

**Министерство науки и высшего образования Российской
Федерации**

**федеральное государственное автономное
образовательное учреждение высшего образования**

**«Самарский национальный исследовательский университет
имени академика С.П. Королева»**

Институт информатики и кибернетики

Кафедра технической кибернетики

Финальный отчёт

Дисциплина: «Технологии сетевого программирования»

Финализация приложения «Книга рецептов»

Выполнил: Диш Софья, Шкуркина Мария

Группа: 6303-010302D

Самара, 2025

АРХИТЕКТУРА ПРИЛОЖЕНИЯ

Проект представляет собой веб-приложение, предназначенное для работы с рецептами. Архитектура построена по модели "клиент-сервер" и включает следующие ключевые компоненты:

1. Клиентская часть (HTML + JS):

- Отвечает за отображение форм регистрации, входа, профиля и различных страниц с рецептами.
- Реализует взаимодействие с сервером через fetch-запросы с JWT-аутентификацией.

2. Серверная часть (Java + Spring):

- Обработывает все HTTP-запросы.
- Реализует REST API.
- Обеспечивает JWT-аутентификацию и управление пользователями, событиями и реакциями.

3. База данных (PostgreSQL):

- Хранит информацию о пользователях и рецептах.

4. Docker-контейнеризация:

- db (PostgreSQL):

Используется стандартный образ PostgreSQL с добавлением необходимых настроек для подключения API-контейнера.

- app (Spring Boot, HTML+JS)

Использует стандартизированный Dockerfile для сборки JAR-файлов с помощью Maven. Доступен по адресу: localhost:8080. Сервер ожидает соединение с базой данных.

ВЗАИМОДЕЙСТВИЕ КОМПОНЕНТОВ

- Запрос от клиента: Клиент отправляет запрос на сервер (например, для получения рецепта).
- Обработка запроса сервером: Сервер принимает запрос, обрабатывает его и обращается к базе данных для извлечения необходимой информации.
- Ответ базы данных: База данных возвращает запрашиваемую информацию серверу.
- Ответ сервера клиенту: Сервер формирует ответ и отправляет его обратно клиенту (например, возвращает рецепты определенной категории).

СТРУКТУРА БАЗЫ ДАННЫХ

- User(id_user, name, password, bday, phone, mail, user_role): Хранит информацию о пользователях.
- Recipe(id_recipe, id_user, title, id_category, description, manual, time, picture): Хранит информацию о рецептах.
- Ingredient(id_ingredient, title, unit_measure): Хранит информацию об ингредиентах.
- Composition_recipe(id_recipes, id_ingredient, quantity): Хранит информацию о том, какие ингредиенты входят в состав рецепта.
- Category(id_category, title): Хранит информацию о категориях.
- Favorite_recipe(id_user, id_recipe): Хранит информацию о том, какие рецепты добавлены в избранное.
- Token(id, access_token, refresh_token, is_logged_out, user_id): Хранит информацию о токенах.

СТРУКТУРА API

Метод	Название метода	URL
POST	Регистрация пользователя	/a/registration
POST	Вход пользователя	/a/login
POST	Смена пароля	/a/password_change
POST	Добавление рецепта	/api/recipes/create
GET	Получение списка рецептов	/api/recipes/search
GET	Получение выбранного рецепта	/api/recipes/getRecipe/{id}
GET	Получить все ингредиенты	/api/ingredients
POST	Создание ингредиента	/api/ingredients/create
GET	Вывод всех категорий	/api/categories
POST	Создание категории	/api/categories/create
GET	Получение одной категории	/api/categories/getCategory/{id}

Метод	Название метода	URL
GET	Получение профиля пользователя	/api/users/profile
GET	Получение всех рецептов пользователя	/api/users/myRecipe
GET	Получение любимых рецептов	/api/users/myFavouriteRecipe
POST	Добавление рецепта в избранное	/api/users/favourite-recipes/{recipeId}
DELETE	Удаление рецепта из избранного	/api/users/removeFavourite/{recipeId}

СТЕК ТЕХНОЛОГИЙ

- PostgreSQL
- Java + Spring
- HTML + JavaScript
- DevOps: Docker, Docker Compose

ДЕМОНСТРАЦИЯ (USE CASES)

1. Регистрация и вход:

- Пользователь переходит на /registration/ и создаёт учётную запись.
- Получает access_token и redirect'ится на /login/.

