1.Waterfall model

Table of contents:

What is waterfall model?

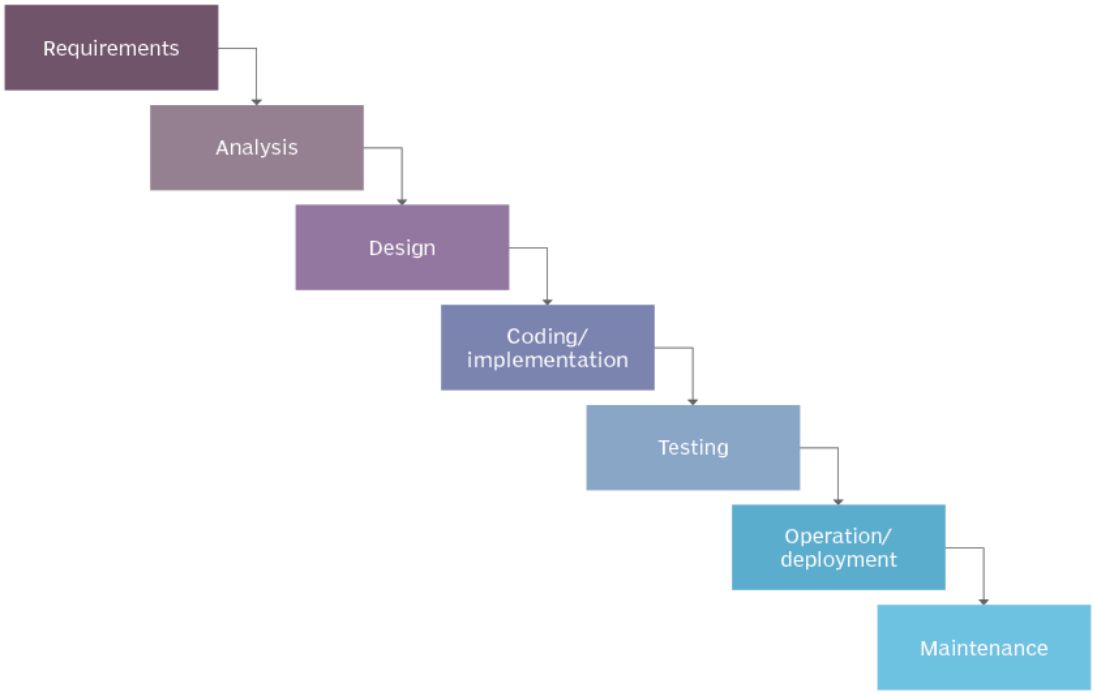
The Waterfall Model was first Process Model to be introduced. It is also referred to as a **linear-sequential life cycle model**. It is very simple to understand and use. In a waterfall model, each phase must be completed fully before the next phase can begin.

This type of **software development model** is basically used for the project which is small and there are no uncertain requirements.

In this model **software testing** starts only after the development is complete. In **waterfall model phases** do not overlap.

Diagram for waterfall model:

Waterfall model



Advantages of waterfall model:

- This model is simple and easy to understand and use.
- It is easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
- In this model phases are processed and completed one at a time. Phases do not overlap.
- Waterfall model works well for smaller projects where requirements are clearly defined and very well understood.

For user:

- Longer operational life.
- Meets user needs and requirement.
- More flexible.

For employees:

- Less needs for supporting system.
- Excellent documentation.

DISADVANTAGES:

- Once an application is in the **testing** stage, it is very difficult to go back and change something that was not well-thought out in the concept stage.
- No working software is produced until late during the life cycle.
- High amounts of risk and uncertainty.
- Not a good model for complex and object-oriented projects.
- Poor model for long and ongoing projects.
- Not suitable for the projects where requirements are at a moderate to high risk of changing.

For users:

- More expensive to maintain.
- More difficult to use.

For employees:

- Requires more training of employees.
- Difficult to customize.
- Requires extensive research into user needs

When to use the waterfall model:

- This model is used only when the requirements are very well known, clear and fixed.
- Product definition is stable.
- Technology is understood.
- There are no ambiguous requirements
- Ample resources with required expertise are available freely
- The project is short.

Examples:

Customer Relationship Management (CRM) systems,

Human Resource Management Systems (HRMS),

Supply Chain Management Systems,

Inventory Management Systems &Point of Sales (POS) systems for Retail chains etc.

