SOFTWARE QUALITY ASSURANCE

Software Quality Assurance is a method used for ensuring quality in software engineering processes. It ensures that developed software meets and complies with the standardized quality specifications. SQA is the process within the Software Development_Life Cycle that is used for checking the developed software to ensure it meets the desired quality measures.

Why SQA Born?

JAMES BACH

CAM KENAR

MOST INFLUENCE PERSON In SQA INDUSTRY

Plan: The first phase is planning. An organization needs to plan and establish the process related objectives and determine the processes that are required to deliver a high-Quality end product.

Do: This phase involves the development and testing of Processes. Also, you can do the changes in the process.

Check: In this phase, you can monitor and modify the processes, and check whether it meets the predetermined objectives.

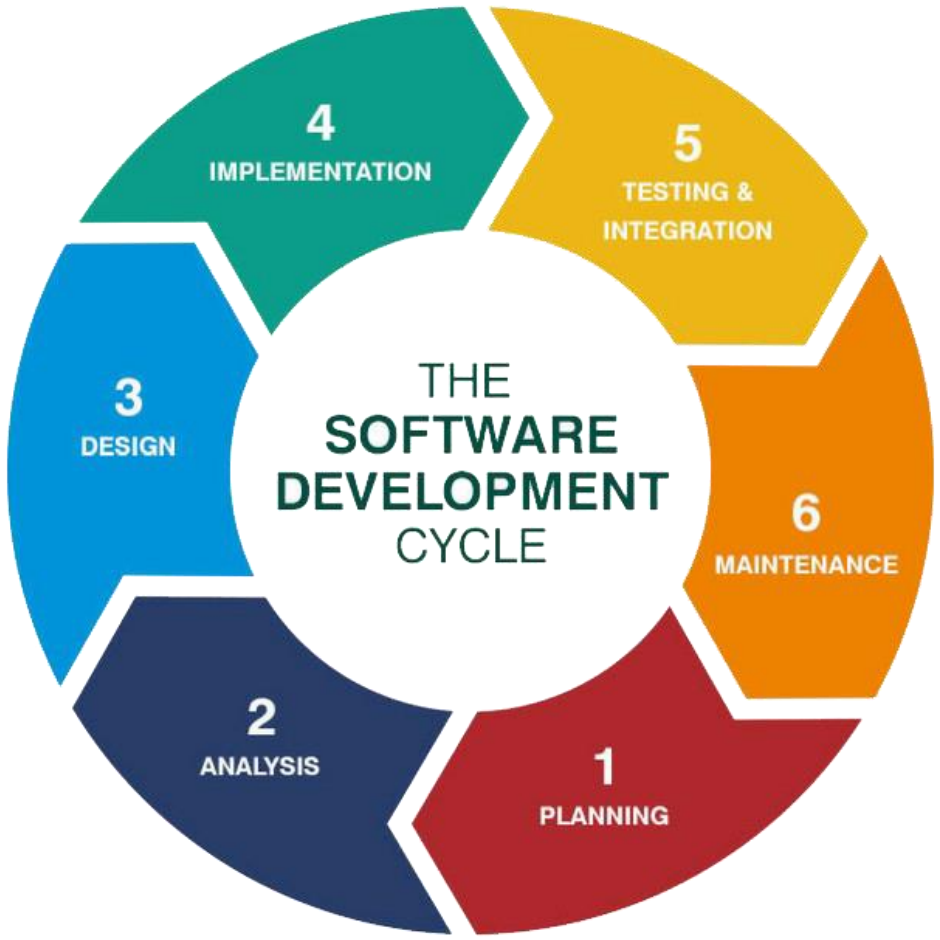
Act: The final phase involves the implementation of actions that are necessary to achieve improvements in the processes.

Quality Assurance	Testing
QA includes activities that ensure the implementation of processes, procedures and standards in context to verification of developed software and intended requirements.	It includes activities that ensure the identification of bugs/error/defects in a software.
Focuses on processes and procedures rather than conducting actual testing on the system.	Focuses on actual testing.
Process-oriented activities	Product-oriented activities.
It is a subset of Software Test Life Cycle (STLC)	Testing is the subset of Quality Control.