2. Отображение рецептов:

- Загружается соответствующая страница рецептов.
- Рецепты представляют собой карточки с названием и временем.

3. Отображение рецепта:

- Показ всей информации рецепта (название, категория, инструкция, описание, ингредиенты, время приготовления).
- Поиск по фильтру (ингредиенту, названию, категории).

4. Создание рецепта:

- Клик по кнопке создания — открывается модальное окно.
- Пользователь вводит данные, подтверждает — новый рецепт появляется в списке.

5. Добавление рецепта в избранное:

- На карточке можно поставить лайк на рецепт, и он появится на отдельной странице.

6. Создание категории и ингредиента:

- Кнопки "Создать категорию" и "Создать ингредиент" доступны только админу.

7. Выход из системы:

- Пользователь может выйти из системы, тогда ему будет доступен только просмотр существующих рецептов.

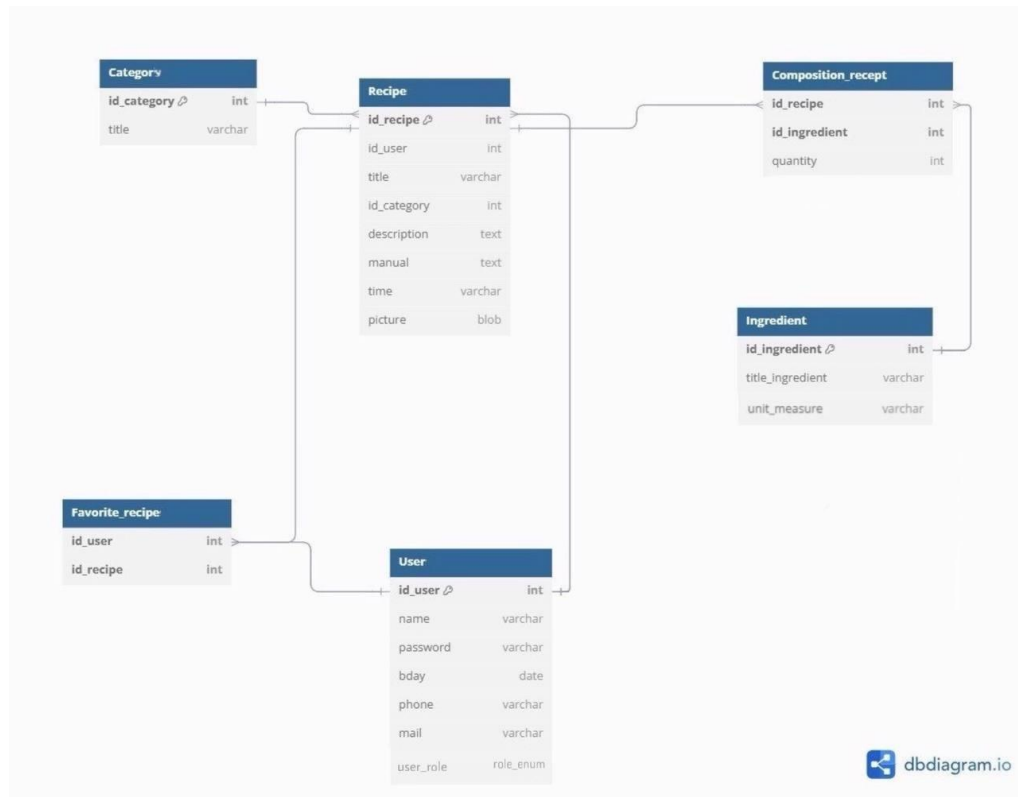


Рисунок 1– Диаграмма БД

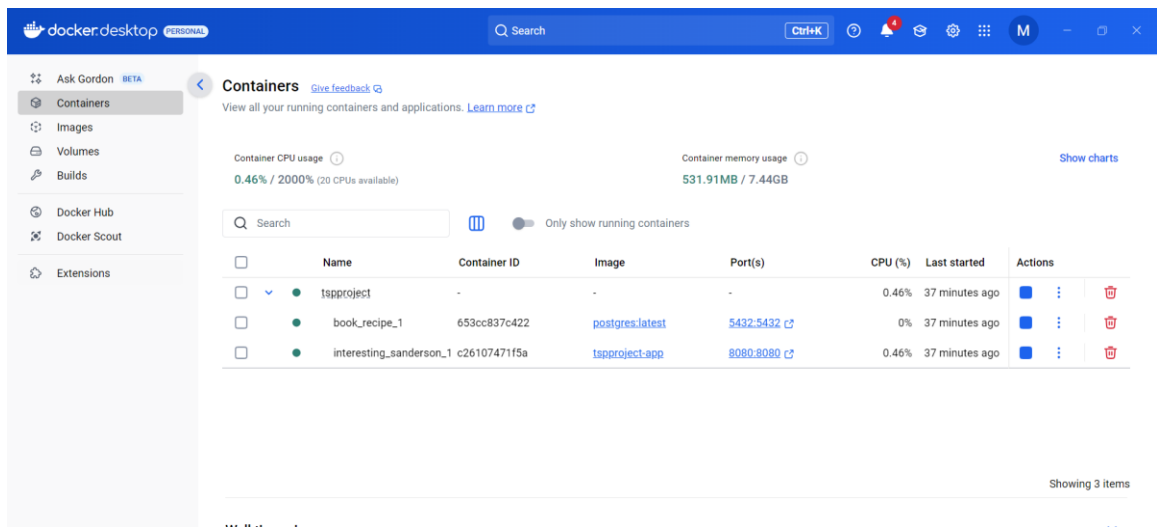


Рисунок 2 – Работающие контейнеры

Регистрация

Имя пользователя:

Пароль:

Телефон:

Почта:

Дата рождения:

Зарегистрироваться

Рисунок 3 – Страница регистрации

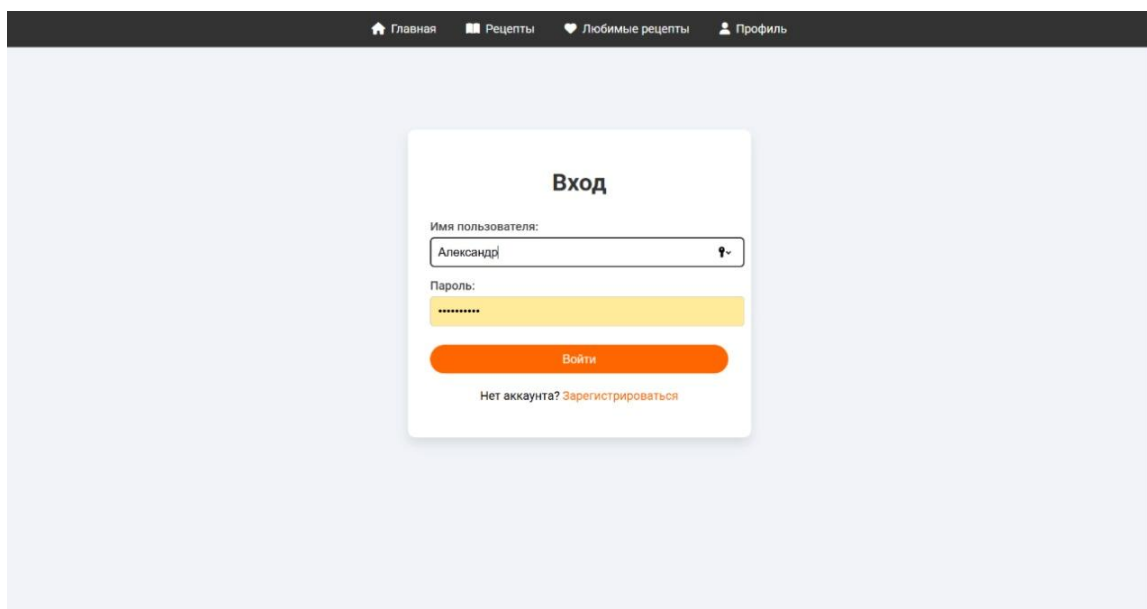


Рисунок 4 – Страница входа

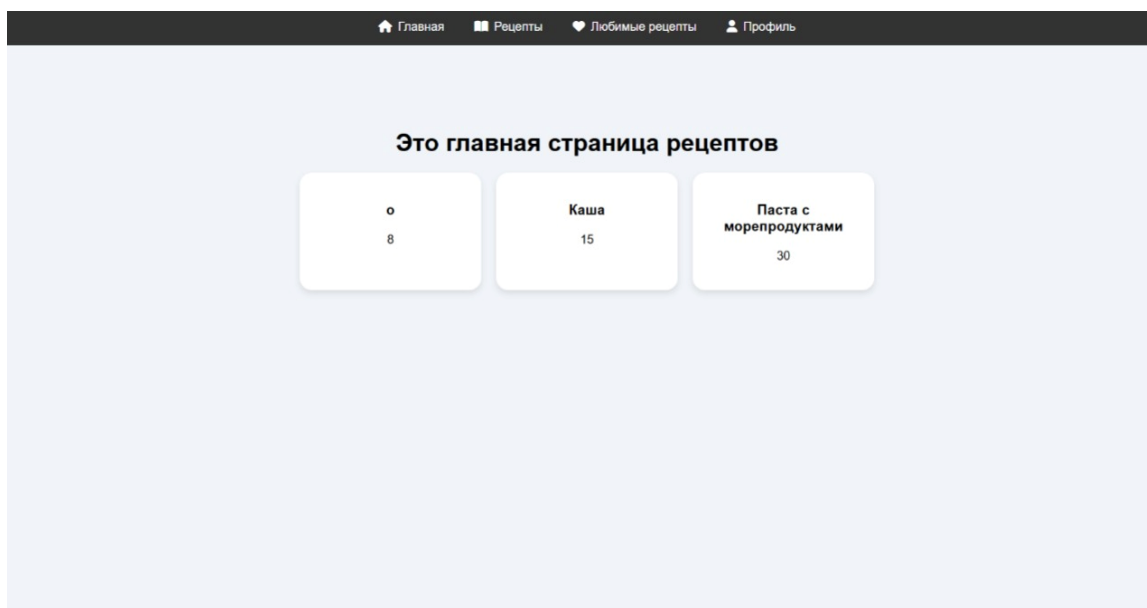


Рисунок 5 – Главная страница

Главная Рецепты Любимые рецепты Профиль

Создать новый рецепт

Название рецепта:

Введите название рецепта

Описание:

Введите описание рецепта

Инструкция:

Введите инструкцию

Время приготовления (в минутах):

Введите время приготовления

Категория:

Завтраки

Ингредиенты:

Молоко

Количество

Добавить ингредиент

Создать Отмена

Александр

Email: a@mail.ru

Телефон: 79871234

Дата рождения: 2000

Создать рецепт

Сменить пароль

Выйти из профиля

Рисунок 6 – Окно создания рецепта

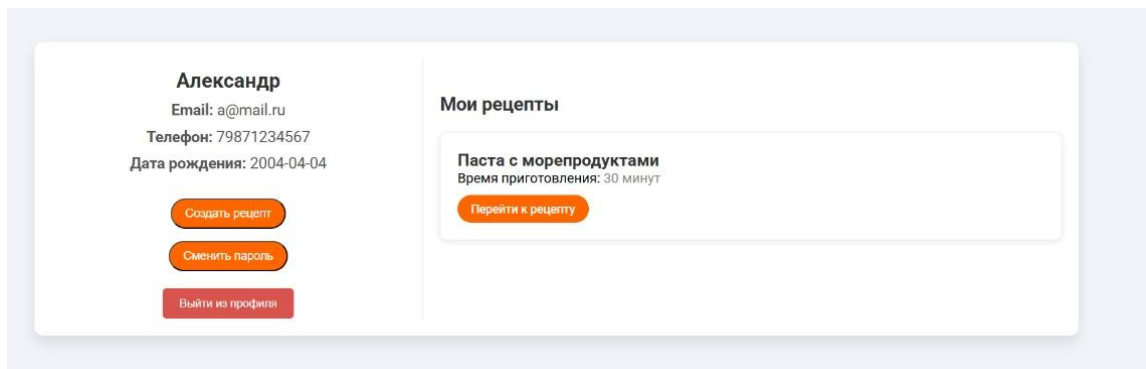


