Sprint Plan 3

Goal: Connect and finish the project by creating a workable version closest to the release plan as we can.

User stories:

User story 1:
As a programmer, I want to be able to solve a problem on a online website, that way I can have a location to practice for my coding interviews.Priority: 10 (total time- 5 hours):

> 1.Finish creating problems table and individual problem page - 5

User story 2:
As a future interviewee, I want to only have to press the submit button for my code to be graded, so that I won't have to stress about whether or not I submitted my work. Priority: 5(total time- 4-5 hours):

> 1. Connect our test cases to Docker to test for success of problem - 3 hours(Isha and Thomas)
> 2. Using flask to obtain users code to test and running tests - 1-2 hours (Alex)

User story 3:
As a user, I want to be able to see a leaderboard where I can compare my speed with my other peers. Priority: 3 (total time- 5 hours):

1. Create a list that keeps track of the top 5 shortest times. 5 minutes (Alex)
2. Grab the user's time spent solving the problem and compare to list, adding into list and sorting list if it is fast enough to qualify for the top 5 times. 30 minutes (Alex)
3. Turn the list into a database so it keeps track throughout various sessions. 4-5 hours (Alex) (tentatively)

Team roles:

Thomas Shaddix: Product Owner
Olivia Wong: Developer
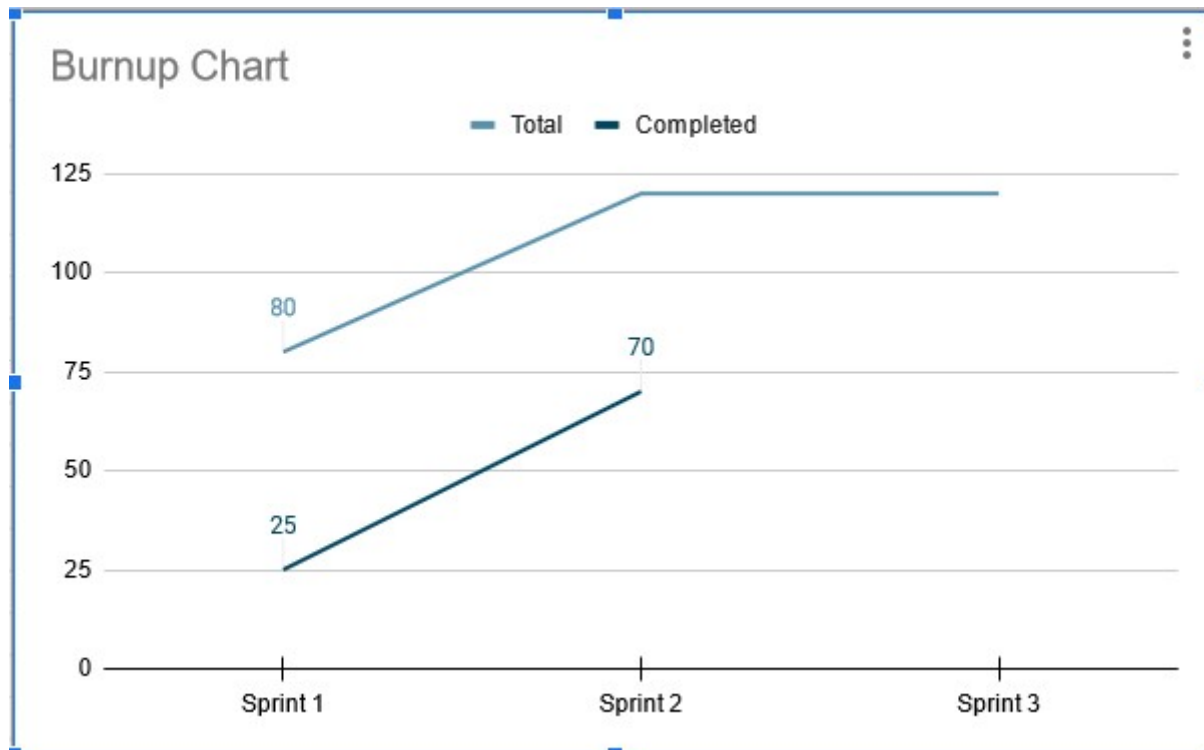Isha Manglik: Developer
Alex Chen: Scrum Master

