**A page (at most) explaining how your design accounts for - or would need to be adjusted for - possible future networking of the game.**

**Before we look into the future possibility of our game network support it becomes vital for us to understand the concept of game network and its fundamentals about the structure of the internet network. From our study, we understood that**our Blokus game from its current design pattern doesn’t support game networking. There are a lot of grey areas for instance, ((1) Although we have multiple player games (2-4 player). We did not consider **online** multiple plays) also, ((2) inconsistent design architecture(eg. poor error handling etc. ) which limits our version of the game to be played online.

Despite this limitation, it is still possible to network our game. At first, Because of power and ease of use of java programming, its potential for future cross-platform multiple game is doable. Also, programming in java helps us provide easy network support. So, for our game networking, we would choose to implement using client-server architecture which is the standard as it is easier to execute. As it requires low bandwidth. So we would use socket which would allow multiple players to connect to a host server that's the game itself and that would then allow data transmission over the network. So Based on our code design we would have to implement a host server through which the data is managed and it will be responsible for connection facilitation and the game action. While the client-server will be responsible for displaying the GUI itself and will be used by the player to connect the host and enjoy the game. Besides creating the server-client sockets we would also need to consider a few changes in the blokus game that we have designed.

As far as the design for offline blokus that we have created and the online networking version that we wish to create. These differ only a few things, as for the most part of the game. It would function in the same manner. One of the differences would be defining a variable which informs the player(s) from the clients-servers end on who’s turn it is while also disabling other players (s) ability to use any functionality of the game until the player has taken his/ her turn. While it will be client-servers responsibility to inform the host server when to enable other player turns.

This is still the basic theoretical approach on the game networking it is possible that the actual implantation may bring about complication which is not accounted for in the explanation.