An Investigation on the Integration of Work Breakdown Structure in Project Management of GitHub Projects

Dan Walsh
Student ID: 7661525
Supervisor: Amanda Brooks
Course: Computing BSc (Hons)
Date of Submission: April 2020
Faculty of Engineering, Environment and Computing, Coventry University

300COM / 303COM Declaration of originality

I Declare that This project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet etc.) has been acknowledged by citation within the main report to an item in the References or Bibliography lists. I also agree that an electronic copy of this project may be stored and used for the purposes of plagiarism prevention and detection.

Statement of copyright

I acknowledge that the copyright of this project report, and any product developed as part of the project, belong to Coventry University. Support, including funding, is available to commercialise products and services developed by staff and students. Any revenue that is generated is split with the inventor/s of the product or service. For further information please see www.coventry.ac.uk/ipr or contact ipr@coventry.ac.uk.

Statement of ethical engagement

I declare that a proposal for this project has been submitted to the Coventry University ethics monitoring website (https://ethics.coventry.ac.uk/) and that the application number is listed below (Note: Projects without an ethical application number will be rejected for marking)

Signed:  Date: **24/04/2020**

Please complete all fields.

First Name:	**Daniel**
Last Name:	**Walsh**
Student ID number	**7661525**
Ethics Application Number	**P102592**
1st Supervisor Name	**Amanda Brooks**
2nd Supervisor Name	**TBC**

This form must be completed and included with your project submission to Turnitin. Failure to append these declarations may result in your project being rejected for marking.

Contents

- ❖ Statement of Originality 1
- ❖ Abstract 4

1. Introduction 5

1.1. Project Aim 5
1.2. Project Objectives 5
1.3. Background and Context 5
1.4. Motivation 5
1.5. Structure 6

2. Literature Review 7

2.1. Introduction 7
2.2. Project Management 7
2.2.1. History 7
2.2.2. Within Business 8
2.2.3. Within Programming 8
2.2.4. Waterfall 8
2.2.5. Agile 9
2.3. Project Structuring 10
2.3.1. Kanban Boards 10
2.3.2. Work Breakdown Structure 10
2.4. Version Control 11
2.4.1. History 11
2.4.2. Git 12
2.4.3. GitHub 12
2.5. Current Software Solutions 12
2.5.1. Microsoft Excel 12
2.5.2. OpenProject 13
2.5.3. Microsoft Azure DevOps 14
2.6. Gap in Knowledge 15

3. Research Methodology 16

3.1. Research Philosophy and Strategy 16
3.2. Data Collection 16
3.3. Implementation 17
3.3.1. Pilot Study 17
3.3.2. Usability Study 17

4. Development 19

4.1. Introduction 19
4.2. Website Design 19
4.2.1. Front-End 19
4.2.1.1. Access Control & Member Management 19
4.2.1.2. Project Creation 20
4.2.1.3. Project Controls 20
4.2.1.4. Part Creation 21
4.2.1.5. Part List 22
4.2.1.6. Bootstrap 23
4.2.2. Back-End 23
4.2.2.1. Page Preparation 23

4.2.2.2. Imported Functions 24
4.2.2.3. Database Queries 24
4.2.2.4. Cross-Site Request Forgery 24
4.3. Database Design 25
4.4. GitHub API 26
4.4.1. Linking GitHub Account 26
4.4.2. Importing Project Details 26
4.4.3. Commit and Issue Tracking 26
4.4.4. Issue Creation 27
4.5. Version Control 28
5. Results Analysis 29
5.1. Analysis of Pilot Study 29
5.2. Analysis of Usability Test 30
5.3. Data Analysis Summary 32
6. Project Management 33
6.1. Development Methodology 33
6.2. Social, Legal, Ethical and Professional Considerations 33
6.2.1. Social and Legal Considerations 33
6.2.2. Ethical and Professional Considerations 35
6.3. Risk Management 37
6.4. Record of Meetings 38
6.5. Feedback 40
6.6. Project Management Reflections 41
7. Conclusions 42
7.1. Summary of Results 42
7.2. Future Development & Research 42
7.3. Reflection 42
8. References 43
9. Appendices 46
9.1. Detailed Project Proposal 46
9.2. Pilot Test 50
9.3. Participant Information Sheet 52
9.4. Consent Form 54
9.5. Usability Test 55
9.6. Pro-forma 58
9.7. Gantt Chart 59
9.8. Ethical Approval Certificate 60
9.9. GitHub Repository 61
9.10. List of Figures 62

Abstract

Work Breakdown Structure (WBS) is a hierarchical structure for a project, with each level of the structure being more detailed than the level above. The current project management solution offered by GitHub does not include WBS, therefore this project looks into researching and developing a solution that integrates WBS with GitHub.

This report investigates the usability of Work Breakdown Structure in the project management process of GitHub projects. Literature is reviewed to give insight into project management processes and solutions that already exist. A web-interface was developed to allow practical usability studies to be carried out to gain understanding about the suitability of WBS in this environment.

An analysis of results from this report show a positive response to the usability of Work Breakdown Structure in this particular use-case. Several points for improvement, specific to the developed web-interface, were raised and areas for future development and research were identified.

1. Introduction

1.1. Project Aim

The aim of this project is to investigate the integration of Work Breakdown Structure into the project management process of GitHub projects.

1.2. Project Objectives

- To create a project management web-interface.
- To integrate GitHub's API into the web-interface.
- To evaluate the web-interface using a usability test.

1.3. Background and Context

Work Breakdown Structure is a method of planning the structure of a project by ordering individual parts in a hierarchical manner. The structure allows each part of the project to be broken down into subparts recursively until all tasks needed to complete the project have been identified.

GitHub is a service that hosts Git repositories as well as offering project management tools. This project aims to integrate Work Breakdown Structure into the project management process of GitHub projects.

1.4. Motivation

My motivation for this project idea was a Youtuber called "Wintergatan" (Wintergatan – YouTube n.d.), who is filming his own complex engineering project; the creation of a marble machine that plays music. In one of his videos, he discussed how it was difficult to manage a project with so many parts and assemblies. He then works with a professional project manager that shows him an Excel spreadsheet system that implements Project Breakdown Structure (PBS), also known as Work Breakdown Structure (WBS). I liked this method of project management because I found it much simpler to use when writing down my plan for complex coding projects. It is also a deliverable-oriented system. That means that I can specify what I need each part to be without having to precisely calculate how it will be done, which I find leaves far more flexibility when coding. Over the time that I've been using WBS I've found several paid subscription sites online that use this system but don't offer integration with GitHub. So, I now want to develop a site that allows me to manage my project online and manage my GitHub repositories all from the same place.

1.5. Structure

The structure of this project is as follows;

- **Initial Research**
  Research state of the art within this area. Areas to include: project management, project management processes (E.g. Kanban).

- **Compare Currently Existing Systems**
  Compare the system that is currently in use by GitHub against other project management tools that are also available.

- **Create Requirements Definition**
  Write up requirements for the project based on findings from literature review.

- **Software Development**
  Develop the web-interface and it's required database.

- **Evaluation**
  Evaluate the web-interface through the use of usability testing, comparing the newly created solution against the project management solution included in GitHub.

- **Analysis and Writeup**
  Analyse results from usability tests and writeup results and findings.

2. Literature Review

2.1. Introduction

This section will firstly introduce an overview of project management and the methods that are used within it. Structuring of projects will be discussed along with how they relate to the project management methods discussed. Finally, this section will cover version control and how GitHub is used within version control and project management.

2.2. Project Management

The Oxford English Dictionary defines project management as “The theory, practice, or occupation of managing projects” (Oxford English Dictionary n.d.). This can include planning and controlling a team to ensure that project requirements are met.

2.2.1. History

Project management can be thought of as ancient, as humans have been using some form of management for centuries; for example, the ancient Egyptian pharaohs used managers for the creation of the Great Pyramid. This and the progressive history of project management are discussed in a blog post from Project Manager by J. Westland (2018).

The post talks about how, more recently, project management is being used across various industries to structure, manage and deliver commercial projects, and the methods used within the management process that have developed over time.

A management tool, which has origins that can be traced back to 1917 and is still used today, is Gantt charts. This tool is a chart that allows visualisation of tasks time allocation and has now moved into digital software tools. A technique, developed in 1957, that fits well with Gantt charts is the Critical Path, which predicts how much time a project will require for completion based on the sequence that tasks are completed.

Another project management method, originally developed around 1962 and still in use today, is Work Breakdown Structure (WBS). This method involves having a complete hierarchical tree structure of parts/tasks needed for project completion. It was originally developed during the US DoD’s Polaris project and is still popular for managing engineering projects, as it’s hierarchical tree structure tracks components and their sub-assemblies clearly.

In 1969 the Project Management Institute was founded, which was the first certification body offering project management certifications, and published 'A Guide to the Project Management Body of Knowledge (PMBOK)', which is still relevant today and in its 6th edition (at time of writing this report).

In 1989 the UK government introduced Projects in Controlled Environments (PRINCE), and subsequently PRINCE2 in 1996, to standardise projects management for information systems. PRINCE2 certifications are still used today, with over 5,000 corporate clients and over 500,000 graduates (PRINCE2 2020).

2.2.2. Within Business

Nicholas and Steyn (2017) discuss the need for project management and how modern projects often involve technical complexity and challenges. These complex modern projects could include large construction projects, such as building a new bridge, down to developing a piece of complex software. They also discuss how project management is used, when businesses give their projects constrained resources and limited time, to meet the projects requirements.

2.2.3. Within Programming

Due to these constrained resources, many companies within the software development industry have taken on the approach of Global Software Development (GSD), which is when software development occurs by multiple teams based in various geographical locations. Niazi et al. (2016) analysed the various challenges of managing these GSD projects and found 19 challenges that cause issues in project management. The second most frequent challenge found was lack of communication, which then gives rise to lack of understanding requirements. Communication between teams is often through email or messenger applications, due to the different time zones. Another challenge found was lack of co-ordination or allocation of tasks, which may be attributed to a lack of communication or use of poor systems for communication.

2.2.4. Waterfall

Pressman (2010: 39-41) explains how the Waterfall methodology is a sequential approach to software development starting with requirements for the project through to development of the software, and finally the delivering of the software product. This means that each step must be completed before moving onto the next, for example the 'Planning' stage must be completed before the 'Modeling' stage can begin.

Pressman also discusses a version of the Waterfall methodology called “the V-model”, which follows the same structure as discussed above, but adds an additional stage at each level for quality assurance. This means that you can imagine the methodology spread out in a large ‘V’ shape. As the team moves down the left-side of the ‘V’ they complete each stage then move onto the next, from detailing requirements to outputting code. Once code has been created, the team moves up the right-side of the ‘V’, which has quality assurance tests that pair with each stage that was completed when coming down the left-side of the ‘V’.

Finally, Pressman discusses some of the issues encountered when using the Waterfall methodology for software development. The main issues that are discussed include:

- A real-life project rarely allows for sequential management, often requiring iteration or modification.

- The Waterfall methodology’s need for all system requirements to be stated at the start of the project is often unrealistic, as customers aren’t always sure what they want or need, and requirement can often change throughout the project.

- The methodology only starts to produce code, which can be shown to the customer, late in the process; that can make involving the customer in development difficult, and can cause large costs and loss of time if changes are needed to be made after customers see the code.

2.2.5. Agile

An alternative methodology is the Agile methodology. Pressman (2010: 68-70) also discusses this methodology and the principles behind it. He discusses how this methodology uses agility to manage projects with the following characteristics:

1. The projects requirements may be added, removed or modified at any point during the development process, even in an unpredictable manner.

2. The projects design and development stages may be worked on at the same time, allowing for designs to be developed and then more designs, or altered designs, to be made for further development.

3. It may be difficult to predict when stages such as analysis, development and testing may be undertaken.

This dealing with unpredictability is where the agility from the Agile methodology is used, as it allows for development teams to work in a manner that can cope with an adapting project. Pressman explains

that this methodology accomplishes this adaptability, without preventing the forward progress of the project, through regular customer feedback. To be able to receive this feedback, development must iteratively produce working code, often in the form of prototypes or individual parts of the project. This code can then be shown to the customer who can evaluate it and provide feedback, which may also incrementally alter the overall project requirements. This feedback, and any altered requirements, can then be dealt with in the proceeding iterations of development and can be further evaluated by the customer.

