Discard botovi

Anđelković, Damjan

Undergraduate thesis / Završni rad

2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Dubrovnik / Sveučilište u Dubrovniku**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:155:394096

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Repository / Repozitorij:

Repository of the University of Dubrovnik

SVEUČILIŠTE U DUBROVNIKU

ODJEL ZA ELEKTROTEHNIKU I RAČUNARSTVO

DAMJAN ANĐELKOVIĆ

DISCORD BOTS

ZAVRŠNI RAD

Dubrovnik, Rujan 2019.

SVEUČILIŠTE U DUBROVNIKU

ODJEL ZA ELEKTROTEHNIKU I RAČUNARSTVO

DISCORD BOTS

ZAVRŠNI RAD

Studij: Primjenjeno/poslovno računarstvo Dubrovnik

Kolegij: Baze podataka

Mentor: Mario Miličević

Student: Damjan Anđelković

Dubrovnik, Rujan 2019.

SAŽETAK

Discord je besplatna aplikacija koja je objedinila mnoštvo različitih funkcija u jedan program. Vrlo je jednostavan za korištenje i za razliku od Teamspeak-a i Skype-a, koristi jako malo resursa te ima veću sigurnost.
Discord kombinira najbolje iz oba programa i upravo radi toga, postao je jako popularan u gaming zajednici.

Botovi su softver kojima je uloga odrađivanje repetitivnih poslova. Oni te poslove odrađuju puno brže od čovjeka. Također ne mogu pogriješiti, ali samo ako čovjek koji ih je programirao nije pogriješio.

Ovaj rad će opširnije opisati svrhu discord-a i botova.
Također će biti prezentiran jedan jednostavan bot koji ima mnoštvo različitih funkcija i objašnjenje svake.

Ključne riječi: discord, bot, javascript, sqlite

ABSTRACT

Discord is a free application which unites a lot of different functions into one program.
It is very easy to use and unlike Teamspeak or Skype, it uses very little resources and has better security.
Discord combines best of both worlds and because of this, it has become very popular in the gaming community.

Bots are software whose main purpose is executing repetitive tasks. They do them far faster than a human being. They also do not make mistakes, but only if the person who wrote them did not make one.

This paper will further explain the purpose of discord and bots. You will have a bigger understanding of what bots can do and what you can use them for.
It will also present one simple discord bot with many functions and each will be explained.

Key words: discord, bot, javascript, sqlite

SADRŽAJ

Sažetak I
Abstract II
1 Uvod .....
1.1 Definicija rada 1
1.2 Svrha i ciljevi rada 1
1.3 Metodologija rada 1
1.4 Struktura rada 1
2 Discord 3
2.1 Kratka povijest 3
2.2 Problemi s Discord-om 3
3 Botovi..... 4
3.1 Vrste botova..... 4
3.1.1 IM i IRC.....4
3.1.2 Socijalni botovi.....4
3.1.3 Komercijalni botovi.....5
3.1.4 Zlonamjerni botovi.....5
3.2 Obrana od zlonamjernih bot-ova.....5
4 Tehnologije korištene u projektu.....6
4.1 Javascript..... 6
4.2 SQLite.....7
4.3 SQL i relacijska baza podataka.....7
4.4 API..... 8
4.5 DB Browser.....9
4.6 VS Code.....10
4.7 JSON..... 10
4.8 Node.js..... 11
4.9 Discord.js..... 11
4.10 Postman..... 12
5 Izrada bot-a.....13
5.1 Prezentacijski sloj 13
5.2 Implementacija baze podataka.....15
5.3 Implementacija funkcionalnosti.....17
6 Opis bot-a.....18
6.1 Upute za discord.....18
6.2 Funkcionalnosti bot-a.....20

7 Zaključak.....31
8 Literatura.....32

1 UVOD

Tek prije nešto više od deset godina su se počeli pojavljivati prvi programi kojima je svrha bila komercijalna globalna komunikacija.
Danas takvih programa imamo napretek, međutim tehnologija stalno napreduje i korisnicima se zahtjevi stalno mijenjaju, stoga programi moraju biti ažurni.
Veliku ulogu igra i jednostavnost. Što više funkcija program može odraditi, a da u isto vrijeme bude i jednostavan, to bolje.
S obzirom da je tehnologija toliko napredna, a korisnika je mnogo, stvorila se potreba za botovima. Botovi su u ovom kontekstu asistenti. Sigurno ste bili u interakciji s botom premda to niste ni znali. Oni su integrirani u skoro svakoj online usluzi.
Svaka akcija se može pretvoriti u algoritam, pa se onda svaka akcija može i automatizirati, tj. može se izraditi bot koji će to odraditi brzo i efektivno.

1.1 Definicija rada

Predmet istraživanja diplomskog rada je istraživanje i analiziranje botova. Na temelju provedenog istraživanja osmisliti će se jednostavan tekstualni bot te izraditi isti. Bot će olakšavati moderaciju jednog Discord servera te pružati par funkcija kojima je svrha zabava te pojednostavljenje nekih svakodnevnih procesa.

1.2 Svrha i ciljevi rada

Svrha rada je pokazati važnost botova u današnjem svijetu i pokazati koliko se naši svakodnevni „rituali“ mogu pojednostavniti i ubrzati koristeći botove. Također će se pokazati princip na kojem rade.

Cilj rada je izraditi jednostavan tekstualni bot te opisati njegove funkcije i pojasniti na kojem principu radi.

1.3 Metodologija rada

U radu će se primijeniti mnoštvo različitih programa i tehnologija.

DB Browser, s kojim će se provjeravati strukturu korištene baze podataka te podatke kojima je napunjena. Korištenjem ovog programa će se provjeravati je li bot radi kao što je predviđeno te će se olakšati bilo kakve pogreške koje će se pojaviti pri programiranju već spomenutog bota.

Virtual studio code, u kojem će se pisati sam kod i preko kojeg će se pokretati bot s terminala.

Discord, sama aplikacija za koju pišemo discord bot. U njoj će se provjeravati radi li sve kako treba.

1.4 Struktura rada

Rad se sastoji od 5 poglavlja koji čine zaokruženu cjelinu. U prvom poglavlju se uvodi čitatelja u područje rada. Navode se predmet, svrha i cilj rada. U drugom poglavlju je obrađen teorijski dio aplikacije Discord, dok je u trećem poglavlju objašnjen teorijski dio botova.

U četvrtom poglavlju je pregled svih tehnologija i programskog jezika korištenog za kreiranje Bot-a.

Peto poglavlje govori o prezentacijskom sloju bot-a, njegovoj izradi te opisuje korištenu bazu podataka.

Šesto poglavlje opisuje funkcionalnosti bot-a.

Na samom kraju rada je naveden zaključak cijelog djela.

2 DISCORD

Discord je besplatna aplikacija koja je dizajnirana za gaming zajednicu, međutim postala je popularna i među ostalim korisnicima jer se moguće pridružiti serveru u kojima se obično raspravlja o jednoj temi, te se ti serveri sastoje od ljudi kojima je barem ta tema zajednička.
Discord od 14 ožujka 2019. ima čak 250 milijuna korisnika.[1]
Aplikacija omogućuje tekstualnu, slikovnu, video te audio komunikaciju među korisnicima.

2.1 Kratka povijest

Koncept Discord-a je kreirao Jason Citron. On je skupa sa svojom ekipom tijekom produkcije igre došao na ideju pojednostavljenja i poboljšanja dotadašnjih programa za komunikaciju.
Discord je postao dostupan za preuzimanje u Svibnju 2015 godine.[2]

2.2 Problemi s Discord-om

Kao i sve društvene platforme, Discord također ima problema s cenzuriranjem i moderiranjem materijala koji korisnici objavljuju. Da bi rješili ovaj problem Discord ima tim koji radi bez prestanka i reagira na prijave. Ne kontroliraju poruke direktno nego imaju sistem koji pronalazi takve poruke. Tim sistemom upravljaju botovi.
Discord planira povećati tim kako se broj korisnika bude povećavao.

3 BOTOVI

Bot dolazi od riječi robot, oni su ustvari software koji radi automatizirane zadatke kao npr. programi za indeksiranje i pretraživanje weba, osobni pomoćnici i ostalo.
Međutim botovi mogu biti i maliciozni, takvi se botovi koriste za krađu informacija, spam, botnet napade.
Više od pola prometa na internetu se sastoji od botova.[3]
Botovi su primjer umjetne inteligencije, što znači da mogu „učiti“ i prilagođavati se prema podacima koji su im dani.

Prvu ideju za umjetnu inteligenciju je imao Alan Turing.
Također je izmislio provjeru koja se naziva Turingov test oko 1950. godine.
Taj test služi za testiranje umjetne inteligencije.
Test se provodi na način da sudac komunicira s drugim čovjekom i s robotom. Ako sudac ne može utvrditi razgovara li s čovjekom ili robotom, smatra se da je robot prošao test.
Sve do 2014. taj test nijedan robot nije prošao, kada ga je prošao robot po imenu „Eugene Goostman“. Radi se o robotu koji simulira 13-ogodisnje dijete i uspio je uvjeriti 33% sudaca da se radi o čovjeku.[4]

U 21. stoljeću dolazi do velikog napretka u polju umjetne inteligencije.[5]
Velike tvrtke i istraživački odjeli uočavaju korisnost bot-ova.
Bot-ovi se počinju koristiti u samovozećim automobilima, video igrama, te evoluiraju na način da razumiju čovjekov glas i koriste se kao pretraživači web-a.
Došlo je do toga da bot-ove koristimo na dnevnoj bazi bez da to i znamo.

3.1 Vrste botova

3.1.1 IM i IRC

IM(Instant messaging) i *IRC*(Internet Relay Chat) se još nazivaju chatbot-ovima i omogućuju ljudima da postavljaju pitanja u tekstualnom obliku na koja onda formuliraju odgovor.
Ovakvi bot-ovi su zaduženi za izvješća o sportskim rezultatima i vremenu te za konverziju valuta i slične zadatke. Također ih možemo naći u tekstualnim kanalima gdje se odvijaju razgovori, rade u pozadini i odgovaraju samo na određene fraze ili cenzuriraju psovke.

3.1.2 Socijalni botovi

Socijalni bot-ovi služe za uspostavljanje usluge ili veze između korisnika na webu.
Ovakvi bot-ovi mogu biti chat bot-ovi ili socijalni bot-ovi, kojima je zadatak imitiranje ljudskog ponašanja.

Korišteni su u izborima 2016 i 2017 u Britaniji.[6]
Programirani su na način da pretraživaju twitter za određene poruke i onda ih retweetaju.
Na ovaj način promoviraju jednu ili napadaju drugu stranu na izborima te utječu na glasove glasača.

3.1.3 Komercijalni botovi

Služe u automatiziranom trgovanju kao što su Apple App Store i Google Play.
Namjena im je ostavljanje lažnih recenzija koje su većinom pozitivne i na taj način zavaravaju korisnika o kvaliteti aplikacije. Međutim, recenzije mogu biti i negativne, te obično služe za napadanje konkurencije.
Također služe kao podrška koja je uvijek dostupna i zamjenjuje čovjeka u razgovoru s kupcima. Na ovaj način će se u budućnosti uštediti milijuni radnih sati te će zaposlenici imati više vremena za važnije zadatke.[7]

3.1.4 Zlonamjerni botovi

Bot-ovi se mogu koristiti za mnoštvo ilegalnih radnji, kao što su prevare, spam, promoviranje YouTube kanala, povećanje prometa na stranicama da bi se privukli ulagači te ostalo.
Bot-ovi se također koriste za varanje u video igrama, jer bot ima brže i bolje reakcije od igrača. U slučaju da napad izvršavaju više od jednog robota onda se tu radi o botnet-u. To je kolektivni skup bot-ova koji napad izvršavaju simultano s više računala i imaju puno veće mogućnosti pa ih je jako teško zaustaviti. Ovo se još zove DoS(Denial of Service) napad.

3.2 Obrana od zlonamjernih bot-ova

Najraširenija tehnika protiv bot-ova je CAPTCHA, oblik Turingovog testa koji služi za provjeru radi li se o čovjeku ili bot-u na način da je korisniku predstavljena grafički šifrirana slika i zadano mu je da prepiše što vidi.
Drugi način obrane od zlonamjernih bot-ova na serverima je uključivanje robots.txt fajla.
Robots.txt je standard koji sadrži pravila koji bot-ovi za web moraju poštivati te upute koje dijelove stranice ne smiju skenirati.
U slučaju da bot ne prati zadana pravila onda mu se zabrani pristup serveru.
Sam tekst fajl na serveru nije dovoljan za provođenje ovih pravila nego je potrebno imati i odgovarajući softver, inače bot neće imati nikakve restrikcije.

