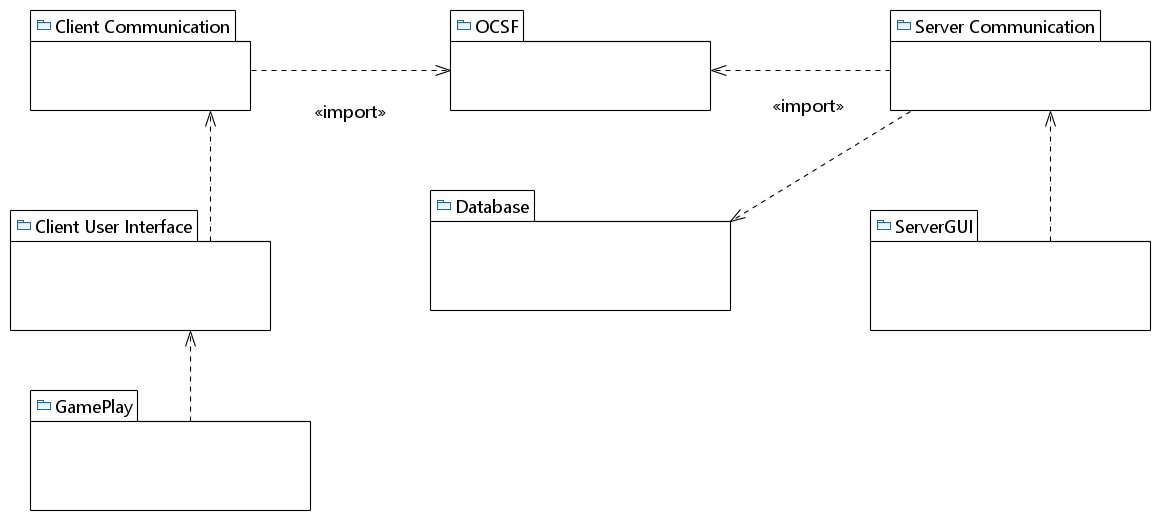
**BlackJack**

**Design Document**

**Team # 5**

**Team Member**: Darius Bowers Nathan Huang Shuo Dai Steven McCormick

**Package Diagram**



**Package (Sub-System) Input/Output**

**Package: Client Communication**

CreateAccountControl

CreateAccountData

GameClient

InitialControl

LoginControl

LoginData

Input:

* Login data from client user interface
* Created account data from client user interface
* IP and port data from user input
* User info from the server, such as username and password for validation
* Server status info
* Dealer info from server
* Game data from other players in the game

Output:

* Login Data sent to server
* Created account data sent to server
* Client status and data sent to server
* IP and port data to connect to server

Output is provided to OCSF, which is forwarded to ServerCommunication

**Package: Client User Interface**

ClientGUI

CreateAccountPanel

GameBoardPanel

LoginPanel

InitialPanel

 HighscoresPanel

Input:

* Input from user in Initial Panel for login or create account
* User input for creating account for username, password, password verification
* User input in login panel for username, password, and button clicks
* User input on Main Panel for starting a game
* Login result for logging in and creating accounts from Client Communication
* Strings for displaying player information from Client Communication, such as score and username

Output:

* Login data sent to client communication
* Created account data sent to client communication
* Start game output sent to Gameplay

Output is sent to ClientCommunication subsystem

**Package: Gameplay**

Card

Deck

Input:

* User account score and username
* User input for gameplay decisions (betting, etc.)
* Dealer info from server
* Game info from client communication
* Client info for connecting to game

Output:

* calculate to client communication
* Gameplay info to client communication

Output is sent to ClientUserInterface, then to ClientCommunication

**Package: Server Communication**

GameServer

Input:

* User info from the database
* Login data
* Created account data
* Connection information from client communication
* Gameplay data from client communication
* Account balance data from client communication
* Client connection information
* Database validation results

Output:

* Gameplay information to client
* User info to the database
* Account balance to database
* Username and balance to Main Panel
* Database validation errors to client

Output is sent to OCSF, then to ClientCommunication subsystem

**Package: Server GUI**

ServerGUI

Input:

* User input from server GUI panel for port and timeout
* User input for listening controls

Output:

* Port number, timeout, and IP to server communication
* Listening commands to start/close server

Output is sent to ServerCommunication subsystem

**Package: Database**

Database

Input:

* User login info from server
* User balance from server

Output:

* User login info to server
* User balance to server

Output is sent to ServerCommunication subclass

**Package: OCSF**

Input:

