

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені І.І.МЕЧНИКОВА

(повне найменування вищого навчального закладу)

Інститут математики, економіки та механіки

(повне найменування інституту, назва факультету (відділення))

Кафедра математичного забезпечення комп'ютерних систем

(повна назва кафедри (предметної, циклової комісії))

## Дипломна робота

бакалавра

(освітньо-кваліфікаційний рівень)

на тему Розробка бота для додатку Discord

Development of bot for Discord application

Разработка бота для приложения Discord

Виконав: студент 4 курсу, групи 1  
напряму підготовки (спеціальності)

6.050102 Комп'ютерна інженерія

(шифр і назва напряму підготовки, спеціальності)

Григорян А.І.

(прізвище та ініціали)

Керівник доц. Антоненко О.С.

(прізвище та ініціали)

Рецензент ст.викл. Геренко О.А.

(прізвище та ініціали)

Рекомендовано до захисту:

Протокол засідання кафедри

№ від « » р.

Захищено на засіданні ЕК №

протокол № від « » р.

Оцінка

(за 4-х бальною шкалою, за шкалою ECTS, бал.)

Завідувач кафедри

Голова ЕК

Є.В. Малахов

(підпис)

(прізвище, ініціали)

О.О. Арсірій

(підпис)

(прізвище, ініціали)

Одеса - 2017

## АНОТАЦІЯ

Дипломна робота присвячена темі розробки бота для популярного месенджера Discord.

Метою роботи є створення розширення над Discord API, що має ієрархічну модульну архітектуру, а також створення бота за допомогою розширеного Discord API.

В результаті проведених в роботі досліджень з аналізу предметної області, вдалося спроектувати ієрархічну модульну архітектуру, що володіє такими властивостями як: відключення / підключення компонентів, наявністю системи ролей, системи подій, а також системи дозволів. Дана архітектура дозволяє створювати ботів, з великою кількістю команд з єдиною системою управління компонентами.

За допомогою розширення був спроектований і створений бот, що складається з десяти модулів, який володіє таким набором функцій: модерація сервера, управління правами на сервері, фільтрація повідомлень користувачів, збір і показ статистики на сервері, математичний калькулятор, показ рекламних повідомлень.

Після розробки бот був протестований на двох серверах Discord з різною тематикою.

## ABSTRACT

The graduate thesis is dedicated to bot development for the popular messenger Discord.

The goal of the work is creation of extension for Discord API, which has an hierarchical module architecture and also bot creation, using Discord API extension.

In result of accomplished work in the field of domain analysis, the development of hierarchical module architecture, which has following properties: components shutting down/connecting, roles management system, events management system and also permission system was succeed. The given architecture allows bots creation, which have big amount of commands with a single components management system. With the help of extension the bot was developed and created, which consists of 10 modules and implements following set of functions: server moderation, permissions managements on the server, users messages filtration, statistic collecting and displaying on the server, mathematic calculator, advertising messages displaying.

After bot development the bot was tested on two Discord servers with various subject matter.

## АННОТАЦИЯ

Дипломная работа посвящена теме разработке бота для популярного месенджера Discord.

Целью работы является создание расширения над Discord API, имеющего иерархическую модульную архитектуру, а так же создание бота с помощью расширенного Discord API.

В результате проведенных в работе исследований по анализу предметной области, удалось спроектировать иерархическую модульную архитектуру, обладающую такими свойствами как: отключение/подключение компонентов, наличием системы ролей, системы событий, а так же системы разрешений. Данная архитектура позволяет создавать ботов, с большим количеством команд с единой системой управления компонентами.

С помощью расширения был спроектирован и создан бот, состоящий из десяти модулей, реализующий следующим набором функций: модерация сервера, управление правами на сервере, фильтрация сообщений пользователей, сбор и показ статистики на сервере, математический калькулятор, показ рекламных сообщений.

После разработки было произведено тестирование бота на двух серверах Discord с разной тематикой.

## ЗМІСТ

	стор.
ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ .....	7
ВСТУП .....	8
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАВДАННЯ.....	11
1.1 Розважальні боти.....	12
1.2 Боти для модерації .....	13
1.3 Вимоги до розроблюваної системи .....	15
2 ПРОЕКТУВАННЯ БОТУ .....	17
2.1 Ієрархічне модульне розширення Discord API .....	17
2.1.1 Команда.....	17
2.1.2 Подія.....	18
2.1.3 Компонент.....	19
2.1.4 Дозвіл .....	19
2.1.5 Модуль .....	19
2.1.6 Бот.....	20
2.2 Побудова архітектурної моделі програми.....	20
2.3 Проектування модулів для бота .....	22
2.3.1 Ядро .....	22
2.3.2 Математический модуль .....	22
2.3.3 Модуль для фільтрації повідомлень .....	24
2.3.4 Модуль реклами .....	24
2.3.5 Модуль модерації.....	24
2.4 Проектування бази даних.....	25
3 ПРОГРАМНА РЕАЛІЗАЦІЯ БОТУ .....	29
3.1 Вибір технологій .....	29
3.2 Проект KingRichard.Data .....	31
3.3 Проект KingRichard.Bot.....	33
3.4 Проект KingRichard.Bot.Modules.....	36
3.5 Проект KingRichard.Bot.Modules.Advertising.....	37
3.6 Проект KingRichard.Bot.Modules.Moderation .....	38

3.7 Проект KingRichard.Bot.Modules.Statistic .....	39
3.8 Проект KingRichard.Bot.Modules.WordFilter .....	40
3.9 Проект KingRichard.Calculator .....	41
3.10 Проект KingRichard.CUI .....	42
3.11 Опис функціонала бота .....	43
3.11.1 Модуль Statistic .....	43
3.11.2 Модуль Core .....	44
3.11.3 Модуль Math .....	47
3.11.4 Модуль Filters .....	47
3.11.5 Модуль Advertising .....	48
3.11.6 Модуль Moderation .....	48
3.12 Тестування бота .....	48
3.12.1 Зібрана статистика .....	49
ВИСНОВОК .....	50
ЛІТЕРАТУРА .....	51
ДОДАТОК А СПИСОК КОМАНД FREDBOAT .....	52
ДОДАТОК Б СПИСОК КОМАНД KAWAIBOT .....	53
ДОДАТОК В РЕГУЛЯРНИЙ ВИРАЗ ДЛЯ РОЗБОРУ НЕНОРМАТИВНОЇ ЛЕКСИКИ .....	55
ДОДАТОК Г ГРАММАТИКА СТРОКОВОГО КАЛЬКУЛЯТОРА .....	56
ДОДАТОК Д СЦЕНАРІЙ СТВОРЕННЯ БАЗИ ДАНИХ .....	60
ДОДАТОК Е ФІЗИЧНА СХЕМА БАЗИ ДАНИХ .....	63
ДОДАТОК Ж ОБ'ЄКТНА СХЕМА БАЗИ ДАНИХ .....	64
ДОДАТОК И ДІАГРАМА КЛАСІВ ПРОЕКТУ KINGRICHARD.DATA .....	65
ДОДАТОК К ДІАГРАМА ЗАЛЕЖНОСТЕЙ ПРОЕКТУ KINGRICHARD.DATA .....	66
ДОДАТОК Л ДІАГРАМА КЛАСІВ ПРОЕКТУ KINGRICHARD.BOT .....	67
ДОДАТОК М ДІАГРАМА ЗАЛЕЖНОСТЕЙ ПРОЕКТУ KINGRICHARD.BOT .....	68
ДОДАТОК Н ВИКОРИСТАННЯ KINGRICHARD.BOT .....	69
ДОДАТОК П ВИХІДНИЙ КОД КЛАСУ PROGRAM ПРОЕКТУ KINGRICHARD.CUI .....	71
ДОДАТОК Р ВІДГУКИ КОРИСТУВАЧІВ .....	72
ДОДАТОК С ПРИКЛАДИ SQL ЗАПИТІВ ДО ЗІБРАНОЇ СТАТИСТИКИ ..	73

## ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ

### Скорочення:

SMS – Short Message Service;

API – Application Programming Interface;

BCL – Base Class Library;

CLR – Common Language Runtime;

ORM – Object-Relational Mapping;

РСУБД – Реляційна Система Управління Базами Даних;

EF – Entity Framework;

EDM – Entity Data Model;

LINQ – Language Integrated Query;

GPPG – Gardens Point Parser Generator.

