JYOTI NIVAS COLLEGE AUTONOMOUS

Post Graduate Center

Department of Computer Science(PG)

E-Journal

On

Software Engineering Scenario

ISSUE 3

OCTOBER 2024

Software Engineering Scenario 1: A global e-learning company is embarking on a project to develop a next generation online learning platform.

Reshmi E

Aneena Thomas

Communication:

In the communication stage, the project team defines the scope, Objectives and requirements for the e-learning platform project. Key features such as courses across various disciplines, interactive learning materials, personalized learning paths and advanced assessment features. The team also establishes a timeline, allocates resources and defines the project's budget.

Requirement gathering and analysis:

We have to understand the need of e-learning company and learners. We have to gather all requirements through Various modes like, Interviews surveys, and workshops. We have to analyse these requirements to identify features like course variety and learning materials. Engage with stakeholders, educators, learners, administrators to understand their needs, define features like course management, user profile, interactive materials, tools, and personalization options. The primary goal of this space is to collect data requirements from the client or stakeholders. This involves understanding their needs, decisions, constraints, and expectations for this software or system being developed. The project manager and possibly other team members engage with the client to elicit and document requirements. This would involve interview, workshops, and discussions to ensure a comprehensive induction. Next, Phase requirement definition. Requirements are defined and specified in detail. These include prioritization, ensuring they are testable and measurable, and resolving any ambiguities. The requirements are reviewed with stakeholders to ensure they accurately meet their needs and accept actions. Verifications are used to ensure that the requirements align with the project goals and scope. Once the requirements are defined and validated, a baseline agreement is typically reached between as they go to deliver the derived results based on feedback from stakeholders. The manager and their own insights team do the same thing giving each other feedback during development as well as at the end of each sprint in what is called a retrospective. The sprint continues until the product is delivered.

Planning:

During the analysis stage, the development team conducts a thorough analysis of the requirements gathered in the planning stage. Break down the functionalities into smaller components such as uses management, course management, content delivery, assessment modules and analytics. Identify potential risks such as technical feasibility of interactive features, scalability issues with a large number of concurrent users, and content diversity challenges. In this stage the development team will comprehensive examination of the requirement collected during the Planning phase. They decompose the requirements into Small tanks,

access potential risks and establish technical specifications. This Phase will assure that the prepared solution is in line with the business. Requirements successfully implemented.

System Design:

As a design first we have to create blueprint of that platform. Design database schema and user interface. Show how courses will be structured, how interactive elements will work and how assessments will be conducted. In the design stage, the team creates a detailed blueprint for the e-learning platform. This includes designing the architecture of the system, main creating wireframes or mock-ups of the user interface, and designing data models and relationships. This include designing overall architecture of the online learning platform considering scalability & integration paints, creating wire frames or prototype of the user interface (UI) to ensure intuitive navigation & usability, defining data models to support course management user profile, content storage and analytics and designing interactive learning materials and assessment features to enhance user engagement and learning Outcomes. Create a blueprint for the plot based on gathered requirements architecture design to determine system components, front-end, back-end, database, dataflow, even interface, API, and integration, system architecture, diagram, database schema, interface, matchups, a prototype.

Implementation:

Develop the platform based on the design Specifications. Write code for frontend, backend and any other necessary components. Implement features such as course browsing user authentication. Once the design is finalized, the development team begins coding and building the e-learning platform according to the specializations outlined in the design stage. Translate the design into actual code and develop the platform writing code based on the chosen technology like ReactJS or Analyzer for back-end like NodeJS or Django, integrating components and ensuring scalability and performance.Continuous integration and version control practices are employed to manage the development process, efficiently. Employs continuous integration practices to ensure that new code integrations do not disrupt existing functionalities.

Testing:

Testing involves in verifying that the platform works correctly and meets requirements. Various types of testing like unit testing, integration testing, System testing and acceptance testing, testing functionality across different browser and devices. The quality assurance team conducts various types of testing to ensure that the e-commerce platform functions correctly and meets the specialized requirements. This includes unit testing, integration testing, system testing, and user acceptance testing. Any defects discovered during testing are reported back to the development team for resolution. Various testing like unit testing, technical individual components, integration testing, technical integration modules, system testing, testing the entire system, and acceptance testing, validating against user requirements, test cases, test reports, identified and fixed defects.

Deployment:

After successful testing, the e-learning platform i**s** ready for deployment. It is deployed to production servers, configured for optimal performance and made accessible to users. Deployment may occur in phases or at all at once, depending on project requirements. Release the platform for use by instructors and learners, prepare for deployment by servers database and network setting. Prepare the platform for release to production environment, plan deployment strategies, set up consumers, configure database, install software, and perform final deployed platform activated area available to users.

