**Checklist:**

**Sprint 2 Review**

1. What went well during the Sprint?

* Everyone was able to contribute by committing changes to our Github repository.
* Everyone was able to pull changes rather easily from the Github repository.
* We were able to meet more frequently than before, sometimes meeting twice a week.
* We had lots of outside communication, regularly communicating via Discord.

1. What problem did the team run into? How were those problems solved?

* We found that it was sometimes hard to delegate tasks and responsibilities, so we would just assign people tasks and have sign up opportunities to help remedy the issue.
* Some of us did not completely understand design patterns, so we would go back and rereview the course material to ensure that we understood design patterns before implementing them in our code.
* When it came to choosing design patterns, we would sometimes disagree on the best pattern to use, so we would settle things by vote.

**Sprint 2 Retrospective**

1. Inspect how this Sprint went with regards to people, relationships, process, and tools;

People: Some people were shy and afraid to pitch in at group meetings, so we may need to do a better job at facilitating cooperation and motivating others.

Relationships: People sometimes have a bit of a tough time understanding the process, so we may have to review things again at times.

Process: We found ourselves moving a little bit more slowly this time, but still making more progress. Perhaps, we are exerting ourselves a bit and making this more complicated than it needs to be?

Tools: This time around, everyone was able to make changes to the repository more easily. We also had people experiment with different IDEs and platforms for the project.

1. Identify and order the major items that went well and potential improvements;
2. We were all able to look ahead and plan our tasks accordingly.
3. Everyone was able to implement some aspect of their user stories in the repository, although it is possible that the quality of our work could be higher.
4. Everyone was able to work on their own code, although we noticed that what we originally planned needed corrections, so we may be better off being more abstract when planning.
5. Create a plan for implementing improvements to the way your team does its work.

* To increase productivity, we would need to work on reducing the amount of time spent delegating by holding ourselves accountable to knowing and understanding the material before coming to meetings.
* Instead of having voluntary sign ups, we may need to assign people tasks randomly to save time and effort.
* We may need to make sure that everyone has easier access to the documents we are working on and is on the same page to avoid lost productivity.