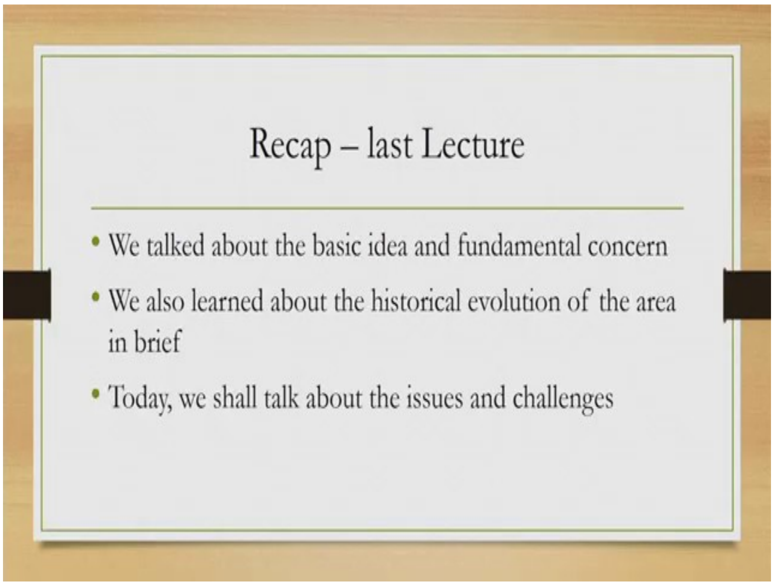
User Centric Computing for Human-Computer Interaction

Prof. Samit Bhattacharya
Department of Computer Science & Engineering
Indian Institute of Technology, Guwahati

Lecture - 02
Issues and challenges

Hello and welcome to the second lecture of the course User Centric Computing for Human-Computer Interaction. So, before going into the content of this lecture, first let me recap what we have learned in the last lecture.

(Refer Slide Time 00:49)

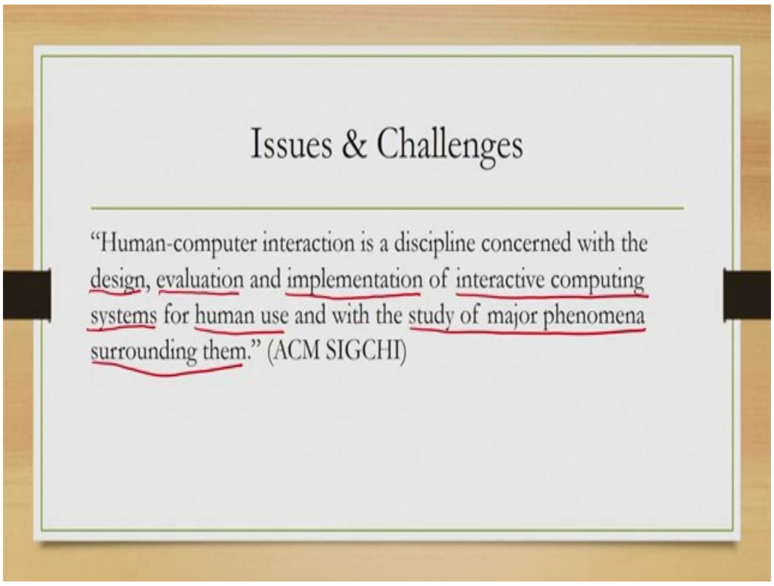


Recap – last Lecture

- We talked about the basic idea and fundamental concern
- We also learned about the historical evolution of the area in brief
- Today, we shall talk about the issues and challenges

So, we primarily talked about the basic idea, what this course is all about and the fundamental concerns that is the user centric design. Now, we also learned about the briefly, about the historical evolution of the field. In this lecture today, we will learn the issues and the challenges that are there in this field. This is meant to give you an overview of the things that that are important considerations in the field, that are important issues for research in the field and that are important issues that gives you scope to do further studies in the field.

(Refer Slide Time 01:30)



Issues & Challenges

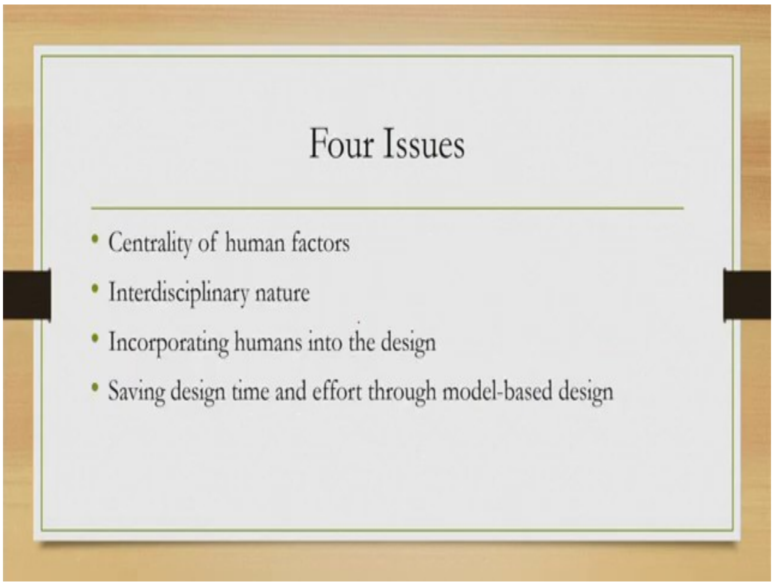
"Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them." (ACM SIGCHI)

Now, before we talk about the issues, let us first see one definition given by ACM SIGCHI about the field, about the area. Now, ACM SIGCHI as you may be knowing, is a body stands for SIGCHI, stands for special interest group on human computer interaction. It is a special interest group of ACM association of computing machinery which actually is a premier body in this field and which tried to define the field.

So, according to their definition, human computer interaction is a discipline that is concerned with the design, let me underline, the design, the evaluation and the implementation of interactive computing systems for human use and the study of the major phenomena surrounding them. Now, so, there are many important concepts, if you may have noticed.

So, first of all we are concerned with the design, evolution and implementation. So, three things of specific group of systems called the interactive computing systems, which are meant for human use; so, that is very important and that is what we are focusing in all the lectures. So, along with these three things design, evolution and implementation the discipline also deals with the study. It is not only design, evolution and implementation of a field, but also study of major phenomena surrounding this systems.

