# C:\Users\t00193238\AppData\Local\Microsoft\Windows\INetCache\Content.Word\projectlogo.pngIntroduction to Database Programming 2018

PokeTrumps Project

Designed & Developed By:

Conor O’Brien

David Herlihy

Jack O’Hara

Mike Edgar

Manfred Buron

Documentation Initial Creation: 04/10/18

About the project:

PokeTrumps is a game in the style of the popular card-game "Top Trumps". This version will use Pokémon as the cards. The objective of the game is to unlock all of the cards. The user will be able to challenge the AI to a battle. Within the battle, users will take turns choosing a statistic to compare with that of the opponent’s card. The winner of each round is the user with the higher statistic. The winner of each match is the user who won the most rounds out of five, with the winner receiving a new Pokemon as reward. The user will only be able to use 5 Pokemon in each battle. The user can choose which Pokemon they want to use in battle from the Pokemon they have unlocked.

## Important decisions behind development

Why did we choose this project?

We initially decided we wanted to do a Pokemon project as it is a game we’re all interested in. We all enjoy playing the games and thought we could develop our own version of a game involving the concepts.

How did we decide on the final concept of the project?

We found that the complexities of the game we wanted to create were too big an operation to undertake. We eventually started thinking about the game in terms of simple statistics and found the concept quickly developed into a game much like Top Trumps.

How did we manage our task development?

We utilized Azure DevOps initially to swiftly plan our project development. Once we had a rough outline of the scale of our project we drafted up our user stories and tasks that we had to complete.

How were the tasks delegated amongst the team?

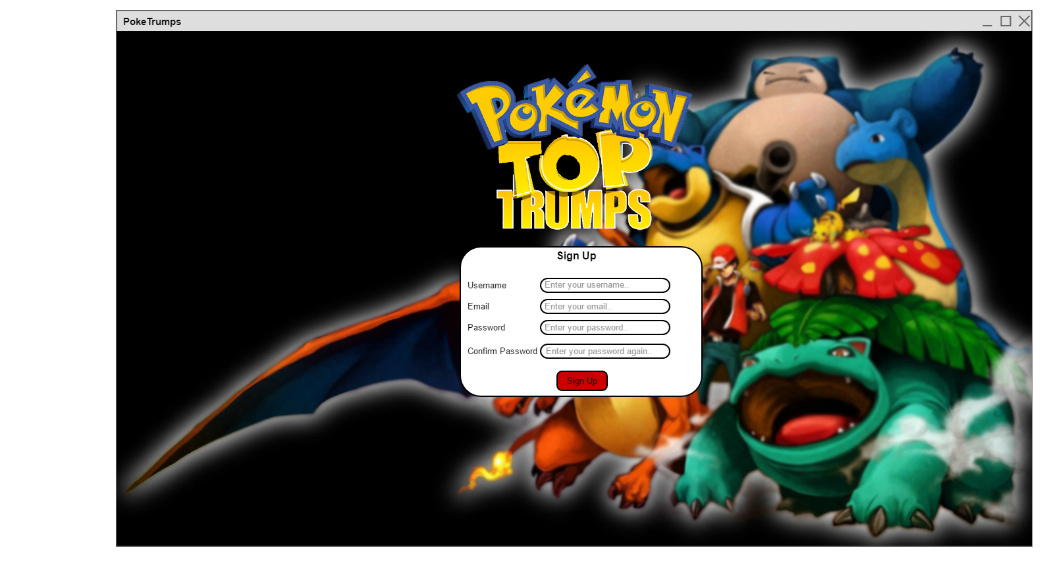
We were curious as to developing individually. In this situation, everyone worked on their own specific features. We found that when everyone did a little bit of every segment, everyone would be more informed on the structure of the project. This proved effective during group conversations we had on how we wanted to progress.

