HEURISTIC EVALUATION

Heuristics listed are the “classic” 10 usability heuristics developed by the Nielsen Norman Group.

HEURISTIC	DIFFICULTIES	OPPORTUNITIES
Visibility of system status The system should always show the status of an on-going operation to the users until it is done. So the user will get a clear understanding of the progress of that particular process/activity.	No status of the page which the user is currently in, Just the information of the page	We use the heading to serve the status of the page which the user is currently in.
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 Icons we’ve used are not representative of how we want to perceive them in the real world.	● Titles are present for each icon in a page ● Natural words are used to describe the titles for each page. ➢ *Expiring food list* ➢ *Wasted food list*
User control and freedom Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave	Lack of button to delete the food items, which are entered into the expiring food list page. There’s no support for undo and redo.	While adding an item into the checklist we can provide an undo option which gives the users freedom to make changes

the unwanted state without having to go through an extended dialogue. Support undo and redo.		as their wish.
Consistency and standards Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.	➢ *Food wasted page* ➢ *Food expiring page*	Users cannot edit the “food wasted” page as it just shows the list of expired foods. But the “food expiring” page shows the users the days left for the expiry of food.
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.	Selecting expiry dates for the food	When the user mistakenly selects the expiry date before the current day, We can make sure to pop up an error message.
Recognition rather than recall Minimize the user's memory load by making objects, actions, and options visible. The user should 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.	➢ Foods entered recently are not shown. ➢ Everytime a food expires a certain date . User needs to recall the item name to register it again.	We can try storing the foods most commonly used by the user . So he may not have to recall them.

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.	Recommending the similar types of foods items does not exist.	We can implement a recommendation system for the food items which are entered by the user.
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.	Color palette doesn't go well with the background of the app.	We can modify the colors which we have used to go well with the background. Make the app look neat with a good design palette.
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.	There is no error message if we select the dates before the current day.	Can implement a pop up message to the user saying invalid operation.
Help and documentation Even though it is better if the system can be used without documentation, it may be necessary to provide help and	No guide for usage of the app	Can add : 1) Help and Documentation page

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.		2) FAQ’s 3) Contact