***Software Engineering******Lecture # 4***

***Objectives:***

* To discuss various phases of Software Development Life Cycle.

***Software Design:***

* Next step is to bring down whole knowledge of requirements and analysis on the desk and design the software product.
* Brings together customer requirements, business needs, and technical considerations to form the “blueprint” for a product
* The inputs from users and information gathered in requirement gathering phase are the inputs of this step.
* Creates a model that provides detail about software data structures, software architecture, interfaces, and components that are necessary to implement the system
* The output of this step comes in the form of two designs; logical design, and physical design.
* Engineers produce meta-data and data dictionaries, logical diagrams, data-flow diagrams, and in some cases pseudo codes.

***There are two kinds of design documents developed in this phase:***

1. ***High-Level Design (HLD):***
   * Brief description and name of each module
   * An outline about the functionality of every module
   * Interface relationship and dependencies between modules
   * Database tables identified along with their key elements
   * Complete architecture diagrams along with technology details
2. ***Low-Level Design (LLD):***
   * Functional logic of the modules
   * Database tables, which include type and size
   * Complete detail of the interface
   * Addresses all types of dependency issues
   * Listing of error messages
   * Complete input and outputs for every module

***Coding:***

* This step is also known as programming phase.
* The implementation of software design starts in terms of writing program code in the suitable programming language and developing error-free executable programs efficiently.

***Testing:***

* An estimate says that **50%** of whole software development process should be tested.
* Errors may ruin the software from critical level to its own removal.
* Software testing is done while coding by the developers and thorough testing is conducted by testing experts at various levels of code such as module testing, program testing, product testing, in-house testing, and testing the product at user’s end.
* Early discovery of errors and their remedy is the key to reliable software.

***Integration:***

* Software may need to be integrated with the libraries, databases, and other program(s).
* This stage of **SDLC** is involved in the integration of software with outer world entities.

***Implementation:***

* This means installing the software on user machines.
* At times, software needs post-installation configurations at user end.
* Software is tested for portability and adaptability and integration related issues are solved during implementation.

***Operation and Maintenance:***

* This phase confirms the software operation in terms of more efficiency and less errors.
* If required, the users are trained on, or aided with the documentation on how to operate the software and how to keep the software operational.
* The software is maintained timely by updating the code according to the changes taking place in user end environment or technology.
* This phase may face challenges from hidden bugs and real-world unidentified problems.

***Maintenance:***

* Once the system is deployed, and customers start using the developed system, following **3** activities occur

***Bug fixing***:

Bugs are reported because of some scenarios which are not tested at all

***Upgrade***:

Upgrading the application to the newer versions of the Software

***Enhancement:***

Adding some new features into the existing software