



Varsity College

ST10075585 Ice Task 1



CREATED BY
TemplateLAB



CREATED BY Nsikelelo Tsoelopele Kumalo

In software development, the Waterfall paradigm and the Incremental paradigm are frequently employed. These models are intended to ensure that software is developed in a systematic, organised, and efficient manner. Read on to learn more about the Waterfall and Incremental models, as well as how they differ from one another.

What is the Incremental Model?

The Incremental Model is a framework for software development wherein the complete system is partitioned into multiple sub-development phases, each accompanied by its corresponding testing phase. These phases, encompassing development and testing, unfold in a sequential manner, thereby endowing the model with a sequential/parallel essence. Due to the requisite functionality of these sequential phases, the developmental cost is elevated in comparison to the Waterfall Model.

The intricate nature of the incremental model surpasses that of the waterfall model. The likelihood of encountering a considerable number of defects during application development is diminished, as testing is conducted concurrently with the application's development.

The incremental approach to software development entails the fragmentation of a project into more manageable components termed "increments." These increments build upon one another, introducing new functionality and features progressively until the final product reaches fruition. This methodology grants greater flexibility, allowing seamless integration of updates into the development process. (Panigrahi, 2022)

What is Waterfall Model?

The Waterfall Model represents the classical approach to software development, wherein every stage of application creation unfolds sequentially. In this model, the entire process is compartmentalized into distinct phases, and a linear and step-by-step progression is followed, ensuring completion of each project phase prior to advancing to the subsequent one. The testing phase is deferred to the culmination of development. Termed as the classical or traditional model, the Waterfall Model is generally deemed inadequate for managing extensive projects.

The stages within the Waterfall Model conventionally encompass requirements gathering and analysis, design, implementation, testing, deployment, and maintenance. The nomenclature "waterfall" is drawn from the seamless transition akin to a cascading waterfall from one phase to another. A noteworthy advantage of the Waterfall Model is its facilitation of clear and concise communication amongst team members, thanks to the well-defined objectives and deliverables for each phase. Nevertheless, its rigidity can impede flexibility and hinder the assimilation of modifications once a phase has reached its conclusion. (Panigrahi, 2022)

Difference between Incremental Model and Waterfall Model

Both the waterfall and incremental models are prescriptive process models. However, they can be distinguished in various ways, including the flexibility of the given software, the risk associated, software testing, maintenance, customer participation, and so on.

Basis	Waterfall Model	Incremental Model
Definition	The Waterfall Model is often referred to as the Classical/Traditional Model. It is also known as a linear-sequential life cycle model since all phases in this model are completed one by one linearly. In this model, we receive software after all coding phases are completed. This model is primarily applied to modest projects. In the waterfall paradigm, there is only one cycle.	Multiple development cycles occur in the Incremental Model, and these cycles are separated into smaller modules. In general, the first module produces working software in an incremental paradigm. Each succeeding module release adds functionality to the prior iteration. In an incremental model, the procedure is repeated until the entire system is completed.
Working version	At the end of the model's life cycle, the working version of the software is provided.	Each iteration includes a working version of the software.
Workflow	A sequential workflow is proposed by this paradigm.	This approach suggests a sequential and parallel process.
User Involvement	The first phase of the model, communication, involves user participation.	The user is involved in the first phase of each iteration.
Feasibility	This process approach does not allow for software modifications.	This process model conveniently handles software changes.
Team Size	The team size is large.	Does not require a large team.
Documentation	Overemphasises documentation.	Documentation is provided, although it is insufficient.
Maintenance	This model requires the least amount of upkeep.	This model encourages upkeep.
Testing and Validation	Each level of development is followed by its own testing. If any validation is required, it could be implemented during this phase.	Following the completion of development, testing is carried out. As a result, if any missing validation needs to be implemented, that phase of development must be identified, and that validation implemented.
Retracking	Not possible	Possible
Type/Nature	The phases, i.e., development and testing, are carried out sequentially, hence the process is sequential/parallel.	The process is sequential/parallel since it occurs in a sequential order. It is a relatively linear sequential design technique, as each phase must be finished before

		moving on to the next. As a result, the nature of this model is Continuous.
Cost and Complexity	Because linear development involves only one phase of development, the cost and complexity are lower than in the Incremental Model.	Because consecutive phases must be functional, the cost is higher than in the Waterfall Model. Furthermore, the complexity exceeds that of the Waterfall paradigm.
Defects	The likelihood of a total number of flaws in application development is low because testing is done concurrently with development.	Because testing is done after development, the likelihood of a total number of flaws in application development is large.
Reusability	Reusability is the least possible in the waterfall model.	Reusability is possible to some extent in the incremental model.
Customer Control over Administrator	The customer has the least control over the administrator in the waterfall model.	The customer has more control over the administrator in comparison to the waterfall model.
Framework Type	The linear framework type is used in the waterfall model.	Linear with iterative framework type is used in the incremental model.
Customer Involvement	The customer is involved only at the beginning of development.	In the incremental model, customer involvement is intermediate.
Number of Cycles	There is only one cycle in the waterfall model.	Multiple development cycles take place in the incremental model.
Overlapping of Phases	Overlapping of phases is not possible in the waterfall model.	Overlapping of phases is possible in the incremental model.
Returning to the Previous Stage/Phase	Returning to the previous stage/phase is not possible in the waterfall model.	Returning to the previous stage/phase is possible in the incremental model.
Testing phase	Testing is done in the waterfall model after the completion of the coding phase.	Testing is done in the incremental model after every iteration of the phase.
Handling Large Projects	The waterfall model can't handle large projects.	The incremental model also can't handle large projects.
Waiting Time for Running Software	There is a long waiting time for running software in the waterfall model.	There is a short waiting time for running software in the incremental model.
Amount of Risk	There is a high amount of risk in the waterfall model.	There is a low amount of risk in the incremental model.
Early-Stage Planning	Early-stage planning is necessary for the waterfall model.	Early-stage planning is necessary for the incremental model.

(Being Intelligent, 2020) (MKS075, 2022)

The most significant difference is that in an Incremental Model, the entire development phase is divided into several sub development phases with corresponding testing phases, whereas in a Waterfall Model, each phase, after completion, flows into the next and the entire testing part is left to be done at the end of the development.

(Panigrahi, 2022)

Bibliography

Being Intelligent, 2020. *Difference Between Waterfall Model and Incremental Model*. [Online]
Available at: <https://beingintelligent.com/difference-between-waterfall-model-and-incremental-model.html>
[Accessed 18 August 2023].

MKS075, 2022. *Difference between Waterfall model and Incremental model*. [Online]
Available at: <https://www.geeksforgeeks.org/difference-between-waterfall-model-and-incremental-model/>
[Accessed 18 August 2023].

Panigrahi, K. K., 2022. *Difference between Incremental Model and WaterFall Model*. [Online]
Available at: <https://www.tutorialspoint.com/difference-between-incremental-model-and-waterfall-model#:~:text=The%20most%20significant%20difference%20that,the%20next%20and%20the%20entire>
[Accessed 18 August 2023].