Use Case Name: Use inputting data into client

Short Description: When the user inputs data on the web client and that data is sent, how that data is interpreted by the rest of the system. Whether there may be errors in it or not, what is returned will be a response corresponding to the situation.

Actors: User [as the user will be the only person or thing that is outside of the system that interacts with the system and subsystems (if there are any)]

Triggers: When the user inputs data from the web client.

Preconditions: User has access to internet, Server is online and is able to respond to requests, and Database is online and functioning correctly.

Post conditions: User’s request goes through presentation’s servlet and the corresponding methods are invoked to process the request and return the correct response to answer the user’s request.

Results: User sees the new page or updated page.

Outline:

Perfect Flow:

1. User inputs data through web client
2. Data is sent through presentation’s servlet
3. Through the list of servlets, the correct one is chosen and the servlet’s methods are invoked to process the request.
4. Through the presentation’s servlets, server and database are contacted (if necessary), the necessary actions are then made and a response is formed.
5. The response goes through presentation’s methods to be translated back into a response that the web can understand.
6. The web client obtains the response and builds the page based on received response to update what the user sees.

Alternate Flow 1 (what the user requested doesn’t exist or is wrong):

1. If the request doesn’t match up with any of the servlets with the correct requirements, the final servlet will be used.
2. Final servlet will be the safety net that will process the outlier requests to return to the user a response without contact with the server or database that includes what went wrong (from incorrect type of request to the incorrect link).