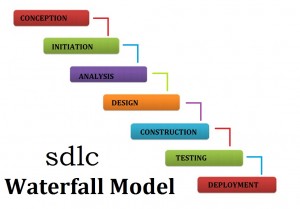
Software Development Life Cycle (SDLC) is a well-defined, structured sequence of stages in software engineering to develop the intended software product. Each phase in the life cycle has its own process and deliverables that feed into the next phase. The life cycle consists of five stages and they are, the analysis of user requirements gathering, designing of the program, coding of program, documenting and testing of system and the operating and maintaining of system.

**Waterfall model**

There are various software development models, and some of it are the Waterfall model, Iterative model and V-model. The Waterfall model is also referred to as a linear-sequential life cycle model. It is known to be the earliest approach that was used for software development and most projects followed the Waterfall approach as it is easy to comprehend and use. The waterfall model occurs in a synchronous manner, because each phase must be completed before the next phase can begin and there is no overlapping in the phases.



The phases of the Waterfall approach are Conception, Initiation, Analysis, Design, Construction, Integration and Testing and Implementation and maintenance. The first phase is Conception, where it triggers when a problem is perceived. It involves identifying goals to be achieved after the problem is solved, estimating benefits in the new system over the current system, and identifying other areas that are affected by the solution.

The second phase is Initiation, where it involves a large scale of study of the customer requirements and defining possible solutions to the customer requirements and cost-benefit justification of these alternatives.

The third phase is Analysis, where it requires carrying out detailed study of the customer requirements and arriving at the exact requirements of the proposed system. Also, it includes freezing the requirements before the design phase begins.

The fourth phase is Design, where it requires translating the identifies requirements into a logical structure.

The fifth, phase is Construction, where it involves combining and testing all the modules developed in the previous phase as a complete system.

The next phase is the Integration and Testing phase. This phase too includes the integration and testing of all modules developed in the previous phase as a complete system.

The final phase is the Implementation and maintenance phase. This phase comprises of converting the new system design into operation, and it may also involve implementing the software system and training the operating staff before the software system is functional.

**Advantages and disadvantages of Waterfall model:**

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| * very easy to understand and also a powerful method of software development. * reduces project time and cost as well as effort. * gives better understanding of the requirements, logic of codes and tests which were conducted on the software. * both the project management stakeholders and developers are aware of the business requirements documentation (BRD) and project management requirements. | * all requirements of the software need to be specified upfront and there is no room for committing mistakes. * changes are not possible because amending something means user will have to recreate the software all over again. * project communications with the clients are extremely limited. * it is a very rigid method which does not entertain any change in requirements and which makes any subsequent functionality changes required extremely difficult and expensive to implement. |

**V-model**

Another software development model would be the V-model, or also known as Verification and Validation model. In V-models, the execution of processes happens in a sequential manner in V-shape. The V-model is an extension of the waterfall model and is based on association of a testing phase for each corresponding development stage. It basically means that for every single phase in the development cycle, there is a directly associated testing phase. V-model is a highly disciplined model and it occurs in a synchronous manner.

The V-shaped model is usually used for small to medium sized projects, where requirements are clearly defined and fixed. Also, when ample technical resources are available with needed technical expertise.



Above is a diagram of V-model. The various phases of the V-model are Requirements, the High-Level Design (HLD), the Low-Level Design (LLD), the Implementation and Coding.

The phase Requirements, involves Business requirement specifications (BRS) and System requirement specifications (SRS). Just like the waterfall model, BRS and SRS begin the life cycle model. However, in V-models, a system test plan is created before a development is started.

For HLD, the phase focuses on system architecture and design. It provides the overview of solution, platform, system, product and process. Also, an integration test is designed in order to test the pieces of the software systems ability to work together.

LLD on the other hand, makes use of the actual software components. This phase is where the actual software components are designed. It defines the actual logic for each and every component of the system. Class diagram with all the methods and elation between classes comes under LLD. In addition to that, component tests are created in this phase as well.

The Implementation phase is where all the coding takes place. Once the coding is complete, the path of execution continues up the right side of the V, where the test plans developed earlier are now put to use.

The last phase of the V-model would be Coding, where module designs are converted into codes by the developers. This phase is usually at the bottom of the V-shaped model.

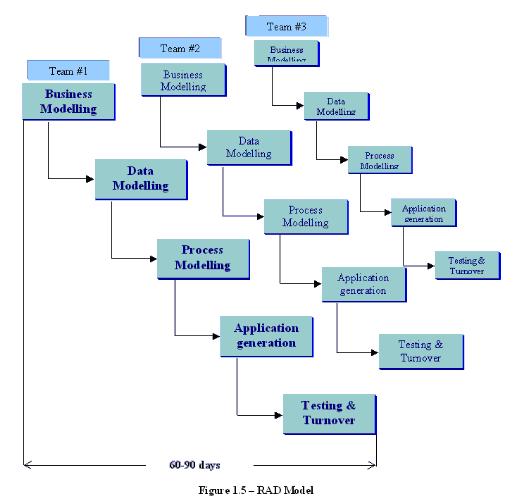
**Advantages and disadvantages of V-model:**

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| * It is simple and easy to use. * Testing activities like planning and test designing happens well before coding. (a lot of time is saved) * Proactive defect tracking – defects are found at early stage * Avoids the downward flow of the defects * Works well for small projects where requirements are easily understood. | * It is very rigid and least flexible. * Software is developed during the implementation phase (no early prototypes of the software are produced) * If user wishes to amend something in the midway, then the test documents along with the requirement documents has to be updated. |

**Rapid Application Development (RAD)**

The third software development model is the Rapid Application Development (RAD) model. It is a type of incremental model. In RAD models, the components of functions are developed in parallel as if they were mini projects. The developments are time boxed, delivered, and then put together into a working prototype. This is so that they can quickly give the customer something to see and use and can receive feedbacks regarding the delivery and requirements in return.

RAD models are usually used when there is a need to create a system that can be modularized in 2-3 months. Also, it is necessary to take into consideration if there is a high availability of designers for modelling, and if the project budget is high. It is important to note that this model should only be chosen if resources with high business knowledge are available, and if there is a need to produce the system in a short span of time, for example in 2-3 months.



Above is a diagram of the RAD-model. The phases in the RAD model are business modelling, data modelling, process modelling, application generation and testing and turnover. The business modelling phase is where the information flow is identified between various business functions.

The data modelling phase is where the information gathered from business modelling is used to define data objects that are needed for the business.

The process modelling phase is where data objects defined in data modelling are converted to achieve the business information flow to achieve some specific business objective. Description are identified and created for CRUD of data objects.

The application generation phase is where automated tools are used to convert process models into code and actual system.

Lastly, the testing and turnover phase is where the testing of new components and all interfaces are involved.

**Advantages and disadvantages of RAD model:**

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| * Reduced development time * Increases reusability of components * Quick initial reviews occur * Encourages customer feedback * Integration from very beginning solves a lot of integration issues | * Dependant on strong team and individual performances for identifying business requirements. * Only system that can be modularized can be built using RAD. * Requires highly skilled developers/designers. * High dependency on modelling skills * Inapplicable to cheaper projects as cost of modelling and automated code generation is very high. |