

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

Факультет компьютерных наук

Кафедра информационных технологий управления и кафедра программирования и
информационных технологий

Приложение “Chef’s Book”

Курсовой проект

09.03.02 Информационные системы и технологии

Обучающийся _____ *Д.И. Евдокимов 3 курс, д/о*

Обучающийся _____ *В.А. Симонов 3 курс, д/о*

Обучающийся _____ *С.С. Павленок 3 курс, д/о*

Руководитель _____ *Тарасов В. С.*

Воронеж 2020

Оглавление

Введение	3
1. Постановка задачи	4
2. Анализ предметной области	5
2.1 Анализ существующих решений	5
2.2 Анализ средств реализации	6
3. Графическое описание работы системы	8
3.1 Диаграмма прецедентов	8
3.2 Диаграмма последовательностей	10
3.3 Диаграмма состояний	11
3.4 Диаграммы активности	12
3.6 Диаграмма взаимодействий	14
3.7 Диаграмма классов	15
3.8 Диаграмма объектов	16
3.9 Диаграмма развертывания	16
4. Графический интерфейс	17
4.1 Страница авторизации	17
4.2 Главная страница	18
4.3 Страница категории	19
4.4 Страница рецепта	20
5 Тестирование	23
5.1 Тестирование Backend	23
5.2 Интеграционное тестирование	24
5.3 GUI тесты	25
Заключение	26

Введение

Мобильные приложения входят в жизнь людей с огромной скоростью. В настоящее время разработано более миллиона мобильных приложений. Все они относятся к различным категориям: игры, здоровье, музыка, кулинария, образование и пр. Мобильные приложения облегчают повседневные задачи, помогают в их выполнении, разнообразят рутинные дела, вовлекают в процесс выполнения, казалось бы совершенно обычных и даже скучных дел, и даже могут помочь улучшить некоторые навыки и способности.

В наше время, практически у каждого человека есть смартфон, который всегда под рукой. Люди стали быстрее получать различную информацию, и сами данные стали доступнее с появлением смартфонов. По статистике, человек, пользующийся смартфоном, в среднем проверяет свой телефон раз в 6,5 минут. Поэтому, актуальность темы мобильных приложений достаточно высока.

Данная работа направлена на создание мобильного приложения, которое облегчит такую повседневную обязанность, как приготовление еды. Отличительной особенностью системы будет простой подбор рецепта по категории, названию или по национальной кухне. А встроенный в приложение список покупок избавит пользователя от необходимости создания заметок в сторонних программах или на бумажном носителе.

1. Постановка задачи

Целью данной работы является разработка мобильной кулинарной книги для Android с функцией фильтрации блюд по категориям, составления списка покупок и добавления рецептов в избранное. Для достижения цели работы, необходимо решить следующие задачи:

- анализ литературы и смежных проектов, связанных с мобильной кулинарией;
- определение требований к мобильному приложению для мобильной кулинарной книги;
- разработка архитектуры мобильного приложения для мобильной кулинарной книги;
- разработка схемы взаимодействия пользователя с интерфейсом приложения;
- создание базы данных с рецептами для мобильной кулинарной книги.

2. Анализ предметной области

Разработка мобильных приложений на сегодняшний день — долгий и трудоемкий процесс. Чтобы приложение было успешным на рынке, необходимо учесть и проработать многие аспекты, такие как дизайн, функционал, хранение и обработка данных, база данных. Важной частью в создании мобильных приложений является создание удобного к применению интерфейса. Вот основные проблемы проектирования мобильных приложений, такие как:

- разные размеры экранов мобильных телефонов и других похожих устройств;
- трудности ввода данных в мобильное приложение;
- легкость в использовании мобильного приложения;
- безопасность хранения данных.

Таим образом, для создания мобильного приложения необходимо разработать простой в использовании и приятный для пользователя интерфейс.

2.1 Анализ существующих решений

В области кулинарии существует большое количество мобильных приложений, работающих на операционной системе Android. Рассмотрим наиболее крупные из них:

Кулинарная книга. Рецепты (разработчик trey0102)

Кулинарная книга — это сборник рецептов, а также отличный помощник для всех, кто любит приготовить что-нибудь вкусненькое. Особенностью приложения являются: возможность добавления в избранное, возможность делиться рецептами в соц сетях, поиск блюд по названиям и ингредиентам. Недостатком данного приложения является отсутствие встроенного списка покупок.

Кулинарная книга: RecetteTek

RecetteTek — это кулинарная книга для всех любителей готовки. С помощью данного приложения Вы сможете создать свою собственную библиотеку рецептов и с легкостью ими управлять. Плюсом данного приложения является возможность создавать свои рецепты, интегрированная система поиска рецептов, список покупок. Минусы данного приложения заключаются в том, что в приложении нет встроенных рецептов, без подключения к интернету невозможно найти рецепт, наличие рекламы.

Поваренная книга рецептов (разработчик Riafy Technologies)

Особенностями данного приложения являются: возможность составлять план питания, составлять список покупок, добавлять предметы в избранное. Минусами данного приложения являются: невозможность поиска рецептов без подключения к интернету, наличие рекламы в обычной версии, ежемесячная платная подписка для премиума версии без рекламы.

2.2 Анализ средств реализации

В качестве средств реализации системы поиска авиабилетов были выбраны следующие технологии:

1. PostgreSQL была выбрана в качестве СУБД. Оно является наиболее популярным в данный момент и регулярно обновляемым.
2. Android SDK был выбран из-за наличия большого функционала, позволяющего запускать тестирование и отладку исходных кодов, проверять совместимость приложения с различными версиями Android.
3. IntelliJ IDEA была выбрана как среда разработки для API. IDEA была выбрана за то, что это интеллектуальная среда разработки, понимающая код. В процессе написания кода программистом она занимается построением синтаксического дерева, определением особенностей размещенных ссылок, анализом возможных путей исполнения операторов и передачи данных. Основываясь на полученных результатах, программа обращает внимание специалиста на существующие ошибки и самостоятельно устраняет их, предоставляет варианты автоматического дополнения кода. Благодаря указанным особенностям она избавляет пользователя от повседневной рутины и позволяет ему сконцентрироваться на более важных задачах.
4. Для разработки API были использованы фреймворк JAVA EE и сервер Wildfly. JAVA EE обеспечивает масштабируемость приложения и целостность данных во время работы системы. Java EE является промышленной технологией и в основном используется в высокопроизводительных проектах, в которых необходима надежность, масштабируемость и гибкость. Платформа позволяет сократить стоимость и сложность разработки, развертывания многоуровневых серверных приложений и управления ими. WildFly – сервер приложений Java EE, полностью разработанным на Java, и, следовательно, может работать в любой операционной системе, как 32-битной, так и 64-битной. WildFly имеет высоко

оптимизированный процесс загрузки; его службы запускаются одновременно, чтобы исключить время ожидания между запуском одной службы и началом другой, а все службы, которые не требуются во время запуска, остаются в режиме ожидания до момента первого использования.

