**PROJECT PROPOSAL**

**TU Bot, A Discord Bot for Professors & Students**

Table of Contents

[Project Abstract 3](#_Toc440365550)

[High Level Requirement 3](#_Toc440365551)

[Conceptual Design 3](#_Toc440365552)

[Background 3](#_Toc440365553)

[Required Resources 4](#_Toc440365554)

## Project Abstract

This project proposes a practical implementation of a multi-purpose Discord bot with the intended use for professors and students to effectively communicate and use meaningful tools in the Discord application. With the hardest parts of quarantine behind us, many students and professors have created Discord accounts and Discord servers to communicate effectively with each other. In the online semesters, many students have picked up the platform and now use it in their everyday life for not just school – as Discord has created a social platform for students to create friendships and properly communicate and collaborate on schoolwork. TU Bot is a concept that strives to be used in all Temple University affiliated servers, with a large scope of helpful tools and commands that would be used for professors and students so that they can make the Discord platform even more convenient for them. Meaningful tools include things like appointment schedulers, academic reminders, categorized/organized notifications, group creators, collaboration toolkits etc. If development time allows it, the bot will strive to web-scrape Canvas for even more uses, perhaps with real-time notifications from Canvas.

## High Level Requirement

This bot requires the user to install the Discord application, create a Discord account, and create a Discord server. The user may use an already existing Discord account or Discord server.

Discord specifically requires Windows 7+, or MacOS 10.11+ on computer devices. On mobile devices, Discord specifically requires iOS 10.0+, or Android 5+. In terms of web browsers, Discord can run on Google Chrome, Firefox 80+, Opera, Microsoft Edge 17+, Safari 11+. A working, stable internet connection is required to utilize all of Discord’s features.

Given an appropriate URL link, any user can add the TU Bot into their Discord server. The user would have to access the URL link and invite the bot to their server. This will give the bot its required permissions to carry out its capabilities without issue. When TU Bot has joined a server, it will host a variety of helpful commands that the user can access with ‘/’ commands in the chat box.

## Conceptual Design

The bot will have a help menu that can be accessed when prompted by a user in the chat box. From there the user can see a full list of commands that can be used by inputting a ‘/’ as an indicator that the bot is taking in a command. These commands will serve many possibilities to make the lives of the student, professor and TA more convenient. This bot is meant to be expanded upon – the list of commands can be added onto as long as implementation allows.

TU Bot, like other Discord bots, are hosted on a user-provided entity. A spare device may have to be used to host the bot 24/7, or perhaps a reliable cloud service. Discord.js is utilized as a foundation for the bot. The entire bot is built using JavaScript. A majority of Discord bots are built on Discord.js, or another official Discord framework. TU Bot will also be built on Discord.js, but it is not open source as it only uses the bare-bones foundation of Discord.js to branch out into its own idea like all other Discord bots.

## Proof of Concept

<https://github.com/cis3296s22/concept-athankim>

## Background

In Discord, bots are the second type of “user” that act as participants in a Discord server. They have become extremely popular, both for the general public who use Discord, and developers who use Discord. Discord bots have the capabilities of responding to user inquiries, web-scraping for useful information, staying up to date in real time with data, etc. The list goes on.

<https://hydra.bot/>

The URL above is the official website for one of the most popular Discord bots, which is used to play music for participating users in a voice chat channel. Although this may seem like a simple feature, the bot features a myriad of helpful commands that accommodate a music player.

<https://gamblingbot.app/home>

The URL above is the official website for another popular Discord bot, which is used to play fun casino minigames with other members of your server, or with yourself. It comes with a sleek and user-friendly UI, along with helpful and easy to understand commands.

The TU Bot does not seek to become a music bot, or a gambling bot. It seeks to become a universally useful tool for any professor or student who hopes to use the Discord application for academic needs at Temple University.

## Required Resources

From the start, anything I need has already been included in the GitHub repository with the importation of Discord.js. In the future, web-scraping may need to be used in the bot to implement helpful features. All of this is possible within the scope of JavaScript.

As for development, all development will be done on the developer’s personal computers. In the future, for stability and longevity, a separate hosting device or cloud server may need to be used/purchased to keep the bot online 24/7.