

Проект "PackMeBot" (хранение PostgreSQL, SQL))

Описание

Тема - telegram бот-советчик по сборам в поездку.

Краткое описание работы системы.

После приветствия Пользователь вводит поочередно основное направление поездки (например, "Командировка"), затем уточнение (например для направления "Командировка" уточнением может быть "Другой город" или "Москва"). Далее система анализирует историю предыдущих поездок всех пользователей и формирует список вещей, которые были взяты пользователями в аналогичные поездки. Список сортируется по категориям, а в категориях по количеству использований. Затем пользователь переходит к укладыванию вещей. При этом после ввода названия уложенной вещи формируются и возвращаются два списка: сложенных вещей и еще не сложенных. После укладывания последней вещи подготовка считается выполненной.

Подробное описание работы системы.

1. Описание взаимодействия с пользователем

Интерфейс взаимодействия системы и пользователя осуществляется посредством текстовых сообщений. Поступающие сообщения разделяются на команды и произвольный текст.

Взаимодействие с пользователем разделяется на следующие стадии:

1. Приветствие;
2. Выбор направления поездки;
3. Получение, анализ и корректировка списка вещей;
4. Укладывание вещей.

При первом подключении к боту telegram автоматически вводит команду /start, что приводит к отображению приветствующего сообщения с описанием возможностей бота и рекомендуемых команд : /new для создания нового сбора (новой поездки) или /help для просмотра всех команд.

Ввод команды /new переводит пользователя на стадию выбора направления поездки.

Направление поездки состоит из двух частей: основного направления и уточнения.

После ввода команды /new на экран отображается список основных направлений. Предлагаемый список основных направлений является результатом статистического анализа истории поездок всех пользователей. Список отсортирован по количеству использований направлений (размер списка указывается аргументом в соответствующем методе).

В случае, если предлагаемое основное направление пользователю не подходит, он может задать своё посредством использования управляющего символа "+". Например, "+На море". При этом, если вводимое направление уже есть в списке, пользователь получит соответствующее поясняющее сообщение.

После ввода основного направления необходимо ввести уточнение. Пользователю также предлагается список наиболее часто используемых уточнений,

относящихся к указанному основному направлению. И также как и в предыдущем случае можно ввести своё уточнение, причем уже без управляющего символа. Например, для основного направления “Командировка” можно ввести “Москва”.

После того как выбор направления поездки произведен, система отображает пользователю предлагаемый список вещей.

Предлагаемый список вещей является результатом статистического анализа истории поездок всех пользователей. Список содержит все вещи, которые пользователи когда-либо брали в поездку по аналогичному направлению. При этом даже если вы выбрали направление и уточнение для поездки, которых не было в списках, а в истории такое направление уже использовалось вами или другим пользователем, то вы получите список использованных ранее вещей. В противном случае будет выведено сообщение “Список пуст”.

Список вещей можно корректировать. Пояснения действий для корректировки списка отображаются в сообщении после вывода списка или с помощью команды /menu.

Для добавления вещи используется управляющий символ “+”. Далее необходимо ввести название вещи и категорию, к которой эту вещь можно отнести (например, “+Колбаса(Продукты)”). Введение категории не является обязательным условием. Вещи с указанными категориями будут группироваться при обновлении списка. Категорию необходимо вводить в скобках.

Удаление вещей из списка доступно двумя способами. В обоих случаях необходимо использовать управляющий символ “-” и далее либо “- Название вещи”, либо “- номер вещи в списке” (например, “-Паспорт” или “-2”).

Список вещей состоит из двух разделов “Осталось сложить” и “Сложено”. При подготовке списка заполняется раздел “Осталось сложить”. Раздел “Сложено” заполняется на следующей стадии - укладывание вещей.

Когда список подготовлен, можно переходить к укладыванию вещей. На этой стадии пользователь вводит название уложенной вещи и она переносится из раздела “Осталось сложить” в раздел “Сложено”. Переход на стадию укладывания осуществляется с помощью команды /pack. В этом режиме корректировка списка недоступна. Однако, при необходимости можно вернуться на стадию подготовки списка вещей с помощью команды /things. И после редактирования списка вновь вернуться к укладыванию с помощью команды /pack.

