Вкладка 1

**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**(Университет ИТМО)**

Факультет **Прикладной информатики**

Направление подготовки **09.03.03**

Образовательная программа **Мобильные и сетевые технологии**

**К У Р С О В О Й   П Р О Е К Т**

Тема: «Помощник рекрутера»

Обучающийся: Дегтярь Глеб Сергеевич, K3141

Санкт-Петербург 2025

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_oottqre3zzsq)

[1 Суть проекта и процесс работы над ним 5](#_7jncxgijvyg)

[2 Проделанная работа для достижения цели и анализ 7](#_t79r0gvvzg6a)

[2.1 Проблема, поставленная передо мной 7](#_d2wt51g5hoze)

[2.2 Решение поставленной задачи 7](#_s7d626hnoid)

[2.3 Анализ проделанной работы 9](#_a8jlsa4czdvk)

[3 Анализ взаимодействия с командой и руководителем 11](#_ru3g5axtw9kk)

[3.1 Взаимодействие с командой 11](#_vnqykc6fgjy8)

[3.2 Взаимодействие с руководителем проекта 11](#_d00hijycbfm4)

[3.3 Оценка работы руководителя проекта 12](#_3apnw7v1rk2n)

[ЗАКЛЮЧЕНИЕ 13](#_1ak78j26zkio)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 15](#_pz70lzrwx1jf)

[ПРИЛОЖЕНИЕ 16](#_ou2sl9yffdde)

[Техническое задание 16](#_l6b91rtqs32b)

# ВВЕДЕНИЕ

«Помощник рекрутера» представляет собой веб-приложение для генерации вопросов с помощью LLM по описанию вакансии, на которые должны ответить кандидаты.

Актуальность данного проекта продиктована сложностью и громоздкостью многоступенчатых процессов найма персонала. Сегодня кандидату приходится проходить через целую серию интервью и этапов оценки, начиная с первичных собеседований с HR-специалистами, тестирований, групповых интервью и заканчивая финальными встречами с руководителями и техническими экспертами. Такой процесс чрезмерно затянут и отнимает значительное количество времени как у соискателей, так и у сотрудников компании. Это снижает эффективность поиска подходящих специалистов, замедляет закрытие вакансий и приводит к потере потенциальных талантов, которые могут уйти к конкурентам с более оперативным подходом к подбору кадров. Кроме того, многоступенчатый отбор может быть психологически утомительным и стрессовым для кандидатов и может оттолкнуть даже заинтересованных соискателей.

Разработанное нами решение предназначено для широкого спектра предприятий, начиная от небольших стартапов и компаний малого бизнеса, где важна скорость закрытия вакансий и гибкость в подборе персонала, и заканчивая крупными корпорациями с высокими требованиями к процессам найма и стандартизации. Оно может стать неотъемлемой частью кадровой стратегии организации, существенно упрощая и ускоряя поиск и отбор квалифицированных специалистов, которые соответствуют всем требованиям, изложенным в описании вакансии.

Цели проекта включали в себя реализацию за отведенное время (с 01.11.2024 по 17.12.2024) работающего прототипа приложения, связав работу каждого из коллег в единое целое. Функционал приложения должен включать экраны “генерация вопросов”, “получение истории сгенерированных вопросов”, “регистрация” и “вход”.

Чтобы достичь поставленной цели, моей команде было необходимо выполнить следующие задачи:

1) составить дизайн-макет экрана входа,

2) составить дизайн-макет экрана регистрации,

3) составить дизайн-макет экрана генерации вопросов,

4) составить дизайн-макет экрана истории запросов,

5) разработать API под страницу входа,

6) разработать API под страницу регистрации,

7) разработать API под страницу генерации вопросов,

8) разработать API под страницу истории запросов,

9) сверстать и подключить к API страницу входа,

10) сверстать и подключить к API страницу генерации вопросов,

11) сверстать и подключить к API страницу истории запросов,

12) провести промпт инжиниринг,

13) предоставить функцию, взаимодействующую с LLM для получения скрининговых вопросов,

14) провести развертывание приложения на стенде.

