**Веб-приложение**

**«Система управления кулинарными рецептами и автоматизации списка продуктов»**

**Руководство программиста**

Киров, 2025 г.

**Аннотация**

Настоящее руководство программиста предназначено для специалистов, осуществляющих разработку, сопровождение и модификацию веб-приложения «Система управления кулинарными рецептами и автоматизации формирования списка продуктов».

Документ содержит сведения о структуре программного комплекса, архитектурных решениях, правилах настройки окружения, сборки, тестирования, резервного копирования и обновления компонентов системы.

Руководство программиста разработано с учётом требований ГОСТ 19.504–79 и ГОСТ Р 59795–2021.

**Содержание**

[1 Назначение и условия применения программы……………………………………………...2](#_Toc212555126)

[1.1 Назначение и функции программы 2](#_Toc212555127)

[1.2 Условия выполнения программы 3](#_Toc212555128)

[1.2.1 Требования к техническому обеспечению 3](#_Toc212555129)

[1.2.2 Требования к программному обеспечению 3](#_Toc212555130)

[1.2.3 Требования к информационному обеспечению 4](#_Toc212555131)

[2.1 Описание основных характеристик 5](#_Toc212555133)

[2.2 Режимы работы программы 5](#_Toc212555134)

[2.3 Временные характеристики 5](#_Toc212555135)

[2.4 Средства контроля и самовосстанавливаемости 6](#_Toc212555136)

[3 Обращение к программе……………………………………………………………………….7](#_Toc212555137)

[3.1 Процедура запуска системы 7](#_Toc212555138)

[3.2 Передача управления и параметров 7](#_Toc212555139)

[4 Входные и выходные данные………………………………………………………………….8](#_Toc212555140)

[4.1 Организация входной информации 8](#_Toc212555141)

[4.2 Организация выходной информации 8](#_Toc212555142)

[5 Сообщения……………………………………………………………………………………...9](#_Toc212555143)

[5.1 Системные сообщения 9](#_Toc212555144)

[5.2 Сообщения об ошибках 9](#_Toc212555145)

[5.3 Действия по сообщениям 9](#_Toc212555146)

# Назначение и условия применения программы

В данном разделе описаны назначение, основные функции и условия применения веб-приложения «Система управления кулинарными рецептами и автоматизации формирования списка продуктов».

## Назначение и функции программы

Веб-приложение предназначено для автоматизации процесса публикации, хранения и взаимодействия с кулинарными рецептами,

а также для формирования агрегированного списка продуктов, необходимых для приготовления выбранных блюд.

Система обеспечивает выполнение следующих функций:

* Авторизация и регистрация пользователей, включая восстановление и смену пароля;
* Создание, редактирование и удаление рецептов с возможностью прикрепления изображения и указания ингредиентов, тегов и описания;
* Просмотр рецептов других пользователей с пагинацией и фильтрацией по тегам;
* Добавление рецептов в избранное и просмотр списка избранных блюд;
* Подписка на авторов и формирование персональной ленты публикаций;
* Создание списка покупок, агрегирующего ингредиенты из выбранных рецептов и исключающего дублирующиеся позиции;
* Скачивание списка покупок в форматах .txt и .pdf;
* Редактирование пользовательского профиля, включая смену аватара и имени;
* Администрирование системы: управление тегами, ингредиентами и модерация пользовательского контента.

## Условия выполнения программы

В настоящем подразделе описаны условия, необходимые для корректного функционирования веб-приложения «Система управления кулинарными рецептами и автоматизации формирования списка продуктов».

### Требования к техническому обеспечению

Для работы системы требуется персональный компьютер или сервер, обеспечивающий возможность запуска контейнеризированного веб-приложения и стабильное соединение с интернетом.

Минимальные системные требования:

* операционная система: Windows 10, Linux (Ubuntu 22.04 LTS) или выше;
* оперативная память: не менее 2 ГБ;
* свободное место на жёстком диске: не менее 1 ГБ;
* разрешение экрана: не менее 1366×768 пикселей;
* установленный современный веб-браузер (Google Chrome, Mozilla Firefox, Safari и др.);
* наличие сетевого подключения;
* наличие клавиатуры и указательного устройства (мышь или тачпад).

### Требования к программному обеспечению

Для успешной разработки, развёртывания и функционирования веб-приложения используются следующие программные средства:

* интерпретатор языка Python версии 3.12+;
* фреймворк Django 5.0 с модулем Django REST Framework;
* библиотека Djoser для реализации аутентификации и управления пользователями;
* система управления базами данных PostgreSQL версии 16;
* веб-сервер Nginx;
* сервер приложений Gunicorn;
* система контейнеризации Docker и Docker Compose;
* среда разработки PyCharm / VS Code;
* система контроля версий Git;
* веб-браузер для взаимодействия с клиентской частью приложения.

