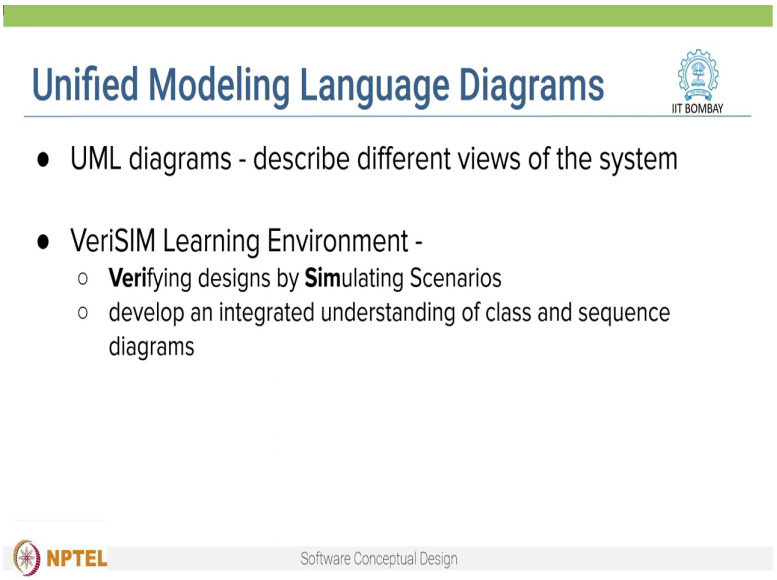
Software Conceptual Design

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

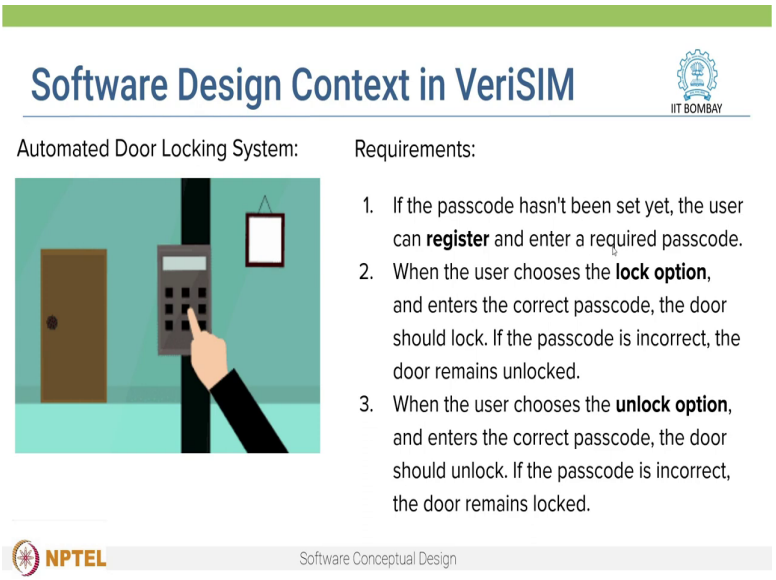
Lecture - 18
VeriSIM : A Learning environment for Comprehending Software Designs

(Refer Slide Time: 00:03)



In the previous video, we looked at the unified modelling language or the UML diagrams. We saw that UML diagrams describe different views of the system. In this video, we will be looking at the VeriSIM Learning environment. VeriSIM stands for verifying designs by simulating scenarios. Using this learning environment, you will be able to develop an integrated understanding of the class and the sequence diagrams.

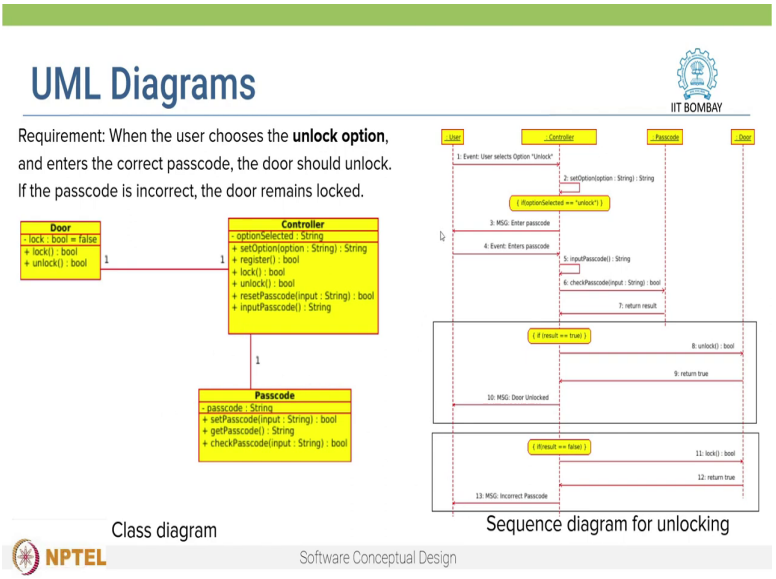
(Refer Slide Time: 00:42)



So, let us look at the software design context in VeriSIM. So, in VeriSIM, you will be introduced to an automated door locking system you will be introduced to the requirements of the system. For example, one requirement can be that the user can register himself or herself by entering a passcode, then the user can choose a lock or an unlock option. So, when a user chooses a lock option and enters the correct passcode, the door should lock.

