|  |  |
| --- | --- |
| *Class Name:*  **Action (INTERFACE)** | |
| *Responsibilities:*  Know data on a specific instruction  Return relevant information to a inquisitor | *Collaborators* |

|  |  |
| --- | --- |
| *Class Name:*  **ActionCard (EXTENDS Card)** | |
| *Responsibilities:*  Knows data on a specific Ivanhoe action card  Return data on targeting specifications, name, colour, description | *Collaborators* |

|  |  |
| --- | --- |
| *Class Name:*  **ActionWrapper** | |
| *Responsibilities:*  Know an action  Know the origin (player) of this action  Facilitate the traceability of the action | *Collaborators*  Action  Player |

|  |  |
| --- | --- |
| *Class Name:*  **Card (ABSTRACT)** | |
| *Responsibilities:*  Knows name, colour  Knows comparability to another card | *Collaborators* |

|  |  |
| --- | --- |
| *Class Name:*  **Client** | |
| *Responsibilities:*  Knows information about the game state  Knows information about the player  Facilitates the transfer of data over the network from client to server Operates on input from the Client Input  Processes this input to send commands or update the game state  Prompts the GUI to update  Provides restrictions on player operations | *Collaborators*  GameState  ClientInput |

|  |  |
| --- | --- |
| *Class Name:*  **ClientInput** | |
| *Responsibilities:*  Know its parent client  Receive input from the command line  Signals the client to operate on this information | *Collaborators*  Client  ValidCommand |

|  |  |
| --- | --- |
| *Class Name:*  **ClientView** | |
| *Responsibilities:*  Know its parent client  Displays and updates this information on the screen  Load images from the game resources  Moves the player between the Lobby and Game views  Receives graphical input from the player and processes it via the client  Provides graphical restrictions to certain actions  Provides graphical options to certain actions | *Collaborators*  Client  GameState |

|  |  |
| --- | --- |
| *Class Name:*  **Colour** | |
| *Responsibilities:*  Knows information on a certain colour profile  Is one of (RED, BLUE, YELLOW, GREEN, PURPLE, NONE)  Knows comparability to another colour or string | *Collaborators* |

|  |  |
| --- | --- |
| *Class Name:*  **Command** | |
| *Responsibilities:*  Knows command arguments  Knows the origin of the command  Can determine validity of command Can return its information to an inquisitor | *Collaborators* |

|  |  |
| --- | --- |
| *Class Name:*  **CommandInterface (INTERFACE)** | |
| *Responsibilities:*  Uses its invoker to execute a command | *Collaborators*  CommandInvoker |

|  |  |
| --- | --- |
| *Class Name:*  **CommandInvoker** | |
| *Responsibilities:*  Can execute a command | *Collaborators*  CommandInterface |

|  |  |
| --- | --- |
| *Class Name:*  **Deck** | |
| *Responsibilities:*  Knows a list of its card members  Knows a list of used members (discard pile)  Can recreate itself when it has no members  Can be asked for the top card  Can be shuffled | *Collaborators*  Card |

|  |  |
| --- | --- |
| *Class Name:*  **Display** | |
| *Responsibilities:*  Knows a list of its card members  Knows the score of its member based on the current tournament  Can be modified in the following ways (add, remove, removeLast, removeAll, removeValue)  Can return the highest or lowest value of its members Can return if one of its members is a Maiden  Can be display via text | *Collaborators*  Card |

|  |  |
| --- | --- |
| *Class Name:*  **DisplayCard (EXTENDS Card)** | |
| *Responsibilities:*  Knows its value  Knows its color  Knows its in-game name  Can return these known values to an inquisitor | *Collaborators*  Colour |

|  |  |
| --- | --- |
| *Class Name:*  **GameState** | |
| *Responsibilities:*  Knows the players in the game  Knows the data on its current tournament  Knows the turn order and who should be next  Knows the last played colour of tournament  Can add or remove cards from displays  Can add or remove cards from hands  Can calculate the highscore among players  Can return an error given a card played at an incorrect condition  Can start or end a tournament  Can return the validity of a target for a card  Can change the Shielded or Stunned state of a player  Can add or remove a token from a player | *Collaborators*  Card  Player  Deck  Colour  Command  Tournament  Token |

|  |  |
| --- | --- |
| *Class Name:*  **Player** | |
| *Responsibilities:*  Knows its color  Knows its name  Knows its network state (in-game, waiting, etc)  Knows its display  Knows its hand  Knows its Shielded or Stunned State  Know if it it’s the current turn  Knows its participation in the current tournament  Can accept a token | *Collaborators*  Display  Card |

|  |  |
| --- | --- |
| *Class Name:*  **PromptCommand (IMPLEMENTS CommandInterface)** | |
| *Responsibilities:*  Knows data on a prompt sent from a server  Knows a message  Knows a target  Knows options for the prompt  Can execute its information to prompt a player | *Collaborators*  Player  Server |

|  |  |
| --- | --- |
| *Class Name:*  **SearchThread** | |
| *Responsibilities:*  Knows its parent Server Seeks incoming connections at its servers network location  Passes these connection to its parent | *Collaborators*  Server |

|  |  |
| --- | --- |
| *Class Name:*  **Server** | |
| *Responsibilities:*  Knows a list of connections to clients and their players  Knows restrictions on starting a game  Knows banned IP  Knows its location on the network  Knows a ‘master’ game state  Knows a set of languages  Knows an action card waiting to execute Can receive a connection from a search thread  Can process input from a server input thread  Can process input from a client connection  Manages order of input in a first in, first out manner  Knows the network state (ready, waiting, in-game) of its clients  Facilitates starting a game  Can relay game information to the console  Can unban or ban a connection  Can create an AI and start its operation | *Collaborators*  SearchThread  ServerThread  ServerInput  GameState  Command  ClientAI |

|  |  |
| --- | --- |
| *Class Name:*  **ServerInput** | |
| *Responsibilities:*  Knows its parent Server Seeks input from the console  Knows the validity of this input  Uses its parent to process this input | *Collaborators*  Server  ValidCommand |

|  |  |
| --- | --- |
| *Class Name:*  **ServerThread** | |
| *Responsibilities:*  Knows its parent Server Interfaces over the network with a particular client  Can update its client  Can receive information from its client  Is aware of the connectivity of its client | *Collaborators*  Server |

|  |  |
| --- | --- |
| *Class Name:*  **Token** | |
| *Responsibilities:*  Knows its colour  Knows its origin  Knows comparability to another token | *Collaborators*  Colour |

|  |  |
| --- | --- |
| *Class Name:*  **Tournament** | |
| *Responsibilities:*  Knows its colour  Knows its title | *Collaborators*  Colour |

|  |  |
| --- | --- |
| *Class Name:*  **ValidCommand** | |
| *Responsibilities:*  Knows a command  Knows a list of arguments  Can return the validity of the command from its data  Uses chains to return complex validity scenarios | *Collaborators*  Command |