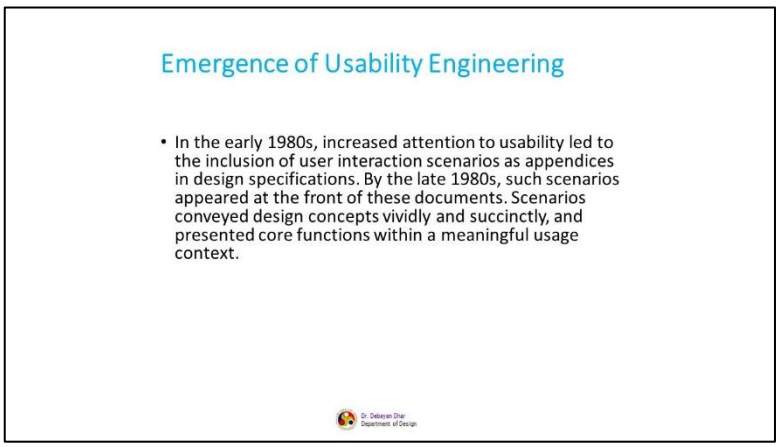
Usability Engineering
Dr. Debayan Dhar
Department of Design
Indian Institute of Technology, Guwahati

Module - 02
Lecture - 06
Usability In Software Development

Welcome to lecture 6 of module 2. In this lecture, we are going to continue our discussion on the Emergence of usability. We are also going to discuss on the structure or the framework, which gradually started emerging during the late 90s that led to the entire discipline of Usability Engineering and user experience in this lecture. So, let us begin.

(Refer Slide Time: 01:20)



So, in early 1980s, there is an increased attention to usability and this led to the inclusion of user interaction scenarios as appendices in design specifications. So, we are here discussing about a document, a structure format which in those times software development organizations used to create and in those document in those documents, in the late 1980s, user interactions scenarios were introduced; but those were at the last of the document as appendices.

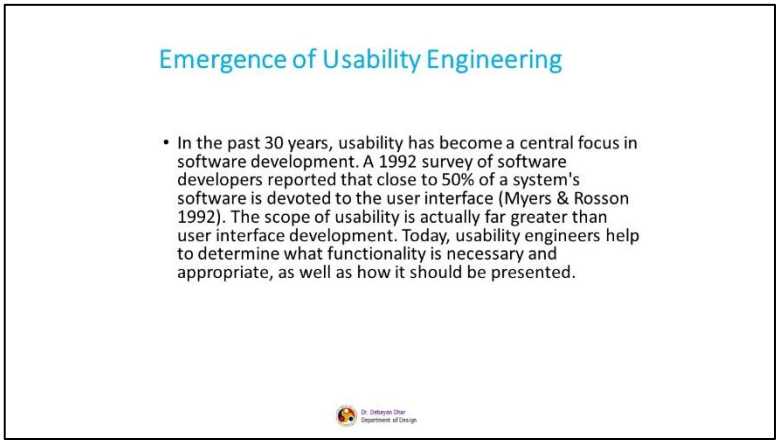
By the time you know early 1990s or late 80s such scenarios appeared at the front of this document. So, you can see migration or the location of these scenarios from first being incorporated in this documents at the last section and then, from those last sections coming up to the early sections to the front of these documents. This shift in the location or the

positioning of user interactions scenarios in these documents convey to us the importance that these specific scenarios holds in the context of software development.

So, scenarios, essentially when we talk about scenarios, it means that the these are concept, these are a structured way of putting design concepts vividly and succinctly and they present core functions within a meaningful usage context. Now, when we talk about scenarios, we are essentially talking about usage situations, specific contexts, when a task is being performed. The core features that supports the user in completion of these tasks, a detailed description of the entire structure is what is depicted in the user interaction scenarios.

Now, it is important to understand here that for the first time, the software developers could see the importance of these interaction scenarios as they understood the software development process. They felt that with these scenarios, they are able to capture or define the usage contexts in its most vivid and succinct way. It allows them to explain the features that are used to complete the task, most importantly in that particular context and therefore, what we see as appendices in early 1980s, we now started seeing these as occupying the front pages of the documents developed by software developers.

