6/1/2016

Science and Engineering

Curtin University

HOOLI XYZ

Connor Beardsmore – 15504319

Brendan Lally – 18407220

Adam Brown – 18338993

Jordan Chou – 18348691

Edward Lim - 18339145

Final Report

Project Design and Management 300

# Abstract

Max 500 words / 1 page. Brief summary of contents and report structure. How work was performed and main outcomes.

This report has been developed for the project client to address the key aspects of the project implementation, design and other project management issues.

Specify access details + links somewhere

Table of Contents

Abstract 1

1.0 Introduction and Objectives 4

2.0 Background 5

3.0 Product Backlog 6

3.1 Account 6

3.2 Administration 6

3.3 Commenting 7

3.4 Posting Submissions 7

3.5 Viewing Submissions 8

3.6 Voting 8

3.7 Rewards 8

3.8 Pivotal Tracker Analytics 9

4.0 Overall Design 11

4.1 Design Details 11

4.1.1 Back-End 12

4.1.2 User Interface 12

4.2 Implementation Details 12

5.0 Project Review 12

5.1 What Went Right 12

5.2 What Went Wrong 14

5.3 What Was Learned 14

5.4 Future Improvements 14

6.0 Conclusions and Summary 15

7.0 References 16

8.0 Appendices 17

8.1 Appendix A: Sprint Reports 17

8.2 Appendix B: Pivotal Tracker Backlog 18

Table of Figures

Figure 1: Cumulative Flow 9

Figure 2: Project Burnup 10

Figure 3: Project Burndown 10

Figure 4: Curtin Ideas Logo 12

Figure 5: Version Control Commits 13

# Introduction and Objectives

Max 2 pages. What report is about. Project objectives, constraints, scope and delimitations. Approach used to complete work. Structure of report. Set the scene for rest of report (context).

# Background

Max 5 pages. Describe initial project management approach and plan. Discuss solution approaches considered. Describe other similar systems and possible solutions you have looked at. Describe tools considered for development and for SCRUM. Provide reason for choice of tools (Salami made us). Pictues of project mockup.

The requirements volatility was quite low throughout the project, with minimal changes to the requirements by the product owner after the initial specification.

Talk about pivotal tracker.

# Product Backlog

Upon receiving the project requirements from the client, the requirements were converted into a collection of 46 user stories. These stories were broken down into groupings for simplicity and to improve modulation when the stories were implemented. The following major groupings were applied to the user stories:

* Account
* Administration
* Commenting
* Posting Submissions
* Viewing Submissions
* Voting
* Rewards

Each user story was allocated into a specific grouping, with each Sprint generally focusing on completing one or two groups of stories. The group allocations for all user stories is displayed below. For a more detailed overview of the product backlog and a breakdown of stories completed per sprint, see the attached file exported from *Pivotal Tracker* in Appendix B.

## 3.1 Account

* As a User, I want to login to the platform so that I can post a submission.
* As a User, I want to logout of my account so that nobody else can access my account.
* As a User, I want to create an account so that I can use the platform.
* As a User, I want to be able to change my password so that I can keep my account secure.
* As a User, I want to access user profile so that I can view specific user’s details.

## 3.2 Administration

* As an Admin, I want access to a Admin platform so that I can perform administrative actions.
* As an Admin, I want to create user profiles so that users can use my site effectively.
* As an Admin, I want to edit submissions so that I can remove offensive language.
* As an Admin, I want to remove submissions so that I can remove inappropriate submissions
* As an Admin, I want an Admin tag so that other Users see my account as an Admin.

## 3.3 Commenting

* As a User, I want to delete my improvements so that I can remove my improvement.
* As a User, I want to edit my improvements so that I can fix spelling errors.
* As a User, I want to delete my comments so that I can remove my comment.
* As a User, I want to edit my comments so that I can fix spelling errors.
* As a User, I want to suggest improvements to a submission so that I can improve upon an idea.
* As a User, I want to comment on a submission so that I can voice my support of the idea.
* As an Admin, I want to be able to delete comments because they may contain inappropriate material.
* As an Admin, I want to edit comments so that I can remove offensive language.
* As an Admin, I want to delete improvements so that I can remove inappropriate improvements.

## 3.4 Posting Submissions

* As a User, I want to post a submission so that I can put my idea onto the platform.
* As a User, I want to delete my submission so that I can remove my submission.
* As a User, I want to edit my submission so that I can fix spelling errors.
* As a User, I want to add images to my submission so that I can add relevant images to my submission.
* As a User, I want to add videos to my submission so that I can add relevant videos to my submission.

