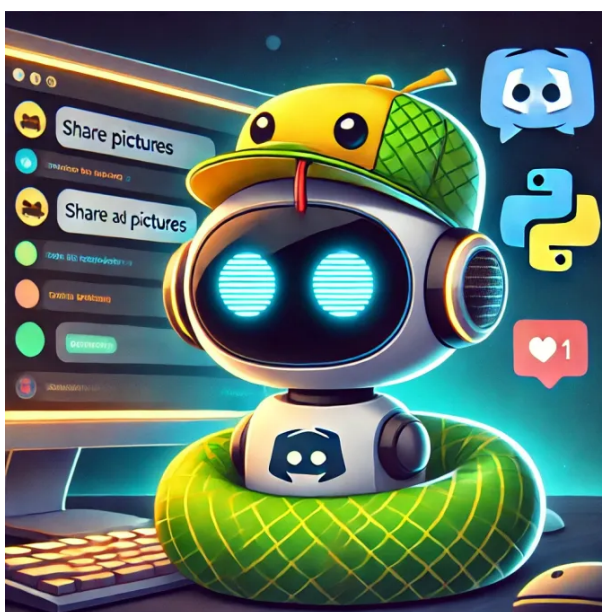


# ZÁVĚREČNÁ STUDIJNÍ PRÁCE

## dokumentace

### Discord Bot - Multifunkční bot pro správu Discord serveru



**Autor:** Jiří Rojek  
**Obor:** 18-20-M/01 INFORMAČNÍ TECHNOLOGIE  
se zaměřením na počítačové sítě a programování  
**Třída:** IT4  
**Školní rok:** 2024/25

## **Poděkování**

Děkuji panu učiteli Mgr. Markovi Lucnemu za rady při vytváření tohoto projektu.

## **Prohlášení**

Prohlašuji, že jsem závěrečnou práci vypracoval samostatně a uvedl veškeré použité informační zdroje.

Souhlasím, aby tato studijní práce byla použita k výukovým a prezentačním účelům na Střední průmyslové a umělecké škole v Opavě, Praskova 399/8.

V Opavě 31. 12. 2024

.....  
Podpis autora

# Obsah

<b>Úvod</b>	<b>6</b>
<b>1 Analýza a návrh</b>	<b>7</b>
1.1 Analýza požadavků	7
1.1.1 Funkční požadavky	7
1.2 Struktura projektu	7
1.3 Vysvětlení pojmů	8
1.3.1 Bot	8
1.3.2 Cog	8
1.4 Použité technologie	8
1.4.1 Discord API	8
1.4.2 Discord.py	8
1.4.3 Knihovna requests	9
1.4.4 Knihovna yt-dlp	9
1.4.5 FFmpeg	9
1.4.6 Knihovna asyncpraw	9
1.4.7 Knihovna Pillow	9
1.4.8 Knihovna Textwrap	9
1.4.9 Knihovna io	9
<b>2 Použité postupy a řešení</b>	<b>10</b>
2.1 Settings.py	10
2.1.1 Import tajných klíčů	10
2.1.2 Výchozí hodnoty	10
2.1.3 Platné možnosti řazení	10
2.1.4 Cesta k FFmpeg	10
2.1.5 Klíčová slova pro náhodné přehrávání	11
2.2 Secret.py	11
2.2.1 Bot Token	11
2.2.2 Reddit API Přihlašovací Údaje	11
2.3 Cog pro YouTube hudbu	11

2.3.1	Popis funkcí .....	12
2.3.2	Závislosti .....	12
2.3.3	FFmpeg .....	12
2.4	Cog pro přehrávání náhodné hudby .....	12
2.4.1	Popis funkcí .....	12
2.4.2	Náhodný výběr .....	13
2.4.3	Závislosti .....	14
2.4.4	FFmpeg .....	14
2.5	Cog pro zobrazení náhodného meme z Redditu .....	14
2.5.1	Popis funkcí .....	14
2.5.2	Náhodný výběr meme .....	15
2.5.3	Závislosti .....	16
2.6	MadaraCog - odesílání náhodného GIFu .....	16
2.6.1	Příkazy .....	17
2.7	Quote Cog - Získávání náhodného citátu .....	18
2.7.1	Příkaz .....	18
2.7.2	Závislosti .....	18
2.8	Základní příkazy .....	19
2.9	Image Cog .....	19
2.10	Write Cog .....	20
<b>3</b>	<b>Výsledky .....</b>	<b>21</b>
3.1	Splněné cíle .....	21
3.2	Zhodnocení projektu .....	23
3.3	Budoucí rozvoj .....	23
<b>4</b>	<b>Závěr .....</b>	<b>24</b>
	<b>Seznam použitých informačních zdrojů .....</b>	<b>25</b>

## **Abstrakt**

Tato práce se zaměřuje na vývoj Discord bota, který poskytuje různé zábavné funkce pro uživatele Discord serveru. Bot je implementován v jazyce Python s využitím knihovny discord.py. Hlavní funkce zahrnují přehrávání YouTube videí, zobrazování Reddit příspěvků, odesílání GIF animací, citátů a obrázků. Důraz je kladen na modularitu kódu a snadnou rozšiřitelnost funkcí.

