ-121

Руководитель: Шутов А.В.

Владимир 2024 г.Оглавление

[ВВЕДЕНИЕ 3](#_Toc185347361)

[**1. Теоретическая часть** 5](#_Toc185347362)

[1.1 Анализ существующих решений 5](#_Toc185347363)

[1.2 Выбор технологий и инструментов разработки 6](#_Toc185347364)

[1.3 Архитектура приложения 7](#_Toc185347365)

[1.4 Структура базы данных 8](#_Toc185347366)

[1.5 Пользовательский интерфейс 9](#_Toc185347367)

[**2. Практическая часть** 14](#_Toc185347368)

[2.1 Создание клиентской части (Android-приложение) 14](#_Toc185347369)

[2.2 Реализация серверной части (PHP-скрипты и API) 15](#_Toc185347370)

[2.3 Интеграция клиент-серверного взаимодействия 20](#_Toc185347371)

[**Заключение** 21](#_Toc185347372)

[**Список литературы** 22](#_Toc185347373)

### ВВЕДЕНИЕ

В современном мире мобильные устройства занимают важное место в повседневной жизни человека. С каждым годом растёт число мобильных приложений, предназначенных для упрощения различных задач — от работы и обучения до отдыха и кулинарии. Одним из популярных направлений разработки является создание приложений с кулинарными рецептами, которые позволяют пользователям быстро находить подходящие блюда, сохранять их, а также делиться своими рецептами с другими людьми.

Удобство использования мобильных приложений и высокая востребованность рецептов делают данную тему особенно актуальной. На фоне увеличивающегося спроса на качественные и функциональные приложения, разработка мобильного решения, объединяющего обширную базу рецептов, интуитивно понятный интерфейс и возможность взаимодействия с серверной частью, представляется значимой и перспективной задачей.

Целью данной курсовой работы является разработка мобильного приложения для Android, которое предоставляет пользователям возможность доступа к коллекции кулинарных рецептов, а также функционал для добавления и редактирования записей. Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Провести анализ существующих решений на рынке мобильных приложений с кулинарными рецептами.
2. Определить подходящие технологии и инструменты для разработки.
3. Разработать архитектуру приложения и структуру базы данных.
4. Реализовать клиентскую и серверную части приложения.
5. Провести тестирование и обеспечить стабильную работу приложения.

Объектом исследования выступает процесс разработки мобильных приложений для управления кулинарными рецептами. Предметом исследования является реализация функциональности приложения на платформе Android с использованием современных технологий клиент-серверного взаимодействия, включая Java и PHP.

Данная работа включает в себя этапы проектирования, программирования и документирования, что позволяет не только продемонстрировать навыки разработки программного обеспечения, но и раскрыть ключевые аспекты создания удобных и эффективных приложений.

## **1. Теоретическая часть**

### 1.1 Анализ существующих решений

На рынке мобильных приложений с кулинарными рецептами представлено множество решений, таких как Yummly, Tasty, AllRecipes и другие. Эти приложения предлагают пользователям широкий функционал: от поиска рецептов по ингредиентам до сохранения избранных блюд и генерации списков покупок. Однако у каждого из них есть свои особенности и недостатки:

* Yummly: предоставляет персонализированные рекомендации на основе предпочтений пользователя. Однако ограничение бесплатного функционала может быть неудобным для пользователей.
* Tasty: ориентировано на визуальный контент с пошаговыми видео-инструкциями. Но отсутствует возможность добавления собственных рецептов.
* AllRecipes: имеет большую базу рецептов и удобный поиск, однако интерфейс приложения может казаться устаревшим.

На основе анализа можно сделать вывод, что ключевыми требованиями для успешного приложения являются:

1. Простой и удобный интерфейс.
2. Широкий набор функциональных возможностей.
3. Персонализация контента под пользователя.

### 1.2 Выбор технологий и инструментов разработки

Для реализации данного проекта выбраны следующие технологии:

1. Android Studio: интегрированная среда разработки для создания приложений на платформе Android.
2. Java: язык программирования, обеспечивающий стабильность и широкий выбор библиотек для мобильной разработки.
3. Volley: библиотека для выполнения HTTP-запросов и взаимодействия с сервером.
4. PHP: серверный язык для создания API и обработки запросов.
5. MySQL: реляционная база данных для хранения информации о пользователях и рецептах.

Выбор данных инструментов обусловлен их популярностью, доступностью и широкими возможностями для реализации всех необходимых функций. Такой стек технологий позволяет разработать приложение, отвечающее требованиям современного рынка.

