Project Update

While we were able to get all the parts done by themselves we did not test interactions between client and server.

Product	Overall Design Inspection
Date	2/27/2016
Author	Alex Shelley
Moderator	Matthew Fouts
Inspectors	Rami Bitar,Matthew Fouts,Alex Meyer
Recorder	Alex Meyer

Defect #	Description	Severity	How Corrected
1	Yet to complete matchmaking service	1	Wrote a function that periodically scans the list of users and updates a list of matched users.
2	Does not secure passwords	2	We will hash and salt passwords before being passed from client
3	Does not request info to make matches	3	Changed it to a preference system, more free
4	Design doc doesn't make any mention of preferred age range	3	Added them to the document

Product	Backend/type_script/sql.ts - file that will connect node.js to the sql files Inspection
Date	2/28/2016
Author	Alex Meyer
Moderator	Alex Shelley
Inspectors	Rami Bitar,Matthew Fouts,Alex Shelley
Recorder	Rami Bitar

Defect #	Description	Severity	How Corrected
1	Find Matches statement UserID_one does not exist	2	Change to UserID1
2	Changes the way blocked is stored to a true or false value in match instead of own table;	1	We moved isBlocked to the same table as Matches to make our aligthrom work better
3			
4			
5			
6			
7			
8			
9			
10			
11			

Product	Front end html inspection- public folder
Date	2/28/2016
Author	Matthew Fouts
Moderator	Rami Bitar
Inspectors	Rami Bitar, Alex Shelley, Alex Meyer
Recorder	Rami Bitar

Defect #	Description	Severity	How Corrected

1	The website's main page still shows the log-in form, even if you're already logged in.	3	
2	Profile picture is not uploading correctly	2	Fix Code.
3	Format is really ugly	3	Any leftover dedicated to fixing UI
4	Messaging not finished	1	Finish.
5	Redirecting when it shouldn't	2	Fixed, error in compiling

Unit Testing

Product	BackEnd/am_unit_testing.js - file that will connect node.js to the sql files Unit Testing
Date	3/1/2016
Author	Alex Meyer

Defect #	Description	Severity	How Corrected
1	SQL statement for matches return illegal parsing error ER_PARSE_ERROR for get user messages	2	change to order by not sort by
2	sure that UserID1 and UserID1 are mathces instead of UserID1 and UserId2	2	Change everything to use UserID1
3	Error: Uncaught, unspecified "error" event. ([object Object])	2	Change "true to True as it can not read it right and make it a json.

Product	BackEnd/testServer.js - file that will connect node.js to the front end Unit Testing

Date	3/1/2016
Author	Matt

Defect #	Description	Severity	How Corrected
1	Username from backend with userID when they get a message	3	Add extra field in getmessages
2	Need to put JSON into Query not Body, breaks everything	1	Put into Query
3	Birthday input system broke testServer.js	2	Updated to match Birthday in Server.js
4			

1. We decided to create our scripts to run tests instead of using a given framework. We did research into a few different types,such as nemo, and decided for our simple app it would be just as easy to write our own test cases and just run them with bash scripts. These test scripts can be found in the BackEnd/tests/ file, but needs to be ran in the BackEnd file. Our strategy was to go through the test log and test what was needed. TestAm.sh will test the basic sql calls and does not need a server running. testServer.sh is for the front end to test to make sure that there calls send the right form to the back end. These are the only two test we used for this case as the rest are used for later integration testing.