### Терміни:

мультимедійний файл - комп'ютерний файл, який містить аудіо- та відеоінформацію;

плей-лист - список звукових доріжок (музичних композицій і іншого роду треків), складений для відтворення.

## ВСТУП

Велику роль в сучасному спілкуванні грають системи миттєвого обміну повідомленнями (месенджери), такі як Telegram, Viber, Skype і т.д. Основне їх призначення - обмін повідомленнями в реальному часі через мережу Інтернет. Можуть передаватися як текстові повідомлення, так і звукові сигнали, зображення, відео. Свою популярність вони здобули в зв'язку з наявністю як текстових, так і голосових чатів, можливістю створення конференцій на велику кількість людей. Більшість месенджерів мають в наявності додатки під мобільні пристрої, що дозволяє порівняти їх з вбудованою функцією надсилання SMS, проте месенджери не вимагають окремої плати за відправку повідомлення, також месенджери дозволяють відправляти мультимедійні дані. У порівнянні з соціальними мережами, які так само зазвичай включають функцію миттєвої відправки повідомлень, месенджери простіші у використанні, та в них відсутні типові для соціальних мереж функції, такі як створення спільнот, розміщення інформації про себе і т.д. Одним з представників є відносно молодий додаток Discord, який був створений 6 березня 2016 року, і за короткий час набув популярності серед ігрових спільнот, а також розробників [1]. Discord в себе включає кращі риси всіх сучасних додатків, призначених для спілкування. Кожен користувач має можливість створити свій сервер, також званий гільдією [2], що являє собою ізольованому колекцію користувачів і каналів на сервері Discord, і прив'язати гільдію до певного географічного регіону. Користувач, який створив сервер, є власником даного сервера, і має необмежені права. Після чого на даному сервері є можливість створити будь яку кількість текстових і голосових чатів, запрошувати інших людей за допомогою тимчасового або постійного посилання на даний сервер, розмежовувати права. Однак однією з головних особливостей програми Discord є створення і підключення до сервера бота.



Бот - спеціальна програма, що виконує автоматично і / або за заданим розкладом будь-які дії через текстовий чат, призначений для людей. Управління ботом відбувається за допомогою команд, команда являє собою повідомлення певного формату: (префікс) (ім'я / алеас) [вхідні параметри]. Зазвичай бот позитивно позначається на працездатності сервера, автоматизуючи однотипну роботу з управління сервером.

Головне завдання ботів на сервері це допомога в модерації та адмініструванні, однак, функціонал ботів цим не обмежений, у зв'язку з цим боти можуть виконувати безліч поставлених завдань будь-якого характеру. Discord надає зручний API на багатьох мовах програмування, як наслідок цього, сфера створення ботів досить швидко розвивається, на момент створення додаток кількість створених ботів перевищує одну тисячу [3]. Однак існують недоліки в Discord API: бот складається тільки з команд і подій. Події відбуваються при виконанні деякої дії на сервері, наприклад, користувач зайшов в голосовий чат, при цьому події не видно користувачам, і ніяк не контролюються адміністратором сервера, що знижує безпеку на сервері. Недоліком команд є відсутність розмежувань за ролями, внаслідок цього, звичайно, будь-який користувач сервера, може виконати будь-яку команду бота, що тільки ускладнює модерацію на сервері. Так само зазвичай не вистачає одного бота зважаючи на обмеженість його набору команд, в зв'язку з цим з'являється необхідність додавання другого, третього, і т.д. ботів, що викликає труднощі у користувачів, через різні префікси команд, перетину функціональності різних ботів. У зв'язку з вищевказаними проблемами одним з методів вирішення є створення розширення над Discord API, яке дозволить більш просто створювати команди, і групувати їх в модулі. Так само власник сервера повинен мати можливість розмежовувати права користування командами, що формує вимогу про деяку просту систему ролей. Власник повинен мати можливість відключати і підключати модулі, команди, з огляду на їх непотрібність або необхідність в конкретний момент часу. Наступною особливістю є можливість схожої взаємодії, як з командами,

так і з подіями. Дане розширення Discord API допоможе замінити зростання кількості ботів на сервері, на збільшення кількості модулів бота, які мають один інтерфейс і взаємодіють один з одним.

Таким чином, метою даної роботи є створення розширення над Discord API, яке являє собою додавання вищевказаних можливостей і має модульну архітектуру, а також створення бота за допомогою розширеного Discord API.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести дослідження предметної області;
- провести огляд існуючих аналогів;
- створити розширення для Discord API;
- створити бота за допомогою розширення;
- забезпечити бота необхідним функціоналом для модерації сервера, збору статистики, базових математичних обчислень, фільтрації каналів сервера;
- впровадити бота на сервер Discord;
- перевірити працездатність бота.

## ВИСНОВОК

У даній роботі був проведений аналіз існуючих ботів та їх методів розробки на базі чого спроектоване та розроблене розширення Discord API, що дозволяє створювати ботів з ієрархічної модульною архітектурою. Використовуючи дане розширення був створений бот, який має ряд переваг, таких як:

- більш гнучке налаштування бота; немає необхідності перезавантажувати систему при простих змінах, таких як відключення або включення компонентів;
- проста процедура створення модулів, і підключення їх до вже готового боту;
- завдяки модульності є можливість скомпонувати тільки ті функціональні модулі, які будуть ідеально підходити під предметну область сервера;
- прозора система подій зробить сервер більш захищеним;
- система дозволів, що дозволяє модулям взаємодіяти один з одним.

Для боту були реалізовані модулі з функціоналом для модерації сервера, збору статистики, базовими математичними обчисленнями, фільтрації каналів сервера, показу реклами, а також модуль для управління ботом. Бот використовувався протягом місяця на двох реальних серверах Discord для автоматизації модерування.

У перспективі розвитку додатку, можна додати нові функціональні модулі, додати можливість динамічного підключення модулів, без перекомпіляції бота. Також в перспективі є можливість вести докладну статистику по серверам для преміум користувачів, і надати різні способи її відображення.

## ЛІТЕРАТУРА

1. Marks T., One year after its launch, Discord is the best VoIP service available [Електронний ресурс]. / Tom Marks // PC Gamer, 13.05.2016 – Режим доступу: <http://www.pcgamer.com/one-year-after-its-launch-discord-is-the-best-voip-service-available> – 01.05.2017
2. Guild Resource [Електронний ресурс]. – Режим доступу: <https://discordapp.com/developers/docs/resources/guild>
3. Discord Bots [Електронний ресурс]. – Режим доступу: <https://bots.discord.pw> – 02.05.2017
4. Telegram FAQ [Електронний ресурс]. – Режим доступу: <https://telegram.org/faq> – 05.05.2017
5. ManyBot [Електронний ресурс]. – Режим доступу: <https://manybot.io/ru> – 10.05.2017
6. Ayana [Електронний ресурс]. – Режим доступу: <https://ayana.io> – 15.05.2017
7. Язык C# [Електронний ресурс]. – Режим доступу: <https://msdn.microsoft.com/ru-ru/library/ms186214.aspx> – 17.05.2017
8. Microsoft SQL Server [Електронний ресурс]. – Режим доступу: <http://searchsqlserver.techtarget.com/definition/SQL-Server> – 17.05.2017
9. ADO.NET Entity Framework [Електронний ресурс]. – Режим доступу: [https://professorweb.ru/my/ADO\\_NET/base/level3/3\\_1.php](https://professorweb.ru/my/ADO_NET/base/level3/3_1.php) – 25.05.2017
10. Плущер-Сарно А. Ю. О семантике слова «мат» / А. Ю. Плущер-Сарно // Большой словарь мата. – Том 1. – Лимбус Пресс, 2005.
11. Григорян А.И., Разработка бота под Discord / А.И. Григорян, А.С. Антоненко // Тезисы докладов 14 Всеукраинской конференции студентов и молодых учёных «Информатика, информационные системы и технологии». — Одесса. — 2017. — С.101-102.

