Design and Implementation of Human – Computer Interfaces
Prof. Dr. Samit Bhattacharya
Department of Computer Science & Engineering
Indian Institute of Technology, Guwahati

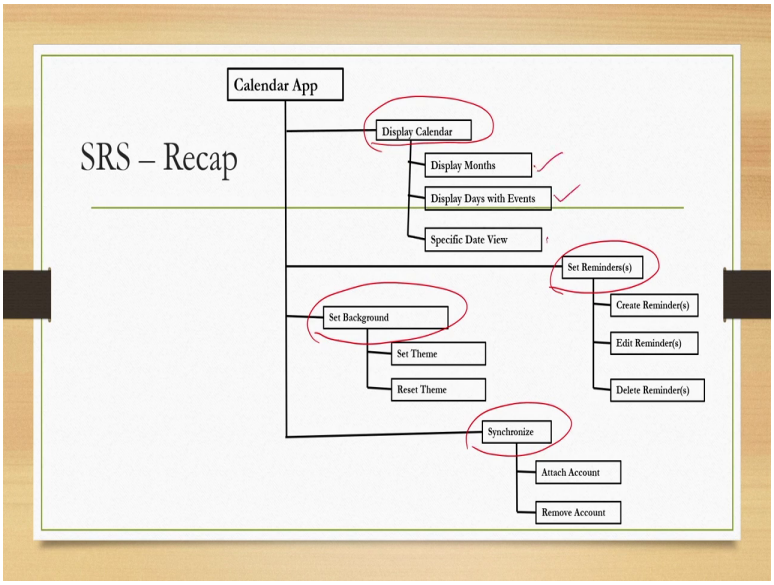
Lecture: 19
Case Study on Prototype Evaluation - II

Hello and welcome to the NPTEL MOOC's course on design and implementation of human computer interfaces. We will continue our discussion on Prototype building and evaluation that is lecture number 17. So, in the previous lecture we have discussed about how to build a prototype and how to get it evaluated with cognitive walkthrough. So, there we talked about one case that is building a calendar application or calendar app.

(Refer Slide Time: 01:14)

So, just to recap we are interested in building a calendar app which is primarily meant to be used by students to manage their academic activities.

(Refer Slide Time: 01:23)

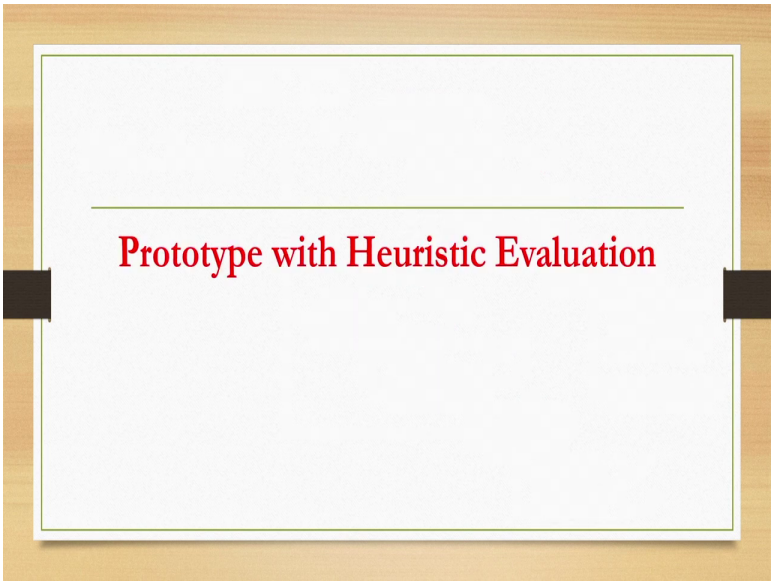


And we have already seen the different functions that have been identified as part of requirement gathering phase which are to be supported by the app. Now the functional hierarchies shown here so, as you can see there are 4 top level functions display calendar, set reminders, set background and synchronize and there are some functions like display month, display days display specific date.

Then create reminders, edit reminders, delete reminders, set theme, reset theme and attach account remove account among them in our previous lecture we talked about ah creating a prototype for supporting the 3 top level functions set background set reminders and Display calendar. Same thing we will do here we will talk about building a prototype for this primarily these 3 top level functions as well as getting it evaluated.

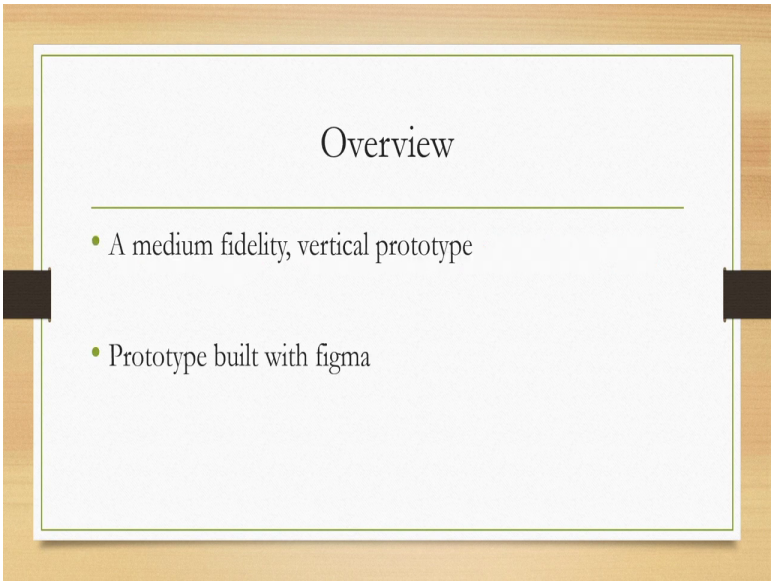
However this time we will talk about evaluation using a different evaluation method that is heuristic evaluation.

(Refer Slide Time: 02:41)



So, the main concern here is building a prototype that supports the functions which are part of the functional hierarchy and getting it evaluated with experts using heuristic evaluation method.

(Refer Slide Time: 02:55)



So, the Prototype that we are going to talk about is a medium fidelity vertical prototype however vertical prototype as you can recollect is not mandatory. So, primarily we will be focusing on the different interfaces that are part of different stages of the Prototype and the Prototype has been built with the figma tool which we use to build our earlier prototype as well as discussed in the previous lecture.

(Refer Slide Time: 03:31)

Now the Prototype has this month view specific date view, year view, background view where background can be changed dark background custom background then other views like deadline views timetable view scheduled of a day that view. So, all these things are supported which are results of the analysis of the functional hierarchy that we have just seen.

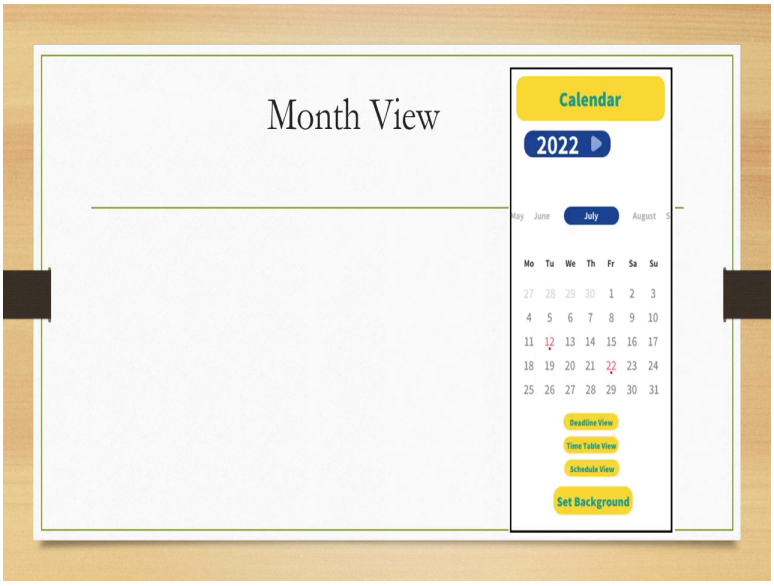
(Refer Slide Time: 04:05)

Let us see the views although this prototype is designed as a vertical prototype to support screen transitions. We can think of it as horizontal prototype where we will focus primarily only on the interfaces for specific views and then we will evaluate it using heuristic evaluation. Let us start

with the year view. So, if we want to get a year view we can see such an interface where the year is mentioned here.