## **Klíčová slova**

Python, Discord API, discord.py, bot, YouTube, Reddit, GIF

**Abstract** This work focuses on developing a Discord bot that provides various entertainment functions for Discord server users. The bot is implemented in Python using the discord.py library. Its main features include YouTube video playback, displaying Reddit posts, sending GIF animations, managing quotes and images. Emphasis is placed on code modularity and the easy extensibility of its functions.

## **Keywords**

Python, Discord API, discord.py, bot, YouTube, Reddit, GIF

## SEZNAM POUŽITÝCH ZKRATEK

API	Application Programming Interface,
HTTP	Hypertext Transfer Protocol,
JSON	JavaScript Object Notation,
REST	Representational State Transfer,
SQL	Structured Query Language,
UI	User Interface.

# ÚVOD

## Představení projektu

Discord se stal jednou z nejpopulárnějších komunikačních platforem, zejména v herní komunitě. S rostoucím počtem uživatelů roste i potřeba efektivní správy Discord serverů. Projekt Discord bota vznikl s cílem poskytnout uživatelům zábavné funkce. Bot využívá Discord API a je implementován v Pythonu s využitím knihovny discord.py.

## Motivace

Hlavní motivací pro vytvoření tohoto bota byla snaha výrazně obohatit server o rozmanité a zábavné prvky, jako jsou vtipné příkazy a možnost přehrávání hudby. Cílem bylo vytvořit nástroj, který by nejen zlepšil celkovou atmosféru a interaktivitu na serveru, ale také poskytl uživatelům více příležitostí ke společné zábavě a sdílení hudebních zážitků.

## Cíle projektu

Hlavním cílem bylo vytvořit modulární a snadno rozšiřitelného Discord bota s následujícími funkcemi:

- Přehrávání YouTube videí včetně náhodného přehrávání.
- Zobrazování příspěvků z Redditu.
- Odesílání GIF animací.
- Odesílání citátů.
- Odesílání obrázků.
- Základní příkazy pro interakci.
- Textové příkazy.

## Struktura práce

Práce je rozdělena do několika hlavních částí:

- Analýza požadavků a návrh řešení představuje cílové funkce a architekturu.
- Implementační část detailně rozebírá realizaci jednotlivých komponent.
- Testování a nasazení popisuje proces testování a nasazení bota.
- Závěrečná část hodnotí výsledky a nabízí výhled do budoucna.

# 1 ANALÝZA A NÁVRH

## 1.1 Analýza požadavků

### 1.1.1 Funkční požadavky

- Zábavné funkce
  - YouTube přehrávač a náhodné přehrávání.
  - GIF animace.
  - Reddit příspěvky.
  - Poslání obrázků.
  - Náhodné citáty.
  - Probouzení uživatelů.
- Správa příkazů
  - Základní příkazy pro interakci.
  - Dynamické načítání cogů.
  - Textové příkazy.
  - Settings.py.

## 1.2 Struktura projektu

```
1 project/
2   main.py           # Hlavní soubor bota
3   settings.py       # Konfigurační soubor
4   secret.py         # Soubor s klíči
5   gifs.txt          # Soubor s odkazy na gif
6   image             # Složka s obrázky
7   fonts             # Složka s fonty
8   cogs/
9       basic.py       # Základní příkazy
10      cog_manager.py  # Správa cogů
11      gifs.py         # GIF příkazy
12      image.py        # Poslání obrázků
13      quote_cog.py    # Citáty
14      random_youtube_play.py # Náhodné YouTube
15      reddit.py       # Reddit příkazy
```



```
16     wakeup.py      # Wake-up funkce
17     write.py       # Textové příkazy
18     youtube.py     # YouTube přehrávač
```

Kód 1.1: Struktura projektu

## 1.3 Vysvětlení pojmů

### 1.3.1 Bot

V Discordu je bot automatizovaný program, který je navržený k tomu, aby vykonával různé úkoly na serverech (guildách) bez potřeby lidské interakce. Boti mohou být naprogramováni tak, aby reagovali na různé příkazy, události nebo prováděli specifické akce, jako je moderování, zábava, automatické odpovědi, hry, integrace s externími službami a mnoho dalšího.

