



Experiment No. 1

Name: Shashwat Shah

SAP ID: 60004220126

Batch: C22

Aim: To identify a suitable life cycle model for your case study and justify your choice

Abstract

This Software Requirement Specification (SRS) outlines the specifications for the development of a comprehensive cloud-based file storage and sharing system, akin to Dropbox. The system aims to provide users with a seamless and secure platform for uploading, organizing, and sharing files across various devices and platforms. Key features include user registration and authentication, efficient file upload/download capabilities, collaborative sharing, version control, and robust security measures.

Product:

The SRS also addresses non-functional requirements such as performance, reliability, usability, and cross-platform compatibility, ensuring a reliable and user-friendly experience. The system design adheres to constraints related to compatibility and regulatory compliance. This document serves as a foundational guide for the development team, offering a clear roadmap for the successful implementation of the cloud file storage system.

With a focus on user-centric design and seamless cross-platform accessibility, the system encompasses pivotal features such as secure user registration, streamlined file upload/download capabilities, efficient file organization, collaborative sharing with customizable permissions, version control for document tracking, and robust security measures including encryption and two-factor authentication.

Features:

- **User Registration and Authentication:**
 - User account creation with a valid email address and password.
 - Secure authentication mechanisms.
 - Password recovery functionality.
- **File Upload and Download:**
 - Ability to upload files of various types and sizes.
 - Support for simultaneous upload and download of multiple files.
 - On-demand file download.



- **File Organization:**
Creation of folders to organize files.
Search functionality for quick file retrieval.
Option to rename and delete files/folders.
- **Sharing and Collaboration:**
File and folder sharing with other registered users.
Definition of permissions for shared items (view-only, edit, etc.).
Real-time collaboration on shared documents.

The scope:

- The proposed cloud-based file storage and sharing system aims to offer a comprehensive and user-friendly platform for individuals and organizations. Users will be able to seamlessly upload, organize, and manage files of various formats and sizes, utilizing features such as file renaming, deletion, and a search functionality for efficient file management. The system emphasizes collaborative sharing, enabling users to share files and folders with customizable permissions to facilitate teamwork. Cross-platform accessibility is a key aspect, with support for web browsers, desktop applications (Windows, macOS, Linux), and mobile applications (Android, iOS), ensuring a consistent user experience across devices. The implementation will include version control for automatic tracking of file versions and the ability to revert to previous states. Security measures such as encryption during transmission and storage, access control mechanisms, and two-factor authentication will prioritize user data privacy. Performance optimization, scalability for accommodating a growing user base, and reliable uptime of 99.9% will be central to the system's success. The user interface will be designed for intuitiveness, complemented by help documentation and tutorials. Integration with third-party services and compatibility with major antivirus software will enhance functionality, while advanced features like file preview, notifications, and an activity log contribute to an enriched user experience. This scope outlines a robust and feature-rich system catering to diverse user needs in file storage and collaboration.

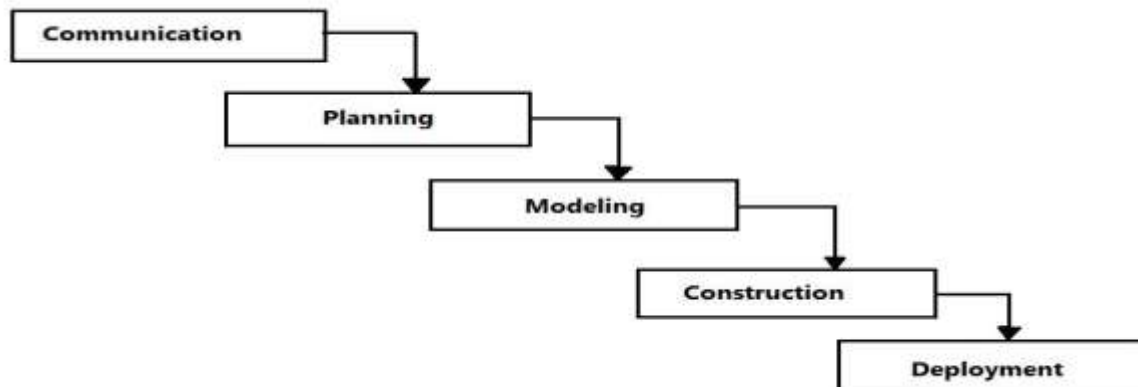
The scope of the proposed cloud-based file storage and sharing system is to provide a versatile and user-friendly platform for individuals and organizations to securely store, manage, and collaborate on digital files