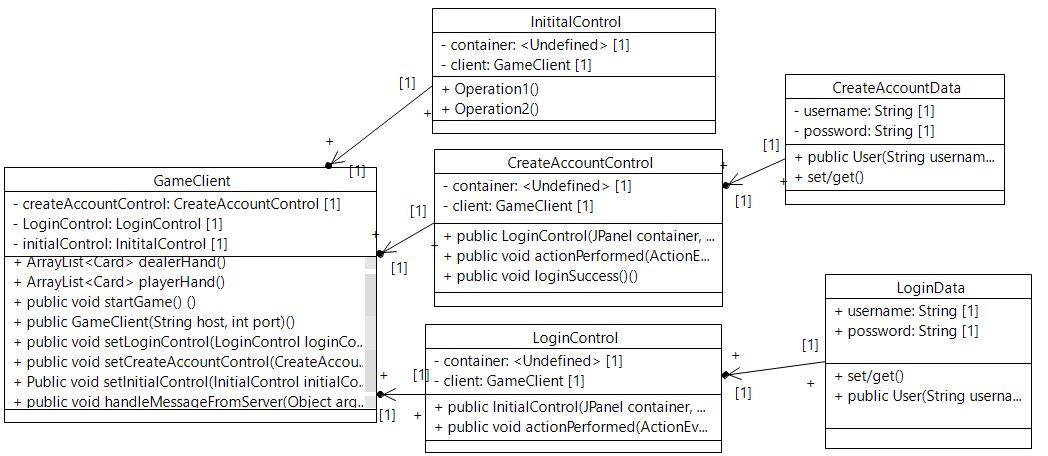
* Client info from client communication
* Server info from server communication

Output:

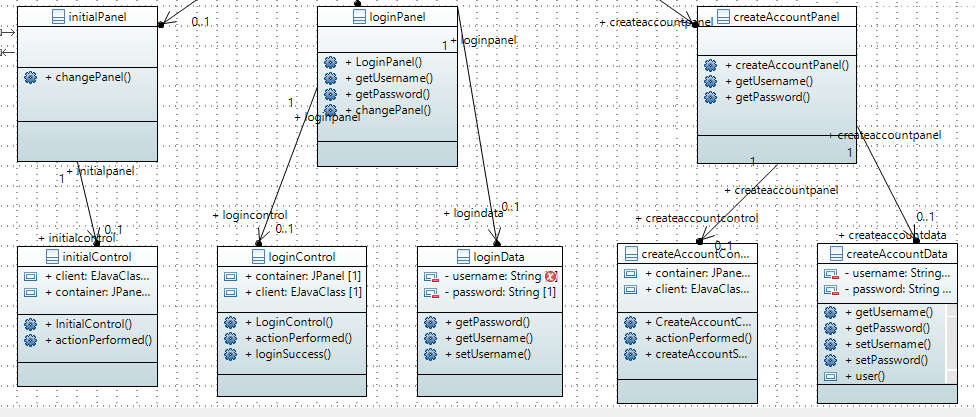
* Client info to server communication
* Server info to client communication

Output is sent to either ClientCommunication subsystem or ServerCommunication subsystem

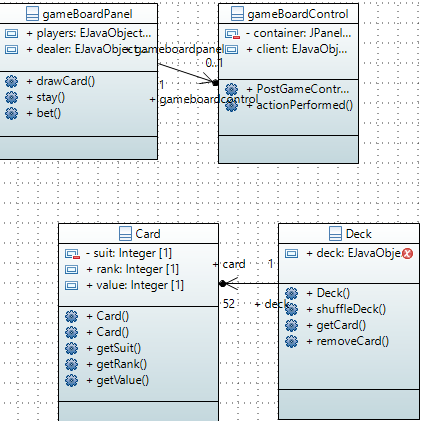
**Class Diagram Part III**



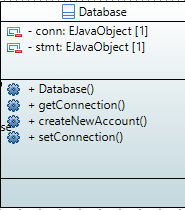
**Client Communication Classes**



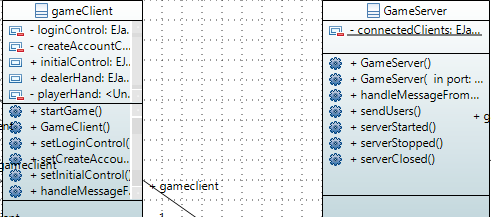
**Client User Interface Classes**



**Gameplay Classes**



**Database Classes**



**Server Communication Classes**