Sprint Retro 1

Sprint: Sprint 1
Date: 14/08/2020
Scrum Master: Alana Tobgui (s3760538)
Development Team: Alana Tobgui (s3760538), Daniel Bound (s3719027), Gleb Bogachev (s3694543), Alex Vo (s3781244)

Things That Went Well

What went well was coming up with our user stories quickly and adding the acceptance criteria and tests for each of them. We assigned tasks and responsibilities evenly and early on so that we could get started on the tasks as soon as possible. ClickUp was a useful tool as we were able to see the priority of the tasks we set and who they were assigned to. We communicated effectively using MS Teams by attending meetings almost every day. The communication and experience of the development team aided in the timely completion of the features required in this sprint.

We implemented all features we intended to, including creating a basic homepage for a user that is logged in or anonymous, as well as a contact us page and an about us page. We also implemented login and sign up functionality which were meant to be done in the next sprint.

Things That Could Have Gone Better

What could have gone better was using the text chat in MS Teams more frequently to ask questions and give updates on our tasks rather than exclusively using voice meetings to do this.

Things That Surprised Us

What surprised us was the use of software that some of us have not used before or have little experience in such as React, Node.js and npm. This required us to do some research on each of them. Since some of us were using different operating systems and the front end was developed on a Mac, we struggled to figure out how to make our application compatible on a Windows operating system.

Lessons Learned

What we learned from the points above is that we should continuously attend daily meetings, communicate through MS Teams and use ClickUp to assign and update tasks. All team members should continuously improve their individual skills in their area so that we are ready for the next sprint.

Final Thoughts

Overall, what we did well and should keep from this sprint is the effective communication and early assigning of tasks so that we can finish the milestone well before the due date. Organising and attending daily meetings allow the team to see the progress that each member has made and gives us a clear objective on everything we need to do.

Our team may not need to change much, but one thing we can do is to have more members working on development of the code to ease the workload off of our front end and back end developers.

Burndown Chart

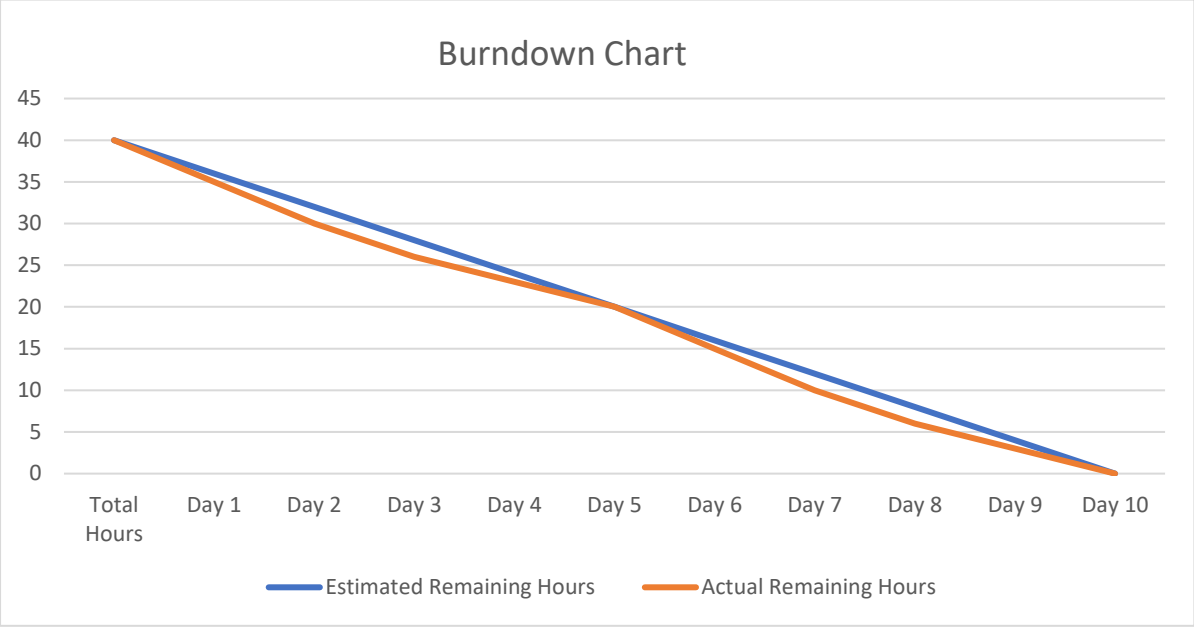

Figure 1

The Figure 1 shown above displays the burndown chart, which is a graphical representation of work left to do versus the time for the current sprint.

The chart shown above is a line graph, with the blue line depicting the ideal burndown or the ideal case of estimated remaining hours whereas the orange line depicts the actual remaining hours of the tasks being taken into consideration for the current sprint.

The vertical axis displays the story points or the total effort whereas the horizontal axis displays the time spent each day.

To calculate the burndown chart, we are first required to calculate the estimated effort which is essentially the available hours in the sprint divided by the number of days, in our case we had 40 hours, over 10 days equating to 4 hours a day.

The next step is to calculate the actual effort spent to complete the tasks each day, which is essentially the difference between the total estimated effort and the time spent each day to complete the sprint. For example, we spent 5 hours working on day 1, 5 hours on day 2, 4 hours on day 3, etc…

The data is expressed in a table below:

	Total Hours	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10
Estimated Remaining Hours	40	36	32	28	24	20	16	12	8	4	0
Actual Remaining Hours	40	35	30	26	23	20	15	10	6	3	0