

Meeting Wednesday 8th December 2015 12pm (Meeting Ended 14:43)

Mo Waqas Aslam, James Grant, Alex

Harry To Do Friday

- Design Mobile App UI
- Polish any Front end tasks

James

- Learned majority of Golang functionality excluding concurrency components (to do by Friday)
- Started coding: Dev environment is completely set up
- Relevant Go Lang libraries have been identified and imported to dev environment

To Do Friday

- Will finish learning concurrency tutorials in Golang
- Code some back end: setup and configure imported classes (i.e socketio server)
- Push code to git
- Write Readme.txt to run Go Lang

Mo

- Has started on Angular coding with cookies
- API is ready to use (User, anon user, lobby)

To Do Friday

- Will have fixed Cookies issue/active session
- Set cookie Timeout if page is closed/inactive
- Some bugs, fix bugs and push changes

Alex

- Completed lobby API in Node.Js
- Users, anon users, lobby are up on Git

To Do Friday

- Need to do docs for api (ReadMe file)
- Moving Serving to node
- Research node/go lang server

Mo, James, Alex: Clear plan on Architecture of application by Friday

**

IMPORTANT

**

**

**

**** TO NODE OR NOT TO NODE ****

Notes

Node server and Go lang server discussion and implementation efficiency

Node may be redundant, done some research and reading to find pros/cons.

Miscommunication as node was thought to be used for client side rather than backend.

Pros and cons of node server and Golang server.

Found node is being used as server side when it can be done in Go lang backend rather than having 2 servers data passing from client side to node to back end is creating extra steps and less efficiency.

Loads of architecture drawings completed

Research

- Go lang to sql: Supported by GoLang
- Look at Golang to JSON: (assumption: should be supported)
 - JSON is built in to Golang and is supported
- Need for node... Yes or No?
- Separate instance for api server??
- Web sockets general readings
- Running game instances

Plan A: JavaScript instance

Plan B: go lang instance

Future implementations:

Mobile app

Client loads game html and JS for game

Someone answers question which is socket connected

Checks the answer with the server

Server processes it and returns the JSON data which has the answer

And will update all clients to their mobile phones

Concurrent routing

