

ПЕТРОЗАВОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНСТИТУТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
КАФЕДРА ИНФОРМАТИКИ И МАТЕМАТИЧЕСКОГО ОБЕСПЕЧЕНИЯ

Направление подготовки бакалавриата

09.03.04 Программная инженерия

Профиль направления подготовки бакалавриата

“Системное и прикладное программное обеспечение”

Курсовая работа по теме:

СОЗДАНИЕ БОТА НА ПЛАТФОРМЕ DISCORD

Выполнил

студент группы 22207:

А. П. Бровкин _____
подпись

Научный руководитель

к.т.н., доцент кафедры ИМО:

С. А. Марченков _____
подпись

Итоговая оценка

оценка

Петрозаводск — 2022

Содержание

Введение	3
1 Реализация	6
1.1 Создание дискорд сервера	6
1.2 Создание бота	7
1.3 Активация бота и его тестирование	10
1.4 Добавление большего функционала (добавляем команды)	14
2 Создание сайта	19
3 Вывод	20
4 Список литературы	21

Введение

Discord - это бесплатное приложение для голосового, видео и текстового общения, которым пользуются десятки миллионов людей, чтобы общаться с друзьями и участниками сообществ.

Несмотря на то, что изначально Discord создавался для игрового сообщества, его аудитория на сегодняшний день намного разнообразнее. Ежедневно люди используют Discord для самых разных целей: от обсуждения художественных проектов и семейных поездок до проверки домашних заданий и организации психологических консультаций. Discord открыт для сообществ самых разных размеров, но наиболее широко используется небольшими активными группами людей для ежедневного общения.

Большинство серверов являются частными, куда можно попасть только по приглашению и где друзья и участники сообществ могут общаться и проводить время вместе. Существуют и более крупные и открытые сообщества, обычно сосредоточенные вокруг определенных тем, например, популярных игр, таких как Minecraft и CS:GO. Здесь люди могут общаться о чем угодно и свободно делиться своим опытом использования Discord.

Люди любят Discord, потому что он является общим домом для их друзей и сообществ, в которых они состоят. Это место, где можно быть самим собой и проводить время с людьми, разделяющими их интересы и увлечения. Здесь нет алгоритма, определяющего, что именно они должны видеть, нет бесконечной прокрутки и новостной ленты. Общение в Discord основано именно на общих интересах.

Также Discord предоставляет обширный API, который разработчи-

ки могут использовать для создания весьма функциональных Discord-ботов, а другие пользователи в последствии могут его использовать для улучшения своего сервера (к примеру бот предоставляющий музыку).

Цели:

- Приобрести навыки и опыт работы с платформой Discord, с точки зрения разработчика
- Повысить свою квалификацию по ходу работы с JavaScript, Node.js, JSON
- Закрепить имеющиеся навыки во время работы с:
 - Языками разметки: LaTeX, Beamer, HTML
 - Языком программирования: JavaScript
 - Языком таблиц стилей: CSS
 - Веб-сервисом для хостинга: GitHub

Задачи:

- Создание дискорд сервера для тестирования работы бота
- Создание бота
- Активация бота и его тестирование
- Добавление большего функционала (добавляем команды)
- Создание сайта для скачивания бота другим пользователям

- Проанализировать полученные результаты и сформулировать выводы, что удалось реализовать, что не удалось, какой получили опыт в результате работы
- Создание документации
- Отправление документации и кода на репозиторий GitHub