2.3. Project Structuring

Project structuring techniques are used within various project management methods to define and layout the tasks, parts and/or goals of a project. Using a structure technique within the management process can give a clear view on what needs to be done and in what order tasks should be completed, so that project completion can be reached within the requirements.

2.3.1. Kanban Boards

A Kanban board is a table that is used within the Kanban methodology, which is a management method that balances capacity with demand to improve workflow. Each column contains tasks at different stages (E.g. 'Backlog', 'To-do', 'Doing' and 'Done') and these tasks are moved across the board by team members as work is completed. Powell (2018) discusses how Kanban boards are used during daily scrum meetings to manage what was done the previous day, and then choose what tasks will be worked on that day. Also discussed, is how each column, except the backlog, can be limited to only contain a certain number of tasks, to ensure they get completed. Powell concludes that Kanban boards can be useful tools within high-mix, low-volume environments (E.g. software development), as it creates a pull mechanism for tasks.

2.3.2. Work Breakdown Structure

The Project Management Institute (2008) describe Work Breakdown Structure (WBS) as a deliverable-oriented hierarchical structure, of a project, that states all tasks that need to be completed for the deliverables to be created. Each level of the structure (going down the hierarchy) is a more detailed breakdown of the level it's linked to above.

2.4. Version Control

Software development projects often consist of multiple programs, which may be developed by one or multiple programmers. Therefore, there is a requirement for these changes to be tracked, so that developers can go back to previous versions of code if errors are made. This is often handled by a version control system.

2.4.1. History

Version Control is the process of managing changes to a collection of information. Within computing, this collection of information is often a collection of source files containing code for a given project. This means that version control software can be used to track changes made to the code and review past versions of code within the collection or project.

As discussed by W. Carstensen (2016), version control originated within engineering for tracking individual components and visualising components in a hierarchical view to see when changes in a component would have a knock-on effect to other components.

Version control for software development originally took the form of office work, as developers had to 'check-out' and 'check-in' code on punched cards, similar to checking in and out technical drawings for engineering; as version control became software-based, the process was automated. These software-based version control solutions allowed developer teams to host their version control on a centralised server; that worked well for development teams that all had access to the centralised server.

One issue of version control software that still exists today is "merge failure", which occurs when two or more changes are applied simultaneously to the same piece of code; this then requires developers to manually merge all the submitted changes.

Version control solutions today mostly don't use the centralised approach but instead use a distributed approach. This means that each developer "clones" the entire repository and keeps a local copy of the full code history. They can then either make changes to the code, and "push" those changes for other developers to see, or "pull" other developers changes to their local copy of the repository.

2.4.2. Git

Git is a version control system (VCS) that allows developers to keep a history of all changes made to files within a repository (a project), but also allows for multiple developers to work on project code at the same time and then have a centralised storage system (E.g. GitHub).

2.4.3. GitHub

GitHub is a service that offers Git repository hosting and is widely used by developers, both inside and outside of business. GitHub is used by over 40 million developers and is hosting over 44 million repositories, with enterprise users in organisations from over 70 countries (GitHub 2019). GitHub delivers this service through a website that allows developers to view and manage their repositories. However, GitHub also offers other features to its users such as Wiki's, which can be used to document software, a web-interface for managing issues and pull requests (features of Git) and project boards that are used to manage the project. The GitHub project boards feature makes use of Kanban boards to allow users to list tasks and sort them (E.g. sort them according to status; 'To Do', 'In Progress' and 'Complete').

2.5. Current Software Solutions

2.5.1. Microsoft Excel

Microsoft Excel is a software solution that allows the creation of spreadsheets. It also offers automation through the use of formulas, several accessibility features and also some advanced features, such as importing a spreadsheet from a photo, as described on its Microsoft product page (Microsoft Excel n.d.).

Spreadsheets can be used in the project management process in a variety of ways, from organisation of tasks through to budgeting. Management of tasks can be achieved by implementing one of the management processes; for example, columns containing lists of tasks to create a Kanban board, or an ordered hierarchical list of tasks to form the WBS (see Figure 1).

	Level	WBS	Task Description	Assigned To	Start	End	Notes
5							
6	1	1	Phase 1				
7	2	1.1	Task Level 2 Description				
8	2	1.2	Task Level 2 Description				
9	3	1.2.1	Task Level 3 Description				
10	3	1.2.2	Task Level 3 Description				
11	4	1.2.2.1	Task Level 4 Description				
12	4	1.2.2.2	Task Level 4 Description				
13	4	1.2.2.3	Task Level 4 Description				
14	2	1.3	Task Level 2 Description				
15	1	2	Phase 2				
16	2	2.1	Task Level 2 Description				
17	3	2.1.1	Task Level 3 Description				
18	3	2.1.2	Task Level 3 Description				
19	1	3	Phase 3				
20	2	3.1	Task Level 2 Description				
21	3	3.1.1	Task Level 3 Description				
22	4	3.1.1.1	Task Level 4 Description				

Figure 1 – Work Breakdown Structure in Microsoft Excel

2.5.2. OpenProject

As described on OpenProject's site (OpenProject n.d.), it is an open source project management software solution that has a main focus based around the agile methodology. OpenProject also provides tools for working with Agile's Scrum and Kanban as well as providing tools for producing roadmaps, bug tracking and budgeting. It allows the creation of Agile items, such as tasks, features and milestones, and then displays these using either a Kanban board, the WBS (see Figure 2) or in a Gantt chart.

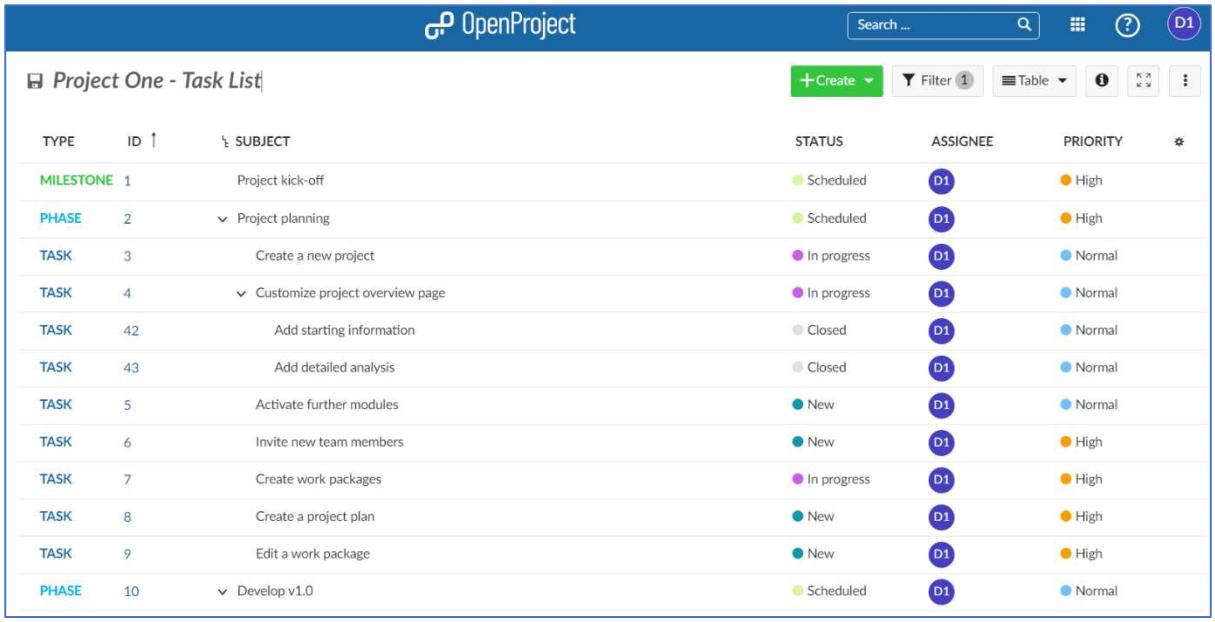

Figure 2 – Work Breakdown Structure in OpenProject

A paper by Abramova, Pires and Bernardino (2016) compares open source project management tools to proprietary ones. They make comparisons between three software solutions, of each type, including OpenProject, and conclude that the two types of software offer similar basic necessary functionalities; though, they also conclude that the proprietary solutions that they used were more user friendly and intuitive.

2.5.3. Microsoft Azure DevOps

Azure DevOps (Azure DevOps n.d.) is a Microsoft product that is made up of a collection of services that allow the planning, developing and testing of projects. The planning phase makes use of the Azure Boards service (Azure Boards n.d.) which is a project planning tool that is Agile focused. It supports Agile's Scrum and uses Kanban boards for task management (see Figure 3). Azure Boards also includes the ability to add extensions, such as support for other Microsoft products (E.g. Teams and GitHub). Its GitHub integration tools allow commits and pull requests on GitHub to be linked to work items within Azure Boards.

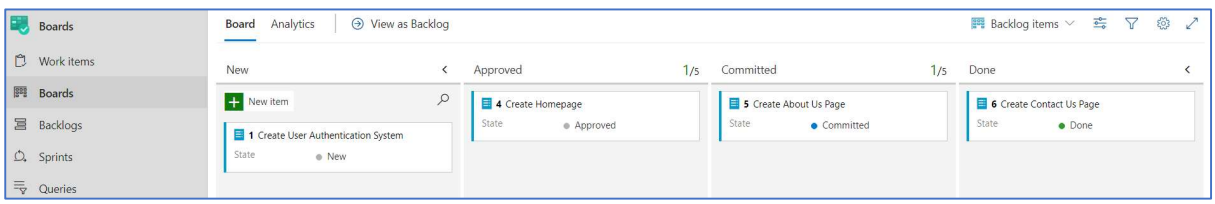

Figure 3 – Kanban Board in Microsoft Azure Boards

2.6. Gap in Knowledge

After researching and reviewing the relevant literature surrounding the topic of project management, its processes, and its software implementations, this project will investigate the use of Work Breakdown Structure in a project management system that is integrated with GitHub. The reviewed literature and currently existing systems show these features in use as project management tools individually but leaves a gap in knowledge when integrated together in one solution. This study will look at the usability of a solution that makes use of both features in a project management system.

3. Research Methodology

3.1. Research Philosophy and Strategy

For the research section of this project, a functional web-interface will be developed and then usability testing will be conducted. Before conducting the usability study, a pilot study will be completed to check the quality of the technique. As discussed by Jim Ross (2016), a pilot study allows the technique to be tested and can help identify issues with the test, including issues with:

- Understanding tasks or questions being given to participants
- Any bias that may be present in the given tasks or questions
- Do the task and questions cover all areas of the interface that need to be tested

A usability study will then be used to help determine the usability of WBS in the management of GitHub projects. Jakob Nielsen (2012) discusses how the usability of web-interfaces is crucial for their survival, because an interface that is unusable makes a user leave. He goes on to discuss that the use of user testing is the most useful method of identifying flaws in the usability of an interface.

3.2. Data Collections

During the research section of this project, primary data was collected using a usability test. To ensure that the test had been clearly written and would be understandable for participants, a pilot test was conducted first. The feedback about the clarity of the tasks and instructions given in the pilot test were used to further develop the test and better phrase some of the tasks. The updated version of the usability test was then sent out to six participants, who all sent back the feedback forms. Six participants were used as this number is supported by research completed by Jakob Nielsen (1989), which advocated simplified user testing, which included testing five users. In a blog post (Nielsen 2000), he discusses how the use of around five users in testing gives the best results and that these smaller tests should be completed more often.

These tests where completed during the COVID-19 period, therefore the tests were sent to participants via email. These emails contained the test sheet, which contained the tasks and questions, the information sheet and the informed consent form. Participants confirmed their consent to participate through email, when returning their responses, due to many of them being unable to print off the consent form, sign it, scan it and return it.

