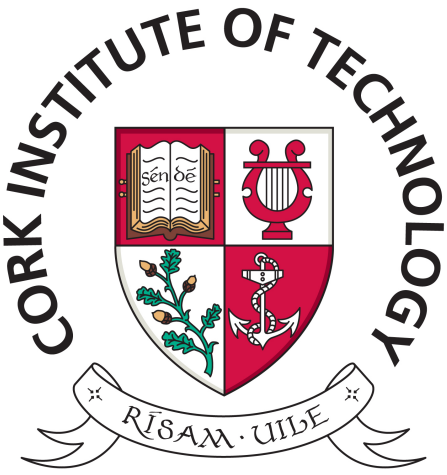


Optimizing Project Management Tool Trello For Agile Development.

by

Dan Coleman

This thesis has been submitted in partial fulfillment for the degree of Bachelor of Science in Computer Systems

in the
Faculty of Engineering and Science
Department of Computer Science

May 2020

Declaration of Authorship

I, Dan Coleman , declare that this thesis titled, 'Increased Freedom in a Project Management Tool Trello' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for an undergraduate degree at Cork Institute of Technology.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at Cork Institute of Technology or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this project report is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: Dan Coleman

Date: 04/02/2020

CORK INSTITUTE OF TECHNOLOGY

Abstract

Faculty of Engineering and Science
Department of Computer Science

Bachelor of Science

by Dan Coleman

Project management techniques are a critical element in the successful delivery of projects in a wide range of disciplines. Many of these techniques are now implemented in a range of software products including Trello, Wrike and Zenhub. This project focuses on one discipline, namely software development. Traditional software development projects are managed using a linear life cycle model, comprised of a number of phases e.g. Waterfall.

Currently, the software development industry is increasingly adopting Agile techniques in their development life cycle. This requires traditional life cycle phases to be broken down into focused tasks for faster delivery via iterative sprints.

This project highlights the lack of support for agile techniques in common project management products. Furthermore, the project then takes one popular product, Trello, to test whether its free version can be extended to better handle agile development.

With Trellos current model you are only allowed to have one extension for the free version which means for most users who use Trello there is not a large degree of customization. My projects goal is to have one central extension (or Power-Up) for Trello that will add several features that allow the tool an increased level of customization allowing a greater level of agility in a users team to track their work how they best see fit.

The tool used to implement the project is called Glitch.com. Which is is an online IDE for JavaScript and Node.js. It includes hosting, sharing and can be connected to GitHub for versioning control. These are loaded into Trello as an iframe. I have This is the chosen method as most of the resources and documentation for creating Trello Power-Ups use Glitch for hosting.

The decision to implement a Power-Up extension over a full bespoke project management tool was based on the time constraints of the project.

Acknowledgements

Special thanks to my fellow students also working on their reports for being a constant source of morale. To Cian, Franz, Nash, Stephen and my mother for helping test my project and giving their valuable feedback. To my first and second semester supervisors Oonagh O'Brien and Byron Treacy for their guidance and understanding, and Monster Energy for fuelling the project.

Contents

List of Figures

List of Tables

Abbreviations

PMT	**P**roject **M**anagment **T**ool
CPM	**C**ritical **P**ath **M**ethod
PERT	**P**rogram **E**valuation **R**eview **T**echnique
UI	**U**ser **I**nterface
JS	**J**ava**S**cript

Chapter 1

Introduction

1.1 Motivation

My motivation for this project comes from my personal interest in project management. It is my goal to become a Scrum Master and because of this I would often take over the role of Scrum Master in college projects. Due to this I frequently was in charge of choosing a project management tool (PMT) for us to use. Wrike defines Project management tools as aids to assist an individual or team to effectively organize work and manage projects and tasks.[1] I have used Trello, Wrike, and Zenhub to track projects and have realised that the freedom given to a Scrum Master to create and visualise work flow is rather limited in each tool. I feel that the core principal of Agile (making small changes to your workflow to find a way that optimises a teams efficiency) isn't being utilised as well as it could be. I have seen that one of the more popular choices for a project management tool is Trello as it is considered easy to set up and start using, and it's dashboard allows users to easily customise how they visualise their workflow. It is typically criticised for having a limited selection of features, especially its free version, due to this Trello is considered a rather situational tool, only really useful in smaller projects. For this project the aim is to make use of Trellos powerful API tools to make Trello more feature rich by adding additional features as a "Power-Up" extension. Thus solving one of it's main critiques and hopefully making Trello a tool that can be used in many more situations, while maintaining its ease of use and lightweight nature.

1.2 Contribution

The main contribution of this project is to project managers or any user who wish to track the work of a project. A common complaint about Trello is that it's lacking

in a number of features that Scrum Masters would consider important (such as Task Pointing, Epics, Blocked Stories or Gantt charts). During my work placement at IBM the company avoided using Trello as it lacking the above features which they felt were needed for tracking progress. A similar issue occurred had during my 3rd Year Project where we decided to go with Wrike over Trello as we felt that the project was too big to use Trello. The projects aim is to benefit Scrum Masters with increased flexibility of the tool, allowing Trello to be usable in many more situations.

1.3 Structure of This Document

Chapter 1 - Introduction:
This sections is a brief overview of the Project.
1.1 Motivation: An insight into what inspired me to propose this project.
1.2 Contribution: A summary of how project contributes to computer science.
1.3 Structure Of This Document: An overview of how this thesis was structured.

Chapter 2 - Background:
This sections covers how the project applies to the field of Computer Science and gives a background into Project Management Tools.
2.1 Thematic Area Within Computer Science: Assigning the Project to the ACM Computing Classification System.
2.2 A Review of Thematic Area: A review of Project Management Tools.

Chapter 3 - Problem:
This sections covers a problem, how the project aims to solve that problem, and gives a list of the requirements for the project.
3.1 Problem Definition: A description of the problem the project aims to solve.
3.2 Objectives: The goals the project aims to achieve.
3.3 Functional Requirements: A list of the Functional Requirements of the Project.
3.4 Non-Functional Requirements: A list of the Non-Functional Requirements of the Project.

Chapter 4 - Implementation Approach:
This section covers how the Project is intends to achieve the solution to the problem outlined in chapter 3
4.1 Architecture: A description of the architecture for the solution to the problem.
4.2 Risk Assessment: A review of the potential risks that could hinder the project.
4.3 Methodology: A description of how I plan to approach the project.
4.4 Implementation Plan Schedule: A schedule for how the time spent on the project

will ideally go.
4.5 Evaluation: A calculation of how much the goal of the project was achieved by.
4.6 Prototype: A description of the prototyping done to estimate the time and difficulty of the final implementation.

Chapter 5 - Implementation:
This section covers a reflection on the problems encountered, along with a description of the solution that was implemented.
5.1 Difficulties Encountered: A list of the issues that were encountered while implementing
5.1.1 Easy Problems: Issues solved with little difficulty.
5.1.2 Medium Problems: Issues that were not easy to solve, but where a solution was found.
5.1.3 Hard Problems: Issues that were to difficult to solve and as a result functional or non functional requirements were not met.
5.2 Actual Solution Approach: If any changes between the initial plans from chapter 4 and the final product, and if so, why they occurred for the following:
5.2.1 Architecture: Architecture for the solution to the problem.
5.2.2 Risk Assessment: Potential risks that could hinder the project.
5.2.3 Methodology: How I plan to approach the project.
5.2.4 Implementation Plan Schedule: Schedule for how the time spent on the project will ideally go.
5.2.5 Evaluation: Calculation of how much the goal of the project was achieved by.
5.2.6 Prototype: The prototype done to estimate scale and difficulty of the final project.

Chapter 6 - Testing and Evaluation:
This section covers the testing process for the project.
6.1 Metric: In this section I cover the testing criteria for the following:
6.1.1 Task Pointing.
6.1.2 Priority.
6.1.3 Streak Counter.
6.1.4 Start Dates.
6.1.5 Blocked Stories.
6.1.6 Non Functional Requirements to be met.

6.2 System Testing: In this section I cover how the testing was implemented in:
6.2.1 Manual Testing: The testing that was done manually by myself.
6.2.2 Survey Testing: The testing that was done by volunteers with feedback given through a survey.

6.3 Results: This section covers the results of the testing of :
6.3.1 Survey Results.
6.3.2 Were all Requirements Met.
6.3.2.1 Functional Requirements Not Met: Covers the functional requirement that were not met and why.
6.3.2.2 Non-Functional Requirements: Covers if each non-functional requirement was met.
6.3.3 Results Conclusion: Final thoughts on the results.

Chapter 7 - Discussion and Conclusions:
This section covers my final thoughts on:
7.1 Solution Review: The solution I implemented to solve the problem
7.2 Project Review: What went well/what I should have focused on more during the project.
7.3 Conclusion: My closing thoughts on the project.
7.4 Future Work: What I would add if I had more time to work on the project.

Chapter 2

Background

2.1 Thematic Area within Computer Science

The aim of the Trello Power-Up is to provide a greater selection of features to the PMT Trello. The most common complaint about Trello is its lacking feature list, to the point where some people don't even consider it a PMT. I hope that by adding features to the tool, especially with the mindset of trying to increase freedom that the tool can be improved as a PMT, while retaining Trellos best assets being lightweight and easy to use.

The project will be a "Power-Up" which is the name given to extensions built using Trellos API. The extension will be written in an online IDE called Glitch.com. Glitch is created by a company of the same name, which branched from Fog Creek Software, who also created Trello. Due to this Glitch is well equipped for creating Trello Power-Ups and comes recommended in the Trello documentation. Glitch will be used to host the code written and loads the code into Trello as an iframe (See Fig:2.1). The language I will be using is JavaScript, with Node.js as my run-time environment. The framework will be Express.

Using the AMC Classification System, the project falls under the topics of: Software and its engineering ->Software creation and management ->Software development process management ->Software development methods ->Agile software development. [Note "->" means the topic on the right is a subsection of the topic on the left]. It fits into this topic as Trello is a PMT, which allows a user to track projects. It does this in the form of a kanban board which is a considered an agile methodology. While there are many Power-Ups available in Trello currently, some even made by the Trello team, these typically add one feature. What I aim to do is create one Power-Up that completely overhauls the amount of functionality available in Trello.

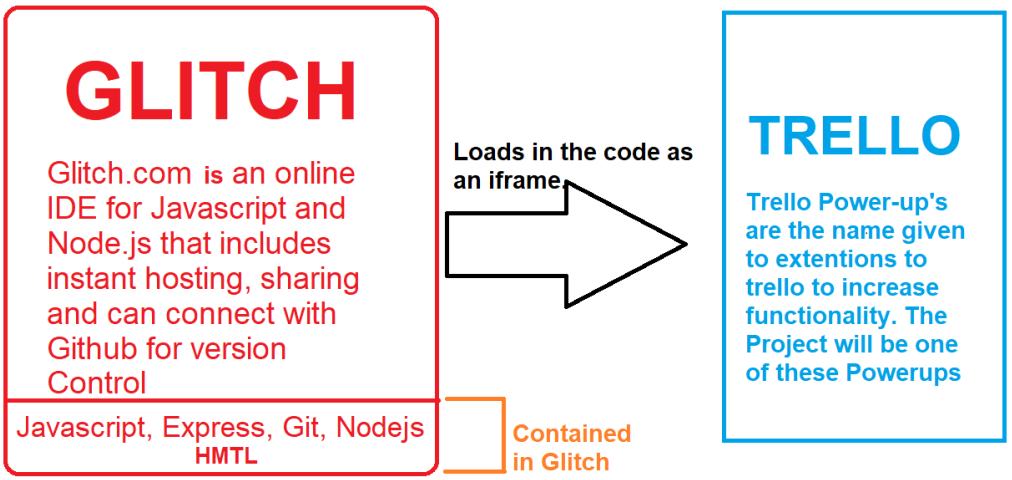


FIGURE 2.1: A basic diagram of the architecture

2.2 A Review of Project Management

The first historically relevant year for the development of project management tools is 1896, with the creation of the Harmonogram. Karol Adamiecki, a Polish economist attempted to display task development using a floating chart, which lay the foundation for project management tools as it is today. Activities and their duration were represented by the position and length of paper strips. In the header of the strips the name and list of preceding activities were given.[2] Project Management as we know it began to take form in 1911 with the publication of "The Principle of Scientific Management" by Frederic Taylor. It was an attempt to help unskilled workers transition to more complex projects by learning simple techniques. In 1917 Henry Gantt created a new scheduling diagram called the "Gantt chart". He plotted a visual timeline and plotted tasks as points with the time it would take to complete the task, he also linked dependent tasks (See Fig:2.2). It's first major implementation is said to be the Hoover Dam in 1931. To this day Gantt charts are used frequently as a tool to measure progress. [3] The American Association of Cost Engineers was formed in 1956 and with the goal to constructively exchange ideas between engineers so that they could develop technical guidance and quality education about Project Management. This is one of the first and to this day one of the leading professional societies of project managers. [5] One year later in 1957 a new project modelling technique called the Critical Path Method (CPM) was developed by two men Morgan R. Walker of DuPont and James E. Kelley of Remington Rand. In essence the critical path is about finding the longest length of time it will take to complete a project task. [3] (See Fig:2.3). The essential technique for using CPM is to construct a model of the project that includes the following:

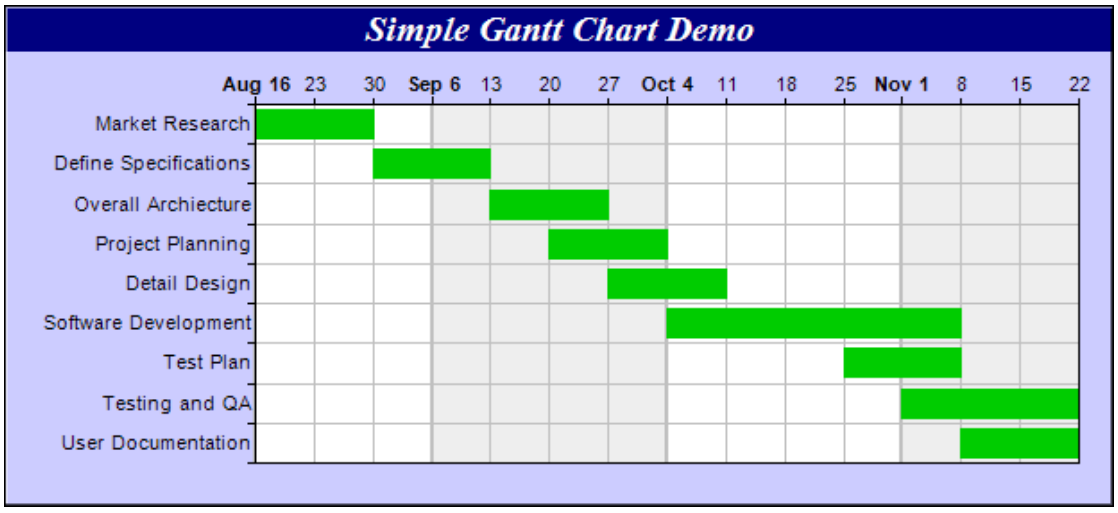


FIGURE 2.2: An example of An example of a Gantt Chart [4]

- A list of the tasks needed to complete a project,
- An estimate of the time that each activity will take to complete,
- The dependencies each task has,
- The deliverable for the project (milestones)

Using these values, CPM calculates the longest path of planned activities to the deliverable, and the earliest and latest that each task can start and finish without making the project longer. This determines which tasks are "Critical" and which ones have "total float" (i.e can be delayed without making the project longer). In project management, a critical path the longest overall duration of the added sequence of project tasks, regardless of if that longest duration has total float or not. This determines the shortest time possible to complete the project. [6]

In 1958 the United States Department of Defences US Navy Special Projects Office developed a similar method to the CPM called Program Evaluation Review Technique (PERT). In PERT three estimates are made, optimistic, expected, and pessimistic. PERT incorporates uncertainty by making it possible to plan a project without knowing all of the details. It is event-oriented technique, focused on the time needed to complete each task and identifying the minimum time required required to finish an entire project. [3] (See Fig:2.4). The key differences between PERT and CPM are: (See Fig:2.5)

- PERT is a project management technique, about planning uncertain, non-repetitive events. While CPM is a statistical technique of project management for planning repetitive, more predictable tasks,

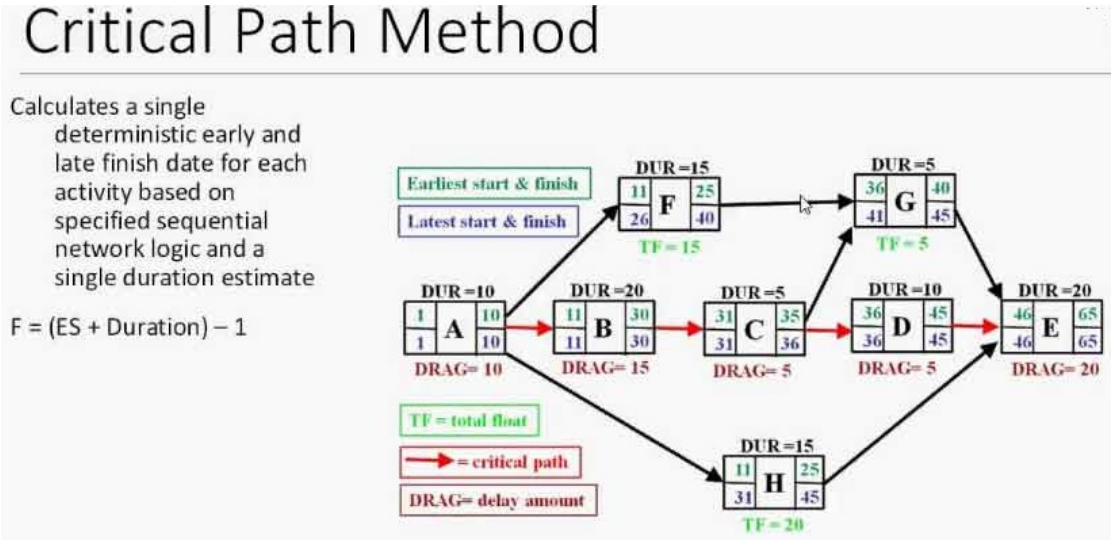


FIGURE 2.3: An example of the Critical Path Method[7]

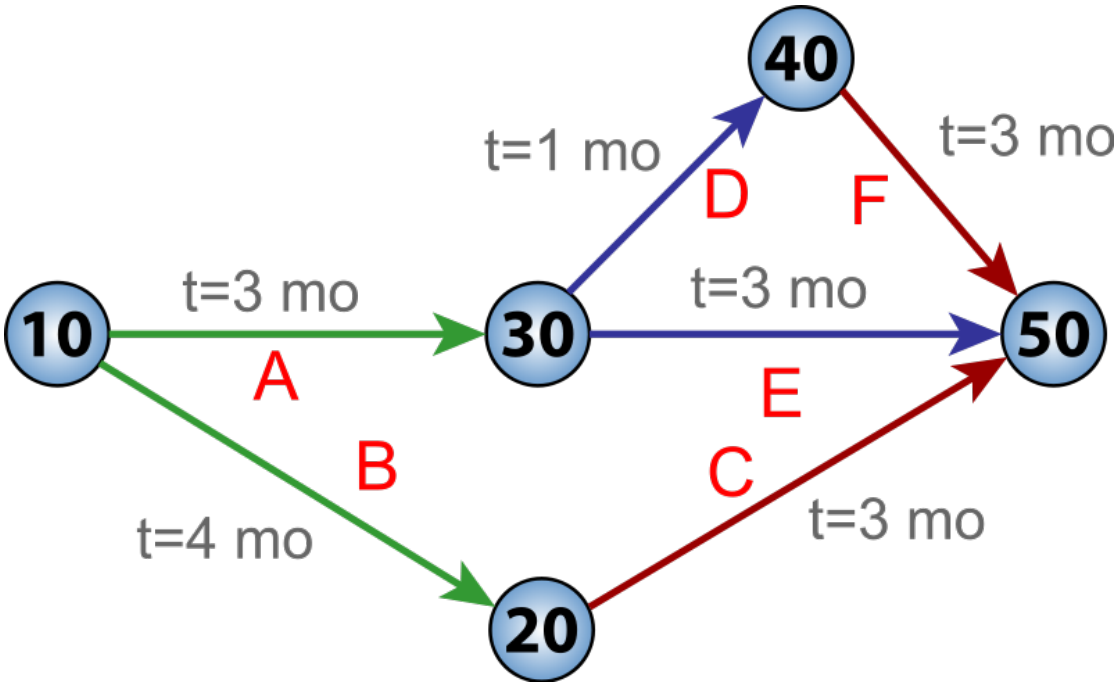


FIGURE 2.4: An example of Program Evaluation Review Technique[8]

- PERT is a technique of planning and control of time. Unlike CPM, which is a method to control costs and time.
- PERT takes three time estimations while CPM takes one [9].