1 Реализация

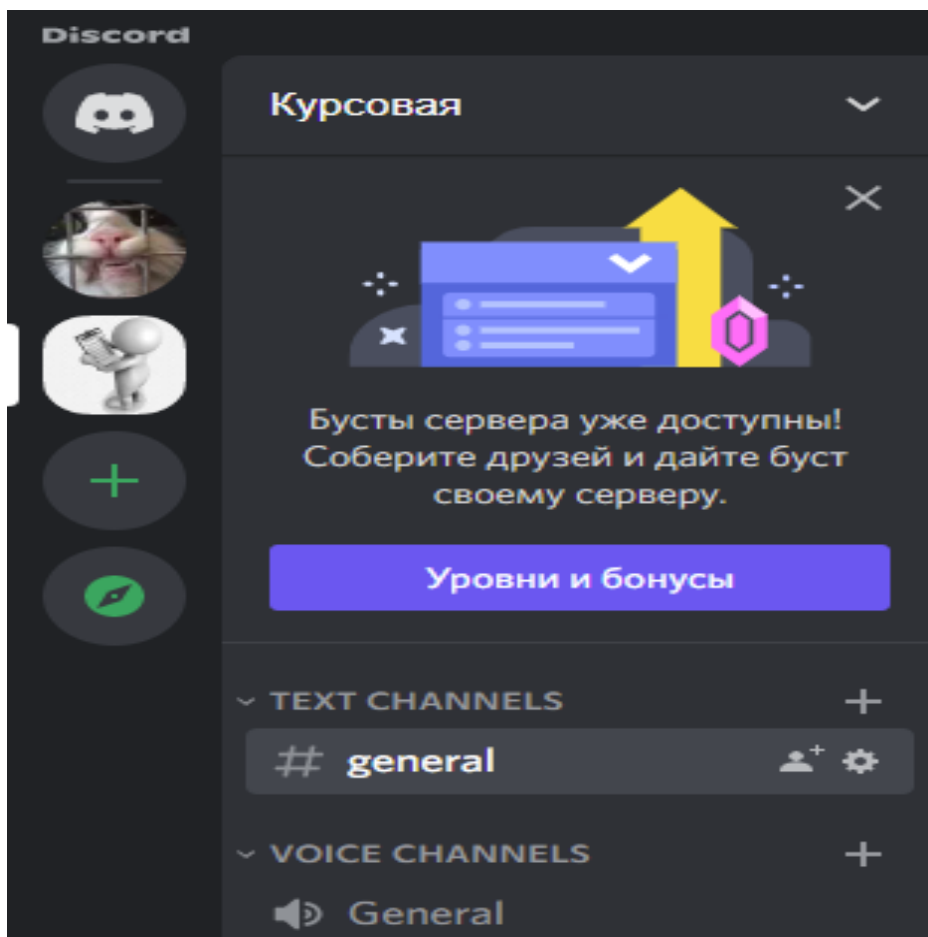
1.1 Создание дискорд сервера

Нажимаем на кнопку "+" в левой колонке

Следующее появившееся окно предложит две опции: "Создать" или "Войти". В данном случае выбираем "Создать"

После выбора между "Создать" и "Войти" указываем название сервера, а также меняем его иконку, указав ее из своих файлов.

Также можем указать регион сервера, но для работы это не пригодится



1.2 Создание бота

Чтобы создать бота, перейдём в Discord для разработчиков. Сначала вам нужно создать приложение, затем в этом приложении создать бота и настроить для него разрешения, и только после этого — добавлять бота на сервер.

- 1. На вкладке Applications выбираем New Application.
- 2. Вводим название будущего приложения и нажимаем Create.
- 3. Приложение создано. Переходим на вкладку Bot и нажимаем Add Bot, чтобы добавить нового бота.
- 4. Соглашаемся добавить бота в наше приложение.
- 5. Бот создан. На вкладке Bot отобразится вся информация о нём. Тут можно изменить его имя, добавить изображение и скопировать токен бота. Этот токен понадобится нам для настройки модуля Discord. Сохраняем токен бота
- 6. Теперь перейдём на вкладку OAuth2 — тут можно настроить разрешения и получить ссылку на нашего бота. В разделе SCOPES выберем bot, в BOT PERMISSIONS отметим разрешения, которые предоставим ему. После копируем автоматически сгенерированную Discord ссылку.
- 7. Перейдём по ссылке и добавим бота на сервер
- 8. На сервере бот оставил приветственное сообщение — значит, что он успешно добавлен и функционирует.

DEVELOPER PORTAL

Applications

Teams

Documentation

DEVELOPER NEWSLETTER

Subscribe to our newsletter and stay updated about cool ways you can build on Discord.

Sign upNo thanks


Applications

Find the perfect feature set for your game in our Game SDK, and sign up for Server Commerce to start selling directly in your server. Get started by creating a new application. We can't wait to see what you make!

Sort By: Date Created

SmallLarge

My Applications



botik

← Back to Applications

SELECTED APP

botik

SETTINGS

General Information

OAuth2

Bot

Rich Presence

App Testers

DEVELOPER NEWSLETTER

Subscribe to our newsletter and stay updated about cool ways you can build on Discord.

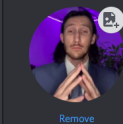
Sign upNo thanks

General Information

What should we call your creation? What amazing things does it do? What icon should represent it across Discord? Tell us here!

By creating an API application, you agree to the [Discord API Terms of Service](#)

APP ICON



Remove

NAME

botik

DESCRIPTION (MAXIMUM 400 CHARACTERS)

Your description will appear in the About Me section of your bot's profile.

Это бот которого я создал для своего курса, но можете пользоваться

TAGS (MAXIMUM 5)

Add up to 5 tags to describe the content and functionality of your application.

#bot x #kursovaya x

APPLICATION ID

974676423363133491

Copy

PUBLIC KEY

ab43fa8ab218a9adf37183eff6c7d6b6a55153a188121538486398d988626b89

Copy

INTERACTIONS ENDPOINT URL

You can optionally configure an interactions endpoint to receive interactions via HTTP POSTs rather than over Gateway with a bot user.

https://nice-example.local/api/interactions

TERMS OF SERVICE URL

Back to Applications

SELECTED APP

botik

SETTINGS

General Information

OAuth2

General

URL Generator

Bot

Rich Presence

App Testers

DEVELOPER NEWSLETTER

Subscribe to our newsletter and stay updated about cool ways you can build on Discord.

Sign up

No thanks

☐ guilds.join
☐ guilds.members.read
☐ gdm.join
☐ rpc
☐ rpc.notifications.read

☐ webhooks.incoming
☐ messages.read
☐ applications.builds.upload
☐ applications.builds.read

☐ activities.write
☐ relationships.read
☐ voice
☐ dm_channels.read

BOT PERMISSIONS

GENERAL PERMISSIONS

☒ Administrator
☐ View Audit Log
☐ Manage Server
☐ Manage Roles
☐ Manage Channels
☐ Kick Members
☐ Ban Members
☐ Create Instant Invite
☐ Change Nickname
☐ Manage Nicknames
☐ Manage Emojis and Stickers
☐ Manage Webhooks
☐ Read Messages/View Channels
☐ Manage Events
☐ Moderate Members

TEXT PERMISSIONS

☐ Send Messages
☐ Create Public Threads
☐ Create Private Threads
☐ Send Messages in Threads
☐ Send TTS Messages
☐ Manage Messages
☐ Manage Threads
☐ Embed Links
☐ Attach Files
☐ Read Message History
☐ Mention Everyone
☐ Use External Emojis
☐ Use External Stickers
☐ Add Reactions
☐ Use Slash Commands

VOICE PERMISSIONS

☐ Connect
☐ Speak
☐ Mute Members
☐ Deafen Members
☐ Move Members
☐ Use Voice Activity
☐ Priority Speaker

GENERATED URL

https://discord.com/api/oauth2/authorize?client_id=974676423363133491&permissions=8&scope=bot

Copy

9

1.3 Активация бота и его тестирование

Для начала работы с кодом нам нужно установить среду разработки.

