

Milestone 3: Features and Functionality

No changes to initial proposal

Milestone 3 Demo Link:

<https://duke.box.com/s/wuxnr3hqu1n7zs9cof2umpuv92f88hj6>

GitHub Link: <https://github.com/2Ahmad5/MonPoke-Champions/tree/main>

Prototype Code Repo: <https://gitlab.oit.duke.edu/ndp25/cs-316-proj>

****GREEN** = completed components by MS 3

****BLUE** = completed components by MS 4

- (20 points) Users Guru: responsible for Account / User Profile / Database design- **Ahbab**
- (20 points) MonPoke Guru: responsible for MonPoke Profile / Design / Database design - **Ahmad**
- (20 points) Purchase Guru: responsible for Microtransaction Store / Ordering / Purchase History Database design - **Nolan Potter**
- (20 points) Social Guru: responsible for Feedback / Messaging / Leaderboard - **SJ**
- (20 points) Game: responsible for game/SQL database interactions/ making the game/ - **Richard**

Functionalities

Quick Overview:

This will be **web-based**, and the actual game (**Unity-based**) will run on a container on the home page of the website. Different **Users** can request to **battle** other users using their **Monpoke**. There will be other tabs on the website corresponding to MonDex, UserProfile. And Leaderboards.

A more detailed overview for the project is linked

<https://docs.google.com/document/d/1Nywsx4oWKuPQ3is1r6MegGpzO3j6jZZRKoMDVjI487I/edit>

User Login / User Profile

Basic requirements:

- A new user can register for a new account; an existing user can log in using username and password.
- Each user account has a system-assigned id. Other account information includes email, full name of user, and password. Users can update all information except the id. Ensure that username is unique among all users.
- Each account is associated with a MonPoke currency. It starts out as \$0, but can be topped up by the user (either through playing the game or buying it using real money).
- Provide a way for user to view their profile. It will show the account number, MonPoke currency, name as well as collected MonPoke and match history.
- Implement a way for users to view OTHER people's profile through a search functionality. Users can search based on other users' username.

Possible additional features:

- Visualize history of balances, and purchase history for the user. This will obviously be hidden when others view your profile.

MonPoke Design / Profile

Basic requirements:

- A separate table just storing all the different characters, their types, attacks, and other similar attributes.
- Users can go on a separate tab to view a MonDex where all the MonPoke's are listed to view relevant information. Options will exist on the tab to filter to make it easier to view MonPokes. Example would be only wanting to view fire types, etc.
- All the MonPokes will be listed as types of species and within a Users data, the MonPokes they have will be stored.
- A variety of MonPoke will reside in this database

Possible additional features:

- Connecting to the MonPoke purchase database to recommend MonPoke the user should buy depending on what they likely need

Microtransaction Store Management

Basic requirements:

- Each user has access to the Microtransaction Store. Items in the store will be packs that give the user a random card, drawn from the database of cards, with higher rarity cards being less likely to be drawn. Different packs will be able to give different types of cards. Each item in the store will have a designated price. All items in the store will be available to the user.
- Users can use an in-game currency to buy one or multiple items provided they have the balance. After completing an order, their balance will be decreased and they will open the pack and the cards will be added to their inventories.
- Users can use “real” money to buy additional currency. It will require the user to add their payment info and then the in-game currency can be added to their balance. For the purpose of the project, we will not deal with real credit cards. - can add currency with a basic form submission at the moment
- Users' purchase history will be tracked and can be viewed in the store interface and will include the date of purchase and purchase cost.
- Users can have a cart that will allow them to mass purchase different items. The cart will remain through log-outs.
- When a user purchases a card, the card they drew pops up with the color of the rarity.

Possible additional features:

- Add a favorites tab where users can select/star items to be designated as favorites which will show up first when they enter the store interface.
- Implement limited time events that can add new discounted bonus offers or limited time packs in the store.

Game

Basic requirements:

- Create a functioning multiplayer card game in Unity that can be run on a website with html.
- Main Menu: The main menu will include buttons for the matchmaking, and settings functions. The matchmaking function will have a text box and a “send invite” button. One player would type in the name of another player and press send invite. The other player would have a box pop up on their main menu notifying them that someone has requested to play against them, to which they can accept or decline. The store will contain packs that players can open with either in-game currency or real money. There will be different types of packs that cost and give different things. The Inventory will allow players to decide what cards they want to include in their 20 card deck. The settings button will include simple setting options.
- Gameplay: The game will practically be Hearthstone retextured. Each player will have a random 5 cards from their 20 card deck given to them at the start of the game. Each player gains 1 mana at the start of their turn until they reach a total of 10. During a turn, players can use minion or spell cards, which may cost them mana, or attack with an already placed minion. Minion cards will each have an attack and health value, when the health value is depleted they are removed from the board. Minions can either attack other minions or the other player. Players start with 20 health and lose when all of their health is depleted.

Considerations:

- Updating SQL databases through Unity Engine / does what happen in the game update the SQL databases

Additional Features:

- Animations
- Sounds and music

Feedback / Messaging

Basic requirements:

- A user can submit a bug report using a form on the website
- A user view their spot on the leaderboard
 - This leaderboard will be generated automatically by periodically sorting all user profiles based on #win (or some other metric we decide later)
 - This leaderboard will be a tab on the website, displaying users, their rank, and their (score)
- There will be a forum that is accessible on the website where users can comment and rate Monpokes

Possible additional features:

- FAQ page
- Upvote functionality for rating/reviews of cards to allow certain reviews to be marked as more or less helpful. By default the top 3 most helpful reviews would be shown first, and then the most recent following these.

c and d. Components that have been completed highlighted in Green. Non highlighted parts to be completed.