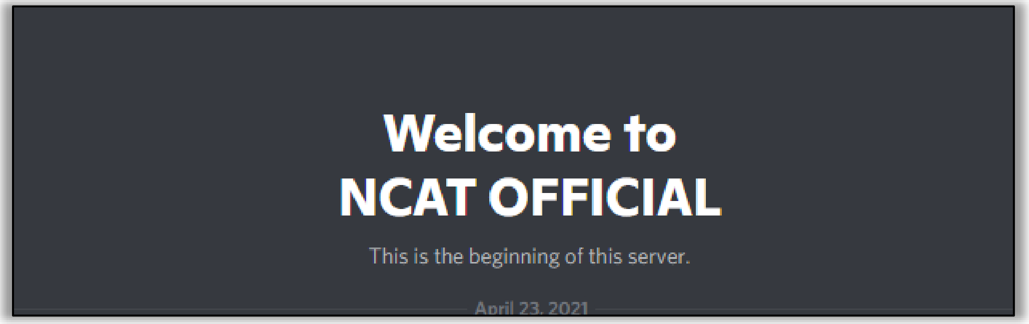
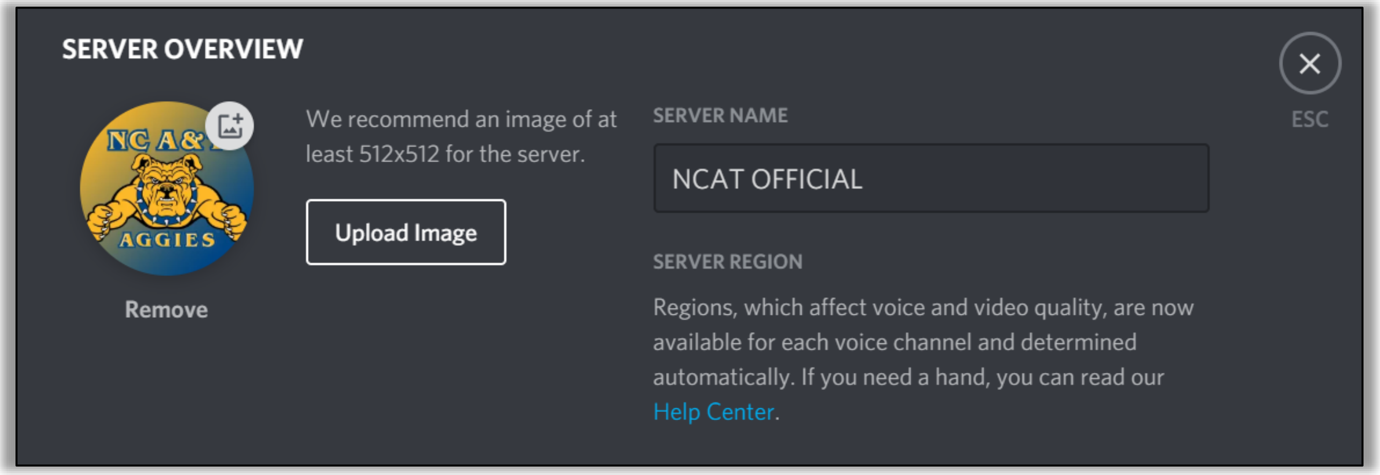
North Carolina
Agricultural and Technical
State University

DISCORD


Welcome to
NCAT OFFICIAL

This is the beginning of this server.

April 23, 2021



We recommend an image of at least 512x512 for the server.

SERVER NAME

NCAT OFFICIAL

SERVER REGION

Regions, which affect voice and video quality, are now available for each voice channel and determined automatically. If you need a hand, you can read our Help Center.

Department of Computer Systems Technology, North Carolina Agricultural and Technical University

Team 1 NCAT Discord Server Final Project Report

Table of Contents

1. TEAM RESUMES 2

2. TEAM POSTER 0

3. WEEKLY JOURNALS 1

3.1 PROJECT OVERVIEW (PROBLEM STATEMENT) 33

4. PROJECT DESCRIPTION 33

4.1 PRODUCT CONTEXT 33
4.2 USER CHARACTERISTICS 34
4.3 ASSUMPTIONS 34
4.4 CONSTRAINTS 34
4.5 DEPENDENCIES 34

5. REQUIREMENTS 34

1.1 USER INTERFACE REQUIREMENTS 34
1.2 USABILITY 35
1.3 PERFORMANCE 35
1.3.1 Capacity 35
1.3.2 Availability 35
1.4 MANAGEABILITY/MAINTAINABILITY 36
1.4.1 Monitoring 36
1.4.2 Maintenance 36
1.4.3 Operations 36
1.5 SYSTEM INTERFACE/INTEGRATION 36
1.5.1 Systems Interfaces 36
1.6 SECURITY 38
1.6.1 Protection 38
1.6.2 Authorization and Authentication 39
1.7 DATA MANAGEMENT (SAGE) 39
1.8 STANDARD COMPLIANCE 40

6. SPECIFICATIONS/IMPLEMENTATION PLAN/FLOW CHART 41

1.9 USER FLOW CHART 41
1.10 UML DIAGRAM 42
1.11 PROJECT DESIGN STEPS 43
1.12 SOFTWARE CODE 44

7. LEARNING STATEMENTS 56

1. Team Resumes

HAU LE

336-615-0318
Hvle1@aggies.ncat.edu

To obtain an internship position with a company where I can utilize my various skills to assist in meeting the goals of the business to ensure proper organization and satisfaction.

EXPERIENCE

05/2014 – 05/2015

ELEVATOR TECHNICIAN, VIETNAMESE THIEN NAM ELEVATOR JOINT STOCK Examine, maintain, and repair elevators as well as other conveyance types.

07/2016 – 05/2019

TUTOR, GREENSBORO, NC

Helped a section of 5 middle school students on better understanding Math functions and their uses. Volunteered to meet with students on a 1-to-1 basis helping them with problems on individual and team projects. Teaching Vietnamese for our community.

06/2016 – 01/2021

RECEPTIONIST-CASHIER, ELITE NAIL DAY SPA

Managed phone calls, scheduled appointments, and fairly distributed work among the employees. Consulted with customers on options for service and coverage, evaluated their needs, and determined the best options for them. Assisted with operations, including overall register and banking documents. Prepared bi-weekly payroll for employees.

05/2014 – 05/2015

NATIONAL HONOR SOCIETY, GREENSBORO, NC

Determine the need for and perform a major overhaul or replacement of defective equipment. Perform proper timekeeping and accounting of materials and expenses for assigned projects. Read and interpret blueprints. Follow appropriate maintenance and service procedures. Meet and assist with third-party municipal inspections.

EDUCATION

AUGUST 2017

ENGINEERING, GUILFORD TECHNICAL COMMUNITY COLLEGE

Associates of Engineering, Information Technology Major Cumulative.
GPA: 3.4

DECEMBER 2021

INFORMATION TECHNOLOGY, NORTH CAROLINA AGRICULTURE & TECHNICAL STATE UNIVERSITY

Computing and Security, Software Engineering, Electric Circuits, Web Design, Networking, Computer Database, Introduction to Linux, Project Management, and Operating System. GPA: 3.87

COMPUTER SKILLS

- Microsoft Office
- Database
- Python & Java
- Computer Hardware
- HTML/CSS

PROJECTS

Researched and analyzed the accounting information system of Tennis Shooting Machine.
Building and developing a boat using solar power with the college team.

RYAN BLOUNT

919-621-9018
Rdblount1@aggies.ncat.edu

Career Objective: I want to contribute to every company I possibly can. I have worked in retail and also have worked in an office environment for the state of North Carolina. I have learned many different selling techniques from working in retail as well as being able to work with many different personalities.

EXPERIENCE

05/2019 – 08/2019

SUMMER INTERN, NORTH CAROLINA DEPARTMENT OF COMMERCE – DIVISION OF EMPLOYMENT SECURITY

Tasks included but were not limited to creating new activity logs, rewriting, and updating procedures, data entry for the NCUI 604 form.

05/2018 – 08/2018

SUMMER INTERN, NORTH CAROLINA DEPARTMENT OF COMMERCE – DIVISION OF EMPLOYMENT SECURITY

Tasks included data entry for employee training. The task required accuracy promptly.

05/2017 – 08/2017

SUMMER INTERN, NORTH CAROLINA DEPARTMENT OF COMMERCE – DIVISION OF EMPLOYMENT SECURITY

Tasks included creating a database to be used to keep track of employees' training upon completion. **03/2016 – 08/2016**

FLOOR STAFF, REGAL ENTERTAINMENT GROUP CROSSROADS STADIUM 20

Tasks included working as an usher and operating the concession stand area. Ushers were to keep the theater clean and neat while offering customer service when needed.

EDUCATION

DECEMBER 2021

INFORMATION TECHNOLOGY, NORTH CAROLINA AGRICULTURE & TECHNICAL STATE UNIVERSITY

The course work includes computer networking, computer software and hardware support, disk storage technology, mainframe system administration, network security, information management, mobile computing, and other areas that utilize computers as the main component of the system.

MAY 2016

HS DIPLOMA, HOLLY SPRINGS HIGH SCHOOL

SKILLS

- Communication
- Efficient in computer systems
- Respectful
- Teamwork
- Trustworthiness
- Able to work under stressful conditions

COMPUTER SKILLS

- Microsoft Office
- Python & Java
    - Visual Studio
    - Adobe Photoshop
- Davinci Resolve
- HTML/CSS

SAGE-MICHAEL BROWN

336-558-0564
Sebrown3@aggies.ncat.edu

I am a very passionate person when it comes to computers and testing/pushing their limits, whether it's overclocking a CPU to run twice as fast or using multiple copies to run a singular application to pressure test it. Computers are the basis on which I was grown and how I was raised because without them we will not be as advanced as we are.

EXPERIENCE

01/2019 – 01/2020

COMPUTER TECH AND SALES, PERSONAL BUSINESS

Built and sold computer parts that I gather from different sources. Sold Custom built fully functional pcs to buyers, whether it was for gaming needs or professional use.

EDUCATION

DECEMBER 2021

INFORMATION TECHNOLOGY, NORTH CAROLINA AGRICULTURE & TECHNICAL STATE UNIVERSITY

Currently maintaining a GPA of 3.85, all if not most of my coursework has encompassed a form of programming through command prompt-based applications GPA: 3.85

JUNE 2016

HS DIPLOMA, JAMAICA COLLEGE

Held a constant 3.4 GPA and was awarded Badminton player of the year for 2015. My course work mainly consisted of science-based topics, nothing that was targeted towards my current career but, they still help in terms of excelling and having information on topics that would seem foreign to me. GPA: 3.4

SKILLS

- Great communication and teamwork
- Efficient in computer systems
- Hardworking & Goal Oriented

COMPUTER SKILLS

- Microsoft Office
- Python & Java
- Electrical Systems
- SQL and Database Knowledge
- Computer hardware certified
- HTML/CSS

ACTIVITIES

In my spare time, I love to give back whether it is by donating to shelters or volunteering at the closest food drive. I played badminton professionally for the past 8 years, while also juggling robotics and cadets at the same time. After getting Hardware certified I ran a small computer tech out of my room, in which I earned money for a couple of months, and became more and more fond of them, with that I also donated some tablets to a small school near where I lived in Jamaica to help as much as I could.

TERRELL PINKNEY

910-884-6263
Tspinkney@aggies.ncat.edu

Senior Information technology undergraduate student actively looking for internship opportunities in Technical Consulting, Data Analyst, Business Analyst, Network Security, and Sales Engineering positions. Highly interested in Rotational and Development programs to gain experience in a variety of areas and to collaborate with different team members.

EXPERIENCE

05/2021 – 08/2021

TECDP INTERN, Cigna & Express Scripts

Project coordination Track execution of project activities Document processes Facilitate meetings Create communications Create PowerPoint presentations Create relationships with project stakeholders. Maintain the organizational structure of the SharePoint site

EDUCATION

DECEMBER 2021

INFORMATION TECHNOLOGY, NORTH CAROLINA AGRICULTURE & TECHNICAL STATE UNIVERSITY

GPA: 3.92
Honors & Awards: Chancellor's List (2018-2020)
Clubs: Electronics Club (2019-Present)

MAY 2017

COMPUTERS TECHNOLOGY CERTIFICATE, FAYETTEVILLE TECHNICAL COMMUNITY COLLEGE

Held a constant 3.4 GPA and was awarded Badminton player of the year for 2015. My course work mainly consisted of science-based topics, nothing that was targeted towards my current career but, they still help in terms of excelling and having information on topics that would seem foreign to me. GPA: 3.4

MAY 2017

HIGH SCHOOL DIPLOMA, JACK BRITT HIGH SCHOOL

SKILLS

- Python and Java
- HTML/CSS
- Applicable use in Virtual Machines with Linux, macOS, and Windows Operating Systems

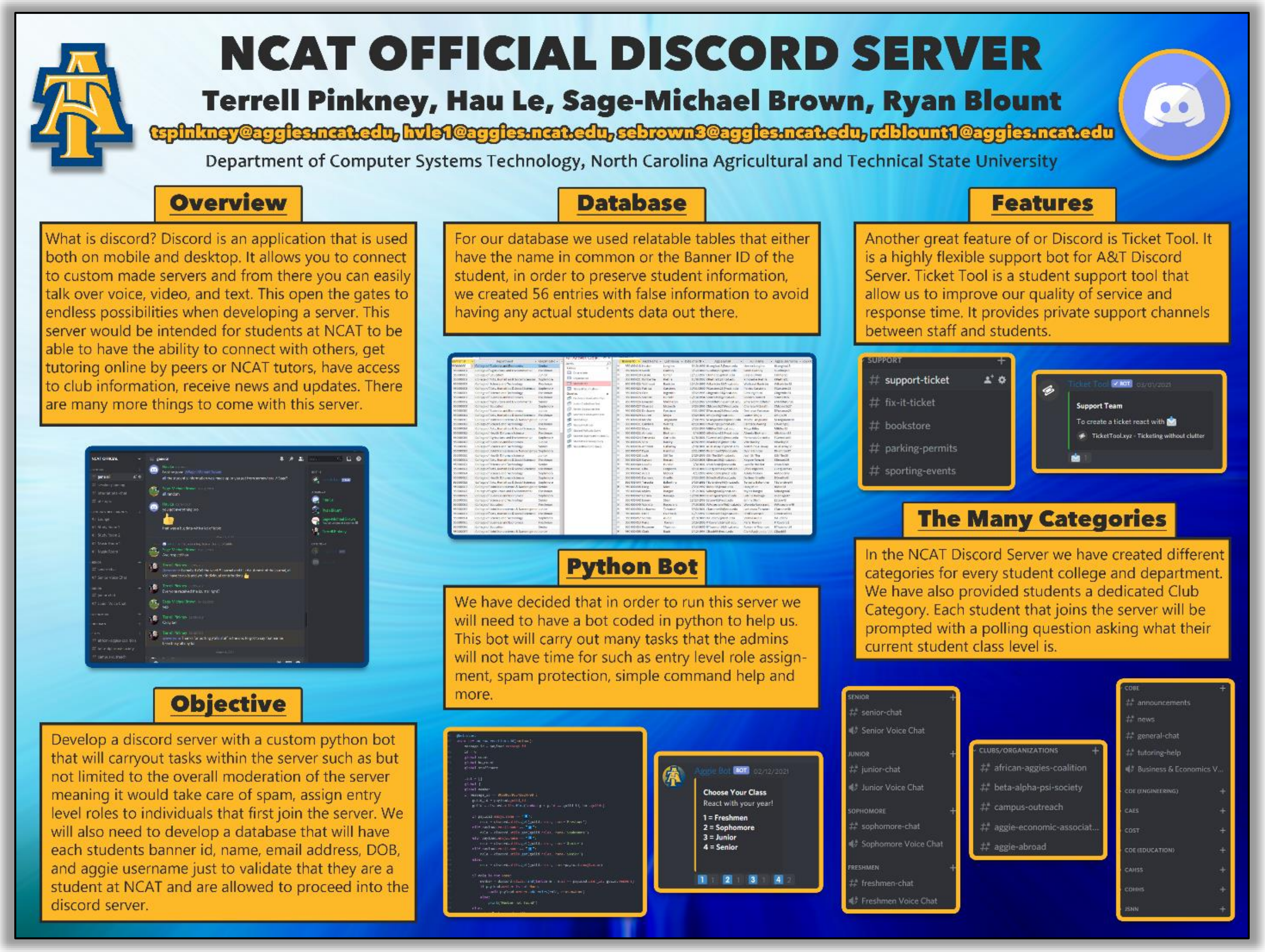
RELEVANT COURSEWORK

Courses: HTML and Web Programming, Introduction to Python Programming, Network Operating Systems, Introduction to Java Programming, Computer Database Management, Network Mainframe Operations, Computer Networking

CERTIFICATIONS AND BADGES

IBM Master the Mainframe 2020 – Level 2, Issued November 15, 2020

2. Team Poster



NCAT OFFICIAL DISCORD SERVER

Terrell Pinkney, Hau Le, Sage-Michael Brown, Ryan Blount
tspinkney@aggies.ncat.edu, hvle1@aggies.ncat.edu, sebrown3@aggies.ncat.edu, rdblount1@aggies.ncat.edu
Department of Computer Systems Technology, North Carolina Agricultural and Technical State University

Overview

What is discord? Discord is an application that is used both on mobile and desktop. It allows you to connect to custom made servers and from there you can easily talk over voice, video, and text. This open the gates to endless possibilities when developing a server. This server would be intended for students at NCAT to be able to have the ability to connect with others, get tutoring online by peers or NCAT tutors, have access to club information, receive news and updates. There are many more things to come with this server.

Objective

Develop a discord server with a custom python bot that will carryout tasks within the server such as but not limited to the overall moderation of the server meaning it would take care of spam, assign entry level roles to individuals that first join the server. We will also need to develop a database that will have each students banner id, name, email address, DOB, and aggie username just to validate that they are a student at NCAT and are allowed to proceed into the discord server.

Database

For our database we used relatable tables that either have the name in common or the Banner ID of the student, in order to preserve student information, we created 56 entries with false information to avoid having any actual students data out there.

Python Bot

We have decided that in order to run this server we will need to have a bot coded in python to help us. This bot will carry out many tasks that the admins will not have time for such as entry level role assignment, spam protection, simple command help and more.

Aggie Bot BOT 02/12/2021

Choose Your Class
React with your year!

1 = Freshmen
2 = Sophomore
3 = Junior
4 = Senior

Features

Another great feature of or Discord is Ticket Tool. It is a highly flexible support bot for A&T Discord Server. Ticket Tool is a student support tool that allow us to improve our quality of service and response time. It provides private support channels between staff and students.

Ticket Tool BOT 03/01/2021

Support Team
To create a ticket react with 📩
TicketTool.xyz - Ticketing without clutter

The Many Categories

In the NCAT Discord Server we have created different categories for every student college and department. We have also provided students a dedicated Club Category. Each student that joins the server will be prompted with a polling question asking what their current student class level is.

3. Weekly Journals

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney
Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: SEPTEMBER 3, 2021 2

1.1 ACCOMPLISHMENT 1: TESTED STUDENT VERIFICATION 2
1.2 ACCOMPLISHMENT 2: CREATED GITHUB ACCOUNTS 2
1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 2
1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 2
1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 2
1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 2
1.7 ISSUES AND SOLUTIONS 2
1.8 GOALS FOR NEXT WEEK 2
1.8.1 Goal 1: Upload current project files on the online repo 2
1.8.2 Goal 2: Learn about Git Bash Commands 2

1 Week Ending: September 3, 2021

1.1 *Accomplishment 1: Tested student verification*

We tested the verification process for when a new student enters the server. Testing was done to determine if there were any issues and bugs with the backend sample of student data in our SQL database.

1.2 *Accomplishment 2: Created GitHub accounts*

Each team member was tasked to create a GitHub account to further project development on an online repo. The online repo will allow each team member to work on different project files simultaneously.

1.3 *Individual Group Member Contribution: Hau Le*

Created a GitHub account for the future scope of the project.

1.4 *Individual Group Member Contribution: Ryan Blount*

Created the online GitHub Repo that will be used for uploading and accessing project files. Also created a GitHub account for the future scope of the project.

1.5 *Individual Group Member Contribution: Sage-Michael Brown*

Created a GitHub account for the future scope of the project.

1.6 *Individual Group Member Contribution: Terrell Pinkney*

Created a GitHub account for the future scope of the project.

1.7 *Issues and Solutions*

- Issues
    - The only issue that the team ran into was getting each team member access to the repo when it was initially created.
- Solutions
    - Each team member had to create a GitHub account to receive access to the online repo.

1.8 *Goals for Next Week*

Our goals for next week are to make sure that each team member has access to the PostgreSQL database, install Microsoft Visual Studio Code, download the GitHub online repo on VS Code.

1.8.1 Goal 1: Upload current project files on the online repo

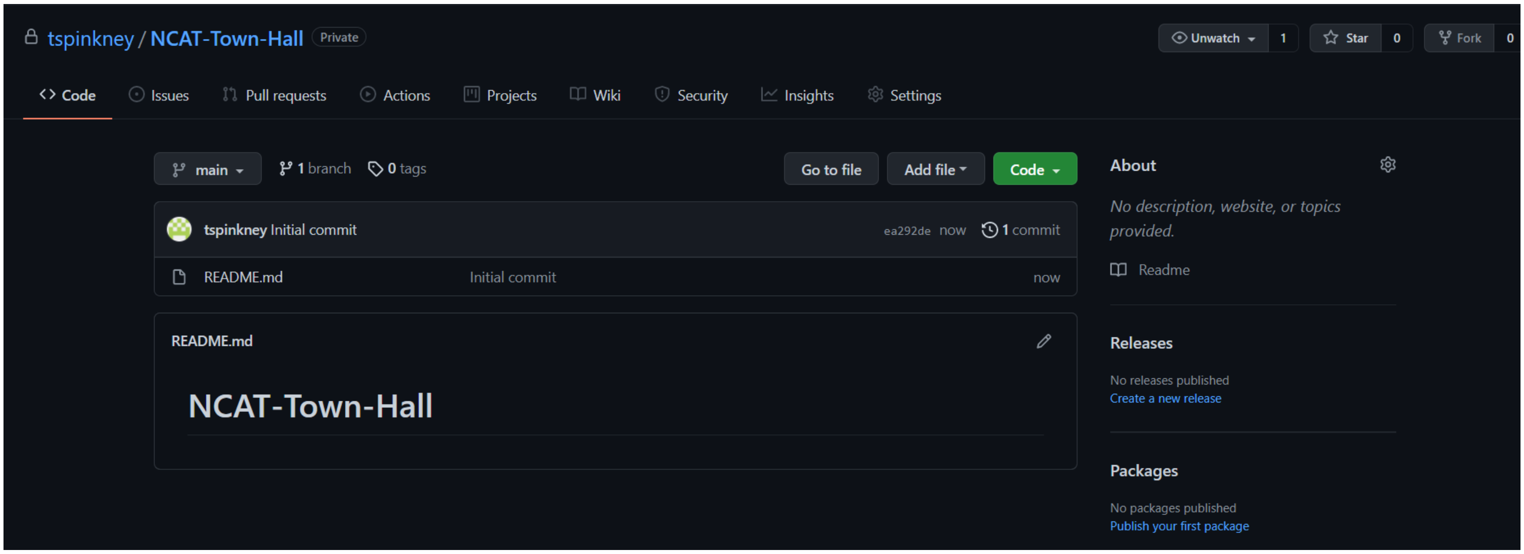
We plan to upload all the existing project files onto the GitHub repo so that everyone on the team has access to previous project files from last semester.

1.8.2 Goal 2: Learn about Git Bash Commands

Each team member must familiarize themselves with using the Git Bash terminal to download the online repo on VS Code.

Attach weekly work here

A screenshot of the initial landing page for the online Git Repo:



tspinkney Initial commit ea292de now 1 commit

README.md Initial commit now

README.md

NCAT-Town-Hall

No description, website, or topics provided.

No releases published

No packages published

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: SEPTEMBER 10, 2021 4
- 1.1 ACCOMPLISHMENT 1: ESTABLISH GITHUB REPO ACCESS 4
- 1.2 ACCOMPLISHMENT 2: ESTABLISH SHAREPOINT ACCESS 4
- 1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 4
- 1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 4
- 1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 4
- 1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 4
- 1.7 ISSUES AND SOLUTIONS 4
- 1.8 GOALS FOR NEXT WEEK 4
  - *1.8.1 Goal 1: Install VS Code 4*
  - *1.8.2 Goal 2: Pull first working branch with GitBash 4*

2 Week Ending: September 10, 2021

2.1 Accomplishment 1: Establish GitHub Repo Access

Each team member was tasked to gain access to the online repo by accepting an invitation through their student email.

2.2 Accomplishment 2: Establish SharePoint Access

Each team member was given access to a Microsoft SharePoint drive that has all the project files from the previous Spring Semester.

2.3 Individual Group Member Contribution: Hau Le

Gained access to the online GitHub Repository and SharePoint

2.4 Individual Group Member Contribution: Ryan Blount

Gained access to the online GitHub Repository and SharePoint. Uploaded python coding script for the Aggie Bot onto the GitHub Repo.

2.5 Individual Group Member Contribution: Sage-Michael Brown

Gained access to the online GitHub Repository and SharePoint

2.6 Individual Group Member Contribution: Terrell Pinkney

Gained access to the online GitHub Repository and SharePoint. Uploaded previous project files from last semester for reference onto the GitHub Repo.

2.7 Issues and Solutions

- Issues
    - N/A
- Solutions
    - N/A

2.8 Goals for Next Week

Our goals for next week are to make sure that each team member has access to the PostgreSQL database, install Microsoft Visual Studio Code, finalize the updated initial project report.