(Refer Slide Time 03:13)

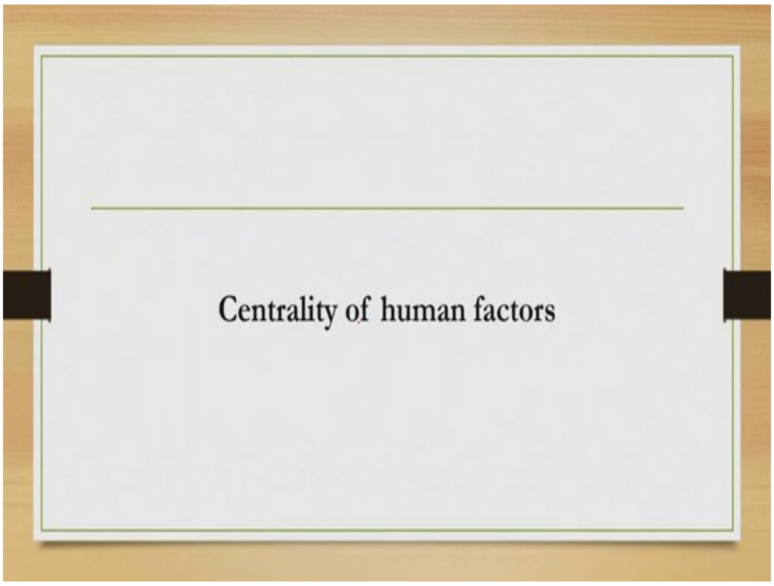


Four Issues

- Centrality of human factors
- Interdisciplinary nature
- Incorporating humans into the design
- Saving design time and effort through model-based design

So, both comes under this discipline. Now, based on this definition of HCI by ACM SIGCHI community, if we see all the issues that are there we can broadly divide them into four major issues or you can say at the same time issues and challenges. First of all the centrality of human factors, then the nature of the discipline, then the challenge of incorporating humans into the design and finally, the challenge of reducing the design time and effort, design and implementation time and effort through model based design.

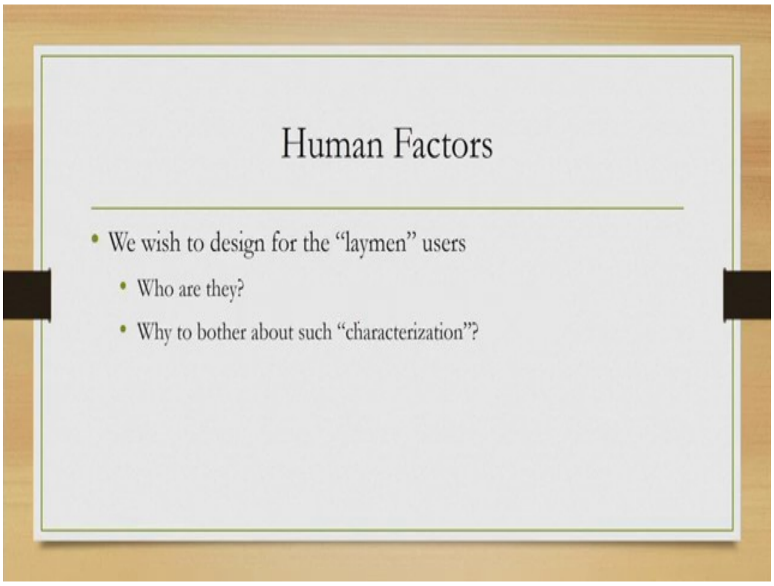
(Refer Slide Time 03:47)



Centrality of human factors

Let us take them up one by one. The first concern, the issue is the centrality of human factors.

(Refer Slide Time 03:56)



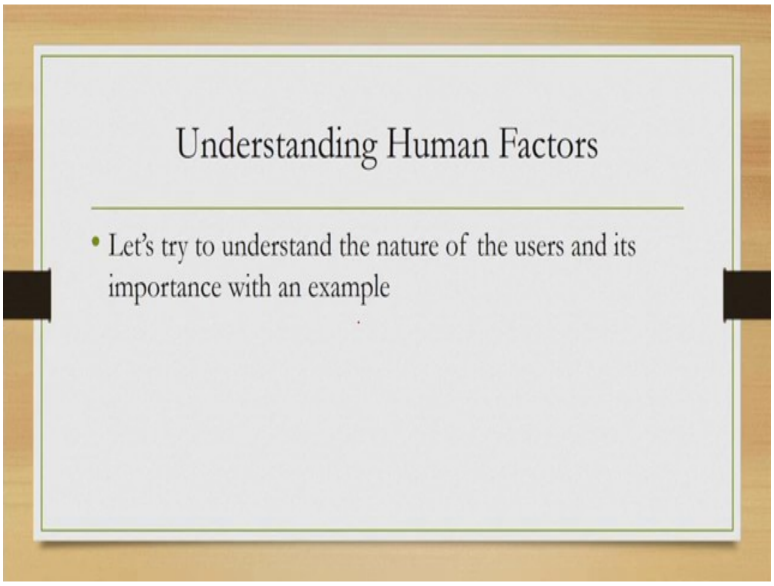
Human Factors

- We wish to design for the "laymen" users
  - Who are they?
  - Why to bother about such "characterization"?

As the definition says, the user centric products are meant for human use. So, essentially we need to take into account the human, in design of the product which we have already mentioned in the last lecture. Now, when we say we have to take into account the humans, what we specifically mean?

If you may recollect from the previous lecture, we said that the products that we are interested in are supposed to be used by laymen users who are not technology experts. Now, why we are emphasizing on this term, who are they and why this emphasis on the term laymen users are not technology expert is important?

(Refer Slide Time 04:38)

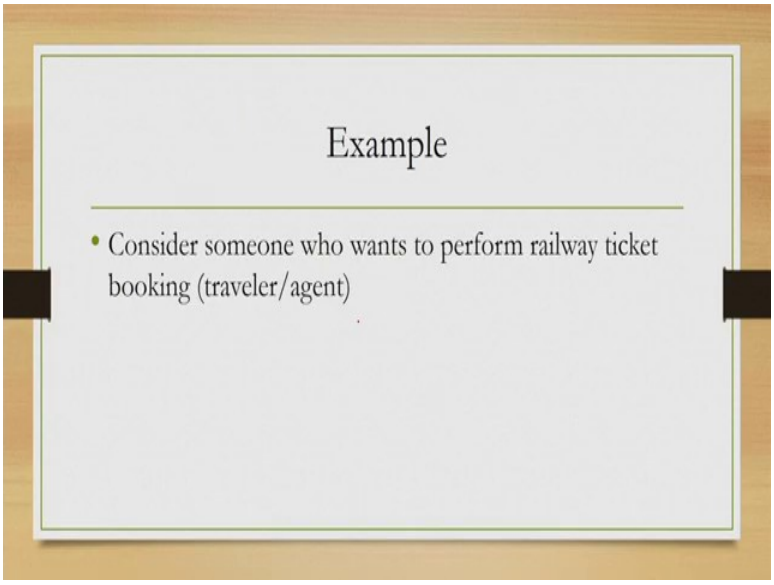


Understanding Human Factors

- Let's try to understand the nature of the users and its importance with an example

Let us try to understand these concerns with an example. Suppose, you are trying to book a railway ticket as a traveller now, what are the tasks that you are expected to perform to book a ticket?