## ДОДАТОК А

### СПИСОК КОМАНД FREDBOAT



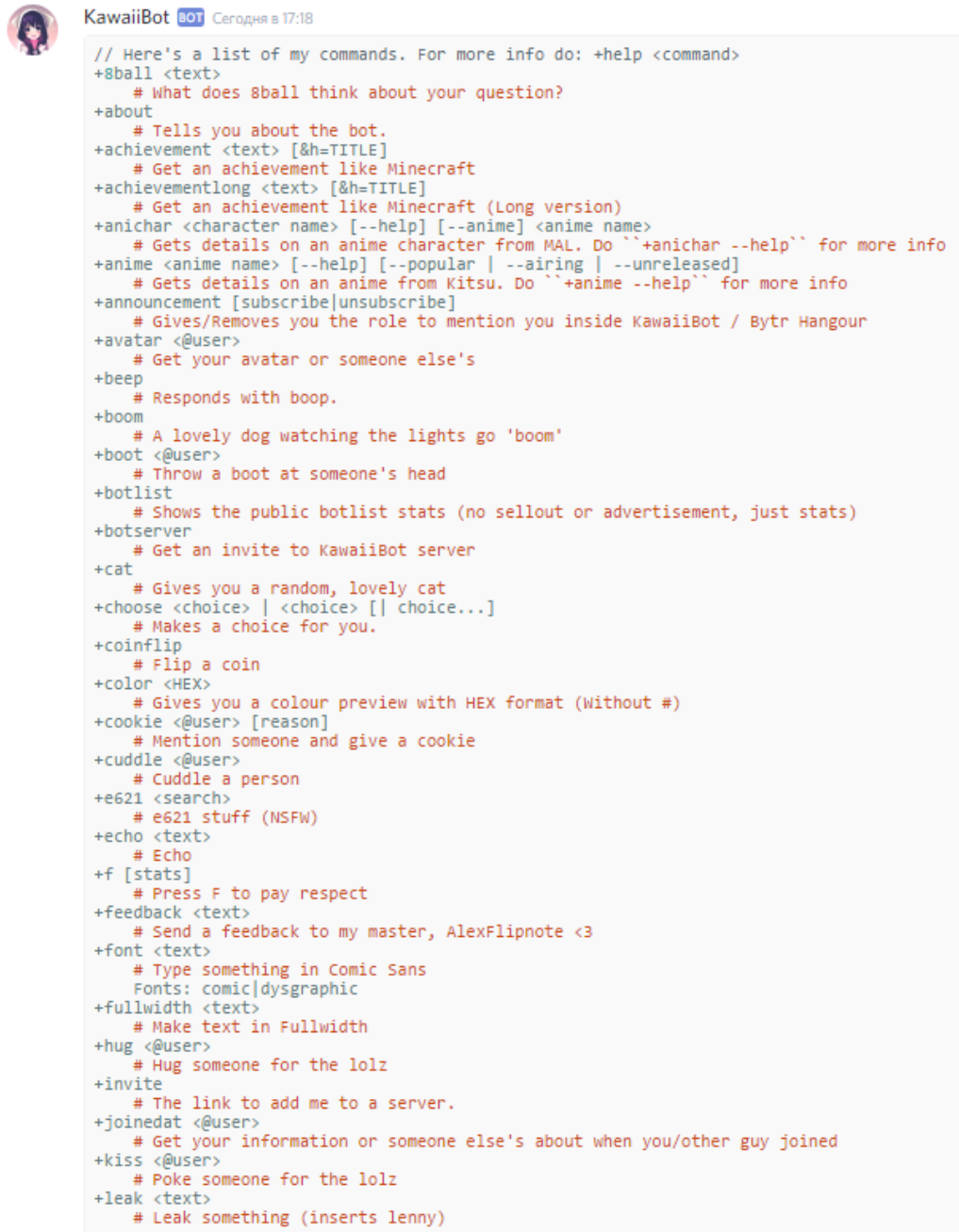
FredBoat🎵 BOT Вчера в 08:47

```
< FredBoat Music Commands >
;;play <url> OR ;;play <search-term>
#Play music from the given URL or search for a track. For a full list of sources visit http://docs.frederikam.com/
;;list
#Display a list of the current songs in the playlist.
;;nowplaying
#Display the currently playing song.
;;skip OR ;;skip n OR ;;skip n-m
#Skip the current song, the nth song in the queue, or all songs from n to m. Please use in moderation.
;;stop
#Stop the player and clear the playlist. Reserved for moderators with Manage Messages permission.
;;pause
#Pause the player.
;;unpause
#Unpause the player.
;;join
#Make the bot join your current voice channel.
;;leave
#Make the bot leave the current voice channel.
;;repeat single|all|off
#Toggle between repeat modes.
;;shuffle
#Toggle shuffle mode for the current queue.
;;reshuffle
#Reshuffle the current queue.
;;forward [[hh:]mm:]ss
#Forward the track by a given amount of time. Example: ;;forward 2:30
;;rewind [[hh:]mm:]ss
#Rewind the track by a given amount of time. Example: ;;rewind 30
;;seek [[hh:]mm:]ss
#Set the position of the track to the given time. Example: ;;seek 2:45:00
;;restart
#Restart the currently playing track.
;;export
#Export the current queue to a hastebin link, can be later used as a playlist.
;;split <url>
#Split a YouTube video into a tracklist provided in its description.
;;select n OR ;;play n
#Select one of the offered tracks after a search to play.
;;gr
#Show the current song played on gensokyoradio.net
;;volume <0-150>
#Changes the volume. Values are 0-150 and 100 is the default. The volume command is deprecated on the public bot.
;;destroy
#Reset the player and clear the playlist. Reserved for moderators with Manage Messages permission.
```

## ДОДАТОК Б

### СПИСОК КОМАНД KAWAIIBOT

Через занадто великий список команд, весь список не вміщується в одне повідомлення Discord, так як існує обмеження в дві тисячі символів, таким чином, всі команди були поділені на 3 частини (рис. Б.1 - Б.3).



The image shows a screenshot of a Discord chat interface. On the left, there is a circular profile picture of a character with dark hair and a pink bow. To the right of the profile picture, the name 'KawaiiBot' is displayed in blue, followed by a blue 'BOT' tag and the text 'Сьогодні в 17:18'. The main content of the message is a long list of commands and their descriptions, formatted in a light blue monospace font. The commands are listed one by one, each followed by a description of its function. The list includes commands like +8ball, +about, +achievement, +achievementlong, +anichar, +anime, +announcement, +avatar, +beep, +boom, +boot, +botlist, +botserver, +cat, +choose, +coinflip, +color, +cookie, +cuddle, +e621, +echo, +f, +feedback, +font, +fullwidth, +hug, +invite, +joinedat, +kiss, and +leak.

```
// Here's a list of my commands. For more info do: +help <command>
+8ball <text>
  # What does 8ball think about your question?
+about
  # Tells you about the bot.
+achievement <text> [&h=TITLE]
  # Get an achievement like Minecraft
+achievementlong <text> [&h=TITLE]
  # Get an achievement like Minecraft (Long version)
+anichar <character name> [--help] [--anime] <anime name>
  # Gets details on an anime character from MAL. Do ``+anichar --help`` for more info
+anime <anime name> [--help] [--popular | --airing | --unreleased]
  # Gets details on an anime from Kitsu. Do ``+anime --help`` for more info
+announcement [subscribe|unsubscribe]
  # Gives/Removes you the role to mention you inside KawaiiBot / Bytr Hangour
+avatar <@user>
  # Get your avatar or someone else's
+beep
  # Responds with boop.
+boom
  # A lovely dog watching the lights go 'boom'
+boot <@user>
  # Throw a boot at someone's head
+botlist
  # Shows the public botlist stats (no sellout or advertisement, just stats)
+botserver
  # Get an invite to KawaiiBot server
+cat
  # Gives you a random, lovely cat
+choose <choice> | <choice> [| choice...]
  # Makes a choice for you.
+coinflip
  # Flip a coin
+color <HEX>
  # Gives you a colour preview with HEX format (Without #)
+cookie <@user> [reason]
  # Mention someone and give a cookie
+cuddle <@user>
  # Cuddle a person
+e621 <search>
  # e621 stuff (NSFW)
+echo <text>
  # Echo
+f [stats]
  # Press F to pay respect
+feedback <text>
  # Send a feedback to my master, AlexFlipnote <3
+font <text>
  # Type something in Comic Sans
  Fonts: comic|dysgraphic
+fullwidth <text>
  # Make text in Fullwidth
+hug <@user>
  # Hug someone for the lolz
+invite
  # The link to add me to a server.
+joinedat <@user>
  # Get your information or someone else's about when you/other guy joined
+kiss <@user>
  # Poke someone for the lolz
+leak <text>
  # Leak something (inserts lenny)
```