That being said, PERT and CPM can be used as complementary tools, as "CPM employs one time estimation and one cost estimation for each activity; PERT may utilize three time estimates (optimistic, expected, and pessimistic) and no costs for each activity. Although these are distinct differences, the term PERT is applied increasingly to all

critical path scheduling." [9]

In 1986 Hirotaka Takeuchi and Ikujiro Nonaka introduced the term scrum in the context

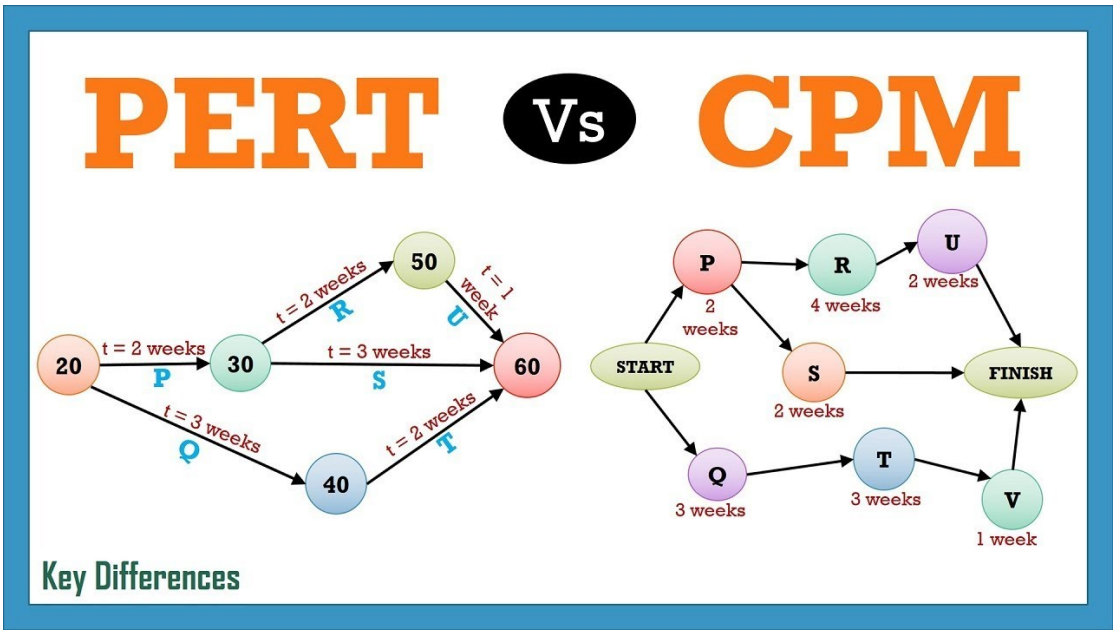


FIGURE 2.5: A comparison of PERT and CPM[10]

of product development in their 1986 Harvard Business Review article, "The New New Product Development Game". Scrum is an agile framework designed for teams of ten or less who brake work down into tasks that can be completed in timed iterations called sprints (these are typically two weeks). Progress is tracked using fifteen minute stand up meetings called "Daily scrums".[11] The key point of Scrum is to work iteratively and incrementally, with a strong focus on communication. A key principle of Scrum is recognising that there will be unpredictable challenges in creating a project which may change how the approach from what was planned, along with that customers will change their mind about what they want from a product (this is called requirements volatility).[12] While iterative and incremental development methods can be traced back as far at the 1950's Agile as a project style wasn't truly standardized until the creation of Agile Manifesto or the Software Development Manifesto, in 2001. Where seventeen people in the industry collectively decided that Agile valued:

- Individuals and Interactions over processes and tools.
- Working Software over comprehensive documentation.
- Customer Collaboration over contract negotiation.
- Responding to Change over following a plan.

Along with these the Agile Manifesto is based on twelve core principals:

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even in late development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicitythe art of maximizing the amount of work not doneis essential.
- Best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly. [13]

As Agile has become more popular more and more Project Management Tools (PMT) have appeared to help users plan, organise, and track the workflow of projects. In this section I shall compare Trello, Wrike and Zenhub. I have chosen Trello as it is a good example of a PMT that prioritises ease of use over a wide array of features, it also has a powerful API tool to allow for further modification. I have chosen Wrike as it is a good example of a tool that is much more feature rich but sacrifices some usability. I have disqualified similar PMTs such as Teamwork Projects as they are similar enough to Wrike in both Dashboard and Usability. Finally, I have chosen Zenhub as it is on the opposite end of the spectrum to Trello, instead focusing more on a wide depth of features at the sacrifice of even more usability. The headings I will be looking at are the following:

Introduction: Some general information on the PMT.

Dashboard: The goal of a dashboard in a PMT should allow understand the summarized project at a glance. It should also highlight tasks that are due soon/require

immediate action. In essence the dashboard exists to refresh a students memory and tell them what they should focus on currently.
Usability: A good PMT should be easy to use, should not require much effort to understand, and should allow a user to easily create projects and task. It should also allow users to easily add users to a project and assign members tasks. [14]
Pros and Cons: This will briefly explain what I believe the PMT does well and poorly.

Trello:
Introduction: Trello is a web-based Project Management Tool that focuses on Kanban-style list-making. It was created in 2011 by Fog Creek Software. It is incredibly lightweight and has strong API support in the form of add-ons called "Power-Ups", for this review I shall be looking at Trello without any add-ons. In the free version a user is allowed one Power-Up. Trellos simplicity has led it to be a popular choice for a PMT. The key to kanban is columns and cards. Each column should represent a specific activity that together compose a "work flow". Typically, in kanban the columns would be simple headings such as "To Do", "In Progress", and "Done". However one of the greatest features of kanban is its customization a user can create columns to suit any work style allowing a user re-purpose the columns into Sprints and work using Scrum. The other key feature to kanban is cards, which is how individual tasks are tracked. Cards should be broken down into single work items typically in kanban.[15]

Dashboard: The dashboard to Trello is simple to understand initially. From a glance users can easily see the labelled columns and cards for each task. There are certain pop-ups that appear on the cards as information is added (called card-badges by Trello) which are useful at a glance to summarize a task (for example it shows the number of attachments a task has or the number of comments if there is at least one). However it this can quickly become cluttered and could be overwhelming to a new user who's added to an existing board (See Fig:2.7). Each card-badge has tool-tips when a user hovers over them, explaining what they mean which can help understanding the project at a glance. Overall Trello's dashboard does a good job at summarizing the project at a glance. Trello is good at informing a user of upcoming events due to being able to set when a task is due. This will then appear on the card as a card-badge. Users can sort a column by these due dates (See Fig:2.8). However, users cannot set a start date which would help Scrum Masters see exactly how long it would take a task took.

Usability: Trello is incredibly easy to set up, on selecting a new board users are given a single column to rename, the ability to create new columns and a button to add cards to each created column. Cards can be made very quickly allowing swift project creation. Inviting people is also incredibly easy as there is an invite button always present on the

screen allowing users to add other users by email, Trello name or a shareable link. Many reviews of Trello praise its simplicity as one of its main appeals. To move cards between columns on Trello one drags the task into the next column. This has been described by Trello as a way to "gamify" the work flow to add further motivation to tasks, claiming that completing tasks will release dopamine (the chemical that motivates people) in the users brain.[16] The only thing that holds Trello back from a usability standpoint is its lacking feature-list. Features lacking from Trello that are commonly compained about include: Gantt Charts, Milestones, setting task priority, no good way to track reoccurring tasks, no calendar support initially, and no way to set dependencies or epic stories.[17][18][19] These lacking features lead many to claim that Trello should not be considered for larger projects.

Pros and Cons: See 2.1

Pros	Cons
Fast real-time updates	Feature lacking
Easy to set up	UI becomes cluttered on bigger projects
Intuitive and satisfying to use	
Powerful API usage allows further customisation	
Supports uploading documents directly to cards	

TABLE 2.1: Trello Pros and Cons

Conclusion: In conclusion Trello is a very user-friendly tool really only let down by a lacking feature list and a Cluttered User Interface (UI). The reason I chose Trello as the PMT to try and improve is due to its API tool it is possible to add features missing from Trello. This allows a solution to its prevalent issue.

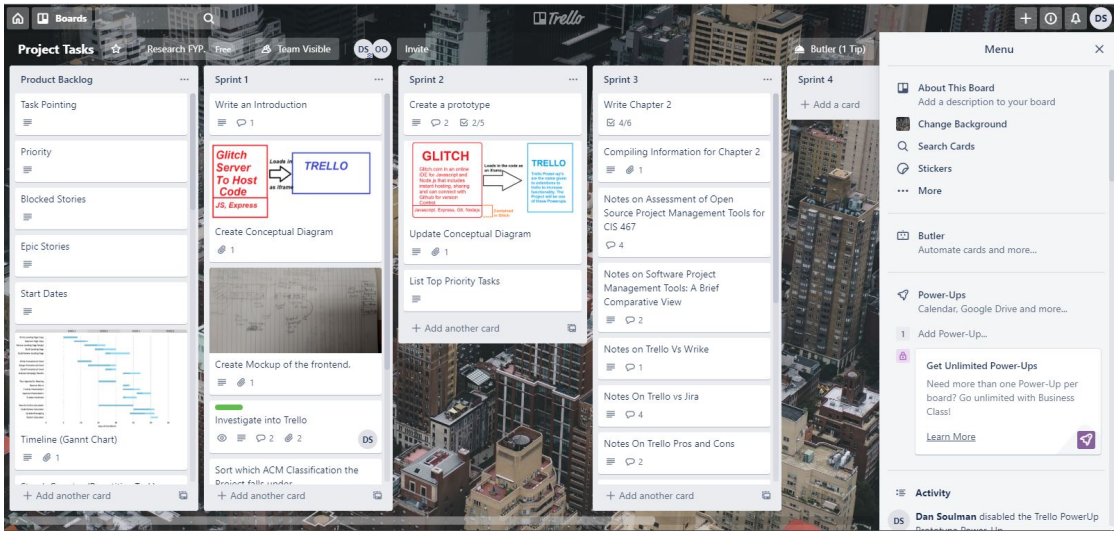


FIGURE 2.6: A Trello board

FIGURE 2.7: A Trello Card with multiple card-badges

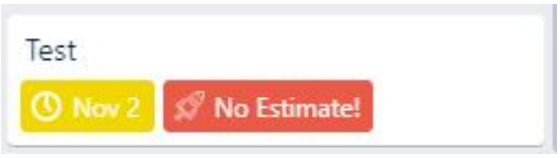


FIGURE 2.8: A Trello Card with a due date

Wrike:

Introduction: The Wrike PMT is a "SaaS-based, work management platform" (SaaS meaning Software-as-a-Service). It's was released in 2006 by the company Wrike.[20] Wrike is a powerful tool, and comes highly recommended by many people, winning a number of awards over the years[21].Wrike has student accounts for free needing only an .edu email account to register however this version is lacking in quite a lot of Wrikes features.

Dashboard: (See Fig:2.9) The Wrike dashboard is broken down into a few sections. To the left is a navigation bar displaying the projects, which can be broken down into folders. This is a very useful tool for sectioning off different kinds of tasks and is freely customizable, allowing customized names and sub-foldering. This allows a large degree of freedom to set up the project how a user wishes. Inside each of these sub folders a user can either display tasks on a list, kanban or table. The only issue to be found is that users cannot change the headings for the kanban columns. There is also a overall dashboard (See Fig:2.10) which allows a user to see tasks in a board view with customizable widgets. This is a premium feature though and requires a paid account to use. The dashboard is a incredibly effective way to summarize a project. As for informing users about due tasks, Wrike has a number of features for the user to choose from. One can set a starting and ending date for each task, there is a built in timer that allows users to track the exact amount of time a task took, there is a time-log that shows time-log entries for all tasks and sub-tasks within a particular Folder or Project.[22] There is also a Calendar feature allowing a user to see what days each task is due. However, all of these features besides start and end dates are locked away behind a paid account, meaning a free user is limited in what they could use.

Usability: Wrike can be rather tricky to set up compared to Trello. I struggled quite a bit setting up my student account initially. Navigating can be daunting for a new user

as there are many different possible views for the same information. This freedom is welcome once a user become comfortable with Wrike however. Wrike has an expansive help page, with interactive tutorials, video guides, and a community page to ask specific questions.[23] Adding a user is rather easy, simply adding there account from a list, however the "Add user" button is small is away from other buttons (its alone on the bottom right corner) which is difficult to find initially (See Fig circled in:2.9).

Pros and Cons: See 2.2

Pros	Cons
Premium version is feature rich.	Free version is heavily lacking features.
Very Secure [16]	Paid version is expensive
Plentiful helpful Documentation	Resource heavy
Nav bar makes moving between projects fast.	

TABLE 2.2: Wrike Pros and Cons

Conclusion: In conclusion Wrike is a very impressive PMT. The paid version is one of the most feature rich PMTs in the market, however it is expensive to use, to get the version with a the time tracking features is 24 dollars 80 cent per user per month which is just under 300 dollars yearly.[24] The free version of Wrike has about as feature rich as Trello but is much more resource heavily and is slower to load and update, however it does have access to folders and sub-folders. For that reason alone, I would suggest Wrike over Trello if your project is to be split up into different teams (for example a mobile team and a website team). However, for a smaller project the lack of a dashboard makes Trello much better for visualizing the project, Trello also loads much faster.

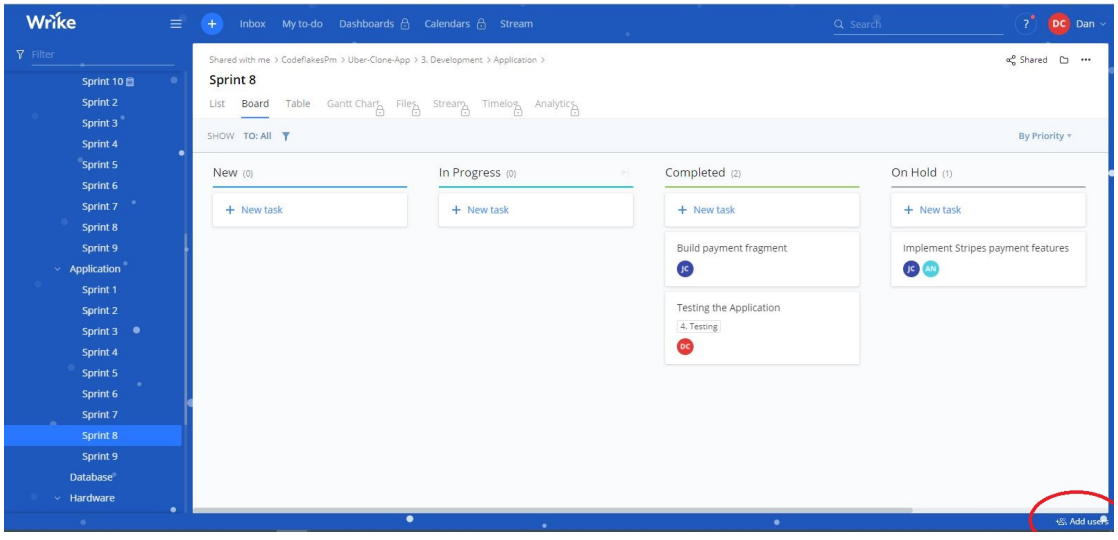


FIGURE 2.9: A Wrike board with add user circled for visibility

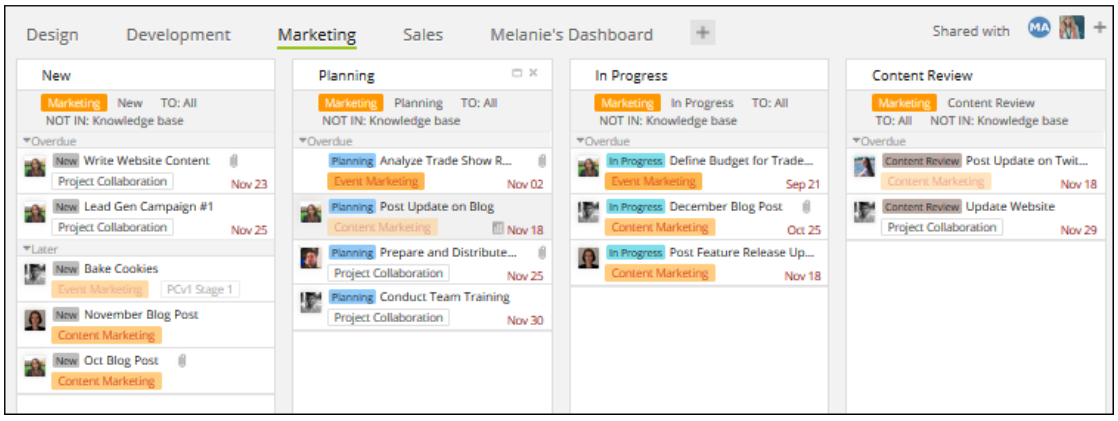


FIGURE 2.10: An example of a Wrike Dashboard

ZenHub:

Introduction: ZenHub is an extension built to integrate directly into GitHub. While other PMT's can add GitHub integration ZenHub is the only PMT that integrates natively into GitHub's UI allowing you to move issues through pipelines on a Kanban board. It was founded in 2014 with the goal of bringing project management closer to the code[25].For public repositories ZenHub is completely free. For Unlimited private repositories it is fifty dollars annually (approximately five times cheaper than Wrike).

