

## CS 411 Track 1 Project Proposal

### Project Title:

**POKÉMON BOSS RUSH**

### Project Summary:

*A Pokémon boss rush where you battle custom gym leaders!!*

Pokémon Boss Rush is a strategy-first, database-driven single player role-playing game designed for the web. Our app removes the time-consuming exploration of traditional Pokémon games and focuses on the experience of team building and strategic battles. Players will create a user profile, access a Pokédex to build a team of six Pokémon, and customize their movesets to prepare them for battle. The core feature of the game is a **boss rush**. A boss rush is a challenging rogue-like where players must defeat a series of four unique gym leaders in a single run.

This project provides a curated experience that fills a very specific niche for Pokémon fans. It allows for strategic experimentation and roleplaying without the high-stakes pressure of competitive player-vs-player battle simulators and the need for hours of in-game exploration of the main game. The Gym Leaders introduce a creative aspect of this app, where each has a distinct personality, appearance and battle AI modeled after a member of our development team.

### Description:

#### **What problem do we want to solve and how is this unique or novel from other similar games?**

The current landscape of Pokémon web apps is largely split between two main categories: encyclopedia-like databases ([Serebii.net](https://www.serebii.net), dataDex) and competitive multiplayer battle simulators (Pokémon Showdown) which are geared towards PvP players. This leaves a significant gap for the vast majority of Pokémon fans who enjoy

the single-player rpg journey and may not be able to dedicate 40+ hours to invest in the original games and/or crafting their perfect Pokémon teams without the need to **catch 'em all**.

Our app aims to solve this by providing a narrow, focused, replayable and enjoyable strategic single-player experience. We are eliminating the “grind” of leveling up and exploring, allowing players to jump right into the most engaging part of the game: team building and battling. Our core features provide the player with the ability to test any team composition in the Pokédex against a series of well-designed, custom-built, intelligent AI opponents. This provides a satisfying experience and gameplay loop that serves as entertainment and a tool for players wanting to improve their strategic thinking ability.

### **Creative Component:**

#### **Dynamic & Personalized Gym Leader AI**

Our project’s primary creative and technical feature is the implementation of four unique gym leaders whose battle logic and teams are modeled after the personalities and strategies of our development team. This feature is technically challenging because it requires us to move beyond simple randomization in Pokémon and move selection and implement a pre-scripted, decision-based engineer for our AI opponents.

For example, one member of the dev team might prefer an offensive glass cannon strategy and their corresponding AI will be set to prioritize special attackers and high-damage moves. Another member of the dev team might prefer Pokémon that inflict status effects, like “toxic” and will be scripted to do so in battle. This feature requires well-thought out backend logic where each AI profile queries the battle state and makes a move choice based on its own unique set of pre-programmed priorities. This creates a personalized and strategic experience for the player, making each gym battle feel more similar to the original Pokémon games.

## Usefulness:

Pokémon Badge Quest/Battle Rush is a fun app for casual gamers and dedicated Pokémon enthusiasts because it serves as both a source of entertainment and a tool for practicing strategy. The basic functions of the app allow users to engage with a Pokédex database. A user can:

- create a profile
- log in to an existing account
- browse a complete Pokédex
- search/filter for Pokémon by name, type or stats

From there, they can perform full CRUD (create, read, update delete) operations on their own battle teams, selecting a team of six Pokémon and customizing their movesets. This team building aspect provides a sandbox for players to experiment with combinations of Pokémon teams that they might not have time to try in the original games.

The app's gameplay loop will also distinguish it from other existing Pokémon apps on the web. Our badge quest/boss rush mode is a persistent challenge **where the player's team carries over from one battle to the next**. We will also implement a user statistics page that tracks metrics on the user profile such as win/loss, badges earned, and possibly even timed battle runs as a reach goal, which may add another layer of personal progression for the player. Our project is unlike Pokémon encyclopedias (Serebii) because we offer an interactive battle experience. Our project is also unlike PvP battle simulators like Pokémon Showdown because our focus is on curating a pressure-free single-player campaign against a custom built unique AI opponent, making it more accessible for casual players and less intimidating for broader audiences. Our potential reach goals include implementing a true roguelike mode with increased difficulty for each battle run, which while challenging, will enhance the players experience and replayability.

## Realness:

### Data Sources:

Our app will be built on two primary, real-world data sources from Kaggle, which will form the contents of our Pokédex and team-builder.