Среда разработки выбирается по удобству использования и практичности, я выбрал Visual Studio Code

Для создания бота используем среду выполнения node.js

Подготовка к написанию кода

- Первым делом создаём папку, после чего открываем её в VS Code
- Теперь создаём файл с неким «описанием» нашего бота, сделаем это через терминал. Вписываем данную строку в терминал: **npm init**

После каждой появившейся строки нажимаем Enter или вписываем свои значения. Значения в этом файле можно будет изменить в любой момент.

- Устанавливаем нужные пакеты и модули, чтобы в дальнейшем наш бот корректно работал.

Поочерёдно вводим следующие команды:

npm install

npm install discord.js axios dotenv

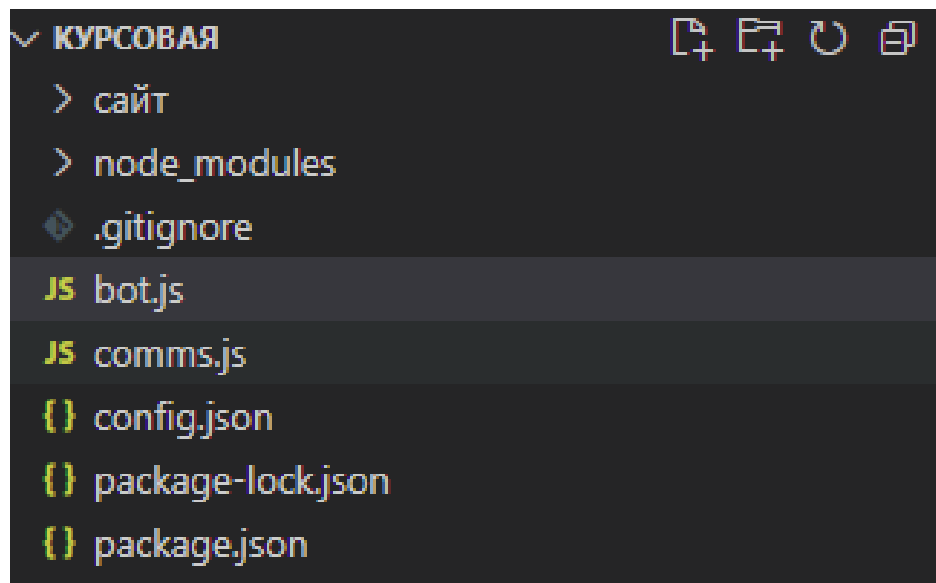
- Итого, в нашей папке появляется 3 объекта: nodeModules, package.json, package-lock.json

Написание кода

Для того, чтобы наш бот появился в сети и мог реагировать на команды, нам нужно написать для него код.

Код будем писать используя два файла, чтобы легко ориентироваться, один для запуска бота, другой для создания команд (также есть и другие файлы, но к ним редко приходится обращаться)

- Для начала создадим файл `config.json`, который будет содержать токен нашего бота и префикс (символ который нужно будет приписать к команде, чтобы бот отреагировал на наше сообщение)
- Далее создаём файл для запуска бота, назовём его `bot.js`
- Напишем для него код, а именно подключаем его к библиотекам, работаем с префиксом и токеном, создаём функцию которая нам поможет понять работает бот или нет, а также функцию реагирования на наши сообщения
- Теперь создаём файл `comms.js`, в нём будут сами команды.
- Также подключаем к библиотекам и связываем с `bot.js`, выставляем префикс
- Пишем первую команду, для того чтобы протестировать что бот работает и формируем список команд, для вызова команды в чате



```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

(node:16128) DeprecationWarning: The message event is deprecated. Use messageCreate instead
(Use `node --trace-deprecation ...` to show where the warning was created)
PS C:\Users\DNS\Desktop\курсовая> node bot.js
botik запустился!
```

```

const Discord = require('discord.js'); // Подключаем библиотеку discord.js
const robot = new Discord.Client(); // Объявляем, что robot - бот
const comms = require('./comms.js'); // Подключаем файл с командами для бота
const fs = require('fs'); // Подключаем родной модуль файловой системы node.js
let config = require('./config.json'); // Подключаем файл с параметрами и информацией
let token = config.token; // «Вытаскиваем» из него токен
let prefix = config.prefix; // «Вытаскиваем» из него префикс

robot.on("ready", function() {
  /* При успешном запуске, в консоли появится сообщение «[Имя бота] запустился!» */
  console.log(robot.user.username + " запустился!");
});

robot.on('message', (msg) => { // Реагирование на сообщения
  if (msg.author.username !== robot.user.username && msg.author.discriminator !== robot.user.discriminator) {
    var comm = msg.content.trim() + " ";
    var comm_name = comm.slice(0, comm.indexOf(" "));
    var messArr = comm.split(" ");
    for (comm_count in comms.comms) {
      var comm2 = prefix + comms.comms[comm_count].name;
      if (comm2 === comm_name) {
        comms.comms[comm_count].out(robot, msg, messArr);
      }
    }
  }
});

robot.login(token); // Авторизация бота