Dashboard:(See Fig:2.11 for sample) Much like Trello, ZenHub uses a kanban system. Users can see all the tasks in relation to the code along with their progress with just one look at the board. Similar to Wrike you can pull up a navigation bar to do simple tasks such as inviting users. There are icons that sometimes appear on the cards though unlike Trello these don't have tool tips when tasks are hover on, meaning users must learn what these mean. Also like Trello you can add labels to a task which allows you to tell at a glance what kind of task it is and you can filter the board by label (See Fig:2.12). As for how it tracks time ZenHub does it entirely through milestones meaning there is no way to set a start or due date for individual tasks. In tracking time ZenHub is less effective than Trello or Wrike.

Usability: The biggest flaw ZenHub has is usability for new users. While initial setup is very simple, as you just create a workspace, create the columns you wish and then add tasks (called issues on ZenHub). Some of the more advanced features such as creating and using Milestones is not very intuitive. The navigation bar allows a user access tutorials. Similar to Wrike there is in depth set of documentation on the help page[26]. which is frequently updated with new tutorials. Adding new users to ZenHub is similar Trello or Wrike, as simple as navigating to the "Invite" button on the navigation bar and then inputting the email of the user you want, you can also add by username instead. Unique to ZenHub it will also make recommendations for who it thinks you will want

to add (See Fig:2.13 for sample).

Pros and Cons: See 2.3

Pros	Cons
Built in auto-generated reporting.	Steep learning curve.
Easy to reference tasks in other tasks	Need a repository to create a board
Easy to follow code progress as tied to Source Code	Tends to crash more than other PMT
	Lacks good way to track time.

TABLE 2.3: ZenHub Pros and Cons

Conclusion: ZenHub has a lot of the features of Wrike but gives access to all of them to both paid and free users. It is significantly more feature dense than Trello. However, it has what I believe to be the steepest learning curve of any PMT and I find that it is the less reliable than the other tools in that it appears to crash more at least in my testing of the tools for this review. I would recommend ZenHub to people who have a good knowledge of Project Management or the time to invest into learning how to use ZenHub. It is a powerful tool once you can use it effectively. It may be worth noting Zenhub does not support tracking individual task times, if that is an important factor to a project ZenHub is not the PMT I would recommend.

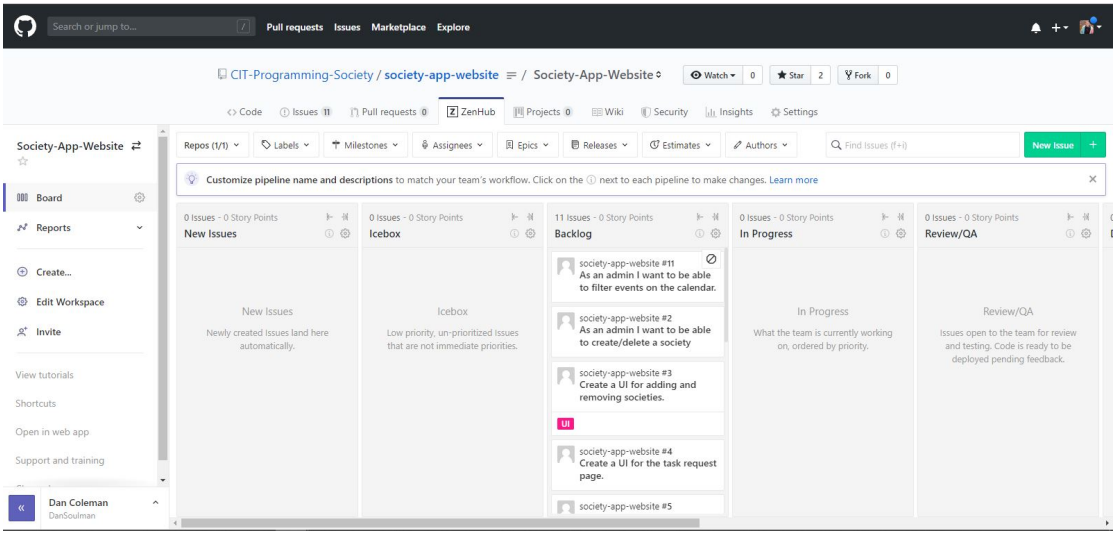


FIGURE 2.11: A ZenHub board

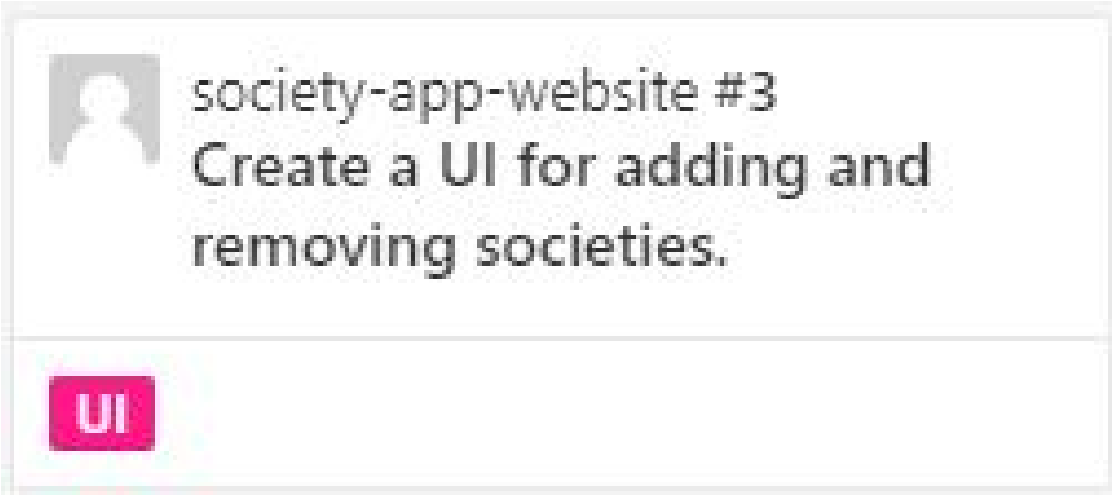


FIGURE 2.12: A ZenHub Card with Label

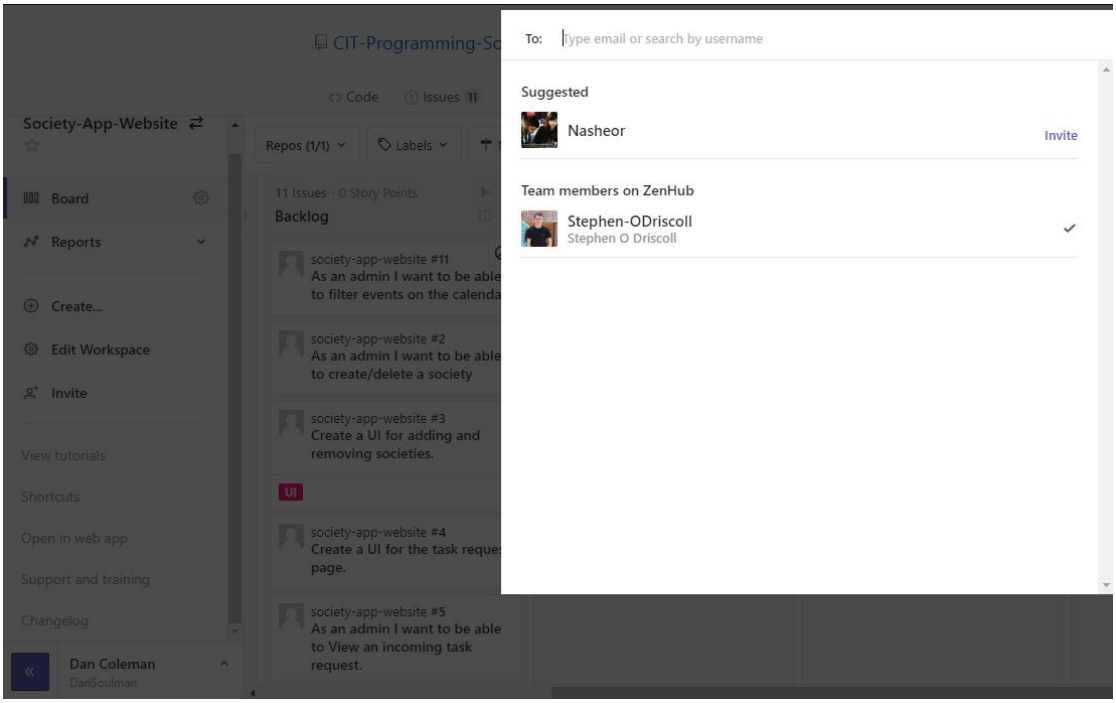


FIGURE 2.13: Invitation using ZenHub

Chapter 3

Problem - Increased Freedom in Project Management Tool Trello

3.1 Problem Definition

The problem I am solving is that while the PMT Trello is lightweight, fast and very beginner friendly making it a very accessible tool, it doesn't boast many features compared to other PMTs. Some of the features it's lacking as a free tool make it unsuitable in an agile environment. What it does, it does quite well but there are many ways to track, sort and visualise a project and Trello does not allow for the following:

- Repetitive Tasks: Agile project teams will often have a task that will be carried out more than once (for example running a Jenkins build).
- Priority: It is regular practice in agile to assign levels of priority to each task to help a scrum master sort the order that tasks should be completed in.
- Setting tasks as Epics/Blocked: In agile it is regular practice to show how tasks are related to other tasks, the two most common are Epic stories (big tasks that to complete require a number of smaller tasks) and Blocked stories (tasks that cannot be completed unless another task has already been completed).
- Timeline chart: Sometimes called Gantt charts, these are a graph showing the order that a task should be completed. These are incredibly useful for visualising how work should be done.
- Start dates: Agile project teams when assigning work often need to set start date for work activities to ensure clarity on expectation.

- Task Length Estimation: Sometimes called "Task Pointing", its a typical agile practice to make estimates on how long each task will take before hand to better estimate how much work can be done in each set sprint.
- A timer: For a better estimate of how long a task took, some agile teams use a timer to measure the length of each task.
- Starring Tasks and Comments: In Trello you can mark boards with a star to signify they are important, however you cannot do this with comments or tasks.
- Polling: Often in agile the team must meet to come to a decision how certain things will be implemented, adding this into the tool itself would allow the option to come to these decisions in a less intrusive way.

3.2 Objectives

The objectives of my project are as follows:

1. Have a functioning Trello Power-Up that expands on the list of features available to a user on Trello.
2. Allow for all of these features to be added in a single Trello Power-Up.
3. Retain the ease of use that makes Trello such an attractive tool.

3.3 Functional Requirements

Priority 1:

1. The Power-Up shall allow tasks to be marked repetitive.
2. The Power-Up shall produce the number of times a repetitive task has been completed.
3. The Power-Up shall allow the level of priority to be set for a task.
4. The Power-Up shall allow a column to be sorted by task priority inside the Sort list.
5. The Power-Up shall allow stories to be set as part of an Epic.
6. The Power-Up shall allow stories to be set as blocked by another.

7. The Power-Up shall produce a chart showing the order of tasks to be completed.
8. The Power-Up shall allow a start date to be set for a task.
9. The Power-Up shall allow input for an estimate for each task.

Priority 2:

1. The Power-Up shall allow tasks to be timed.
2. The Power-Up shall produce a warning for tasks that haven't been worked on after a set time.
3. The Power-Up shall allow a task to be "starred" (marked as important).
4. The Power-Up shall allow a comment to be "starred" (marked as important).
5. The Power-Up shall allow a polling system to allow a team to make decisions.
6. The Power-Up shall allow the number of sub-tasks on a card to be displayed as a percentage.
7. The Power-Up shall allow columns to be minimised.

3.4 Non-Functional Requirements

1. The Power-Up shall be easy to use for users who know very little about Project Management.
2. The Power-Up shall not reduce usability by introducing a cluttered dashboard.
3. The Power-Up UI shall be consistent with the existing Trello UI.
4. The Power-Up shall be simple to add or remove from a board for users who are not technically advanced.

Chapter 4

Implementation Approach

4.1 Architecture

The solution I have went with to implement the Power-Up is to host the work on Glitch.com and insert it into Trello. The Power-Up then runs on top of Trello to add the functionality into the pre-existing base. Glitch is an online web IDE for JavaScript, HTML and Node.js, it also includes instant hosting and automated deployment (see 4.1)[27]

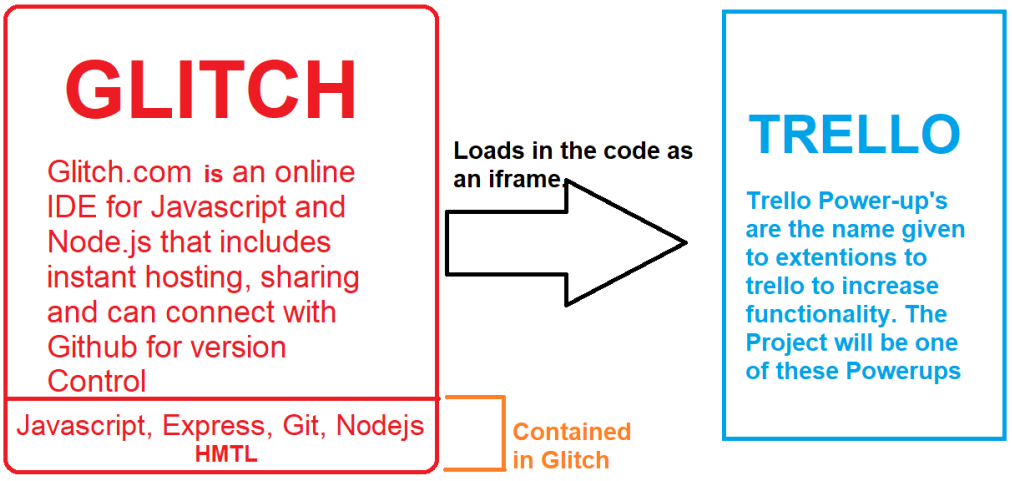


FIGURE 4.1: A basic diagram of the architecture

I chose Glitch for a few reasons:
Glitch (the company who makes Glitch) was originally called Fog Creek, who were also the company that made Trello. This leads me to believe that Glitch will be a good solution to host the Power-Up, a belief that is confirmed by Trellos Power-Up documentation which suggests glitch as the hosting service. [28]

The second reason I have chosen to use Glitch is that it runs on JavaScript and Node.js, these are the tools I used most often in my work placement.
The final reason I have chosen Glitch is that it makes hosting the site easy, to host the site we need only take the URL of the Glitch where my Power-Up is hosted and insert it into Trello's Power-Up section under Iframe Connector URL (see Fig:4.2).

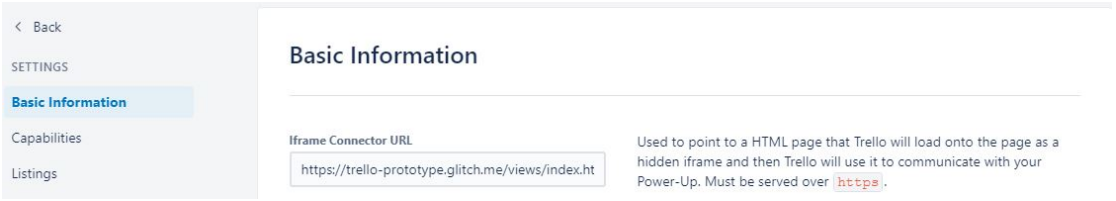


FIGURE 4.2: Inserting iframe URL into Power-Up info

I have decided to go with JavaScript as it is the language I used during work placement so I am somewhat familiar with it, along with it being the language used in Glitch which as explained above makes the Power-Up easier to implement. I will also be using Glitches built in git support for versioning.

4.2 Risk Assessment

There are a number of risks to my project. In this section I shall address each of the risks and what I could do to mitigate them: The first risk is the most fatal were it to occur. That being that as my project is based on Trello's framework and integrates with the tool if for some reason Trello or Glitch were to go down, my project would need to be overhauled massively. In my research section I looked into Trello and it is doing successful and has no reason to stop being a service, this is the same with Glitch as both are made by the same company and are rather largely used products. If it did, the only solution available would be to write a basic PMT myself to integrate my functions into which would still not solve my projects aim of improving Trello. This would be the death of my project. However the chance of it happening are about as likely as something Amazon Web Service going down, and I care about this topic enough to accept the risk. Another possible risk is that as an amateur programmer that there may be a feature I am not skilled enough to implement within the Power-Up tool, this error is more likely to occur but wouldn't necessarily end my project prematurely. The projects goal is to expand the features of Trello to allow it to be a better tool, while an impossible feature would weaken the quality of the Power-Up there are many other features still available to implement. The solution to an impossible feature would simply be to skip it and move to a more manageable feature. The final risk I can think of is similar in that I may run out of time to implement the project to the scope I am aiming

for in this project. This one is very likely as even with prototyping giving me a rough estimate of how much time features would take, I cannot predict the future. My skills of a programmer are average at best and I will work as best I can to implement the top priority list.

4.3 Methodology

As I have been using Trello, and am the only person working on this project I elected to use an the Agile methodology Kanban with the concept of sprints taken from Scrum. At the beginning of each two week sprint I set myself a number of tasks that I thought would be attainable to do around my college work and as I found spare time I would mark one as "To do" and focus on that task until it was complete. I often kept in mind the Kanban rule of limiting the number of pending requests (or in my case tasks I was working on) making the process more sensitive and revealing inefficiencies. Once I was happy with that task, I would mark it as complete and would select a new task from the list. I also used the Kanban method of having each task be a card. The online Kanban board I used was Trello (see Fig:4.3).