Для выполнения работы каждому из участников проекта была отведена определенная роль в нем. Так, как дизайнеру проекта, мне необходимо было создать эффективный и понятный дизайн страниц “входа”, “регистрации”, “генерации скрининговых вопросов” и “истории предыдущих генераций”, а также определить возможные направления дальнейшего развития проекта и подготовить дизайн страниц, которые могли бы быть добавлены в будущем, так например, была также разработана “приветственная” страница сервиса.

Для выполнения поставленных задач, мне необходимо было обучиться работе с figma, а также изучить правила создания дизайна для web-сервисов.

# 1 Суть проекта и процесс работы над ним

Веб-приложение «Помощник рекрутера» предназначено для HR-специалистов в компаниях любого размера: оно трансформирует процесс найма на работу, становясь важным инструментом в процессе поиска сотрудников. C использованием приложения рекрутеры смогут в удобном, специализированном интерфейсе самостоятельно генерировать скрининговые вопросы по любым технологиям из требований вакансии, не привлекая специалистов, проводящих техническое интервью, время которых, как правило, обходится работодателям дороже. Такое решение дает возможность отсеивать неподходящих кандидатов силами рекрутеров на начальных этапах, что оптимизирует процесс приема на работу.

Процесс работы над проектом можно разделить на четыре этапа: первый начался с выдачи заданий для ознакомления с предметной областью и стеком, необходимым для разработки. Затем каждый участник команды стал выполнять свои индивидуальные задачи, которые в будущем сформируют веб-приложение. Каждый из коллег завел репозиторий GitHub и регулярно синхронизировал его с добавлением новой функциональности.

Для контроля выполнения заданий еженедельно проводились созвоны, где все обновляли информацию о статусе своих задач, задавали интересующие вопросы и отправляли работу на оценку руководителю, чтобы либо оперативно перейти к следующей задаче, либо заняться доработкой текущей.

Сначала каждый выполнял свою работу отдельно, но немногим позднее была налажена связь между бэком, фронтом и деплоем. На этом этапе пришлось провести масштабную работу над ошибками, чтобы работа одного участника корректно открывалась на устройстве другого. Чуть позже получилось интегрировать работу промпт-инженера.

Финальным шагом стал созвон, посвященный защите проекта: там был определен порядок спикеров, доработана презентация и обсуждены вопросы, которые потенциально могут прозвучать на защите.

Работа шла планомерно с соблюдением сроков, установленных руководителем проекта. Во время интеграции результатов работы нескольких людей нужно было держать контакт с членами команды: они были рады подсказать, если в процессе появлялись ошибки. Такое взаимодействие было плодотворным и полностью поддерживалось тимлидом.

# 2 Проделанная работа для достижения цели и анализ

## 2.1 Проблема, поставленная передо мной

В начале работы над проектом передо мной были поставлены следующие задачи:

1. познакомиться с предметной областью,
2. создать репозиторий, провести локальную настройку,
3. реализовать POST-запрос,
4. реализовать GET-запрос,
5. реализовать DELETE-запрос,
6. подключить запросы к роутеру,
7. добавить функционал по получению истории запросов,
8. ознакомиться с библиотекой FastAPI Users
9. добавить возможность регистрации и авторизации,
10. подключение функции вызова к LLM

Суть проблемы заключается в реализации серверной части веб-приложения. Это работа с базой данных, куда записывается информация о запросах пользователей, написание API в виде POST, GET, DELETE запросов.

## 2.2 Решение поставленной задачи

Для решения поставленной задачи было необходимо углубить свои знания в языке Python, познакомиться с ООП и работой с данными. Поработать с Alembic, Swagger, описать базу данных через SQLAlchemy и написать API при помощи фреймворка FastAPI.

При выполнении работы мне помогали материалы и советы руководителя, официальные инструкции к используемым инструментам, различные форумы.

Первым шагом в моей работе стало создание пустого проекта по шаблону, который рекомендовал тимлид. Он представляет собой набор директорий, который, при правильной настройке, превращается в понятную структуру.

Затем мне пришлось познакомиться с SQLAlchemy (рис. 1), где с помощью принципов объектно-ориентированного программирования можно взаимодействовать с базой данных PostgreSQL с помощью языка Python.



Рисунок 1 - Логотип программной библиотеки SQLAlchemy

При работе с базой данных мне удалось применить знания и навыки проектирования БД, полученные на соответствующем предмете. Я определил атрибуты каждой записи и их формат, создав основу базы данных проекта.