```

```

const config = require('./config.json'); // Подключаем файл с параметрами и информацией
const Discord = require('discord.js'); // Подключаем библиотеку discord.js
const prefix = config.prefix; // «Вытаскиваем» префикс

// Команды //

function test(robot, mess, args) {
  mess.channel.send('Test!')
}

// Список команд //

var comms_list = [{
  name: "test",
  out: test,
  about: "Тестовая команда"
}];

// Name - название команды, на которую будет реагировать бот
// Out - название функции с командой
// About - описание команды

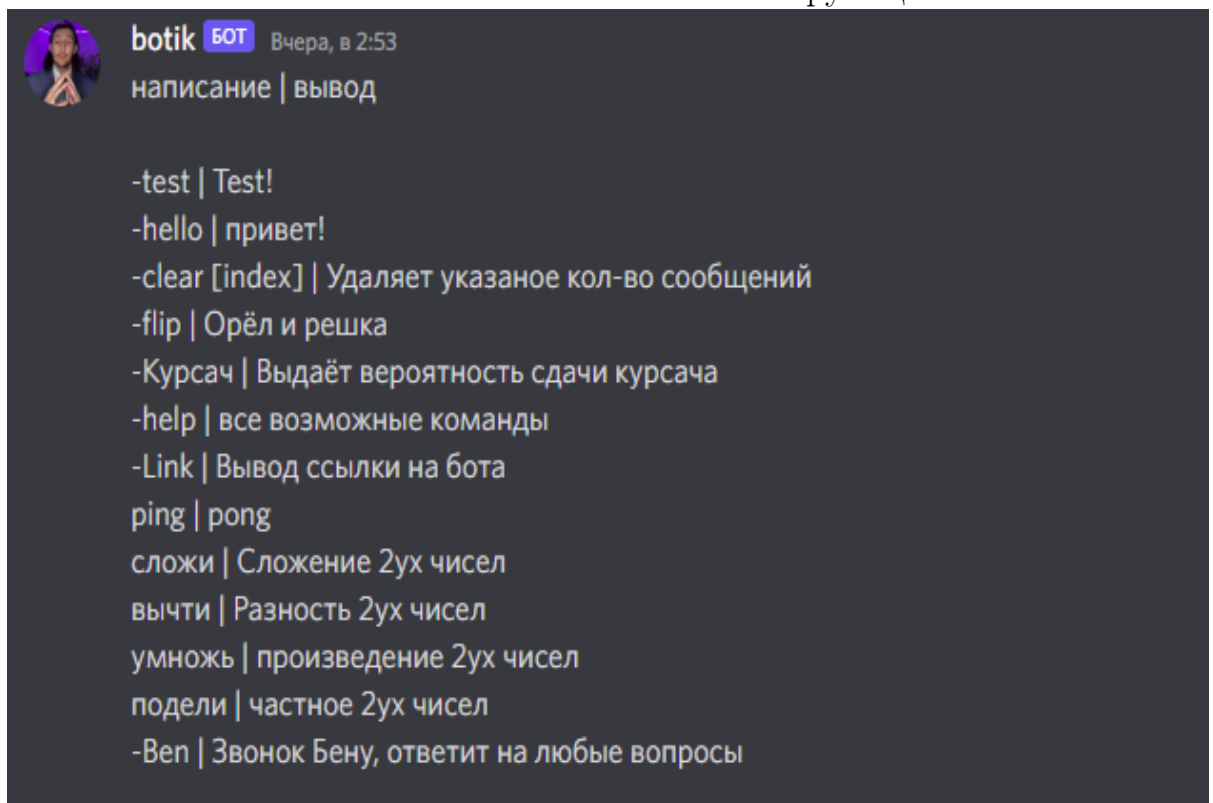
module.exports.comms = comms_list;

```

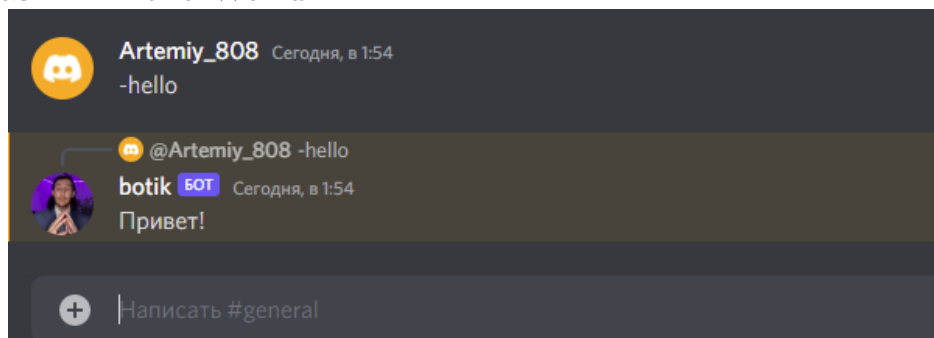
1.4 Добавление большего функционала (добавляем команды)