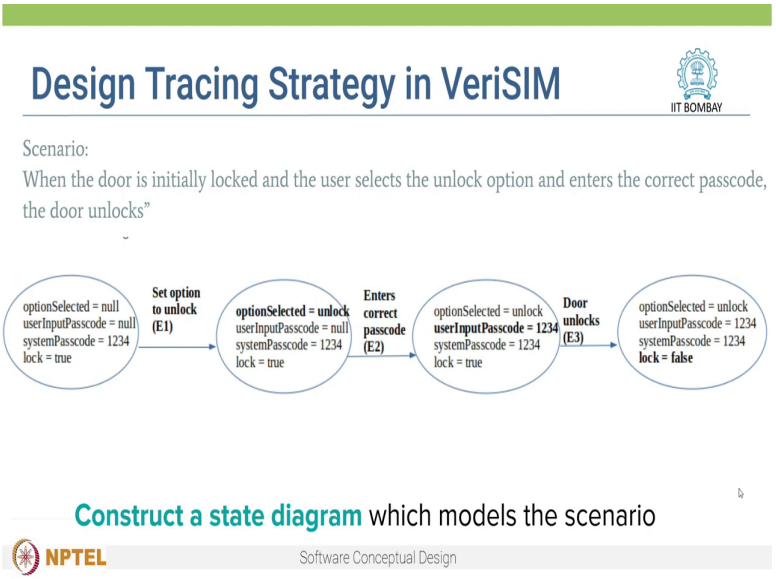
However, if the passcode is incorrect, the door should remain unlocked; a similar requirement is there for the unlock option as well.

(Refer Slide Time: 01:37)



Now, these requirements are modelled using various UML diagrams, such as the class diagram, which is shown here on the left and one of the sequence diagrams for unlocking is shown on the right. The automated door locking system has a class diagram and several sequence diagrams which model the requirements of the system.

(Refer Slide Time: 02:08)



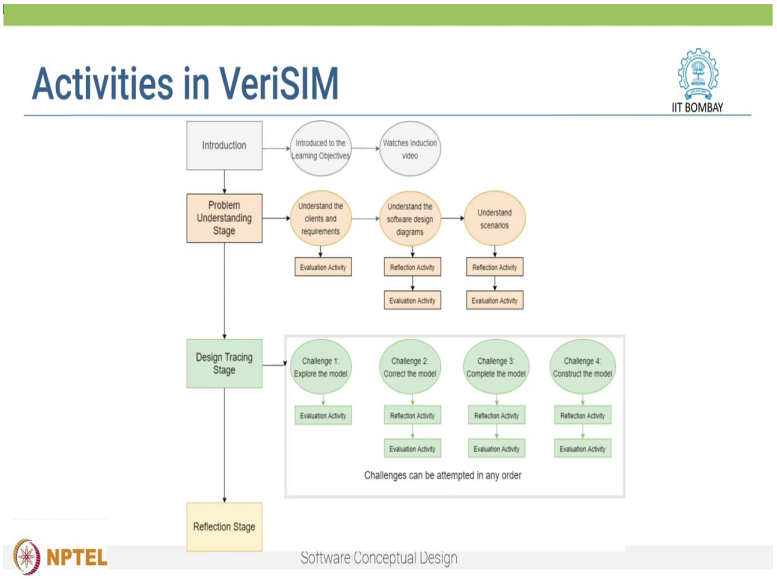
So, in VeriSIM, you will be introduced to a strategy known as the design tracing strategy. So, consider a scenario of the automated door locking system. So, let us say that the door is initially locked, and the user selects the unlock option and enters the correct passcode on doing this, the door should unlock. So, what is design tracing? In design tracing, you construct a state diagram which models a given scenario. So, in this case, this state diagram models this given scenario.

We can see that the transitions in the state diagram correspond to parts of the scenario. So, we see initially that the user selects the unlock option and then they enter the correct passcode and finally, is the door unlocks. So, these transitions correspond to messages in the sequence diagram, and based on these transitions, we see that the states which contain data variables and their values these keep on changing.

So, for example, when the user selects the unlock option, the value for the variable option selected changes to unlock. These variables are taken from the class diagram, and the transitions are messages which are taken from the sequence diagram. So, as you construct

these state diagrams, this will help you model different scenarios in the system and thereby help you develop an integrated understanding of the class and the sequence diagrams.

(Refer Slide Time: 04:14)



So, what are the activities in VeriSIM? You will be first introduced to what VeriSIM is what the learning objectives are, then you will move on to the problem understanding stage, where the requirements and the design diagrams for an automated door locking system will be given to you. The next stage is the design tracing stage, where you will attempt different challenges which will help you construct the state diagram which you saw in the previous slide and these challenges can be attempted in any order.

And finally, you will go through the reflection stage, which will help you reflect on what you have learnt in VeriSIM. After each of these activities, there are evaluation as well as reflection activities which will help you reflect on what you have learnt in that activity. So, this information is enough for you to now go and attempt VeriSIM. Happy learning.