(Refer Slide Time: 05:24)



So, in the past or the last 30 years, usability has become a central focus in software development. A survey in 1992 of software developers highlighted that close to 50 percent of a system software is devoted to the user interface. This scope of usability is actually far greater than user interface development. Today, in recent times, usability engineers help

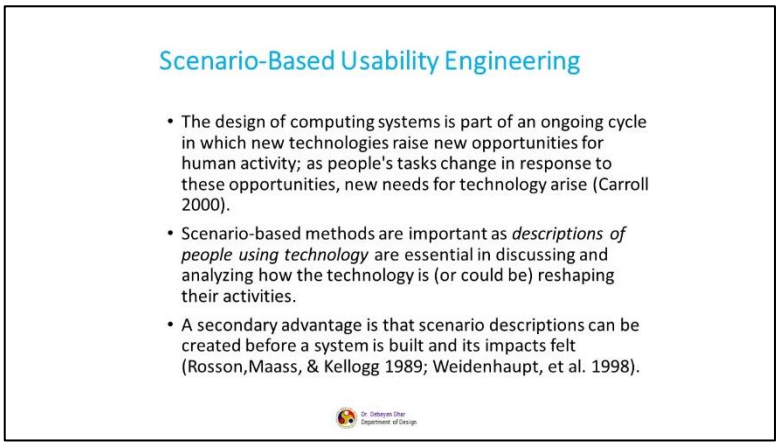
to determine what functionality is necessary and appropriate as well as how it should be presented.

It is important for us to understand the gradual realization of the importance of usability, but by software developers. As we have discussed earlier, initially designing the interface, designing the look and feel and the functions of the interface and how the interface would work in exchange of communication for exchanging communication between the humans and the systems was of paramount in importance at the early stage of development of graphical user interface.

As issues related to software usage, issues related to the crisis of mass software usage started happening. A gradual realization of importance of usable features importance of the medium being able to work in a way that is usable and satisfiable by the customers or the end users became of paramount importance. So, at this stage, what we realize or what we see is that the scope of working towards usable systems is far greater than designing that interface only. It is not limited to just designing the medium so that the experience is satisfiable.

Now, what we see? In current situations, we see usability practitioners, user experience designers. They getting involved into deciding the core functions, the core features of the software. So, they have now claimed their role in the research and development domain, where they contribute by studying users, by studying their requirements, by studying their needs and then, coming up with ways, with unique features to address those needs. So, now, they take on more active role by focusing on designing features that work really well with the customers.

(Refer Slide Time: 09:31)



With the advent of usability practitioners, designing or contributing to the design of core functions and features of software products, the advent of scenario-based usability engineering started growing. Design of computing systems is part of an ongoing cycle in which new technologies raise new opportunities for human activity; as people's tasks change in response to these opportunities, new needs for technology arise.

The advent of web revolution started from users just reading information presented to them through an interface. It moved from a static interface that presents information to an interface in which the end user can interact with. For the first time, he looks at the interface and tries to generate some content and these contents are textual sentences, pictures, videos and he is contributing to the movement which later is known as the movement for user generated content in the web.

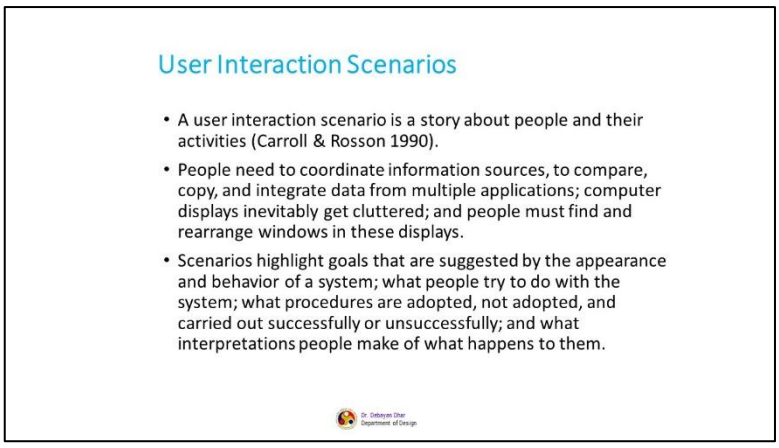
Now, this tells us the story that as and when technology was introduced, user's behavior and their activities evolved. From just reading to a list of contents, we started interacting with these contents, we started even generating content and this is a paramount shift in the nature and behavior of users over the years of new technologies that started evolving and this shift in user behavior, this change in the evolving user behavior warrants a way through which we can understand, we can appreciate and we can define the boundaries of this interaction.