Thomas Shaddix: User story 2, Connect test cases to Docker to test success
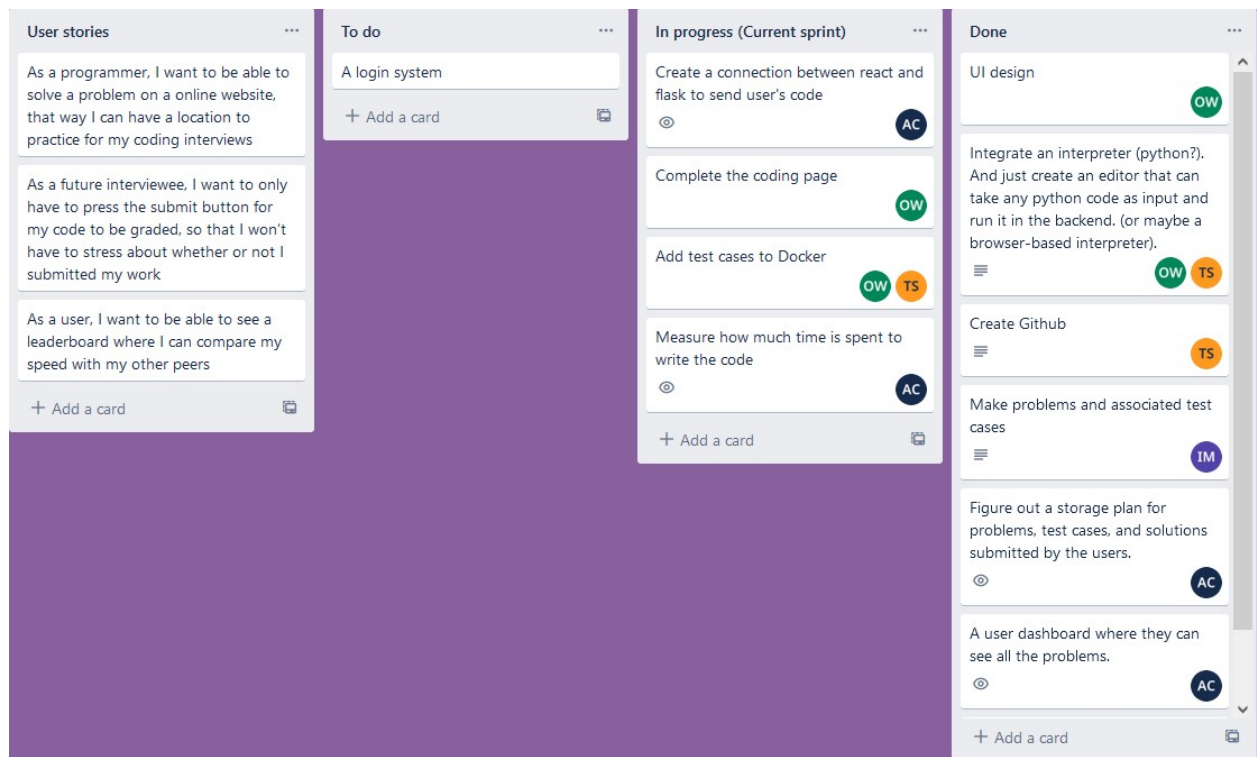Olivia Wong: User story 1, Finish creating problems table and individual problem page
Isha Manglik: User story 2, Connect test cases to Docker to test success
Alex Chen: User story 2: Use flask to obtain user's code to test and run tests

Burnup chart

Scrum Board

User stories	To do	In progress (Current sprint)	Done
As a programmer, I want to be able to solve a problem on a online website, that way I can have a location to practice for my coding interviews	A login system	Create a connection between react and flask to send user's code AC	UI design OW
As a future interviewee, I want to only have to press the submit button for my code to be graded, so that I won't have to stress about whether or not I submitted my work	+ Add a card	Complete the coding page OW	Integrate an interpreter (python?). And just create an editor that can take any python code as input and run it in the backend. (or maybe a browser-based interpreter). OW TS
As a user, I want to be able to see a leaderboard where I can compare my speed with my other peers		Add test cases to Docker OW TS	Create Github TS
+ Add a card		Measure how much time is spent to write the code AC	Make problems and associated test cases IM
		+ Add a card	Figure out a storage plan for problems, test cases, and solutions submitted by the users. AC
			A user dashboard where they can see all the problems. AC
			+ Add a card

Meeting dates: Thursday 8am,Friday 2pm, Sunday 6pm, Monday 1pm