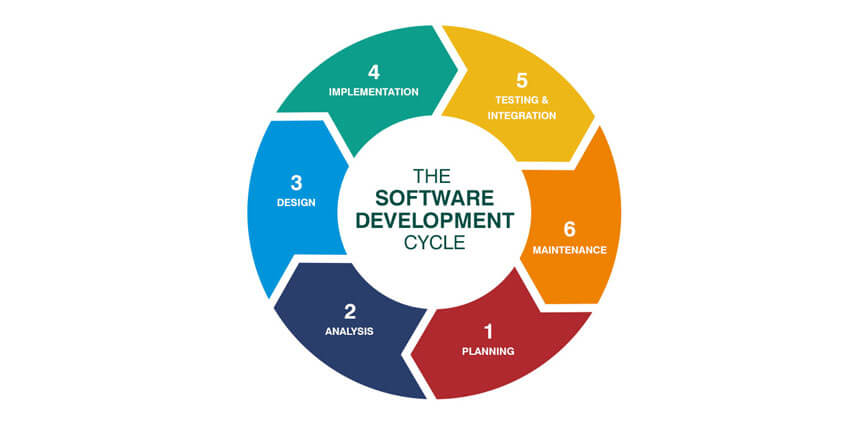
# **1.7 SDLC methods**

The Full form of SDLC is Software Development Life Cycle. SDLC is a process to develop high qualified software. SDLC have six stages. Software life cycle models describe phases of the software cycle and the order in which those phases are executed. Each phase produces deliverables required by the next phase in the life cycle.

## **1.7.1 Phase of SDLC**



**Fig 1.7-1: Phase of SDLC**

1. **Planning**: requirements are gathered in this phase. This phase is the main focus of the project managers and stake holders. Meetings with managers, stake holders and users are held in order to determine the requirements like, who is going to use the system? How will they use the system?  What data should be input into the system?  What data should be output by the system? All the answers are in this planning step.

2. **Analysis**: This is the documentation stage. After requirement analysis, requirements are clearly documented and approved from the customer.

3. **Design**: Based on the requirements, more than one design is proposed. Designed are also documented. It’s called DDS (Design Document Specification). After analysis the DDS, the best design approach is selected for the product.

4. **Implementation**: The work is divided in modules/units and actual coding is started. Since, in this phase the code is produced so it is the main focus for the developer. This is the longest phase of the software development life cycle.

5. **Testing**: In this stage tester try to find bugs and check if the software meets the requirements or not. During this phase all types of functional testing like unit testing, integration testing, system testing, are done as well as non-functional testing are also done.

6. **Maintenance**: After the product is tested then it’s ready to release. Release date depends on organization and market conditions. This stage is also contain maintenance. Based on feedback and customer demand, software maintenance happen.

