# SOFTWARE ENGENEERING PART 2

**1) What is SDLC?**

- **Ans) Software Development Life Cycle (SDLC) is the application of standard business practices to building software applications.**
- **The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates. It's typically divided into six to eight steps.**
- **It is also called as Software Development Process.**
- **SDLC is a framework defining tasks performed at each step in the software development process.**
- **ISO/IEC 12207 is an international standard for software life-cycle processes.**

**2) What are steps involved in SDLC?**

**Ans i) Planning: Project leaders evaluate the terms of the project.**

- **Planning can also include feedback from stakeholders.**
- **Planning should clearly define the scope and purpose of the application.**
- **It plots the course and provisions the team to effectively create the software.**
- **It also sets boundaries to help keep the project from expanding or shifting from its original purpose.**

**(ii) Define Requirements: Defining requirements is considered part of planning to determine what the application is supposed to do and its requirements. Requirements also include defining the resources needed to build the project.**

**(iii) Design and Prototyping: The Design phase models the way a software application will work. Some aspects of the design include: Architecture, User Interface, Platforms, Programming, Communications, Security.**

**(iv) Software Development: This is the actual writing of the program.**

- **A small project might be written by a single developer, while a large project might be broken up and worked by several teams.**
- **An Access Control or Source Code Management application is used in this phase. These systems help developers track changes to the code. They also help ensure compatibility between different team projects and to make sure target goals are being met.**

**(v) Testing: It's critical to test an application before making it available to users.**

- **Testing should ensure that each function works correctly.**
- **Different parts of the application should also be tested to work seamlessly together—performance test, to reduce any hangs or lags in processing.**
- **The testing phase helps reduce the number of bugs and glitches that users encounter. This leads to a higher user satisfaction and a better usage rate.**

**(vi) Deployment: In the deployment phase, the application is made available to users. Deployment can be complex and may take time and effort.**

**(vii) Operations and Maintenance: In this phase, users discover bugs that weren't found during testing.**

**These errors need to be resolved, which can spawn new development cycles.**

**3) Explain the phases of waterfall model with advantages and disadvantages**

**Ans) Phases of Waterfall Model:**

**(i) Feasibility Study**

**(ii) Requirements analysis and specification:**

**a) Requirement gathering and analysis:**

**b) Requirement specification:**

**(iii) Design**

**(iv) Coding and Unit testing**

**(v) Integration and System testing**

**(vi) Maintenance**

**Advantages:**

**Simplicity: The linear and sequential nature of the Waterfall model makes it easy to understand and use. Each phase has specific goals and deliverables.**

**Drawbacks:**

**Difficult to incorporate change requests.**

**Incremental delivery not supported.**

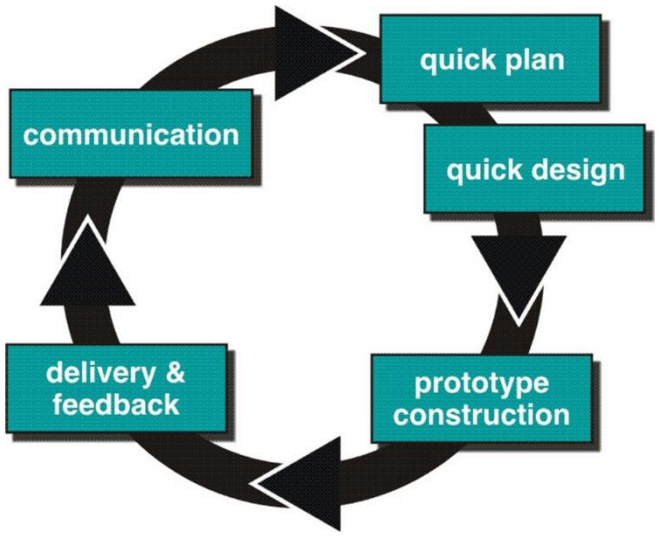
**Overlapping of phases not supported.**

