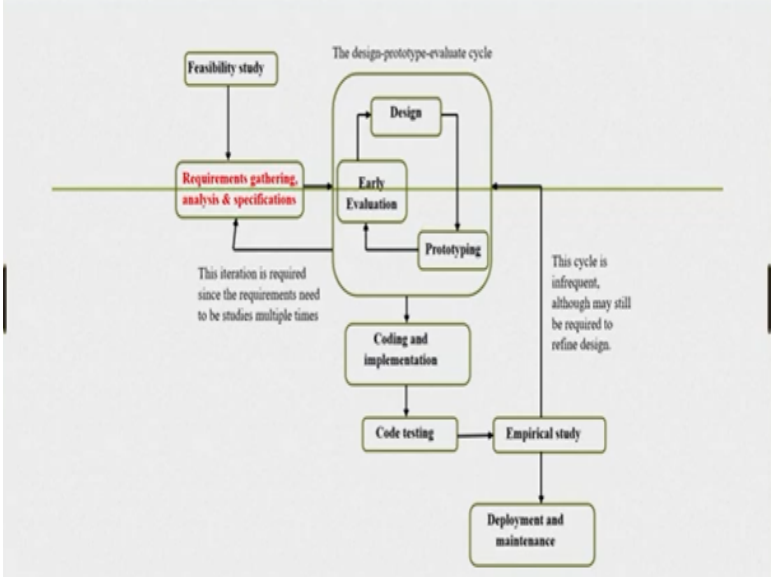
Design & Implementation of Human - Computer Interfaces
Dr. Samit Bhattacharya
Department of Computer Science & Engineering
Indian Institute Technology - Guwahati

Module No # 03
Lecture No # 11
Case Study-Non-Functional Requirements of SRS

Hello and welcome to the NPTEL MOOCs course on design and implementation of human computer interfaces lecture number 10. Here we are going to continue with our previous case study on calendar application development. Before we begin we will again go through the basic idea of the interactive system development life cycle for easier recollection and then we will continue with the example.

(Refer Slide Time: 01:28)



So we are currently focusing on a particular stage of the overall interactive system development life cycle which comprises many stages. What are those stages? Just to have a quick recap we have the feasibility study stage, requirement gathering analysis and specification stage, design stage, prototyping stage, early evaluation of the prototype stage, coding and implementation stage, code testing stage, empirical study stage and finally deployment and maintenance stage.

So all together we have 9 stages that make this interactive system development life cycle. Now as you can see among these stages some stages form cycles, which means they are generally performed in a loop iteratively at many places we can find such iterations. For example in the

design prototype and evaluation stages create a cycle as shown here in this block. That means that when we go for design then this design has is prototyped and quickly evaluated to find out problems with the design and this process goes on iteratively Till we arrive at a design that no longer has many issues.

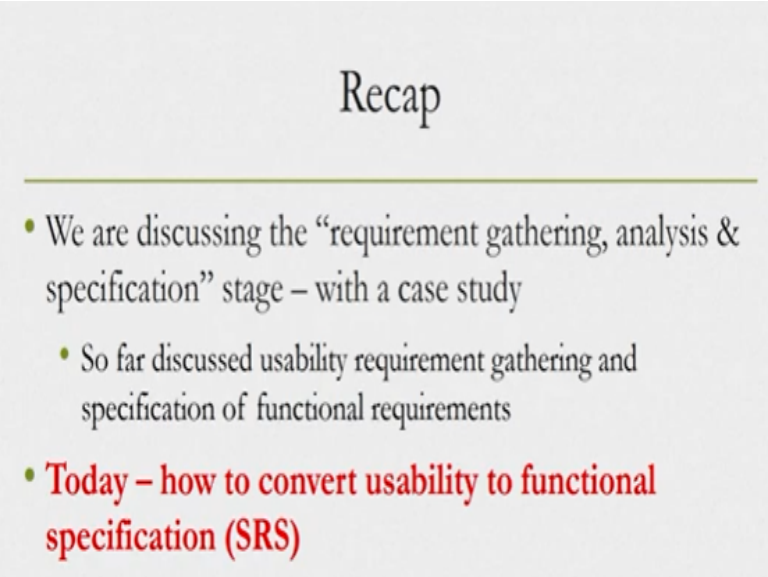
Now, when we say it does not have many issues we essentially refer to the fact that the design is now likely to be leading to a usable product another important thing that we should again keep in mind is that when we are talking of the term design. We are actually referring to 2 types of designs one is the design of the interface and interaction that is from the user's point of view. Another one is the design of the system which is from the systems point of view.

Now when we talk of the design prototyping evaluation cycle essentially, we are referring to the design of the interfaces and interactions. In the case of the design of the system we may not require prototyping and quick evaluation, however, there are other ways to evaluate system designs quickly which are covered under the code testing stage. Similarly, there is another bigger cycle that we can see from this design prototype.

Evaluate cycle to empirical study and back so this bigger cycle is likely to be infrequent. Otherwise, it will lead to huge costs and resource requirements. So only once or twice may happen there is even a smaller cycle between the design and requirement gathering. So if the design is found to be having too many problems then we may like to have the requirement gathering to refine the design so this smaller cycle may also be there.

Among these stages we are currently discussing the requirement gathering analysis and specification stage so we have already covered the basic concepts and currently, we are focusing on understanding these concepts in terms of one case study one example is system development, which is a calendar application.

(Refer Slide Time: 05:30)



So in the requirement gathering stage, we learned what is the key requirement from the point of view of an interactive system? So what we have learned is that the key requirement from the point of view of an interactive system is the usability of the system now usability requirement is applicable for interface and interaction design. It is not necessarily a requirement for system design, so currently we are focusing on the interface and interaction design for which usability is a key requirement.