5. Java использовался для реализации Android-приложения. Это проверенный временем язык программирования, существует много различных материалов по написанию Android-приложения на Java. Одним из наиболее весомых плюсов Java является наличие стандартной библиотеки, которая постоянно пополняется, так что в ней можно найти алгоритмы и данные для огромного количества задач. Так же автоматическая сборка мусора в Java избавляет от необходимости вручную прописывать удаление неиспользуемых объектов.
6. Для спецификации REST API был использован Swagger. Его прелесть заключается в том, что он дает возможность не только интерактивно просматривать спецификацию, но и отправлять запросы – так называемый Swagger UI.
7. Для тестирования был выбран самый известный и самый используемый фреймворк JUnit. Это удобный и простой в использовании фреймворк.

3. Графическое описание работы системы

Для описания работы системы была использована графическая нотация UML. В данном разделе представлены диаграммы и описания, где они требуются.

3.1 Диаграмма прецедентов



Рисунок 1 - Диаграмма прецедентов

При взаимодействии с сайтом у пользователей есть определенный список возможностей, который более наглядно изображен на рисунке 1:

Авторизованный пользователь имеет возможность:

- искать рецепты по названию;
- искать рецепты по категории;
- искать рецепты по национальной кухне;
- формировать список покупок;
- добавлять рецепты в избранное.

Гость имеет возможность:

- авторизоваться;
- искать рецепты по категории;
- искать рецепты по национальной кухне;
- искать рецепты по названию.

Администратор имеет возможность:

- редактирование данных (удаление, добавление, изменение);

3.2 Диаграмма последовательностей

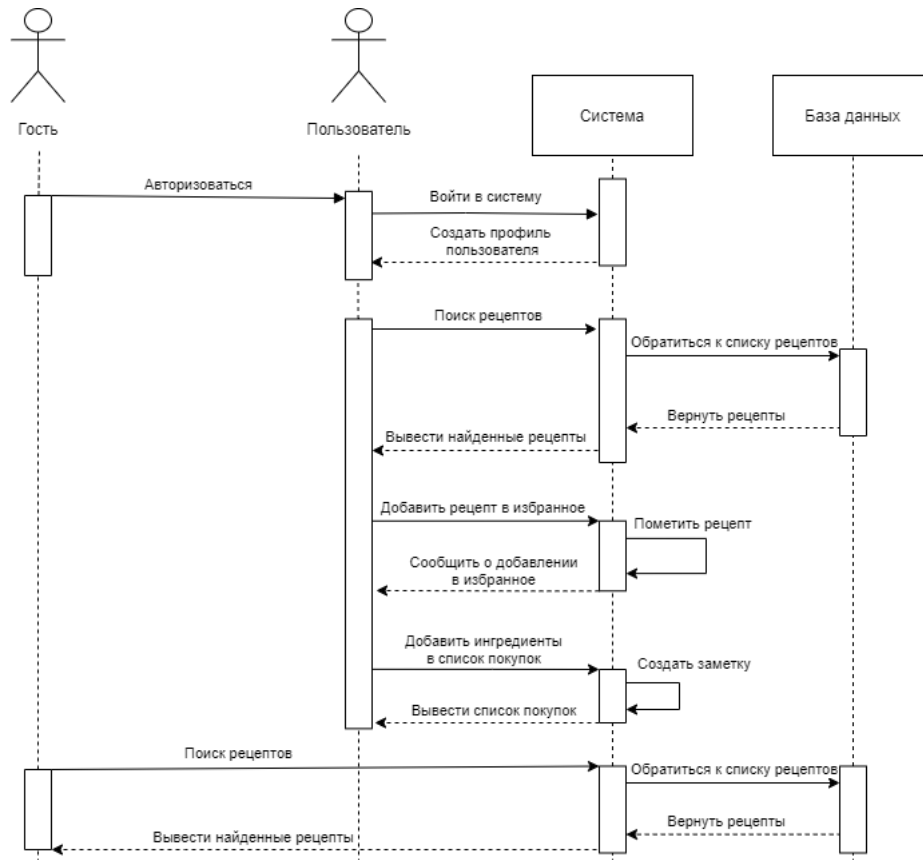


Рисунок 2 - Диаграмма последовательностей

Диаграмма последовательности — диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл какого-либо определённого объекта и взаимодействие актеров ИС в рамках какого-либо определённого прецедента.

Сначала гость авторизуется в системе, система создает уникальный профиль пользователя. Далее авторизованный пользователь может:

- искать рецепты по поиску;
- добавлять рецепты в избранное;
- добавлять ингредиенты в список покупок.

Во время поиска рецепта система обращается к базе данных и ищет совпадение. Если совпадение найдено, то база данных посылает данные о рецепте в систему, а системы выводит данный рецепт пользователю.

Когда авторизованный пользователь добавляет рецепт в избранное система помечает этот рецепт и сообщает пользователю о том, что данный рецепт добавлен в избранное.

Во время добавления ингредиента в список покупок система создает заметку, а затем выводит обновленный список покупок пользователю.

