Web Based Scrum Board

Project Milestone 5: Final Report

CS 3704 Fall 2024 (83365)

Aditya Rao
Aidan Carraretto
Akshara Gandrakota
William Burriss

Abstract

When it comes to the different fields of engineering, software engineering is one of the newest fields that is prevalent today. Civil engineering, for example, has been around for thousands of years. With this new field of engineering, new methodologies must be developed to help keep software engineers organized and improve efficiency on large scale projects. As discussed in lecture, the 1995 CHAOS Report revealed that software engineering was a very expensive and inefficient process [1]. Since then many different systems have been developed to fix the issue of expensive and laborious software projects. One system that proved to be very effective in certain settings was the Agile methodology [2]. The Agile system is effective because it was crafted with the knowledge that project requirements will change over time [2]. A common implementation of the Agile methodology is the Scrum board. There are many services that provide a web based Scrum board so that developers can easily implement Agile methodologies. Our solution to the problem of inefficient software development is to create our own web based Scrum board with additional features that many alternatives fail to implement.

Introduction

Scrum boards are extremely helpful; in fact, 87% of teams that employ agile methods use stand up meetings to discuss scrum boards and the resulting project planning [3]. They help when it comes to breaking a project down into more manageable steps and play a large role in team management. However, there are many drawbacks that a traditional scrum board presents. One big one is with remote working; many remote working employees can find it hard to keep up with the milestones discussed relating to scrum boards, as they cannot be in person. With the rise of global teams and the adaptation of flexible working after the pandemic, this issue has become even more prominent. Some employees also find it harder to make time for these meetings in their busy schedules, especially as project deadlines approach. Traditional scrum boards can also make it hard to make sure everyone is staying on track. There is no concrete system to observe accountability which poses a big problem, particularly in larger sized teams. Without automated update systems, managers rely on manual updates from team members, which can cause things to be missed and gaps in communication. Some have even said that traditional scrum boards can lead to scope creep, since there is not as much structure as they would like there to be [4]. With projects becoming more complex, the need for a new scrum board that fills the holes of a traditional scrum board is greater than ever. This tool should be able to bridge gaps in communication and allow for seamless project tracking, as well as helping identify any slowdowns. By implementing this board, managers, developers, and stakeholders will be on the same page as their team, especially in risk heavy and fast-paced environments like those engineers work in.

Motivating Example

The usage of Agile software engineering processes have been on the rise, and with it comes a rise in the prevalence in Scrum boards and techniques. As such, an application of this project can be generalized and applied to many roles and scenarios in software development and a project's pipeline. For example, a newcomer who was just brought on board for a team's ongoing project is going to need to learn and understand the project's concepts and tasks quickly and efficiently. Even without prior knowledge of this newcomer joining the project, the team likely wants an easy way to organize their tasks and collaborate on a shared scrum board remotely. For a software engineering project like the Top Five class example, the team would be able to divide tasks between members, assigning a certain number to each member that they can also put on each task.

Additionally, they can create further boards that don't necessarily need to be Scrum boards - the column rename/deletion features adds a level of modularity and customization to each board to meet the needs of such a project, like categorizing tasks. As an example, the team can also create a Kanban board, creating any number of columns to insert tasks that are associated with each column.

With all of these boards set up, the newcomer would also have an easier time understanding the project with another team member showing them through the boards created by the team and what they mean for the project.

Related Work

There exist many online scrum boards, as well as online boards that can be used as a scrum board among other organization options. Examples include Jira, Miro, and Trello. Jira and Trello both handle their user experiences through easy to understand menus, while tools like Miro are more focused as a collaborative whiteboard, although they can still be used as a scrum board through certain tools, such as sticky notes. Tools like Jira and Trello allow users to create categories for tasks (such as "In Progress," "Complete," etc.) and add "cards" under these categories. Each card can have a description and people assigned to them, allowing teams to specifically state what to do for a certain task and who should work on it.

