

Проект: Telegram-бот “Умный менеджер задач с приоритетами”

1. Цель проекта

Разработать Telegram-бота, который позволяет пользователю создавать задачи, управлять ими, получать напоминания и автоматически определять тип и приоритет задачи с помощью искусственного интеллекта (ИИ).

2. Функциональные требования

2.1. Основные функции бота

Бот должен уметь:

1. Создавать новую задачу

- Пользователь вводит описание задачи.
- Бот фиксирует:
 - текст задачи
 - дату/время дедлайна (если есть)
 - статус (активная / выполнена)
 - приоритет (низкий / средний / высокий — можно вручную или через ИИ)

2. Просматривать задачи

- Показать все задачи списком
- Показать только активные
- Показать выполненные

3. Редактировать задачу

- изменить текст задачи
- изменить дедлайн
- изменить приоритет
- изменить статус

4. Удаление задачи

5. Система напоминаний

- Если у задачи есть дедлайн, бот напоминает заранее (за 30 мин / 1 час / 1 день — на выбор)

3. ИИ-функционал (обязательно)

Бот должен использовать ИИ для:

3.1. Классификация типа задачи

По тексту задачи бот определяет её категорию:

- учёба
- работа
- дом
- личное
- другое

3.2. Прогнозирование приоритета

ИИ должен по описанию задачи определить приоритет:

- низкий
- средний
- высокий

Это может работать через:

- ML-модель
- или простую rule-based NLP-логику

4. Хранилище данных (на выбор)

Вариант 1 — JSON-файл (простой)

- Хранится локально в проекте: `data/tasks.json`
- Формат: список словарей

Вариант 2 — SQLite база данных (продвинутый)

- Файл: `data/tasks.db`
- Требуются таблицы:
 - `tasks` (`id`, `text`, `deadline`, `priority`, `status`, `category`, `user_id`)

5. Нефункциональные требования

1. Бот должен работать 24/7 (можно использовать:
 - PythonAnywhere
 - Render
 - Railway
 - локальный запуск
2. Код должен быть структурирован на модули.
3. Обязательное использование виртуального окружения: `venv`.
4. Должен быть README-файл с инструкцией по запуску.
5. Обязательное логирование ошибок.

6. Технологии

- Python 3.x
- Библиотека для Telegram: **aiogram** или **python-telegram-bot**
- NLP / ИИ:
 - sklearn
 - или HuggingFace
 - или простые правила

7. Отчётность

Студент должен предоставить:

1. Готовый проект (папка с кодом)
2. Отчёт в PDF, где отражено:
 - 1) цель проекта
 - 2) архитектура бота
 - 3) выбранное хранилище
 - 4) реализация ИИ-части
 - 5) примеры работы
 - 6) выводы