Sprint 2 Retrospective

**Team 71 - Really Good Boys**

|  |  |
| --- | --- |
| **Student Number** | **Team Member Name** |
| N9123971 | Fahd Ahmed |
| N9365681 | Charles Alleman |
| N9723960 | Mikhail Burkot |
| N9721657 | Ashley Husband |
| N9745734 | Joshua Lowe |

Tutor: Jesse St. Germain

Version: 1.0

Date: 06/10/2017

**What We Did Well**

**Charles:**

**Mikhail:**

1. During this sprint I found that our time management was better than last sprint for the first half. This is clearly identified when comparing both sprint 1 and 2 burn down charts. I think this was mainly due to the fact that this time we prioritized this subject over others and wanted to be a more effective team.
2. Our communication was yet again still in very good condition. Our time to respond to website queries, artifacts and other was very quick (probably one hour max), with majority being answered in the first couple minutes.

**Ashley:**

**Joshua:**

1. During this sprint we maintained good communication and with our extended knowledge on git we were able to have a steady flow of work progression throughout the project. We were able to allocate tasks quite effectively and efficiently.
2. With our feedback from last sprint we were able to divvy up tasks more effectively, distributing a wider variety to each of our members. This helped in maximising our portfolios to include a more diverse representation of work we completed.

**Fahd:**

**What Could Be Improved**

**Mikhail:**

1. Like I said previously in ‘what we did well’ in how we managed to successfully maintain a better burndown distribution for the first half, unfortunately it wasn’t the case for the other half. This was mainly derived from other university subjects once again and members having to prioritise them as this one was on schedule during that point of time.
2. Lastly the developers had difficulty throughout numerous days where a part of the code would stop working. This was due to how ineffectively we operated GitHub. Due to this, most of us spent the our time fixing what got broken during a merge, pull or fetch. This time could of been eliminated with developers worked in separate branches and double/triple checked that all website functionality is working before pushing/merging to master branch.

**Joshua:**

1. Same as last time we had a bit of difficulty in maintaining a steady workflow throughout the project. It wasn’t as detrimental this time though because we were generally better organised. This is a hard area to improve as we were very busy with other assessment and commitments outside university, however, fitting in work wherever we could may have helped.
2. As mikhail said, we did break each others code a few times during merges, (probably mostly me doing the breaking), I should have operated under a different branch rather than committing to the master. Most of the breaks had to do with the database which pulling from git doesn’t really effectively merge together like it does with code.

**What Will Change Going Forward**