Both types of scrum board applications allow teams to remotely work together on an online scrum board to establish requirements and assign tasks. Although it is possible to create a whiteboard tool with a focus on scrum boards, our team will be focusing on the menu/page-based UX of web-based scrum boards. It is not only more feasible to create this type of application instead of a live whiteboard, but it also gives the potential for our app to streamline the user experience and provide a more basic but easier to understand interface.

Implementation

For our implementation we opted to use figma to create wireframes and a simple interactive prototype. During the first few phases we decided figma would be ideal for creating a simple wireframe so we could get an idea of how our design would look. As we progressed through the project we made our prototype more in depth and better looking. We decided to begin work on a more interactive design; one issue we encountered was that not everyone knew the same technology stack. This made creating our prototype difficult. However, one of the features of figma we discovered was the ability to create a mock prototype without having to write code. This is ultimately what we decided to use to create our prototype. By using figma we could not only rapidly make changes to the prototype, but we already had a wireframe which made the process much quicker.

On top of figma allowing us to make changes to the wireframe we already have there are also a number of other benefits to using figma. By having an interactive prototype we were able to interact with our design and see how easy it was to use. We could see how well the screen flowed and how the user experience would go.

Deployment Plan

Should we decide to deploy our project there are a number of different techniques that we would implement. For one as engineers we would hold daily scrum meetings while we are in the process of creating our website. We would also utilize github or git in general to maintain our codebase. I believe that we would have a big emphasis on testing and making sure our code works. One way we might do this would be blackbox testing which we discussed in class and in the last phase. We might elect to use github actions or a similar service to automatically deploy our code and run a number of tests on it. This ensures that flawed code does not get pushed into production. It also makes it easier for us as developers to deploy our code.

As for the features we would like to add, we would first translate our prototype into a real web application. We would likely use a modern tech stack such as SQL, Node, and React, or something similar. We would add all of the features detailed in the prototype which we have already created. On top of that we would add the ability to filter tasks by user, a signin system, and user roles.

Discussion

In the future, we will probably implement the functionality discussed using technologies such as SQL, Node, and React. The future of this project mainly lies in actual implementation of features. Then after that it is necessary for us to maintain the project. We could use docker for our containerization.

A possible idea for the future is a team availability feature. This feature would build upon the filtering feature to have it so that each team member can show their availability when they are being filtered during things such as Scrum Meetings. This could make it easier for some team members to collaborate and set up meetings in Scrum itself. Another possible extension idea for our project could be a time tracking feature to show how much time has been spent on certain tasking. This would allow for managers leading these meetings to be able to know how tasks are being handled. For limitations, this comes down to how the data would be managed as well as keeping the application up with the amount of users. If we can work with those factors, we can manage those limitations.

Conclusion

The web-based scrum board we worked on was created with the main purpose of bridging any gaps that may exist in current scrum boards. Many software developers have said that traditional scrum boards are rigid and can even lead to creep due to the unclear setup. Due to this, our scrum board put a large focus on giving users the freedom to assign tasks to their liking, create names outside of the default ones, and group them however they wish. We also added in a notification system, alerting users whenever a task has been added, assigned, completed, or moved. With these features, we were able to address common problems of lowered communication, lack of accountability, and the difficulty of keeping up when working remotely. There are also functionalities based on roles, allowing managers to oversee projects and assign tasks, while others in the team are limited to what they can access. Stakeholders are given viewing privileges, and any other settings can be customized. Through the implementation of these features, the web-based scrum board is a high-level system that fosters collaboration and encourages agile project development.

References

---

[1] C. Brown, 'SE Process I', 2024.
[2] C. Brown, 'SE Process II', 2024.
[3] Stray, V., Moe, N. B., & Bergersen, G. R. (2017, April 14). *Are daily stand-up meetings valuable? A survey of developers in software teams*. SpringerLink. https://link.springer.com/chapter/10.1007/978-3-319-57633-6_20
[4] Chandana. (2024, September 24). Scrum project management: Advantages and disadvantages. Simplilearn.com. https://www.simplilearn.com/scrum-project-management-article

GitHub Repository:
https://github.com/ACarr33/ISD-Blacksburg-Software-Team-Project