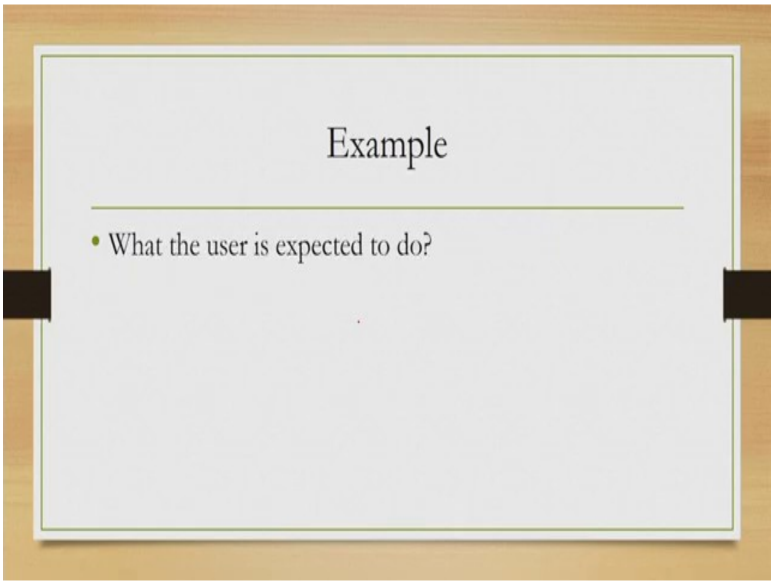
(Refer Slide Time 04:44)



Example

- Consider someone who wants to perform railway ticket booking (traveler/agent)

(Refer Slide Time 04:53)

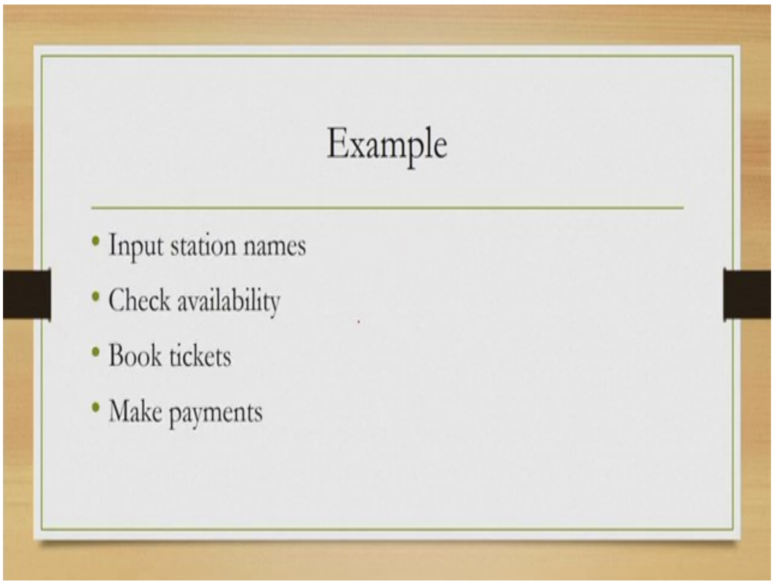


Example

- What the user is expected to do?

Many tasks actually.

(Refer Slide Time 04:55)

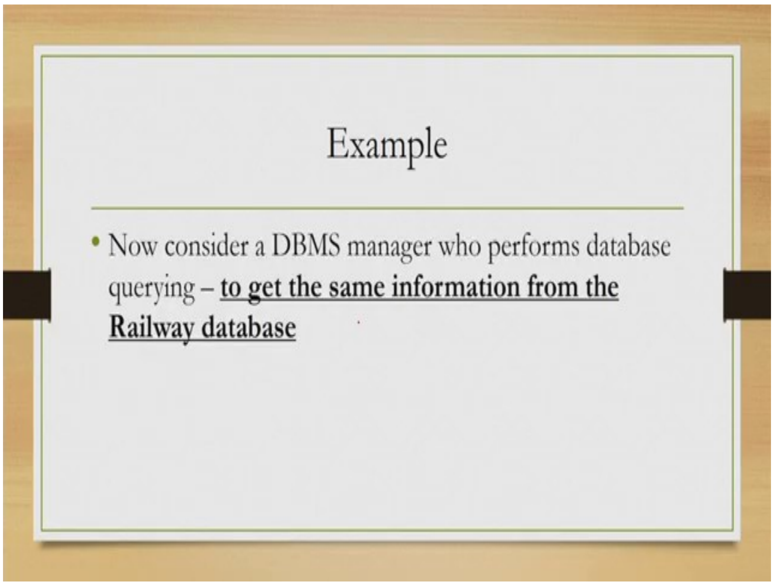


Example

- Input station names
- Check availability
- Book tickets
- Make payments

First of all you need to give the name of the stations as input between which stations you want to travel, then you probably need to check the availability of trains between those stations and then make a choice which train you want to travel by. Then you have to book tickets again that requires the checking of ticket availability in that particular train, coach, berth, if you have some preference of course, and finally, you need to make payments. So, these are the things you are supposed to perform to book a ticket for your travel and you do it with a or you expect an interface to help you do it.

(Refer Slide Time 05:38)

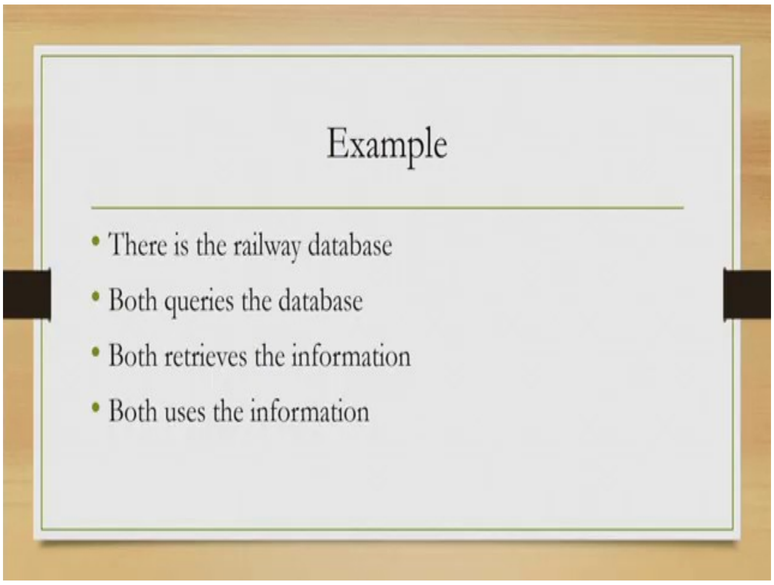


Example

- Now consider a DBMS manager who performs database querying – **to get the same information from the Railway database**

Now, consider a DBMS or database manager who is asked to write few queries to retrieve some information. Now, this information may be related to the availability of train between stations, the availability of ticket on a train, the availability of particular berths in a train, compartment so on. Now, are there any difference between what you want to do and what the DBMS database manager wants to do?