### 1.3.2 Cog

V Discord botech (používajících knihovnu discord.py) je Cog způsob, jak organizovat a strukturovat kód do modulárních částí, což usnadňuje správu a údržbu bota, zejména když je bot větší a obsahuje více příkazů nebo funkcí. Cog je v podstatě třída nebo modul, který obsahuje související příkazy, události nebo funkce. Představ si, že máš různé části bota (například příkazy pro moderování, příkazy pro hry, příkazy pro hudbu atd.), každý z těchto bloků může být umístěn do samostatného cogu. To znamená, že místo toho, abys měl všechny příkazy v jednom souboru, můžeš je rozdělit do více souborů, což zjednoduší jejich správu.

## 1.4 Použité technologie

### 1.4.1 Discord API

Discord API poskytuje rozhraní pro programování aplikací, které umožňuje vývojářům vytvářet boty a aplikace pro platformu Discord. Například:

- Systém příkazů a událostí.

### 1.4.2 discord.py

Discord.py je Python knihovna pro práci s Discord API. Poskytuje:

- Asynchronní přístup k API pomocí asyncio.
- Systém extensions a cogs pro modularitu.

### **1.4.3 Knihovna requests**

Knihovna pro HTTP požadavky

### **1.4.4 Knihovna yt-dlp**

Knihovna pro extrakci videí a zvukových stop z různých webových stránek, především z YouTube.

### **1.4.5 FFmpeg**

Nástroj pro zpracování multimédií (audio a video).

### **1.4.6 Knihovna asyncpraw**

Asynchronní wrapper pro Praw (Python Reddit API Wrapper), což je knihovna, která poskytuje přístup k Reddit API.

- Používá se pro získání příspěvků z Redditu.
- Používá se zde asynchronní verze (async), což znamená, že operace s Redditem nejsou blokující a neovlivňují výkon bota.
- Reddit API.

### **1.4.7 Knihovna Pillow**

Knihovna pro práci s obrázky v Pythonu. Používá se k vytváření obrázků, vykreslování textu a ukládání obrázků ve formátu PNG.

### **1.4.8 Knihovna Textwrap**

Knihovna pro zalamování textu do více řádků na základě šířky.

### **1.4.9 Knihovna io**

Knihovna pro práci s binárními daty v paměti.

## 2 POUŽITÉ POSTUPY A ŘEŠENÍ

### 2.1 Settings.py

Soubor `settings.py` v Discord botu je soubor, který se používá k ukládání různých nastavení a konfigurací, které bot potřebuje k tomu, aby správně fungoval. Tento soubor může obsahovat různé informace, jako jsou API klíče, tokeny, ID serverů, příkazy pro bota a další nastavení.

#### 2.1.1 Import tajných klíčů

Soubor `settings.py` importuje následující tajné klíče z `secret.py`:

- `BOT_TOKEN` - Token pro připojení bota k Discordu.
- `REDDIT_CLIENT_ID` - Klientské ID pro připojení k Reddit API.
- `REDDIT_CLIENT_SECRET` - Klientský tajný klíč pro připojení k Reddit API.

#### 2.1.2 Výchozí hodnoty

- `DEFAULT_SUBREDDIT` - Výchozí subreddit, který se používá pro příkazy související s Redditem. Výchozí hodnota je `"memes"`.
- `DEFAULT_SORT` - Výchozí způsob řazení příspěvků z Reddit. Výchozí hodnota je `"hot"`.

#### 2.1.3 Platné možnosti řazení

`VALID_SORTS` je množina možných hodnot pro řazení příspěvků z Reddit. Platné možnosti jsou:

- `"hot"`.
- `"new"`.
- `"top"`.
- `"rising"`.
- `"controversial"`.

#### 2.1.4 Cesta k FFmpeg

`FFMPEG_PATH` obsahuje cestu k souboru `ffmpeg.exe`, který je vyžadován pro správnou funkci přehrávání hudby z YouTube. Výchozí hodnota je:

`C:/ffmpeg/bin/ffmpeg.exe`

### 2.1.5 Klíčová slova pro náhodné přehrávání

KEYWORDS je seznam klíčových slov, které se používají pro náhodné přehrávání hudby z YouTube. Tento seznam obsahuje různé žánry a typy hudby, například:

- "pop music".
- "rock hits".
- "top charts".
- "hip hop music".
- "indie songs".
- "jazz music".

## 2.2 Secret.py

Soubor `secret.py` obsahuje následující tajné klíče, které jsou používány v celém projektu.

### 2.2.1 Bot Token

BOT\_TOKEN je token, který slouží k autentizaci a připojení bota k Discord API. Tento token je jedinečný pro každého bota a nesmí být veřejně sdílen. Hodnota tohoto tokenu je:

```
MTI4Nzc2NTg2MDI5OTMwOTExOA.GTFsAj.xE9C4bJ0CX7bq5nwcD-v_e2l1nq7pmtQ0gC1Aw
```

### 2.2.2 Reddit API Přihlašovací Údaje

Pro připojení k Reddit API jsou potřeba následující přihlašovací údaje.

