# Введение

* Цель проекта:
  + Создать телеграм бота, который помогает руководителю проектов структурировать информацию по его проектам.
* Описание функциональности бота (после общения с заказчиком)
  + В бота загружается протокол совещаний в “.docx“ формате.
  + После загрузки отчёта руководителем ему на выбор доступны три варианта отчёта - по проектам, по ответственным, по дате.
  + После выбора варианта, бот анализирует протокол и формирует в ответном сообщении информацию о каждом проекте с задачей, сроками и исполнителями.
  + Отчёт руководителя содержит данные по всем проектам. В блоке с задачей есть поля статус выполнения (да/нет), комментарий.
  + Отчёт ответственному отправляется после загрузки файла руководителем. Он содержит только его задачи с информацией по каждой задачи – какой это проект, подпроект, срок, статус, комментарий, руководитель.

# Архитектура бота

* Общая схема работы

Модуль для взаимодействия с пользователями и получения данных: Обработка команд пользователя при помощи библиотеки для работы с Telegram API Telebot.

↓

Модуль для обработки .docx файлов: Извлечение данных из протоколов совещаний в формате .docx, после ключевого слова “РЕШЕНИЕ:”, добавление ключей “Проект:”, “Подпроект:”, “Задача:”, “Статус:”, ”Комментарий:”

↓

Модуль для отправки данных на сервер.

↓

База данных PostgreSQL: Хранения информации о задачах, пользователях, комментариях и статусах выполнения задач.

↓

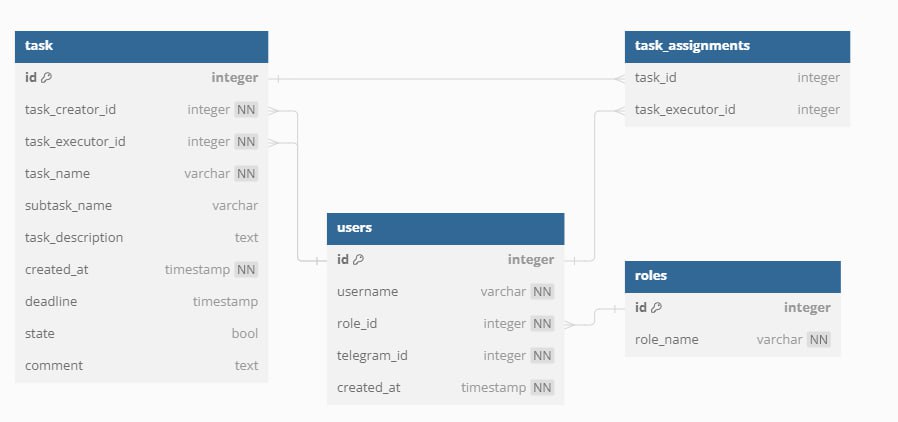
Модуль для получения данных с сервера.

↓

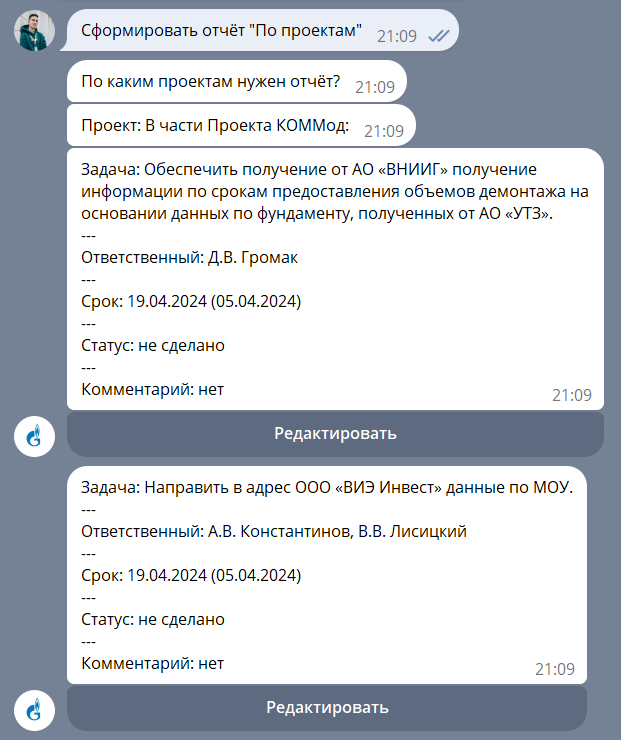
Модуль для создания майндкарт: Каждая задача формируется отдельным сообщением с разделителями “---” между строками для лучшей читаемости.

↓

Модуль для отправки сообщений пользователям: Отправка полученной майнд-карты при помощи библиотеки для работы с Telegram API Telebot.

* Используемые технологии
  + Python - язык программирования, на котором написан бот.
  + Telebot - библиотека для работы с Telegram API.
  + psycopg2 - библиотека для работы с PostgreSQL.
  + re - модуль для работы с регулярными выражениями.
* База данных
  + 

# Функциональные возможности

* **Отчетность для руководителя**
  + Загрузка протокола совещания в формате .docx
  + Варианты отчета для руководителя:
    - По проектам
    - По ответственным
    - По дате
  + Отчет по проекту:
    - Статус выполнения задачи (да/нет)
    - Возможность оставления комментария к задаче
    - Добавление новой задачи
  + Пример отчета по проекту
    - 
* **Отчетность для ответственных**
* Отправка таблицы с задачами, сроками и проектом в личные сообщения от бота

# Демонстрация работы бота

* Пример работы бота на загруженном протоколе совещания (вставить видео)
* (то которое было до этого)
* + запишу видео по коду

# Заключение

* Полученные результаты
  + Локальная обработка
    - Бот считывает docx
    - Бот обрабатывает docx
    - Бот выводит отчёт в telegram чат
  + Сервер
    - Бот считывает docx
    - Бот обрабатывает docx
    - Бот отправляет на сервер
* Планы по развитию проекта
  + Сервер:
    - Разделить базу данных на несколько
    - Написать запросы к БД чтобы выводить отчёт в телеграм чат с данными из БД (для отчётов по проектам, пользователям, дате)
    - Сделать работающими другие кнопки
    - Сортировка данных внутри таблиц по датам
    - Исправить ошибки в работе программы

# Команда

Разработчики:

1. Гордеев Никита Владиславович (tg: @nikitagordeev10) — Middle Python разработчик, Тимлид
2. Алимов Рашид Насирович (tg: @olivka\_djan) — Middle Python разработчик
3. Джалкубаев Карим Джамбулович (tg: @tudjsjs) — Junior Python разработчик

Менеджеры:

1. Корчевнюк Мария Юрьевна — ИТ-консалтинг
2. Орищенко Вероника Сергеевна (tg: @veronika\_orishenko) — 1С специалист
3. Живаева Алина Владимировна (tg: @alinazhivaeva) — 1С специалист

# Материалы

* https://github.com/nikitagordeev10/assistant\_rp\_gazprom/

# Вопросы и ответы

* Будем рады ответить на ваши вопросы

Начало формы