Modeling as a Design Technique

2.0 Definition:

A model is an abstraction before building any system a prototype may be developed. The main purpose of model is for understanding of the system.

2.1 Introduction for modeling

Abstraction plays very important role in handling complexity. Abstraction means hiding detailed information and only important information is conveyed to the user. Engineers, artists build models for their designs. One can see even in the computer system. Operating system helps the user to use computer in a convenient and efficient way without knowing much about software and hardware present in the computer system.

2.2 Modeling

Designer build different kinds of models for various purposes before constructing things. For example car , airplane, pencil sketches for oil painting, blueprints of machine parts, Plan for house construction etc., Models serve many purposes

Testing a physical entity before building it

Engineers test scale models for airplanes, cars and boats in wind tunnels and water tanks to improve their dynamics. Also the use of simulation (Computer model) makes cheaper than building a complete system and correct if there is any flaw in it.

Communication with customers

Architects and product designers develop model to show their customers. Trial demonstration can be arranged.

Visualization

Storyboard of movies, television shows and advertisements show how their ideas flow.

Reduction of complexity

The human can understand if it is expressed in simple means. The model helps to bring complex things into simple.

2.3 Abstraction

Abstraction is process of masking unimportant details. For example the user see the automobile vehicle in a different way compared to a mechanic. All abstractions are incomplete and inaccurate. A good model essentially captures the important aspects of a problem and omits the others.

2.4 The three models

To capture the important aspects of the system in different viewpoints we use three models namely class model , state model and interaction model

The class model represents the static , structural , data aspects of a system.

The state model represents the temporal, behavioral , control aspects of system.

The interaction model represents the collaboration of individual objects of a system.

For example in a Software system, data structures part represents the class model , sequence operations in time represents the state model and communication among objects represents interaction model.

The three kinds of model separate a system into distinct views. The different models are not completely independent. Combination of all model results in better understanding of the system.

2.5 Class model

The class model describes the structure of objects in a system – their identity, their relationships to other objects , their attributes and their operations. The class model provides context for the state and interaction model.

Class diagrams representation used in the class model. Classes define the attribute values carried by each object and the operations that each object performs or undergoes.

2.6 State model

The state model describes those aspects of objects concerned with time and the sequencing of operations , events that mark changes, states that define the context for events and the organization of events and states.

The state model represents control i.e., the aspect of a system that describes the sequences of operations that occur, without regard for what the operations do, what they operate on, on how they are implemented.

A state diagram represents the state model. Each state diagram show the state and event sequences permitted in a system for one class of objects.

2.7 Interaction Model

The interaction model describes interaction between objects – how individual objects collaborate to achieve the behavior of the system as a whole.

Use cases, sequence diagrams and activity diagrams document the interaction model. In use case it shows the interaction between the system and outside actors. Sequence diagram show the objects that interact and the time sequence of their interactions. Activity diagrams show the flow of control among the processing steps of a computation.