Рисунок Б.1 – Перша частина відповіді на команду «help»

```

+letteremoji <search>
  # Search on the ~urban~ wiki for the 'best' answers
+lewd
  # Post an image saying lewd
+makememe <URL to image> | <top> | <bottom>
  # Make a meme from a picture URL (Bot should have permission to delete messages)
+manga <manga/novel name>
  # Gets details on a manga from MAL.
+meme (list|symbols|show) <meme> | <top> | <bottom>
  # Make a ~dank~ meme
+memeuser <@user> | <top> | <bottom>
  # Make a meme from the mentioned user
+notwork
  # Post an image saying 'That is not how it works you little shit'
+osu [mode] sig [username] [hex color] | [mode] <user|best|recent> [username]
  # Checks information regarding Osu
+pat <@user>
  # Pat someone for the lolz
+penguin
  # Gives you a random, lovely penguin
+ping
  # Responds with pong.
+poke <@user>
  # Poke someone for the lolz
+poll <question> | <duration> [| emojis...]
  # Creates a poll
+quote <quote> | <author>
  # Make a fake/real quote
+ratewaifu <something> --custom <NUMBERS>
  # Rates what you desire or your waifu <3
+remindme <reminder> in <[0 days] [00 hours] [00 minutes] [000 seconds]> | remove <text in reminder> | list
  # Set reminders.
+reverse <text>
  # Reverse something for the fun 🤪 nuf os
+roll [[min-]max]
  # Roll a number between the given range.
+rule34 <search>
  # rule34 stuff (NSFW)
+servergroup
  # Makes a picture with the server avatar surrounded with random user icons
+shared <user>
  # Get your information or someone else's
+ship <@user1> <@user2>
  # Ships 2 people
+slots
  # Le slot machine
+steam <info|sig> <steamid>
  # Get information about someone's steam
+support
  # Get the official support links
+tableflip <@user> [text]
  # Mention someone and make someone or yourself tableflip (WARNING: Long text)
+throw <@user>
  # Hit someone with an item like MarioKart (Some quotes are from Team Fortress 2)
+time <adress|ZIP code|city|country>
  # Get the time of the place you wish

```

Рисунок Б.2 – Друга частина відповіді на команду «help»

```

+urban <search>
  # Search on the ~urban~ wiki for the 'best' answers
+user <user>
  # Get your information or someone else's
+value <amount> <CODE> to <CODE>
  # Convert between currencies
+weather <city|country>
  # Get the weather of your request
+woop <text>
  # +woop

```

Рисунок Б.3 – Третя частина відповіді на команду «help»

## ДОДАТОК В

### РЕГУЛЯРНИЙ ВИРАЗ ДЛЯ РОЗБОРУ НЕНОРМАТИВНОЇ ЛЕКСИКИ

```

\w{0,5}[xx]([xx\s\!@#$$%^&*+-\|\/]{0,6})[yy]([yy\s\!@#$$%^&*+-
\|\/]{0,6})[ёілееюийя]\w{0,7}|\w{0,6}[пр]([пр\s\!@#$$%^&*+-
\|\/]{0,6})[іие]([іие\s\!@#$$%^&*+-
\|\/]{0,6})[Ззс]([Ззс\s\!@#$$%^&*+-
\|\/]{0,6})[дd]\w{0,10}|[ccs][yy]([yy\s\!@#$$%^&*+-
\|\/]{0,6})[4чкк]\w{1,3}|\w{0,4}[бб]([бб\s\!@#$$%^&*+-
\|\/]{0,6})[лл]([лл\s\!@#$$%^&*+-
\|\/]{0,6})[уя]\w{0,10}|\w{0,8}[её][бб][лскеёеыиаа][найёйвл]\w{0
,8}|\w{0,4}[ее]([ее\s\!@#$$%^&*+-
\|\/]{0,6})[бb]([бb\s\!@#$$%^&*+-
\|\/]{0,6})[уy]([уy\s\!@#$$%^&*+-
\|\/]{0,6})[н4ч]\w{0,4}|\w{0,4}[ееё]([ееё\s\!@#$$%^&*+-
\|\/]{0,6})[бb]([бb\s\!@#$$%^&*+-
\|\/]{0,6})[нn]([нn\s\!@#$$%^&*+-
\|\/]{0,6})[уy]\w{0,4}|\w{0,4}[ее]([ее\s\!@#$$%^&*+-
\|\/]{0,6})[бb]([бb\s\!@#$$%^&*+-
\|\/]{0,6})[ооaa@]([ооaa@\s\!@#$$%^&*+-
\|\/]{0,6})[тннт]\w{0,4}|\w{0,10}[ёе]([ёе\s\!@#$$%^&*+-
\|\/]{0,6})[б]\w{0,6}|\w{0,4}[пп]([пп\s\!@#$$%^&*+-
\|\/]{0,6})[иеei]([иеei\s\!@#$$%^&*+-
\|\/]{0,6})[дd]([дd\s\!@#$$%^&*+-
\|\/]{0,6})[ооaa@eeei]([ооaa@eeei\s\!@#$$%^&*+-
\|\/]{0,6})[pr]\w{0,12}

```



## ДОДАТОК Г

### ГРАММАТИКА СТРОКОВОГО КАЛЬКУЛЯТОРА

```

main_expression : expression {
    Result = $1;
}
;

expression      : md_expression {
    $$ = $1;
}
| expression Plus md_expression {
    $$ = $1 + $3;
}
| expression Minus md_expression {
    $$ = $1 - $3;
}
;

md_expression    : pow_expression {
    $$ = $1;
}
| md_expression Mult pow_expression {
    $$ = $1 * $3;
}
| md_expression Divide pow_expression {
    $$ = $1 / $3;
}
;

pow_expression   : f_expression {
    $$ = $1;
}
| pow_expression Pow f_expression {

```

```

        $$ = Math.Pow($1, $3);
    }
;

f_expression      : u_expression {
        $$ = $1;
    }
| uf_expression {
        $$ = $1;
    }
| df_expression {
        $$ = $1;
    }
;

uf_expression      : Sqrt u_expression {
        $$ = Math.Sqrt($2);
    }
| Ln u_expression {
        $$ = Math.Log($2);
    }
| Sin u_expression {
        $$ = Math.Sin($2);
    }
| Cos u_expression {
        $$ = Math.Cos($2);
    }
| Tangent u_expression {
        $$ = Math.Tan($2);
    }
| Cotangent u_expression {
        $$ = 1 / Math.Tan($2);
    }
| ModuleBracket u_expression ModuleBracket {
        $$ = Math.Abs($2);
    }

```

```

    }
    | u_expression Exclamations {
        $$ = RichardMath.Factorial($1);
    }
;

df_expression    : Log u_expression u_expression {
        $$ = Math.Log($3) / Math.Log($2);
    }
;

u_expression     : atom {
        $$ = $1;
    }
    | Plus atom {
        $$ = $2;
    }
    | Minus atom {
        $$ = -$2;
    }
;

atom             : Double {
        $$ = Convert.ToDouble($1,
CultureInfo.InvariantCulture);
    }
    | Int {
        $$ = Convert.ToDouble($1,
CultureInfo.InvariantCulture);
    }
    | Pi {
        $$ = Math.PI;
    }
    | Exp {
        $$ = Math.E;
    }

```

```
}  
| LeftBracket expression RightBracket {  
    $$ = $2;  
}  
| LeftBracketS expression RightBracketS {  
    $$ = $2;  
}  
;
```

## ДОДАТОК Д

### СЦЕНАРІЙ СТВОРЕННЯ БАЗИ ДАНИХ

```

CREATE TABLE [dbo].[Guild] (
    [Id]      BIGINT      NOT NULL,
    [Name]    NVARCHAR (MAX) NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC)
);

CREATE TABLE [dbo].[User] (
    [Id]      BIGINT      NOT NULL,
    [Name]    NVARCHAR (MAX) NOT NULL,
    [Type]    INT         NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [CK_User_Type] CHECK ([Type]>=(0) AND
[Type]<=(1))
);

CREATE TABLE [dbo].[Advertising] (
    [Id]      INT         IDENTITY (1, 1) NOT NULL,
    [Text]    NVARCHAR (MAX) NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC)
);

CREATE TABLE [dbo].[ApplicationLog] (
    [Id]      INT         IDENTITY (1, 1) NOT NULL,
    [UserId]  BIGINT      NOT NULL,
    [Time]    DATETIME    NOT NULL,
    [ApplicationName] NVARCHAR (MAX) NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_ApplicationLog_ToUser] FOREIGN KEY ([UserId])
REFERENCES [dbo].[User] ([Id])
);

CREATE TABLE [dbo].[BlockedCommand] (
    [Id]      INT         IDENTITY (1, 1) NOT NULL,
    [GuildId] BIGINT      NOT NULL,
    [CommandName] VARCHAR (MAX) NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_BlockedCommand_ToGuild] FOREIGN KEY
([GuildId]) REFERENCES [dbo].[Guild] ([Id])
);

CREATE TABLE [dbo].[BlockedModule] (
    [Id]      INT         IDENTITY (1, 1) NOT NULL,
    [GuildId] BIGINT      NOT NULL,
    [ModuleName] NVARCHAR (MAX) NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),