2.8.1 Goal 1: Install VS Code

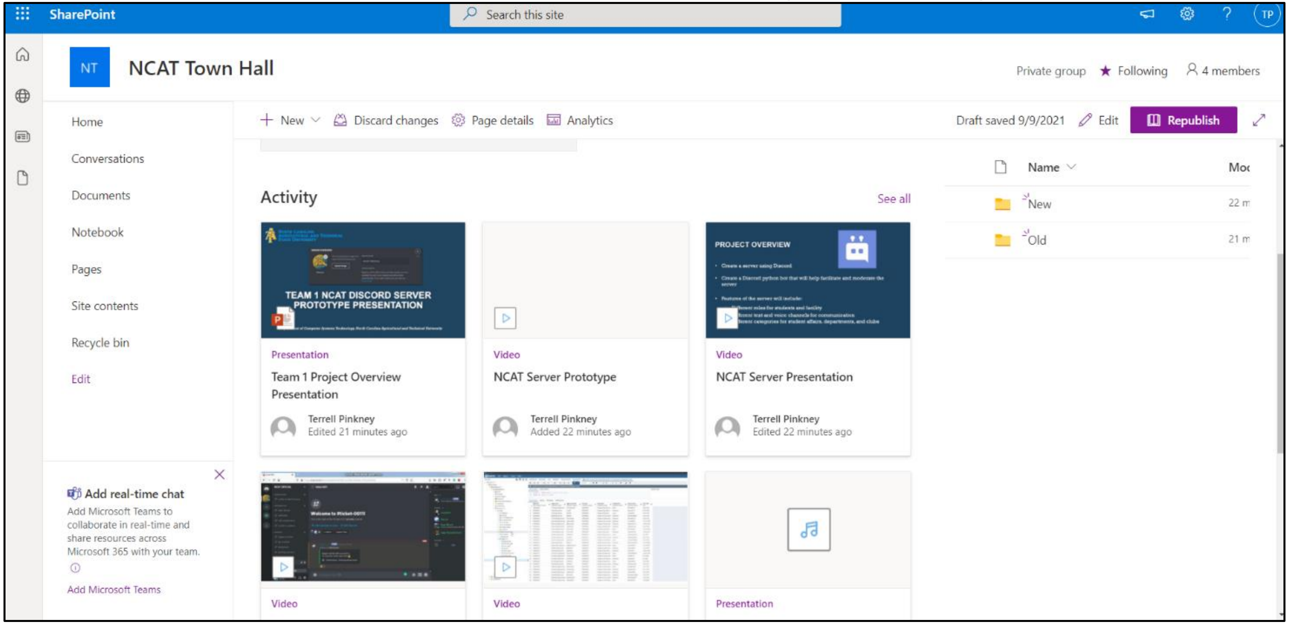
Each team member is to install Microsoft Visual Studio code to start working on code for the Aggie Bot and to link the current PostgreSQL database to the bot through VS Code.

2.8.2 Goal 2: Pull first working branch with GitBash

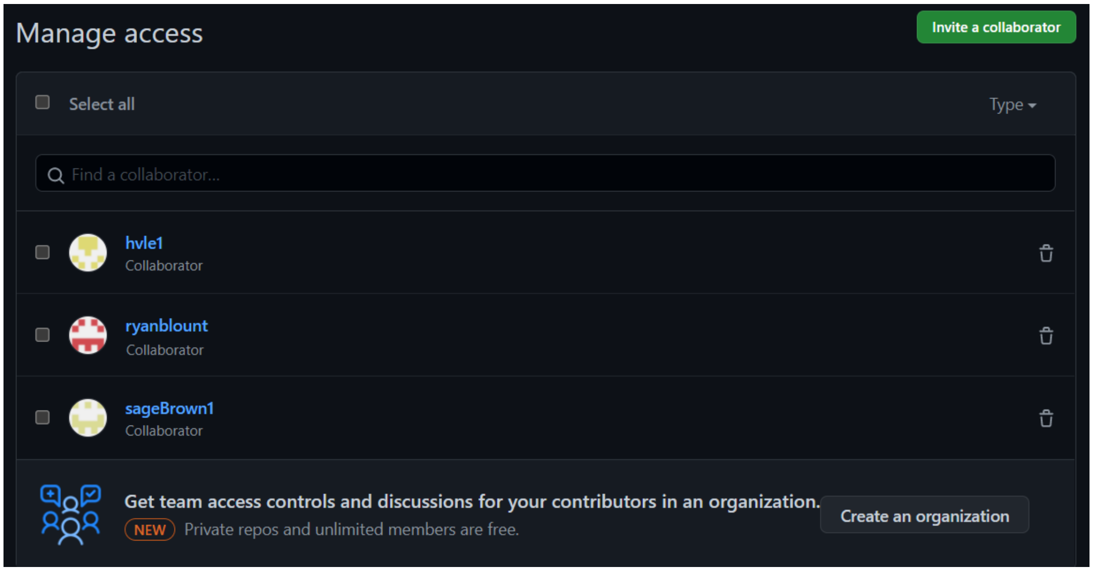
Each team member must successfully pull a working version of the python code so that they can contribute to adding more features to the Aggie Bot.

Attach weekly work here

A screenshot of the team SharePoint for uploading and viewing new project files:



A screenshot of each team member joining the online repository for the project on GitHub:



☐ hvle1
Collaborator

☐ ryanblount
Collaborator

☐ sageBrown1
Collaborator

Get team access controls and discussions for your contributors in an organization.
NEW Private repos and unlimited members are free.

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: SEPTEMBER 17, 2021 7

1.1 ACCOMPLISHMENT 1: INSTALLED MICROSOFT VISUAL STUDIO CODE 7
1.2 ACCOMPLISHMENT 2: INSTALLED GIT FOR BASH SHELL IN VISUAL STUDIO CODE 7
1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 7
1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 7
1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 7
1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 7
1.7 ISSUES AND SOLUTIONS 7
1.8 GOALS FOR NEXT WEEK 7
1.8.1 Goal 1: Finish creating the local branch 7
1.8.2 Goal 2: Pull first working branch with GitBash 7

3 Week Ending: September 17, 2021

3.1 Accomplishment 1: Installed Microsoft Visual Studio Code

All team members were able to successfully install Microsoft Visual Studio Code onto their devices.

3.2 Accomplishment 2: Installed Git for bash shell in Visual Studio Code

All team members were able to successfully install Git and load the software into Visual Studio Code.

3.3 Individual Group Member Contribution: Hau Le

Successfully installed Microsoft Visual Studio Code. Installed Git software for Visual Studio Code shell command.

3.4 Individual Group Member Contribution: Ryan Blount

Successfully installed Microsoft Visual Studio Code. Installed Git software for Visual Studio Code shell command.

3.5 Individual Group Member Contribution: Sage-Michael Brown

Successfully installed Microsoft Visual Studio Code. Installed Git software for Visual Studio Code shell command.

3.6 Individual Group Member Contribution: Terrell Pinkney

Successfully installed Microsoft Visual Studio Code. Installed Git software for Visual Studio Code shell command. Started the cloning process for the GitHub repository by creating the first local branch.

3.7 Issues and Solutions

- Issues
    - N/A
- Solutions
    - N/A

3.8 Goals for Next Week

Our goals for next week are to finish cloning the GitHub Repo onto Visual Studio Code and to push the local branch online so that each team member can pull from the master branch.

3.8.1 Goal 1: Finish creating the local branch

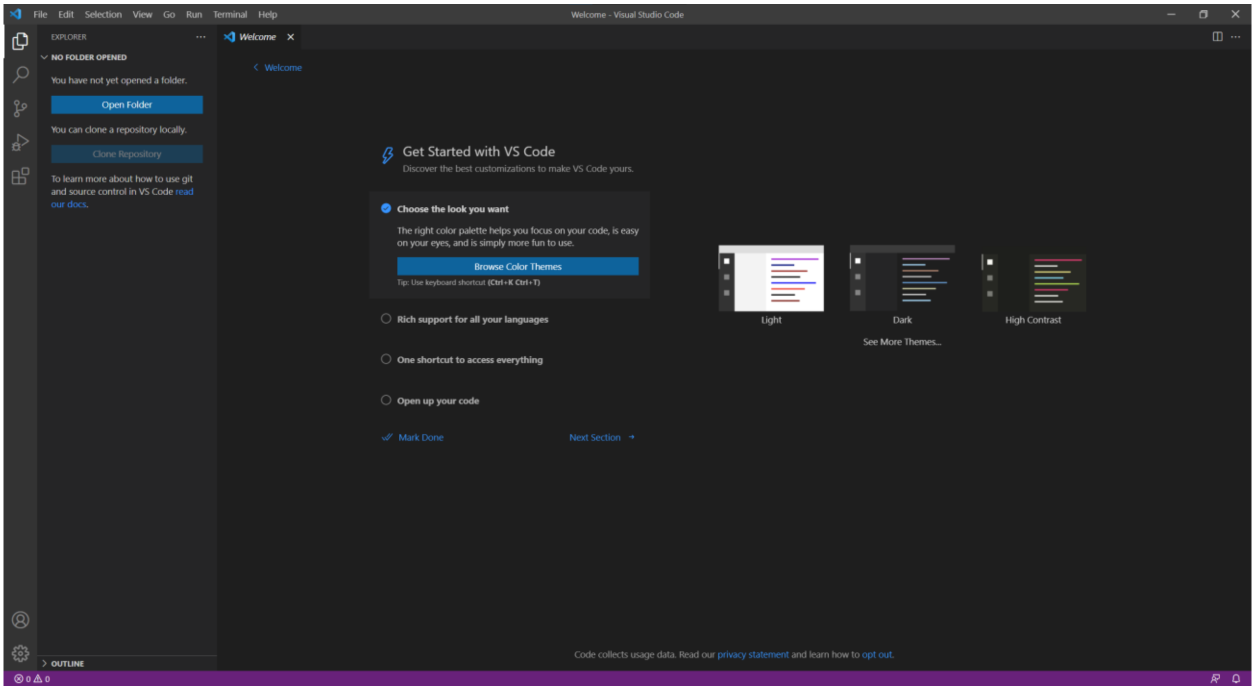
One team member must clone the online repo and create the master branch so that other team members can use the Git pull command to clone their branches.

3.8.2 Goal 2: Pull first working branch with GitBash

Each team member must successfully pull a working version of the python code so that they can contribute to adding more features to the Aggie Bot.

Attach weekly work here

A screenshot of Microsoft Visual Studio Code installed:



You have not yet opened a folder.

You can clone a repository locally.

To learn more about how to use git and source control in VS Code read our docs.

Get Started with VS Code

Discover the best customizations to make VS Code yours.

Choose the look you want

The right color palette helps you focus on your code, is easy on your eyes, and is simply more fun to use.

Tip: Use keyboard shortcut (Ctrl+K Ctrl+T)

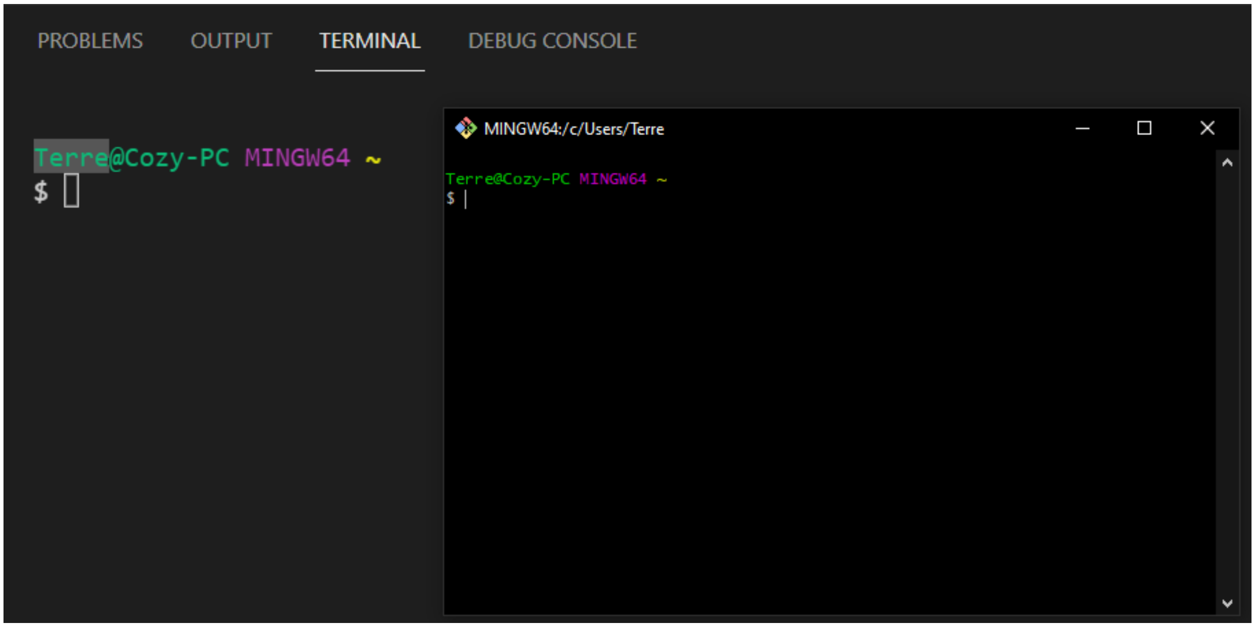
○ Rich support for all your languages

○ One shortcut to access everything

○ Open up your code

Code collects usage data. Read our privacy statement and learn how to opt out.

A screenshot of Git Bash installed:



```
Terre@Cozy-PC MINGW64 ~
$ 
```

```
Terre@Cozy-PC MINGW64 ~
$ |
```

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: SEPTEMBER 24, 2021 10

1.1 ACCOMPLISHMENT 1: INSTALL PYTHON 10
1.2 ACCOMPLISHMENT 2: N/A 10
1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 10
1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 10
1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 10
1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 10
1.7 ISSUES AND SOLUTIONS 10
1.8 GOALS FOR NEXT WEEK 10
1.8.1 Goal 1: Continue adding feature to the Discord bot 10
1.8.2 Goal 2: Continue working on updating the Midterm Report 10

4 Week Ending: September 24, 2021

4.1 Accomplishment 1: Install Python

- Each team member was able to successfully install python to work on the aggie bot .py file in Visual Studio Code.

4.2 Accomplishment 2: N/A

4.3 Individual Group Member Contribution: Hau Le

- Started researching for new ideas that can be added to the code for the Aggie Discord Bot.

4.4 Individual Group Member Contribution: Ryan Blount

- Started researching for new ideas that can be added to the code for the Aggie Discord Bot.

4.5 Individual Group Member Contribution: Sage-Michael Brown

- Started researching for new ideas that can be added to the code for the Aggie Discord Bot.
- Added a feature that allows students to send and review code for different classes

4.6 Individual Group Member Contribution: Terrell Pinkney

- Started researching for new ideas that can be added to the code for the Aggie Discord Bot.

4.7 Issues and Solutions

- Issues
    - N/A
- Solutions
    - N/A

4.8 Goals for Next Week

4.8.1 Goal 1: Continue adding feature to the Discord bot

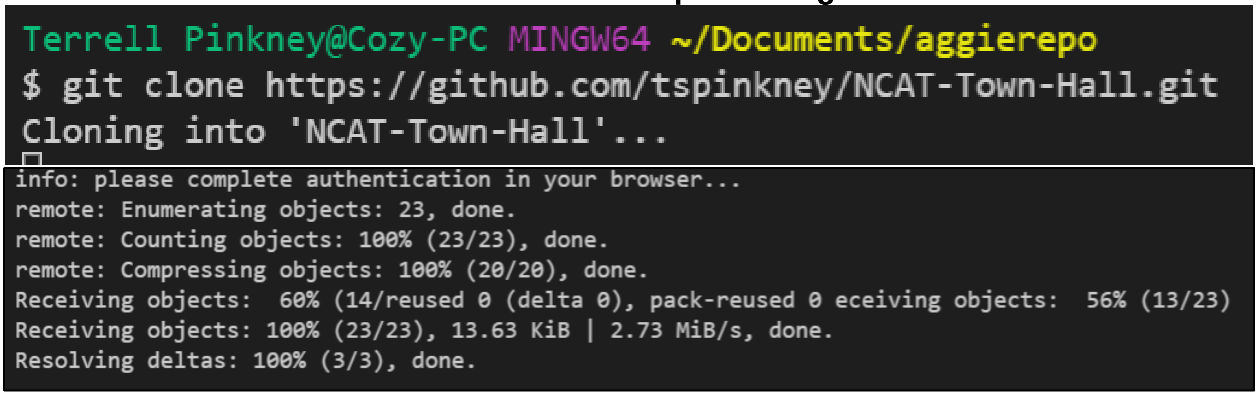
- We plan to add more useful features to the bot which include but are not limited too:
    - Student Trivia for practicing for exams
    - Student feedback for professor lectures
    - Student listening parties and music features

4.8.2 Goal 2: Continue working on updating the Midterm Report

- We will also continue updating the Midterm Report and preparing for our Midterm Presentation.

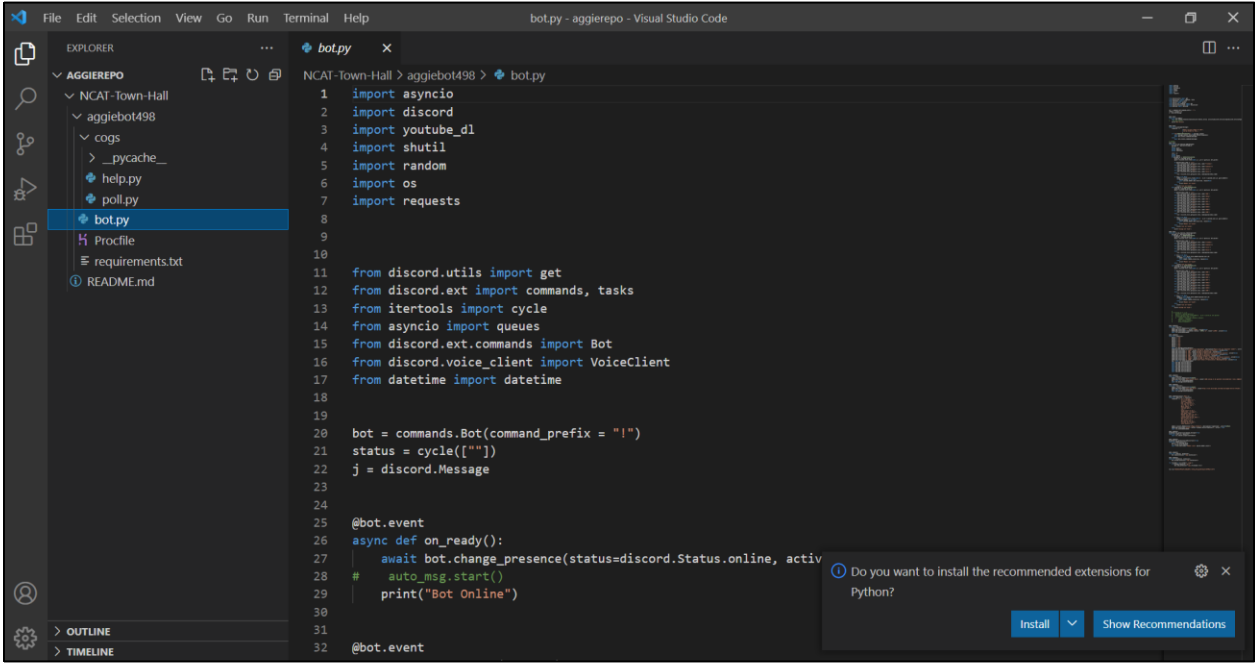
Attach weekly work here

A screenshot of the GitHub repo cloning into VS code



```
Terrell Pinkney@Cozy-PC MINGW64 ~/Documents/aggierepo
$ git clone https://github.com/tspinkney/NCAT-Town-Hall.git
Cloning into 'NCAT-Town-Hall'...
info: please complete authentication in your browser...
remote: Enumerating objects: 23, done.
remote: Counting objects: 100% (23/23), done.
remote: Compressing objects: 100% (20/20), done.
Receiving objects:  60% (14/reused 0 (delta 0), pack-reused 0 eceiving objects:  56% (13/23)
Receiving objects: 100% (23/23), 13.63 KiB | 2.73 MiB/s, done.
Resolving deltas: 100% (3/3), done.
```

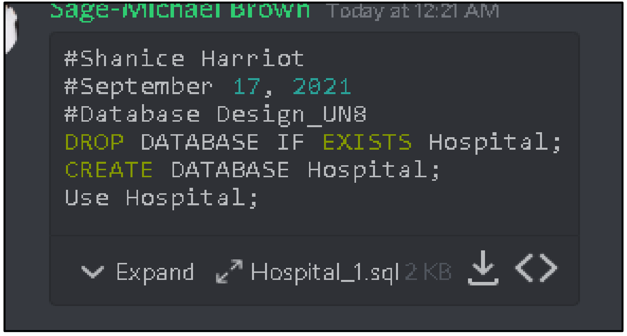
A screenshot of repo cloned into VS code:



```python
import asyncio
import discord
import youtube_dl
import shutil
import random
import os
import requests



from discord.utils import get
from discord.ext import commands, tasks
from itertools import cycle
from asyncio import queues
from discord.ext.commands import Bot
from discord.voice_client import VoiceClient
from datetime import datetime


bot = commands.Bot(command_prefix = "!")
status = cycle([""])
j = discord.Message


@bot.event
async def on_ready():
    await bot.change_presence(status=discord.Status.online, activ
#    auto_msg.start()
    print("Bot Online")


@bot.event
```

Do you want to install the recommended extensions for Python?

A screenshot of a student sending code for review in the Discord server:



Sage-Michael Brown Today at 12:21 AM

```sql
#Shanice Harriot
#September 17, 2021
#Database Design_UN8
DROP DATABASE IF EXISTS Hospital;
CREATE DATABASE Hospital;
Use Hospital;
```

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: OCTOBER 8, 2021 13

1.1 ACCOMPLISHMENT 1: CREATE QUESTIONS FOR QUIZ FEATURE FOR BOT 13
1.2 ACCOMPLISHMENT 2: ADD NEW IMAGES AND DETAILS TO MIDTERM REPORT 13
1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 13
1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 13
1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 13
1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 13
1.7 ISSUES AND SOLUTIONS 13
1.8 GOALS FOR NEXT WEEK 13
1.8.1 Goal 1: Continue adding feature to the Discord bot 13
1.8.2 Goal 2: Continue working on updating the Midterm Report 13

5 Week Ending: October 8, 2021

5.1 Accomplishment 1: Create questions for quiz feature for bot

- We were able to create at least 20 questions that our bot can ask students in the server.

5.2 Accomplishment 2: Add new images and details to Midterm Report

- We were able to add updated content and images to the Midterm Report.

5.3 Individual Group Member Contribution: Hau Le

- Researched and created NC A&T related questions for the Discord Bot to generate in the server.

5.4 Individual Group Member Contribution: Ryan Blount

- Implemented each question into the python code on VS code.

5.5 Individual Group Member Contribution: Sage-Michael Brown

- Researched and created NC A&T related questions for the Discord Bot to generate in the server.

5.6 Individual Group Member Contribution: Terrell Pinkney

- Implemented each mutiple choice answer into the python code on VS code. Also workedo n Midterm Report and updated key details and images from work in progress of project.

5.7 Issues and Solutions

- Issues
  - N/A
- Solutions
  - N/A

5.8 Goals for Next Week

5.8.1 Goal 1: Continue adding features to the Discord bot

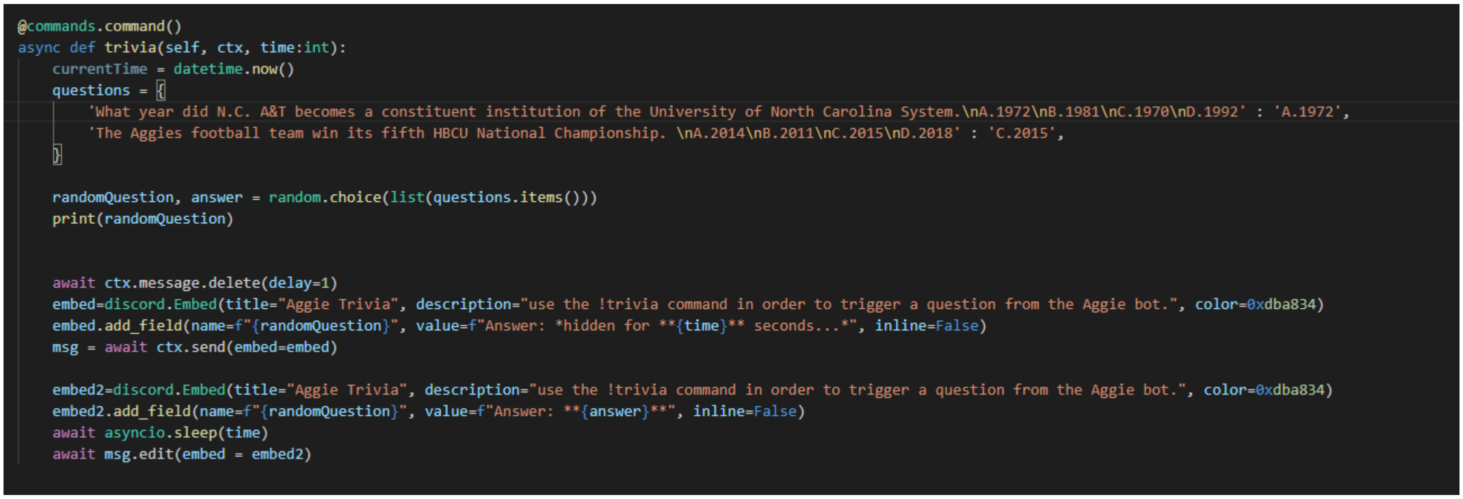
- We plan to continue adding useful features to the bot which include:
  - Student feedback for professor lectures
  - Student listening parties and music features
  - More tutoring resource features

5.8.2 Goal 2: Continue working on updating the Midterm Report

- We will also continue updating the Midterm Report and preparing for our Midterm Presentation Video.