SDLC

Software Development Life Cycle

Software Development Life Cycle is a process that produces software with the highest quality and lowest cost in the shortest time possible



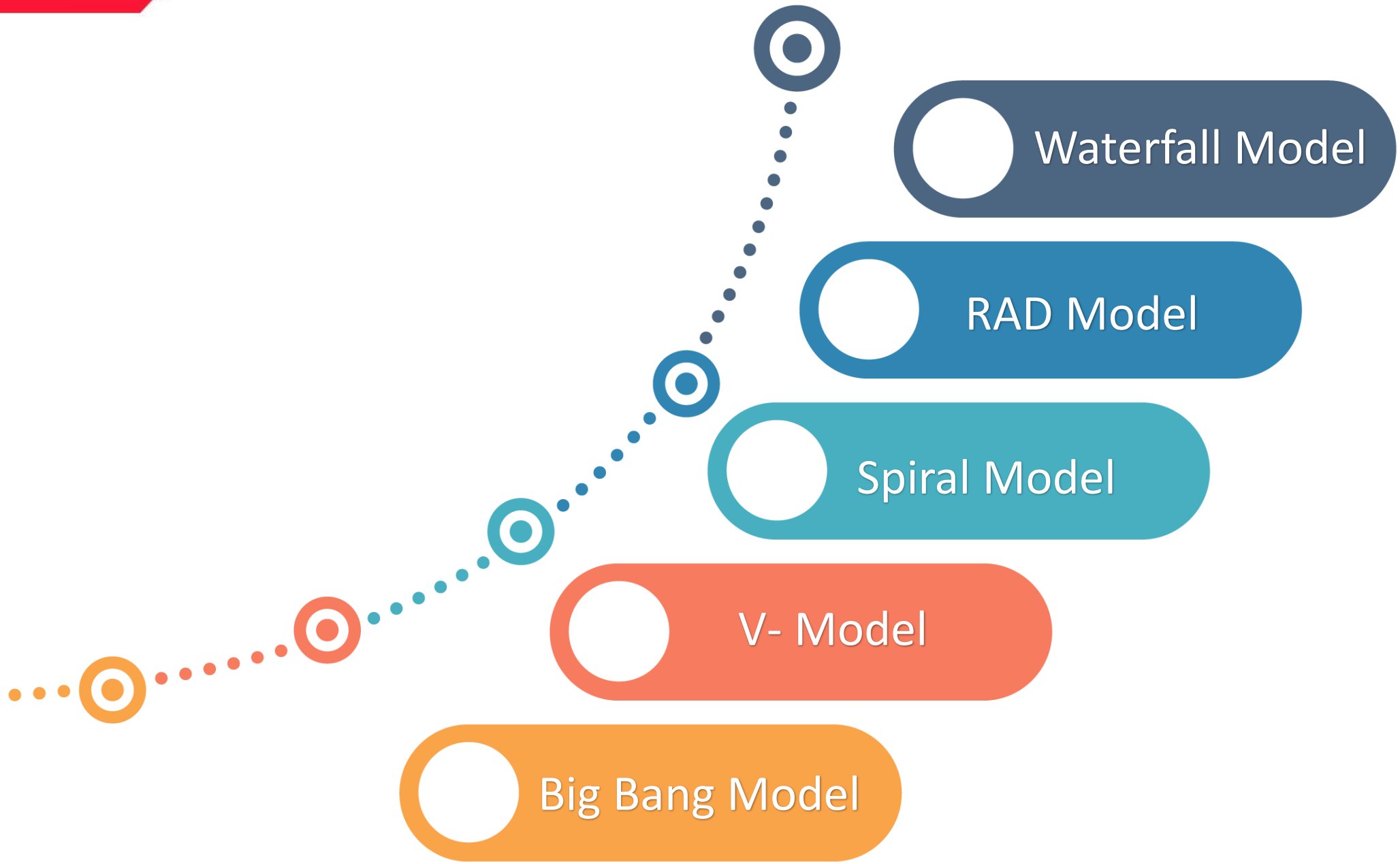
Analysis: Business analyst and Project Manager set up a meeting with the customer to gather all the information like what the customer wants to build, who will be the end-user, what is the purpose of the product.