4 TEHNOLOGIJE KORIŠTENE U PROJEKTU

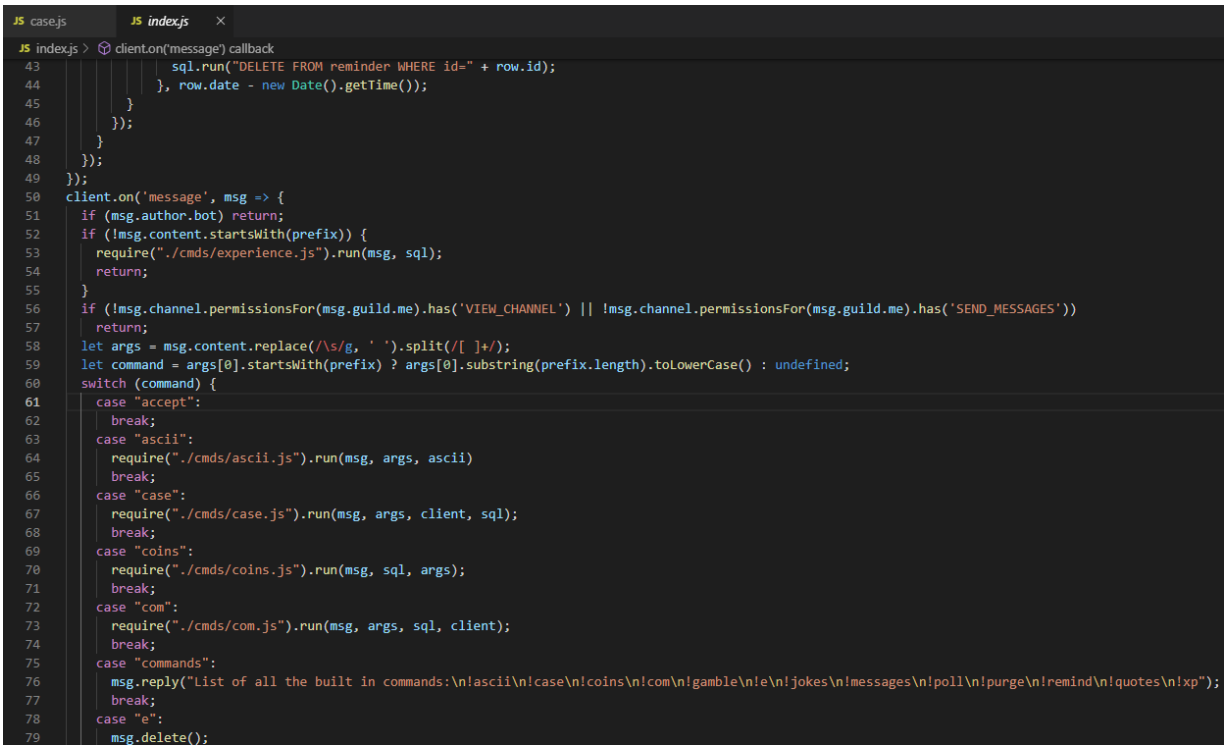
4.1 Javasript

JavaScript(JS) je objektno orijentiran jezik koji se najčešće koristi kao skriptni jezik za web stranice, ali može se koristiti i u okruženjima bez web preglednika.
Podržava stilove programiranja temeljene na događajima, funkcionalne i imperativne.

JavaScript se većinom pokreće na strani klijenta i koristi se za dizajniranje izgleda web stranice prilikom otvaranja te iste.

Osnovna sintaksa ovog jezika je slična Javi i C++ jezicima ali se ovi jezici razlikuju u dizajnu. JavaScript je pod utjecajem jezika Scheme i Self.

JavaScript je otvoren za mnoge sigurnosne napade jer se odvija na strani klijenta. Ovo se može spriječiti ispravnim dizajnom web aplikacija.
Primjer Javascript jezika vidimo na slici 1.

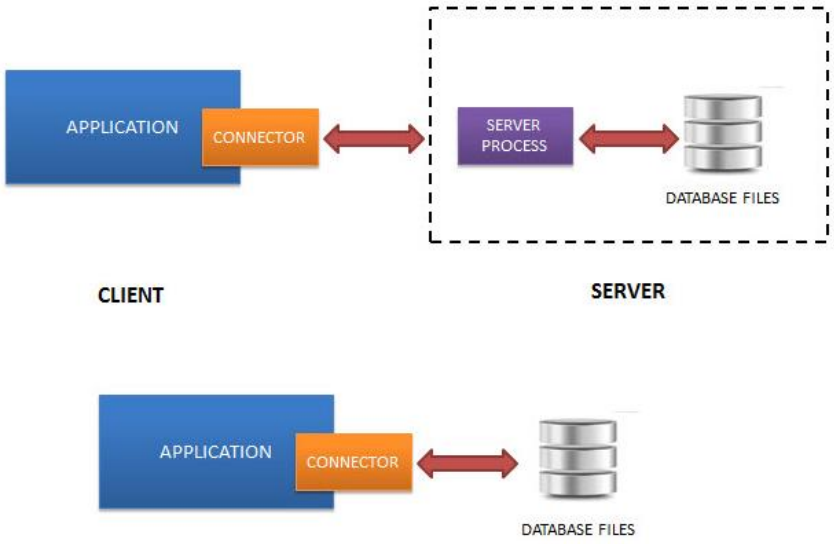


Slika 1- Primjer JavaScript jezika

4.2 SQLite

SQLite je relacijska baza podataka koja se temelji na C programskoj biblioteci.
Ona je najkorištenija baza podataka, koristi se više nego sve ostale baze kombinirano.[8]

Za razliku od MySQL-a, SQLite baza ne zahtjeva server da bi radila. On je integriran u aplikaciju koja pristupa bazi. Vizualizacija na slici 2.(SQLite dole, ostale baze gore).



Slika 2 - Razlika SQLite-a i ostalih baza [9]

S obzirom da ne SQLite ne koristi server, ne mora se instalirati prije korištenja, nema procesa koji se treba konfigurirati, započeti i prekinuti.[10]

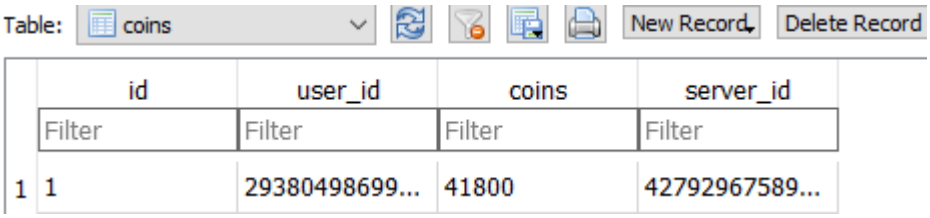
4.3 SQL i relacijska baza podataka

SQL (Structured Query Language) je programski jezik dizajniran za upravljanje podacima koji se nalaze u bazi podataka. Podatke možemo spremati, mijenjati, dohvaćati te brisati.
SQL upiti i ostale operacije se pišu kao izjave. Najkorištenije izjave su:
INSERT, SELECT, UPDATE, DELETE.

Da bi bot mogao raditi određene naredbe on mora na neki način spremiti podatke, koje onda može koristiti, mijenjati ili brisati.
Za spremanje tih podataka nam služi relacijska baza podataka.
U bazi podataka podaci su organizirani u tablicama koje sadrže stupce i retke. Svaka tablica ima barem jedan stupac koji je ujedno i jedinstveni ključ i svaki redak ima različiti ključ.
Stupci predstavljaju vrijednost atributa objekta, a retci predstavljaju objekt.
Primjer SQL-a koji ubacuje podatke u tablicu „coins“.

```
sql.run(`INSERT INTO
coins(user_id,coins,server_id)VALUES(${293804986991213},${41800},${427929675892
321})`)
```

Nakon ubacivanja će tablica izgledati kao na slici 3.



Slika 3- Primjer baze podataka

4.4 API

API je skup podprogramskih definicija, protokola i alata za izgradnju softvera.
To je skup jasno definiranih metoda komunikacije između komponenti.
Korištenjem API-ja se znatno olakšava razvoj računalnog programa tako što se pružaju građevni blokovi koje programer sastavlja.

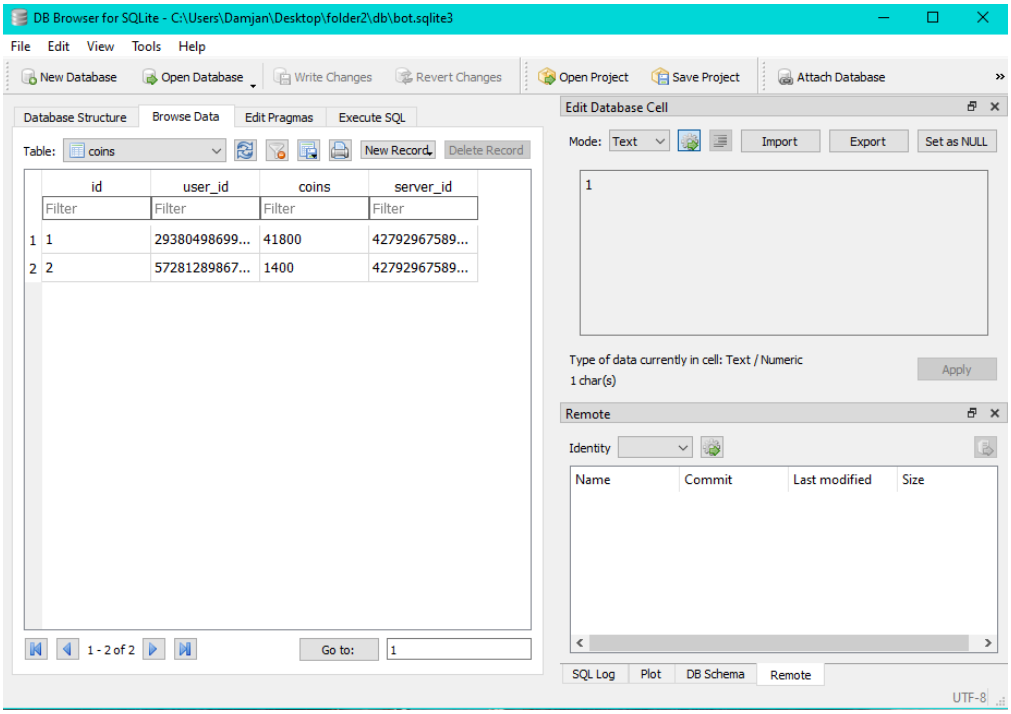
U ovom projektu su korištena dva različita API-ja.
API koji generira slučajan vic te API koji generira slučajne citate.
U sljedećem kodu vidimo API zahtjev u JavaScriptu

```
request('https://sv443.net/jokeapi/category/Programming?format=json', { json:
true }, (err, res, body) => {
                    if (err)
                        msg.reply(err.message)
                    if (body.type == "twopart")
                        msg.reply(body.setup + "\n" + body.delivery);
                    else
                        msg.reply(body.joke);
                });
```

U navedenom zahtjevu dobijamo podatke u formatu JSON i proslijeđujemo ih na Discord server.

4.5 DB Browser

DB Browser je open source alat za kreiranje, dizajniranje i editiranje fajlova kompatibilnih s SQLite-om.
U projektu je korišten za za provjeru stanja baze podataka prije i poslije pozivanja naredbe za Bot-a, na temelju usporedbe podataka se može saznati je li naredba radila kako treba.
Slika 4. prikazuje izgled DB Browser programa



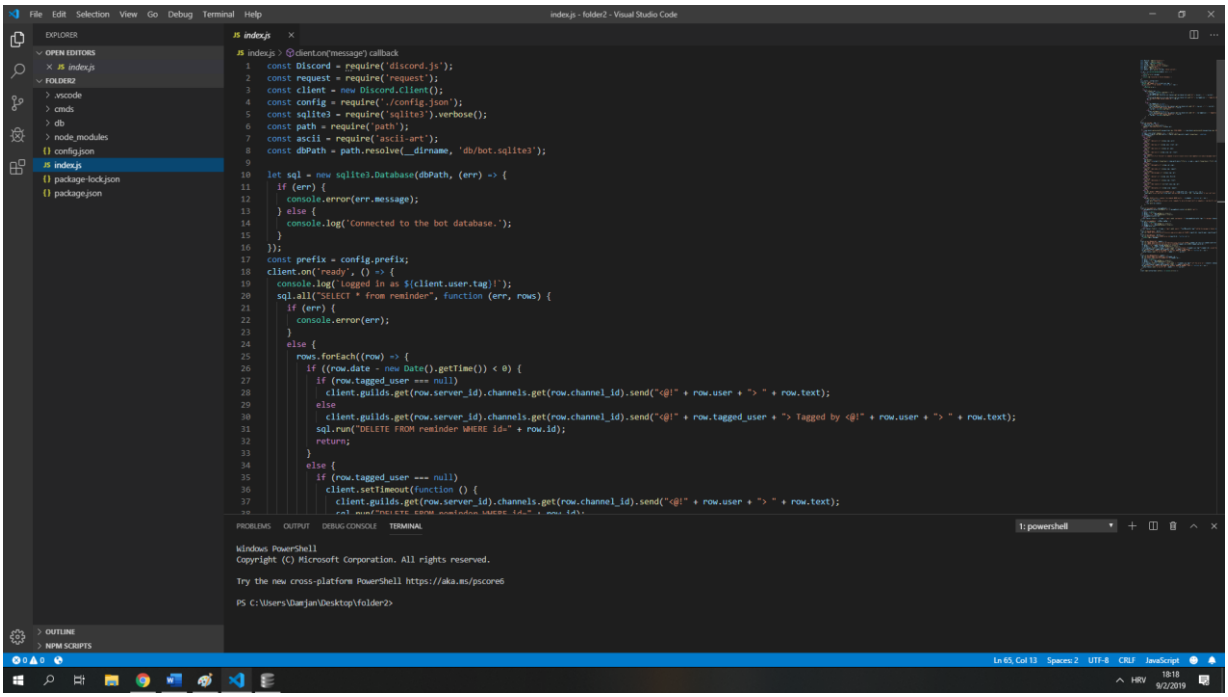
Slika 4- Izgled DB Browser programa

4.6 VS Code

VS Code je editor razvijen od strane Microsoft-a za Windows, Linux i macOS.
Baziran je na Electron-u, okvir koji se koristi za razvijanje Node.js aplikacija.