Бот запущен и вы можете им пользоваться, а значит можно приступить к добавлению новых команд

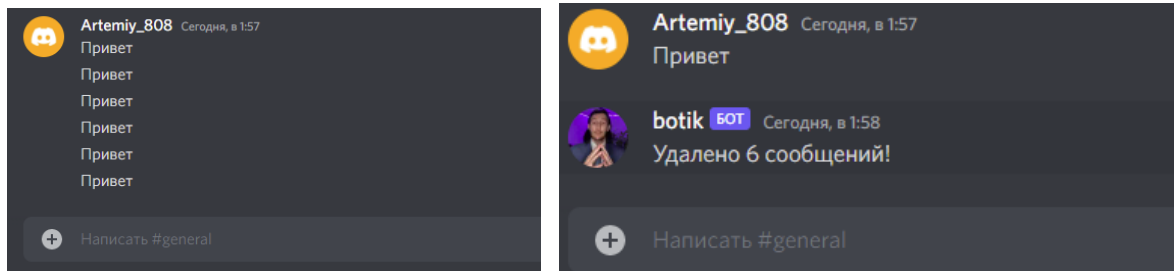
- Начнём с своеобразного Флешфорварда, создаём команду help которая будет выводить все наши команды, это необходимо для того чтобы новые пользователи могли ознакомиться с функционалом бота



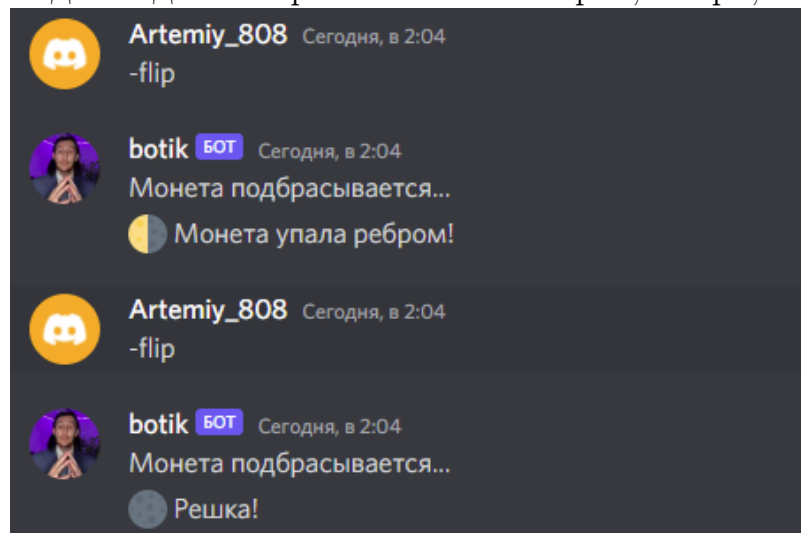
- Первая команда которую мы создадим (после теста конечно), будет своеобразным Hello World!



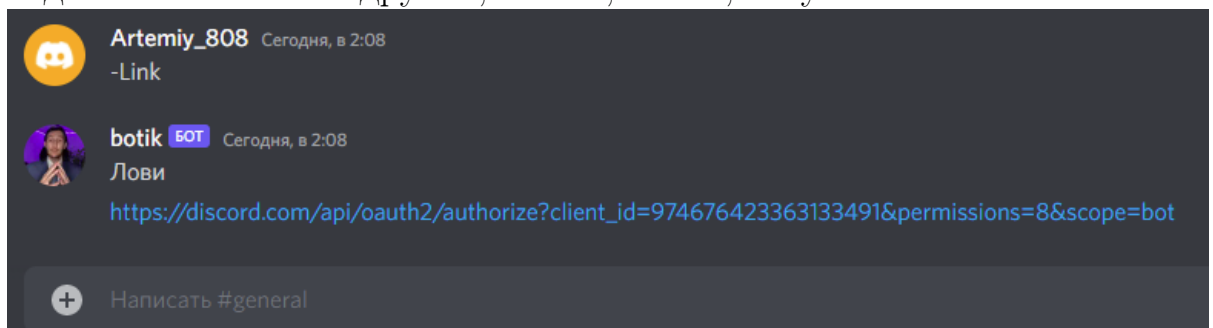
- Следующая команда будет удалять конкретное кол-во сообщений указанное нами