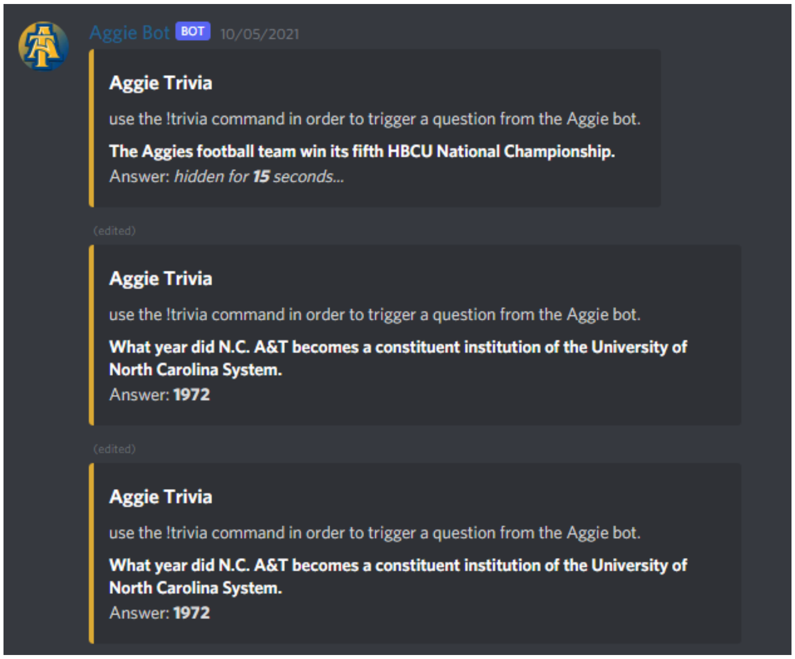
Attach weekly work here

A screenshot of the code for the first trivia bot question:



```
@commands.command()
async def trivia(self, ctx, time:int):
    currentTime = datetime.now()
    questions = {
        'What year did N.C. A&T becomes a constituent institution of the University of North Carolina System.\nA.1972\nB.1981\nC.1970\nD.1992' : 'A.1972',
        'The Aggies football team win its fifth HBCU National Championship. \nA.2014\nB.2011\nC.2015\nD.2018' : 'C.2015',
    }

    randomQuestion, answer = random.choice(list(questions.items()))
    print(randomQuestion)


    await ctx.message.delete(delay=1)
    embed=discord.Embed(title="Aggie Trivia", description="use the !trivia command in order to trigger a question from the Aggie bot.", color=0xdba834)
    embed.add_field(name=f"{randomQuestion}", value=f"Answer: *hidden for **{time}** seconds...*", inline=False)
    msg = await ctx.send(embed=embed)

    embed2=discord.Embed(title="Aggie Trivia", description="use the !trivia command in order to trigger a question from the Aggie bot.", color=0xdba834)
    embed2.add_field(name=f"{randomQuestion}", value=f"Answer: **{answer}**", inline=False)
    await asyncio.sleep(time)
    await msg.edit(embed = embed2)
```

A screenshot of the bot asking different multiple choice questions in the Discord server:



Aggie Bot BOT 10/05/2021

Aggie Trivia

use the !trivia command in order to trigger a question from the Aggie bot.

The Aggies football team win its fifth HBCU National Championship.

Answer: *hidden for **15** seconds...*

(edited)

Aggie Trivia

use the !trivia command in order to trigger a question from the Aggie bot.

What year did N.C. A&T becomes a constituent institution of the University of North Carolina System.

Answer: **1972**

(edited)

Aggie Trivia

use the !trivia command in order to trigger a question from the Aggie bot.

What year did N.C. A&T becomes a constituent institution of the University of North Carolina System.

Answer: **1972**

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: OCTOBER 15, 2021 16

1.1 ACCOMPLISHMENT 1: FINISH TEST PHASE 1 FOR THE TRIVIA BOT FUNCTION 16
1.2 ACCOMPLISHMENT 2: TESTED THE BOT FUNCTIONALITY IN REGARDS TO TRIVIA QUESTIONS 16
1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 16
1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 16
1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 16
1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 16
1.7 ISSUES AND SOLUTIONS 16
1.8 GOALS FOR NEXT WEEK 16
1.8.1 Goal 1: Continue adding features to the Discord bot 16
1.8.2 Goal 2: Code the bot so that it can collect student question/answer ratio 16

6 Week Ending: October 15, 2021

6.1 Accomplishment 1: Finish test phase 1 for the trivia bot function

- We were able to finish the initial code for testing the bot when asking questions on the server.

6.2 Accomplishment 2: Tested the bot functionality in regards to trivia questions

- We were able to find specific errors and bugs when testing out a student answering a question.

6.3 Individual Group Member Contribution: Hau Le

- Researched and created NC A&T-related questions for the Discord Bot to generate in the server.

6.4 Individual Group Member Contribution: Ryan Blount

- Implemented each question into the python code via VS code.

6.5 Individual Group Member Contribution: Sage-Michael Brown

- Researched and created NC A&T-related questions for the Discord Bot to generate in the server. Provided useful and updated database information for each student.

6.6 Individual Group Member Contribution: Terrell Pinkney

- Implemented each multiple-choice answer into the python code on VS code. Finalized the project midterm and helped test the trivia feature of the bot.

6.7 Issues and Solutions

- Issues
  - The bot would prompt a student with the same question twice
  - The bot would send a response showing two correct answers instead of one.
- Solutions
  - Since the questions are randomized, we had to create and code more questions so that a user would not see the same question twice.

6.8 Goals for Next Week

6.8.1 Goal 1: Continue adding features to the Discord bot

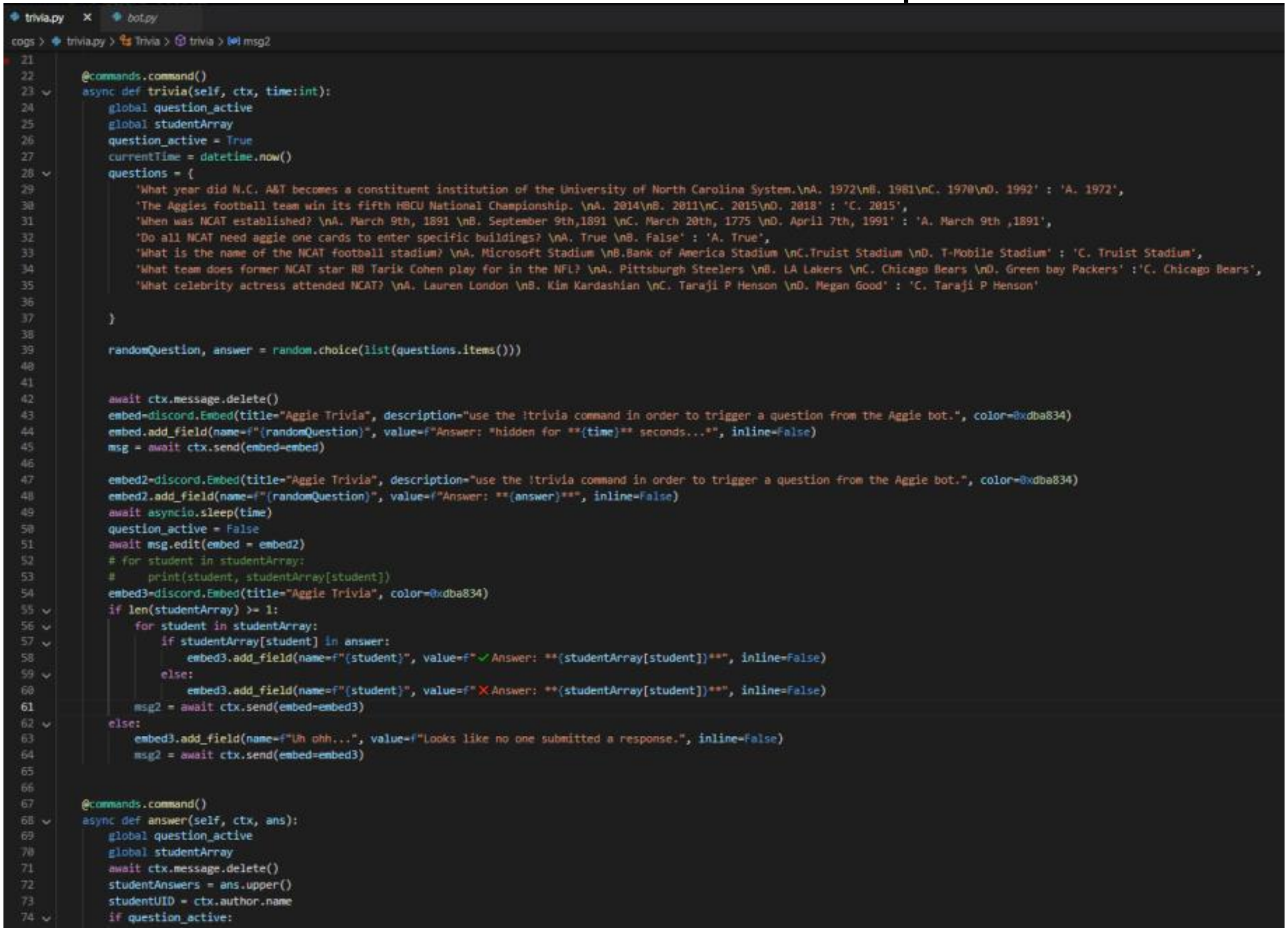
- We plan to continue adding useful features to the bot which include:
  - Student feedback for professor lectures
  - Student listening parties and music features
  - More tutoring resource features
  - Student game nights and online game sessions.

6.8.2 Goal 2: Code the bot so that it can collect student question/answer statistics

- We plan to update the code on the bot so that it can give students results showcasing their question/answer statistics from the beginning to end of each month,

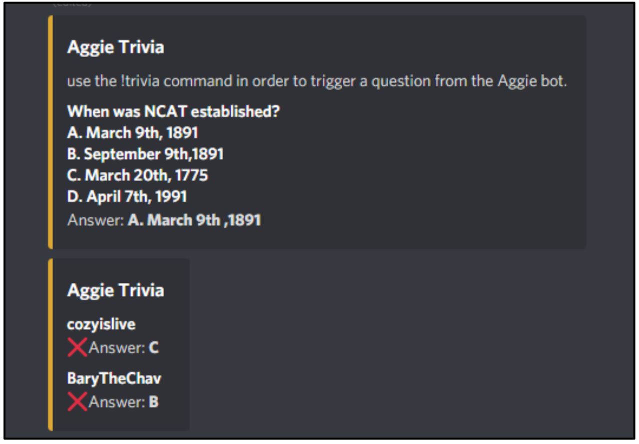
Attach weekly work here

A screenshot of the code for the trivia bot question:



```python
    @commands.command()
    async def trivia(self, ctx, time:int):
        global question_active
        global studentArray
        question_active = True
        currentTime = datetime.now()
        questions = {
            'What year did N.C. A&T becomes a constituent institution of the University of North Carolina System.\nA. 1972\nB. 1981\nC. 1970\nD. 1992' : 'A. 1972',
            'The Aggies football team win its fifth HBCU National Championship. \nA. 2014\nB. 2011\nC. 2015\nD. 2018' : 'C. 2015',
            'When was NCAT established? \nA. March 9th, 1891 \nB. September 9th,1891 \nC. March 20th, 1775 \nD. April 7th, 1991' : 'A. March 9th ,1891',
            'Do all NCAT need aggie one cards to enter specific buildings? \nA. True \nB. False' : 'A. True',
            'What is the name of the NCAT football stadium? \nA. Microsoft Stadium \nB.Bank of America Stadium \nC.Truist Stadium \nD. T-Mobile Stadium' : 'C. Truist Stadium',
            'What team does former NCAT star RB Tarik Cohen play for in the NFL? \nA. Pittsburgh Steelers \nB. LA Lakers \nC. Chicago Bears \nD. Green bay Packers' :'C. Chicago Bears',
            'What celebrity actress attended NCAT? \nA. Lauren London \nB. Kim Kardashian \nC. Taraji P Henson \nD. Megan Good' : 'C. Taraji P Henson'

        }

        randomQuestion, answer = random.choice(list(questions.items()))


        await ctx.message.delete()
        embed=discord.Embed(title="Aggie Trivia", description="use the !trivia command in order to trigger a question from the Aggie bot.", color=0xdba834)
        embed.add_field(name=f"{randomQuestion}", value=f"Answer: *hidden for **{time}** seconds...*", inline=False)
        msg = await ctx.send(embed=embed)

        embed2=discord.Embed(title="Aggie Trivia", description="use the !trivia command in order to trigger a question from the Aggie bot.", color=0xdba834)
        embed2.add_field(name=f"{randomQuestion}", value=f"Answer: **{answer}**", inline=False)
        await asyncio.sleep(time)
        question_active = False
        await msg.edit(embed = embed2)
        # for student in studentArray:
        #     print(student, studentArray[student])
        embed3=discord.Embed(title="Aggie Trivia", color=0xdba834)
        if len(studentArray) >= 1:
            for student in studentArray:
                if studentArray[student] in answer:
                    embed3.add_field(name=f"{student}", value=f"✓ Answer: **{studentArray[student]}**", inline=False)
                else:
                    embed3.add_field(name=f"{student}", value=f"✗ Answer: **{studentArray[student]}**", inline=False)
            msg2 = await ctx.send(embed=embed3)
        else:
            embed3.add_field(name=f"Uh ohh...", value=f"Looks like no one submitted a response.", inline=False)
            msg2 = await ctx.send(embed=embed3)


    @commands.command()
    async def answer(self, ctx, ans):
        global question_active
        global studentArray
        await ctx.message.delete()
        studentAnswers = ans.upper()
        studentUID = ctx.author.name
        if question_active:
```

A screenshot of the bot asking a sample question to a student in the server:



Aggie Trivia

use the !trivia command in order to trigger a question from the Aggie bot.

When was NCAT established?
A. March 9th, 1891
B. September 9th,1891
C. March 20th, 1775
D. April 7th, 1991
Answer: **A. March 9th ,1891**

Aggie Trivia

cozyislive
✗Answer: **C**

BaryTheChav
✗Answer: **B**

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: OCTOBER 22, 2021 19

1.1 ACCOMPLISHMENT 1: TESTED FOR ANY ERRORS AND BUGS WITH THE TRIVIA QUESTION FEATURE FOR THE BOT 19
1.2 ACCOMPLISHMENT 2: ADDED MORE RESPONSES TO THE BOT 19
1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 19
1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 19
1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 19
1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 19
1.7 ISSUES AND SOLUTIONS 19
1.8 GOALS FOR NEXT WEEK 19
1.8.1 Goal 1: Continue adding features to the Discord bot 19

7 Week Ending: October 22, 2021

7.1 Accomplishment 1: Tested for any errors and bugs with the trivia question feature for the bot

- We were able to test the trivia question feature for the bot and found different errors and bugs.

7.2 Accomplishment 2: Added more responses to the bot

- We were able to add more responses and feedback to the bot after a student answers a question. The bot now displays the percentages of each answer a student chooses after answering a question and the correct/incorrect answer from each question. Additionally, the bot will provide a response when the channel is idle and a student does not answer the question after a specific amount of time.

7.3 Individual Group Member Contribution: Hau Le

- Created the response for when a student does not answer questions after an allotted time from when the server is idle.

7.4 Individual Group Member Contribution: Ryan Blount

- Formatted each response written by other team members into the bot so that the bot could provide more feedback to students when answering questions.

7.5 Individual Group Member Contribution: Sage-Michael Brown

- Created the response that shows the percentages of how many students answered each question.

7.6 Individual Group Member Contribution: Terrell Pinkney

- Helped test each trivia question and response from the bot to find bugs and errors with team member Ryan.

7.7 Issues and Solutions

- Issues
    - Errors when showing the percentages of students answers
    - Errors when showing the correct and incorrect answers back to the students

- Solutions
    - We had to fix a couple of logical errors in the code to fix the responses in the bot.

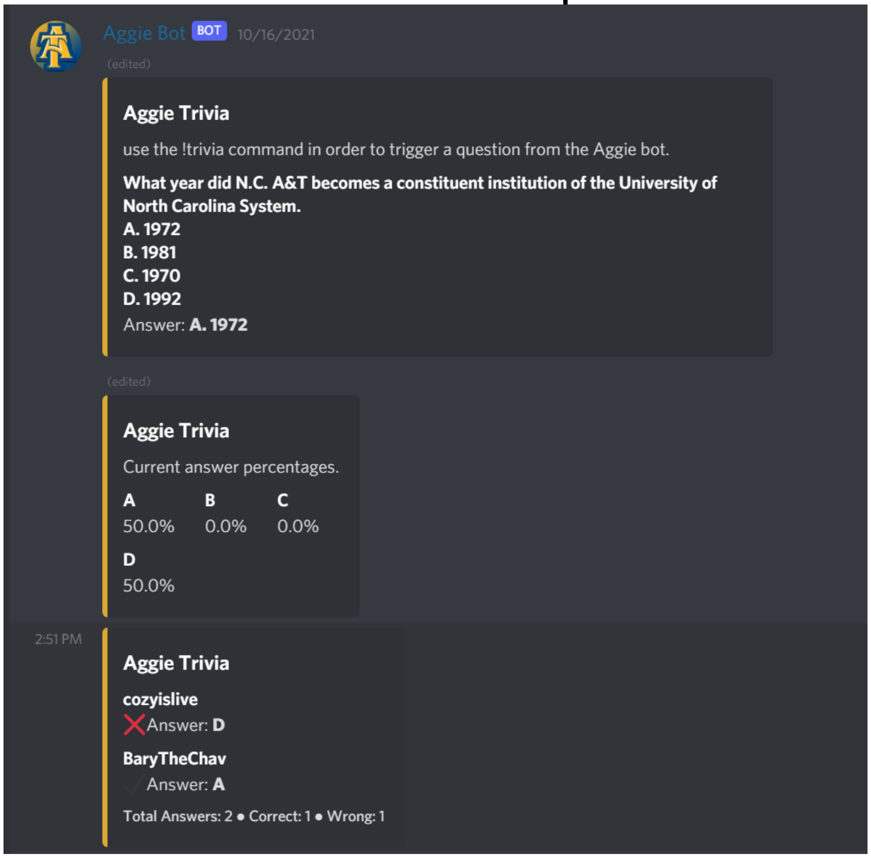
7.8 Goals for Next Week

7.8.1 Goal 1: Continue adding features to the Discord bot

- We plan to continue adding useful features to the bot which include:
    - Student feedback for professor lectures
    - Student listening parties and music features
    - More tutoring resource features
    - Student game nights and online game sessions.

Attach weekly work here

A screenshot of the bot asking a student a question and providing the correct and incorrect answer for the question:



Aggie Trivia

use the !trivia command in order to trigger a question from the Aggie bot.

What year did N.C. A&T becomes a constituent institution of the University of North Carolina System.
A. 1972
B. 1981
C. 1970
D. 1992
Answer: **A. 1972**

Aggie Trivia

Current answer percentages.

A	B	C
50.0%	0.0%	0.0%

D
50.0%

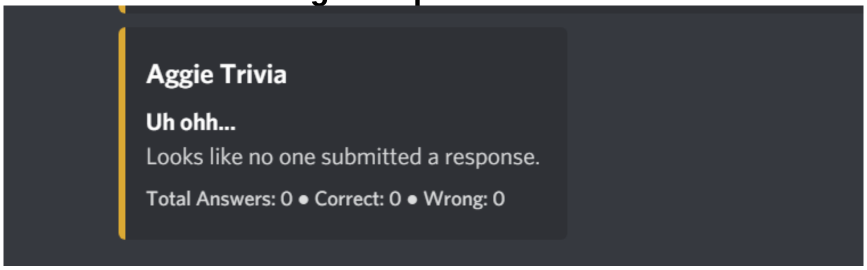
Aggie Trivia

cozyislive
❌Answer: **D**

BaryTheChav
✓Answer: **A**

Total Answers: 2 • Correct: 1 • Wrong: 1

A screenshot of the bot sending a response when no one answers a question.



Aggie Trivia

Uh ohh...
Looks like no one submitted a response.

Total Answers: 0 • Correct: 0 • Wrong: 0

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: OCTOBER 29, 2021 22

1.1 ACCOMPLISHMENT 1: EDITED THE INTIAL ROLE ASSIGNING CHANNEL 22
1.2 ACCOMPLISHMENT 2: N/A 22
1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 22
1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 22
1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 22
1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 22
1.7 ISSUES AND SOLUTIONS 22
1.8 GOALS FOR NEXT WEEK 22
1.8.1 Goal 1: Continue adding features to the Discord bot 22

8 Week Ending: October 29, 2021

8.1 Accomplishment 1: Edited the intial role assigning channel

- We made a more intuitive role greeting page that will allow students from different departments to select a specifc emoji to be sorted into their respective roles.

8.2 Accomplishment 2: N/A

8.3 Individual Group Member Contribution: Hau Le

- Helped add emojis for each department

8.4 Individual Group Member Contribution: Ryan Blount

- Programmed the bot to prompt students with the emoji poll to assign roles in the server

8.5 Individual Group Member Contribution: Sage-Michael Brown

- Created each department that will appear in the role channel

8.6 Individual Group Member Contribution: Terrell Pinkney

- Helped test the role assigning poll response from the bot

8.7 Issues and Solutions

- Issues
  - N/A
- Solutions
  - N/A

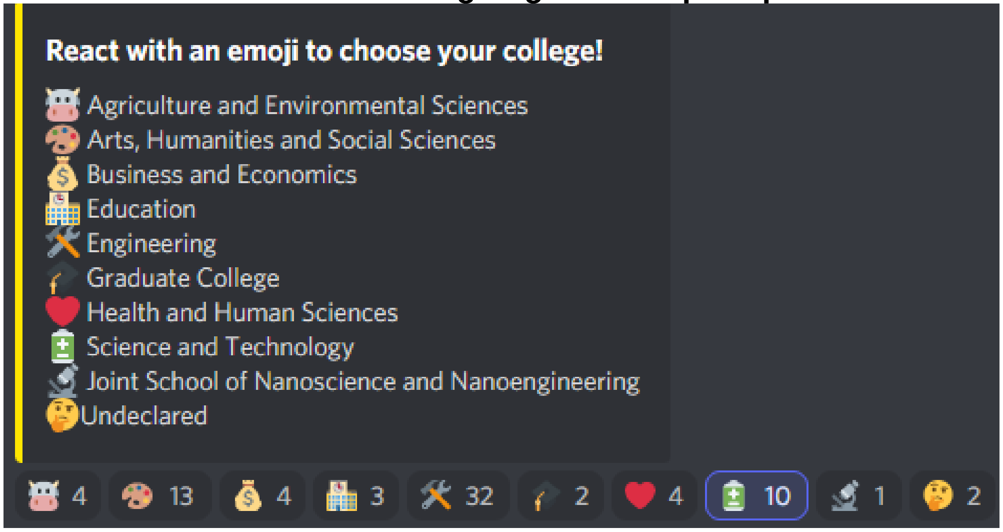
8.8 Goals for Next Week

8.8.1 Goal 1: Continue adding features to the Discord bot

- We plan to continue adding useful features to the bot which include:
  - Student feedback for professor lectures
  - Student listening parties and music features
  - More tutoring resource features
  - Student game nights and online game sessions.

Attach weekly work here

A screenshot of new role assigning channel prompt from the bot:



React with an emoji to choose your college!

🐮 Agriculture and Environmental Sciences
🎨 Arts, Humanities and Social Sciences
💰 Business and Economics
🏫 Education
🛠 Engineering
🎓 Graduate College
❤️ Health and Human Sciences
🔋 Science and Technology
🔬 Joint School of Nanoscience and Nanoengineering
🤔Undeclared

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: NOVEMBER 5, 2021 25

1.1 ACCOMPLISHMENT 1: RESEARCHED IDEAS FOR A STUDENT LEVELING SYSTEM 25
1.2 ACCOMPLISHMENT 2: RESEARCHED IDEAS FOR CREATING A FAQ CHANNEL IN THE SERVER 25
1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 25
1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 25
1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 25
1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 25
1.7 ISSUES AND SOLUTIONS 25
1.8 GOALS FOR NEXT WEEK 25
1.8.1 Goal 1: Continue adding features to the Discord bot 25
1.8.2 Goal 2: Create FAQ questions for the FAQ channel 25

9 Week Ending: November 5, 2021

9.1 Accomplishment 1: Researched ideas for a student leveling system

- Each team member researched online resources and examples for implementing a level system for when a student responds in the server.

9.2 Accomplishment 2: Researched ideas for creating a FAQ channel in the server

- Each team member researched python examples of implementing FAQ response features with the bot.

9.3 Individual Group Member Contribution: Hau Le

- Researched online examples and resources for new leveling system features.

9.4 Individual Group Member Contribution: Ryan Blount

- Researched online examples and resources for new leveling system features.

9.5 Individual Group Member Contribution: Sage-Michael Brown

- Researched online examples and resources for new leveling system features.

9.6 Individual Group Member Contribution: Terrell Pinkney

- Researched online examples and resources for new leveling system features.