3.3 Диаграмма состояний

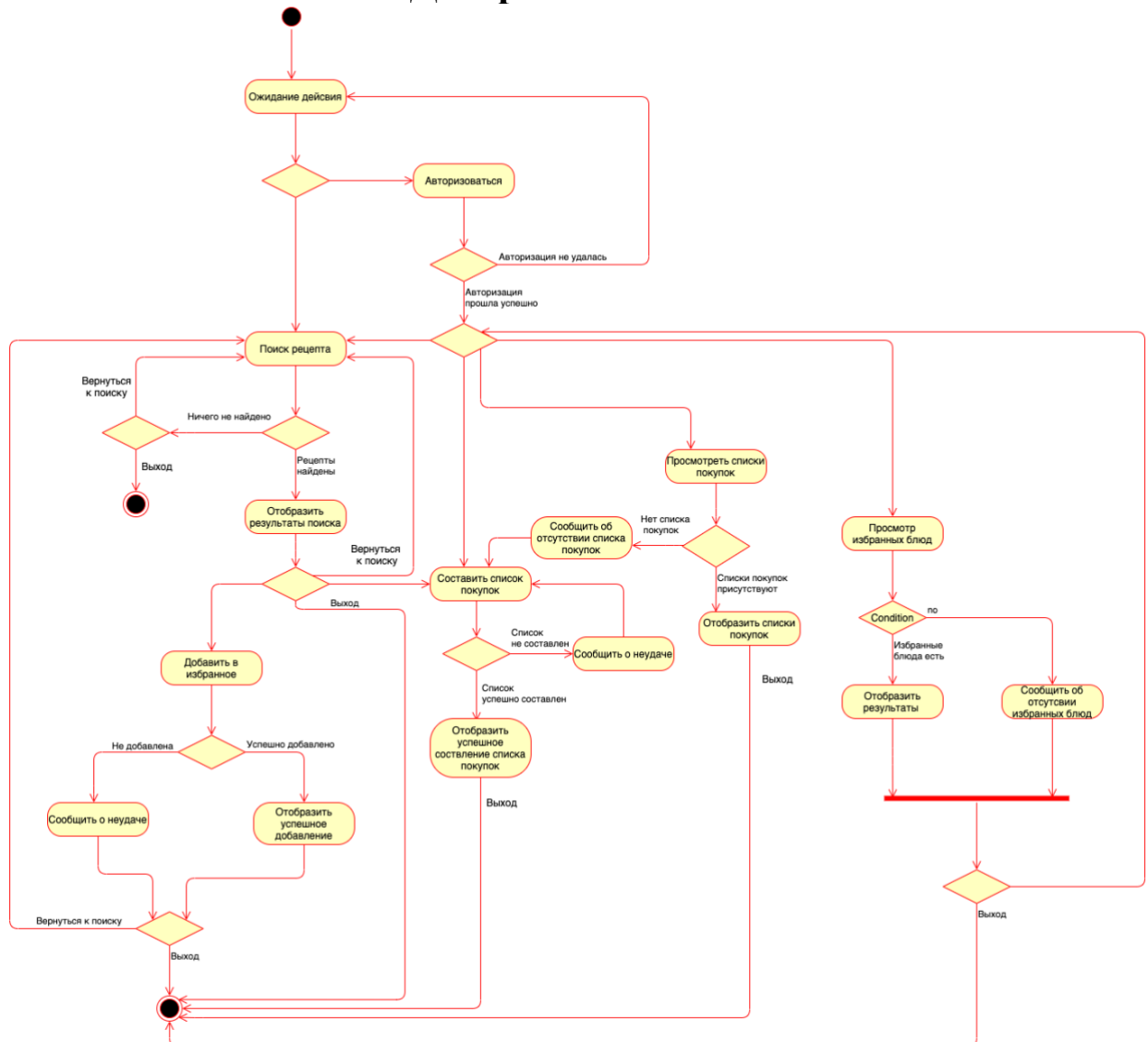


Рисунок 3 - Диаграмма состояний

Для наглядного представления состояний в которых может находиться система составлена диаграмма изображенная на Рисунке 3.

3.4 Диаграммы активности

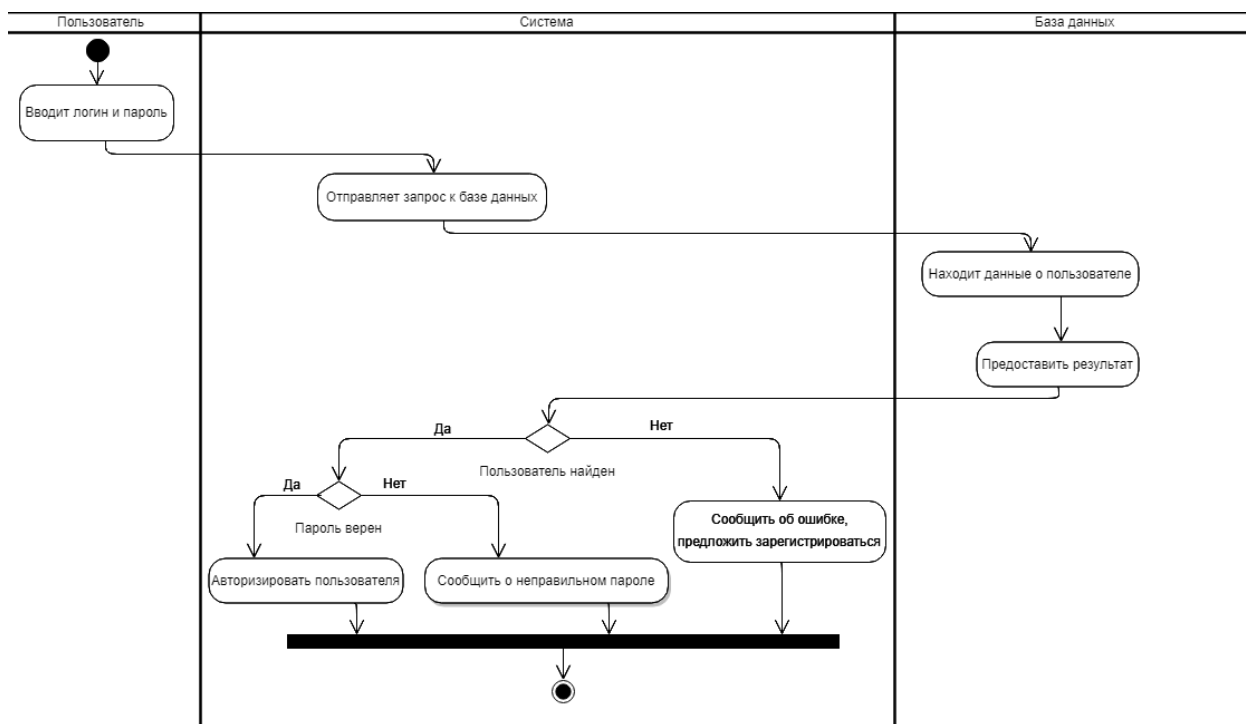


Рисунок 4 - Диаграмма активностей(авторизация)

Диаграмма активности авторизации в системе изображена на Рисунке 4. На данной диаграмме присутствуют 3 части (дорожки): пользователь, система и база данных.