## 3.5 Viewing Submissions

* As a User, I want to view existing submissions so that I can see all submissions posted.
* As a User, I want to search all submissions so that I can find one relevant to my field of study.
* As a User, I want to sort submissions based on popularity so that I can see the most popular submissions.
* As a User, I want to sort submissions based on date so that I can see the newest submissions.
* As a User, I want to sort submissions based on views so that I can see the most popular submissions.
* As a User, I want to categorize my submissions so that other users can find it easily.
* As a User, I want to view submissions based on category so that I can view relevant submissions.

## 3.6 Voting

* As a User, I want to down-vote on submissions so that I can acknowledge submissions I dislike.
* As a User, I want to up-vote on submissions so that I can support submissions I like.
* As a User, I want to see the total votes for a submission so that I can see how popular it is.
* As a User, I want to up-vote on improvements so that I can support improvements I like.
* As a User, I want to down-vote on improvements so that I can acknowledge improvements I dislike.
* As a User, I want to only vote once for a submission so that voting is fair and equal for all users.
* As a User, I want to un-vote on submissions so that I can change my mind on voting.

## 3.7 Rewards

* As a User, I want other users to see my level when I post so that I can show off my level.
* As a User, I want to gain levels based on my points so that I am rewarded for utilizing the platform.
* As a User, I want to gain points when my submission receives votes so that I am rewarded for good work.
* As a User, I want to gain points for posting improvements so that I am encouraged to post again.
* As a User, I want to gain points for posting submissions so that I am encouraged to post again.
* As a User, I want to gain points for voting on submissions so that I am encourages to vote again.

## 3.8 Pivotal Tracker Analytics

The following figures illustrate the progress through the product backlog over the period of the project. Figure 1 shows the cumulative flow of points, while Figure 2 and Figure 3 display the project burnup and burndown respectively.

