Josh Scotton

Abdallah Al-Sulaihat

Nick Cannon

Yahya Osman-Husein

Review

1. GUI

a. Interactions

i. Do all interactions on each respective GUI operate?

ii. Does the GUI have each interaction properly labeled or described?

iii. Does the GUI have the proper connections to the associated code?

2. Database

a. Cohesion

i. Is there code allowing a connection between the software and the database?

ii. Does the database have the capacity to store all required information and fields?

3. Code

a. Maintenance

i. Is everything coupled and cohesive?

ii. Is all the code manageable and commented?

iii. Is there any errors?

iv. Are good coding conventions used?

b. Testing

i. Have all the classes been tested individually?

ii. Is there appropriate code for connection to the database?

iii. Have the classes been grouped and tested in different groups?

iv. Have the classes been integrated with the GUI and interactions tested?

4. Software

a. Testing

i. Has the software been tested with success?

ii. Has the simulation been properly integrated into the software?

iii. Has every button and interaction pane been used and tested?

Our biggest glaring issue is currently integrating all of the different aspects into one software. The GUIs seem to be fine but as of yet have not been integrated and tested with the code corresponding to the different interactions. The general fields for the database have been created but have yet to be integrated as well.

Another issue is with the simulation, we have yet to figure out how to execute the open-source code and because of that, cannot integrate it into the software.

As a group there has been minimal testing on the different aspects of the software outside of the GUI.