**My progress so far up until Nov. 7, 2014:**

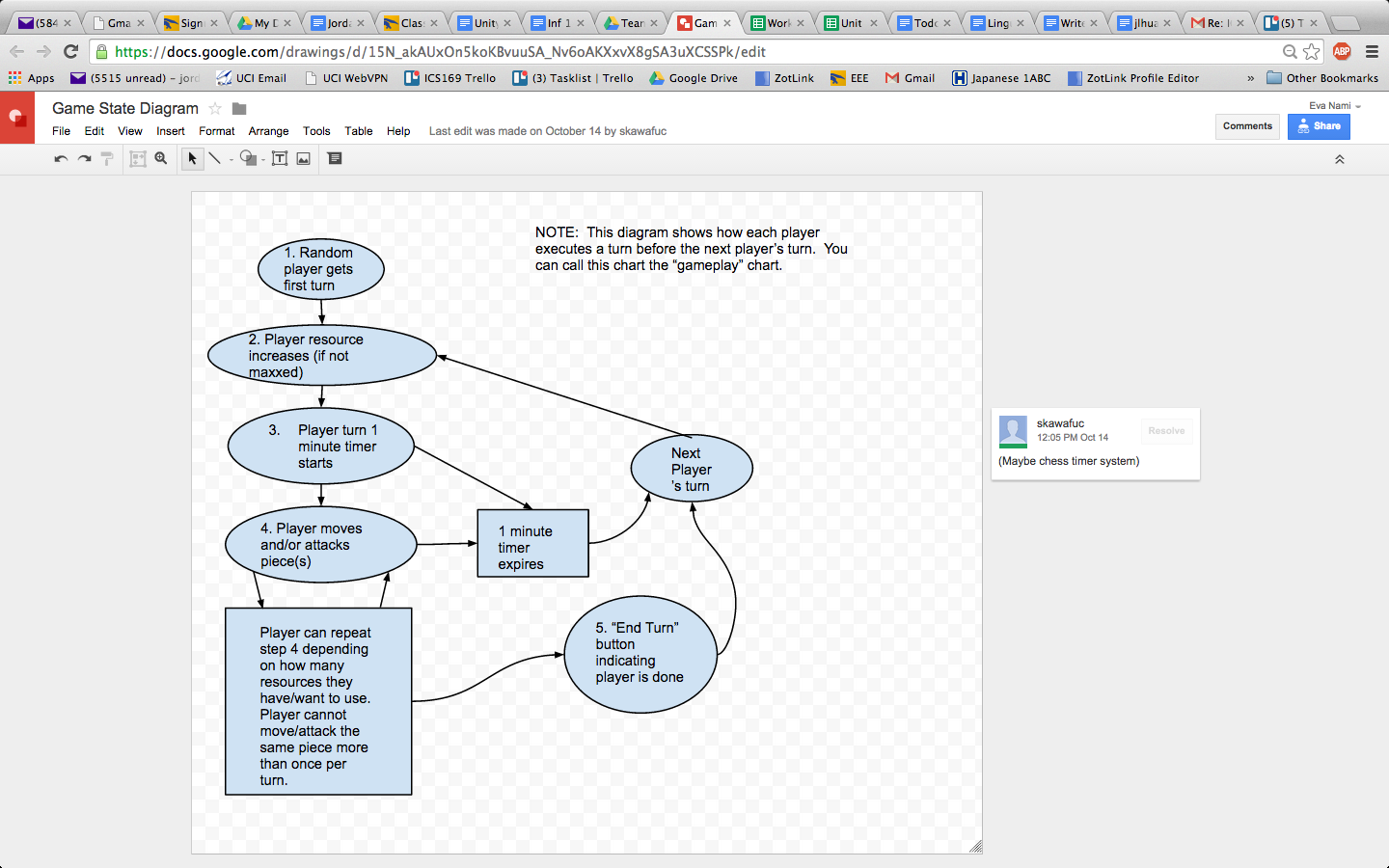
1. What I finished so far:

