Центр Цифрового Образования Детей "IT-cube. Курск"

**Разработка комплекса ботов «Rotex» для двусторонней Discord-Telegram связи**

**и создания общих каналов между Discord серверами**

Авторы проекта:

Урманов Даниил  
Волобуев Михаил

Руководитель проекта:

Борисова Мария Валерьевна

г. Курск

**Введение**

В наши дни популярность соц. сетей неминуемо растёт, а их количество и разнообразие постоянно обновляется. Существует множество разнообразных мессенджеров, каждый из которых уникален по своим свойствам, функциям и предназначению. Например, Telegram позволяет обмениваться как голосовыми, так и видео сообщениями, не выходя из чата, а Discord поддерживает создание голосовых каналов. У каждого из мессенджеров своя уникальная особенность, которой нет у других.

В наши дни у любой крупной организации есть своя группа в Telegram или сервер в Discord-е, а иногда и то, и другое сразу. Помимо прочего, группы людей, объединённые по интересам, нередко создают свои сообщества сразу в нескольких соц. сетях, самыми популярными среди которых являются Discord и Telegram, заслужившие призвание своим удобством и обширным функционалом.

Нередко, при наличии и Discord сервера, и Telegram группы у сообщества или организации, неизбежно появляются люди, которые есть в Telegram, но по тем или иным причинам не могут или не хотят использовать Discord для связи с остальными участниками. Такая же ситуация иногда возникает с теми, кто привык пользоваться Discord но не желает переходить в Telegram. Также, существуют ситуации, в которых у сообщества есть сразу несколько серверов Discord, между которыми постоянно приходится переключаться или быть на всех сразу, участникам, чтобы держать со всеми связь.

**Проблема**

Главной проблемой большого количества соц. сетей у одного сообщества является необходимость для участников присутствовать на всех сразу для того, чтобы получить возможность общаться со всеми. Нередко членам сообществ приходится заводить новые аккаунты в мессенджерах для того, чтобы получить возможность поддерживать связь с группой по работе или интересам.

**Цели и задачи**

Основной задачей данного проекта является простое и доступное каждому объединение Telegram-группы и Discord канала в один чат при помощи простых команд. Также, данный проект должен позволить создавать общие каналы между Discord серверами. Помимо прочего, необходимо реализовать пересылку всех типов сообщений, включая голосовые и видео Telegram-сообщения.

**Этапы разработки**

1. **Постановка задачи:**   
   Реализовать решение для организации двусторонней Discord-Telegram связи и создания общих каналов между Discord серверами.
2. **Выбор метода решения:**  
   В качестве основного языка проекта был выбран Python, поскольку в его экосистеме присутствует множество открытых библиотек как для Discord, так и Telegram ботов, включая библиотеки, использующие асинхронное программирование.
3. **Разработка алгоритма решения задачи:**

Программа должна хранить данные общих чатов всех Discord серверов и Telegram групп, где она используется. Также, основным критерием должна стать скорость работы, позволяющая обрабатывать большое количество серверов одновременно.

Не менее важна простота и доступность для обычного обывателя. Необходимо сделать настройку общих чатов с помощью нескольких простых и понятных команд.

Для реализации двусторонней связи между серверами и группами необходимо создать два бота, один для Telegram, а другой для Discord мессенджера. Боты должны иметь общую базу данных, благодаря которой будет возможна работа связи между Discord каналом и Telegram группой.

На основании вышеизложенного, основными библиотеками были выбраны:  
discord.py (для реализации Discord бота)

aiogram (для реализации Telegram бота)

requests (для пересылки сообщений между чатами)

sqlalchemy (для взаимодействия с БД через ORM)

Также, для реализации базы данных решено было использовать SQLite СУБД.  
  
В ходе разработки алгоритма решения задачи, было решено добавить к функционалу Discord бота несколько дополнительных функций, зачастую необходимых на Discord серверах и упрощающих жизнь пользователей: создание эбмед-сообщений и прогноз погоды.

