SERP Group 9 - Retrospective Report 3

Meeting time: 6 pm 12th Sep 2020
Attendees: Basil, Daniel, Josh
The meeting was conducted on the 12th of September, 2 days after the client meeting. The main items of discussion were:

- Discussions of the effectiveness of estimation techniques and guidelines for story points
- Planning and strategy for future sprints and the features that will be delivered incrementally
- Discussion of features necessary to support the client's desired user experience

What went well:
This sprint was the first after the major version, 1.0 release of the project. As such, it involved the initial development of the major components that would support the fundamental core features of the newly defined simulation and training tool. The team defined the architecture for this tool which comprised a web client, intermediary Node.js server, and a logical python model. The corresponding components which implemented this architecture were developed this sprint.

The client side visualisation of the outputs from the logic model was primarily completed by Basil, where he utilised a javascript drawing library to create a prototype demonstrating how we would present the SEIRD outputs of the epidemic equations in a dynamic, engaging manner. While not presented in the context of the complete user interface, we received positive feedback from the client on the features and style of the visualisation, while noting some areas for improvement to aid interpretation.

Another component of the architecture developed during this sprint was the web server which communicated with the client and the logic model. Josh was largely responsible for this task. He successfully set up a Node.js server, established the fundamentals for a web socket communication between this server and the python model, and also deployed the project to a production server so that it can be accessed remotely. Along with setting up this component, Josh also set up GitHub actions to allow for automated testing workflows of each component. This included our existing test suite, and new unit tests for the logic model which Daniel co-authored.

The final component developed during this sprint was the Python logic model, which is what actually computes the calculations, holds the instance data of the simulation, and controls the flow of execution. Daniel completed all the tasks related to this component, which involved refactoring the existing SEIRD epidemic equations into an object orientated program with more dynamic control over parameters and functions. This component was the most developed during this sprint, and should support most functionality of future features.

Another continued success of our project this sprint was our communication with the client. We collectively agreed on a scheduled weekly meeting time, where we will discuss progress and activities with the client in between officially assessed meetings.

What was lacking:
One thing we did not implement in this sprint, which is important to the overall deliverables of the project was an interactable user interface that will allow the user to manipulate the variables of the simulation during its runtime. It will need to be designed and implemented in a way that promotes a quality user experience.

Another aspect that is missing from this release is the communication between the client and the server, fundamentally important to allow a user to change values over the course of the simulation. We have created the code in the server that will pass the relevant data to the client, but we have yet to actually send it and prepare the client for its arrival. To do this we will need to employ the web server created during this sprint and enable the client to receive and handle these messages.

One thing that was brought to light during this meeting was that the purpose of the tool, and its visualisation are not immediately apparent to the uninitiated. Although we as the developers and the client could understand our progress, other people who were not as familiar with the project details struggled to make sense of it. This is partly due to the interface and communication not being fully implemented yet, and we believe it will become easier to understand once the user interacts with it and sees how they can affect the model. However, a large element of this is that the visual communication of the model is lacking. The dots that appear on the screen do a poor job of signifying the simulation, and although once explained they make more sense they are not immediately apparent to the user. We will work on this by differentiating the dots in not only colour but also in shape, and research some more ways that we can make this change more noticable.

What did you learn:
We learned several things while working on this sprint. The first of which were better ways to handle the implementation of testing throughout our system. We invested some time setting up GitHub actions, this automates the running of all of the test suites that we have created. This simplifies the running of the tests to ensure that they occur automatically whenever anything is pushed to the master branch, or when a pull request is created. This also prevents us from merging non functional code (In terms of the test suites we have written).
The next thing that we learnt was how to better create user stories and determine their complexity estimates. We have determined as a group that spending time working together on writing the user stories and assigning them different complexity scores together greatly assists with the organisation and distribution of the workload between each of us.
We also learnt about some visualization libraries that allowed us to create the visuals that will help the users of the system interpret how the modelled virus is spreading throughout their controlled population.

What are you planning on doing in the next sprint:
This next sprint will be about fleshing out the functionality of the model. Completing the communication between the server and the client that will enable interactive time steps and full simulations. We will also begin the design of the user interface with UX principles as a key focus and continue sharing our progress with the client through weekly meetings.

Sprint 3 Burndown Chart