TataruBot

# Main Objective:

TataruBot is a discord bot designed to automate the process of administration on the discord server Cleric’d. The bot will have the following features:

1. Automatically assigns new members to default role.
2. Dynamically grand and revoke access to individual text chat channels dedicated to individuals in an associated voice channel and purge the channel of all content once the voice channel is empty for a certain period.
3. Allow members to assign themselves to in game roles so they can be tagged when members are looking for specific roles for content.
4. Change the server location based on a text command when connection to sever becomes weak (admin only).
5. Store historical information about members to track activity, flag inactive members to be removed.
6. Store and retrieve resources for raiding, including guides, documents, and triggers.
7. Store and retrieve random chat commands for fun (copypastas and memes)
8. Create a loot management system for the raid team
   1. Keep track of loot added
   2. Retrieve loot history for pieces
   3. Suggest loot distribution
9. Any other features requested by users.

# Technology Used:

The bot will be written in C# using the Discord.Net library. In additional, AWS will be used for databases and for storage purposes. SQL will be used to send and retrieve requests to the database.

# Core Project Setup

* Main (executes the bot)
* BotClient (features event processing)
* S3 Module
  + S3Client (connects to AWS S3 to process data)
  + S3Connector (Basic independent async library for collecting and retrieving data
* SQL Module
  + SQLClient (process data from the database)
  + SQLConnector (Independent connector sending and retrieving requests)

# Milestones:

**Version 1.0:** (To be completed by September 3rd, 2019) COMPLETE

1. Automatically assigns new members to default role.
2. Dynamically grand and revoke access to individual text chat channels dedicated to individuals in an associated voice channel and purge the channel of all content once the voice channel is empty for a certain period.
3. Allow members to assign themselves to in game roles so they can be tagged when members are looking for specific roles for content.
4. Change the server location based on a text command when connection to sever becomes weak

Version 1.1: (To be completed by September 16th, 2019) Store historical information about members to track activity, flag inactive members to be removed.

1. Store and retrieve resources for raiding, including guides, documents, and triggers.
2. Store and retrieve random chat commands for fun (copypastas and memes)

Future Releases:

1. Create a loot management system for the raid team
   1. Keep track of loot added
   2. Retrieve loot history for pieces
   3. Suggest loot distribution
2. Any other features requested by users.

# Detailed Implementation Plan

## Automatic Assignment of New Members to Default Role

The role of “Member” will be assigned to anyone who joins the server. This role ID will be stored in a local json document and loaded in before the bot is launched.

## Dynamic Assignment to Chat focused Voice Channel and Channel Purge

This feature will use the UserVoiceStateUpdated event. This event will provide two objects, the user voice state before and after their state has changed. The ID of the channel values before and after will be captured (null if no voice channel). The ID of the channel before the state change will be used to remove the associated rank from the member. The ID of the channel after the state change will be used to add the associated rank to the member. The associated ranks will be saved in the form of json values within a local configuration file.

## Dynamic Role Assignment Based on Reactions

This feature will use the ReactionAdded and ReactionRemoved event. It will parse the associated message, should the message ID match the ID of a provided value through configs. If the ID of the message matches, it will check the ID of the emoji used. If the emoji matches a provided internal list, it will assign the member with the associated rank. The associated rank will be provided via a local json configuration file.

Roles:

1. Healer
2. Tank
3. Ranged DPS
4. Melee DPS
5. Caster DPS

## Dynamic Voice Server Relocation

Upon detecting the message @TataruBot change server, the server location will be changed. It will cycle from us-west, us-east, us-central, and us-south. These values can be provided in the command to change to a specific server.

## Database Access for fight guides, videos, triggers, and timelines

Use SQL commands to match with existing text commands to pull from database.

Discord commands:

* @TataruBot get me <fight> <resources>
  + <fight> will be any of these values:
    - e1s
    - e2s
    - e3s
    - e4s
    - ucob
    - uwu
    - Any new savage/ultimate fights
  + <resources> can be any combination of the following:
    - timeline
    - guide
    - triggers
  + “get me” is optional
* @TataruBot add <fight> <resources> <url>
  + Can only be triggered in admin channel
  + “add” is required