### 1.3 Архитектура приложения

Архитектура приложения основана на клиент-серверной модели, которая обеспечивает взаимодействие между мобильным приложением и сервером. Основные компоненты:

1. Клиентская часть: Android-приложение, написанное на Java, которое отправляет запросы к серверу и обрабатывает ответы.
2. Серверная часть: реализована на PHP с использованием MySQL для управления данными.
3. API: обеспечивает взаимодействие между клиентской и серверной частями через HTTP-запросы с использованием библиотеки Volley.

Коммуникация между клиентом и сервером происходит через REST API. Это позволяет легко масштабировать приложение и вносить изменения в серверную часть без необходимости обновления клиента.

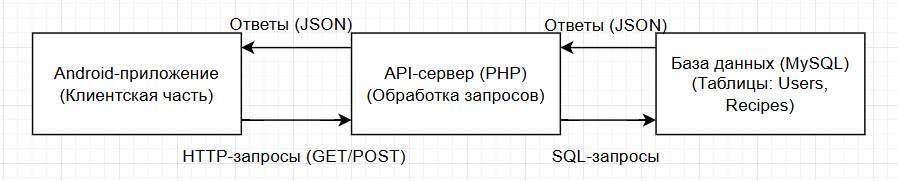


Рис.1 – «Архитектура приложения»

### 1.4 Структура базы данных

Для хранения данных используется реляционная база данных MySQL. Основные таблицы базы данных:

1. Users: хранит информацию о зарегистрированных пользователях (ID, имя пользователя, пароль, роль).
2. Recipes: содержит информацию о рецептах (ID, название, описание, ингредиенты, пользователь-создатель).

Каждая таблица связана друг с другом с использованием внешних ключей для обеспечения целостности данных.

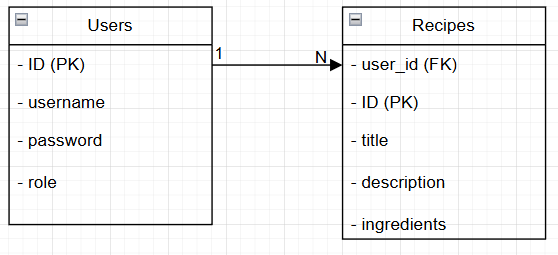


Рис. 2 – «ER-диаграмма базы данных»

### 1.5 Пользовательский интерфейс

Пользовательский интерфейс приложения был разработан с акцентом на простоту и удобство использования. Основные экраны:

1. Экраны авторизации и регистрации: включает формы для ввода логина и пароля, а также кнопку регистрации. (Рис. 3 и Рис. 4)
2. Главный экран: отображает список рецептов, пользователь с ролью «admin» имеет возможность редактировать список рецептов. (Рис. 5)
3. Экран рецепта: содержит полное описание рецепта, список ингредиентов (Рис. 6)

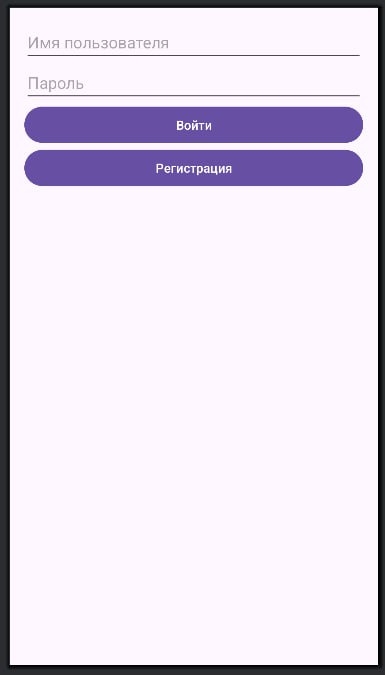


Рис. 3 – «Экран авторизации»

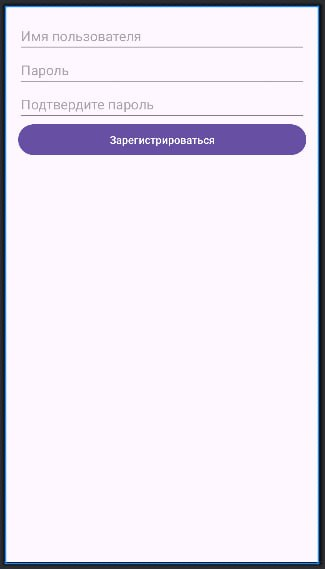


Рис. 4 – «Экран регистрации»