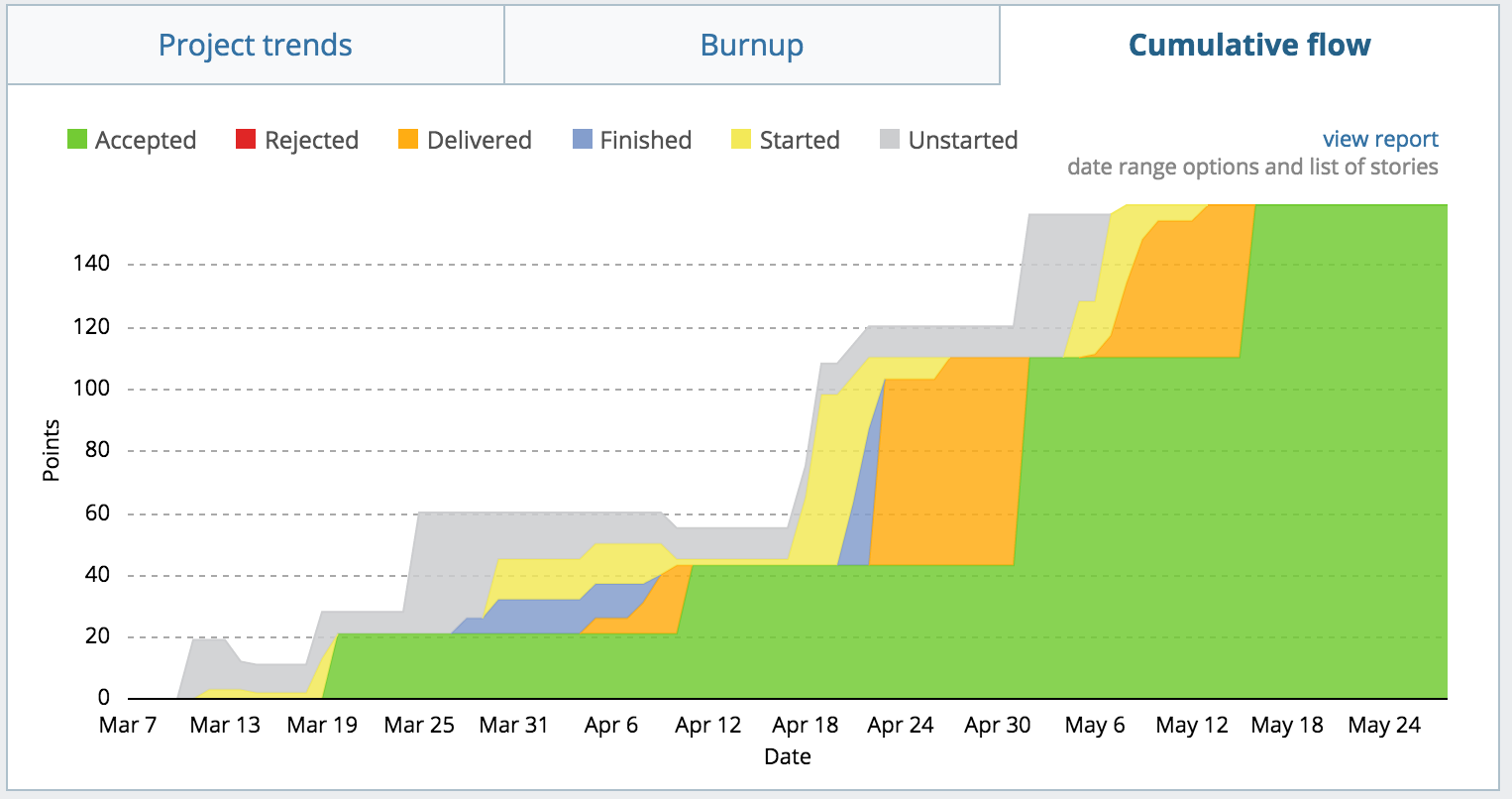


Figure 1: Cumulative Flow

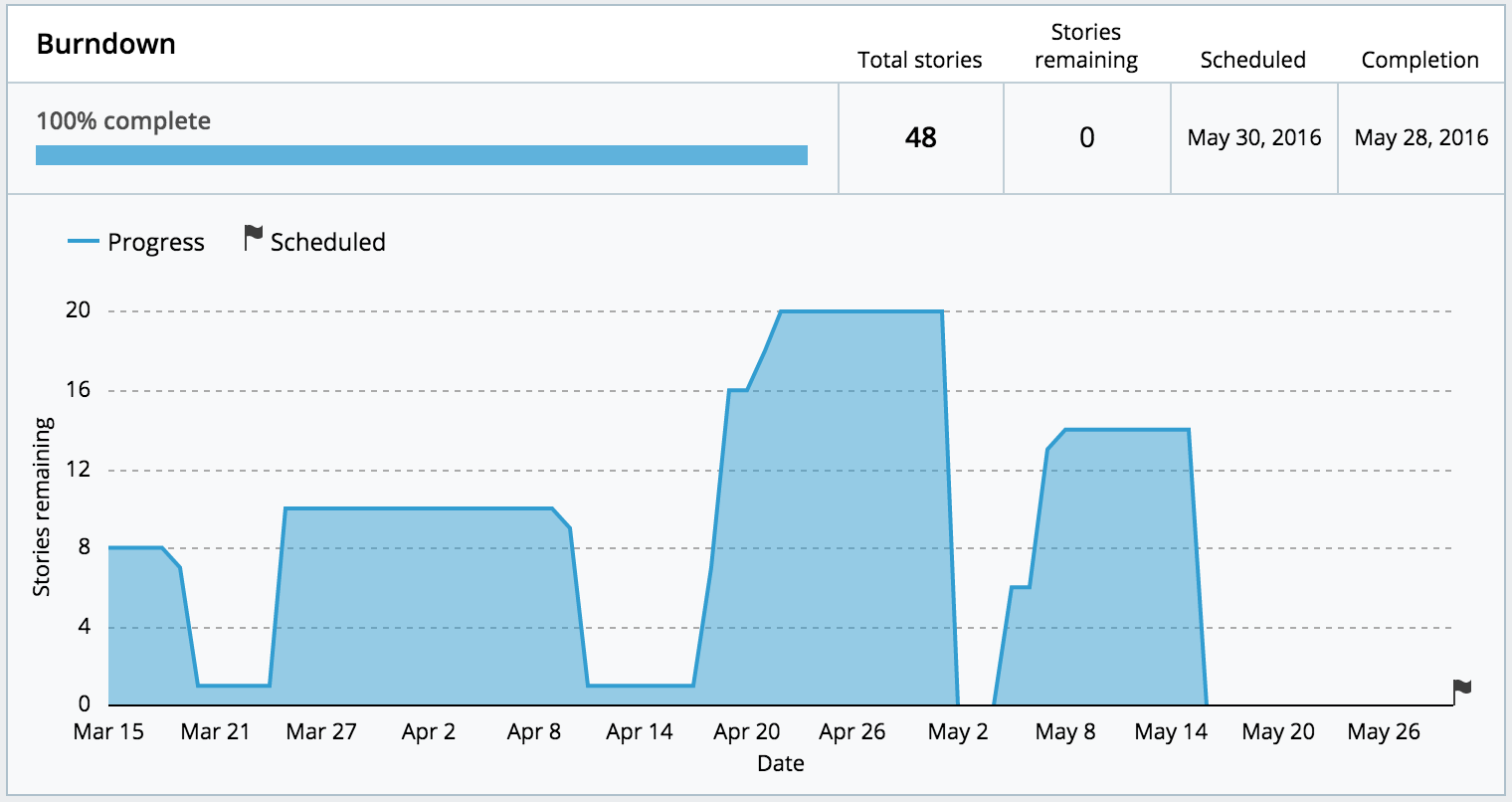


Figure 3: Project Burndown

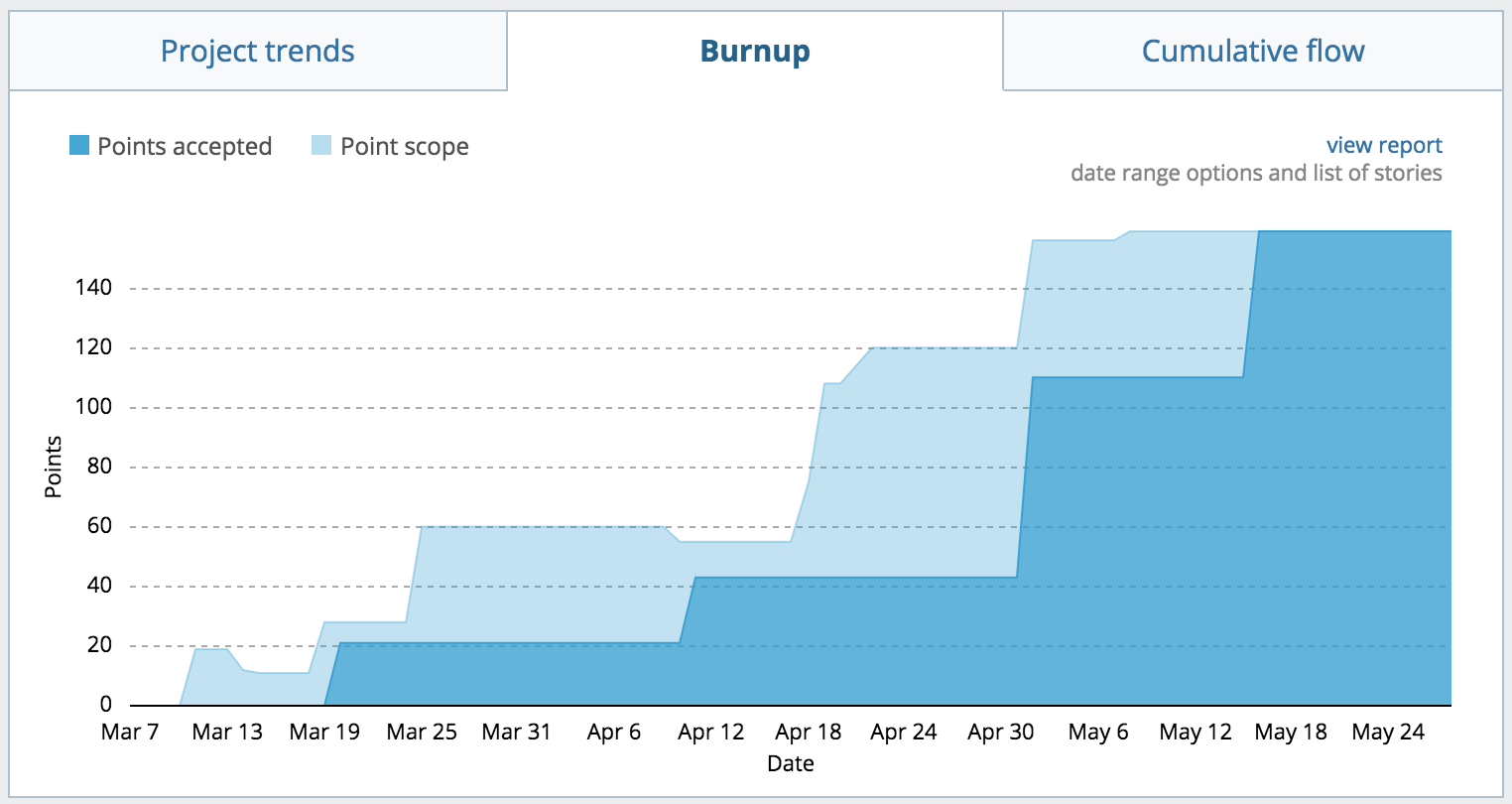


Figure 2: Project Burnup

# Overall Design

## 4.1 Design Details

*Relevant design details. Uml, er diagram etc. provide reasons for choices. Report changes to user stores due to the design process. So many ER diagrams + relations*

*UI design, CSS and template design*

* Design Introduction
* Backend design
  + Submissions
    - Creation
    - Deletion
    - Modification
  + Comments/improvements
  + Voting
  + Reward system
* UI design
  + Goals
  + Design change over time
  + Final design



Figure 4: Curtin Ideas Logo

After initially reading through and understanding the requirements for the project, each user story was created and placed in a sub-group: admin, user accounts, submissions, voting, commenting or rewards.

It was decided quite early on in the project to implement the website using the Python web-framework, Django. This would allow for easy to use and powerful admin features, along with well-defined ways to implement all the functionality deemed necessary for the project to work (ADD MORE REASONS WHY USED DJANGO). Django’s model system seemed very applicable for creating objects such as users, submissions and comments which were central to the overall design.