# User Functions

## Account Management

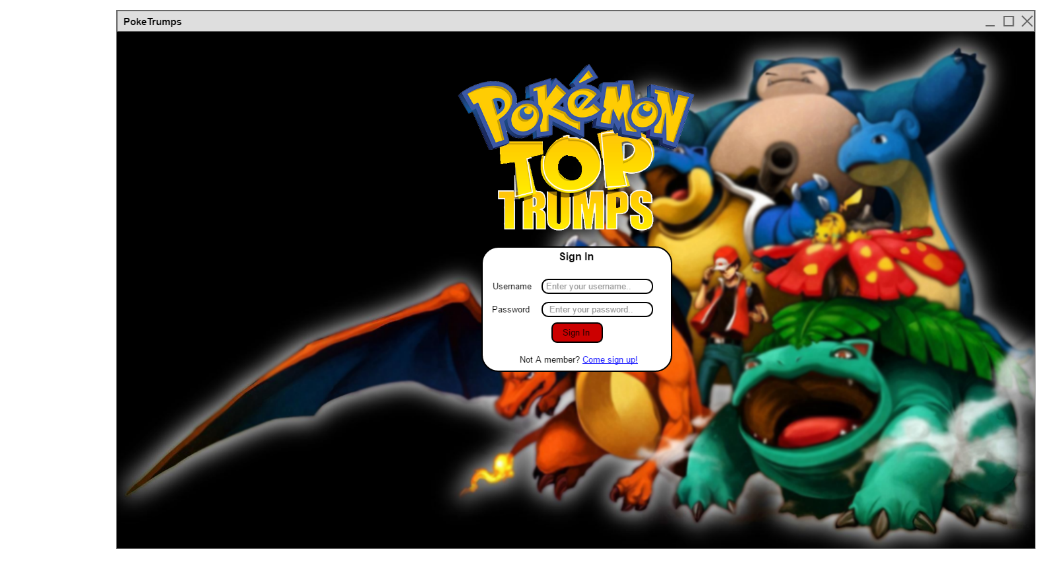
### Sign Up

Before playing the game, users must create an account that will be stored on the database. A user must enter a valid unique username, a valid unique email address, a valid password and a confirmation password that matches the password. If these fields are entered correctly a user will receive 5 random Pokemon to begin the game.



### Log In

If a user has already registered an account, they can log in again using their username and password. If these fields are both accepted and match the information in the database, they can log in to their account.



### Log Out

Users are able to exit their current session via the log out button. This is easily accessible on the home screen.

### Add Friend

Users can add their friends to their friends list via the main menu search feature. Users search for their friends by searching for their friend’s username. All usernames are unique so this does not encounter problems. Once found, users click the ‘Add’ button.

### Remove Friend

Users are able to remove any friends from their friends list if they decide they wish to. They select the specific friend from the list and click ‘Remove’.

### Get Starting Pokemon

Users will be able to receive 5 starting Pokemon from the total Pokemon database. These will be randomly chosen and 5 will be the specific amount given.

### Display Pokemon List

Users can check to see how many Pokemon they have obtained through their battles and what Pokemon they are currently using in their party and compare them with each other.

### Edit Team

The user can choose to swap some of their Pokemon between their main party of 5 and between their total Pokemon collected using this function. A user can’t swap a Pokemon with its self.

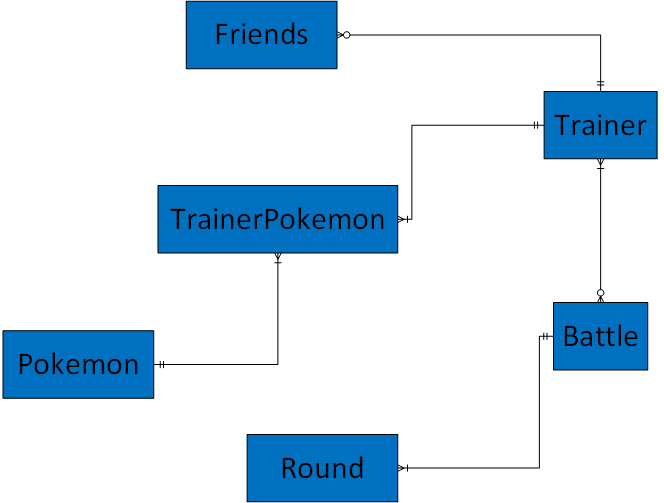
### Select Favourite

The user can choose which of the Pokemon within the Pokemon they own so that it is easier to locate them when viewing all collected Pokemon.

