### **SDLC (Software Development Life Cycle) – Summary**

### SDLC is a systematic process used to develop high-quality software in a structured and efficient manner. It defines a series of steps that software developers follow to design, develop, test, and maintain software applications. The main goal of SDLC is to produce software that meets customer requirements, stays within budget, and is completed on time.

### **Phases of SDLC:**

### **1. Requirement Gathering and Analysis**

* Understand what the client or user wants from the software.
* Conduct meetings or interviews with stakeholders for detailed input.
* Analyze feasibility in terms of technical, operational, and financial aspects.
* Create the Software Requirement Specification (SRS) document.
* Get approval from stakeholders before moving to the next phase.

### **2. Design**

* Convert requirements into a blueprint or architecture.
* Define software modules, data flow, and interface designs.
* Prepare High-Level Design (HLD) and Low-Level Design (LLD).
* Choose suitable technologies, tools, and platforms.
* Validate the design before starting development.

### **3. Implementation or Coding**

* Developers write the actual code based on the design.
* Follow coding standards and use version control tools.
* Develop the application module-wise or feature-wise.
* Perform unit testing during development.
* Result is a functional and testable software product.

### **4. Testing**

* Conduct various types of testing to find bugs and errors.
* Types include Unit Testing, Integration Testing, System Testing, and User Acceptance Testing.
* Ensure that the software meets all specified requirements.
* Check performance, security, and user experience.
* Create test reports and fix issues if found.

### **5. Deployment**

* Deploy the tested software to the production environment.
* Can be released in phases such as beta and full version.
* Monitor for any real-time issues after deployment.
* Provide support and training to end users if required.
* Document the deployment process for future use.

### 

### **6. Maintenance**

* Resolve bugs and errors that occur after deployment.
* Add improvements or new features as per user needs.
* Adapt the software to updated environments or technologies.
* Perform routine updates and system optimization.
* Maintain updated documentation for all changes.