2.Iterative model

Content of table shown:

What is iterative model?

The iterative process model is a software development life cycle (SDLC) approach in which the initial development work is conducted based on initial requirements that are clearly defined, and subsequent features are added to this base software product through iterations until the final system is completed.

to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).

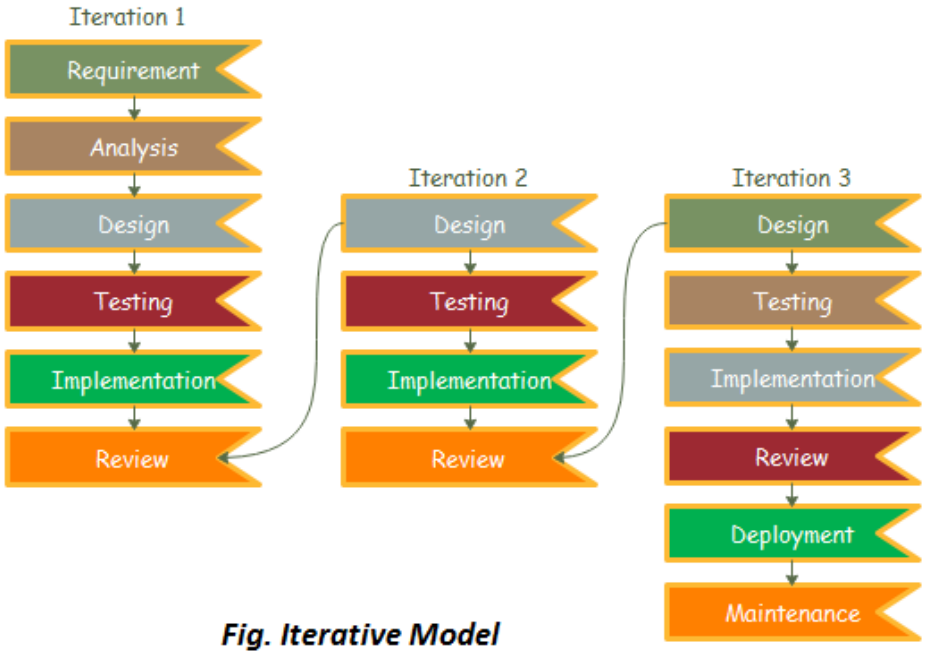


Fig. Iterative Model

Advantages of iterative model

- Testing and debugging during smaller iteration is easy.
- A Parallel development can plan.
- It is easily acceptable to ever-changing needs of the project.
- Risks are identified and resolved during iteration.
- Limited time spent on documentation and extra time on designing.
- Customer feedback is based on working products rather than technical specifications.

Disadvantages of iterative model

- It is not suitable for smaller projects.
- The entire procedure is difficult to manage.
- There is a need for proper management.
- More Resources may be required.
- Design can be changed again and again because of imperfect requirements.
- Requirement changes can cause over budget.
- Project completion date not confirmed because of changing requirements.

When to use the iterative model?

1. When requirements are defined clearly and easy to understand.
2. When the software application is large.
3. When there is a requirement of changes in future.

Example:

- Prototyping,
- Rational Unified Process (RUP),
- agile development,
- Rapid Application development.

3.spiral model

Table of content shown:

What is spiral model?

The spiral life cycle model is a iterative software development model which is the **generally implemented in high risk project**.

