**Software Development Life Cycle**

It is a framework that define tasks performed at each step in the software development process. SDLC is a structure that is followed by a development team within the software organization. It is used in systems engineering, information systems and software engineering to describe a process for planning, creating, testing and deploying an information system. The life cycle also defines a methodology for improving the quality of the software and the overall development process.

**Software Development Life Cycle Phases**

Software development life cycle models describe phases of the software cycle and the order in which those phases are executed. Where each phase produces deliverables which is required by the next phase in the cycle. The below are 6 phases in every software development life cycle model that are required.

Requirement gathering and analysis

This phase requires business requirement that must be gathered. It is the main focus for project managers and stake holders. Managers, stake holders and users conduct meetings to determine requirements like, who is going to use the system and what are the data should be output by the system. After such requirements are gathered, they will be analysed in the system and development is also studied.

Design

This phase prepares the system and software design which were studied in the first phase. The system design specify hardware and the system requirements also helps in defining overall system architecture. It also serve as the input for the next phase of the model.

Implementation or coding

This phase receives system design documents, the work will be divided in modules or units and the actual coding will be started. The code will be produced and it is the main focus for the developer. This is the longest phase of the software development life cycle.

Testing

Once the code is developed and is tested against the requirements to make sure product is actually solving the needs addressed and gathered during the requirements phase. Unit testing, integration testing, system testing and acceptance testing are done in this phase.

Deployment

This phase will deploy the product to the customer for their usage after all the successful testing are done.

Maintenance

This phase will take care problems upon once customer’s meets actual problems when the customers starts using the developed system and needs to be solved from time to time.

**Waterfall Model**

Waterfall model referred to as a linear-sequential life cycle model. Each phase must be completed fully first before the next phase can begin. Waterfall model is used for the project which is small with no other uncertain requirements. Once the first phase end, a review will then takes place to determine if the project is on the right path and whether or not to continue or discard the project. Waterfall model testing only starts after the development is complete and the waterfall model phases does not overlap.

**Advantages of Waterfall Model**

* Waterfall model is simple and easy to understand and use
* Each phase has specific deliverables and a review process thus making it easy to manage
* Waterfall model works well in smaller projects

**Disadvantages of Waterfall Model**

* It is hard to change something that was not well-thought in the concept stage once the application is in the testing stage
* High amounts of risk and uncertainty
* Poor model for long and ongoing projects

**Iterative Model**

Iterative life cycle model does not attempt to start with a full specification of requirements. Development begins by specifying and implementing just part of the software, which then will be reviewed in order to identify further requirements. The process will then be repeated thus, producing a new version of the software for each cycle of the model.

**Advantages of Iterative Model**

* Detect defects at early stages as iterative model allows building and improving the product step by step
* Less time is spent on documenting and more time given for designing
* Get reliable user feedback when presenting sketches and blueprints of the product to users for their feedback

**Disadvantages of Iterative Model**

* Each phase of iterative is rigid with no overlaps
* Expensive system architecture or design as not all requirements are gathers up front for the entire lifecycle

**RAD Model**

Rapid Application Development model is a type of incremental model. The components or functions are developed in parallel that acts like they were mini projects. Its developments are time boxed, delivered and then assembled into a functioning prototype. It can also quickly give the customer something to see and use so to provide feedback regarding the delivery and the requirements.

**Advantages of RAD Model**

* Reduced development time
* Encourages customer feedback
* Increases the reusability of components

**Disadvantages of RAD Model**

* Requires skilled developers or designers
* Not applicable for cheaper projects as the cost of modelling and automated code generation is very high
* Only system that can be modularized, can only be built using RAD model