

# HR Zero - проект для хакатона Лидеры Цифровой Трансформации 2024

## Что успели сделать за 300 часов!

- Бэкэнд и фронт сервиса для проведения собеседований
- 3 микросервиса:
  - для авторизации пользователей по ролям
  - основной бизнес-логикой проведения собеседований
  - чаты для общения с соискателями
- Личный кабинет соискателя
- Скрейпер для сбора вакансий и данных о кандидатах
- AI для анализа соответствия резюме вакансиям

## Ссылки

- Сайт: <https://hrzero.ru/>
- Репозиторий: <https://github.com/lct-2024/lct-2024>

## Учётные записи для тестирования

**Внимание!** Для удобства проверки демо-стенда, проверка паролей была отключена и для логина можно вводить любой пароль.

- HR: hr@example.com
- Нанимающий менеджер: manager@example.com
- Кандидат 1: sasha@example.com
- Кандидат 2: mary@example.com

## Стек технологий

- Бэкенд на Common Lisp.
- Межсервисное взаимодействие: OpenRPC
- На фронтенде: React + Redux
- База данных - PostgreSQL в облаке.
- Развёртывание всех компонент в Docker.

## Архитектура проекта

Для бэкенда мы реализовали микросервисную архитектуру из 3 сервисов:

- passport - хранит информацию о пользователях и их правах.
- chat - позволяет общаться соискателям и сотрудникам компании.
- ats - хранит все данные и логику работы с соискателями, воронки найма и тд.

Бэкенды общаются между собой и фронтендом с использованием JSON-RPC.

API документировано в формате OpenRPC (аналог OpenAPI и Swagger) и может быть просмотрено в интерфейсе <https://playground.open-rpc.org/>. Для этого достаточно ввести на сайте один из следующих эндпоинтов:

- <https://passport.hrzero.ru/>
- <https://chat.hrzero.ru/>
- <https://ats.hrzero.ru/>

Чтобы в Playground выполнять запросы от определённого пользователя, нужно в левой нижней части экрана задать авторизационный токен таким образом:

```
{
  "headers": {
    "Authorization": "..."
  }
}
```

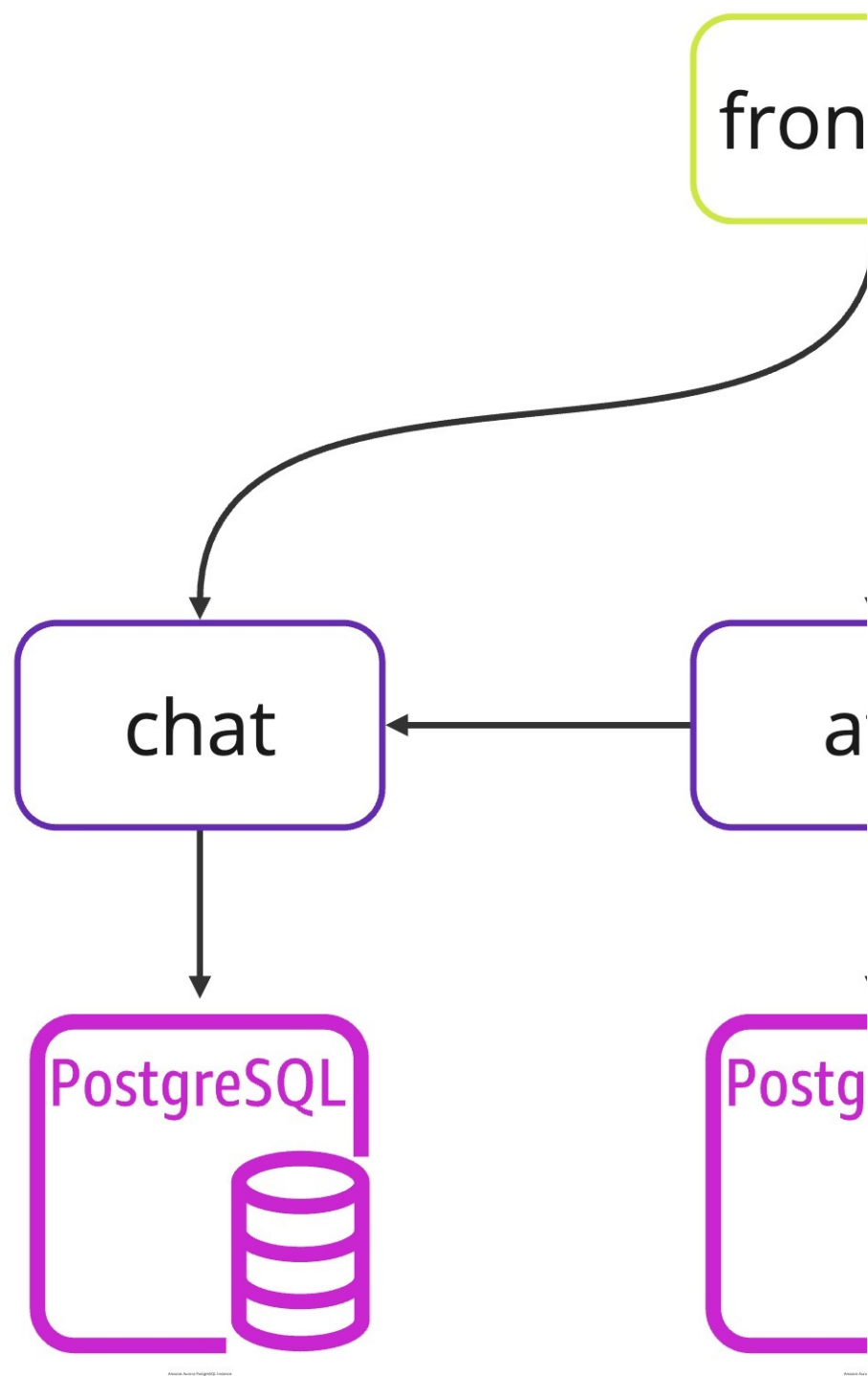
Вот токены пользователей с разными ролями:

