

# System Development Methods

## CT00046-3-2

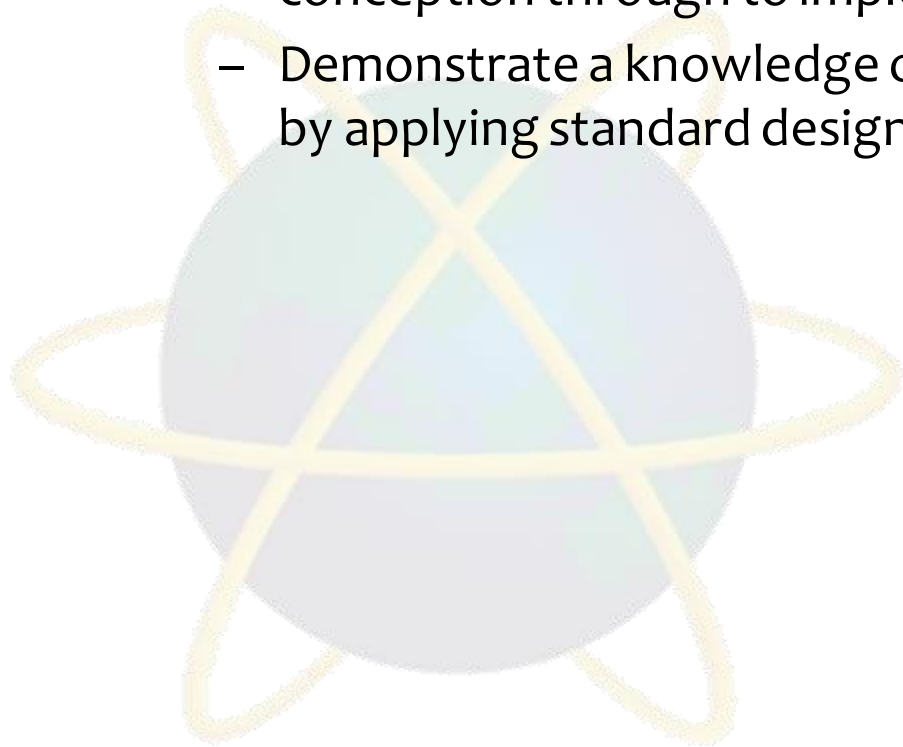


**A · P · U**  
ASIA PACIFIC UNIVERSITY  
OF TECHNOLOGY & INNOVATION

## System Design – Part 2

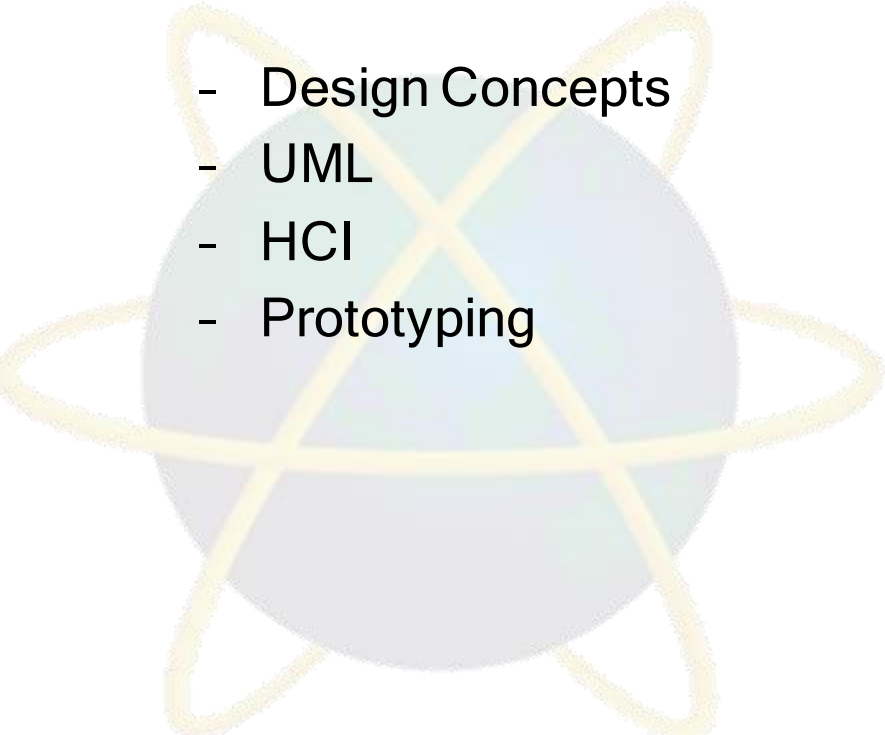
# Learning Outcome

- At the end of the module, you should able to:
  - Apply UML analysis, design and implementation techniques to develop a simple prototype, with a suitable interface, from conception through to implementation.
  - Demonstrate a knowledge of the fundamental issues of prototyping by applying standard design principles.



