virtusa®
Accelerating Business Outcomes

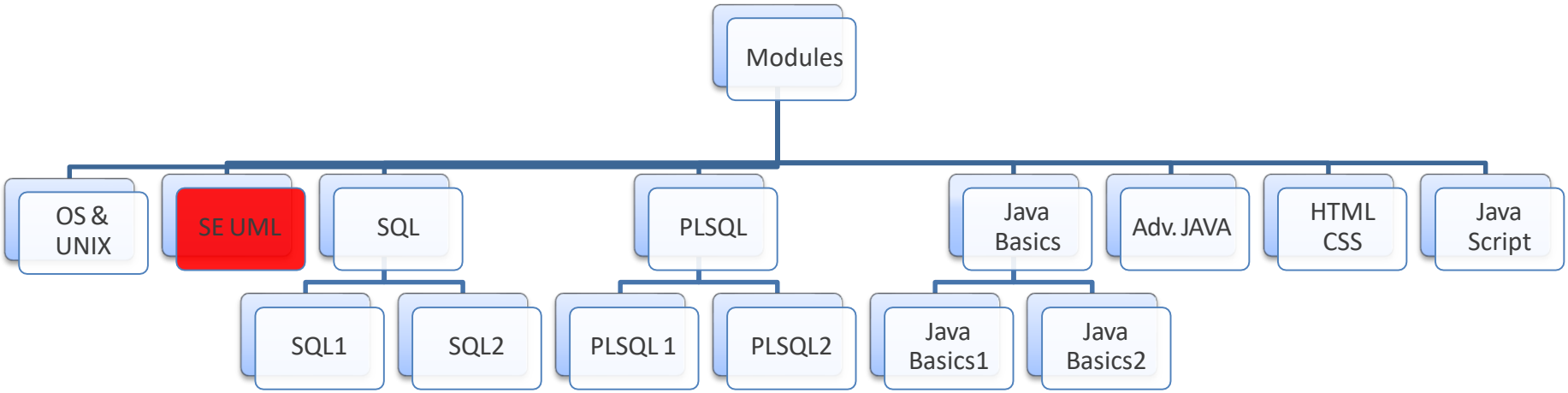
SE and UML

Module Overview

Purpose:

The following module hierarchy presents the technical modules required to build the basic IT skills and acquaints you with relevant technology basics.

The current module (highlight in red) is contributing on Basics of Software Engineering practices during software product development and the basics of Unified Modeling Language.



* **Recommended duration to complete SE & UML module: 8 hours**

Module Objectives

By the end of this module, you will be able to:

- Define software engineering practices during application development
- Define Software Development Life Cycle
- Use Waterfall model during product development
- Develop UML Diagrams using OOPS features

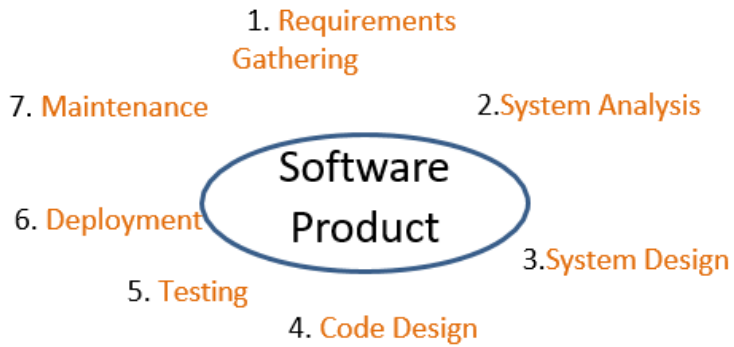
Software Engineering - Define software engineering practices during application development

What is Software Engineering?

- Software engineering is an engineering branch associated with development of software product using well-defined scientific principles, methods and procedures.
- The outcome of software engineering is an efficient and reliable software product.

