**Software Development Life Cycle Models and Methodologies**

Introduction

[Software development life cycle](http://en.wikipedia.org/wiki/Software_development_process) ([**SDLC**](http://en.wikipedia.org/wiki/Systems_Development_Life_Cycle)) is important for the software project success, the good software engineer should have enough experience and knowledge to prefer an choose one model than another based on the project context.

Therefore, it may be required to choose the right SDLC model according to the specific concerns and requirements of the project.

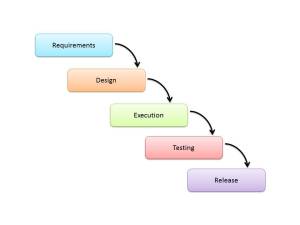
# Types of Software developing life cycles (SDLC)

* [**Waterfall Model**](https://melsatar.blog/2018/02/16/the-waterfall-model-a-different-perspective/)
* **V-Shaped Model**
* [**Spiral**](http://en.wikipedia.org/wiki/Spiral_model)**Method (**[**SDM**](http://en.wikipedia.org/wiki/Software_development_methodology)**)**

## Waterfall Model

### Description

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 and most widely known that was used for software development.



**The usage**

Projects which not focus on changing the requirements, for example, projects initiated from request for proposals ([RFPs](http://en.wikipedia.org/wiki/Request_for_proposal" \o "Request for proposal)), the customer has a very clear documented requirements

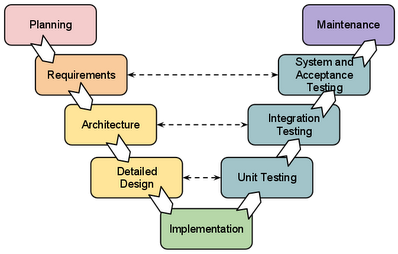
### Advantages and Disadvantages

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| * Easy to explain to the users. * Structures approach. * Stages and activities are well defined. * Helps to plan and schedule the project. * Verification at each stage ensures early detection of errors/misunderstanding. * Each phase has specific deliverables. | * Assumes that the requirements of a system can be frozen. * Very difficult to go back to any stage after it finished. * A little flexibility and adjusting scope is difficult and expensive. * Costly and required more time, in addition to the detailed plan. |

V-Shaped Model

Description

It is an extension of the waterfall 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 major difference between V-shaped model and waterfall model is the early test planning in the V-shaped model.



The usage

* Software requirements clearly defined and known
* Software development technologies and tools are well-known

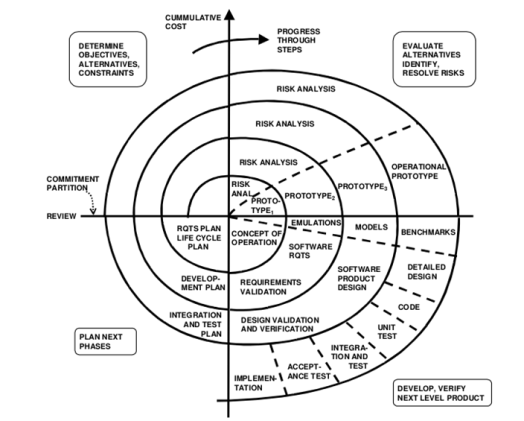
Advantages and Disadvantages

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| * Simple and easy to use * Each phase has specific deliverables. * Higher chance of success over the waterfall model due to the development of test plans early on during the life cycle. * Works well for where requirements are easily understood. * Verification and validation of the product in early stages of product development. | * Very inflexible, like the waterfall model. * Adjusting scope is difficult and expensive. * The software is developed during the implementation phase, so no early prototypes of the software are produced. * The model doesn’t provide a clear path for problems found during testing phases. * Costly and required more time, in addition to detailed plan |

Spiral Model (SDM)

Description

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.



The usage

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

Advantages and Disadvantages

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| * Estimates (i.e. budget, schedule, etc.) become more realistic as work progressed because important issues are discovered earlier. * Early involvement of developers. * Manages risks and develops the system into phases. | * High cost and time to reach the final product. * Needs special skills to evaluate the risks and assumptions. * Highly customized limiting re-usability |