

УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Облачные и туманные вычисления»

Проект

Разработка Telegram бота "InterviewBot"

Студент

Кузнецов М. А.

P34131

Преподаватель

Перл О. В.

Санкт-Петербург, 2023 г.

Содержание

<i>Сведения о приложении</i>	2
<i>Реализация решения</i>	3

Сведения о приложении

Телеграмм-бот InterviewBot, который в себе будет содержать вопросы по разным топикам и темам, относящихся к IT в целом. Например, градация вопросов по языкам, алгоритмическим задачам и т. п. Возможны также и более общие темы -- устройство Linux, System Design и т. д. Еще, как возможность, пользователи сами смогут расширять существующую базу вопросов, то есть, через интерфейс бота добавлять свои вопросы. Другими словами – все это является некой совместной расширяемой базой знаний.

Реализация решения

На предыдущем этапе был определен следующий стек разработки:

- [Go](#) – в качестве главного языка программирования бота.
- [Telegram Bot API](#) – библиотека для удобного взаимодействия с API Telegram.
- [PostgreSQL](#) – open-source база данных, которая предоставит весь необходимый функционал для хранения данных.
- [pgx](#) – библиотека для Go для работы с базой данных PostgreSQL.

А также следующие роли:

1. **Гость** – это любой человек, который может найти бота через поиск, перейти по ссылке, взаимодействовать с ним. Данная роль разрешает использование основных и базовых команд бота. Назначается по умолчанию каждому новому пользователю.
2. **Администратор** – для получения данной роли требуется admin-key. Данный ключ генерируется автоматически при каждом запуске экземпляра бота на сервере. С помощью команды /adminkey можно авторизоваться в боте и получить дополнительный доступ к другим командам, которые позволяют управлять вопросами и темами, просматривать предложения.

К приложению была также добавлена роль super-admin, которую перенимает на себя разработчик бота. Данная роль конфигурируется только через .yaml файл конфигурации путем передачи ID пользователя Telegram, которому будет назначена данная роль. Пользователь с данной ролью в дополнении ко всем прочим командам может:

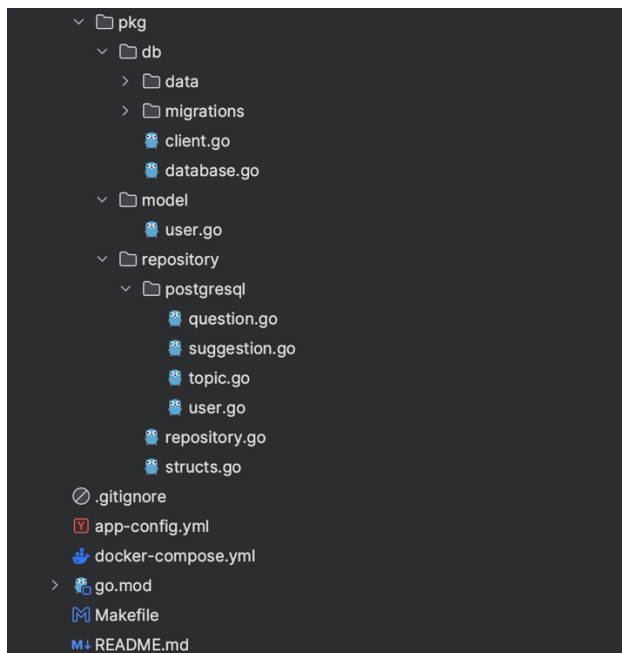
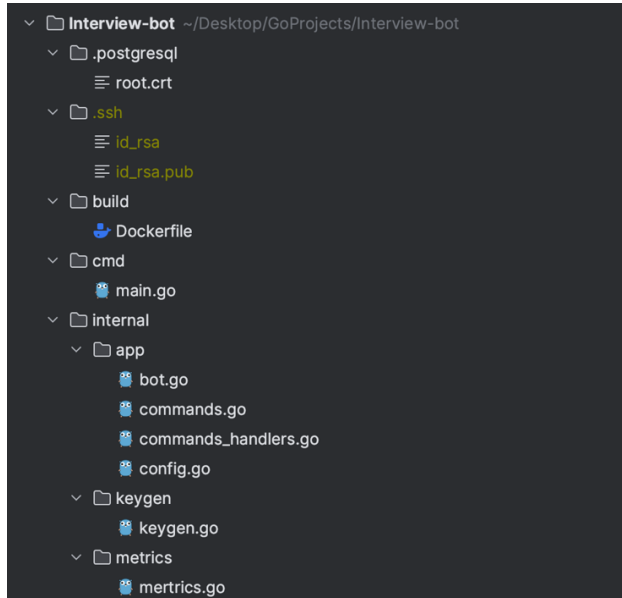
1. /block – запретить глобальный доступ к боту
2. /allow – разрешить глобальный доступ к боту
3. /blocka – запретить авторизацию для новых администраторов
4. /allowa – разрешить авторизацию для новых администраторов

Также не возникла необходимость в создании хранилища для фотографий для вопросов. Оказалось, что Telegram позволяет хранить изображения на своих серверах, доступ к которым можно получить по их уникальному ID. Таким образом, в базе данных храниться ID изображения, которое при показывании вопроса просто извлекается напрямую с серверов Telegram, а не через хранилище на сервере.