(Refer Slide Time 06:10)

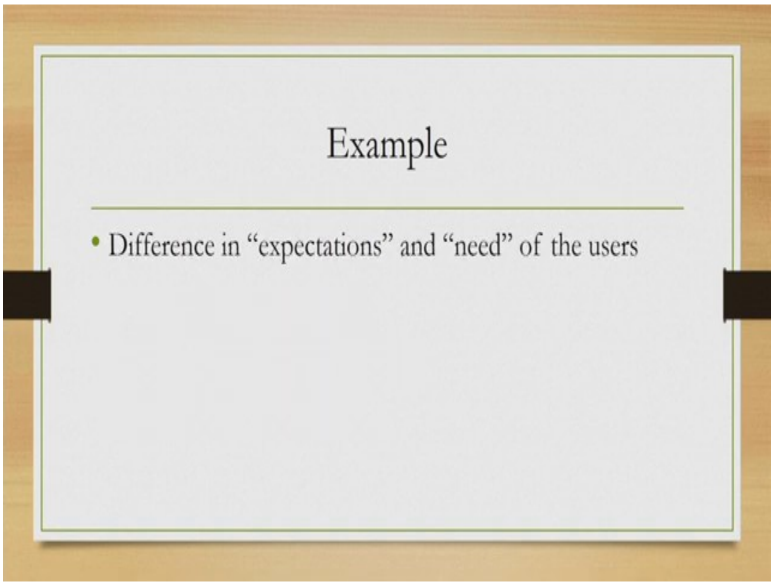


Example

- There is the railway database
- Both queries the database
- Both retrieves the information
- Both uses the information

Technically, both of you are doing the same thing and that same thing is that there is a railway database, from that database you want certain information. So, you were querying the database and you are fetching the information, based on the information you are performing some action or you are using the information.

(Refer Slide Time 06:28)



Example

- Difference in "expectations" and "need" of the users

But then what is the difference between you as a traveller booking a ticket and for which you need certain information and a database manager getting the same information in a different way, by querying the database directly. Now, the difference lies in the needs and expectations of the user.

So, in your case you expect that there is a quote unquote simple interface, which will allow you just to input the station name or which will display to you in a way you understand the availability of tickets or the trains that are running between those stations or it the availability of berths. And, also there is some interface which help you make payments in an easy way, that type of expectations are not there for a database manager. He or she does not need that quote unquote simple interface to get the information.

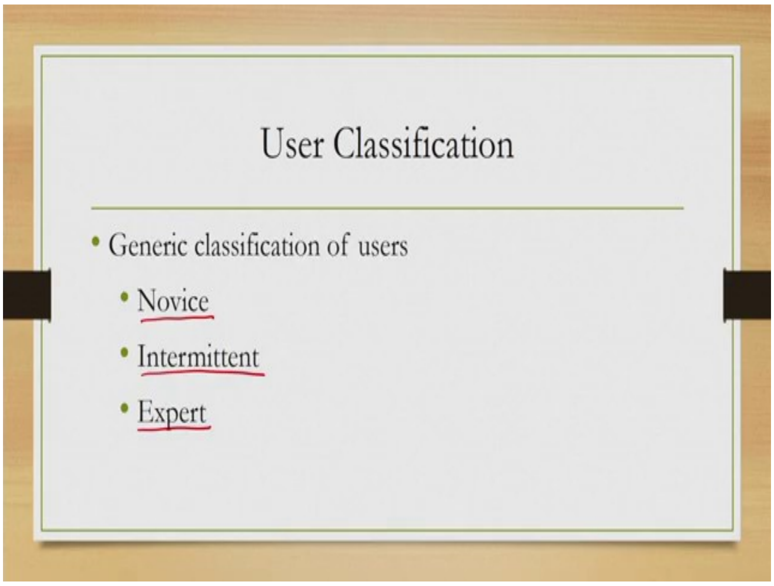
(Refer Slide Time 07:28)

User Classification

- We need to "know the users" in order to design

So, the moral of the story is when you are trying to design for somebody, some user, we need to know for whom we are designing so that we have some idea of the needs and the expectations of the user. So, we need some user characterization, some user classification.

(Refer Slide Time 07:51)



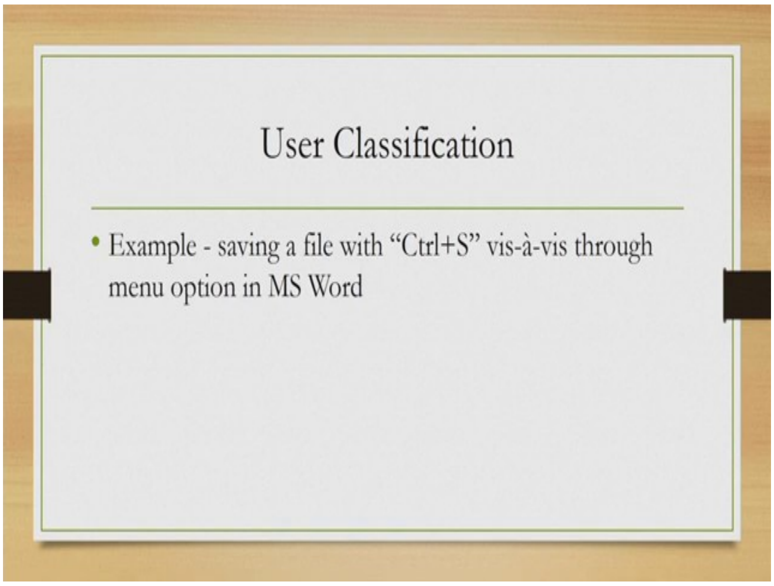
User Classification

- Generic classification of users
  - Novice
  - Intermittent
  - Expert

There is a broad classification of user we can make use off. According to this broad classification, any user belongs to one of the three classes; Novice, Intermittent and Expert. Here, note that when we are using the term user we are again emphasizing on the term Laymen users, who are not technically sound or at least are not aware of the particular technology behind the system he or she wants to use.

Now, those users we can divide into three groups: novice, as I said novice, intermittent and experts. Now, novice users are those who are using the system for the first time, intermittent users are those who use the system very occasionally not frequently and expert users are those who knows everything of the system and use the system frequently. But again, I want to emphasize that here by using the term knows the everything about the system referring to the interface not the technology aspect of the system.

(Refer Slide Time 09:02)



User Classification

- Example - saving a file with “Ctrl+S” vis-à-vis through menu option in MS Word

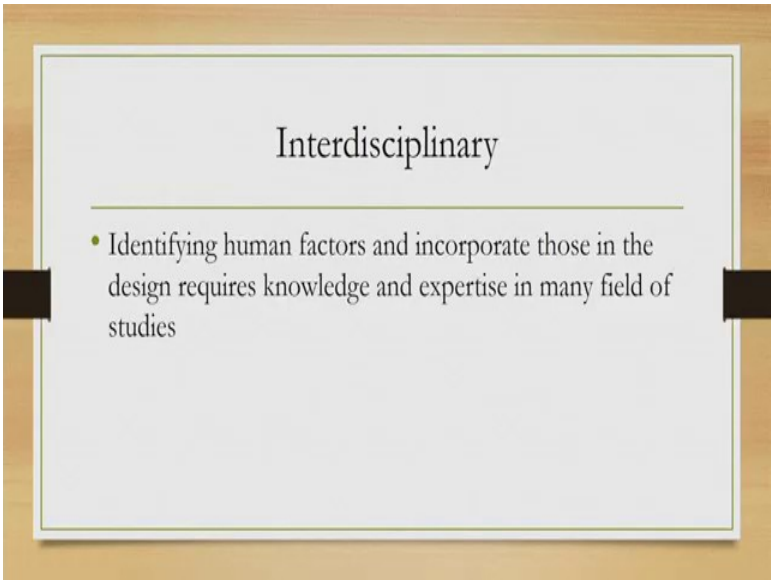
As an example let us consider one design, which we are familiar with. So, all of you probably have used some word processor system popular one is MS Word, which allows you to create documents, write texts. Now, you want to save your text what you will do? In most of the word processors that we use there are two options provided, two ways we can do this; one is menu based option. So, we go to the file menu, then there is save option and we use the save option, we click on it and the file gets saved, the other one is what is called hotkey option, a combination of keys.