Vrlo je prilagodljiv. Korisnicima dopušta izbor teme, prečica za tipkovnicu te instaliranje ekstenzija koje dodaju nove funkcionalnosti. Izvorni kod je besplatan i otvoren.

Korišten je iz razloga što uključuje podršku debugiranja te ugrađen terminal što je bilo potrebno za ovaj projekt. Primjer izgleda programa možemo vidjeti na slici 5.



Slika 5- Izgled VS code programa

4.7 JSON

JSON je format razmjene podataka. Koristi se iz razloga što je ljudima jednostavan za čitanje i pisanje, a strojevima jednostavan za analiziranje i generiranje.
Temelji se na podskupu JavaScript jezika.

JSON se sastoji od dvije strukture.
Prva je kolekcija parova imena i vrijednosti, a druga je poredana lista vrijednosti.

Primjer JSON objekta

```
{
   "quoteText": "Imagination is the highest kite one can fly.",
   "quoteAuthor": "Lauren Bacall"
}
```

Primjer JSON objekta s poljem JSON objekata

```
{
"quotes":[
        {
          "quoteText": "Imagination is the highest kite one can fly.",
          "quoteAuthor": "Lauren Bacall"
        },
        {
          "quoteText": "He who lives in harmony with himself lives in harmony with the
        universe.",
          "quoteAuthor": "Marcus Aurelius"
        }
]
}
```

Iz primjera se može vidjeti da objekt započinje lijevom vitičastom zagradom i završava desnom vitičastom zagradom.
Između svakog imena i vrijednosti nalazi se dvotočka.
Parovi imena i vrijednosti su odvojene zarezom.
U slučaju polja sve je isto osim što polje počinje lijevom ulgatom zagradom i završava desnom uglatom zagradom.
Vrijednosti mogu biti string, broj, true, false, null, i drugi JSON objekt.

4.8 Node.js

Node.js je serverska JavaScript platforma koja sadrži minimalnu biblioteku.
Omogućuje nam izvršavanje JavaScript koda izvan preglednika.

4.9 Discord.js

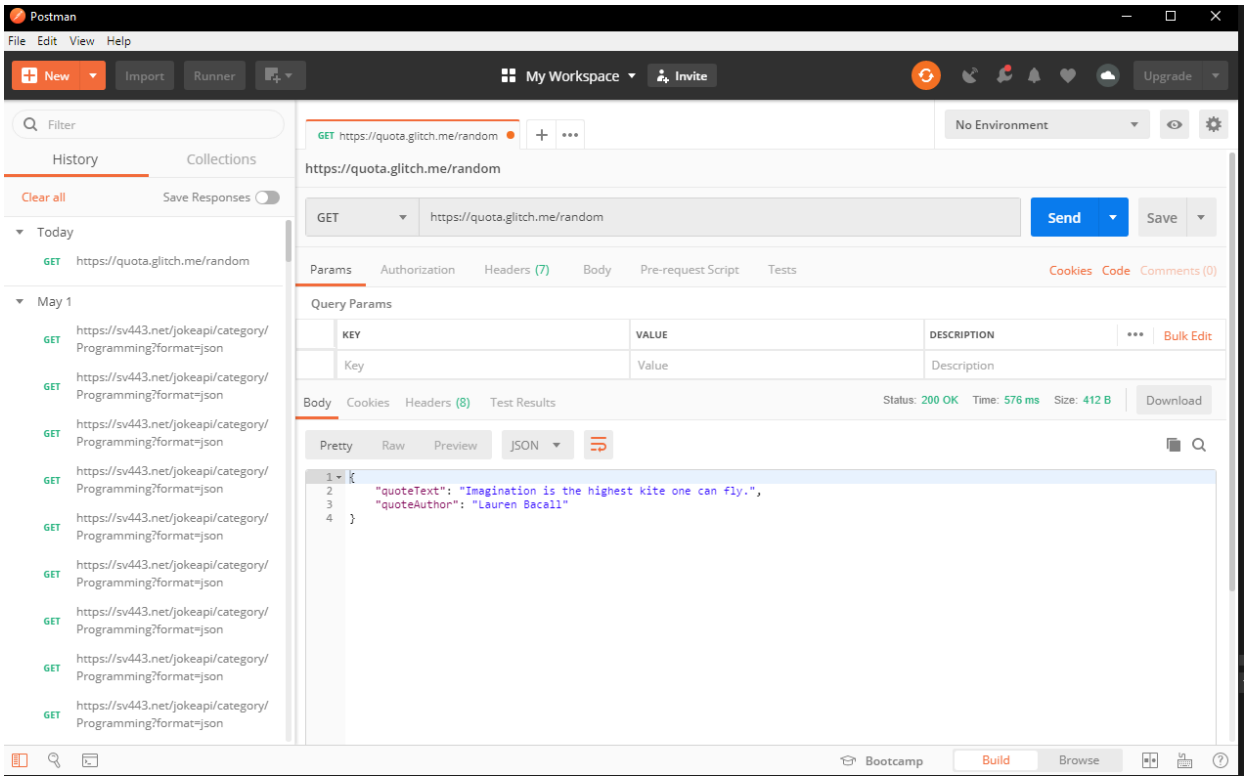
Node.js modul koji omogućuje interakciju s Discord API-jem.

```
const Discord = require('discord.js');
const client = new Discord.Client();
```

Dvije linije koje se moraju nadodati na početak index.js filea da bi se mogao koristiti discord.js.

4.10 Postman

Postman je jedan od najpopularnijih alata koji se koristi u testiranju API-ja. Izgled Postman alata možete vidjeti na sljedećoj slici 6.



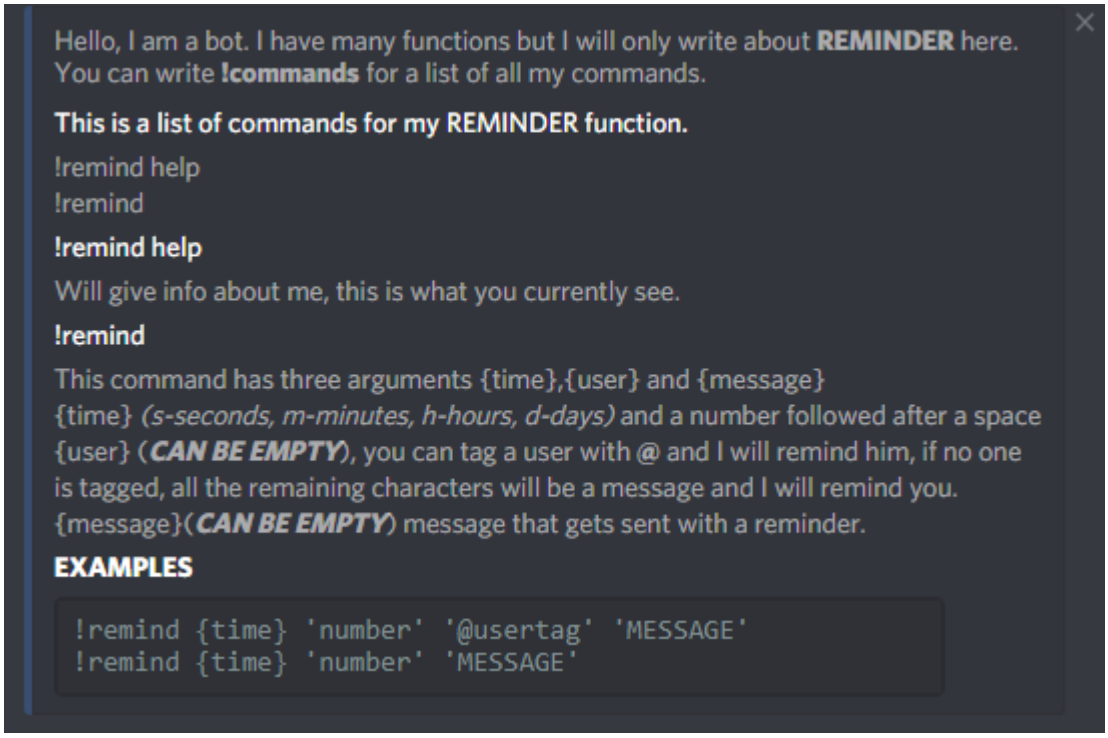
Slika 6- Izgled Postman programa

5. IZRADA BOT-a

U ovom poglavlju će se opisati na koji način smo koristili navedene tehnologije u prijašnjim poglavljima.

5.1 Prezentacijski sloj

S obzirom da bot radi unutar discord aplikacije potrebno je bilo samo isprogramirati njegove funkcije. Jedini dizajn kojim se projekt bavi je objašnjenje svake naredbe koje se postiže markup jezikom u embed formatu i vidimo ga na slici 7.
Markup se koristi tako da dodavanjem * na početak i kraj slova postanu kurzivna, dvostrukom * slova postanu bold, i slično.



Slika 7- embed format

Sljedeći kod prikazuje kako izgleda embed format i korištenje markup jezika.

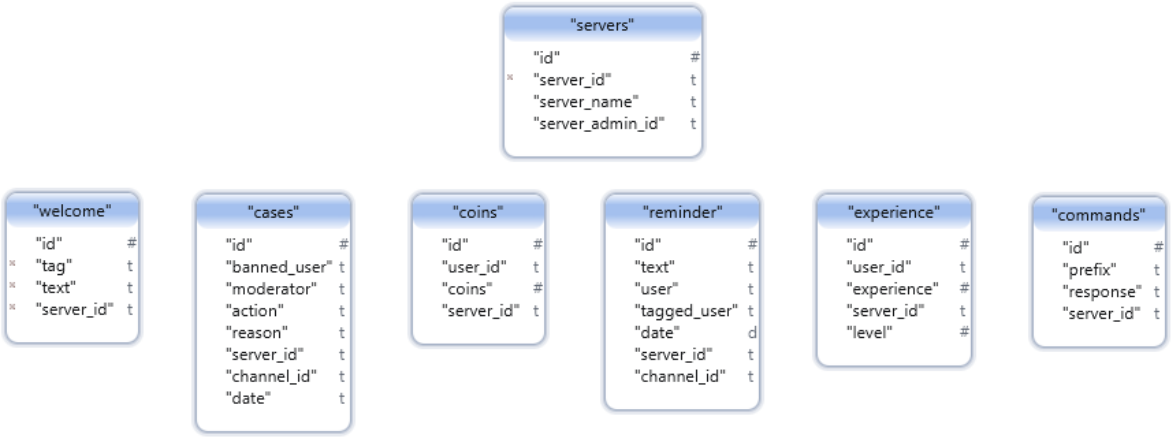
```
const embed = {
          "description": "Hello, I am a bot. I have many functions but I will
only write about **REMINDER** here. You can write **!commands** for a list of
all my commands.",
          "color": 3755632,
          "fields": [
            {
              "name": "This is a list of commands for my REMINDER function.",
              "value": "!remind help\n!remind"
            },
```

```
            {
              "name": "!remind help",
              "value": "Will give info about me, this is what you currently
see."
            },
            {
              "name": "!remind",
              "value": "This command has three arguments {time},{user} and
{message}\n{time} *(s-seconds, m-minutes, h-hours, d-days)* and a number
followed after a space\n{user} (***CAN BE EMPTY***), you can tag a user with
**@** and I will remind him, if no one is tagged, all the remaining characters
will be a message and I will remind you.\n{message}(***CAN BE EMPTY***) message
that gets sent with a reminder."
            },
            {
              "name": "**EXAMPLES**",
              "value": "```\n!remind {time} 'number' '@usertag'
'MESSAGE'\n!remind {time} 'number' 'MESSAGE'```"
            }
          ]
        };
```

5.2 Implementacija baze podataka

Za spremanje podataka korištena je SQLite 3 baza, a za pristup podacima DB Browser za SQLite.