Every Life Cycle:

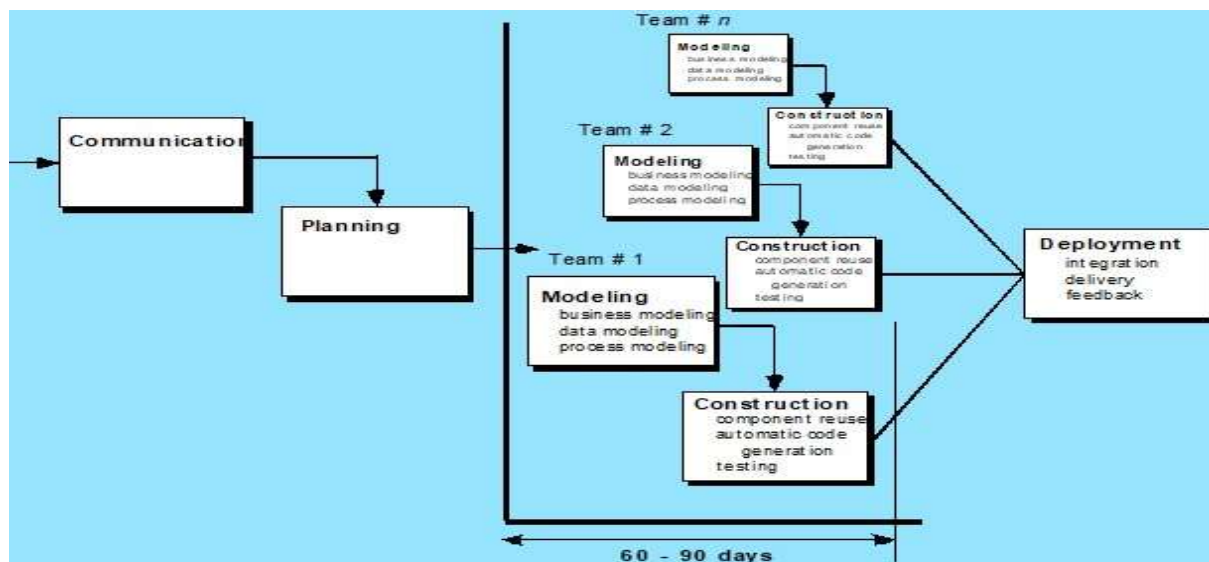
- 1) **Waterfall Model:** The waterfall model is a linear sequential approach to software development, where each stage of the development cycle follows the previous one. While this model was widely used in the past for its linear and sequential approach, it may not be the best fit for web3 based e-



commerce due to its Lack of flexibility. In projects, requirements are often subject to change due to the fast-evolving nature of the technology. Higher risk of failure, Delayed testing, and limited customer involvement In a project, it is essential to have continuous customer feedback and involvement to ensure that the final product meets their needs and expectations.