- REDDIT\_CLIENT\_ID - Klientské ID pro autentizaci aplikace na Redditu. Hodnota tohoto ID je:

```
abcdefghijklmnop1234567hijklmn89opqrst
```

- REDDIT\_CLIENT\_SECRET - Klientský tajný klíč pro autentizaci aplikace na Redditu. Hodnota tohoto klíče je:

```
zxy9876543210mnopqrstuvwxyz
```

## 2.3 Cog pro YouTube hudbu

Tento cog je určen pro přehrávání hudby z YouTube. Implementuje metody pro přehrávání skladeb a jejich zastavení v hlasovém kanálu na Discordu.

### 2.3.1 Popis funkcí

- **play(ctx, url)**

Tento příkaz přidá skladbu do fronty a spustí její přehrávání. Pokud bot není připojen v hlasovém kanálu, připojí se k němu.

- **stop(ctx)**

Tento příkaz zastaví aktuálně přehrávanou skladbu v hlasovém kanálu. Pokud žádná skladba nehraje, bot informuje uživatele.

### 2.3.2 Závislosti

Pro funkčnost tohoto cogu je nutné mít nainstalované následující knihovny:

- `discord.py` - Knihovna pro interakci s Discord API.
- `yt-dlp` - Knihovna pro stahování a přehrávání hudby z YouTube.

Tento cog využívá `yt-dlp` pro stahování a extrakci zvuku z YouTube, který následně přehraje v hlasovém kanálu.

### 2.3.3 FFmpeg

Pro správné přehrávání hudby je třeba mít nainstalovaný FFmpeg. Cestu k binárkám FFmpeg specifikuje proměnná `FFMPEG_PATH` v souboru `settings.py`.

## 2.4 Cog pro přehrávání náhodné hudby

Tento cog umožňuje přehrávání náhodně vybraných skladeb na základě klíčových slov. K dispozici jsou metody pro připojení do hlasového kanálu a přehrávání nebo zastavení hudby.

### 2.4.1 Popis funkcí

- **play\_random(ctx)**

Tato metoda připojí bota k hlasovému kanálu, pokud ještě není připojen, a začne přehrávat náhodně vybranou píseň. Písně jsou vybírány na základě klíčového slova z předdefinovaného seznamu (`KEYWORDS`). Pro každé klíčové slovo se provádí vyhledávání na YouTube, a první nalezený výsledek je přehrán.

- **stop\_random(ctx)**

Tato metoda zastaví aktuální přehrávanou skladbu a odpojí bota od hlasového kanálu. Pokud bot není připojen k žádnému kanálu, informuje o tom uživatele a neprovádí žádnou akci.

### 2.4.2 Náhodný výběr

Pro přehrávání skladeb bot používá náhodný výběr klíčového slova ze seznamu KEYWORDS, který je definován v souboru settings.py. Tento seznam obsahuje témata jako například "pop music", "rock hits" nebo "jazz music". Po výběru klíčového slova provádí bot vyhledávání na YouTube pomocí yt-dlp a přehraje první nalezený výsledek.

```
1 import random
2
3 class MusicPlayer(commands.Cog):
4     def __init__(self, bot):
5         self.bot = bot
6         self.keywords = ["pop music", "rock hits", "top charts",
7                          "hip hop music", "indie songs"]
8
9     async def play_next_song(self, ctx):
10         # Vyber náhodné klíčové slovo z listu
11         keyword = random.choice(self.keywords)
12         await ctx.send(f"Vybrané téma: **{keyword}**.")
13         Hledám a přehrávám písničku...)
14
15         song_url = await self.search_youtube(keyword)
16
17         if not song_url:
18             await ctx.send(f"Nepodařilo se najít písničku
19                             pro téma: {keyword}. Zkousím další...")
20             return await self.play_next_song(ctx)
21
22         # Přehrání nalezené skladby
23         source = discord.FFMpegPCMAudio(song_url, executable=FFMPEG_PATH)
24         self.voice_channel.play(source, after=lambda e:
25             self.bot.loop.create_task(self.after_song(ctx, e)))
```

Kód 2.1: Náhodný Výběr klíčového slova a přehrávání hudby

V tomto příkladu kódu je náhodně vybráno klíčové slovo z proměnné keywords, která obsahuje seznam témat. Po výběru klíčového slova bot provede vyhledávání na YouTube a přehraje první nalezený výsledek.

### 2.4.3 Závislosti

Pro funkčnost tohoto cogu je nutné mít nainstalované následující knihovny:

- `discord.py` - Knihovna pro interakci s Discord API.
- `yt-dlp` - Knihovna pro stahování a přehrávání hudby z YouTube.

Tento cog využívá `yt-dlp` pro stahování a extrakci zvuku z YouTube, který následně přehraje v hlasovém kanálu.