Maintenance & Support:

Post deployment, the e-learning platform enters maintenance stage. It is monitored and maintained to ensure continued functionality and. performance. This involves addressing bugs, applying updates, and making enhancements based on user feedback and evolving business needs. Ensure ongoing performance and user Satisfaction provide technical Support to user queries and issues. Maintain the platform, performance & security, update the platform to add new feature, fix bugs and improve performance based on user feedback. The maintenance stage may continue indefinitely as the platform evolves over time. Ensure ongoing functionality, fix bugs, and enhance features as needed. Provide more support, motivate performance, call the user, feed bails, and implement update and purchases regular updates, bug fixes, and new feature release.

Software Engineering Scenario 2: The stages of SDLC that describes "The project manager meets with the client to gather requirements and understand the scope of the project"

Aarthi J

Anusha C.C

This scenario belongs to the planning and analysis stage of the software development lifecycle SDLC. In this stage, the project manager meets with the client to gather detailed requirements, understand the client's needs, and define the scope of the project.

Current phase Planning and Analysis:

Planning is a crucial step in everything, just as in software development. In this same stage, requirement analysis is also performed by the developers of the organization. This is obtained from the customer's inputs and any departments on surveys. This information from this analysis forms the building blocks of the basic project. The quality of the project is a result of planning. In this phase, Project manager engages with Client to understand and document the requirements and expectations for the project.

- **Requirement elicitations:**
  This is the process of gathering information from Stakeholders, which can include interviews, workshops, Surveys and other techniques to understand what System needs.
- **Requirement Analysis:**
  Once requirements are gathered, they are analysed to ensure they are clear, complete. This phase often involves refining requirements, identifying and potential and requirements based on feasibility.
- **Requirement Documentation:**
  The finalized requirements are documented in a requirements specification

During the analysis stage, the development team conduct a thorough analysis of the requirement gathered in the planning stage. They break why down the requirements into smaller tasks, identify risks, and define technical specifications. This stage ensures that the solution aligns with business needs and is feasible to implement. The project manager meets with the client to gather requirements and understand the scope of the project aligns with the initial phase of software Development Life cycle (SDLC), known as Communication. The project team causes on comprehensive requirements from client through meeting, workshop. The goal is to define the project's scope, objectives, and key features such as project's functionalities user authentication and payment processing. the project manager ensures clarity by documenting these requirements in details, which serves the foundation for subsequent phases. Once requirements are gathered, the project would typically move to next phase. It establishes a clear communication with the client to gather requirements and understand project goals. The primary aim is to ensure all client requirement are captured accurately and both parties have a clear understanding. In the communication stage, the project team defines the scope, objectives and requirements for the project. The team also establishes a timeline, allocates resources, and defines the

project's budget, conduct meetings to elicit and document requirements. Define project objectives Job timeline, and resource needs. Identify initial project risks. The phase concludes with comprehensive documentation of requirements and agreements on project scope, time live and resource allocations, providing a solid foundation for the subsequent phases of the project. In this phase the gathered requirements are analysed, documented and reviewed to ensure that they are complete, consistent and feasible. this phase involves Creating detailed specifications defines system functionalities and identifying any potential Risks constraints that may that may impact the project. It defines how the platform will be structured, including components like servers' databases and external integrations. It creates a prototype of mock-ups to visualize the platform's user interface. It specifies the technical details such as a data models, algorithms and APIs that will be used to implement the platform. In this phase, the manager meets with the client to gather detailed requirements understand the scope of project, and identify the goals and objectives of one software development project. The main purpose of this phase is to ensure that all stakeholders have a clear understanding of what document needs to be developed and to document requirement accurately. It uses a techniques like interviews workshops and surveys to gather detail requirements, and it requires a Structure format.

System Design

- Architectural Design:
  In this stage, Overall System architecture is designed. This includes defining structure of system, its Components, their relationship
- Detailed Design:
  Once the architecture is defined, detailed design of individual system Components takes place. This involves designing data structure, algorithm, interfaces & technical details.
- Prototyping:
  Depending on the project and its Complexity, prototyping may occur during this phase to validate design decision & gather feedback from stakeholders.
- Design Documentation:
  Similar to the requirements documentation phase design decisions and Specifications are documented in detail. It server blueprint for developer during the Implementation phase.

