Software Conceptual Design

Dr. Sridhar Iyer
Dr. Prajish Prasad
Dr. T. G. Lakshmi
Department of Computer Science and Engineering
Indian Institute of Technology, Bombay

Lecture - 19
Software Design Quality Perspectives

In the previous video, we understood the importance of design comprehension and how experienced designers comprehend a given design. We also interacted with the VeriSIM learning environment. Verisim helped us understand how to comprehend a given design by tracing various scenarios in the design.

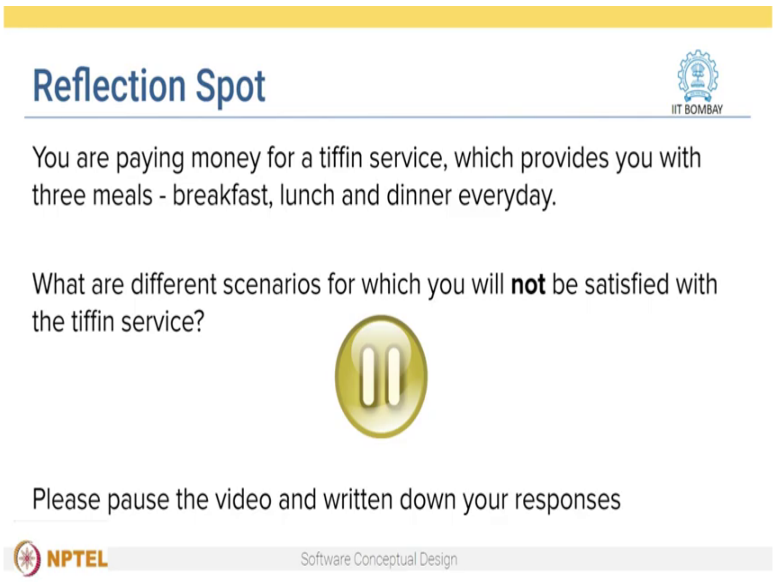
Ok. So, we have modelled created a design comprehended, and now we also have an integrated understanding of the design.

Yes, along with comprehending a design, we also have to evaluate whether the design is correct.

Design is correct in what sense? Can you help me understand this with an example?

Yes, let us take an example from a non-software context it might help us understand the concept better.

(Refer Slide Time: 01:15)



Reflection Spot

You are paying money for a tiffin service, which provides you with three meals - breakfast, lunch and dinner everyday.

What are different scenarios for which you will **not** be satisfied with the tiffin service?

Please pause the video and written down your responses

IIT BOMBAY
NPTEL
Software Conceptual Design

Let us say that you are paying money for a tiffin service which provides you with three meals, breakfast, lunch and dinner every day. So, what are different scenarios for which you will not be satisfied with the tiffin service? You can pause this video, write down your responses and then proceed.

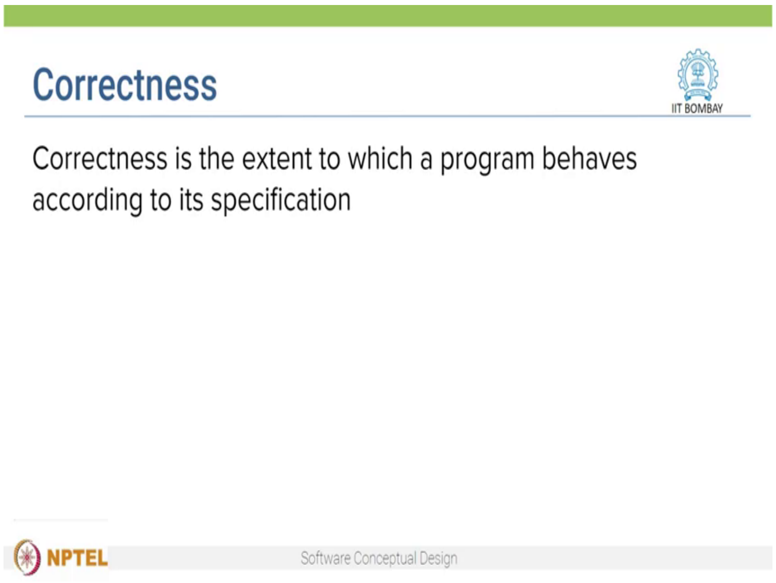
So, the first thing is that I want my food to be delivered on time, and I do not want any mix-ups happening. For example, the service should not be giving me breakfast for lunch.

Yes, exactly, so what you actually mentioned now these are the basic requirements right? The tiffin service should make sure that these basic requirements are satisfied this is known as correctness.

Ok. So, going back to the software design context, we need to ensure the correctness of the design we make sure of the design satisfies all the requirements we can say that the design is correct in some sense.