Naziv baze podataka je bot, a baza se sastoji od 7 tablica.
Te tablice su: cases, coins, commands, experience, reminder, servers i welcome.
Sve tablice imaju jedinstveni ključ id, i strani ključ server_id kojim su povezane s tablicom servers. Korištenu bazu podataka možemo vidjeti na slici 8.



Slika 8- Korištena baza podataka

U tablicu **welcome** spremaju se sve poruke dobrodošlice/pozdrava koje admin izmisli.
Te poruke se pojave nakon što korisnik postane novi član ili izađe iz servera.
Sastoji se od atributa:

- *id* - jedinstvena brojčana oznaka koja se automatski generira i inkrementira(uveća za 1) za svaki novi unos. Taj atribut je ujedno i ključ tablice.
- *tag* - tekst koji označuje je li poruka dobrodošlice ili poruka pozdrava.
- *text* - poruka dobrodošlice/pozdrava. Atribut server_id je strani ključ i predstavlja id od servera na kojemu postoje ove poruke.

U tablicu **cases** spremaju se podaci o svim slučajevima gdje je došlo do zabrane ili izbacivanja člana na tom discord serveru.
Sastoji se od atributa:

- *id* - jedinstvena brojčana oznaka koja se automatski generira i inkrementira za svaki novi unos. Taj atribut je ujedno i ključ tablice.
- *banned_user* - id korisnika kojemu je zabranjeno pristupanje serveru.
- *moderator* - id korisnika koji mu je zabranio pristup serveru.
- *action* - akcija koja se uzela protiv člana, ili zabrana pristupa ili izbacivanje iz servera, u kojem slučaju je moguć ponovno pridruženje serveru.
- *reason* - razlog zašto je član izbačen ili zašto mu je zabranjen pristup serveru.

- *server_id* - strani ključ i predstavlja id od servera iz kojega je član izbačen ili mu je zabranjeno ponovno pridruživanje.
- *channel_id* - id kanala u kojemu se dogodio incident, tj. id kanala u kojem je izvedena naredba.
- *date* - datum kada se izvela naredba.

U tablicu **coins** spremaju se podaci o izmišljenoj valuti, tj. koliko koji član ima novčića.
Sastoji se od atributa:

- *id* - jedinstvena brojčana oznaka koja se automatski generira i inkrementira za svaki novi unos. Taj atribut je ujedno i ključ tablice.
- *user_id* - id korisnika koji ima određen broj novčića(moguće je da ih nema).
- *coins* - broj koliko novčića taj korisnik ima.
- *server_id* - strani ključ i predstavlja id od servera na kojem taj korisnik ima toliko novčića.

U tablicu **reminder** spremaju se podaci o podsjetniku.
Sastoji se od atributa:

- *id* - jedinstvena brojčana oznaka koja se automatski generira i inkrementira za svaki novi unos. Taj atribut je ujedno i ključ tablice.
- *text* - tekst koji će se prikazati kao poruka podsjetnika kada istekne zadano vrijeme.
- *user* - id korisnika koji je pozvao podsjetnik.
- *tagged_user* - id označenog korisnika kojega će se podsjetiti nakon isticanja vremena. Može biti isti kao i user, u tom slučaju korisnik je sam sebi ostavio podsjetnik.
- *date* - datum i vrijeme kada je pozvana naredba u milisekundama, tom broju je pridodano vrijeme koje treba proći prije podsjetnika.
- *server_id* - strani ključ i predstavlja id od servera na kojem je pozvan podsjetnik.
- *channel_id* - id kanala u kojem je pozvan podsjetnik, u tom istom kanalu će se i izvesti.

U tablicu **experience** spremaju se podaci o bodovima koje svaki korisnik na serveru ima.
Sastoji se od atributa:

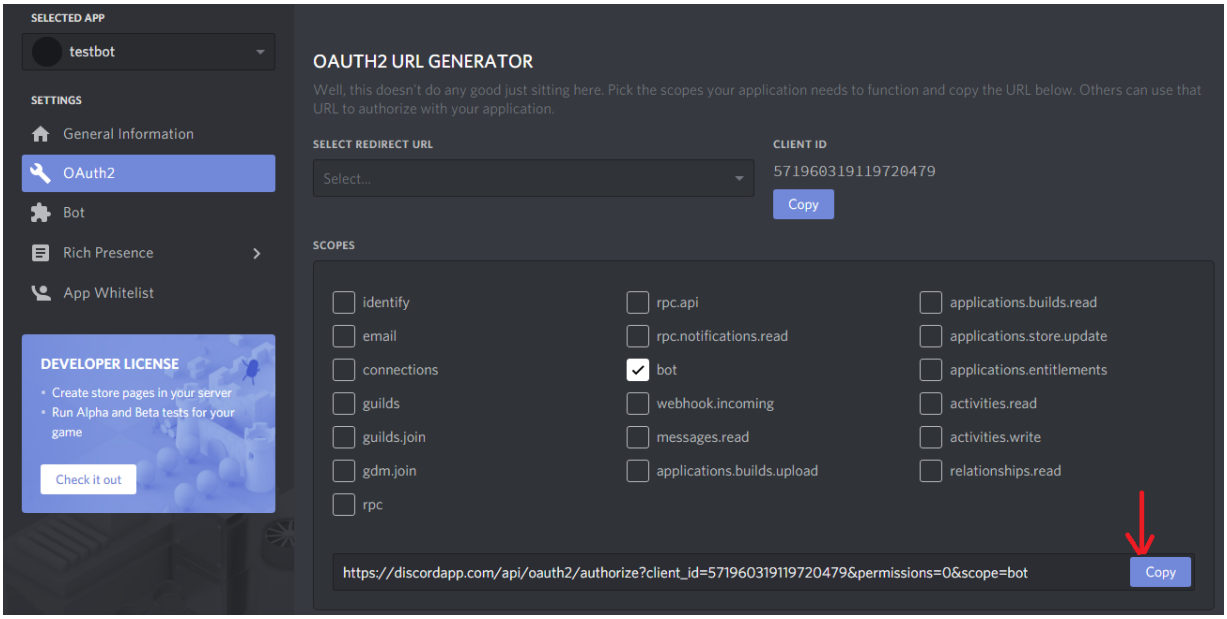
- *id* - jedinstvena brojčana oznaka koja se automatski generira i inkrementira za svaki novi unos. Taj atribut je ujedno i ključ tablice.
- *user_id* - id korisnika koji ima određen broj bodova.
- *experience* - označava koliko taj korisnik ima bodova.
- *server_id* - strani ključ i predstavlja id od servera na kojem taj korisnik ima toliko bodova.
- *level* - označava na kojoj je razini korisnik, nakon svake razine bodovi se vraćaju na 0, a potrebni bodovi za sljedeću razinu se povećavaju.

U tablicu **commands** spremaju se sve nove naredbe koje inače ne postoje, a pravi ih admin servera.
Sastoji se od atributa:

- *id* - jedinstvena brojčana oznaka koja se automatski generira i inkrementira za svaki novi unos. Taj atribut je ujedno i ključ tablice.
- *prefix* - označava kojim prefiksom počinje naredba, tj. što se treba napisati da bi došlo do odgovora bota.
- *response* - poruka koja se pošalje nakon određenog poziva.
- *server_id* - strani ključ i predstavlja id od servera na kojem postoje te naredbe.

5.3 Implementacija funkcionalnosti

Da bi se bot mogao koristiti sve što je potrebno je imati discord aplikaciju te imati svoj profil. Također bot mora biti pozvan na server u koji ste se pridružili, ovo obavlja admin servera. Potrebno je logirati se na discordapp.com/developers/applications te kopirati link koji vidimo na slici 9. i nakon toga ga otvoriti i dopustiti bot-u da se pridruži serveru.



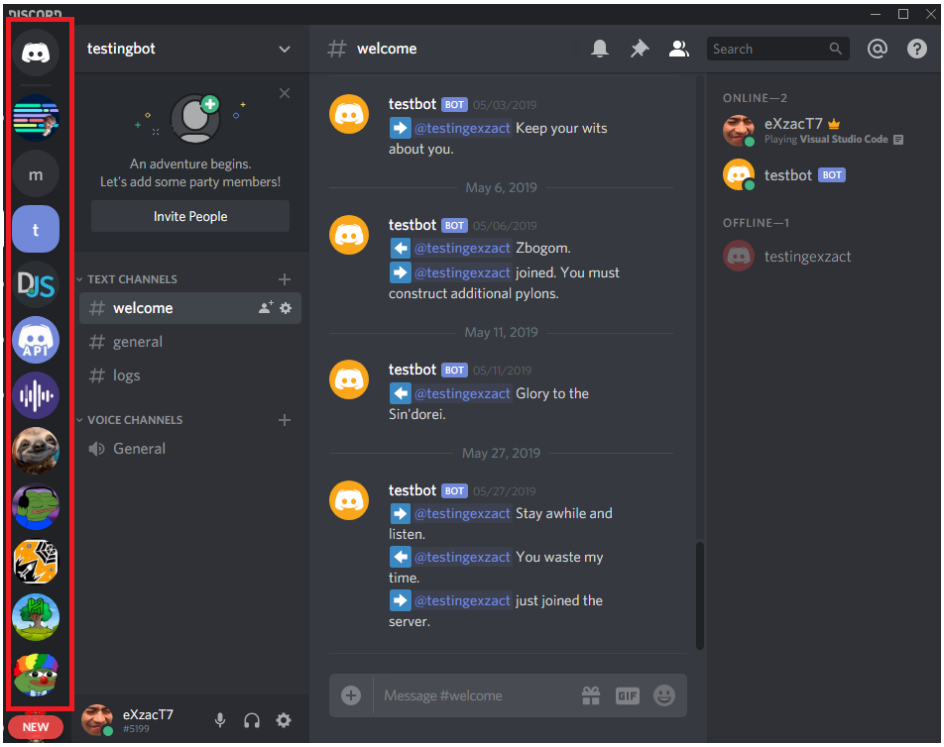
Slika 9- Upute dodavanja bot-a na server

6. OPIS BOT-a

U ovom poglavlju se ukratko opisuje korištenje discord-a, onoliko koliko je potrebno da bi korisnik bio u stanju iskoristiti funkcije bot-a.

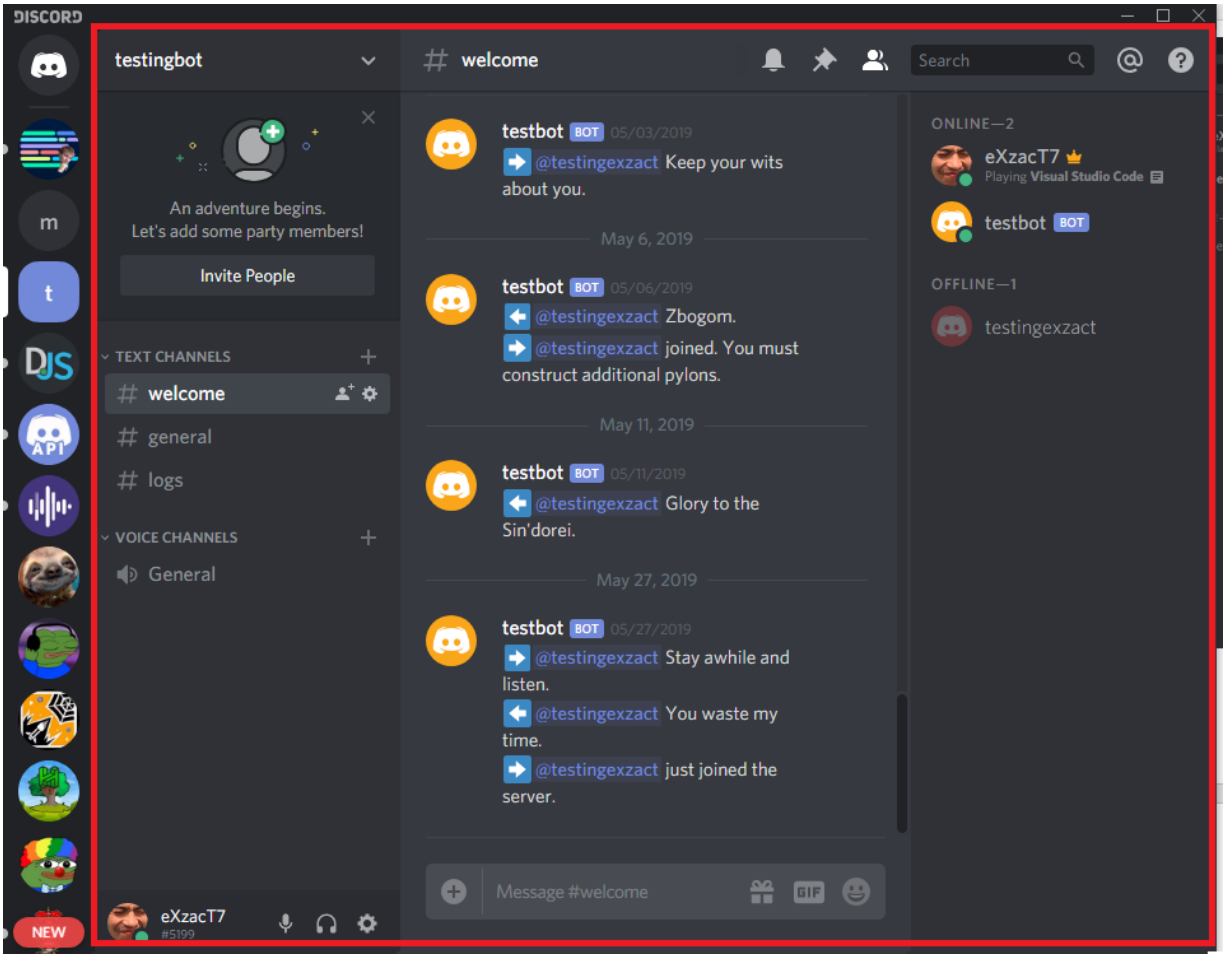
Zatim se detaljno opisuju sve funkcionalnosti bot-a.