9.7 Issues and Solutions

- Issues
  - N/A
- Solutions
  - N/A

9.8 Goals for Next Week

9.8.1 Goal 1: Continue adding features to the Discord bot

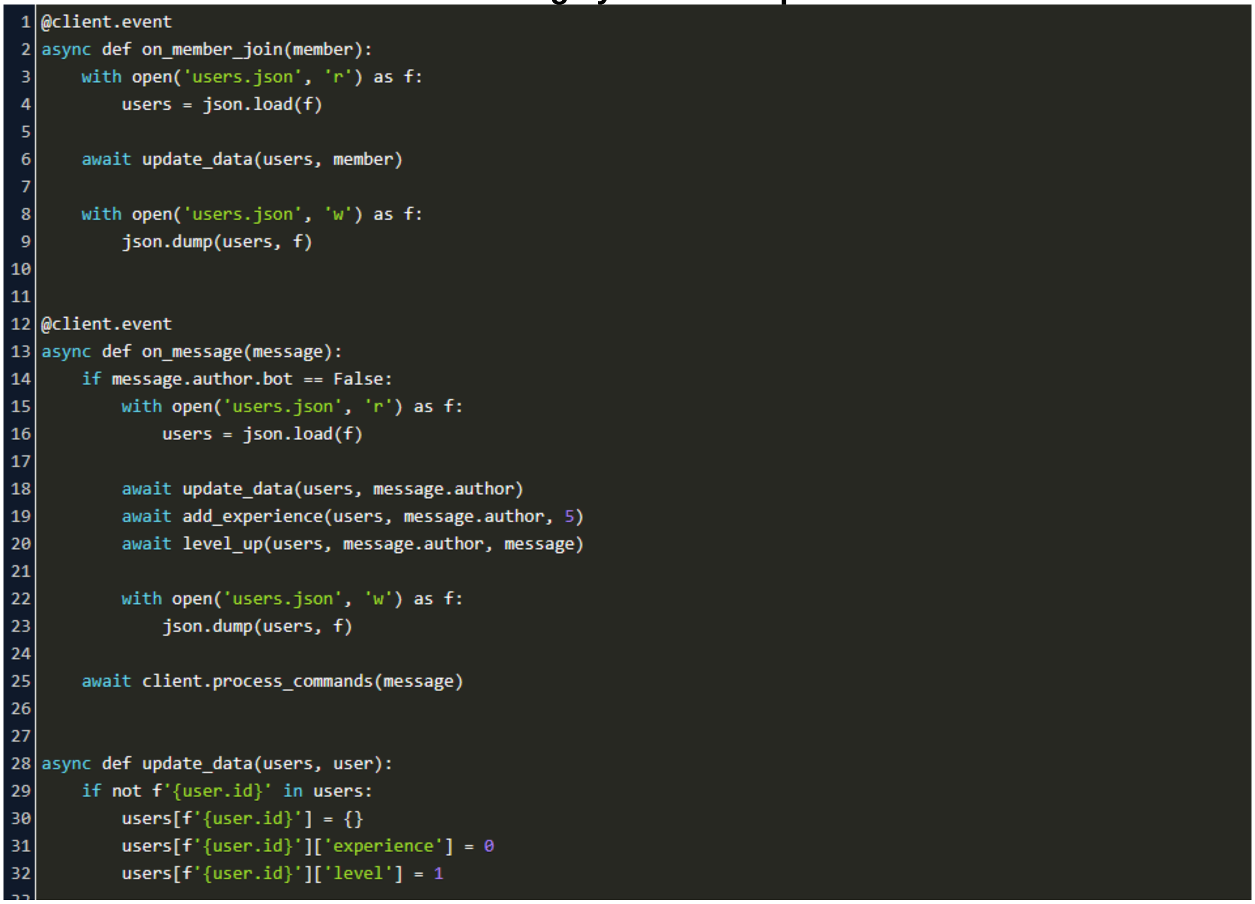
- We plan to continue adding useful features to the bot which include:
  - Student feedback for professor lectures
  - Student listening parties and music features
  - More tutoring resource features
  - Student game nights and online game sessions.

9.8.2 Goal 2: Create FAQ questions for the FAQ channel

- Each team member must research commonly asked A&T questions from different departments and services (Financial Aid, Registration, Housing, etc).

Attach weekly work here

A screenshot of leveling system examples found online:



```
@client.event
async def on_member_join(member):
    with open('users.json', 'r') as f:
        users = json.load(f)

    await update_data(users, member)

    with open('users.json', 'w') as f:
        json.dump(users, f)


@client.event
async def on_message(message):
    if message.author.bot == False:
        with open('users.json', 'r') as f:
            users = json.load(f)

        await update_data(users, message.author)
        await add_experience(users, message.author, 5)
        await level_up(users, message.author, message)

        with open('users.json', 'w') as f:
            json.dump(users, f)

    await client.process_commands(message)


async def update_data(users, user):
    if not f'{user.id}' in users:
        users[f'{user.id}'] = {}
        users[f'{user.id}']['experience'] = 0
        users[f'{user.id}']['level'] = 1
```

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: NOVEMBER 12, 2021 28

1.1 ACCOMPLISHMENT 1: SUCCESSFULLY CREATED A BOT RESPONSE FEATURE 28
1.2 ACCOMPLISHMENT 2: TEST THE BOT FOR BUGS WHEN RESPONDING TO QUESTIONS 28
1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 28
1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 28
1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 28
1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 28
1.7 ISSUES AND SOLUTIONS 28
1.8 GOALS FOR NEXT WEEK 28
1.8.1 Goal 1: Continue adding features to the Discord bot 28
1.8.2 Goal 2: Add more FAQ for the bot 28

10 Week Ending: November 12, 2021

10.1 Accomplishment 1: Successfully created a bot response feature

- We were able to create a new feature that would allow the discord bot to respond to questions that students may ask.

10.2 Accomplishment 2: Test the bot for bugs when responding to questions

- We tested the bot's functionality when responding to specific questions from a user on the server.

10.3 Individual Group Member Contribution: Hau Le

- Created questions and answers relating to Financial Aid for the bot when answering student questions.

10.4 Individual Group Member Contribution: Ryan Blount

- Took each question created from each team member and added those questions into the bot.

10.5 Individual Group Member Contribution: Sage-Michael Brown

- Created questions and answers relating to Graduation for the bot when answering student questions.

10.6 Individual Group Member Contribution: Terrell Pinkney

- Created questions and answers relating to Parking for the bot when answering student questions.

10.7 Issues and Solutions

- Issues
    - There was an issue with the bot answering questions for specific categories.
    - There was an issue with the bot when producing helpful links to navigate students to specific websites.
- Solutions
    - To fix the issue with answering different types of questions, we programmed the bot to read specific keywords in the questions to decide how to answer.
    - To fix the hyperlinks for bot responses we had to embed the links into the keywords using markup language.

10.8 Goals for Next Week

10.8.1 Goal 1: Continue adding features to the Discord bot

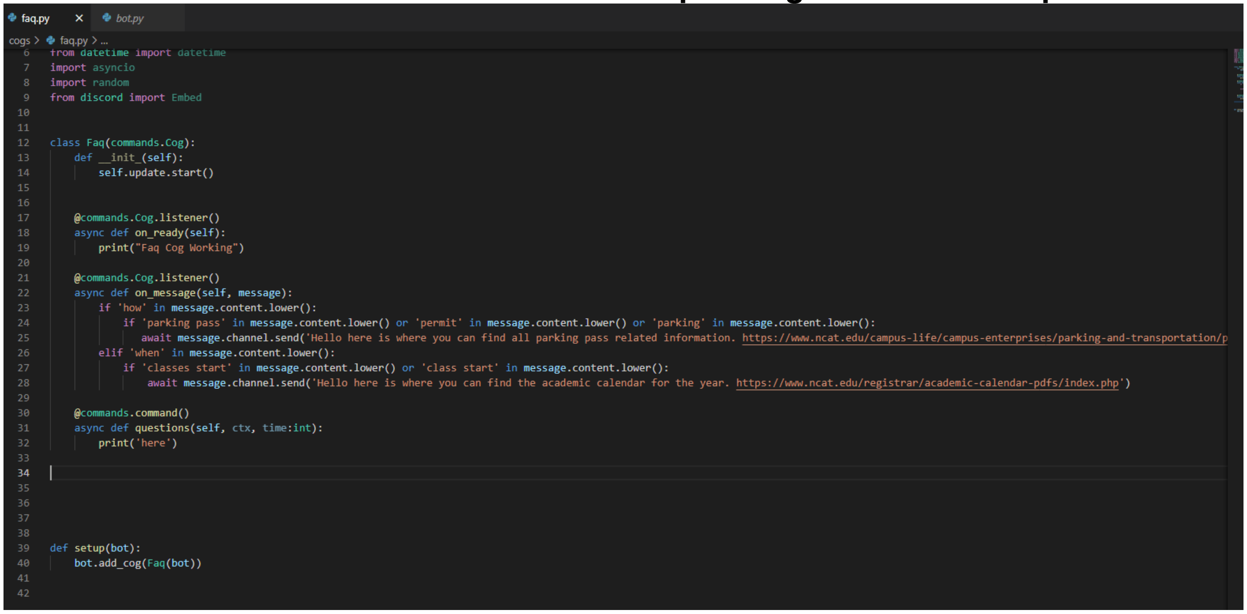
- We plan to continue adding useful features to the bot which include:
    - Student feedback for professor lectures
    - Student listening parties and music features
    - More tutoring resource features
    - Student game nights and online game sessions.

10.8.2 Goal 2: Add more FAQ for the bot

- We plan to add more answers to questions students may have about different categories. We also plan to test the bot for bugs when responding to these questions.

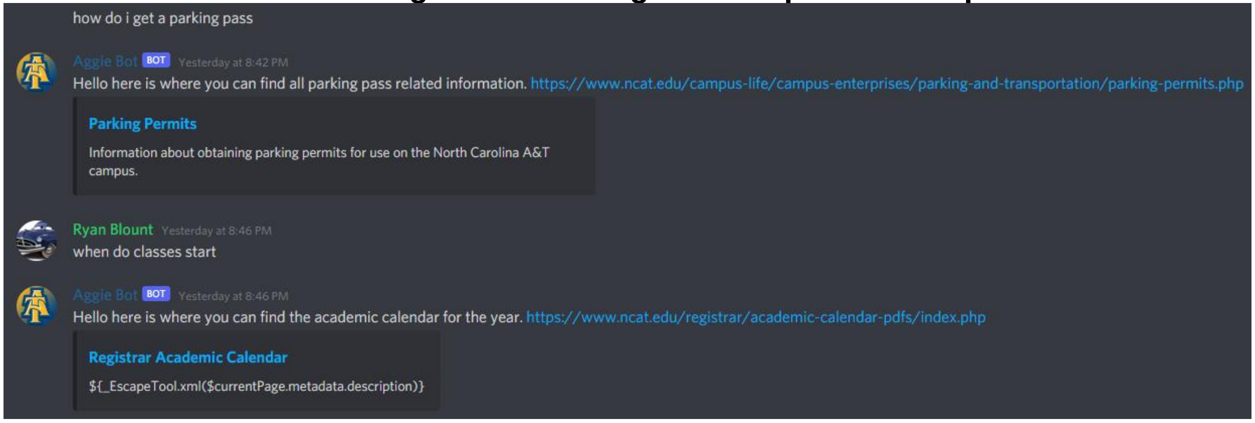
Attach weekly work here

A screenshot of the code for the bot responding to a student's question:



```python
from datetime import datetime
import asyncio
import random
from discord import Embed


class Faq(commands.Cog):
    def __init_(self):
        self.update.start()


    @commands.Cog.listener()
    async def on_ready(self):
        print("Faq Cog Working")

    @commands.Cog.listener()
    async def on_message(self, message):
        if 'how' in message.content.lower():
            if 'parking pass' in message.content.lower() or 'permit' in message.content.lower() or 'parking' in message.content.lower():
               await message.channel.send('Hello here is where you can find all parking pass related information. https://www.ncat.edu/campus-life/campus-enterprises/parking-and-transportation/p
        elif 'when' in message.content.lower():
            if 'classes start' in message.content.lower() or 'class start' in message.content.lower():
                await message.channel.send('Hello here is where you can find the academic calendar for the year. https://www.ncat.edu/registrar/academic-calendar-pdfs/index.php')

    @commands.command()
    async def questions(self, ctx, time:int):
        print('here')






def setup(bot):
    bot.add_cog(Faq(bot))


```

A screenshot showing a test running the bot question respond feature:



how do i get a parking pass

Aggie Bot BOT Yesterday at 8:42 PM

Hello here is where you can find all parking pass related information. https://www.ncat.edu/campus-life/campus-enterprises/parking-and-transportation/parking-permits.php

Parking Permits

Information about obtaining parking permits for use on the North Carolina A&T campus.

Ryan Blount Yesterday at 8:46 PM

when do classes start

Aggie Bot BOT Yesterday at 8:46 PM

Hello here is where you can find the academic calendar for the year. https://www.ncat.edu/registrar/academic-calendar-pdfs/index.php

Registrar Academic Calendar

${_EscapeTool.xml($currentPage.metadata.description)}

Team 1 Project Journal

Fall 2021

Submitted By: Terrell Pinkney

Hau Le, Ryan Blount, Sage-Michael Brown, Terrell Pinkney

Contents

1 WEEK ENDING: NOVEMBER 18, 2021 31

1.1 ACCOMPLISHMENT 1: FAQ CHANNEL WELCOME MESSAGE FROM BOT 31

1.2 ACCOMPLISHMENT 2: TEST THE BOT FOR BUGS WHEN RESPONDING TO QUESTIONS 31

1.3 INDIVIDUAL GROUP MEMBER CONTRIBUTION: HAU LE 31

1.4 INDIVIDUAL GROUP MEMBER CONTRIBUTION: RYAN BLOUNT 31

1.5 INDIVIDUAL GROUP MEMBER CONTRIBUTION: SAGE-MICHAEL BROWN 31

1.6 INDIVIDUAL GROUP MEMBER CONTRIBUTION: TERRELL PINKNEY 31

1.7 ISSUES AND SOLUTIONS 31

1.8 GOALS FOR NEXT WEEK 31

1.8.1 Goal 1: Continue adding features to the Discord bot 31

11 Week Ending: November 18, 2021

11.1 Accomplishment 1: FAQ channel welcome message from bot

- We programmed the bot to print a welcome message in the FAQ channel to specify what questions can be answered from the bot.

11.2 Accomplishment 2: Test the bot for bugs when responding to questions

- We tested the bot’s functionality when responding to specific questions from a user in the FAQ channel.

11.3 Individual Group Member Contribution: Hau Le

- Created questions and answers relating to Financial Aid for the bot when answering student questions.

11.4 Individual Group Member Contribution: Ryan Blount

- Programmed the bot to respond to questions that team member Sage asked for testing purposes.

11.5 Individual Group Member Contribution: Sage-Michael Brown

- Helped team member Ryan test the bot by answering the different types of questions in the FAQ channel.

11.6 Individual Group Member Contribution: Terrell Pinkney

- Created questions and answers relating to Parking for the bot when answering student questions.

11.7 Issues and Solutions

- Issues
    - There was an issue with the bot answering questions for specific categories.
- Solutions
    - To fix the issue with answering different types of questions, we programmed the bot to read specific keywords in the questions to decide how to answer.

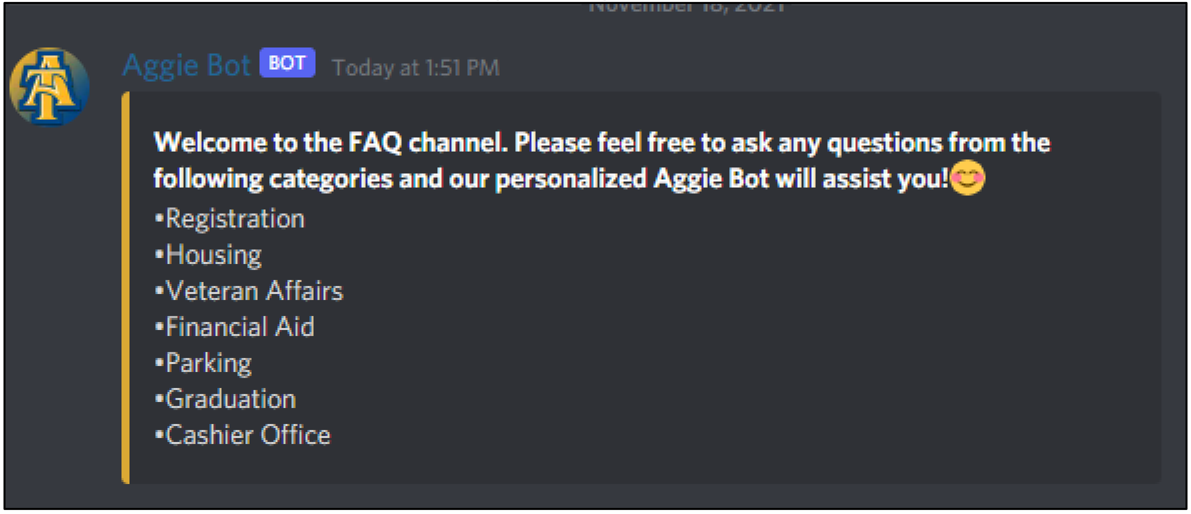
11.8 Goals for Next Week

11.8.1 Goal 1: Continue adding features to the Discord bot

- We plan to continue adding useful features to the bot which include:
    - Student feedback for professor lectures
    - Student listening parties and music features
    - More tutoring resource features
    - Student game nights and online game sessions.

Attach weekly work here

A screenshot of the bot greeting students in the FAQ channel:

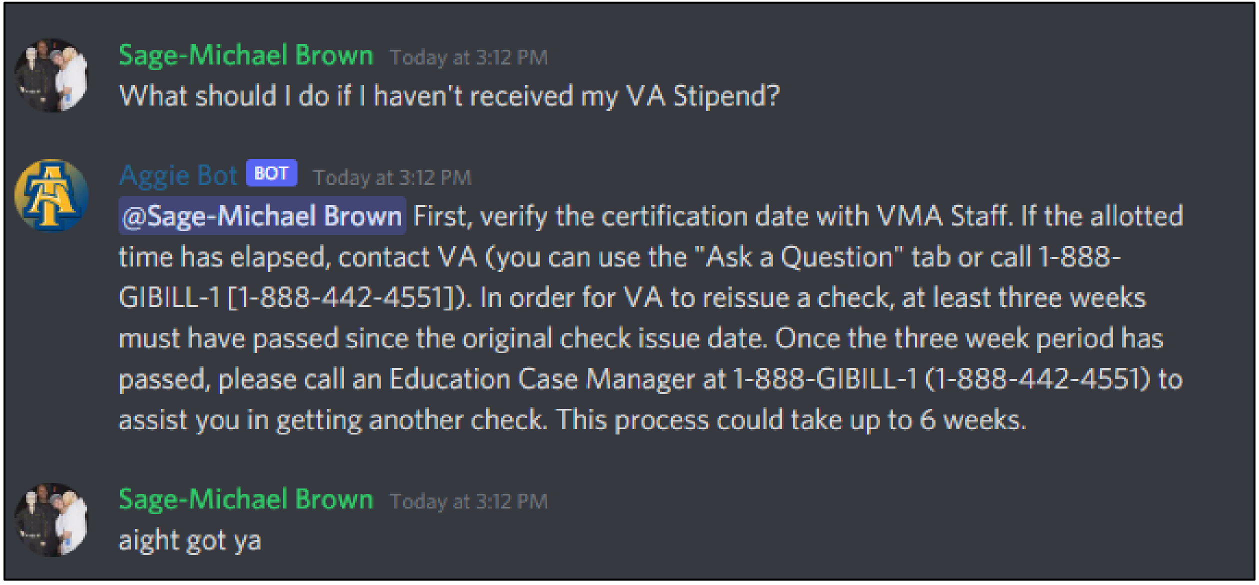


Aggie Bot BOT Today at 1:51 PM

Welcome to the FAQ channel. Please feel free to ask any questions from the following categories and our personalized Aggie Bot will assist you!😊

•Registration
•Housing
•Veteran Affairs
•Financial Aid
•Parking
•Graduation
•Cashier Office

A screenshot of the bot answering a question about the Veteran Affairs office:



Sage-Michael Brown Today at 3:12 PM
What should I do if I haven't received my VA Stipend?

Aggie Bot BOT Today at 3:12 PM
@Sage-Michael Brown First, verify the certification date with VMA Staff. If the allotted time has elapsed, contact VA (you can use the "Ask a Question" tab or call 1-888-GIBILL-1 [1-888-442-4551]). In order for VA to reissue a check, at least three weeks must have passed since the original check issue date. Once the three week period has passed, please call an Education Case Manager at 1-888-GIBILL-1 (1-888-442-4551) to assist you in getting another check. This process could take up to 6 weeks.

Sage-Michael Brown Today at 3:12 PM
aight got ya

Executive Summary

3.1 Project Overview (Problem Statement)

The intended use for this server would be to allow the students at NCAT to be able to make connections with other students and also have 24/7 access to tutoring or homework help. Students will be able to join voice chats and share their screens at the same time. This would also be a great place to send campus-wide announcements and news out to students. The server will help students interact with each other safely without the need to meet in person. This will help students with interacting with each other during the COVID-19 pandemic and potentially help decrease the spread of the virus between students. This gives more free range in terms of communicating to students, whether it is to update users, welcome freshmen, or provide students with the notification of opportunities available. We hope to provide a safe environment for all students of every community to be welcomed and free to express themselves in a way deemed appropriate.

4. Project Description

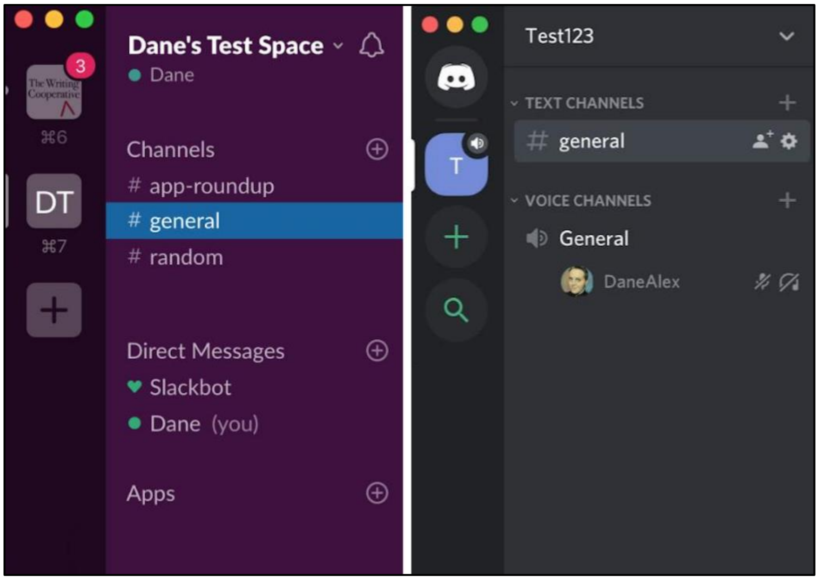
Background Information about Discord and its features

The project will be a discord server that makes use of the automated chatbots featured in discord. Discord is a group-chatting app that gives gamers or other individuals a place to build communities and communicate with each other. Inside of discord, users can create servers that can manage members inside the server, topics, rules, and channels. Discord also allows users to make video/voice calls with other people and gives gamers access to stream games. Discord works like other apps such as Slack, Microsoft Teams, and HeySpace. Our project will make use of the chatbot feature that is offered by Discord. Discord bots perform different automated tasks in a server, such as moderating what content appears, banning users from the server, and welcoming new members who join the server. Discord allows users to create bots with computer programming knowledge.

4.1 Product Context

Several community-based messaging and or VoIP platforms such as Microsoft Teams, Slack, and Telegram provide similar features to Discord at a more secure and enterprise-level with communication and professionalism. Group Chatting Applications such as Facebook Messenger, WhatsApp, and GroupMe also provide similar messaging services to Discord. Unlike similar platforms, Discord's is geared towards gamers and the gaming community/market. Discord allows users to stream gameplay from video games, create a variety of community servers with different guidelines and permissions and support a simple user-friendly interface.

Slack and Discord Comparison



4.2 User Characteristics

General Customer Profiles

- Student and or facility members
- No experience is required as there are short tutorials on the basic use of discord.
- Must have the ability to operate a computer/cellphone
- Social, intuitive characteristics

4.3 Assumptions

- Staff Availability
    - If there is a low staff percentage of workers for each Channel in the server, then several day-to-day maintenances, security, and operation tasks will become difficult to do.
- Friendly Interface
    - While there are plenty of free online communication programs out there, Discord still stands out because it has a wide variety of chat options. It combines all the best features of commonly used programs, such as Skype and Slack, with a friendly interface. An application wouldn't be of much use if it had a complex interface and was difficult to use.
- High Cost
    - If we use Discord to support a small group of A&T students, it won't cost much, but to develop and use more features we need to pay a lot of money for the service on Discord Server.

4.4 Constraints

- Cost of running the MS Access Database on the AWS PostgreSQL Cloud service
- Cost of keeping the server running on Discords Server Plan
- Receiving access to NC A&T Student information and database
- User limits
    - Limits on users and channels per server preclude new fans from popular servers and limit growth for indie developers.
- Memory space
    - Discord can take up a fair amount of RAM if you install Discord on the device or use it on a web browser.
- Safety in large servers
    - Discord is a communication app between students, which means it attracts its fair share of trolls and unethical users like hacking, harassment. Our private servers should be safe, and we need a high-security level for them.