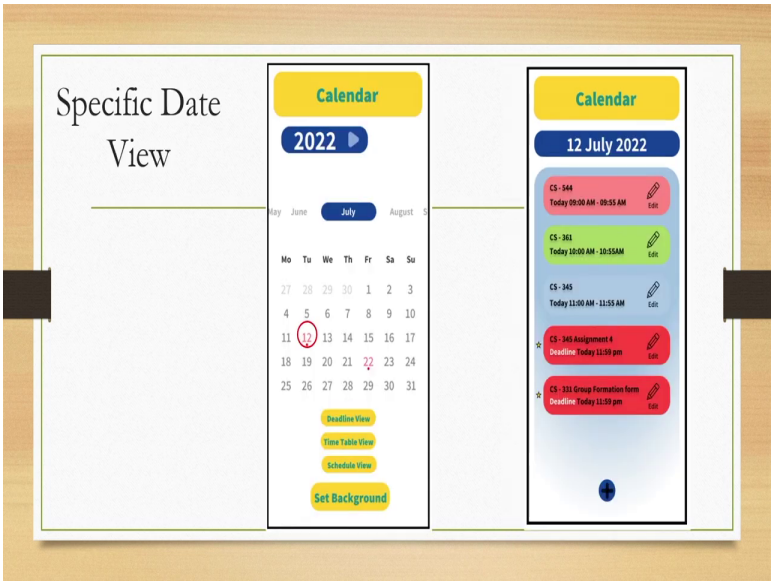
And then the months are mentioned on the top and if I select a specific month the current month the days are shown here and the year can be changed. To change the year this right hand interface appears where we can change the year.

(Refer Slide Time: 05:03)



The month view as we have just mentioned in the year view itself the month view is available this is for the current month it can be for other months as well. So, we have to select the month from the top menu.

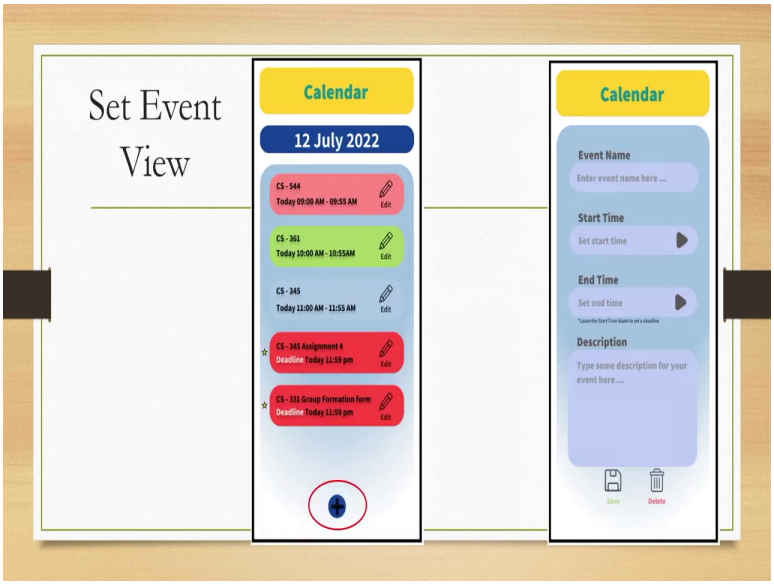
(Refer Slide Time: 05:21)



In this month view we can also have specific dates and if we choose a date like here then for this date we get to see the events that are scheduled on that day likes shown here in this type of interface. So, for 12th July that specific date in the year 2022 these are the events that are scheduled CS544 a course CS361 another course 3 4 5 another course assignment submission some other activities.

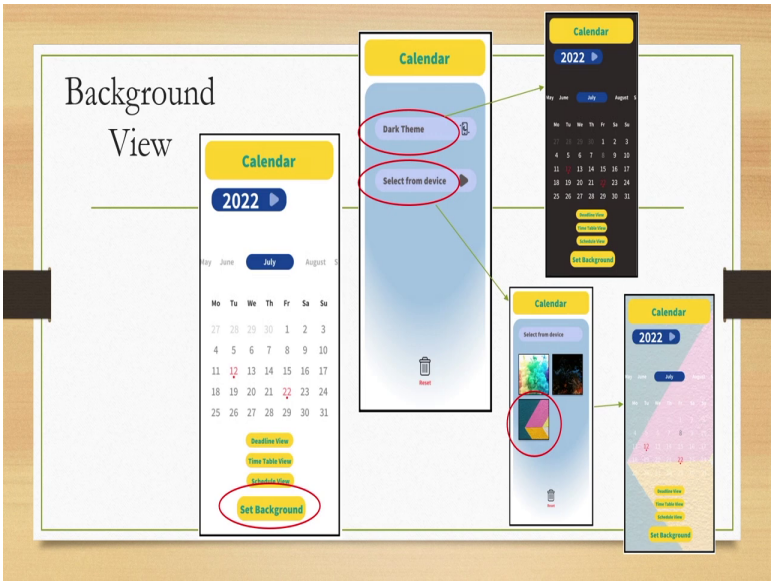
So, these are part of that activities and events that can be viewed if we select a date. So, that is the interface for specific date.

(Refer Slide Time: 06:17)



Then we can set events for a date. So, if we choose a date then this interface appears and below this interface there is this button and if we select this button then the other interface on the right hand side appears where we can mention the event name start time of the event end time of the event and some brief description which is optional. And as you can see below we can save it save this event or we can delete also if we think that that is not necessary to save.

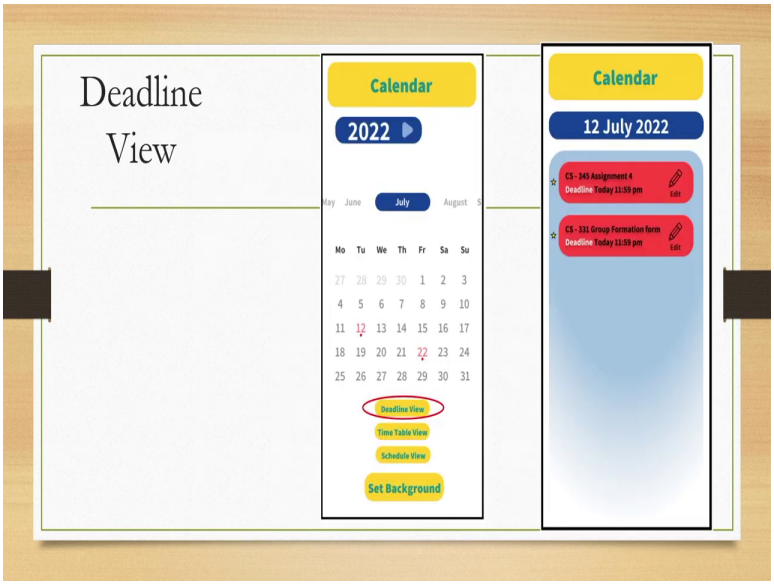
(Refer Slide Time: 06:54)



Then comes the background view. In this view we start with the year view where we get to see the days of the year and below that interface there is this set background option. So, once we select it the next interface appears in the middle where we can have either dark theme or some other theme that we can select from the device. If we choose the dark theme then that is a pre stored theme and this top interface appears with the background changed to the dark colour.

If we make the other selection that is select from device then some other background themes may appear depending on what are stored in the device and we can choose from here. Like if we select this particular theme then this next interface shows the theme applied on the year view as shown in this interface.

(Refer Slide Time: 08:10)

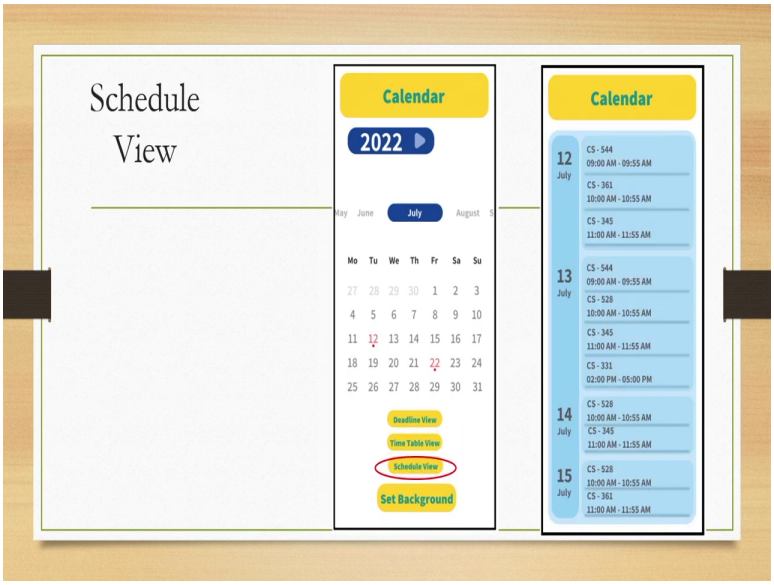


For deadline view we have this deadline view option here once it is selected it shows all the deadlines for a specific date which can be chosen from here.

