

ECS522U – Graphical User Interfaces (2021)

COURSEWORK ASSIGNMENT 3: EVALUATION

Group Members:

HARISON SUNDARAMOORTHY

HARDIK SACHDEVA

RONIK KAMLESH

EEMAN KANWAL

RIKHIL SHAH

Contents

1. Summary of Evaluation.....	2
2. Evaluation Process.....	3
3. Findings.....	4
4. Proposed Improvements.....	10
5. Member Contribution	13

1. Summary of Evaluation

We began by completing a comprehensive and thorough evaluation of the educational weather application we developed. To ensure the evaluation process was unbiased, each evaluator was given testing criteria (the 10 heuristics) and asked to identify problems based on each criterion. Each evaluator also rated the severity of each heuristic to quantify how severely each heuristic was being violated (if at all).

Some of the key heuristics which were being violated by the problems present in the application were to do with error prevention aspects of the application as well as the visibility of the status of the system. These were addressed by proposed improvements to the application which included having a dedicated loading screen and error message alerts – to ensure the user is kept informed about the processing of erroneous inputs.

The proposed improvements aimed to limit the number of heuristics (usability principles) which were being violated by the application. If all proposed improvements were to be implemented, the application would be even more user friendly and easier to use for the primary stakeholder.

2. Evaluation Process

1. Testing criteria:

We completed the evaluation of our application by checking all aspects against the 10 heuristic principles (usability principles). We determined if there were any problems which violated any of the usability principles and also attempted to locate areas of the application which lack the minimum threshold to be user-friendly. Along with the heuristic principles, the evaluators also conducted cognitive tests to be satisfied that it is convenient for every user to understand the interface.

2. Assign the task to the relevant evaluators:

After having the list of criteria, three group members (who had a good understanding of the evaluation process) were assigned to perform the evaluation on the application. These group members were: Rikhil, Harison and Hardik.

3. Providing the evaluators with instructions:

After being assigned the evaluation task, the evaluators discussed the standard criteria for evaluating the application so that they each had the same set of instructions while assessing the application – to ensure the evaluation process was unbiased.

4. Evaluation process

The evaluators started the process of evaluation by getting familiarised with the user interface followed by analysing every component of the application with the heuristic and other cognitive principles.

5. Noticing the problems

The evaluators made an individual list of the problems they experienced while browsing through the application. They arranged the results in the specific heuristic category depending upon the problems they faced.

6. Discussing the results

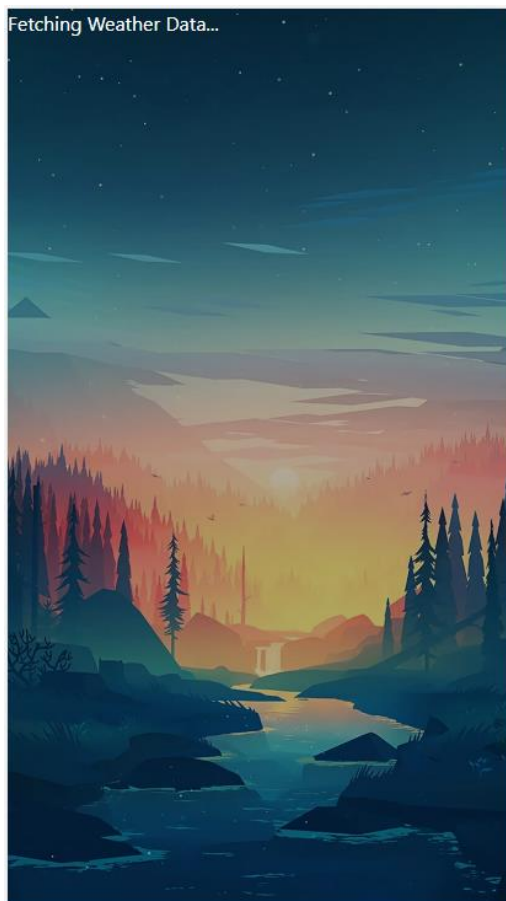
After the completion of the individual reports by the evaluators, the observations were compared and a conclusive report for the evaluation was generated.

