Software Conceptual Design
Dr. Sridhar Iyer
Dr. Prajish Prasad
Dr. T. G. Lakshmi
Indian Institute of Technology, Bombay

Lecture - 22
Software Development Process Recap

Since we are at the last week of the course it is important to reflect on the learnings and concepts we learnt in the previous weeks so that we can make connections between them.

Yes, I agree also making connections between the newly learned concepts and the earlier known concepts is an important exercise.

So, in week 1 we looked at how a relatively large software like Amazon is built, it is made up of several interworking modules.

You are right, and this concept is known as modularization.

(Refer Slide Time: 00:48)

Modularization is used to divide a software system into multiple independent and co working modules. These modules are capable of carrying out the desired functionalities, and they form the basic building blocks for the entire software.

(Refer Slide Time: 01:12)

Reflection Spot

Why did we learn about the various phases in software design and the software development models?

Please pause the video and write down your responses

Also, in week 1 we learnt about the different phases and the software development models. So, learners here is a question for you why did not we learn about the various phases in software design and the development models? Pause the video write your answer in the notebook, and resume.

Software development does not start immediately with programming and creating a software it begins with understanding the needs of the end user, creating a plan, verifying it and then implementing it using a programming language. After creating the software we test it and then deploy it at the customer's end.

So, software development is a systematic process of problem-solving.

Yes, you are right, the software development phases are broad and could be done iteratively and in many different ways.

So that is why it is important for software developers to know and understand these phases and processes. We also compared and contrasted software design we design in other domains, specifically architecture.

Yes, the abstract and intangible nature of software adds to the complexity in software design and development. However, there are similarities in the design processes across the domains, much like the steps to problem-solving.

Well, in week 2 and 3 we created software designed models using the FBS design framework and the UMI diagrams. How do these models map to the software development phases and the processes?

Yes, that is an important question we will discuss this in detail in the next video.