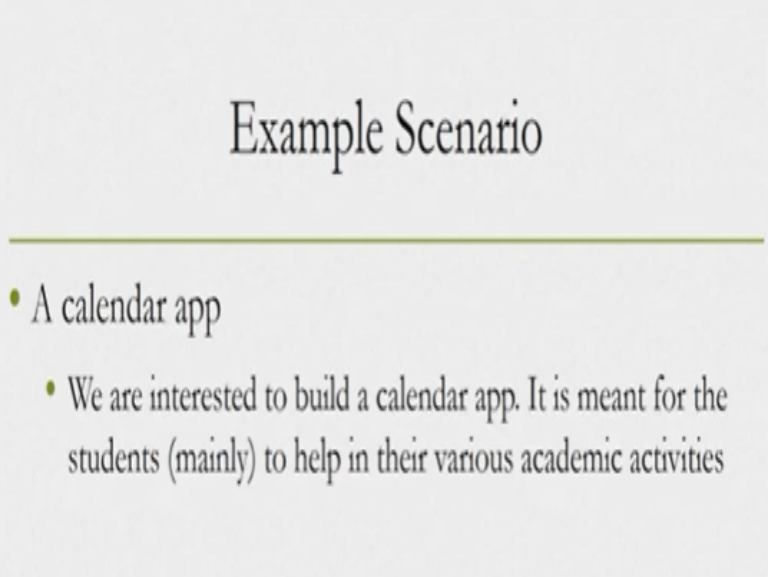
Also, we have learned about how to find out functional requirements and specify it note here that usability requirement comes under the non-functional requirement. Now there are other ways to specify requirements these are functional requirements that can directly translate to features of a system. So we have seen what are usability requirements? And we have also discussed how to find them out through contextual inquiry?

In the case study, we have seen how to find out through contextual inquiry usability requirements for the calendar application. Separately in another lecture, we have seen how to find out the functional requirements for the calendar application? Now in this lecture, we are going to see how these 2 requirements can be combined and a common specification of requirements can be generated in other words.

Here we are going to talk about how to make use of the usability requirements? That we have identified through the contextual inquiry method to generate functional requirements that can be

integrated into the software requirement specification document, which primarily consists of a functional requirement specification.

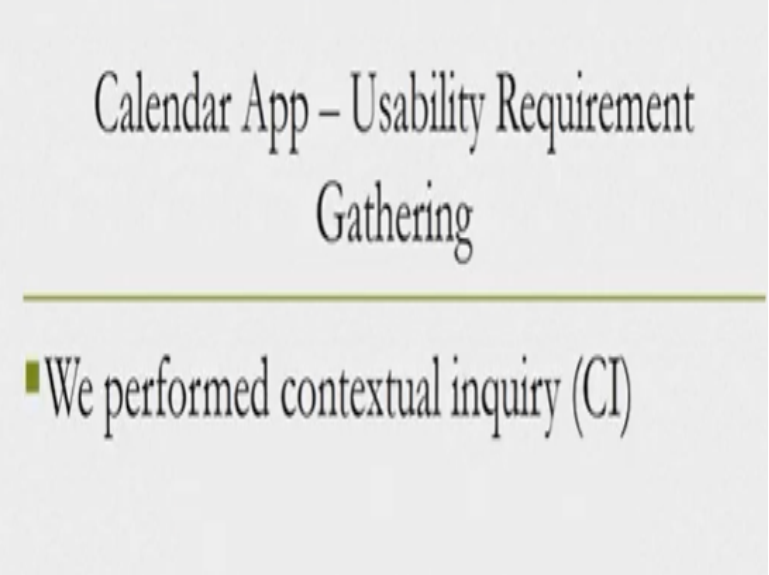
(Refer Slide Time: 07:43)



So we will continue with the calendar application that we were covering in the earlier lectures to recap so we are interested to build a calendar application. Now the application is meant for students, primarily students enrolled in colleges and universities for higher studies, and the application is meant to help them in their academic activities. So the user group, as well as the user context, are specified with these statements.

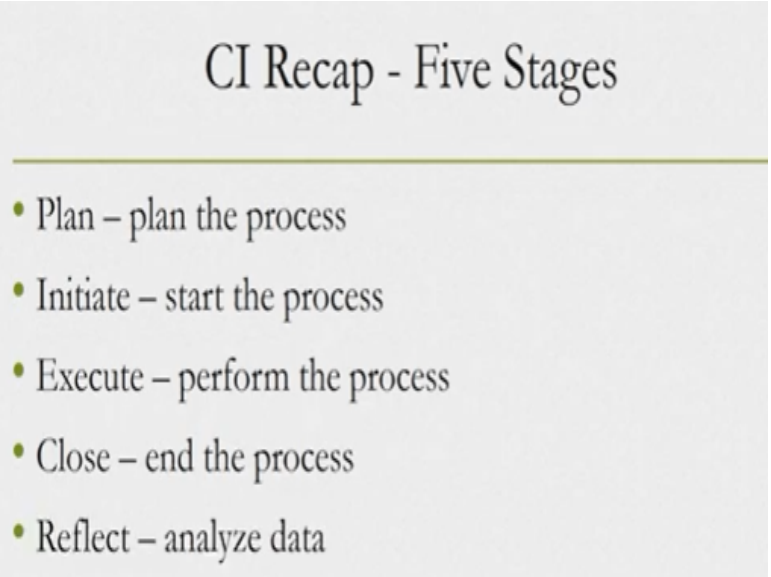
Now earlier we have seen how to find out usability requirements for the calendar app from the contextual inquiry which is one of the many methods available to identify usability requirements.

(Refer Slide Time: 08:34)



So we assumed that we performed a contextual inquiry to identify the usability requirements.