(Refer Slide Time: 08:28)

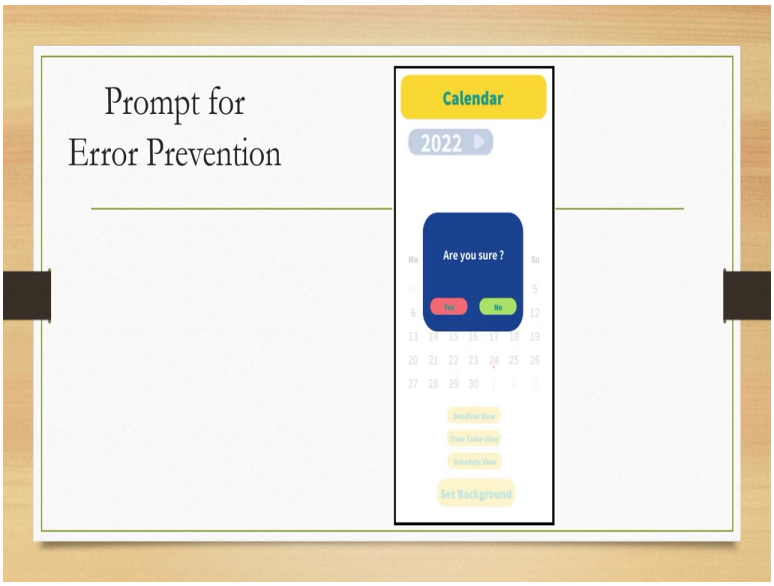
Similarly to see the timetable we have this time table view for a specific date that we can select from here and the timetable for the day date is shown in this interface on the right hand side. Here you can see additional information along with the timetable.

(Refer Slide Time: 08:51)



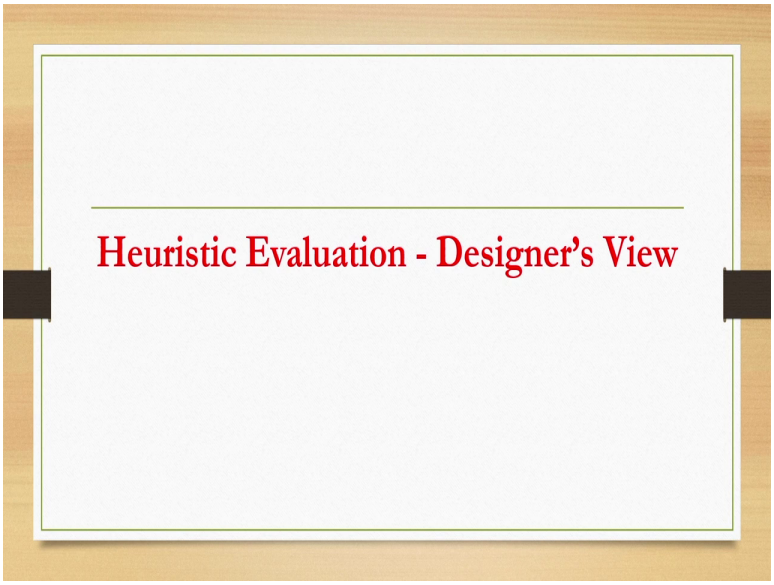
Another option present there is schedule view. So, this view is meant for schedule of the month like 12 July 13 July 14th July. So, what are the schedules it will show you the schedule for the month in this type of interface where it can be scrolled up or down.

(Refer Slide Time: 09:19)



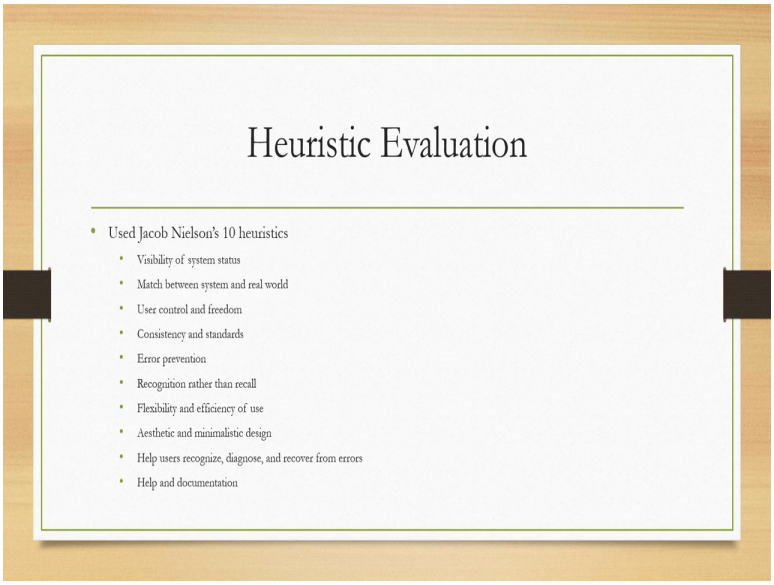
Another feature of the interface although that is not exactly part of the functional requirement hierarchy. Functional hierarchy is a dialog box that is added to prevent errors. So, if there is something needs to be saved some changes have been made or you want to close the app before that this type of confirmatory messages appear where you can confirm or decline to confirm the changes. So, that is all about the prototype.

(Refer Slide Time: 10:07)



Now let us evaluate this prototype with heuristic evaluation.

(Refer Slide Time: 10:20)

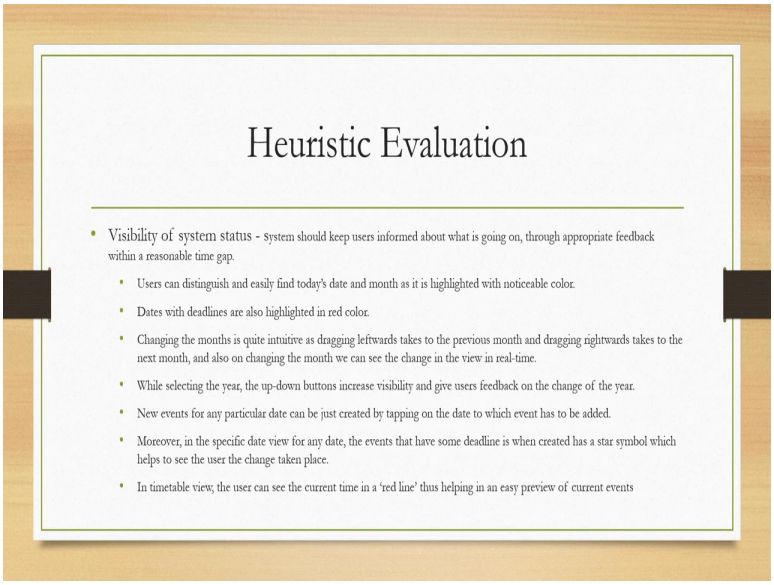


So, in order to perform heuristic evaluation we will follow the 10 heuristics by Nielsen just to recap. So, these 10 heuristics are visibility of system status, match between the system and the real world, user control and freedom, consistency and standards, error prevention, recognition rather than recall, flexibility and efficiency of use, aesthetic and minimalist design help users recognize diagnose and recover from errors help and documentation.

So, these are the things that are expected to be there on the interface the system status should be visible to the user there should be a match between the system and the real world user should have control and freedom for operations. The system should follow consistency and standards it should support error prevention mechanisms, it should support recognition rather than recall it should provide the user flexibility and efficiency of use.

It should have aesthetic and minimalist design, it should help users recognize diagnose and recover from errors and finally it should have help and documentation support. These are 10 things that needs to be checked for ensuring that the interface is usable.