3. Findings

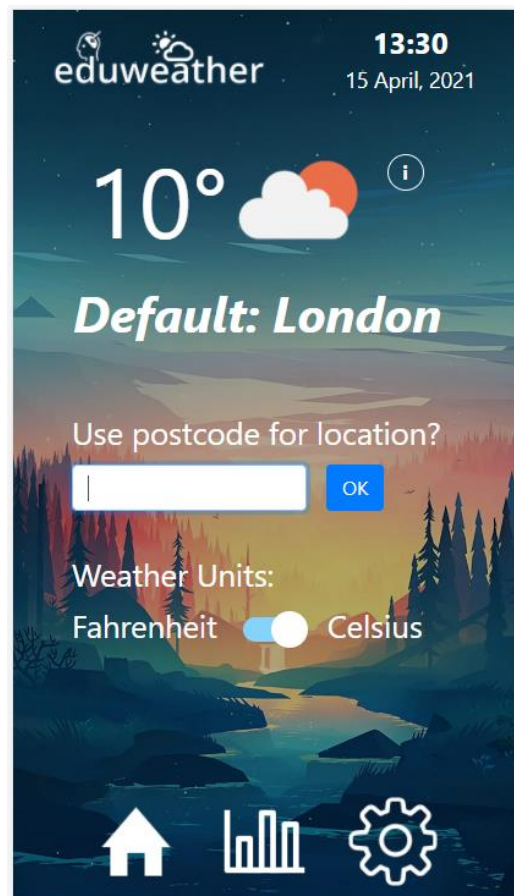
In this section, the problems discovered with the weather application will be analysed and combined with a heuristic evaluation. The heuristics violated by each discovered problem will be discussed and each heuristic will be rated based on a severity scale. The severity rating for each heuristic will be determined by the number of problems which violate it as well as their impact and persistence.

The evaluators who will determine the severity rating for each heuristic will be three evaluators from our group: Rikhil, Harison and Hardik. Each member will give an independent rating based on the analysis of each heuristic.

Problem 1 – Loading screens lack detail:



Screenshot 1



Screenshot 2

The loading message in Screenshot 1 is displayed whilst the application fetches weather data from the API. However, the same message is also displayed whilst the application waits for the user to either allow or block geolocation access. As a result, the message could be misleading to some users as they may think that weather data is being fetched when in actual fact, the application is waiting for their input.

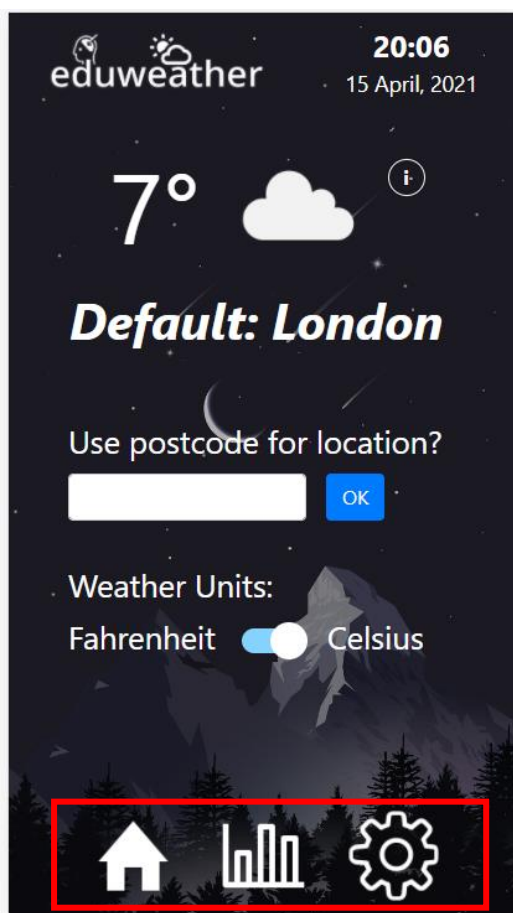
If the user blocks geolocation access, they can manually enter a postcode to state their location. However, when the OK button is clicked (in Screenshot 2), there is no feedback to the user to indicate that weather data was successfully fetched for the entered postcode. This means that when the fetch is unsuccessful, the user is kept uninformed.

The Visibility of System Status heuristic states that the user should be kept informed throughout the usage of the application. Our weather application does show a loading message when it is first launched (see Screenshot 1 below).

Problem 2 – Logo and time are unnecessary:

The logo and time, displayed at the top of every screen (see Screenshot 2), are unnecessary as they do not provide useful information to the user. Both take a significant portion of the already limited screen space. In addition, the weather application is made for smartphones which already displays date and time information in the notification bar.

