ТЕХНИЧЕСКАЯ ДОКУМЕНТАЦИЯ

Телеграм-бот «OverEdit»

Содержание

[1 Описание проекта 3](#_Toc157674519)

[2 Основные функциональные возможности: 4](#_Toc157674520)

[3 Уточнение структуры данных: 5](#_Toc157674521)

[4 Формы представления данных 6](#_Toc157674522)

[5 Структура базы данных 8](#_Toc157674523)

[6 Разработка алгоритма решения задачи: 9](#_Toc157674524)

[7 Определение языка, структуры программы и требований к техническим средствам 10](#_Toc157674525)

[7.1 Язык программирования 10](#_Toc157674526)

[7.2 Структура программы 10](#_Toc157674527)

[7.3 Требования к техническим средствам 11](#_Toc157674528)

# Описание проекта

Название проекта: телеграмм-бот для предоставления оверлеев (видеороликов) для творческих проектов.

Цель проекта: разработка телеграмм-бота с целью предоставления оверлеев в видеороликах для редакторских проектов пользователей.

# Основные функциональные возможности:

* отображение оверлеев (видео) пользователю в Telegram;
* выбор оверлеев по тематике;
* добавление и удаление оверлеев в избранное;
* визуализация оверлеев;
* интеграция с мессенджером Telegram.

# Уточнение структуры данных:

**Пользователь:**

* Telegram ID пользователя (user\_id) - уникальный идентификатор пользователя в Telegram;
* никнейм (username) - никнейм пользователя в Telegram;
* тема, которую предпочитает пользователь (current\_theme) - текущая тема оверлеев, которую пользователь выбрал или в которой он заинтересован.

**Оверлеи (видео):**

* номер оверлея (overlay\_id) - уникальный идентификатор оверлея.
* название (title) - название оверлея;
* тема (theme) - тематика оверлея (например, «Бабочки», «Звезды», «Фон»);
* тип (type) - тип оверлея (например, «видео»);
* ссылка на файл оверлея (file\_url) - ссылка на файл с видео оверлеем;
* пометка «избранное» (favorite) - флаг, указывающий, добавлен ли оверлей в избранное пользователем.

**Избранное пользователя:**

* Telegram ID пользователя (user\_id) – уникальный идентификатор пользователя в Telegram;
* номер оверлея (overlay\_id) – уникальный идентификатор оверлея, добавленного пользователем в избранное.

# Формы представления данных

Контекстная диаграмма приложения для Телеграмм «OverEdit» представляет собой высокоуровневую иллюстрацию взаимодействия данного приложения с его внешней средой. Контекстная диаграмма представлена на рисунке 4.1.

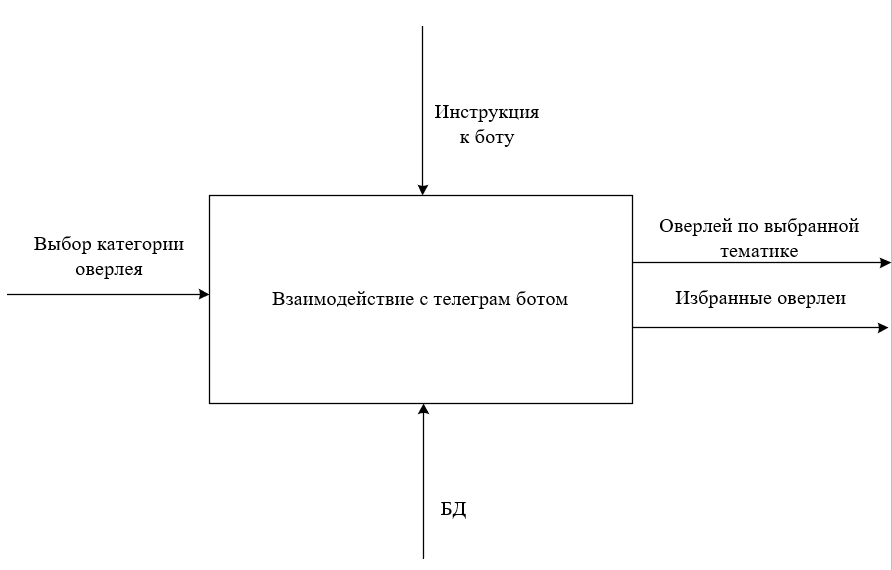


Рисунок 4.1 – Контекстная диаграмма

Ниже приведено описание основных элементов контекстной диаграммы.