Scenario based methods are important as descriptions of people using technology are essential in discussing and analyzing how the technology is or could be reshaping their activities. A secondary advantage is that scenario descriptions can be created before a

system is built; that means, even before the initial components and the structure of the systems are being planned out before even that phase, we can start delimiting or defining the scenario situations.

And therefore, what happens is this, it allows us to define the use case or the usage context to its fullest details. We are now in a position with this scenario-based detailing to understand the dimensions of the interaction, the way each decision is being taken by our user, how they are getting influenced from various information sources, while they are completing a task. This ensured the movement of scenario-based usability engineering.

(Refer Slide Time: 14:05)

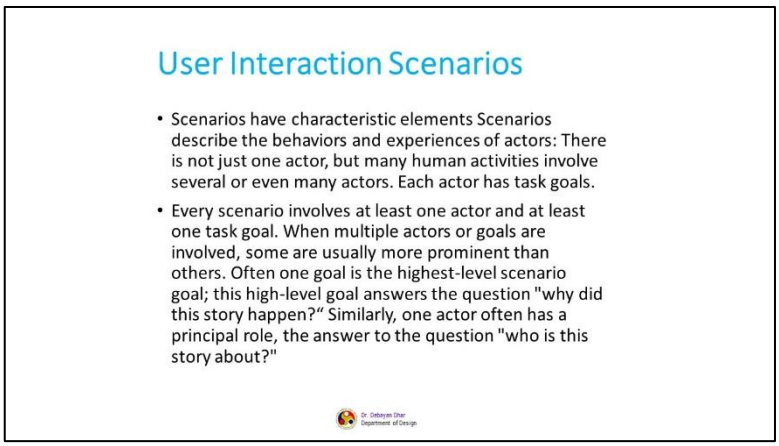


The advent of scenario-based usability engineering saw the rise of user interaction scenarios. These became an essential component to the scenario-based descriptions, we have been talking about. A user interaction scenario is a story, it is a story about people and their activities; the tasks, their behaviors in those situations. People need to coordinate information sources, to compare, copy and integrate data from multiple applications; computer displays inevitably get cluttered because of rich content and people, your customers must find and rearrange windows in these displays in order to achieve their goals and complete the tasks.

Scenarios highlight goals that are suggested by the appearance and behavior of a system; what people try to do with the system; what procedures are adopted or not adopted and which are the procedures that could be successfully or even unsuccessfully completed; what are the interpretations people make of what happens to them. These are the primary concerns that gets depicted through the user interaction scenarios.

This provides a rich quality of data, extreme depiction of the usage scenario and therefore, allows usability practitioners to define the scope of design intervention, they would probably conceive for the situations that have been presented to them.

(Refer Slide Time: 16:40)

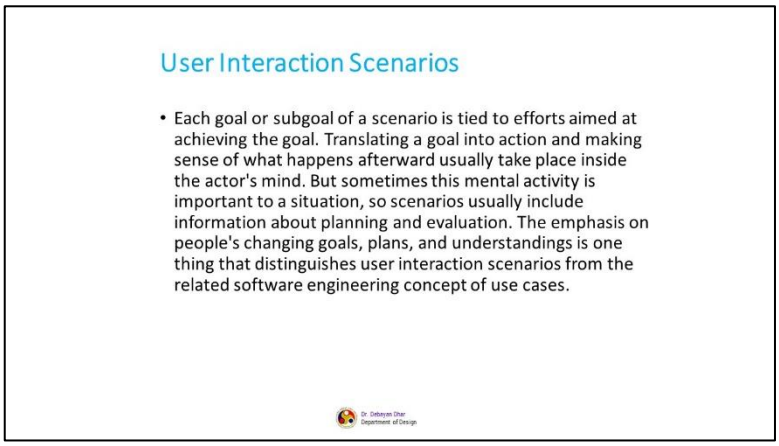


Now, these user interaction scenarios have characteristic elements. These scenarios describe the behaviors and experiences of the actors, the people, your users, who are involved in those situations which are being depicted. That is not just one actor, but many. There are many human activities involved and therefore, it involves many actors and each actor has task goals.

Every scenario involves at least one actor and at least one task goal. When multiple actors or goals are involved, some are usually more prominent than others. Often, one goal is the highest-level scenario goal; this high-level goal answers the question “why did this story happen?” Similarly, one actor often has a principal role, the answer to the question “who is the story about?”

These are the foundational objectives that allows us to do explicitly define the user interaction scenarios.