Usage of Software Engineering in Application Development

- Software is more than just a program code. A program is an executable code, which serves some computational purpose. Software is considered to be collection of executable programming codes, associated libraries and documentations. Software, when made for a specific requirement is called software product.
- Engineering on the other hand, is all about developing products, using well-defined, scientific principles and methods.



Software Engineering

References	http://www.tutorialspoint.com/software_engineering/software_engineering_overview.htm

Software Development Life Cycle - Define Software Development Life Cycle

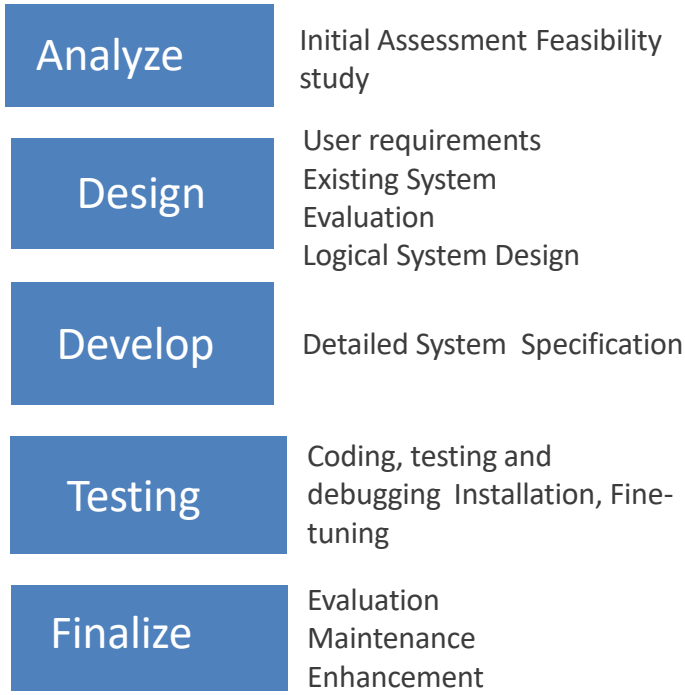
What is Software Development Life Cycle?

- SDLC, Software Development Life Cycle, is a process used by software industry to design, develop and test high quality software.
- The SDLC aims to produce a high quality software that meets or exceeds customer expectations, reaches completion within the time and cost estimates.
- It is also called as Software development process.

Usage of SDLC phases

Analyze->Design->Develop->Testing->Finalize

- The Software Development Life Cycle (SDLC) is a framework defining tasks performed at each step in the software development process.
- ISO/IEC 12207 is an international standard for software lifecycle processes. It aims to be the standard that defines all the tasks required for developing and maintaining software.



Software Development Life Cycle

References

- http://www.tutorialspoint.com/sdlc/sdlc_overview.htm

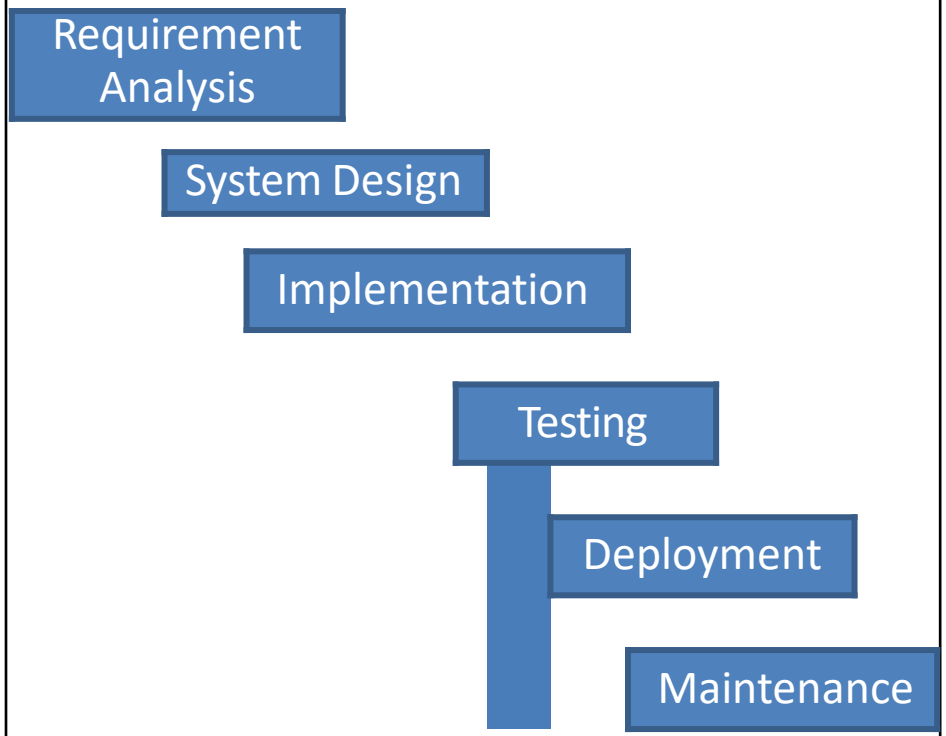
Waterfall Model - Define Waterfall Model