(Refer Slide Time: 08:41)



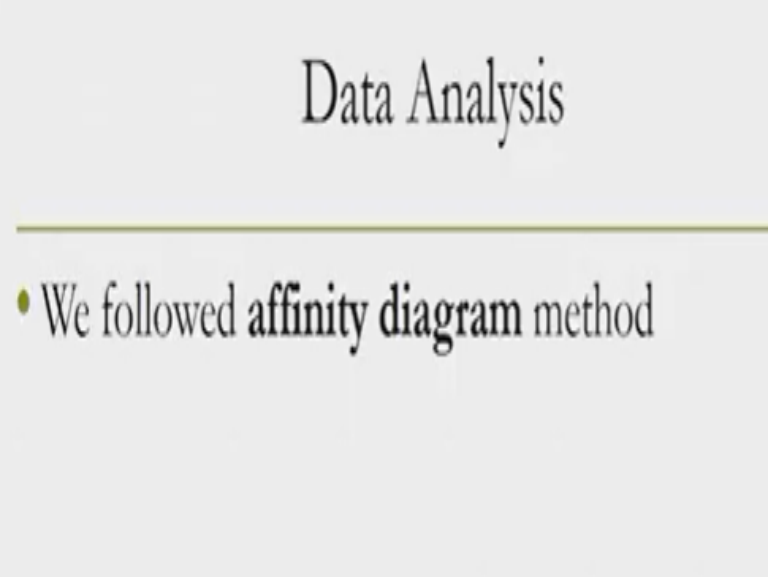
Just for a quick recap what is contextual inquiry? So it comprises it is a process that comprises 5 stages it starts with a planning stage where we plan for the overall inquiry process. Preferably in the form of a script then initiate stage where we start the process by communicating with the potential users from whom we will collect data as well as their higher authorities in the work setting if any such authorities exist.

In the third stage now we go for actual data collection which means we observe the users and take note of their behaviour, while they perform their tasks in the work setting. If you may

recollect contextual inquiry is primarily an observation of user behaviour in a work setting in their natural work setting. And occasionally this observation can be augmented with semi-structured interviews for clarification on certain observations.

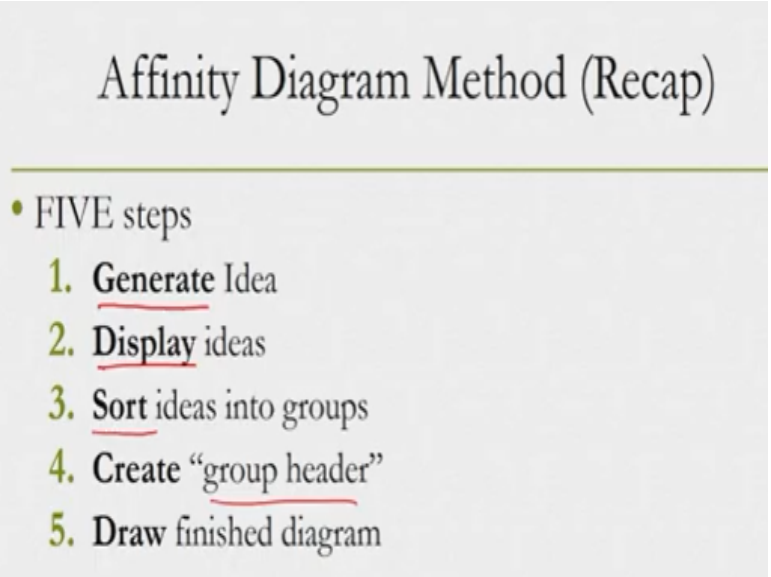
So the third stage is the actual observation and data collection stage. In the fourth stage, we close the observation process by sending preferably thank you notes to the participants and creating a rapport, so that in the future their services can again be availed if such need arises. In the fifth and final stage we go for analysis of the observed data. There are many analysis methods and we specifically learned about one particular method namely the affinity diagram method.

(Refer Slide Time: 10:42)



Now in the affinity diagram method again there are 5 stages and through those stages, we can analyze the data.

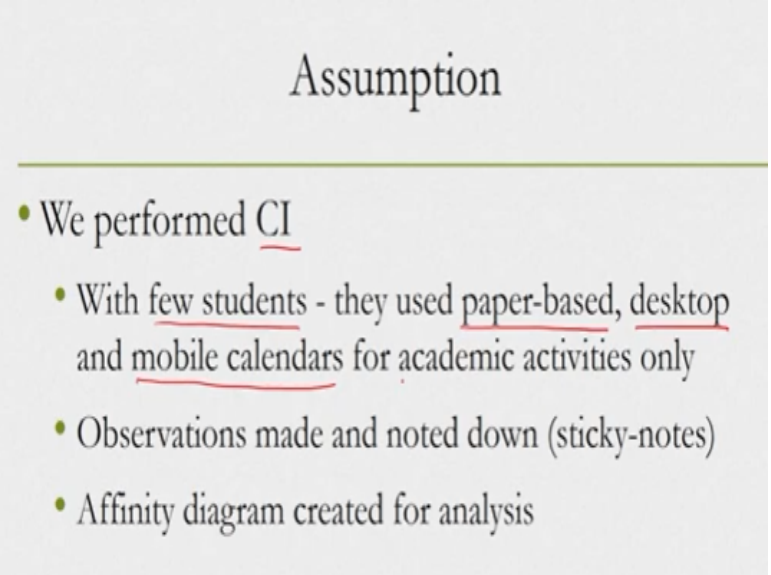
(Refer Slide Time: 10:52)



So what are those 5 steps or stages, first is generate idea from observation then display the ideas on some place it may be a wall, may be some screen, it may be whiteboard anywhere. Now the ideas can be displayed in the form 5 of sticky notes and then we need to perform brainstorming in a team to sort those ideas into groups of similar ideas. In the fourth stage we have to assign some group header to each group of similar ideas and finally we need to draw the finish diagram.

So these are the 5 steps or stages through which we can analyze the data that we have collected during observations in a contextual inquiry process. Now as, we have mentioned in the earlier lecture, where we discussed contextual inquiry for the case study.

