

# Software Engineering

Prepared by: Neha Tripathi

***Software life cycle model***

**or**

***Software development life cycle (SDLC) model***

**or**

***Software development process model***

# Software Life Cycle

- Software life cycle (or software process):
  - **series of identifiable stages that a software product undergoes during its life time:**
    - Feasibility study
    - Requirements analysis and specification,
    - Design,
    - Coding,
    - Testing
    - Installation
    - Maintenance.

# Software Life Cycle Model

- **A software life cycle model (or process model):**
  - a **descriptive and diagrammatic model** of software life cycle:
  - identifies all the **activities** required for product development,
  - establishes a precedence **ordering** among the different activities,
  - Divides life cycle into **phases**.
- Several different activities may be carried out in each life cycle phase.
  - For example, the design stage might consist of:
    - structured analysis activity followed by
    - structured design activity.

# Why Model Life Cycle ?

- A written description:
  - **forms a common understanding of activities among the software developers.**
  - **helps in identifying inconsistencies, redundancies, and omissions in the development process.**
  - **Helps in tailoring a process model for specific projects.**
- The development team must identify a suitable life cycle model:
  - and then adhere to it.
  - Primary advantage of adhering to a life cycle model:
    - **helps development of software in a systematic and disciplined manner.**
- When a program is developed by a single programmer ---
  - he has the freedom to decide his exact steps.

- **When a software product is being developed by a team:**
  - **there must be a precise understanding among team members as to when to do what,**
  - **otherwise it would lead to chaos and project failure.**
- **A software project will never succeed if:**
  - one engineer starts writing code,
  - another concentrates on writing the test document first,
  - yet another engineer first defines the file structure
  - another defines the I/O for his portion first.
- **A life cycle model:**
  - **defines entry and exit criteria for every phase.**
  - **A phase is considered to be complete:**
    - **only when all its exit criteria are satisfied.**

- The phase exit criteria for the software requirements specification phase:
  - Software Requirements Specification (SRS) document is complete, reviewed, and approved by the customer.
- A phase can start:
  - only if its phase-entry criteria have been satisfied.
- **It becomes easier for software project managers:**
  - **to monitor the progress of the project.**
- When a life cycle model is adhered to,
  - **the project manager can at any time fairly accurately tell,**
    - **at which stage (e.g., design, code, test, etc. ) of the project is.**
  - **Otherwise, it becomes very difficult to track the progress of the project**
    - the project manager would have to depend on the guesses of the team members.
- This usually leads to a problem:
  - known as the 99% complete syndrome.

- Many life cycle models have been proposed.
- We will confine our attention to a few important and commonly used models.
  - **classical waterfall model**
  - **iterative waterfall,**
  - **evolutionary,**
  - **prototyping, and**
  - **spiral model**



# Summary

- Software engineering is:
  - **systematic collection of decades of programming experience**
  - **together with the innovations made by researchers.**
- A fundamental necessity while developing any large software product:
  - **adoption of a life cycle model.**
- Adherence to a software life cycle model:
  - **helps to do various development activities in a systematic and disciplined manner.**
  - **also makes it easier to manage a software development effort.**

Thank you!