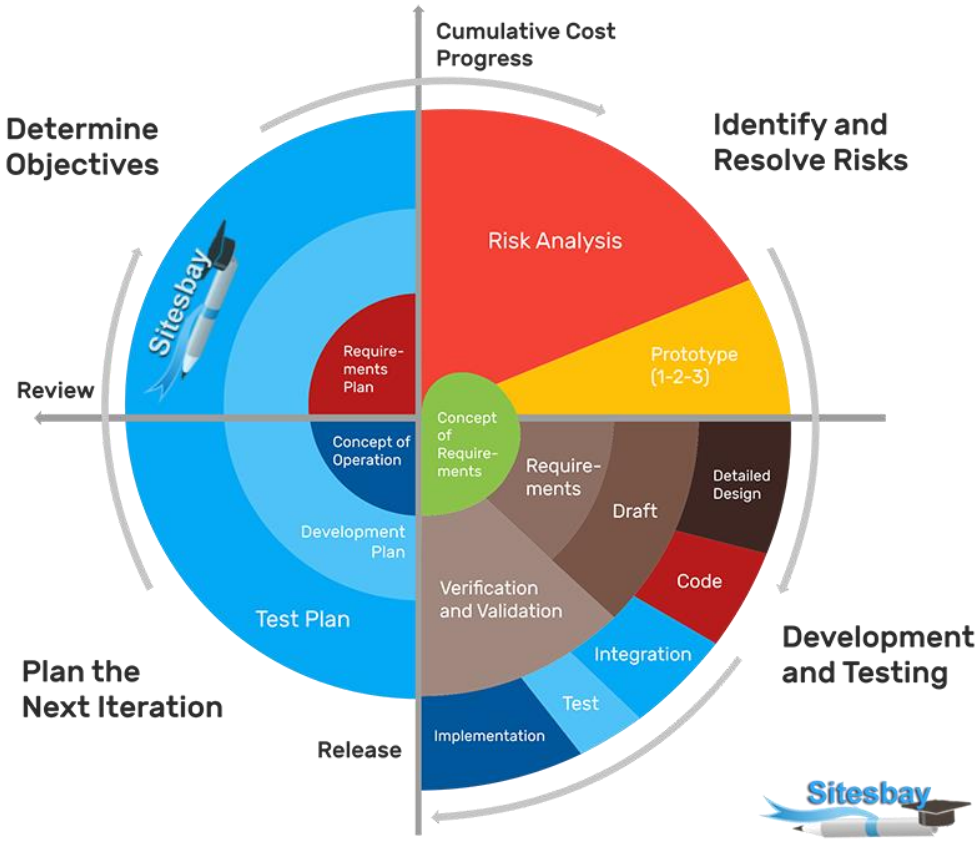
The Spiral Model allows the product to be rolled out and refined in each phase of the spiral, with the ability to build prototypes in each stage. **A prototype is created at the beginning of each phase as a risk management technique**

It's a combination of waterfall model and iterative model

Each phase of spiral model starts with requirement analysis and end with client reviewing

Software is developed in a series of incremental release.

Spiral model image:



Advantages of spiral model:

1. The spiral model is perfect for projects that are **large and complex** in nature as continuous prototyping and evaluation help in mitigating any risk.

2. Because of its **risk handling ability**, the model is best suited for projects which are very critical like software related to the health domain, space exploration, etc.

3. This model supports the client feedback and **implementation of change requests** (CRs) which is not possible in conventional models like a waterfall.

4. Since customer gets to see a prototype in each phase, so there are higher chances of customer satisfaction.

Dis advantages of spiral model

1. Because of the prototype development and risk analysis in each phase, it is very **expensive and time taking**.

2. It is **not suitable for a simpler and smaller** project because of multiple phases.
3. It requires **more documentation** as compared to other models.
4. Project **deadlines can be missed** since the number of phases is unknown in the beginning and frequent prototyping and risk analysis can make things worse.

When to use Spiral model:

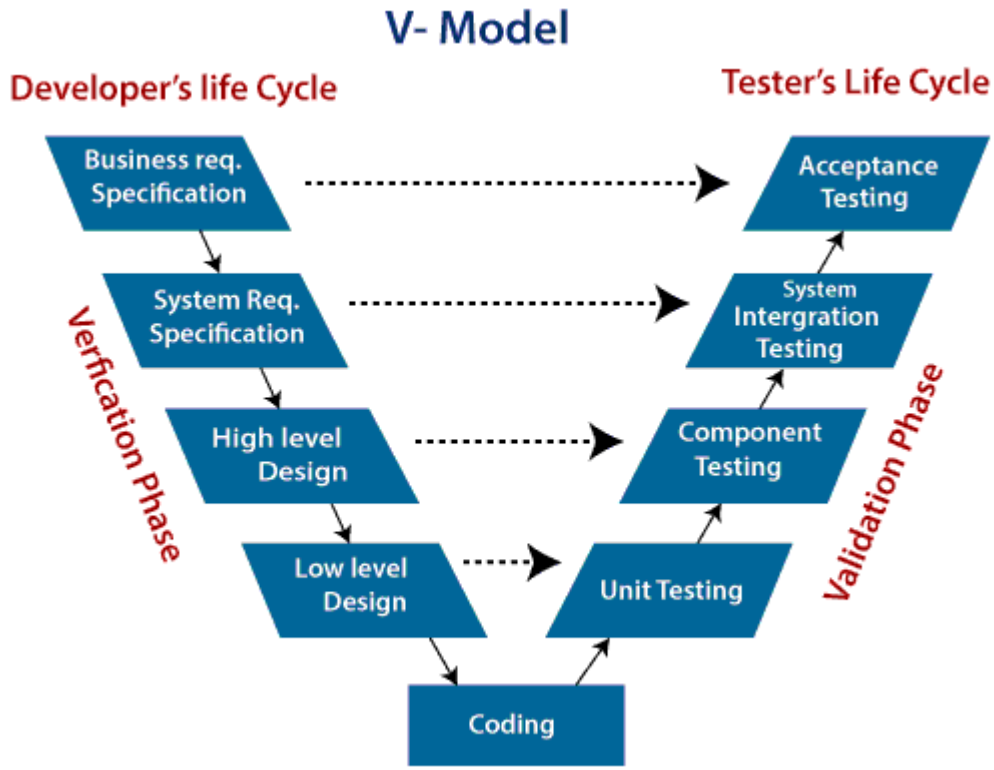
- When costs and risk evaluation is important.
- For medium to high-risk projects.
- Long-term project commitment unwise because of potential changes to economic priorities.
- Users are unsure of their needs.
- Requirements are complex.
- New product line.

Example

The best-suited example for the spiral model is **MS-Excel** because MS-Excel sheet having several cells, which are the components of an excel sheet.

4.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: It involves a static analysis method (review) done without executing code. It is the process of evaluation of the product development process to find whether specified requirements meet.

Validation: It involves dynamic analysis method (functional, non-functional), testing is done by executing code. Validation is the process to classify the software after the completion of the development process to determine whether the software meets the customer expectations and requirements.

There are the various phases of Validation Phase of V-model:

Unit Testing: A unit is the smallest entity which can independently exist, e.g., a program module.

Integration Testing: Integration tests are performed to test the coexistence and communication of the internal modules within the system

System Testing: Check the entire functionalities of system.

Acceptance Testing: a quality assurance (QA) process that determines to what degree an application meets end users' approval.

ALPHA: it's the initial phase of validating whether a new product will perform as expected.

Its check with developer side.

BETA: its second phase in which sampling of intended audience tries the product one.

Its check with end user side.

When to use V-Model?

- When the requirement is well defined and not ambiguous.
- The V-shaped model should be used for small to medium-sized projects where requirements are clearly defined and fixed.
- The V-shaped model should be chosen when sample technical resources are available with essential technical expertise.

Advantage (Pros) of V-Model:

- Easy to Understand.
- Testing Methods like planning, test designing happens well before coding.
- This saves a lot of time. Hence a higher chance of success over the waterfall model.
- Avoids the downward flow of the defects.
- Works well for small plans where requirements are easily understood.

Disadvantage (Cons) of V-Model:

- Very rigid and least flexible.
- Not a good for a complex project.
- Software is developed during the implementation stage, so no early prototypes of the software are produced.
- If any changes happen in the midway, then the test documents along with the required documents, has to be updated.

Example:

- Projects initiated from a request for proposals (RFPs), the customer has a very clear documented requirements.
- Military projects.
- Mission Critical projects, for example, in a Space shuttle.
- Embedded systems.
- Projects with defined and clear requirements.

5.Big bang model

In this model, developers do not follow any specific process. Development begins with the necessary funds and efforts in the form of inputs. And the result may or may not be as per the customer's requirement, because in this model, even the customer requirements are not defined.

This model is ideal for small projects like academic projects or practical projects. One or two developers can work together on this model.

