**My Artifacts**

### Artifact 1 – Mobile Application to Full Stack MERN Web Application

I chose this assignment to upgrade my CS360 Mobile Design and Architecture assignment augment because it fits two of the milestone goals Software Design and Engineering and Databases. The Original work was a mobile java application therefore converting it into a MERN stack web application both align with my full-stack career goals as well as fit the criteria. Build a MERN stack requires implementation of a database therefore it also assisted in hitting the criteria for goal number three Database.

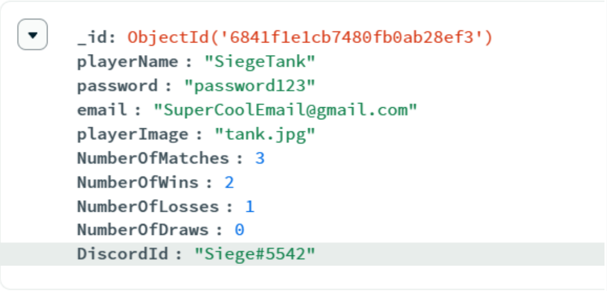
This application goal was to create a matchmaking queue for real world games, such as magic the gathering or Warhammer 40k in your favorite local stores. You'll be able to log into one of two example accounts.

SiegeTank or BigTanker, using the emails of SuperCoolEmail@gmail.com or SuperCoolEmailAsWell@gmail.com and both use password123. Other username/password will be rejected.

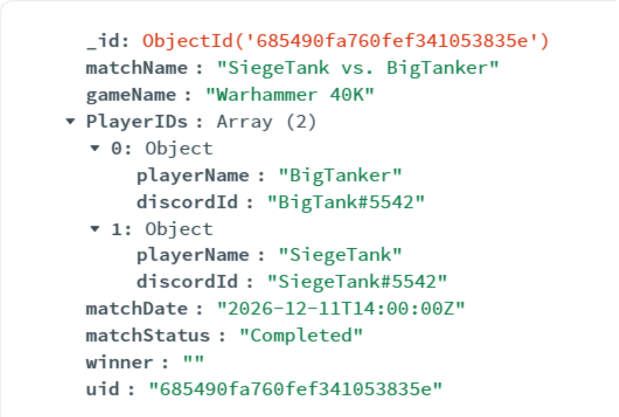
Hit the "Find Match button, If your account is currently not in queue you will be added to and then removed from queue upon being matched. To test this, you can either log into both accounts or queue up with yourself, I've left this up intentionally as to allow for smoother demo testing. You will be given the name of your opponent and their discord_Id which you would use to contact them and schedule a time. This also allows you to implement a delete match to remove a match you are no longer participating in and update the time of your match after you've coordinated your match time.

This is persistent if you create a match between the two you'll load in and see the account match has rendered. If you only create a match against yourself (SiegeTank vs SiegeTank) you will not see it when you are logged in as big Tanker.

This program utilizes a MongoDB database free account and references an offsite database from the example servers to show how the program would develop in the future. I've included two brief snapshots of how the database is laid out but more can be seen in the milestone 2 narrative if you wish to look into the final version there.



```
_id: ObjectId('6841f1e1cb7480fb0ab28ef3')
playerName : "SiegeTank"
password : "password123"
email : "SuperCoolEmail@gmail.com"
playerImage : "tank.jpg"
NumberOfMatches : 3
NumberOfWins : 2
NumberOfLosses : 1
NumberOfDraws : 0
DiscordId : "Siege#5542"
```



```
  _id: ObjectId('685490fa760fef341053835e')
  matchName : "SiegeTank vs. BigTanker"
  gameName : "Warhammer 40K"
▾ PlayerIDs : Array (2)
  ▾ 0: Object
      playerName : "BigTanker"
      discordId : "BigTank#5542"
  ▾ 1: Object
      playerName : "SiegeTank"
      discordId : "SiegeTank#5542"
  matchDate : "2026-12-11T14:00:00Z"
  matchStatus : "Completed"
  winner : ""
  uid : "685490fa760fef341053835e"
```

### Artifact 2- Converting BST to a Balanced AVL Binary Search Tree.

I chose an assignment from CS300 Data Structure and Algorithms Analysis and Design, of converting a Binary Search tree that parsed a word document of courses and placed them into a BST. I improved on this for the Data Structures in algorithms in two ways, first I added an additional features that allowed the user to choose from a list of degree plans and see how many courses they still had to take in order to finish their degree.
I also implemented the project from an unbalanced BST into an AVL and a balanced search tree which improve search time from a O(n) while unbalanced to a balanced one of O(log n).

### Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science

I have created a code review that details how I intended to grow a mobile application into a larger full stack web application. I utilized this to emphasis an understanding of the need to constantly review and improve upon each project. During my development process I stagnated I would utilize industry practices where in short of having a compatriot coder to work with I my father an former mainframe programmer who would bounce ideas back at me for review. This along furiously commenting my code so that it can be understood and read in at a future date.

### Design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts

I included a list of detailed comments I my code that allow for future develops to understand why and what occurred so that they can return to the files and make changes and improve upon the existing code base. I adhered to standard UI/UX design practices by implementing, UX/UI of consistency, usability and clarity in each design I intention repeat patterns that build familiarity in a user while also focusing on clean Ui's that are easy to understand.

**Design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution, while managing the trade-offs involved in design choices**

I improved upon a CS300 Binary Tree Assignment, Data Structures Analysis and Design Course, I improved upon it by adding an additional feature to compare it with a list of courses as well as improved upon the search speed by converting a normal BST in a Adelson-Volsky and Landis self-balancing binary tree or an AVL tree. A standard binary tree suffers from the potential to have the data skewed to one side of the search tree over time. This will create longer search times for the majority of items in the search tree. By balancing the search tree, I created decreased the time the average search will take. While this disadvantages are that it increases insertion time for new items, a course list database would receive more query's during a term than it would updates, so I deemed it to be valuable improvement.

**Demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals**

I converted a Mobile application from CS360 to full-stack web application is a MERN stack, which is a popular framework stack for large companies such as Netflix, UberEATS, Airbnb, Instagram and Walmart. It focuses of single page applications that will dynamically write and update the webpage as the user makes server requests. I chose this for my final project as it included several industry standards frameworks used throughout the industry using this stack align with my future full-stack development goals as well the ability to set production schedules that lead to actualized production.

**Develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources**

During Full MERN stack application I implemented a log-in feature to my assignment. Log-in are one of the core principles of AAA software security, that show an understanding that you sue always authentic your users. While this program isn't ready for a need for security, I felt including the use of the first steps dictated that in the future I would plan for a Authenticated, Authorized and Accountable design program.