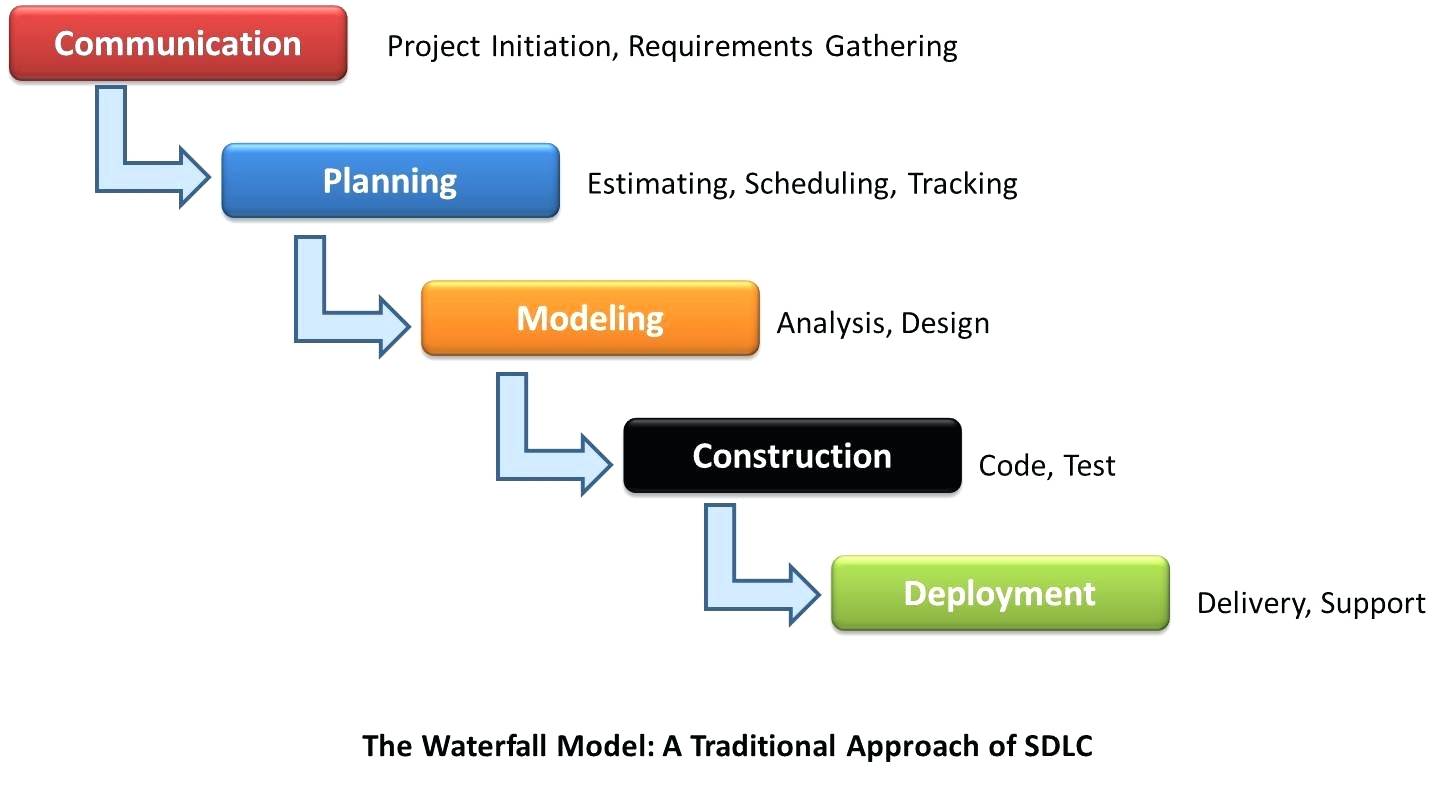
## **1.7.2 SDLC Models**

There are different kinds of SDLC process model. There are two kinds of models -

1. Traditional models
2. Agile models

Traditional Models:

1. **Waterfall model**: The Waterfall Model is a linear sequential flow. In which progress is seen as flowing steadily downwards (like a waterfall) through the phases of software implementation. This means that any phase in the development process begins only if the previous phase is complete.

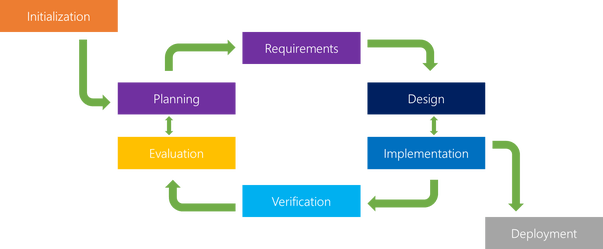


**Fig 1.7-2: Waterfall Model**

1. **Iterative Model**:

It is developed to overcome the weaknesses of the waterfall modelIt starts with an initial planning and ends with deployment with the cyclic interactions in between.This method is based on repeated cycle (iterative). The main plot of this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).

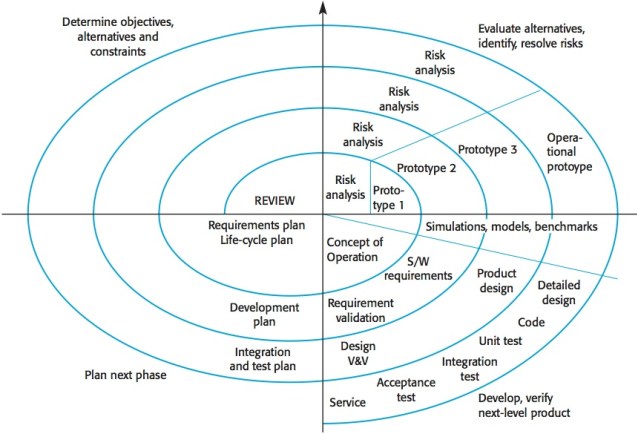
Phases of iterative model are:



**Fig 1.7-3: Iterative Model**

Iterative process is an incremental process. More than one iteration may process at a time. In this model risk can easily analyze. But this model is so complex. Skilled people need for management.