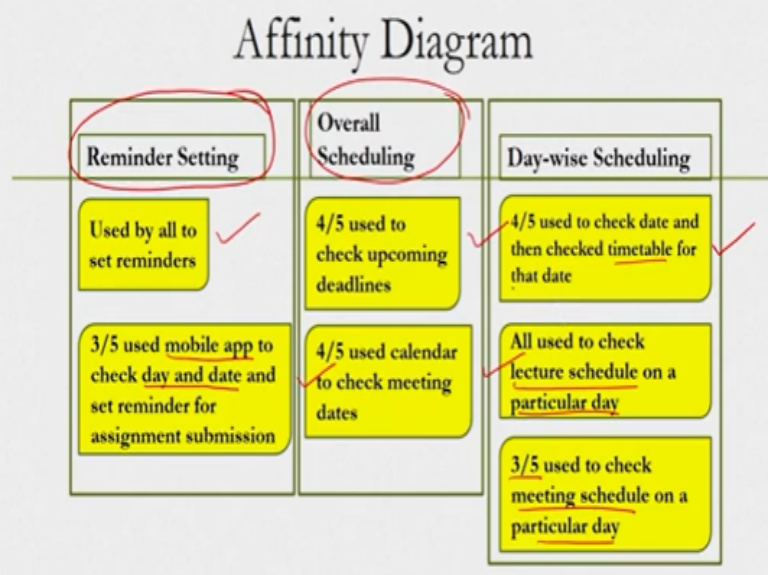
(Refer Slide Time: 12:07)



We mentioned that we performed CI hypothetically with some students, where we assumed that the students used a paper-based calendar, a computer-based calendar. Here by the term computer. So, in the desktops Computers are referred to as calendars that are available on mobile phones and they use the calendars only for academic activities.

So we ensured that the work setting remains the same and while they are using these calendars in those methods so you observed their behaviour and noted down, whatever notes we needed to take to record their behaviour. And to note down, we use sticky notes and then we performed affinity diagram method to analyze the data where we created the affinity diagram from the observations.

(Refer Slide Time: 13:18)



So we discussed in detail the final diagram and how we arrived at that diagram what observations we made how we grouped them together and how we got to know the final diagram? So we will skip those steps here and only reproduce the final affinity diagram that we obtained after performing the method. So we managed to get a few groups one group contains 2 observations one is the calendars were found to be used by all the users to set reminders, and 3 out of 5 users were found to use mobile apps to check days and dates and set reminders for assignment submission.

So these 2 observations that we have noted down were found to be related and we put them together in a group with a group heading reminder setting. Similarly, we have made another group of observations 2 observations where there 4 out of 5 users used the calendar systems. Either physical or mobile or desktop based to check upcoming deadlines and again 4 out of 5 users use the calendars to check meeting dates.

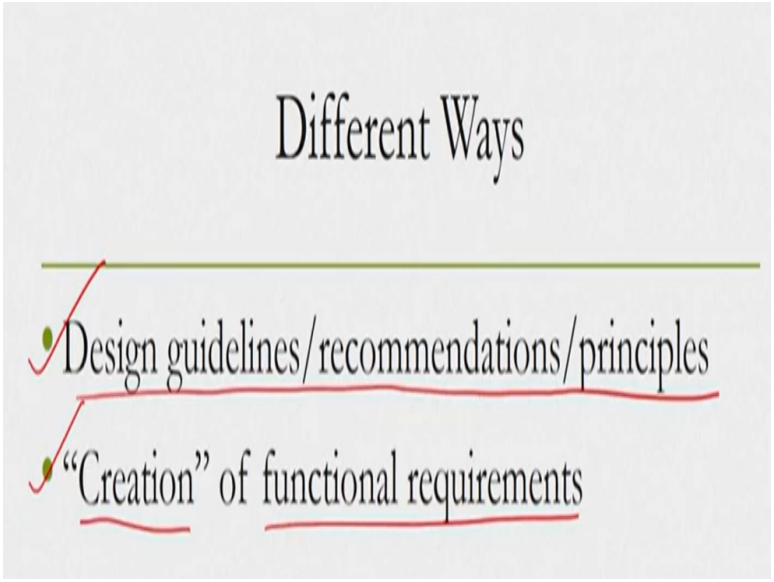
So these 2 again are related and we put them together under a group with a group heading overall scheduling. We have also created a third group of observations having 3 observations one is 4 out of 5 users were found to use the calendars to check dates and then check the timetable for that particular date. All were found to use the calendars to check lecture schedules on a particular day and 3 out of 5 were found to use the calendar to check the meeting schedule on a particular day.

So in all the cases as you can see a particular day is a common link so accordingly, we group them together these 3 observations and provided a group heading for day-wise scheduling. So at the end of the affinity diagram method, we found 3 groups of observations with 3 headings reminder setting, overall scheduling and device scheduling. So this is the output of the contextual inquiry that we have performed for the calendar application development.

(Refer Slide Time: 16:22)

Now the question is what to do with this output? So we made some observations and created an affinity diagram now how we are going to use this affinity diagram that is going to be useful for the development of the system.