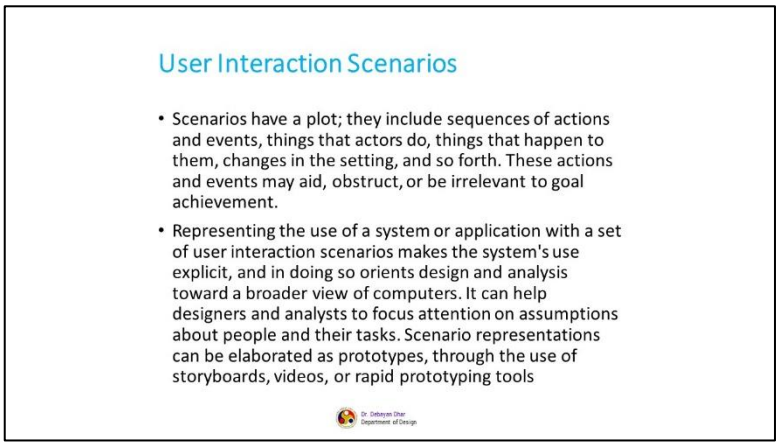
(Refer Slide Time: 18:30)



Each goal or sub goal of a scenario is tied to efforts aimed at achieving the goals. Translating a goal into action and making sense of what happens afterward usually take place in the actor's mind. But sometimes his mental activity is important to a situation, and therefore, scenarios usually include information about planning and evaluation. The emphasis on people's changing goals, plans and understandings is one thing that distinguishes user interaction scenarios from the related software engineering concept of use cases.

Many a time, we see that in the design industry among usability practitioners use cases are used and defined every now and then to highlight the user interaction scenarios. But there is a difference in terms of how user interaction scenarios were envisaged, which we have discussed and there is a difference how use cases are defined. We will take a look at these in detail. Before discussing the use cases, it is important that we learn the components of scenarios; how they are or how they have been rather classically defined.

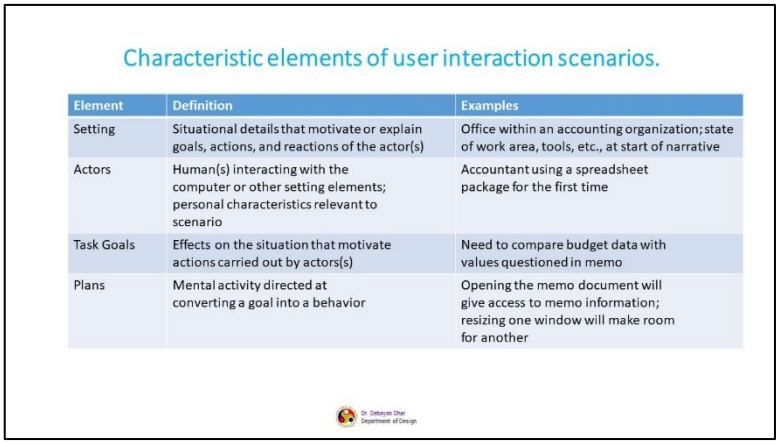
(Refer Slide Time: 20:51)



Now, scenarios have a plot and these include sequence of actions, events, things that actors do, things that happen to them, changes in the setting and so forth. Now, these actions and events may aid, obstruct or be irrelevant to goal achievement. Representing the use of a system or application with a set of user interaction scenarios makes the systems use explicit and in doing so orients design and analysis toward a broader view of computers.

It can help designers and analysts to focus attention on assumptions about people and their tasks. Scenario representations can be elaborated as prototypes through the use of storyboards, videos, or rapid prototyping tools.

(Refer Slide Time: 22:08)



These are some of the characteristic elements of user interaction scenarios. We have the elements like setting, actors, task goals, plans. Now, settings; by settings, we mean situational details that motivate or explain goals, actions and reactions of the actors.

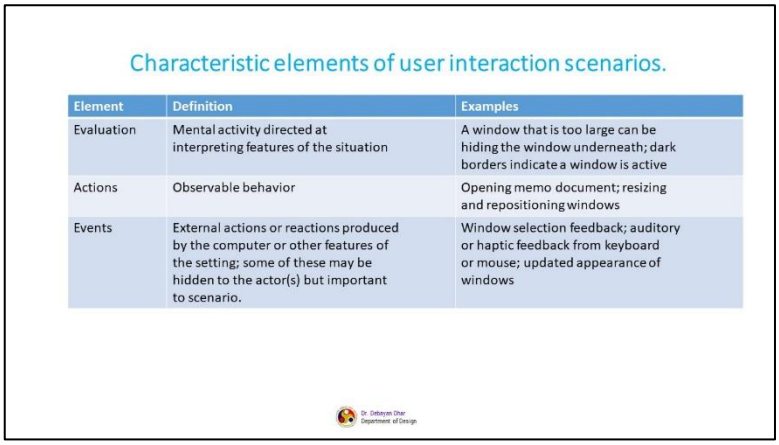
Examples, office within an accounting organization; state of work area, tools, etcetera, at start of the narrative. That is what we define the narrative. Even if you see a movie, you would see that it starts by defining the narrative in terms of the situation at which the story is being unfolded.

