



**Academic Year: 2023\_24**

**Name : Dhruvin Chawda**

**Sap/div : 60004210159/C2**

**Sub: Software Engineering (SE)**

### **Experiment No. 1**

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

#### **Theory:**

There are different types of models that are suitable in different situations.

1. Waterfall model
2. Incremental Model
3. Prototyping Model
4. RAD Model
5. Spiral Model
6. Concurrent model
7. Agile Model - XP
8. Agile Model - Scrum
9. Agile Model - DSDM
10. Agile Model - ASD

#### **Performance:**

1. For case study we have took **online virtual meeting software**
2. Abstract:

In an era dominated by remote work and global connectivity, the demand for efficient virtual collaboration tools has surged exponentially. The advent of online meeting software addresses the critical need for seamless communication among geographically dispersed individuals and teams. This software serves as a transformative solution, bridging the physical gap and enabling real-time interactions through virtual meeting rooms. The product, online meeting software, is a digital platform that facilitates video conferencing,



**Academic Year: 2023\_24**

allowing users to conduct meetings, presentations, and discussions irrespective of their physical locations. Users access the software through web browsers or dedicated applications on various devices, ensuring flexibility and accessibility. The primary users of this software include professionals engaged in remote work, businesses conducting virtual meetings, educators facilitating online classes, and individuals seeking a convenient means of communication.

**Features :**

1. **Video Conferencing:** Enables face-to-face interactions through video feeds, fostering a sense of connection among participants.
2. **Audio Communication:** Supports real-time audio communication, allowing users to engage in discussions and exchange ideas.
3. **Chat Functionality:** Provides a chat feature for instant messaging during meetings, enhancing communication and facilitating quick exchanges.
4. **Screen Sharing:** Allows participants to share their screens, enabling presentations, demonstrations, and collaborative work on documents.
5. **File Sharing:** Facilitates the exchange of files within the virtual meeting environment, streamlining collaboration and information sharing.
6. **Scheduling:** Enables users to schedule and organize meetings in advance, with options for recurring sessions.
7. **Accessibility:** Accessible through web browsers or dedicated applications on various devices, ensuring flexibility for users across different platforms.
8. **User-Friendly Interface:** Designed with an intuitive and user-friendly interface, making it easy for participants to join meetings and navigate through features.



**Academic Year: 2023\_24**

3. Analyze every life cycle model and choose a model that best fits your problem :

How various life cycle model can be used for this software :

#### **Waterfall model :**

The Waterfall Model, a linear software development methodology, guides the creation of online meeting software through distinct phases. Initially, user needs are meticulously identified, and features like video conferencing and chat are defined, establishing project requirements. Subsequently, a detailed blueprint covering architecture and user interface is developed. This guides the coding phase, emphasizing core functionalities such as video conferencing, chat, document collaboration, and scheduling. Rigorous testing follows, addressing bugs and ensuring functional, performance, and usability standards. Upon successful testing, the software is deployed for users through web browsers or dedicated applications, accompanied by documentation and support. Post-deployment, ongoing maintenance includes issue resolution, regular updates, and integration of user feedback, ensuring the software's reliability and relevance.

#### **Incremental model :**

The Incremental Model provides a structured approach to developing online meeting software, delivering a functional product in successive increments while continuously refining it based on user feedback. The process begins with initial planning, identifying core functionalities like video conferencing and basic chat. The first increment involves developing and releasing the initial version with these core features. User feedback is then gathered to evaluate preferences and areas for improvement. Subsequent increments iteratively build upon the previous ones, incorporating user input and expanding capabilities. Emphasis is placed on scalability as the user base grows, integrating advanced features such as document collaboration and screen sharing. The final increment results in a fully-featured version, incorporating all planned functionalities and refinements based on user feedback. Ongoing maintenance ensures stability, addressing post-release issues,



**Academic Year: 2023\_24**

providing regular updates, and remaining responsive to evolving user needs. This iterative and adaptable approach ensures continuous improvement in response to the dynamic expectations of online meeting software users.

**Prototyping model :**

The Prototyping Model for online meeting software involves engaging stakeholders to identify key user requirements, emphasizing features like video conferencing and chat. A basic prototype is then created, showcasing core functionalities and providing a tangible representation. Interactive user sessions gather feedback on usability and feature preferences, facilitating iterative refinement. This continuous cycle incorporates user input into successive prototype iterations, gradually enhancing features and the user interface. Ongoing user involvement ensures alignment with user expectations. Transitioning to full-scale development is guided by the refined prototype, with concurrent testing to promptly address issues. The final software release incorporates validated features and improvements from the prototyping process, with a commitment to continuous user feedback for future updates.