4.5 Dependencies

Channel Management

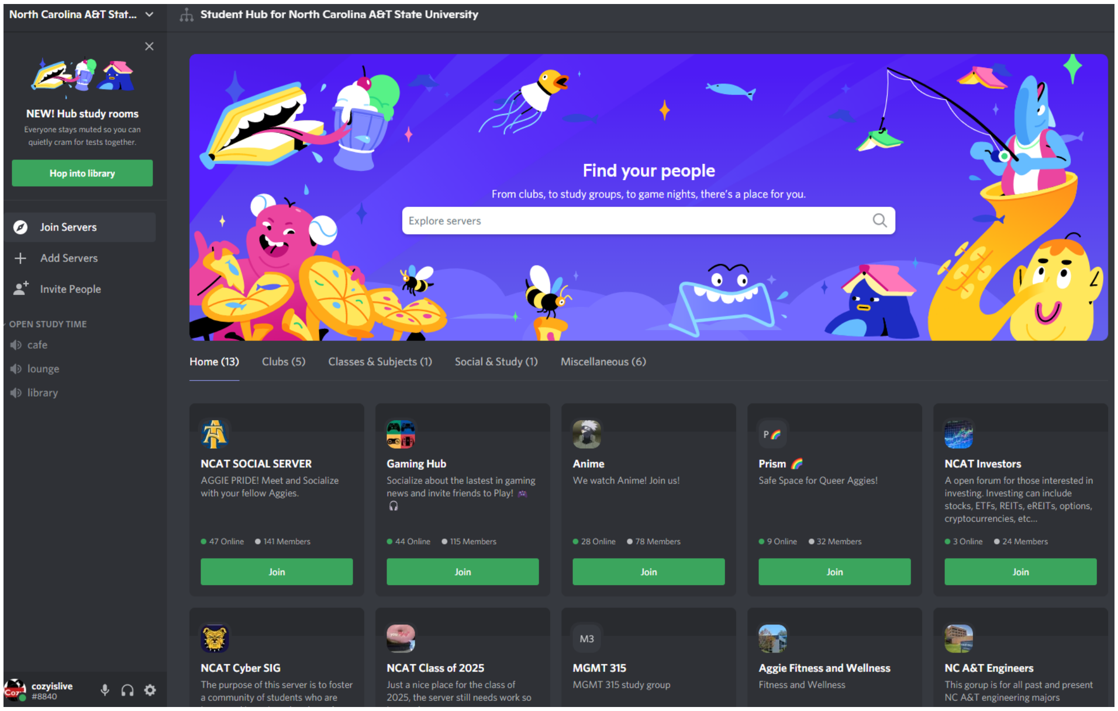
For each channel in the server, there must be an admin who controls the permissions and restrictions of that specific channel. Each Channel will require administrators so that they can facilitate rules and permissions for all of the different channels in the server.

5. Requirements

1. Requirements for the discord would consist of having the discord downloaded.
2. We would also need to make sure our bot is online running on an online server 24/7.
3. Access to the NCAT student database would also be required if we wanted to verify that each student was an NCAT student.
4. We would need administrators and staff and faculty to help keep the server running.
5. Tutors will be needed to help students with their work.

1.1 User Interface Requirements

The requirements for discord are minimal in terms of computer specifications, both when it comes to RAM and processor power. Discord is an application that can run on multiple operating systems and devices, whether it be a phone, computer, or tablet. The interface greets the user with simple and easy ways to find communities that have a common interest whether it’s looking for a gaming, social, or educational wise. We hope to use this to our advantage to create a secure place for students to interact and socialize with each other like we once did in the Student Center. To achieve this, we are both the creators and users in the server we try to apply what we as students know we would like if one was created before this. Things such as club allocation and year selection enable us to ensure students are within the same area as their counterparts whether it is a classification-based or college department-based.



Everyone stays muted so you can quietly cram for tests together.

From clubs, to study groups, to game nights, there's a place for you.

NCAT SOCIAL SERVER
AGGIE PRIDE! Meet and Socialize with your fellow Aggies.
47 Online 141 Members

Gaming Hub
Socialize about the lastest in gaming news and invite friends to Play!
44 Online 115 Members

Anime
We watch Anime! Join us!
28 Online 78 Members

Prism 🌈
Safe Space for Queer Aggies!
9 Online 32 Members

NCAT Investors
A open forum for those interested in investing. Investing can include stocks, ETFs, REITs, eREITs, options, cryptocurrencies, etc...
3 Online 24 Members

NCAT Cyber SIG
The purpose of this server is to foster a community of students who are

NCAT Class of 2025
Just a nice place for the class of 2025, the server still needs work so

MGMT 315
MGMT 315 study group

Aggie Fitness and Wellness
Fitness and Wellness

NC A&T Engineers
This gorup is for all past and present NC A&T engineering majors

1.2 Usability

We use a bot to aid with controlling the server as well as collect reports if there are issues within the server. The Aggie Bot will do most of the main interactions in terms of communicating and allocating roles to users. This Aggie Bot will monitor chats and ensure that the users are who they are to ensure safety. The bots function uses a "!" as the initial starting point for the function for the bot to realize it is being called. We as moderators will need to do small tasks such as monitoring the Aggie Bot to ensure that it is working properly and taking reports that the bot is unable to aid users with.

Things such as user "eye candy" which includes emotes and banners are all a part of the user interface. For people not familiar with discord, the bot is there to aid in terms of learning commands, and users such as us the creators, are there to aid in any way possible.

1.3 Performance

The server will be able to support about 20,000 users who are NC A&T students and faculty members. The Aggie bot will categorize each student and faculty member into their specific department/facility when joining the server for the first time. The Aggie bot will be able to answer questions and give feedback to any user on the server under any circumstances for each channel and category.

1.3.1 Capacity

The maximum number of students and staff that could be supported would be 200,000.

1.3.2 Availability

- The Server will operate 24/7 Monday-Friday from 6:30 AM to 10:30 PM
- Level of availability required
- The Server will be accessible from any location as long as the user has access to a Wi-Fi connection and a cellular or digital device (cellphone and or computer)
- There will be scheduled downtimes on the server

  - Students will be notified a week before scheduled maintenance
- During maintenance times students will be unable to access specific channels in the server.
  - All student departments and colleges will still be accessible during maintenance hours.

1.4 Manageability/Maintainability

1.4.1 Monitoring

For each category in the server, there must be an administrator who oversees all users that join and interact with each channel. Each administrator of a particular channel will be given the ability to allow students to view, join, and communicate in their channel.

1.4.2 Maintenance

- Every semester, the student database will be updated to include newly admitted students, international students, and transfer students.
- Twice every month the server will undergo necessary maintenance to ensure that each channel and category is working properly
- Every week the Aggie Bot will be updated with new scripts to better handle different scenarios and interactions with students.

1.4.3 Operations

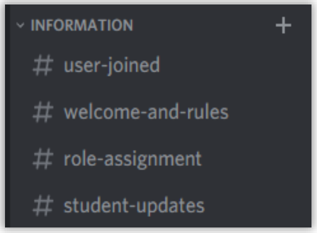
- The NCAT server will be backed up by Discords online database
- The Ticketing Tool will operate normally with student interactions

1.5 System Interface/Integration

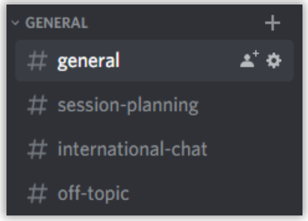
The server will be integrated with a database server created from Microsoft Access. Details about the Access Database and Server integration will be discussed primarily in the 1.7 Data Management Section.

1.5.1 Systems Interfaces

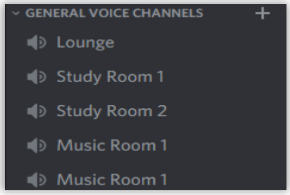
The homepage of the server will consist of different categories that will have different channels under each category. Each of the channels will relate to different facilities and services affiliated with NC A&T. The first category will be the information section and will have general information about student updates and will establish rules that must be followed in the server.



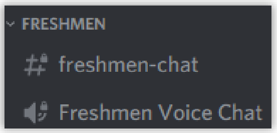
The general category section will have information for international students and allow students to plan group or individual sessions via voice calls or video calls.

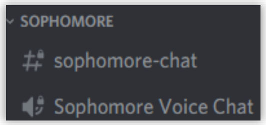


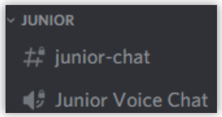
The general voice channel will allow students to join different study rooms to talk amongst each other and discuss student affairs and events.

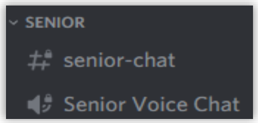


After initially joining the server, each student will be categorized by their grade level i.e. Freshman, sophomore, junior, and senior. Each student, through specific permissions and restrictions, will be able to join and collaborate in their respective grade levels in different student categories.

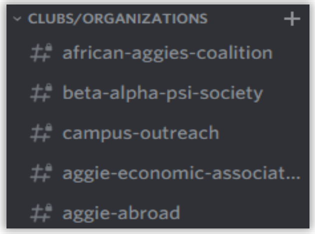

FRESHMEN
freshmen-chat
Freshmen Voice Chat


SOPHOMORE
sophomore-chat
Sophomore Voice Chat

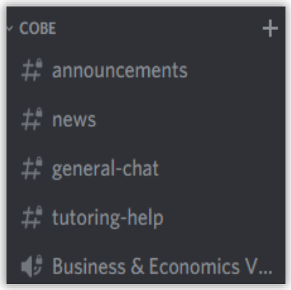

JUNIOR
junior-chat
Junior Voice Chat

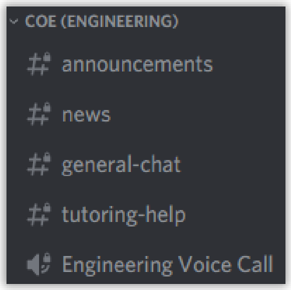

SENIOR
senior-chat
Senior Voice Chat

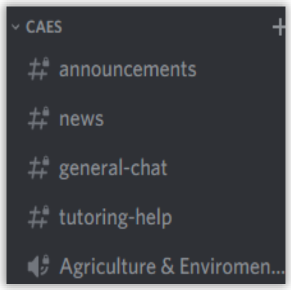
Clubs will also be playing a big role in the server as well. We have created a club/organization category that will allow students to join and communicate with their club members and leaders. Not all students will have access to viewing these different clubs and organizations as club leaders will be given administrative roles that will allow them to give specific students access to viewing and joining clubs.

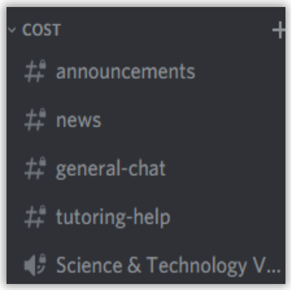

CLUBS/ORGANIZATIONS
african-aggies-coalition
beta-alpha-psi-society
campus-outreach
aggie-economic-associat...
aggie-abroad

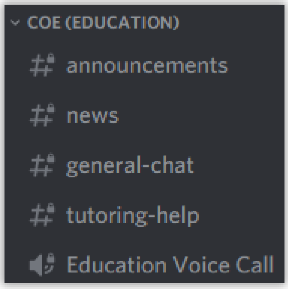
We created each of the categories so that the aggie bot can categorize each student into their respective college department and organization. Inside each category, some channels provide useful services for students. The first few categories are for each grade level. Students are not restricted with their grade level chats as they can join other grade level chats if they wish to. Inside each college category, we have an announcement and news channel, a general chat for students, tutoring help for students in the department, and a voice channel for students to communicate with each other. In the clubs or organization category, we have different club channels. Not just anybody can view each channel on the server. The discord server is divided into roles based on classification and department. For example, if a computer science student joined the server, they would be given the role of the college of engineering school (COE). Once they receive their role, they will only be able to view their specific school/department. This also applies to clubs as students who are not members of a specific club will not have access to viewing the club. For a student to access an organization, they must be a registered A&T student. After this, club leaders can assign students to specific club roles, which will allow them access to view the club or organization.

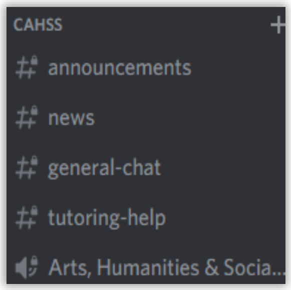

COBE
announcements
news
general-chat
tutoring-help
Business & Economics V...

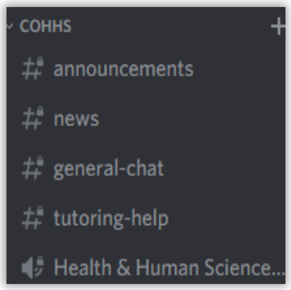

COE (ENGINEERING)
announcements
news
general-chat
tutoring-help
Engineering Voice Call


CAES
announcements
news
general-chat
tutoring-help
Agriculture & Enviromen...

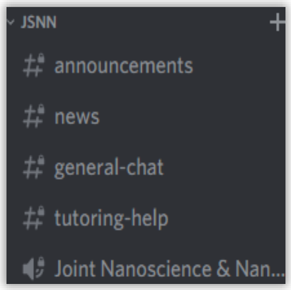

COST
announcements
news
general-chat
tutoring-help
Science & Technology V...


COE (EDUCATION)
announcements
news
general-chat
tutoring-help
Education Voice Call


CAHSS
announcements
news
general-chat
tutoring-help
Arts, Humanities & Socia...



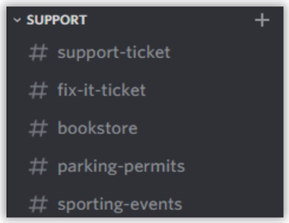
COHHS
announcements
news
general-chat
tutoring-help
Health & Human Science...



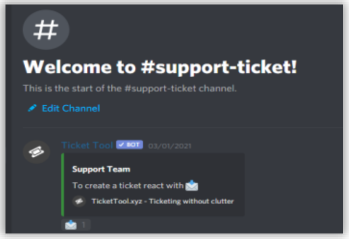
JSNN
announcements
news
general-chat
tutoring-help
Joint Nanoscience & Nan...

As you can see towards the left, there are multiple department categories so that student can easily find their department when initially joining the server for the first time.

The Ticket Tool is a student support tool that allows us to improve our quality of service and response time. We have provided private support channels between staff and students. Moreover, we have created five different channels in the support service section, and they will help us categorize exactly which problems students will need help with. The first channel is #support-ticket which allows students to request some different kind of general support. The second one is #fix-it-ticket will take students to a web user interface, and require them to log in, so they can find the solutions related to IT. Similarly, #bookstore provides students another web user interface with all information of textbook, supplies, electronic, returns, and refunds policy. Besides that, we also have channels that will help students get their parking permits easily or any problems related to parking. Lastly is the sporting events channel that includes sports schedules around school as well as the sports clubs, and we can also help students sign up for each event or buy tickets.



SUPPORT
support-ticket
fix-it-ticket
bookstore
parking-permits
sporting-events



Welcome to #support-ticket!

This is the start of the #support-ticket channel.

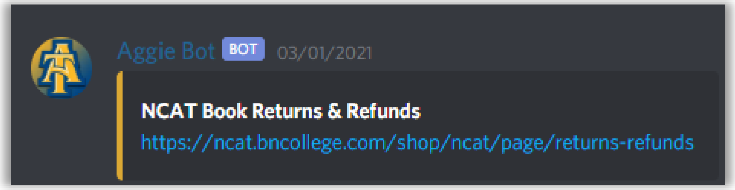
Edit Channel

Ticket Tool BOT 03/01/2021

Support Team

To create a ticket react with 📩

TicketTool.xyz - Ticketing without clutter



Aggie Bot BOT 03/01/2021

NCAT Book Returns & Refunds

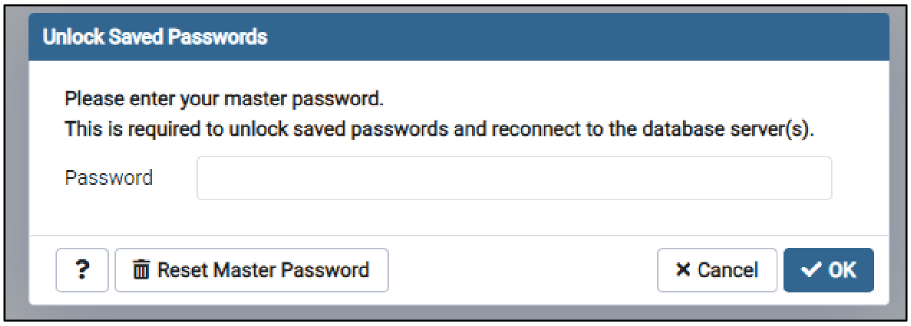
https://ncat.bncollege.com/shop/ncat/page/returns-refunds

1.6 *Security*

1.6.1 *Protection*

For this part of the project, we decided to use PostgreSQL to have a test database to access student information. We create a sample student database to test authentication and clearance in the server. To access the database we created, it will need an encrypted password which only members of the group have access to. With this, we hope to be able to access student names and banner IDs to ensure that the server is secure and no outside user who is not a member of the NCAT. We would like for graduate-level students to be given administrative privileges so that they can moderate each different server. A useful feature of discord is the server logs, which allow us to see any changes that have been made in the server from moderators or administrators.

PostgreSQL Student Database Locked & Secured



Please enter your master password.
This is required to unlock saved passwords and reconnect to the database server(s).

1.6.2 Authorization and Authentication

Users must identify themselves when joining the server for the first time. This is to avoid information and data leaks from each server. We will use each student’s unique Banner ID number to authorize and establish that each person is a current NCAT student. Authorization is the only thing we can ensure fully and completely to students and a safe environment to meet and talk with friends whether it’s a group project or just to communicate with each other. expletives and improper communication using obscene language.

1.7 Data Management (Sage)

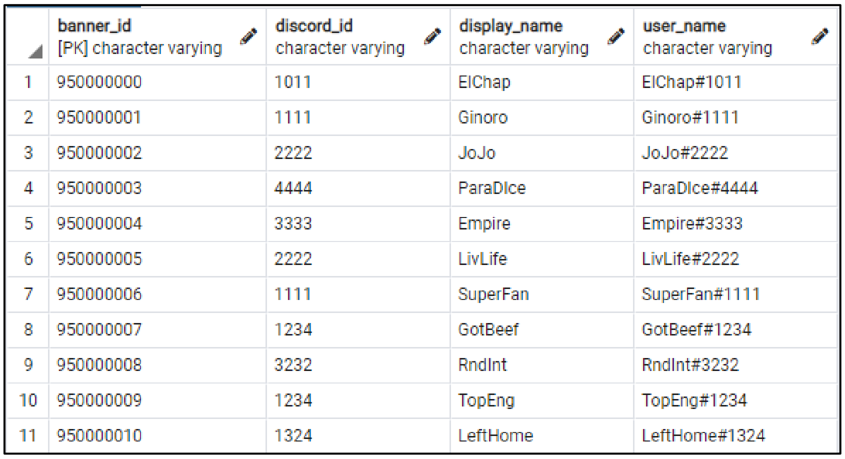
Initial

Due to this hopefully being a fully functional part of being an Aggie, it requires data that is owned by the university. This information includes names, banner ids, and any information that will aid in controlling who enters the server and leaves. The main constraint with this is that we have no access to student data to test, so this is the reasoning for a false data set with data that has no reliance on A&T information. The data entries we use are common things every student has being a major, classification, banner ID, and date of birth. With that, we can separate students to be grouped based on department and classification. as well as clubs and groups. There are 56 entries within the database, and all data sets have values for each section. Using the banner ID as the primary key for most tables within the database.

Formulas used in the database:

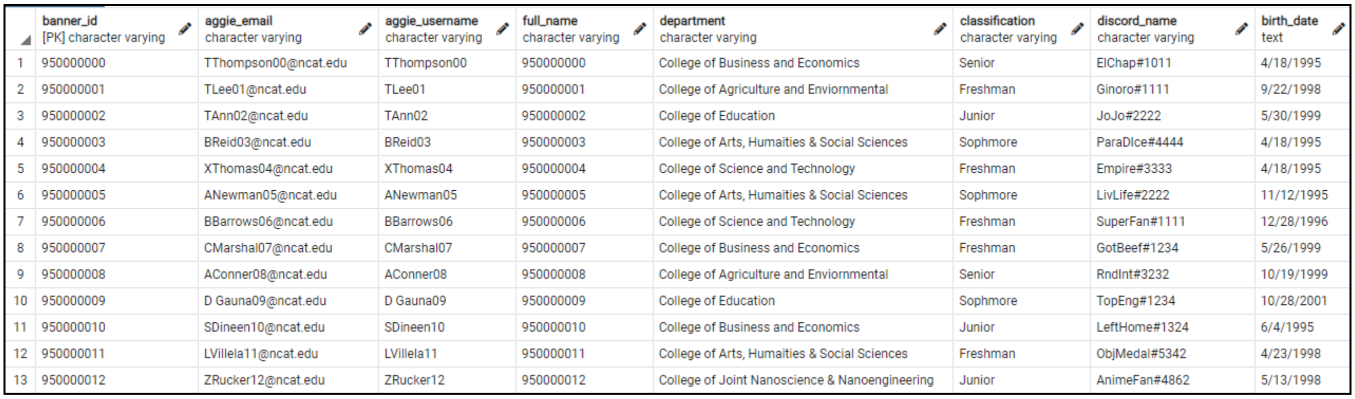
- Age of Individual-Age: DateDiff("yyyy",[Date of Birth],Now())
- Full Discord ID- [Discord Name] & "#" & [Discord ID]
- Individual Full name-[First Name] & " " & [Last Name]
- Full Name of individual-Left([First Name],1) & [Last Name] & Right([Banner ID],2)
- Student Email-[Aggie Username] & "@ncat.edu"

Sample student data for each NCAT student’s discord account name:



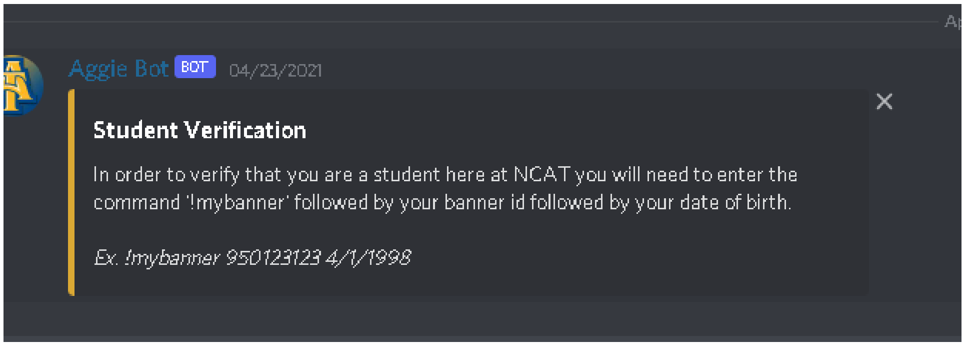
	banner_id [PK] character varying	discord_id character varying	display_name character varying	user_name character varying
1	950000000	1011	ElChap	ElChap#1011
2	950000001	1111	Ginoro	Ginoro#1111
3	950000002	2222	JoJo	JoJo#2222
4	950000003	4444	ParaDIce	ParaDIce#4444
5	950000004	3333	Empire	Empire#3333
6	950000005	2222	LivLife	LivLife#2222
7	950000006	1111	SuperFan	SuperFan#1111
8	950000007	1234	GotBeef	GotBeef#1234
9	950000008	3232	RndInt	RndInt#3232
10	950000009	1234	TopEng	TopEng#1234
11	950000010	1324	LeftHome	LeftHome#1324

Sample student data that includes detailed NCAT student details:



	banner_id [PK] character varying	aggie_email character varying	aggie_username character varying	full_name character varying	department character varying	classification character varying	discord_name character varying	birth_date text
1	950000000	TThompson00@ncat.edu	TThompson00	950000000	College of Business and Economics	Senior	ElChap#1011	4/18/1995
2	950000001	TLee01@ncat.edu	TLee01	950000001	College of Agriculture and Enviornmental	Freshman	Ginoro#1111	9/22/1998
3	950000002	TAnn02@ncat.edu	TAnn02	950000002	College of Education	Junior	JoJo#2222	5/30/1999
4	950000003	BReid03@ncat.edu	BReid03	950000003	College of Arts, Humaities & Social Sciences	Sophmore	ParaDIce#4444	4/18/1995
5	950000004	XThomas04@ncat.edu	XThomas04	950000004	College of Science and Technology	Freshman	Empire#3333	4/18/1995
6	950000005	ANewman05@ncat.edu	ANewman05	950000005	College of Arts, Humaities & Social Sciences	Sophmore	LivLife#2222	11/12/1995
7	950000006	BBarrows06@ncat.edu	BBarrows06	950000006	College of Science and Technology	Freshman	SuperFan#1111	12/28/1996
8	950000007	CMarshal07@ncat.edu	CMarshal07	950000007	College of Business and Economics	Freshman	GotBeef#1234	5/26/1999
9	950000008	AConner08@ncat.edu	AConner08	950000008	College of Agriculture and Enviornmental	Senior	RndInt#3232	10/19/1999
10	950000009	D Gauna09@ncat.edu	D Gauna09	950000009	College of Education	Sophmore	TopEng#1234	10/28/2001
11	950000010	SDineen10@ncat.edu	SDineen10	950000010	College of Business and Economics	Junior	LeftHome#1324	6/4/1995
12	950000011	LVillela11@ncat.edu	LVillela11	950000011	College of Arts, Humaities & Social Sciences	Freshman	ObjMedal#5342	4/23/1998
13	950000012	ZRucker12@ncat.edu	ZRucker12	950000012	College of Joint Nanoscience & Nanoengineering	Junior	AnimeFan#4862	5/13/1998

Current



Student Verification

In order to verify that you are a student here at NCAT you will need to enter the command '!mybanner' followed by your banner id followed by your date of birth.