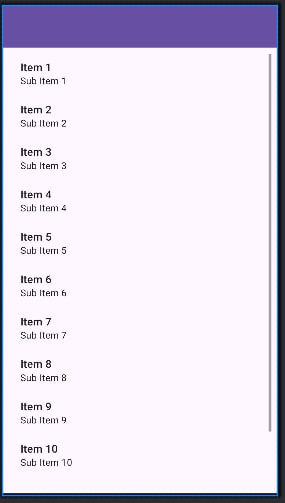


Рис. 5 – «Главный экран»

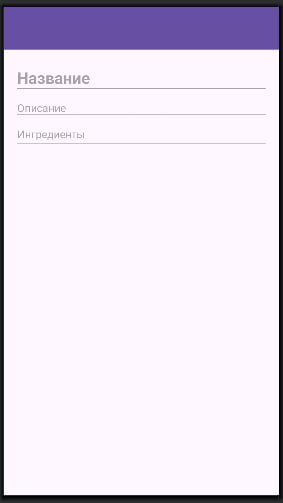


Рис. 6 – «Экран рецепта»

## **2. Практическая часть**

### 2.1 Создание клиентской части (Android-приложение)

Клиентская часть была реализована с использованием языка программирования Java и среды разработки Android Studio. Основные этапы разработки включали:

1. Настройка проекта:
   * Создание нового проекта в Android Studio.
   * Настройка зависимостей, включая библиотеку Volley для работы с HTTP-запросами.
   * Добавление базовых настроек (иконки, название приложения, версия SDK).
2. Реализация экранов приложения:
   * Экран авторизации и регистрации: позволяет пользователям войти в существующую учетную запись или создать новую.
   * Главный экран: отображает список рецептов
   * Экран добавления рецепта: предоставляет возможность добавлять новые рецепты с указанием названия, описания и ингредиентов.
3. Интеграция клиент-серверного взаимодействия:
   * Настроены API-запросы для получения списка рецептов, авторизации пользователя и добавления данных.
   * Реализована обработка ошибок сети (например, проверка наличия соединения).

### 2.2 Реализация серверной части (PHP-скрипты и API)

Серверная часть была реализована с использованием языка PHP. Она отвечает за обработку запросов, отправленных клиентом, и взаимодействие с базой данных. Основные реализованные функции:

1. API для работы с пользователями:
   * Регистрация нового пользователя. (Рис. 7)
   * Авторизация пользователя. (Рис. 8)