```

```

        CONSTRAINT [FK_BlockedModule_ToGuild] FOREIGN KEY
([GuildId]) REFERENCES [dbo].[Guild] ([Id])
);

CREATE TABLE [dbo].[Filter] (
    [Id] INT IDENTITY (1, 1) NOT NULL,
    [Filter] NVARCHAR (MAX) NOT NULL,
    [GuildId] BIGINT NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_Filter_ToGuild] FOREIGN KEY ([GuildId])
REFERENCES [dbo].[Guild] ([Id])
);

CREATE TABLE [dbo].[RoleConformity] (
    [Id] INT IDENTITY (1, 1) NOT NULL,
    [Role] NVARCHAR (MAX) NOT NULL,
    [RealRole] NVARCHAR (MAX) NOT NULL,
    [GuildId] BIGINT NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_RoleConformity_ToGuild] FOREIGN KEY
([GuildId]) REFERENCES [dbo].[Guild] ([Id])
);

CREATE TABLE [dbo].[Session] (
    [Id] INT IDENTITY (1, 1) NOT NULL,
    [StartTime] DATETIME NOT NULL,
    [EndTime] DATETIME NOT NULL,
    [UserId] BIGINT NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_Session_ToUser] FOREIGN KEY ([UserId])
REFERENCES [dbo].[User] ([Id])
);

CREATE TABLE [dbo].[UserToGuild] (
    [GuildId] BIGINT NOT NULL,
    [UserId] BIGINT NOT NULL,
    PRIMARY KEY CLUSTERED ([GuildId] ASC, [UserId] ASC),
    CONSTRAINT [FK_UserToGuild_ToGuild] FOREIGN KEY ([GuildId])
REFERENCES [dbo].[Guild] ([Id]),
    CONSTRAINT [FK_UserToGuild_ToUser] FOREIGN KEY ([UserId])
REFERENCES [dbo].[User] ([Id])
);

CREATE TABLE [dbo].[VoiceChannel] (
    [Id] BIGINT NOT NULL,
    [GuildId] BIGINT NOT NULL,
    [Name] NVARCHAR (MAX) NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_VoiceChannel_ToGuild] FOREIGN KEY ([GuildId])
REFERENCES [dbo].[Guild] ([Id])
);

```

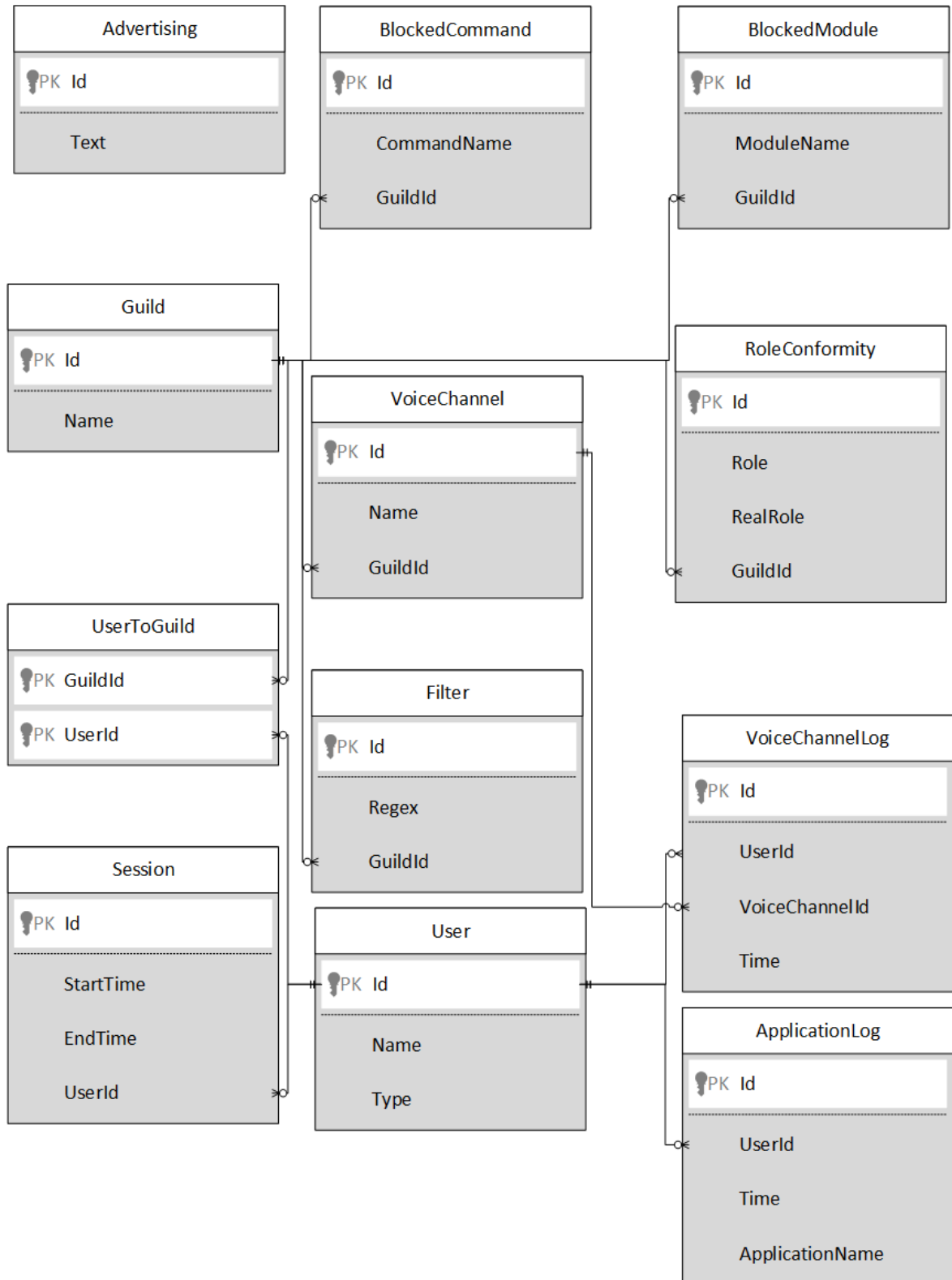
```

CREATE TABLE [dbo].[VoiceChannelLog] (
    [Id] INT IDENTITY (1, 1) NOT NULL,
    [UserId] BIGINT NOT NULL,
    [VoiceChannelId] BIGINT NOT NULL,
    [Time] DATETIME NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_VoiceChannelLog_ToUser] FOREIGN KEY
([UserId]) REFERENCES [dbo].[User] ([Id]),
    CONSTRAINT [FK_VoiceChannelLog_ToVoiceChannel] FOREIGN KEY
([VoiceChannelId]) REFERENCES [dbo].[VoiceChannel] ([Id])
);

