**2025.08.11 ТЗ. RAG-система + Telegram-бот. Ассистент для агробизнеса**

**1. Описание проекта**

**1.1. Определение проблемы и идеи проекта**

**Проблема:**  
Аграрные специалисты, фермеры и владельцы сельхозбизнеса тратят много времени на поиск информации в инструкциях, регламентах, методичках и научных материалах. Информация часто хранится в разных форматах (PDF, DOCX, TXT), что усложняет поиск и обработку.

**Решение:**  
Создание интеллектуального Telegram-бота, который использует RAG-технологию (Retrieval-Augmented Generation) для извлечения релевантных данных из базы знаний и генерации точных ответов на вопросы пользователя.

**Идея:**  
Разработать Telegram-ассистента для агробизнеса, который:

* Принимает вопросы в чате.
* Находит нужную информацию в базе знаний (документы TXT, DOCX, PDF).
* Отвечает понятным языком с указанием источников.
* Работает 24/7 и ведёт аналитику запросов.

**1.2. Цель проекта**

Интеграция RAG-системы в Telegram-бота с возможностью пополнения базы знаний и автоматической индексации новых документов.

**2. Используемые технологии**

* **Язык программирования:** Python
* **Telegram API:** python-telegram-bot
* **LLM API:** OpenAI API (Chat Completions, Embeddings)
* **Обработка данных:** PyPDF2, python-docx, tiktoken
* **Хранение эмбеддингов:** FAISS
* **Интеграция RAG:** собственная реализация, опционально LangChain
* **База данных:** SQLite (лог запросов, статистика)
* **Развёртывание:** VPS или облачный сервер (DigitalOcean, AWS, Hetzner)

**3. Среда для разворачивания**

* VPS с Python 3.10+
* Доступ к Telegram Bot API Token
* Директория /agrobot/data/knowledge для базы знаний
* Возможность обновления базы знаний без остановки бота

**4. Основные функции**

**4.1. Функциональные требования**

**Для пользователя:**

1. Отправка вопроса в чат бота.
2. Получение ответа на основе базы знаний.
3. Указание источника данных (файл, страница).

**Для администратора:**

1. Загрузка новых файлов в базу знаний (PDF, DOCX, TXT).
2. Автоматическая индексация без перезапуска бота.
3. Просмотр статистики запросов и затрат на API.

**4.2. Ключевые технические детали**

**Архитектура:**

* Модуль Telegram-бота (bot.py) — обработка команд и сообщений.
* Модуль RAG (rag\_pipeline.py) — поиск релевантного контекста + генерация ответа.
* Модуль индексации (create\_index.py) — добавление новых документов.
* SQLite — хранение логов запросов и статистики.

**Структура базы данных:**

* users — user\_id, username, first\_seen.
* query\_logs — id, user\_id, query, response, tokens\_prompt, tokens\_completion, cost\_usd, timestamp.

**5. Реализация проекта**

**5.1. Этапы разработки**

1. **Анализ требований и проектирование.**
2. **Разработка RAG-системы.**
3. **Интеграция с Telegram-ботом.**
4. **Тестирование (разные сценарии использования).**
5. **Развёртывание на сервере.**

**5.2. Тестирование**

* Проверка работы поиска и генерации ответа.
* Проверка корректной индексации новых документов.
* Тестирование нагрузки (много одновременных запросов).

**6. Внедрение и поддержка**

1. Запуск бота на сервере.
2. Передача администратору панели управления документами.
3. Обучение пользованию системой.
4. Возможность доработки под дополнительные задачи (CRM, интеграция с API поставщиков).

**7. Сроки, стоимость и условия**

* **Сроки:** 14–21 календарный день.
* **Стоимость:** от 95 000 рублей (в зависимости от функционала).
* **Условия:** работа по договору с ИП, предоплата 50%.