1. **тестирование программы:**  
   В ходе тестирования были обнаружены и успешно исправлены несколько незначительных багов. Тестирование проводилось в реальных условиях будущей работы, с несколькими уже существующими Discord серверами и Telegram-группами.
2. **запуск в эксплуатацию:**

Для хостинга бота был выбран сервис VPS хостинга <https://timeweb.com/ru/>.

На данный момент, комплекс ботов «Rotex» успешно запущен и работает стабильно. Ознакомиться с результатами работы в действии можно перейдя по ссылкам в разделе «Результаты разработки»

**Алгоритм работы**

Вся программа запускается одним исполняемым файлом «run\_app».

При первом запуске программы автоматически создаётся каталог tempdata для хранения временных файлов, пересылаемых между Discord-каналами и Telegram-группами. Данный каталог автоматически очищается от временных файлов через равные промежутки времени с помощью фонового потока.

Также, при первом запуске автоматически создаётся база данных RotexData.sqlite

Проект состоит из 7 основных файлов:

1. run\_app – сценарий командной строки, запускающий всю программу.
2. discord\_bot.py – скрипт, отвечающий за работу Discord бота.
3. telegram\_bot.py – скрипт, отвечающий за работу Telegram бота.
4. db\_import.py – модуль, импортируемый в два основных скрипта и отвечающий за взаимодействия с базой данных.
5. commands.py – словарь, хранящий в себе информацию о доступных командах Discord бота. Используется при вызове пользователем команды /help
6. tokens.py – модуль, импортируемый в два основных скрипта, содержащий в себе API токены, необходимые для работы Discord и Telegram ботов.
7. requirements.txt – все необходимые для работы программы зависимости.

Для пересылки сообщений из Telegram в Discord, а также для пересылки сообщений между Discord серверами используются Discord вебхуки, а для отправки сообщений из Discord в Telegram используются прямые POST запросы к Telegram API. При получении сообщения в группе или Discord сервере, где установлен бот, создаётся GET запрос к базе данных. В случае, если канал, где отправлено сообщение, будет обнаружен в данных Discord сервера или Telegram группы, в ячейке связанных с другими серверами или Telegram группами каналов, сообщение будет пересылаться по каналам, привязанным к каналу в таблицах связанных групп и каналов.

Вся информация в базе данных хранится в виде Id Discord каналов и Id Telegram групп.

Всего в базе данных 5 таблиц.

1. GUILDS\_DATA – данные Discord серверов, содержащие информацию о связанных каналах других Discord серверов и Telegram групп.
2. GROUPS\_DATA – данные Telegram групп, содержащие информацию о связанных каналах Discord серверов.
3. LINKED\_CHANNELS – все связанные между собой Discord каналы.
4. LINKED\_GROUPS – пары связанных между собой Discord каналов и Telegram групп
5. WEBHOOKS – пары id Discord канала: привязанный к нему вебхук

К одной Telegram группе может быть привязан лишь один Discord канал. В то время как для общего канала между Discord серверами количество участников не ограничено.

Касаемо таких дополнительных функций Discord бота, как прогноз погоды и создание эмбед сообщений: Прогноз можно получить с помощью простой команды /weather, указав название города или страны как аргумент. Для получения прогноза погоды используется <https://openweathermap.org/api>.

Для создания эмбед-сообщений используется упрощённый json-синтаксис. Подробное руководство по созданию эмбед-сообщений можно просмотреть с помощью /post\_help. Информация о всех доступных командах Discord-бота и Telegram-бота, а также о том, как создавать общие чаты - /help

**Результаты разработки**

На данный момент, комплексботов для двусторонней Discord-Telegram связи

и создания общих каналов между Discord серверами размещён на виртуальном частном сервере и стабильно работает. Чтобы воспользоваться функционалом проекта, необходимо:

пригласить в целевую Telegram-группу следующего бота:  
<https://t.me/RotexTgBot>  
затем добавить на Discord сервер бота по следующей ссылке: <https://discord.com/api/oauth2/authorize?client_id=1088509697524519115&permissions=8&scope=bot%20applications.commands>

Ссылка на репозиторий проекта:

<https://github.com/Ranger295/YandexLyceum>