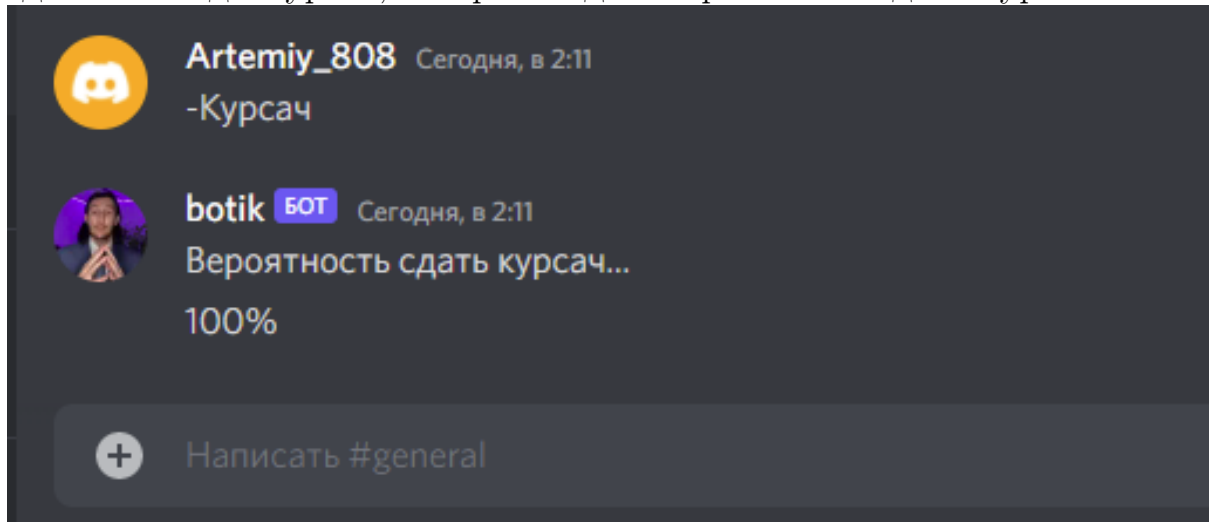
- Для того чтобы разрешать споры между нашими дорогими пользователями сделана команда flip или же орёл и решка, при вызове команды выпадает одно из трёх состояний: Орёл, Ребро, Решка



- Команда для того чтобы получить ссылку на бота, который можно поделиться со своим другом, мамой, папой, бабушкой имя её link

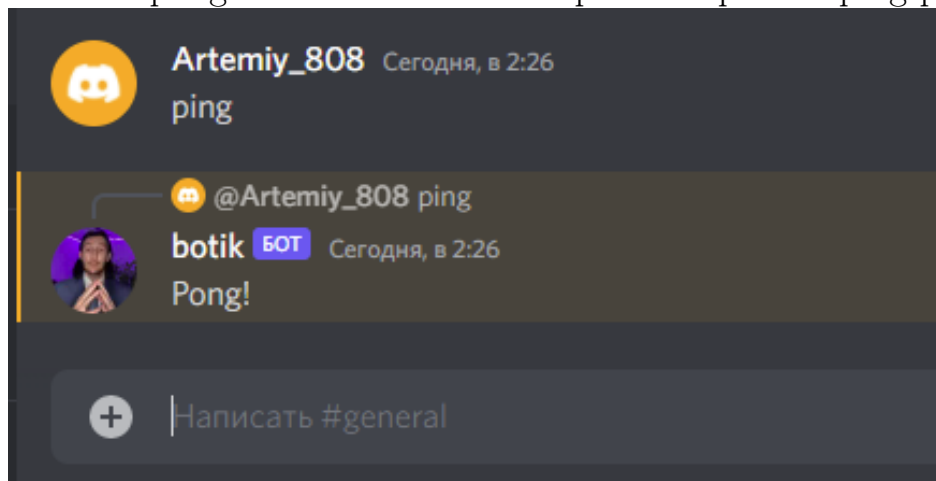


- Далее чтобы скоротать ожидание сдачи курсовой работы была создана команда Курсач, которая выдаёт вероятность сдачи курсовой