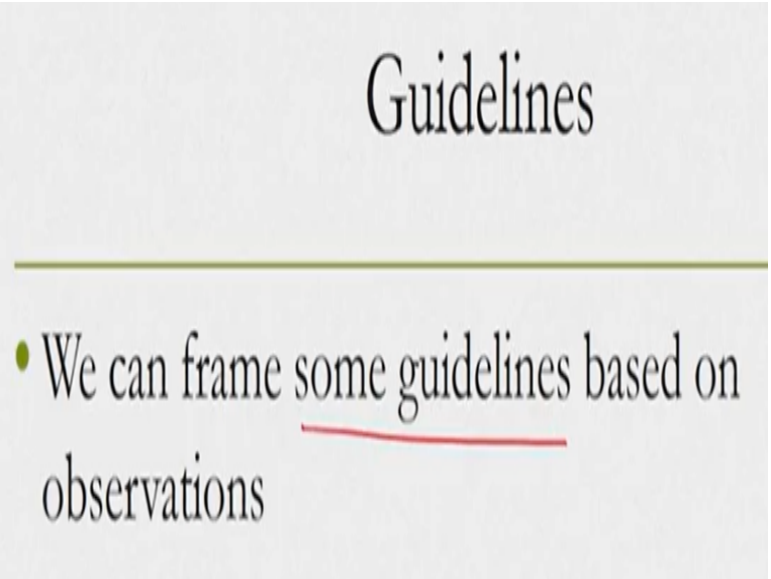
(Refer Slide Time: 16:43)



We can actually make use of these observations or rather we can actually make use of this output of the process in either of the 2 ways. One is either we can come up with a set of design guidelines or recommendations or principles to guide the overall design of the interface. So this is one method one way of using the output the other way of using the output is to create functional requirements.

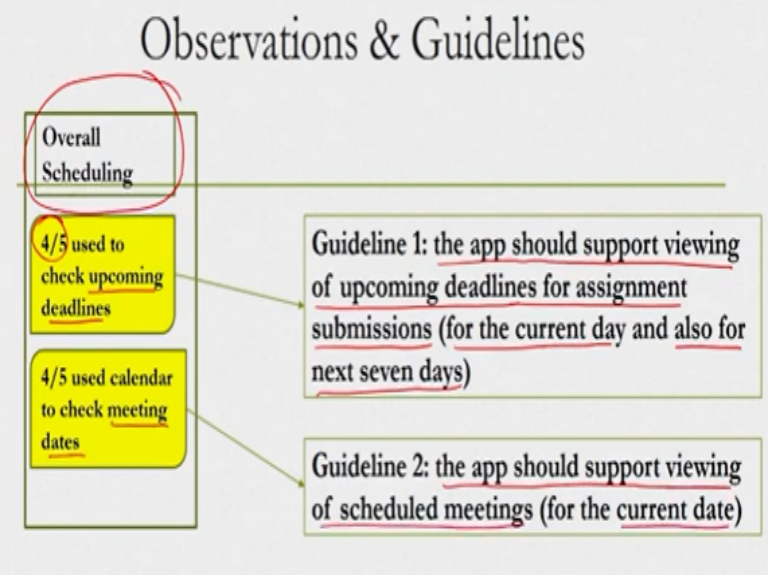
So one is to create guidelines for design other one is to create functional requirements for system development. Now when we say design guidelines we specifically refer to the design of the interface rather than the system. So given the output that we have just seen for the calendar app what kind of guidelines we can generate?

(Refer Slide Time: 17:50)



In fact we can frame more than one guideline from the output that we got at the end of the process.

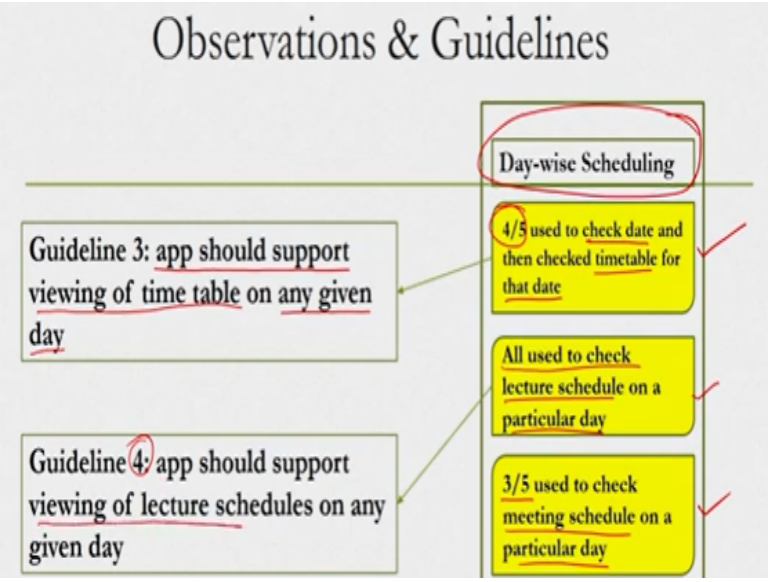
(Refer Slide Time: 18:00)



Let us concentrate on the first group of observation that is overall scheduling. Here we have seen that 4 out of 5 users were found to have used the calendars to check upcoming deadlines as well as meeting dates. From these observations we can make some guidelines one guideline can be the app resultant app should support viewing of upcoming deadlines for assignment submission for the current day and also for next 7 days this may be one guideline.

That is the app should have some support for a specific functionality which we can frame based on the observation. Similarly based on the other observation we can frame another guideline that is the app should support viewing of scheduled meetings for the current date. So in the group there are 2 observations each observation can give us some idea about a particular guideline that is going to be helpful in coming up with a product, that is likely to be usable.

(Refer Slide Time: 19:32)