Problem 3 – User may not know which screen they are on:

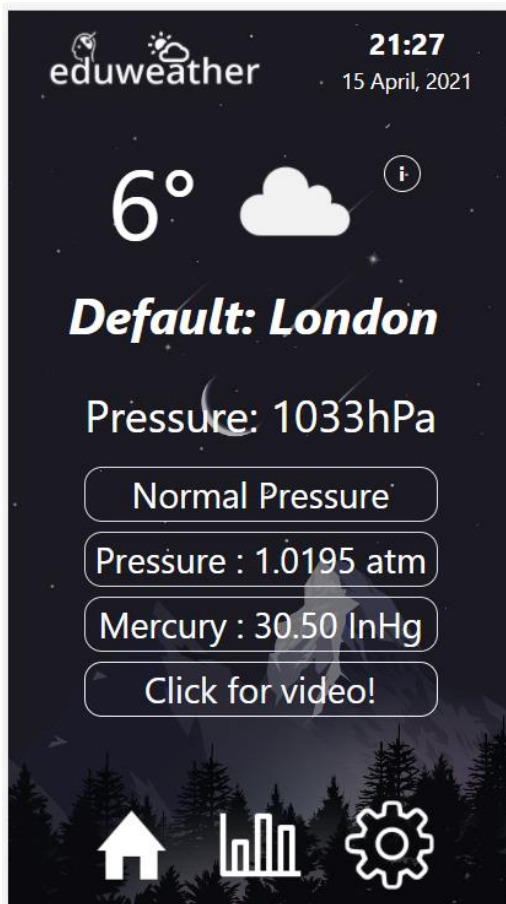


Screenshot 3

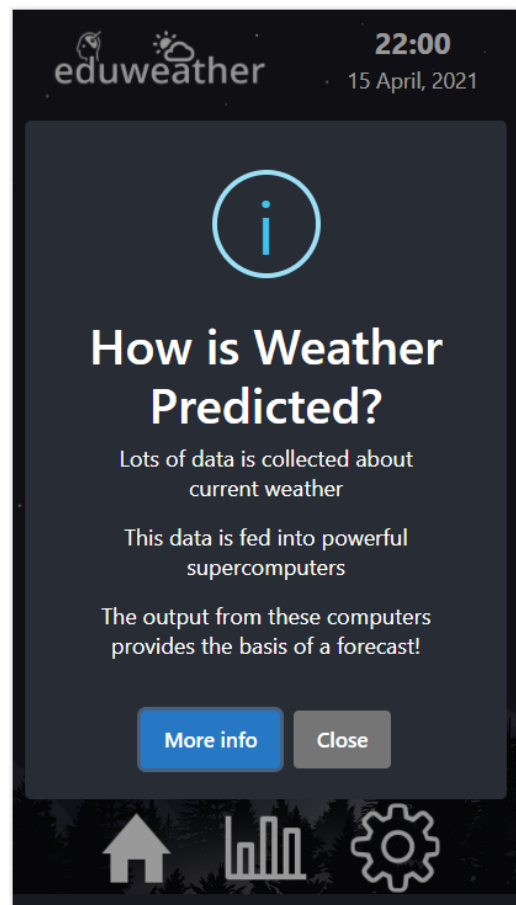
The weather application does not provide the user with information regarding which screen they are currently on.

The navigation bar (highlighted in red in Screenshot 3) is what is used to navigate between the different screens in the application. However, the application provides no feedback when a navbar button is clicked. This makes it difficult for the user to determine which screen they are on.

For example, in Screenshot 3, the user is on the Settings screen. This would be unclear to someone who has never used the application as the word 'Settings' is not displayed anywhere on the screen.

Problem 4 – Too much text:

Screenshot 4



Screenshot 5

The two screenshots above show how the application contains lots of text. Some users may find this overwhelming and would prefer to learn about weather conditions through pictures and videos. The large amount of text could lead to less user engagement with the application.

Problem 5 – User preferences are not saved:

When the user exits the application, their preferences are not saved. This means that when the user relaunches the application, they must spend time choosing their preferences again. An example of this occurs with the user's preferred weather units (Screenshot 2). If the user selects Fahrenheit as their preferred units for temperature, this is not saved. The user's location preferences are also not saved when the application is closed. This means that the user is prompted to allow geolocation access every time they launch the application – some users may find this bothersome.

Problem 6 – Lack of error messages:

Screenshot 6

The application lacks error messages in key areas. For example, when the OK button is clicked in Screenshot 6, nothing is displayed to the user.

