

# Sprint Meeting

Time: December 5, 7:20 pm to 8:20 pm

Location: Discord Call

Description: This meeting consisted of brainstorming a list of problems with the current product. We used these problems to build a product backlog with a series of user stories. Lastly, we reviewed the user stories and created a sprint backlog with an initial set of tasks for the sprint.

## Problems

**[Problem 1]** Lack of support for generating multiple types of problems

**[Problem 2]** No easy option to generate problems for each student

**[Problem 3]** Unable to export problem sets to Canvas

## Product Backlog

**[Epic 1]** As an instructor, I want to create “fill in the blank” parson’s problems from code files, so that I can test my student’s ability to complete the missing ideas in code. **[Problem 1]**

**[Epic 2]** As an instructor, I want to generate multiple-choice style parson’s problems from code files, so that I can test my student's ability to discern incorrect parts of code blocks. **[Problem 1]**

**[Epic 3]** As an instructor, I want to generate 1 problem per problem type for each student so that each student will have an independent set of different types of problems to holistically test their understanding of the course content. **[Problem 2]**

**[Epic 4]** As an instructor, I want to be able to export my generated problem sets to Canvas, so that I can efficiently create Canvas Quizzes for my students by simply importing the exported problem sets. **[Problem 3]**

## Sprint 12/05/2022 - 12/13/2022 Backlog

**[Task 1]** Create a FillInTheBlankProblemGenerator class

- Assignee: **Ramki**
- Status: TODO
- User Story: **[Epic 1]**

**[Task 2]** Write unit tests for FillInTheBlankProblemGenerator

- Assignee: **Ramki**
- Status: TODO
- User Story: **[Epic 1]**

**[Task 3]** Modify ProblemSetGenerator to use FillInTheBlankProblemGenerator to generate “fill in the blank” problems

- Assignee: **Ramki**
- Status: TODO
- User Story: **[Epic 1]**

**[Task 4]** Modify GenerationForm to allow instructors to select “fill in the blank” problem type

- Assignee: **Tanush**
- Status: TODO
- User Story: **[Epic 1]**

**[Task 5]** Modify GenerationForm tests to ensure “fill in the blank” problem type can be selected in GenerationForm

- Assignee: **Tanush**
- Status: TODO
- User Story: **[Epic 1]**

**[Task 6]** : Modify GenerationForm to allow the # of problems per problem type to be set by the user

- Assignee: **Rohan**
- Status: TODO
- User Story: **[Epic 3]**

**[Task 7]** Modify GenerationForm tests to ensure # of problems per problem type implementation is respected

- Assignee: **Rohan**
- Status: TODO
- User Story: **[Epic 3]**

**[Task 8]** Modify GenerationResult component to display the “fill in the blank” problem type.

- Assignee: **Tanush**
- Status: TODO
- User Story: **[Epic 3]**