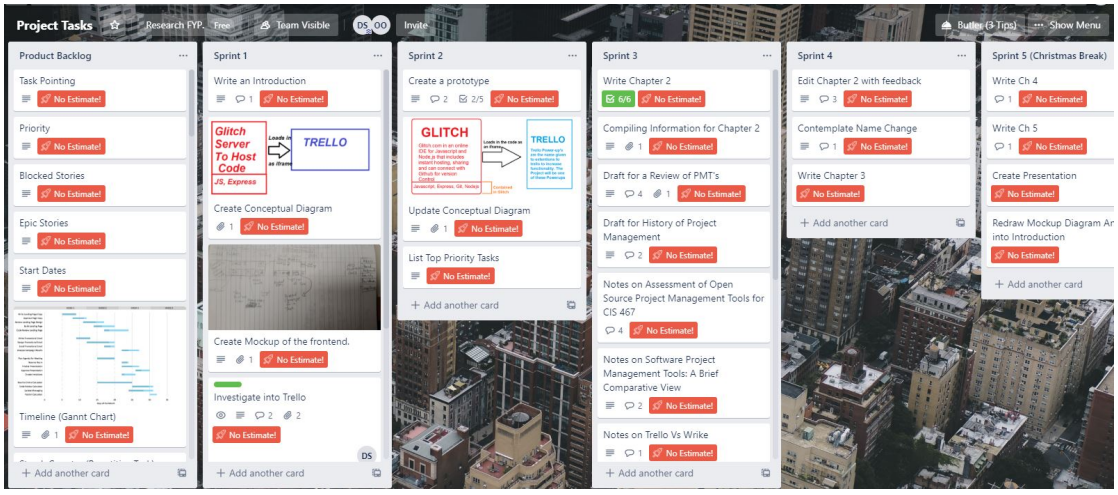


FIGURE 4.3: My Trello Board

To prepare for the research section there was two topics I felt I needed to cover for the background:

- The history behind Project Management to see how Project Management had gotten to where it had, and what had been tried but didn't work.
- A comparison of current online PMT's to allow me to see what was being done the same across all of them as well as what each felt made for an important PMT.

To prepare for the history section I created a Trello card and using Google Scholar looked for any papers I could on the topic, I would read up on these and take any notes I felt would be relevant to my topic and would write a draft of that paragraph along with saving the link to the paper so that referencing it would be easier later. As there weren't many papers on the topic I went to less academic sources to fill in any gaps in my knowledge I felt were missing in explaining the history and repeated the process of writing draft paragraphs. When this was complete I then combined all of these sources into a final draft where I tried to tie together the information into a chronological narrative to make the information easier to digest. This section was then sent to my supervisor for any feedback and with that feedback I wrote a final version. The history section of the paper took about a three days of work from start to final draft, and then an hour once I had my feedback to finalise.

To prepare for the comparison took much more time. Initially when I began my project to come up with features I wrote a list of features I thought would be nice to have in a PMT. I came up with 48 features of varying importance and I compared them against if Trello had them or not, upon further research I had learned of the Power-Up page and then also listed if the feature was currently available as a Power-Up currently to see if the feature was feasible. Then I decided I would compare three PMTs that were somewhat different to see if there were any features I had missed, what had been done well, and what could be improved. To prepare for this comparison, I accessed the three PMT's, built a small project on each and tested out what I could do on each. I took notes for each on these on Trello cards so that I could access these notes as needed from any device. Then I read a few papers that had previously compared PMT's and took note of the criteria they though was important and the points they had about each PMT and considered these. Finally I constructed my criteria for how I would compare the PMT's and I wrote my comparison. From research to my final draft took about a week, at which point I sent it to my supervisor for feedback. After receiving this feedback I spent one last day making edits with the feedback in mind.

4.4 Implementation Plan Schedule

The implementation schedule for the Power-Up is rather simple. Each Sprint I will select one feature from the top priority list and attempt to implement it, starting with more simple features. Early on as I am still wrapping my head around working with Trello I do not expect progress to be very fast. Once I have become better at working with Trello I expect the process to speed up and I will attempt to complete more than a feature a week. Being realistic with my programming abilities I do not intend to complete all of

the features on my top priority list but if I do I have a second list with lower priority tasks I can work on. I will choose these based off how quickly I can implement them.

4.5 Evaluation

As I add each feature I will consult the functional requirements to see that the implementation allows you to complete the function it is to fulfill. After that I will go through each non-functional requirement and ensure that the feature still adheres to each of them to ensure that the feature is of an acceptable standard along with achieving the goals the project is intended to achieve. Along with this there will be basic manual testing to ensure that there aren't any situations that the code breaks in. For each feature I will also include a criteria that is needed for it to be considered Done.

4.6 Prototype

As time was very limited throughout the semester I made a very basic prototype as a proof of concept. Initially I had planned my prototype to be one simple, completed feature (setting Priority), however despite following the documentation the prototype still wasn't working as intended, and with my time limited I made a simple button appear on the dashboard and a drop down in an individual card accessed with another button. Unfortunately it has yet to save the estimate selected and thus can't display it on the card button Creating the prototype even in this basic form was a massively beneficial, it showed that the Power-Up wasn't going to be as easy as I hoped to implement. This led me to massively scaling back the amount of work that I expect to complete (see images below)

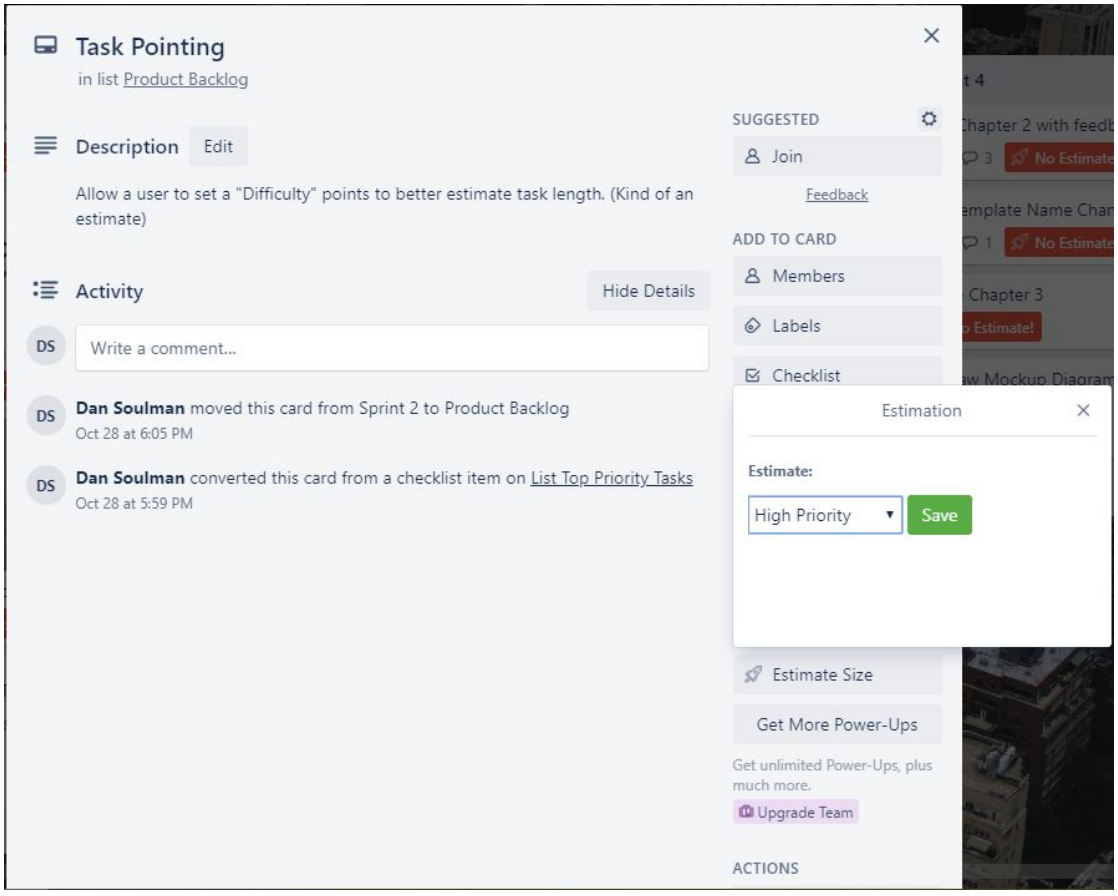


FIGURE 4.4: Estimate size button clicked on card

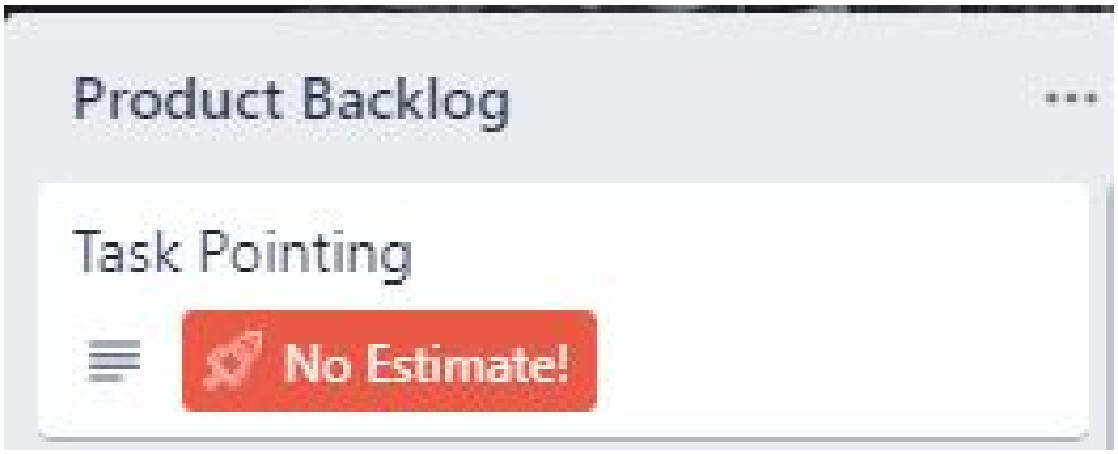


FIGURE 4.5: Estimate Card Button

Chapter 5

Implementation

5.1 Difficulties Encountered

Throughout the Project I encountered a number of difficulties as expected. As recommended by the draft for the thesis, I will break down the tasks into three categories: Enumerate the different difficulties you have found when developing your solution approach. Create three categories of difficulties:

- **Easy**: Tasks that were solved with little difficulty.
- **Medium**: Tasks that were not easy to solve, but that I managed to find solutions or workarounds that allowed me to achieve the functionality I had in mind.
- **Hard**: Tasks that were too difficult to solve. As a result of this, the solution to this problem did not achieve a functional requirement, non-functional requirement or use case it had been intended to achieve.

5.1.1 Easy Problems

Issue 1: Card States didn't save

1. Description of the issue: Initially in my Prototype I had run into an issue where upon selecting an option inside my priority window it would close and forget the selection, reverting back to the initial value.
2. How did it affect the original project design?: Had this issue persisted it would have been a massive issue, as without being able to save the options the feature would have been entirely useless. This would have meant that the projects goal

of increasing the feature count would have not been achieved, rendering the entire project a failure.

3. What did you do to solve the issue?: Having spent the winter break away from the code base due to burnout, I realised in minutes that the solution was caused by fixing the syntax in calling my script in the HTML file. This fixed the issue instantly.

Issue 2: Card detail badges didn't appear

1. Description of the issue: While working on Task Pointing, I realised that one of the places where information was to be displayed (called a card-detail-badge) wasn't appearing at all despite the code for it all being implemented. (See 5.1)

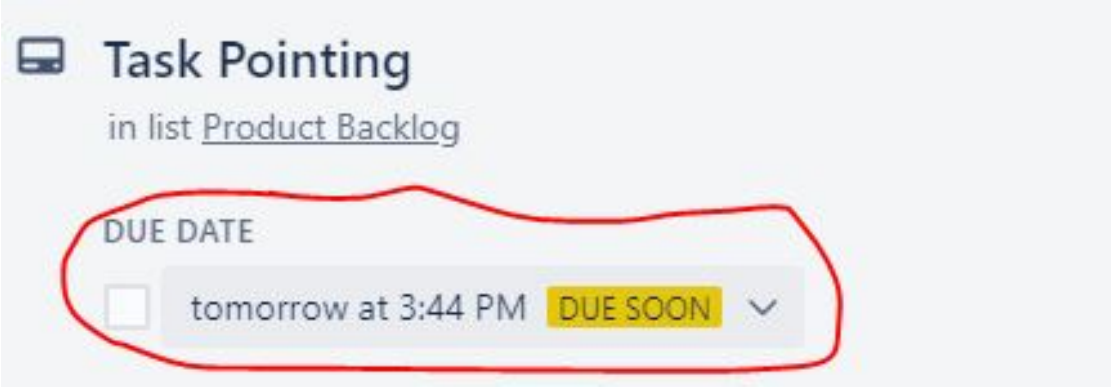


FIGURE 5.1: An example of a card detail badge

2. How did it affect the original project design?: Had this issue persisted it wouldn't have been incredibly detrimental to my final product, however it was a useful way to display information and the code was present so I took the time to fix it.

3. What did you do to solve the issue?: After a short debugging session I realised that I had disabled them in the power-up options erroneously, thankfully the console command pointed me straight to this issue.

Issue 3: The counter for Streak Counter saves the value passed in previously

1. Description of the issue: In the feature where I count repetitive tasks, or the "Streak Counter" I had an issue where if you saved the state of the counter it wouldn't save. Then when you selected a new value and saved again. It would save the number from the first attempt.

2. How did it affect the original project design?: This issue would of course be a disaster, as the feature would not work if this issue persisted. This issue caused the feature to carry on into the next Sprint as the first day I couldn't find out that the solution was.

3. What did you do to solve the issue?: After returning to my code the next day in my schedule to work on the project I realised my error in about half an hour. It turned out I had been accessing the information for the object incorrectly in parts of my Streak Counter window. Essentially, I had been creating a second variable that stored the old value and was then called back. After making sure all the lines of code called the same object the counter passed around the current value.

Issue 4: Creating Icons

1. Description of the issue: To make the project as presentable as possible I wanted to create custom icons for each of the features implemented. As this was in line with Trello which uses icons for it's features. However, the tool I typically use for the project Paint does not support transparent backgrounds. This left all my icons as a white square as opposed to the icons I would draw.

2. How did it affect the original project design?: This issue wasn't a massive issue for my project but I felt that having icons really would add to the initial presentation of the project. I wanted to ensure that the Power-Up left a good first impression with users.

3. What did you do to solve the issue?: To make sure I could use transparent icons I downloaded a free trial of Adobe Photoshop and spent an hour creating each of the icons. While I find Photoshop rather difficult to use, I am quite proud of the end result of my icons. They make the Power-Up look more in line with Trello's original design. (See 5.2 for original icon and 5.3 for new icon)

5.1.2 Medium Problems

Issue 1: Scaling the project past one feature

1. Description of the issue: For numerous weeks due to difficulty understanding how exactly Trello was storing information I couldn't have multiple card-badges display on a card at once.

FIGURE 5.2: The original rocket icon used as a placeholder

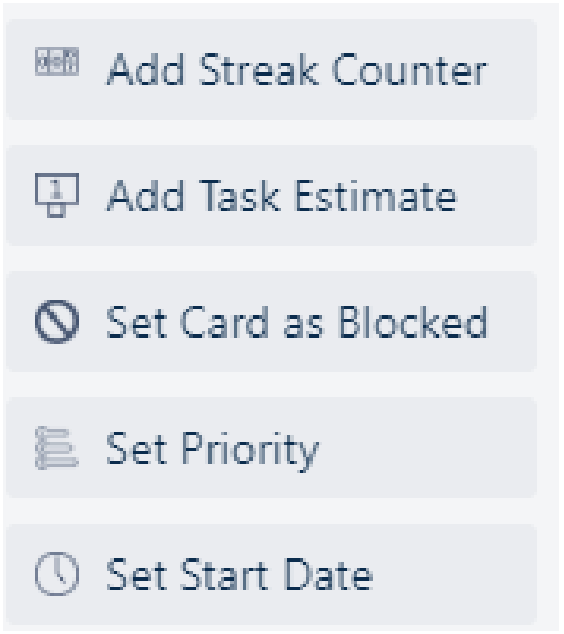


FIGURE 5.3: The new icons to represent each feature

2. How did it affect the original project design?: This affected the Project heavily as I could not find a method of displaying card-badges and card-detail-badges (icons to show information) with information from multiple different features at once. If the issue persisted, I wouldn't be able to integrate information into the Trello dashboard as seamlessly, which may reduce usability one of my core non-functional requirements. I couldn't allow the features to exist as separate Power-Ups as that would fail one of my core objectives for my Project. Solving the issue affected the schedule of implementation as I knew if I ignored it and continued working on features I would run back into this issue.

3. What did you do to solve the issue?: I worked for almost two full sprints searching online for any similar issues. There are very few people who work on Trello Power-ups as a hobby, so I managed to run out of online solutions rather quickly. I even created an account on Atlassian's Developer Community (which is the official forum for Trello development) to see if I could find a solution there but I couldn't find any help there either. Getting desperate I asked a few of my peers with more JavaScript experience could they find what I was doing wrong, and two of them were kind enough to spend a day sitting with me and trying different ideas, but it was to no avail and I wasn't willing to spend more time on a seemingly impossible task. Eventually while spending time looking at exactly how the data was being stored to solve a different problem, I realised I had not called the information from the card object correctly due to a misunderstanding on what the "shared" variable did. Upon realising this I went back once again to try and solve this issue and it worked.

Issue 2: Displaying the value of Streak Counters on opening the window

1. Description of the issue: While the Streak Counter keeps track of its value even after saving and closing the window, it will not display the current count until the value had been changed.

2. How did it affect the original project design?: Without being able to see this value, the feature is bothersome to use and could confuse new users. This would fail the non-functional requirement that it stays easy to use for all users.

3. What did you do to solve the issue?: By the time this issue had cropped up, I was already far behind how far I was supposed to be in the Sprint, and it seemed every feature went a few days over estimate. This was not my best work-around but as I couldn't spend any more time working on this relatively simple feature, I made the hard decision to just include a button that displayed the current value by taking 0 from the value which made it appear.

Issue 3: Trellos Documentation isn't comprehensive

1. Description of the issue: I found Trellos official documentation for Power-Ups often expected the user to fill in the blanks on concepts themselves. For example, under "Datetime pop-up" in the section "UI Functions" it shows an example of a date picker, and above the code sample is the caption "here is the code required to open the popup". Perhaps this is just me misinterpreting the caption but the code will not give you any of the UI objects required to recreate the date-picker in the image,

nor will it give you all of the variables available from the image. Trello Power-ups include custom CSS that make new creations fit Trellos UI, and so I don't think it would have been a huge leap in logic for them to include a completed sample of the date-picker UI, or to change the text to better clarify additional code was needed.

