

Project Title: Chrome Extension - All-in-One Productivity Enhancer

Team Number: 5

- Sarthak S Kumar (SRN: PES2UG21CS482)
- Sathish Kumar G (SRN: PES2UG21CS484)
- Sanath Kumar R (SRN: PES2UG21CS474)
- Sumukha N M (SRN: PES2UG22CS822)

SDLC Model:

Implementing the project using Lean Agile is a wise choice, as it emphasizes efficiency, adaptability, and delivering value to the end-users early and often. Let's align the Lean Agile approach with the project components:

Lean Agile Principles and Project Components:

1. Iterative and Incremental Delivery:

- Break down the project into small, manageable increments, delivering a potentially shippable product after each sprint (e.g., 2-4 weeks per increment). Each increment should have a set of features that add value to the end-users.

2. Feedback Loops and Continuous Improvement:

- Encourage frequent feedback from stakeholders, end-users, and team members. Use this feedback to adapt and improve the product continuously.

3. Empowering Teams and Individuals:

- Encourage teams to self-organize and make decisions. Trust their expertise and empower them to find the best ways to implement the functionalities assigned to them.

4. Value Stream Mapping:

- Map the value stream for each feature to identify bottlenecks, delays, or unnecessary steps in the development process and work on streamlining the workflow.

5. Visual Management and Information Radiators:

- Use visual tools like Kanban boards or Scrum boards to visualize the workflow, track progress, and make the project's status transparent to all stakeholders.

Aligning Lean Agile with Project Components:

1. Sprint Planning:

- Conduct sprint planning sessions at the beginning of each sprint to select and prioritize features from the backlog for that sprint.

2. Daily Stand-ups:

- Hold daily stand-up meetings to discuss progress, challenges, and plan for the day's work, ensuring everyone is aligned and aware of the project's status.

3. Sprint Reviews and Retrospectives:

- At the end of each sprint, conduct a review meeting with stakeholders to showcase completed features and gather feedback. Also, conduct a retrospective to reflect on the sprint and identify areas for improvement.

4. Minimum Viable Product (MVP):

- Identify and prioritize a Minimum Viable Product (MVP) that includes essential features to deliver value early to the users.

5. Kanban for Workflow Management:

- Use Kanban boards for managing the workflow efficiently.

Architectural Style:

1. Service-Oriented Architecture (SOA)

Description:

Service-Oriented Architecture (SOA) is an architectural style where components of the system are organized as services. These services are loosely coupled, meaning they can operate independently, yet can be integrated to achieve specific functionalities. Each service performs a specific business function and can be accessed and used independently by other components or services within the system.

Justification and Benefits:

- Modularity and Reusability:

- SOA allows breaking down the system into modular, independent services. Each feature in your project (e.g., authentication, chat, file compression) can be developed as a separate service.
- These services can be reused in future projects or scaled independently, promoting modularity and reusability.

- Scalability:

- Individual services can be scaled horizontally or vertically based on demand. For instance, if the chat service experiences heavy usage, only that specific service can be scaled without affecting other parts of the system.

- Interoperability:

- SOA enables interoperability between different services. Each service can be developed using different technologies or programming languages, as long as they communicate through standardized protocols such as HTTP/REST or SOAP.

- Flexibility and Adaptability:

- As services are loosely coupled, changes in one service do not impact other services. This flexibility allows for easier updates and modifications without disrupting the entire system.

- Centralized Management:

- SOA allows for centralized management of services. This means that each service can have its own development and deployment cycle, making it easier to manage the project as a whole.

- Enhanced Performance:

- By breaking down complex tasks into smaller services, each service can be optimized individually for performance, leading to overall enhanced system performance.

- Security and Access Control:

- Security mechanisms can be implemented at the service level, ensuring controlled access to specific functionalities. This is crucial for features like user authentication and data privacy.

Considering the diverse functionalities of your project and the need for flexibility, scalability, and maintainability, a Service-Oriented Architecture (SOA) would provide the required structure and benefits to your system. Each component of your project, such as user authentication, chat, file compression, notes, and others, can be developed as individual services, enhancing the overall efficiency and maintainability of the system.