The user should be presented with an error message as they have clearly entered an invalid postcode. Instead, the application does not display anything.

This means the user is kept uninformed about the processing status of their input. This could be frustrating for some users who may expect an output from their input – even if their input is erroneous.

Heuristic Evaluation:

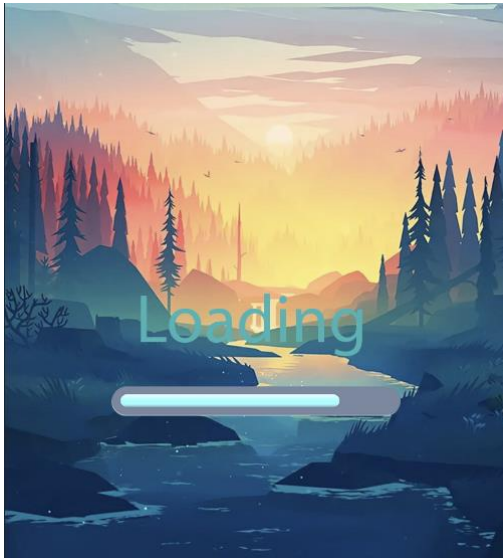
Heuristic Principle	Problems which violate the Heuristic	Severity Rating
Visibility of system status	<p>Problem 1 – user is kept uninformed when screens are loading. Therefore, system status is not always visible.</p> <p>Problem 3 – By not displaying the current screen, the current system status is also not clearly visible.</p>	<p>Rikhil – 3 Harison – 3 Hardik – 2</p>
Match between system and the real world	None of the problems identified violate this heuristic.	0 – Symbols used for different aspects of the application (e.g. settings button) are easily identifiable for the primary stakeholder.
User control and freedom	None of the problems identified violate this heuristic.	0 – All screens of the application are easily accessible from the home page and the home page is always accessible via the navbar at the bottom.
Consistency and standards	Problem 5 – Consistency is lacking as preferences are not saved when the application is closed.	<p>Rikhil – 1 Harison – 2 Hardik – 1</p>
Error prevention	Problem 6 – The lack of error messages mean that no <i>good</i> error messages are displayed by the application.	<p>Rikhil – 4 Harison – 3 Hardik – 3</p>
Recognition rather than recall	Problem 3 – The user should be able to immediately recognise which screen they are currently on. They should not have to remember.	<p>Rikhil – 3 Harison – 3 Hardik – 3</p>
Flexibility and efficiency of use	Problem 5 – If the user is able to save their preferences, their experience of using the application will be faster and more efficient.	<p>Rikhil – 1 Harison – 2 Hardik – 1</p>

Aesthetic and minimalist design	<p>Problem 2 – The logo and time are extraneous information which violates this usability principle (heuristic).</p> <p>Problem 4 – Large blocks of text about how weather is predicted and weather conditions are extraneous and overwhelming for users.</p>	<p>Rikhil – 3</p> <p>Harison – 3</p> <p>Hardik – 2</p>
Helps users recognize, diagnose, and recover from errors	None of the problems identified violate this heuristic.	0 – Currently, the application does not display error messages. However, when API requests fails, the application uses the last known previous location until the API request succeeds.
Help and documentation	None of the problems identified violate this heuristic.	0 – Documentation for the application is not required as the application is intuitive to use. However, the video produced in Assignment 2 could be a form of documentation if users require help.

4. Proposed Improvements

In this section, we will focus on proposing improvements for the problems which had the highest severity scores (from Section 3 of this report). This is because it is important to prioritise the improvements for the application which cause major usability problems for the primary stakeholder.

Problem 1 – Improvement:



Problem 1 – after

The first improvement we can make consists of improving the loading screen.

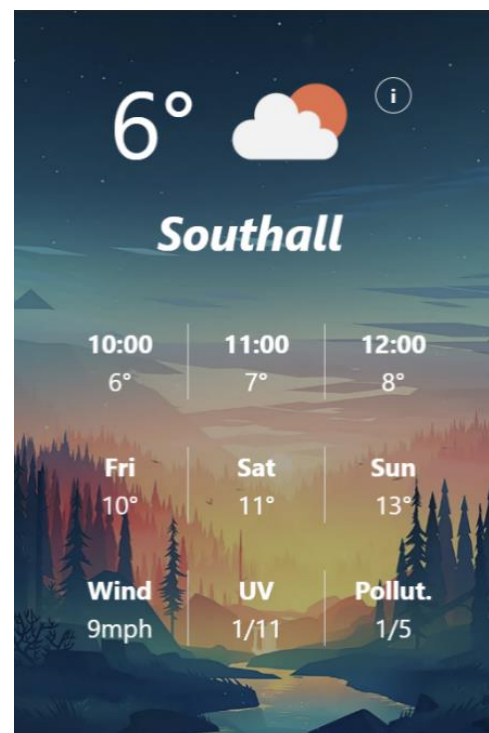
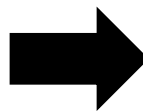
Currently the loading screen only says fetching weather data regardless of what the application is trying to load, which also goes against showing the user the current state.

