



GAME FACE ON

This was week 13 and we tackled a number of major and minor issues. These ranged from fixing minor bugs on the gameplay side to getting our first version of the application published on Android.

Publishing the Application

This week was a milestone week for the project in this aspect. We got our first application published into the Android market. This was available on the Android market place early in the week but there was some issue with signing the application correctly. This prevented the application from being installed on any Android devices. We are looking into how to best resolve this issue as soon as possible. On the iOS front, our first build of the application has been entered into Apple's pipeline for their review process after much struggle. This was mainly a problem since we now have a new Apple Developer Account to publish the application which meant we had to go through the entire process again. We expect to hear back for the review in over a week's time.

Internal Play Test

This week we were initially targeting to conduct a play test either at ETC or on CMU main campus at the spring carnival. But since we are targeting a major play test with MetaVerse ModSquad next week, we decided to focus our energy on refining the game play and making the whole architecture as stable as possible. This meant we conducted internal play tests on Wednesday, Thursday and Friday using mobile devices mimicking actual user

scenarios as much as possible. Through this we found out critical areas where we need to improve. All the major tasks and bugs we identified were logged and can be referenced as part of this spreadsheet. We also discussed and improved on the visual feedback for the game heavily to make it simple, much more attractive and appealing.

Server Stress Test

This was a big thing we did not anticipate spending this much time on testing the server. We uncovered quite a bit of issues with the server once we delved into testing stress testing the server. The server was not able to handle the load of 100+ users (simulated) jumping on the network at the same time. Looking into our existing implementation of the server we uncovered several key issues with multi-threading in a thread unsafe Unity environment and also mostly exhausting our input network buffer for the incoming packets of UDP data. For this we have decided to move to a different architecture of the server. This would be outside Unity but still remain in C# so we can capitalize on the existing code base. But for better parallel execution of the incoming UDP requests



GAME FACE ON

from the client devices, we will be using a Thread-Pool for assigning incoming requests from users to different threads in the pool. Overall this architecture theoretically is much more stable and should allow us to handle more incoming traffic than our existing implementation.

Professional QA and Play Testing

We are currently working with main campus and MetaVerse ModSquad to conduct a major play test for our game. There is a lot of paperwork involved in order to get the required approval from main campus which is taking time to coordinate and setup. Our initial plan is get the setup done during the next week and a couple of people from the team will most likely fly out to orchestrate the entire play test. This would be a really good opportunity for us to see how the players respond in such a large environment.

Soft Opening

Coming Monday we have Soft Opening. For this we have decided not to do a presentation but to let the faculty play the game with a live demo. This game would be setup in our

project room itself and it would be a good opportunity for us to get direct feedback from faculty members on the game. Our core focus at this stage is to bring the game to a near bug-free near complete version before we demo it to the faculty. We will be prepping up our website for faculty review at Soft Online Opening which will include gameplay videos, past play test videos and much more.

That's it from this week, see you all next week!

Mayank Grover