### 2.4.4 FFmpeg

Pro správné přehrávání hudby je třeba mít nainstalovaný FFmpeg. Cestu k binárkám FFmpeg specifikuje proměnná `FFMPEG_PATH` v souboru `settings.py`.

## 2.5 Cog pro zobrazení náhodného meme z Redditu

Tento cog umožňuje zobrazení náhodného meme obrázku z Redditu na základě zadaného subredditu a typu řazení. Uživatel může specifikovat subreddit a typ řazení, přičemž existují výchozí hodnoty.

### 2.5.1 Popis funkcí

- **`meme(ctx, subreddit=None, sort_type=None)`**

Tato metoda umožňuje uživateli získat náhodný meme obrázek z Redditu. Uživatel může zadat název subredditu a typ řazení (např. "hot", "new", "top", "rising", "controversial").

Pokud nejsou tyto argumenty zadány, použijí se výchozí hodnoty definované v `settings.py`.

- `subreddit`: Název subredditu pro vyhledání meme (výchozí je nastavený v `settings.py`).
- `sort_type`: Typ řazení příspěvků na Redditu, např. "hot", "new", "top", "rising", nebo "controversial" (výchozí je nastavený v `settings.py`).

Metoda ověřuje platnost zadaného typu řazení a následně vybírá příspěvky z Redditu podle specifikovaných kritérií. Vybere náhodný příspěvek a zobrazí ho v embed zprávě.

- **`cog_unload()`**

Tato metoda je volána při vypnutí cogu, kdy se zavře připojení k Reddit API. Používá se k uvolnění prostředků.

### 2.5.2 Náhodný výběr meme

K získání náhodného meme obrázku bot používá metodu `choice()` pro náhodný výběr platného příspěvku z Redditu, který není již zobrazený. Platné příspěvky jsou filtrovány na základě formátu souboru a bezpečnosti (nevhodný obsah je vynechán).

```
1 import random
2 import asyncpraw as praw
3
4 class Reddit(commands.Cog):
5     def __init__(self, bot):
6         self.bot = bot
7         self.reddit = praw.Reddit(
8             client_id=REDDIT_CLIENT_ID,
9             client_secret=REDDIT_CLIENT_SECRET,
10            user_agent="script:randommeme:v1.0
11                (by u/Agitated_Arachnid891)"
12        )
13        self.used_posts = set()
14
15    @commands.command()
16    async def meme(self, ctx: commands.Context,
17        subreddit: str = None, sort_type: str = None):
18        """
19        Zobrazí náhodný meme obrázek z vybraného subredditu a typu řazení.
20        """
21
22        subreddit = subreddit or DEFAULT_SUBREDDIT
23        sort_type = sort_type or DEFAULT_SORT
24
25        if sort_type not in VALID_SORTS:
26            await ctx.send(f"Invalid sort type '{sort_type}'.
27                Valid options are: {'', ' '.join(VALID_SORTS)}")
28            return
29
30        subreddit_obj = await self.reddit.subreddit(subreddit)
31        post_list = []
32
33        if sort_type == "hot":
```



```

34         async for post in subreddit_obj.hot(limit=30):
35             post_list.append(post)
36     elif sort_type == "new":
37         async for post in subreddit_obj.new(limit=30):
38             post_list.append(post)
39
40     # Výběr náhodného příspěvku
41     random_post = random.choice(post_list)
42
43     # Poslání odkazu na obrázek přímo
44     await ctx.send(f"Náhodný meme z r/{subreddit}: {random_post.url}"
45                   (Post created by {random_post.author.name if random_post.author
46                     else 'N/A'})")
47
48     # Uložíme post.id jako zobrazený příspěvek
49     self.used_posts.add(random_post.id)
50
51     # Po zobrazení deseti příspěvků vyprázdníme seznam
52     if len(self.used_posts) >= 10:
53         self.used_posts.clear()
54
55     def cog_unload(self):
56         """
57         Zavře připojení k Reddit API při odinstalování cogu.
58         """
59         self.bot.loop.create_task(self.reddit.close())

```

Kód 2.2: Zobrazení náhodného meme z Redditu

### 2.5.3 Závislosti

Pro spuštění tohoto cogu je potřeba nainstalovat následující závislosti:

- discord.py: Knihovna pro interakci s Discord API.
- asyncpraw: Asynchronní klient pro interakci s Reddit API.

## 2.6 MadaraCog - odesílání náhodného GIFu

Tento cog umožňuje odesílat náhodné GIFy na základě textového souboru, přičemž zajišťuje, že se nepošle ten samý GIF, který byl poslán naposledy. Také obsahuje příkaz pro odesílání

GIFu s konkrétním odkazem.