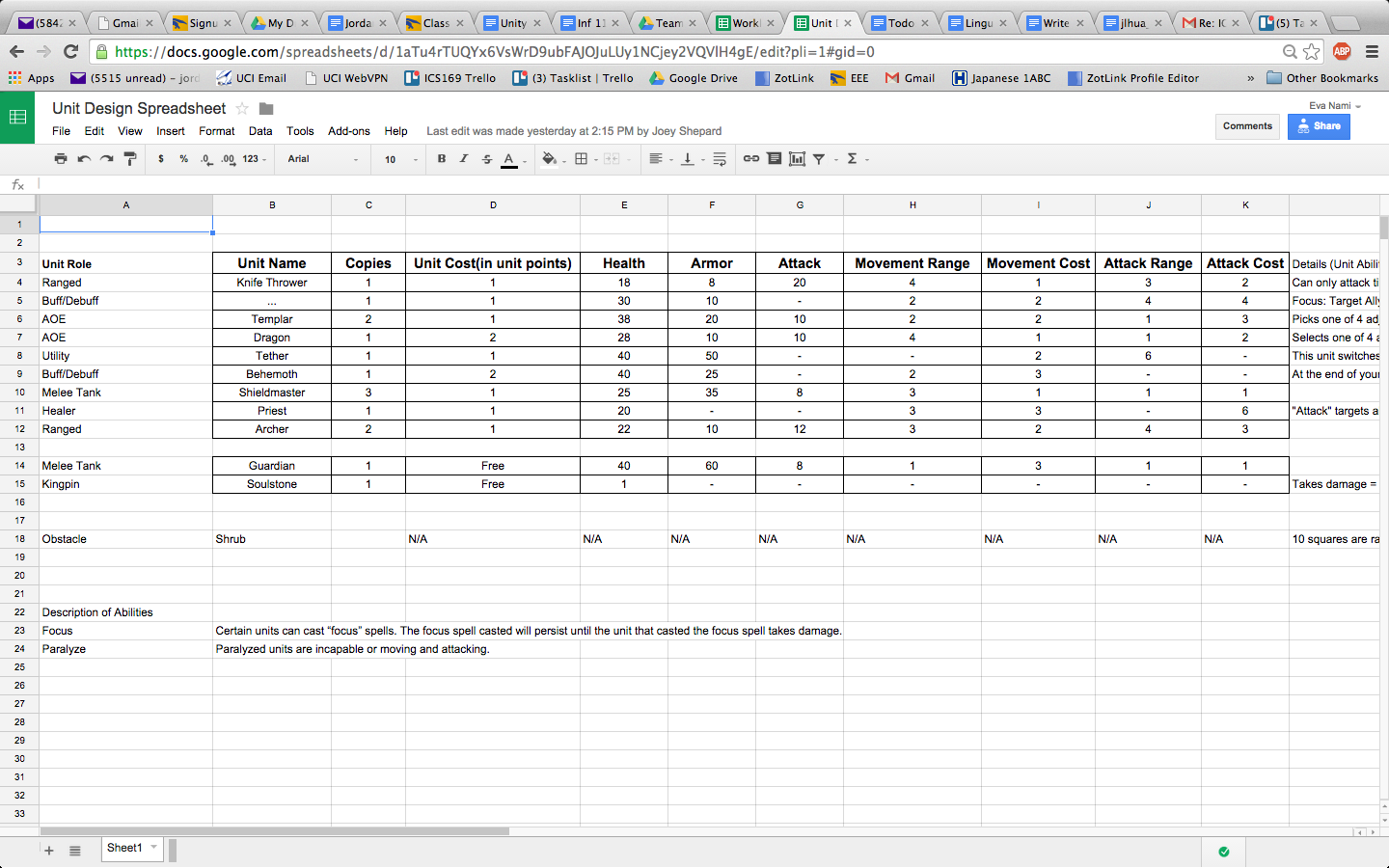
For the first 3 weeks and ongoing, I carefully updated all of the UML State diagrams, the game design document, and the unit design spreadsheet to update any changes we discussed every meeting for designing the game. After we carefully divided up the tasks on what part of the game that each person works on based on the final revision of the game design document, I worked on implementing most of the setup screen. On the setup screen, I completed spawning each of the pieces with their appropriate attributes. For my brainchild design of dragging and dropping the the units on the grid, I implemented the drag-drop controls and snapping each piece onto half of the grid that Steven migrated over to the setup screen from the board scene, on my own. I of course made sure all my code actually helps Daniel, Joey, and Steven to complete their tasks, and I made multiple revisions. I made sure that all my code is considerate of what the others added into the unity project, so I am always looking at their code and helping designing their implementation with them.