Следующим шагом было написание API в виде трех запросов. На этом этапе моим незаменимым инструментом стал официальный туториал FastAPI (рис. 2), где с понятными ремарками приводятся примеры нужного синтаксиса. Здесь были полезны и советы руководителя: Егору удалось на ранней стадии заметить мою ошибку, которая ставила под угрозу работоспособность всего проекта. Со своей стороны, я сделал необходимые выводы и не допустил повторения проблемы.



Рисунок 2 - Логотип веб-фреймворка FastAPI

Важным этапом стала работа со Swagger (рис. 3). Это позволило мне сгенерировать машино- и человекочитаемую документацию в формате JSON, которая был полезна для фронтенд-части проекта при создании экранов генерации и истории запросов. Документация показывает, какие данные (и в каком формате) отправляются и принимаются в обработку.



Рисунок 3 - Логотип инструмента Swagger для создания документации

## 2.3 Анализ проделанной работы

Работа была крайне непростой, но я горд полученным результатом проекта. Поскольку это было мое первое участие в разработке, у меня не было достаточного опыта в программировании и, более того, мне не хватало кругозора в области веб-программирования, поэтому изучать все в сжатые сроки было довольно тяжело, тем не менее, у меня получилось выполнить основные задачи.

Основные трудности были связаны с изучением стека новых технологий и ранее неизвестных возможностей языка Python. Например, с большим трудом мне давалось объектно-ориентированное программирование, которое использовалось при создании моделей SQLAlchemy, также непросто было понять, как реализовать подключение запросов к роутеру.

Единственная задача, которая так и осталась невыполненной, это реализация регистрации и авторизации при помощи новой библиотеки FastAPI Users. Этот экран нужно было сделать совместно с членом команды, занимающимся фронтендом, но мы единогласно решили оставить эту задачу из-за недостатка времени. Поскольку эта часть работы изначально была описана в качестве дополнительного задания, проект ничего не потерял в функциональности из-за ее отсутствия.

Нашей команде удавалось работать планомерно. Единственной причиной, которая, возможно, помешала исполнению всех заданий в срок стала болезнь человека на роли дизайнера.

Подводя итоги, можно с уверенностью сказать, что мне удалось:

1) улучшить свои навыки в языке Python: все, с чем я столкнулся за полтора месяца работы над курсовым проектом сейчас помогает мне академически,

2) освоить основные концепции бэкенд-разработки: теперь я понимаю устройство веб-приложений изнутри,

3) познакомиться с различными фреймворками, библиотеками и инструментами,

4) улучшить свои навыки работы с Git,

5) лучше ориентироваться в GitHub,

6) правильно вести репозиторий: сейчас я знаю, какая информация в программном коде может считаться приватной и почему из-за этого стоит делать репозиторий закрытым.

# 3 Анализ взаимодействия с командой и руководителем

## 3.1 Взаимодействие с командой

Общение с командой осуществлялось в чате, созданном руководителем. Там мы договаривались о созвонах, обсуждали технические моменты разработки веб-приложения. Наиболее тесное общение с членами команды началось тогда, когда пришло время интегрировать работу каждого из участников в общий продукт. Во время этого, мы обращались и получали ответы от каждого, в чьей области ответственности происходила проблема. Мы как члены команды показали настоящую сплоченность, продемонстрировали свое критическое мышление и готовность помогать друг другу.

## 3.2 Взаимодействие с руководителем проекта

Коммуникация с руководителем проекта осуществлялась через мессенджер или Яндекс.Телемост. К Егору можно было обратиться по любому вопросу и оперативно получить исчерпывающий ответ. Если же кто-то сталкивался с особенными трудностями, всегда можно было попросить о созвоне в индивидуальном порядке, где находилось решение для каждой проблемы.

При выборе проекта, я обращал внимание на специализацию магистра-руководителя и мне повезло, что я оказался на позиции бэкенд-разработчика где был под чутким контролем Егора, который является настоящим специалистом в этой сфере. Я часто обращался к руководителю за советом и помощью, всегда получая развернутый комментарий.