- [illegible]

Замените многоточие из примера на один из токенов.

Документация по API автоматически собирается из docstring в коде и складывается в файлы `openrpc-spec.json` внутри папки каждого из микросервисов. Затем по этим файлам автоматически генерируется клиент для того, чтобы можно было удобно обращаться из одного микросервиса в другой.

## Вот так выглядит архитектура сервиса



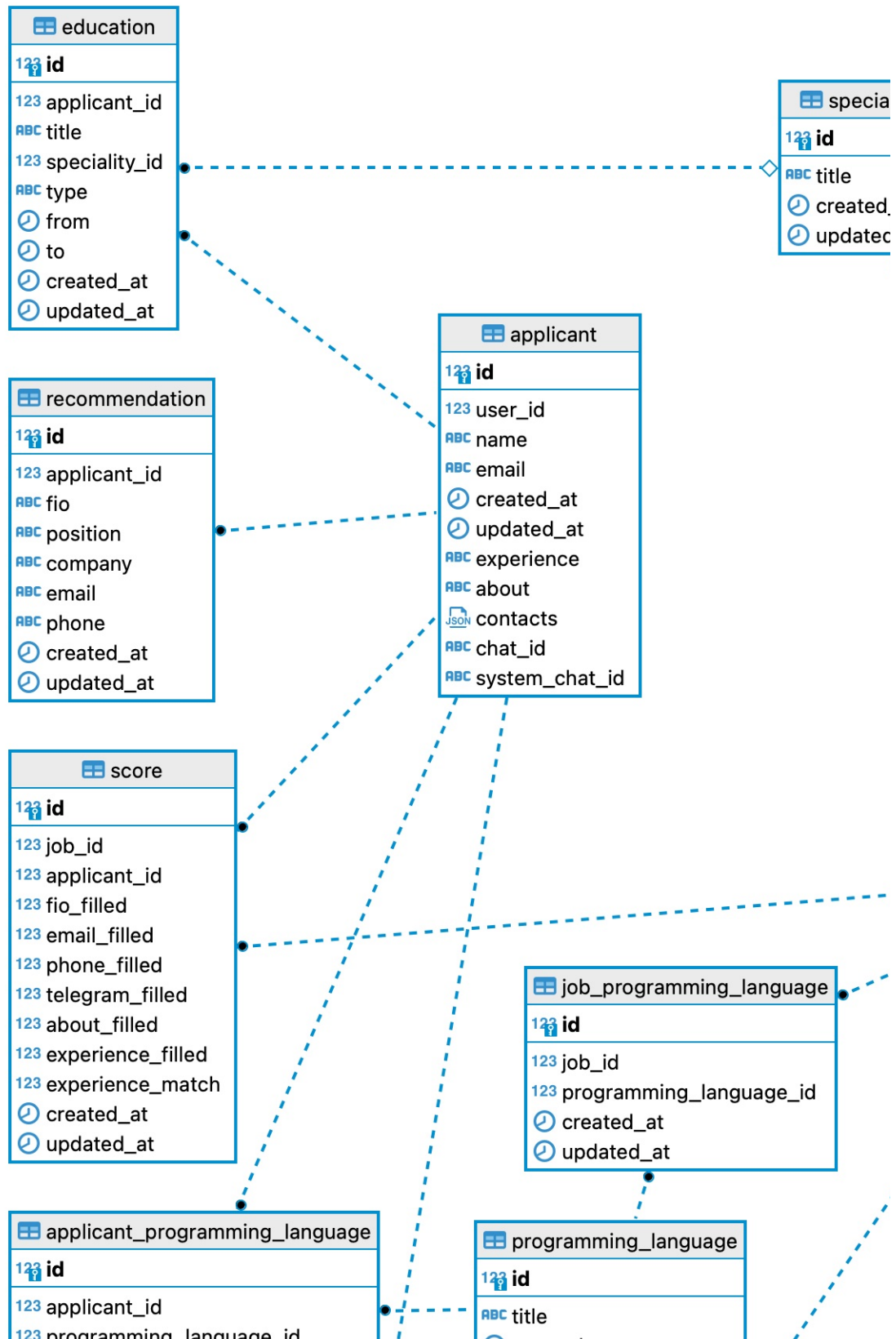
#### Директории

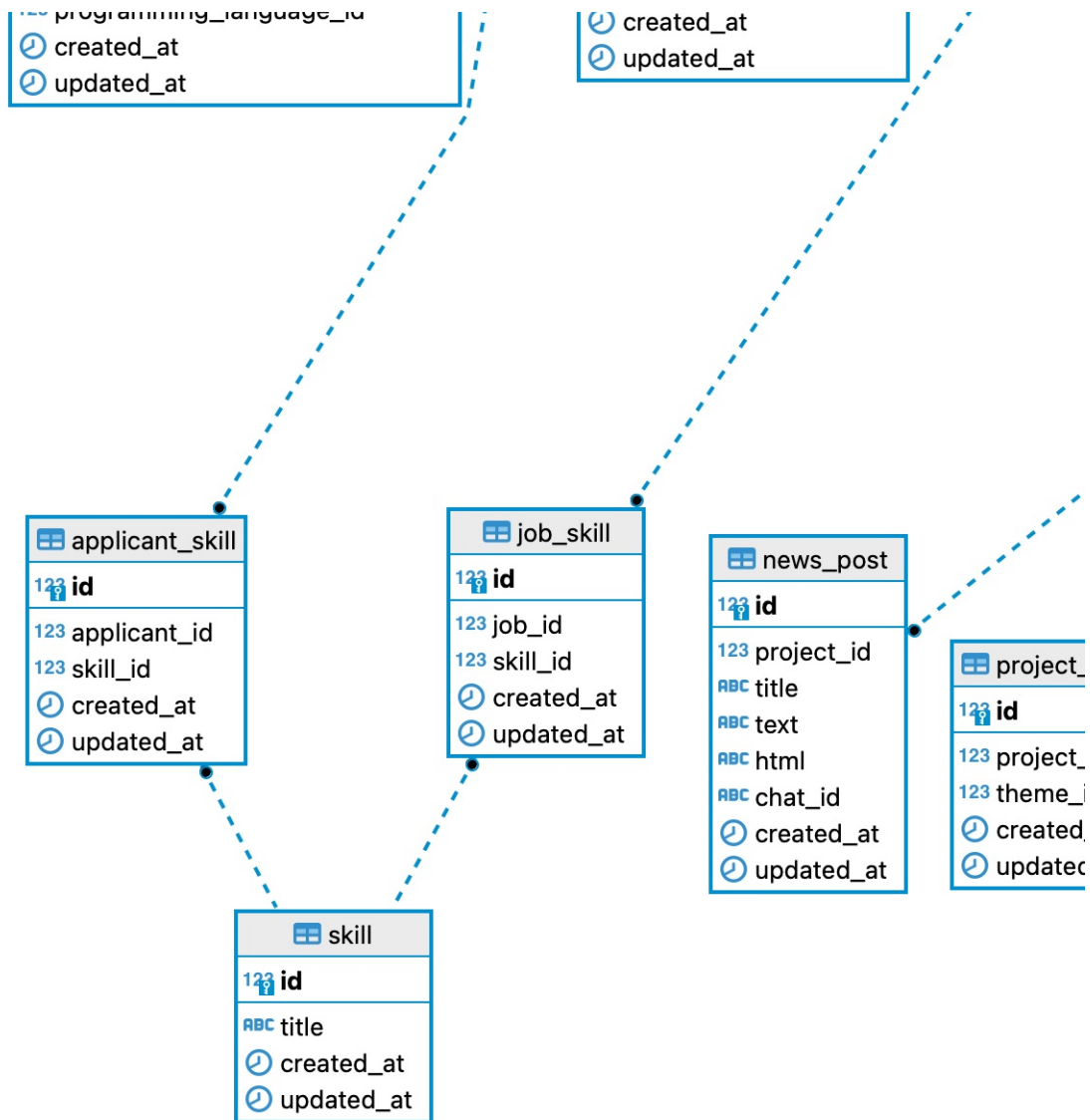
- /frontend/ - код фронтенда.
- /backend/ - код бэкенда.