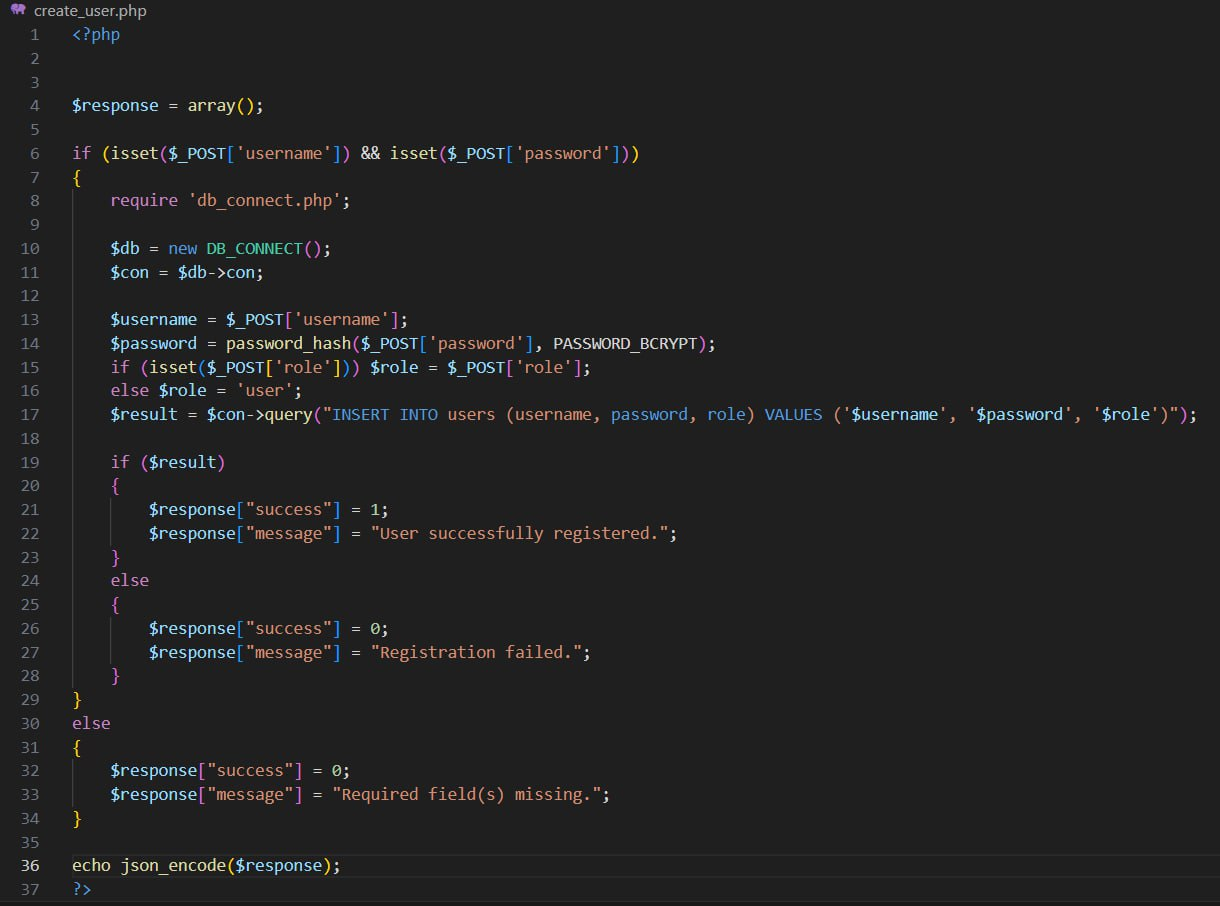


Рис. 7 – «Регистрация»

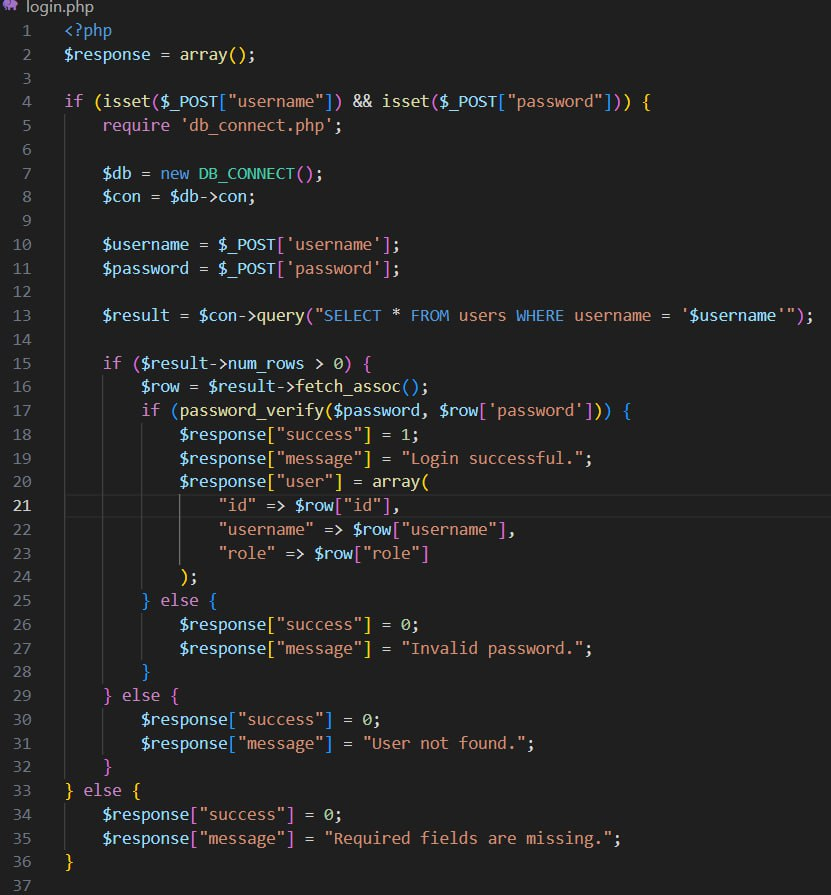


Рис. 8 – «Авторизация»

1. API для работы с рецептами:
   * Получение списка всех рецептов. (Рис. 9)
   * Добавление нового рецепта. (Рис. 10)
   * Удаление и редактирование существующих рецептов (функционал для администратора). (Рис. 11 и Рис. 12)