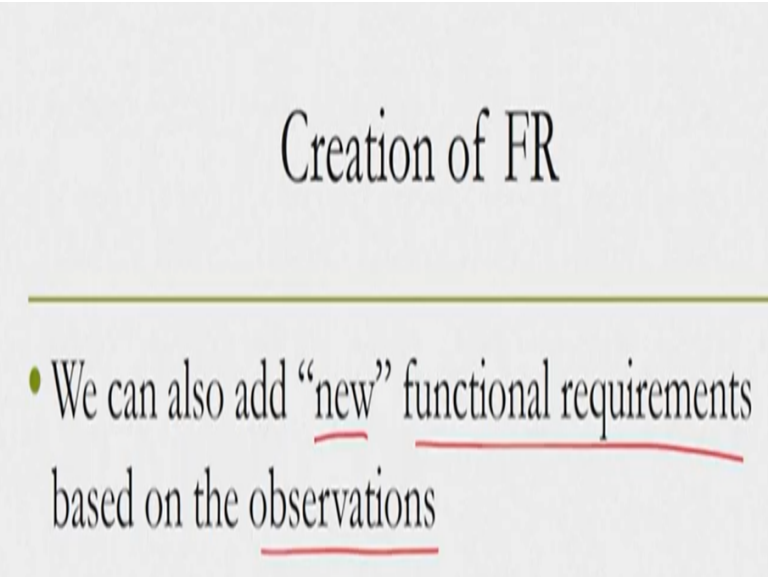


Let us see the other group of observations the day-wise scheduling under this group there are 3 observations. Observation one 4 out of 5 users were found to use the calendar to check the date and timetable for that date. Second observation all users were found to use the calendar to check lecture schedule on a particular day. And the third observation is 3 out of 5 users were found to have used the calendar to check meeting schedule on a particular day. So we have made 3 observations.

So let us see how from these observations we can again frame some guidelines? One guideline which we can call guideline 3 can be the final product that should support viewing of the timetable on any given day. Again this indicates what functionality the final product should have rather than how to have that functionality. Similarly, another observation can lead us to frame another guideline that is guideline number 4.

That is the calendar app or the product should support viewing of lecture schedules on any given day. Again this comes directly from the observations that the users are likely to use these functions quite frequently in their work setting which in turn is going to make the product usable.

(Refer Slide Time: 21:25)

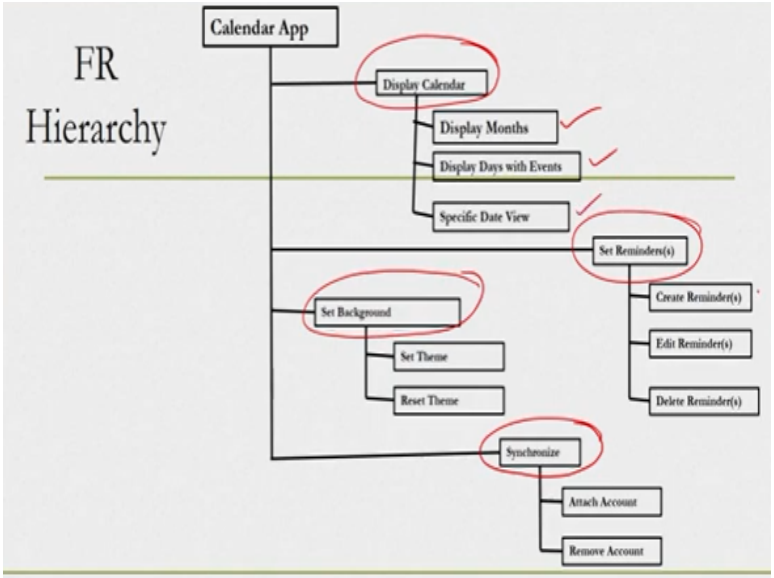


So that is one way of making use of the output of the contextual inquiry process. That is the framing of guidelines as we have seen we have framed 4 guidelines. These 4 guidelines can be used to add functionalities to the final product or the calendar app which are likely to be used frequently by the specified group of users that is the students in the specified work context that is the academic settings.

And that in turn is likely to lead to more efficiency effectiveness and satisfaction of the end product as per the standard definition of usability. Now there is as we have mentioned another way of making use of the findings that is the addition of new functional requirements based on the observations. So earlier we have seen that how to create the functional requirements that is primarily by interviewing the clients or customers.

And based on that we can create a functional requirement and specify it using a specific notation. The usability study also can lead us to identification of new functional requirements that we can add to the other functions in the functional requirement hierarchy.

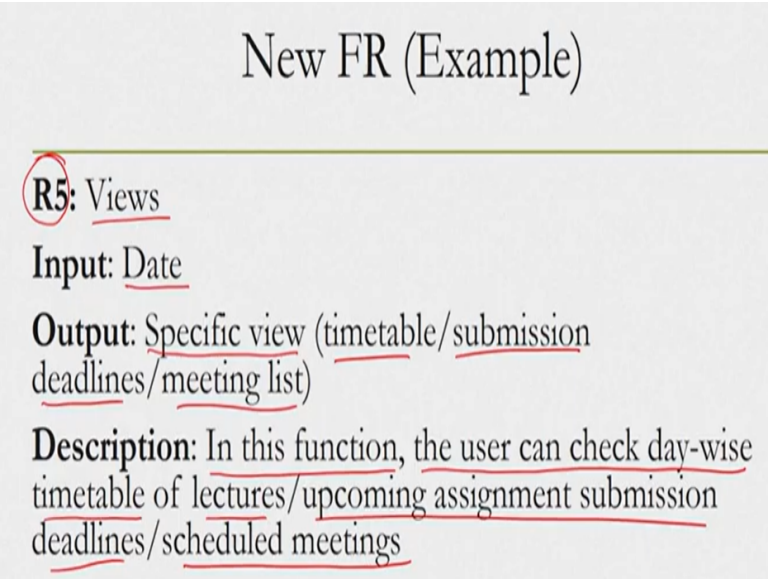
(Refer Slide Time: 23:11)



So in this case let us see what new functions we can add before that let us just quickly recap what functional requirement hierarchy we have created for the calendar app? So there were 4 top-level functions display calendar, set a reminder, set background, and synchronize. Under display calendar we had display months display days, and specific date view under set reminders. We have created reminders edit reminders and deleted reminders.

Under set background, we have set theme to reset theme and under synchronize, we have attached the account remove account. So these are the functions that we have identified and the hierarchy we have created along with the specification of the hierarchy.

