

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут» імені Ігоря Сікорського
Факультет інформатики та обчислювальної техніки
Кафедра автоматики та управління в технічних системах

Лабораторна робота № 5

З дисципліни: «Компоненти програмної інженерії – 2.
Моделювання та аналіз програмного забезпечення»

Виконав:

студент групи ІТ-73

Старовойтов Руслан Олександрович

Перевірив:

Галушко Д. О

Київ 2020

Тема: Разработка конструктора Telegram-ботов.

Цель: создание даталогической модели.

В базе данных планируется хранить информацию о платежах пользователей и информацию для вычисления стоимости услуг. Поэтому необходимо обеспечить надёжность хранения данных. Желаемую надёжность может обеспечить реляционная система управления базами данных (РСУБД). Желательно, чтобы для этой РСУБД уже существовала библиотека, для работы с ORM. Использование ORM даёт возможность ускорить процесс тестирования сервиса за счет создания БД в памяти компьютера при выполнении unit тестов. Так же при использовании ORM можно использовать code first подход, что сэкономит время при необходимости менять даталогическую модель в процессе разработки.

Требования к РСУБД:

- 1) бесплатность;
- 2) соответствие требованиям к надёжности;
- 3) наличие библиотек для работы с выбранной ORM;
- 4) простая установка и настройка.

Согласно этим требованиям был выбран Postgres.

Ниже приведена даталогическая модель.

Опишем требования к таблицам.

Нужно, чтобы пользователь мог войти в аккаунт с помощью email и через протокол авторизации OAuth от Telegram. Сущность **Account** будет хранить имя и количество средств на счету пользователя. Сущности **EmailSign** и **TelegramSign** будут хранить информацию для входа зарегистрировавшихся пользователей в аккаунт. Если пользователь регистрируется с помощью своего email, то до момента подтверждения почты информация о пользователе будет храниться в сущности **TemporaryAccount**.

Нужно, чтобы пользователь мог сменить свой пароль в случае регистрации через email. Запросы на смену пароля до момента подтверждения владения email-ом будем хранить в сущности **PasswordChangeRequest**.

Нужно, чтобы у каждого пользователя могли быть боты. Для ботов должна собираться статистика. Для ботов должна сохраняться информация о бекенд сервере, на котором они запущены. Для ботов нужно сохранить список запросов от из пользователей. Сущность **Bot** будет хранить информацию о токене, разметку бота (наполнение его дерева), информацию о его собственнике. Сущность **Bot-Statistics** будет хранить показатели кол-ва заказов, кол-во уникальных сообщений, кол-во уникальных пользователей. Сущность **LaunchedBotServerEndPoint**

будет хранить информацию о сервере на котором бот запущен. Сущности **Order**, **Inventory**, **InventoryItem**, **SessionText**, **SessionFile**, **Item** хранят всю информацию о запросе пользователя бота (прикреплённый текст/файлы, количество конкретных товаров).

Нужно, чтобы каждый аккаунт содержал разный набор ответов на запросы пользователей бота. При смене ответа на странице управления ботом сообщение будет отправляться пользователю бота. Сущности **OrderStatusGroup** и **OrderStatus** будут содержать информацию о таком ответе. Например: при переключении ответа из состояния «Не просмотрено» на статус «В обработке» пользователю в Telegram от имени бота будет отправлено сообщение “Ваш заказ находится в обработке”.

Нужно, чтобы владелец бота мог назначить модератора для того, чтобы он мог менять содержимое разметки бота. Сущность **Moderator** будет хранить информацию о модераторе конкретного бота.