Не могу не отметить вклад Егора в процесс защиты проекта: он, имея опыт хакатонов, составил порядок выступления и поделился своим видением того, как должна проходить защита. Благодаря тимлиду нам удалось отчитаться о завершении проекта без стресса и паники, получив высокую оценку, чему я очень доволен.

## 3.3 Оценка работы руководителя проекта

Я высоко оцениваю работу Егора. Он ответственный и пунктуальный, всегда готов прийти на помощь и подсказать. У меня остались только положительные эмоции от нашего взаимодействия: я получил ценные знания, основанные на реальном опыте, которые, уверен, пригодятся мне и в будущем.

# ЗАКЛЮЧЕНИЕ

Базовые требования к проекту были реализованы, за исключением, возможно, верстки дизайна из-за причин, описанных выше. Единственная задача, так и оставшаяся на бумаге - это реализация экранов регистрации и авторизации. К сожалению, выполнить ее не удалось из-за нехватки времени и отсутствия необходимого опыта. Однако отсутствие этого экрана отнюдь не критичное: задача входила в список дополнительных требований и не является основной, поэтому можно с уверенностью сказать, что проект выполнен успешно.

В результате получился рабочий продукт, размещенный в сети и готовый к использованию по назначению, что является отличным завершением работы. Оглядываясь назад, я доволен пройденным путем: мне удалось научиться новым вещам и поработать в команде, получая практический опыт разработки.

Область моей ответственности в этом проекте - бэкенд - играет ключевую роль в архитектуре системы и является центральной, системообразующей частью сервиса. Именно он обеспечивает связь между всеми компонентами веб-приложения, над которыми трудились мои коллеги, превращая их усилия и наработки в цельный, полнофункциональный продукт. Благодаря нашей слаженной работе, весь функционал приложения стал доступен пользователю и работает корректно, что особенно важно для реализации поставленных целей.

Однако невозможно не отметить выдающийся вклад каждого члена команды. Работа над проектом была бы невозможна без упорства, креативности и профессионализма моих товарищей. Мы не просто выполнили поставленные задачи — мы показали себя как по-настоящему сплоченный коллектив, заслуживший максимальную оценку за командную работу.

Для каждого из нас этот проект стал не просто учебным заданием, а значимым этапом учебы. Мы не только углубили свои знания, но и открыли для себя новые горизонты в программировании, взаимодействии с коллегами и управлении сложными процессами разработки. Во время работы мы значительно выросли как специалисты, расширив свои навыки и компетенции. Трудно представить выполнение этого проекта без кого-либо из членов команды. Каждый из нас внес важный вклад в его реализацию, и только благодаря совместным усилиям мы сумели превратить первоначальную идею в работающий, инновационный продукт. Наше веб-приложение готово к применению в процессе реального найма и не имеет аналогов в сети, что делает его действительно интересным решением.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Неустроев А.В. Сохранить JSON данные в базу данных sqlalchemy// StudNet. — 2016. — №9. — С. 1-2. — URL: <https://cyberleninka.ru/article/n/sohranit-json-dannye-v-bazu-dannyh-sqlalchemy> (дата обращения 02.11.2024)
2. Баланов А. Бэкенд-разработка веб-приложений. Архитектура, проектирование и управление проектами. Учебное пособие — СПб.: Лань, 2024. — 544 с.: ил. — (Серия «Высшее образование»). — ISBN 978-5-4461-1796-3 (дата обращения 10.11.2024)
3. FastAPI. — URL:<https://fastapi.tiangolo.com/tutorial/> (дата обращения 10.11.2024)
4. SQLAlchemy. — URL: <https://docs.sqlalchemy.org/en/20/tutorial/index.html> (дата обращения 05.11.2024)
5. Python Documentation. — URL: <https://docs.python.org/3/tutorial/index.html> (дата обращения: 01.11 - 01.12.2024)

# ПРИЛОЖЕНИЕ

## Техническое задание

**1. Название проекта**

Помощник рекрутёра.

**2. Цель**

Целью проекта является разработка веб-приложения “Помощник рекрутера”, основной функционал которого состоит в генерации скрининговых вопросов собеседования по описанию вакансии. Веб-приложение должно включать дизайн, серверную и клиентскую часть и должно быть размещено в публичном доступе в интернете.