During this phase, the design team transforms business requirements into a detailed system architecture. The goal is to create a system that is feasible, robust, and brings value to the organization. The design phase also involves creating documents that aid in the installation and operation of the new system, such as the architecture document, critical priority analysis, implementation, plan maintenance, manual, and user manual. Depending on the project, the design phase products may also include diagrams, flowcharts, sketches, and HTML screen designs. SRS, (software requirement specification), is a reference for software designers to come up with the best architecture for the software. Hence, with the requirements defined in SRS, multiple designs for the product architecture are present in the design document specification, (DDS). The project manager and possibly other team members engage with the client to elicit and document requirements. This would involve interview, workshops, and discussions to ensure a comprehensive induction. Next, Phase

requirement definition. Requirements are defined and specified in detail. These include prioritization, ensuring they are testable and measurable, and resolving any ambiguities. The requirements are reviewed with stakeholders to ensure they accurately meet their needs and accept actions. Verifications allow verification to ensure that the requirements align with the project goals and scope. Once the requirements are defined and validated, a baseline agreement is typically reached between as they go to deliver the derived results based on feedback from stakeholders. The manager and their own insights team do the same thing giving each other feedback during development as well as at the end of each sprint in what is called a retrospective. The sprint continues until the product is delivered. When this happens, the team starts over again on a new task in the development consistency. Some of work can be automated such as code management and automated testing continues into action ensures all the ingredients are prepared then accumulated properly to form the item continuous delivery.

Agile model Working

Shwetha V

Thanusree

Agile model is an iterative model that means it does it work repeatedly. In an agile model company should give the output that statistics Customer's requirements and take feedback and work for it in a short time. Agile model it will team works they will have good Communication among them and they take decision by taking feedback from everyone or every team member. They do a requirement change in any change. In Agile model team communicates every now and then make changes if required. In Agile model they focus on Customer Satisfaction and they try to give the software or user requirements within a short span of time. The Company will work in collaboration like design team requirement teams all work together.

In Agile model work will be divided into small work among themselves.

In software development business units ask for the products they want developed providing their user stories and requirements. The agile team has to take all those user stories and figure out how to make all the products so that they all get delivered at the right time to the various business Teams. It is called the Kanban process where the Japanese word Kanban means user stories or tickets used in project planning. The tickets will be organized in order. These are called sprints. Scrum process-team developers commit for the code they will deliver for the current sprints. Development team creates the product and makes changes based on stakeholders and product manager's feedback. Work can be automated. There is continuous integration. Downtime is a specific time frame allocated to Depay or update changes for a software product in a real-time environment. This happens in software product which is developed using waterfall model. Cisco is using agile methodology for subscription billing platform (SBP).

Downtime is a small part of the waterfall model. It is traditional way of developing software using the software development life cycle where the whole product is considered as one single unit the phases are dependent on each other, changes contain high risk. Agile methodology was introduced to overcome the problems faced in the waterfall model. The entire product process into microservices as phases which are faster to execute and deploy changes on the go. Agile based products are developed using agile life cycle - Plan, Design, launch, Develop, test, Review, Deploy.

Agile methodology was introduced to overcome the problems faced in the waterfall model, where Agile based products are developed by bratling the entire product process into micro services. which is further to execute and & display changes on the go Then there is no need to worry about other or pressures take while working on one particular phase, avoiding product failure. Agile based products don't require any particular timeframe to deploy changes. Agile based products are developed using the Agile lifecycle. At first the developed implemented in product is actual working environment for receive from client and stakeholders to check its deliverables and functionality.

After client Reviews official product is launched in a real world environment where Agile methodology focus on satisfying the consumer needs by efficiently utilizing the resources and avoiding additional risk or deviation in the product. Providing trial beta version of the software for the end user software towards its deliverables and results crucial be helpful in refining and reviewing like adobe photoshop etc. In cisco, one of the popular leaders in globally is curing agile methodology for their subscription billing platform, as it was originally developed using waterfall model After adopting the agile methodology, cisco product improved its accurate efficiency where defects were reduced by 40% compared to the previous release call and defect removal efficiency increased by 14%. In software development, business unit ask for the product they want developed, providing their user stories and requirement in another words their order tickets The Agile team has to take all those tickets and figure out how to make all the products so that they all get delivered at the right time to various business team that's called the Kanban process, based on Japanese cord Kanban for the tickets used in project planning. The scrum master, in consultation with the project owner and development team, organizes the ticket into groups that can be the tackled at once. The groups are then placed in order so the first things that need so get done is placed first and so on.

In agile development, there groups are called sprints, typically a two-week period where the various developers work time on their tickets so they're done in next sprint. Developers commit for the code they deliver for the current sprint, the process occurred in what is called the scum process. Developers. implement what they committed to, making adjustment as they go to deliver the desired result based on feedback from stakeholders, product manager & their own insights development team giving feedback each other during the development work of earn sprint is what is called as retrospective. In agile development some of the work can be automated such as code management & automated testing continuous delivery set you deliver code author each sprint, not wait until the end of each sprint. That gets the software to the business factor through in smaller chunks. Agile Model was primarily designed to help a project adapt quickly to change requests.