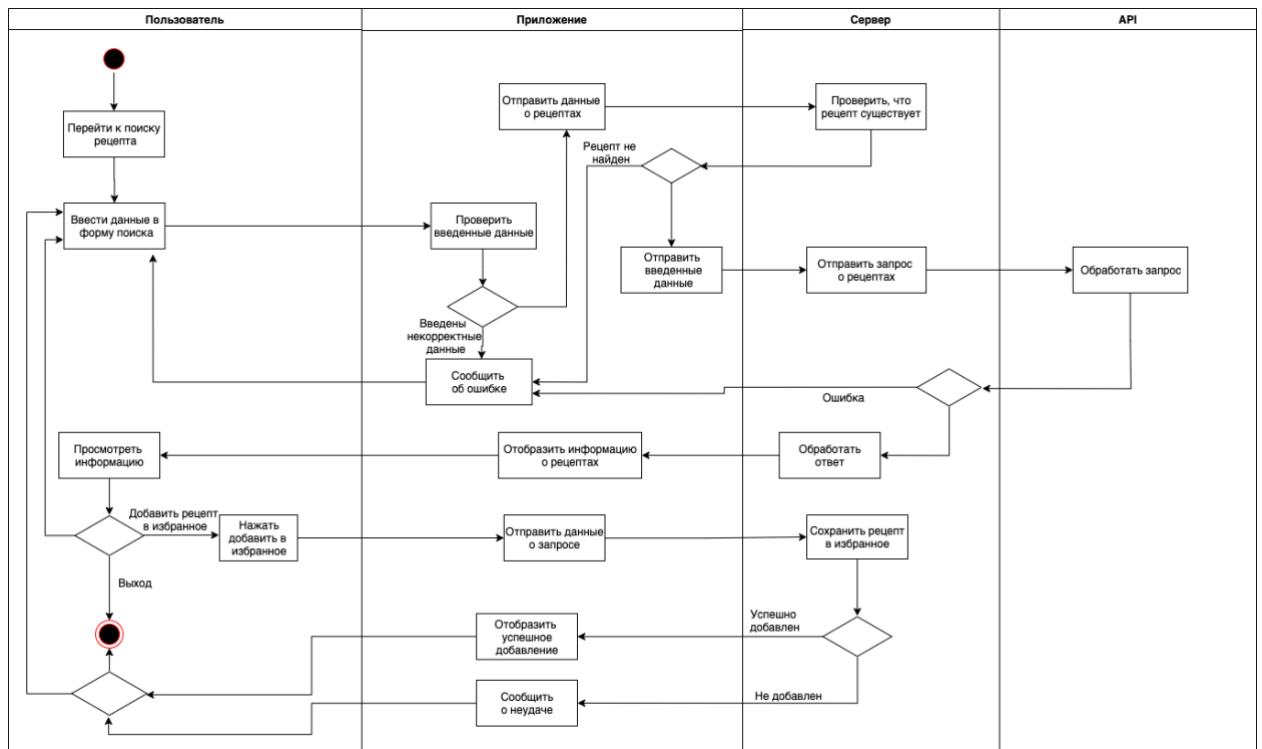


Рисунок 6 - Диаграмма активностей (поиск и добавление рецепта в избранное)

Диаграмма активности, отражающая поиск и добавление рецепта в избранное изображена на Рисунке 6. На данной диаграмме присутствуют 4 части (дорожки): пользователь, приложение, сервер, API.

3.6 Диаграмма взаимодействий

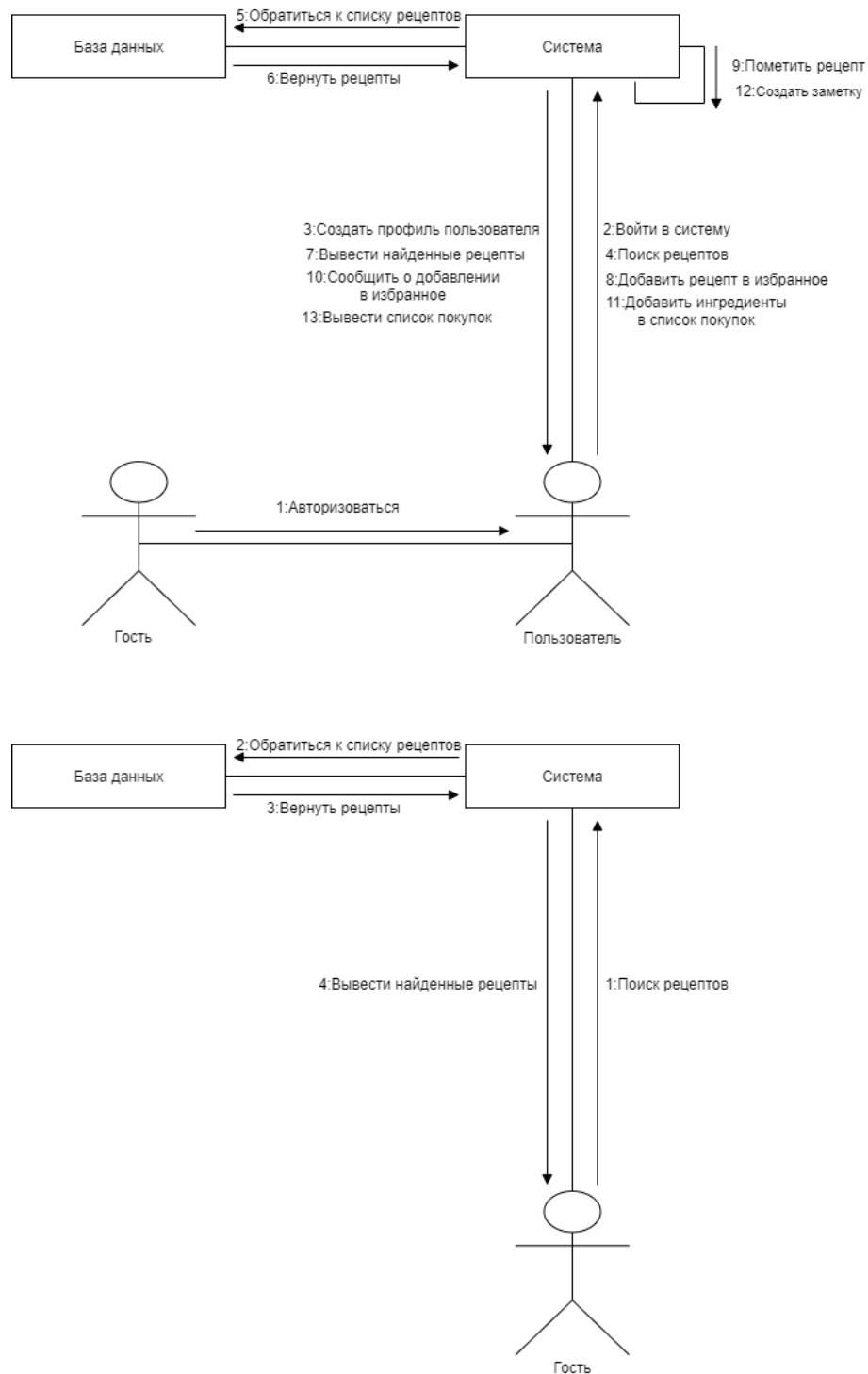


Рисунок 7 - Диаграмма взаимодействий

Диаграмма взаимодействий, изображенная на Рисунке 7, позволяет видеть все взаимодействия запросов в системе. Так же она позволяет определить,