So, in case of MS-Word or word pad, we can use control S control button plus the S button on your keyboard. So, this combination of keys we usually use the term hotkeys. Now, why these two options are given for the same task? Here, our task is to save the file, but the way the system is designed we have two ways to do the same thing; one is using menu, one is using the hotkeys. Why it is so? Again, it is done based on the user classification. So, if you are a Novice user, you are not likely to be very familiar with the hotkeys, the key combinations, the shortcuts and you would like to use menu's which are designed based on a principle of recognition rather than recall.

So, from menu you can actually recognize the actions to be performed and you can do it rather than keeping in mind the particular hotkey combination that are required, but if you are an expert or frequent user, then you will prefer the hotkey combination as that is the faster option come between the two. So, menu based option although, it is easier to perform for a Novice user. It takes more time than a hotkey option. So, that is why these two options are kept depending on who the user is.

Our next issue is which discipline this area belongs to? Now, as I said we need to know or incorporate human factors.

(Refer Slide Time 11:26)

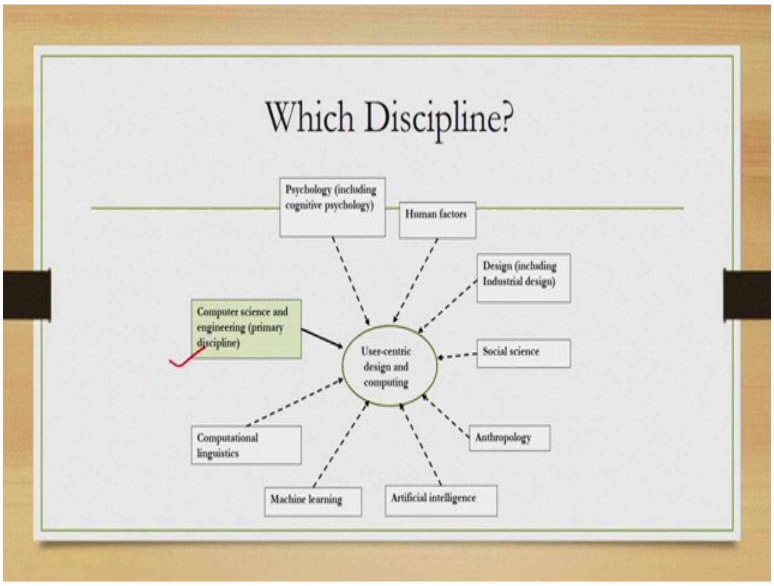


Interdisciplinary

- Identifying human factors and incorporate those in the design requires knowledge and expertise in many field of studies

Now, human factors is a broad term and it requires knowledge and expertise from different areas of study. So, this broad area of user centric design or user centric computing actually is a combination of multiple sub areas or it is an interdisciplinary field with inputs taken from many disciplines.

(Refer Slide Time 11:46)



Which Discipline?

Psychology (including cognitive psychology)

Human factors

Design (including Industrial design)

Computer science and engineering (primary discipline)

User-centric design and computing

Social science

Computational linguistics

Anthropology

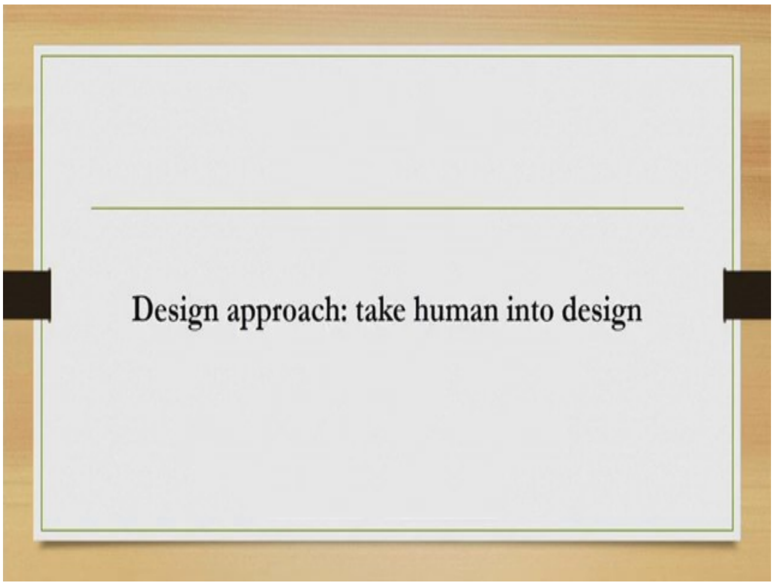
Machine learning

Artificial intelligence

So, let us see few of those. So, in it psychology including cognitive psychology, we need a discipline known as human factors, we need expertise in discipline such as design, industrial design, then social science, anthropology, machine learning, artificial intelligence, computational linguistics and computer science. There are many more which are not that prominent as compared to these disciplines

Now, among them according to ACM SIGCHI the computer science and engineering is the primary discipline, others are supporting discipline. So, if you want to master this area, computer science and engineering is probably the one you should master first and other you can use knowledge from other disciplines, for making the products more user centric.

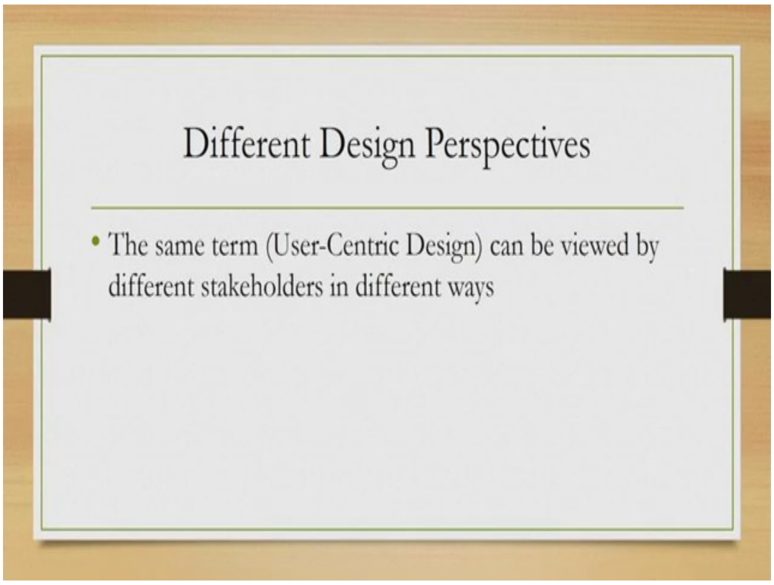
(Refer Slide Time 12:42)



Design approach: take human into design

Our third issue is the approach, design approach which approach we should take of course, our approach is to take human into the design. So, we should have a design approach which takes the human into the design.

