Sprint 1 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:**

1. Quick and efficient communication on how to solve a bug in the system (Mikhail reporting bug in system, josh fixed within a couple hours)
2. Separation of system into paper work and programming to reduce the amount of requirements to multitask programming and designing of system (Fahd design website, charles build html, josh integrate into django)

**Mikhail:**

1. The seamlessly online active communication was very effective. Having everyone actively responding in the chat to help with a general response, code or documentation was very helpful. It kept everyone on the same page and the overall flow of the project.
2. The way we laid out the user stories (story checklist spreadsheet) along with the task checklist from the (sprint 1 release & sprint plan document) was very easy to read and track. I believe this made it easier on ourselves when developing the code and the first sprint plan release.

**Ashley:**

1. Effective cataloguing of tasks that allowed us to see what still needed to be done and helped me (as SCRUM-master) designate tasks of sufficient priority to developers.
2. Good team communication allowed our team take portions of tasks and divvy them up among developers who were more skilled at certain portions than others, resulting in more efficient work.

**Joshua:**

1. Whether it was GitHub or Google Drive, having the synchronized group workspace was great in keeping up to date when paired with a group chat. Also, Charles’ excel skills were pretty convenient when planning our development
2. Honestly, I liked the use of scrum roles and disparity between development and documentation, primarily because I enjoy developing, whereas for documentation - not so much.

**Fahd:**

**What Could Be Improved**

**Charles:**

1. Focus on setting deadlines for tasks to be completed
2. Distribution of easy and difficult/time consuming software (Josh has literally 40 projects completed which includes database and table creation because he setup the framework and works harder than everyone)

**Mikhail:**

1. Distribution of artifacts should be more equal (both documentation and code). This unit focuses on both roles (developer and IS) to contribute to each others roles and during this sprint we had Josh doing the majority of the code, leaving the IS students with either limited/no code left to complete and vice versa for the documentation side.
2. Feel like we struggled time wise in the early stages of development (losing more than a week worth of work). This was mainly due to the fact that everything was new to the group and with all of us having time conflicts outside of this unit. If we all establish an effective timetable and be more strict on our deadline schedules this will improve the overall performance (which would be seen in the burndown chart).

**Ashley:**

1. Inconsistent time management. Development on the website moved at a shaky pace, and meetings were very inconsistently timed or forgotten about.
2. Misunderstanding aspects of the development tools and the assignment criterias meant that our estimations of some of the task times were very off base.

**Joshua:**

1. As Mikhail said, we had a slow start. A major aspect of this project’s goal was to encourage a steady flow of commitment from the start – to which I understand simulates a work environment. However, it was hard to set such a pace in a learning environment due to committing a lot of time elsewhere (e.g. Work, other units). It’s hard to improve in this area because it isn’t very predictable. Although, I do believe it could have been improved if we started on the first day, no matter how little headway we could make. I think that would set the momentum for a slow accumulation of progress.
2. When planning the sprint tasks, we weren’t very familiar with the Django environment, which lead to a lot of inaccuracies in predicting what tasks we need the most, and how long they would take. So, I’d say a solid improvement would be familiarizing ourselves with the tools that we will be using, so we can plan effectively.

**Fahd:**

**What Will Change Going Forward**

Our time management will be improved by setting proper deadlines and establishing a core meeting time to stick to. Now that we’ve familiarised ourselves with the development tools, our workflow should be more consistent from here on. Thoroughly checking the criteria sheet and checking with the tutor will help give us a better understanding of the tasks required of us.