(Refer Slide Time: 11:50)



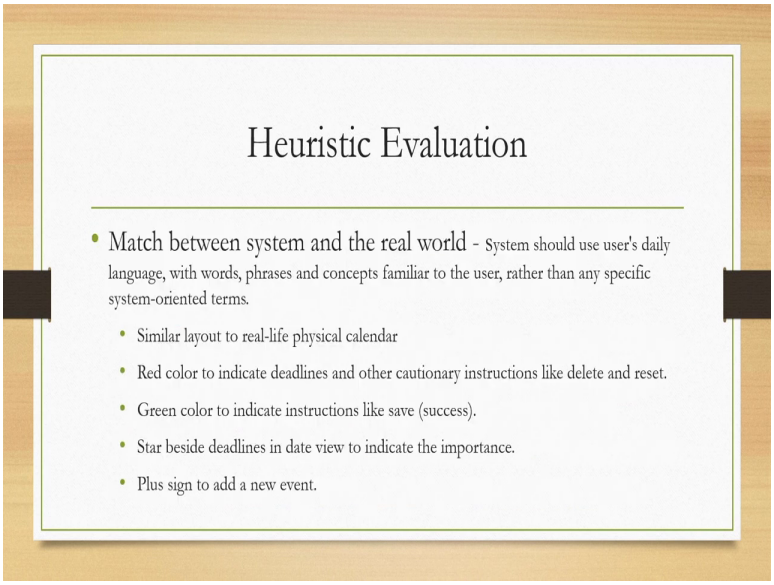
Now from the designer's point of view let us see what can be the view with respect to these 10 heuristics. First is visibility of system status the system should keep users informed about what is going on through appropriate feedback within a reasonable time gap. According to the designer the users in this system can distinguish and easily find today's date and month as it is highlighted with noticeable colour dates with deadlines are also highlighted in red colour.

Changing the months is quite intuitive as dragging left towards text to the previous month and dragging right towards text to the next month and also on changing the month we can see the change in the view in real time. While selecting the year the up down buttons increase visibility

and give users feedback on the change of the year. New, events for any particular date can be just created by tapping on the; date to which event has to be added.

Moreover in the specific date view for any date the events that have some deadline is when created has a star symbol which helps to see the user the change taking place. In timetable view the user can see the current time in a red line thus helping in an easy preview of current status. So, these are what the designer thinks about conforming to the first heuristic that is visibility of system status.

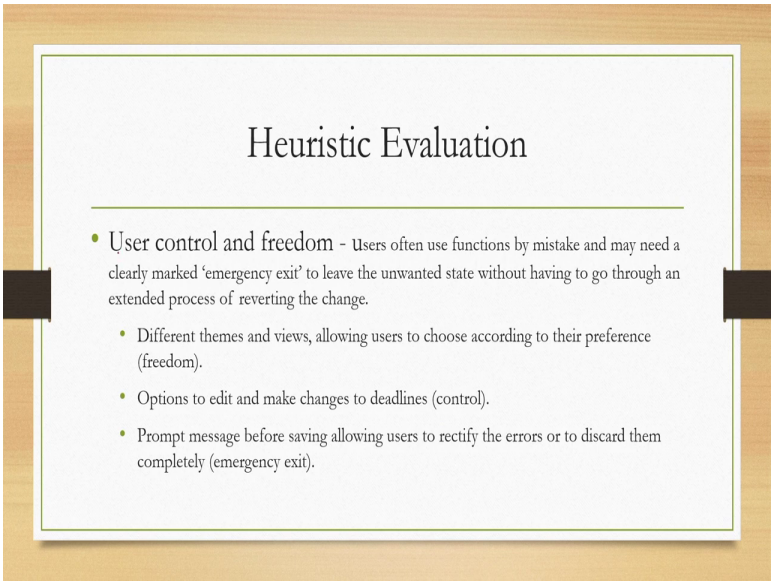
(Refer Slide Time: 13:27)



Next heuristic is match between system and the real world. What it means is that the system should use users Daily Language with words phrases and Concepts familiar to the user rather than any specific system oriented terms and what the designer thinks about this particular design similar layout to real life physical calendar red colour to indicate deadlines and other cautionary instructions like delete and reset.

Green colour to indicate instruction like save for success a star beside deadlines in dead view to indicate the importance plus sign to add a new event. So, these are the things that are added to ensure that there is a match between system and real world as per the designer.

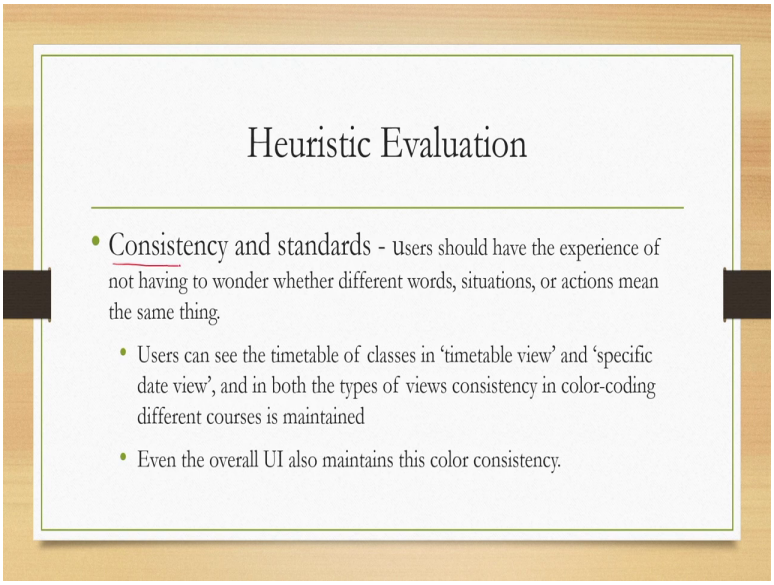
(Refer Slide Time: 14:15)



Next is user control and freedom. So, this is the third heuristic users often use functions by mistake and may need a clearly marked emergency exit to leave The unwanted state without having to go through an extended process of rewriting the change that is what is required to ensure this third heuristic is satisfied. So, what the designer thinks about the design with respect to this third heuristic different themes and views are there in the system allowing user to choose according to their preference. So, this implies freedom.

Option to edit and make changes to deadlines implies control. Prompt message before saving allowing users to rectify the errors or to discard them completely emergency exit. So, according to the designer the design supports this third heuristic as well.

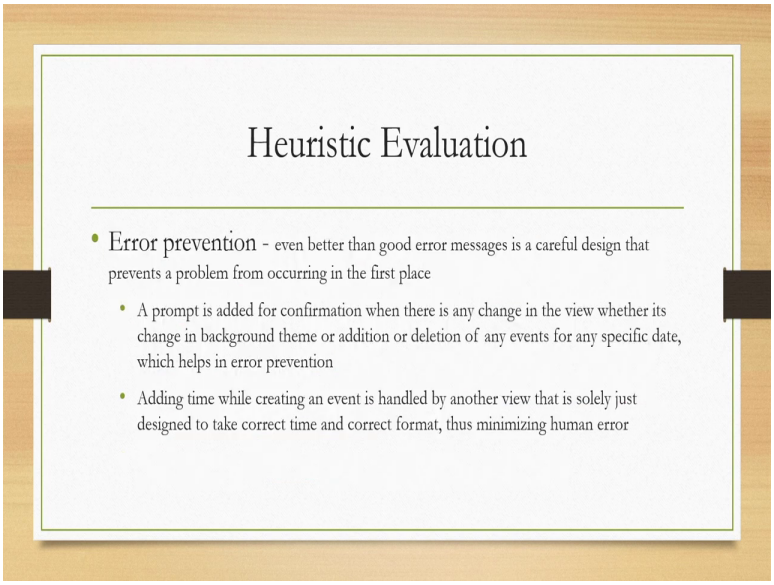
(Refer Slide Time: 15:17)



The 4th heuristic is consistency and standards what it says is that users should have the experience of not having to wonder whether different word situations or actions mean the same thing or different things. In the design for this calendar app what the designer thinks with respect to the 4th heuristic. In the design users can see the timetable of classes in timetable view and specific date View and in both the types of views consistency in colour coding for different courses is maintained that means it supports consistency.

Even the overall UI also maintains this colour consistency. So, consistency is supported however nothing is mentioned about following the standard.

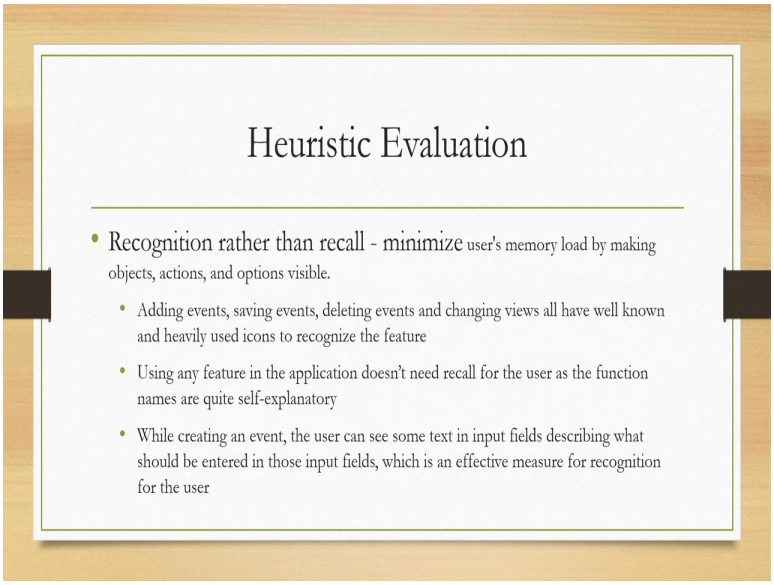
(Refer Slide Time: 16:03)



Error prevention is the fifth heuristic, even better than good error messages is a careful design that prevents a problem from occurring in the first place. What the designer thinks a prompt is added for confirmation when there is any change in the view whether it is change in background theme or addition or deletion of any events for any specific date which helps in error prevention. So, designer thinks that the prompt feature or the dialog box that we have seen is supporting this heuristic.