**3.** **Spiral Model**: This Spiral model is a combination of iterative development process model and sequential linear development model i.e. the waterfall model with a very high emphasis on risk analysis. This model of development combines the features of the prototyping model and the waterfall model.

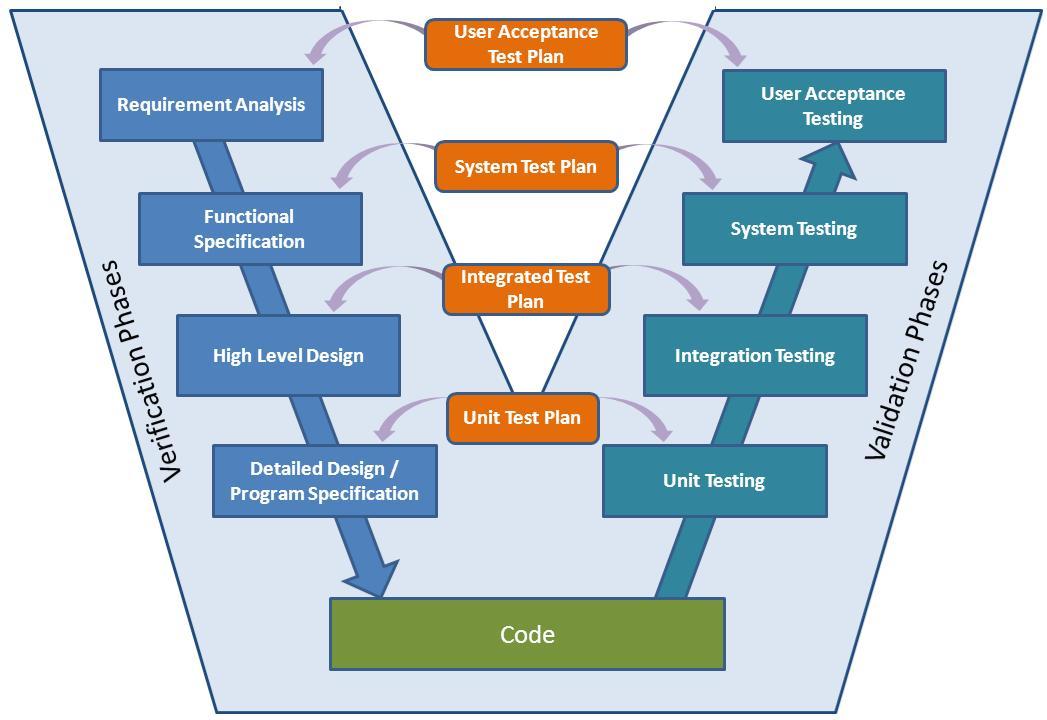


**Fig 1.7-4: Spiral Model**

The spiral model is favored for large, expensive, and complicated projects. This model has four phases. A software project tasks repeatedly passes through these phases in iterations called Spirals. This one is mainly used for medium to high-risk projects, requirements are complex, significant changes are expected in the product during the development cycle. Spiral model is used for big projects and when requirements are not stable.

1. **V – Model**:

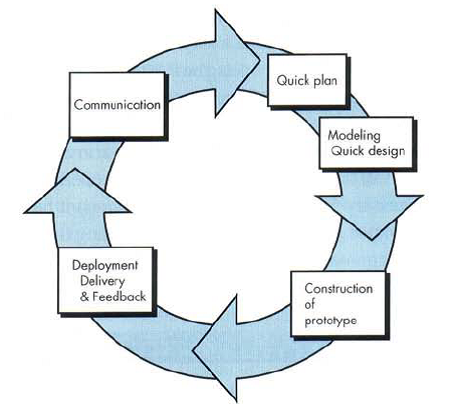
This is also known as Verification and Validation Model.  Instead of moving down in a linear way, the process steps are bent upwards after the implementation and coding phase, to form the typical V shape. The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage. The major difference between V-shaped model and waterfall model is the early test planning in the V-shaped model. Like waterfall model here requirements are well defined and fixed and there will be no undefined or ambiguous requirements. This is mainly used for short projects.