## Applicant Tracking System

Основная бизнес-логика тут. Этот микросервис хранит данные о вакансиях, кандидатах, этапах собеседований, которые прошёл каждый из кандидатов и прочее.

Вот структура базы данных этого микросервиса (описание схемы backend/ats/schema.sql):





Так же, этот микросервис содержит дополнительные алгоритмы для парсинга резюме из файлов загруженных пользователями, и для матчинга резюме и вакансий.

А ещё, здесь же в папке `backend/ats/src/spiders/` есть код скрейперов, с помощью которых производился сбор дополнительных данных о вакансиях и кандидатах.

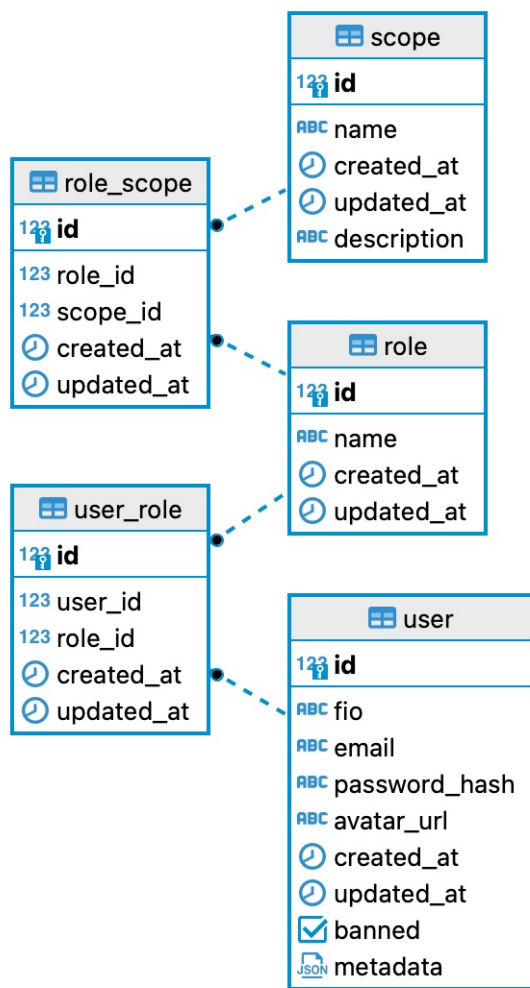
### Passport (Информация о пользователях, ролях и правах)

Этот микросервис хранить данные о пользователях. Email используется как идентификатор. В базе хранятся хэши от паролей. Для аутентификации запросов между фронтендом и микросервисами мы используем JWT токены, в которых зашифрованы основные данные о пользователе и его правах.

С каждым пользователем системы может быть ассоциирована одна или несколько ролей и каждая из ролей может быть ассоциирована с несколькими правами (scopes). Эта система позволяет гибко распределять полномочия между сотрудниками компании.

Мы используем две роли, дающие расширенные полномочия: `hr` и `manager`

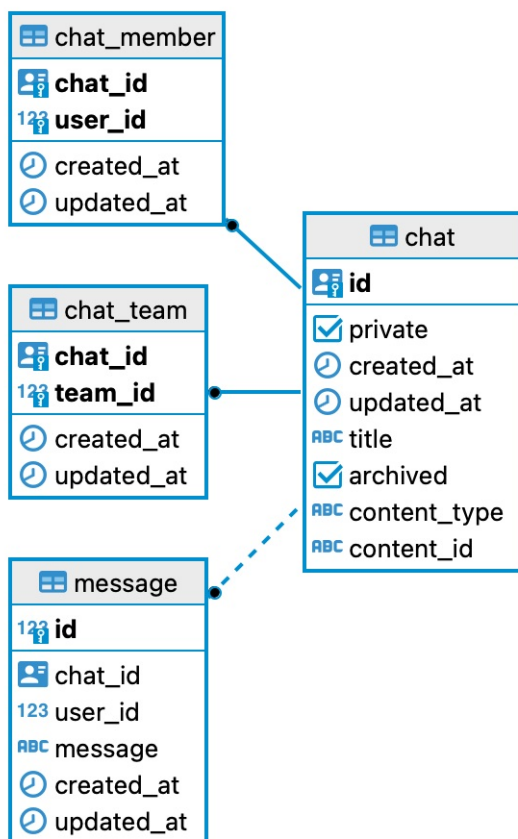
Вот так выглядит структура базы этого микросервиса:



## Chat

Чаты для общения с кандидатами и между сотрудниками компании. Так же мы используем отдельные чаты для доставки до пользователя таких системных сообщений, как уведомления о переходе на новый этап собеседований, предложений согласовать время собеседования и тд.. В дальнейшем, для уведомлений можно будет подключать новые каналы оповещения, такие как email, telegram, Whatsup.