Adding time while creating an event is handled by another view that is solely just designed to take correct time and correct format thus minimizing human error. Again the designer thinks because of the particular design error prevention is possible and this particular heuristic is satisfied.

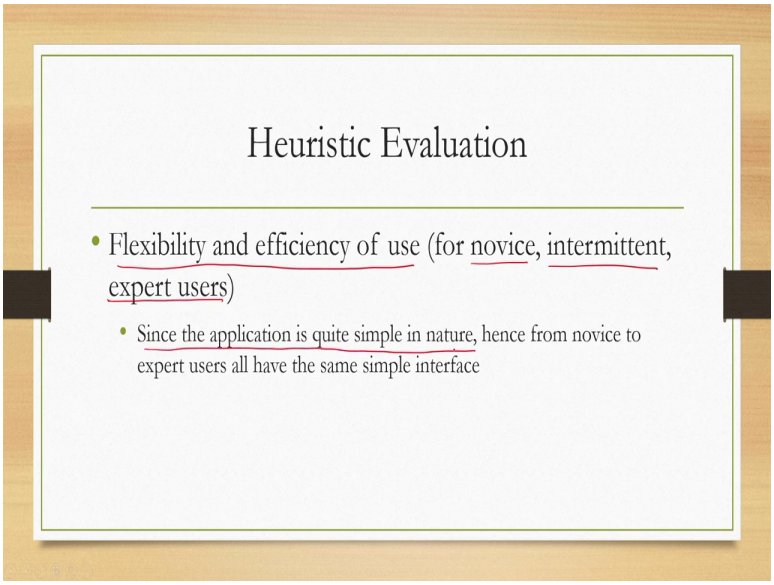
(Refer Slide Time: 17:06)



Next heuristic is recognition rather than recall which implies that the design should minimize users memory load by making objects action and options visible. In the design adding events saving events deleting events and changing views all have well known and heavily used icons to recognize the feature according to the designer. Using any feature in the application does not need recall for the user as the function names are quite self explanatory again according to the designer.

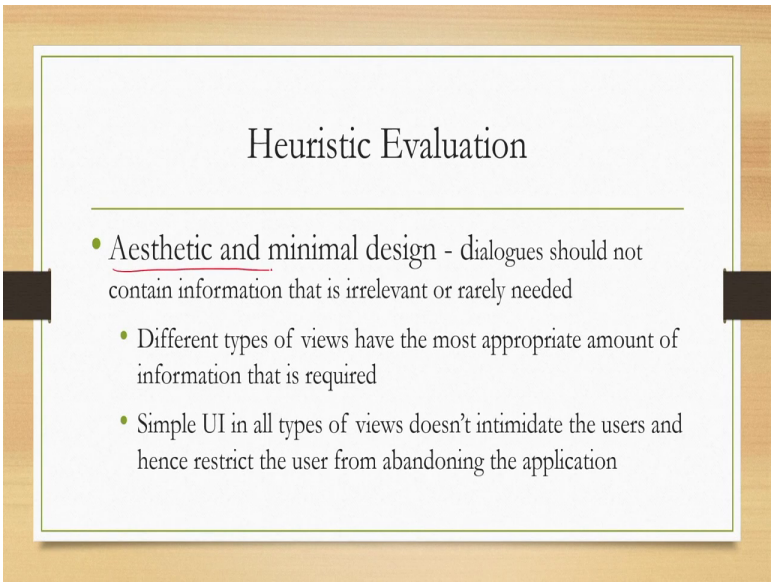
While creating an event the user can see some text in the input fields describing what should be entered in those input fields which according to the designer is an effective measure for recognition for the user. So, the designer thinks because of this particular way of Designing the design supports recognition rather than recall.

(Refer Slide Time: 18:18)



Next is flexibility and efficiency of use for different user categories namely novice intermittent and expert users. In this design since the application is quite simple in nature as perceived by the designer hence from novice to expert users all have the same simple interface according to the designer. So, designer thinks the interface is quite simple. So, it does not matter whether the user is novice or expert all will be able to use it. So, it provides flexibility and efficiency of use.

(Refer Slide Time: 18:47)

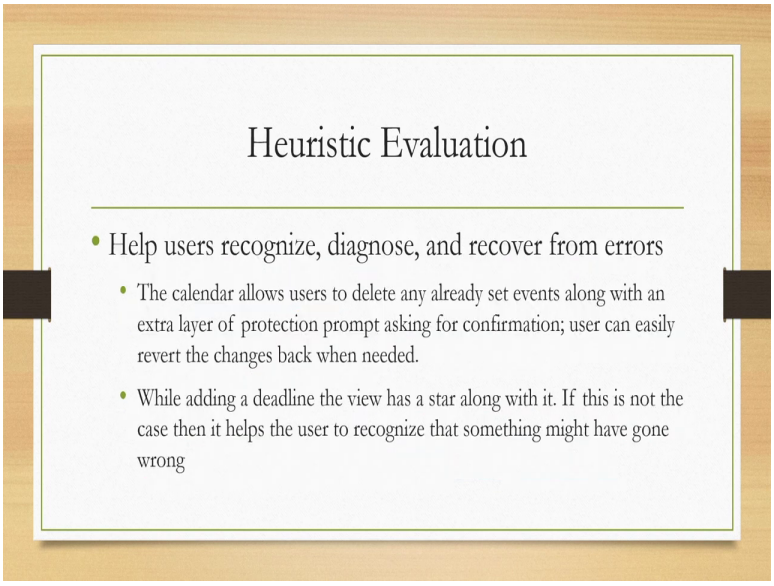


Next is aesthetic and minimalist design which states that dialogues should not contain information that is irrelevant or rarely needed. Different types of views have the most appropriate amount of information that is required all these qualifications are made by the designer that

whether the information is most appropriate or not again simple UI in all types of views does not intimidate the users and hence restrict the users from abandoning the application.

So, this is what the designer thinks that the amount of information is appropriate not intimidating and does not lead to rejection of the application.

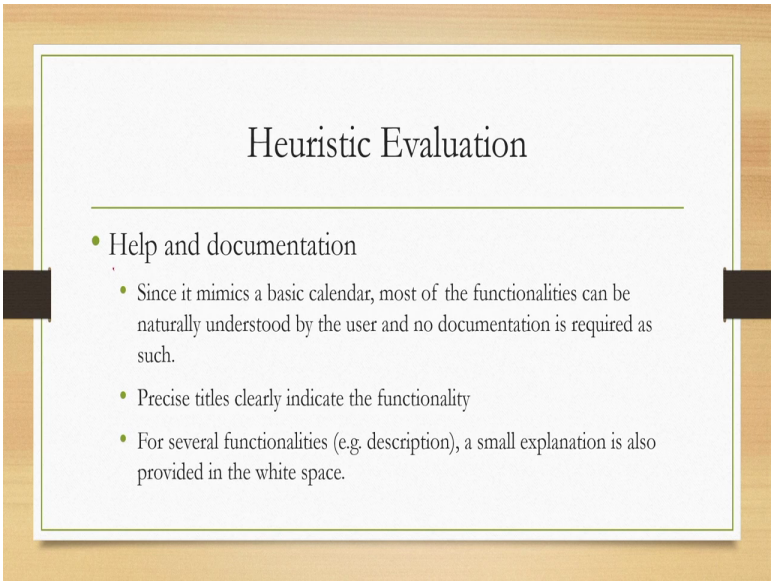
(Refer Slide Time: 19:33)



By the user next heuristic is the design should help the users recognize diagnose and recover from errors. In the design the calendar allows users to delete any already set events along with an extra layer of protection prompt asking for confirmation. User can easily revert, the changes back when needed. The designer feels so, while adding a deadline The View has a star along with it if this is not the case then it helps the user to recognize that something might have gone wrong.

So, again there is some indication regarding erroneous situations and the designer thinks all these design choices made the system help users recognize diagnose and recover from errors.