3.3. Implementation

3.3.1. Pilot Study

A pilot study was completed first, to test the clarity and understandability of the tasks and questions within the usability test. This was achieved by giving the usability test to a participant and requesting feedback on the clarity of the tasks given and the understandability of the questions that evaluate users' experience. This study was completed and several points for improvement were given as feedback. These points included several areas where the participant felt tasks lacked the required level of guidance and also where some questions could be made more relevant to the tasks that had been carried out. These improvements were considered, and several alterations were made before the newly updated usability test was sent out to the larger study group.

3.3.2. Usability Study

A usability study was undertaken after the pilot study had been completed and the relevant alterations had been made to the test's task list and evaluation questions. Participants were given 13 tasks to complete that instructed them to use the main features of the system; these were:

- Creation of projects
- Creation of parts
- Manipulation of projects
- Modification of parts
- GitHub integration features:
    - Linking a project to a GitHub repository
    - Importing details from a GitHub repository (E.g. Title or Description)
    - Commit tracking
    - Issue tracking
    - Issue creation

Once the participants had attempted all of the given tasks, they were asked several questions that were designed to evaluate their experience of using the system. For the first part of the evaluation questions, the participants were asked to rate (from 'strongly disagree' to 'strongly agree') nine

statements about the ease of use of the features covered in the test's tasks. They were then asked to give reasons for any feature they had given a rating of 'strongly disagree' or 'disagree' to. This was done to be able to get feedback on what improvements could be made to features that participants had found difficult to use.

Participants were then asked several open-ended questions, starting with whether they had been able to complete all of the tasks set in the test, and if not, to explain why. They were also asked if, during their use of the system, there had been anything they had found difficult to use and why. Both of these questions, together, give insight into possible areas for improvement and give the participant opportunity to discuss any difficulties they had during the test.

Finally, the participants were asked about the layout and design of the interface. These questions were designed to be less about the participants ability to complete the given tasks and more about their opinion on the overall structure and usability of the interface. Feedback could be given on a range of areas from positioning of controls, to the fonts and colour schemes used.

4. Development

4.1. Introduction

During the development stage of the research, multiple technologies were brought together to create the required web-based system. The system makes use of a web-interface as its front-end for users to interact with and a database and back-end for processing and storing data that users enter or request.

4.2. Website Design

The website is made up of a front-end user interface for displaying and controlling projects, and a back-end for data processing. The front-end uses HTML, CSS and JavaScript to display data and controls, which are then used by the user to send data to the back-end for processing and storage in the systems database. The website was developed using Google Chrome as the test web-browser, a local MySQL database and a locally hosted Apache webserver.

4.2.1. Front-End

The front-end interface was developed mainly using HTML and CSS, with JavaScript being used to add functionality to dynamic features (E.g. Collapsible lists) and for using the GitHub API.

4.2.1.1. Access Control & Member Management

One of the requirements of this system is that users should only be able to access their own data or data that other users have given them access to. To achieve this, the website uses unique accounts for each user that are password protected. Users have automatic access to any project that was created using their own account, but users can also be invited to join projects created by other users. This system works by allowing a project owner (the account that created the project) to invite other users to join their project. Users are identified by their username and can be invited either as a user with read and edit permissions or just read permissions. The reason for having accounts with only read permissions is that project owners may wish to allow their customers or other personnel to view the management process or work progress without allowing them to interact with any of the management data.

4.2.1.2. Project Creation

Another requirement of this system is to allow users to create 'projects'. The projects are used to manage access rights, customise options to best suit the needs of the project being undertaken and contain 'parts' (explained below) to be completed. When creating a project, a user can choose to create a blank project or a GitHub project. A blank project will allow the user to use all of the project management tools within the system but won't be able to use any of the GitHub integration tools. A GitHub project allows the user to make use of all the project management and GitHub integration tools, such as commit tracking and issue creation. When creating a GitHub project, users can also choose to automatically import details from an existing GitHub repository, removing the need to manually provide a project title and description. GitHub projects and GitHub integration tools can only be used by users that have linked their GitHub account with their account on this system.

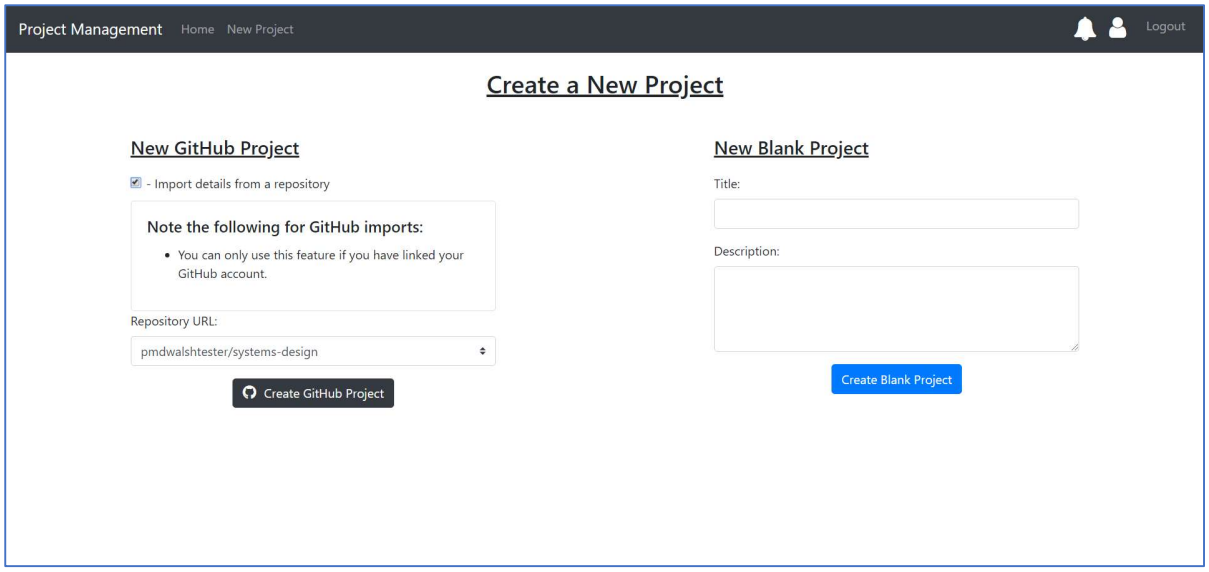

Figure 4 – Project Creation Page

4.2.1.3. Project Controls

A third requirement of the system was that it needed to be able to meet the individual needs of different development teams. These teams may be implementing various management methodologies and may have various requirements for what details they store during the management process. To meet these varied requirements, the system allows each project to have its own individual settings so that each one can be customised as required. These project settings cover a range of areas throughout the management process such as automation options, various details options and management of users.

4.2.1.4. Part Creation

This system uses the word 'part' instead of 'task' because it is based on the Work Breakdown Structure (WBS), which was initially used in engineering where there would be assemblies, parts and subparts. These 'parts' and 'subparts' are interchangeable with software developments 'tasks' and 'subtasks'.

When creating a new part, or subpart, the only required information is a name for the part. This means that each part can be identifiable to users, but they can choose how much, or how little, detail they provide in a part. Parts can also be provided with the following, optional, details:

- Description – Used to give more detail about the part or its requirements.
- Date Range – Used to specify the date a part is started and finished. These could be target dates or the actual dates. To maximise the user's ability to use this feature as needed, the dates are independent, so either both dates can be given or just one.
- Status – Used to quickly identify what stage an individual part is at. A part's status can be one of the following:
    - Unassigned
    - Assigned
    - In Progress
    - Needs Inspection
    - Complete
    - Issue
    - Abandoned
- Priority – Used to specify whether an individual part is either low, normal or high priority.
- Version – Used to specify the version number that an individual part is on.
- Progress – Used to show the percentage of progress made on an individual part, and it's used in automation of management by marking parts with a 100% progress with a 'complete' status (dependant on project settings).
- Category – Used to group multiple parts using a single word or phrase.

Also, the priority, version, progress and category options can be enabled or disabled globally for a project, from that project's settings, to allow its owner to customise what data they wish to be stored.

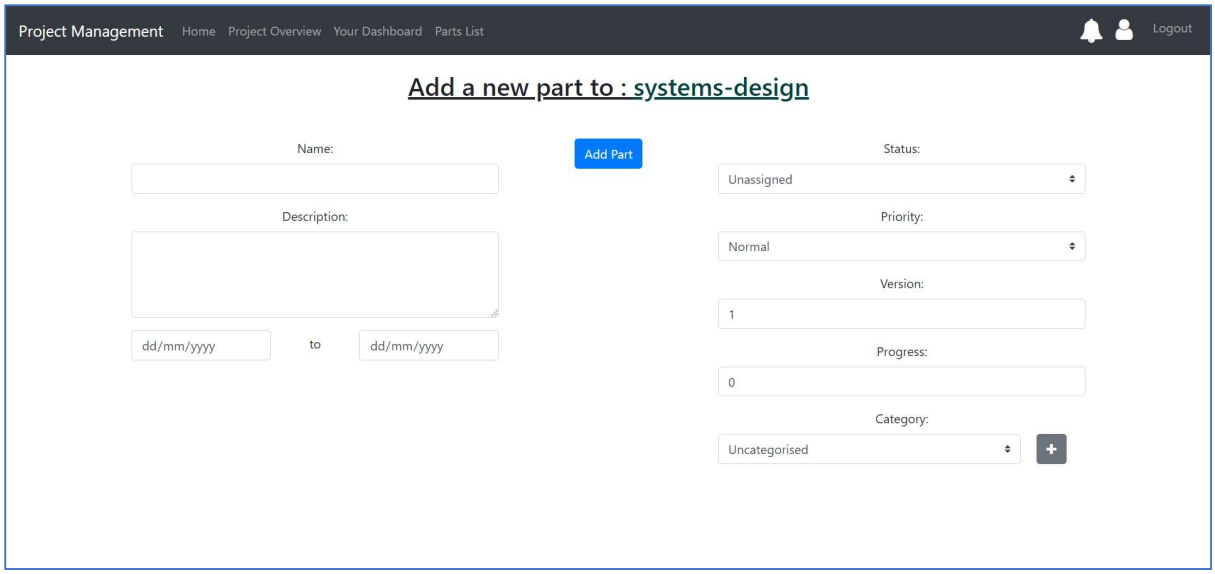

Figure 5 – Part Creation Page

4.2.1.5. Part List

Another feature of the system is the way in which the parts are structured and displayed to the user. This system makes use of WBS, which is a hierarchical structure, therefore the system presents a list of all parts in this hierarchical manner. The parts list makes use of multiple collapsible lists to present parts and uses indented lists to signify subparts. The top level parts of the hierarchical list are ordered by date and time of creation, with their respective subparts directly below them in an indented list.

On the parts list page, are badges that display the current status of each part and controls to add a new top level part or subparts of already existing parts. The parts displayed in the list can also be searched by their status; for example, a user wishing to find all parts that haven't yet been assigned to a user can search the list for all parts with the 'unassigned' status.