### Pokemon Stats and Information:

Source: <https://www.kaggle.com/datasets/rounakbanik/Pokémon>

Format: CSV

Data Size: Contains information for approximately 800 Pokemon

Schema: The dataset includes columns of information for each Pokemon across 8 generations including: Pokédex\_number, name, type, attack, defense, sp\_attack, sp\_defense, speed, abilities, generation and is\_legendary

### Pokemon Images:

Source: <https://www.kaggle.com/datasets/kvpratama/Pokémon-images-dataset>

Format: Images (png)

Data Size: Contains approximately 809 images

Schema: The dataset contains standardized images for each Pokemon across 8 generations with their names.

Combining Datasets: we will join this dataset with the primary stats dataset on the Pokemon's name to display a visual representation of the Pokemon's form from the original games in the app's Pokédex, the user's team and during battle runs.

## Database Design

Our app's functionality will be centered around database operations and CRUD actions for user-generated behavior:

- A new user **creates** a user profile, which updates our Users table with a new data entry.
- When building a team, the user **creates** a new Team entry, which is linked to their **user\_id**.

- This team contains six **pokemon\_instance(s)**, which are each linked to a specific Pokemon in the Pokédex table.
- Users can **read** their profile information and **save** their team distribution.
- Users can **update** their team by swapping Pokemon or changing their team's movesets
- Users can **delete** whole teams or specific Pokemon from their team
- During battle runs, the app will **query for and read** Pokemon data to calculate important aspects of battles like damage and will **update** the state of the Pokemon (like HP) so that the Pokemon's status will persist between battles.

### Functionality:

Our website delivers multiple UI views, such as a login/create an account view, a profile view, a Pokédex and team-building view, and a gym leader lobby view. Once a user creates an account, we are able to store all game-related information with that user's username, email, and password in the database. After they login, the user is brought to their profile page that allows them to navigate to other pages, like the ones mentioned above.

Upon arrival to the homepage, the user is met with a navigation bar on the left, and a profile-specific navigation bar on the right. In the navigation bar on the left, the user can navigate to view the teams they've created and create, update, and delete teams. This is important because the purpose of our app is for the user to fight "gym leaders" that own a gym, and they must create a team of Pokémon to defeat them. Additionally, they can view the gyms they've completed, the entire Pokémon collection (Pokédex), and the badges they've accrued throughout the game. The pages that display the gyms the users have completed and the badges they've earned serve as a memory book to remind the users of their accomplishments. The Pokédex page serves as a resource for the users to learn more about what attributes different Pokémon have, which allows users to make informed decisions when creating their team.

Along with the left navigation bar, we have a right profile-specific navigation bar that allows users to log out of their account and view information specific to their

account. The information specific to the user's account includes their username, email, password, badge level (novice, intermediate, expert), win/loss rate, average battle time, and badges earned. On this same page, we also include the option to delete the user's account.

Between the left navigation bar and the right profile-specific navigation bar, we have our main area which displays the user, the current gym they need to defeat, as well as a button to allow the user to enter the gym battle. Once the user enters the gym, they are met with a description of the gym from the gym leader, and have the ability to choose the team of their choice to fight against the gym leader. During the battle, users can choose which move to use in the Pokémon's move set. If the battle results in a win, users are able to navigate to the next battle or home page, and if the battle results in a loss, users are able to retry or navigate to the home page.

The information used to display the Pokémon in the Pokédex would be taken from the datasheets above, and the gym leaders' Pokémon team will be custom-designed, with unique personalities and Pokémon team strategies. The gym leaders will be designed based on the preferences of our development team, and the playstyle and strategy of the gym leaders may be left up to randomness or an algorithm we design, but this will be further explored as we continue developing. Lastly, users will not have to worry about logging out of their account, because we will be using their username and emails as keys to tie their game information to their game account.

Our user interaction flow is as follows:

### **1. Create/Login**

- a. a new user creates a profile with a username, email address and password
- b. If the user is returning, they can login with their username, email address and password

### **2. Pokédex & Team Builder**

- a. a user can browse and search the Pokédex for a specific Pokémon. They can filter by type, generation and/or stats.

- b. They can select 6 Pokémon to create their team. As a preset, allow all Pokémon to hold one health potion and allow the user to carry 1-2 restore potions.
- c. For each Pokémon they select, they can select 4 moves from its movepool
- d. The team can be saved to the user profile, and can be edited, updated, saved or deleted (CRUD) from the user.