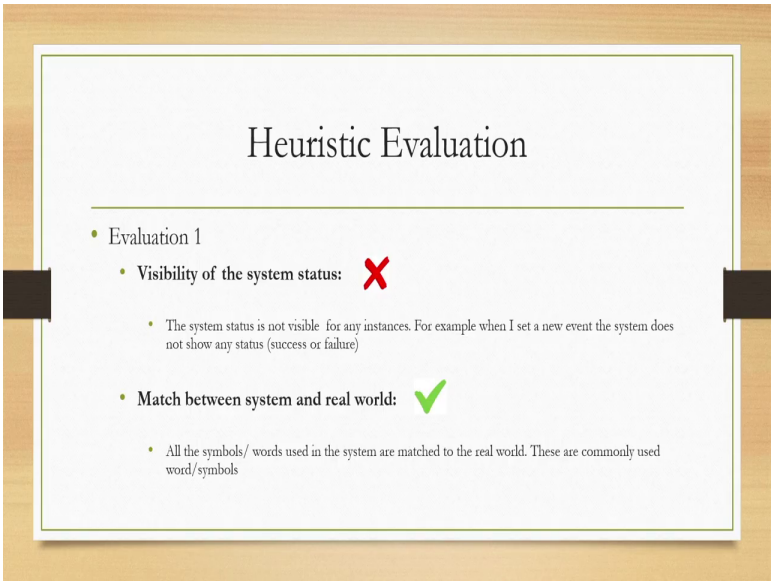
(Refer Slide Time: 20:27)



Finally help and documentation support for the design since the calendar mimics a basic calendar most of the functionalities can be naturally understood by the user and no documentation is required as such which the designer feels. In other words there is no documentation provided in the design and the designer feels that is not required. Precise titles clearly indicate the functionality.

So, the titles itself can be treated as documentation as per the designer and that those titles can be used to help the users use the system. For several functionalities a small explanation is also provided in the white space. So, that is not there for all the functionalities but for several functionalities. So, that is one sort of documentation according to the designer. So, in summary designer thinks that there are several design decisions made which indicates that the 10 heuristics by Nielsen are supported in the design which in effect implies that the design is usable. Now let us see whether that is indeed true with respect to some independent evaluation by experts.

(Refer Slide Time: 21:47)



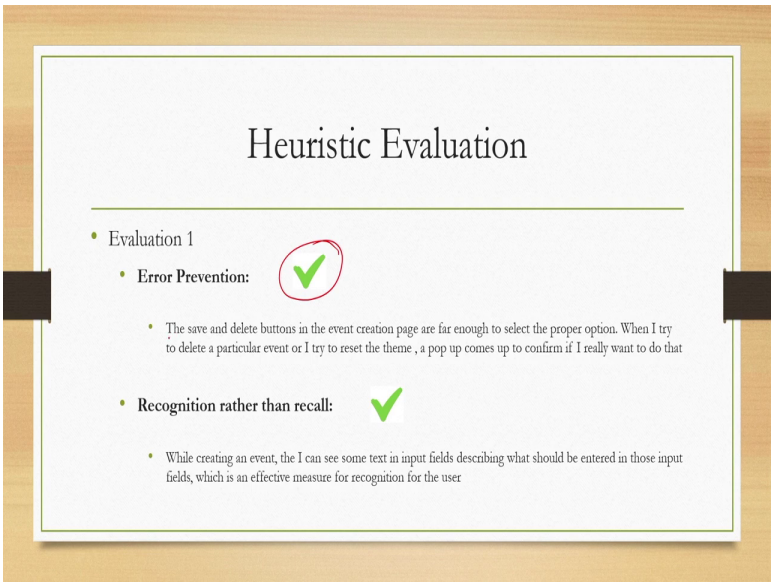
So, let us assume that heuristic Evolution has taken place just to quickly recap in heuristic evaluation we have a team of 3 to 5 expert users who are domain experts. They are asked t check the interfaces and see whether the interface design satisfies the heuristics and they are asked to prepare reports. Each evaluator prepare reports and at the end those are compiled to identify points of concerns.

Let us assume that we have followed such a approach such an approach and let us discuss the results. Suppose there is one evaluator who produced a report which we are terming as evaluation one. So, the evaluator found that the visibility of system status is not satisfied. The evaluator feels that the system status is not visible for any instances. For example when I set a new event the system does not show any status whether it is a success or a failure.

So, the view of the evaluator is that system status is not visible. Match between system and real world that is satisfied all the symbols words used in the system are matched to the real world these are commonly used word and symbol. So, the second heuristic is satisfied according to the first evaluator. User control and freedom not satisfied there are no undue and redo features for some actions consistency and standards not satisfied the system is not consistent with the various calendar applications.

The deadline view timetable View and the schedule view buttons are placed in the places which is not common in the calendar apps. A separate menu bar should be provided to select these options. So, according to the evaluator these two heuristics are not satisfied.

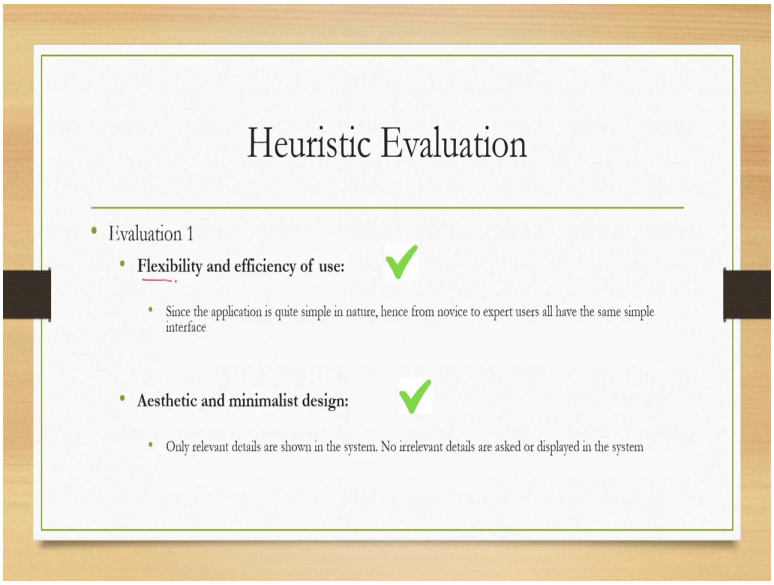
(Refer Slide Time: 23:53)



But the designs supports error prevention the save and delete buttons in the event creation page are far enough to select the proper option. When I try to delete a particular event or I try to reset the theme a pop-up comes up to confirm if I really want to do that. So, these according to the evaluator help preventing error. Evaluator also feels that recognition rather than recall is supported in this design.

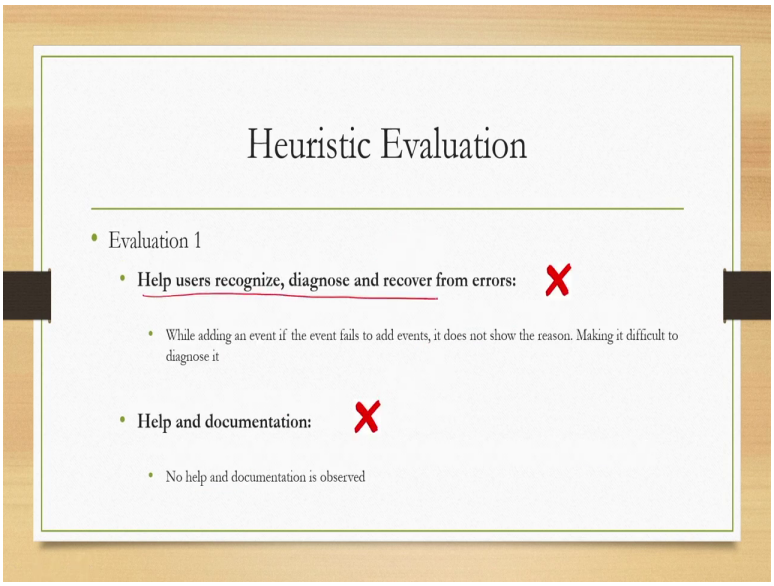
While creating an event I can see some text in input Fields describing what should be entered in those input fields which is an effective measure for recognition for the user.

(Refer Slide Time: 24:32)



Regarding flexibility and efficiency of use the evaluator feels it is supported since the application is quite simple in nature hence from novice to expert users all have the same simple interface. So, the heuristic is supported aesthetic and minimalist design heuristic is also supported only relevant details are shown in the system no irrelevant details are asked or displayed in the system.