В остальном функционал бота остался прежним и удовлетворяет заданным на 1-м этапе требованиям.

Детали реализации

Структура проекта:



.postgresql – хранит сертификат на подключение к БД Yandex Cloud.

.ssh – необходимые ssh-ключи для подключения к Compute Yandex Cloud.

build/Dockerfile – *(не используется)* Dockerfile для сборки проекта.

Используется при локальной разработке, не используется при разворачивании на сервере.

cmd – содержит запускающий бота файл. Точка входа в программу.

Internal – содержит внутренний код работы и бизнес-логику бота.

app – основные настройки и запуск бота на основе [Telegram Bot API](#).

keygen – генератор случайного админ-ключа. Может быть перезаписан на свой через файл конфигурации.

metrics – работа с метриками для Prometheus.

pkg – код, относящийся к работе с хранилищем и БД:

db – содержит конфигурацию по настройке и подключению к БД Postgres.

model – модели для работы с БД.

repository – реализация паттерна Репозиторий для работы с БД.

app-config.yml – файл конфигурации приложения. Он имеет следующий вид:

```
bot:
  vm-address: "ADDRESS"
  bot-token: "TELEGRAM_BOT_TOKEN"
  superadmin-id: TELEGRAM_PERSON_ID
  force-admin-key: true | false
  admin-key: CUSTOM_ADMIN_KEY

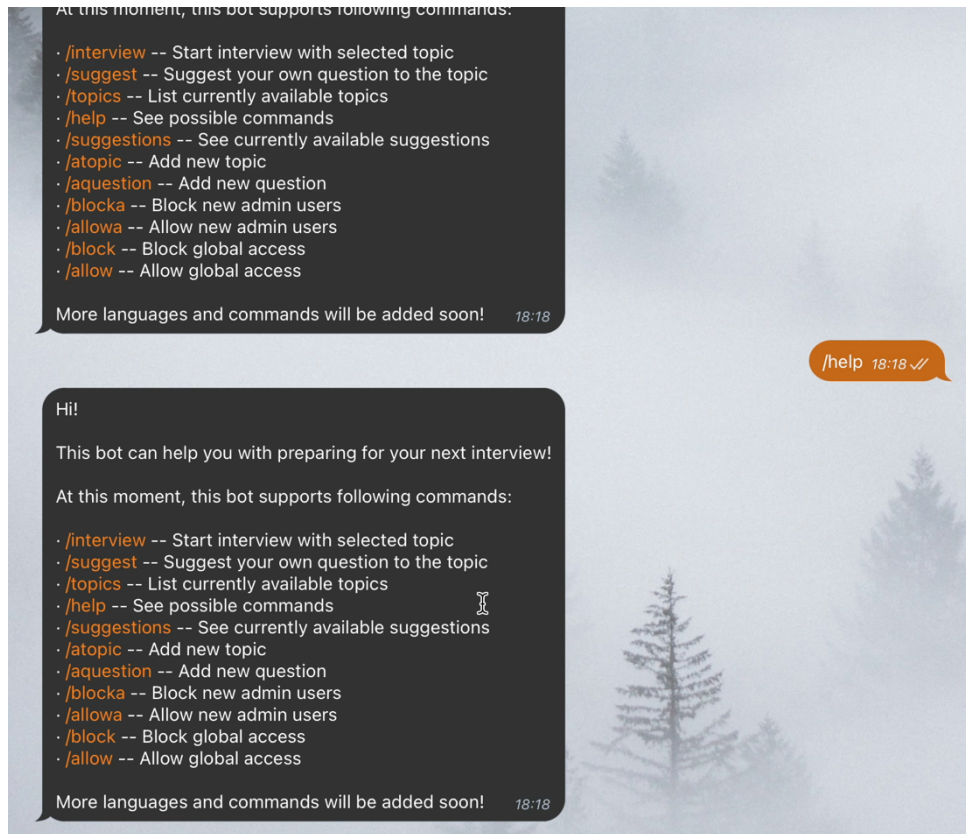
db:
  host: "ADDRESS"
  port: PORT
  user: "USERNAME"
  password: "PASSWORD"
  dbname: "DATABASE_NAME"
  ca: "PATH_TO_CA_CERT"
```

force-admin-key – если true, то перезапишет случайный ключ на указанный в поле admin-key.

docker-compose.yml – *(не используется)* компоуз-файл для разворачивания dev-среды. Не используется при разворачивании на сервере.

