Elementi del gruppo:

Alessandro Bellotta

Mattia pejano

Andrea Cerfoglia

Nicola Colantuono

Nicola Colantuono

COMMUNICATION PROTOCOL

Progetto IS gruppo AM29

Sommario

[Connection and setup phase 2](#_Toc132381216)

[Take phase 4](#_Toc132381217)

[Place phase 5](#_Toc132381218)

[End game phase 6](#_Toc132381219)

Messages and network overview……………………………………………………………………………7

[Conclusion 8](#_Toc132381220)

# Connection and setup phase

When the client sends a request of connection to server there will be two possible responses.

1. If the player is the first to connect, Server will be replied with a **insertParameterTOCreateGameRequest**.

Server send:

Login\n

Inserire username, numero dei giocatori\n

1. If the Server is already open, it will reply with **gameExist** and **joinMatchRequest** response.

Server send:

Login\n

Inserire username\n

This will continue until the number of clients connected to the server is equal to the number insert by the first player. A timer will start when the first player connects to the server and after that the server will be closed if the game has not started.

# Take phase

Server:

È il tuo turno\n

Scegli cosa pescare\n

Client:

[[int Xi] [int Yi], [int Xj][int Yj],[int Xk][int Yk]]

Server response:

action valid [y,n]

Immagine che contiene diagramma

Descrizione generata automaticamente

Methods invocation

After server receives the coordinates of the tiles the player wants to take from the

board, it creates an instance of takeObjecArgs and the view method takesArgsObservable.notify is called, with that instance as parameter.

notifyChanges notifies the client about every change occurred in the virtual view.

# Place phase

Server:

è il tuo turno\n

scegli dove piazzare\n

Client:

[int column]

Server response:

action valid [y, n]

Immagine che contiene diagramma

Descrizione generata automaticamente

Methods Invocation

After server receives the message with nickname and bookshelf's column number where the player wants to place the tiles, an instance of PlaceObjectArgs is created, with nickname and column number as parameter and the correspondent methods in the view will be called.

Model method PlaceObject already has turn handling functions, so when the tiles are placed in the bookshelf the current player changes and the next turn starts.

There are also all the functions to check if the board needs the refill, points calculations for common goals and personal goals and calculations of endgame phase and winner.

notifyChanges notifies the client about every change occurred in the virtual view.

# End game phase

The game can end in three different ways:

1. When a player completes his bookshelf and the current player is the first of the turn (all players have to complete last turn).
2. one player has disconnected.
3. the method to refill the board has to be called but in the Tile bag there are not enough tiles.  
   End game message will be sent to all clients connected and cause the disconnection from the server.

**EndMessage :** [“il vincitore è + String nickname”]

Immagine che contiene diagramma, schematico

Descrizione generata automaticamente

# Messages and network overview

We have decided to implement the network side of the project using only Socket technology with TCP connection protocol.

The messages exchanged between client and server will be implemented using the serializable interface and the classes named GameDTO and PlayerDTO as parts of the DTO (Data Transfer Object) pattern. These classes have the purpose of representing the changeable parts of the game such has bookshelf and board, so that when these parts change during the game, the view can be notified and can show the changes.