### **3. The Gym Lobby**

- a. Users can enter a gym lobby UI where they can read a description (scouting report) of each custom gym leader created by the dev team.
- b. When ready, they can begin a boss rush/run.
- c. Starting a new boss rush can create a new battle log entry onto the user profile.

### **4. Battling**

- a. The user can engage in a SIMPLE turn-based battle against each gym leader. Users can attack, switch Pokémon or use items held by Pokémon or revive.
- b. If the user wins, they can advance to the next gym leader with the current status of their Pokémon team persisting. If they lose, the run ends.

### **5. View Results and Statistics**

- a. The user's profile is updated with their win/loss (can also possibly time the runs)
- b. They can read a simple battle summary about their run
- c. Users can view badges they have earned
- d. Possibly view more challenges (e.g. beat all gym leaders without losing a Pokémon, etc a reach goal though!)

For our low-fidelity UI mockup, please view the powerpoint in our repository labeled "Pokémon\_UI\_Design" and below:



Login Page
<div><div>Pokemon Boss Rush</div><div><div>Username:</div><div></div></div><div><div>Password</div><div></div></div></div> <div><div>Login</div><div>Register</div></div>







	Header – Create account							
	<div><div>Please answer the following:</div><table><tr><td>Username</td><td>[insert text]</td></tr><tr><td>Email</td><td>[insert text]</td></tr><tr><td>Password</td><td>[insert text]</td></tr></table><div>Create</div></div>	Username	[insert text]	Email	[insert text]	Password	[insert text]	
Username	[insert text]							
Email	[insert text]							
Password	[insert text]							
	Footer							













Navbar	Header - Homepage	Profile Navbar
<div>Team button</div> <div>Gyms button</div> <div>Pokedex button</div> <div>badges button</div>	<div><div><div><div><div></div></div><div>user</div></div><div><div></div></div></div><div>Enter Gym</div></div>	<div>Logout button</div> <div>Profile button</div>
	Footer	





	Header - Pokedex	
	<div>search</div> <div><div><div><div></div></div><div></div></div><div><div><div></div></div><div></div></div><div><div><div></div></div><div></div></div><div><div><div></div></div><div></div></div></div> <div></div>	







Header – clicked/selected pokemon icon	
<div>search</div> <div>  </div>	<div>pokemon name</div> <div> <div>icon</div> <div>  </div> </div> <div>description</div>
Footer	





Header - team	
<div>search</div> <div> <div> <div>Team Name</div> <div>      </div> </div> <div>Edit</div> <div>Delete</div> </div> <div> <div>  </div> <div>Create a new team</div> </div>	



Header – view team	
<div>team name</div> <div>   </div>	<div>pokemon name</div> <div>icon [select a pokemon to view]</div> <div>description</div>
Footer	



Header - team	
<div>search</div> <div><div>Team Name</div><div></div><div>Edit Delete</div></div> <div><div></div><div>Create Team Name: <input type="text"/></div><div>Next</div></div>	
Footer	




Header – create new team	
<div>search</div> <div></div>	<div>pokemon name</div> <div>icon<div></div></div> <div>description</div> <div><div>Add</div><div>Delete</div></div> <div>team name:  </div>
Footer	



Header – create new team	
<div>search</div> <div></div>	<div>pokemon name</div> <div>icon<div></div></div> <div>description</div> <div><div>team name:     </div><div>Save</div></div>
Footer	


	Header - gyms	
	<div>search</div> <div>     </div>	
	Footer	


	Header – view gym	
	<div> <div>search</div> <div>  </div> </div> <div> <div>gym leader name</div> <div> <div>icon</div>  </div> <div> <div>description</div> <div>log of date completed</div> </div> </div>	
	Footer	












Header – view locked gym		
	<div>search</div> <div></div>	<div>gym leader name</div> <div><div>icon</div><div></div></div> <div><div>description</div><div>gym has not been unlocked yet</div></div>
Footer		

Header - Badges		
	<div>search</div> <div><div></div><div></div><div></div></div>	
Footer		

