Heuristic Evaluation

A heuristic is a simple principle or “rule of thumb”. A heuristic evaluation is one where an evaluator or group of evaluators walks through the design of an interface and decides whether or not it complies with these “rules of thumb”. It provides good insight into possible usability problems that might damage the user experience.

We have evaluated our model according to Jakob Nielsen’s 10 Usability Heuristics for User Interface Design as follows:

Heuristic	**Challenges**	**Measures**
Visibility of the system’s status Users should be given feedback on what is happening within a reasonable timescale.	The application should constantly update itself with the changes made	A plant growing animation to indicate progress and A worn out plant to indicate fallbacks/overdues/decline
Match between the system and real world Information should be presented in a means familiar to the user including language and conventions rather than terms developed for the system. Information should be presented in a logical order	To make it seem like a user-friendly app instead of a mere calendar	Interactive design with short phrases and buzzwords to make it more friendly and easy to navigate
User control and freedom Users make mistakes. There should be an “emergency exit” which is easy to find and exit the current system state without having to jump through hoops. Undo and redo functions are essential.	Already scheduled tasks deletions/changes updation	Undo/Redo mechanism provided to delete/modify tasks which will be updated in the calendar
Consistency and standards. Words, actions, situations, etc. should always mean the same thing and users should be able to understand that.	Making reuse of words look non repetitive	Actions like Task submitted/Completed conveyed uniquely and in informal terms i.e. to make it more interactive
Error prevention Preventing error is better than clear error messages. Eliminate error conditions or make users aware that they may be about to occur and ask them if they want to proceed.	Evaluating Quizzes in the assessment widget.	Quizzes are made based on the structured notes provided and are evaluated. Each answer asks for confirmation before submission.

Recognition rather than recall Reduce the load on a user's memory. Make actions, options, and objects visible. Users shouldn't have to remember things from one screen to the next. Instructions should be easy to access when needed.	The sole purpose of this app is to eliminate the reliability on our memory for trivial tasks	Constant alerts, alarms and notification for respective tasks scheduled
Flexibility and efficiency of use the use of accelerators, where appropriate, may be invisible to new users but improve the efficiency of use for experienced users. Actions could be customized by users.	Implementing easier ways of providing inputs	Drag and drop mechanism to provide ease of giving inputs.
Aesthetics and minimalist design Dialogue should not give irrelevant or rarely useful information. The more data in dialogue the more it diminishes the overall visibility of individual points for the user.	To implement a unique yet simple and attractive design	Simple colour co-ordination with animations and widgets such as quote of the day, etc.
Help users recognize, diagnose and recover from error This means error messages should be in clear language and avoid the use of codes. They should explain the problem and offer a solution to that problem.	Failure of the app in case of a breakdown	If the app is down, the dashboard is displayed after an error message
Help and documentation The best approach is to construct a system which needs no help or documentation but if it is necessary – it should be easy to search, be based on the tasks the user wishes to execute, offer concrete steps to follow and be kept to a minimum.	To enable the user to understand how to navigate through the app	A "genie guide" option provided to showcase how to navigate thru the application and its usage