Another example is the tutorial "Building a Trello Power-Up".[29] The document is set up in multiple parts to walk you through the basics of creating a Power-Up, the most recent part, part three promises that they will cover building end-points and card badges (an especially useful feature in my case). Part three released in 2017 and there is still no sign of the promised part four. Leaving me to work it out from elsewhere.

2. How did it affect the original project design?: The blanks I am left to fill in while following the documentation have increased the time each feature takes to implement, which in turn decreases the number of features I am able to implement in the allotted time. This also meant that my Start Date feature was changed from a date-time picker into a more simple text box to ensure it was completed in time.

3. What did you do to solve the issue?: There was no solve for this issue, I felt it would discourage me from using Trello Power-Up's as a solution in the future.

5.1.3 Hard Problems

Issue 1: Time Constraints

1. Description of the difficulty: Due to the project taking more time than estimated to complete I did not manage to complete all the task from my top priority feature list. I had suspected that this was likely to happen as my original list was optimistic.

2. How did it affect the original project design?: A number of things were changed due to these time constraints. Most notably the exclusion of the Epic Story and Gantt Chart features. These features were the most difficult to implement and had to be cut early on to ensure that the other features would be completed to the standards I was happy with. While this means that not all of the functional requirements I had set for the project were met, I do not think that this means that my solution does not solve the problem. Many useful features were added to Trello that allow for a greater range of choice for users to track their projects. This means that the solution still solved the issue of limited choices in project tracking in Trello.

Issue 2: Sorting by Priority

1. Description of the difficulty: I had planned to allow users to sort the cards in a column by Priority, however the way Trello uses sorting in its documentation it takes the information from a list of cards as opposed to the information on the card stored in the board object. In the length of time I had allotted to implement this feature I could not get the sorting algorithm to access the custom data fields I had added to cards on the board object inside the card list. Due to this the priority of the card could not be used to sort.
2. How did it affect the original project design?: The functional requirement that cards could be sorted by priority was not met. As mentioned previously this didn't stop the problem from being solved, as there was still more freedom in setting Priority than in Trello's initial code base.

5.2 Actual Solution Approach

In this section I will cover if any changes have occurred between my initial plan and my current product and if so, why these changes occurred.

5.2.1 Architecture

My initial plan for the Architecture in summary was to host the Power-Up on Glitch.com and insert it into Trello as an iframe. The Power-Up would then run on top of Trello to add the functionality into the pre-existing base. Unlike many projects where different technologies might turn out to be a better choice to implement the solution to a problem. My goal has been to see how useful I could make my solution using the already existing option of Trello Power-Ups because of this goal, my architecture has not changed at all.

5.2.2 Risk Assessment

I had four possible risks in my initial assessment. Which were in order of least likely to most likely:

1. Trello would shut down.
2. Glitch would shut down.
3. I would not be a talented enough Programmer to implement a given feature.

4. I would not have enough time to implement each feature.

First, I will go through each risk and give an update on them. Then I shall also include any new risks that have appeared

Trello/Glitch will shut down

At the time of writing there seems to be no chance that Trello or Glitch will be shutting down. The company that purchased Trello, Atlassian has been doing well recently, with stock increasing since I began my project (See 5.4). While I couldn't find any stock information for Glitch given how closely the two work in tandem I can only assume that there is also no fear of them shutting down.

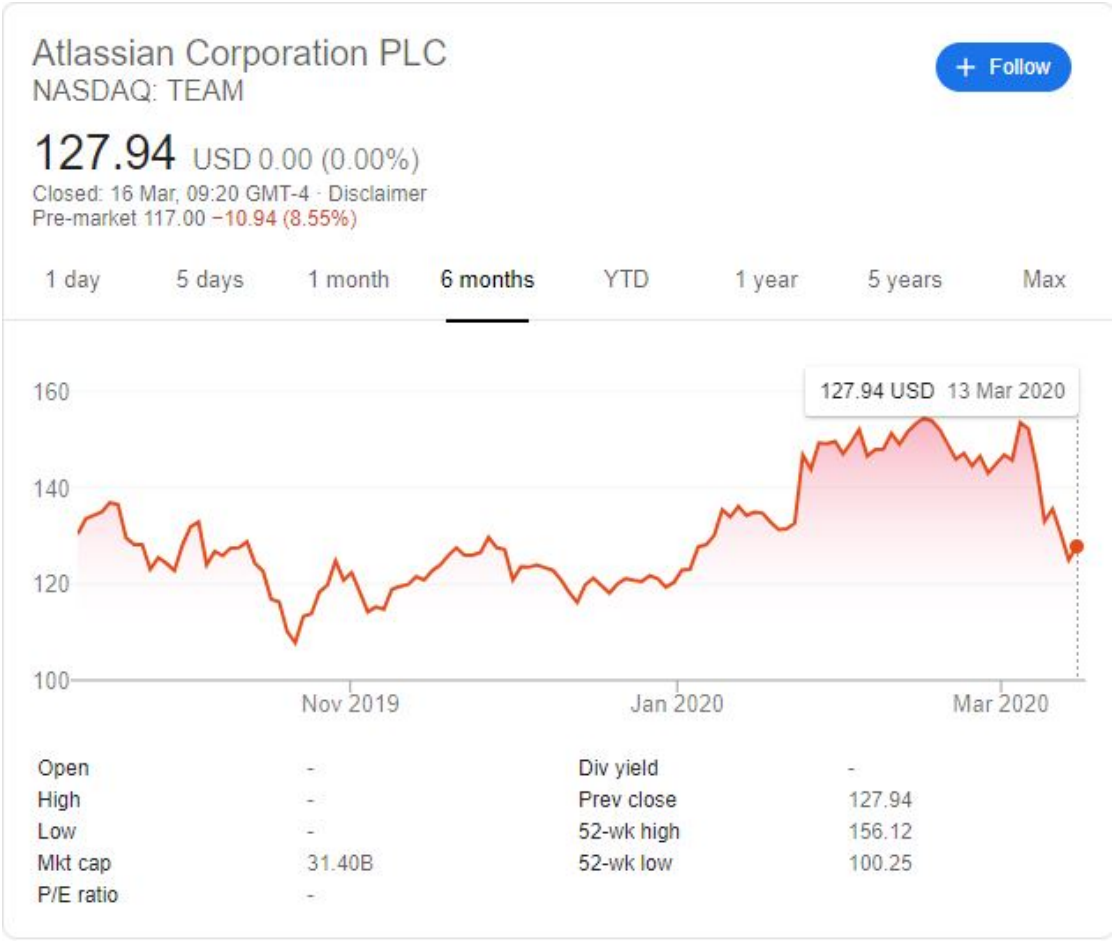


FIGURE 5.4: Image of the Atlassian stock as of March 13th

Not being talented enough to implement features/ Not having enough time

This issue does occasionally crop up. For example I had to cut my "Gantt Chart" feature very early on as the amount of time it would have taken to implement an updating graph showing relations between other tasks would have taken a massive amount of time and working around the Trello Library which does not natively support such a feature. To

implement this feature would have taken almost the entire implementation phase and so it was cut for time. Along with this my general inexperience working with Trello's libraries have meant that each feature has taken longer to complete than intended, which have put a few features at the very real risk of being incomplete. Time is my number one issue with this project as completing it around all of my other projects has left me with a very strict amount of time to work on features.

New Risk: Fighting against the Trello Power-Up Library

This is a risk that I should have realised might crop up, to suit Trellos business model the way that Trello's Power-Up library allows you to implement things is heavily favoured in making small, single function Power-Ups and adding many of them separately into Trello. Due to this, certain tasks such as displaying multiple card-badges took longer to implement than I had intended. More complex solutions I had planned such as an updating table for blocked stories have had to be reworked into more simple solutions due to the difficulty I had using the library.

5.2.3 Methodology

My initial plan for my Project Methodology was to use Kanban but with Sprints as the headers to help me more easily plan out my time. Then I would work on a single feature at a time until I was fully convinced that it was an acceptable standard before moving on. I have stuck to this as best I could although following in Agile methodology, I have made changes where needed to help optimise my workflow. I have stuck to two-week Sprints (the Scrum standard length) and typically try to focus on a single task. My supervisor Byron gave me the great idea of to have a small group of people use the product and give feedback to ensure usability. Due to this I no longer consider each feature fully completed other users test the product and ensure the features are at a passable quality. I tracked the project using Trello. This was especially helpful as it meant with every feature I completed I had access to that feature and could use it to help plan further. To ensure that I could manage completing course work and project work, I would dedicate two full days a week to the project (Monday and Friday) to the project, as I had no classes these days. The remaining weekdays would be dedicated to coursework.

5.2.4 Implementation Schedule

My initial plan for the Implementation Schedule was to select a simple feature initially and work on it while moving on to more complex features as time went on. I was hopeful that by the end I would complete more than a feature a week but knew this was unlikely

to actually happen. I've heard of very few projects that reach the schedule they hope for, as its natural for roadblocks to occur. My more realistic estimate was that I wouldn't finish Gantt Charts as that would be incredibly complex to implement and that it was possible more features wouldn't make the deadline. My current progress is more in line with my realistic estimate, with each feature taking about 3 weeks to implement. I am not worried as due to the nature of my project as long as one feature was added the range of options was increased, allowing greater flexibility in customizing Trello and the initial goal of optimising Trello for agile development was technically met. With each feature however the legitimacy of that claim is stronger, and I will continue to complete as many as I can.

5.2.5 Evaluation Plan

My initial plan of ensuring quality is to check each functional and non-functional requirement is met along with a small list of done criteria on a feature to feature basis. Along with this if possible, I want to have people come in and test my product and give any feedback they might have to ensure that the product is usable. While this is my intent, with the recent outbreak of Covid-19 and the current self-isolation in affect it will most likely have to be done digitally. If that is the case, I will likely conduct the survey over an online tool such as "Survey Monkey". I will do this with about 4 weeks left to the deadline to ensure that the project is as far along as I can but still ensures enough time to make some changes.

5.2.6 Prototype

During the time of writing my chapter four the Prototype was still not fully functional however within the first week of my implementation phase I had fixed the issue as covered in my Difficulties Encountered section. My prototype allowed users to set a Priority for each task. Since then the Power-Up has increased in scale to include a number of additional features. Due to the nature of Trello Power-Ups very little from my prototype went unused with the code base being used to create the full project as connecting it to Trello was already functioning as intended.

Chapter 6

Testing and Evaluation

In my project there were two forms of testing done on the features. The first was a set of arbitrary tests I conducted to ensure the quality of each feature, the second was that I conducted a survey with a small group of willing testers who gave feedback on each section that was completed at the time of surveying. With the results of both sets of test data I will go through each feature comment on the testing process and see if each requirement was met. I will then cover the non-functional requirements and give my evidence on why I believe these were met.

6.1 Metrics

6.1.1 Task Pointing

Tests I made to ensure quality:

- Test that the feature does not slow down Trello's use by more than one second: **Passed.**
- Test that adding number estimates doesn't break the dashboard or card badge displays: **Passed.**
- Test that adding text estimates doesn't break the dashboard or card badge displays: **Passed.**
- Test that adding a combination of text and numbers doesn't break the dashboard or card badge displays: **Passed.**
- Test that the pointing can be changed 5 times and still save correctly: **Passed.**

- Have a user test it to ensure that it is easy to use: **See Results Section.**

Relevant Requirements

- The Power-Up shall allow input for an estimate for each task: **Functional Requirement met**

6.1.2 Priority

Tests I made to ensure quality:

- Test that the feature does not slow down Trello's use by more than one second: **Passed.**
- Test that the Priority can be changed 5 times and still save correctly: **Passed.**
- Have a user test it to ensure that it is easy to use: **See Results Section.**

Relevant Requirements

- The Power-Up shall allow the level of priority to be set for a task: **Functional Requirement met**
- The Power-Up shall allow a column to be sorted by task priority inside the Sort list: **Functional Requirement not met. (See Results for explanation)**

6.1.3 Streak Counter

Tests I made to ensure quality:

- Test that the feature does not slow down Trello's use by more than one second: **Passed.**
- Test that the counter can be saved as positive and negative number without breaking the displays. **Passed.**
- Test that the counter can be changed 5 times and the number still saves correctly. **Passed.**
- Have a user test it to ensure that it is easy to use: **See Results Section.**

Relevant Requirements

- The Power-Up shall allow tasks to be marked repetitive: **Functional Requirement met**
- The Power-Up shall produce the number of times a repetitive task has been completed: **Functional Requirement met**

6.1.4 Start Date

Tests I made to ensure quality:

- Test that the feature does not slow down Trello's use by more than one second: **Passed.**
- Test that the incorrect format, month, day, or year have validation: **Passed.**
- Test that validation message appears instead of date for all possible displays: **Passed.**
- Test that leap years are correctly validated: **Passed.**
- Test that the Date can be changed 5 times and still show the correct data: **Passed.**
- Have a user test it to ensure that it is easy to use: **See Results Section.**

Relevant Requirements

- The Power-Up shall allow a start date to be set for a task: **Functional Requirement met.**

6.1.5 Blocked Stories

Tests I made to ensure quality:

- Test that the feature does not slow down Trello's use by more than one second: **Passed.**
- Test that the result can be changed 5 times and still show the correct data: **Passed.**
- Have a user test it to ensure that it is usable: **See Results Section.**

Relevant Requirements

- The Power-Up shall allow stories to be set as blocked by another: **Functional Requirement met**

6.1.6 Non Functional Requirements to be met across the Power-Up

1. The Power-Up shall be easy to use for users who know very little about Project Management.
2. The Power-Up shall not reduce usability by introducing a cluttered dashboard.
3. The Power-Up UI shall be consistent with the existing Trello UI.
4. The Power-Up shall be simple to add or remove from a board for users who are not technically advanced.

6.2 System Testing

6.2.1 Manual Testing

My manual testing method was to come up with multiple possible use cases of how each feature could be used whenever possible. Then I made sure that each of these possible use cases was accounted for in the functionality. For example, if a user had missed a repetitive task they may wish to turn a positive count into a negative count or vice versa, I would manually recreate these myself to ensure it was possible. While this was not the most in depth testing method, I wanted to place a focus on allowing multiple options for how users would make use of a feature. If the data could pass through Trello correctly, display each mention of the data correctly, could be changed multiple times while maintaining the information, and met each functional requirement, I typically considered the task to be functioning.

An instance where my validation was stricter was in the Start Date feature. While I aimed for freedom of use as the main focus for most tasks, I did make it a point to force users to input dates as MM/DD/YYYY. This was due to Trello using this format for its similar Due Date feature. Along with this I felt it was unlikely that a user would want to input a date that did not exist, so by having a standardised date format it allowed me to implement stricter validation. I followed a similar process of thinking of different ways a user might accidentally put in an incorrect date without realising, for example incorrectly thinking a year was a leap year, mixing up the day and month in the format,

or forgetting how many days were in a month. As I wanted the user to be aware of these possible errors, I wrote separate validation warnings for each error explaining where the user went wrong.

6.2.2 Survey Testing

In my survey group I took four computer system students who were familiar with project management and Agile processes, and I also asked my mother to test the project and give feedback from the standpoint of someone who is not familiar with computers and has no knowledge with project management. While the testing pool was not exceptionally large, I am grateful for the insight it gave me into the project.

To allow the users to test the project easily I created a Trello account with five identical boards and assigned each tester a board to allow them to test out the project as they saw fit. I then asked them to fill out a short survey asking if they were satisfied with each feature, which features would they use in the future in given the option and finally was the Power-Up easy to use. The scale I used to rate the features was dissatisfied to very satisfied.

- Dissatisfied which meant that a feature was not well implemented, and still needed work.
- Satisfied meant that the feature was implemented successfully and was considered usable.
- Very Satisfied meant that the feature was implemented especially well.

I also left a blank feedback section for testers to give any suggestions on how to improve the feature along with any bugs I may not have caught.

6.3 Results

6.3.1 Survey Results

In this section I will go through each question and give a brief overview of the results. For the first feature, task pointing 100 percent of testers were very satisfied with the feature. No bugs were caught in feedback and no suggestions on ways to improve it were mentioned. (See 6.1)

For the second feature, priority I once again had 100 percent very satisfied. There were no bugs or suggested improvements left as feedback for this feature. (See 6.2)

For the third feature, the repetitive task (or streak) counter I received 80 percent very satisfied with one user saying that they were satisfied but not very satisfied. Nobody was dissatisfied with the results of the feature. I received feedback on this feature. The first was the suggestion that I add a label that clarified that to increment the counter you must click the button multiple times. The other feedback was that while the feature was well implemented that they couldn't think of a situation where it would be useful. This feedback does not surprise me as I am aware repetitive task counting is a niche feature. (See 6.3 and 6.9)

For the fourth feature, the start date I received 100 percent very satisfied. The only feedback I received about this was that one of my testers caught a bug where I had not accounted for validation to check for negative numbers. This has since been rectified. (See 6.4)

For the fifth feature, the blocked stories I received 100 percent very satisfied. There were no bugs or suggested improvements left as feedback for this feature. (See 6.5)

I next asked users to mark which features they would use in future Trello Projects. I was happy to see that 100 percent of users would use Task Pointing, Priority, and Blocked Stories. Start Date got an 80 percent rate from users, with the niche Streak Counter feature having 40 percent. Considering that I did not expect every user to have a use for every feature, I am still very happy with these results. (See 6.6)

Next to ensure the project had maintained Trello's usability I asked if users found the Power-Up to be usable. I am proud to report that 100 percent of users agreed that the Power-Up was easy to use. (See 6.7)

Finally, I left a section for testers to leave feedback. I am happy to see that there was constructive feedback and that the overall impression was that people were impressed by the Power-Up. I ensured that their suggestions were implemented and that the bug mentioned in the testing ground was addressed. 6.8 and 6.9)

6.3.2 Were all Requirements Met?

6.3.2.1 Functional Requirement not met:

The Power-Up shall allow a column to be sorted by task priority inside the Sort list: This requirement was initially planned but was not implemented. This was

because the card data that was stored by board, which is how I accessed the data in all other instances was different to the card data stored in lists, which is how sorting is implemented in Trello Power-Ups. I had written a sorting algorithm, I could not find a way for the algorithm to access the information on the card's priority from a list. This meant that my algorithm could not use the priority to sort the cards. Due to this I had to cut the feature.

The Power-Up shall allow stories to be set as part of an Epic: This requirement was cut early in the Implementation phase as I know I would not be able to complete it to an acceptable standard in time. I would like to implement it in a later version of this project.

The Power-Up shall produce a chart showing the order of tasks to be completed. This requirement was also cut early in the Implementation phase as I know I would not be able to complete it to an acceptable standard in time.