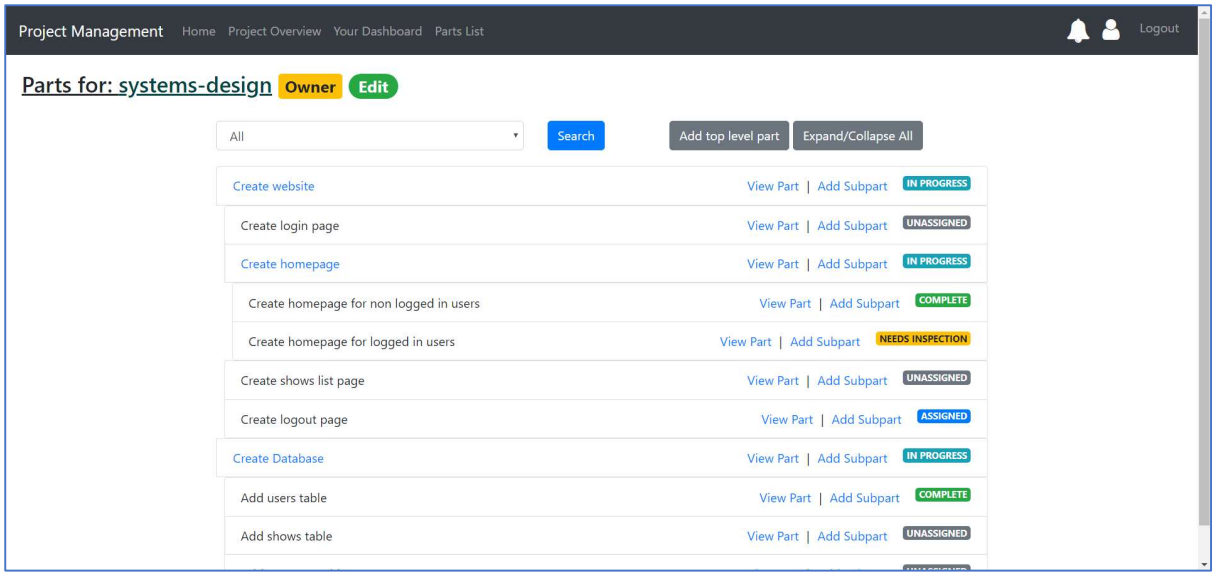

Figure 6 – Parts List Page

4.2.1.6. Bootstrap

During the development of this system the main focus was functionality, however the layout and style of the site was also important from a usability perspective.

To ensure clean styling and a simple layout, the Bootstrap (Bootstrap n.d.) library was used. It is a free, open-source library containing CSS and JavaScript that can be used to create responsive front-end interfaces. Bootstrap makes use of a grid-based solution and responsive components so that pages can be designed to remain clear on any sized device. This system makes use of multiple Bootstrap pre-built components, such as badges and cards, to structure pages. Using this solution, more development time could be spent focused on the functionality of the site rather than coding the site's CSS.

4.2.2. Back-End

The back-end of the website was developed mainly using PHP with MySQL being used for interaction between the back-end and the database.

4.2.2.1. Page Preparation

Almost every page that a user can visit on the site requires some sort of preparation before being served and this is achieved through PHP. When a user requests a page from the webserver, the system first ensures that the user is authenticated (has logged into the system) and has permission to view

the requested page. This is required to ensure users can't access data, such as projects or parts, without the appropriate permissions. The system then inserts the relevant navigation menu, dependant on which page is being requested, and inserts any data that has been retrieved from the database. The page is then served to the user for them to view and interact with.

4.2.2.2. Imported Functions

To maximise code reuse and minimise the overall file size of the system, it makes use of PHP's ability to import one PHP file into another. This ability is used by creating 'function' files that solely contain PHP functions that may be used in multiple pages across the site. If a page requires access to a function it imports the relevant function file, which is stored with the '.php' extension to ensure the source code cannot be viewed by malicious users.

4.2.2.3. Database Queries

Due to the system storing the majority of users' data in a MySQL database, the back-end PHP code needs to have a connection to that database. This is achieved through PHP's 'mysqli' extension, which enables the system to create a connection with the database and then pass queries and results over the created connection. The system also contains MySQL queries in the PHP code that can initially create the database, which this system uses, if it doesn't yet exist.

4.2.2.4. Cross-Site Request Forgery

Cross-Site Request Forgery (CSRF) is a type of cyber attack where the attacker is attempting to get the user, that has already authenticated into a system (E.g. A bank account), to make a change or action rather than retrieve data. As discussed by the Open Web Application Security Project (OWASP) (n.d.), these attacks, if targeting normal users, may attempt to force the user to alter data (E.g. change their email address). But, if the attack is targeting an administrator account, the security of the entire web application and the data held within may be jeopardised.

These attacks work by assuming the user has already authenticated into the system being attacked, and then the attacker sends a link for the user to click on, perhaps in a chat or email. The link takes the user to a location on the system where an action can be started automatically.

An example use of this attack on this system would be that a user is authenticated into the system and then clicks a link received in an email. The link is a URL that is used to delete projects (E.g.

domain.co.uk/delete-project.php?id=364), and if there are no protections against CSRF the project would be successfully deleted, because the user has already been authenticated.

To protect against this, every action that a user can make on the site uses an anti-CSRF code. This is a random string that is generated at the start of every session and is inserted into every page, so that if the user makes an action, both the action and the anti-CSRF code gets sent to the back-end.

Therefore, using the example of deleting projects above, if the attacker still wanted to successfully direct the user to the delete page they would also need to provide that users anti-CSRF code (E.g. domain.co.uk/delete-project.php?id=364&**antiCSRF=kf83jr74jjhdu5ii87hd3**…). This means that attackers would need a new method to firstly gain access to a user's anti-CSRF code before making use of this attack. This is further safeguarded by making each code invalid once a user logs out of the site or when their session expires. A new random code is then generated once the user returns to the site.

4.3. Database Design

During the development of this system, the MySQL database used to store data was also developed. The final database design is made up of 11 tables that are all related to each other. The database has been normalised, to reduce data redundancy and improve integrity. All data stored within the database is stored unencrypted, apart from the passwords for user accounts. Account passwords are hashed during the registration process and the produced hash is then stored.

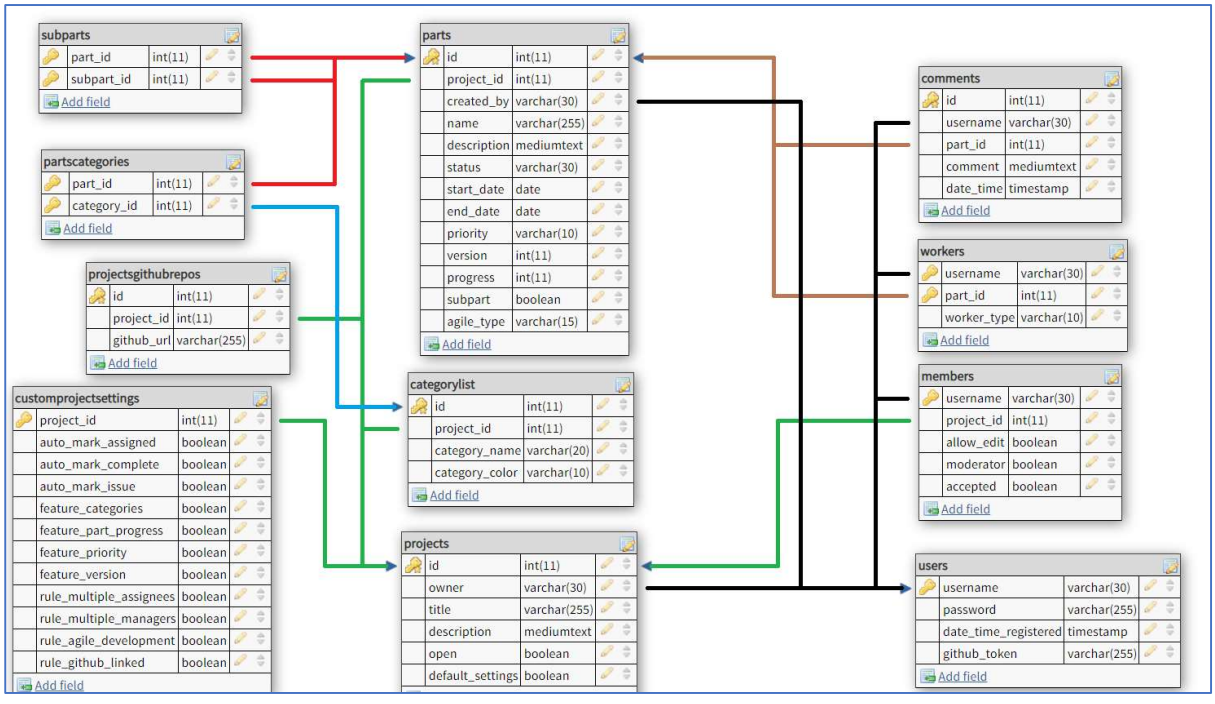

Figure 7 – Database Structure Diagram

The relationships in this database also take advantage of the 'CASCADE' delete option to ensure relationship integrity. This means that if a project is deleted, any parts that are related to that project are also automatically deleted. This occurs recursively through all relevant tables to ensure no record has a foreign key that references a record that has been deleted.

4.4. GitHub API

4.4.1. Linking GitHub Account

To allow this system to integrate with GitHub, users need to link their account on this system with their GitHub account. This is done using the authentication method provided in the GitHub API. This method is described by GitHub as their "web application flow", which involves the user clicking a link in their account settings that redirects the user to GitHub, where they are identified and authenticated. GitHub then redirects the user back to this system. It also provides an access token that is stored in this system's database and used for all authentication required GitHub interactions. Storing and using this access token means that a user only needs to link their GitHub account once to be able to permanently use GitHub integrated features, rather than having to repeatedly provide their GitHub credentials. Should a user wish to remove the created access token, they can do so from their GitHub account settings or by deleting their account on this system.

4.4.2. Importing Project Details

One feature of this system is the ability to not only link a project to a GitHub repository, but also to import the title and description from a GitHub repository and use them as the title and description of the new project. This allows project managers to maintain consistency across platforms and saves time. This is achieved by firstly retrieving all public and private repositories linked to the users GitHub account and displaying them in a dropdown box. The user can then select which repository they would like to import the title and description from, then these details are retrieved from the selected repository and used in the newly created project.

4.4.3. Commit and Issue Tracking

The main GitHub integration feature of this system is to be able to link commits and issues, hosted on a GitHub repository, to parts hosted within this system. The ability to track commits that are linked to parts means that its progress can be tracked and files related to a part can be identified. Tracking issues that are linked to a part will mean that any problems can be seen, not only from a development

point of view, but also from a management point of view. Tracking issues in a repository is also used in one of the automated management features, as any part that has an issue linked to it gets marked with the “Issue” status to allow parts with active GitHub issues to be quickly identified.

The tracking of commits is achieved using a reference string, which users must include in the commit’s title or description. This reference string always starts with “pm=” and then includes the part’s identification number (E.g. “pm=59983”). The tracking of issues is similar to that of commits, but the reference string must be included in the title of the issue.

4.4.4. Issue Creation

The final GitHub integration feature that has been implemented in this system is the ability to create GitHub issues from within this system. The advantage of generating issues from within this system is that the reference string for the part, that the issue is being created for, is automatically included in the issue title. This removed the need for the user to manually find and include the relevant part number.

This is achieved by allowing users to add an issue to any part within a project. Upon starting the issue creation process, the user is asked which of the GitHub repositories (as a project can be linked to multiple GitHub repositories) the issue should be published to. The issue title and chosen repository is passed to the GitHub API for it to be published on that relevant repository.

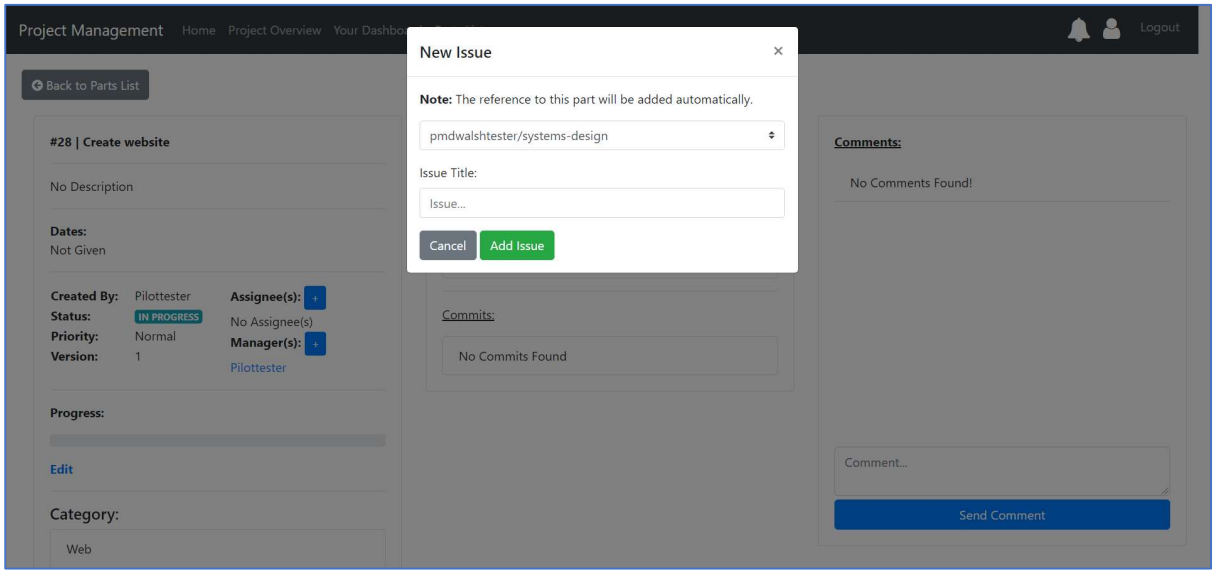

Figure 8 – Issue Creation Modal

4.5. Version Control

Due to the nature of PHP, both the front-end and back-end code are stored in one Git repository hosted on GitHub. Using GitHub allowed the code to be backed-up to a remote server, minimising the risk of losing code. Git allowed easy back tracking to working versions of code, should any errors be made during development. It also allowed the creation of branches, which were used for separating new features. These feature branches were only merged to the master branch after they had been completed and tested, to ensure there were no faults. GitHub's 'issues' feature was also used to track any bugs or additional requirements that were encountered during development.