Рисунок 7 – Профиль обычного пользователя

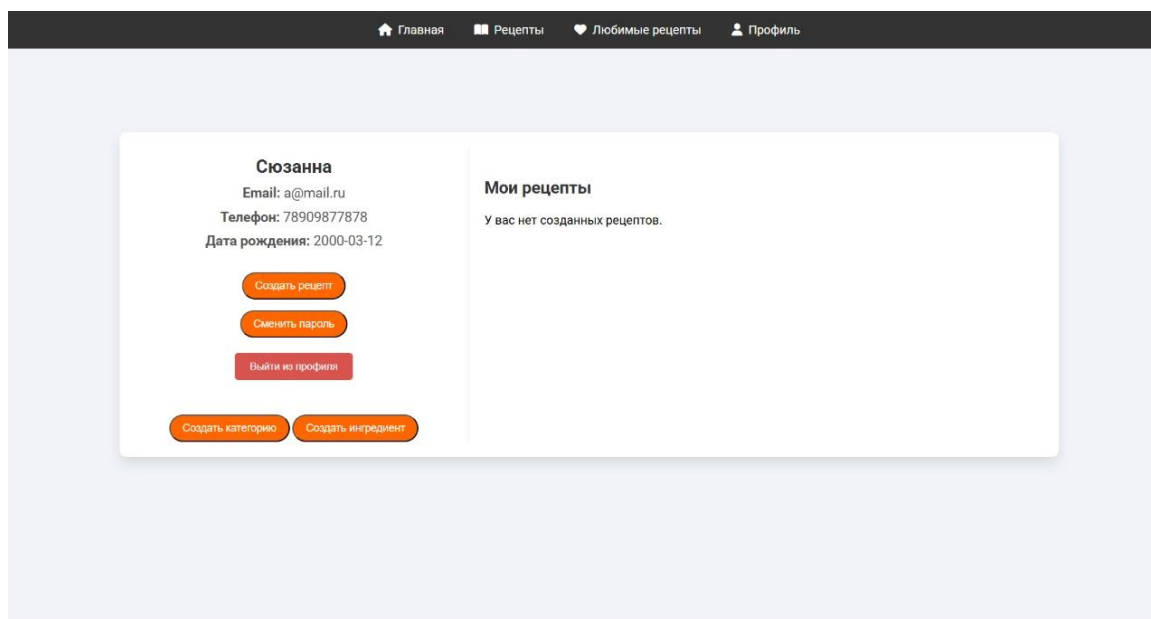


Рисунок 8 – Профиль администратора

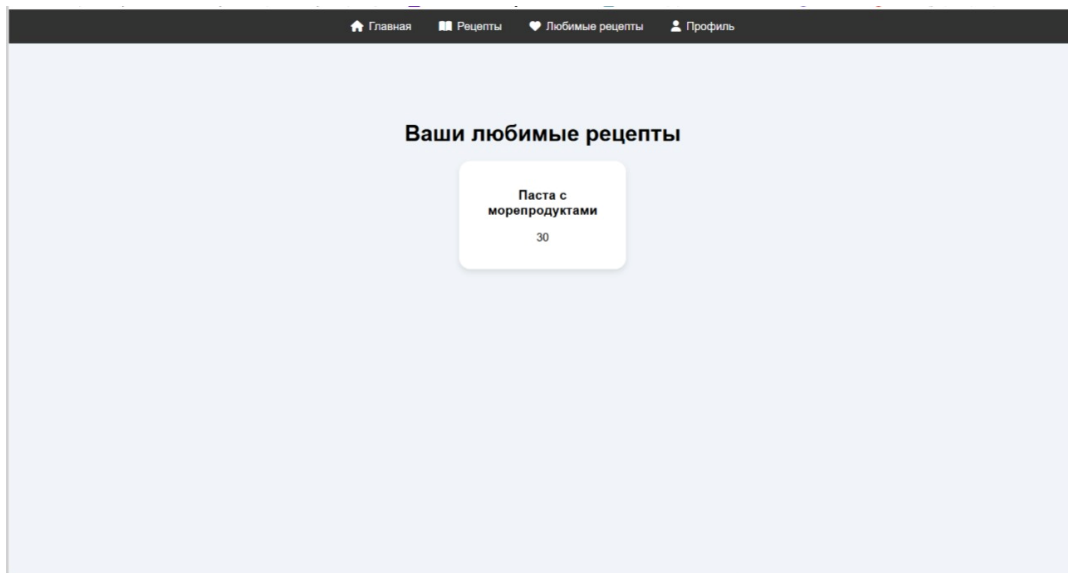


Рисунок 9 – Любимые рецепты

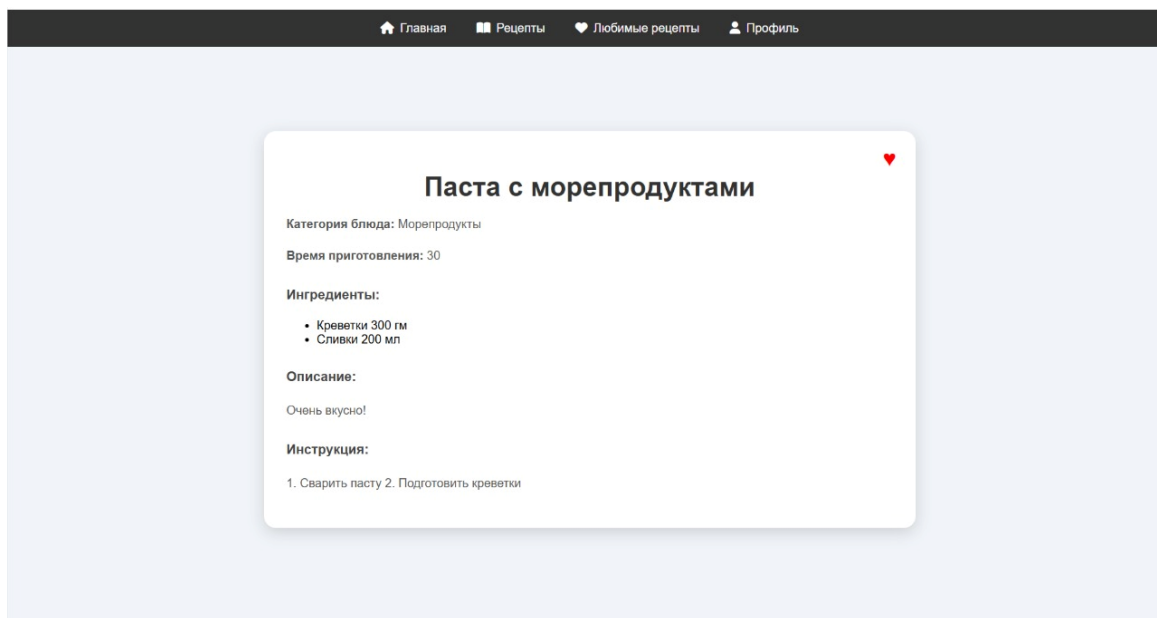