6.3.2.2 Non Functional Requirements:

The Power-Up shall be easy to use for users who know very little about Project Management: As previously mentioned I asked my mother who has very little knowledge on Project Management or Agile methodology to go through the project and give any feedback on what she thought would help make the project more understandable. She mentioned that the small labels above each feature in their window had helped her understand the purpose of each feature. She also mentioned that the concept of clicking a button multiple times to increment and decrement the count was confusing and suggested a second label for Streak Counter clarifying this. With this feedback implemented I can say I am confident that my Power-Up meets this requirement. **The Power-Up shall not reduce usability by introducing a cluttered dashboard:** On the dashboard each feature in use adds an additional badge to the dashboard. Each has an icon to help differentiate badges. This had been the requirement I was most worried about but there was no feedback about it in the survey, and the general consensus when I had discussed it with fellow students when I was deciding if each feature should have a dashboard badge was that they were useful and the information was easy to follow. With all of this in mind I am confident my Power-Up meets this requirement. (See 6.10).

The Power-Up UI shall be consistent with the existing Trello UI: Trello Provide custom CSS for Power-Ups to ensure that the UI in a Power-Up is consistent with the existing Trello UI. Due to this as long as I included the CSS in each file it was very easy to meet this requirement. (See 6.11 and 6.12).

The Power-Up shall be simple to add or remove from a board for users who are not technically advanced: To add the Trello Power-Up in its current state is a little trickier than it would be if I were to publish the Power-Up for it to be available in the store. However, for this to be possible I would have to ensure a number of complex legal procedures and sign a lengthy contract with Trello holding me accountable for my Power-Up.

I was not comfortable signing a contract unassisted so instead to access my Power-Up a user must use the custom Power-Up admin portal. This process takes approximately ten minutes one time and is detailed step-by-step in the "Building A Trello Power-Up" documentation available online.[28] Once this has been done one only needs to insert the URL of my Power-Up **"https://trello-prototype.glitch.me/views/index.html"** into the iframe URL section, and the project is available in the Power-Up section as it would any other official Power-Up. (See 6.13).

Disabling the Project is the same as enabling it once it has been added through the admin panel. You just access the Power-Up settings in the side bar and select disable. Even if the first enabling is a little complicated it is well documented and takes very little time, overall, I am confident that this requirement is met. (See 6.14).

6.3.3 Results Conclusion

In closing I am very happy with my survey results. The general opinion of my project was very positive, all critique was constructive and massively helpful in the closing weeks to improve my project. I am very proud to see that many of the features would be used by 100 percent of users. It is unfortunate I could not reach all of my functional requirements, but I still believe that the solution I have created solves the problem it aimed to solve, so I am still confident in the final product. I am happy to see that every non-functional requirement was met.

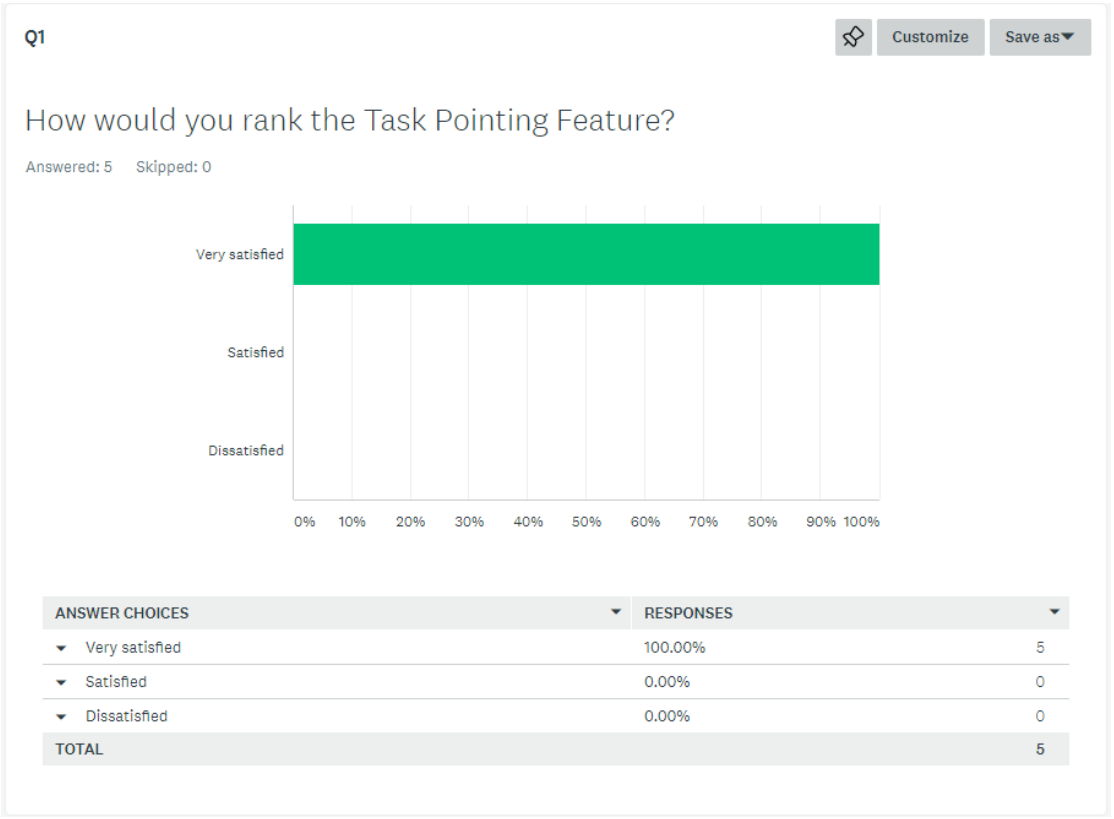


FIGURE 6.1: Task Pointing Survey Feedback

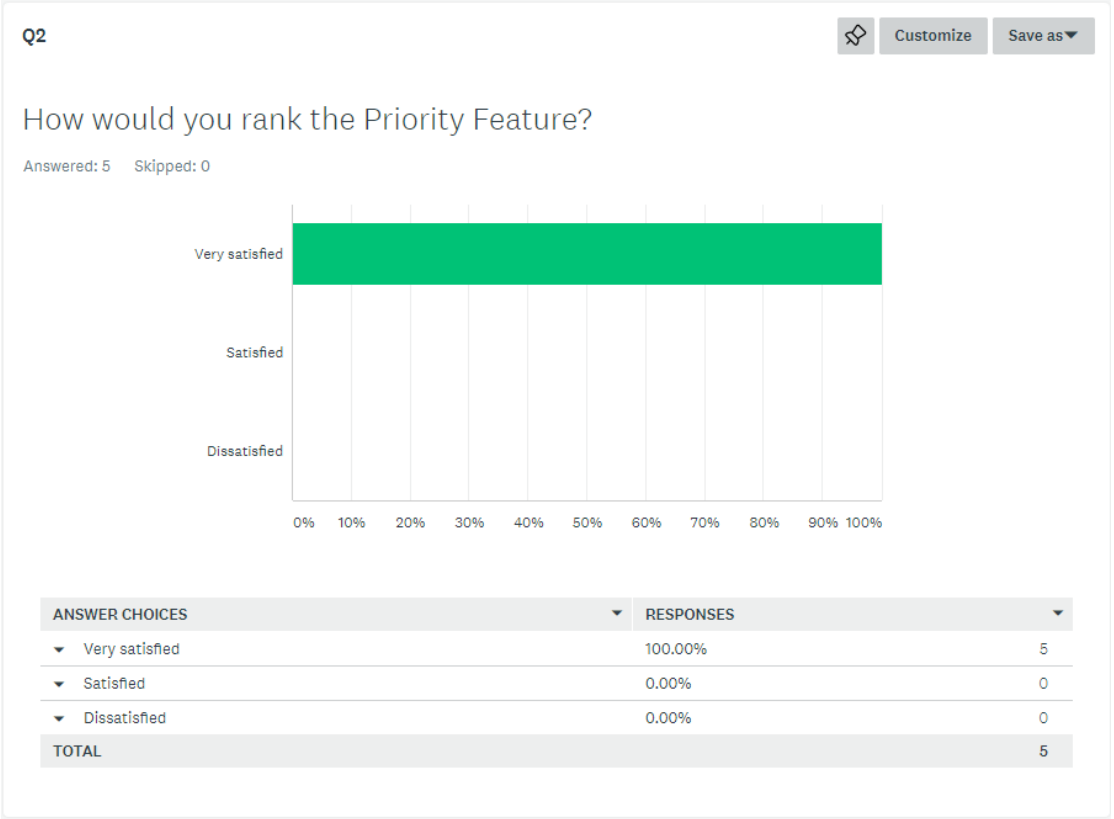
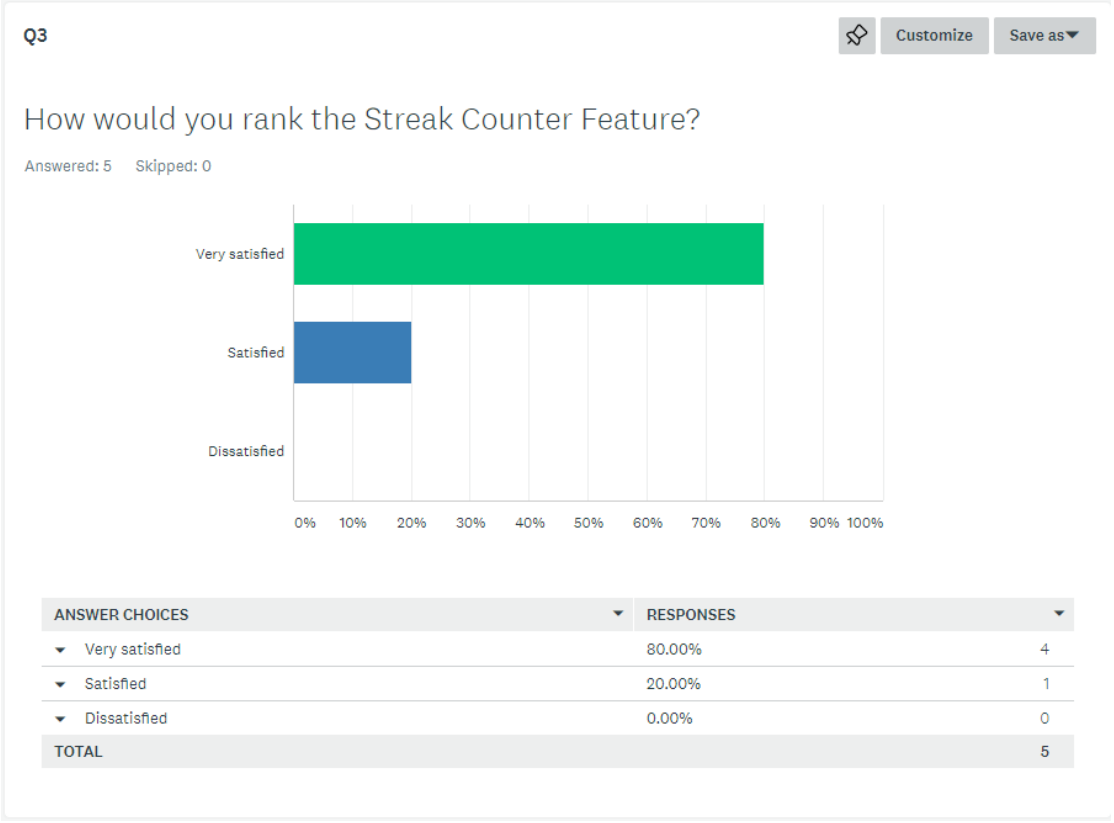
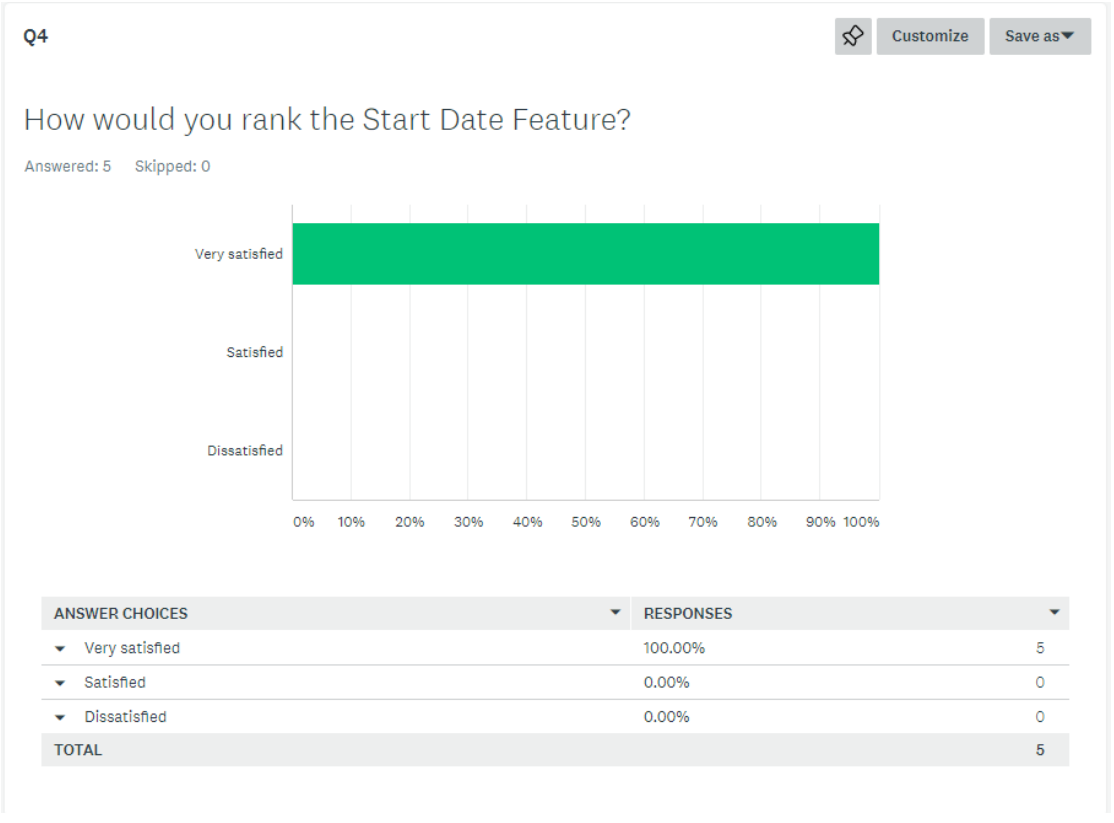


FIGURE 6.2: Priority Survey Feedback



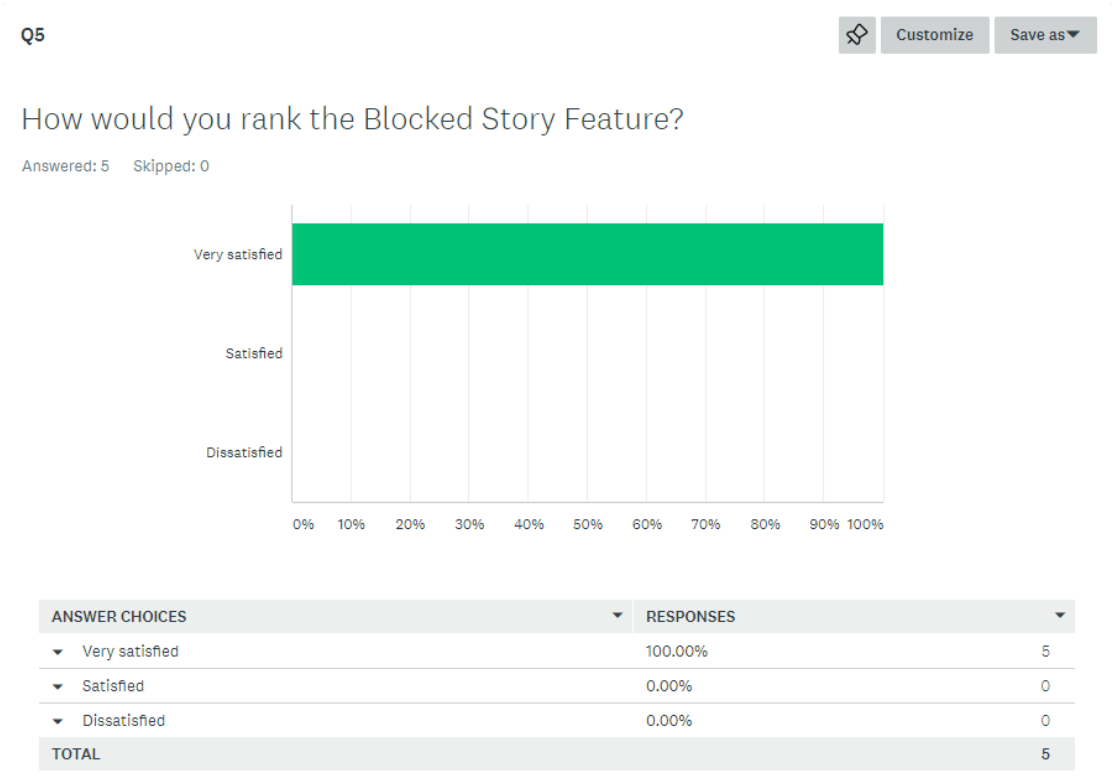
ANSWER CHOICES	RESPONSES	
Very satisfied	80.00%	4
Satisfied	20.00%	1
Dissatisfied	0.00%	0
TOTAL		5

