

James Taylor

Final Report for TwentyOne

Group SB_04

Answer these questions within your answers:

- (a) What are the key events that are to remembered?
- (b) What are the key lessons that you learnt that you can carry over to future projects. Think back starting from the time the teams were assigned, then project proposals were made, screenshots developed, requirements, architecture, framework demo, test plan, and finally the demo.
- (c) What would you do different if you were assigned the same team and the same project again?

1.1 What Went Wrong

A few things did not go quite as planned. To start, websockets was frustrating to learn and attempt to implement and was never actually implemented. This ended up being the main hindrance in our game, and I am still not very happy about how far we got in regards to websockets before the demo

In general client side fell behind in the last sprint because of single player -- We had to do single player and also try to catch up to what our team members were doing server side -- and it did not all get finished.

1.2 What Went Right

A lot went right. We built an app from nothing to a largely functional blackjack game that you can not only play with your friends online, but also train against a computer controlled dealer.

I went from having 0 experience with Android Studios to being a fairly competent Android app developer. I also had no idea how a client connected to a server, and a server connected to a database. This might be the most useful thing that came out of the project, as it is crucial knowledge moving forward in my software engineering career.

Our group worked very well together. We had some long nights (even some that went into the early morning). We communicated well and each contributed a somewhat equivalent amount. I don't think that anyone felt like they were left out or doing everything.

1.3 Lessons Learnt

- Team Work

In general, our team worked very well together. The only thing that we could have done better is to keep on each other more about not waiting until the last half of the sprint.

- Technologies Used

I learned a lot about Android Studios. Android Studios does a lot for you, but it can be kind of difficult to learn how to originally use it.

I also got to experience the wonder that is Volley and HTTP Requests for the first time. This was very frustrating and confusing at first, but once I learned what was really going on between the client and server it became more clear.

- Processes (Reqs, Design, Coding, Debugging, Testing)

Our design philosophy was pretty much non existent, which looking back I would totally do differently. We just made something work and then moved on in the client. I think it was a mix of trying to make something on time for the demo and learning new technologies. I feel like I forgot to design good code. No custom interfaces, not as object oriented as it needed to be. In the end it ended up being more difficult. This was a very important take away -- It is worth it to take the time in the beginning to design your project well.