**README for Online Pokémon Card Store Server and Client**

# Introduction

This README is to provide information about the Online Pokémon Card Store Server and its client program. The server is used to allow users to perform various actions related to buying and selling Pokémon cards. The instructions on implemented commands and instructions for running the server and client are listed below.

# Server

The server is the main control for managing the databases that store user and card information. The server is meant to receive commands from the client program.

## How to Run the Server

1. Open a terminal or command prompt.
2. Navigate to the project directory where server.py is stored.
3. Run the server file with the following command: **python server.py [host]**
   1. [host] is an optional argument that will be used to connect to the server host in the argument. If no argument is listed, then the local host IP will be used “127.0.01”.
4. The server will now be open and waiting for commands from the client.

## Commands

### BUY

The ‘**BUY**’ command allows users to purchase Pokémon cards from the server. This command involves the following parameters:

* ‘card\_name’: The name of the Pokémon card.
* ‘card\_type’: The type of the Pokémon card.
* ‘card\_rarity’: The rarity of the Pokémon card.
* ‘card\_price’: The price of each card.
* ‘quantity’: The number of cards to purchase.
* ‘user\_id’: The ID of the user making the transaction.

When the command is executed, the server verifies the parameters, afterwards it checks the balance of the user. If successful, then the user’s balance is updated, and the cards are added to the user’s information, either adding it to the database or updating the existing one.

### SELL

The ‘**SELL**’ command allows users to sell Pokémon card to the server. This command uses the following parameters:

* ‘card\_name’: The name of the Pokémon card.
* ‘quantity’: The number of cards to purchase.
* ‘card\_price’: The price of each card.
* ‘user\_id’: The ID of the user making the transaction.

When the command is executed, the server verifies the parameters, afterwards it checks to make sure the number of cards are equal or less than the quantity of cards. If the conditions are met, the the card information is updated or removed if there are no more cards that the user owns, and the user’s balance is updated.

### LIST

The ‘**LIST**’ command allows users to see the list of cards owned by a specific user. This command uses the following parameters:

* ‘user\_id’: The ID of the user whose cards are to be shown.

If the user ID provided is valid, then it will show the list of cards owned by the user or display that the user owns no cards.

### BALANCE

The ‘**BALANCE**’ command allows the user to check the balance of a specific user ID. This command uses the following parameters:

* ‘user\_id’: The ID of the user whose balance will be shown.

If the user ID provided is valid, then it will return the specified user’s balance.

## Closing the Server

Ending the program can be done in two different means.

### QUIT

The QUIT command will end the client’s program, while keeping the server open for the next client program.

### SHUTDOWN

The SHUTDOWN command will end the server’s program.

# Client

The client allows users to interact with the server by sending commands and receiving responses.

## How to Run the Client

1. Open the terminal or command prompt.
2. Navigate to the project directory where client.py is stored.
3. Run the client file with the following command: **python client.py [server\_host]**
   1. [server\_host] is an optional argument that will be used to connect to the server host in the argument. If no argument is listed, then the local host IP will be used “127.0.01”.
4. The client will attempt to connect to the server and prompt the user for commands. If the client is unable to connect to the server, then the program will end.

## Sending Commands

* Enter the desired command from the list from the server section in the client’s input, and the client will send the command to the server to be processed.
* To exit the client, send the ‘**QUIT’** command.