### 2.6.1 Příkazy

- **madara**: Odesílá náhodný GIF ze souboru `gifs.txt`, přičemž zajišťuje, že není stejný jako ten předchozí.
- **gif**: Odesílá pevně definovaný GIF.

```
1  class MadaraCog(commands.Cog):
2  def __init__(self, bot):
3      self.bot = bot
4      self.last_gif = None # Uchování posledního GIFu
5
6  @commands.command(help="Pošle náhodný GIF z gifs.txt,
7      který není stejný jako poslední.")
8  async def madara(self, ctx):
9      try:
10         with open('gifs.txt', 'r') as file:
11             gifs = file.readlines()
12
13         gifs = [gif.strip() for gif in gifs] # Odstranění bílých míst
14         random_gif = random.choice(gifs)
15
16         while random_gif == self.last_gif:
17             # Zajištění, že GIF není stejný jako ten předchozí
18             random_gif = random.choice(gifs)
19
20         self.last_gif = random_gif
21         await ctx.send(random_gif)
22
23     except FileNotFoundError:
24         await ctx.send("Soubor gifs.txt nebyl nalezen!")
25     except Exception as e:
26         await ctx.send(f"Došlo k chybě: {e}")
```

Kód 2.3: Implementace náhodného odesílání GIFu

## 2.7 Quote Cog - Získávání náhodného citátu

Tento cog umožňuje uživatelům získat náhodný citát s autorem z veřejného API. Kód využívá API `zenquotes.io` pro získávání citátů a jejich odeslání do kanálu na Discordu.

### 2.7.1 Příkaz

- **quote**: Získává a zobrazuje náhodný citát s autorem.

```
1 class Quote(commands.Cog):
2     def __init__(self, bot):
3         self.bot = bot
4
5     @commands.command(help="Zobrazí náhodný citát z API.")
6     async def quote(self, ctx):
7         try:
8             # API endpoint pro náhodný citát
9             response = requests.get("https://zenquotes.io/api/random")
10
11            # Zkontroluj, zda je odpověď úspěšná (status kód 200)
12            if response.status_code == 200:
13                data = response.json()
14                # Extrahuj citát a autora z odpovědi API
15                quote = data[0]['q']
16                author = data[0]['a']
17
18                # Odešli citát do Discordu
19                await ctx.send(f'"{quote}" {author}')
20            else:
21                # Pokud API neodpoví úspěšně, odešli chybovou zprávu
22                await ctx.send
23                ("Nemohu získat citát, zkuste to prosím později.")
24        except Exception as e:
25            # Zachytí chybu a odešle ji do Discordu
26            await ctx.send(f"Došlo k chybě: {str(e)}")
```

Kód 2.4: Získávání náhodného citátu z API

### 2.7.2 Závislosti

Pro spuštění tohoto cogu je potřeba nainstalovat následující závislosti:

- discord.py: Knihovna pro interakci s Discord API.
- requests: Knihovna pro HTTP požadavky (pro získání citátu z API).

## 2.8 Základní příkazy

Cog basic.py obsahuje základní interaktivní příkazy:

- ping: Odpoví na příkaz .ping zprávou "Pong!".
- pong: Odpoví na příkaz .pong zprávou "Ping!".
- hello: Odpoví na příkaz .hello pozdravem, přičemž zmíní autora příkazu.

```

1 @commands.command(help="Odpoví na příkaz .ping s 'Pong!'")
2 async def ping(self, ctx):
3     """
4     Odpoví na příkaz .ping s "Pong!".
5     """
6     await ctx.send("Pong!")
7
8 @commands.command(help="Odpoví na příkaz .hello s pozdravem.")
9 async def hello(self, ctx):
10    """
11    Odpoví na příkaz .hello s pozdravem.
12    """
13    await ctx.send(f'Ahoj, {ctx.author}!')
```

Kód 2.5: Implementace základních příkazů

## 2.9 Image Cog

Tento cog umožňuje odesílání obrázků ze složky.

- image [číslo obrázku]: Odesílá obrázek ze složky ./image. Pokud není číslo zadáno, obrázek je vybrán náhodně.
- list\_images: Vypíše seznam všech obrázků ve složce ./image, seřazených podle čísel.

```

1 def __init__(self, bot):
2     self.bot = bot
3     self.image_folder = "./image" # Cesta ke složce s obrázky
```

```

4      @commands.command(name="image", help="Pošle obrázek ze složky.
5          Zadejte číslo obrázku nebo nechte náhodně vybrat.")
6      async def send_image(self, ctx, image_number: int = None):
7          if image_number is not None:
8              if 1 <= image_number <= len(images):
9                  selected_image = images[image_number - 1]
10             else:
11                 await ctx.send(f"Zadejte číslo mezi 1 a {len(images)}.")
12                 return
13         else:
14             selected_image = random.choice(images)

```

Kód 2.6: Ukázka příkazu

- discord.py: Knihovna pro interakci s Discord API.

