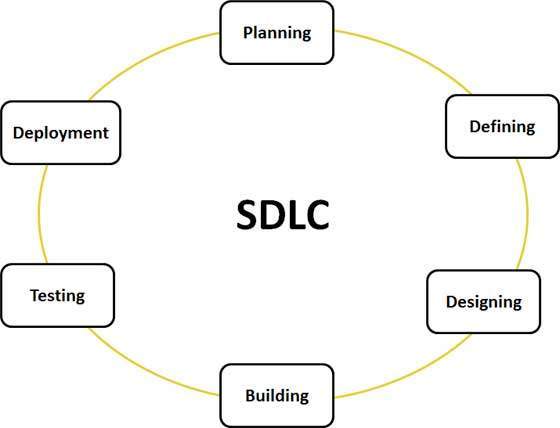
Software Development Life Cycle (Kimberly Tan Jia Ying)

SDLC is the acronym of Software Development Life Cycle. It is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

Here are the different stages of SDLC:

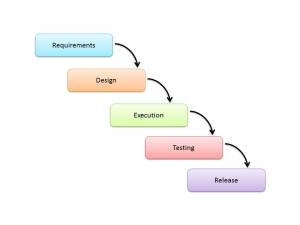


There are various software development life cycle models defined and designed which are followed during software development process. These models are also referred as "Software Development Process Models". Each process model follows a Series of steps unique to its type, in order to ensure success in process of software development.

So here are some of the SDLC models:

1. **Waterfall Model**

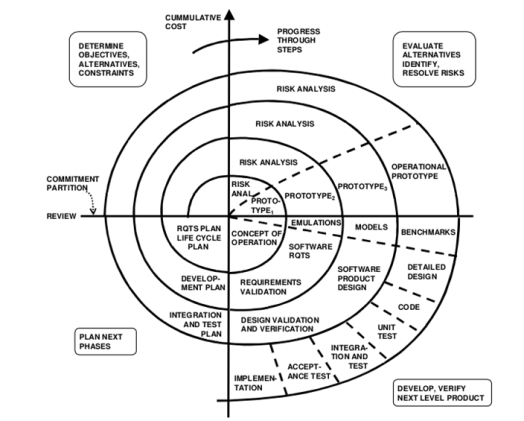
The waterfall Model is a linear sequential flow. In which progress is seen as flowing steadily downwards (like a waterfall) through the phases of software implementation. This means that any phase in the development process begins only if the previous phase is complete. The waterfall approach does not define the process to go back to the previous phase to handle changes in requirement. The waterfall approach is the earliest approach that was used for software development.



The Waterfall Model is used in projects which not focus on changing the requirements, for example, projects initiated from request for proposals.

1. **Spiral Method (SDM)**

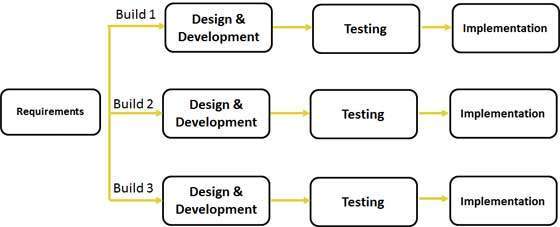
It is combining elements of both design and prototyping-in-stages, in an effort to combine advantages of top-down and bottom-up concepts. This model of development combines the features of the prototyping model and the waterfall model. The spiral model is favored for large, expensive, and complicated projects. This model uses many of the same phases as the waterfall model, in essentially the same order, separated by planning, risk assessment, and the building of prototypes and simulations.



It is used in shrink-wrap large applications and systems which built-in small phases or segments.

1. **Iterative Model design**

Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. At each iteration, design modifications are made and new functional capabilities are added. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).



Iterative and Incremental development is a combination of both iterative design or iterative method and incremental build model for development. "During software development, more than one iteration of the software development cycle may be in progress at the same time." and "This process may be described as an "evolutionary acquisition" or "incremental build" approach."

In incremental model the whole requirement is divided into various builds. During each iteration, the development module goes through the requirements, design, implementation and testing phases. Each subsequent release of the module adds function to the previous release. The process continues till the complete system is ready as per the requirement.

The key to successful use of an iterative software development lifecycle is rigorous validation of requirements, and verification & testing of each version of the software against those requirements within each cycle of the model. As the software evolves through successive cycles, tests have to be repeated and extended to verify each version of the software.

Software Development Life Cycle (Li Jin)

Software Development Life Cycle is a process used by software industry to design, develop and test high quality software, and also followed for a software project, within a software organization. The purpose of SDLC is to provide the high quality software that meets or exceeds customer expectations, reaches completion within time and cost estimates. SDLC consists of a detailed plan which is describing how to develop, maintain, replace and alter or enhance specific software. The SDLC is defined as a methodology to improve the quality of software and the overall development process.