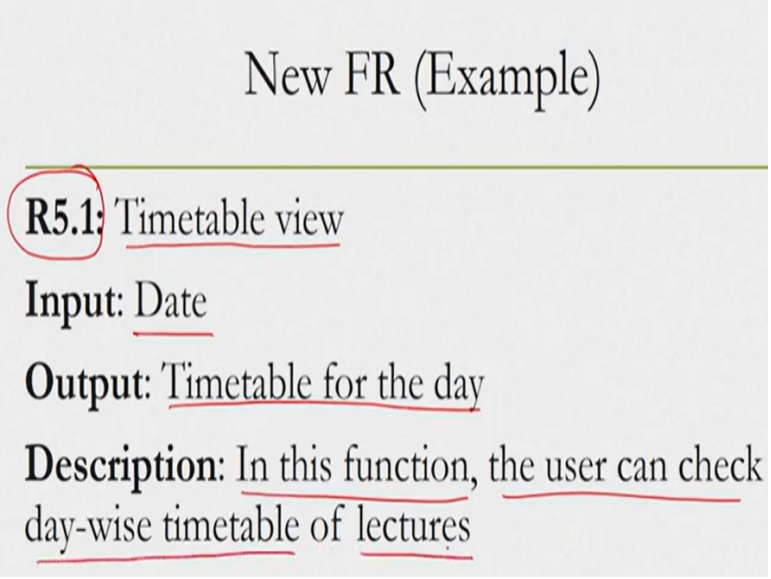
(Refer Slide Time: 24:05)



Now based on the affinity diagram we can actually find out newer functions and add it to the existing hierarchy. Let us see how we can do that? So based on the observations we can add a top-level function view since it is a top-level function and the fifth function in the hierarchy, so we are giving it a label R5. Now its input is any date and output is a specific view for that particular date. Now the view map, the view may be related to the timetable view submission deadlines view, or meeting list view.

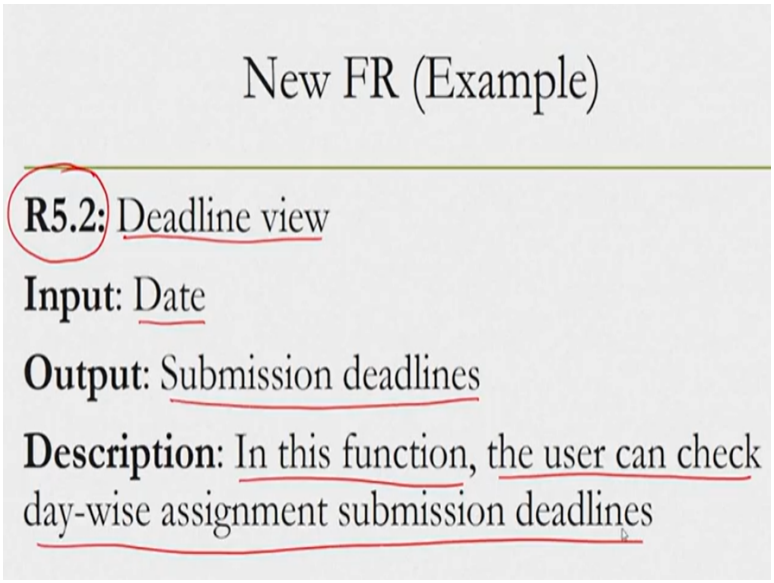
And as we learned earlier so for this top-level function we need to add some description. Although it is optional it is always preferable to add a description so in this case what we can do is we can add a simple description like in this function, The user can check day wise time tables of lectures or upcoming assignment submission deadlines or scheduled meeting lists that is the overall purpose of this view function which is a top-level function. Now under view we can have sub-functions depending on the purpose.

(Refer Slide Time: 25:33)



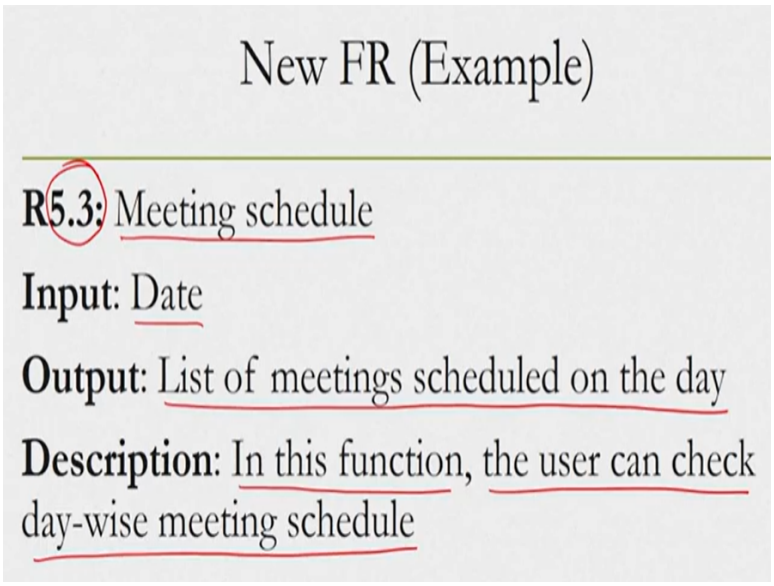
So one sub-function can be timetable viewed accordingly we have given it a level R5.1. Now its input is the date and the output is the timetable for the day it can have a simple description like in this function, the user can check the day-wise timetable of the lectures.

(Refer Slide Time: 26:03)



Similarly, we can have another sub-function deadline view the name is deadline view label is 5.2 because it is the second sub-function under the top level function view which itself is having a level 5 because it is the fifth function in the hierarchy. Now again its input is date and output is submission deadlines the description is quite straight forward. In this function the user can check day-wise assignment submission deadlines that are the purpose of the sub-function.