**Risk handling not supported.**

**Limited customer interactions.**

**4) Write a short note on prototyping**

**Ans**




**Best approach when:**

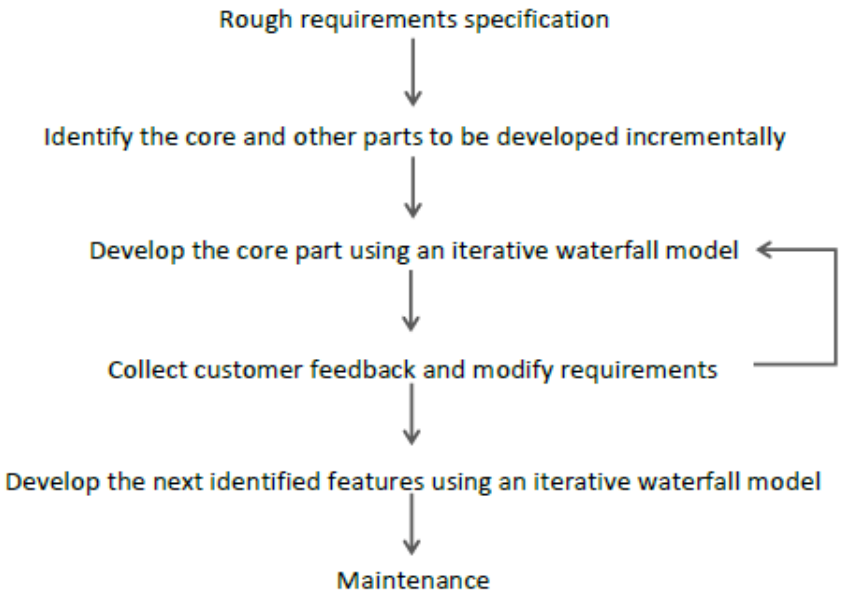
- **Objectives defined by customer are general but does not have details like input, processing, or output requirement.**
- **Developer may be unsure of the efficiency of an algorithm, O.S., or the form that human machine interaction should take.**
- **It can be used as standalone process model.**
- **Model assist software engineer and customer to better understand what is to be built when requirement are fuzzy.**
- **Prototyping start with communication, between a customer and software engineer to define overall objective, identify requirements and make a boundary.**
- **Going ahead, quick planning and modeling (software layout visible to the customers/end-user) occurs.**
- **Quick design leads to prototype construction.**
- **Prototype is deployed and evaluated by the customer/user.**
- **Feedback from customer/end user will refine requirement and that is how iteration occurs during prototype to satisfy the needs of the customer.**
- **Prototype can be serve as “the first system”.**

- **Both customers and developers like the prototyping paradigm. Both must be agree that the prototype is built to serve as a mechanism for defining requirement.**
- **Customer/End user gets a feel for the actual system**
- **Developer get to build something immediately.**

**5) Explain evolutionary model with its advantages and disadvantages**

**Ans) Evolutionary model is also referred to as the successive versions model and sometimes as the incremental model.**

- **In Evolutionary model, the software requirement is first broken down into several modules that can be incrementally constructed and delivered**
- **The development first develops the core modules of the system. The core modules are those that do not need services from the other modules. The initial product skeleton is refined into increasing levels of capability by adding new functionalities in successive versions.**
- **Each evolutionary model may be developed using an iterative waterfall model of development.**
- **Is normally useful for very large products, where it is easier to find modules for incremental implementation.**
- **Is used when the customer prefers to receive the product in increments so that he can start using the different features as and when they are developed rather than waiting all the time for the full product to be developed and delivered.**



➢ **Advantages of Evolutionary Models:**

- **Normally useful for very large products.**
- **User gets a chance to experiment with a partially developed software much before the complete version of the system is released.**
- **Helps to accurately elicit user requirements during the delivery of different versions of the software.**
- **The core modules get tested thoroughly, thereby reducing the chances of errors in the core modules of the final products.**
- **Avoids the need to commit large resources in one go for development of the system.**

➢ **Disadvantage of Evolutionary Model:**

- **Difficult to divide the problem into several versions that would be acceptable to the customer and which can be incrementally implemented and delivered.**

**6) Explain phases of agile model.**

**Ans)**

**(i) Requirements gathering.**

**(ii) Design the requirements.**

**(iii) Construction/ iteration.**

**(iv) Testing/ Quality assurance.**

**(v) Deployment.**

**(vi) Feedback.**

**7) Write shot note on agile model**

**Ans) Based on iterative development.**

- **Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning.**
- **The project scope and requirements are laid down at the beginning of the development process.**
- **Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.**

- **Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks.**
- **The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements.**
- **Each iteration involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.**

**8) Explain the steps and goals for reverse engineering**

**Ans) Goals of reverse engineering**

**(i) Cope with Complexity.**

**(ii) Recover lost information.**

**(iii) Detect side effects.**

**(iv) Synthesise higher abstraction.**

**(v) Facilitate Reuse.**

**9) Describe objectives of re-engineering.**

**Ans) Objectives of Re-Engineering:**

- **To describe a cost-effective option for system evolution.**
- **To describe the activities involved in the software maintenance process.**
- **To distinguish between software and data re-engineering and to explain the problems of data re-engineering.**

**10) Explain the steps involved in re-engineering**

**Ans) Steps in Re-Engineering:**

**(i) Recording data flow.**

**(ii) Recording control flow.**

**(iii) Review extracted design.**

**(iv) Generate documentation.**