FIGURE 6.3: Streak Counter Survey Feedback



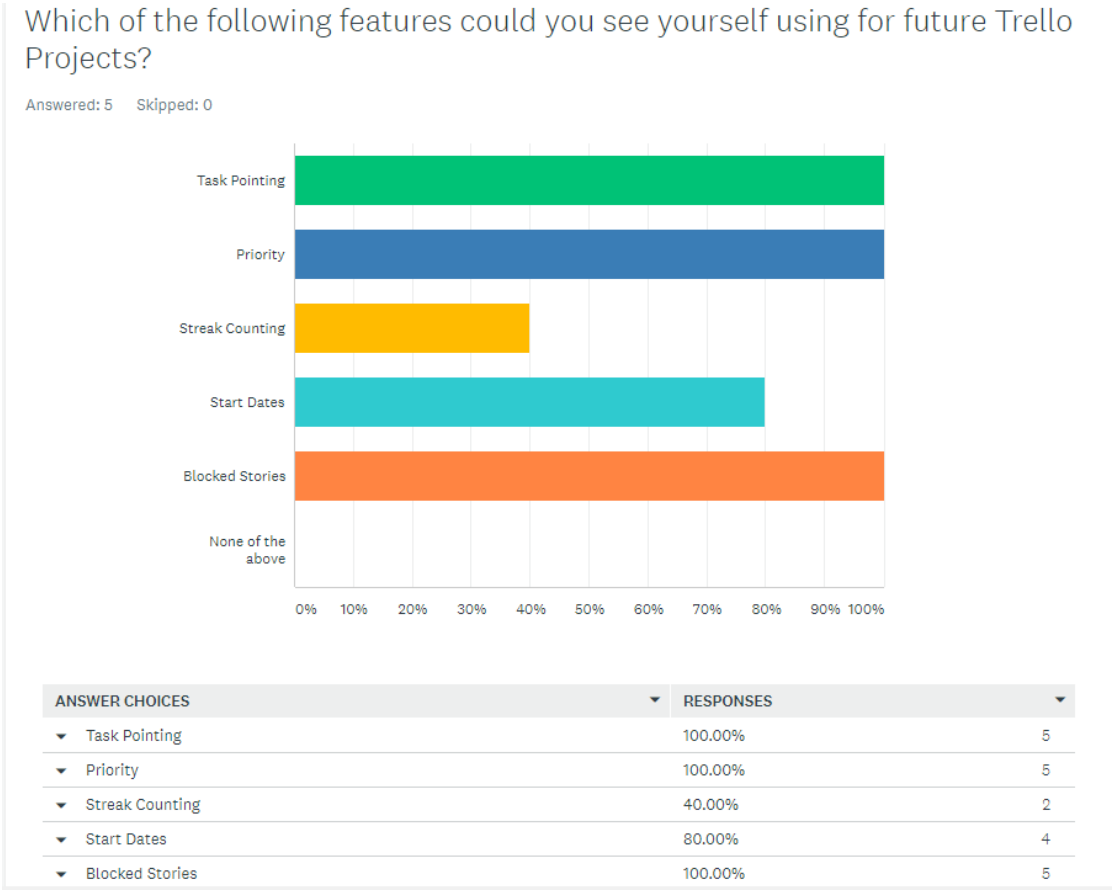
ANSWER CHOICES	RESPONSES	
Very satisfied	100.00%	5
Satisfied	0.00%	0
Dissatisfied	0.00%	0
TOTAL		5

FIGURE 6.4: Start Date Survey Feedback



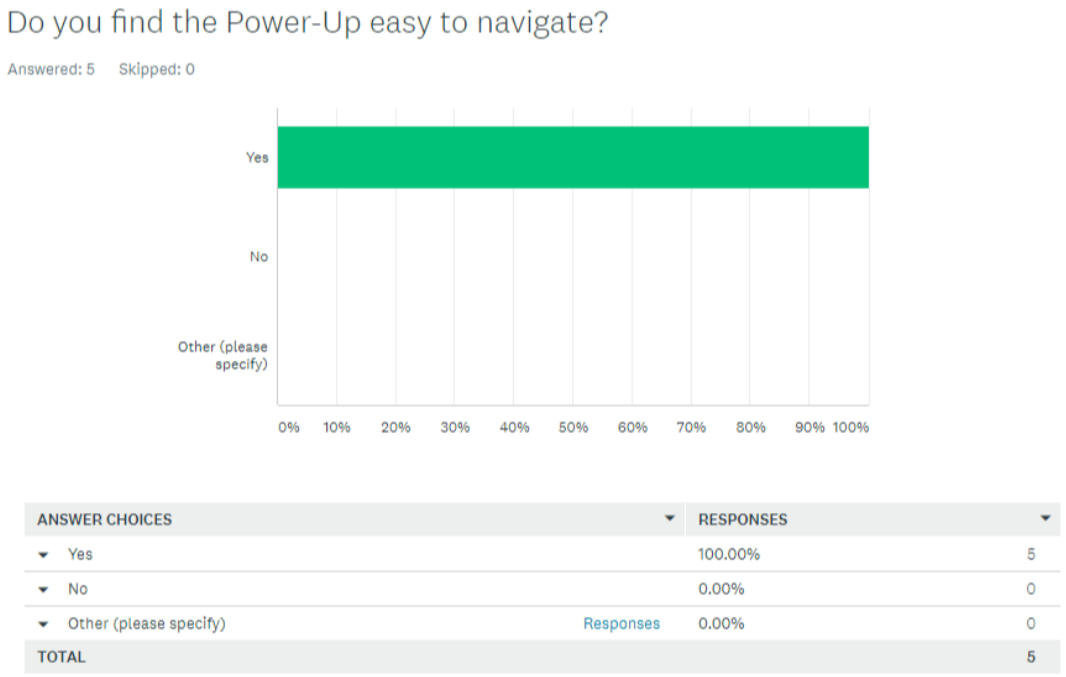
ANSWER CHOICES	RESPONSES	
Very satisfied	100.00%	5
Satisfied	0.00%	0
Dissatisfied	0.00%	0
TOTAL		5

FIGURE 6.5: Blocked Stories Survey Feedback



ANSWER CHOICES	RESPONSES	
Task Pointing	100.00%	5
Priority	100.00%	5
Streak Counting	40.00%	2
Start Dates	80.00%	4
Blocked Stories	100.00%	5

FIGURE 6.6: Features Users would use in the future Survey Feedback



ANSWER CHOICES		RESPONSES	
Yes		100.00%	5
No		0.00%	0
Other (please specify)	Responses	0.00%	0
TOTAL			5

FIGURE 6.7: Usability Survey Feedback

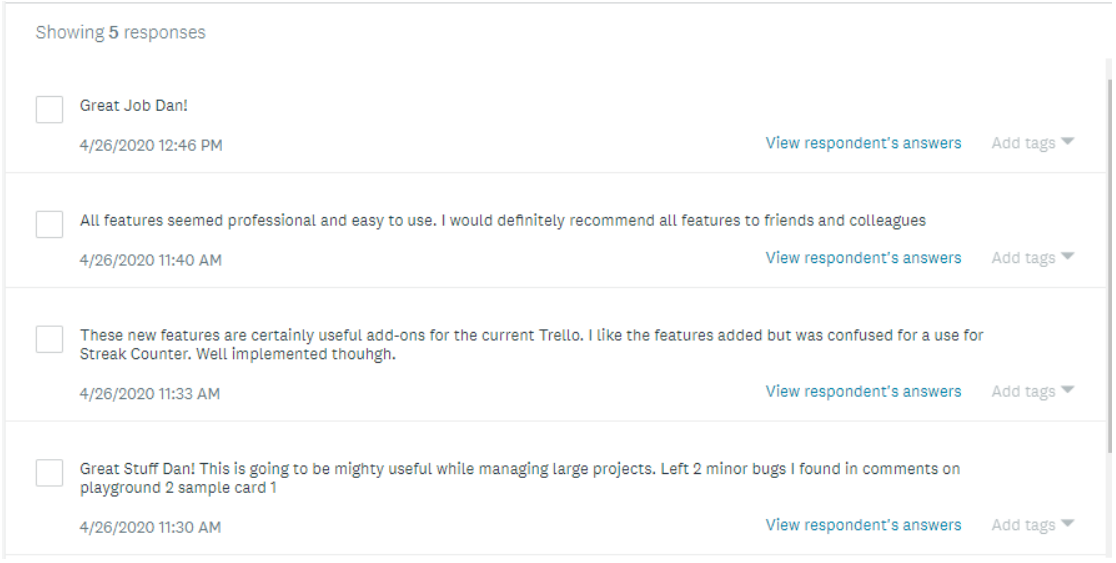


FIGURE 6.8: Additional Survey Feedback (1 of 2)

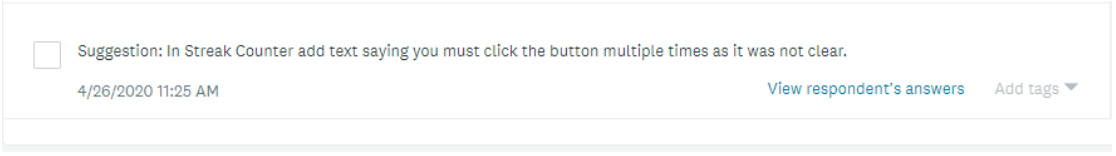


FIGURE 6.9: Additional Survey Feedback (2 of 2)

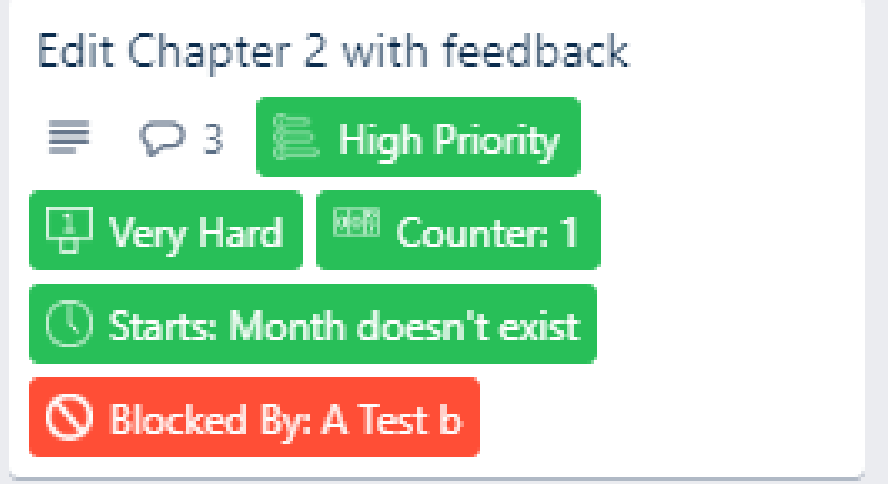


FIGURE 6.10: Image of the card with all badges visible at once

```
<head>
  <link rel="stylesheet" href="https://p.trellocdn.com/power-up.min.css" />
  <style>
    select {
      height: 30px;
    }
  </style>
```

FIGURE 6.11: Trello has it's own css style sheet provided

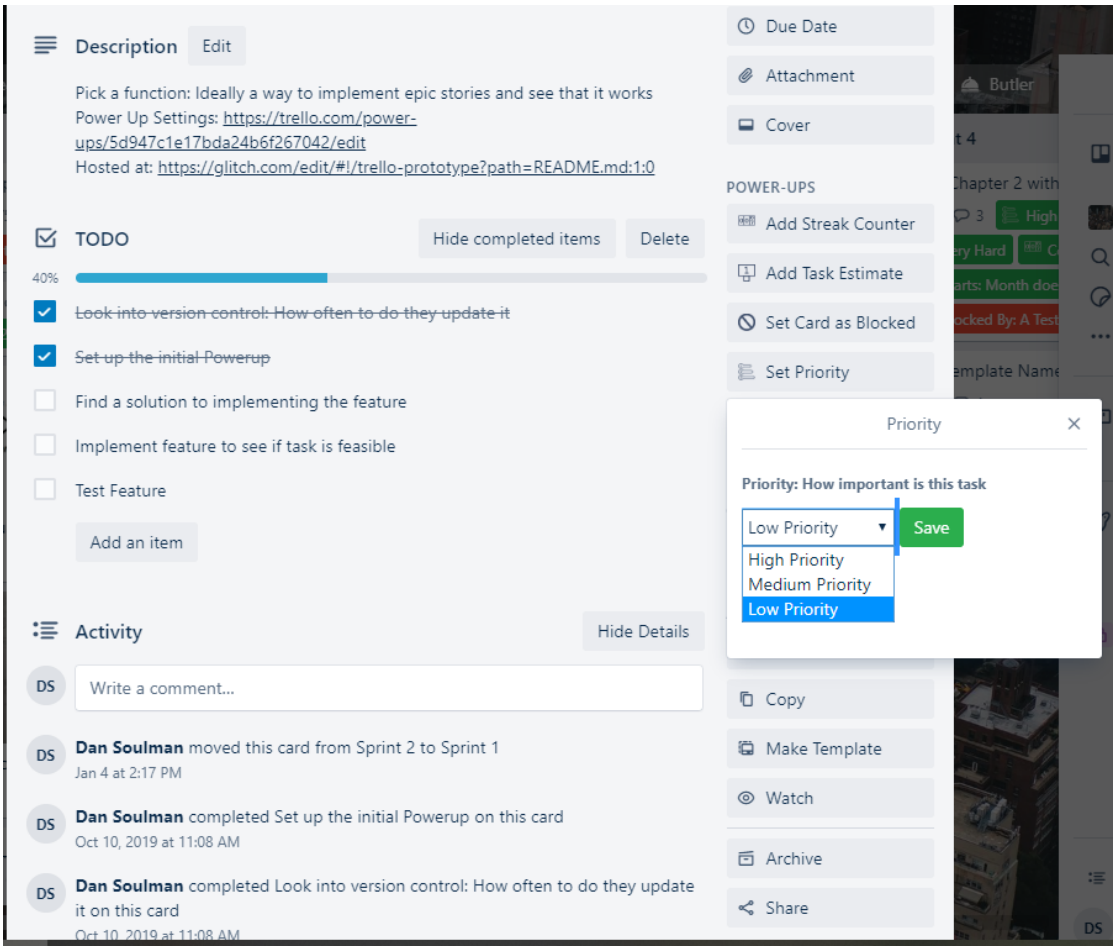


FIGURE 6.12: A sample of a Power-Up Feature using this style sheet

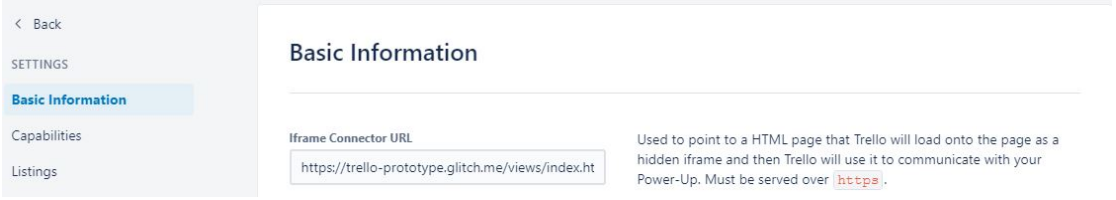


FIGURE 6.13: Trello Power-Up Admin Panel

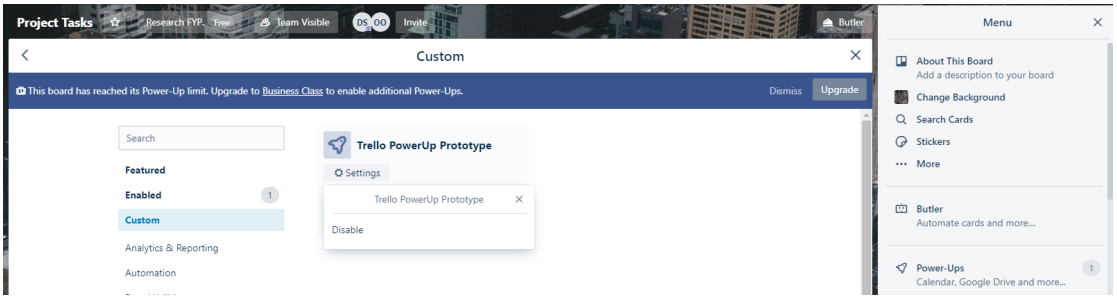


FIGURE 6.14: How to Disable/Enable the Power-Up

Chapter 7

Discussion and Conclusions

7.1 Solution Review

The problem that I set out to solve at the beginning of this project was that Trello is a feature light PMT. My research shows that what people wanted from Trello was additional features so that users could decide exactly how they wanted to track their project. Along with this I wanted Trello to better fit how an agile environment works, which is giving users and their teams the freedom to make minor adjustments to find out for them what works and does not work and as the project goes on to be able to use what they think worked while avoiding what did not work to allow for an efficient workflow. I also wished to ensure that the ease of use Trello offered was not hindered by my Power-Up as its ease of use makes it a fantastic option as a PMT, especially for those who do not have experience in those tools.

With my solution I have added five additional features to Trello, giving the tool the additional functionality that users wanted. By looking at the feedback I received from my survey, 100 percent of students said that if they had the option, they would use at least two of my five features. With 40 percent saying that they would use every feature if given the chance (see 6.6). This shows that my goal of adding additional features with a focus on useful agile methodology has been a massive success. All the feedback I received was positive and suggests that the solution was useful. Any constructive feedback was taken into account and implemented to ensure that users had the best possible experience.

While it is quite unfortunate that some of the functional requirements were cut, I believe this was the right call. The final solution still solves the problem, and the extra time I spent giving the other features lead to incredibly positive feedback on them.

Along with this I achieved every non-functional requirement I set out to achieve, I am very happy with doing this as ensuring the ease of use Trello has was a key component of the implementations. From the survey results we can see this ease of use was successfully maintained (see 6.7).

In conclusion I am very confident that my solution was successful, and the feedback from testers proves this.

7.2 Project Review

What I think I did well: I believe that despite the issues that I ran into, the project progressed quite well. I feel one of my strengths as a project manager is that I am aware of my programming skill level. I had estimated that I would initially struggle with working against Trello's code base to allow multiple features in a single Power-Up and planned my project accordingly. Due to this the project did not suffer when I ran into this exact issue. I was also aware that time is very often a disastrous issue when it comes to projects so I think it was a good call to choose a project where the solution was obtained early on, and any future work built upon the solution to make it a better solution. Finally, I feel that I made a good call on scaling the project down after the first sprint. It is a difficult decision to cut features but as I've stated previously the survey results for the features I did complete were very good. And I feel that rushing the project to complete another feature or two would have harmed the end product.

What I think could have been improved: I think that greater focus could have been placed on the design of the streak counter feature. At the time I was concerned that if my features looked too similar, I would be told my project did not look very difficult (which was feedback I had received from my research phase) and my grade would be affected. Due to this I went out of my way to make the streak counter more complicated to implement. This harmed the project as many people thought the streak counter feature was too complicated and harmed the ease of use. Along with this it made implementing the feature more difficult which took up more time that could have been spent elsewhere. I believe that I should have spent more time before the project brushing up on JavaScript. We had not done any JavaScript modules; I picked it up during placement. While I believe JavaScript was the right choice, I do think if I had prepared better the first few weeks of implementing would have been smoother.

Key Skills learned during the project: The first skill to mention is that my JavaScript and Node.js have improved quite a bit. I am much more comfortable with integrating JS into HTML along with accessing and manipulating existing data objects.