Design: In this phase, the requirement gathered in the SRS document is used as an input and software architecture that is used for implementing system development is derived.

Implementation: Implementation/Coding starts once the developer gets the Design document. The Software design is translated into source code. All the components of the software are implemented in this phase

Testing: Testing starts once the coding is complete and the modules are released for testing.

Maintenance: After the deployment of a product on the production environment, maintenance of the product i.e. if any issue comes up and needs to be fixed or any enhancement is to be done is taken care by the developers.


Waterfall Model
RAD Model
Spiral Model
V- Model
Big Bang Model

WATERFALL MODEL

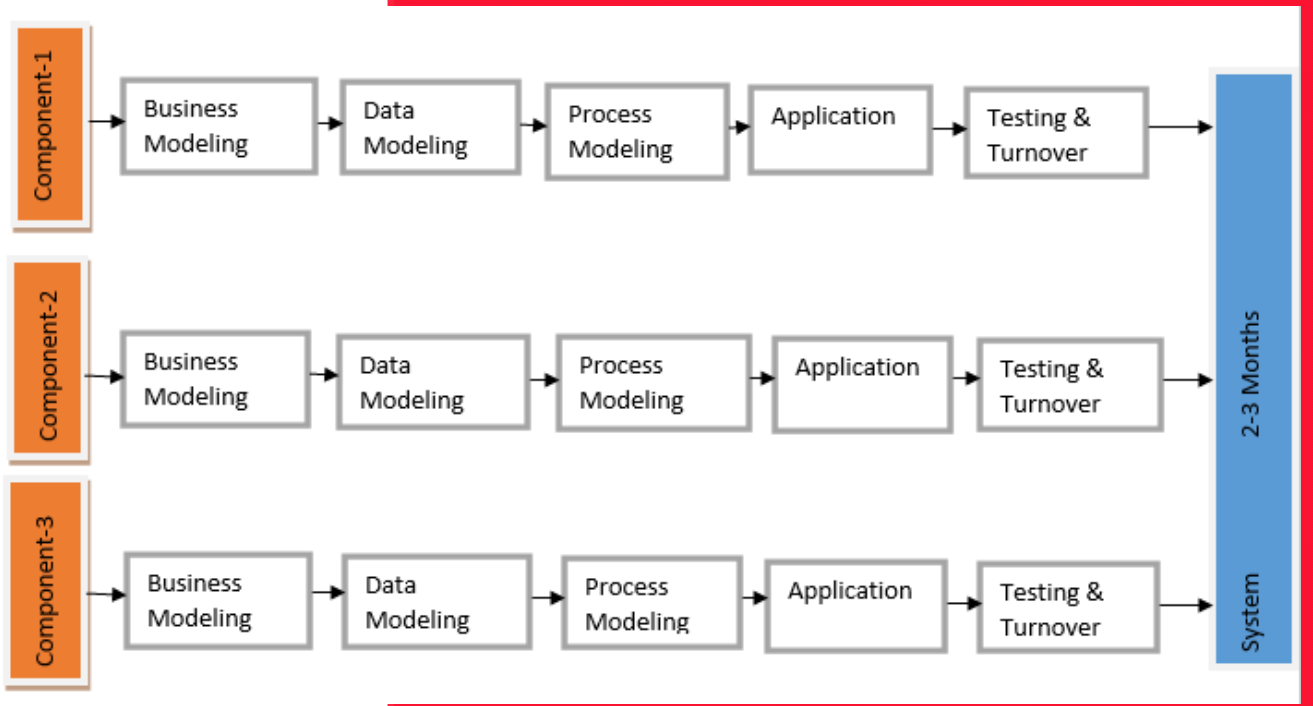
- Analysis
- Requirement Specification
- Design
- Development
- Testing
- Deployment