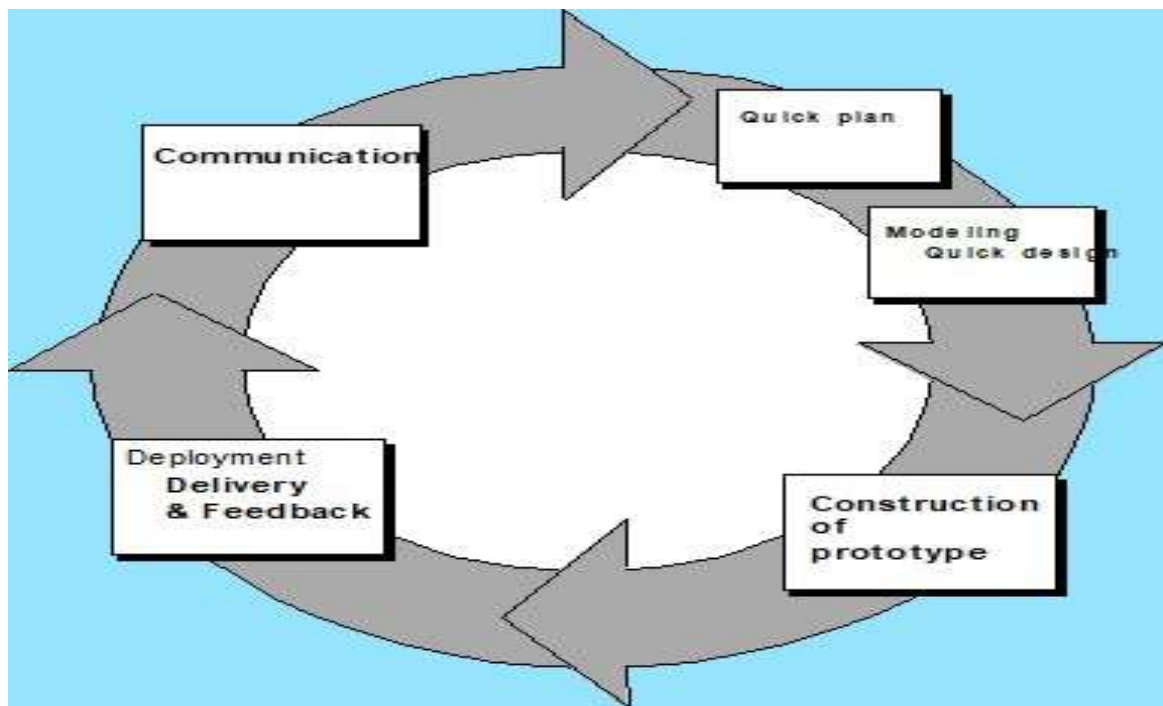


- 2) **RAD Model:** The Rapid Application Development (RAD) model is a software development approach where the emphasis is on speed and flexibility. The Rapid Application Development (RAD) model prioritises fast prototyping and iterative development, but its disadvantages are Lack of documentation. In projects, documentation is essential for security and audit purposes, and the lack of it may create vulnerabilities and compliance issues. Limited testing, Limited scalability, Limited collaboration, . In web3 e-commerce projects, it is essential to have a collaborative and cross-functional team to ensure that the product meets the needs of both customers and the business.