### Требования к информационному обеспечению

Информационное обеспечение системы реализуется на основе реляционной базы данных PostgreSQL.

Все данные организованы в логически взаимосвязанные таблицы, обеспечивающие целостность и непротиворечивость данных.

Основные сущности БД:

* пользователи — данные о зарегистрированных участниках системы;
* рецепты — описание блюд, фотографии, автор и дата публикации;
* ингредиенты — перечень продуктов с единицами измерения;
* теги — классификация рецептов по категориям;
* список покупок — агрегированные ингредиенты для выбранных рецептов;
* избранное и подписки — данные о взаимодействиях между пользователями.

Каждая таблица имеет первичный ключ (id) и внешние связи для обеспечения целостности (ON DELETE CASCADE).

Хранилище данных поддерживает экспорт в форматах .sql, .csv, .pdf и .txt.

# Характеристика программы

В настоящем разделе приведено описание основных характеристик и особенностей веб-приложения «Система управления кулинарными рецептами и автоматизации формирования списка продуктов».

## Описание основных характеристик

Разрабатываемая система представляет собой веб-приложение клиент–серверной архитектуры, обеспечивающее пользователям возможность публикации, редактирования и просмотра кулинарных рецептов, а также автоматизацию формирования списка продуктов для выбранных блюд.

Система функционирует на основе Django + Django REST Framework и взаимодействует с базой данных PostgreSQL, предоставляя REST API для клиентских запросов.

Серверная часть обрабатывает данные пользователей, контролирует права доступа, формирует списки ингредиентов и обеспечивает экспорт данных в форматах .txt и .pdf.

Приложение ориентировано на три категории пользователей:

* Гость — просмотр рецептов и страниц без авторизации;
* Зарегистрированный пользователь — публикация рецептов, создание списков покупок, подписки и избранное;
* Администратор — управление тегами, ингредиентами, пользователями и модерация контента.

## Режимы работы программы

Веб-приложение поддерживает несколько режимов работы, определяемых ролью пользователя:

Режим «Гость»:

* просмотр главной страницы с рецептами;
* использование фильтров и тегов;
* ознакомление со страницами «О проекте» и «Технологии»;
* попытка взаимодействия с функционалом избранного, подписок и покупок с предложением авторизоваться.

Режим «Пользователь»:

* регистрация, вход и смена пароля;
* создание, редактирование и удаление собственных рецептов (CRUD);
* добавление рецептов в избранное и удаление из него;
* подписка на других авторов и просмотр ленты публикаций;
* добавление рецептов в список покупок и выгрузка агрегированных ингредиентов в .txt или .pdf;
* изменение личных данных и аватара профиля.

Режим «Администратор»:

* управление тегами и ингредиентами;
* модерация рецептов и пользовательских данных;
* просмотр статистики активности пользователей;
* редактирование или удаление контента, нарушающего правила.

Все режимы доступны через веб-браузер при наличии подключения к сети Интернет.

## Временные характеристики

Веб-приложение «Система управления кулинарными рецептами и автоматизации формирования списка продуктов» обеспечивает стабильную работу при стандартных условиях эксплуатации и подключении к сети Интернет со скоростью не ниже 10 Мбит/с.

Показатели производительности:

* время отклика интерфейса при навигации по страницам — не более 3 секунд;
* загрузка главной страницы с рецептами — не более 5 секунд;
* выполнение операций CRUD с рецептами (создание, редактирование, удаление) — не более 2 секунд;
* фильтрация рецептов по тегам или авторам — до 3 секунд;
* формирование и скачивание списка покупок (.txt/.pdf) — не более 7 секунд;
* авторизация пользователя и получение токена доступа — не более 2 секунд;
* обновление страницы профиля или аватара — не более 3 секунд.

Время восстановления работоспособности системы после сбоя составляет не более 2 часов, включая процедуры восстановления контейнеров и базы данных PostgreSQL из резервных копий и проверку целостности файлов конфигурации Docker.

## Средства контроля и самовосстанавливаемости

Система реализует комплексный подход к контролю целостности данных и самовосстановлению, основанный на принципах надёжности и отказоустойчивости серверных компонентов.

Контроль целостности данных обеспечивается за счёт:

* двухуровневой валидации вводимой информации — на стороне клиента (JavaScript/HTML5) и сервера (Django REST Framework);
* строгой проверки форматов входных данных и загружаемых изображений рецептов (допускаются только .jpg, .png);
* обеспечения ссылочной целостности в базе данных PostgreSQL с помощью внешних ключей и каскадных операций;
* транзакционной обработки операций, предотвращающей потерю данных при ошибках записи.