Then, we have actors. Actors are humans interacting with the computer or other setting elements; personal characteristics relevant to the scenario. Example, accountant using a spreadsheet package for the first time. Here, you see there is an accountant and he have a goal to use the spreadsheet and he is using it for the first time.

Then, we have task goals. These are effects on the situation that motivate actions carried out by actors. Examples like need to compare budget data with values questioned in the memo or in in some files. Then, we have plans. Mental activity directed at converting a goal into a behavior. Examples, opening the memo document. It will give access to memo information; resizing one window will make room for another. These are some of the examples of plants-based elements.

Then, we have element that is evaluation. Mental activity directed at interpreting features of the situation. Examples a window that is too large can be hiding the window underneath; dark borders indicate a window is active. You can see when you start sharing some kind of windows through online sessions, you would see how it is being highlighted.

(Refer Slide Time: 24:55)

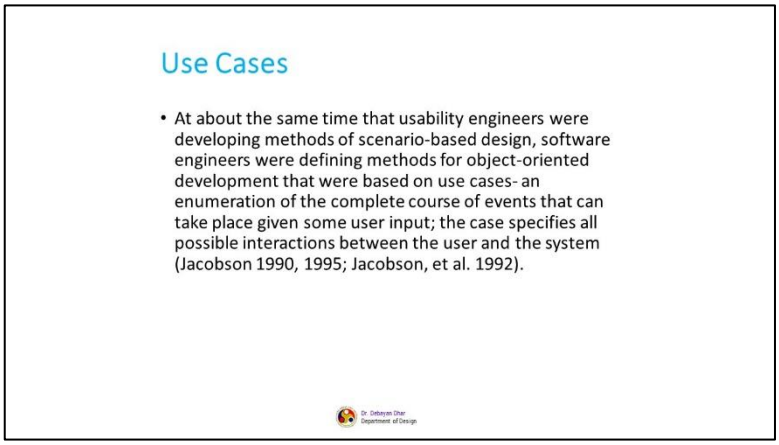


Elements like actions, observable behavior. So, actions means behavior that we can observe. Opening a memo document; resizing and repositioning windows and finally, we have events. Events are external actions or reactions produced by the computer or other

features of the setting; some of these may be hidden to the actors, but they are important to scenario.

Examples like windows selection feedback; auditory or haptic feedback from keyboard or mouse; updated appearance of windows. These are some of the characteristic elements of user interaction scenarios.

(Refer Slide Time: 25:48)



Now, let us start discussing about use cases and see how they are different from user interaction scenarios. So, now at the same time that usability engineers were developing the methods of scenario-based design, which we have; which we have just discussed. Software engineers were defining methods for object-oriented development that were based on use cases- an enumeration of the complete course of events that can take place given some user input; the use the case specifies all possible interactions between the user and the system.

In object-oriented analysis and design, use cases have the same feature as user interaction scenarios. They decompose a service requested of a system into series of inputs and system responses. But there is a difference also. One difference is that a use case is more generic; it is general and it includes all possible responses to an input, all the possible responses. If a user requests say cash withdrawal, the user's balance is first checked and the response depends on the result of that check. Thus, a scenario can be seen as one instance of a use case.