Figure x shows the UML diagram for the design of the project.

### 4.1.1 Back-End

One of the major design issues that was central to the project as a whole was the submission. The submission model needed to store information about a user-generated idea including the title, author, category, the idea itself and links to external webpages. The majority of the project features in some way interact with the submission, be that viewing, commenting on, suggesting improvements or voting. As well, submissions had to be both editable and removable by the correctly privileged users/admin.

We wanted a user to be able to create an account storing basic personal and contact information that could be used to identify the user and for them to participate in submitting, commenting and voting on submissions. A user would have to be currently logged onto the system to be able to submit, comment and vote on submissions. However, submissions could be viewed by anyone of the general public without requiring an account.

### 4.1.2 User Interface

The interface design went through numerous iterations before eventually settling on the current design.

The main aims of the interface design were to create a sleek, modern interface which was easy to navigate and to use.

## 4.2 Implementation Details

Document the implementation including tasks, effort, features and task distribution (planned and actually completed). Screenshots of functionality. Changes to initial design. What testing was done? \*lol none.

# Project Review

## 5.1 What Went Right

Numerous aspects of the project worked well for the group and contributed to the overall success of the project. Throughout all Sprints, the moral of the team and the work ethic produced by each team member was strong. Being able to choose our team members was a likely producing factor for this, with everyone in the team already knowing each other well before the start of the project.

The aspect that helped the most in the product development was the high level of communication between all team members. Every member attended the weekly SCRUM meetings that were held and these worked to guide the team in completing their designated stories for each sprint. In addition to these formal meetings, communication was also performed via a *Facebook* group message. This allowed us to be in constant communication and ask for help or clarification instantly when it was required.

Another team-based part of the project that went well was the utilization of pair programming. Every story was allocated not to an individual, but to a pair of team members. This greatly enhanced performance as members could bounce ideas off one another and troubleshoot bugs and issues in tandem. As a result, this contributed to the high morale of the group.