Ex. !mybanner 950123123 4/1/1998

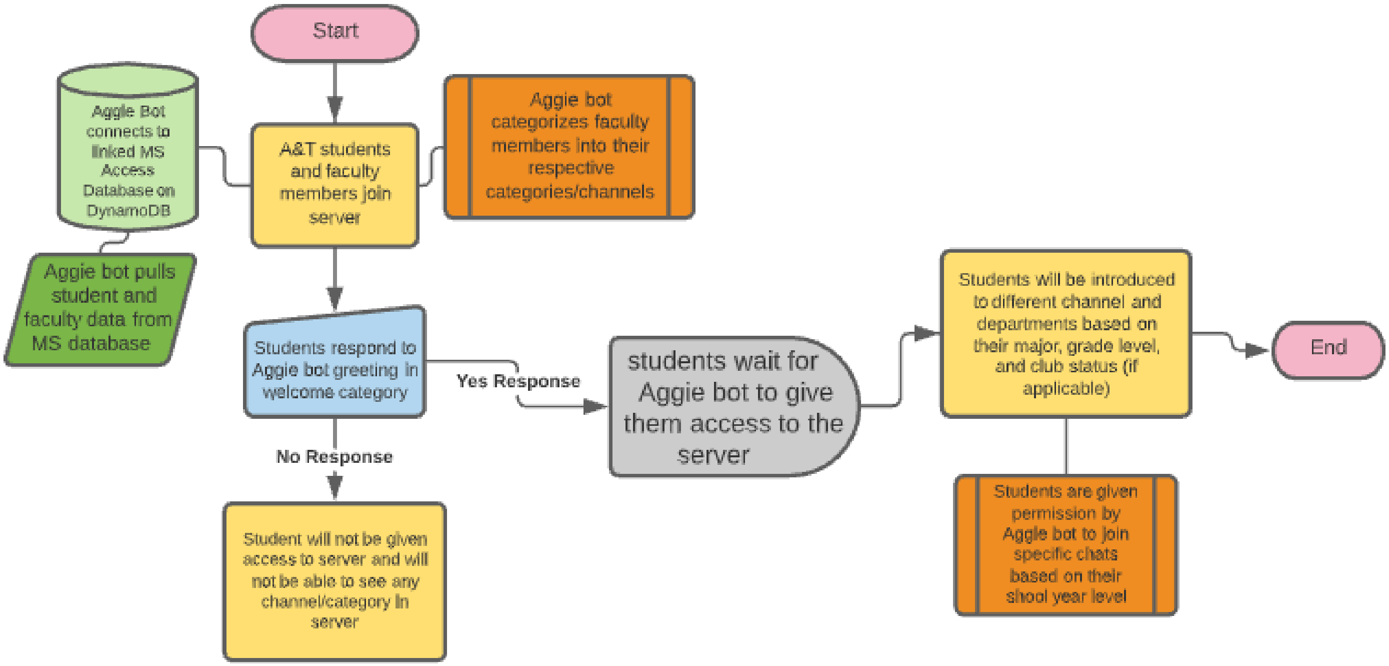
The image above displays the bot asking a user for his/her info to grant them access to the server, otherwise the user will not be able to open any chat or use any voice channel to connect with verified students. PGAdmin is where all this information is stored to be drawn from the bot, it stores the user's name, email, banner id, and discord id assuming that this is not the first time the user is using the application. With this, we make the verification more automated and less reliant on human intervention unless a problem occurs. Inside of pgadmin lies all user-based information such as what college they belong to, to their classification within the school.

1.8 Standard Compliance

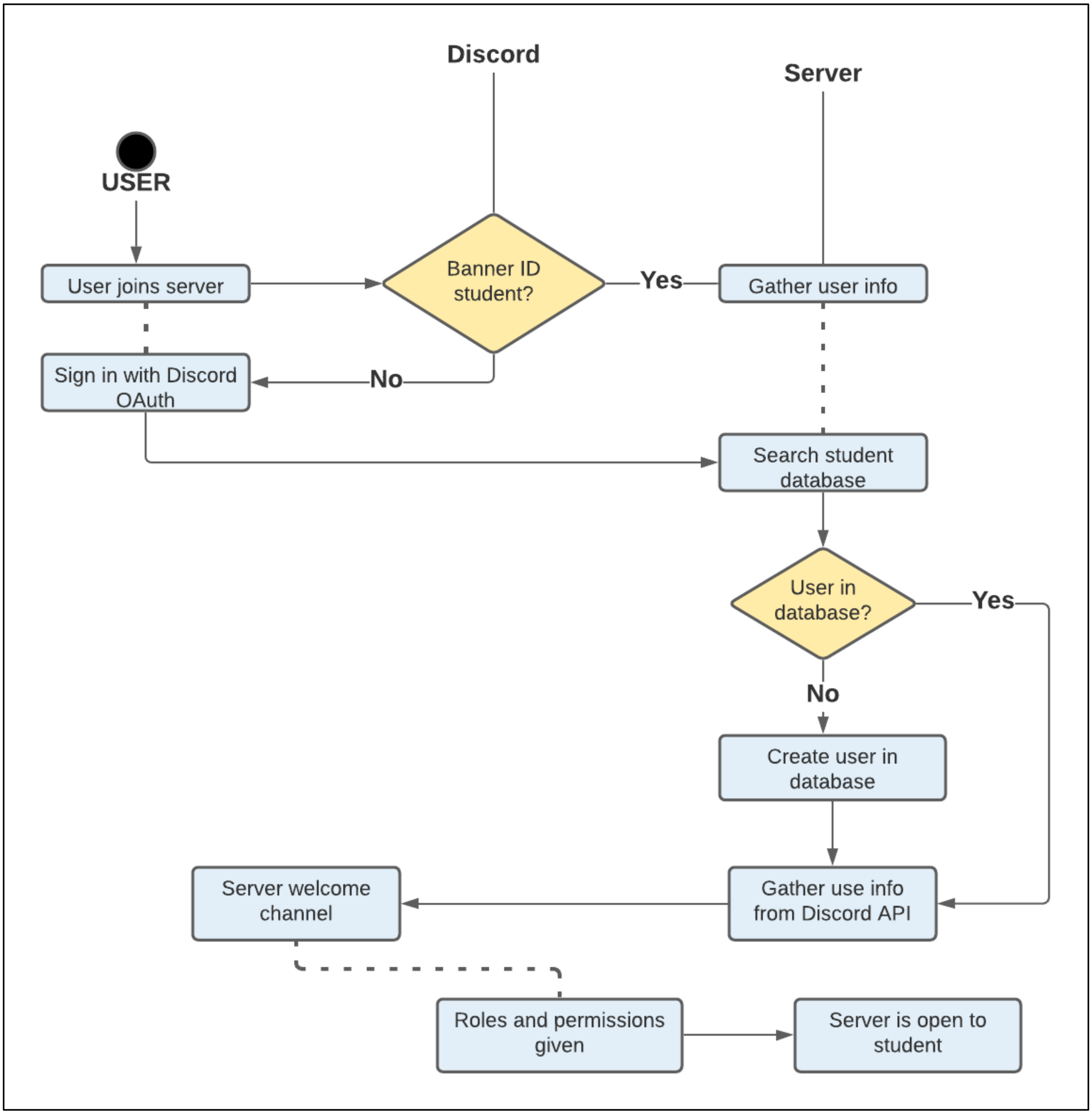
The Discord server will abide by all NC A&T State University policies, regulations, and guidelines and will not implore the safety and privacy of all NC A&T students and staff members.

6. Specifications/Implementation Plan/Flow Chart

1.9 User Flow Chart


Start
Aggie Bot connects to linked MS Access Database on DynamoDB
A&T students and faculty members join server
Aggie bot categorizes faculty members into their respective categories/channels
Aggie bot pulls student and faculty data from MS database
Students respond to Aggie bot greeting in welcome category
Yes Response
No Response
students wait for Aggie bot to give them access to the server
Students will be introduced to different channel and departments based on their major, grade level, and club status (if applicable)
End
Students are given permission by Aggie bot to join specific chats based on their shool year level
Student will not be given access to server and will not be able to see any channel/category in server

1.10 UML Diagram



1.11 Project Design Steps

(Step by step details from initial design to finished product)

Step 1. Designing the Server:

  a. Creating the channels and categories
    - To create the initial server and have the server up and functioning we had to first create the server on discord and assign each team member an administrative role.
    - After assigning administrative roles each team member created different channels and categories in the server.
    - Once all of the channels and categories were created, we started to add more features such as emojis.

  b. Creating NCAT A&T-themed emojis
    - We created a variety of A&T-themed emojis for the student to use when communicating in different channels on the server.

  c. Creating Roles and Permissions
    - The last step for creating the server was to create different roles for each of the categories in the serve. These roles help establish a student's identity in the server. These roles included each college department and student classification level.

Step 2. Programming the discord bot

  a. Adding python code to personalize the bot
    - The next designing phase was to create the Aggie bot which would facilitate and monitor each channel in the server. The bot would also authenticate each person that would join the server.
    - We created a python file that consisted of code that would allow the bot to complete different tasks in the server such as giving students detailed information about new and events, or support for issues on A&T web services.

Step 3. Creating the database

  a. Creating the first database on Microsoft Access
    - To test the server for functionality, we had to create a database filled with a randomly created NC A&T student database. This database was initially created on Microsoft Access and included information such as name, date of birth, banner ID, and aggie email.

  b. Exporting database data into Excel cvs. file
    - After creating the tables and queries of the database we exported the data into a cvs. file so that we could import the data into a database software called PostgreSQL.

  c. Transferring previous database data into PostgreSQL database
    - We then created a new student database on PostgreSQL and imported the data from the previous database into the new one.

Step 4. Linking the database with the server

  a. Adding code to the bot to link server to database
    - After we created the new database, we were able to add some code to the python file for the bot so that the server could have access to the database through AWS cloud services.
    - Once the python file was updated the server was able to work as intended.

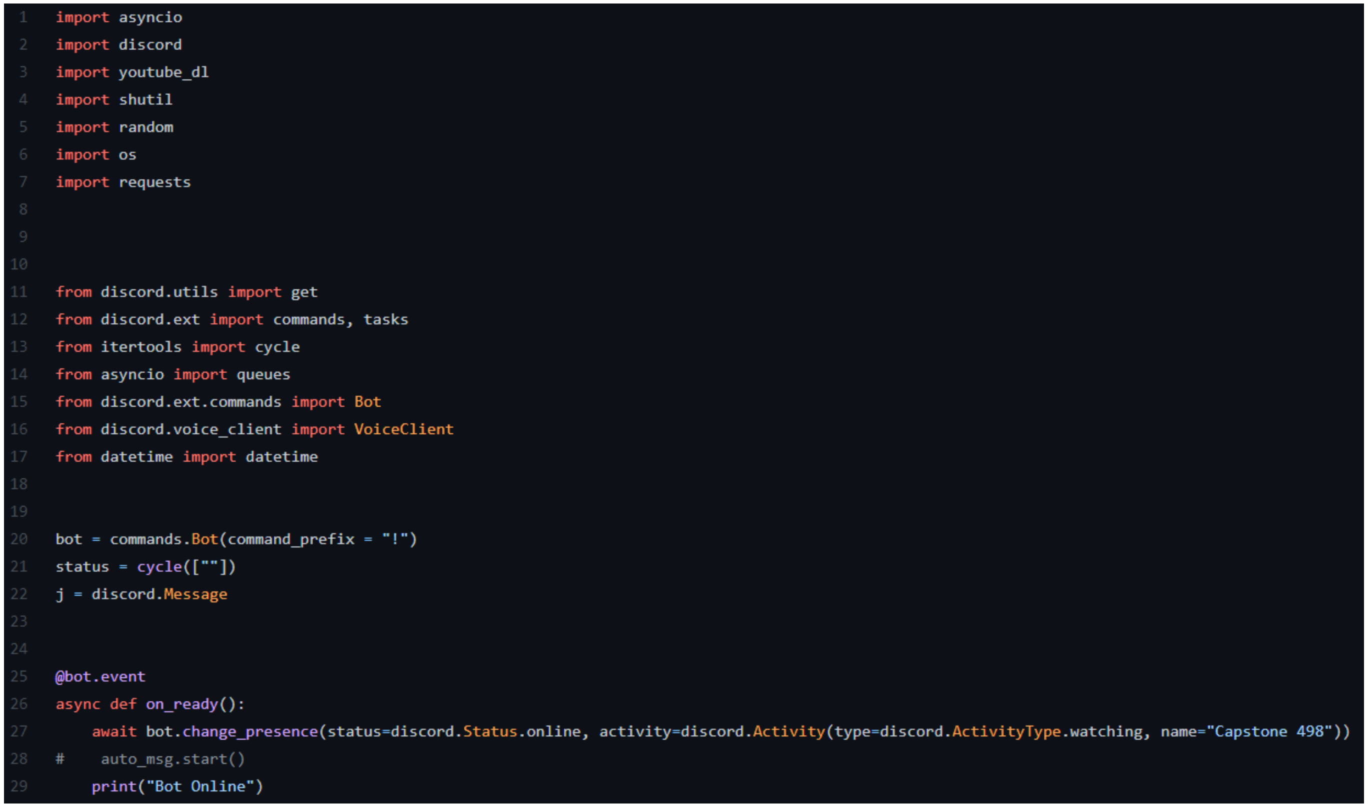
Step 5. Testing server functionality

  a. Test bot functionality with different features
    - Test the Aggie bot for asking Trivia questions in different channels
    - Test the Aggie bot for answering different questions in the FAQ channel

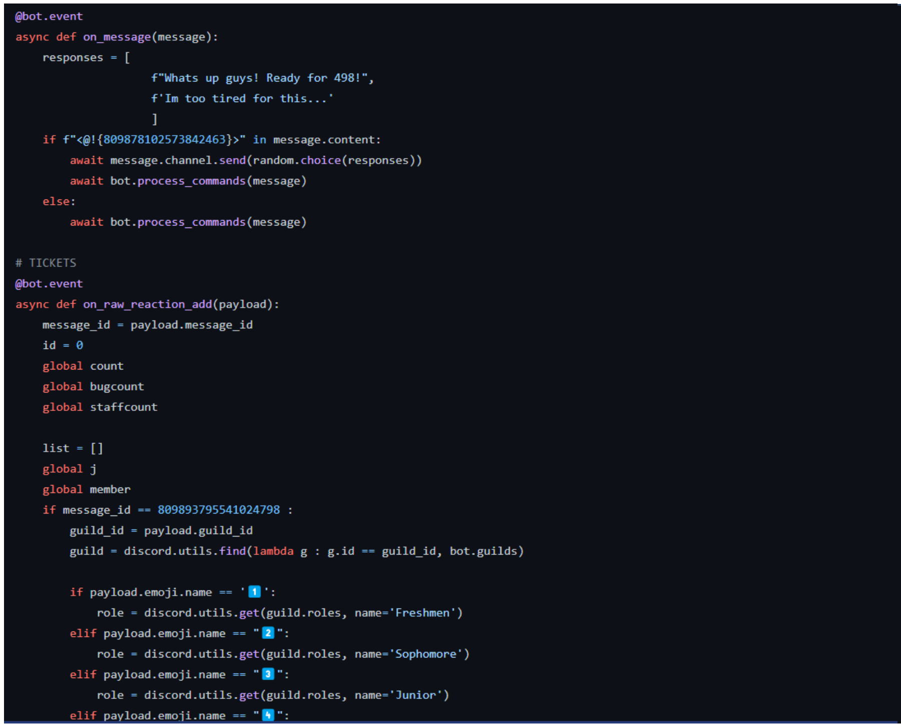
  b. Test server for roles
    - Test for when a new student initially joins the server
    - Test for the roles that are assigned to students who are new to joining the server.

1.12 Software Code

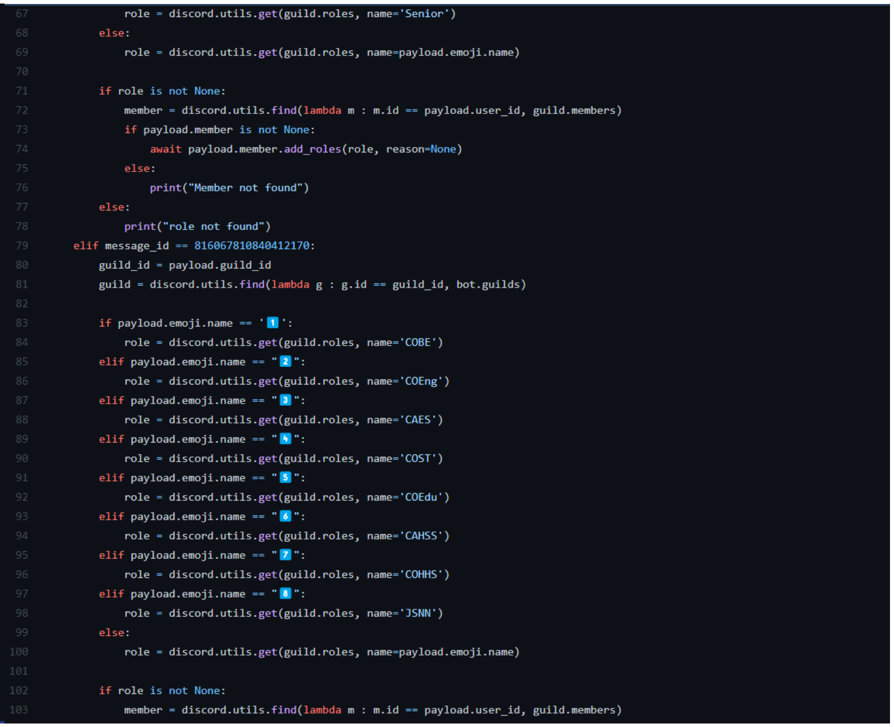
- Code for primary bot features



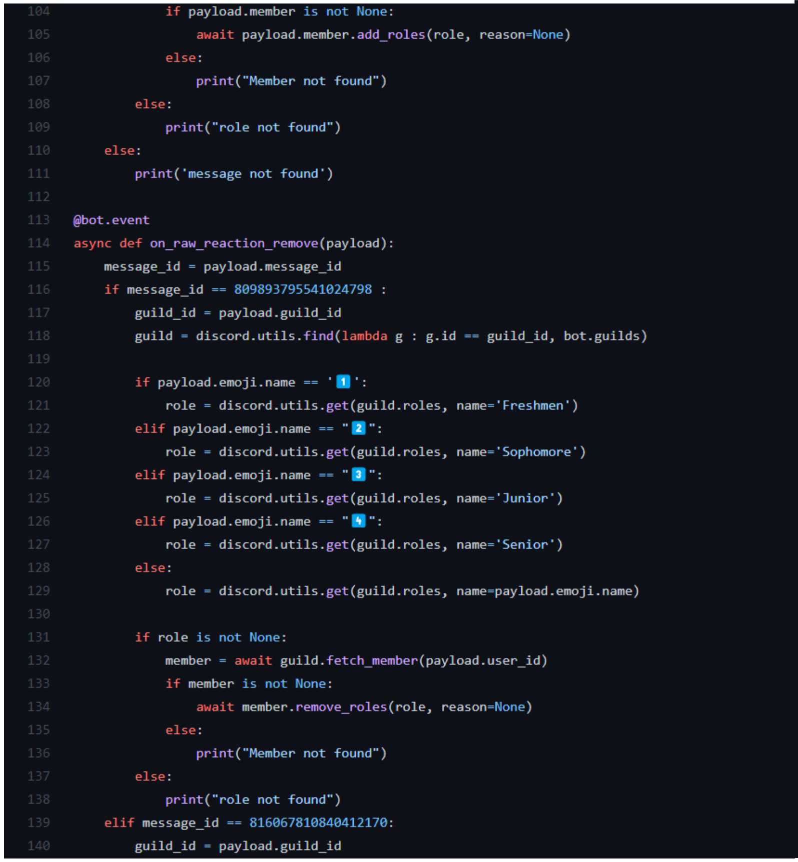
```python
import asyncio
import discord
import youtube_dl
import shutil
import random
import os
import requests



from discord.utils import get
from discord.ext import commands, tasks
from itertools import cycle
from asyncio import queues
from discord.ext.commands import Bot
from discord.voice_client import VoiceClient
from datetime import datetime


bot = commands.Bot(command_prefix = "!")
status = cycle([""])
j = discord.Message


@bot.event
async def on_ready():
    await bot.change_presence(status=discord.Status.online, activity=discord.Activity(type=discord.ActivityType.watching, name="Capstone 498"))
#    auto_msg.start()
    print("Bot Online")
```



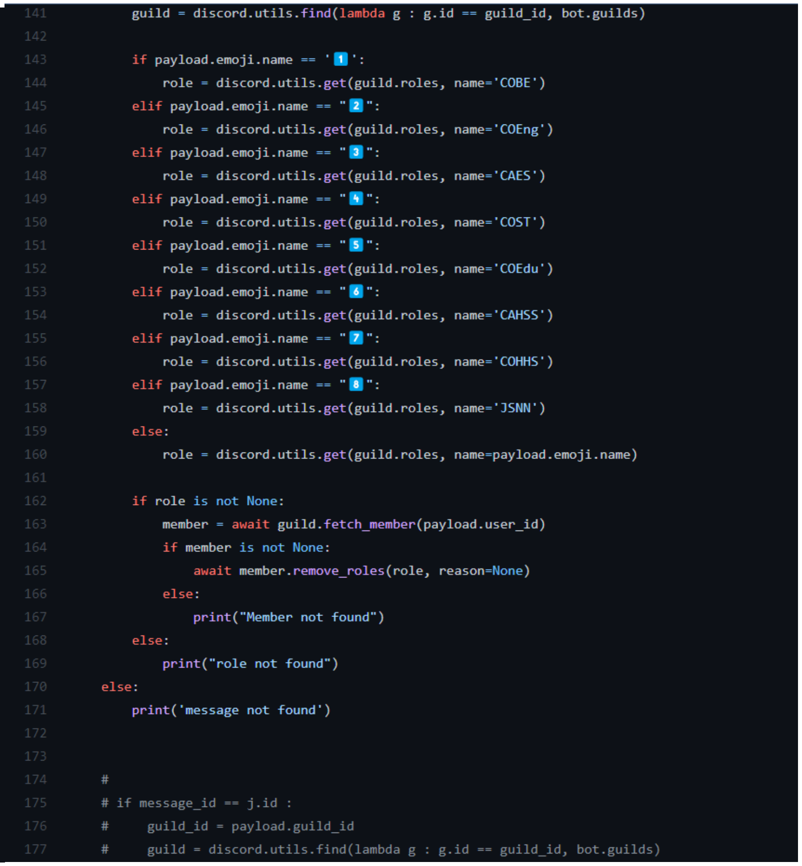
```
@bot.event
async def on_message(message):
    responses = [
                    f"Whats up guys! Ready for 498!",
                    f'Im too tired for this...'
                    ]
    if f"<@!{809878102573842463}>" in message.content:
        await message.channel.send(random.choice(responses))
        await bot.process_commands(message)
    else:
        await bot.process_commands(message)

# TICKETS
@bot.event
async def on_raw_reaction_add(payload):
    message_id = payload.message_id
    id = 0
    global count
    global bugcount
    global staffcount

    list = []
    global j
    global member
    if message_id == 809893795541024798 :
        guild_id = payload.guild_id
        guild = discord.utils.find(lambda g : g.id == guild_id, bot.guilds)

        if payload.emoji.name == '1️⃣':
            role = discord.utils.get(guild.roles, name='Freshmen')
        elif payload.emoji.name == "2️⃣":
            role = discord.utils.get(guild.roles, name='Sophomore')
        elif payload.emoji.name == "3️⃣":
            role = discord.utils.get(guild.roles, name='Junior')
        elif payload.emoji.name == "4️⃣":
```



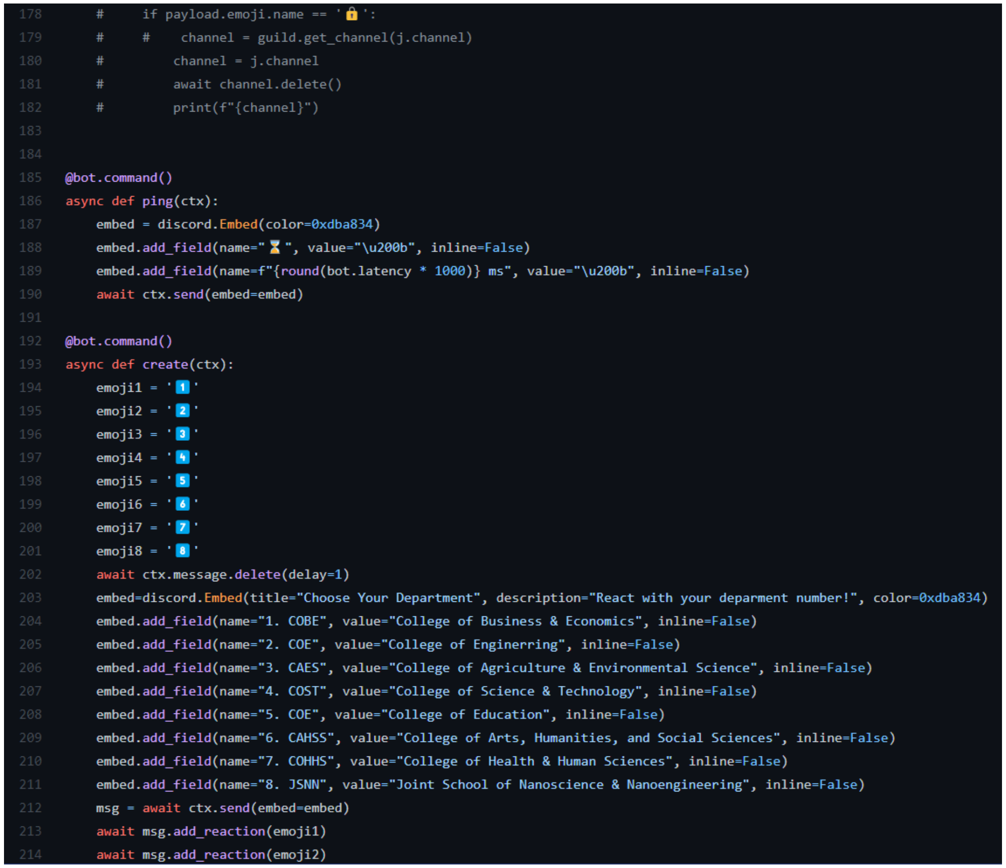
```
            role = discord.utils.get(guild.roles, name='Senior')
        else:
            role = discord.utils.get(guild.roles, name=payload.emoji.name)

        if role is not None:
            member = discord.utils.find(lambda m : m.id == payload.user_id, guild.members)
            if payload.member is not None:
                await payload.member.add_roles(role, reason=None)
            else:
                print("Member not found")
        else:
            print("role not found")
    elif message_id == 816067810840412170:
        guild_id = payload.guild_id
        guild = discord.utils.find(lambda g : g.id == guild_id, bot.guilds)

        if payload.emoji.name == '1️⃣':
            role = discord.utils.get(guild.roles, name='COBE')
        elif payload.emoji.name == "2️⃣":
            role = discord.utils.get(guild.roles, name='COEng')
        elif payload.emoji.name == "3️⃣":
            role = discord.utils.get(guild.roles, name='CAES')
        elif payload.emoji.name == "4️⃣":
            role = discord.utils.get(guild.roles, name='COST')
        elif payload.emoji.name == "5️⃣":
            role = discord.utils.get(guild.roles, name='COEdu')
        elif payload.emoji.name == "6️⃣":
            role = discord.utils.get(guild.roles, name='CAHSS')
        elif payload.emoji.name == "7️⃣":
            role = discord.utils.get(guild.roles, name='COHHS')
        elif payload.emoji.name == "8️⃣":
            role = discord.utils.get(guild.roles, name='JSNN')
        else:
            role = discord.utils.get(guild.roles, name=payload.emoji.name)

        if role is not None:
            member = discord.utils.find(lambda m : m.id == payload.user_id, guild.members)
```