(Refer Slide Time 12:59)

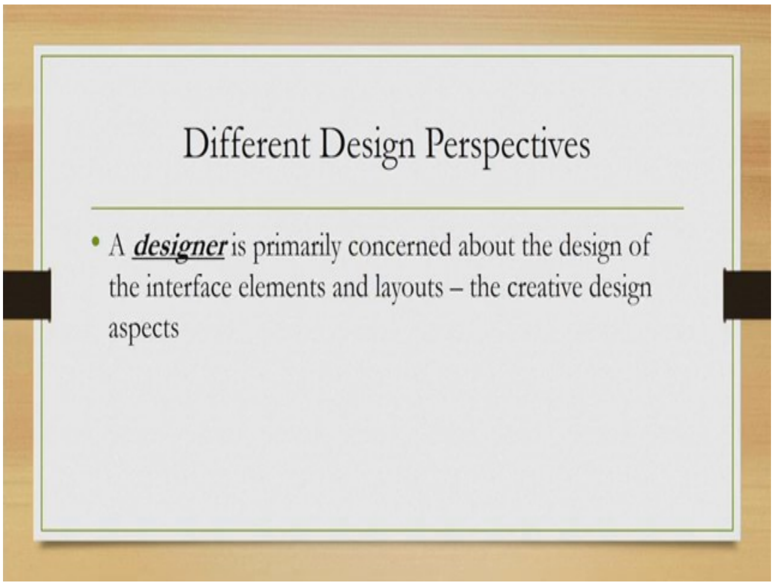


Different Design Perspectives

- The same term (User-Centric Design) can be viewed by different stakeholders in different ways

Now, this actually is something we should be very careful about, because when we talk of design actually, it refers to different things to different people depending on who is our target audience.

(Refer Slide Time 13:11)

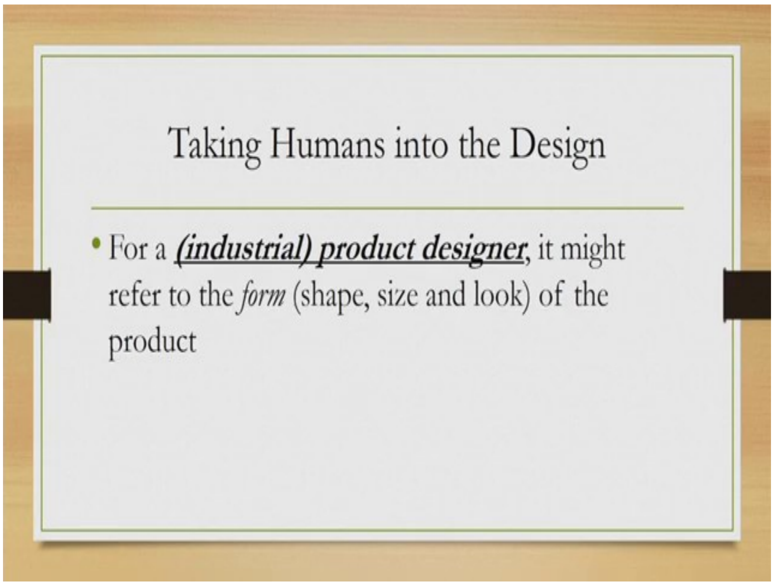


Different Design Perspectives

- A ***designer*** is primarily concerned about the design of the interface elements and layouts – the creative design aspects

So, for a designer the term user centric design may refer to design of elements and layouts particularly the creative design aspects.

(Refer Slide Time 13:19)

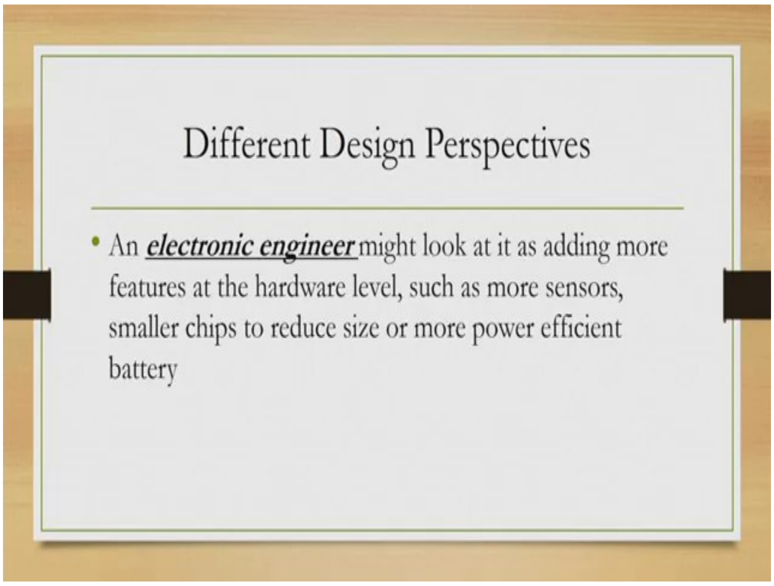


Taking Humans into the Design

- For a ***(industrial) product designer***, it might refer to the *form* (shape, size and look) of the product

For an industrial product designer it may mean something different, it may mean the form factor of the product related to shape, size, look.

(Refer Slide Time 13:33)

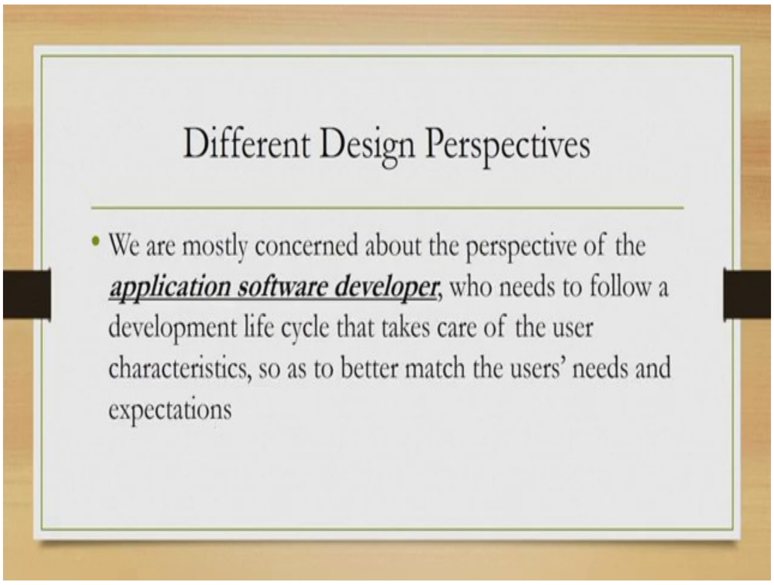


Different Design Perspectives

- An ***electronic engineer*** might look at it as adding more features at the hardware level, such as more sensors, smaller chips to reduce size or more power efficient battery

To an electronics engineer the term user centric design may indicate completely different things. The electronic engineer may be more concerned about the hardware features, the features at the hardware level, such as more sensors, reduced chip size, batteries which are more efficient; these are the things that are of primary concern to an electronic engineer.