```

## ДОДАТОК Е

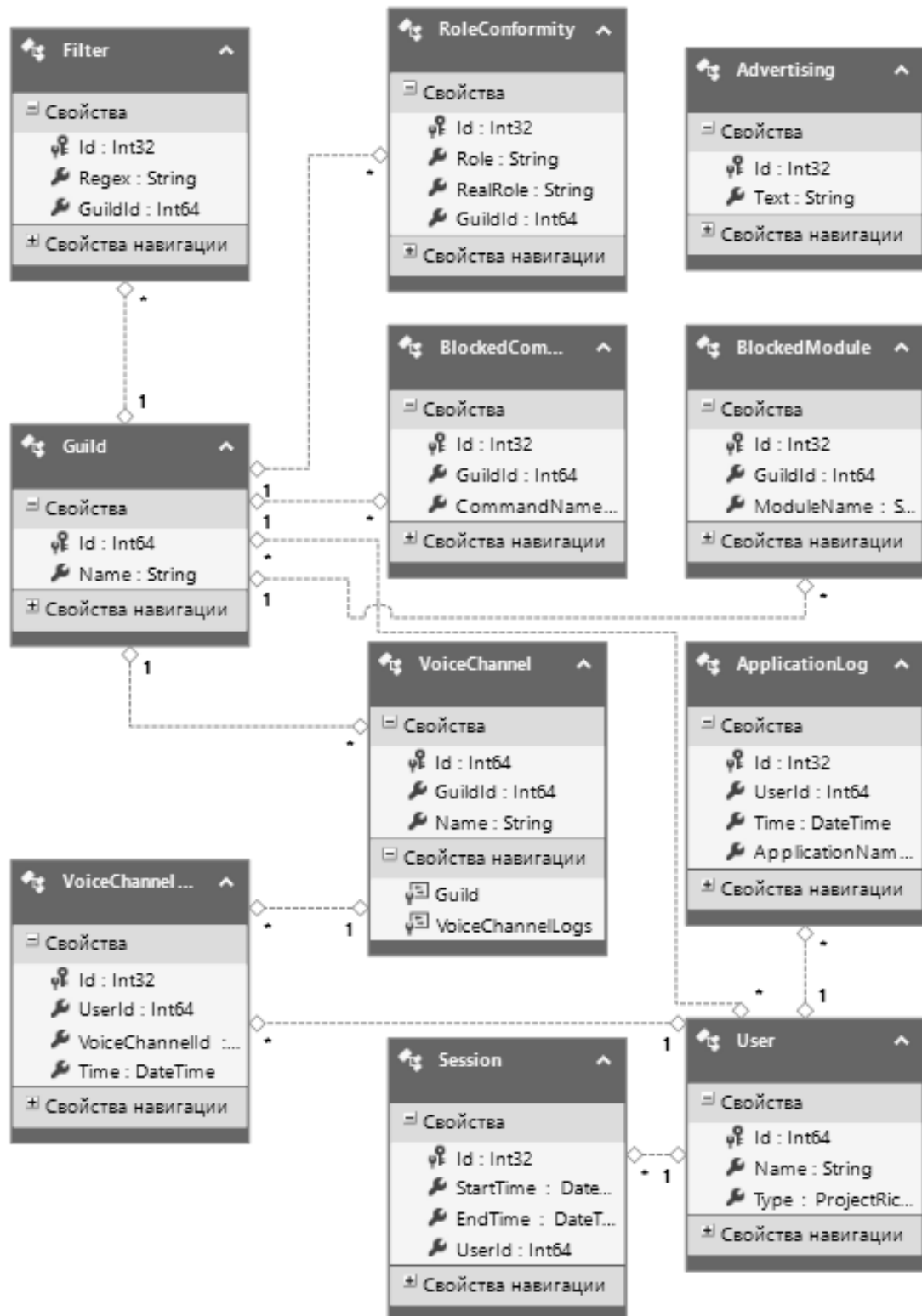
### ФІЗИЧНА СХЕМА БАЗИ ДАНИХ





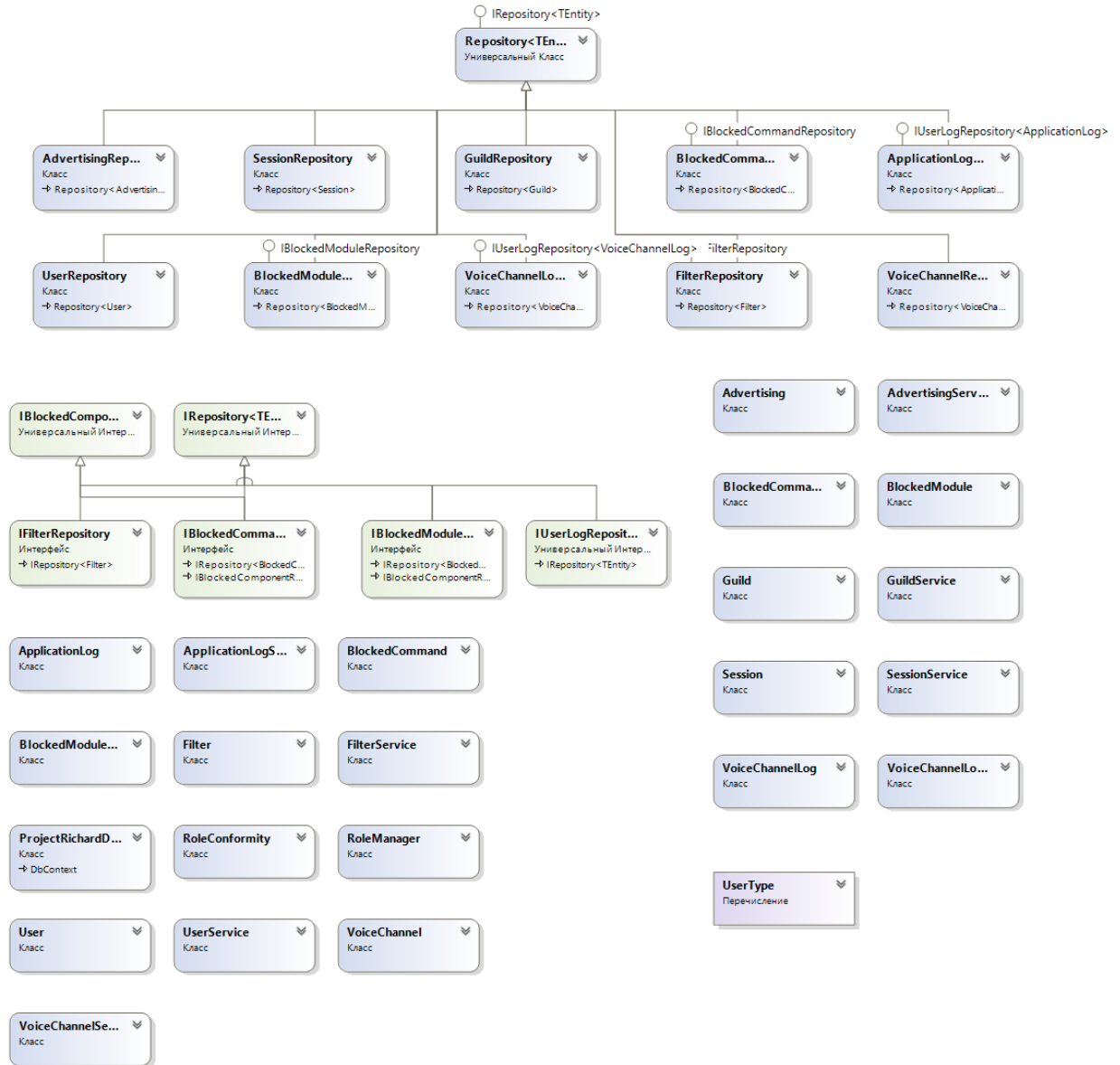
# ДОДАТОК Ж

## ОБ'ЄКТНА СХЕМА БАЗИ ДАНИХ

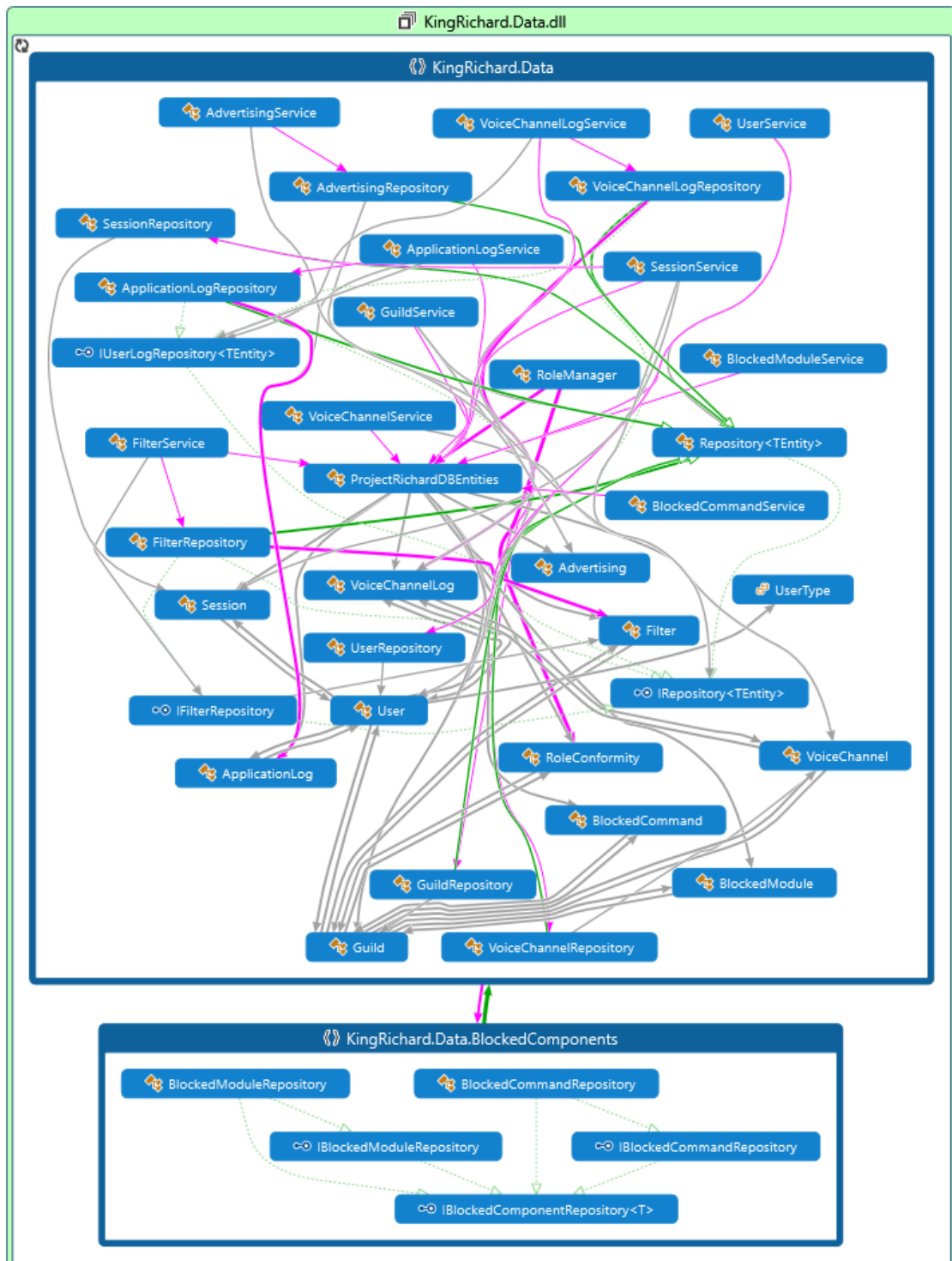


## ДОДАТОК И

### ДІАГРАМА КЛАСІВ ПРОЕКТУ KINGRICHARD.DATA

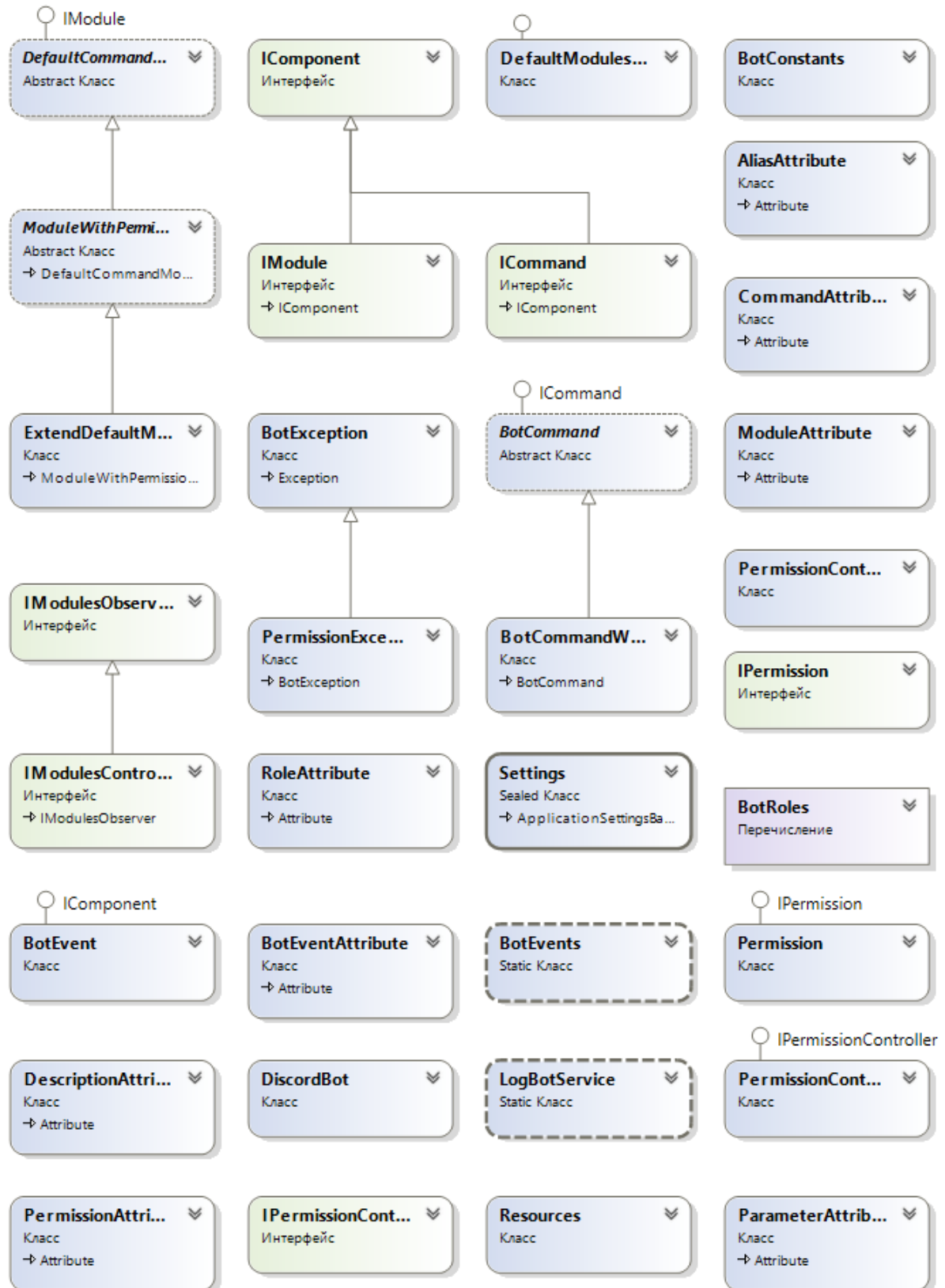


ДОДАТОК К  
ДІАГРАМА ЗАЛЕЖНОСТЕЙ ПРОЕКТУ KINGRICHARD.DATA



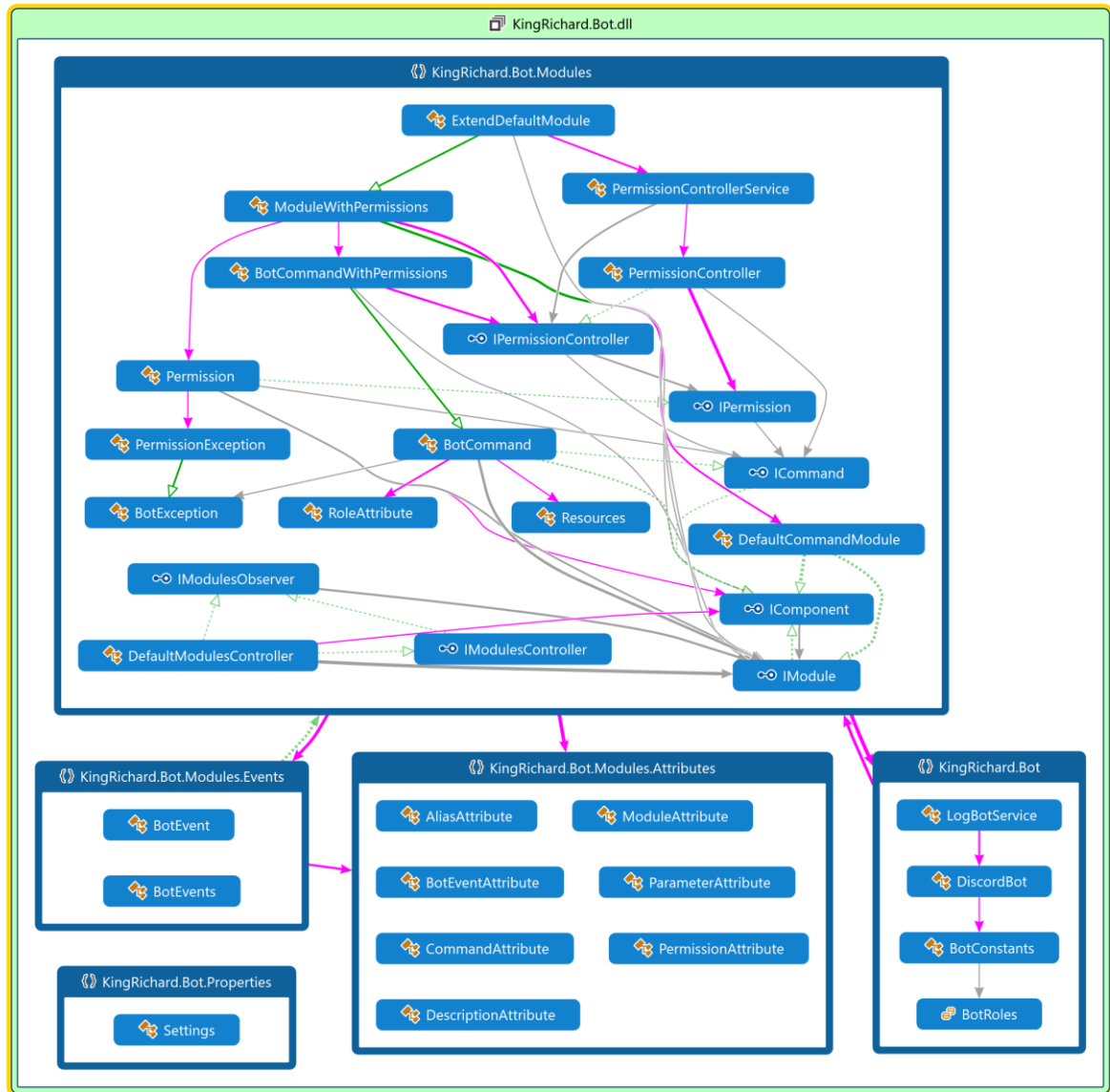
## ДОДАТОК Л

### ДІАГРАМА КЛАСІВ ПРОЕКТУ KINGRICHARD.BOT



# ДОДАТОК М

## ДІАГРАМА ЗАЛЕЖНОСТЕЙ ПРОЕКТУ KINGRICHARD.BOT



## ДОДАТОК Н

### ВИКОРИСТАННЯ KINGRICHARD.BOT

Приклад створення модуля, з ім'ям `MathModuleName`, подгруженого з `CommandsConstants`, з командою, що має ім'я `CommandsConstants.CalculateCommandName`, але асом `CommandsConstants.CalculateCommandAlias` і одним вхідним параметром `CommandsConstants.Expression` типу `ParameterType.Unparsed`, так само в оголошенні створення команди і модуля міститься опис, котрий допоможе користувачеві розібратися в команді:

```
[Module (CommandsConstants.MathModuleName) ]
[Description (CommandsConstants.MathModuleDescription) ]
    public class MathModule : ExtendDefaultModule
    {
        [Command (CommandsConstants.CalculateCommandName) ,
Description (CommandsConstants.CalculateCommandDescription) ]
        [Alias (CommandsConstants.CalculateCommandAlias) ]
        [Parameter (CommandsConstants.Expression,
ParameterType.Unparsed) ]
        public async Task Calculate (CommandEventArgs e)
        {
            //реалізація команди
        }
        // інші елементи класу
    }
```

Приклад створення події з ім'ям `CommandsConstants.AddGuildEventName`, яке відбудеться під час додавання нового сервера.

```
[Module (CommandsConstants.SettingsModuleName) ]
```

```

[Description(CommandsConstants.SettingsModuleDescription)]
    public class SettingsModule : ExtendDefaultModule
    {
        ...

        [BotEvent(CommandsConstants.AddGuildEventName,
BotEvents.JoinedServer)]
        public void AddGuild(object sender, ServerEventArgs
args)
        {
            //реалізація події
        }
