Intermediate Software Design Final Presentation

Web-based scrum board

Blacksburg Software Team
Akshara Gandrakota, Aidan Carraretto, William Burriss, Aditya Rao

Problem Statement

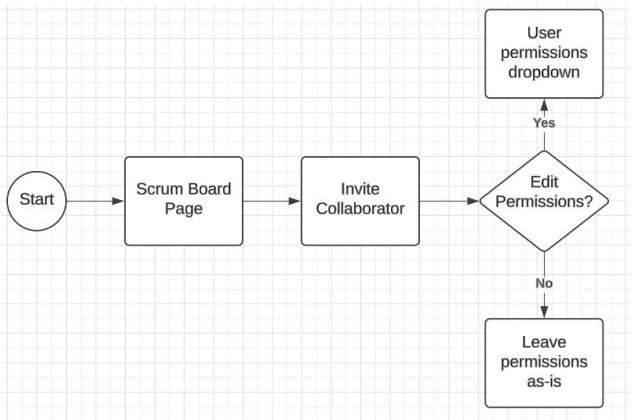
- Scrum boards are an important part of many companies, as they help break a project down into more manageable increments
- However, many teams can be limited by different factors
  - Remote working: traditional scrum boards are reviewed in person, so it becomes hard for people working remotely to stay updated
  - Different work schedules: while scrum meetings can be very helpful, not everyone can make time for these meetings in their schedule
  - Staying accountable: when milestones are discussed in a scrum meeting, it can be hard to make sure that all the employees are doing the work they were assigned, especially in large teams
- Therefore, there needs to be a more applicable, collaborative, and user-friendly approach available

Proposed Solution

- For our solution we decided to create a web based Scrum board to help keep developers on track and solve many issues faced in the world of software engineering
- We decided to make a basic Scrum board, but along with that have other features such as:
  - Tasks filters. One feature we planned to add was the ability for users to apply filters to the Scrum board. Some filters we plan on adding include: Task owner, Parent task, and Task creation date.
  - Custom naming conventions. Typically Scrum boards use the naming convention of stories and epics. However we plan to add a feature which would allow users to change the traditional naming convention.
  - Customization of categories. Traditional scrum boards only have to-do, completed, in-progress, but we wanted to have users be able to add in their own columns of their own choosing
- So far our model currently has the basic features implemented.

Use Cases and Test Plan

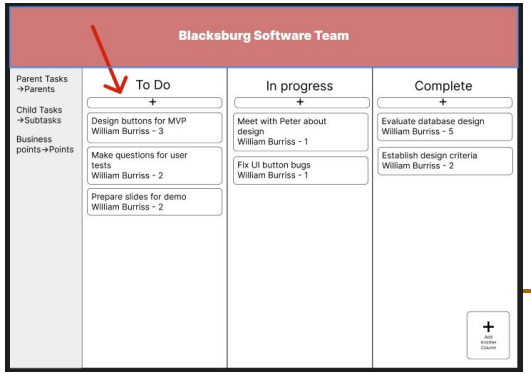
- Use cases mostly concern individual system parts
  - Creating a new board
  - Creating a new column
  - Creating a new task
  - Moving tasks/setting priority
  - Inviting collaborators
  - etc.
- Ended up similar to our black box test plan
  - Made it easier to plan how the system would be tested - many of our tests revolve around how a user would be expected to use the system
  - Test plans additionally included simulating a scrum board for a sample project



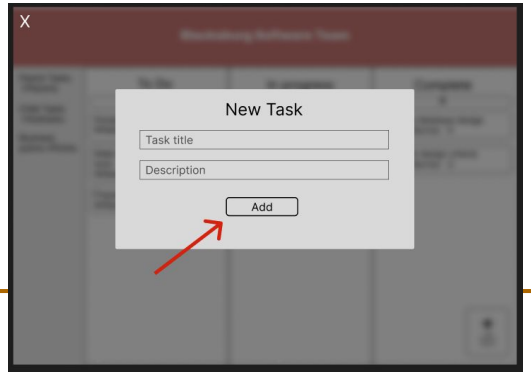
Visual Representation

Our mock user interface

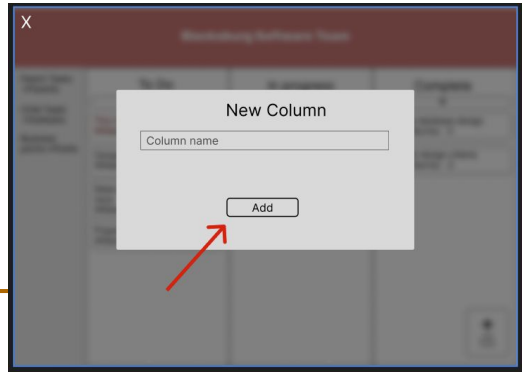
- Was created using Figma
- Allows us to simulate a working prototype using flows
- Provides us with a design that allows for many different types of evaluation



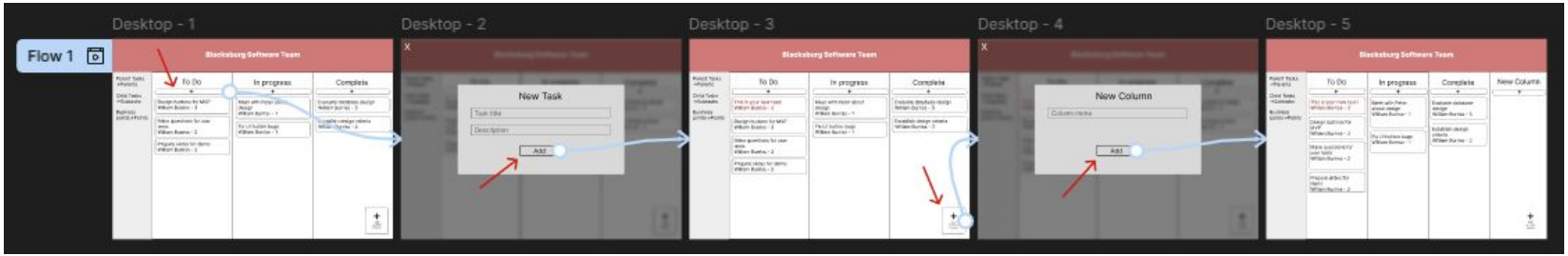
Home screen



Task adding screen



Column adding screen



Flow overview

Limitations, Future Work

Limitations

- Time constraints
- Figma
- Common knowledge

Future work

- Add more features to the Figma Flow
- Create working website
- Evaluate design to identify more necessary features

Processes and Tools

Processes

- Scrum
- Iterative Model

Tools

- Figma for UI design
- Figma Flows and prototyping for creating our mock UI

What We Learned

- Maintaining an agile approach in the project
- Importance of requirements elicitation
- Keeping track of collaboration