6.1 Upute za Discord



Slika 10 - Upute korištenja discord-a(1)

Na slici 10 je označena traka svih servera kojima je korisnik pridružen. Svaki server ima svoju sličicu ili oznaku u slučaju da admin servera nije izabrao sliku. Pritiskom na kružić servera popunjavaju se polja u sredini programa.



Slika 11 - Upute korištenja discord-a(2)

U gornjem lijevom kutu slike 11 vidimo ime servera. Ispod toga je lista svih tekstualnih i glasovnih kanala u koje možemo pisati ili se pridružiti i razgovarati. Kanale dodaje vlasnik servera i drugi članovi kojima je to omogućeno preko raznih uloga.

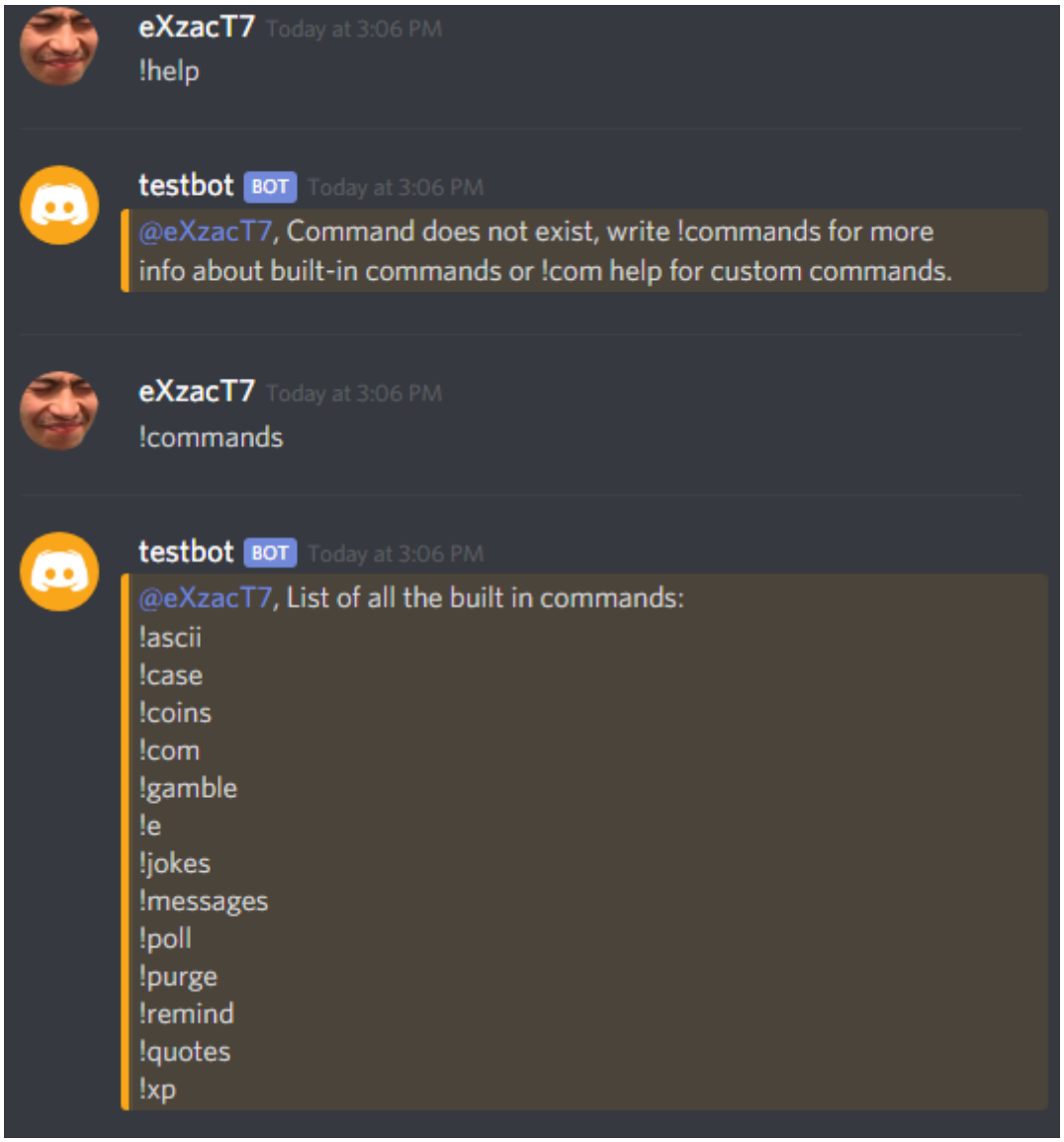
U sredini označenog pravokutnika vidimo sve poruke označenog kanala, u ovom slučaju „welcome“, ovo je izborni kanal i u njega se ispisuju poruke kad se netko pridruži ili izađe iz servera. Korisnicima je također omogućeno napisati i bilo kakvu drugu poruku, ali za to su namjenjeni drugi kanali. U osnovi bi admin trebao spriječiti pisanje u ovakve kanale.

S desne strane vidimo listu svih članova toga servera, uključujući i članove koji trenutno nisu prisutni.

Discord ima još puno funkcionalnosti ali njihovo objašnjenje i funkcije nisu potrebne za korištenje napisanog bot-a.

6.2 Funkcionalnosti bot-a

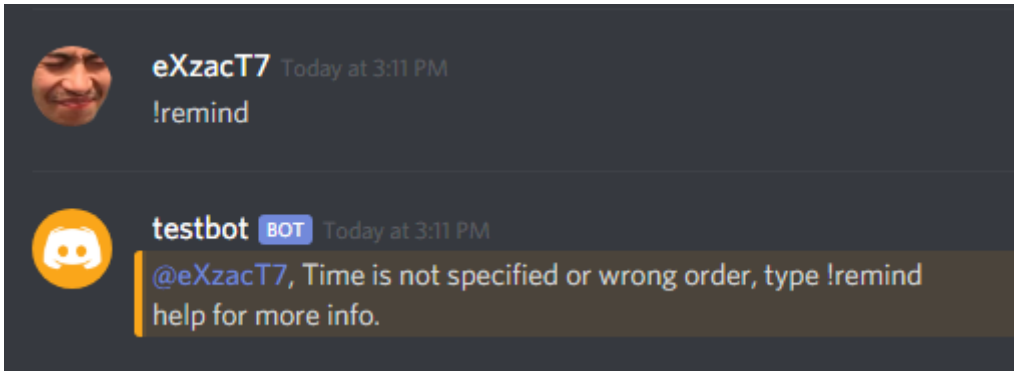
Bot čita sve poruke poslane u serveru kojem je pridružen ali odgovara samo na poruke koje počinju s uskličnikom.
U slučaju da korisnik napiše naredbu koja ne postoji, bot odgovara s porukom da ta naredba ne postoji i daje mu daljne upute kako se bot koristi te koje sve naredbe podržava.



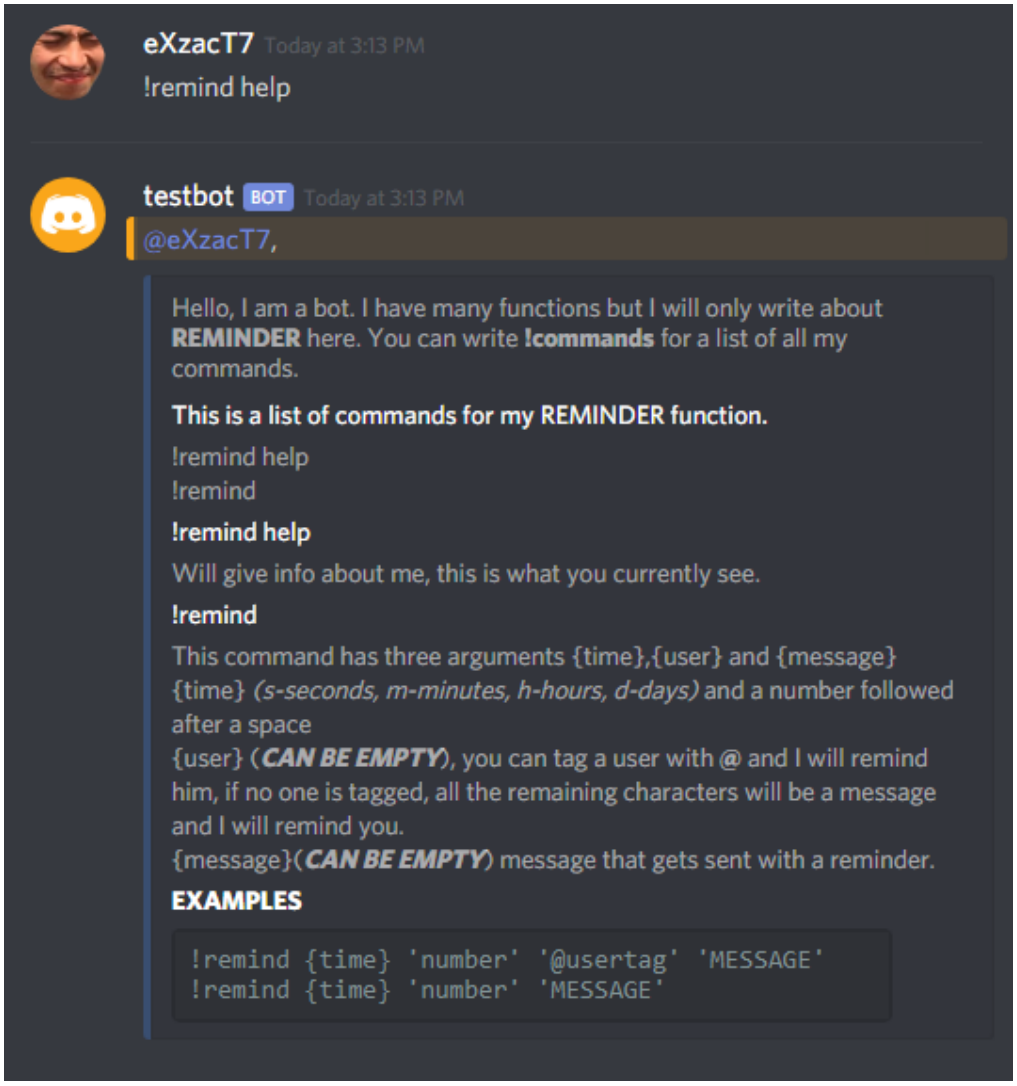
Slika 12- Primjer pozivanja naredbe te odgovor

Kao što se vidi na slici 12, bot ima 13 različitih već ugrađenih naredbi. O njima detaljnije u nastavku.

U slučaju da korisnik pozove postojeću naredbu ali je ne napiše u potpunosti točno, on odgovara gdje je korisnik napravio pogrešku te uvijek napominje da korisnik može u nastavku naredbe napisati „help“ za detaljan opis naredbe što se može vidjeti iz slike 13 i 14.