The repository only includes credentials for development databases (hosted locally only) and development GitHub API credentials, all of which were destroyed after the development finished. At no point were the credentials for the live site, it's database or linked GitHub API account, included within the GitHub repository.

5. Results Analysis

5.1. Analysis of Pilot Study

Before conducting the usability study on the full group of participants, a pilot study was undertaken to find any errors in the test, and to refine the given tasks and evaluation questions. This test consisted of 11 tasks and four evaluation questions, one of which made use of a numerical scale to rate features used within the test, with the remaining three questions being open-ended and about the participants experience.

One of the first issues noted in the feedback was that instructions for a task, which involved adding a category to a part, were too vague for the participant to be able to understand what they were being asked to do. To fix this issue, the task was split into two tasks, adding more detail on where participants should be. It also added clarity on what was expected of them.

A second issue raised in the feedback was that the participant felt the task instructions didn't give enough guidance on where to find the 'project settings'. The purpose of this task was to test the usability of the settings menu and understandability of the help tooltips, not a user's ability to find the menu, so an alteration was made to give more guidance on where to find this menu.

The final issue, about the test's tasks, that was highlighted in the feedback was that not enough guidance had been given for the creation of a new project. The participant noted that they were unsure if they should be creating a new 'GitHub project' or a new 'blank project'. To address this issue, the task was altered to specify that the participant should be creating a new 'GitHub project'.

Feedback was also given for the four evaluation questions that the participants answered, after completing the tasks given in the test. The first suggested improvement was to switch the first question's rating system to a Likert scale ('strongly disagree' to 'strongly agree') from the numeric scale being used (1 to 10). This would give more understanding of how participants felt about each feature used.

A further suggestion was to ask why any features were given a poor rating or why any tasks could not be completed. This would then give insight into areas for improvement for the system that participants had not found easy to use.

The final suggested improvement was to include an open-ended question that would allow the participants to make comments about the general look of the system. This would give insight into the overall design quality of the system; therefore, two open-ended questions were added. One focused on the general layout and structure of the system while the other focused on the general design, look and feel of the overall system.

5.2. Analysis of Usability Test

A usability study was completed after the completion of the pilot study. This study was undertaken during the COVID-19 period, so participants were sent the test and evaluation questions via email. The feedback given in these evaluation question forms gives insight into the usability of the system and the overall design of the site.

The first question asked the participants to rate (using a Likert scale) various features used during the tasks given in the test. This scale was converted to numeric values ('strongly disagree' = 1 to 'strong agree' = 5) and the mean response and median response were calculated for each feature statement. A statement with both a mean and median above three shows an overall positive response to the statement. Below three shows an overall negative response to the statement. The range (the difference between the highest response and the lowest response) was also calculated for each statement. A statement with a range of two or less shows that the participants were aligned in their responses. A range above two shows that the statement created polarised responses. Each feature statement and their respective mean, median and range values are shown in the following table:

Feature Statement	Mean Response	Median Response	Range
I found navigating to the project easy to do	4.3	4	1
I found it easy to view individual parts	3.8	4	2
I found the category system easy to use	4	4	3
I found the assigning system easy to use	4	4	3
I found project settings easy to use	4	4	2
I found creating a project easy	4.1	4.5	3
I found the member management system easy to use	3.6	4	3
I found part creation easy	3.6	3.5	3
I found issue creation easy	4.1	4	2

All the features gained a score above three in both their mean response and median response, which could show that the overall response to the usability of system was positive. Five out of nine feature statements also gained a range higher than two, suggesting that participants of this study had varied opinions on those features. The response to this question shows that, overall, the system is usable, but has areas that need improvement. The open-ended questions, covered next, may identify these areas for improvement or reinforce the strengths of the system.

The second question asked participants to give reasons for any rating of 'disagree' or 'strongly disagree'. The feedback received covered a range of areas within the system. The first of these identified menu controls for navigating to the 'parts list' as difficult to locate. A possible fix for this

could be to re-evaluate the structure of the project overview page to ensure all navigation controls can be clearly identified. Another participant identified a possible improvement for the member management system; this was inviting members to join a project and then controlling their read/write permissions separately. This suggestion would also allow users to then alter member permissions without removing them from the project and then re-inviting them with altered permissions.

The next question asked participants if they had been able to complete all tasks and if not, why. All of the participants responded that they had been able to successfully complete all of the tasks given in the test, with one participant specifying that they still felt the task instructions needed to be clearer. These responses indicate that the system is navigable and that the features tested were accessible.

The fourth question asked participants if there had been any features that gave them difficulty, regardless of whether or not a task had been successfully completed. The feedback obtained from this question indicated that the 'project settings' option was difficult to find and some controls (E.g. 'New part' controls) were not where the participants expected to find them.

The next question asked about the overall layout of the interface, with focus being on the structure of pages. The feedback received on this question varied, with participants giving both praise and possible improvements for the interface. Praise for the interface highlighted that priority options stand out on pages and the structure allowed the interface to be "intuitive" with controls that allow easy use of the system. Many of the suggested improvements focused on the alignment of controls or moving them to be in more intuitive locations on the page, as well as improving the responsive layout of the interface, so that it remains clear when used on smaller screens. A final suggested improvement was to include an option to allow users to add new parts directly from the project dashboard page, rather than navigating to the parts list to do this.

The final question asked participants to give their opinion on the overall design of the interface, focusing on the ease of use and clarity of the interface. This question resulted in several points for improving the interface, both by altering current components and through the inclusion of additional functionality to improve ease of use. There were several suggestions about the look of the interface, including: allowing users to customise the background colour of the interface, using different font sizes on different levels of the parts list, making headings use consistent font size and weight across the entire site and aligning the option controls within the parts lists with each other. The suggested additional functionality included: adding a control to allow users to cancel the 'part creation' process without using their browsers 'back' button, including more help tooltips across the interface (similar to those used in the project settings) and allowing users to search and filter their project list, which would be useful when a user is a member of a large number of projects.

5.3. Data Analysis Summary

Analysis of the results from the pilot and usability studies proved interesting with points raised supporting the system and also for improving it. The data provided a positive response for the usability of the system during completion of tasks but also displayed some polarised opinions. The results also displayed positive feedback on tasks that included GitHub integration tools (E.g. creation of issues). This could also give an indication that the Work Breakdown Structure (WBS), paired with GitHub integration, is a viable solution for management of projects on GitHub. The points for improvement were mainly based on the structure and layout of certain pages within the system, with no negative feedback given on the usability of the WBS.

6. Project Management

6.1. Development Methodology

During the development process in this project, the Agile methodology was applied. This methodology can be customised, through choosing which approaches and processes are used, to suit the needs of the project being undertaken.

Firstly, the Agile Scrum methodology (Takeuchi and Nonaka 1986) was partially implemented, including the iterative development approach it uses, while excluding features aimed towards teams of developers, such as meetings with the team leader (known as the 'Scrum Master') in 'daily stand-ups'. When a new feature was to be added, a list of tasks was created and each task was completed, one at a time, until the feature was considered complete. This process was then repeated for the next feature to be added from the feature list.

The second approach used in the Agile method was Lean Software Development (Poppendieck and Poppendieck 2003), which focuses on delivering the most value possible during a set development time period. This was achieved by only working on tasks that were required to implement the feature, and not adding additional functionality that didn't contribute to getting the feature to a 'complete' state.

The final approach used in the Agile method was Feature Driven Development, which involves a model-driven process where iterations are focused on developing a single feature. This process allows for an overall model to be made that gives insight into the features that will be required. Development is then planned and carried out based on iterations completing the development of all features on this 'feature list'. The first iteration of project development is then considered complete and the system can then be tested.

6.2. Social, Legal, Ethical and Professional Considerations

While completing this project, I will be conducting development of a software solution and also conducting research. I will, therefore, need to make considerations towards the social, legal, ethical and professional implications of my project.

6.2.1. Social and Legal Considerations

For the legal considerations of the project, I needed to consider both my developed solution and the research that would be carried out.

Firstly, data protection needed to be considered, as it falls under the General Data Protection Regulation (GDPR) and the Data Protection Act 2018 (DPA), which is the UKs implementation of GDPR. These laws relate to how personal data is stored and the rights of individuals in regard to control over their data. For this projects research, these laws effect how the results of the research are stored, what data can be collected and also what information needs to be given to participants of the research. To attempt to ensure compliance, I created an information sheet with general information about the purpose, procedure of the research and participant rights in relation to GDPR and the DPA. I also got all participants to sign an informed consent form that confirms they had read and understood the information sheet. As the produced solution is a website, it has a privacy policy that specifies what data is collected and stored and where this data is stored. Also, due to users being able to upload data (in the form of text and web addresses) to the site there are terms and conditions, which users must agree to during account creation, that cover the storage of this data.

Secondly, the Copyright, Designs and Patent Act 1988 (CDPA) needs to be considered. This law relates mainly to the copyright of the works and the length of time that they last. It also covers categories and enforcement measures of copyrights, patents and trademarks. For this project's research the CDPA effects the uses of other works, such as in the literature review and the originality of this project. If any works have been used or referenced at any point in this report then they will have been appropriately referenced using the CU Harvard referencing style. There is also a statement of originality in this report that declares that the contents of the project are the original works of the author except where work is referenced. For this project's produced solution the CDPA is relevant because users are able to upload data to the site. This uploaded data should not be in violation of the CDPA and due to the website displaying this uploaded data to other users (that are within the same team as the uploading user) the CDPA is again relevant. In order to ensure compliance with the CDPA the terms and conditions of the website state that any uploaded data must not be in violation of the CDPA. There are also conditions about the license and transferring of data uploaded by users that covers the website being able to display and modify this data.

The third legal aspect that needs consideration is the Communications Act 2003. This law relates to communications that can be sent using multiple methods from telecommunication to social media. The main focus regarding this act, for this project, is that any communications should be professional and not have contents that are intended to be malicious. To attempt compliance with this act, any communication with the projects supervisor or research participant was made professionally; also, its

content had no malicious intent and was only related to subjects that would reasonably be expected. The act also applies to the produced solution due to users being able to upload text and web addresses (URLs), which may then be viewed by other users of the website. Therefore, the websites terms and conditions clearly state 18 rules for the content of data uploaded to the website by users.

Social considerations were also made when planning and completing this project. Firstly, throughout all areas of this project and report, best effort has been made to remain professional and prevent any cause of offense to any person or group. Secondly, when creating a web-interface, considerations are made to best ensure it is inclusive and accessible to all persons and groups; this includes:

- Providing alternative text when displaying images, for users that are visually impaired.
- Using gender neutral terminology and icons.
- Avoiding use of slang and abbreviations, which could be difficult to read for users that need to translate pages into a different language.