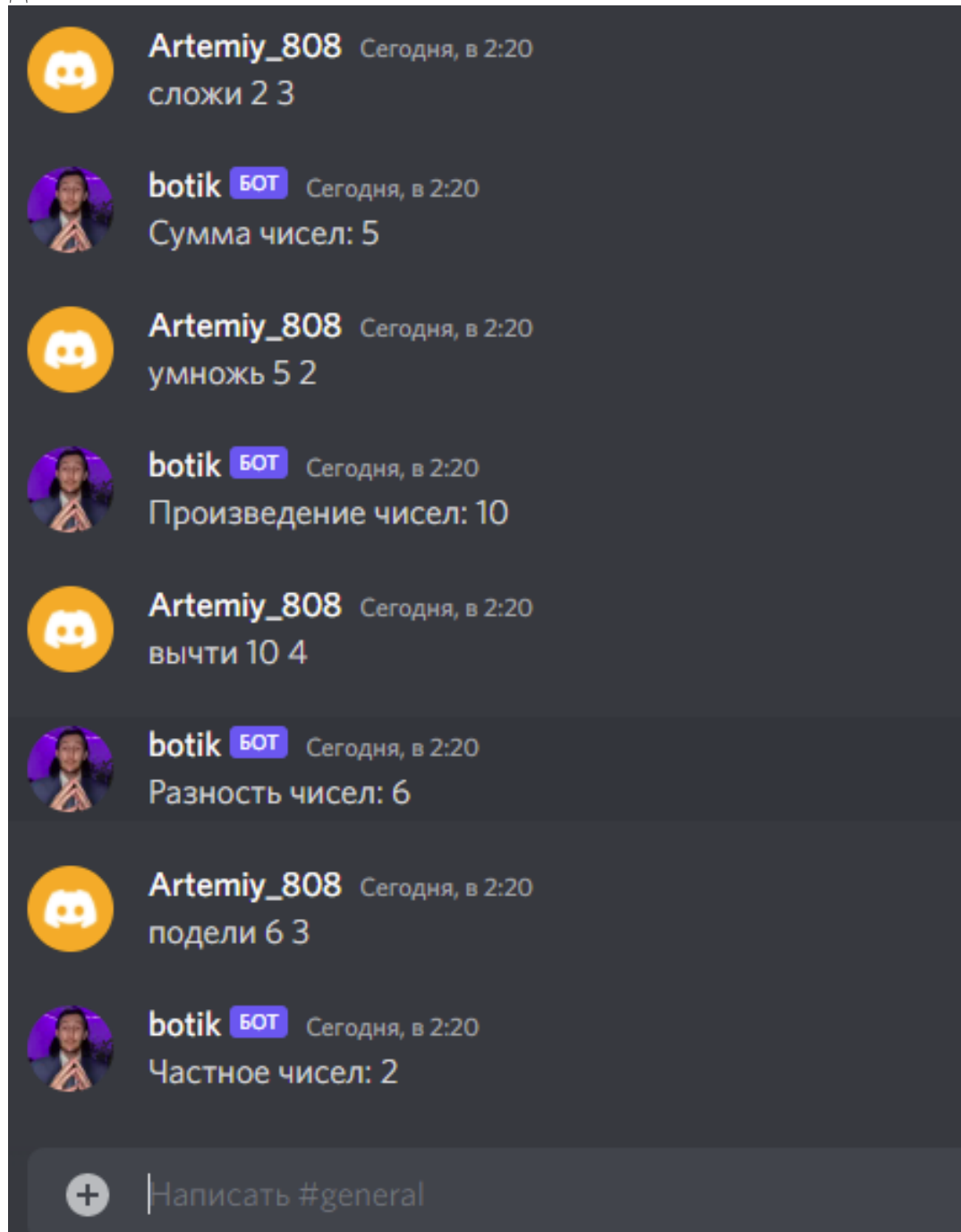


Вау, мне повезло и моя вероятность сдачи курсовой 100 процентов!!!
(после сотого вызова команды)

- Сидите дома, а хочется позаниматься спортом, тогда для вас команда ping(по сути что и test, но для неё не нужен префикс), на неё бот ответит pong и так можно своеобразно играть в ping-pong

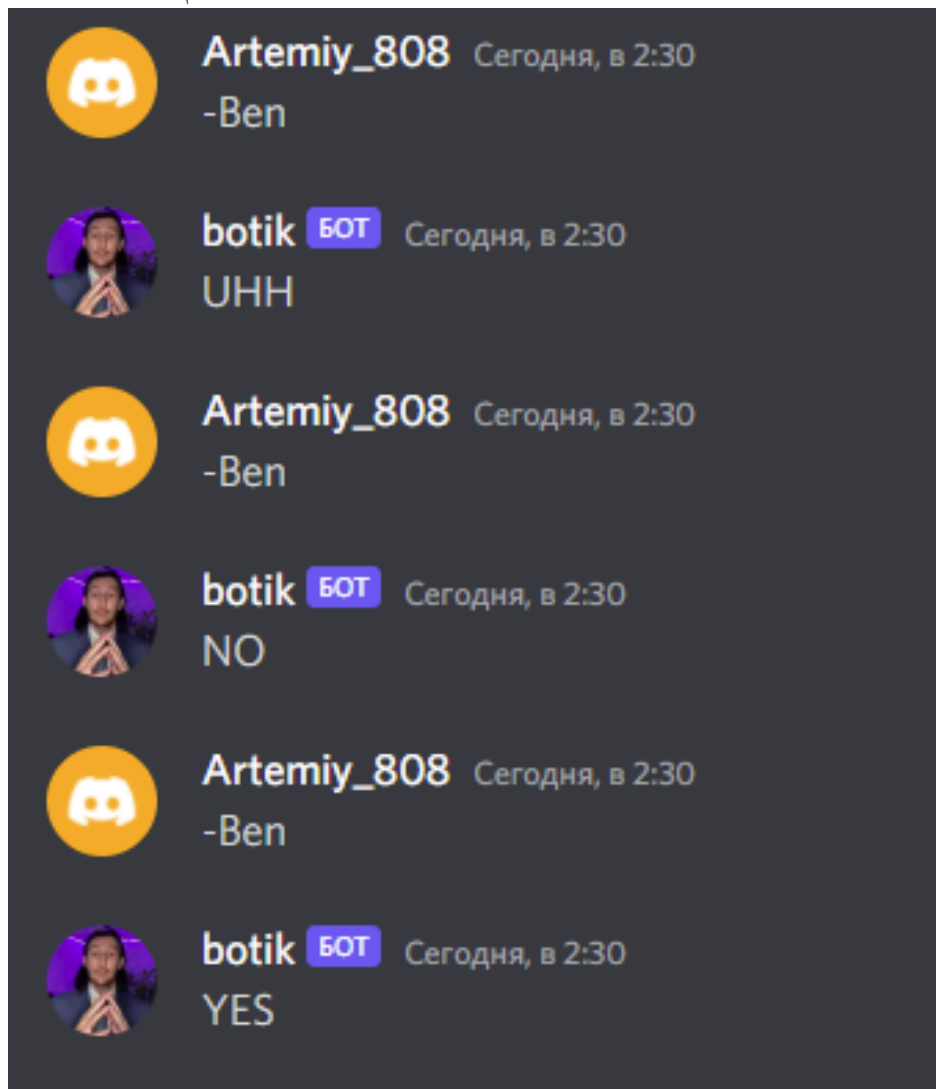


- Если вдруг у вас нету калькулятора под рукой, а вы забыли простейшие правила вычитания, сложения, умножения и деления, то можно воспользоваться следующими командам: сложи, вычти, умножь и подели