какие способы взаимодействия с системой есть в распоряжении пользователей

3.7 Диаграмма классов

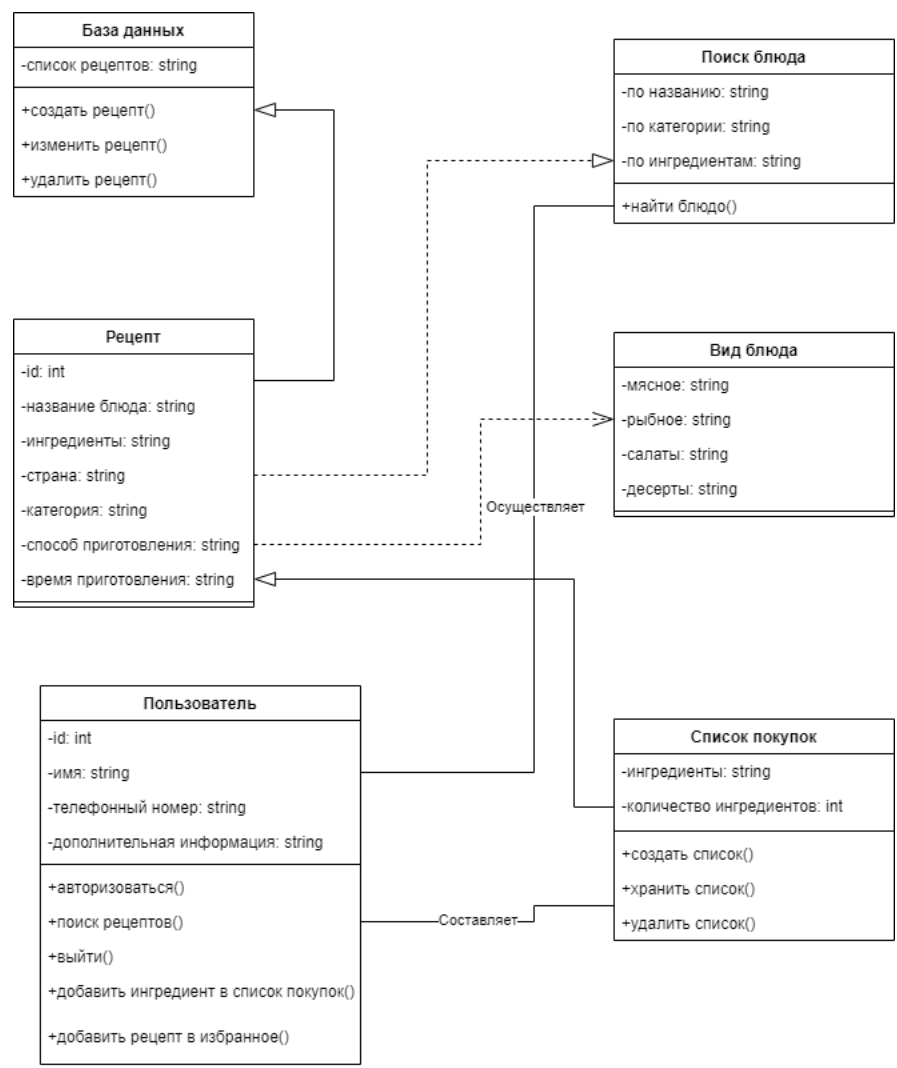


Рисунок 8 - Диаграмма классов

На рисунке 8 изображена диаграмма классов, которая отображает классы системы, их атрибуты и их взаимодействия.

3.8 Диаграмма объектов

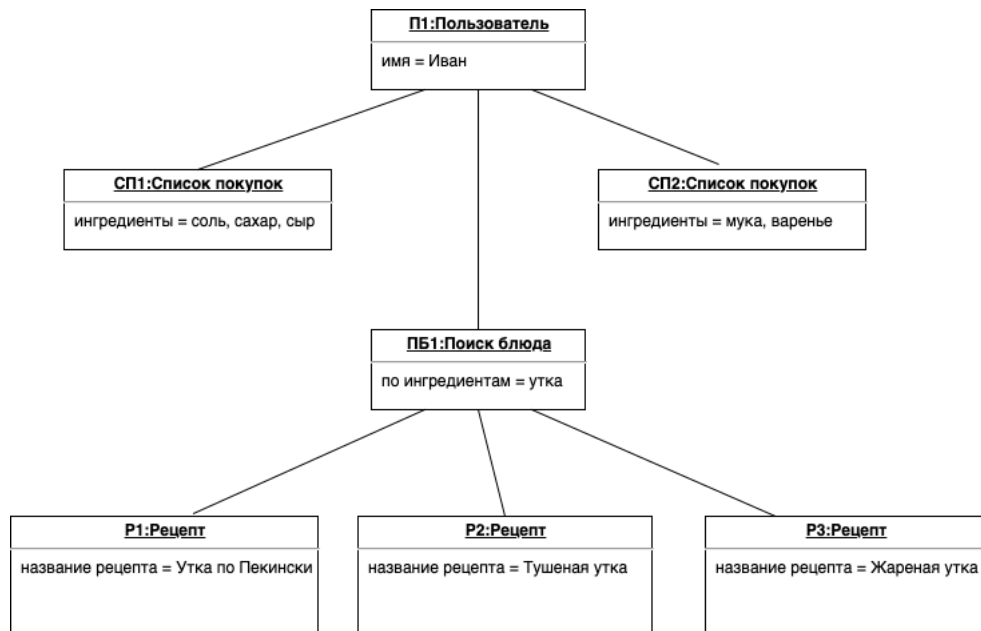


Рисунок 9 - Диаграмма объектов

Диаграмма, изображенная на Рисунке 9, показывает множество объектов - экземпляров классов и отношений между ними в некоторый момент времени.

3.9 Диаграмма развертывания

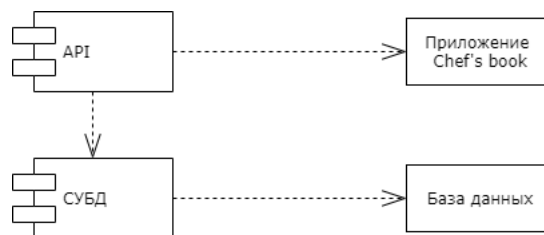


Рисунок 10 - Диаграмма развертывания

Приведенная на Рисунке 10 диаграмма визуализирует элементы и компоненты программы, которые существуют на этапе ее исполнения.