Makefile – содержит набор команд для сборки и выгрузки приложения на сервер.

Пример работы

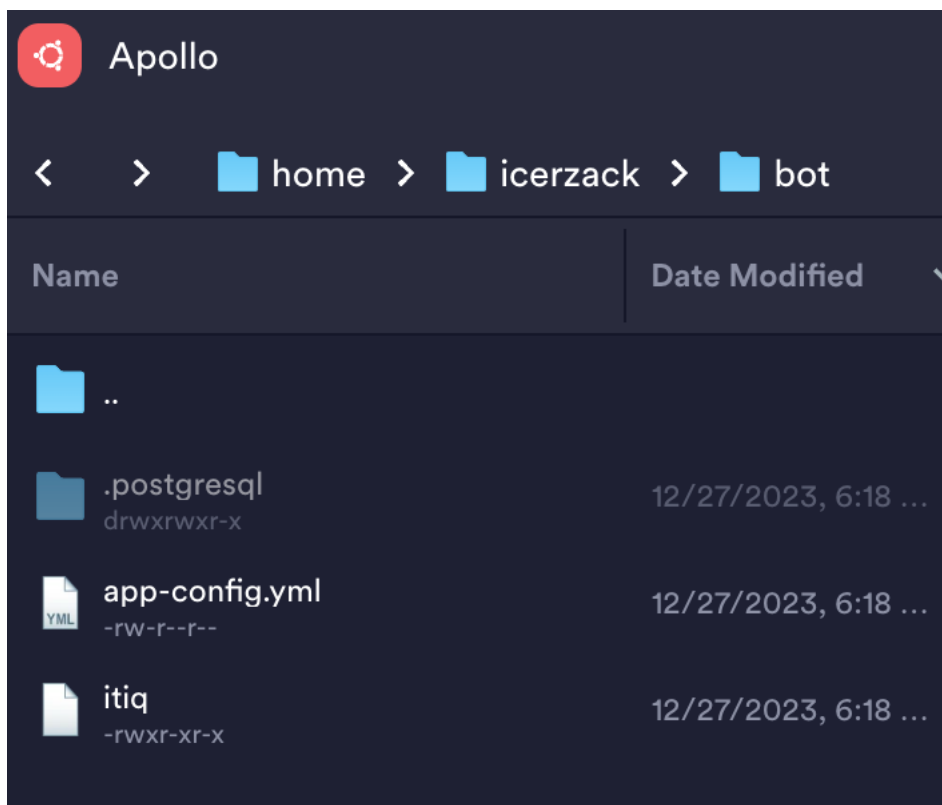


(пкм – воспроизведение)

Ссылка: @ITIQTGTelegram_Bot

Сервисы Yandex Cloud

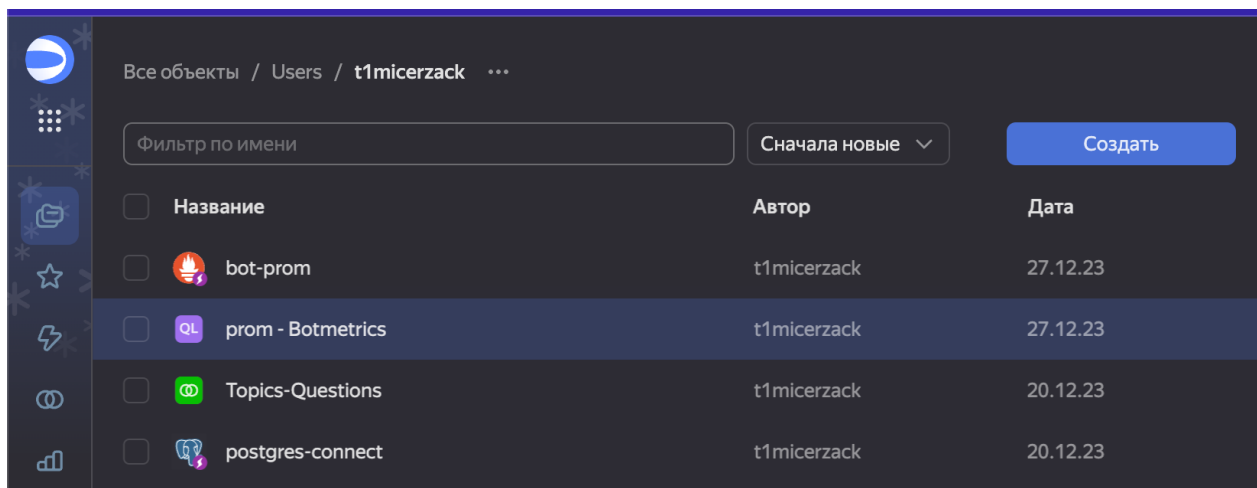
Apollo – виртуальная машина, минимальная конфигурация, прерываемая. На ней запускается собранная программа:



