SDLC(Software Development Life Cycle) Process

SDLC Phases:

1. Requirements Gathering
2. System Analysis
3. System Design
4. Coding/Implementation
5. Testing
6. Release/Deployment
7. Maintenance

    a. Corrective Maintenance
    b. Enhasive Maitenance

People Invovled

=> Client
=> CEO
=> Project Manager(PM)
=> Business Analyst(BA)
=> System Analyst(SA)
=> Technical Architect
=> Developer
=> Tester
=> CCB Team(Change Contol Board) / Production Support Team/ Maintenance Team

Software Bidding = a proposal given by the client about a new s/w to be implemented

Kick Of Meeting = to select a person as project Manager

1. Requirements Gathering Phase :

Responsible Person : Business Analyst
Input : User Requirements
Output : BRS(Business Requirements Specification) Document
URS(User Req. spec.) doc
CRS(Customer Req Spec) doc

2. System Analysis :
==========

Responsible Person : System Analyst
Input : BRS Doc
Output : FRS(Functional Requirements Spec. Doc)
SRS(Software Requirements spec. doc)

Feasibility Study

i. Economical Feasiblity Study
ii. Resource Feasibility Study

iii. Operational Feasiblity Study

Resource Manager

3. System Design:

Responsible Person : Technical Architect
Input : FRS/SRS Doc
Output : HLD(High Level Design) Doc [main modules information]
LLD(Low Level Design) Doc [internal details of each module]

- ER Diagrams(Entity Relationshiop diagrams)
- DFD's (Data Flow Diagrams)
- FlowCharts
- Use Case Diagrams

4. Coding:
==========

Responsible Person : Developer
Input : HLD, LLD Doc
Output : Software Product

5. Testing:
==========
Responsible Person : Tester
Input : software product
Output : Quality software

SDLC Models:
---

* Waterfall Model
=> When all the client requirements are clearly specified and they are simple in size
=> Phase by Phase

Advantages:
1. is is simple and all the requirements are clearly mentioned

Disadvantages:
1. REsource Wastage will happen
2. TEsting is happening in the later stages, so cost of fixing the defect will be higher
3. changes to the existing system is difficult to perform

* Incremental Model
=> whan all the recquiedment are clearly mensioned but they are in huge size
=> build by build s/w will be implements

Advantages:
1. Resource wastage will be avoided

Disadantages:
1. longer period of time to get final version of the s/w
2. TEsting is happening in the later stages, so cost of fixing the defect will be higher
3. changes to the existing system is difficult to perform

* Prototype Model
=> provide a sample s/w called prototype for client avaluation

Advantages:
1. no need to perform any changes to the existing s/w

Disadvantages:
1. It is costly when compared to other models
2. TEsting is happening in the later stages, so cost of fixing the defect will be higher

* Spiral Model
=> used for product based s/w
=> version by version s/w will be implemented

Advantages:
1. no dead-line pressure
2. Fastly we can provide a flavor of the s/w product to the customers

Disadvantages:
1. we don't know the end date
2. maintaining the resources for a longer period of time is difficult
3. TEsting is happening in the later stages, so cost of fixing the defect will be higher

* V-Model
verification and validation model

* Agile Model