Student Name: Alan Heslop

Student Number: 199302280 (bh83dl)

Programme: MSc Digital and Technology Solutions (Software Engineering)

Project Title**:** Software Engineering Aspects of Continous Development and the Effect within Project Teams

Project Title**:** Software Engineering Aspects Using Version Control Systems such as GitHub and the Collaborative Effect this has on Project Team Members.

Client/Sponsor: Angus Greenland (DXC)

Supervisor (if known): Gavin McClary

## Abstract

*The current approach for junior members of the team, which require code reviews or have questions is to book an allocated amount of time with a senior member. Booking time to deal with basic code quality reviews can reduce the output of the project due to the senior member being unavailable for meetings, and their commitment is divided up between meetings. Using an application that can provide a response that guides you to a better understanding of the issue, delivers empowerment, and provides Continuous Development with each action to the repository. From this project, junior team members would be able to progress and learn through their own momentum. An overall improvement in the production of quality of code, self-development, and team morale." Will providing a junior team member with the tools for self-learning improve Continuous Development?*

## Aim & Objectives

The aim of this project is: To design, develop and implement a Proof of Concept (PoC) application that allows less-experienced members of the team to use continuous integration (GitHub/githooks/AWS/code editor etc/cloud watch) in a form of set driven commands to automate the checking of any commits, using client-side hooks to a GitHub repository. This is part of a wider approach to Continuous Development. The use of this is to save time for the senior member needing to check the commit for validation, secrets not hidden, or overall not a clean build that may affect any merge. In addition, less-experienced members of the team will self learn by reviewing the feedback from errors.

The return should be presented to the user in a well-known instant messaging application (Slack) using the ideals of ChatOps mixed with DevOps, with the potential for Microsoft Teams integration in the future.

The objectives to achieve this aim are:

1. Implement a working PoC utilising concurrent technologies.
2. Design and implement ChatOps processes.
3. Design and Implement DevOps processes.
4. Provide users with an up-to-date process of use.
5. Create a fresh user interface for the users to respond to actions.
6. Design architecture and implementation on AWS.

* Does introducing digital learning improve pupil enjoyment?
* Does introducing digital learning as a method of assessing achievement of success criteria improve pupil enjoyment in S1 science?

Can we measure?

* Speed
* Scalability
* Cost

Does introducing continuous integration as a method of development

## Research

The research question is: *“Can the implementation and practical use of client and server-side GitHooks fundamentally develop a DevOPs junior member’s knowledge? Also, can the use of Continuous Integration/Continuous Delivery (CI/CD) improve the collaboration within teams?”*

Research areas of interest include Instant messaging, GitHub, cloud development tools, Advantages and disadvantages of using this method.

The goal of reviewing the literature on **code quality using automation for code review** is to identify the advantages and disadvantages of using this method, highlighting the importance of a newcomer’s experience and guiding them through this process of code review.

The practical application of the research is expected to includethe best approach to developing a workable PoC which utilises GitHooks, GitHub, a nominated chat client (Slack) and AWS services. The practical element will enable an investigation into the effectiveness of DevOps, processes and the tools for learning opportunities.

## Practical Element

The practical element will involve planning, designing, developing, and testing a PoC of a “chatbot” using ChatOps methods and DevOps procedures utilising a cloud-based environment and will demonstrate collaborative and learning improvements throughout.

Implementation of an Agile/Scrum methodology will be approached due to the nature of Continuous Development as it requires an accurate and efficient system for planning and tracking tasks and sprints with the team development.

The practical element will be developed using approaches such as client-side githooks, GitHub repositories, and a cloud computing aspect for developing the chatbot. Primarily, the language chosen will be Python and a GIT CLI will need to be configured.

The practical outcome that this project aims to provide is: A workable PoC that the client can provide to new recruits/less experienced team members as part of a training package for continuous integration and DevOps.

The sponsor/client will use it to provide an introduction at a low level to GitHub/git commits and how to troubleshoot any validation/formatting errors.

To evaluate the success of the practical outcome I would require various team members who are less experienced to run through the installation/guidelines of this application. This will help improve the application based on multiple users’ interactions.

## Learning and Development

This project involves knowledge and skills including software development, cloud technologies, microservices, GitHub, continuous integration, and monitoring.

Areas for development include Cloud Technology, Software Development, Agile Project Management, and feedback from less experienced team members will be improved through engaging in the research and practical aspect of the project lifecycle.