“This SDLC model is the oldest and most straightforward. With this methodology, we finish one phase and then start the next. Each phase has its own mini-plan and each phase “waterfalls” into the next. The biggest drawback of this model is that small details left incomplete can hold up the entire process”

RAD MODEL

Rapid Application Development model is a software development process based on prototyping without any specific planning or minimum planning.

- Business
- Data
- Process
- Application
- Testing & Turnover

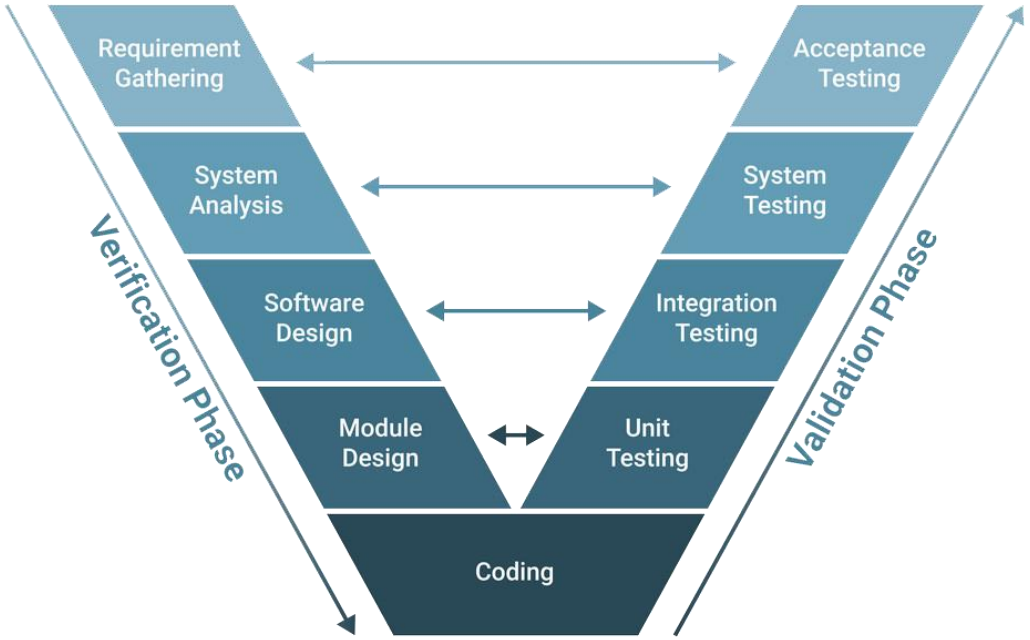


In RAD model, there is less attention paid to the planning and more priority is given to the development task. It focuses on input-output source and destination of the information. Development of each module requires basic Waterfall steps.

SPIRAL MODEL

The most flexible of the SDLC models, the spiral model is similar to the iterative model in its emphasis on repetition. The spiral model goes through the planning, design, build and test phases over and over, with gradual improvements at each pass. We use the model when the requirements kept changing. We don’t know when the projects will be end.

V MODEL



V-Model also referred to as the Verification and Validation Model. In this, each phase of SDLC must complete before the next phase starts. It follows a sequential design process same as the waterfall model. Testing of the device is planned in parallel with a corresponding stage of development.

Verification: Verification is the process of confirming that software meets its specification.

Validation: Validation is the process of confirming that software meets the user's requirements.