Yes, you are right. When you check for correctness, you ensure that the design satisfies all that your system should do.

(Refer Slide Time: 02:49)



Correctness

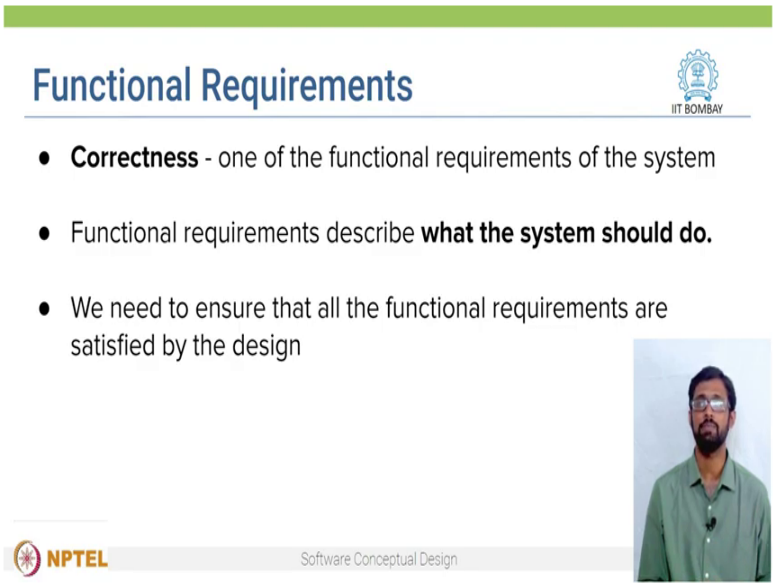
IIT BOMBAY

Correctness is the extent to which a program behaves according to its specification

NPTEL Software Conceptual Design

Correctness is the extent to which a program behaves according to its specification. Learners, this is what you did in VeriSIM as well, using design tracing, you checked whether various scenarios in the design correctly satisfied all the requirements of the automated door-locking system.

(Refer Slide Time: 03:20)



Functional Requirements

- **Correctness** - one of the functional requirements of the system
- Functional requirements describe **what the system should do.**
- We need to ensure that all the functional requirements are satisfied by the design

IIT BOMBAY
NPTEL
Software Conceptual Design

Correctness is one of the functional requirements of the system. Functional requirements describe what the system should do. We need to ensure that all the functional requirements are satisfied by the design.

So, checking if the functional requirements are satisfied is one way to evaluate the given design, but is it the only way are there other perspectives we need to consider while evaluating a given design?

Yes, there are several other ways to evaluate a software design, it is necessary to evaluate a design against various quality attributes.

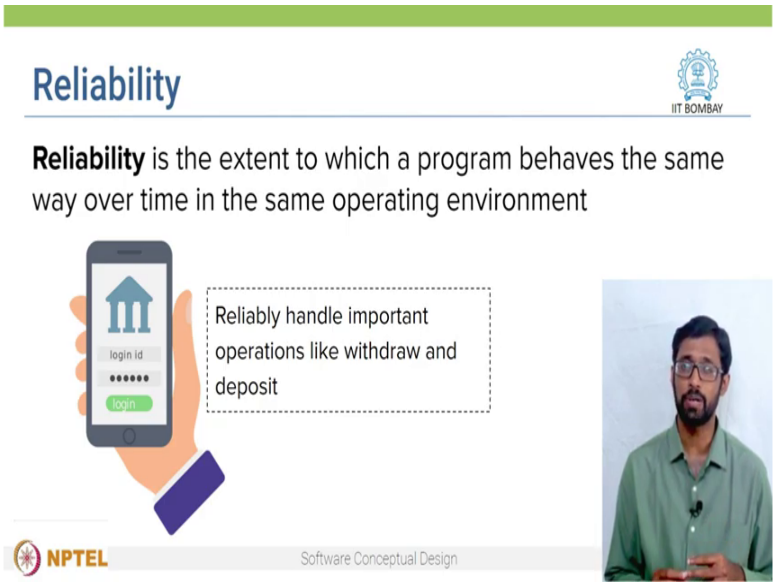
Quality attributes? can you help me understand this with an example?

Yes, let us go back to the tiffin service example. What else do you expect from the tiffin service?

Ok. Next, I want the service to be reliable for example, if I miss more than three meals a week, I would not be happy.

Exactly this is termed as reliability.