We can improve this by having a generic loading screen that we can reuse for any part of the application which displayed the previous loading screen. This will make it more consistent and in-line with intuitive designs.

Problem 2 – Improvement:



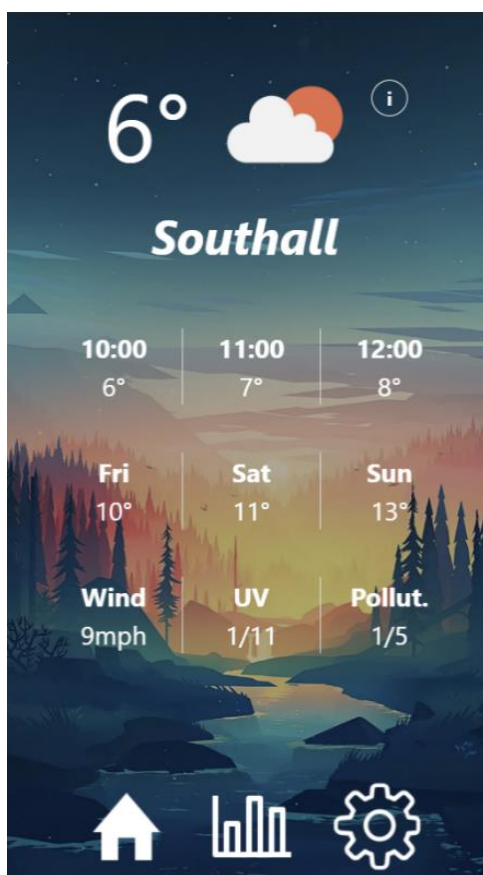
Problem 2 – before



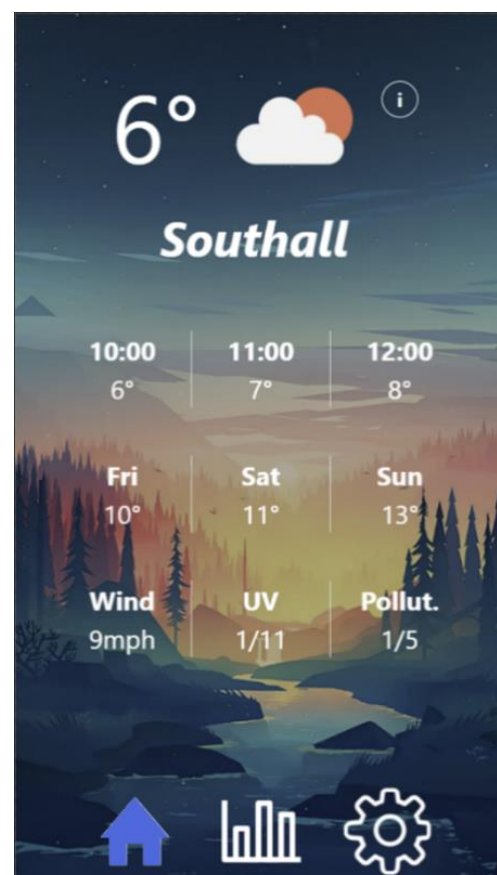
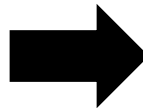
Problem 2 – after

The second improvement we can make is to our user interface based on the Aesthetic and Minimalist Design heuristic. On our homepage, we have unnecessary information that a student would not need – for example, the time and the name of the application at the top. Both take a significant portion of the screen – and screen space is already very limited. Also, assuming that this application will be used on a mobile device, the time should already be in their notification bar. Moreover, the date does not add any useful information for our primary stakeholder who are students. Therefore, the solution to this problem would be to remove the unnecessary information – the result of the improvement can be seen in the screenshots above.

