**[Networked System Development](https://bb-gbc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_7062576_1&course_id=_216644_1&group_id=&mode=view)**

20% of your total mark.

Remember!  Git commits are required during every hour of your work.  It is not acceptable to hand in project files, you must hand in a repo with commits over time that illustrate the development of your work.

Congratulations!  The start of this assignment marks a very notable milestone in your journey becoming a networked gameplay programmer.

You have now navigated the huge amount of complexity in the technologies that go into making networked gameplay possible.  Further, the client-server solution that we have developed thus far is enough to make online pvp possible which is, in and of itself, great, but what is more is this.

You can turn a function call into a line of text.  You can turn a line of text into a function call.  You can move lines of text between computers.  Let that seep in fully.  Imagine how your lists of ClientToServer and ServerToClient transfer types could continue to expand.  Imagine the places that you might find yourself calling your SendNetworkData function.  Imagine the lines of code that you might add to the end of your client/server RecieveIncomingData functions.  In a sense, everything that you must do now is just a review and repetition of what you have already done.

Moving forward the challenge will be to develop solutions that interact with the structure that you have already established.  Your challenges are as follows.

Challenge 1:  Develop player to player communication, create a series of prefixed msgs.

Challenge 2:  Develop the ability for a client to join a game room as an observer.

Challenge 3:  Develop a replay system.

Optional Challenge 1:  Develop player to player communication, text msg exchanges.

Optional Challenge 2:  Separate “instance” servers for hand off of in-progress game rooms.

Optional Challenge 3:  Metrics tracking.  Sends to a server every key the player pushes.

Optional Challenge 4:  Design and Develop a matchmaking system.

Optional Challenge 5:  Server redundancy

Optional Challenge 6:  Replace the transport layer that we are using with that of another framework tool.  Traditional socket, Photon, Mirror, MLAPI, ect.

Optional Challenge 7:  Push the envelope on real time play.  Take on the positional update requirements of an online car racing game.  That is, make it so that positional updates and inputs are as in sync as possible.

Optional Challenge 8:  Design a suite of tests which could be used to -as fully as possible- test EVERY part of your code.

Optional Challenge 9:  Leaderboards

Optional Challenge 10:  Incorporate your networked play solutions developed in this course with another project that you are doing.

Etcetera, Optional Challenges:  Name a feature and then develop it.

You will be given the help that you are able to ask for.  Do ask well in advance and for what you specifically need help with.

Where each of you will be required to submit separate projects, you are not facing these challenges alone.  Where you are not allowed to directly copy the work of your peers, you are allowed and being directly encouraged to exchange information with each other.  Consider this challenge, can you create an outline to a solution that is so effective that everybody is able to then write the solution?  A person who is capably filling a lead programmer role would be able to do so.

I will facilitate at least one brainstorming session to help you (as an entire class) map out your thoughts on each of the challenges.  Furthermore, this is not a Unity course, so I will develop for you any UI once you (as a class) have requested it from me.  This brainstorming session is formally scheduled for week 7, but I expect to deliberate on a given topic whenever it arises.  In many ways, this is a class wide activity.  Do your best to help your fellow scholars out.

You are almost there, rise to the challenge of development!  If you are able to develop the three mandatory challenges in this assignment, you can certainly consider yourself to be a capable networked developer.

Completion of optional challenges will merit a letter of excellence.  If you are able to develop solutions for the challenges listed in the optional list, you are pushing the hours of practice required to become an expert.

Rubric:

Completion Challenge 1: 15%

Completion Challenge 2: 25%

Completion Challenge 3: 50%

Completion of Optional Challenge 1: 10%

Completion of optional challenges can be used to merit marks appropriately.  Speak with me for more details.