Your assignment will be marked as follows: 50% for correctly implementing/testing the requirements of the game per se using TDD and JUnit, and 50% for implementing and testing networking, again using TDD and JUnit. A feature that is not tested correctly using TDD/JUnit will not receive marks.

Here is the order in which you **must** tackle the functionality of the game:
(as we will verify in your repository)

- process a select command so that the target, attack direction, attack time, defense direction, and direction time of a relevant player are accessible. How you design this is up to you and there are several possible approaches. You are NOT evaluated on the merits of your design.
- given 2 players who have selected their target, attacks and defenses, process the roll of one of these 2 players to resolve the attack:
    - you must test attack times
    - you must test getting the actual direction of an attack.
    - you must then test actual direction against the target's selected defense.
    - you should test the case where both time and matching actual direction against selected defense make the attack successful
- Then process the second roll to resolve the second attack and generate the msg OUTCOME sent to each player listing ALL player names and number of wounds (ie 0 or more)
- Test resolving attacks involving 3 then 4 players. This ends the testing of game functionality per se.
- Test, for 2, 3 and 4 players, initiating a session and having the server detect the required number of players and issuing back to the clients the "Game ready" message
- For 2 players, test sending the select messages from the 2 clients in any order and having the "ROLL TO RESOLVE" message sent to all players. Repeat for 3 and then 4 players.
- For 2 players, test sending the roll messages from the 2 clients in any order and having the server send the OUTCOME message back to each player. Repeat for 3 and then 4 players.
- You must test several rounds

Please note that any of the items listed above may require several TDD iterations!