Iterative MODEL

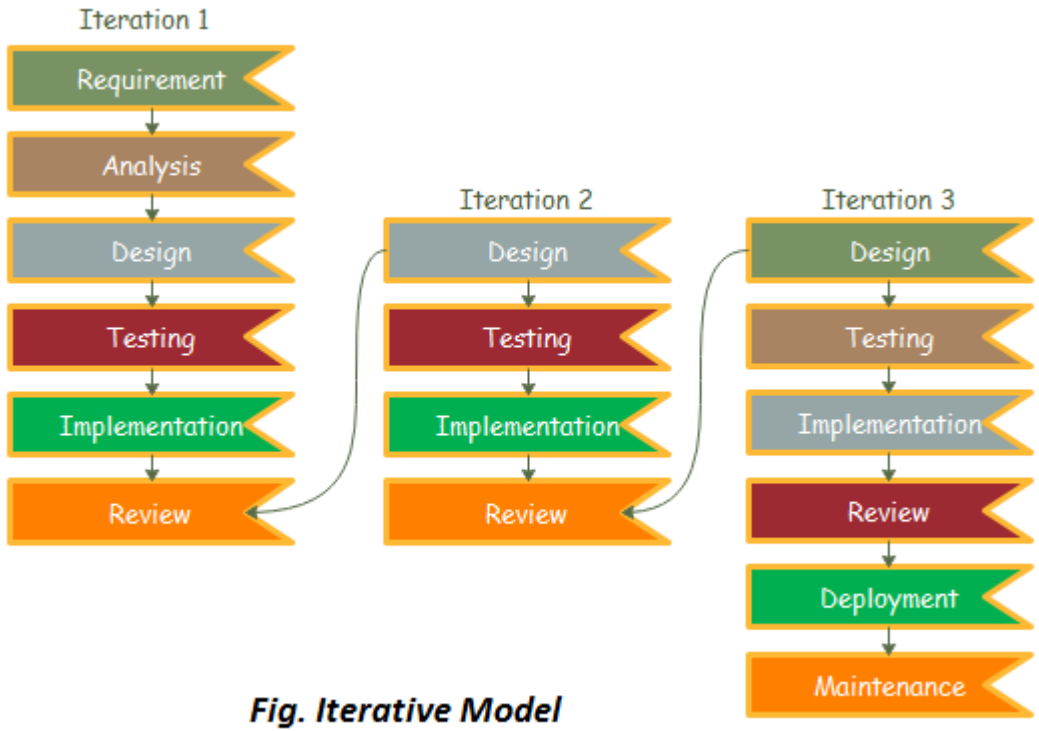
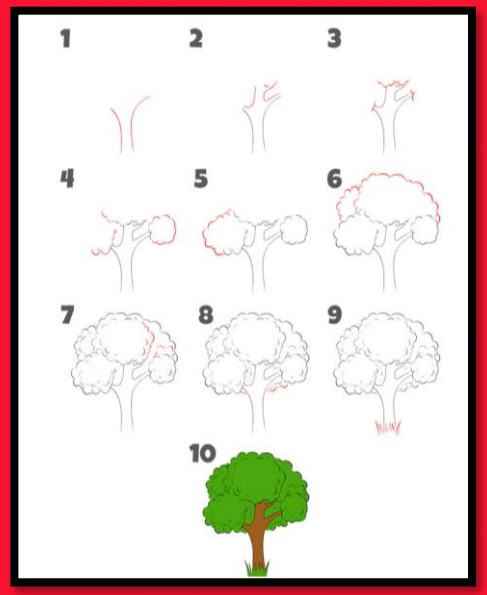
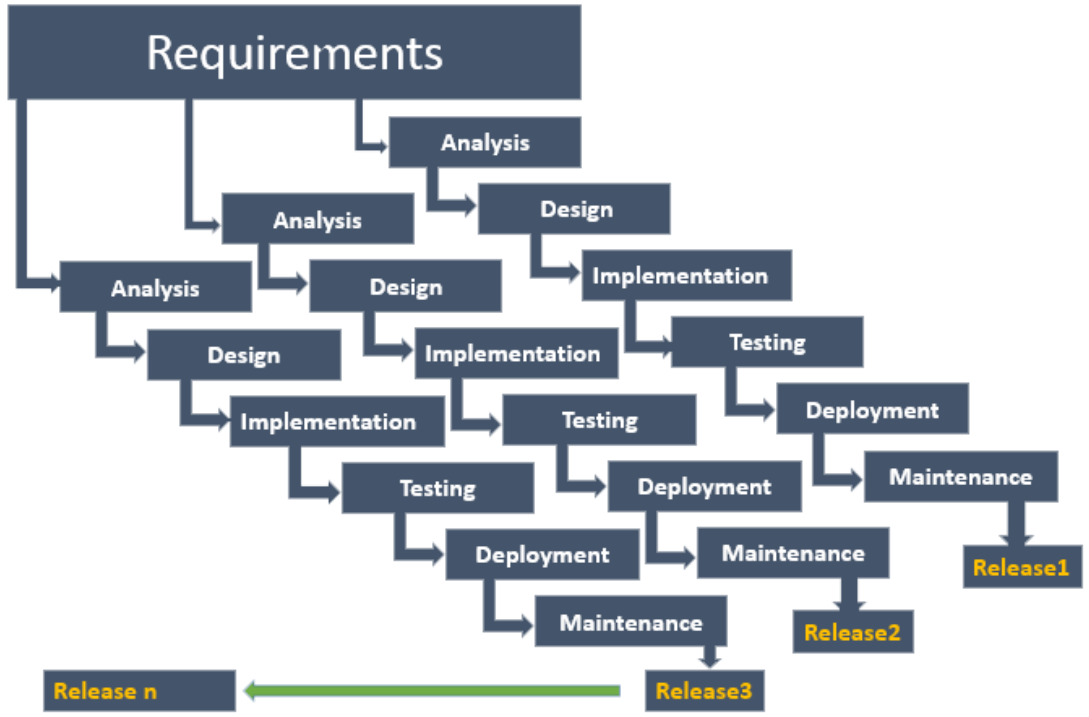