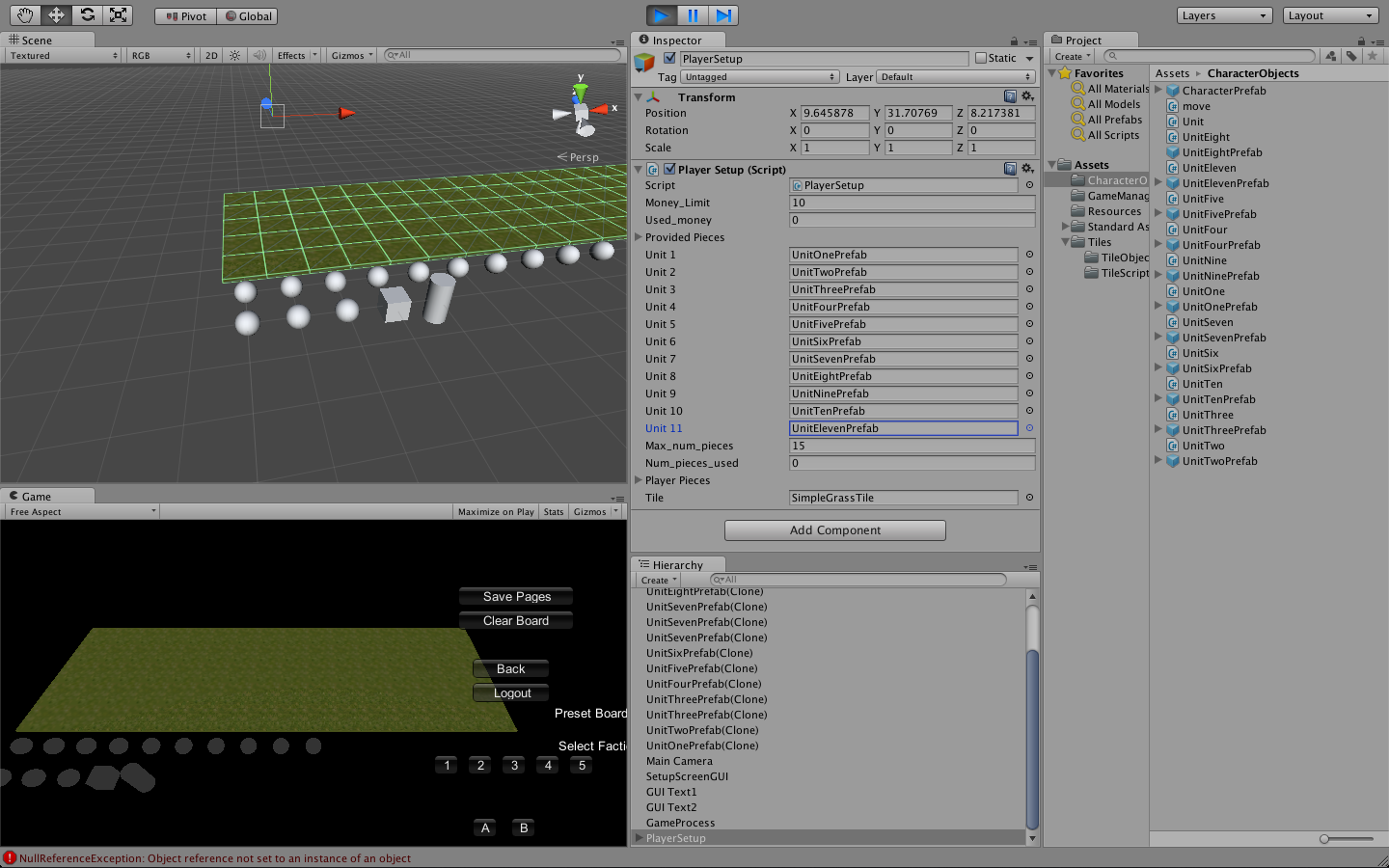
1. What I did for the project that can’t be seen on the workload sheet:
   1. I did paper modeling to test to make sure that the game design does not have any issues before we began proceeding with how to implement the final game’s design, both alone, and presenting at the beginning of all of the team meetings.
   2. While implementing each of the pieces as physical C# code, I implemented it in a way in which their ID’s, roles, and abilities were represented in a way that can be transferred as a byte to the server. We later scratched the idea of sending the ID’s, roles, and abilities of each piece the way I designed it, so all that code in particular for the pieces were all deleted.
2. What I am still working on:

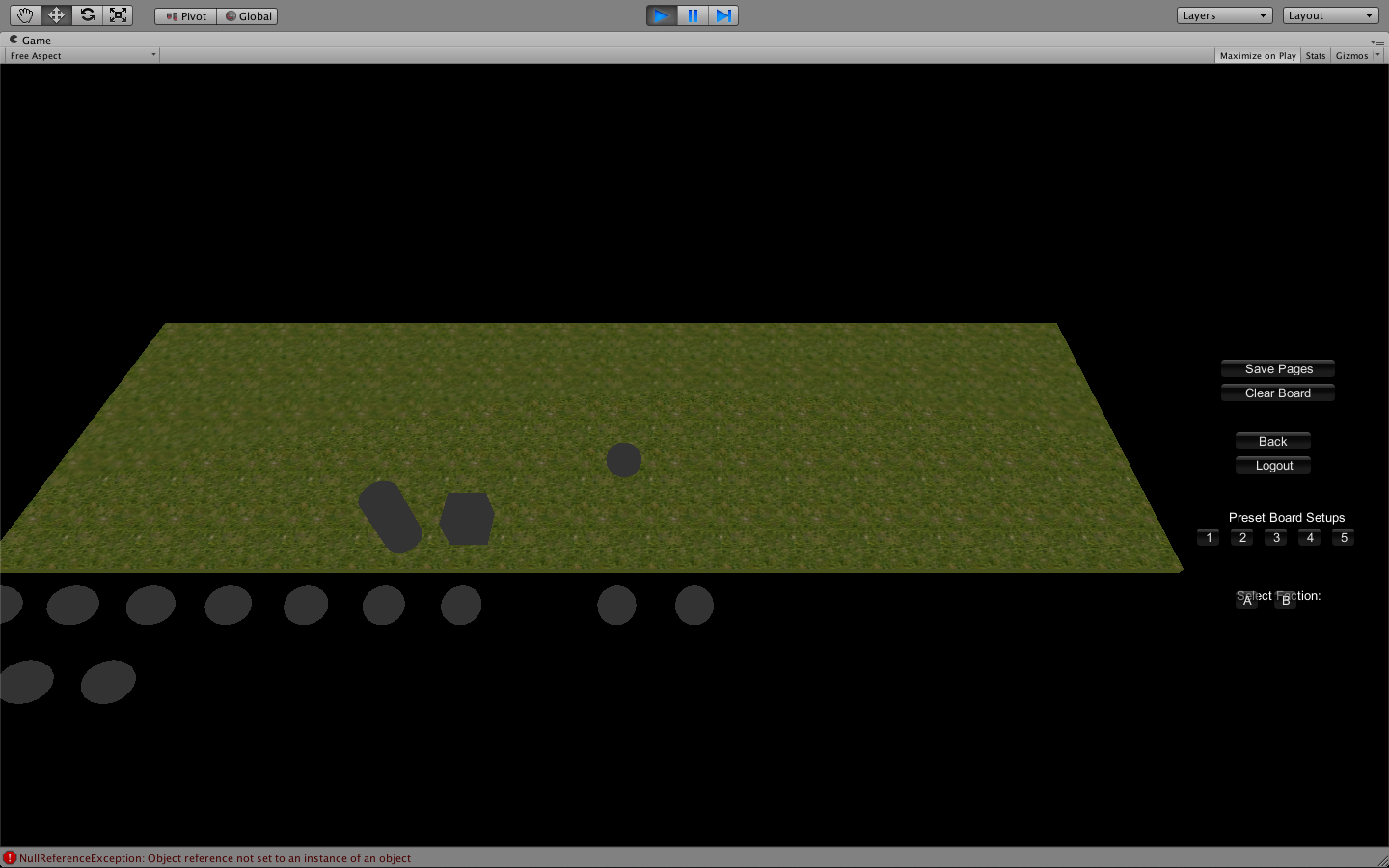
At the moment, I am trying to to get the setup screen ready so that the player information recorded from player’s interaction on the setup screen can be ready for usage by the server and all the other scenes of the unity project, saving the arrangement in how the players move his/her pieces on the board, and designing a system that limits how many pieces are allowed to be placed onto the board. If we decide to make any slight changes on the board scene particularly, everything completed so far may have to be re-implemented again, or at the very least be put back as an ongoing task. But for now, the tasks I completed are completed.

Some screenshots:









/\*

Unfortunately, the inherent design of our particular kind of game required work strategies in which nearly every aspect of the tasks, even the ones assigned to us, turned out to be completely collaborative. Because Steven was working on the gameplay aspect of the units on the board scene, and I worked on setting up the units on the player setup scene, the mere fact that we were implementing a different part of 1 topic of the game design (the game pieces at the same time), meant that we had no choice but to carefully see each other’s progress back and fourth in person. We corrected and modified each other’s code, and a lot of code was usually deleted on a daily basis on both my end of the task and Steven’s end simply because we had to make sure that what I did with setting up the units on the setup screen did not affect any code that used the same assets on the board scene.

\*/