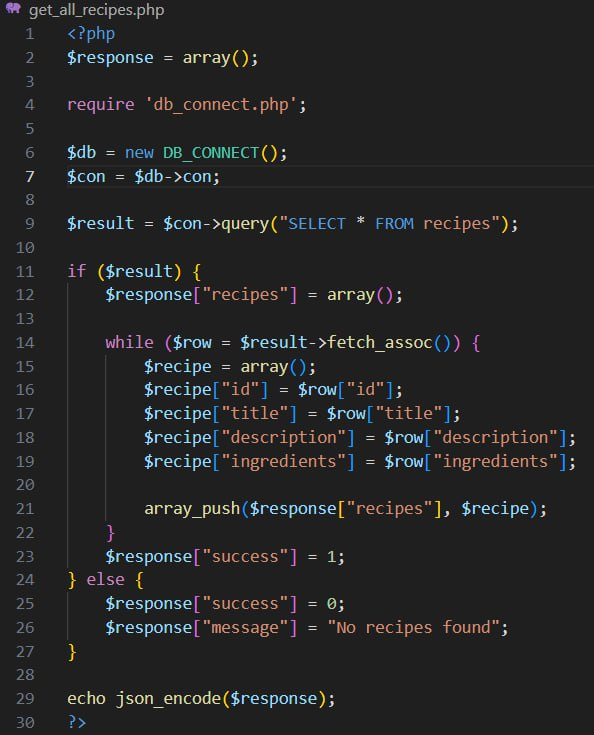


Рис. 9 – «Получение списка рецептов»

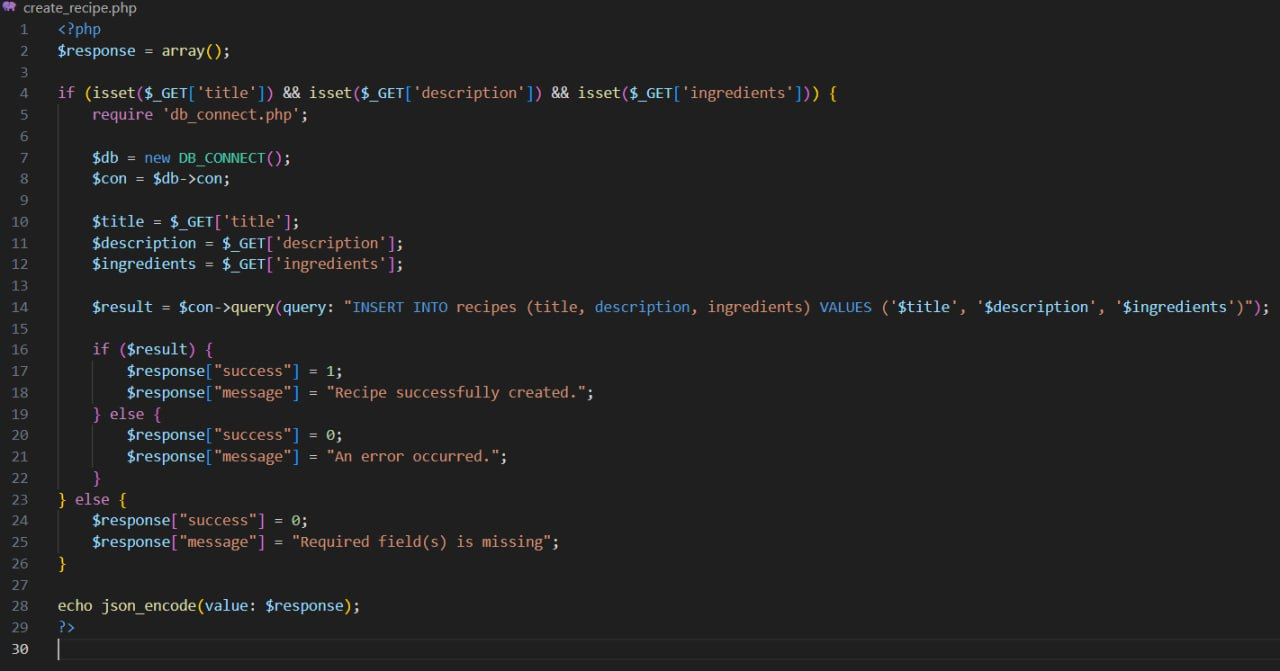


Рис. 10 – «Создание рецепта»



Рис. 11 – «Редактирование рецепта»

