Part A  
Description  
Project: Rainbow Six Siege Team Stats Tracker

The Rainbow Six Siege Team Stats Tracker is a dedicated database application that I have designed to precisely manage and analyze the performance statistics of my current competitive Rainbow Six Siege team. Rainbow six siege is a Ubisoft Game in which two teams of five players choose from a variety of specific operators, or agents, with specific utility to either protect or defend specific objectives. With the growing complexity and immense strategic depth of the game, players and their teams really require in depth tools to monitor players progress, success, failures, and find areas of growth. Doing this regularly can give a team a competitive edge, which is extremely important in this environment. While this can be done by hand, and with VOD reviewing, it is never as accurate as being able to look at specific statistical values. This application aims to collect detailed player statistics directly from Ubisoft's official platform through a custom API, allowing for real-time performance tracking and historical data analysis.

The core functionality of my Tracker revolves around the three principal entities of the game: Players, Maps, and Operators. See these three entities are very important for checking players strengths and weaknesses. First we can check on a players overall statistics, which allows us to see where they tend to lack, overall win loss, their K/D ratio, or even their KOST which is a measure of their support in game. We can get similar stats per operator and map to see if specific maps or operators need work for a player, or if they thrive on a certain operator and map.

After getting all this data, each player's performance data, which consists of their respective in game operators and played maps, is added to our database. The relationships between the entities I chose are designed to help users get insights into other players such as their operator efficiency or how adaptable they are across different maps and roles.

The Team Stats Tracker in its infancy stands to be a super useful tool for any serious rainbow six siege teams. By leveraging the use of accurate and up to date player data, I can basically allow teams to strategically analyze their whole team and make decisions that can really be the difference between victory and defeat in the competitive arena that is Rainbow Six Siege.

# Part B E/R Diagram

A diagram of a game

Description automatically generated

# Part C

Relational Schema

A screenshot of a computer program

Description automatically generated

BCNF form?  
yes this is in BCNF Form since there are no transitive dependencies and no partial dependencies.

Part D  
I basically am pulling the Data using an API from Ubisofts servers, I then store using SQLite and python the data I’ve pulled into the appropriate tables.

# Part E

When I was coming up with a user interface, I was originally creating a Website. However when I was testing the use of a website with my team, it wasn’t really as easy to have everyone join a meeting and have to watch my screen or individually all go to the website to the same stats page I was on. Instead I decided to create a Discord Bot so that my teams discord server could easily discuss the stats in meeting. Everyone can see the same message sent by my bot in real time, and so we can easily compare and go over team data.   
To Use my bot, first make a discord server and then invite my bot using the link:  
<https://discord.com/oauth2/authorize?client_id=1204233624925511691&permissions=8&scope=bot>

My bot has a help function /R6help which shows you all the commands at any time, but you can also choose

/refresh` - Update the database with the latest data.

/showtopbyrank` - Show top 5 players by ranked points.

/showtopbykd` - Show top 5 players by K/D ratio.

/showtopbykost` - Show top 5 players by KOST.

/playerstats <username>` - Show stats for a specific player.

/playermaps <username>` - Show map data for a specific player.

/playerops <username>` - Show operator data for a specific player.

# Part F (Source Code included)

Part G  
Honestly this was a lot of work. It came during a bad part of my life too, I was in between flying to countries because my grandmother was passing away and then I ended up staying out of country till my grandmother passed a few days ago. Every bump in the road this project had was like a nightmare to me, and it really took a lot out of me to finally get everything working. That being said I know there are a lot of features I still think are needed to make this bot a super great bot, and it needs some cleaning. I think overall this project really taught me a lot of perseverance, and that even when something isn’t working, and when nothing is going right, that there is always a way to keep moving forward.

## Credits to CNDRD for creating and maintaining the API <https://github.com/CNDRD/siegeapi>