**Fig 1.7-5: V- Model**

1. **Prototyping Model**:

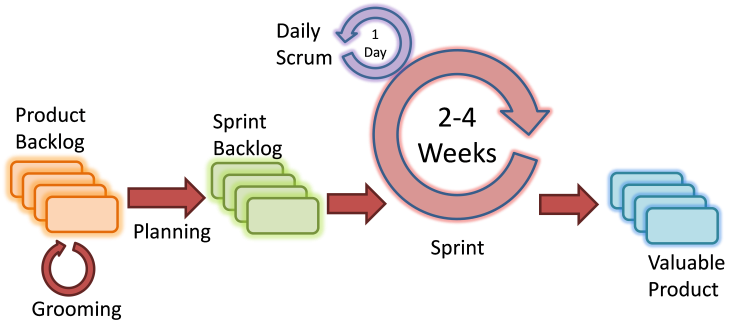
This model is mainly used for understanding the user requirements clearly. This one helps developer to understand what functionality and system look customer is expecting to build. It is an activity that can occur in software development. It used to visualize some component of the software to limit the gap of misunderstanding the customer requirements by the development team. This also will reduce the iterations may occur in waterfall approach and hard to be implemented due to the inflexibility of the waterfall approach. Iteration occurs as the prototype is tuned to satisfy the needs of the customer.



**Fig 1.7-6: Prototyping Model**

1. **Agile Methods**:

It is based on iterative and incremental development, where requirements and solutions evolve through collaboration between cross-functional teams. In Agile model, the tasks are divided to time boxes (small time frames) to deliver specific features for a release. These are done in iteration process It can be used with any type of the project, but it needs more engagement from the customer and to be interactive. Also, it can be used when the customer needs to have some functional requirement ready in less than three weeks and the requirements are not clear enough. This model gives flexibility to developers and here the resource requirements are minimum. But it will be hard to manage or processing the tasks if the project has complex dependencies.



**Fig 1.7-7: Agile Method**

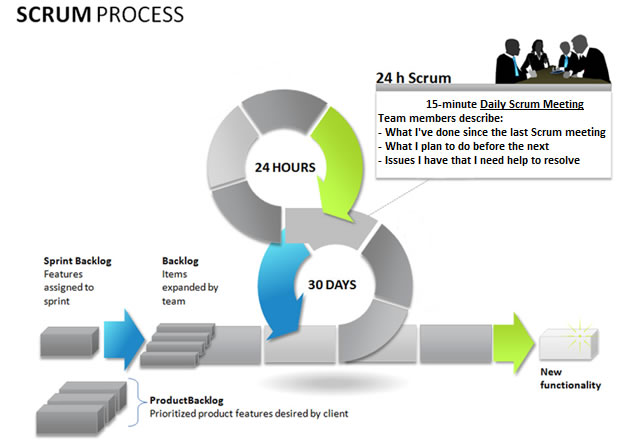
**1. Scrum**:

Scrum have three phase. These are:

1.1. Pre-game

1.2. Development

1.3. Post-game

**Fig 1.7-8: Scrum Process**

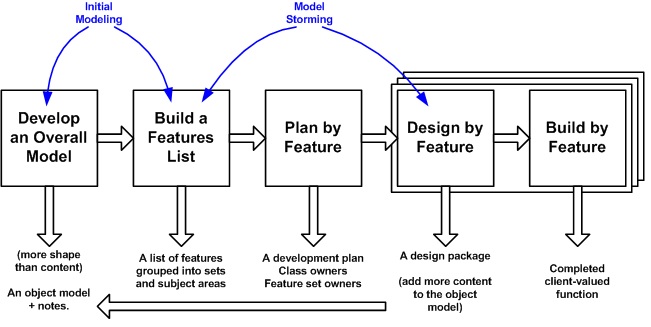
In pre-game phase planning and design are happen. It includes requirements analysis, risk calculation and other documentation part. And in development phase, project is built. Postgame phase come when there is no more requirement. And then the project is being released.

**2. Extreme Programming (XP):** Agile Modeling (AM) is a practices-based software process. In XP method, software may release after every iteration.XP is the most specific of the agile frameworks regarding appropriate engineering practices for software development.



**Fig 1.7-9: Extreme Programming (XP)**

**3. Feature Driven Development (FDD):** FDD is iterative and incremental software development process. FDD have short iteration process. FDD consist five basic activities.



**Fig 1.7-10: FDD**

First two activities are the overall model of the project. The final three activities are iterated for each feature.