**Spiral model :**

The Spiral Model, applied to the development of online virtual meeting software, involves a strategic and iterative approach. Initially, project goals and constraints are identified, and a development plan is established, emphasizing the model's risk-driven nature. Through systematic risk analysis, potential challenges are evaluated, prioritized, and considered throughout the development process. The model's iterative cycles commence with the creation of a small, functional prototype, focusing on core features like video conferencing. User feedback is then collected and used to refine the software in subsequent iterations, introducing additional features such as document collaboration and enhanced user interfaces. As the software evolves, scalability and integration aspects are emphasized to accommodate a growing user base. The final iterations progressively expand the software's capabilities, addressing user feedback and meeting evolving



**Academic Year: 2023\_24**

requirements. The model culminates in the release of the final version, incorporating all planned functionalities and refinements based on user feedback and risk assessments. Ongoing maintenance ensures the software's stability and responsiveness to user needs, completing the Spiral Model life cycle.

**Rapid Application Development (RAD) Model :**

The RAD Model, applied to online virtual meeting software, prioritizes swift user engagement and iterative development. Stakeholders swiftly define key user requirements, emphasizing features like video conferencing and chat. Rapid prototypes are created, serving as tangible representations for immediate user feedback and validation. Through interactive sessions, user input guides iterative development, enabling rapid adjustments to align with user expectations. Parallel development and testing processes accelerate the delivery of functional components. Features are incrementally added based on user feedback, including document collaboration and screen sharing. Continuous user involvement, quick deployment cycles, and iterative refinement ensure the software evolves responsively. Continuous integration practices maintain overall stability, while regular feedback loops and agile project management methodologies foster adaptability and collaboration throughout the software development lifecycle.

**Agile model - scrum :**

For this model particularly the Scrum approach, for online virtual meeting software involves creating a prioritized product backlog outlining features and functionalities. Sprint planning sessions define the scope for each iteration, selecting backlog items based on priority and complexity. Daily stand-up meetings ensure team alignment, fostering collaboration and transparency. During sprints, development work is executed in small, manageable increments, resulting in a potentially shippable product at each sprint's end. Continuous testing practices, including automated and manual testing, maintain software reliability. Sprint review meetings showcase completed features, gathering stakeholder feedback for continuous improvement. Sprint retrospective meetings reflect on team performance, allowing



**Academic Year: 2023\_24**

adjustments for future sprints. Embracing agile principles of adaptability and flexibility enables the software to evolve based on user feedback and changing needs. Regular user engagement gathers feedback on features and usability, guiding continuous improvement in subsequent sprints. Backlog refinement ensures ongoing prioritization based on changing requirements. This iterative approach, encompassing sprint cycles, allows for continuous adaptation to changing priorities and the delivery of incremental value in the dynamic and user-centric realm of virtual meeting software development.

**From waterfall, incremental, prototyping, spiral, RAD, agile(scrum) the agile model fits the best for the online virtual meeting software.**

Explanation :

Why Others May do not Fit :

Waterfall model : Waterfall is less suitable for the online virtual meeting software due to its rigid, sequential nature. In a dynamic environment where requirements evolve, the linear progression of Waterfall may lead to challenges in accommodating changes effectively.

Incremental model : While Incremental models allow for phased development, they may lack the agility needed to quickly respond to rapidly changing user needs in the online virtual meeting context. Agile provides more flexibility and faster iterations.

Prototyping model : Prototyping is valuable for gathering early user feedback, but it might not be as conducive to the rapid and continuous development required for evolving features in online virtual meeting software.

Spiral model: The Spiral Model's emphasis on risk management and in-depth planning may be perceived as more time-consuming for a software domain that requires rapid adaptation. Agile allows for a more streamlined and adaptive approach.

RAD (Rapid Application Development) : RAD shares similarities with Agile but might lack the structured ceremonies and defined roles present in Scrum. Scrum's



**Academic Year: 2023\_24**

specific framework and practices make it more tailored for collaborative development in the virtual meeting software domain.

Why agile model fits the best :

The Agile Model, specifically the Scrum approach, is well-suited for online virtual meeting software due to its adaptability and responsiveness to the dynamic nature of this domain. In the context of virtual meetings, user requirements can evolve rapidly as technology advances and user preferences change. Agile's flexible and iterative development methodology allows development teams to promptly adjust to these evolving needs. Scrum's emphasis on regular, short development cycles (sprints) aligns with the need for continuous improvement and quick adjustments in online meeting software. The iterative nature of Scrum enables the team to receive and incorporate user feedback at the end of each sprint, ensuring that the software remains aligned with user expectations. The dynamic environment of virtual meetings, with constant advancements in communication technology, demands a development approach that can quickly adapt to emerging trends. Agile's incremental delivery of features ensures that users can experience and provide feedback on new functionalities regularly, facilitating a more user-centric and responsive development process.

**Conclusion:**

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