Finally, considerations have been made to ensure that the completion of this project and its results won't negatively impact any group or category of persons. The project will follow professional practises, and collect and store data as guided by the law.

6.2.2. Ethical and Professional Considerations

Ethical considerations have been made for the collecting and storing of personal information. As this project may collect this information, either in the developed solution or in the research stages of the project, the data will be stored in accordance with the law. To ensure ethical integrity is maintained, considerations will be made to ensure only necessary details are collected and subsequently stored. Also, all requested information, given tasks and asked questions will be reviewed to ensure that they can cause no offence to any individuals or groups.

Profession considerations made, include reviewing the four key principles of the British Computer Society's (BCS) code of conduct:

- Any designed solution will be accessible to everyone and considerations will be made to ensure no person or group is excluded.

- Throughout the project and this report, no competence is claimed untruthfully and professionally integrity is maintained.
- Respect is always shown to all organisations, groups and individuals during the completion of this project, and professional and ethical standards are maintained throughout.
- Considerations are made to uphold the reputation and standards of the computing profession.

6.3. Risk Management

Risk	Effect of risk on project	Risk prevention or response	Probability (Low / Medium / High)	Impact on project (Low / Medium / High)
Sample size for usability tests too small	Would make results of usability tests statistically insignificant	Request more participants than necessary in-case some are unavailable or unable to participate	Medium	Medium
Underestimation of amount of time needed to complete project work	Completion of work would fall behind schedule or project would not be completed before deadline	Mark estimates as appropriate as possible and then add an extra period of time per task in case of over-run. Also allow for "catch-up" gaps in time plan in case the project is behind schedule at any point	Low	Medium
Repetitive Strain Injury (RSI) while working on code or digital write-up	Injury to myself, would result in taking time away from completing work for the project	Task regular breaks when completing work and take extended breaks if I feel any aches or strains starting	Medium	Medium
Sudden illness preventing me from completing project work	I would be unable to complete work for the project, the project would fall behind schedule.	Should I fall ill, I will communicate with the University and will either still complete the project within the given time or apply for an extension on medical grounds	Low	High
Time for ethical approval to be given is longer than expected	Certain project work couldn't be carried out until approval was given. Project may be delayed as a result	Put application for ethical approval in at earliest possible time and allow for some work to be delayed until approval is given	Low	Medium

6.4. Record of Meetings

Date	Location	Purpose	Feedback / Actions
20/11/2019	ECB	Discuss project idea	Research gap in knowledge for the idea (a project management solution using WBS). Check for existing solutions.
14/01/2020	ECB	Discuss project proposal	Compare GitHub's current solution against "good" project management solutions from elsewhere. Suggested layout for miniature literature review. Suggested layout for background and motivation for the project. Suggested layout for primary research plan.
21/01/2020	ECB	Discuss project proposal	Some content from miniature literature review can be removed and put in the main literature review for the final project report.
28/01/2020	ECB	Discuss project proposal	Add more detail about GitHub's current management solution in the miniature literature review. Add more explanation of Kanban in the miniature literature review.
07/02/2020	ECB	Discuss ethics form	An additional sheet is required in the ethical approval application, which includes a set of sample questions to be used in the usability study.
18/02/2020	ECB	Discuss introduction of final report	Cover aims and objectives of the project. Discuss background and motivation for the project. Set out how the project will occur (structure).
25/02/2020	ECB	Discuss main literature review	Requires: Overview of project management. Consider discussion of various methodologies and their various structures.
03/03/2020	ECB	Discuss main literature review	Discuss version control and Git to give the reader a better understanding of GitHub. Include a section about the identified gap in current knowledge.
10/03/2020	ECB	Discuss main literature review	Include history of certain sections (E.g. project management and version control). Possibly include information about PRINCE.

20/03/2020	ECB	Discuss research methodology and project management sections	Research methods should include usability testing and consider using pilot testing. Discuss why these methods are being used. Project management should include the methodology being implemented during the project.
26/03/2020	Microsoft Teams	Discuss of pro-forma	Action points & feedback given in section; **6.5.**
02/04/2020	Microsoft Teams	Discuss pilot study and usability study	Pilot study results are used to refine the usability test's tasks and questions. Usability test is then conducted and results are analysed.
09/04/2020	Microsoft Teams	Discuss results analysis	Results should be written up separately (pilot study & usability study) then final overall summary of the results can go in "Data Analysis Summary"
17/04/2020	Microsoft Teams	Ask questions about contents of sections within the report	Include screenshots that are directly relevant to the text in-line, other, less relevant screenshots can go in the appendix. Conclusion can be written in 1st person and it contains a reflection of the project as a whole.

6.5. Feedback

[Feedback from project supervisor after submission of pro-forma]

For final write-up try and ensure the following;

- *Write in the third person (apart from motivation section)*
- *Avoid personal bias; no statements like 'I feel', "I think", "I believe" etc…*
- *Avoid sweeping statements without relevant facts and a suitable reference. For example, "Crime has increased across the globe."*
- *Sentences should not be over long – make them concise and easy to read. Limit the use of connectives; 'however', 'therefore' etc…*
- *Double-check spelling and grammar – ask somebody not connected with project to read it to ensure it is understandable.*
- *Make sure it flows – between sections and within sections (particularly important with literature review).*
- *Relate writing back to your project question / topic i.e. content must be relevant*

Additional points raised during the feedback discussion, while reviewing the pro-forma, were the following:

- Ensure an abstract is included in the beginning of the paper that covers what this paper is about and what the findings were
- Possible methods of proceeding with a pilot study and usability study during the COVID-19 period
- Ensure statement of originality is included
- When analysis of study results begins, ensure coverage of all feedback and comment on how results or given feedback could be used to direct future development

These raised points were considered and implemented during the development of this paper and the following actions were taken in response to the given feedback:

- All work was checked to ensure it was phrased in the third person perspective
- Opinionated and sweeping statements were avoided and references to suitable sources have been used where appropriate and/or possible
- A pilot study and usability study were conducted, to allow feedback and analysis of the created system, in an appropriate manner that took into account the situation ongoing at the time
- Results were analysed without bias, covering all feedback given

6.6. Project Management Reflection

During the process of completing this project, I have found new strengths and weaknesses in my management skills. I have also faced unexpected management challenges presented by the COVID-19 situation, which occurred during the completion of this project.

Firstly, I think the choice of development methodology gave me the flexibility to structure and develop this project in a way that suited me. The use of Feature Driven Development to create an overall model of required features and then combining this approach with Scrum iterations allowed for a structured and well-planned development process. The inclusion of Lean Software Development forced me to focus on tasks that contributed to the completion of required features and helped prevent feature creeping; This is when additional functionality is continuously added, preventing a task from ever being considered 'complete'. I did find the creation of burndown charts unhelpful and instead opted to mark on the created Gantt charts (see Section 9.7) how the actual process faired against the planned timescale, allowing me to clearly see if I was on schedule or not.

Secondly, I anticipated that my project would require a significant amount of code development, to be able to include all of the desired features, however I still underestimated this amount. The final project code source was made up of 35 files with approximately 5,600 lines of code. Planning of time management went well during development, but this section of the project was larger than I expected it to be.

Finally, I found that the database, used by the system, became far more complex than I first anticipated. The final database included 11 tables with multiple relationships and SQL features used. The normalisation process was completed without issue, but I found that I had to make decisions to prioritise making the tables concise as possible. Overall, I am pleased with the resultant database design.

In conclusion, I feel the overall management of this project went well and challenges were overcome, while still maintaining the focus of the project. I also think that I responded well to the challenges presented by the COVID-19 situation and adapted to the changing circumstances. If I were to repeat this project, I would try to maximise the amount of time allocated to code development. I would also have liked to of been able to implement changes based on the feedback from the first usability study and then complete a second study.

7. Conclusions

7.1. Summary of Results

The purpose of this project was to investigate the integration of Work Breakdown Structure (WBS) in project management of GitHub projects. To do this, a web-interface was developed that allows project management to be undertaken using the WBS, and it also offered GitHub integration tools. The results from the usability study, based on the interface, were positive, but also showed some polarised responses. Multiple feedback suggestions were also given, which could be implemented to improve the usability of the interface. Overall, the results from the research conducted in this project presented a positive response to the usability of the WBS in project management of GitHub projects.

7.2. Future Development & Research

For future development of the system made during this project, firstly a review and implementation of the suggested improvements, given in the feedback from the usability study, could be carried out. The main focus for this development iteration was the functionality of the system, so improvements to the structure of pages, to improve the usability and simplicity of the interface, could also be made. Development of the 'parts list' could be completed to maximise its use of the WBS and make it as clear and usable as possible.

For future research, a second usability study could be completed after making alterations. Also a study that compares the use of GitHub's current solution (a Kanban board) against this system, which implements the WBS, could offer insight into areas of interest.

7.3. Reflection

In conclusion, the project succeeded in investigating the usability of a project management solution based around the Work Breakdown Structure with GitHub integration, but has left further areas for development and research. The project has explored the general usability of the WBS for structuring a project and integrating it as a management tool for projects based around GitHub. Several challenges were overcome, and both the project and the researcher adapted well to obstacles raised by the unprecedented COVID-19 situation.

8. References

Abramova, V., Pires, F., and Bernardino, J. (2016) *Open Source vs Proprietary Management Tools.* Springer, Cham

Azure Boards (n.d.) *Azure Boards | Microsoft Azure* [online] available from https://azure.microsoft.com/en-gb/services/devops/boards/ [n.d.]

Azure DevOps (n.d.) *Azure DevOps Services | Microsoft Azure* [online] available from https://azure.microsoft.com/en-gb/services/devops/ [n.d.]

Bootstrap (n.d.) *Bootstrap · The most popular HTML, CSS, and JS library in the world.* [online] available from https://getbootstrap.com/ [n.d.]

Carstensen, W. (2016) *A Brief History of Version Control* [online] available from https://www.red-gate.com/blog/database-devops/history-of-version-control [7 November 2016]

GitHub (2019) *The State of the Octoverse* [online] available from https://octoverse.github.com/ [30th September 2019]

Microsoft Excel (n.d.) *Microsoft Excel, Spreadsheet Software* [online] available from https://www.microsoft.com/en-gb/microsoft-365/excel [n.d.]

Niazi, M., Mahmood, S., Alshayeb, M., Riaz, M., Faisal, K., Cerpa, N., Khan, S., and Richardson, I. (2016) *Challenges of project management in global software development: A client-vendor analysis.* CrossMark

Nicholas, J. and Steyn, H. (2017) *Project Management for Engineering, Business and Technology* 5th edn. London & New York: Routledge, 7-8

Nielsen, J. (1989) *Usability engineering at a discount*

Nielsen, J. (2000) *Why You Only Need to Test with 5 Users* [online] available from https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/ [18 May 2000]

Nielsen, J. (2012) *Usability 101: Introduction to Usability* [online] available from https://www.nngroup.com/articles/usability-101-introduction-to-usability/ [3 January 2012]

OpenProject (n.d.) *OpenProject - online project management software - free and open source* [online] available from https://www.openproject.org/ [n.d.]

Open Web Application Security Project (n.d.) *Cross Site Request Forgery (CSRF)* [online] available from https://owasp.org/www-community/attacks/csrf [n.d.]

Oxford English Dictionary (n.d.) *project management, n.* [online] available from https://oed.com/view/Entry/267391 [n.d.]

Poppendieck, M. and Poppendieck, T. (2003) *Lean Software Development: An Agile Toolkit* Boston: Addison Wesley

Powell, D. (2018) *Kanban for Lean Production in High Mix, Low Volume Environments*

Pressman, Roger S. (2010) *Software Engineering – A Practitioner's Approach* 7th edn. McGraw-Hill, 39-41

Pressman, Roger S. (2010) *Software Engineering – A Practitioner's Approach* 7th edn. McGraw-Hill, 68-70

PRINCE2 (2020) *PRINCE2 Project Management Certification* [online] available from https://www.prince2.com/uk [2020]

Project Management Institute (2008) *A Guide to the Project Management Body of Knowledge* PMBOK Guide – 4th Edn. Project Management Institute Inc, 116

Project Manager (2018) – Westland, J. *History of Project Management* [online] available from https://www.projectmanager.com/blog/history-project-management [24 May 2018]

Ross, J. (2016) *Conduct a Pilot Test First* [online] available from https://www.infragistics.com/community/blogs/b/ux/posts/conduct-a-pilot-test-first [26 September 2016]

Takeuchi, H. and Nonaka, I. (1986) *The new new product development game*

Wintergatan – Youtube (n.d.) *Wintergatan – YouTube* [online] available from https://www.youtube.com/user/wintergatan2000 [n.d.]

