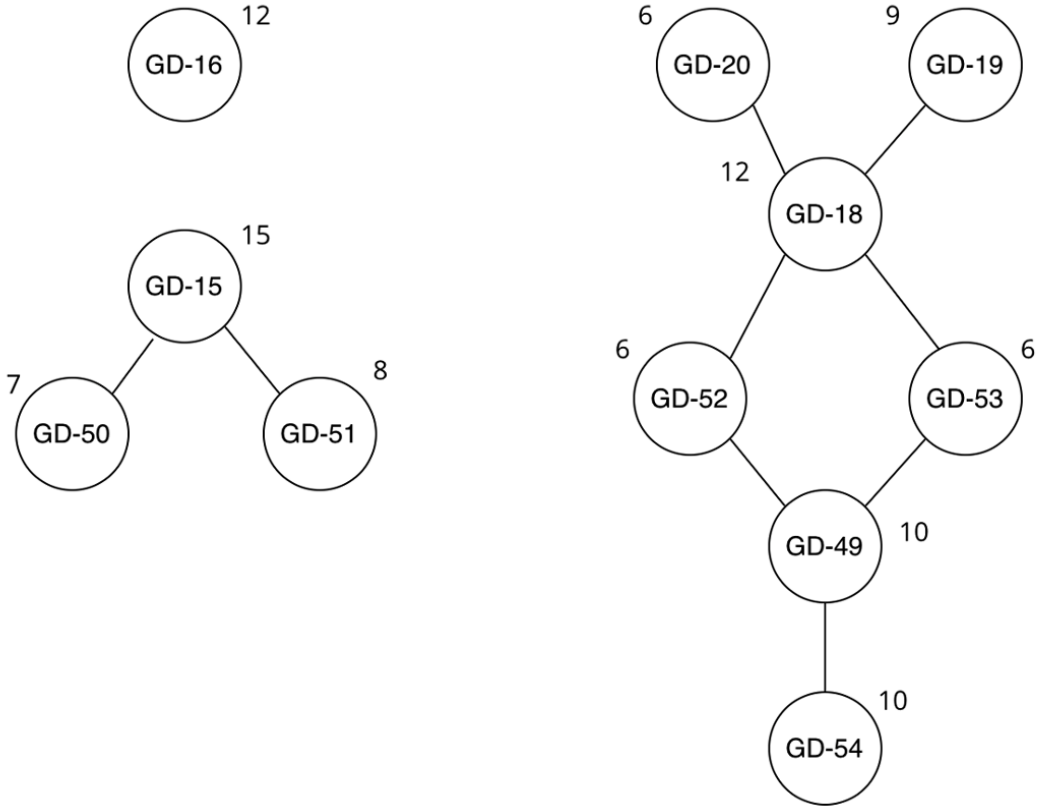
Schedule



TaskID	Time	Dependencies	Critical Path
GD-16	12	-	-
GD-15	15	GD-50, GD-51	-
GD-50	7	-	-
GD-51	8	GD-51	-
GD-20	6	GD-18	-
GD-19	9	GD-18	*
GD-18	12	GD-52, GD-53	*
GD-52	6	GD-49	-
GD-53	6	GD-49	*
GD-49	10	GD-54	*
GD-54	10	-	*

Critical Path:
GD-19, GD-18, GD-53, GD-49, GD-54

Sprint Schedule Explanation

The way we kept our sprint on schedule was very similar to sprints 1, 2 and 3. Since this strategy worked for both sprints, we decided to mimic the strategy as close as possible. Furthermore, we ensured that all members were on the same page during the first stand-up meeting.

To keep our sprint on schedule, firstly, we ensure that every team member receives a task that that they are interested in working on or are strong in. Since this was the last sprint, to ensure this principle was kept, mappings of each member to each task were kept consistent in terms of frontend and backend tasks. That is, members working on the frontend for the first 3 sprints, were given frontend tasks. Similarly, members working on the backend for the first 3 sprints, were given backend tasks.

Next, as stated in the sprint guidelines, we made sure we held each other responsible for our progress through at least 3 stand-ups per sprint. Despite only three official stand-up messages per person being posted, we often have smaller meetings, for team members working on similar tasks, to bounce ideas off of each other and pair program.

Lastly, at the beginning of the sprint, we make sure not to overcommit in terms of velocity. After the first two sprints, this became a lot easier to judge. However, since this was the last sprint, there were more tasks to complete naturally, which led to a higher velocity that normal.