**3. Сроки выполнения**

Начало 01 ноября 2024 г.

Окончание 20 декабря 2024 г.

**4. Руководитель проекта**

Кондрашов Егор Юрьевич.

**5. Термины и сокращения**

Рекрутинг – процесс поиска, привлечения и найма сотрудников.

Скрининговые вопросы – вопросы, которые задаются на скрининговом интервью.

Скрининговое интервью – ранний этап процесса найма, который позволяет рекрутерам оценить большой круг кандидатов и определить, соответствуют ли они минимальным требованиям, предъявляемым к должности.

**6. Технические требования**

1. Возможность по переданному описанию вакансии получить 5 вопросов для скринингового интервью, подходящих к требованиям из вакансии.
2. Возможность просмотреть историю своих предыдущих сгенерированных вопросов.
3. Возможность зарегистрировать по паре логин и пароль (опционально).
4. Возможность войти в существующий аккаунт по логину и паролю (опционально).
5. Веб-сервис доступен в сети интернет.

**7. Содержание работы**

| № | Этапы проекта | Сроки выполнения этапов | Ответственный за этап | Вид представления результатов этапа |
| --- | --- | --- | --- | --- |
| 1 | Разработка технического задания | 1-10 ноября | Кондрашов Егор Юрьевич | Документ, содержащий техническое задание. |
| 2 | Анализ предметной области и проектирование | 11 ноября - 17 ноября | Дегтярь Глеб Сергеевич | Заведённые задачи на реализацию проекта; созданные репозитории. |
| 3 | Разработка | 18 ноября - 11 декабря | Миронов Иван Андреевич | Программный код бэкенда, фронтенда, Prompt Engineering и DevOps частей; проект в Figma с дизайн макетом. |
| 4 | Тестирование | 12 - 19 декабря | Матюков Дмитрий Владимирович | Отчёт о тестировании. |
| 5 | Защита проекта (сдача отчета и представление доклада с презентацией) | 20 декабря | Кармазин Арсений Захарович | Отчёт о проделанной работе, презентация. |

**8. Задачи проекта**

Задачи сгруппированы по предметной области и выполняющему их исполнителю.

Бэкенд, исполнитель Кармазин Арсений Захарович:

– провести подготовительные работы,

– реализовать получение скрининговых вопросов по описанию вакансии,

– реализовать получение истории запросов пользователя,

– добавить регистрацию и авторизацию пользователей на серверной части.

Фронтенд, исполнитель Миронов Иван Андреевич:

– провести подготовительные работы,

– реализовать экран "Получение скрининговых вопросов по описанию вакансии",

– реализовать экран "Получение истории запросов пользователя",

– реализовать экран "Регистрация по логину и паролю",

– реализовать экран “Вход”.

Промпт-инжиниринг, исполнитель Матюков Дмитрий Владимирович:

– провести подготовительные работы,

– выполнить промпт-инжиниринг,

– определить оптимальный вид входных данных,

– реализовать функцию для получения скрининговых вопросов.

Дизайн, исполнитель Дегтярь Глеб Сергеевич:

– провести подготовительные работы,

– выполнить дизайн экрана "Получение скрининговых вопросов по описанию вакансии",

– выполнить дизайн экрана "Получение истории запросов пользователя",

– выполнить дизайн экранов "Регистрация" и "Вход".

Деплой, исполнитель Мухсин Афаккир:

– провести подготовительные работы,

– склонировать репозитории на стенд,

– предоставить доступ к бэкенду и фронтенду через веб-сервер,

– добавить SSL.

Другие задачи:

– исполнитель Кондрашов Егор Юрьевич: составить техническое задание,

– исполнитель Матюков Дмитрий Владимирович: провести тестирование и полишинг,

– исполнитель Кармазин Арсений Захарович: подготовить доклад и презентацию.

**9. Основные результаты работы и формы их представления**

Основными результатами проекта являются:

– доступный в сети Интернет и выполняющий приведённые в пункте 6 технического задания требования веб-сервис,

– программный код бэкенда,

– программный код фронтенда,

– программный код Prompt Engineering части и документы с описанием проведённых экспериментов,

– программный код DevOps части,

– проект в Figma, содержащий дизайн-макет сервиса.