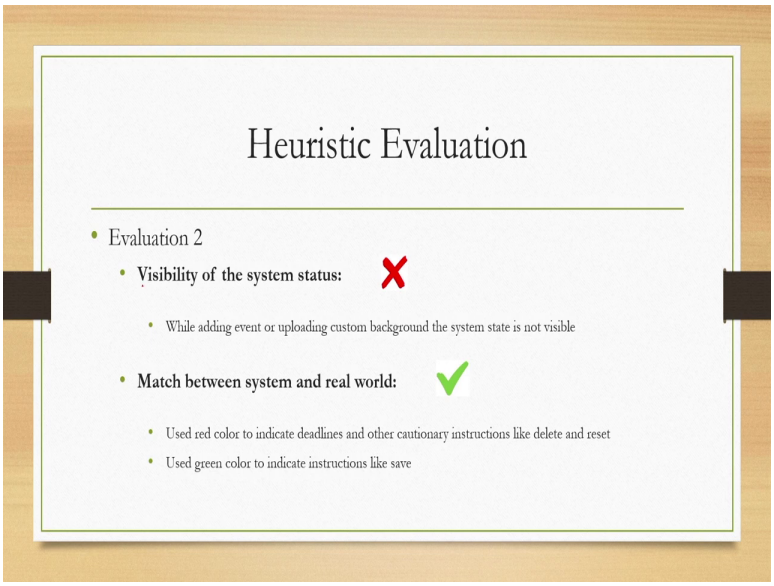
(Refer Slide Time: 24:58)



Help users recognize diagnose and recover from errors not supported according to the evaluator while adding an event if the event fails to add events it does not show the reason making it difficult to diagnose it. Help and documentation is also not supported no help and documentation

is observed. So, these are the opinions according to evaluator one. Let us see another evaluation report let us call it evaluation two.

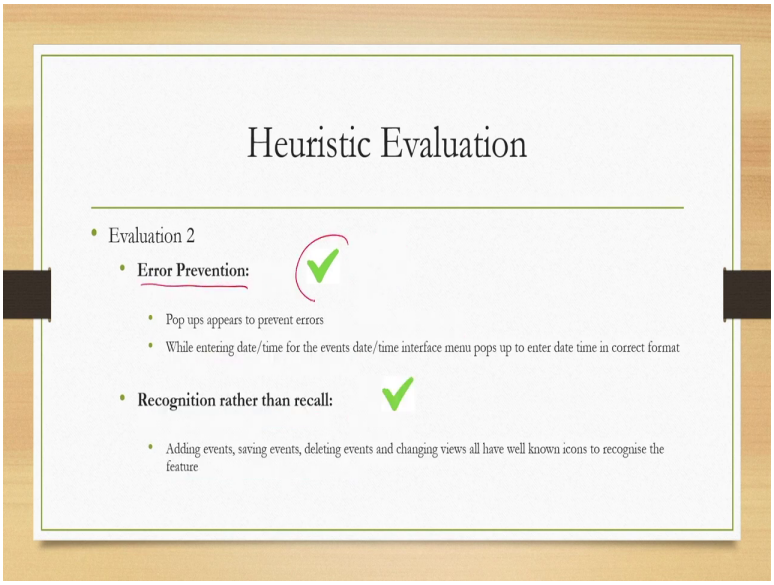
(Refer Slide Time: 25:30)



Here also according to the second evaluator visibility of system strategies not supported while adding event or uploading custom background the system state is not visible. Match between system and real world supported used red colour to indicate deadlines and other cautionary instructions like delete and reset. Used green colour to indicate instructions like save user control and freedom according to this evaluator it is supported when delete option is selected a prompt pops up to undo it once event is created it can be edited.

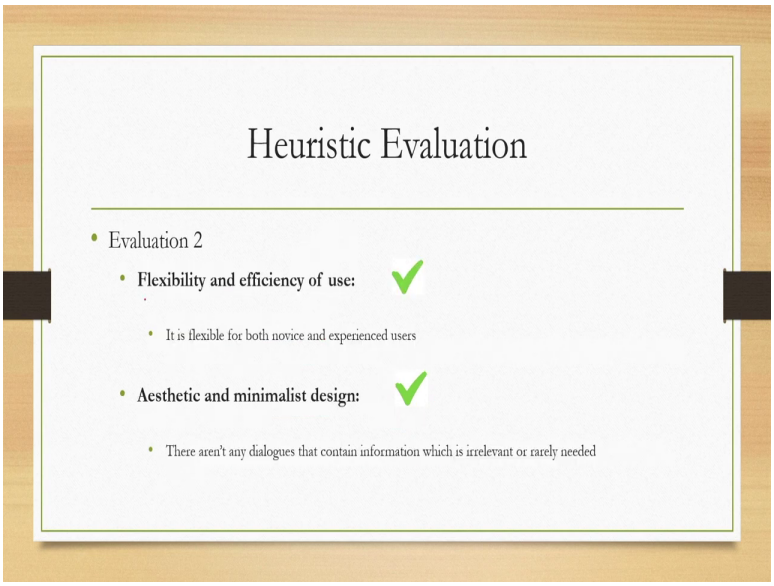
Regarding consistency and standards not supported the menus are placed in the odd place according to the evaluator.

(Refer Slide Time: 26:20)



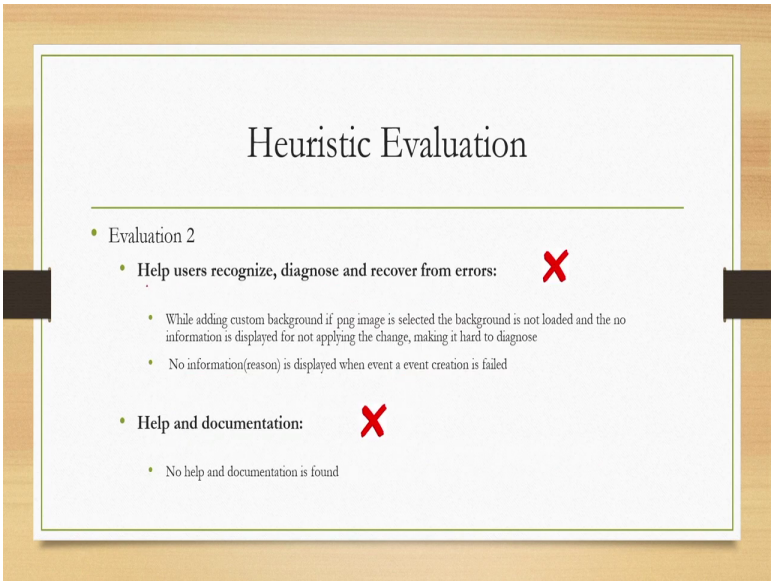
Error prevention supported pop-ups appears to prevent errors while entering date time for the events date time interface menu pops up to enter date time in correct format. Recognition rather than recall supported adding events, saving events, deleting events and changing views all app well known icons to recognize the feature.

(Refer Slide Time: 26:44)



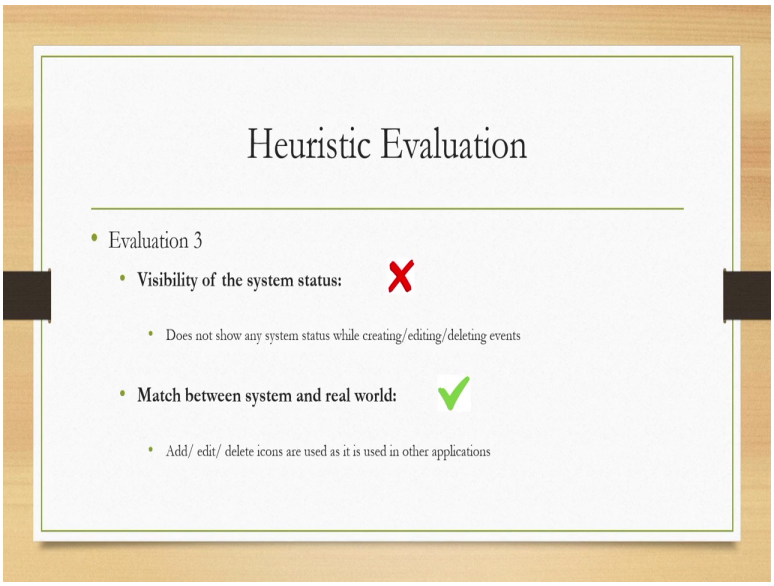
Flexibility and efficiency of use supported. The system is flexible for both novice and experienced users, aesthetic and minimalist design supported there are not any dialogue that contain information which is irrelevant or rarely needed. So, this is the view of the evaluator.