9. Appendices

9.1. Detailed Project Proposal

Topic for Investigation (Research Question / Problem Statement)

I will be investigating integration of project management into GitHub.

GitHub's current system makes use of Kanban boards. While this does allow for some project management, it can be difficult to structure large projects that often contain tasks with sub-tasks. The nature of column limiting (discussed in the initial literature review) means a need for having many individual cards for each subtask. This can then be difficult to organise within GitHub's Kanban board system and visualising the structure of the project then becomes difficult.

A different method for structuring large projects could be to use the Project Breakdown Structure (PBS – also known as Work Breakdown Structure (WBS)), which lists tasks and then the sub-tasks that make up each task. Tasks can be nested several times within this structure and a general rule of thumb is that a task, that has no sub-task, should require no more than 80 hours of work. This structure can be used within multiple project management methods (E.g. Agile, Waterfall, Etc…) similarly to Kanban boards.

This can be achieved by taking existing project management methods and integrating them with GitHub through a web-interface. A stand-alone website will be developed which includes project management tools that can be integrated into a user's GitHub repository via the GitHub API. The aim of this is to allow users to have a centralised place to plan, structure and track project progression.

Initial Literature Review

The Oxford English Dictionary (n.d.) define project management as "The theory, practice, or occupation of managing projects". This can include planning and controlling a team to ensure that project requirements are met.

Nicholas and Steyn (2017) discuss the need for project management and how modern projects often involve technical complexity and challenges. They also discuss how project management is used, when businesses give their projects constrained resources and limited time, to meet the projects requirements.

Due to these constrained resources, many companies have taken on the approach of Global Software Development (GSD), which is when software development occurs by multiple teams based in various geographical locations. Niazi et al. (2016) analysed the various challenges of managing these GSD projects and found 19 challenges that cause issues in project management. One of the most common challenges was lack of communication, which then gives rise to lack of understanding requirements. Communication between teams is often through email or messenger applications due to the different time zones. Another challenge found was lack of co-ordination or allocation of tasks, which may be attributed to a lack of communication or use of poor systems for communication.

Software development projects often consist of multiple programs which may be developed by one or multiple programmers. Therefore, there is a requirement for these changes to be tracked so that developers can go back to previous versions of code if errors are made. This is often handled by a version control system.

Git is a version control system (VCS) which allows developers to keep a history of all changes made to files within a repository (a project), but also allows for multiple developers to work on project code at the same time and then have a centralised storage system (E.g. GitHub).

GitHub is a service that offers Git repository hosting and is widely used by developers, both inside and outside of business. GitHub is used by over 40 million developers and is hosting over 44 million repositories, with enterprise users in organisations from over 70 countries (GitHub 2019). GitHub delivers this service through a website that allows developers to view and manage their repositories. However, GitHub also offers other features to its users such as Wiki's, which can be used to document software, web-interface for managing issues and pull requests (features of Git) and project boards, which are used to manage the project. The GitHub project boards feature makes use of Kanban boards to allow users to list tasks and sort them (E.g. sort them according to status; 'To Do', 'In Progress' and 'Complete').

Kanban is the process of regulating the supply of components using a card or board.

A Kanban board is a table that is used within the Kanban methodology. Each column contains tasks at different stages (E.g. 'Backlog', 'To-do', 'Doing' and finally 'Done') and these tasks are moved across the board by team members as work is completed. Powell (2018) discusses how Kanban boards are used during daily scrum meetings to manage what was done the previous day and then choose what tasks will be worked on that day. Also discussed is how each column, except the backlog, can be limited to only contain a certain number of tasks to ensure they get completed. Powell concludes that Kanban boards can be useful tools within high-mix, low-volume environments (E.g. software development) as it creates a pull mechanism for tasks.

GitHub currently uses Kanban boards as it's project management feature however, another method of displaying tasks, that GitHub does not currently use, is Work Breakdown Structure.

The Project Management Institute (2008) describe Work Breakdown Structure (WBS) as a deliverable-oriented hierarchical structure of a project that states all tasks that need to be completed for the required deliverables to be created. Each level of the structure (going down the hierarchy) is a more detailed breakdown of the level it's linked to above.

This research project will implement the methods of the Work Breakdown Structure and integrate them with GitHub repositories.

Client, Audience or Motivation

My motivation for this project idea was a Youtuber called "Wintergatan" (Wintergatan – YouTube n.d.), who is filming his own complex engineering project; the creation of a marble machine that plays music. In one of his videos, he discussed how it was difficult to manage a project with so many parts and assemblies. He then works with a professional project manager that shows him an excel spreadsheet system that implements Project Breakdown Structure (PBS) also known as Work Breakdown Structure (WBS). I liked this method of project management because I found it

much simpler to use when writing down my plan for complex coding projects. It is also a deliverable-oriented system which means that I can specify what I need each part to be without having to precisely calculate how it will be done, which I find leaves far more flexibility when coding. Over the time that I've been using PBS, I've found several, paid subscription, sites online that use this system but don't offer integration with GitHub. So, I now want to develop a site that allows me to manage my project online and manage my GitHub repositories all from the same place.

This system would benefit anyone who required project management software but, more specifically, developers that need to manage projects that incorporate GitHub services. This means that a potential client for my project would be software developers that use GitHub to host their Git repositories.

The main benefit of this system is a more feature-rich and flexible management solution that integrates with a version control system.

Primary Research Plan

- **Initial Research**
  Research state of the art within this area. Areas to include: project management, project management processes (E.g. Kanban). Week Commencing: 13/01/2020

- **Compare Currently Existing Systems**
  Compare the system that is currently in use by GitHub against other project management tools that are also available. Week Commencing: 03/02/2020

- **Create Requirements Definition**
  Writeup requirements for the project based on findings from literature review. Week Commencing: 10/02/2020

- **Software Development**
  Develop the web-interface and it's required database. Week Commencing: 17/02/2020

- **Evaluation**
  Evaluate the web-interface through the use of usability testing, comparing the newly created solution against the project management solution included in GitHub. Week Commencing: 9/03/2020

- **Analysis and Writeup**
  Analyse results from usability tests and writeup results and findings. Week Commencing: 16/03/2020

Intended Project Outcome

The intended outcome of my project is to have a fully functional website that implements project management features which allow users to plan, structure and track development of a project which links to a GitHub repository.

These project management features should allow a user to manage their project using a selection of management methods, while being capable of displaying complex project structures though the Project Breakdown Structure (PBS/WBS).

This website should also have a range of features that integrate GitHub into the management process. These should include allowing files, hosted on GitHub, to be linked to individual tasks, linking pull request to tasks and tracking any issues raised on the project repository and which files they're associated with.

To evaluate the web-interface, usability testing should be undertaken to compare usage of this project's web-interface against the solution currently used by GitHub, their Kanban boards.

Bibliography

GitHub *The State of the Octoverse* [online] available from https://octoverse.github.com/ [30th September 2019]

Niazi, M., Mahmood, S., Alshayeb, M., Riaz, M., Faisal, K., Cerpa, N., Khan, S., and Richardson, I. (2016) *Challenges of project management in global software development: A client-vendor analysis.* CrossMark

Nicholas, J. and Steyn, H. (2017) *Project Management for Engineering, Business and Technology* 5th edn. London & New York: Routledge, 7-8

Oxford English Dictionary (n.d.) *project management, n.* [online] available from https://oed.com/view/Entry/267391 [n.d.]

Powell, D. (2018) *Kanban for Lean Production in High Mix, Low Volume Environments*

Project Management Institute (2008) *A Guide to the Project Management Body of Knowledge* PMBOK Guide – 4th Edn. Project Management Institute Inc, 116

Wintergatan – Youtube (n.d.) *Wintergatan – YouTube* [online] available from https://www.youtube.com/user/wintergatan2000 [n.d.]

9.2. Pilot Test

This usability test is to investigate the usability of a web-interface designed to allow project management of GitHub projects using a hierarchical approach.

You will be accessing this web-interface through a live version located at;

[redacted]

Your login details are:

Username: [redacted]

Password: [redacted]

Please complete the following tasks in the order they are given;

1. Login to [redacted] with the credentials provided above
2. Navigate to the **systems-design** project
3. Navigate to the parts list
4. Add the **Web** category to the part called **Create login page**
5. Assign the part called **Create logout page** to the user **john36**
6. Alter the project settings to allow each part to be given a version number
7. Mark the project as finished by setting it to **Closed** (The current project status is shown to the right of the **Project Stats** heading)

8. Create a new project that imports the details from the "**pmdwalshtester/usability-test**" GitHub repository.
9. Invite the user with the username **john36** to join your project as a user that can edit on the project.
10. Create the first part with the following details;
    a. Name: Create Website
    b. Dates: 10/04/2020 – 15/05/2020
    c. Status: Unassigned
    d. Priority: Normal
11. On the part you just created, create a new issue with the title **Login page needed** [Depending on your connection, you may need to refresh the page to see the newly created issue]

Thank you for your assistance in this usability test, could you please now answer some quick questions about your experience and return this document to the researcher.

Usability Evaluation

Please rate the following features of the web-interface (1 = Bad – 10 = Good):

Feature	Rating
Overall ease of use	
Overall layout	
Part List layout	
Individual part layout	
Project creation	

Were you able to complete all of the tasks? If no, please specify which task numbers you couldn't complete:

What do you think of the part list layout compared to a Kanban board layout:

Were there any features you found difficult to find/access/use:

9.3. Participant Information Sheet

PARTICIPANT INFORMATION SHEET

You are being invited to take part in research on integration of project management into GitHub. Daniel Walsh, a student at Coventry University is leading this research. Before you decide to take part, it is important you understand why the research is being conducted and what it will involve. Please take time to read the following information carefully.

What is the purpose of the study?

The purpose of the study is to investigate and evaluate a project management system that has GitHub integration.

What are the benefits of taking part?

By sharing your experiences with us, you will be helping Daniel Walsh and Coventry University to better understand the effects of project management integration with version control tools.

Are there any risks associated with taking part?

This study has been reviewed and approved through Coventry University's formal research ethics procedure. There are no significant risks associated with participation.

Do I have to take part?

No – it is entirely up to you. If you do decide to take part, please keep this Information Sheet and complete the Informed Consent Form to show that you understand your rights in relation to the research, and that you are happy to participate. Please note down your participant number (which is on the Consent Form) and provide this to the lead researcher if you seek to withdraw from the study at a later date. You are free to withdraw your information from the project data set at any time until the data is fully anonymised in our records on 14/03/2020. You should note that your data may be used in the production of formal research outputs (e.g. journal articles, conference papers, theses and reports) prior to this date and so you are advised to contact the university at the earliest opportunity should you wish to withdraw from the study. To withdraw, please contact the lead researcher (contact details are provided below). Please also contact the Research Support Office (ethics.eec@coventry.ac.uk) so that your request can be dealt with promptly in the event of the lead researcher's absence. You do not need to give a reason. A decision to withdraw, or not to take part, will not affect you in any way.

What will happen if I decide to take part?

You will be asked to complete a number of tasks using a new project management web-interface and then you will be asked a number of questions regarding your experience with the new project management system. The usability test will take place in a safe environment at a time that is convenient to you. Ideally, we would like to screen record your usage of the web-interface, so a computer will be provided, and the usability test will take place in a fairly quiet area. The usability test should take around 15 minutes to complete.