Ещё важная особенность этих команд что для них не нужно использовать префикс

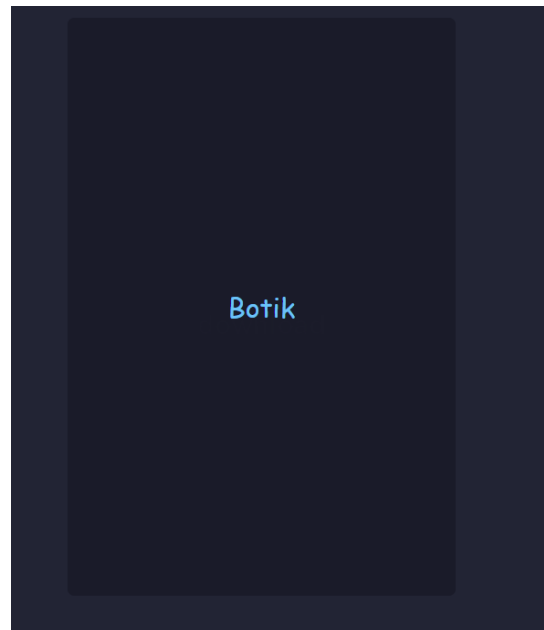
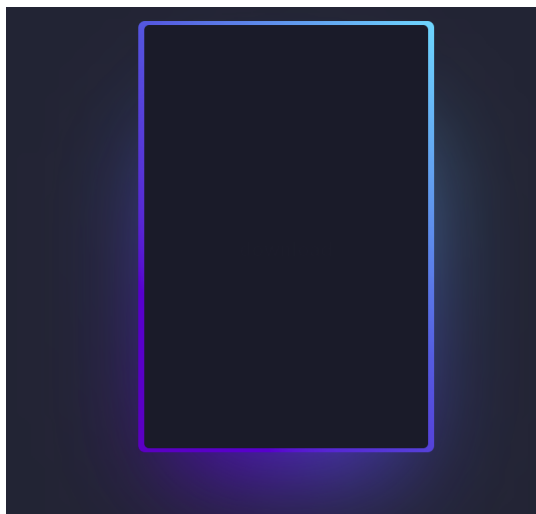
- Ну и последняя команда -Ben, в сети развернулась старая мобильная игра my talking ben, ему можно позвонить и он будет давать вам 4 ответа да, нет, отвращение и смех, я решил добавить своеобразную версию "Ben'a" в функционал своего бота, картинки он не имеет, но на сообщения отвечает



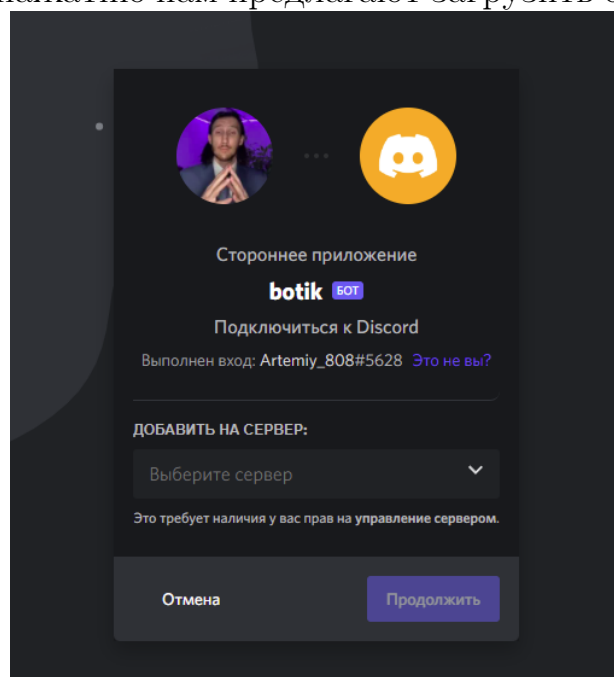
2 Создание сайта

Для того чтобы человек не имеющий ссылки мог скачать бота, создадим сайт в котором будет кнопка с url-ссылкой

Сайт имеет следующий вид:



Представляет собой карту с красивой анимацией, при наведении на неё освещение пропадает, как бы отдавая свою энергию тексту, из-за чего он появляется, а по нажатию нам предлагают загрузить бота на наш сервер



3 Вывод

В своей курсовой работе была рассмотрена работа с Discord для разработчиков, а именно создание discord bot'а, от добавления его на собственный сервер, до активирования с небольшим количеством команд, которым могут пользоваться другие пользователи. В ходе работы был получен опыт работы с Node.js, API, JavaScript, Discord for Developer, JSON, а также закрепление полученных навыков работы с CSS, HTML, GitHub, Latex, Beamer.

Большинство поставленных задач было выполнено, за исключение более глубокой работы с API, например подключение к другому сайту с целью вывода картинки по команде в текстовый канал Discord

Список литературы

4 Список литературы

1. Основная документация discord.js — URL: <https://discord.js.org//>
2. Документация discord.js №2 — URL: <https://discordjs-fork.readthedocs.io/en/latest/index.html>
3. Руководство discord.js — URL: <https://discordjs.guide/before-you-begin>
4. Руководство discord.js №2 — URL: <https://anidiots.guide/>
5. Discord developers — URL: <https://discord.com/developers>
6. Node.js — URL: <https://nodejs.org/en/>
7. Visual Studio Code — URL: <https://code.visualstudio.com/>