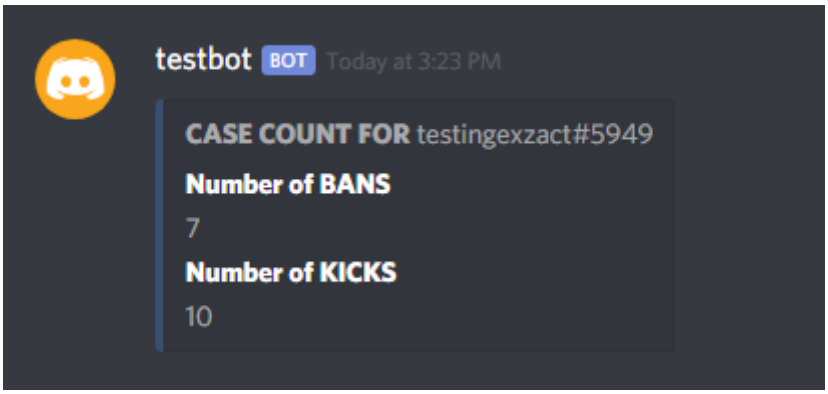


Slika 13 - Primjer krivog pozivanja naredbe



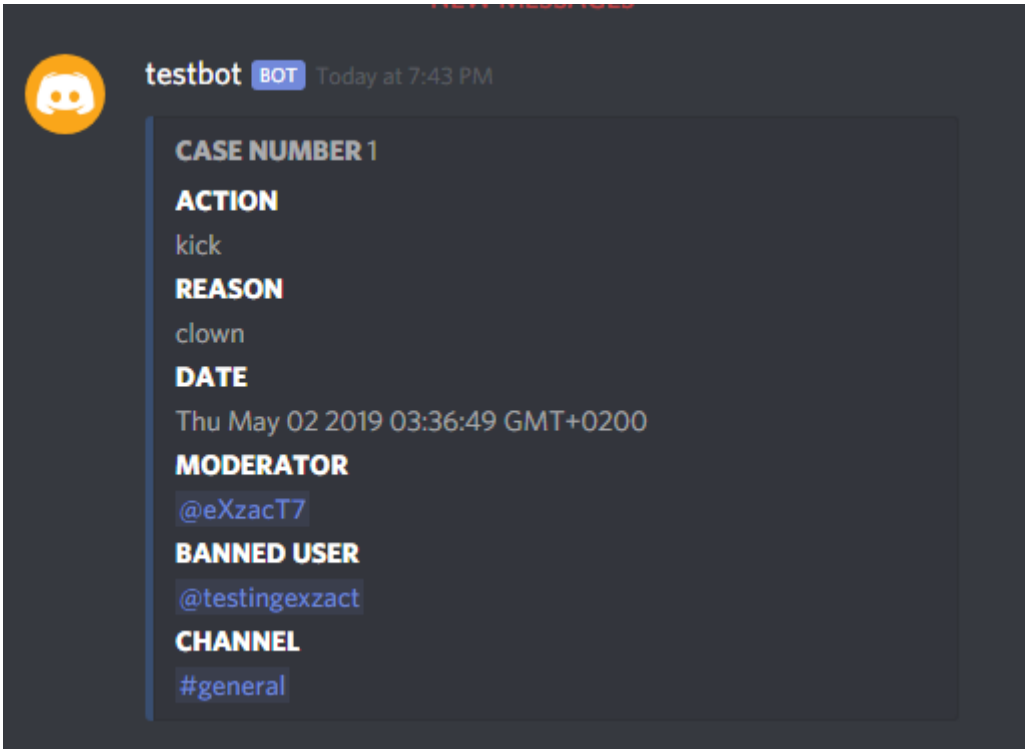
Slika 14- Primjer pozivanja univerzalne help naredbe

Naredba !case nam služi za izbacivanje ili zabranu pristupa članu servera. Nareba sve podatke o incidentu zapisuje u bazu podataka i kasnije joj pristupa pozivom !case check i označivanjem korisnika. Ova naredba nam vraća podatke koje vidimo na slici 15, a oni nam govore koliko je puta korisnik izbačen ili koliko mu je puta dana zabrana pristupa.



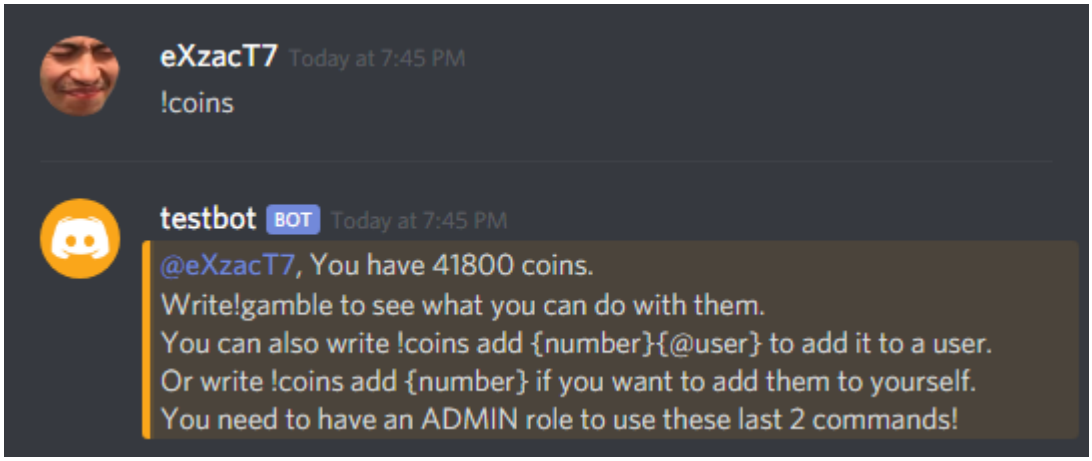
Slika 15 - Case naredba(1)

Također se mogu vidjeti detaljni podaci o zasebnom slučaju pozivanjem !case „case_id“ check, slika 16.



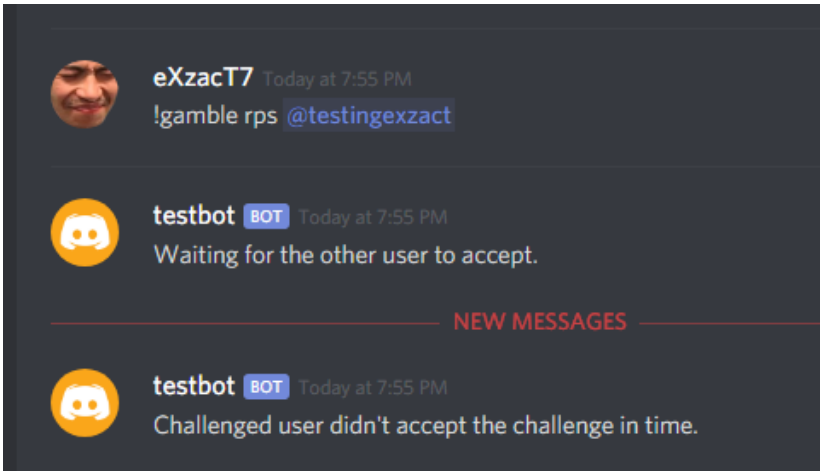
Slika 16 - Case naredba(2)

Naredba !coins služi za prikaz koliko korisnik koji je pozvao ima izmišljene valute.
Novčiće dobije svaki član kad se pridruži serveru.Također ih je moguće dobiti kada član dođe na sljedeću razinu, a to se postiže aktivnošću u serveru ili igrajući protiv drugih članova.
O tome više u nastavku.
U slučaju da je korisnik ujedno i admin onda može koristiti ovu naredbu za dodavanje valute sebi ili drugim članovima.
Evidencija o novčićima se vodi u bazi podataka.
Na slici 17 vidimo što se dogodi korištenjem naredbe.



Slika 17 - Coins naredba

Naredba !gamble sadrži 3 različita načina za korištenje novčića.
Prvi način je igra kamen-škare-papir. Moguće je igrati samo protiv drugog člana servera i suma za koju se igra je 100 novčića. Izazvani član mora prihvatiti izazov porukom !accept, u slučaju da ne napiše ništa izazov se otkazuje, ovo vidimo na slici 18.

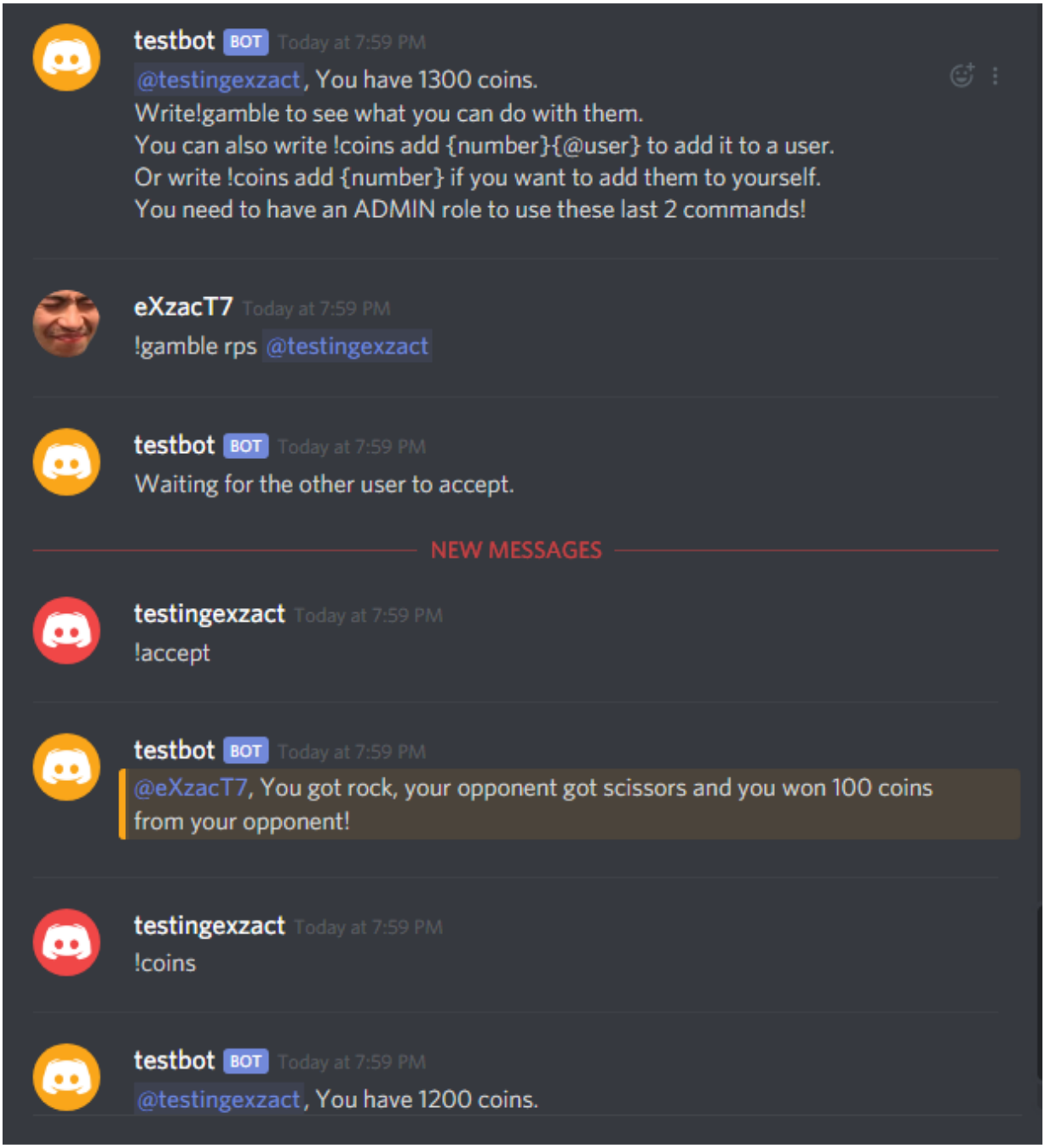


Slika 18 - Gamble naredba(1)

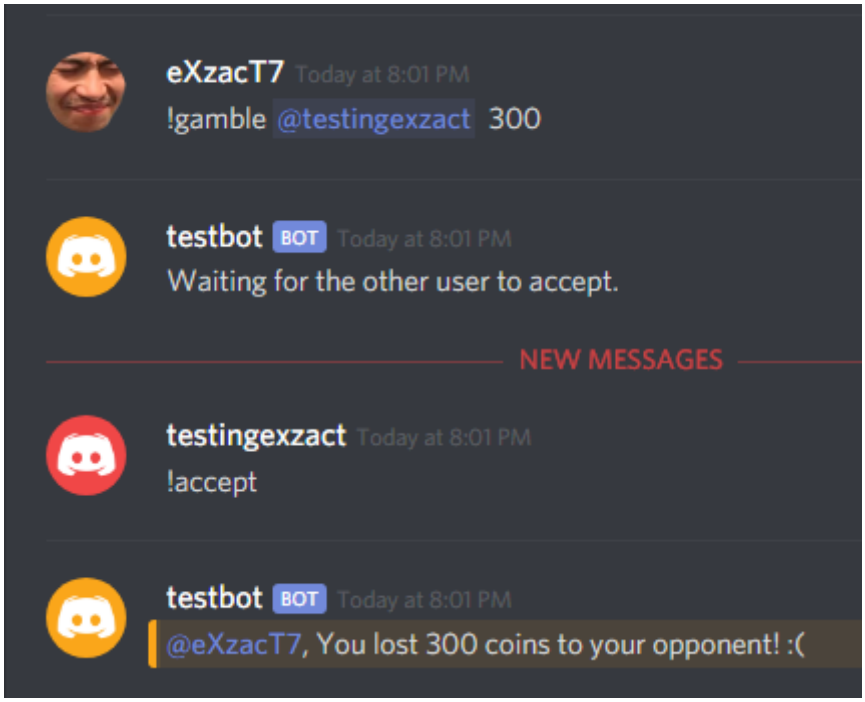
Druga igra je igra pogađanja. Igra se samo protiv kompjutera na način da korisnik napiše broj od 1 do 100, a kompjuter generira 2 broja, jedan kao svoj pokušaj, a drugi koji je ustvari broj koji se trebao pogoditi. Pobjednik je onaj tko je bio bliže generiranom broju.
U trećoj igri postoji 50% 50% šansa za pobjedu između člana koji je pozvao naredbu i člana koji je označen. Označeni član također mora prihvatiti izazov tako što napiše !accept, inače se igra

otkazuje. Igra se za napisani broj novčića, te u slučaju ako korisnik ili izazvani član nemaju toliko novčića javlja se poruka greške te se igra obustavlja.