4. Графический интерфейс

4.1 Страница авторизации

На данной странице размещены поля:

- для ввода логина;
- для ввода пароля;

После того как пользователь вводит свои данные, то он нажимает по кнопке “Войти в аккаунт” и если данные введены корректно, то переходит на главную страницу приложения.

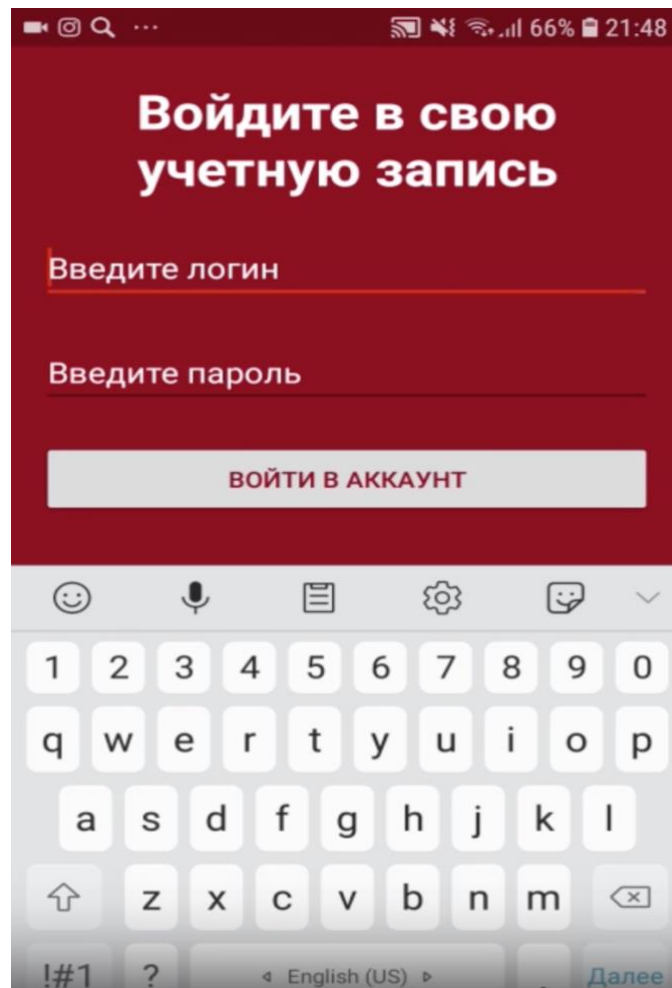


Рисунок 11 – Страница авторизации

4.2 Главная страница

Выводится при старте приложения и предоставляет следующие возможности для пользователя:

- осуществить поиск рецептов;
- перейти на страницу одного из предложенных рецептов;
- перейти на страницу одной из категорий рецептов.

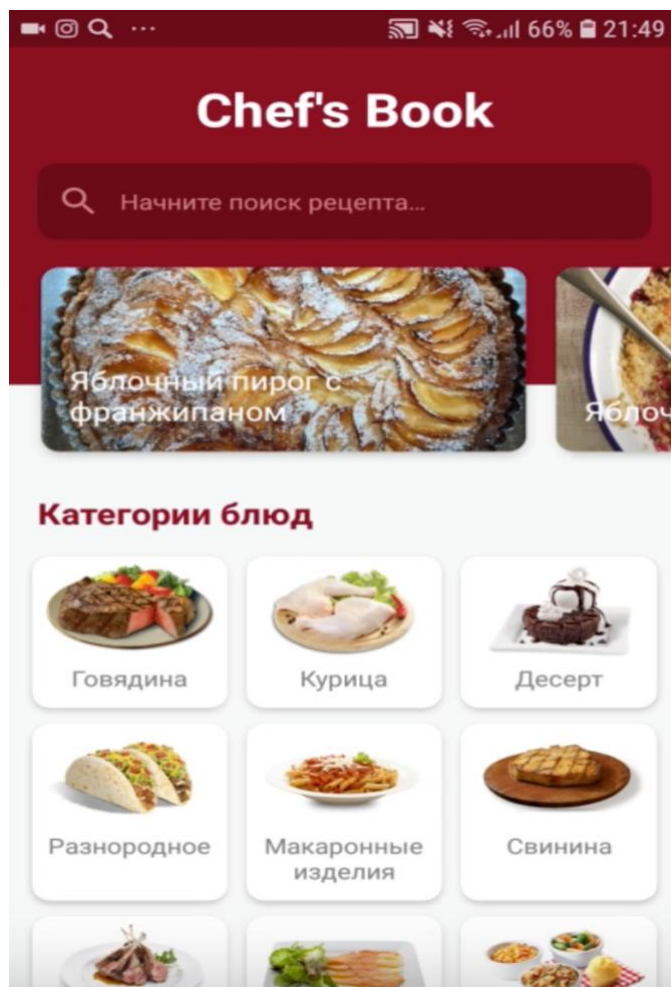


Рисунок 12 – Главная страница приложения

4.3 Страница категории

На данной странице пользователь может:

- выбрать другую категорию рецептов;
- проглядеть все блюда, которые относятся к данной категории;
- прочитать описание категории;
- перейти на страницу рецепта;
- вернуться на главную страницу.