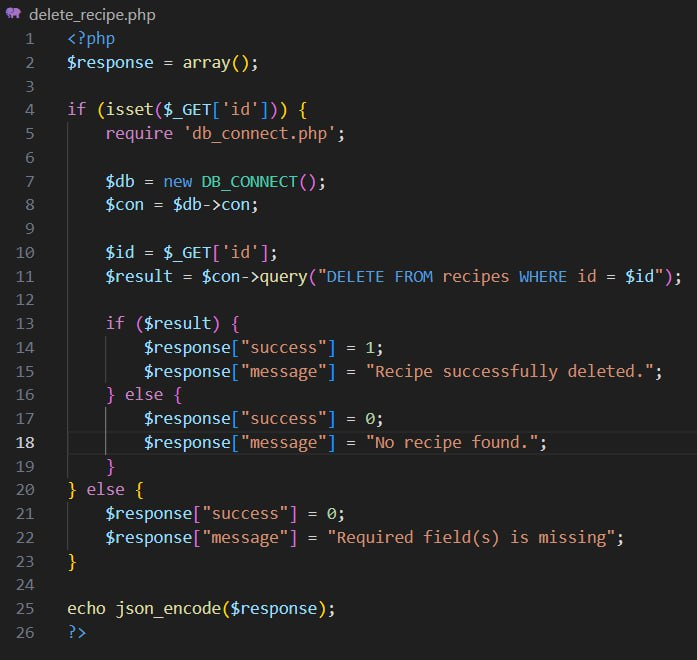


Рис. 12 – «Удаление рецепта»

1. Обработка ошибок:
   * Реализован контроль за корректностью данных, отправляемых клиентом.
   * Возвращаются сообщения об ошибках с кодами (например, 400 — неправильные данные, 401 — неавторизован).
2. Настройка базы данных:
   * Созданы необходимые таблицы для пользователей и рецептов.
   * Настроены индексы и связи для обеспечения целостности данных.

### 2.3 Интеграция клиент-серверного взаимодействия

Интеграция осуществлялась путем настройки HTTP-запросов с использованием библиотеки Volley. Основные этапы:

1. Настройка авторизации:
   * Клиент отправляет данные пользователя (логин, пароль) на сервер.
   * Сервер возвращает токен, который используется для выполнения всех последующих запросов.
2. Получение списка рецептов:
   * Реализован запрос для получения данных с сервера.
   * Клиент отображает данные в списке с использованием RecyclerView.
3. Добавление рецептов:
   * Клиент отправляет данные нового рецепта на сервер через POST-запрос.
   * Сервер обрабатывает запрос и добавляет запись в базу данных.
4. Тестирование запросов:
   * Для проверки корректной работы API использовался инструмент Postman.
   * Выполнено тестирование основных операций (GET, POST, DELETE).

## **Заключение**

В ходе выполнения курсовой работы была достигнута основная цель — разработка мобильного приложения для управления кулинарными рецептами на платформе Android. Проведенный анализ существующих решений позволил определить ключевые требования к функциональности приложения, что стало основой для проектирования и реализации.

На этапе проектирования были разработаны архитектура приложения и структура базы данных. Реализованная архитектура на основе клиент-серверной модели обеспечила стабильное взаимодействие между клиентской частью (Android-приложением) и сервером, созданным на языке PHP с использованием базы данных MySQL. Также удалось интегрировать основные функции приложения, такие как регистрация и авторизация пользователей, просмотр и добавление рецептов, а также управление данными через API.

В рамках реализации проекта удалось решить все поставленные задачи:

1. Проведен анализ существующих приложений, что позволило учесть их преимущества и недостатки.
2. Выбраны технологии и инструменты, соответствующие требованиям современного рынка разработки мобильных приложений.
3. Создано Android-приложение с интуитивно понятным интерфейсом, поддерживающее основные функции для работы с рецептами.
4. Реализовано взаимодействие клиента и сервера с использованием REST API.
5. Обеспечена стабильная работа приложения в процессе тестирования.

Таким образом, цели курсовой работы были успешно достигнуты. Разработанное приложение соответствует поставленным требованиям и обладает потенциалом для дальнейшего развития.

## **Список литературы**

1. Android Developers. Документация по разработке приложений на платформе Android [Электронный ресурс]. – Режим доступа: <https://developer.android.com>.

2. Официальная документация Java [Электронный ресурс]. – Режим доступа: <https://docs.oracle.com/javase/>.

3. PHP Manual. Официальная документация PHP [Электронный ресурс]. – Режим доступа: <https://www.php.net/manual/>.

4. MySQL Documentation. Руководство по работе с базами данных MySQL [Электронный ресурс]. – Режим доступа: <https://dev.mysql.com/doc/>.