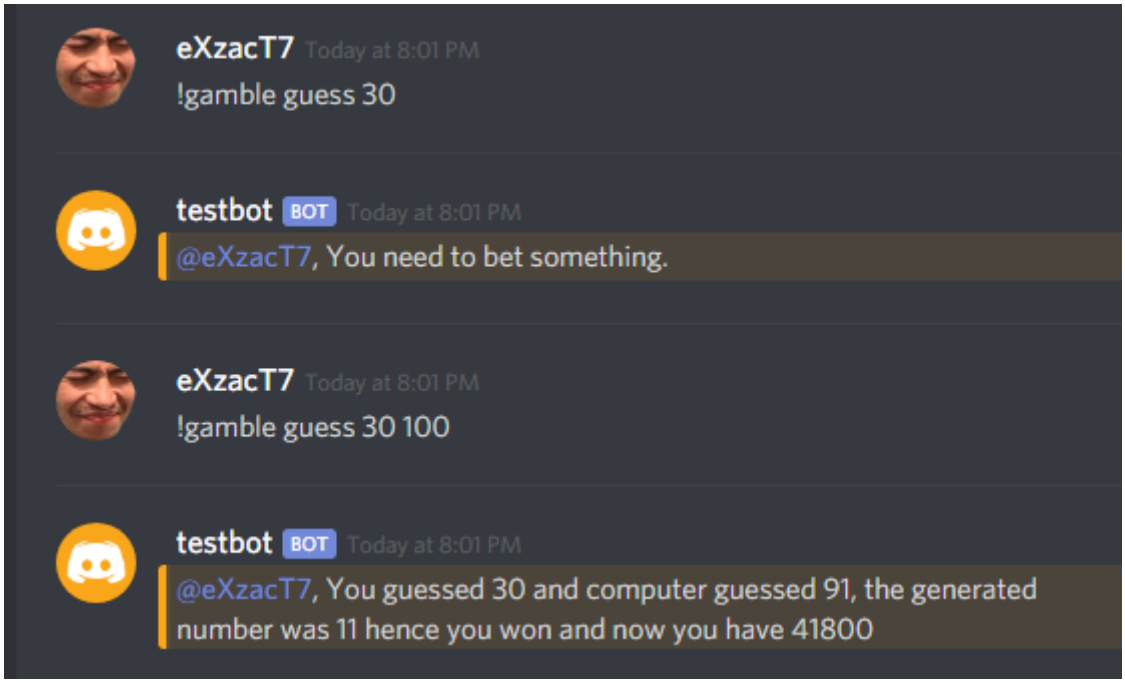
Na slici 19 i 20 i 21 vidimo primjere korištenja naredbe !gamble.



Slika 19 - Gamble naredba(2)

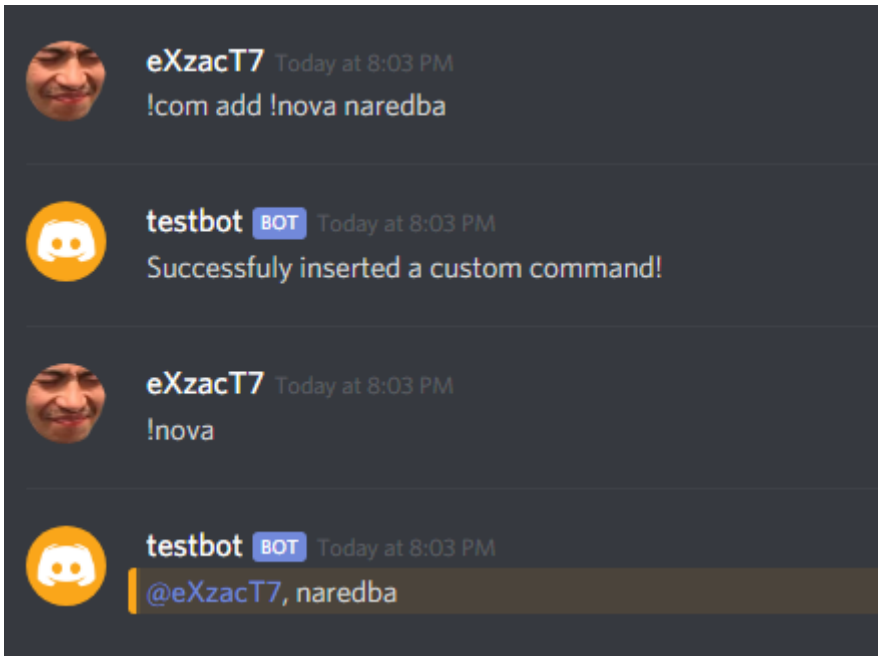


Slika 20- Gamble naredba(3)



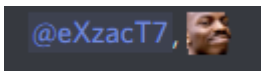
Slika 21 - Gamble naredba(4)

Naredba !com služi za dodavanje novih jednostavnih naredbi, tj. dodavanje novih riječi na koje bot reagira i odgovara s zadanom porukom. Također je moguće vidjeti listu naredbi te brisati naredbe. Sve nove naredbe se zapisuju u bazu podataka. Primjer korištenja naredbe možemo vidjeti na slici 22.



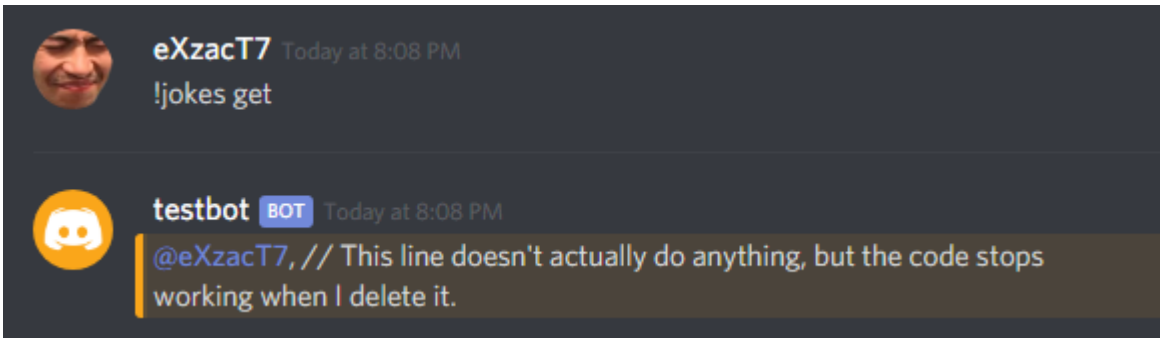
Slika 22- Com naredba

S obzirom da botovi imaju discord nitro, oni mogu koristiti GIF reakcije.
Sljedeća naredba !e iskorištava to na način da korisnik pozove naredbu !e te u nastavku izabere GIF. Primjer na slici 23.



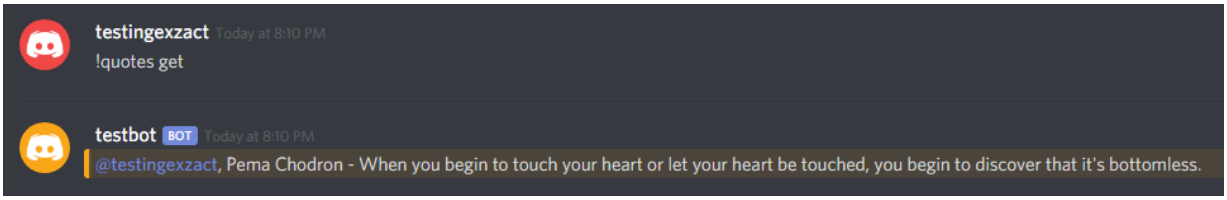
Slika 23- emote naredba

Naredba !jokes dohvaća slučajan vic s API-ja. Primjer na slici 24.



Slika 24- Jokes naredba

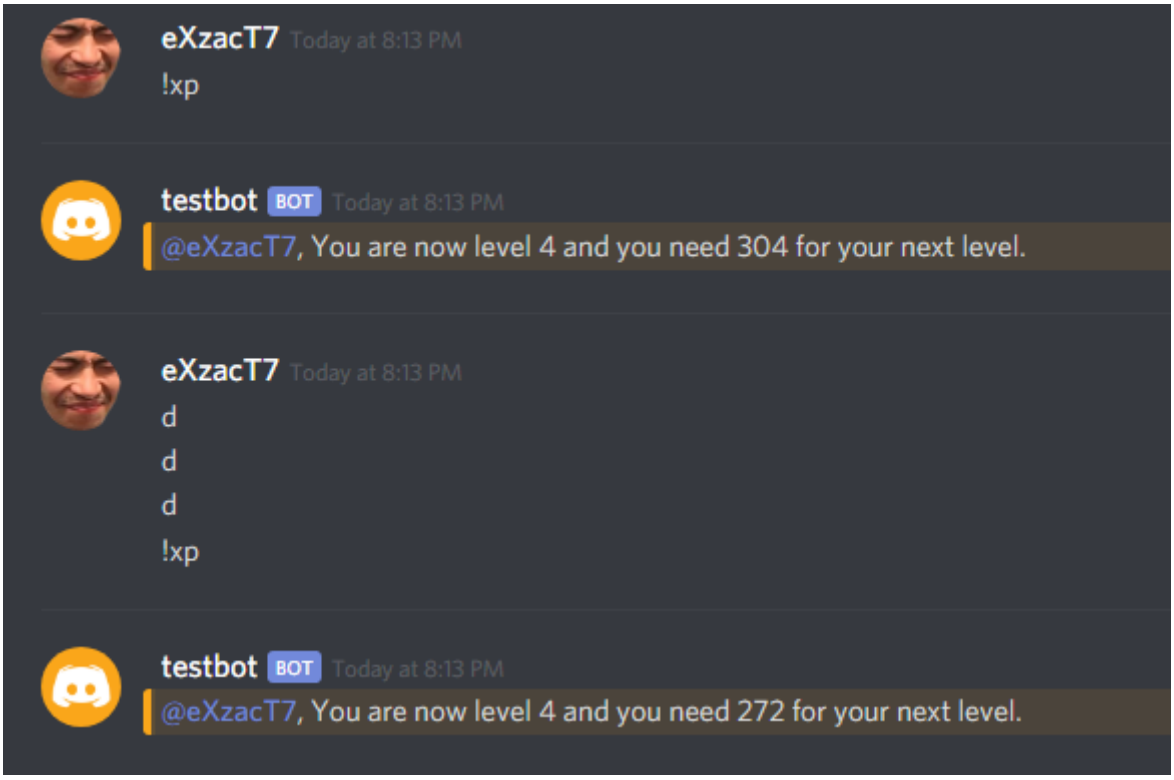
Naredba !quotes radi na isti način. Primjer na slici 25.



Slika 25- Quotes naredba

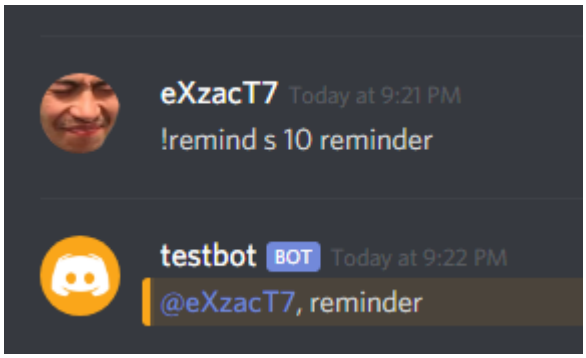
Naredba !purge služi za brisanje X prijašnjih poruka u kanalu kojem je pozvana. Ovu naredbu može koristiti samo admin.

Naredba !xp prikazuje trenutne bodove korisnika, njegovu razinu, te potrebne bodove da bi korisnik prešao na sljedeću razinu. Kao što je već prije spomenuto u objašnjenju naredbe !coins, korisnici dobijaju bodove sukladno s njihovom aktivnošću i nikako drugačije. Nakon prijelaza na veću razinu korisnik dobija novčiće. Što je veća razina to se dobije više novčića. Podaci se zapisuju u bazu podataka. Korištenje naredbe vidimo na slici 26.



Slika 26- Xp naredba

Naredba !remind služi za postavljanje podsjetnika. Korisnik može podsjetiti sebe ili nekog drugog člana. Uz podsjetnik se može ostaviti i poruka. Za ovu naredbu sam koristio bazu podataka jer u slučaju pada servera bot gubi sve zapisane podsjetnike. Na slici 27 možemo vidjeti primjer podsjetnika nakon 10 sekundi s porukom „reminder“.



