

# Project Backlog Overview

---

## Technical User Stories (for Development)

**Epic:** SCRUM-19 | **Status:** To Do

### User Stories

#### **Database Creation (SCRUM-20)**

*As a developer, I want to create a Database with all necessary information*

- **SCRUM-43:** Develop Database Model

#### **Domain Model Development (SCRUM-42)**

*As a developer, I want to create a Domain Model so that I can easily develop API endpoints*

- **SCRUM-46:** Visualize data using relational diagrams or graphics
- **SCRUM-47:** Split data into entities, value-objects
- **SCRUM-48:** Build useful Aggregates
- **SCRUM-49:** Implement as C# code

#### **Disconnect Handling (SCRUM-21)**

*As a developer, I want to handle disconnects so that battles can be resumed*

- **SCRUM-50:** Save game state on potential connection loss or regularly
- **SCRUM-51:** Implement waiting screen (1-2min)
- **SCRUM-52:** Either add reconnect button or do it automatically when connection is restored

#### **Team Composition Validation (SCRUM-22)**

*As a developer, I want to validate team compositions so that players cant use invalid teams*

- **SCRUM-53:** Define checks for validation
- **SCRUM-54:** Visually display what the user did wrong



## Account Management

**Epic:** SCRUM-1 | **Status:** To Do

### User Stories

#### **Account Creation (SCRUM-23)**

*As a new player, I want to create an account so I can save my progress*

- **SCRUM-55:** Choose a framework that handles account creation/validation (like clerk)
- **SCRUM-56:** Follow the frameworks documentation to implement it

### Secure Login (SCRUM-24)

*As a returning player, I want to log in securely so I can access my saved data*

- **SCRUM-57:** Kinda the same as above

### Profile Statistics (SCRUM-25)

*As a player, I want to view my profile statistics so I can track my progress*

- **SCRUM-58:** Update backend to save history of all matches
- **SCRUM-59:** Create endpoint which provides necessary statistics
- **SCRUM-60:** Develop UI that displays statistics in a nice manner

### Account Management (SCRUM-26)

*As a player, I want to change my username and password so that I can maintain my account*

- **SCRUM-61:** Check docs of authentication provider
- **SCRUM-62:** Enable or implement if necessary

---

## Team Management

**Epic:** SCRUM-16 | **Status:** To Do

### User Stories

#### Team Building (SCRUM-27)

*As a player, I want to build a team of 6 Pokemon so that I can prepare for battles*

- **SCRUM-63:** Create Endpoint that provides necessary information e.g getAllPokemon()
- **SCRUM-64:** Implement UI that displays Pokemon and allows user to choose 6 of them e.g Datagrid

#### Pokemon Stats Viewing (SCRUM-29)

*As a player, I want to view detailed stats of each Pokemon so that I can make strategic choices*

- **SCRUM-65:** Either add detailed information to initial request or implement another one to provide these details
- **SCRUM-66:** Implement UI that displays Pokemon Details (Stats etc)

#### Move Customization (SCRUM-30)

*As a player, I want to customize my Pokemons moves so that I can create different strategies*

- **SCRUM-67:** API endpoint which provides all possible moves of specific pokemon

- **SCRUM-68:** UI that dynamically displays the moves and allows user to switch each one of the 4
- **SCRUM-69:** Update team validation to check for valid move combinations

### Team Configuration Management (SCRUM-31)

*As a player, I want to save multiple team configurations so that I can switch between them*

- **SCRUM-70:** Update DB to store team configurations of users
- **SCRUM-71:** Update Backend to save team configurations
- **SCRUM-72:** Update frontend, add save option to team creation screen
- **SCRUM-73:** Add team configuration screen which displays and allows user to edit them

---

## Battle System

**Epic:** SCRUM-4 | **Status:** To Do

### User Stories

#### AI Battle (SCRUM-32)

*As a player, I want to challenge AI to a battle so that I can practice my strategy*

- **SCRUM-74:** Develop opponent model

#### Player vs Player Battle (SCRUM-33)

*As a player, I want to battle against other players so that I can test my skills*

- **SCRUM-77:** Update Backend to use something like SignalR for real-time-communication
- **SCRUM-78:** Update Frontend to allow player to search for online matches

#### Move Selection (SCRUM-34)

*As a player, I want to select moves during battle so that I can attack my opponent*

- **SCRUM-79:** Update Backend to allow for turn-based combat and add attack Endpoint
- **SCRUM-80:** Develop UI that enables user to call the Endpoint
- **SCRUM-81:** Write (Domain-Layer) Unit Test to make sure the calculations are correct

#### Pokemon Switching (SCRUM-35)

*As a player, I want to switch between my Pokemon during battles so that I can adapt my strategy*

- **SCRUM-83:** Update Frontend to allow users to swap active Pokemon

#### Battle Animations (SCRUM-36)

*As a player, I want to see battle animations so that I can follow the action*

- **SCRUM-84:** Figure out a way to get matching animations

- **SCRUM-87:** Update Frontend to display animations on attack

### Battle Forfeit (SCRUM-14)

*As a player, I want to forfeit a battle so that I can end an unfavorable match*

- **SCRUM-88:** Add Endpoint forfeit()
  - **SCRUM-89:** Add Forfeit Button to UI
- 

## Social Features

**Epic:** SCRUM-17 | **Status:** To Do

### User Stories

#### Friend Management (SCRUM-37)

*As a player, I want to add friends so that I can easily find battle partners*

- **SCRUM-90:** Check if the authentication system allows for friends, OTHERWISE:
- **SCRUM-91:** Update DB to store each user with his friends
- **SCRUM-92:** Update Backend to include Endpoints to add and remove friends
- **SCRUM-93:** Update Frontend to include Social Screen which displays friends and provides access to Endpoints

#### Profile Viewing (SCRUM-38)

*As a player, I want to view other players' profiles so that I can learn about their stats and strategies*

- **SCRUM-94:** Add Endpoint getStats(ID)
- **SCRUM-95:** Update UI to allow user to view stats by right clicking a friend (optionally opponents match-history)

#### Friend Challenges (SCRUM-39)

*As a player, I want to send battle challenges to friends so that I can compete with them*

- **SCRUM-97:** Update Frontend

#### Battle History (SCRUM-40)

*As a player, I want to see my battle history so that I can review my performance*

- **SCRUM-98:** Update DB to store Battle History
  - **SCRUM-99:** Add GET Endpoint for Battle History
  - **SCRUM-100:** Update Frontend to allow users to view their Battle History
- 

**Total:** 5 Epics | 20 User Stories | 58 Tasks | All items currently in "To Do" status