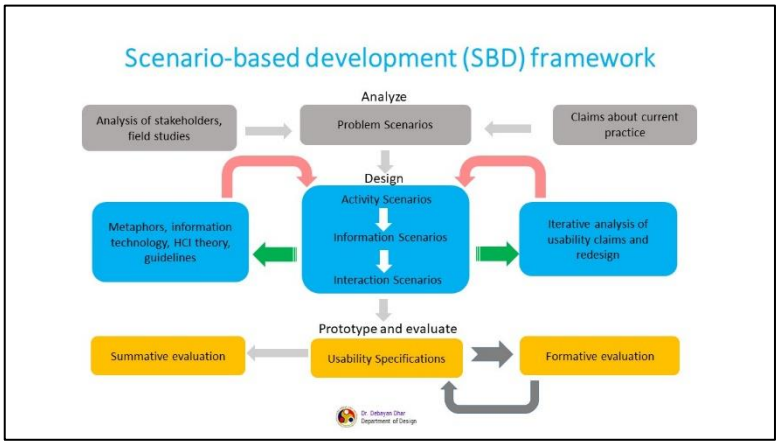
While use case highlights all possible responses to an input on, a user interaction scenario is only one instance of that use cases. It specifies an execution threat for a particular

starting state and a set of events. Another difference between use cases and user interaction scenarios is the content included in the episode. Use cases are intended to be complete description of what a system will do. But user interaction scenarios specify functionality, but always in the context of use. This is important.

The context of use is of paramount importance in scenario-based descriptions or user interaction scenarios; while the use cases are generic, they are not dependent on a specific context of use. Scenarios focus less on completeness of the coverage, directing attention instead to the design rational and possible side effects of user interaction scenarios.

So, therefore, many a time, this debate is always there whether while the software development process is going on, the focus should be to define the use cases or should the focus be to enlist all the possible user interaction scenarios. And there is this debate between the software developers and the designers, the usability practitioners; one way to integrate the two methods is to develop use cases as a functional specification and of the user system exchanges and right scenarios that raise and consider the usability implications of these exchanges.

(Refer Slide Time: 30:12)



The gradual realization of scenario based descriptions, user interaction scenarios and the onset of use cases started the movement of scenario based development framework. Now, we know that it is the correct enlisting of the situations, the authentic enlisting of situations that are of paramount importance which will guide the team, the development team, the design team to define and develop the software accurately that meet the requirements of the user.

In order to do that a scenario-based development framework emerged during these situations. And what you see here is the framework that was being adopted by the design community, the usability practitioners to ensure that the scenario-based descriptions and the use cases are accurately defined and then, design interventions are conceived. What you see here is a structure a structure of three step process; the first step it focuses on analysing problem scenarios and then, it focuses on the design; after the design interventions are conceived, it then focuses on prototyping and evaluate.

Now, it is important to understand here that analysis of stakeholders and field studies plays an important role in defining the problem scenarios. Extensive user studies in order to define the claims about the current practice is important to define the problem scenarios as well. So, these two components analysis of stakeholders and field studies and claims about current practice from user study, contributes to the development of problem scenarios and this is the analysis phase that we are talking about.

The next phase, once we are equipped with the accurate definition of our requirements, with the most accurate definition of the requirement from the analysis phase, we then move on to the design phase and in the design phase, we generally focus on three important situations. We focus on activity scenarios, we and then on information scenarios and then, we focus on the interaction scenarios right. We will discuss about each of these scenarios in detail in subsequent lectures.

But it is important for us to understand now that the design phase is influenced by two important situations; one these are influenced by metaphors, information technology, HCI theory and guidelines. So, while designing, all these knowledge influences the designer to accurately design an intervention for the requirement.

Similarly, iterative analysis of usability claims and redesign; it is an iterative process of analysing the claims, the situations, the demands, focus and address of usability issue and then, constantly, quickly, getting feedback from the users and testing it out also informs the design phase.

Once the detailed design is developed and a prototype is designed, it is then when the usability testing phase is adopted. The usability testing phase depends on actually or essentially two important aspects; the one that we focus on the iterative analysis is called as the formative evaluation, the other one which is the phase, where the entire product is

conceived prototype and then, tested is called the summative evaluation. Now, this framework is the basis for the user center design approach that we are going to discuss in the subsequent lectures.

I would like to draw your attention here that while discussing the user center design approach that ensures usable products, we are going to focus on various HCI theories, we are going to focus on various principles heuristics and guidelines, we are also going to focus on foundational concepts of information technology metaphors. Because these are the principal components that would influence designers or the usability practitioners focus while they are designing an intervention for the requirements.

Here, once this part is addressed and we have a design intervention, it is now important for the usability practitioner to realize how the concepts are iteratively tested. So, how we go for formative evaluation so that the claims can be addressed and quick redesign can be initiated.

In the subsequent lectures, we will discuss about this framework, each section of the framework in detail and this would ensure that we can design and come up with a software system or application that is usable, that provides experiences that lasts for long and encourages your customers to become loyal customers of your product.