Example: - Agile development is likes running a restaurant. Everyone has to work together to deliver the customer and get good feedback feedback. The customer first places Hodel in the form of order tickets. The agile team. has to take the ticket and prepare the ordered product to deliver it on time to various customer of business unit. It has to follow Kanban process where the tickets are arranged in order and the chef prepare dish. On agile development they are called sprint who takes and prepare the order. The process of taking and delivering the order of the business is the agile development. Agile Methodology is a project management framework that breaks project down into severe dynamic phases commonly known as sprints.

These phases are dependent as each other. How a agile different from waterfall model:

- It is dependent on microservices.
- Agile consumes more time
- Agile methodology aims to be the consumer requirement.

In this development each Individual have their own specific tasks and they have to work together and then the talk will be combined or integrated together. we can take this reactant management where everyone has their specific tasks to complete but they have to work together at night time to serve the customers. They have to be able to prepare all kind of meals based on customer requirement what they want to eat on Each day, the term will be successfully, only if the customer is satisfied with the meals which has been served in restaurant management. The process will start by taking one order. The server will collect the order in form of tickets that will be sent to kitchen and then cooks the rough.

In the same way, the developer, after gathering the information from the client, starts development of software and after the completion of software product, they will expect the client to give the review so that the developer will be able to do modifications or updating based on their client requirement. Downtime in a software engineering can be defined as the time frame allocated to deploy as update chosen for a software product. In a real-time environment, downtime will happen because most of software are using waterfall model. Once after adapting the agile methodology, it has overall efficiency and defects were reduced compared to previous release downtime. Downtime is a small part in waterfall model and it tests the whole model as single unit in waterfall model development of one stage will start after completing the other stage. Adding the new feature and making update to existing feature is called as downtime. The waterfall model is earliest and traditional model for software development.

Agile products are developed using agile lifecycle. The developed cycle product is executed in real-time environment and review are given by client for modification or updating of product it mainly focuses on satisfying the client needs by efficiently utilises the records and avoiding additional results or deviation in the product. The Agile model refers to the Iterative approach to delivering a software product. Agile methodology is a project management approach that involves breaking the project into phases and emphasizes continuous collaboration and improvement. Teams follow a cycle of planning, executing, and evaluating.

Agile methodology was introduced to overcome the drawbacks faced in Waterfall model, where Agile-based products are developed by breaking the entire product process into micro-services or phases which is faster to execute and deploy changes on the go. There is no need to worry about other or previous tasks while working on one particular phase, avoiding product failures. Agile-based products don't require any particular timeframe downtime to deploy changes. Unlike the Waterfall model, where the whole product is treated as one single unit and each process is dependent on the preceding process, where deploying chances leads to downtime. Agile-based products are developed using the Agile lifecycle. The Agile software development lifecycle is the structured series of stages that a product goes through as it moves from beginning to end. The Agile lifecycle will vary slightly depending on the project management methodology. It contains six phases, concept, inception, iteration, release, maintenance, and retirement.Agile software development is a lot like a serving a restaurant. Everyone has their own specific task but here to work together so all the pieces come together at their risk time and the have to be able to make all types of meals. that customers want each day. The term is only successful when the product is satisfactorily delivered to the customer.

In software development, business units ask for the products they want to develop, providing their users and requirement/ their own tickets. The agile term has to take all those tickets and find out how to make it and get it delivered at the risk time, the business term. This is used in Kanban process, based on the Japanese work Kanban for the tickets used in project planning. The scrum master is constrained with the product owner in development term like organizers the tickets is groups that can be tasked at once. The groups are placed in order as that the first things that need to get done first.

In Agile development, these groups are used in sprints, where the various developers work on their tickets so they are done in their further sprint in a restaurant. The executive chef shows the cooking term, everything that needs to be done for the set of orders. The cooks commit to what dish they will work on.

In Agile development, the same process occurs in what is used in the server process. In these terms, developers likewise commit for the code they will deliver for the amount sprint developers implement what they commit to making adjustments as they go to deliver the desired result based on feedback from stakeholders. The product message, etc. By the feedback, the term makes improvement in their future work. The sprint continues until the product is delivered.

When that happens, the term starts over again or a new product that gets the software to the business faster through smaller chunks. It's like a sun restaurant that lets you order different bites of food throughout the meal, delivering one nearby rather than deliver the whole meal all at once. Downtime is a specific time frame allocated to display or update changes for the software product in a rest time environment, and this happens most of the software we use today are developed using the waterfall model. For example, Cisco, one of the popular leaders in IT and networking globally, is using Agile methodology for their subscription billing platform as it was originally developed using the waterfall model after adopting the Agile methodology. Cisco's product improved its overall efficiency where defects were reduced by 40% compared to the previous selectors and defect removal efficiency increased by 14%.