(Refer Slide Time: 04:46)



Reliability

Reliability is the extent to which a program behaves the same way over time in the same operating environment

Reliably handle important operations like withdraw and deposit

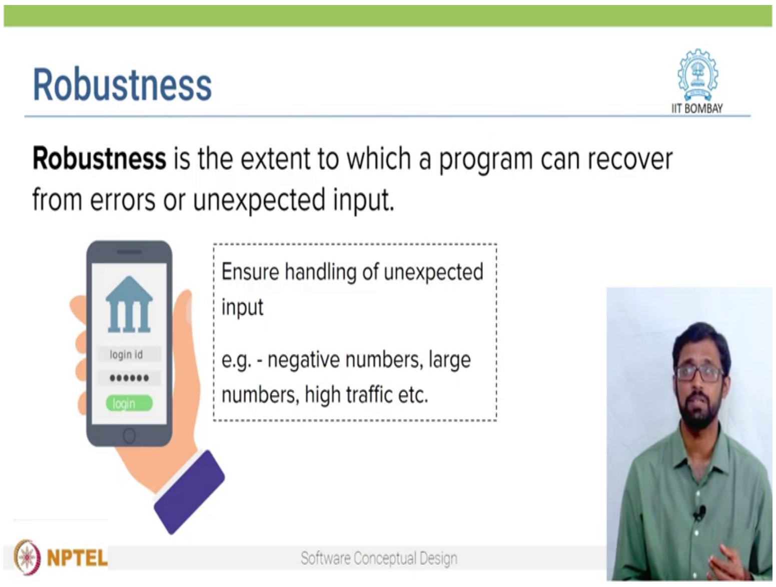
IIT BOMBAY
login id
login
NPTEL
Software Conceptual Design

Reliability is the extent to which a program behaves the same way over time in the same operating environment. In case of software, we need to ensure that it does not crash often; otherwise, users will get frustrated. For example, in a mobile wallet like Amazon Pay, reliability is even more important. We must ensure that the wallet reliably handles all operations especially important operations like withdrawal and deposit.

I also want the service to handle unexpected situations reasonably well. For example, if the main chef falls sick, is there a replacement? Is there any other option to serve the customers?

Ok. So this is similar to the quality attribute of being robust even in unexpected situations.

(Refer Slide Time: 05:59)



Robustness

Robustness is the extent to which a program can recover from errors or unexpected input.

Ensure handling of unexpected input

e.g. - negative numbers, large numbers, high traffic etc.

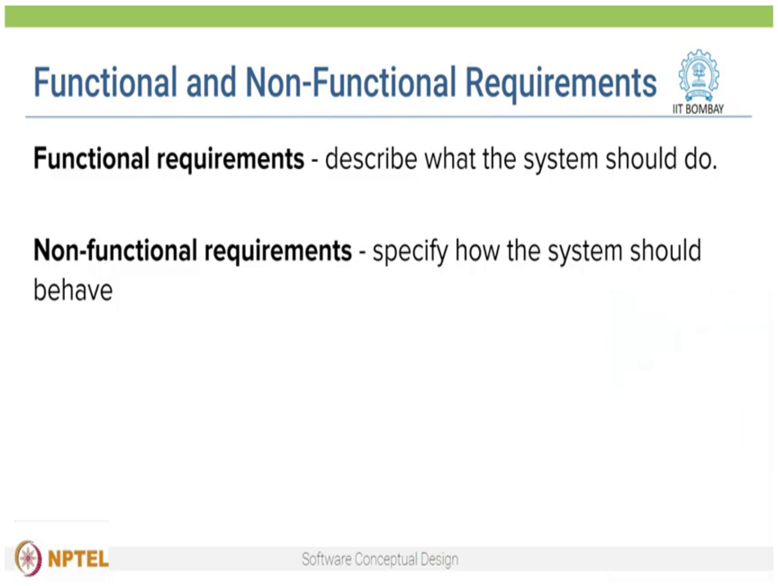
IIT BOMBAY
login id
login
NPTEL
Software Conceptual Design

Robustness is the extent to which a program can recover from errors or unexpected input. For example, in a mobile wallet system like Amazon Pay we have to ensure that the service handles unexpected input like negative numbers and large numbers and high traffic as well.

So, while checking for quality attributes like reliability and robustness, we are, in a way checking how the system is behaving and not exactly checking the system requirements. So, are these quality attributes different from the functional requirements of the system?

Yes, you are right attributes like reliability and robustness are known as the non-functional requirements of the system.

(Refer Slide Time: 06:58)



Functional and Non-Functional Requirements

Functional requirements - describe what the system should do.

Non-functional requirements - specify how the system should behave

IIT BOMBAY
NPTEL
Software Conceptual Design

Non-functional requirements essentially specify how the system should behave. For example, we looked at two non-functional requirements reliability and robustness.

Ok. So, functional and non-functional requirements have to be considered while designing, creating and evaluating the system.

Yes, you are right to summarize in this video, we discussed functional and non-functional requirements. There are other non-functional requirements such as performance portability, security, and so on. Do check out the additional resources for more information.