**Выбор категории оверлея:** когда пользователь запускает бота и взаимодействует с командой "Категории📋", бот предоставляет пользователю список категорий оверлеев, таких как "Видео🎞️", "Картинки🌆" и др. Пользователь выбирает нужную категорию, и бот предоставляет дополнительные опции, связанные с выбранной категорией (например, тематики видео).

**Инструкция к боту:** при команде "/start" бот приветствует пользователя и предоставляет краткую инструкцию по использованию. Это может включать в себя обзор доступных команд, например, "Помощь❓", "Категории📋", "Избранное❤" и др. Также бот может предложить выбрать категорию оверлеев.

**База данных (БД):** бот использует базу данных для хранения информации о пользователях, оверлеях и их статусе (избранное). В таблице пользователей хранятся Telegram ID, никнейм и текущая тема пользователя. В таблице оверлеев хранятся номер оверлея, название, тема, тип (видео, картинка), ссылка на файл и флаг "Избранное". База данных обеспечивает сохранение пользовательских настроек и управление доступом к оверлеям.

**Оверлей по выбранной тематике:** когда пользователь выбирает тематику оверлея, например, "Бабочки🦋", "Звезды✨", "Фон🌌", бот обращается к базе данных, извлекает соответствующие оверлеи и предоставляет пользователю возможность выбрать конкретный оверлей. Бот может также предложить просмотр предыдущих и следующих оверлеев в выбранной тематике.

**Избранные оверлеи:** когда пользователь выбирает опцию "Избранное❤", бот обращается к базе данных, извлекает избранные оверлеи для данного пользователя и предоставляет их в виде списка. Пользователь может просматривать избранные оверлеи и взаимодействовать с ними, например, добавлять или удалять из избранного. Бот обновляет базу данных согласно действиям пользователя.