(Refer Slide Time: 27:04)



Help users recognize diagnose and recover from errors not supported. While adding custom background if some PNG image is selected the background is not loaded and the information is insufficient that means no information is displayed for not applying the change making it hard to diagnose. Now information is displayed when event creation fails help and documentation is also not supported no help and documentation is found in the prototype. So, this is the view of evaluator 2.

(Refer Slide Time: 27:44)

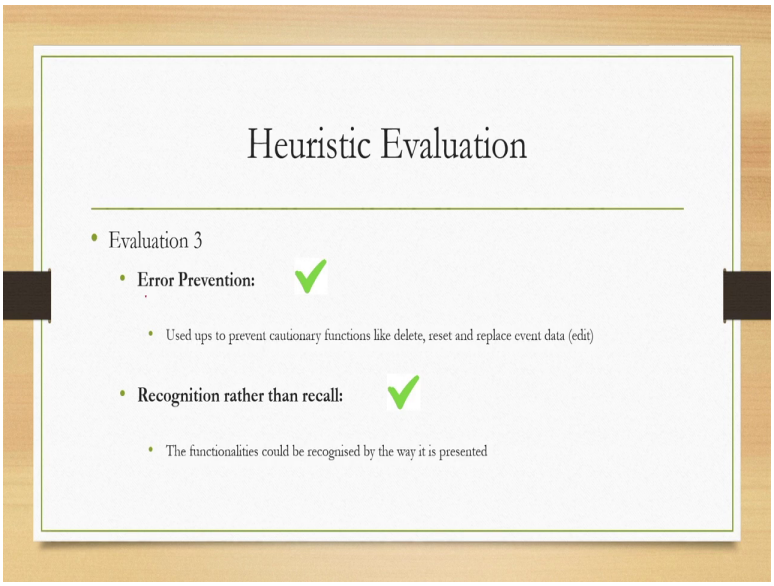


Let us look at the view of a third evaluator. Visibility of system status not supported does not show any system status while creating editing or deleting events. Match between system and real world supported add edit delete icons are used as it is used in other applications.

(Refer Slide Time: 28:03)

User control and freedom supported users have the control over the created events, consistency and standards not supported the set background button looks bigger than other buttons.

(Refer Slide Time: 28:20)



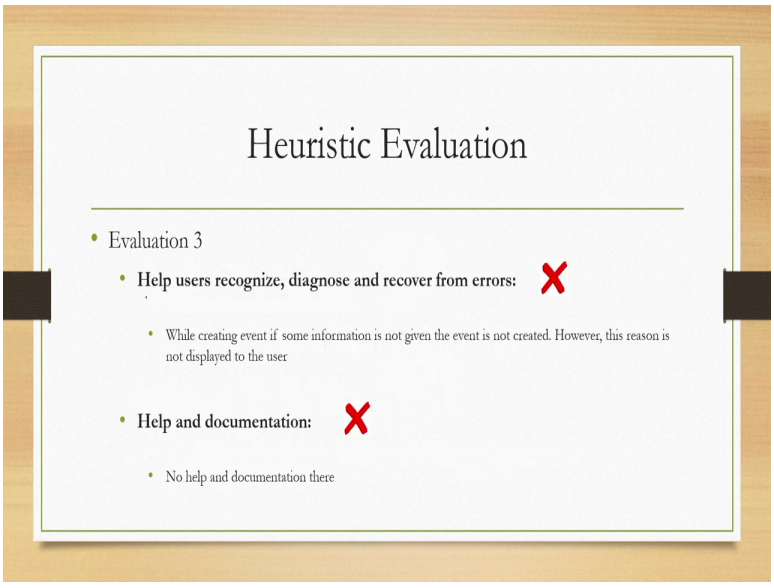
Error prevention supported used pop-ups to prevent functions like delete, reset and replace event data or rather during edit bit of operations pop-ups were there to get confirmation. So, that helps

in error prevention. Recognition rather than recall supported the functionalities could be recognized by the way it is presented.

(Refer Slide Time: 28:54)

Flexibility and efficiency of use supported. The system is very simple and can be used by both novice and expert users. Aesthetic and minimalist design supported only relevant details are given while viewing dates events and creating new events.

(Refer Slide Time: 29:12)



Help users recognize diagnose and recover from errors, not supported. While creating event if some information is not given the event is not created however this reason is not displayed to the

user. Help and documentation not supported no help and documentation there. So, we have seen different opinions rather different responses to the evaluation by different evaluators. At the end we are supposed to compile them together and identify common points of concerns which will eventually lead to refinement of the design.

(Refer Slide Time: 29:56)

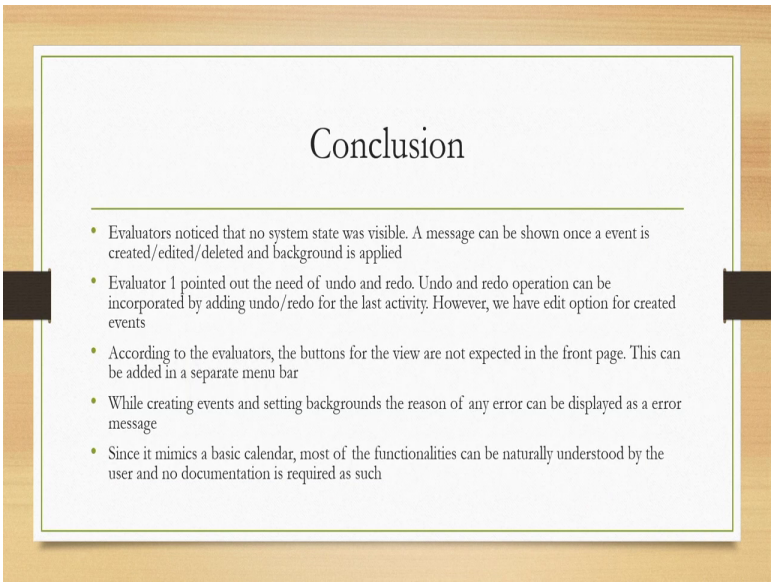
So, if we summarize the evaluations what we see is that in case of the first heuristic visibility of system status evaluator one things it is not supported two things not supported 3 also things not supported. Second one match between system and real world all 3 things supported. Third one user control and freedom one thing not supported 2 and 3 things think there it is supported. Consistency and standards none of the evaluated things it is supported.

Error prevention recognition rather than recall, flexibility of use, aesthetic and minimalist design all the evaluators things these 4 are supported in the proposed design represented in the form of prototype. Whereas the remaining two help users recognize diagnose and recover from errors and help and documentation support all the evaluators think these two are not supported. So, only in case of user control and freedom there is a difference of opinion between the evaluators.

Otherwise in all the other cases all the other heuristics, 9 heuristics the opinion is same. Now here we can say that because majority thinks it is supported out of 3, 2 things it is supported for this particular third heuristic user control and freedom we may consider it to be supported

whereas for heuristic 1, heuristic 4, 9 and 10 these 4 heuristics are not supported. So, we have to do something in the design to support these heuristics.

(Refer Slide Time: 31:57)



We can summarize as the evaluators noticed that no System state was visible a message can be shown once an event is created edited deleted and background is applied. So, that is what can be done to refine the design. Evaluator one pointed out that there is a need for undo and redo. Undo and redo operation can be incorporated by adding undo redo for the last activity. However we have edit option for created events.

According to the evaluators the buttons for the view are not expected in the front page first interface this can be added in a separate maneuver. So, these are some suggestions that can be used to refine the design. So, that the heuristics are supported while creating events and setting backgrounds the reason of any error can be displayed as an error message. Since it mimics a basic calendar most of the functionalities can be naturally understood by the user and no documentation is required as such.

So, although that particular heuristic is not supported all the evaluators pointed out that the interface is simple. So, documentation may not really be required. So, it does not matter whether the; particular heuristic is supported and documentation is added or not because this particular application is very simple. So, at the end what we get is some suggestions first status check of

the quality of the design with respect to the heuristics and then based on the responses we can figure out whether we need to make improvements or it is not really necessary.

Even if the heuristic is not supported like in the case of help and documentation. So, that is all for this topic. So, in this lecture we have learned about how to create prototype how it looks and how we can apply cognitive work through or heuristic evaluation to evaluate the prototypes what are the steps what reports we get how to compile them and how to draw conclusion. Based on this conclusion we may go for refinement of design which is followed by again prototyping and evaluation or we may say that there are no further issues.

So, we can freeze the design and go to the code design. I hope you enjoyed the content of this lecture looking forward to meet you in the next lecture, thank you and goodbye.