The Waterfall model used for software development, where the output for software of one page as the input for the preceding page, consisting of series of steps. This method is simple and easy to understand where the preservatives are pre-known document ensures and technologies remain static where there is no need for ambiguous requirements due to the delivery changes in the Waterfall model will have high risk. To overcome the Waterfall model, Agile methodology was introduced. In Agile model, the products are developed by dividing the entire product into shares, small which will be executed fast. The agile model refers to the iterative approach to delivering a software product. Agile methodology is a project management approach that involves breaking the project into phases and emphasizes continuous collaboration and improvement. Teams follow a cycle of planning, executing and evaluating. Agile methodology was introduced to overcome the drawbacks. faced in waterfall model, where agile based products. developed by breaking the entire product process into. Microservices or phases which is faster to execute and deploy changes on the go.

There is no need to worry about other or previous tasks while working on one particular phase avoiding product failure. Agile based products don't require any particular time

frame downtime to deploy changes. Unlike the waterfall mode where the whole product is treated as one single unit and each process is dependent on the preceding processes where deploying changes leads to downtime. Agile based products life cycle is developed using the Agile. The Agile Software development life cycle is the structured series of stages that a product goes through as it moves from beginning to end." The Agile life cycle will vary slightly depending on the Project Management methodology. It contained six phases. Concept, Inception, Iteration, Release, Maintenance and retirement. In agile development each. their own specific tasks individual has and they have to work together and then the task will be combined of integrated together.

For example, if we take restaurant management. In restaurant management system everyone has their own specific tasks to complete but still they have to work together so all the pieces come together to right serve the customers. They have that time to be able to prepare all on the 8 kinds meals automats requirement what the they want to eat on each day. The team will be successful only if customer is satisfied with the meals. Which has been served.

In restaurant Start by taking the orders of the customer. After taking the order and the management, the process will Bearer will collects the order in the form of tickets that will be sent to Kitchen and then work through. In the software development, the business man will ask for the products they wanted to develop by gathering the Stories and requirements, for exp the agile team will take all the tickets and decide how to make all the products delivered at night. Kanban process so that they will be that is called as in restaurant, tickets will be arranged So that he preparing the ingredients knows what to prepare in time and the meals comer together at the right time. In restaurant the cooking them will be learning how to prepare the meal by the Scrum proc. chef. this process is called. After delivering the product to the cook will expect to get feedback about the meds which has been served In the same way the developers after gathering the information from the client starts development of the software and after the completion of the Software that product they will expect quieted reviews so the developer will be able to do modifications of updates based on requirements. Downtime defined in a software engineering can be as the timeframe allocated to deploy of update changer for a software product in a real time because environment. Downtime will happen are wring most of the software waterfall model. After adopting that it has overall agile methodology efficiency and defects were reduced compared to previous release. Downtime is a small part in waterfall model and it treats the whole product has single unit. In waterfall model development of one step will start after completing the other Stage adding the new features and making updates to the exiting feature is called as downtime the waterfall model in the list and the traditional model for software development this model is easy to understand and it is the entire well documented the final region is available only after completing. software lifecycle process changes in the waterfall model will have high risk.

Example:

A tap and pay will First account is funded with then transaction machine where the machine validates whether the well sufficient funds and will be made To overcome the waterfall model Agile was introduced methodology the product are developed model by dividing the entire product into there’s which will " need to execute fact In agile model there is no way, premium Stage to about the complete before going to another stage but in

waterfall model be one stope will Connected with another stage. Waterfall model will not will not proceed to another stage before previous stage. Agile products executed in time completion of developed product using Agile lifecycle. The developed agile product is antienvironment and are given as per modification of updating. Agile methodology focuses on chest needs by efficiently and awaiting product Satisfying the utilizing additional the resources. The product rents of demotions in waterfall model Sequential Phase. no overlap Spiral. Combination of 100 iteration and sequential development suitable for large, complex projects with fixed requirements are iterative and incremental approach responds quickly to change in requirements throughout the development. Iterative sequential approach where project is divided into smaller cycles. Allows for feedback. and improvement after each cycle. suitable for projects suitable for project of user requirements emphasizes collaboration Automatic feedback and flexibility to change. where initial requirements are but can profits Stable from product refinements and improvements. each iteration produces a working version of the software. four on functionality Jour on delivering of the system. A working System.