.postgresql – содержит .crt для подключения к БД

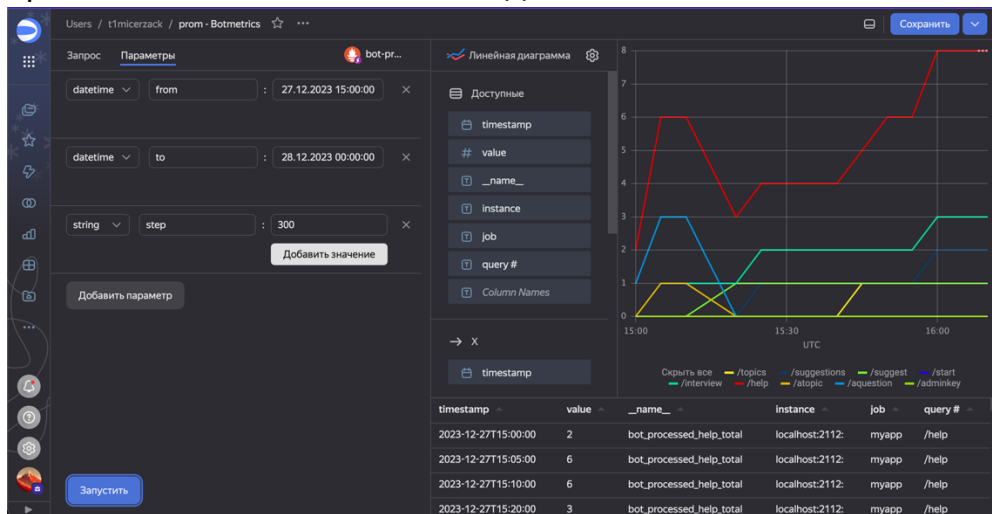
app-config.yml – конфигурация приложения

itiq – запускающий файл



DataLens – для сбора метрик. В качестве примера созданы два чарта\дашборда:

1. prom-Botmetrics (QL-чарт) для Prometheus как пример вывода по времени использования команд:



2. Topics-Questions для PostgreSQL как пример вывода содержимого БД:

The screenshot shows the Topics-Questions interface. It displays a table with columns for ID, Name, Source, Type, Aggregation, and Description. The table contains several rows of data.

#	Имя	Источник поля	Тип	Агрегация	Описание
1	id	questions.id	# Целое число	Нет	
2	q_text	questions.q_text	Строка	Нет	
3	a_text	questions.a_text	Строка	Нет	
4	image_id	questions.image_id	Строка	Нет	
5	topic_name	questions.topic_name	Строка	Нет	
6	created_at	questions.created_at	Дата и время	Нет	
7	updated_at	questions.updated_at	Дата и время	Нет	

The screenshot shows the Managed Service for PostgreSQL interface. It displays the overview of the Hephaestus cluster, including general information, availability, and resources.

Обзор

Общая информация

- Имя: Hephaestus
- Идентификатор: c9q7t2n29b4rusl32edn
- Дата создания: 20.12.2023, в 12:27
- Окружение: PRODUCTION
- Версия: 15

Доступность Alive

Все хосты работают нормально, все запущенные операции были успешно выполнены.

Ресурсы

PostgreSQL

- Класс хоста: c3-c2-m4 (2 vCPU, 100% vCPU rate, 4 ГБ RAM)
- Хранилище: 10 ГБ network-ssd

3 917,30 ₽ в месяц

Тарифы и цены

- PostgreSQL. Intel Ice Lake. 100% vCPU: 2 347,20 ₽
- Публичный IP-адрес - PostgreSQL: 172,80 ₽
- Быстрое сетевое хранилище — PostgreSQL: 130,10 ₽
- PostgreSQL. Intel Ice Lake. RAM: 1 267,20 ₽

Hephaestus – DbaaS от Yandex Cloud, в котором мы храним наши вопросы, топики, предложения.