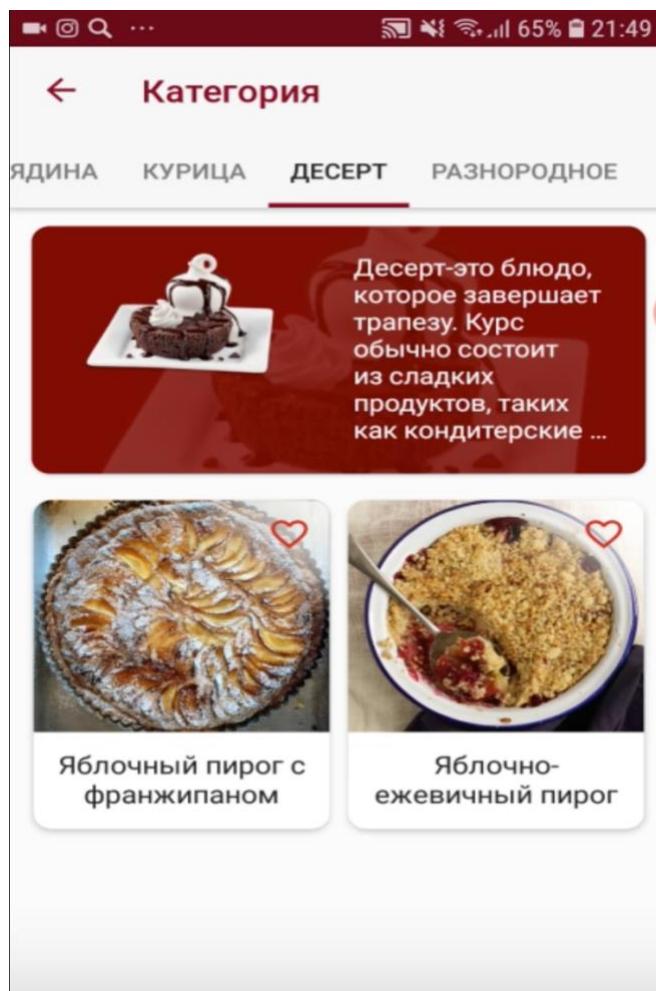


Рисунок 13 – Страница категории блюд

4.4 Страница рецепта

На данной странице пользователь может:

- посмотреть фото готового блюда;
- узнать к какой категории относится данный рецепт;
- узнать к кухне какой страны относится данный рецепт;
- прочитать инструкцию по приготовлению блюда;
- посмотреть какие продукты необходимы для приготовления блюда
- перейти на Youtube и поглядеть видеорецепт.

