**Федеральное государственное бюджетное образовательное**

**учреждение высшего образования «Российский экономический университет имени Г.В. Плеханова»**

Факультет Плехановская школа бизнеса «Интеграл»

**ДОКУМЕНТАЦИЯ**

по дисциплине «Технологии разработки веб-приложений на Django»

Тема: «Веб-приложение для пекарня “Веселый хлебушек” на Django»

**Выполнили:**

Белинская Яна Валентиновна

Абукаева Илина Рамазановна

Студенты Факультета Плехановская школа бизнеса «Интеграл»

Группа 15.07В-ПИ1/23б

**Преподаватель:**

Нефедов Юрий Викторович

2024

Москва

# Введение

В данном проекте разрабатывается веб-сайт для пекарни под названием "Веселый хлебушек". Сайт будет предоставлять функционал для оформления заказов, выбора продукции, обработки платежей, а также отслеживания доставки. Основная цель разработки — создание удобной и функциональной онлайн-платформы для взаимодействия с клиентами, где пользователи могут выбрать из разнообразных продуктов пекарни и заказать их с доставкой на дом.

Для реализации веб-сайта используется фреймворк **Django**, который позволяет быстро разрабатывать и развертывать веб-приложения с использованием Python. В проекте будут реализованы основные функциональные требования для работы с продуктами, заказами, клиентами и платежами, а также обеспечена высокая доступность и удобство интерфейса.

**Основная задача:**

**Разработка веб-сайта для пекарни "Веселый хлебушек" с функционалом для оформления заказов, выбора продукции, обработки платежей и отслеживания доставки.**

**Подзадачи:**

1. **Анализ требований.**

2. **Проектирование интерфейса.**

3. **Настройка окружения.**

4. **Разработка моделей.**

5. **Создание функционала для работы с продуктами.**

6. **Разработка функционала для оформления заказов.**

7. **Тестирование.**

8. **Запуск и развертывание.**

# Основные функциональные и нефункциональные требования

## Функциональные требования сайта для пекарни

## 1. Регистрация и аутентификация пользователей:

## Возможность зарегистрироваться на сайте с указанием личных данных (имя, email, телефон, пароль).

## Аутентификация пользователей для оформления заказов и отслеживания статуса заказа. (email и пароль)

## 2. Просмотр продуктов:

## Пользователь должен иметь возможность просматривать доступные для покупки продукты: хлеб, булочки и другие изделия.

## У каждого продукта должна быть подробная информация (название, описание, цена, изображение).

## 3. Оформление заказа:

## Пользователь может добавить несколько продуктов в корзину и удалить лишние.

## Возможность оформления заказа с указанием количества каждого продукта и суммы.

## Выбор способа оплаты (оплата - При получении).

## 4. Управление заказами:

## Пользователь видит какие заказы он уже сделал и в какое время их примерно доставят.

## Доставка заказа на указанный клиентом адрес.

## 5. Административная панель:

## Возможность администрировать сайт через Django-админку.

## Управление продуктами: добавление, редактирование, удаление продуктов.

## Управление заказами.

## Нефункциональные требования

1. Юзабилити:

Удобный и интуитивно понятный интерфейс.

Простота в использовании как для клиента, так и для администратора пекарни.

2. Безопасность:

Все пользовательские данные должны быть защищены с использованием шифрования.

3. Поддерживаемость:

Документация

Модульность

4. Кроссбраузерность:

Сайт должен корректно отображаться в основных браузерах.

5. Скалируемость:

Горизонтальная и вертикальная масштабируемость.

# Схемы

1. Схема прецендентов

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Рисунок - Схема прецендентов

1. Диаграмма классов

Изображение выглядит как текст, диаграмма, снимок экрана, План

Автоматически созданное описание

Рисунок - Диаграмма классов

1. Диаграмма последовательностей

Изображение выглядит как текст, снимок экрана, диаграмма

Автоматически созданное описание

Рисунок - Диаграмма последовательностей

# Древовидная структура сайта

1. Древовидная структура сайта

Моя пекарня

│

├── Главная страница (home.html)

│ ├── Добро пожаловать в пекарню

│ └── Приветственное сообщение

│

├── Каталог продуктов (catalog.html)

│ ├── Просмотр всех продуктов

│ │ ├── Продукт 1

│ │ ├── Продукт 2

│ │ └── Продукт 3

│ ├── Фильтры товаров

│ └── Поиск продуктов

│

├── Корзина (cart.html)

│ ├── Список товаров в корзине

│ │ ├── Продукт 1

│ │ ├── Продукт 2

│ │ └── Продукт 3

│ ├── Подсчет общей стоимости

│ └── Оформление заказа

│

├── Оформление заказа (checkout.html)

│ ├── Список товаров в заказе

│ ├── Выбор способа оплаты

│ └── Ввод адреса доставки

│

├── Профиль пользователя (profile.html)

│ ├── Личные данные

│ │ ├── Имя

│ │ ├── Электронная почта

│ │ └── Телефон

│ ├── История заказов

│ └── Выход

│

├── Страница входа (login.html)

│ ├── Ввод почты

│ ├── Ввод пароля

│ └── Ссылка на регистрацию

│

├── Страница регистрации (register.html)

│ ├── Ввод полного имени

│ ├── Ввод электронной почты

│ ├── Ввод телефона

│ └── Ввод пароля

│

└── Доставка (delivery\_detail.html)

├── Информация о заказе

│ ├── Номер заказа

│ ├── Дата доставки

│ ├── Статус доставки

│ └── Способ оплаты

└── Адрес доставки

# Структура шаблонов

Основные структуры шаблонов:

**1. base.html:**

Это базовый шаблон, от которого наследуются другие страницы. В нем находится общая структура HTML-документа, включая заголовок, подключение стилей (CSS) и скриптов (JavaScript), а также навигационное меню.

В блоке <title> определяется название страницы.

В блоках navbar и footer определяется навигация и подвал сайта, которые используются на всех страницах.

В основном контенте, на месте {% block content %}, будут вставляться уникальные данные каждой страницы.

**2. catalog.html:**

Страница каталога товаров, где пользователи могут просматривать доступные продукты.

Для каждого продукта отображается изображение, название, описание, цена и категория.

Есть кнопка "Добавить в корзину" для добавления товаров в корзину.

**3. cart.html:**

Страница корзины покупок, где отображаются добавленные товары, их количество, цена и возможность изменить количество товаров или удалить их.

Подсчитывается общая сумма покупок и есть кнопка для оформления заказа.

**4. checkout.html:**

Страница оформления заказа, где пользователи могут увидеть список товаров в корзине, выбрать способ оплаты, ввести адрес доставки и оформить заказ.

**5. profile.html:**

Личный кабинет пользователя, где отображаются его данные (например, имя, почта, телефон) и возможность выйти из аккаунта.

**6. login.html:**

Страница входа в систему для авторизации пользователя по электронной почте и паролю.

**7. register.html:**

Страница регистрации, где новый пользователь может создать аккаунт, введя свое имя, электронную почту, телефон и пароль.

**8. delivery\_detail.html:**

Страница с информацией о доставке, где отображаются детали доставки, такие как номер заказа, дата доставки, статус и способ оплаты.