Agile methodology is a project management approach. that prioritizes cross-functional collaboration and Continuous Improvement, it divides projects into smaller phases and guides. team through Cycles of planning execution, and evaluation. A Doing agile development is a lot and running restaurant. everyone has their own tasks, but they have to work together to all the pieces come together at the right time and they have to be able to make all of meals based on want. The whatever customer, happen TO each day team is successful when the product Dis satisfactorily Stouts with the food order. Order On tickets that the delivered to the customer. The process in a restaurant The Server Collects the Kitchen then work through Same in the Software development, business sales units. ask for the products they want developed providing user requirements. So that the right time. that is called as word Kanban they can deliver at Kanban process based on the Japanese in agile development, these groups are called. Sprints, typically a two-week period where the various developer work developers' implement what adjustments as they go to on their tickets so they're done in time, they committed to, making deliver the desired result based on feedback from stakeholders. agile development teams give feedback during the development work, as well as at feedback during the end of each the development work. Sprint in what is called a retrospective. Other the Sprints continues until the product is delivered happens, the and when that team Starts over again on a new product. In agile Some of the work can be automated such as code management Through focusing on customer concerns and stakeholder team is well positioned to product feedback, the result agile that satisfy the right people.

downtime is a specific time home allocated to deploy e for a software's we use today or update change are developed using waterfall model. for egg: - cisco One of the popular leaders in IT and networking globally, is using agile methodology for their Subscription billing platform, SBP, as it was originally developed using the waterfall model. After adopting the agile technology method of Cisco's product improved its overall efficiency where defects were reduced. by 40% compared to the preview releases and defect removal efficiency increased by 147- Upon exploring downtime downtime is a small part of the waterfall model. Unit one single where the whole product is treated as one Adding new updating that existing feature in a waterfall model-based product needs Specific timeshare known downtime to a disturbance in the workflow of an organization, when applying

changes in a waterfall-based product. produce irrelevant results or product failure. method & simple and easy to understand, where prerequisites are pre-known, high This is documented require and technology remains static where there is no need. ambiguous requirements due to the for delivery. delay in software changes in waterfall model contain risk for e.g.: a tap and Pay machine Agile-based products don't require any particular time frame to deploy changes Unlike the waterfall model, where the whole project i single unit and each process is dependent treated as Are company implementing agile in their workspace now-a-days Companies are moving toward agile methodology to it flexibility and advantages over traditional Systems. It is primarily designed to help a project adapt quickly to change. The Agile Model refers to a go took on multiple place to development process, they have come together at the right time. E.g.: If a restaurant is being taken as example everyone, I have to work together so bit they will have everything comes together so that. have to make meals that customers want and customers will only wart what they need. They will Now the kitchen works through an software development business units air for product they want to develop by their order tickets providing users stores and requirements the agile team has to take all those ticket and figure how to make all products so that are at right time they to various business team called.

The key differences between Waterfall, Spiral, V-Model, Incremental, Agile, Prototyping, Big Bang, and Iterative models in terms of their development approach, timeline adaptability, risk mitigation, and the level of customer collaboration

Kavya N

Krishna Shree

Waterfall model	**Spiral model**
Waterfall model is simple and easy	Spiral model is more Complex.
Waterfall model works in sequential method	Spiral model works in evolutionary method
Waterfall model is applicable for Small project	Spiral model works in large project.
waterfall model is less expensive	Spiral model is very expensive
The waterfall model works in sequential method	while the spiral works in the evaluation method
The waterfall model is adopted by customer	While the spiral model is adopted by developers
It is applicable for small projects	It is used for large projects
It is comparatively inexpensive	It is very expensive
It is based on linear to frame work type	It is based on linear and iterative to framework type
It works in a sequential method	It works in evolutionary method
It is adopted by customers	It is adapted by developers
Requirements and early-stage planning is necessary	Risk is identified and rectified earlier
Suitable for small system	suitable for large system
working model of the product can only be generated in the end	working model of the product produces in each iteration
In Waterfall Model, requirements and early-stage planning is necessary.	When in Spiral Model, requirements and early-stage planning is necessary, if required.
It is adapted by customers.	It is adapted by developers.
It works in a sequential method.	It works as the evolutionary method.
It is based on linear framework type.	It is based on linear and iterative framework type.

Incremental model	V-model
Sequential approach where each phase must be completed before moving on to next.	Iterative model combining elements of both waterfall and prototyping models.
Recurrence planning and documentation are required upfront.	Risk-driven approach where risks are identified and resolved in each iterative.
Changes are difficult to implement once the project moves to next phase.	Allows for early user feedback and changes based on feedback.
Suitable for projects with well-defined requirements and low risk.	Well-suited for large, complex projects with high risk.
Limited flexibility for changes during the development process.	Emphasizes on prototyping and incremental development to mitigate risk.
In the waterfall model, errors or risks are identified and rectified after the completion of stages.	In the spiral model, errors or risks are identified and rectified earlier.
Waterfall model is applicable for small projects.	Spiral model is used for large projects.
Customer involvement is minimum in waterfall model.	In the spiral model, customer involvement is high.
Flexibility to change in waterfall model is difficult.	Flexibility to change in spiral model is not difficult.
Error and risks are identified and rectified after the completion of stages.	Error and risks are identified and rectified earlier.
Requirements and early-stage planning is necessary.	Requirements and early-stage planning is necessary if required.
Works in a sequential method.	Works in the evolution any method.
Error and risks are identified and rectified after the completion of stages.	Error and risks are identified and rectified earlier.
Requirements and early-stage planning is necessary.	Requirements and early-stage planning is necessary if required.
Flexibility to change is difficult.	Flexibility to not change is not difficult.