Рисунок 10 – Страница рецепта

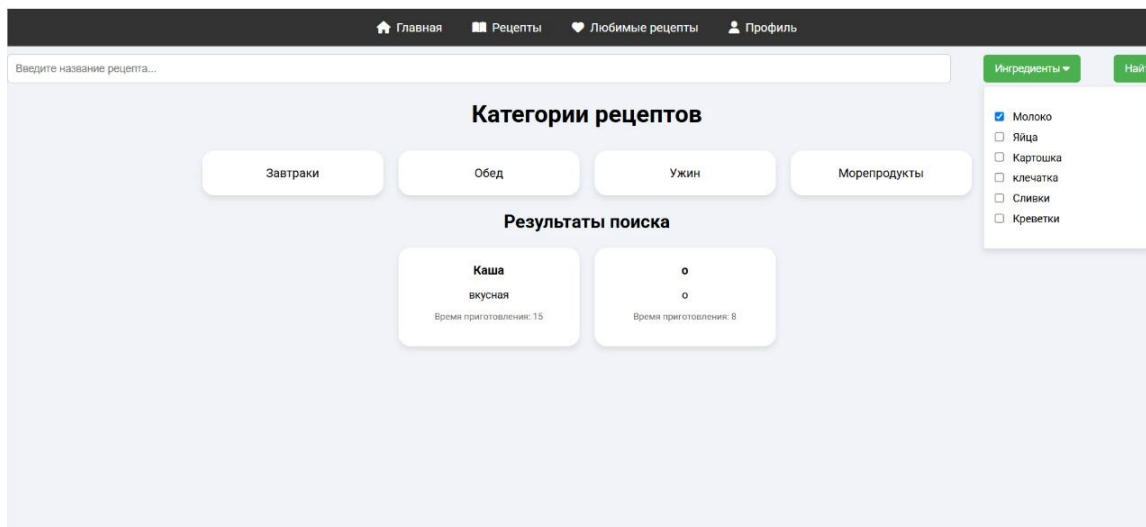


Рисунок 11 – Страница поиска рецептов по фильтрам, названию и категории

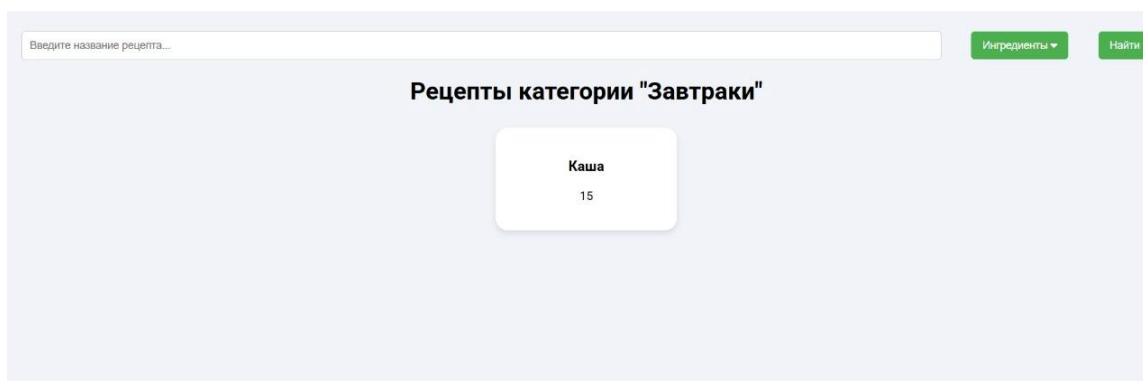


Рисунок 12 – Рецепты выбранной категории

