

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ
УНИВЕРСИТЕТ»
ИНСТИТУТ ПЕРСПЕКТИВНОЙ ИНЖЕНЕРИИ
ДЕПАРТАМЕНТ ЦИФРОВЫХ, РОБОТОТЕХНИЧЕСКИХ СИСТЕМ И
ЭЛЕКТРОНИКИ

КУРСОВАЯ РАБОТА

по дисциплине

«Интернет-технологии»

на тему:

«Разработка веб-приложения для управления задачами»

Выполнил:

Милосердов Илья Яковлевич

Студент 4 курса группы _ПИН-б-3-22-1

Направления подготовки

09.03.03 Прикладная информатика

заочной формы обучения

Проверил:

Старший преподаватель департамента цифровых, робототехнических систем
и электроники института перспективной инженерии

Щеголев А.А.

Ставрополь, 2026

Пояснительная записка к курсовой работе

Тема: Разработка веб-приложения для управления задачами

Введение

Целью данной курсовой работы является разработка веб-приложения для управления задачами (Task Manager). Приложение предназначено для организации личного и командного времени, отслеживания статуса выполнения задач и повышения продуктивности.

1. Анализ предметной области

В современном мире управление задачами является ключевым аспектом успешной деятельности. Существующие решения часто либо слишком сложны, либо имеют недостаточный функционал. Разрабатываемое приложение должно сочетать простоту использования с современными подходами к разработке.

2. Техническое задание

Функциональные требования:

- Регистрация и аутентификация пользователей.
- Создание, редактирование и удаление задач.
- Просмотр списка задач с фильтрацией по статусу (Новая, В работе, Выполнена).
- Назначение приоритетов задачам.

Нефункциональные требования:

- Клиент-серверная архитектура.
- Использование Docker для контейнеризации.
- Современный пользовательский интерфейс (SPA).

3. Проектирование

Архитектура системы:

Система построена по принципу разделения ответственности (Backend и Frontend).

- Backend: Python (Django + Django REST Framework). Предоставляет REST API.
- Frontend: TypeScript (React + Vite). Реализует пользовательский интерфейс.
- База данных: SQLite (для разработки), PostgreSQL (в продакшене).

Схема базы данных (Модели):

- User: Стандартная модель пользователя Django.
- Task:
 - `title` (Название)
 - `status` (Статус)
 - `priority` (Приоритет)
 - `assigned_to` (Исполнитель)

4. Реализация

Использованные технологии:

- Язык программирования: Python 3.10, TypeScript.
- Фреймворки: Django 5.0, React 18, TailwindCSS.
- Инструменты: Docker, Git.

Описание программных модулей:

- `backend/tasks`: Модуль управления задачами.

- `frontend/src/pages/TaskBoard`: Компонент отображения доски задач.

5. Тестирование

Проведено функциональное тестирование основных сценариев:

1. Успешная регистрация и вход.
2. Создание задачи (отображается в колонке "Новая").
3. Смена статуса задачи.

Заключение

В ходе курсовой работы было разработано и протестировано веб-приложение для управления задачами. Все требования технического задания выполнены.