What is Waterfall Model

- Waterfall model is the simplest model of software development paradigm.
- It says that all the phases of SDLC will function one after the other in a linear manner, i.e., when the first phase is finished then only the second phase will start and so on.

Usage of Waterfall Model

- Waterfall model is suggested in product development, if the requirements are known to the user.
- If the base data exists, user can add new data on top of it.



Waterfall Model

References	http://www.tutorialspoint.com/software_engineering/software_development_life_cycle.htm

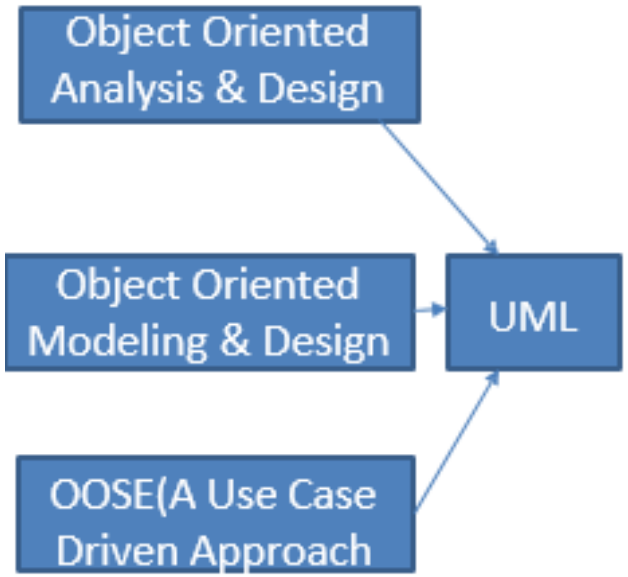
Unified Modeling Language - Define UML

What is Unified Modeling Language?

- Unified Modeling Language is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.
- UML was created by Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.
- UML is different from the other common programming languages like C++, Java, COBOL etc.
- UML is a pictorial language used to make software blue prints.

Usage of UML

- An object contains both data and methods that control the data. The data represents the state of the object. A class describes an object and they also form hierarchy to model real world system. The hierarchy is represented as inheritance and the classes can also be associated in different manners as per the requirement.