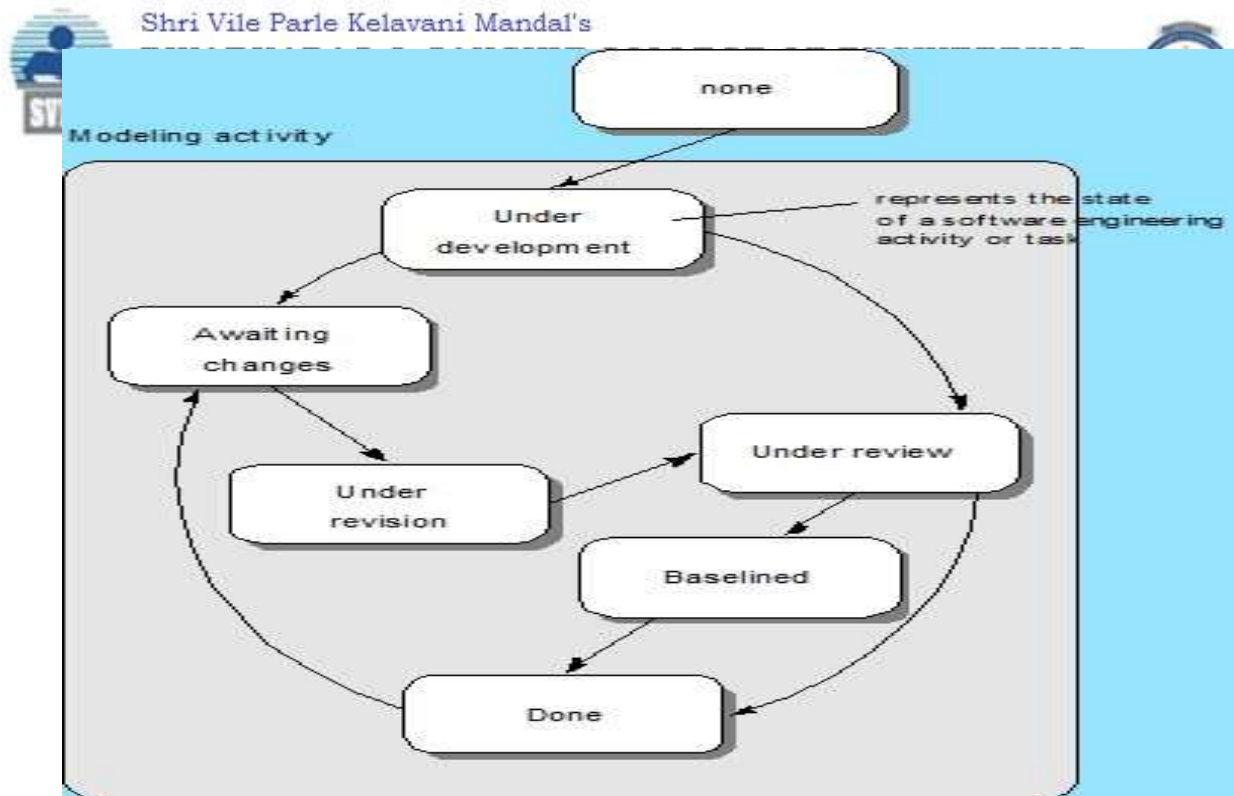




Prototyping Model: The prototyping model, which involves creating a working model of a system to test and refine its functionality before building the final product, has several disadvantages in the context of Dropbox. Time and Cost, Creating prototypes can be time-consuming and expensive. Limited Scope, Risk of Misinterpretation, Technical Complexity can be highly complex, with many different components and technologies involved. Creating accurate prototypes that reflect this complexity can be challenging, and errors or oversights in the prototyping stage can have significant consequences later on.



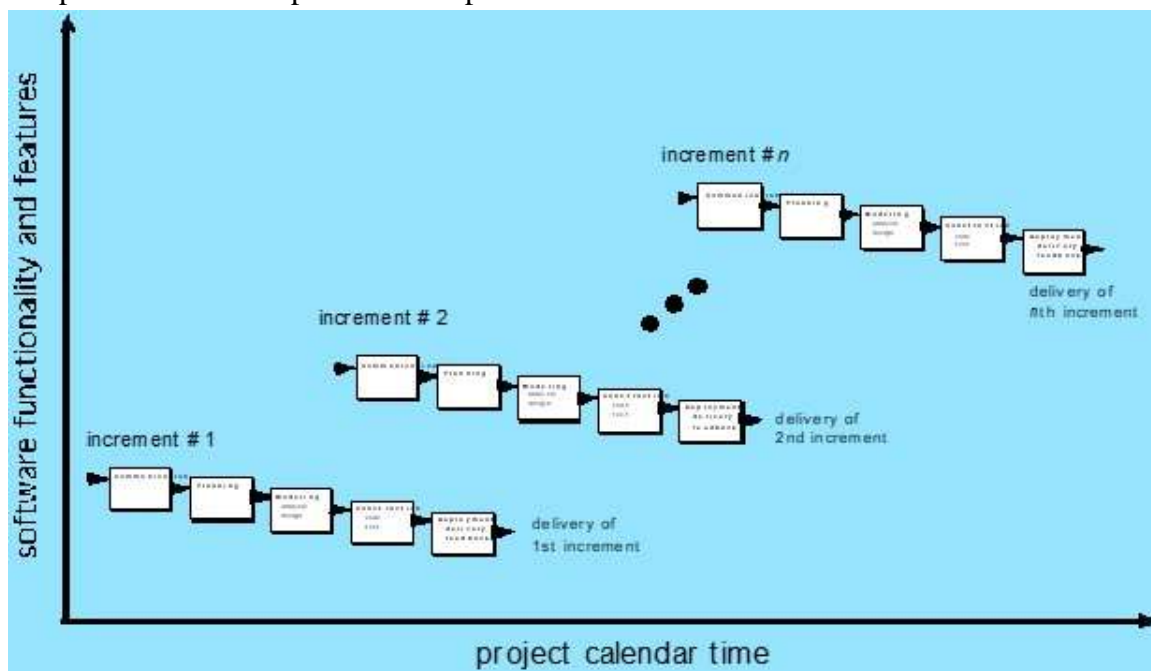
- 3) **Concurrent Model:** The concurrent model is a software development approach where multiple stages of the development cycle are executed simultaneously. Disadvantages include performance bottlenecks, difficult to debug, data inconsistency, longer development time