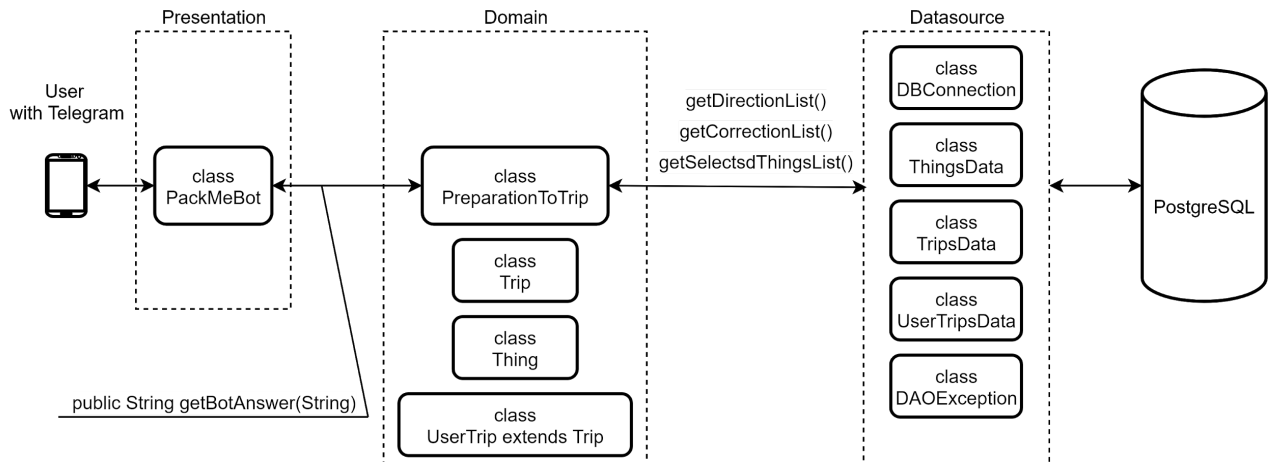
После укладывания последней вещи сбор считается завершенным. Данные о подготовке к этой поездке записываются в базу данных.

Дополнительные команды.

/stop - останавливает сбор и уничтожает введенные данные о направлениях и вещах. После использования этой команды необходимо создать новую поездку командой /new;

/list - вспомогательная команда, позволяющая отобразить текущий список вещей на экран.

2. Структурная схема проекта



Проект имеет трехуровневую структуру, в которую входят следующие уровни:

1. Presentation
2. Domain
3. Data source

2.1 К уровню Presentation относится класс `PackMeBot.java`, отвечающий за связь с приложения с пользователем посредством библиотек, предоставляемых Telegram.

2.2 В уровне Domain реализована бизнес логика приложения. Взаимодействие между уровнями осуществляется с помощью класса `PreparationToTrip.java`. Классы `Trip.java`, `UserTrip.java` и `Thing.java` являются описаниями объектов, используемых в работе приложения.

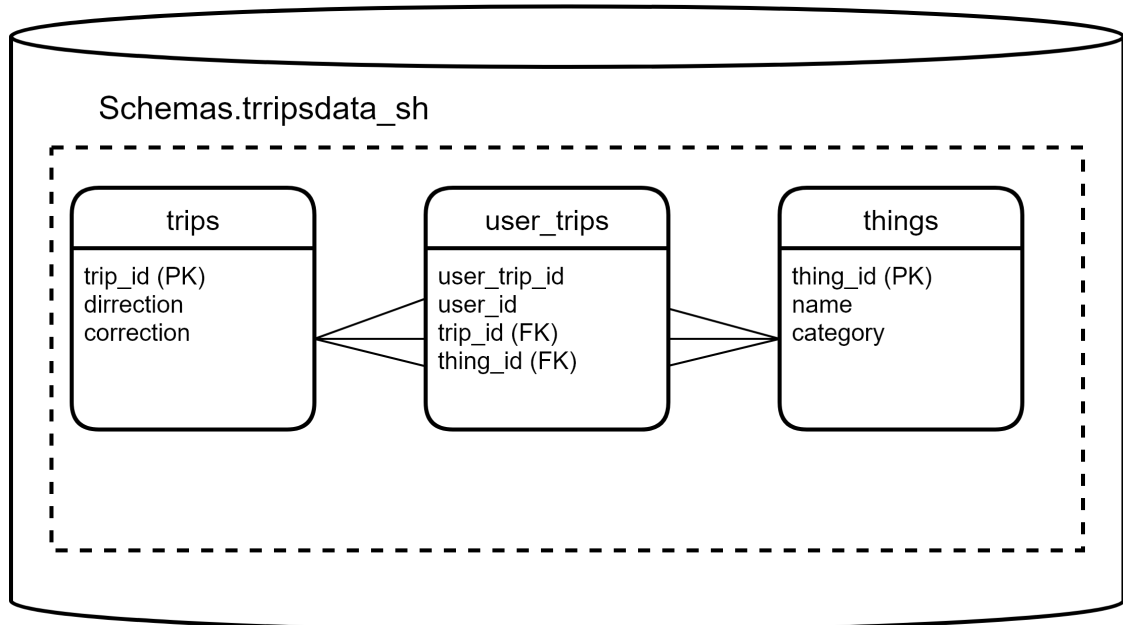
2.3 Уровень Data source представляет собой реализацию паттерна DAO. В `datasource.package` собраны интерфейсы, описывающие взаимодействие классов объектов с базой данных, классы, реализующие соответствующие интерфейсы. А также класс для связи с БД и класс для идентификации исключений, выброшенных на уровне DAO.

3. Описание способа хранения и обработки данных

Для хранения информации о вещах, направлениях поездок и истории поездок пользователей в проекте используется база данных PostgreSQL 13.

В базу данных входят следующие таблицы:

TripsData PostgreSQL



Структура базы данных представлена тремя таблицами: “trips”, “things” и “user_trip”.

Таблица “trips” предназначена для хранения вариантов направлений поездок и уточнений к ним.

Таблица “things” предназначена для хранения общего списка используемых пользователями вещей.

Таблица “user_trip” реализует связь типа “Многие ко Многим” между таблицами “trips” и “things”. Также в нее записывается дополнительная информация о пользователе, собирающимся в поездку.