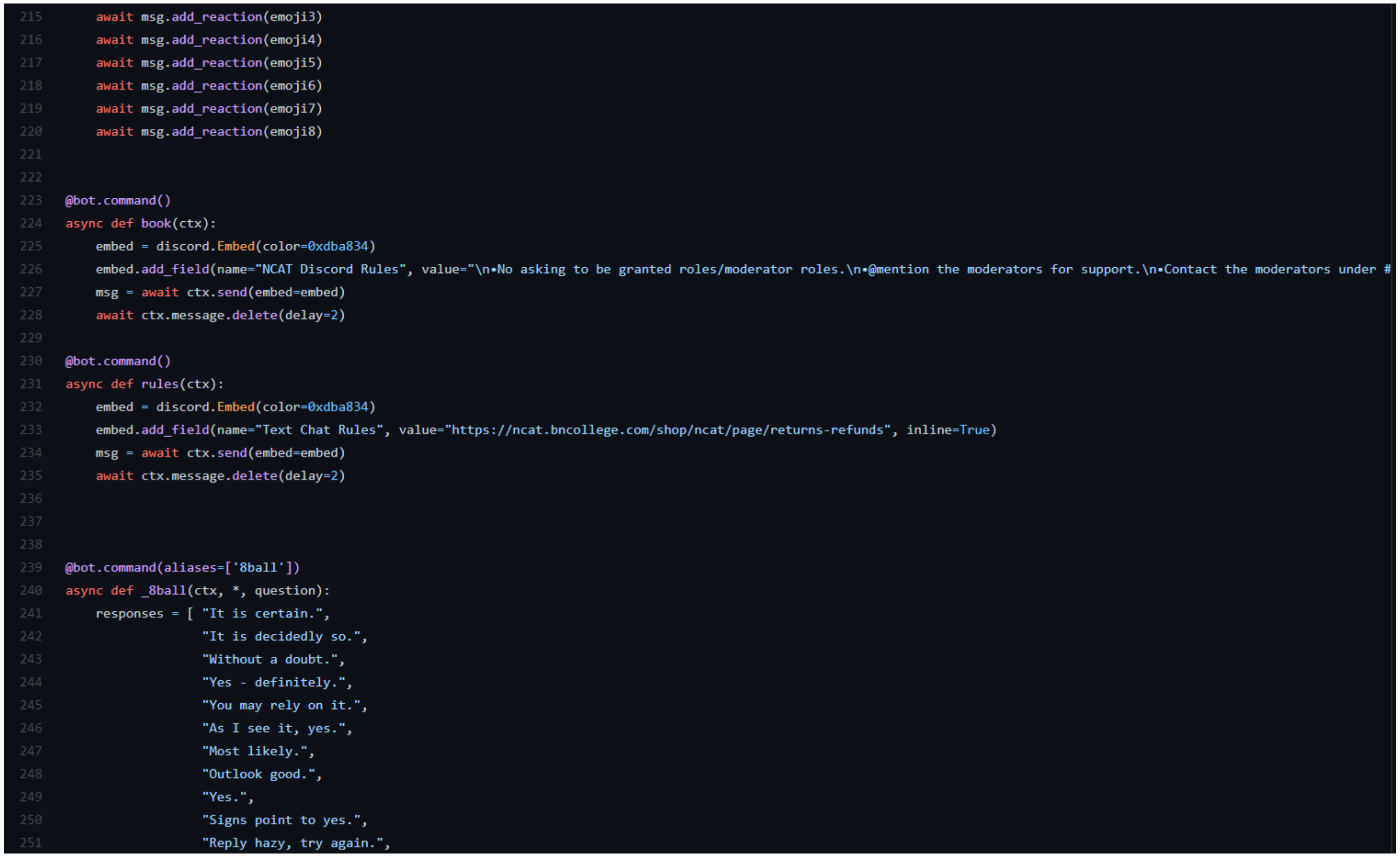
```
            if payload.member is not None:
                await payload.member.add_roles(role, reason=None)
            else:
                print("Member not found")
        else:
            print("role not found")
    else:
        print('message not found')

@bot.event
async def on_raw_reaction_remove(payload):
    message_id = payload.message_id
    if message_id == 809893795541024798 :
        guild_id = payload.guild_id
        guild = discord.utils.find(lambda g : g.id == guild_id, bot.guilds)

        if payload.emoji.name == '1️⃣':
            role = discord.utils.get(guild.roles, name='Freshmen')
        elif payload.emoji.name == "2️⃣":
            role = discord.utils.get(guild.roles, name='Sophomore')
        elif payload.emoji.name == "3️⃣":
            role = discord.utils.get(guild.roles, name='Junior')
        elif payload.emoji.name == "4️⃣":
            role = discord.utils.get(guild.roles, name='Senior')
        else:
            role = discord.utils.get(guild.roles, name=payload.emoji.name)

        if role is not None:
            member = await guild.fetch_member(payload.user_id)
            if member is not None:
                await member.remove_roles(role, reason=None)
            else:
                print("Member not found")
        else:
            print("role not found")
    elif message_id == 816067810840412170:
        guild_id = payload.guild_id
```



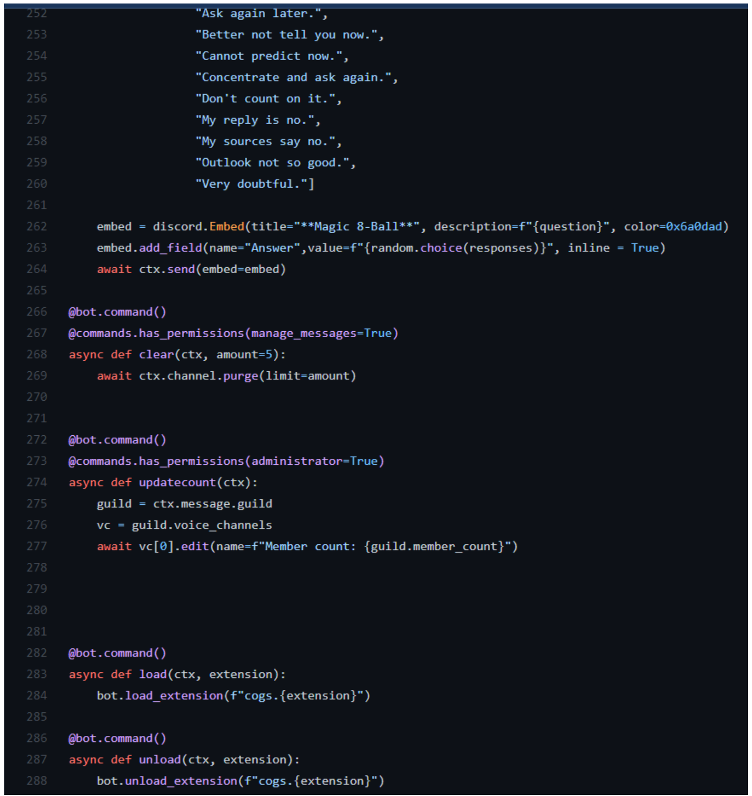
```
        guild = discord.utils.find(lambda g : g.id == guild_id, bot.guilds)

        if payload.emoji.name == '1️⃣':
            role = discord.utils.get(guild.roles, name='COBE')
        elif payload.emoji.name == "2️⃣":
            role = discord.utils.get(guild.roles, name='COEng')
        elif payload.emoji.name == "3️⃣":
            role = discord.utils.get(guild.roles, name='CAES')
        elif payload.emoji.name == "4️⃣":
            role = discord.utils.get(guild.roles, name='COST')
        elif payload.emoji.name == "5️⃣":
            role = discord.utils.get(guild.roles, name='COEdu')
        elif payload.emoji.name == "6️⃣":
            role = discord.utils.get(guild.roles, name='CAHSS')
        elif payload.emoji.name == "7️⃣":
            role = discord.utils.get(guild.roles, name='COHHS')
        elif payload.emoji.name == "8️⃣":
            role = discord.utils.get(guild.roles, name='JSNN')
        else:
            role = discord.utils.get(guild.roles, name=payload.emoji.name)

        if role is not None:
            member = await guild.fetch_member(payload.user_id)
            if member is not None:
                await member.remove_roles(role, reason=None)
            else:
                print("Member not found")
        else:
            print("role not found")
    else:
        print('message not found')


    #
    # if message_id == j.id :
    #     guild_id = payload.guild_id
    #     guild = discord.utils.find(lambda g : g.id == guild_id, bot.guilds)
```



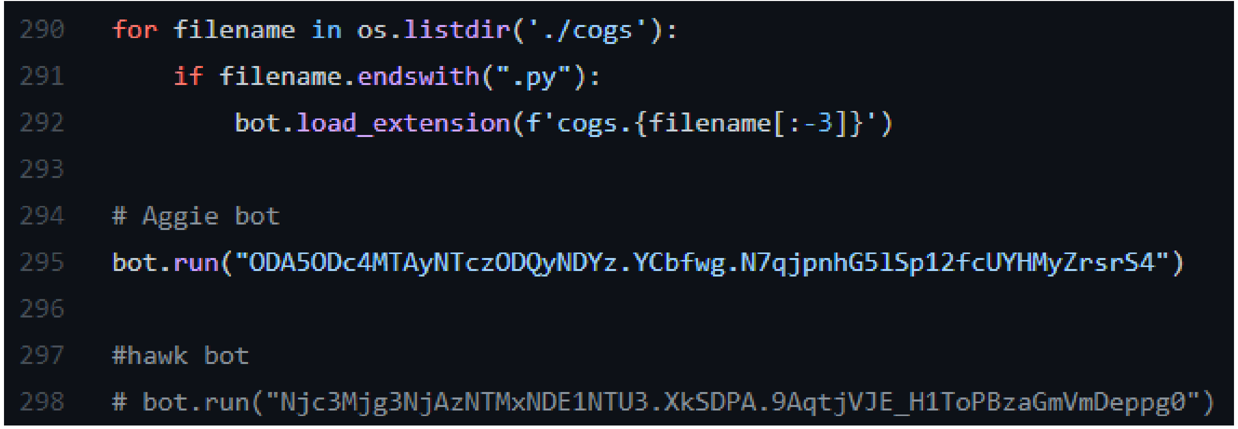
```python
    #     if payload.emoji.name == '🔒':
    #     #    channel = guild.get_channel(j.channel)
    #         channel = j.channel
    #         await channel.delete()
    #         print(f"{channel}")


@bot.command()
async def ping(ctx):
    embed = discord.Embed(color=0xdba834)
    embed.add_field(name="⏳", value="\u200b", inline=False)
    embed.add_field(name=f"{round(bot.latency * 1000)} ms", value="\u200b", inline=False)
    await ctx.send(embed=embed)

@bot.command()
async def create(ctx):
    emoji1 = '1️⃣'
    emoji2 = '2️⃣'
    emoji3 = '3️⃣'
    emoji4 = '4️⃣'
    emoji5 = '5️⃣'
    emoji6 = '6️⃣'
    emoji7 = '7️⃣'
    emoji8 = '8️⃣'
    await ctx.message.delete(delay=1)
    embed=discord.Embed(title="Choose Your Department", description="React with your deparment number!", color=0xdba834)
    embed.add_field(name="1. COBE", value="College of Business & Economics", inline=False)
    embed.add_field(name="2. COE", value="College of Enginerring", inline=False)
    embed.add_field(name="3. CAES", value="College of Agriculture & Environmental Science", inline=False)
    embed.add_field(name="4. COST", value="College of Science & Technology", inline=False)
    embed.add_field(name="5. COE", value="College of Education", inline=False)
    embed.add_field(name="6. CAHSS", value="College of Arts, Humanities, and Social Sciences", inline=False)
    embed.add_field(name="7. COHHS", value="College of Health & Human Sciences", inline=False)
    embed.add_field(name="8. JSNN", value="Joint School of Nanoscience & Nanoengineering", inline=False)
    msg = await ctx.send(embed=embed)
    await msg.add_reaction(emoji1)
    await msg.add_reaction(emoji2)
```



```python
    await msg.add_reaction(emoji3)
    await msg.add_reaction(emoji4)
    await msg.add_reaction(emoji5)
    await msg.add_reaction(emoji6)
    await msg.add_reaction(emoji7)
    await msg.add_reaction(emoji8)


@bot.command()
async def book(ctx):
    embed = discord.Embed(color=0xdba834)
    embed.add_field(name="NCAT Discord Rules", value="\n•No asking to be granted roles/moderator roles.\n•@mention the moderators for support.\n•Contact the moderators under #
    msg = await ctx.send(embed=embed)
    await ctx.message.delete(delay=2)

@bot.command()
async def rules(ctx):
    embed = discord.Embed(color=0xdba834)
    embed.add_field(name="Text Chat Rules", value="https://ncat.bncollege.com/shop/ncat/page/returns-refunds", inline=True)
    msg = await ctx.send(embed=embed)
    await ctx.message.delete(delay=2)



@bot.command(aliases=['8ball'])
async def _8ball(ctx, *, question):
    responses = [ "It is certain.",
                  "It is decidedly so.",
                  "Without a doubt.",
                  "Yes - definitely.",
                  "You may rely on it.",
                  "As I see it, yes.",
                  "Most likely.",
                  "Outlook good.",
                  "Yes.",
                  "Signs point to yes.",
                  "Reply hazy, try again.",
```

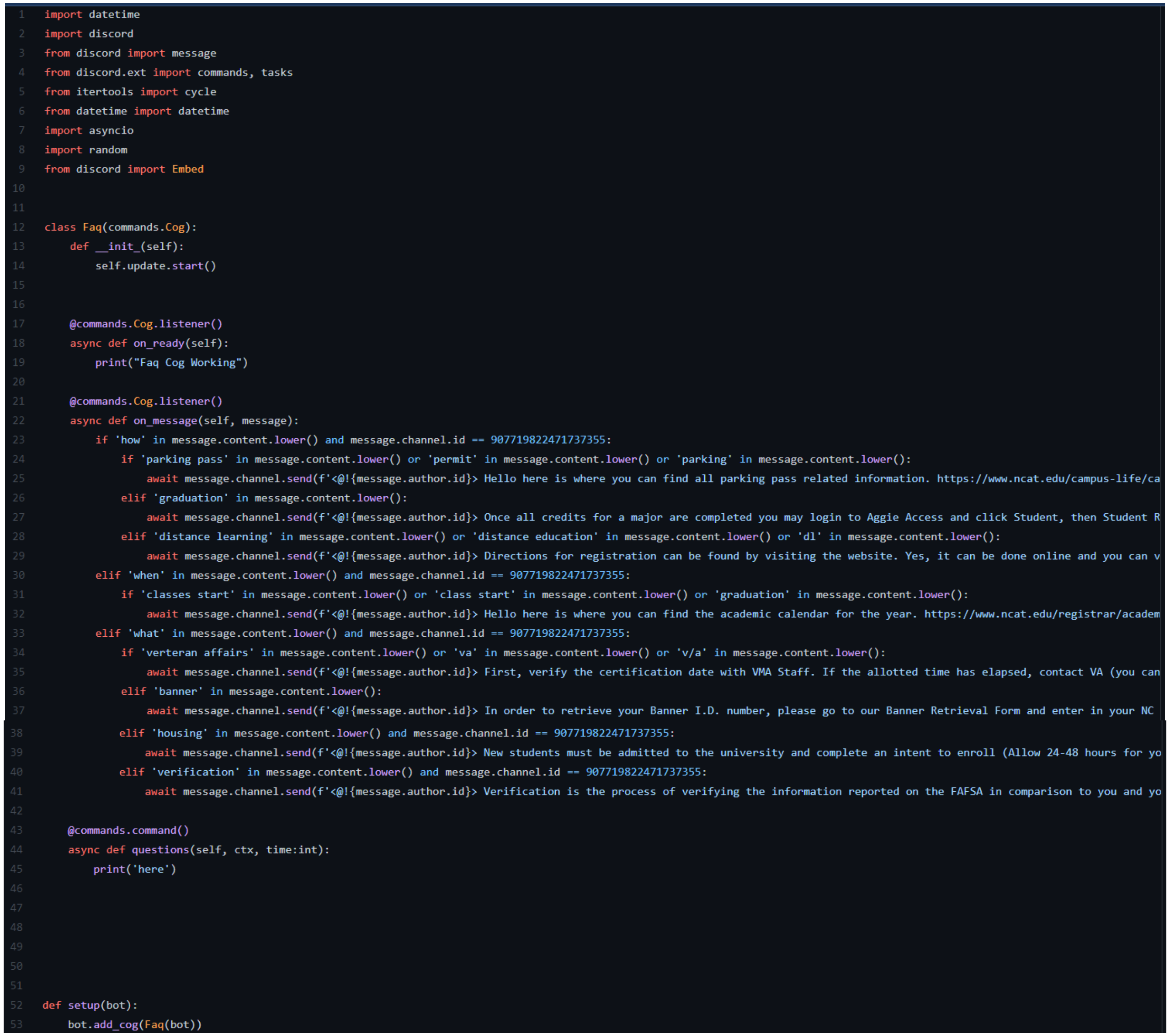


```
                  "Ask again later.",
                  "Better not tell you now.",
                  "Cannot predict now.",
                  "Concentrate and ask again.",
                  "Don't count on it.",
                  "My reply is no.",
                  "My sources say no.",
                  "Outlook not so good.",
                  "Very doubtful."]

    embed = discord.Embed(title="**Magic 8-Ball**", description=f"{question}", color=0x6a0dad)
    embed.add_field(name="Answer",value=f"{random.choice(responses)}", inline = True)
    await ctx.send(embed=embed)

@bot.command()
@commands.has_permissions(manage_messages=True)
async def clear(ctx, amount=5):
    await ctx.channel.purge(limit=amount)


@bot.command()
@commands.has_permissions(administrator=True)
async def updatecount(ctx):
    guild = ctx.message.guild
    vc = guild.voice_channels
    await vc[0].edit(name=f"Member count: {guild.member_count}")




@bot.command()
async def load(ctx, extension):
    bot.load_extension(f"cogs.{extension}")

@bot.command()
async def unload(ctx, extension):
    bot.unload_extension(f"cogs.{extension}")
```



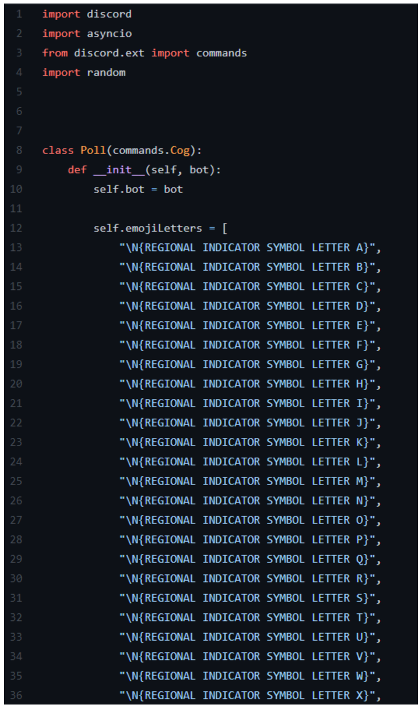
```
for filename in os.listdir('./cogs'):
    if filename.endswith(".py"):
        bot.load_extension(f'cogs.{filename[:-3]}')

# Aggie bot
bot.run("ODA5ODc4MTAyNTczODQyNDYz.YCbfwg.N7qjpnhG5lSp12fcUYHMyZrsrS4")

#hawk bot
# bot.run("Njc3Mjg3NjAzNTMxNDE1NTU3.XkSDPA.9AqtjVJE_H1ToPBzaGmVmDeppg0")
```

- Code for bot answering question in the FAQ channel

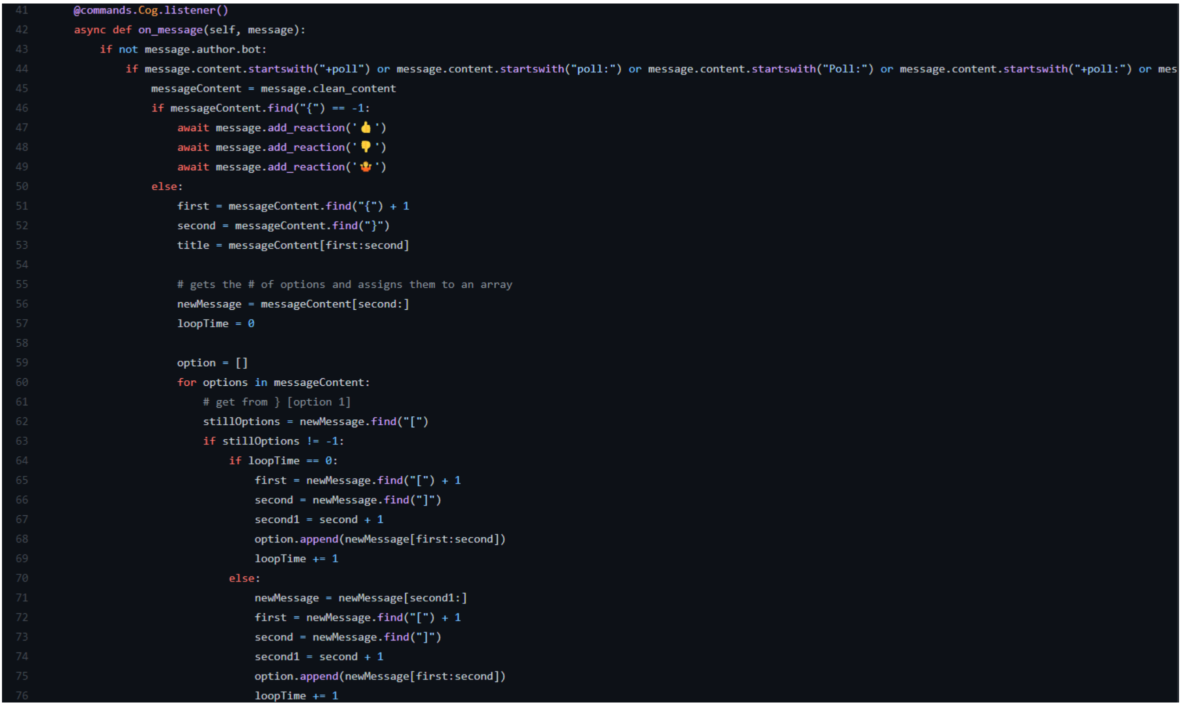


```
import datetime
import discord
from discord import message
from discord.ext import commands, tasks
from itertools import cycle
from datetime import datetime
import asyncio
import random
from discord import Embed


class Faq(commands.Cog):
    def __init_(self):
        self.update.start()


    @commands.Cog.listener()
    async def on_ready(self):
        print("Faq Cog Working")

    @commands.Cog.listener()
    async def on_message(self, message):
        if 'how' in message.content.lower() and message.channel.id == 907719822471737355:
            if 'parking pass' in message.content.lower() or 'permit' in message.content.lower() or 'parking' in message.content.lower():
                await message.channel.send(f'<@!{message.author.id}> Hello here is where you can find all parking pass related information. https://www.ncat.edu/campus-life/ca
            elif 'graduation' in message.content.lower():
                await message.channel.send(f'<@!{message.author.id}> Once all credits for a major are completed you may login to Aggie Access and click Student, then Student R
            elif 'distance learning' in message.content.lower() or 'distance education' in message.content.lower() or 'dl' in message.content.lower():
                await message.channel.send(f'<@!{message.author.id}> Directions for registration can be found by visiting the website. Yes, it can be done online and you can v
        elif 'when' in message.content.lower() and message.channel.id == 907719822471737355:
            if 'classes start' in message.content.lower() or 'class start' in message.content.lower() or 'graduation' in message.content.lower():
                await message.channel.send(f'<@!{message.author.id}> Hello here is where you can find the academic calendar for the year. https://www.ncat.edu/registrar/academ
        elif 'what' in message.content.lower() and message.channel.id == 907719822471737355:
            if 'verteran affairs' in message.content.lower() or 'va' in message.content.lower() or 'v/a' in message.content.lower():
                await message.channel.send(f'<@!{message.author.id}> First, verify the certification date with VMA Staff. If the allotted time has elapsed, contact VA (you can
            elif 'banner' in message.content.lower():
                await message.channel.send(f'<@!{message.author.id}> In order to retrieve your Banner I.D. number, please go to our Banner Retrieval Form and enter in your NC
            elif 'housing' in message.content.lower() and message.channel.id == 907719822471737355:
                await message.channel.send(f'<@!{message.author.id}> New students must be admitted to the university and complete an intent to enroll (Allow 24-48 hours for yo
            elif 'verification' in message.content.lower() and message.channel.id == 907719822471737355:
                await message.channel.send(f'<@!{message.author.id}> Verification is the process of verifying the information reported on the FAFSA in comparison to you and yo

    @commands.command()
    async def questions(self, ctx, time:int):
        print('here')






def setup(bot):
    bot.add_cog(Faq(bot))
```