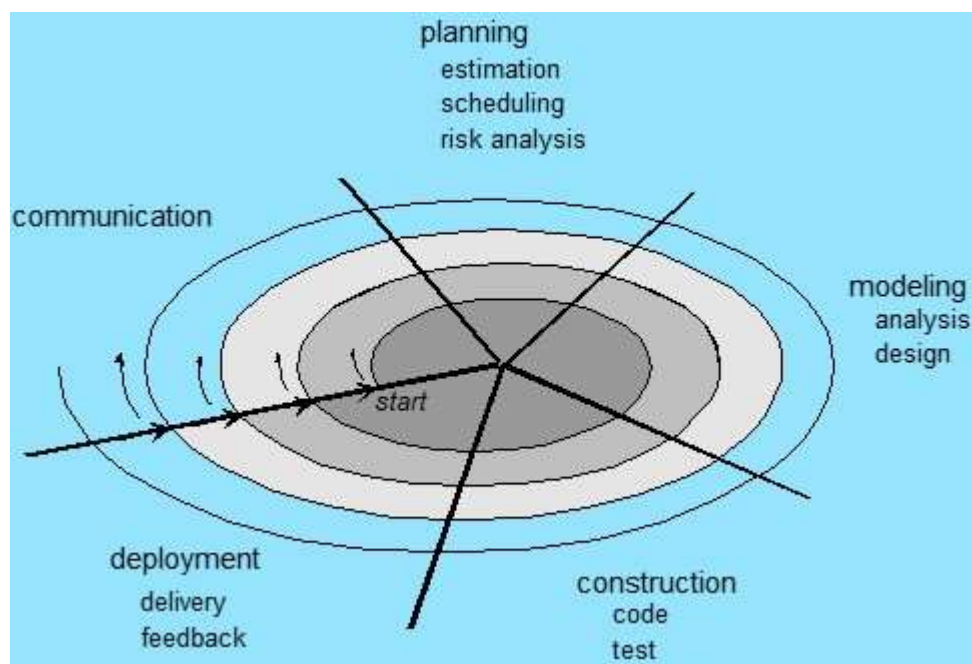
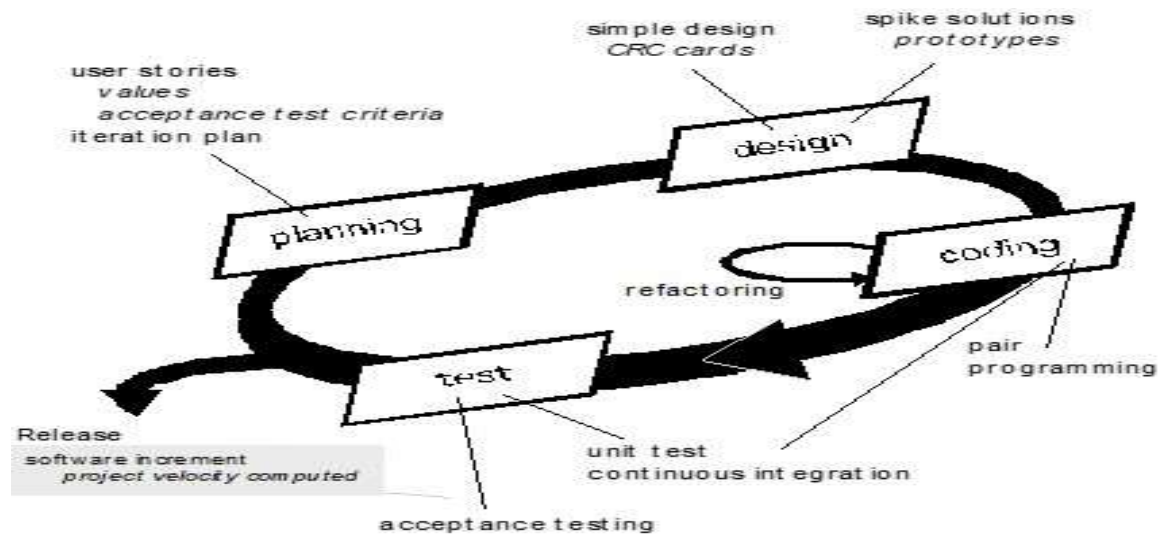
- 4) **Agile-XP**: Agile XP may not scale well to larger teams or more complex projects. Agile XP requires a high level of collaboration and engagement from all team members, which can be a disadvantage when working with a team that is not experienced with or committed to Agile development practices. Limited documentation: software over comprehensive documentation, which can be a disadvantage for that require detailed business rules and security requirements.
- 5) **Agile-FDD**: FDD requires a high level of collaboration and engagement from all team members, which can be a disadvantage when working with a team that is not experienced with or committed to Agile development practices. FDD prioritises flexibility and adaptability over predictability, which can be a disadvantage for websites that require strict adherence to project timelines and budgets.
- 6) **Agile-Crystal**: Crystal requires a high level of collaboration and engagement from all team members, which can be a disadvantage when working with a team that is not experienced with or committed to Agile development practices. Crystal focuses on a limited scope of work in each iteration, which can be a disadvantage when developing a website that involves multiple complex components and features.
- 7) **Agile-Adaptive software development**: ASD may not scale well to larger teams or more complex projects. This can be a disadvantage for websites, which may involve many complex and interconnected components and features. ASD requires a high level of collaboration and engagement from all team members, which can be a disadvantage when working with a team that is not experienced with or committed to Agile development practices.