## 2.10 Write Cog

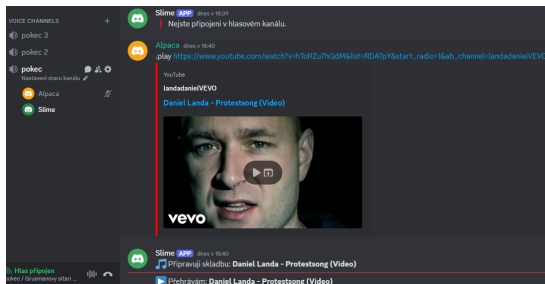
Tento cog vznikl po několika hodinách diskuzí s ChatGPT. Původně jsem měl v úmyslu implementovat funkci, která by generovala obrázky na základě textu zadaného uživatelem pomocí AI. Během vývoje se však objevily problémy s přístupem k AI a problémy s verzí programu, nakonec jsem skončil u funkce, která vykreslí zadaný text na bílé pozadí ve vybraném písmu. I když to nebyl původní záměr, výsledek je krásný sám o sobě, zejména když je kombinován s příkazem quote.

## 3 VÝSLEDKY

### 3.1 Splněné cíle

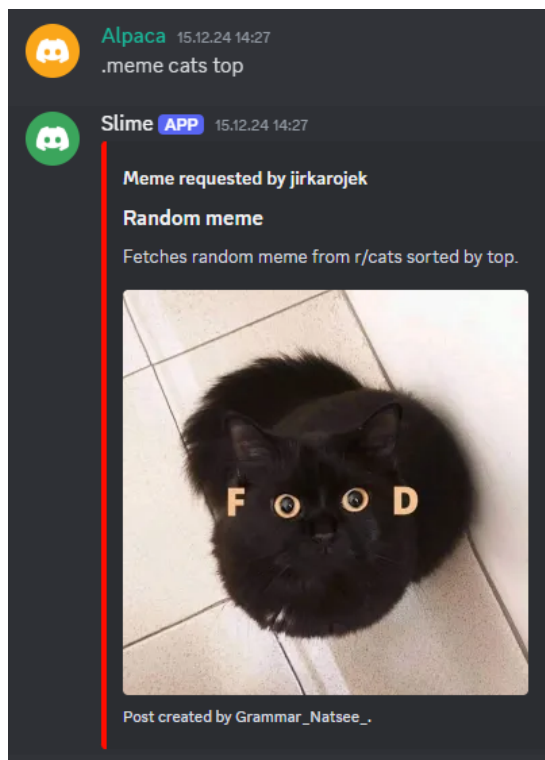
V rámci projektu se mi podařilo úspěšně implementovat funkce.

- Přehrávání hudby z YouTube.



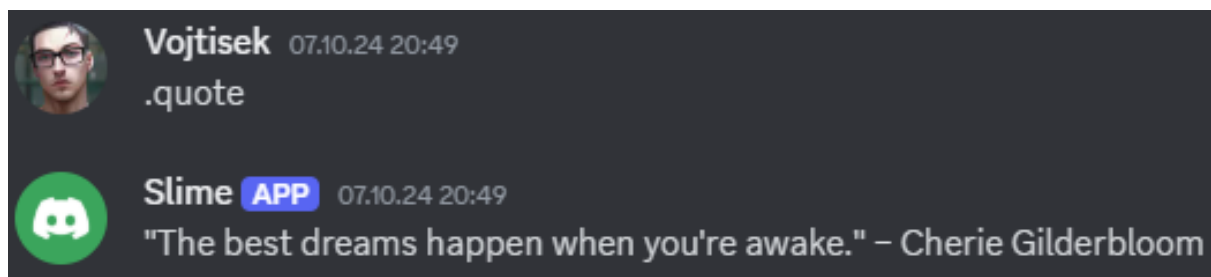
Obrázek 3.1: Ukázka z YouTube

- Náhodné přehrávání hudby.
- Posílání příspěvků z Redditu.



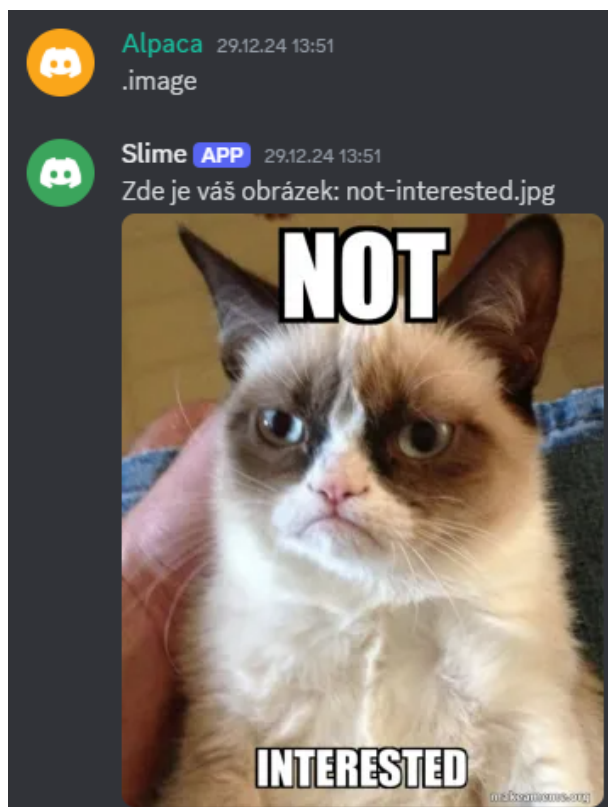
Obrázek 3.2: Ukázka meme z Redditu

- Posílání citátů.