# Key Terms you must be able to use

- If you have mastered this topic, you should be able to use the following terms correctly in your assignments and exams:

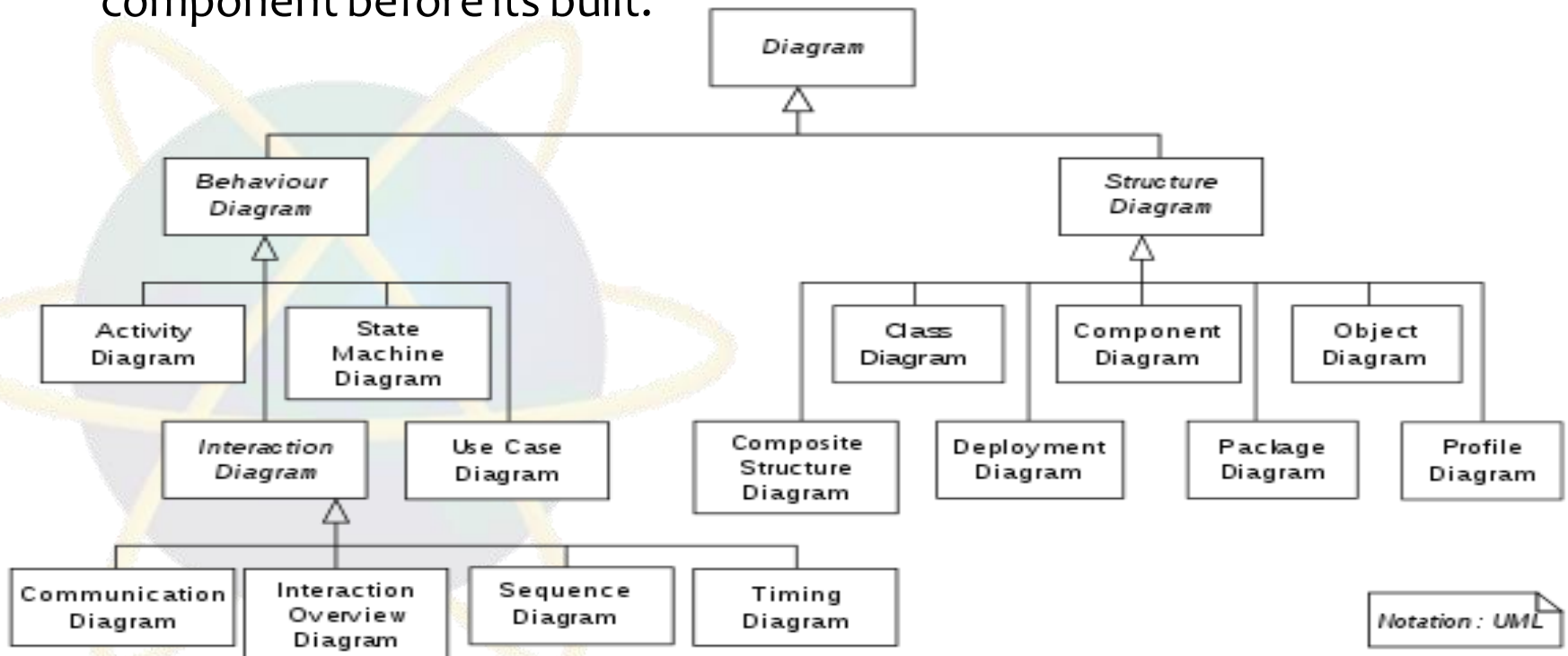
- 
- Design Concepts
  - UML
  - HCI
  - Prototyping

# Influence of Software Design

- Process Centered Design
  - Designing software based on automation of process
  - Ex; Ticketing Machine, Manufacturing process, Hotel booking, etc.
- Data Centered Design
  - Designing software based on processing large data.
  - Ex; Big Data, Shopping, Immigration system, flight management.
- User Centered Design
  - Designing software based on human behavior
  - Ex; AI, games, social sites, navigation applications, etc.