        // інші елементи класу
    }

```

Приклад дозволу, при порушенні якого, буде виведено повідомлення **CommandsConstants.RolePermissionMessage**.

```

[Module(CommandsConstants.RolesModuleName)]
[Description(CommandsConstants.RolesModuleDescription)]
    public class RolesModule : ExtendDefaultModule
    {

[Permission(CommandsConstants.RolePermissionMessage)]
        public bool CheckRole(CommandEventArgs args,
ICommand command)
        {
            //реалізація дозволу
        }
        // інші елементи класу
    }

```

## ДОДАТОК П

## ВИХІДНИЙ КОД КЛАСУ PROGRAM ПРОЕКТУ KINGRICHARD.CUI

```
using KingRichard.Bot;
using KingRichard.Bot.Modules;
using KingRichard.Bot.Modules.WordFilter;
using KingRichard.Bot.Modules.Statistic;
using KingRichard.Bot.Modules.Advertising;
using KingRichard.Bot.Modules.Moderation;

namespace KingRichard.CUI
{
    class Program
    {
        static void Main(string[] args)
        {
            DiscordBot bot = LogBotService.CreateBot();

            bot.BotToken = Properties.Settings.Default.BotToken;

            bot.Modules.Add(new StatisticModule());
            bot.Modules.Add(new SettingsModule(bot.Modules));
            bot.Modules.Add(new MathModule());
            bot.Modules.Add(new FilterModule());
            bot.Modules.Add(new AdvertisingModule());
            bot.Modules.Add(new ModerationModule());

            bot.Start();
        }
    }
}
```



## ДОДАТОК Р

### ВІДГУКИ КОРИСТУВАЧІВ

Відгук звичайного користувача:

«Я, находилась на сервере Discord где функционировал бот Richard. Очень понравилась функция, которая блокировала сообщения с нецензурной лексикой. Так же идея с модулями очень интересная, в других ботах такого не видел. Не понравилась реклама, которая всплывала некоторое время. В перспективе было бы неплохо добавить больше развлекательного контента»

Відгук модератора:

«Я, пользуюсь ботом "Richard" для приложения "Discord" уже в течении месяца. В данном боте я часто использую команды -say и -clear, т.к. первая является довольно-таки забавной озвучкой текста, а вторая помогает содержать текстовый чат в надлежащем виде, удаляя лишние сообщения пользователей или других ботов с помощью одной команды, вместо ручного удаления каждого сообщения. Так же данный бот позволяет менять его работу "на ходу", благодаря модулям, что вполне удобно. Тем не менее, в данном боте можно реализовать и другие модули, например, прослушивание музыки из того же YouTube, или каких-то других сервисов, доступных для онлайн-прослушивания музыки, что лично для меня является крайне полезной функцией. Функционалом бота на данный момент удовлетворён»

Відгук адміністратора:

«Я, являюсь администратором сервера «20-17», месяц назад добавил бота на свой сервер, после чего его модерация стала намного удобнее, так как намного проще и быстрее написать короткую команду в чате, чем искать те же функции в графическом интерфейсе Discord, очень удобно что есть система ролей, которая позволила настроить модераторов на сервере. Не понравилось, что бот имеет слишком много прав»

## ДОДАТОК С

### ПРИКЛАДИ SQL ЗАПИТІВ ДО ЗІБРАНОЇ СТАТИСТИКИ

**Запит для аналізу найпопулярнішого голосового чату:**

```
SELECT [Guild].Name AS 'Guild name', [VoiceChannel].Name AS
'Voice channel name',
      ChannelLogList.Count AS 'Count'
FROM (SELECT MAX([VoiceChannelLog].VoiceChannelId) AS 'Id',
      COUNT([VoiceChannelLog].VoiceChannelId) AS 'Count'
      FROM [VoiceChannelLog]
      GROUP BY [VoiceChannelLog].VoiceChannelId) AS
ChannelLogList
      JOIN [VoiceChannel] ON ChannelLogList.Id =
[VoiceChannel].Id
      JOIN [Guild] ON [VoiceChannel].GuildId = [Guild].Id
      ORDER BY [Guild].Name, ChannelLogList.Count DESC
```

**Запит для аналізу часу перебування користувачів у додатку Discord:**

```
SELECT MAX(tDiff.Name) AS 'name',
convert(TIME,dateadd(ms,MAX(tDiff.Diff)*1000,0),114) AS 'max',
convert(TIME,dateadd(ms,MIN(tDiff.Diff)*1000,0),114) AS 'min',
convert(TIME,dateadd(ms,AVG(tDiff.Diff)*1000,0),114) AS 'avg'
      FROM (SELECT Name, StartTime, EndTime, DateDiff(ss,
StartTime, EndTime) as 'Diff'
      FROM [User] JOIN [Session] ON [User].Id =
[Session].UserId) AS tDiff
      GROUP BY tDiff.Name
```