Cost of incremental model is low	Cost of V-model is expensive
It is iterative	It is not iterative
It is Complex to use.	It is simple to use
incremental model is flexible	V- model is less flexible
Incremental model is development model where the products is analysed, designed, implemented and tested incrementally until the product is finalized.	V Model is software development model but a software development and testing are not concurrent
Cost is low	It is expensive
It is more flexible	It is little flexible
Guarantee of success is low	Guarantee of success is high
In this user involvement is also only at the beginning	In this user involvement is also only in beginning.
It is a software development model Where the product is analysed, designed, implemented & tested incrementally until the product is finished	It is a software development fool model but development & testing are not concurrent
Testing is done in incremental model after every Iteration of phases	In V model testing activities Start with the first stage.
flexibility to change is easy	little flexibility to change is flexible
In this user involvement is only at the beginning	In this user involvement is also only in the beginning
Testing is done in incremental model after every iteration of phase.	In V-Model, testing activities start with first stage.
Incremental Model is a software development model where the product is analysed, designed, implemented, and testing incrementally until the product is finished.	V-Model is software development model, but development and testing are concurrent.
It is easy to change and flexible.	It is feasible little.
Divides the project into small modules that are developed and delivered.	Corresponds each stage of development lifecycle with a testing phase.
Each increment adds new functionality to previous increment.	Emphasizes on verification and validation activities throughout the development.
Increment model is iterative.	V-Model is not iterative.
Flexibility to change in incremental model is easy.	Flexibility of V-Model is little flexible.
Testing is done in incremental model after every iteration of phase.	In V-Model, testing activities start with the first stage.

Cost of incremental model is low.	Cost of V-Model is expensive.
Guarantee of success through incremental model is low.	Guarantee of success through V-Model is high.
Incremental model is a software development model where the product is analysed, designed, implemented and tested incrementally until the product is finished.	V-Model is a software development model but development and testing are not concurrent.
It is a software development model where the product is analysed, designed, implemented, and tested incrementally until the product is finished.	V-Model software development testing are not concurrent.
Flexibility to change in incremental model is easy.	Flexibility to change is little.
Guarantee of success through incremental model is low.	Guarantee of success through V-Model high.

Waterfall model	**Agile**
waterfall model is more linear	Agile relies on Continual change
Testing is done only in Test phase	Testing is done after
Follows Sequential design method	Follows an incremental approach.
Project are manager plays an essential role	Project manager plays entire team.
Sequential development process in predefined phase	Iterative development in short sprints
The process is documented and follows the fixed structure and requirements	Flexible and adopted methodology
The requirements and scope are definitive once agreed upon	Project development requirements and scope is expected to change over the course of iterative development process
Suitable for straight forward and predictable projects	Suitable for short projects in high-risks situation
Fixed individual responsibilities, particularly in management positions	Responsibilities and hierarchical structure can be interchangeable between team members
linear & sequential approach	Iterative & incremental approach
Rigid & predictive planning	Adaptive & flexible planning
minimal feedback & resistance to changes	frequent feedback and change acceptance
risk identified & mitigated during planning	Early risk identification and & mitigated
more applicable for smaller projects	more applicable to larger and complex projects auth various requirements
suitable for straightforward projects in predictable circumstances.	Suitable for short projects in high-risk situations
Sequential development process in predefined paces.	iterative development, short sprints.
The process is documented and follows the fixed structure.	flexible and adaptive methodology.
The SDLC paces are followed in order with no	The SDLC phases overlap and begin early in