(Refer Slide Time 13:55)

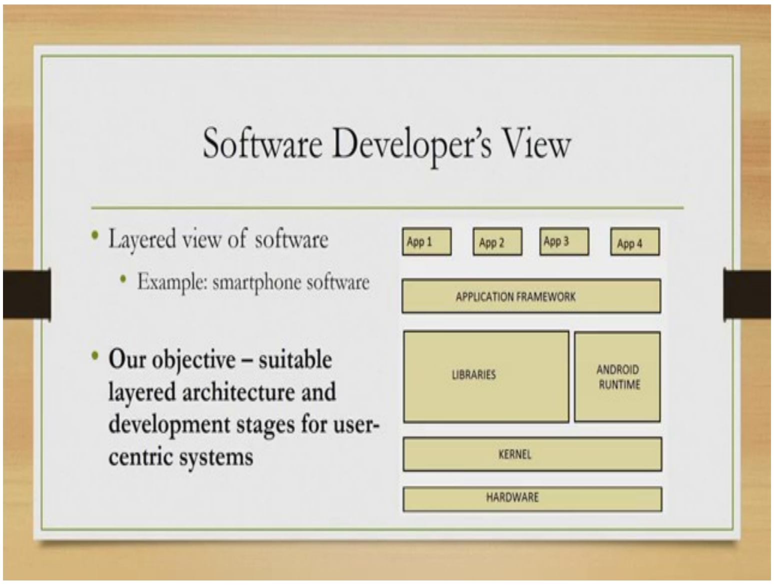


Different Design Perspectives

- We are mostly concerned about the perspective of the ***application software developer***, who needs to follow a development life cycle that takes care of the user characteristics, so as to better match the users' needs and expectations

Now, our concern in this course at least is the concern of an application software developer. So, we will basically, be focusing on the design of application software for interactive systems and so, when we refer to user centric design, we will refer to the design of the software for that product. Now, an application software developer needs to follow a development lifecycle to build the software and in this lifecycle we have to incorporate the user's needs and expectations.

(Refer Slide Time 14:28)



Software Developer's View

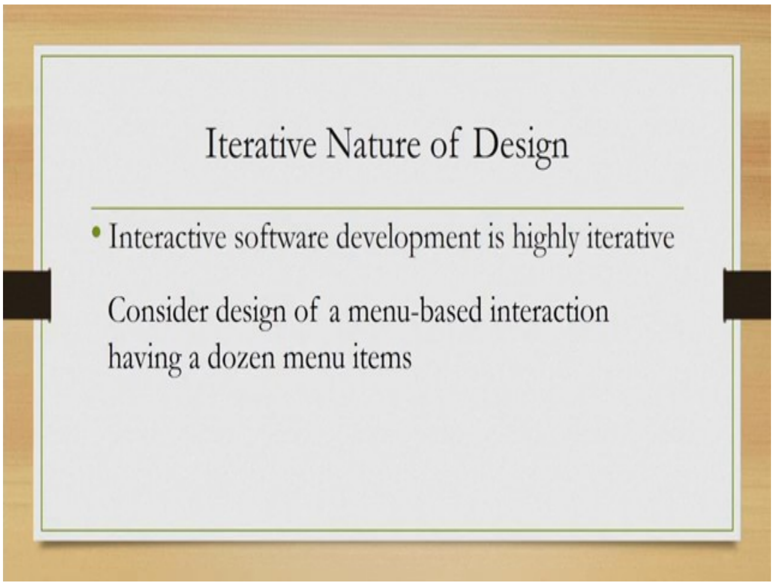
- Layered view of software
  - Example: smartphone software
- Our objective – suitable layered architecture and development stages for user-centric systems

One way to understand this view of the application software developer is to consider layered software designs. For example, the android software that are very common in the smartphones, we can view it as a layered system, different layers are designed to perform different activities as shown in this image. So, at the bottom we have the hardware layer, then we have the kernel layer, the core of operating system, then runtime and libraries are there in the intermediate layer, then one framework and on top of it a different applications are running.

So, this is one way of viewing the software and when we are talking of the view of application software, our objective is basically something like this. We need to have a suitable layered architecture that takes into account the users needs and explicitly. In other words we need to have a specific layered view of the user centric software and our final concern is model based design.

So, as you know when we are talking of design of a software, some of you may have done the software engineering course, as part of your undergraduate program and you know that software development is an iterative process, lots of iterations are there as an example, just to refresh our knowledge let us consider the design of a menu based interaction, where we have a dozen menu items, twelve menu items.

(Refer Slide Time 16:01)



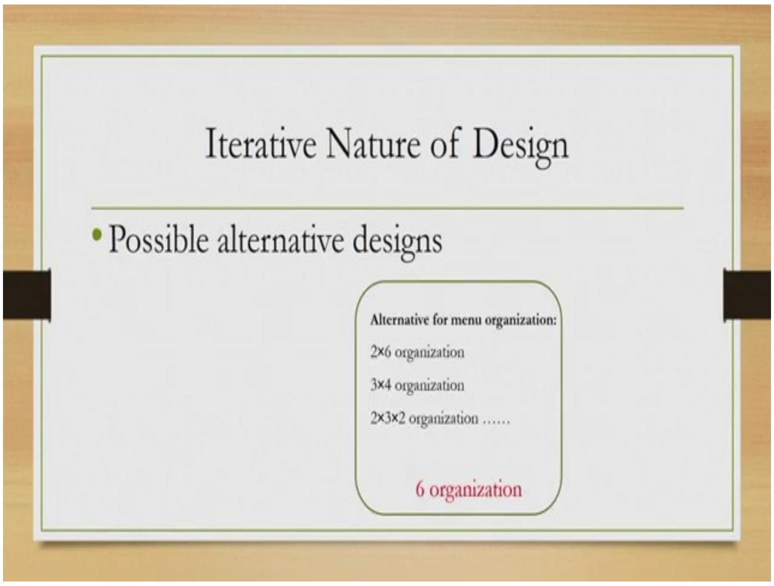
Iterative Nature of Design

- Interactive software development is highly iterative

Consider design of a menu-based interaction having a dozen menu items

So, how to design this menu?