Problem 3 – Improvement:

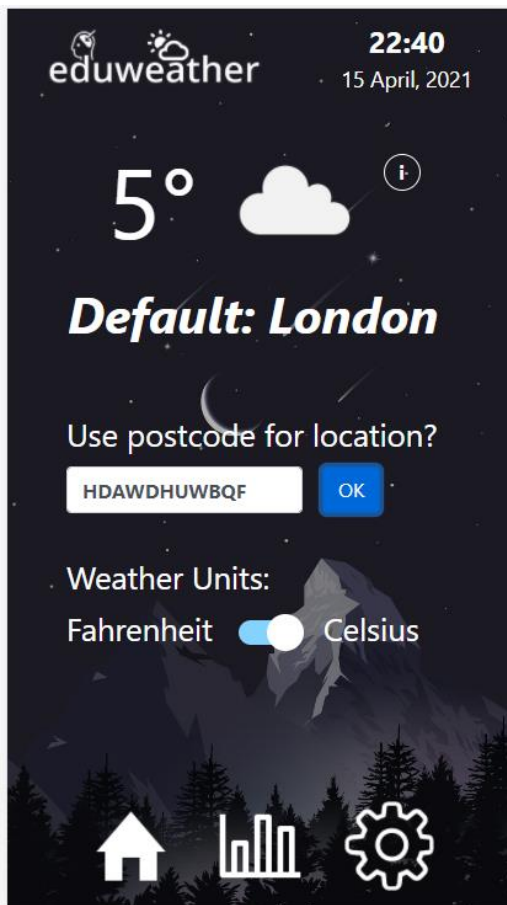
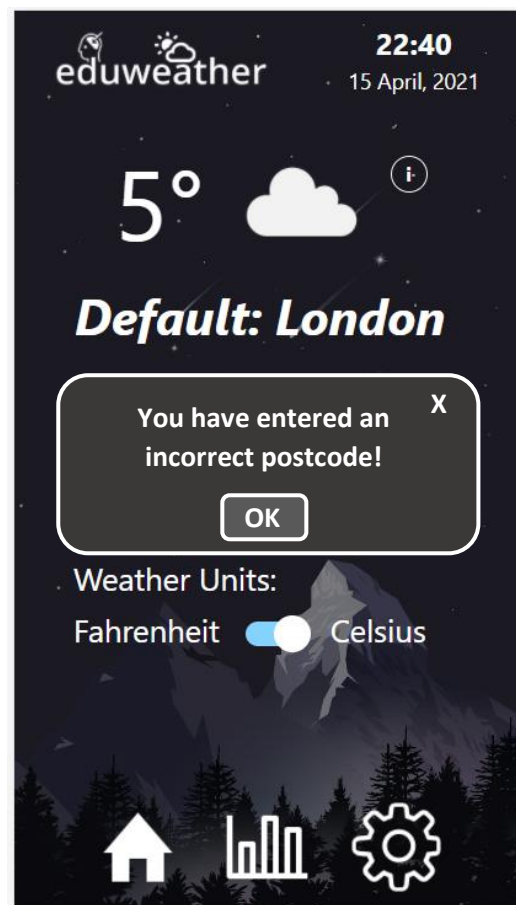
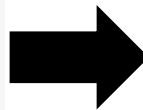


Problem 3 – before



Problem 3 – after

Another improvement we can make is based on the Visibility of System Status heuristic. While in the app, we currently, do not know what state we are in, for example in the home page or the records page. We can improve this by changing the colour of the button which belongs to the current page we are currently looking at. This would make navigating between the pages more intuitive for the primary stakeholder. They would also be able to immediately tell which page they are currently only by simply glancing at the navbar.

Problem 6 – Improvement:*Problem 6 – before**Problem 6 – after*

The improvement for Problem 6 involves displaying an alert message when the user enters an incorrect postcode. A similar alert message would be displayed in other areas of the application where error messages need to be displayed. This will keep the primary stakeholder informed as to whether their inputted data has been processed. Currently, they are kept uninformed when they enter an incorrect postcode or if the server is unable to locate their postcode – which is not beneficial. This improvement would mean that the Error Prevention heuristic is no longer violated as a useful error message is being displayed.

5. Member Contribution

Name	Tasks Completed
HARISON SUNDARAMOORTHY	Proposed improvements
HARDIK SACHDEVA	Evaluation process, Proof reading
RONIK KAMLESH	Summary of evaluation
EEMAN KANWAL	Proof reading
RIKHIL SHAH	Findings, Proposed improvements, Proof reading