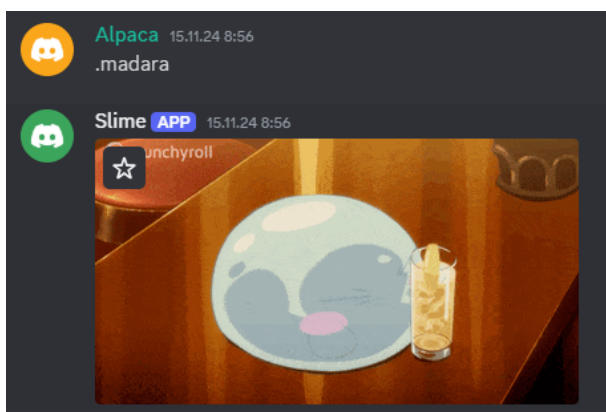
Obrázek 3.3: Ukázka citátu

- Posílání obrázků.



Obrázek 3.4: Ukázka obrázku

- Posílání GIFů.



Obrázek 3.5: Ukázka GIFu

## 3.2 Zhodnocení projektu

Projekt rozšířil možnosti komunikace na serveru a obohatil je o několik zábavných funkcí.

## 3.3 Budoucí rozvoj

Na základě zkušeností z vývoje a zpětné vazby uživatelů byly identifikovány oblasti pro další rozvoj aplikace. Možná vylepšení:

- Spojení funkce vykreslení textu na obrázek a funkce s citáty.
- Vylepšení náhodného přehrávání hudby z YouTube.
- Rozšíření přehrávání hudby o Spotify.
- Nové upozornění na hry zdarma.



## 4 ZÁVĚR

Discord bot byl úspěšně implementován jako zábavný nástroj pro Discord servery. Hlavní funkce zahrnují:

- Přehrávání hudby z YouTube včetně náhodného přehrávání.
- Zobrazování meme obrázků z Redditu.
- Odesílání GIF animací.
- Zobrazování náhodných citátů.
- Základní interaktivní příkazy.
- Systém probouzení uživatelů.
- Textové příkazy.

Zdrojový kód projektu je dostupný na GitHubu:

[https://github.com/JirkaITRojek/mat\\_project/tree/master](https://github.com/JirkaITRojek/mat_project/tree/master). Bot je implementován v Pythonu s využitím knihovny discord.py a využívá modulární architekturu založenou na systému cogů. Tato architektura umožňuje snadné přidávání nových funkcí a údržbu kódu. Při vytváření tohoto projektu byly použity různé nástroje a technologie, včetně generativního jazykového modelu ChatGPT (verze 4), který pomohl při generování textu, návrhu struktury projektu a asistoval při analýze nápadů. Tento nástroj byl využit pro zefektivnění práce a urychlení procesu.

## SEZNAM POUŽITÝCH INFORMAČNÍCH ZDROJŮ

- 1 YouTube tutoriál k základům -

[https://www.youtube.com/watch?v=93DrbetB6oM&list=PLwqYQaS6jxfmCUTbFU-\\_d5M4yGTmKS0gk&index=11&ab\\_channel=Paradoxial](https://www.youtube.com/watch?v=93DrbetB6oM&list=PLwqYQaS6jxfmCUTbFU-_d5M4yGTmKS0gk&index=11&ab_channel=Paradoxial).

- 2 YouTube tutoriál k Redditu -

[https://www.youtube.com/watch?v=eblCb7LQ\\_pc&list=PLwqYQaS6jxfmCUTbFU-\\_d5M4yGTmKS0gk&index=6&ab\\_channel=Paradoxial](https://www.youtube.com/watch?v=eblCb7LQ_pc&list=PLwqYQaS6jxfmCUTbFU-_d5M4yGTmKS0gk&index=6&ab_channel=Paradoxial).

- 3 YouTube tutoriál k YouTube -

[https://www.youtube.com/watch?v=dRHUW\\_KnHLs&ab\\_channel=Computeshorts](https://www.youtube.com/watch?v=dRHUW_KnHLs&ab_channel=Computeshorts).

- 4 asyncio Documentation -

<https://docs.python.org/3/library/asyncio.html>.