Node.js made working with the web browser much easier. In the future I could definitely see myself looking for a career in JavaScript as I enjoy working with it. If I were to write a similar project, I would definitely use both JS and Node.js again. Another skill worth mentioning is researching. Before this project I had never done researching to this scale or on this level of expertise. I really enjoyed researching both the history of project management as well as comparing and contrasting the different PMTs. If I was to do another project, researching is what I would look forward to most. A skill I struggled with before this project was technical speaking, however this skill was put in to practice numerous times throughout the project. The report, supervisor meetings, presentations and demos all forced me to spend more time learning technical terms and speaking confidently about them. This skill will hopefully not only help in future projects, but also in helping me achieve my goal of becoming a Scrum Master, as being confident in technical speaking is a key role for a Scrum Master. A minor skill that I am quite pleased with learning, was on how to better use Photoshop. I used it to create the icons used in the Power-Up and while they are not to a professional standard by any means I am proud of how they turned out. In future projects I would consider making any custom icons I need in Photoshop.

7.3 Conclusion

In conclusion, after comparing the free version of several different Project Management Tools available on the market I felt that Trello was the best example of a simple interface that is fast and easy to understand with no additional documentation. However the free version is limited in its number of features which hinders the freedom that agile is founded on, the aim of my solution was to write a "Power-Up" in JavaScript and HTML that included a number of the features that were commonly requested, allowing a user a greater degree of freedom to customise the project to what was most effective for them. I chose Trello as it had a user-friendly UI, making it easy to use and I felt that by adding these sought-after features Trello could become a more effective project management tool. The method I went about achieving this was by making use of Trello's "Power-Up" extension library and working on creating a single Power-Up (thus allowing the features to be used by users of any version of Trello) that maintained these ease of use Trello was a shining example of. Having completed my project and receiving feedback I can confidently say this solution was a success, with 100 percent of surveyed users stating that if given the chance they would make use of the Power-Up. I am incredibly proud of this project.

7.4 Future Work

If I had additional time, there are a number of small additions I would like to make. Including:

1. Allow users to add custom priorities to the drop-down list that can be set for Priorities.
2. Implementing a date picker in line with the one in Due Dates for Start Times.
3. Implement a table that displays all tasks that are blocking a task for Blocked Stores.

Along with this in a future version I would like to implement the following features:

1. Allowing a checklist of tasks to be duplicated as cards without removing them from the checklist. This would essentially allow you to create Epic Tasks.
2. Allowing a task to be timed and the timer to display how long a task has been worked on.
3. Allowing a user to star a comment or task (marked as important).
4. Allowing a user to create polls for comments.
5. Allowing columns to be minimised (hide the cards in that column).

Along with this I would like to redesign my Streak Counter with a more intuitive design. The survey feedback shows that it was the feature people were least impressed with (see 6.3). I would like it to be a feature, that even if most people do not have a use for, that all users are very satisfied with its design.

Bibliography

[1] Wrike, "What are project management tools?" 2019. [Online]. Available: https://www.wrike.com/project-management-guide/faq/what-are-project-management-tools/

[2] historicprojects.com, "The harmonogram," 2019. [Online]. Available: https://www.historicprojects.com/The_Harmonogram.html

[3] J. Westland, "History of project management," 2019. [Online]. Available: https://www.projectmanager.com/blog/history-project-management

[4] "Sample gantt chart image." [Online]. Available: https://www.advsofteng.com/doc/cdnetdoc/gantt.htm

[5] AACE, "About aace," 2019. [Online]. Available: https://www.aacei.org/about-aace

[6] P. Samuel L. Baker, "Critical path method (cpm)," 2004. [Online]. Available: https://web.archive.org/web/20100612142236/http://hspm.sph.sc.edu/COURSES/J716/CPM/CPM.html

[7] "Sample cpm image." [Online]. Available: http://www.businessstudynotes.com/wp-content/uploads/2017/09/Critical-Path-Method.jpg

[8] "Sample pert image." [Online]. Available: https://en.wikipedia.org/wiki/Program_evaluation_and_review_technique

[9] S. S, "Difference between pert and cpm," 2009. [Online]. Available: https://keydifferences.com/difference-between-pert-and-cpm.html

[10] "Pert cpm comparison image." [Online]. Available: http://i.ytimg.com/vi/dFTG3ohAcso/maxresdefault.jpg

[11] M. G. Software, "Daily scrum meeting." [Online]. Available: https://www.mountaingoatsoftware.com/agile/scrum/meetings/daily-scrum

[12] J. Henry and S. Henry, "Quantitative assessment of the software maintenance process and requirements volatility," February 1993. [Online]. Available: https://dl.acm.org/citation.cfm?id=170868

[13] A. v. B. A. C. W. C. M. F. J. G. J. H. A. H. R. J. J. K. B. M. R. C. M. S. M. K. S. J. S. D. T. Kent Beck, Mike Beedle, "Manifesto for agile software development," 2001. [Online]. Available: https://pdfs.semanticscholar.org/3eda/bb96a07765704f9c6a1a5542e39ac2df640c.pdf

[14] K. Almoqhim, "Assessment of open source project management tools for cis 467," 2016. [Online]. Available: http://scholarworks.gvsu.edu/cgi/viewcontent.cgi?article=1264&context=cistechlib

[15] M. Rehkopf, "What is a kanban board?" [Online]. Available: https://www.atlassian.com/agile/kanban/boards

[16] B. Brandall, "Why you get more done when you gamify your life." [Online]. Available: https://blog.trello.com/get-more-done-gamify-your-life

[17] N. Ceta, "Trello vs wrike the right tool for project management." [Online]. Available: https://tallyfy.com/trello-vs-wrike-project-management/

[18] J. C. A. Ambriz, "Trello vs. jira: Compared from a developers perspective." [Online]. Available: https://www.toptal.com/agile/trello-vs-jira-comparison

[19] K. Griffin, "The pros and cons of trello." [Online]. Available: http://www.projectmanagers.net/i/the-pros-and-cons-of-trello/

[20] Wrike, "About wrike." [Online]. Available: https://www.wrike.com/newsroom/content/uploads/2018/03/Wrike_Company_Description.pdf

[21] "Wrike: Recognition and awards." [Online]. Available: https://en.wikipedia.org/wiki/Wrike#Recognition_and_awards

[22] Wrike, "Timelog view." [Online]. Available: https://help.wrike.com/hc/en-us/articles/210323505-Timelog-View

[23] ——, "Wrike help website." [Online]. Available: https://help.wrike.com/hc/en-us

[24] ——, "Wrike pricing." [Online]. Available: https://www.wrike.com/price/?ga_campaign=(ROI)+Sitelinks&ga_adgroup=Price&targetID=kwd-12217672964&gclid=EAIaIQobChMIi5SDjuTL5QIVWeDtCh3sAQ27EAAYASABEgLduvD_BwE

[25] D. LaBelle, "What is zenhub? detailed zenhub overview and explanation of zenhub features," 2019. [Online]. Available: https://thedigitalprojectmanager.com/zenhub-overview/

[26] Zenhub, "Zenhub help website." [Online]. Available: https://help.zenhub.com/support/home

[27] Wikipedia, "Glitch (company)." [Online]. Available: https://en.wikipedia.org/wiki/Glitch_(company)

[28] B. Cook, "Building a trello power-up." [Online]. Available: https://tech.trello.com/power-up-tutorial-part-one/

[29] ——, "Building a trello power-up part 3." [Online]. Available: https://tech.trello.com/power-up-tutorial-part-three/

Appendix A

Code Snippets

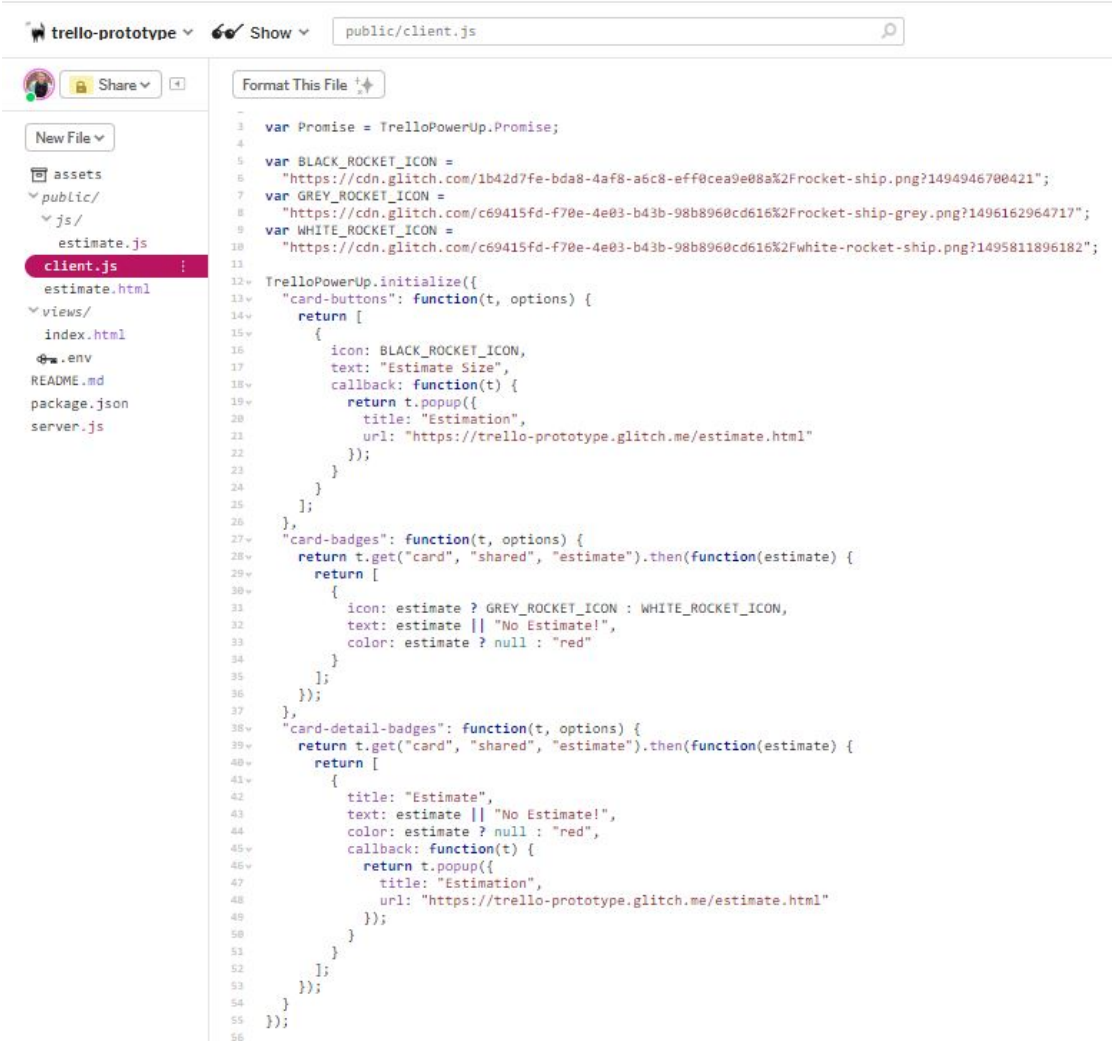


FIGURE A.1: Prototype: Client.js is used to initialize functions of Trellos Library

```
/*||---------||
  ||I C O N S||
  ||---------||*/
//Allow the program to access the images in the assets section.
var CLOCK_ICON =
  "https://cdn.glitch.com/1a623816-70a1-42fc-bf8b-fd43e9e3210b%2FClock%204.2.png?v=1586190859641";
var POINT_ICON =
  "https://cdn.glitch.com/1a623816-70a1-42fc-bf8b-fd43e9e3210b%2FTaskPoint.png?v=1586192665199";
var PRIORITY_ICON =
  "https://cdn.glitch.com/1a623816-70a1-42fc-bf8b-fd43e9e3210b%2Fpriority.png?v=1586192855045";
var COUNTER_ICON =
  "https://cdn.glitch.com/1a623816-70a1-42fc-bf8b-fd43e9e3210b%2FCounter.png?v=1586193416033";
var BLOCKED_ICON =
    "https://cdn.glitch.com/1a623816-70a1-42fc-bf8b-fd43e9e3210b%2FBlocked.png?v=1586797613534";
```

FIGURE A.2: Client.js: These Global Variables allow the code to load in the images for the Icons

```
TrelloPowerUp.initialize({  //Initialises the Power-Up features in use f
  /*||--------------------------------------||
    ||  C  A  R  D     B  U  T  T  O  N  S   ||
    ||Appear on the right hand side of cards ||
    ||--------------------------------------||*/
  "card-buttons": function(t, options) {
    return [
      {
        //Priority Button
        icon: PRIORITY_ICON,
        text: "Set Priority",
        callback: function(t) {
          return t.popup({
            title: "Priority",
            url: "https://trello-prototype.glitch.me/priority.html"
          });
        }
      },

      {
        //Task Estimate Button
        icon: POINT_ICON,
        text: "Add Task Estimate",
        callback: function(t) {
          return t.popup({
            title: "Task Estimate",
            url: "https://trello-prototype.glitch.me/taskpoint.html"
          });
        }
      },

      {
        //Start Date Button
        icon: CLOCK_ICON,
        text: "Set Start Date",
        callback: function(t) {
          return t.popup({
            title: "Start Date",
            url: "https://trello-prototype.glitch.me/startdate.html"
          });
        }
```

FIGURE A.3: Client.js: The Card Buttons allow users to open a popup with a given features html page

```
/*-------------------------------------------------/
/            C A R D   B A D G E S                /
/       The ones that appear on the dashboard     /
/-------------------------------------------------*/
"card-badges": function(t, options) {
  return t.getAll().then(function(data) {
    var card = data.card.shared; //Used for getting custom variables

    return [
      {
        icon: card.priority ? PRIORITY_ICON : null,
        text: card.priority || "",
        color: card.priority ? "green" : null
      },
      {
        icon: card.taskpoint ? POINT_ICON : null,
        text: card.taskpoint || "",
        color: card.taskpoint ? "green" : null
      },
      {
        icon: card.count ? COUNTER_ICON : null,
        text: card.count ? "Counter: " + card.count : "",
        color: card.count ? "green" : null
      },
      {
        icon: card.startdate ? CLOCK_ICON : null,
        text: card.startdate ? "Starts: " + card.startdate : "",
        color: card.startdate ? "green" : null
      },
      {
        icon: card.cardName ? BLOCKED_ICON : null,
        text: card.cardName ? "Blocked By: " + card.cardName : "",
        color: card.cardName ? "red" : null
      }
    ];
  });
},
```

FIGURE A.4: Client.js: Card Badges Display the any saved information from a feature on the dashboard

```
/*-----------------------------------------------|
|    C A R D   D E T A I L   B A D G E S   |
|   The ones that appear on the top of a card     |
|-----------------------------------------------*/

"card-detail-badges": function(t, options) {
  return t.getAll().then(function(data) {
    var card = data.card.shared; //Used for getting custom variables

    return [
      {
        title: "Priority",
        text: card.priority || "Set Priortiy?",
        color: card.priority ? null : "blue",
        callback: function(t) {
          return t.popup({
            title: "Priority",
            url: "https://trello-prototype.glitch.me/priority.html",
            target: "Trello Landing Page"
          });
        }
      },
      {
        title: "Task Estimate",
        text: card.taskpoint || "Set Task Estimate?",
        color: card.taskpoint ? null : "blue",
        callback: function(t) {
          return t.popup({
            title: "Task Point",
            url: "https://trello-prototype.glitch.me/taskpoint.html",
            target: "Trello Landing Page"
          });
        }
      },
```

FIGURE A.5: Client.js: Card Badges Display the any saved information from a feature on the top of a card

```
/* global TrelloPowerUp */
var t = TrelloPowerUp.iframe();

// Elements with IDs are available as properties of `window`.
window.startdate.addEventListener("submit", function(event) {
  // Stop the browser trying to submit the form itself.
  event.preventDefault();
  return t
  //This sets the start date either the correct date or the error message if it was invalid.
    .set("card", "shared", "startdate", validateDate(window.startDateTextfield.value))
    .then(function() {
      t.closePopup();
    });
});

t.render(function() {
  return t
    .get("card", "shared", "startdate")
    .then(function(startdate) {
      startdate = validateDate(startdate);
      window.startDateTextfield.value = startdate;
    })
    .then(function() {
      //Sizes window to allow all buttons to be seen at once
      t.sizeTo(document.body).done();
    });
});
```

FIGURE A.6: startdate.js: Deals with passing in the information and recalling it if the popup opens again

```
// Takes in the date and makes sure it's valid.
function validateDate(date) {
  var error = ""; //This accounts for any errors

  //Regex for format
  if (!/^\d{1,2}\/\d{1,2}\/\d{4}$/.test(date)) {
    error = "Invalid Date Format";
  }

  //Breaking up the date into month, day and year
  var split = date.split("/");
  // setting the split up date chunks to variables to make it easier to work with
  var month = split[0];
  var day = split[1];
  var year = parseInt(split[2], 10);

  //Deal with inputting zero
  if (day == "00" || month == "00") {
    error = "Invalid Date Format";
  }
  //Deal with the months
  if(month < 0 || month > 12){
    error = "Month doesn't exist";
  }
  //Deal with the no of days in a month
  if ( (month == "01" && day > 31) ||(month == "03" && day > 31) ||(month == "05" && day > 31) ||(month == "07" && day > 31) ||
       (month == "08" && day > 31) ||(month == "10" && day > 31) ||(month == "12" && day > 31) )
  {
    error = "Day " + day + " does not exist in Month " + month;
  }
  //30 days for April, July, Sept and November
  if ( (month == "04" && day > 30) ||(month == "06" && day > 30) ||(month == "09" && day > 30) ||(month == "11" && day > 30) )
  {
    error = "Day " + day + " does not exist in Month " + month;
  }
  //We need to account for leap years in february
  if (month == "02") {
    if ( ( year % 400 == 0 || (year % 100 != 0 && year % 4 == 0) ) ) {
      if (day > 29){
      error = "Day " + day + " does not exist in Feb " + year;
      }

      //Otherwise
    } else if (day > 28) {
      error = "Day " + day + " does not exist in Month " + month;
    }
  }

  //If the date passed validation returns inputted date otherwise it returns the error
  if (error == "") {
    return date;
  } else {
    date = error;
    return date;
  }
}
```

FIGURE A.7: startdate.js: Date Validation Function

```
<html>
  <head>
    <link rel="stylesheet" href="https://p.trellocdn.com/power-up.min.css" />
    <style>
      select {
        height: 30px;
      }
    </style>
    <script src="https://p.trellocdn.com/power-up.min.js"></script>
  </head>
  <body>
    <form id="startdate">
      <label for="startdate">
        Start Date: When will you begin this task (format is MM/DD/YYYY)
      </label>
      <input id="startDateTextfield" type="text" />
      <button type="submit" class="mod-primary">Save</button>
    </form>
    <script src="./js/startdate.js"></script>
  </body>
</html>
```

FIGURE A.8: startdate.html: Pop-up that allows the user to enter the date