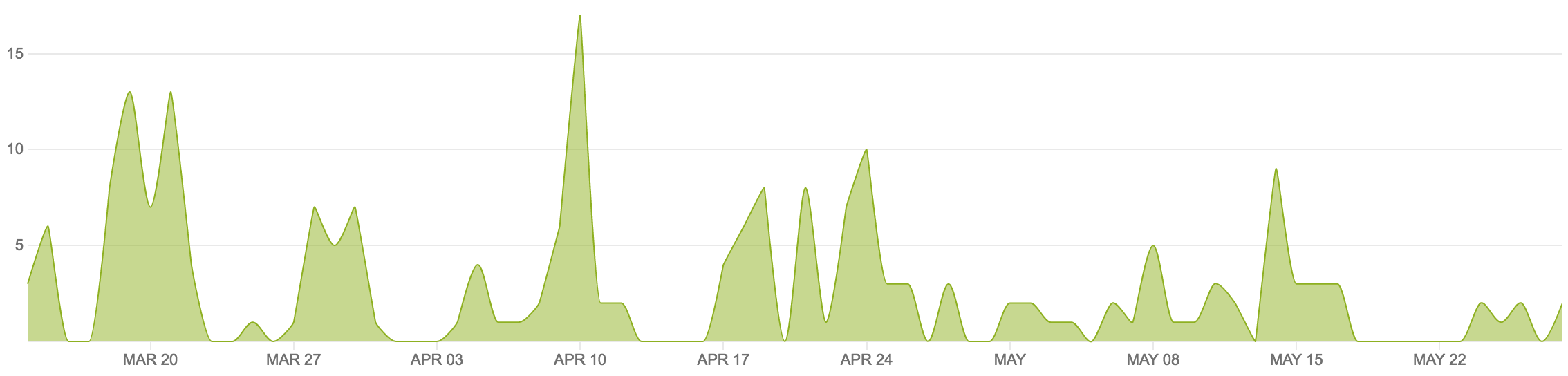


Figure 5: Version Control Commits

Numerous tools utilized throughout the project worked in our favor during product development. The use of version control via the *BitBucket* platform enabled team members to concurrently work on the project without their work interfering with one another. The commits pushed to this platform are shown in Figure 5 and analysis of this helps us to analyze our group and individual contributions over time. It also enabled us to rollback and help solve any regression errors input into the source code. The issue tracking tool *Pivotal Tracker* worked to keep the team on track and provide a metric on how our performance was going and whether we were on track or not. Without this, we expect we would not have fully implemented the solution by the final release date.

* Choice of web development frameworks (Django)
* Ease of hosting using PythonAnywhere

## 5.2 What Went Wrong

Despite the majority of aspects being productive during the project, some aspects of the project could have been performed better.

* Unfamiliar with Agile, SCRUM and Web Development
* Slow start and low productivity early on
* Poor time management at times
* External constraints

## 5.3 What Was Learned

The team has learned a lot from the project and grown significantly as a result of the project.

* Project management skills
* Time management skills
* Web development
* Issue Tracking
* Experience with Agile and the SCRUM framework
* Use of Version Control
* Group Work

## 5.4 Future Improvements

The group has several recommendations for future participations of the Project Design and Management unit. These are issues we discovered throughout our participation of the unit and may not be relevant to certain groups and their personal group-dynamic. Firstly, the weighting of the project was weighted heavily towards the SCRUM process and the Sprint and Final reports. We propose placing some of this weighting into the platform and its actual functionality. Thus, teams that produce a fully working and functional solution can receive marks for this. As a team that completed all functionality, this would have benefitted us significantly.

During the project, the one-on-one time with the project client was extremely limited. Through Sprint 3, only a small 10 minute window was held with the client. We feel that more time with the project client to discuss specifics on aspects such as styling and functionality implementation would benefit both our team and the final solution. However, this may be difficult considering the time constraints of both parties and the constraints of a university unit in general.

# Conclusions and Summary

Max 2 pages. Summary of what report has covered. Describe what was achieved in product development. Describe what was gained in project experience. Suggestions to product owner to consider in future releases. Expansions!

# References

Grellier, Jane, and Veronica Goerke. 2010. *Communication Skills Toolkit*. 1st ed. South Melbourne, Vic.: Cengage Learning.

Park, Thomas. "Darkly." Bootswatch:. Accessed March 28, 2016.

https://bootswatch.com/darkly/.

"Pivotal Tracker | Agile Project Management". 2016. *Pivotal Tracker*. https://www.pivotaltracker.com.

Schwalbe, Kathy. *Information Technology Project Management.* 7th ed. Boston, MA:

Thomson Course Technology, 2006.

Twalib, Salim. *"Agile Lecture."* Class lecture, Project Design and Management from Curtin University, Perth, Australia, February 29th 2016.

# 8.0 Appendices

## 8.1 Appendix A: Sprint Reports

The full Sprint reports for sprints 1 through 4 have been included as an attachment to this final report.

## 8.2 Appendix B: Pivotal Tracker Backlog

All data tracked within the issue-tracking tool *Pivotal Tracker* has also been included as a *csv* attachment to this final report.