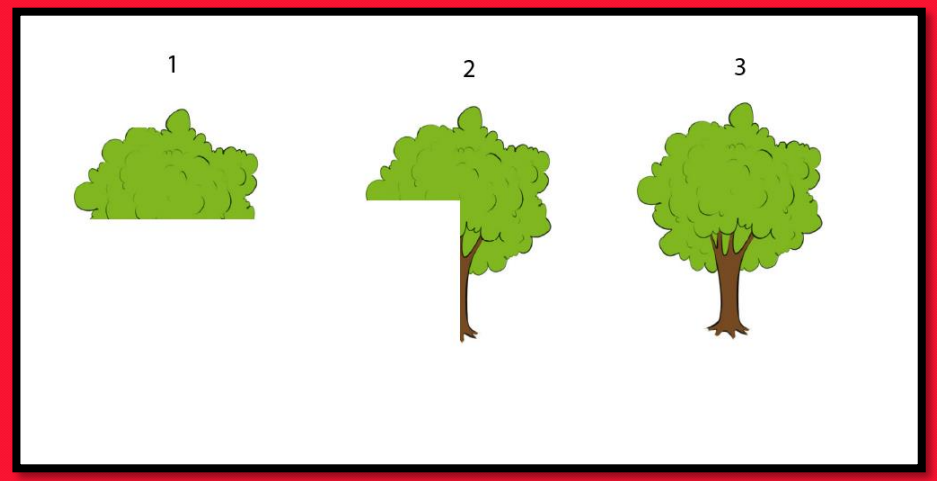
Fig. Iterative Model



In this Model, you can start with some of the software specifications and develop the first version of the software. After the first version if there is a need to change the software, then a new version of the software is created with a new iteration. Every release of the Iterative Model finishes in an exact and fixed period that is called iteration.

INCREMENTAL MODEL





Incremental Model is a process of software development where requirements divided into multiple standalone modules of the software development cycle. In this model, each module goes through the requirements, design, implementation and testing phases. Every subsequent release of the module adds function to the previous release. The process continues until the complete system achieved.