



Process Model Document

05.11.2023

—

Team F

Adrian

Arooz Singh

Fahimul Ghaffar

Libin Joji

Mamun

Overview

To decide on our process model we first looked at a couple of factors about the project undertaken.

I. Project Requirements

The project requires the usage of react, node, javascript and html to implement a website with a working login feature that lets users view performance reviews based on username and project once successfully logged in.

II. Project Size and Complexity

We took into consideration that the project was a relatively small and non complex one as it only required five people. Because of this our deadlines were more flexible with meetings held regularly to assess progress.

III. Familiarity with the technology


Another major factor was that we decided to use react to build our website instead of bottle. A reason for this is that it gave us an opportunity to work with a software most of us were unfamiliar with. Due to this, ample time had to be given to each designer to get familiar with the software.

Our Team Process Model

Based on the above mentioned factors we decided our process model would consist of first creating an initial implementation, reviewing it and then slowly evolving it with new features. These features would be split up into small increments that would allow us to regularly conduct meeting reviews and decide on updates where necessary. This way each member could stay updated on and also control the flow of the project while simultaneously working on their own work.

I. Processes

Our processes and activities included weekly scrum meetings, discussions online through a group chat, code reviews whenever necessary and pull requests. Our pull request policy consists of presenting the code to the other team members during a scrum meeting or notifying the other team members of a change online and a



thorough code review conducted by two or more members after which the PR is accepted.

II. Sequence

At the beginning of a sprint, the SCRUM master delegates roles and tasks to members based on their strengths and interests. We agree upon a flexible deadline set at least two days before the actual deadline after which we regularly update the other team members on our progress during scrum meetings. Once a team member has completed their task they present their code and a code review is done.

III. Decision Points

Decisions are made at the beginning of each sprint by the SCRUM master and during each scrum meeting where we discuss potential changes to the tasks or roles, a change to the deadline, additional tasks etc.

IV. Resources

Our team decided to use the React framework for our project because one of our members was already familiar with React, and it provided the rest of us with an exciting opportunity to learn something new. We used Pycharm as our IDE as advised by the professor. We also used node js as it helped us build networked applications using javascript. Node allows us to handle requests, manage databases and in serving dynamic content to clients.

V. Documentation

We prioritized thorough documentation throughout our project, ensuring that every aspect of our development process and decision making was well documented. This ensured that team members who were unable to attend team meetings were still caught up with the activities of the rest of the group.

VI. Iteration

Our process model incorporated iterative cycles, allowing for refinement and improvement based on feedback.

Team Meetings

We conducted scrum meetings during every single class, a note taker and a scrum master was decided at the beginning before moving onto discussions. During meetings regarding implementation and updates, we discussed what our project should look like, assigned tasks according to each designer's strengths and preferences and updated our product backlog which included our to-do lists, bugs, features etc. We took a look at each other's code, their progress, any trouble they might have had while working on the code and made decisions to either drop features we felt unable to complete on time or push it to the next sprint.

VII. Pull Requests


We each worked on separate local branches that would be pushed to the online branch after work was complete. Then during our regular scrum meetings we conducted code reviews and once the code was approved by all members we merged it into the master branch.

VIII. Code Reviews

Code reviews were conducted during our scrum meetings, once someone had completed implementing a feature.

Evolution from our previous process model

While we still conduct regular scrum meetings, our previous meetings before sprint 1 included less discussions about the status of our programs. Our code reviews were not done properly and a proper system for pull requests was not followed despite being agreed upon. After receiving our feedback for sprint 1 we have held stricter team meetings, assigning tasks properly, and everyone updated the team on their progress. We held proper code reviews where the designer explained the working of their code to the rest of the team, while the team pointed out bugs or problems that might arise. Merges to the master branch were only done during team meetings once every present team member had seen and approved the code. We also thoroughly went through the marking scheme and checked out each point this time. In the context of the process model, roles and responsibilities would be properly identified at the beginning of each sprint and documented. A timeline would be created based on how long each member thinks their



specific task might take. Based on feedback, changes are made to the project and we decide whether additional changes can be made before the sprint deadline.

Improvements to our current process model

Our current process model can be further improved with better communication and documentation. Establish clear documentation standards for code reviews, ensuring that each review includes documentation of changes made, reasoning behind decisions, and potential impacts. Better usage of github and kanban board to track tasks, assign responsibilities, and monitor progress. This provides a centralized view of the project's status and helps team members stay organized. At the end of each sprint we could conduct meetings to reflect on the team's performance, identify areas for improvement, and implement changes in the process model accordingly. Take better advantage of unit tests to ensure that errors don't occur when changes are being made. Encourage team members to actively participate in discussions outside their immediate areas of expertise, which can lead to a better understanding of the project for everyone and improve collaboration.