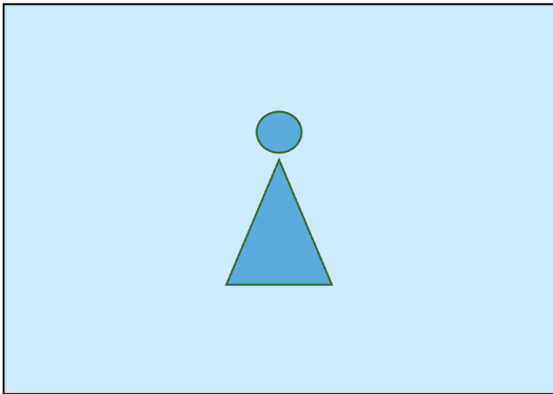










	Header – clicked/selected badge icon		
	<div>search</div> <div>  </div>	<div>badge name</div> <div> <div>icon</div> <div>  </div> </div> <div>battle summary</div>	
	Footer		









Navbar	Header - Profile		home button
	<div>  </div> <div>Username</div> <div>Badge Level</div>	<div> <div>Username</div> <div>Email</div> <div>Password</div> </div> <div> <div>Win/loss rate</div> <div>Average battle time</div> <div>Badges earned</div> </div> <div>Delete account</div>	
	Footer		

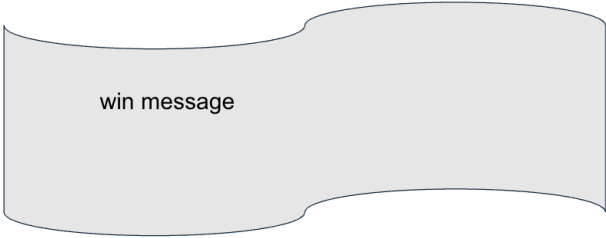
	Header – First enter gym	home button
	<div>  <p>Gym leader</p> </div> <div>Text... welcome to the gym.. etc</div>	
	Footer	

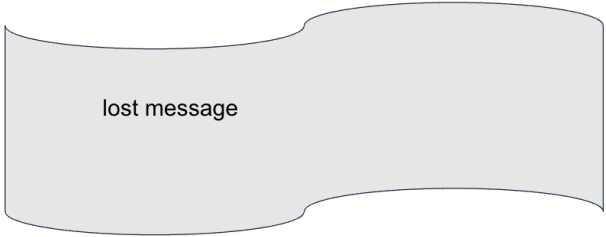
	Header – gym battle choose team	
	<div>  </div> <div>Please select your team</div> <div> <div>search</div> <div> <div>Team Name     </div> <div>Team Name     </div> </div> </div>	
	Footer	



Header – gym battle choose team		
		<div>search</div> <div> <div>Team Name     </div> <div>Team Name     </div> </div>
	<div>You've chosen [Team name]. Great choice!</div>	<div>You've selected [Team name]</div> <div>Continue to battle</div>
Footer		

	Header – Gym Battle		home button
<div>Team name</div> <div>   </div> <div>   </div> <div>   </div> <div>Health potions</div>	<div>Username</div> <div>  </div> <div>           Display only one:            - moves to choose from (buttons)            - then statement of move            - effect         </div>	<div>Gym Leader Name</div> <div>  </div> <div>           Display only one:            - statement of move            - effect         </div>	
	Footer		

	Header – Gym Battle	
	<div><div>Win</div><div><p>win message</p></div><div><div>Next button</div><div>home button</div></div></div>	
	Footer	

	Header – Gym Battle	
	<div><div>Lose</div><div><p>lost message</p></div><div><div>play again button</div><div>home button</div></div></div>	
	Footer	

## Project Work Distribution Overview:

- Design of database, QA
  - *Person(s): Everyone*
  - Create UML, write SQL queries, indexing
- Data Acquisition
  - *Person(s): Everyone*
  - Retrieve data from Kaggle Pokémon datasets and input into project databases
- Lead Database Engineer
  - *Person: Mengmeng Fang*
  - Implement UML design into code, ongoing database management/integration
- Lead Back-End Engineer
  - *Person: Pierre Font*
  - API Development, CRUD actions, User authentication
- Lead Front-End Engineer & UI/UX Designer
  - *Person: Hannah Kim*
  - Design views, Front-End testing, event handling, UI/UX implementation
- Team Captain & Project Manager
  - *Person: Hannah Kim*
  - Manages division of tasks, keeps track of milestones, works to maintain team cohesion
- Tech Lead & AI Engineer
  - *Person: Tanjie McMeans*
  - Technical mentorship, lead code reviews, resolving technical blockers, implementing personality-driven opponent AI, full stack development

