Name: Stephen
Date: June 17, 2016
Current Module: Networking in Python
Project Name: Safetydome

Project Goals:

The goal of this project was to make a restful application that through the navigation of html could get to locations created by the programmer. We were to create routes and query and give information back based on what router they chose

Considerations:

- Does the database exist
- Can it be connected to
- How will the application respond if an unknown page is requested
- If an unknown combatant or battle is asked for
- If there are no battles recorded

Initial Design:

My program is structured using Flask and giving the routes to set up. It use flask to take in the routes and then uses SQL to make queries to a database and pushes them back based on the webpage.

Data Flow:

The dataflow of this program simplys follows the routes given in Flask and if the url matches the the function will be called and inside each function a SQL query will be ran and the information asked for in each route will be given.

Communication Protocol:

None

Potential Pitfalls:

- SQL Injection
- Not making sure the information that comes back is correct
- Passing information if an incorrect way
- Making sure users only have read only access

Test Plan:

User Test:

1. Test connection to ~/
2. Test connection to /combatants
3. Test connection to /combatant/<id>
4. Test connection to /battle

5. Test connection to /battle/<id>-<id>
6. Test connection to /results

Automated Test:

None

Test Cases:

1. Connected to localhost:8054 to see the index page
2. Connected to localhost:8054/combatants by the url and by clicking the link from the index page
3. Connected to localhost:8054/combatants/<id> by the url and by clicking any combatant name
4. Connected to localhost:8054/battle by the url and by clicking the link from the index page
5. Connected to localhost:8054/battle/<id>-<id> by the url and also by clicking the battle id number in the battle id column on the battle page
6. Connected to localhost:8054/results by the url and also by clicking the link from the index page

Expected Result:

For each of these I expected to navigate to the correct url/route and have the proper information displayed without any errors.

Conclusion:

For this project one problem I have was making sure the information from my SQL queries was getting properly passed into the html templates. Another important thing I have to pay attention to was that I was querying the proper information for each Flask route. A possible future change would be to use SQLAlchemy instead of Flask.