Механизмы самовосстановления системы включают:

* автоматическое сохранение пользовательской сессии при длительной неактивности;
* регулярное резервное копирование базы данных с помощью встроенных CRON-задач Docker;
* возможность восстановления данных из резервной копии в течение 2 часов после критического сбоя;
* автоматическую перезапуску контейнеров при сбое сервисов nginx, gunicorn или postgres;
* систему логирования и мониторинга ошибок через стандарт Django (logging + middleware уведомлений).

Пользовательский контроль ошибок реализуется посредством:

* отображения информативных сообщений об ошибках при вводе данных;
* логирования всех исключений в серверном журнале;
* уведомлений администратора о критических событиях в системе.

# Обращение к программе

В настоящем разделе описаны процедуры запуска, инициализации и взаимодействия с веб-приложением «Система управления кулинарными рецептами и автоматизации формирования списка продуктов».

## Процедура запуска системы

Запуск приложения осуществляется через веб-браузер или локальный сервер в среде Docker.

Для начала работы необходимо выполнить следующие действия:

* Клонировать репозиторий проекта с официального источника: https://github.com/username/recipe-manager
* Перейти в корневую директорию проекта: cd recipe-manager
* Запустить контейнеры приложения командой: docker-compose up -d
* После успешного развёртывания открыть браузер и перейти по адресу: http://127.0.0.1/ или http://localhost/.
* После загрузки веб-интерфейса пользователь попадает на страницу авторизации, где необходимо ввести логин и пароль, указанные при регистрации.
* После успешной проверки учетных данных система перенаправляет пользователя:
* на главную страницу (для гостей и зарегистрированных пользователей);
* в панель администратора Django (для пользователей с ролью администратора).

После входа веб-приложение полностью готово к работе и обеспечивает доступ к функциям, предусмотренным ролью пользователя.

## Передача управления и параметров

Управление системой выполняется через интерактивный веб-интерфейс, построенный на основе Django Templates и REST API.

Передача параметров между клиентом и сервером осуществляется с помощью HTTP-запросов:

* GET — получение данных (рецепты, ингредиенты, теги, профили);
* POST — создание новых объектов (рецепты, списки покупок);
* PUT/PATCH — обновление данных (редактирование рецепта, изменение профиля);
* DELETE — удаление объектов (рецептов, избранных, покупок).

Пользователь взаимодействует с системой через:

* формы ввода данных (создание и редактирование рецептов);
* выпадающие списки и фильтры (поиск по тегам, авторам, ингредиентам);
* кнопки действий («Добавить в избранное», «Скачать список покупок»);
* диалоговые окна подтверждения (удаление рецепта, выход из аккаунта).

Все операции сопровождаются валидацией данных в реальном времени на стороне клиента и сервера.

Критические действия требуют подтверждения через всплывающее окно.

После выполнения операции данные автоматически обновляются на экране без необходимости перезагрузки страницы (через AJAX или fetch-запросы).

# Входные и выходные данные

В данном разделе описана организация используемой входной и выходной информации веб-приложения «Система управления кулинарными рецептами и автоматизации формирования списка продуктов».

## Организация входной информации

Входная информация поступает в систему от пользователей через веб-интерфейс и API-запросы.

Она включает несколько категорий данных в зависимости от выполняемых действий и роли пользователя.

Основные категории входных данных:

* Учетные данные пользователей — логин, пароль, имя и адрес электронной почты, используемые при регистрации и авторизации. Все данные проходят обязательную валидацию на стороне клиента и сервера.
* Профиль пользователя — аватар, краткая информация, изменение пароля и контактные данные.
* Информация о рецептах — название, список ингредиентов, количество, единицы измерения, теги, описание, способ приготовления и изображения блюд (в форматах .jpg, .png, размером до 5 МБ).
* Информация о взаимодействиях пользователей — добавление рецептов в избранное, подписка и отписка от авторов.
* Информация для формирования списка покупок — набор выбранных рецептов, из которых система агрегирует ингредиенты.
* Параметры фильтрации и поиска — теги, авторы, ингредиенты, диапазон времени приготовления, наличие изображений.
* Данные администрирования — добавление и редактирование тегов, ингредиентов, управление пользователями и модерация контента.

Все входные данные проходят многоуровневую проверку:

* валидация форм и форматов файлов на клиенте (HTML5, JavaScript);
* серверная проверка типов данных, длины строк и уникальности записей через Django REST Framework;
* защита от XSS и CSRF-атак при передаче данных через HTTP-запросы.