Структура базы чатов:



Технически, чаты устроены так, что доступ к ним может быть ограничен для определённой группы сотрудников компании.

## Принципы обработки вакансий

### Парсинг вакансий из резюме

Процесс обработки загруженных файлов очень простой:

- загруженный файл преобразуется из формата Word или PDF в обычный текст с помощью программы pandoc
- затем мы используем специальный промпт к YandexGPT, чтобы по этому тексту сформировать JSON с полями в которые извлечены контактные данные, биография, навыки и данные об опыте и образовании соискателя.

Промпт к нейросети можно посмотреть в файле `backend/ats/src/ai/prompts/cv.txt`

### Подписка на новые вакансии

Для подписки мы сохраняем в базе фильтр, заданный пользователем: категорию, город, специальность, проект.

Далее, при появлении новой вакансии, проверяем, соответствует ли она каким-либо подпискам и отправляем соответствующим соискателям уведомления. Сейчас этот процесс запускается по событию прямо в backend микросервиса ats, но в будущем, если потребуется обрабатывать тысячи соискателей и сотни новых вакансий в день, стоит перевести систему на очередь с отдельными обработчиками.

## Матрица трассировки требований

Для тестирования мы описали 10 сценариев и составили такую матрицу трассировки требований:



100% ▼

p.

%

.0

.C

1

1

По ум...



---

B34:B38



```
fx реалізовано наполовину (back ats.create_news_post)
```

	A	B	C
1	Требование/Кейс	Примечание	С Кейс 1
11	Отправка комментария в чат	реализовано	
12	Колонка "Соответствие" в разделе "Вакансии"	реализовано	
13	Страница с рекомендациями по соответствию резюме вакансии	не реализовано	
14	Кнопка "Соответствие" в разделе "Вакансии"	не реализовано	
15	Последние сообщения из всех чатов с кандидатами в ЛК рекрутера	не реализовано	
16	Вход в ЛК по email	реализовано	
17	Алгоритм соответствия резюме вакансиям	реализовано	
18	Кнопка "Создать" в разделе "Вакансии", видная только HR	не реализовано	
19	Страница создания Вакансии	не реализовано	
20	Кнопка "Удалить" у каждой вакансии разделе "Вакансии", видная только HR	не реализовано	
21	Кнопка "Удалить" в профиле вакансии видная только HR	не реализовано	
22	Удаление вакансии на бэкенде	реализовано	
23	Кнопка "Открыть" на закрытых вакансиях в разделе "Вакансии"	не реализовано	
24	Кнопка "Открыть" в профиле вакансии видная только HR	не реализовано	
25	Открытие вакансии на бэкенде	реализовано	
26	Кнопка "Закрыть" на закрытых вакансиях в разделе "Вакансии"	не реализовано	
27	Кнопка "Закрыть" в профиле вакансии видная только HR	не реализовано	
28	Закрытие вакансии на бэкенде	реализовано	
29	Отображение откликнувшихся кандидатов в профиле вакансии	не реализовано	
30	Кнопка "Пригласить" на Этап 1, переводит кандидата на этап собеседования	не реализовано	
31	Кнопка "Собеседование" и попап с календарем	не реализовано	
32	Чат с системными уведомлениями в ЛК Соискателя	не реализовано	
33	Раздел "Новости"	реализовано	
34	Кнопка "Создать" в разделе "Новости"	реализовано	
35	Страница новости	не реализовано	
36	Кнопка "Удалить" на странице новости	не реализовано	
37	Кнопка "Редактировать" на странице новости	не реализовано	
38	Форма редактирования новости	не реализовано	
39			

[Ссылка на Excel документ](#) в Google Docs.

## Как собрать PDF с документацией

```
sudo apt-get install pandoc wkhtmltopdf
```

```
pandoc README.md \
  -t html \
  --pdf-engine-opt=--enable-local-file-access \
  --title 'ЛЦТ24 "Мы из Будущего"' \
  -o README.pdf
```