Slika 27- Reminder naredba

Kod radi na principu da pod datum ubaci datum kad je pozvana naredba i nadoda vrijeme koje je korisnik zadao.

```
sql.run("INSERT INTO reminder(text,user,tagged_user,date,server_id,channel_id)
VALUES (?,?,?,?,?,?)", [text, msg.member.id, null, msg.createdTimestamp + retur
ntime, msg.guild.id, msg.channel.id]
```

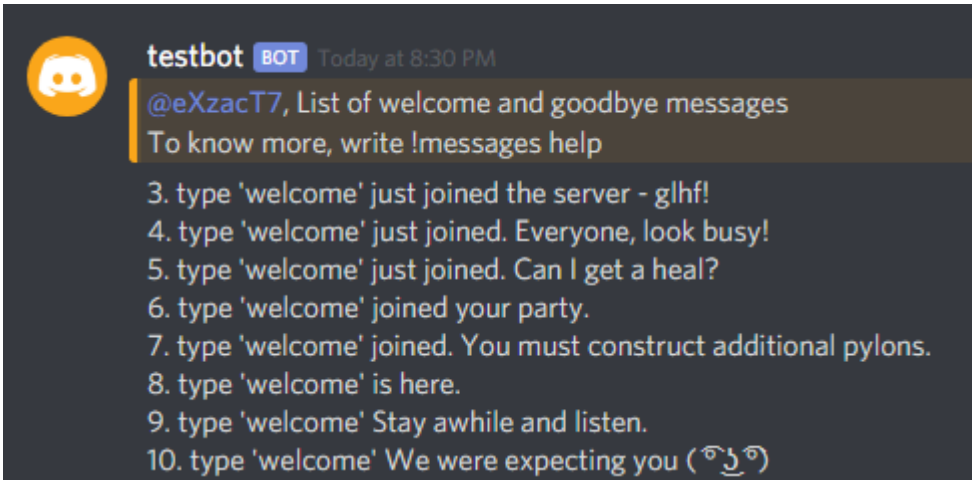
Navedena naredba uopće nije potrebna ako ne dođe do pada servera, međutim u slučaju da dođe do pada servera i onda ponovnog paljenja bot-a, postoje dva slučaja.
Bot provjerava je li zadano vrijeme prošlo i u prvom slučaju, ako nije, podsjetnik će se postaviti na točno vrijeme, a to vrijeme je zapisano vrijeme u bazi(već spomenuto poviše) i oduzme trenutno vrijeme.

```
 client.setTimeout(function () {
              client.guilds.get(row.server_id).channels.get(row.channel_id).sen
d("<@!" + row.user + "> " + row.text);
              sql.run("DELETE FROM reminder WHERE id=" + row.id);
            }, row.date - new Date().getTime());
```

Drugi slučaj je da je zadano vrijeme prošlo i u tom slučaju će se poslati podsjetnik te će se izbrisati svi podaci za taj podsjetnik iz baze.

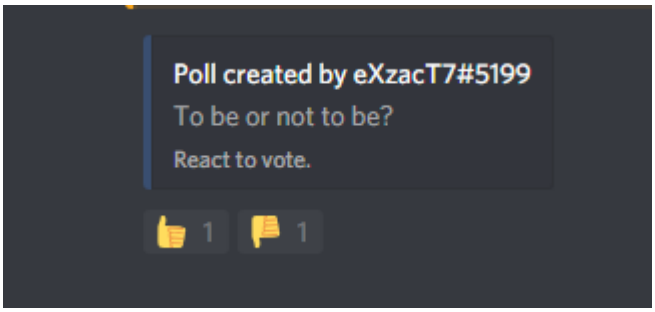
```
  if (row.tagged_user === null)
            client.guilds.get(row.server_id).channels.get(row.channel_id).send(
"<@!" + row.user + "> " + row.text);
          else
            client.guilds.get(row.server_id).channels.get(row.channel_id).send(
"<@!" + row.tagged_user + "> Tagged by <@!" + row.user + "> " + row.text);
          sql.run("DELETE FROM reminder WHERE id=" + row.id);
```

Naredba !messages radi na sličnom principu kao i !com. Korisnik može dodavati i brisati prilagođene poruke dobrodošlice i pozdrava koje se pojavljuju pri svakom pridruženju te odlasku člana iz servera. Također se može vidjeti lista već postojećih poruka skupa s njihovim ID-em radi lakšeg brisanja. Podaci se zapisuju u bazu podataka. Primjer se vidi na slici 28.



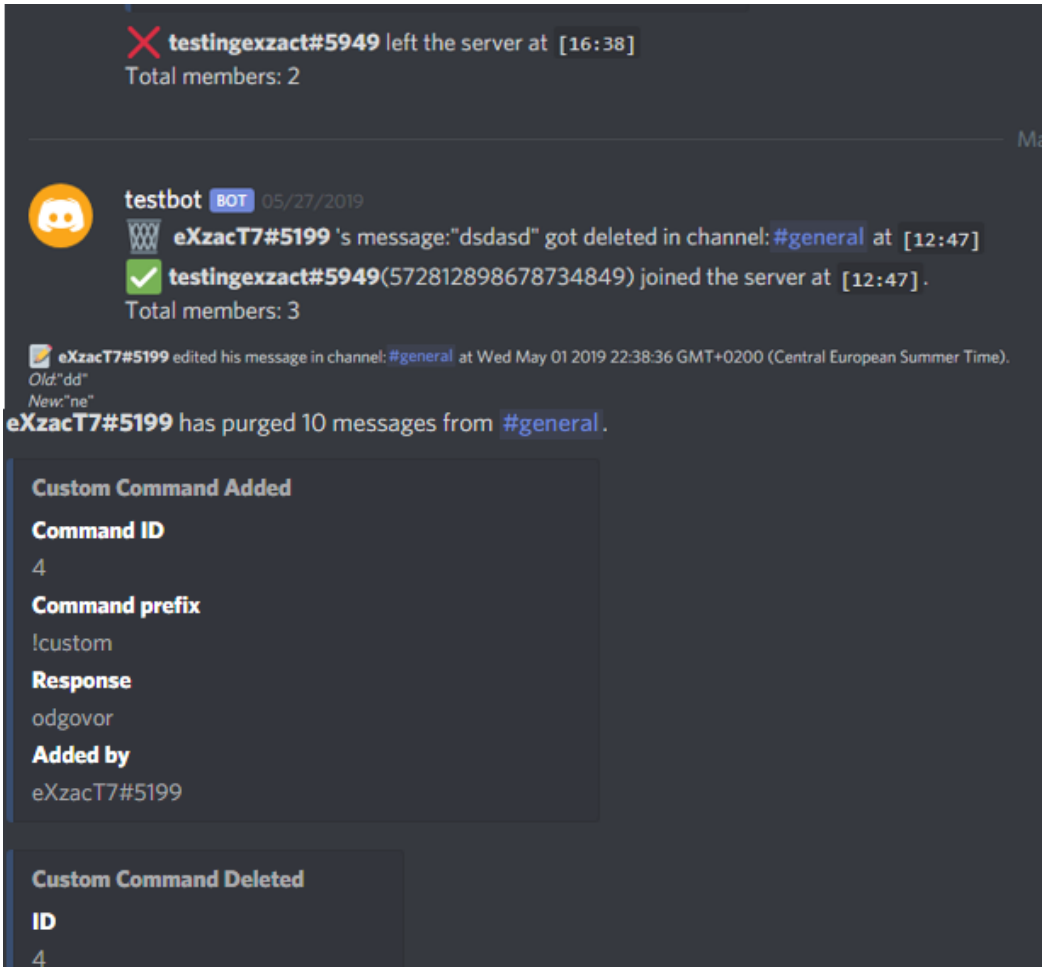
Slika 28- Messages naredba

Zadnja naredba je !poll, ona služi za lakše glasovanje i dogovaranje članova na serveru.
Sve što je potrebno napisati je !poll „pitanje“ i generira se sljedeći prozorčić s dva ponuđena odgovora, palac gore i palac dole, koji u prijevodu znače da ili ne. Ovo se može vidjeti na slici 29.



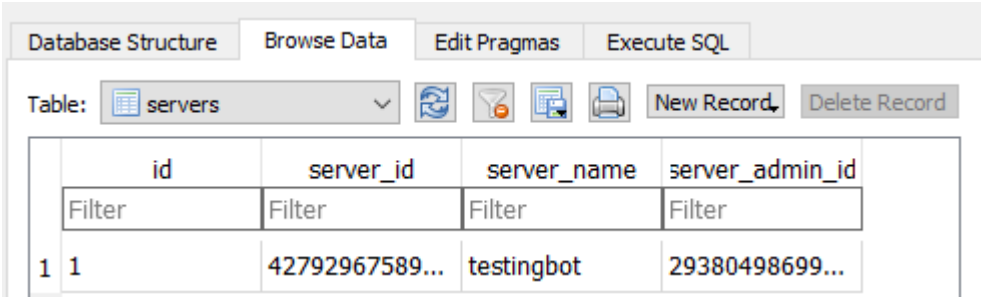
Slika 29- Poll naredba

Osim navedenih naredbi bot također vodi evidenciju o različitim događanjima na serveru te ih zapisuje u kanal „logs“.
Zapisuje pridruženja serveru, izlaske iz servera, promijenjene poruke, izbrisane poruke, zapisnik o prekršaju(ban/kick), dodavanje novih poruka dobrodošlice/pozdrava, dodavanje novih naredbi, korištenje naredbe !purge. Primjer zapisnika se vidi na slici 30.



Slika 30- Zapisnik

Bot također vodi evidenciju o serverima na kojima je prisutan. U bazu podataka zapisuje id servera, id admina te ime servera. Sve ostale tablice su povezane s ovom tablicom stranim ključem server_id, na ovaj način se osigurava da sve naredbe rade kako su zamišljene i da se ne može pristupati podacima drugih servera. Na slici 31. se vide zapisani podaci samo jednog servera jer se bot koristi samo na jednom serveru.



Slika 31- Tablica s podacima o serverima

7. ZAKLJUČAK

Prije početka projekta bilo je potrebno upoznavanje s javascript jezikom, proučavanje discord.js dokumentacije te izabiranje baze podataka koja se koristi u projektu.

S obzirom da bot radi u discord aplikaciji, u projektu nije bila potrebna faza dizajniranja nego se rad samo bavi funkcionalnostima bot-a. Također je bilo potrebno istražiti o već ponuđenim bot-ovima da bi se lakše razumio način na koji oni rade.
Baza podataka se postupno izmjenjivala i proširivala kako su se dodavale nove naredbe.
Bot je pisan u JavaScript jeziku te se za dohvat i ažuriranje baze podataka koristi isti. Da bi bot mogao raditi 24 sata svaki dan potrebno ga je staviti na hosting, ali za potrebe projekta korišten je node.js i bot radi sve dok se program ne izgasi.

Bot, koji je finalni rezultat ovog završnog rada, služi za olakšavanje moderacije discord servera te ga članovi mogu koristiti i u slobodno vrijeme za svoje potrebe. Ima mnoštvo različitih naredbi koje se mogu pozivati, međutim samo admin može koristiti naredbe koje služe za moderaciju. Članovima je to onemogućeno osim ako imaju određenu ulogu, a tu ulogu dodjeljuje admin. Ostale naredbe služe za korištenje u slobodno vrijeme i nemaju veze s moderiranjem.
Ovaj bot može služiti kao dobra baza za daljnje unaprijeđivanje s obzirom da je vrlo lako dodavati nove funkcije bez mijenjanja prijašnjih. Sve što je potrebno je dodavanje nove tablice u bazu podataka u slučaju da funkcija to zahtjeva.

8. LITERATURA

[1] www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/
[2] www.pcgamer.com/one-year-after-its-launch-discord-is-the-best-voip-service-available/
[3] https://www.imperva.com/blog/bot-traffic-report-2016/
[4] https://www.theguardian.com/technology/2014/jun/08/super-computer-simulates-13-year-old-boy-passes-turing-test
[5] https://chatbotslife.com/bots-a-tech-trend-for-2018-f47365324b69
[6] https://spectrum.ieee.org/computing/software/how-political-campaigns-weaponize-social-media-bots
[7] https://www.forbes.com/sites/mnewlands/2017/12/22/these-chatbot-usage-metrics-will-change-your-customer-service-strategy/#382ca6cf364a
[8] https://www.sqlite.org/mostdeployed.html
[9] https://www.sqlitetutorial.net/wp-content/uploads/2015/12/RDBMS-Client-Server-Architecture.jpg
[10] https://www.sqlitetutorial.net/what-is-sqlite

IZJAVA

Izjavljujem pod punom moralnom odgovornošću da sam završni rad izradio samostalno, isključivo znanjem stečenim na studijima Sveučilišta u Dubrovniku, služeći se navedenim izvorima podataka i uz stručno vodstvo mentora dr.sc.Maria Miličevića, kome se još jednom srdačno zahvaljujem.

Damjan Anđelković