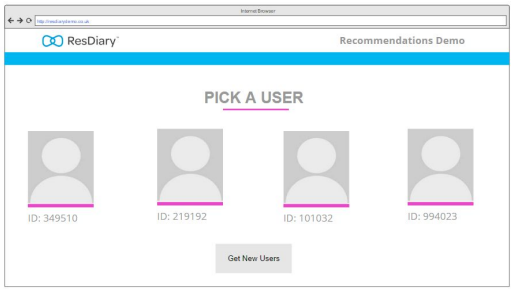
The issue arose of easily showing that the algorithm does in face make sensible recommendations. Asking the client to look at the number and claiming that “complex maths and papers” say that this result is valid just won’t cut it. There should be a visual representation that the algorithm is functioning correctly. Furthermore despite the recommendation itself being the crux of the project, a horrific mess of a demo that poorly showcases a well built system should also be viewed as a failure. The correct combination however of a well performing system and slick and smooth visual showpiece should result in a great demo and better project overall.

Thus the front end of the application shall undergo a design overhaul and strive to accomplish the following:

- Always keep the demonstration showcase in mind. There should be a logical flow and order. It should be clean, slick and transition smoothly.
- There should be a visual representation that the recommendations being made are sensible. It should be obvious to a non-technical person that the recommendations being made are valid and sensible.
- Additional statistics such as those shown previously on the back end of the application can be implemented.
- It should showcase the average user of the application rather than a preselected handful who has the best recommendations. It should be a new random pool of users to show how the application would function in a real-world setting.

1. User Selection



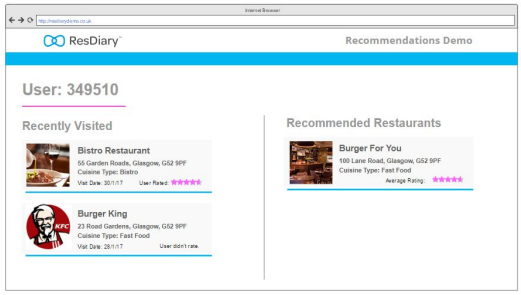
Several potential methods were proposed for selecting a random user / pool of users:

1. Generate a random pool of 5 to select from (with ability to refresh and generate additional users)
2. Load straight into a random user’s recommendation (screen 2 below)

Of the two the first was the chosen as the planned implementation, though the second method will also probably be implemented with a modified form of A/B testing potentially being done to see which flows better from a presentation perspective.

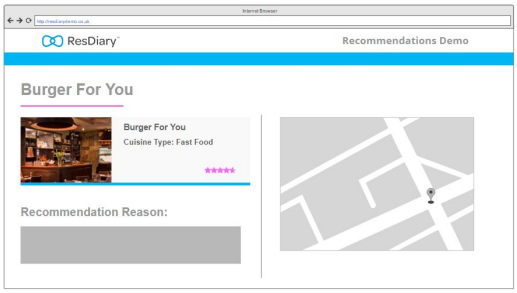
Another suggestion that was made was for rather than refresh the random user pool that a scroll wheel could be implemented to cycle through potential users to select from. This is an aesthetic difference which may be investigated if time permits.

2. User Recommendations Screen



Upon selecting a user it should then transition to a screen dedicated to that specific user's recommendations. One visual way of showing that the application is making sensible recommendations can be made here. By showing both the user's recently visited restaurants and their new recommendations there should be a clear, visual connection between the restaurants the user frequents and the recommendations being made. This also shows that new restaurants are being recommendation and not restraunts the user has recently visited. It shows visually that if a user has eaten at 60 Chinese restaurants in Glasgow that the system clearly isn't working if it's recommending a chip shop in Bristol.

3. Further Restaurant Information



Selecting a recommended restaurant then should provide additional information about the restaurant. The back end statistics can be located here so a technical "math & paper" justification is provided for a specific recommendation. Furthermore an additional visual recommendation reason could be provided via Google map integration. It was proposed to include a Google map view of where the restaurant is located similar to the ResDiary.com website. A further visual aid for recommendation reason would be to show where the user's recommendations are and where the user has previously eaten with markers on the map. This shows that proximity of location is a factor in the recommendation being made.