Data Protection and Confidentiality

Your data will be processed in accordance with the General Data Protection Regulation 2016 (GDPR) and the Data Protection Act 2018. All information collected about you will be kept strictly confidential. Unless they are fully anonymised in our records, your data will be referred to by a unique participant number rather than by name. If you consent to being audio recorded, all recordings will be destroyed once they have been transcribed. Your data will only be viewed by the researcher/research team. All electronic data will be stored on a password-protected computer file on Coventry University's OneDrive. All paper records will be stored

in a locked filing cabinet within Coventry University. Your consent information will be kept separately from your responses in order to minimise risk in the event of a data breach. The lead researcher will take responsibility for data destruction and all collected data will be destroyed on or before 16/03/2020.

Data Protection Rights

Coventry University is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance with the General Data Protection Regulation and the Data Protection Act 2018. You also have other rights including rights of correction, erasure, objection, and data portability. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to the University Data Protection Officer – enquiry.ipu@coventry.ac.uk

What will happen with the results of this study?

The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will always be made anonymous in any formal outputs unless we have your prior and explicit written permission to attribute them to you by name.

Making a Complaint

If you are unhappy with any aspect of this research, please first contact the lead researcher, Daniel Walsh. If you still have concerns and wish to make a formal complaint, please write to Amanda Brooks:

Amanda Brooks

Assistant Lecturer in Computing and Computer Science

Coventry University

Coventry CV1 5FB

Email: ab3728@coventry.ac.uk

In your letter please provide information about the research project, specify the name of the researcher and detail the nature of your complaint.

9.4. Consent Form

Participant No.

INFORMED CONSENT FORM:
An investigation on the Integration of Project Management into GitHub

You are invited to take part in this research study for the purpose of collecting data on the integration of project management into GitHub.

Before you decide to take part, you must **read the accompanying Participant Information Sheet.**

Please do not hesitate to ask questions if anything is unclear or if you would like more information about any aspect of this research. It is important that you feel able to take the necessary time to decide whether or not you wish to take part.

If you are happy to participate, please confirm your consent by circling YES against each of the below statements and then signing and dating the form as participant.

1	**I confirm that I have read and understood the Participant Information Sheet for the above study and have had the opportunity to ask questions**	**YES**	**NO**
2	**I understand my participation is voluntary and that I am free to withdraw my data, without giving a reason, by contacting the lead researcher and the Research Support Office at any time until the date specified in the Participant Information Sheet**	**YES**	**NO**
3	**I have noted down my participant number (top left of this Consent Form) which may be required by the lead researcher if I wish to withdraw from the study**	**YES**	**NO**
4	**I understand that all the information I provide will be held securely and treated confidentially**	**YES**	**NO**
5	**I am happy for the information I provide to be used (anonymously) in academic papers and other formal research outputs**	**YES**	**NO**
6	**I am happy for the interview to be audio recorded**	**YES**	**NO**
7	**I agree to take part in the above study**	**YES**	**NO**

Thank you for your participation in this study. Your help is very much appreciated.

Participant's Name	**Date**	**Signature**
Researcher	**Date**	**Signature**

9.5. Usability Test

This usability test is to investigate the usability of a web-interface designed to allow project management of GitHub projects using a hierarchical approach. GitHub is used to store and manage versions of code during software development and the interface you will be testing has been designed to allow developers to manage their project and tasks needed to complete the project.

You will be accessing this web-interface through a live version located at;

[redacted]

Your login details are:

Username: [redacted]
Password: [redacted]

Consent:

By returning this form to the researcher, you confirm that you are agreeing to your answers being used in the research. No identifiable information is being collected.

Please complete the following tasks in the order they are given;

1. Login to [redacted] with the credentials provided above
2. Navigate to the **systems-design** project
3. Navigate to the parts list (all parts)
4. View the part called **"Create login page"**
5. Add the **Web** category to the category section of the part called "**Create login page"**
6. Navigate back to the parts list and then view the part called **"Create logout page"**
7. Assign the part called "**Create logout page"** to the user **john36**
8. Navigate to the project overview and alter the project settings to allow each part to be given a version number
9. Mark the project as finished by setting it to **Closed** (If completed correctly, the word "Closed" should be shown to the right of the Project Stats heading)

Please navigate back to the **Home** page and then continue with the tasks below;

12. Create a new project (new GitHub project) that imports the details from the "**pmdwalshtester/usability-test**" GitHub repository.
13. Invite the user with the username **john36** to join your project as a user that can edit on the project.
14. Create the first part with the following details;
    a. Name: Create Website
    b. Dates: 10/04/2020 – 15/05/2020
    c. Status: Unassigned
    d. Priority: Normal
15. On the part you just created, create a new issue with the title **Login page needed** [Depending on your connection, you may need to refresh the page to see the newly created issue]

Thank you for your assistance in this usability test, could you please now answer some quick questions about your experience and return this document to the researcher.

Usability Evaluation

Please rate the following features of the web-interface (Strongly Disagree – Strongly Agree):

Feature	Rating
I found navigating to the project easy to do	Choose an item.
I found it easy to view individual parts	Choose an item.
I found the category system easy to use	Choose an item.
I found the assigning system easy to use	Choose an item.
I found project settings easy to use	Choose an item.
I found creating a project easy	Choose an item.
I found the member management system easy to use	Choose an item.
I found part creation easy	Choose an item.
I found issue creation easy	Choose an item.

If you chose "strongly disagree" or "disagree" above, please specify why:

Were you able to complete all the tasks? If no, please specify which task numbers you couldn't complete and why:

Were there any features you found difficult to find/access/use, if so please say why:

What did you think of the overall layout of the interface? (E.g. Were controls/menus/etc where you expected to find them?)

What did you think of the overall design of the interface? (E.g. Was content readable? Were there any colours or fonts that cause issues for you?)

9.6. Pro-forma

300COM/303COM Project Progress Report Pro-forma

An Investigation on The Integration of Project Management Into GitHub

The primary element of my project is the development of a web based project management solution. My current progress on this solution is good and it is near completion. The website has functioning login systems, project and task creation and has some of the basic GitHub integration implemented. I still need to add some more advanced GitHub integration features such as creation of issues.

REPORT SECTION

Section	% Complete
INTRODUCTION	90
LITERATURE REVIEW	90
DESCRIPTION OF PRIMARY METHOD	20
RESULTS / FINDINGS	0
EVALUATION/INTERPRETATION OF RESULTS	0
PROJECT MANAGEMENT CHAPTER	80
CONCLUSIONS / RECOMMENDATIONS / DISCUSSION OF FUTURE WORK	0
REFERENCES	N/A (Added as needed)

Questions for supervisor:

- What should be covered in the “Background and Context” section of the introduction?
- What should be covered in the “Experimental Analysis” section of the research methodology?

9.7. Gantt Chart

Figure 9 shows the Gantt chart for the writing of this report. The blue bars represent the time allocation that was planned at the beginning of the project and the orange bars represent how the time was actually used.

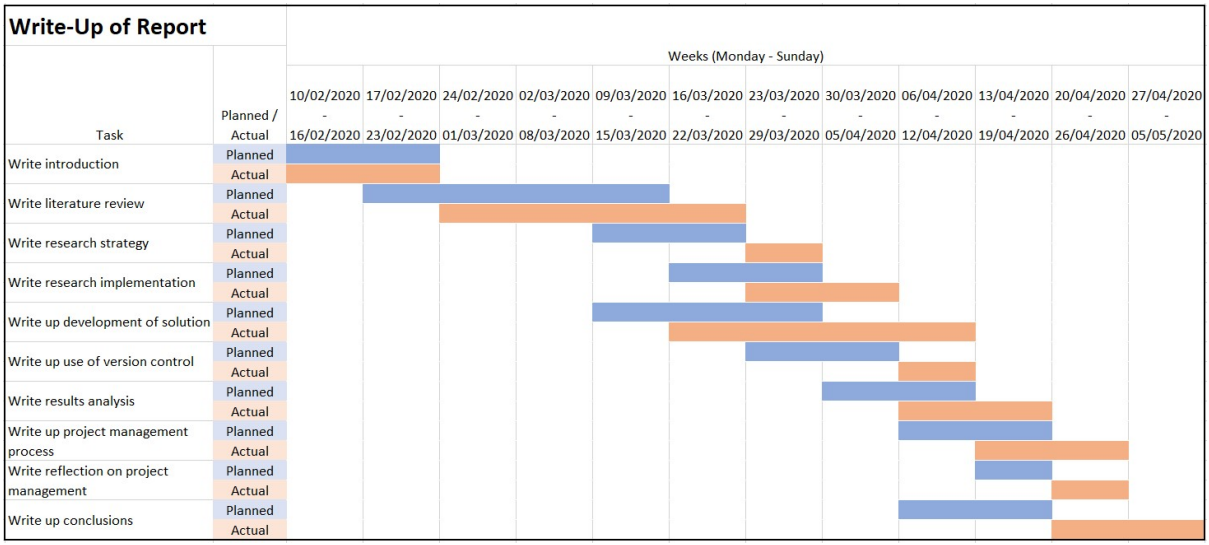

Figure 9 – Gantt Chart for Report Writing

Figure 10 shows the Gantt chart for the development of the software solution. This development was conducted alongside the writing of the final report. Again, the blue bars represent the time allocation that was planned at the beginning of the project and the orange bars represent how the time was actually used.

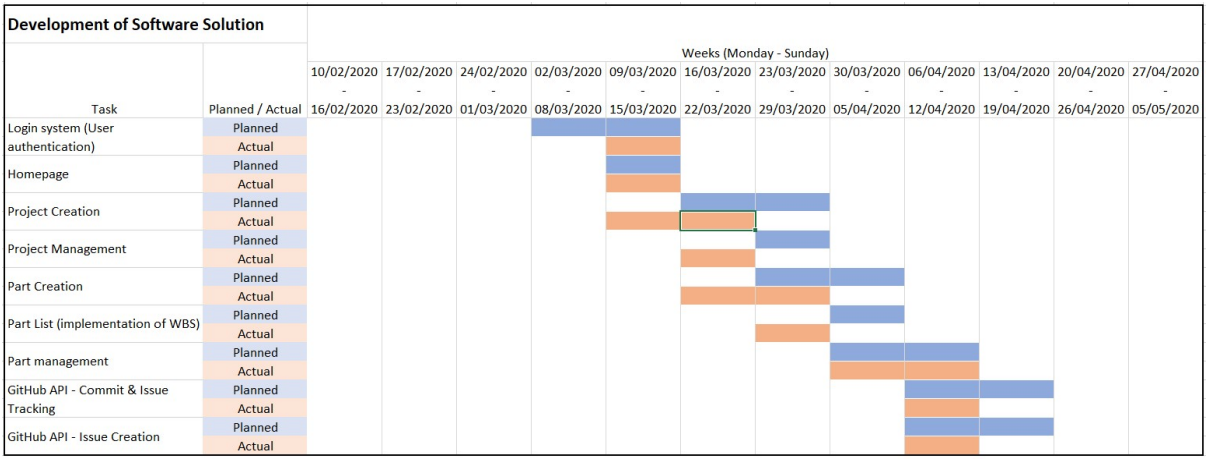

Figure 10 – Gantt Chart for the Development of the Software Solution

9.8. Ethical Approval Certificate

Certificate of Ethical Approval

Applicant:

Daniel Walsh

Project Title:

An investigation on the integration of project management into GitHub

This is to certify that the above named applicant has completed the Coventry University Ethical Approval process and their project has been confirmed and approved as Medium Risk

Date of approval:

19 February 2020

Project Reference Number:

P102592

9.9. GitHub Repository

https://github.coventry.ac.uk/walshd9/303COM-Project-Management

9.10. List of Figures

Figure 1 – Work Breakdown Structure in Microsoft Excel 13

Figure 2 – Work Breakdown Structure in OpenProject 14

Figure 3 – Kanban Board in Microsoft Azure Boards 14

Figure 4 – Project Creation Page 20

Figure 5 – Part Creation Page 22

Figure 6 – Parts List Page 23

Figure 7 – Database Structure Diagram 25

Figure 8 – Issue Creation Modal 27

Figure 9 – Gantt Chart for Report Writing 59

Figure 10 – Gantt Chart for the Development of the Software Solution 59