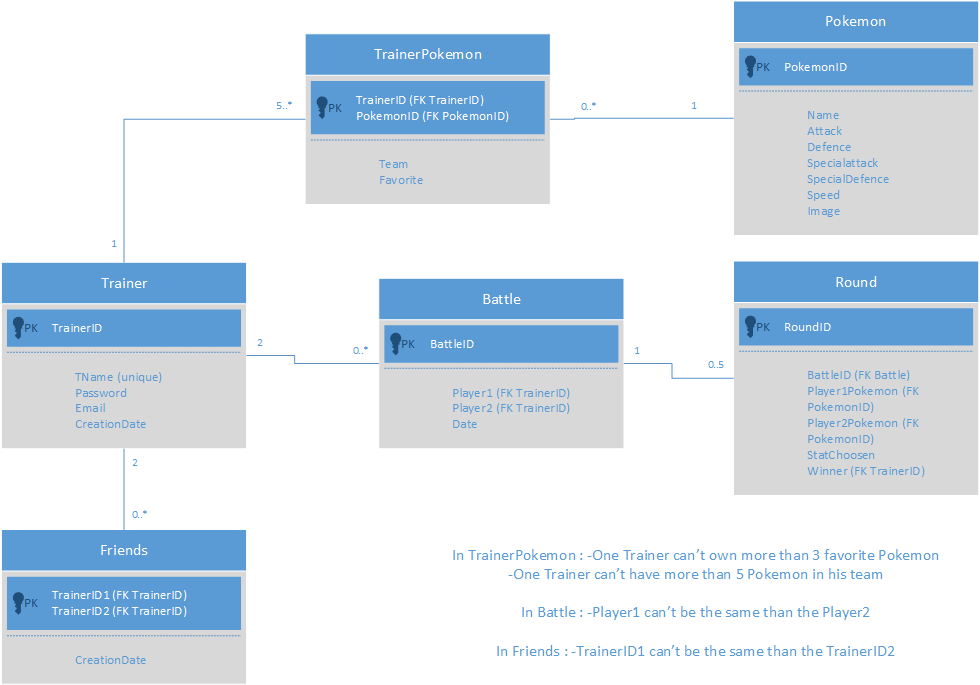
Start Battle

The user may choose to initiate a battle with the AI

Enterprise Model



Entity Relationship Diagram



Discussion of the design

Functional Model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entity  -------------  Function | Trainer | Pokémon | Battle | Round |
| Create Account | **X** |  |  |  |
| Login | **X** |  |  |  |
| Log Out | **X** |  |  |  |
| Add Friend | **X** |  |  |  |
| Remove Friend | **X** |  |  |  |
| Get Starting Pokémon | **X** | **X** |  |  |
| Display Pokémon List | **X** | **X** |  |  |
| Edit Team | **X** | **X** |  |  |
| Select Favourite | **X** | **X** |  |  |
| Start Battle | **X** |  | **X** |  |
| Flip Coin | **X** |  | **X** |  |
| Choose Pokémon | **X** | **X** | **X** | **X** |
| Choose Stat |  | **X** | **X** | **X** |
| Check Round Result | **X** |  | **X** | **X** |
| Check Current Round |  |  | **X** | **X** |
| Is Winner | **X** |  | **X** |  |
| Get New Pokémon | **X** | **X** |  |  |

**Database Schema**

**Trainers Table**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| **trainerID(PK)** | INT | Unique identifier for a Trainer |
| tName | VARCHAR2(15) | Username for a trainer |
| password | VARCHAR2(10) | Password for the trainer to log in |
| email | VARCHAR2(30) | Email for the trainer’s account |
| creationDate | DATE | Date user created account |

**Pokemon Table**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| **pokemonID(PK)** | INT | Unique identifier for a Pokemon |
| name | VARCHAR2(10) | Name of the Pokemon |
| attack | INT | Attack stat |
| defence | INT | Defence stat |
| specialAttack | INT | Special Attack stat |
| specialDefence | INT | Special Defence stat |
| speed | INT | Speed stat |
| Image | BLOB | Image of the Pokemon |

**TrainerPokemon Table**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| **trainerID(PK)** | INT | Identifier for which triner this Pokemon belongs to |
| **pokemonID(PK)** | INT | Identifier for which Pokemon this Pokemon is associated with |
| team | BOOLEAN | Shows if the Pokemon is selected in the Trainer’s team |
| Favourite | BOOLEAN | Shows if the Pokemon is selected in the Trainer’s favourite Pokemon |

**Friends Table**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| **trainerID1(PK)** | INT | Unique identifier for a Trainer 1 |
| **trainerID2(PK)** | INT | Unique identifier for a Trainer 2 |
| creationDate | DATE | Data user’s became friends |

**Battle Table**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| **battleID(PK)** | INT | Unique Identifier for the Battle |
| trainerID1(PK) | INT | Identifier for Trainer 1 |
| trainerID2(PK) | INT | Identifier for a Trainer 2 |
| battleDate | DATE | Data of the Battle |

**Rounds Table**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| **RoundID(PK)** | INT | Unique identifier for the Round |
| **battleID(FK)** | INT | Unique Identifier for the Battle |
| **Player1Pokemon(FK PokemonID)** | INT | Identifier for Pokemon 1 |
| Player2Pokemon(FK PokemonID) | INT | Identifier for Pokemon 2 |
| StatChosen | ENUM | Stat chosen for battle |
| Winner (FK TrainerID) | DATE | Data of the Battle |

Pokemon Stats