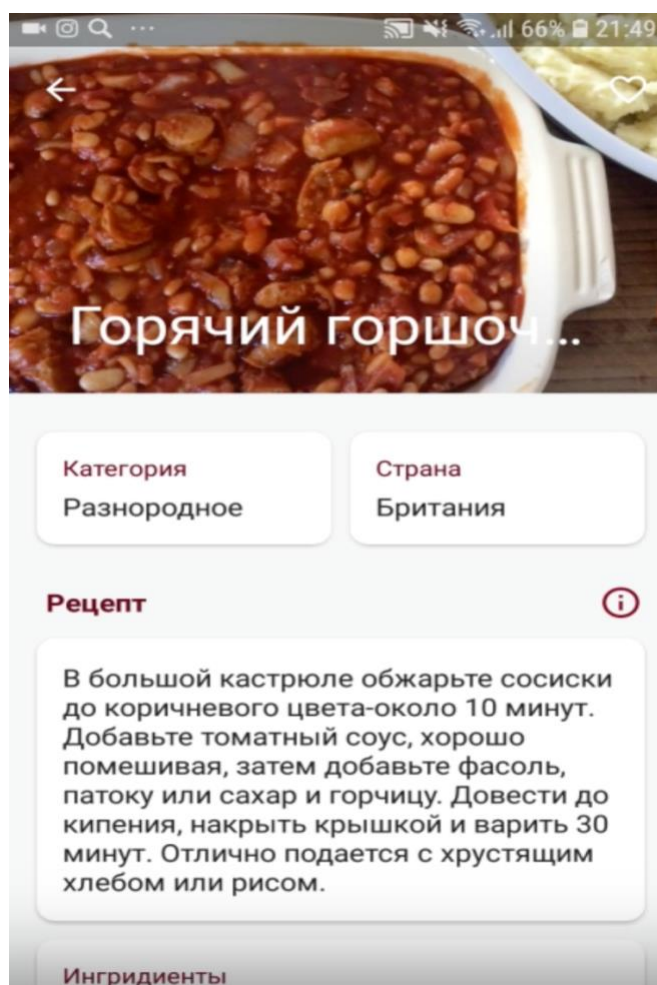


Рисунок 14 – Страница рецепта

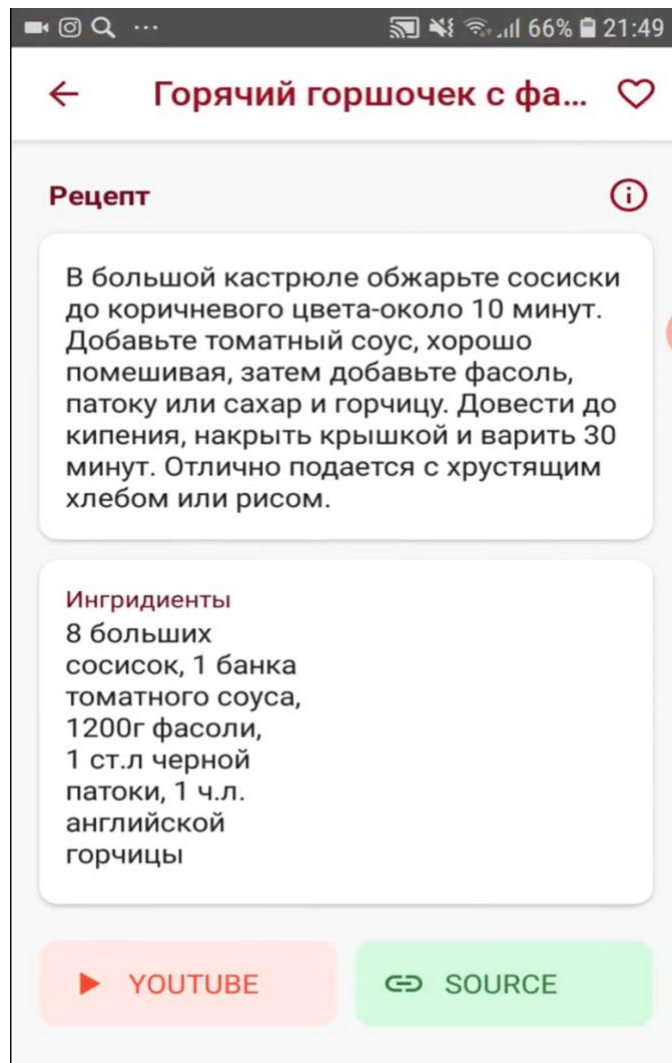


Рисунок 15 – Страница рецепта

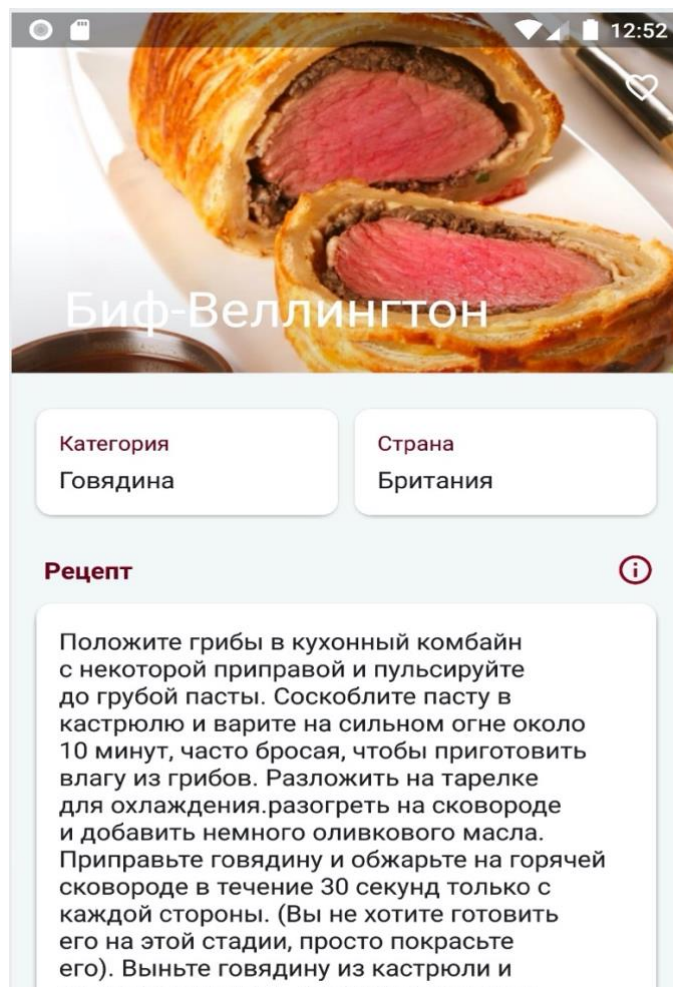


Рисунок 16 – Страница рецепта

5 Тестирование

В целях проверки работоспособности системы в целом и подсистем приложения были проведены тесты по принципу «белого ящика» следующего вида:

- тестирование Backend;
- интеграционное тестирование;
- GUI тесты.

5.1 Тестирование Backend

Для тестирования Backend были проведены следующие тесты:

- Handle user login;
- Security testing;
- Complex route testing.

Результаты тестирования продемонстрированы на Рисунке 17.

Backend test 1: handle user login			Backend test 2: security testing			Backend test 3: complex route testing		
Шаг	Действие	Проверка	Шаг	Действие	Проверка	Шаг	Действие	Проверка
1	Запустить Backend приложение	Проект скомпилировался без ошибок, сервер активен	1	Запустить Backend приложение	Проект скомпилировался без ошибок, сервер активен	1	Используя валидный bearer token перейти по маршруту с параметром (к примеру mealCategory/category={category})	405 ошибка отсутствует, маршрут вывел все блюда за прошлые категории
2	Перейти по маршруту регистрации, используя валидные данные	Логин произошел успешно, в response headers содержится bearer token, созданный под данного пользователя	2	Перейти по маршруту регистрации, используя валидные данные	Логин произошел успешно, в response headers содержится bearer token, созданный под данного пользователя			
			3	Полученный bearer token подставить в header и перейти по маршруту, который выводит список категорий	405 ошибка отсутствует, мы получили все данные о категориях, хранящихся в БД			Тесты будут дополняться по мере включения в приложение новых функций
Приоритетность кейсов			Критерий приёмки					
Test case	Priority							
1	High							
2	High							
3	High							

Рисунок 17 - Тестирование Backend

5.2 Интеграционное тестирование

Результаты интеграционного тестирования продемонстрированы на Рисунке 17.

Integration test 1:			Приоритетность кейсов		Критерий приёмки
Шаг	Действие	Проверка	Test case	Priority	
1	Запустить сервер с API и запустить андроид приложение	Оба приложения работают корректно	1	High	100%
2	Изучить главный экран андроид приложения	Контент, предоставляемый API, отображается в приложении(текст, картинки)			
3	Перейти в детальный обзор блюда	Более детальная информация о блюде (рецепт, ссылка на ютуб, ингредиенты, страна и т.д.) была загружена из базы данных корректно		Тесты будут дополняться по мере включения в приложение новых функций	
4	Перейти в сортировку по категориям блюд	API правильно поставил набор блюд, которые действительно находятся в этой категории			

Рисунок 18 - Интеграционное тестирование

5.3 GUI тесты

Для тестирования Backend были проведены следующие Test cases :

- Smoke GUI app hellcheck;
- GUI test of detail recipe view;
- Smoke GUI category view testing.

Результаты GUI тестов продемонстрированы на Рисунке 19.

Test case 1: Smoke GUI app hellcheck			Test case 2: GUI test of detail recipe view			Test case 3: Smoke GUI category view testing		
Шаг	Действие	Проверка	Шаг	Действие	Проверка	Шаг	Действие	Проверка
1	Запустить приложение	Приложение запущено корректно и без ошибок	1	Запустить приложение	Приложение запущено корректно и без ошибок	1	Запустить приложение	Приложение запущено корректно и без ошибок
2	Поскроллить главный экран "Вверх", "Вниз"	Интерфейс при скроллинге отображается корректно и не тормозит	2	Кликнуть на блюдо в верхней панели блюд	Открылось подробное описание блюда	2	Проверить категории	Выбрать одну из категорий и перейти
3	Поскроллить панель верхних блюд	Блюда перемещаются без фризов	3	Просмотреть детально информацию о блюде	Страница скроллится, ошибок в отображении текста и картинок нет	3	Просмотреть информацию и описание блюда внутри категории	Текст и изображения категории отображаются корректно
			4	Нажать на кнопку Youtube	Проверить, что нас перекинуло на соответствующее видео на портале youtube.com			
Критерий приёмки								
Test case		Priority						
		1 High						
		2 High						
		3 High						
Тесты будут дополняться по мере включения в приложение новых функций								

Рисунок 19 – GUI тесты

Заключение

В ходе выполнения был проведен анализ предметной области мобильных приложений связанных с рецептами, были составлены диаграммы прецедентов, последовательностей, активности, состояния, взаимодействий, классов, объектов и развертывания.

В результате работы было реализовано Android приложение, которое позволяет осуществлять поиск рецептов различных блюд по категориям, по названию, по национальной кухне, а также имеет возможность добавлять рецепты в избранное.

Были выполнены следующие задачи:

1. Разработана Front-end часть приложения, находящаяся на телефоне пользователя;
2. Разработана Back-end часть приложения, развернутая на локальном сервере;
3. Была создана связь между Базой данных и Приложением с помощью REST API;
4. Разработана базы данных, расположенная на удаленном сервере;

Было проведено тестирование системы и установлено, что она работоспособна и функционирует корректно на практически всех входных данных.