## Организация выходной информации

Выходная информация формируется системой в зависимости от выполняемых операций, типа запроса и роли пользователя.

Основные виды выходных данных:

* Визуальные данные веб-интерфейса:
* страницы с каталогом рецептов, профилем пользователя и лентой публикаций;
* таблицы с ингредиентами, тегами и списками покупок;
* карточки рецептов с изображением, описанием, тегами и автором;
* формы редактирования и диалоговые окна подтверждения.
* Отчётные и экспортные данные:
* агрегированные списки ингредиентов для выбранных рецептов в форматах .txt и .pdf;
* результаты фильтрации и поиска рецептов (в формате JSON при использовании API);
* административные отчёты (журнал действий, статистика пользователей).
* Системные уведомления:
* сообщения об успешных операциях (создание, редактирование, удаление);
* предупреждения при некорректном вводе данных;
* уведомления об ошибках сервера или отсутствии соединения.
* Служебная информация:
* логи веб-приложения (ошибки, системные события, обращения к API);
* резервные копии базы данных PostgreSQL;
* метрики производительности (время отклика, количество активных пользователей, нагрузка на сервер).

# Сообщения

В данном разделе описаны системные и пользовательские сообщения, выдаваемые в ходе работы веб-приложения «Система управления кулинарными рецептами и автоматизации формирования списка продуктов», а также рекомендации по действиям при их получении.

## Системные сообщения

Система информирует пользователя о результатах выполнения операций посредством всплывающих уведомлений,

визуальных индикаторов статуса и динамических изменений интерфейса.

Основные случаи отображения системных сообщений:

* при успешной авторизации — пользователь автоматически перенаправляется на главную страницу, соответствующую его роли (пользователь или администратор);
* при создании или редактировании рецепта — отображается уведомление «✅ Рецепт успешно сохранён»;
* при удалении рецепта — всплывающее сообщение «🗑 Рецепт удалён»;
* при добавлении рецепта в избранное — уведомление «⭐ Рецепт добавлен в избранное» (или «Удалён из избранного» при повторном нажатии);
* при подписке/отписке от автора — сообщение «Вы подписались на автора» / «Подписка отменена»;
* при генерации списка покупок — система уведомляет «🛒 Список покупок успешно сформирован»;
* при экспорте списка покупок — автоматически начинается загрузка файла в формате .txt или .pdf, без дополнительных уведомлений;
* при обновлении профиля — сообщение «Изменения успешно сохранены»;
* при модерации администратором — система выводит «Изменения применены» после завершения операции.

Изменения состояния данных визуально отражаются в интерфейсе — обновлением карточек, изменением цвета кнопок, статусных меток и динамических элементов без необходимости перезагрузки страницы.

## Сообщения об ошибках

Система классифицирует ошибки по категориям и отображает информативные уведомления с пояснениями.

|  |  |  |
| --- | --- | --- |
| **Ситуация** | **Сообщение** | **Причина** |
| Ошибка входа | «Введены неверные логин или пароль» | Неправильные учетные данные |
| Ошибка регистрации | «Пользователь с таким email уже существует» | Нарушение уникальности |
| Ошибка при сохранении рецепта | «Не все обязательные поля заполнены» | Пропуск полей формы |
| Ошибка загрузки изображения | «Недопустимый формат файла. Разрешены JPG и PNG» | Ошибка формата файла |
| Ошибка добавления в избранное | «Ошибка выполнения запроса. Повторите попытку позже» | Потеря соединения с сервером |
| Ошибка экспорта | «Не удалось сформировать файл» | Ошибка генерации PDF |
| Ошибка доступа | «Недостаточно прав для выполнения операции» | Попытка администратора или пользователя вне своей роли |
| Системная ошибка | «Произошла ошибка на сервере. Повторите попытку позже» | Непредвиденный сбой приложения |

## Действия по сообщениям

Рекомендации пользователю при возникновении ошибок:

* при ошибке авторизации — проверить правильность введённого логина и пароля, при необходимости воспользоваться функцией восстановления;
* при ошибке загрузки файла — убедиться, что изображение соответствует требованиям формата и не превышает лимит по размеру (5 МБ), затем повторить загрузку;
* при ошибках валидации форм — заполнить все обязательные поля и проверить корректность форматов данных;
* при ошибке экспорта списка покупок — убедиться в стабильности интернет-соединения и повторить операцию;
* при ошибках доступа или отказе сервера — выполнить повторную попытку позже или обратиться к администратору системы через встроенную форму обратной связи.