# Структура базы данных

Концептуальная ER-диаграмма базы данных «over» представлена на рисунке 5.1

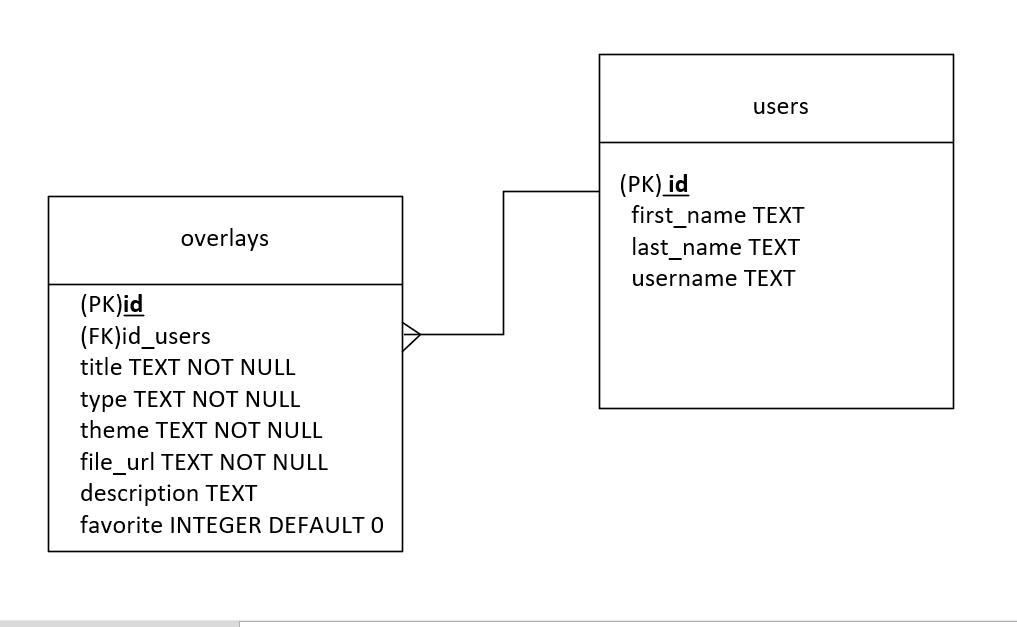


Рисунок 5.1 – Концептуальная ER-диаграмма

Таблица «overlays» содержит информацию о доступных оверлеях, включая их название, тип, тему, URL-адрес файла, описание и флаг избранного.

Таблица «users» хранит информацию о пользователях, включая их идентификатор, имя, фамилию и имя пользователя в Telegram.

# Разработка алгоритма решения задачи:

В коде телеграмм бота OverEdit представлен набор функций для обработки команд и нажатий кнопок пользователей, связанных с выбором оверлеев (видео). Исходный код Телеграм бота представлен в приложении Е. Ниже приведены основные шаги алгоритма решения задачи, реализованного в коде:

**Инициализация бота и базы данных:**

* бот инициализируется с помощью токена и создает соединение с базой данных SQLite.

**Обработка команды «/start»:**

* при получении команды `/start` бот добавляет пользователя в базу данных, если его данные еще не были записаны;
* отправляется приветственное сообщение с кнопками для навигации по функционалу бота.

**Обработка нажатия кнопок:**

* бот обрабатывает нажатия кнопок с помощью функции «callback\_handler»;
* при нажатии кнопки «Помощь» выводится информация о доступных командах и действиях пользователей;
* при выборе категории оверлеев (видео) бот отображает соответствующие видео в зависимости от выбранной тематики;
* пользователь может переключать видео вперед или назад, просматривать избранное, добавлять видео в избранное и удалять из него.

**Избранные видео:**

* при выборе команды «Избранное» бот показывает пользователю список избранных видео;
* пользователь может добавлять просматривать избранные видео.

**Обработка ошибок:**

* бот продолжает работу в режиме опроса (polling) и обрабатывает исключения, чтобы избежать остановки работы при возникновении ошибок.

# Определение языка, структуры программы и требований к техническим средствам

## Язык программирования

Для разработки телеграмм бота OverEdit был выбран язык программирования Python. Этот выбор обоснован простотой и гибкостью языка, а также наличием богатых библиотек и фреймворков для работы с телеграмм API (используется библиотека «telebot»). Python также обеспечивает удобство в обработке баз данных и взаимодействии с другими API.

## Структура программы

Программа структурирована следующим образом:

**Подключение библиотек и модулей:**

* «telebot»: для работы с телеграмм API;
* «sqlite3»: для работы с базой данных SQLite.

**Определение переменных:**

* Переменные для хранения текущего видео, индекса текущего видео, текущей темы и идентификатора пользователя.

**Функции для работы с базой данных:**

* «create\_connection»: установка соединения с базой данных;
* «add\_user\_to\_database»: добавление пользователя в базу данных.

**Обработчики команд и нажатий кнопок:**

* обработчик команды «/start»;
* обработчик нажатий кнопок («callback\_query\_handler»).

**Функции для отображения интерфейса бота:**

* «start»: приветственное сообщение и кнопки;
* Различные функции для формирования разных типов клавиатур (InlineKeyboardMarkup).

**Функции для отправки видео:**

* «send\_video:» отправка видео с заданными параметрами.

**Цикл обработки сообщений:**

* Бесконечный цикл «while True» для опроса обновлений.

## Требования к техническим средствам

Минимальные требования для серверной части чат-бота:

* операционная система: Windows 7 / Linux Debian 12;
* ОЗУ: 4 гб;
* жесткий диск: 5гб;
* быстрое и надежное сетевое подключение с высокой скоростью передачи данных;
* минимальные требования для клиентской части;
* устройство, поддерживающее браузер или приложение Telegram (смартфон, ПК);
* доступ к интернету для взаимодействия с чат-ботом.