(Refer Slide Time 16:08)



Iterative Nature of Design

- Possible alternative designs

Alternative for menu organization:

2×6 organization

3×4 organization

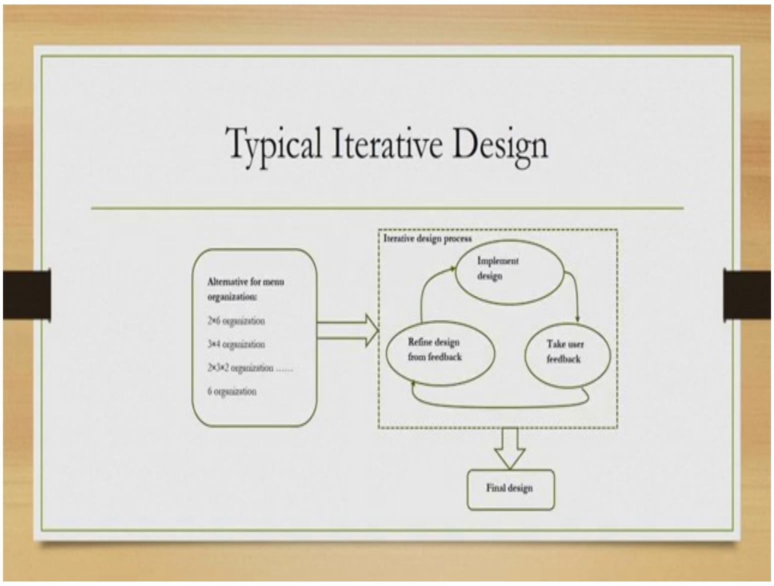
2×3×2 organization ……

6 organization

We can have many possible alternative designs. Here, we can have six possible organizations; one organization can be two top level menu options, under each top level menu option we have 6 sub menus items. Another possible organization can be two top level menu items, under each top level menu item we have in one case; we have three sub menu items in one case, we have four sub menu items yet another organization can be that we have three top level menu items under one top level item, we have two sub menu items, under another second top level item we have three sub menu items and under the third top menu item we have two sub menu items.

In this way we can have many alternative designs, alternative ways of organizing the menu items. Now, how to choose the right alternative, how to choose, which one is best for my purpose?

(Refer Slide Time 17:06)

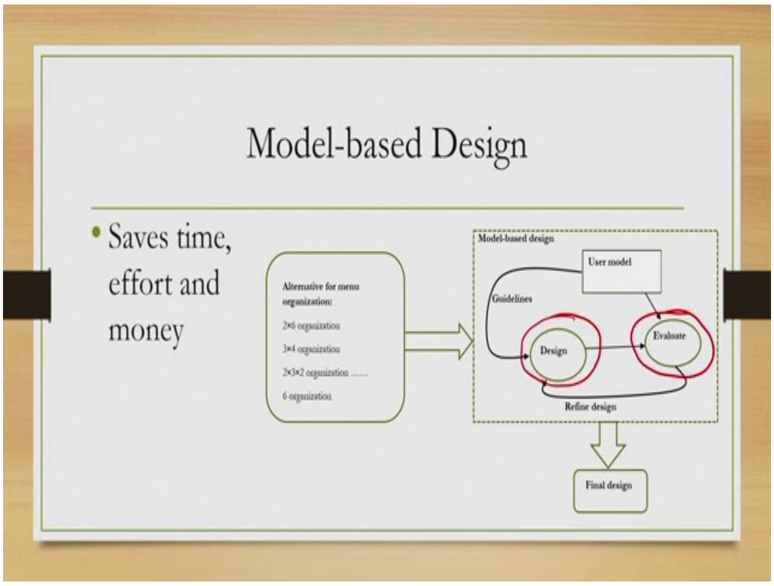


Typical Iterative Design

So, typically in software engineering what we do? We basically go through an iterative cycle. So, first we implement our design, then we need to take feedback from users, based on the feedback we need to refine our design, then implement the refined design, then again take the feedback and this loop continues till we are satisfied with the feedback and decide that no further refinements are necessary.

Of course, this is a very simplified view of the actual iterative process which we will be discussing in another lecture later, but the simplified view is sufficient for our present purpose which is just to show the nature of iteration. So, this cycle continues till we arrive at a final design and typically the cycle is time consuming and along with time it needs lots of effort in implementing design, collecting feedback, refinement of the design and that also involves money.

(Refer Slide Time 18:18)



Model-based Design

- Saves time, effort and money

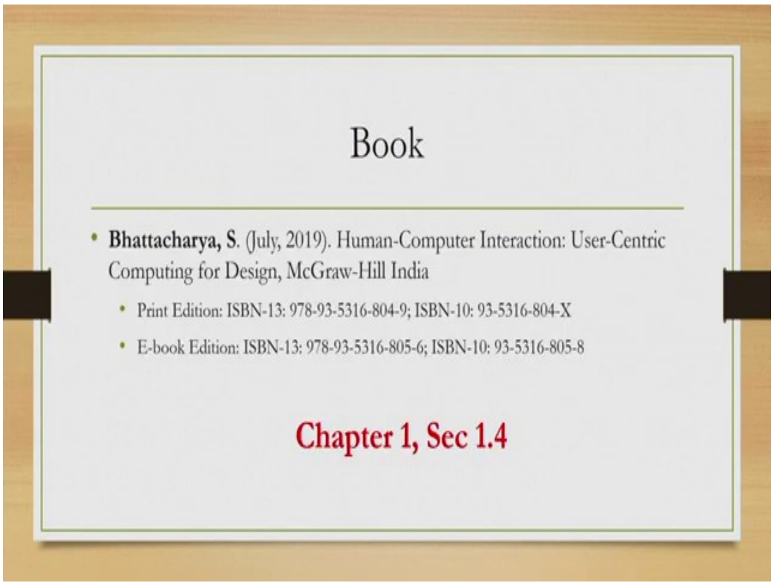
So, you need to employ manpower that is costly and the longer the manpower is engaged the money increases. Now, some elements of this design cycle, we can actually automate by what we can call model based design. So, we can have some user models these models can be utilized for designing the product, at least initial designs as well as for evaluation without always asking some users to come and evaluate or give feedback we can employ these models for evaluation.

Now, the cycle looks something like this that we have these models the model actually can be used for design as you can see this arrow. Now, there we can make use of guidelines which is one way of representing a qualitative model of the user then these guidelines can be utilized for design and once this design is done we go for evolution.

Now, for evolution we can employ user models typically those called computational models of the user, then based on the evaluation results we refine the design, this of course, involves efforts as before and then the cycle continues, but here the time it takes to complete this cycle is expected to be less, because certain aspects are automated. So, the evolution will take less time compared to the previous cycle and design refinement is the one which will take time, but overall time will reduce which entails the reduction in effort and cost.

So, this is one approach in user centric design, which is a challenge. On the one hand we want to reduce the time effort cost, but the challenge is how to model the user, how to use the model in the design, whether the models will be sufficient for evaluation and all these issues.

(Refer Slide Time 20:23)



Book

- **Bhattacharya, S.** (July, 2019). Human-Computer Interaction: User-Centric Computing for Design, McGraw-Hill India
  - Print Edition: ISBN-13: 978-93-5316-804-9; ISBN-10: 93-5316-804-X
  - E-book Edition: ISBN-13: 978-93-5316-805-6; ISBN-10: 93-5316-805-8

Chapter 1, Sec 1.4

Now, whatever I have discussed today are taken from this book particularly, section 1.4 of chapter 1 of the book.

Thank you and goodbye.