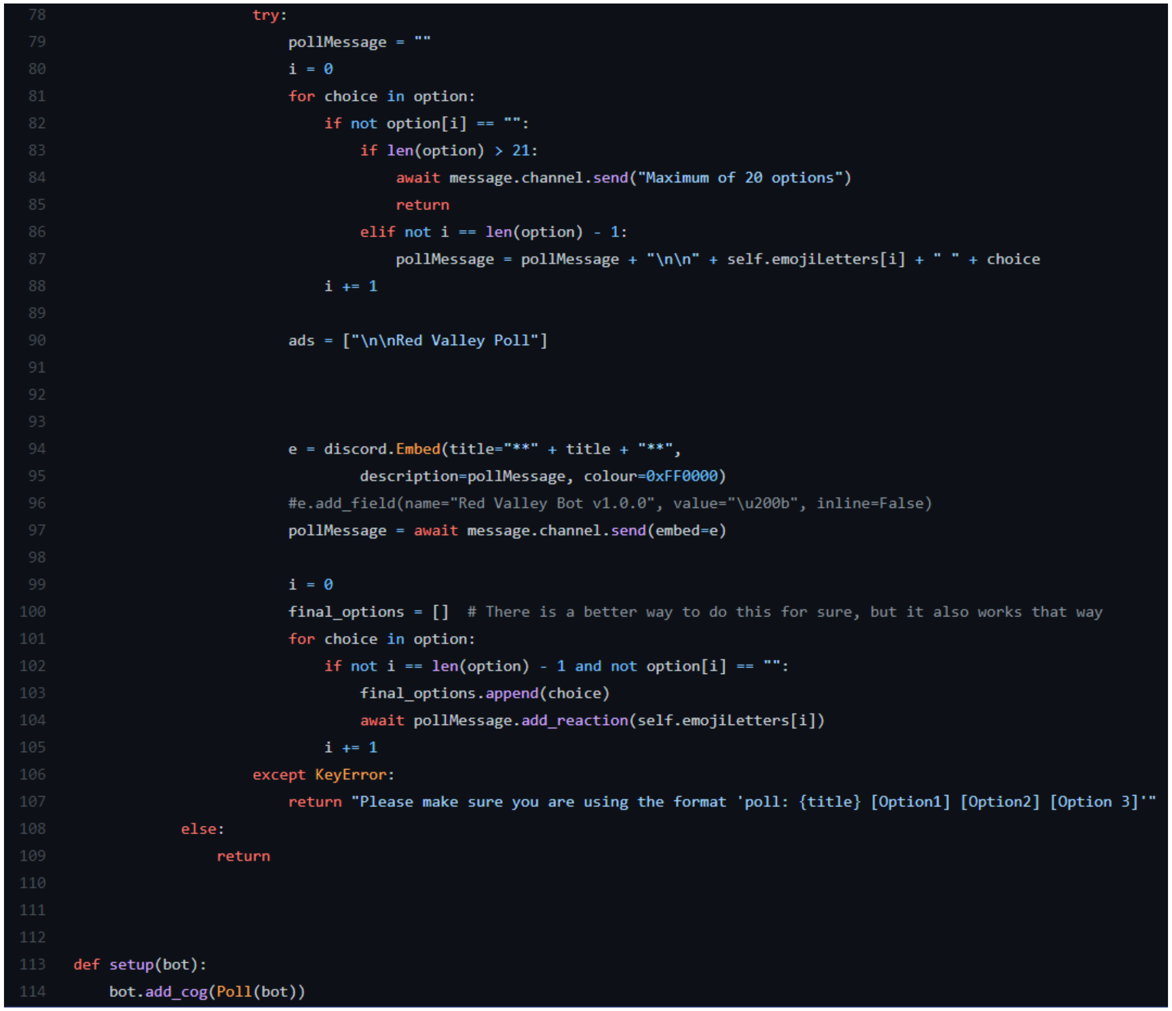
- Code for bot periodic poll questions



```
import discord
import asyncio
from discord.ext import commands
import random



class Poll(commands.Cog):
    def __init__(self, bot):
        self.bot = bot

        self.emojiLetters = [
            "\N{REGIONAL INDICATOR SYMBOL LETTER A}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER B}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER C}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER D}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER E}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER F}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER G}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER H}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER I}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER J}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER K}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER L}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER M}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER N}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER O}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER P}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER Q}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER R}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER S}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER T}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER U}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER V}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER W}",
            "\N{REGIONAL INDICATOR SYMBOL LETTER X}",
```

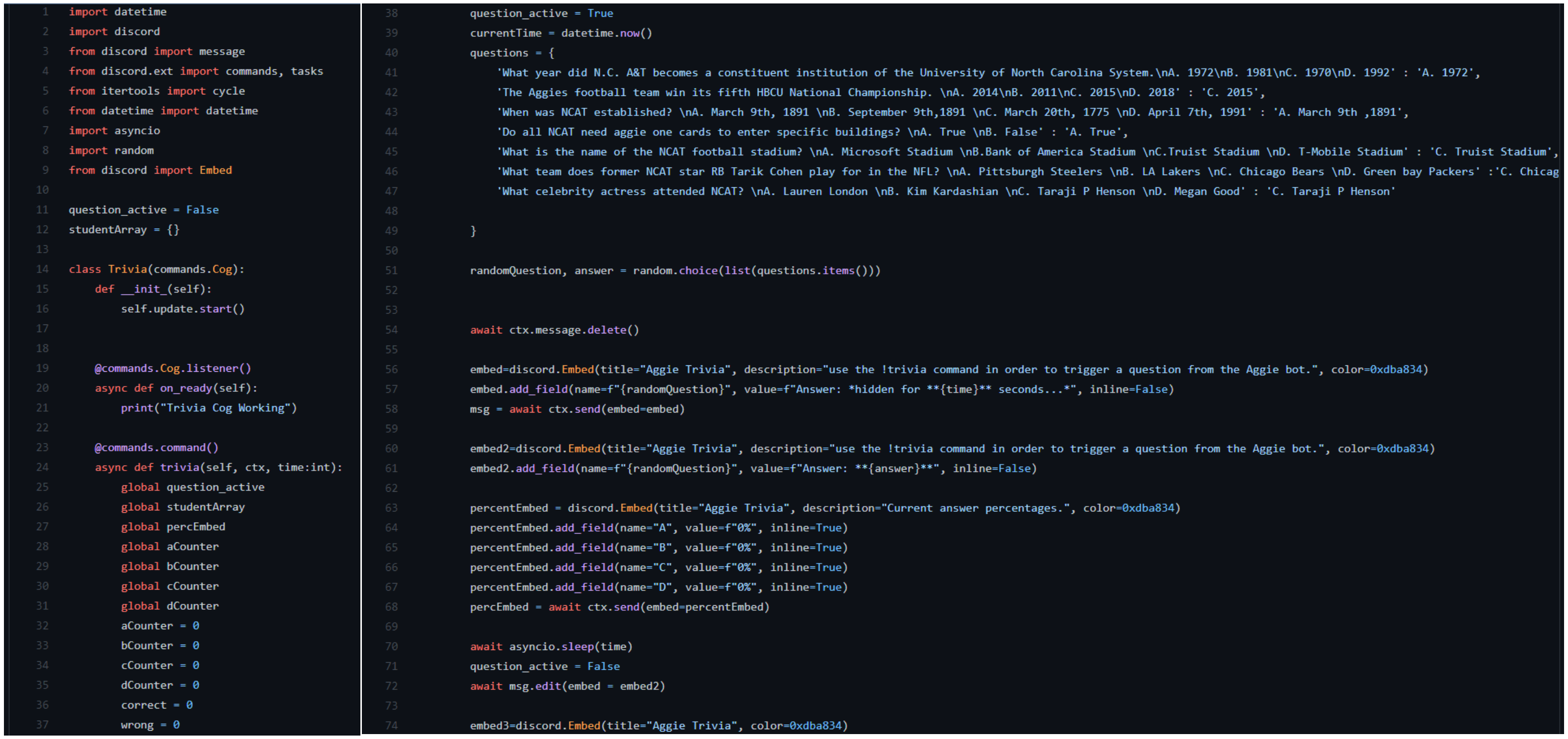


```
    @commands.Cog.listener()
    async def on_message(self, message):
        if not message.author.bot:
            if message.content.startswith("+poll") or message.content.startswith("poll:") or message.content.startswith("Poll:") or message.content.startswith("+poll:") or mes
                messageContent = message.clean_content
                if messageContent.find("{") == -1:
                    await message.add_reaction('👍')
                    await message.add_reaction('👎')
                    await message.add_reaction('🤷')
                else:
                    first = messageContent.find("{") + 1
                    second = messageContent.find("}")
                    title = messageContent[first:second]

                    # gets the # of options and assigns them to an array
                    newMessage = messageContent[second:]
                    loopTime = 0

                    option = []
                    for options in messageContent:
                        # get from } [option 1]
                        stillOptions = newMessage.find("[")
                        if stillOptions != -1:
                            if loopTime == 0:
                                first = newMessage.find("[") + 1
                                second = newMessage.find("]")
                                second1 = second + 1
                                option.append(newMessage[first:second])
                                loopTime += 1
                            else:
                                newMessage = newMessage[second1:]
                                first = newMessage.find("[") + 1
                                second = newMessage.find("]")
                                second1 = second + 1
                                option.append(newMessage[first:second])
                                loopTime += 1
```

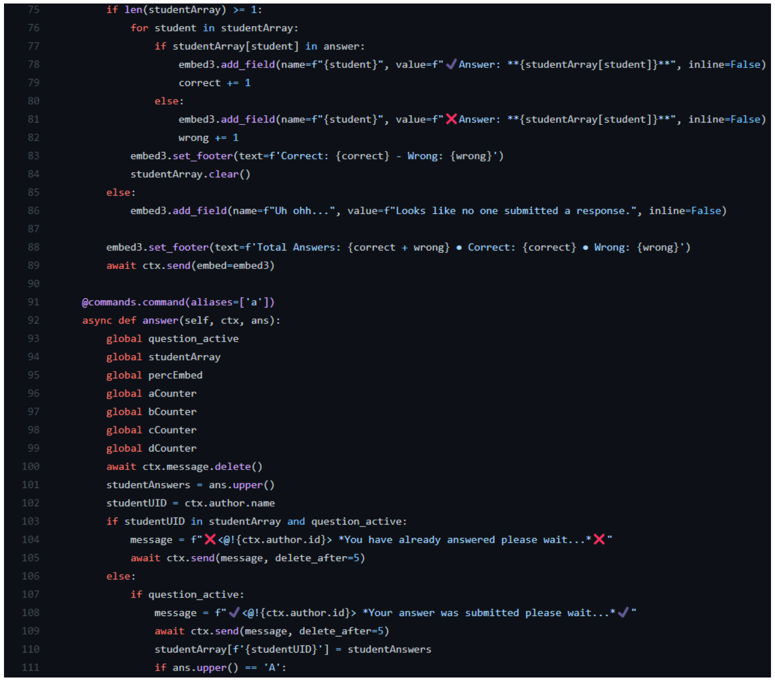


```
                    try:
                        pollMessage = ""
                        i = 0
                        for choice in option:
                            if not option[i] == "":
                                if len(option) > 21:
                                    await message.channel.send("Maximum of 20 options")
                                    return
                                elif not i == len(option) - 1:
                                    pollMessage = pollMessage + "\n\n" + self.emojiLetters[i] + " " + choice
                            i += 1

                        ads = ["\n\nRed Valley Poll"]



                        e = discord.Embed(title="**" + title + "**",
                                description=pollMessage, colour=0xFF0000)
                        #e.add_field(name="Red Valley Bot v1.0.0", value="\u200b", inline=False)
                        pollMessage = await message.channel.send(embed=e)

                        i = 0
                        final_options = []  # There is a better way to do this for sure, but it also works that way
                        for choice in option:
                            if not i == len(option) - 1 and not option[i] == "":
                                final_options.append(choice)
                                await pollMessage.add_reaction(self.emojiLetters[i])
                            i += 1
                    except KeyError:
                        return "Please make sure you are using the format 'poll: {title} [Option1] [Option2] [Option 3]'"
            else:
                return



def setup(bot):
    bot.add_cog(Poll(bot))
```

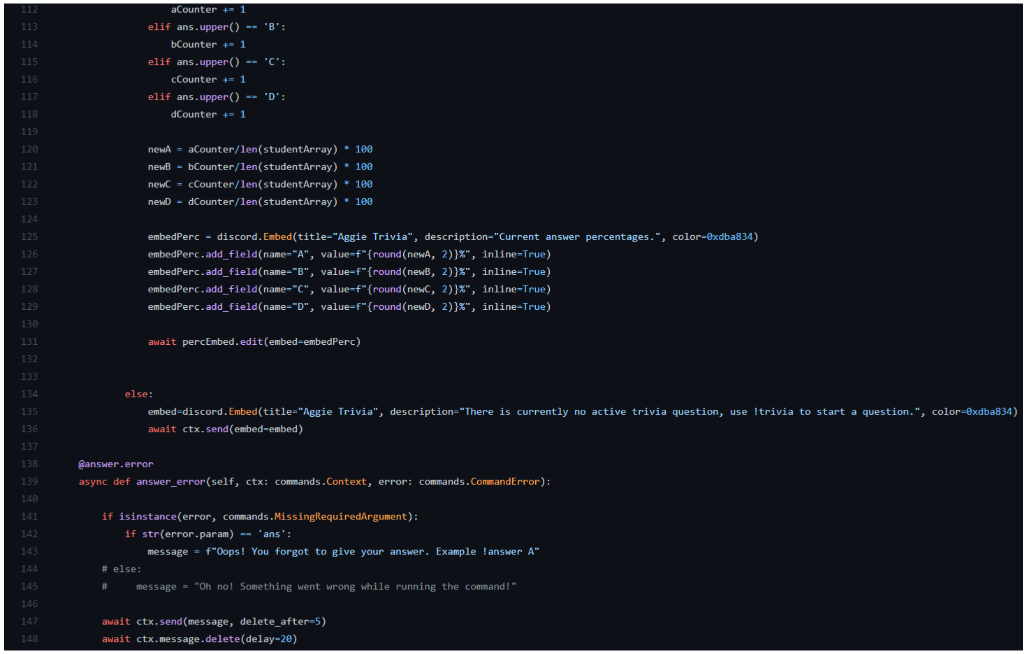
- Code for bot to ask trivia questions in the server



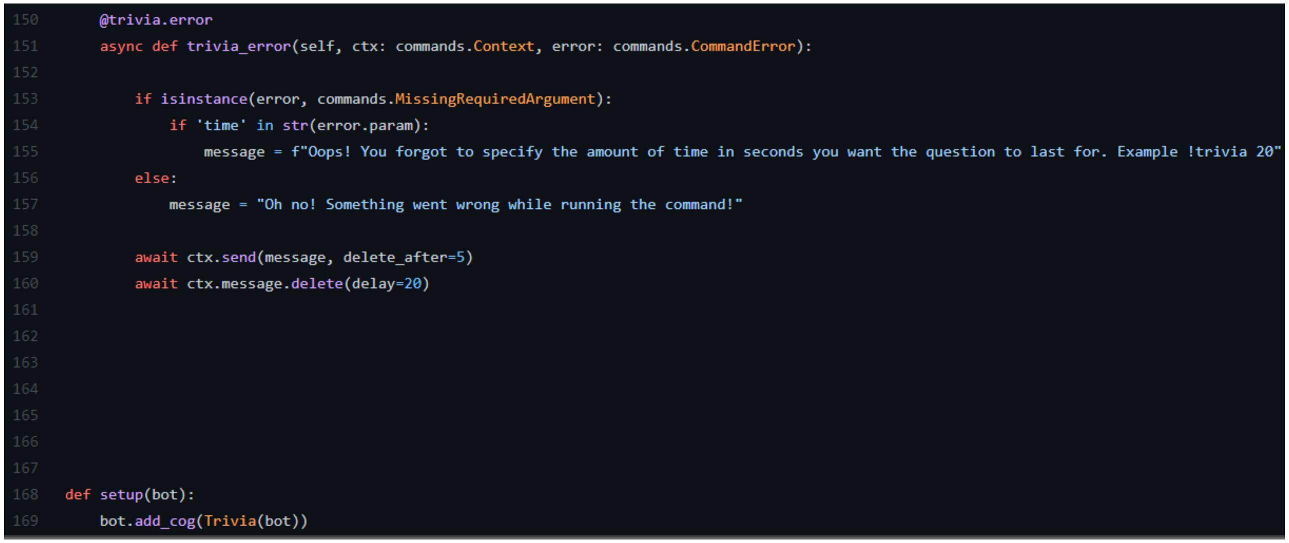
```python
import datetime
import discord
from discord import message
from discord.ext import commands, tasks
from itertools import cycle
from datetime import datetime
import asyncio
import random
from discord import Embed

question_active = False
studentArray = {}

class Trivia(commands.Cog):
    def __init_(self):
        self.update.start()


    @commands.Cog.listener()
    async def on_ready(self):
        print("Trivia Cog Working")

    @commands.command()
    async def trivia(self, ctx, time:int):
        global question_active
        global studentArray
        global percEmbed
        global aCounter
        global bCounter
        global cCounter
        global dCounter
        aCounter = 0
        bCounter = 0
        cCounter = 0
        dCounter = 0
        correct = 0
        wrong = 0
        question_active = True
        currentTime = datetime.now()
        questions = {
            'What year did N.C. A&T becomes a constituent institution of the University of North Carolina System.\nA. 1972\nB. 1981\nC. 1970\nD. 1992' : 'A. 1972',
            'The Aggies football team win its fifth HBCU National Championship. \nA. 2014\nB. 2011\nC. 2015\nD. 2018' : 'C. 2015',
            'When was NCAT established? \nA. March 9th, 1891 \nB. September 9th,1891 \nC. March 20th, 1775 \nD. April 7th, 1991' : 'A. March 9th ,1891',
            'Do all NCAT need aggie one cards to enter specific buildings? \nA. True \nB. False' : 'A. True',
            'What is the name of the NCAT football stadium? \nA. Microsoft Stadium \nB.Bank of America Stadium \nC.Truist Stadium \nD. T-Mobile Stadium' : 'C. Truist Stadium',
            'What team does former NCAT star RB Tarik Cohen play for in the NFL? \nA. Pittsburgh Steelers \nB. LA Lakers \nC. Chicago Bears \nD. Green bay Packers' :'C. Chicag
            'What celebrity actress attended NCAT? \nA. Lauren London \nB. Kim Kardashian \nC. Taraji P Henson \nD. Megan Good' : 'C. Taraji P Henson'

        }

        randomQuestion, answer = random.choice(list(questions.items()))


        await ctx.message.delete()

        embed=discord.Embed(title="Aggie Trivia", description="use the !trivia command in order to trigger a question from the Aggie bot.", color=0xdba834)
        embed.add_field(name=f"{randomQuestion}", value=f"Answer: *hidden for **{time}** seconds...*", inline=False)
        msg = await ctx.send(embed=embed)

        embed2=discord.Embed(title="Aggie Trivia", description="use the !trivia command in order to trigger a question from the Aggie bot.", color=0xdba834)
        embed2.add_field(name=f"{randomQuestion}", value=f"Answer: **{answer}**", inline=False)

        percentEmbed = discord.Embed(title="Aggie Trivia", description="Current answer percentages.", color=0xdba834)
        percentEmbed.add_field(name="A", value=f"0%", inline=True)
        percentEmbed.add_field(name="B", value=f"0%", inline=True)
        percentEmbed.add_field(name="C", value=f"0%", inline=True)
        percentEmbed.add_field(name="D", value=f"0%", inline=True)
        percEmbed = await ctx.send(embed=percentEmbed)

        await asyncio.sleep(time)
        question_active = False
        await msg.edit(embed = embed2)

        embed3=discord.Embed(title="Aggie Trivia", color=0xdba834)
```



```
        if len(studentArray) >= 1:
            for student in studentArray:
                if studentArray[student] in answer:
                    embed3.add_field(name=f"{student}", value=f"✔Answer: **{studentArray[student]}**", inline=False)
                    correct += 1
                else:
                    embed3.add_field(name=f"{student}", value=f"❌Answer: **{studentArray[student]}**", inline=False)
                    wrong += 1
            embed3.set_footer(text=f'Correct: {correct} - Wrong: {wrong}')
            studentArray.clear()
        else:
            embed3.add_field(name=f"Uh ohh...", value=f"Looks like no one submitted a response.", inline=False)

        embed3.set_footer(text=f'Total Answers: {correct + wrong} • Correct: {correct} • Wrong: {wrong}')
        await ctx.send(embed=embed3)

    @commands.command(aliases=['a'])
    async def answer(self, ctx, ans):
        global question_active
        global studentArray
        global percEmbed
        global aCounter
        global bCounter
        global cCounter
        global dCounter
        await ctx.message.delete()
        studentAnswers = ans.upper()
        studentUID = ctx.author.name
        if studentUID in studentArray and question_active:
            message = f"❌<@!{ctx.author.id}> *You have already answered please wait...*❌"
            await ctx.send(message, delete_after=5)
        else:
            if question_active:
                message = f"✔<@!{ctx.author.id}> *Your answer was submitted please wait...*✔"
                await ctx.send(message, delete_after=5)
                studentArray[f'{studentUID}'] = studentAnswers
                if ans.upper() == 'A':
```



```
                    aCounter += 1
                elif ans.upper() == 'B':
                    bCounter += 1
                elif ans.upper() == 'C':
                    cCounter += 1
                elif ans.upper() == 'D':
                    dCounter += 1

                newA = aCounter/len(studentArray) * 100
                newB = bCounter/len(studentArray) * 100
                newC = cCounter/len(studentArray) * 100
                newD = dCounter/len(studentArray) * 100

                embedPerc = discord.Embed(title="Aggie Trivia", description="Current answer percentages.", color=0xdba834)
                embedPerc.add_field(name="A", value=f"{round(newA, 2)}%", inline=True)
                embedPerc.add_field(name="B", value=f"{round(newB, 2)}%", inline=True)
                embedPerc.add_field(name="C", value=f"{round(newC, 2)}%", inline=True)
                embedPerc.add_field(name="D", value=f"{round(newD, 2)}%", inline=True)

                await percEmbed.edit(embed=embedPerc)


            else:
                embed=discord.Embed(title="Aggie Trivia", description="There is currently no active trivia question, use !trivia to start a question.", color=0xdba834)
                await ctx.send(embed=embed)

    @answer.error
    async def answer_error(self, ctx: commands.Context, error: commands.CommandError):

        if isinstance(error, commands.MissingRequiredArgument):
            if str(error.param) == 'ans':
                message = f"Oops! You forgot to give your answer. Example !answer A"
        # else:
        #     message = "Oh no! Something went wrong while running the command!"

        await ctx.send(message, delete_after=5)
        await ctx.message.delete(delay=20)
```



```python
    @trivia.error
    async def trivia_error(self, ctx: commands.Context, error: commands.CommandError):

        if isinstance(error, commands.MissingRequiredArgument):
            if 'time' in str(error.param):
                message = f"Oops! You forgot to specify the amount of time in seconds you want the question to last for. Example !trivia 20"
        else:
            message = "Oh no! Something went wrong while running the command!"

        await ctx.send(message, delete_after=5)
        await ctx.message.delete(delay=20)








def setup(bot):
    bot.add_cog(Trivia(bot))
```

7. Learning Statements

Team Member 1: Ryan Blount

- It has been a great experience to work will each team member while creating the Discord Server. I learned how to create emojis, server banners, and other unique features on a discord server. Through research and communication with my team members, I learned how Discord utilizes bot AI technology to perform useful tasks for the different channels and categories we created on the server. I also learned how users can be giving administrative roles so that they can facilitate and manage the different permissions in each channel and category.

Team Member 2: Sage-Michael Brown

- I have been able to work with three of the most amazing people, while working we have had talks on projects, computers, gaming, and just normal life. The most important thing I have probably learned while working with everyone is about teamwork and developing my knowledge of databases. This project so far has instilled in me security while working and the value of everyone's input making everything come together to produce a final product.

Team Member 3: Hau Le

- There are so many things I learned while working with my teammates. First of all, I have learned that group projects can be both rewarding and frustrating in many ways such as we have to deal with the deadline every week and a lot of work under the condition going on right now. Our group allows people with different viewpoints and knowledge levels to learn from each other. We also create a space for people to have fun, helping each other solve problems. I really appreciate my teammates helped me a lot in this project because I'm an international student and I'm currently taking a Database class. Everything feels so different to me, and I think I don't have enough skill to work on it much, but our group showed me how to use and work with Discord Server. Although this is my first capstone project, I have learned a lot from the people around me. This gives me the opportunity to learn from them as well as their culture and hobby in life.

Team Member 4: Terrell Pinkney

- Starting the project from the initial brainstorming phase was interesting and was a great experience. I learned about the many features that Discord provides from its users through my team members and research. Every team member collaborated very well with each other in regard to complete each task for the server. So far, we learned how we can link the student database to the python code for the Aggie Bot. Through research, we found multiple applications that allowed users to upload excel, Access, and SQL databases onto an online server. We decided to use a service called PostgreSQL that utilizes Amazon's cloud service AWS. It was great that we were able to find a service that will allow our bot to make use of the database we created to test the functionality of our server.