UML Structure

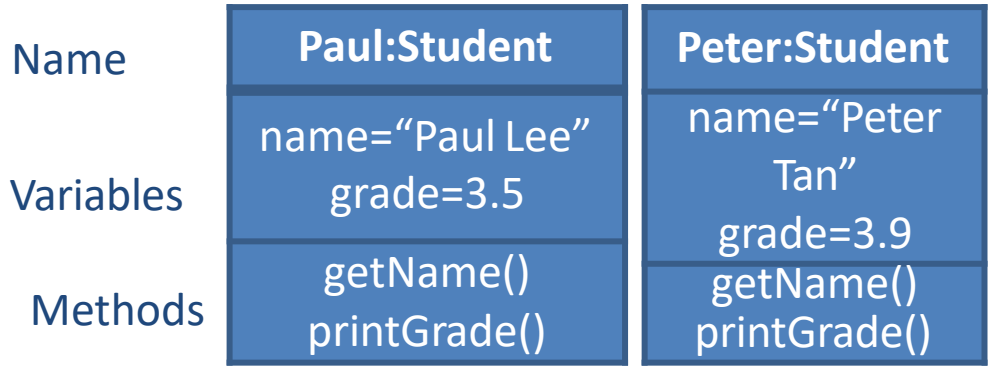
References

- http://www.tutorialspoint.com/uml/uml_standard_diagrams.htm

UML Continued - OOPs Features

Following are some fundamental concepts of object oriented world:

- **Objects:** Objects represent an entity and the basic building block.
- **Class:** Class is the blue print of an object.
- **Abstraction:** Abstraction represents the behavior of a real world entity.
- **Encapsulation:** Encapsulation is the mechanism of binding the data together and hiding them from outside world.
- **Inheritance:** Inheritance is the mechanism of making new classes from existing one.
- **Polymorphism:** It defines the mechanism to exist in different forms.



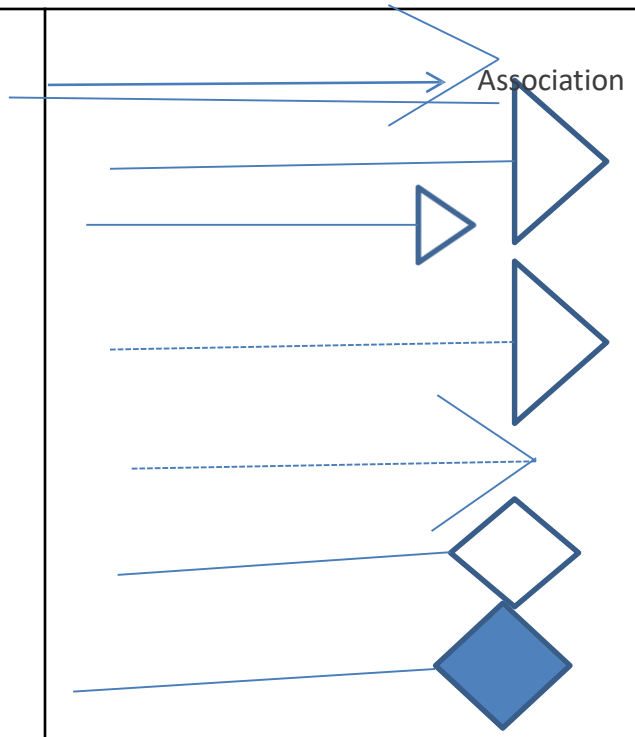
Two instances of the class Student

References

- http://www.tutorialspoint.com/uml/uml_standard_diagrams.htm

UML Continued - UML Diagrams and Notations

- **Use case diagram :** Captures high level artifacts of the system
- **Class diagram :** Captures structural aspects of the system
- **Sequence diagram:** Captures objects communication
- **State chart diagram:** Captures state transitioning of objects
- **Activity diagram:** Captures the system flow from source to sink
- **Component diagram:** Builds the component to use
- **Deployment diagram:** Deploys the built-in component



UML Class Diagram Notations

References	• http://www.tutorialspoint.com/uml/uml_standard_diagrams.htm

Self Check?

Instructions to write Self Evaluation Sheet:

Open the excel sheet, refer SE UML sheet, write down the solutions for all questions, save a local copy in your machine.

Lab Assignment

- Refer ***Assignment Document*** for this module to proceed with **Lab Assignment.**
- Do **submit the Solutions** for the given assignment and refer the ***Participant guide*** for submission procedure.

Module Summary

Now that you have completed this module, you will be able to:

- Use Software engineering practices in application development
- Define Software Development Life Cycle phases
- Use Waterfall and Iterative model
- Use Unified Modeling Language Diagrams

Thank you!