

### **Selected process model:**

For our generative education system using AR, the initial requirements of the project are vague and may be subjected to changes as development progresses. Hence an agile process model will be appropriate in this scenario. Considering the fact that our project will deal with continuous changes due to customers changing requirements, we may need to release the system in steps. And so, the Scrum framework will be the best choice for the project.

### **Reasons for choosing this model:**

Scrum is one of most widely used agile process models, due to its lightweight management practices, transparency among developers as well as frequent consumer feedback. Scrum uses an approach that is both iterative and incremental. There is also a benefit of regular inspection of the progress. Everyone can see every part of the project, from inside and outside the team which helps customers/stakeholders to observe if the features are working as desired.

In this model, the development process is divided into shorter intervals, called 'sprints'. For each sprint the requirements are prioritized and developed accordingly. Additionally the product Backlog list is constantly updated with new and more detailed items. An iteration continues until the customer is satisfied with the features implemented. As our project relies on customer feedback, we will be able to review our progress before release

Scrum emphasizes collaboration and frequent communication between team members and stakeholders. This is beneficial for developing learning modules, course contents, and progress tracking functionalities as it allows for continuous refinement based on feedback.

Scrum promotes regular communication through daily stand-up meetings, sprint reviews, and sprint planning sessions. This supports the development of collaborative communication features, n-way communication between teachers and students, and live interaction functionalities.

Overall, the Scrum process framework will nicely fit within the scope of our project as it provides the best approach for development with continuous progress tracking & feedback as well as deep customer engagement.

### **Why are other models insufficient?**

Since our project requirements initially are not well defined, the plan driven frameworks will fail in this regard. So the agile process is best suited in terms of ambiguous & changing requirements.

Extreme Programming (XP) focuses heavily on technical practices such as pair programming and test-driven development which could be beneficial for ensuring high-quality code and timely delivery. But XP might not provide as much structure for managing the overall development process and stakeholder collaboration as Scrum does. Scrum provides clearer roles and artifacts to focus on specific goals than XP.

DSDM emphasizes the importance of frequent delivery and active user involvement. While this aligns well with the requirements for iterative development and stakeholder engagement, DSDM's focus on fixed time and cost constraints might not be as flexible for accommodating evolving requirements as Scrum. DSDM

focuses on engineering activities and may include roles beyond the development team, meanwhile Scrum focuses on the operational team with more standardized terminologies.

FDD focuses on features and may require extensive planning beforehand. Moreover, in FDD the operational team consists of a large group which are then divided into smaller groups to work in parallel. Scrum focuses on smaller teams and is more oriented towards customer feedback to improve its results.

### **Project roles and responsibilities:**

In a development project, members of the team will be given specific roles, each with their own sets of obligations to ensure the development progress continues as planned. The number and scope of these roles will vary with the complexity & needs of the project. For our project the following roles are included:

- **Scrum Master:**

- Make sure the development team abides by Scrum principles and track their progress through daily Scrum meetings & reviews.
- Prevent team from over committing to elusive requirements during sprint planning as well as aid in estimation of task progress and sub-task creations.
- Assist product owner in managing & prioritizing product backlogs to achieve clear requirements for the project.
- Advise development team on organization of tasks & manage internal obstacles through workflow improvements.

- **Product Owner:**

- Communicate with the development team to define the product goal for the Scrum team.
- Create & manage product backlog as well as review & finalize tasks related to product backlog.
- Prioritize & verify requirements defined in the backlog.
- Represent the needs of shareholders and discuss with them to change or create additional requirements for the backlog.

- **Scrum Development Team**

- Produce increments of working software based on product backlog requirements.
- Ensure product quality through identifying the best approach for development. This can be achieved through consulting with the Scrum master, testing prototype builds and including quality assurance tasks.
- Create an estimation of the time required for a sprint as well as commit to the necessary goals to achieve during that sprint.
- Collaborate with Scrum Master and other members of the team to organize & delegate tasks to suit overall development progress.

- **Management Group**

- Manage the overall project by interacting with both the developers and shareholders to ensure smooth delivery of the product.

- Participate in review meetings to ensure backlog requirements are met as demanded from the shareholders.
- Take part in final decision makings and make sure all agreements between developers and shareholders are met accordingly. Also make sure all standards and necessary protocols are maintained throughout the project.