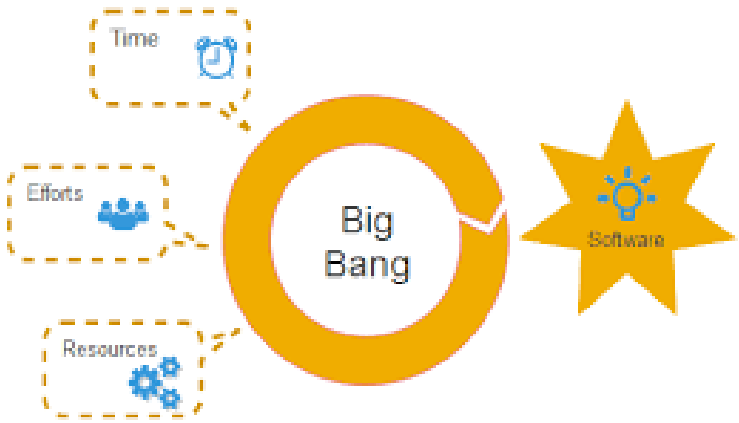


Fig. Big Bang Model

Advantages:

- There is no planning required.
- Simple Model.
- Few resources required.
- Easy to manage.

- Flexible for developers.

Disadvantages:

- There are high risk and uncertainty.
- Not acceptable for a large project.
- If requirements are not clear that can cause very expensive.

When to use the big bang model:

As we discussed above, this model is required when this project is small like an academic project or a practical project.

This method is also used when the size of the developer team is small and when requirements are not defined, and the release date is not confirmed or given by the customer.

Difference Between Waterfall Model, Iterative Model, Spiral Model, V Model & RAD Model

Waterfall Model	Iterative Model	Spiral Model	Big bang model	V Model	RAD Mod
The Waterfall model is **simple and easy.**	It's **simple and easy**	The spiral model is a lot **more complex**	Big-bang model is **simple**	**Easy to Understand.**	**Easy to understand**
The waterfall model works in a **sequential method**.	The iterative model works in an **incremental method**	While the spiral model works in the **evolutionary method.**	The Big Bang model is **a start-from-scratch** SDLC model	It is also a **sequential execution process**.	It's an **increme model**

Flexibility to change in waterfall model **is Difficult**.	**Flexibility** to change in incremental **model is Easy**.	**Flexibility** to change in spiral model is **not Difficult.**	**Flexibility** to change **in big ba**ng **model is Easy**	Flexibility of V-model is **Little flexible**.	This model **flexible fo change.**
Waterfall model is **comparatively inexpensive.**	Cost of incremental model is **low**	While cost of spiral model is **very expensive**	It is **cost-effective**	Cost of V-model is **expensive.**	Cost is **low**
The waterfall model is applicable for **small projects**	It's also used for **large project**	the Spiral model is used for **large projects**.	applicable for **small projects**	applicable for **small projects**	Not appropriat handle the **lar project**
In waterfall model requirements and **early- stage planning** is necessary.	Iterative model requirements and **early - stage planning** is necessary	While in spiral model requirements and **early - stage planning is necessary** if required	**No early-stage planning**	While in spiral model requirements and **early-stage planning is necessary** if required	**No early-sta planning**
It requires **least maintenance**.	Its **maintainable**	It requires **typical maintenance**	Its **easy to maintenance**	Its **maintenance is less**	**Easy to maintenanc**
Time frame is **very long**	Time frame is **long**	Time frame is **long**	**Short** time frame	Time frame is **long**	**Short time** fra
No overlapping phase	Overlapping **phase is there**	**No overlapping** phase	**Overlap of system rollout**	**No overlapping** phase	Overlappin **phases is the**
After completion of development phase, the testing will start	**After every iteration** the testing will start	At the **end of engineering phase**, the testing will start	Testing is an integration testing strategy wherein ***all units are linked at***	**After completion of development** the testing will start	**After complet of developm** the testing w start

			once, resulting in a complete system		
It's based on **linear framework** type	It is based on **linear and iterative framework** type.	It is based on **linear and iterative** framework type.	**don't move in linear** way	V-model's steps **don't move in linear** way.	It's based on **li framework** t
There is **high amount risk** in waterfall model.	There is **low amount risk** in iterative model.	There is **medium to high amount risk** in spiral model.	There is **high amount risk** in big bang model	There **is high rigid** in v model.	There is **lo amount risk** RAD mode
Reusability is **extremely unlikely**.	To a **certain extent, reusability** is possible.	To a **extent, reusability certain** is possible.	Reusability is **extremely unlikely**.	reusability is **possible**	reusability i **possible**