# Unified Modeling Language (UML)

- Modeling techniques to visualize the design of a system.
- Supports Object Oriented development approach.
- UML can be used to design, discuss, present and event test a software component before its built.



# TWO different views of UML modeling

- Static / structural view
  - Emphasizes the static structure of the system using objects, attributes, operations and relationships.
  - Used for Data and Process oriented development approach.
  - Presented through - class and composite structure diagrams.
- Dynamic / behavioral view
  - Emphasizes the dynamic behavior of the system by showing collaborations among objects and changes to the internal states of objects.
  - Used for User and some Process oriented development approach.
  - Presented through sequence diagrams, activity diagrams and state machine diagrams.

# Prototyping

- Process of creating model(prototype) of the system.
- Features
  - Only shows the main interface, functions and features.
- Advantage
  - ‘3D’ view of product, capture more problems / opinions
  - Display for early user feedback / confirmation / satisfaction.
- Disadvantage
  - May cost money and time to create / modify
  - Developer unclear of the scope of the prototype (end-up over doing it)
  - Not all function can be seen in a prototype.

# Dimensions of Prototyping

- Horizontal prototype
  - Prototype provides a broad view of an entire system or subsystem
  - Focusing on user interaction more than low-level system functionality, such as database access.
  - Useful for - demonstration to users and estimating project.
- Vertical prototype
  - Prototype elaborates a single subsystem or function.
  - It is useful for obtaining detailed requirements for a given function, such as refinement database design.
  - Useful for - study and testing.



# Types of Prototype

- Throw-away Prototype
  - Basic form of prototype, created for demonstration purpose.
  - Only shows ‘dummy’ representation of the actual product.
- Evolutionary Prototype
  - Entire product is built in stages based on added / changing requirement.
  - Commonly this prototype is developed to become the final product.
- Incremental Prototype
  - Several prototypes are merged and refined to become final product.
- Extreme Prototype
  - Widely used for Web Development.
  - Product is delivered in fully functional scale, by stages
  - Stage 1 - Usually Static Pages (HTML)
  - Stage 2 – All Functions (Coding)
  - Stage 3- Other services / components such as database.

# Human Computer Interaction (HCI)

- Concepts of creating software which are safe and pleasant to use.
- Concerns mostly on interface design, between a computer system and user.
- Things to consider in HCI;
  - Users abilities/ disabilities
  - User's age group (interest, legal contents, etc)
  - User's computer literacy (Novice, Intermediate, Expert)
  - User's Socio-culture (perception of color, language, religious sensitivity, etc)
  - Purpose of Software (Command line, GUI, Interactive, 3D, etc.)

## Quick Quiz - HCI

- Consider this situation
  - Designing an educational website for children between 3 to 5 years old.
- Why are the design factors that you would consider for this website, in term of:
  - Color?
  - Font size?
  - How any number of words to use? Long sentences?
  - How would a child enter data?
  - How would a child receive data from the site?
  - How would they start the website? Can they remember the web address? Do they need to know any other commands?

# Quick Quiz – HCI

## *Possible answer*

- Why are the design factors that you would consider for this website, in term of:
  - Color?
    - Children at that age can only recognize Primary Colors.
  - Font size?
    - Doesn't need to be large, they can still see clearly.
  - How any number of words to use? Long sentences?
    - Few, they cant read properly. More pictures and metaphors.
  - How would a child enter data?
    - Click on picture, voice reply
  - How would a child receive data from the site?
    - Animation, voice reply

# Tutorial

1. List the UML models that you could find. *How many have you tried creating?* Study the use/function of each model (where and when it should be used).
2. Match UML Models with corresponding STRUCTURED DIAGRAMS such as DFD. *Can you use only UML for an entire project?*
3. What type of prototype would you build for the below systems (and why)?
  - Mobile based Game.
  - Lazada type of Website.
  - A Calculator application for PC.

# Next Session

- System Implementation