- **Incremental Model**: The Incremental Model is a software development model in which the development process is divided into small, manageable phases or increments. Each increment builds on the previous increment, and the system is developed and tested incrementally. This model is suitable for platforms because it allows for iterative development, which means that the platform can be continuously improved and adapted to meet changing user needs and business requirements. Using an incremental model allows for the rapid delivery of new features, which can help keep the website competitive and up-to-date.



- **Agile Model-Scrum**: Scrum emphasises collaboration and communication among team members, which is particularly important in developing a website. This type of website requires expertise in multiple areas, including blockchain development, smart contract programming, and e-commerce functionality. Scrum is suitable for platforms because it allows for frequent releases and rapid feedback, which is critical for keeping up with the fast-changing market and customer demands.
- **Spiral Model**: The Spiral Model is a software development model that combines the iterative nature of the Incremental Model with the risk management and prototyping of the Prototyping Model.



The selection of a Software Engineering (SE) model depends on various factors, including project requirements, team expertise, and the development environment. For a cloud-based file storage and sharing system, a model that accommodates iterative development, flexibility, and client/user feedback would be beneficial. The Agile model is a suitable choice for such a project.

Agile Model:

Iterative and Incremental: Agile promotes an iterative and incremental approach, allowing for continuous development and improvement. This aligns well with the evolving nature of cloud-based systems and the need for regular updates and enhancements.



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Autonomous College Affiliated to the University of Mumbai)

NAAC Accredited with "A" Grade (CGPA : 3.18)



Flexibility: Agile allows for changes to be made even late in the development process, which is beneficial when adapting to user feedback or incorporating new features.

Client/User Involvement: Regular client or user involvement is a key principle of Agile. This is crucial for a file storage and sharing system, as user requirements may evolve, and user feedback is valuable for refining features.

Collaboration and Communication: Agile emphasizes collaboration among team members and encourages open communication. This is vital for a project where features like collaborative sharing and real-time updates are key components.

Frequent Deliverables: Agile encourages the delivery of functional software in short, incremental cycles. This aligns with the expectations of users who may prefer regular updates and new features.

Adaptability: Agile is adaptable to changes in requirements and priorities, making it suitable for projects with evolving needs, such as a file storage and sharing system

While Agile is a popular choice, it's important to consider the specific characteristics of your team, project constraints, and the preferences of stakeholders. Additionally, hybrid models or variations of Agile, such as Scrum or Kanban, can be tailored to suit the unique requirements of your development project.

Conclusion:

In this experiment we have studied various process models and are able to apply suitable process model for our application.