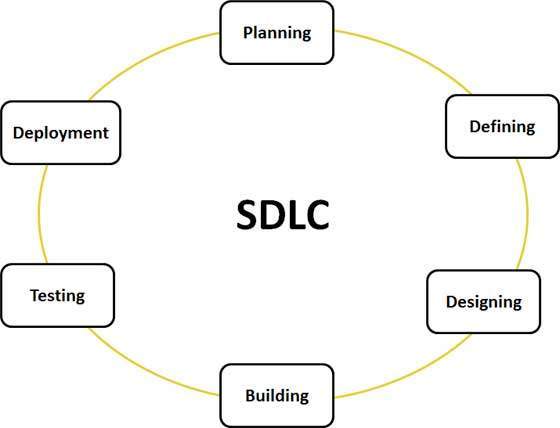


Figure1: Various Stages of a Typical SDLC

The SDLC include the following phases:

* **Planning**

The most important part in this SDLC is the requirement analysis. In this project, there are 3 things need to be thought:

* Identifying the problem, the objectives and the resources such as personnel and costs.
* Studying the ability of proposing alternative solutions after meeting with customers, Mr. and Mrs. Wang.
* Thinking about how to make the product better than other competitors.
* **Defining**
* Determining and documenting the end-user’s requirements, should know that what their expectations are for the system, and how it will perform.
* A feasibility study will be made for the project as well, such as determining whether it is organizationally, economically, socially, technologically feasible.
* Maintaining strong communication with clients or Mr. and Mrs. Wang to ensure that the project is done by the right way and the function as well.
* **Designing**
* Defining the system, such as the elements of the system, the components, the security level, modules, architecture and the different interface.
* Designing the user interface and what kind of the function will be contained.
* **Building**
* Developing the system with the programming languages such as C#, C++, Java and PHP.
* **Testing**
* Ensuring that the system us not easily exploitable to hackers.
* Making sure the actual outcomes are compared and equal to the predicted and desired outcomes.
* **Deployment and Maintenance**
* Deploying the tested system to the real market.
* Maintenance the system which is based on the customer feedback.

**SDLC Models**

The development models are the various processes or methodologies that are being selected for the development of the project depending on the project’s aims and goals. There are many development life cycle models that have been developed in order to achieve different required objectives. The models specify the various stages of the process and the order in which they are carried out.

Following are the most important and popular SDLC models followed in the industry:

1. Waterfall model
2. Agile model
3. Big Bang model

**Waterfall Model**

Waterfall approach was the first SDLC Model and it is used widely in Software Engineering to make sure the success of the project. In the waterfall approach, the whole process will be separated in some different phases. In this model, normally, the outcome of one phase acts as the input for the next phase sequentially.

 Figure 2: Different Phases of the Waterfall Model

**Agile Model**

Agile development model is also a type of Incremental model. Software is developed in incremental, rapid cycles. This results in small incremental releases with each release building on previous functionality. Each release is thoroughly tested to ensure software quality is maintained. It is used for time critical applications



Figure 3: Agile Model

**Big Bang Model**

Big Bang Model is without any specific process. The development just starts with the required money and efforts as the input, and the output is the software developed which may or may not be as per customer requirement.

This is no formal development followed and only little planning is required, and usually, this model is followed for small projects where the development teams are very small.

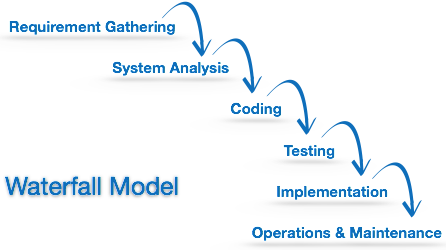
Software Development Life Cycle (Sim Zhi Yu)

SDLC, Software Development Life Cycle is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

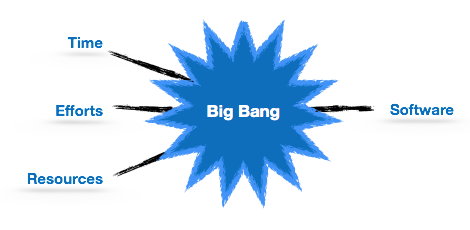
There are six phases in every Software development life cycle model:

1. Requirement gathering and analysis
2. Design
3. Implementation or coding
4. Testing
5. Deployment
6. Maintenance

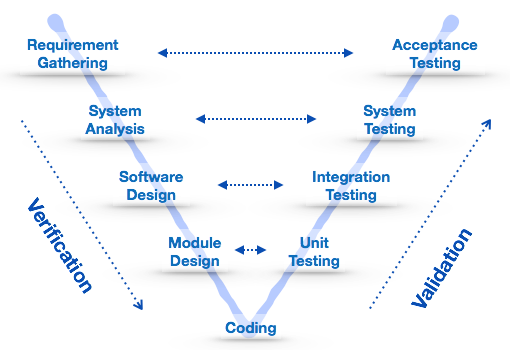
There are various software development life cycle models defined and designed which are followed during software development process. Each process model follows a series of steps unique to its type, in order to ensure success in process of software development. Following are the SDLC models followed in the industry:

1. **Waterfall Model**

Waterfall model is the earliest SDLC approach that was used for software development. The waterfall model illustrates the software development process in a linear sequential flow; hence it is also referred to as a linear-sequential life cycle model. This means that any phase in the development process begins only if the previous phase is complete. In waterfall model phases do not overlap.

1. **Big Bang Model**

Big bang model comprises of focusing all the possible resources in software development and coding, with very little or no planning. The requirements are understood and implemented as they come. Any changes required may or may not need to revamp the complete software.

1. **V-Model**

V-Model is an extension of the waterfall model and is based on association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle there is a directly associated testing phase. This is a highly disciplined model and next phase starts only after completion of the previous phase.

**References (Kimberly Tan Jia Ying)**

<http://www.tutorialspoint.com/sdlc/sdlc_iterative_model.htm>

<https://melsatar.wordpress.com/2012/03/15/software-development-life-cycle-models-and-methodologies/>

**References (Li Jin)**

<http://www.tutorialspoint.com/sdlc/sdlc_overview.htm>

<http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/>

**References (Sim Zhi Yu)**

<http://www.tutorialspoint.com/sdlc/sdlc_tutorial.pdf>

<http://www.tutorialspoint.com/software_engineering/software_development_life_cycle.htm>