overlap.	the SDLC.
Team members focused on their responsibility only designing their respective SDLC.	All the team members focused on end-to-end completion for the project.
Sequential approach with distinct for-requirement design	Iterative and incremental approach with focus on delivering working software in short iterations.
emphasizes on documentation and planning upfront.	Emphasizes on collaboration, adaptability, and responding to change.
Changes are difficult to implement once the project moves to next phase.	Welcomes changes, requirements, even late in development process.
Suitable for projects with well-defined and stable requirements.	Suitable for projects with evolving requirements and where customer feedback is crucial.
Limited flexibility for changes during the development process.	Provides flexibility and rapid delivery of evolved to stakeholders.
Testing is done after the completion of the development phase.	Development and testing are concurrent processes.
Follows sequential design method.	Follows incremental approach.
Breaks down the project development lifecycle into distinct phases.	Breaks down the project development lifecycle into sprints.
The project manager plays an essential role.	Projects are managed by the entire team.
Do not allow any changes in the requirement after the process starts.	More flexible in terms of changing the requirements whenever required.
Team coordination or synchronization is limited.	High degree of team coordination. Synchronization
Follow a sequential model.	Follow the incremental approach.
Measurement of success is in terms of completed and reviewed out facts.	It is flexible as there is a possibility of changing the sequential requirements even after solving the development process.
It has less customer interaction.	It has high customer intervention.
The team may consist more members.	It has a small team up and smaller in the team. The fewer people work on it with that they can move for the test.
Test plan is reviewed after complete development.	Test plan and reviewed after each sprint.

Prototyping	Big bang
It is an iterative approach.	It is adaptive
requirement · less defined	minimal initial planning
development - rapid development of prototypes	Development without formal process.
Flexibility-high accommodates easily	High reacts to customer feedback
Adaptability-supports evolving requirements	Reactive to change in the requirements
Emphasizes evolving and refining requirements through iterative prototypes	Assumes stable and well defined requirements from the start
May appear to extend initial timeline but often reduces overall development time and costs	Initial planning phases can be time confusing, changes later on can be more costly and time-intensive
Iterative and incremental, refining requirements through feedbacks loops	Linear approach where development, testing and implementation on occur sequentially
Reduce risks by validating assumptions early with iterative feedback	Carrier higher risk as issues may only surface late, impacting the entire system
Allows for changes based on ongoing user feedback	Less flexible once development starts, with changes potentially disruptive and costly

Agile	Iterative
Team can review during mid-sprint planning	Relies on baseline iteration plan.
Anyone in the team can identify, prepare and execute the test case	Tester identify, prepare and execute test cases
Executed by team members	Handled by testers
Iterative and increment	Iterative
Iterative cycles	Iterative cycles
Prototyping emphasizes iterative refinement & user feedback to gradually build a complete system	Big bang model focuses on rapid & quick deployment without Structured phases
It helps mitigate risks by identifying issues early through user involvement & feedback loops	It poses higher risks due to nature & minimal upfront planning.
It allows for changes & improvement based on feedback making it suitable for projects with evolving requirements	It is faut, may struggle with accommodating changes after initial deployment
It is called for project where requirement are unsure	It is ideal for profit where immediate functionality is crucial

It is suitable fees both large & small project	It is suitable for small project
Iterative and incremental refining requirement through feedback loops.	Linear approach where development, testing, and implementation occur sequentially.
Reduces risk by validating assumptions early with iterative feedback.	Carries higher risk as issues. May only surface late, impacting the entire system.
Allows for changes based on ongoing user feedback.	Less flexible once development starts. Until changes potentially disruptive and costly.
Involves creating a working model of the software to gather feedback and refine requirements.	Development approaches where all components are developed simultaneously and entered at the end.
Allows users to interact with a prototype to provide early feedback.	Minimal planning and documentation upfront.
Helps in identifying and resolving decision issues early in the development process.	High risk due to lack of incremental development and testing.
Suitable for projects with unclear or evolving requirements.	Suitable for small projects with well-defined requirements and limited complexity.
Can be time consuming and may lead to scope creep of not managed effectively.	Limited visibility into project process until the integration phase.
Risks are managed through iterative testing and feedback.	Risk of failure is significant due to lack of initial planning.
In Prototyping Model, cost is high.	In Big Bang Model, cost is relatively low.
Prototyping Model are well documented.	Here it is poor or minimal as focus on rapid development.
Initial requirements are collected.	Here minimum or no formal requirements gathering.
The Prototyping Model involves creating a preliminary version of the software a prototype.	The Big Bang Model is a software development where development starts with a broad outline.
High user involvement throughout the development process.	Low to minimal user involvement until the product is delivered.
involves an initial version of a software system to gather feedback and validate design decisions.	involves standing the development process with all requirements gathered up front and proceeding with full-scale development without intermediate stages.
The primary goal of prototyping is to understand user requirements better, explore design all examples, and clarify specifications alone before the full development efforts begin.	This approach occurs that all requirements are involved and well-embedded from the beginning, minimizing the need for structural development or prototyping.

Prototypes are developed quickly with format functionality to demonstrate that any function	Begin with a comprehensive plan and the entire system is dealt with in one go rather than in minutes.
can be efficient for all well-defined projects.	Helps in identifying and resolving potentials securely. Improves communication.
Prototyping is iterative.	Big Bang is sequential.