Recipes For Life
Appendix 6.4
Nielsen's 10 Usability Heuristics Usability Test

Heuristic	Description	Application
Visibility of system status	The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.	The application keeps the user informed by telling the user if the actions are completed successfully e.g. if an issue occurred editing a recipe or if there is no internet connection a message will appear telling the user this. If an edit text box is empty or does not meet the criteria the system will inform the user of the actions that need to be taken before they can proceed.
Match between system and the real world	The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.	The application mainly uses icons that follows application conventions for icons such as a plus sign for add and a dustbin for delete. Occasionally a symbol like the font symbols did not represent the real world as much as the student would have liked but this was due to the icons available with the right licenses for use.
User control and freedom	Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.	When the user is in the application and going through a process like adding a new recipe each dialog in this step offers the ability for the user to close the dialog quickly. Unfortunately the student did not look into undo/redo but this is something to explore in the future.
Consistency and standards	Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.	Icons and phrases maintain the same throughout the application to keep consistency. They are common icons and phrases that meet common cultural conventions. Icons and phrases do not change meaning throughout the app and so maintain consistency.
Error prevention	Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.	The application has lots of error checking in place for example it checks that textboxes which should not be empty aren't empty and tells the user to fill the text box in before accepting the information or checking the password meets the correct criteria before accepting. The user is also given confirmation dialogs before committing serious actions like deleting a recipe or cookbook or updating a recipe.
Recognition rather than recall	Minimize the user's memory load by making objects, actions, and options visible. The user should	All options are made visible in the application page and are easy to access. No option is inside another

	not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.	option so it should be straightforward to see the option and access it.
Flexibility and efficiency of use	Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.	There wasn't really a need for accelerators in the application. Recent search suggestions were stored to speed up users recent searches. In the future it might be good to have an accelerator for frequent users to easily access frequent ingredients or method steps that they use in recipes.
Aesthetic and minimalist design	Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.	The application is as straight forward as possible with no irrelevant information in the dialogues.
Help users recognize, diagnose, and recover from errors	Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.	Error messages are all in plain language outlining the issues. Although the student believes they could suggest more constructive solutions and will explore this in the future.
Help and documentation	Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.	A user guide was created to help the user use the application although this is just a word document. The student would like to have a help guide in the application or a video in the application showing the user how to use the app. This is something to explore in the future.