

Sprint 2 Report

Product Name: Automatic Cell Counting

Team Name: Cell Counting Project

Date: 05/4/2021

Actions to stop doing:

- Stop waiting to ask questions so long. We spend too much time trying to test hypotheses when we are working in a field we have no experience in. It would be better simply to get advice from someone who is more experienced with the topics at hand.
- Stop putting all changes in one branch. There are a couple times when our changes modified others' work.
- Stop leaving temporary code in our commits. We had accumulating print statements that made reading our outputs more difficult later on in our development.

Actions to start doing:

- Ask TA for advice more often. Akila has a ton of really useful prior knowledge about these kinds of projects
- Reconsider our user story points every so often. We were a little inaccurate in terms of how long we thought some tasks would take and how much time we would have to commit to the project.
- While we updated the scrum board a few times we should be doing it every scrum meeting.
- Make a document of all lingering questions we have and their test/google results and sub-problems, their relevant context, and who is working on them.
- We can only test in Google Collab, so we are making all of our changes in the Google Collab. We need to commit our changes from Google Collab to our GitHub more frequently.

Actions to keep doing:

- Working together to solve problems we encountered and helping each other out.
- Team communication is really good and we are aware of what each of us is working on and whether we need help or not.

Old actions to keep doing

- Daily scrum meetings are within the time limit. We were good at summarizing our progress and needs while getting all necessary information across.
- Important documents and other notes are in the Drive for everyone to use. It's well organized and there has been very little confusion as to where documents are located.
- Using the Trello to organize our tasks. We did have a very good organization on Trello during the times we did use it. It helps keep the whole sprint in perspective and helps keep us from getting lost in the weeds.

Work completed:

- We completed the UI download option

- We are able to train the model without running into input errors due to the difference between our Jsons and their jsons.
- We are able to run predictions without encountering formatting errors.
- We finished annotating images

Work not completed:

- We are still not able to train well enough
- Either we are still not able to predict accurately or we are not interpreting the output correctly.
- Nothing from our model has been hooked up to the GUI yet.
- We were not able to make sure our model is not counting specks of dirt or visual clutter since we could not get it to predict accurately.

Work completion rate:

User stories completed: 1

Total number of estimated ideal work hours: 54.5

Total number of days in past sprint: 14

User Stories/Day: $1/14 = 0.07$

Ideal Work hours/Day: $54.5/14 = 3.89$

- We heavily underestimated how stuck we would get to get the model to train correctly and get an output that we could visualize and understand. So we had to increase our total ideal work hours a lot to accommodate for the increasingly difficult user story.

Burnup chart

