Aalto University
School of Science
and Technology

CSE-C2610
Software Project

Learning diary

Blomqvist

Teemu Blomqvist 222804
Neronet Team 11

December 5, 2015

Contents

1 Project review I

In sprint 1 our team decided to work on our user stories in pairs. With 5 developers that amounted to two pairs and one person doing work solo. I thought this was an okay idea in theory, but we hadn't really discussed how we would implement pair work in practice. A problem we had with my pair was that we tried to focus on one user story at a time, but the user stories we implemented were too small for us to split the work properly, and often I had no part to work on. As a result I spent a lot of time not coding myself but simply trying to help my partner with whatever problems arised. In hindsight this was still somewhat useful as I was fully aware of how my partner implemented each of the functions, and we solved problems faster. On the other hand, I think we could have used our time more efficiently. In an optimal scenario we would both have something to work on at all times, and we could ask each other for help on problems and then from time to time would sync up on what the other has done (also possibly asking the other for clarification on his own code) so that both are fully aware of how all parts work. If we still keep on working in pairs this will be something to remember.

Our team didn't really utilize the advantages of our work practices. We do all our work in the same space, presumably to allow easy communication. When it came time to merge the parts of our work that needed to work together, both pairs had no idea on how the other pair's part worked and thus merging them properly became difficult. In hindsight we had a lot of time to work out the necessary interfaces between our two parts, but both pairs were simply focused on getting their part of the work done. This is something we should obviously improve.

In sprint 0 we spent most of the time trying to figure out what would be the best way to implement our system, and while we did end up using a lot of man hours on that, it did help a lot in the development, as everyone now knows on a high level of abstraction how the system is supposed to work. I don't think we could have really started developing a usable system without that knowledge.

My contributions: I did 3 user stories with my pair: - Being able to specify experiments - Being able to specify clusters - Getting a status report of all the defined clusters and experiments