

«Создание музыкального бота с голосовыми командами для приложения Discord на языке программирования Python»

Ученик :
Ездаков Вадим
Сергеевич
10 «И» класс
Государственное
Бюджетное
Образовательное
Учреждение школа
номер 1547
г. Москва

ПРОБЛЕМА:

Отсутствие музыкального бота с поддержкой голосовых команд для приложения Discord.

ЦЕЛЕВАЯ АУДИТОРИЯ: 12+ (пользователи мессенджера Discord)

ЦЕЛЬ: Создать музыкального бота для приложения Discord .

ЛИТЕРАТУРА: статьи на Habr и GitHub, документации библиотек используемых в проекте

ЗАДАЧИ:

1. Погрузиться в контекст задачи.
2. Создать программу.
3. Провести тест.
4. Провести анализ полученного результата.
5. Разместить свой продукт на платформе Discord.

ВЫБОР ПЛОЩАДИ И ЯЗЫКА ПРОГРАММИРОВАНИЯ ДЛЯ СОЗДАНИЯ ПРОДУКТА

Достоинства площадки

Бот создается и будет размещен на площадке Discord. Discord — это мессенджер с поддержкой голоса и видео. Discord занимает ведущую позицию по популярности среди своих аналогов и предоставляет возможность воспользоваться библиотекой `Discord.py` [1] для использования возможностей данной площадки.

Выбор языка программирования

Для реализации своего проекта мной был выбран язык программирования Python из-за его преимуществ перед другими языками в плане читабельности кода, а следовательно и в скорости его реализации. Стоит сказать что Python изучают во многих школах России и он является одним из наиболее популярных языков в мире.

РАССМОТРЕНИЕ АНАЛОГОВ

Probot [2]:

Многофункциональный бот имеющий большой набор функций от автоматической выдачи ролей до воспроизведения музыки.



Pancake [3]:

Бот располагающий функциями воспроизведения музыки, выдачи ролей и модерации канала.



Yukikaze [4]:

Музыкальный бот поддерживающий воспроизведения музыки из ВК, Яндекс музыки, Spotify, SoundCloud.



АКТУАЛЬНОСТЬ И НОВИЗНА

Данный продукт в основах реализации проигрывания музыки ничего нового в себе не несёт, а вносит некоторое улучшение в плане удобства для пользователя.

Голосовых команд нет ни у одного по крайней мере популярного аналога в Discord – это самое главное усовершенствование, которое добавляет данный продукт.

На данный момент актуальны боты, работающие с разными площадками из-за того, что у слушателей плейлисты с их любимой музыкой работают на разных площадках.

ПОСТАНОВКА ЗАДАЧ

1. Реализация подключения бота к голосовому каналу
2. Осуществление скачивания и воспроизведения аудиозаписей
3. Своевременное покидание голосового канала
4. Дополнительные функции (Голосовые команды, функции плеера)

ПРЕДЛАГАЕМЫЕ ФУНКЦИИ

Реализуем подключение к голосовому каналу в Discord используя библиотеку `Discord.py`, которая поможет узнать индекс голосового канала и подключить к нему бота (код см. Приложение [5]). Покидание канала и функции плеера осуществляется, также используя данную библиотеку.

Скачивание осуществим с помощью библиотек для взаимодействия с API той площадки откуда будет скачен файл, например `youtube_dl` [6] для скачивания с YouTube или `vk_audio` [7] для скачивания из ВКонтакте.

Распознавание голоса сделаем с помощью API `Google speech`.

ПОЛУЧЕННЫЙ ПРОДУКТ

На данный момент продукт представляет собой файл запускающийся в интерпретаторе. Чтобы начать проигрывание нужно чтобы бот был запущен со стороны владельца, пользователю же требуется добавить бота на сервер по ссылке, после зайти в голосовой канал и отправить команду содержащую ссылку на поддерживаемый ресурс в текстовый канал сервера, затем бот подключится в канал, скачает по ссылке контент и воспроизведёт его в голосовой канал, чтобы бот вышел из канала нужно также вписать соответствующую команду.

Функции плеера такие как пауза или очередь находятся в стадии разработки, а распознавание речи и голосовые команды воспроизведения пока не реализованы.

ПЕРСПЕКТИВЫ РАЗВИТИЯ

1. Добавление поддержки разных площадок
2. Улучшение текстового интерфейса (добавление кнопок под сообщением с возможностью при нажатии на них ставить на паузу или переключать треки в очереди)