Importance of design . Software design is an iterative process through which requirements are translated into a “blueprint” for constructing the software. Design is a highly creative stage in software development where the designer plans – how the system or program should meet the customer’s requirements – how to make system effective and efficient. Stages of design ,Understand the problem – Look at the problem from different angles to discover the design requirements , Identify one or more solutions – Evaluate possible solutions and choose the most appropriate , Describe solution abstractions – Use graphical, formal or other descriptive notations to describe the components of the design

Object Oriented Design 1. Understand System and interactions – Use case Diagrams – Activity Diagrams – Use case Scenarios. Object Oriented Design 2. Design System Architecture – Subsystems and communication between the subsystems. Object Oriented Design 4. Develop Design Models – Describing a system at a high level of abstraction – Design Model types , Structural models , Dynamic models – Is it necessary to model software systems? IT1060 - Software Process Modeling SLIIT - Faculty of Computing What Is a Model

Object Oriented Design 4. Develop Design Models – Design Model types • Structural models • Dynamic models – Modeling Languages • UML • SysML • Refer https://modeling-languages.com/# IT1060 - Software Process Modeling SLIIT - Faculty of Computing What Is the UML? • The UML is a language for • Visualizing • Specifying • Constructing

Dynamic models : . Describes the dynamic structure of the system and shows the interactions between the system objects. Interactions that may be documented include the sequence of service requests made by objects and the state changes that are triggered by these object interactions. (You will learn them in SE next semester

Use Case Model; –Graphically represent the proposed functionality of the new system. –Use Case Model captures the functional requirements of a system. –Help to demonstrate the high-level behavior of the proposed system to the clients What is a Use Case Diagram? IT1060 - Software Process Modeling SLIIT - Faculty of Computing

System is something which perform function(s). System Boundary Represents the boundary between the (physical) system and the actors who interact with the (physical) system. A Use Case Diagram shows the interaction between the system and entities external to the system. These external entities are referred to as Actors

Below mentioned are the main types of relationships used in use case diagrams. Association. indicates that an actor participates in (i.e. communicates with) the use case. Generalization. Actor Generalization is drawn from the concept of inheritance in Object Oriented Programming. A child actor Inherits all of the characteristics and behavior of the parent actor

Extend – The base use case implicitly incorporates the behavior of another use case at certain points called extension points. – The base use case may stand alone, but under certain conditions its behavior may be extended by the behavior of another use case. Generalization – The child use case inherits the behavior and meaning of the parent use case. – The child may add to or override the behavior of its parent.