(Refer Slide Time: 26:45)

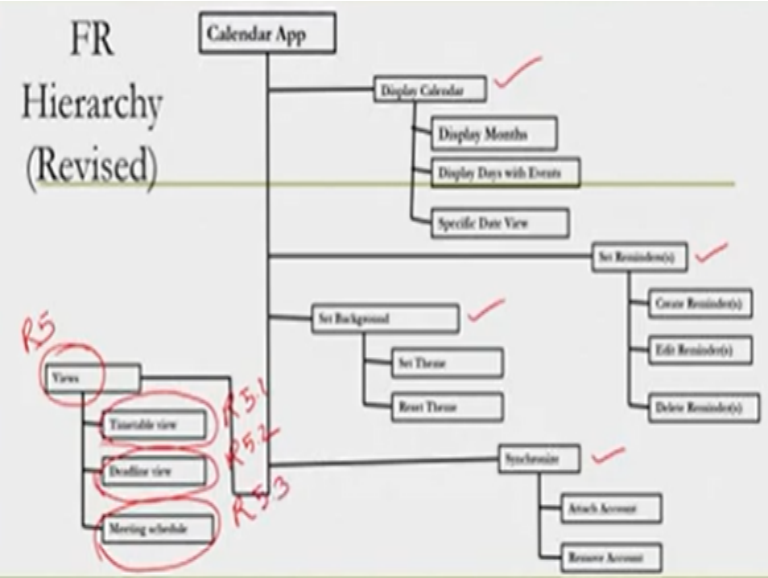


There can be a third sub-function as well under the function view which is meeting schedule. Accordingly, the level is 5.3 input is debt output is list of meetings scheduled on that day and description again can be very simple in this function. The user can check the device meeting

schedule so what we have done? From the observations we identified is that this scheduling is important and for that, the user needs to get the information to provide the information we can add certain functional requirements to the functional hierarchy.

Now through these requirements, we are indicating to the design team that these are to be provided as functions of the final product or as features of the final product which will be very much beneficial to the end users and will lead to improved usability of the product. So we have seen one addition of functional requirement top level addition that is a view along with its input-output and description under view we have added 3 sub-functions for 3 activities. One is viewing the deadlines meeting schedules and lecture schedules.

(Refer Slide Time: 28:22)



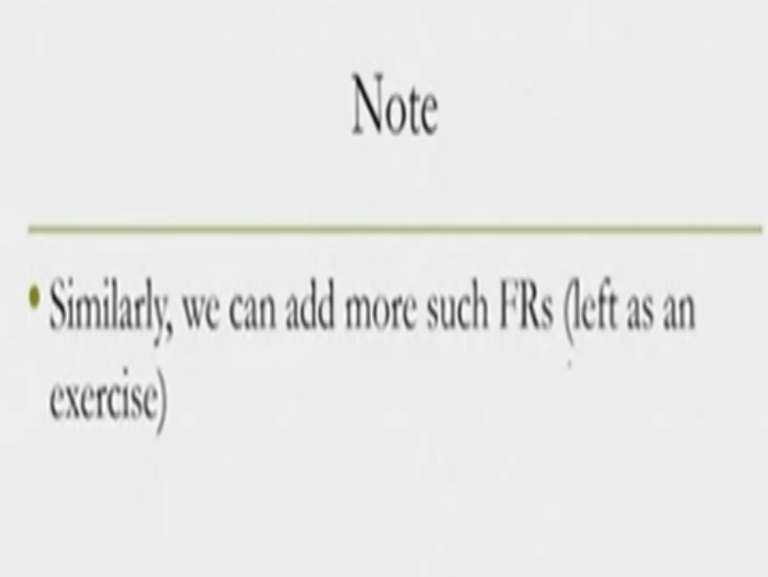
So now we can add it to the hierarchy. So we have already added these 4 top level functions display calendar, set reminder set background and synchronize. Now a fifth top level function along with its hierarchy we can add this is the views this is the fifth top-level function. Accordingly, we have given it a level R 5 and then under it, we have time table view, deadline view and meeting schedule view.

So accordingly these are labelled as 5.1, 2 to so here you can see how we made use of the outcome of the contextual inquiry process which is in the form of an affinity diagram. To identify and add a new hierarchy of functional requirements in the existing hierarchy, that is the second

way of using the outcome of the contextual inquiry process. The other way just to recollect is to frame design guidelines for use in a later design stage.

Now here, of course, we have shown one example, only depending on the other observations also we can add few more functions in the form of hierarchy and further, enhance the functional hierarchy that we have just seen.

(Refer Slide Time: 30:19)



However will not provide any more details on those functions as well as the hierarchy and I leave it for you as a take home exercise. So I would like to request all of you to go through the observations and add more functions in the hierarchy to make it complete. So that is all for today so what we have learned here is? We have gone through the final part of the case study to learn about how to make use of the outcome of a contextual inquiry method.

To enhance the software requirement, specification documents by identifying and adding more functions in the functional hierarchy. Now one thing you must keep in mind is that usability requirements are generally called non-functional requirements. So it is not mandatory to be able to convert such requirements into functional requirements however in many of the cases you will find that that is possible for certain usability requirements.

And accordingly you should try to convert those usability requirements into functions. So that in a later stage we find it easier to implement the overall system with improved usability but I

would like to re-emphasize the point that usability requirements are non-functional requirements, so it is not mandatory to be able to convert those requirements into functional requirements. So in a design situation if you find that you manage to find out some usability requirements but unable to convert them to functional requirements you need not worry that is anyway part of the overall specification process.

So for non-functional requirements we specify them separately then functional requirements for which we follow a certain convention which we have discussed throughout the previous lectures. So what matters most is? At the end you get a software requirement specification document which includes both non-functional requirement as well as functional requirements. So if you are unable to convert usability requirements into functional requirements then you can keep that requirement as non-functional requirement in the SRS document, and have the functional requirements specified as per the convention that we have seen.

With that I would like to end this lecture I hope you could understand the material and you will be able to translate it into practice I also hope that you enjoyed the learning and I look forward to see you all again in the next lecture that is all for this lecture thank you and goodbye.