

## Техническое тестовое задание на позицию AI Lead разработчика

### Цель задания

Оценить ваши практические навыки в разработке ИИ-решений, включая работу с Retrieval-Augmented Generation (RAG), языковыми моделями (LLM) и базовой интеграцией.

### Формат выполнения

Задание выполняется дома.

Дедлайн: 10 дней с момента получения задания.

Формат сдачи:

- Репозиторий на GitHub или GitLab с кодом, README и примерами.
- Текстовое описание архитектуры решения (в README или отдельным файлом).
- Ссылку на репозиторий, презентационные, сопроводительные материалы и прочее отправьте на email

### Описание задания: Реализация простого ИИ чат-бота на базе RAG

Вам необходимо разработать простой ИИ чат-бот на базе RAG для сайта ООО «СФН» (<https://sfn-am.ru/>), который отвечает на вопросы инвесторов, используя базу знаний.

### Условия выполнения

- Использовать язык программирования Python.
- Для LLM использовать GigaChat API (первые 1 млн токенов предоставляются бесплатно после регистрации) или любую открытую модель (например, Hugging Face - facebook/bart-large).
- Для RAG использовать FAISS для хранения эмбедингов.
- База знаний: вам необходимо самостоятельно собрать данные, например, спарсить информацию с сайта <https://sfn-am.ru/> (разделы FAQ, информация о ЗПИФ, объекты недвижимости) или использовать публичные финансовые документы.
- Реализовать взаимодействие с чат-ботом через API (FastAPI) или, как альтернативный вариант, через Telegram-бота.
- Добавить базовую документацию (файл README) с инструкцией по запуску.
- Нарисовать верхнеуровневую архитектуру RAG-пайплайна (например, в draw.io / Miro) и предоставить рекомендации по её улучшению в будущем (в текстовом виде, в README или отдельным файлом).

### Требования к выполнению

- Подготовить базу знаний: собрать данные, разбить их на чанки, сгенерировать эмбединги и сохранить в FAISS.
- Реализовать RAG-пайплайн: принимать запрос пользователя, искать релевантные данные, генерировать ответ с помощью LLM.

- Добавить обработку ошибок (например, на случай отсутствия релевантных данных или сбоя API).
- Написать README с инструкцией по запуску: установка зависимостей, запуск чат-бота, примеры запросов.
- Нарисовать верхнеуровневую архитектуру RAG-пайплайна и дать рекомендации по улучшению.

#### Дополнительно (опционально)

- Добавить классификацию запросов (например, определение категории: "дивиденды", "продажа паёв").
- Реализовать базовый мониторинг (например, логирование запросов и ответов).

### Примеры запросов и ответов в ИИ чат-бот

#### Пример 1: Запрос о том, что такое инвестиционный пай

- **Запрос:** "Что такое инвестиционный пай (Пай)?"
- **Ответ:** Инвестиционный пай — это ценная бумага, подтверждающая долю пайщика в имуществе Фонда, право на доверительное управление, выплату компенсации при погашении и участие в собрании пайщиков. Учёт прав ведётся в реестре, данные подтверждаются выпиской. Паи обращаются свободно после формирования фонда, с ограничениями для квалифицированных инвесторов.

#### Пример 2: Запрос о тестировании перед покупкой паёв

- **Запрос:** "Перед покупкой паев нужно пройти тестирование. Что это значит?"
- **Ответ:** Для покупки паёв фондов, не предназначенных для квалифицированных инвесторов, требуется пройти тестирование на понимание продукта. Тест из семи вопросов доступен в СберБанк Онлайн перед покупкой. Пересдавать можно неограниченно.

#### Пример 3: Запрос о возврате средств из ЗПИФ

- **Запрос:** "Возможно ли вернуть средства из ЗПИФ до прекращения фонда?"
- **Ответ:** Паи розничных фондов ООО «СФН» можно реализовать на вторичном рынке: для квалифицированных инвесторов — через брокера, для неквалифицированных — через договор купли-продажи или брокера. На бирже — только если фонд листингован. Рыночная цена устанавливается участниками торгов, а не управляющей компанией.

### Критерии оценки

- **Функциональность:**
  - Работает ли RAG-пайплайн (поиск, генерация ответа).
  - Корректность взаимодействия (API или Telegram-бот).
- **Качество кода:**

- **Оптимизация и дополнительные функции:**
  - Эффективность RAG
  - Наличие дополнительных функций.
- **Архитектура и рекомендации:**
  - Наличие и качество архитектуры RAG-пайплайна.
  - Ценность рекомендаций по улучшению.
- **Объяснение решения**

## **Инструкции по сдаче**

1. Создайте репозиторий на GitHub или GitLab.
2. Поместите в репозиторий:
  - Код проекта.
  - Файл README с инструкцией по запуску (установка зависимостей, запуск чат-бота, примеры запросов).
  - Верхнеуровневую архитектуру RAG-пайплайна и рекомендации по улучшению (в README, отдельным файлом или в формате изображения).
  - Короткое видео (3–5 минут) или текстовое описание архитектуры (можно в README).
  - Ссылку на ИИ бота в телеграмме (если реализовали)
3. Отправьте ссылку на репозиторий и сопроводительные материалы на email