## Research Ethics

This research does not involve human participants and does not need ethical approval from the University of Sunderland Research Ethics Committee

# References

*Bibliography*

1. DEV Community. 2019. *5 ways to create a junior developer-friendly culture*. [online] Available at: <https://dev.to/httpspauline/5-ways-to-create-a-junior-developer-friendly-culture-3n4> [Accessed 16 February 2022].
2. Atlassian, n.d. *Git Hooks | Atlassian Git Tutorial*. [online] Atlassian. Available at: <https://www.atlassian.com/git/tutorials/git-hooks> [Accessed 16 February 2022].

# Evidence / Appendices

## Letter/Email of support from Client / Sponsor

*Guidance: short letter from your sponsor client. It can be as simple as:*

I can confirm that I support Alan Heslop’s project proposal “Software Engineering Aspects of Continous Development and the Effect within Project Teams” for their MSc project as detailed in the Client-Sponsor MSc Project Proposal Template.

|  |
| --- |
|  |
| Angus Greenland  (Project Client)  Date: 15/02/2022 |

## Supervisor (if applicable)

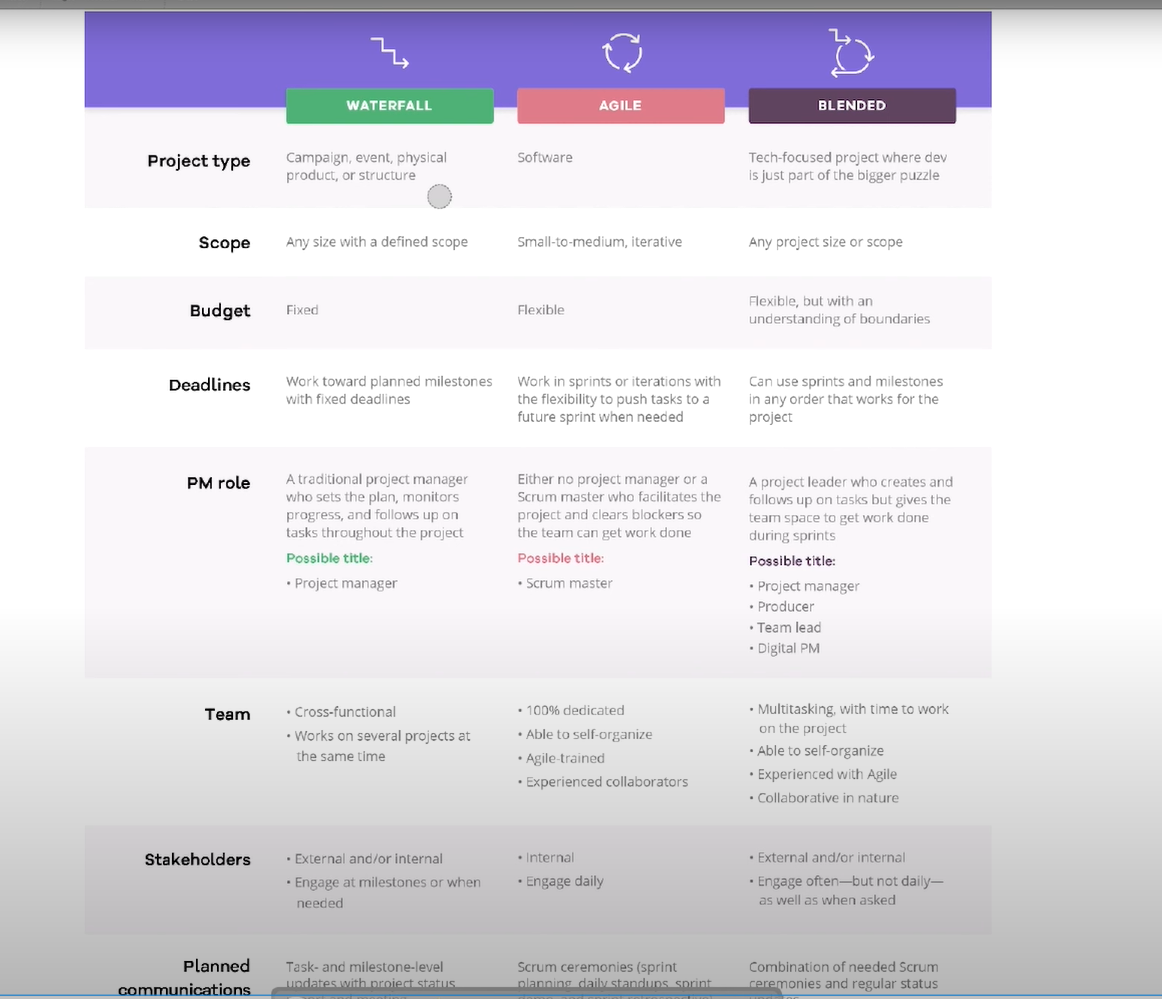
*Guidance: short email of support from a supervisor*

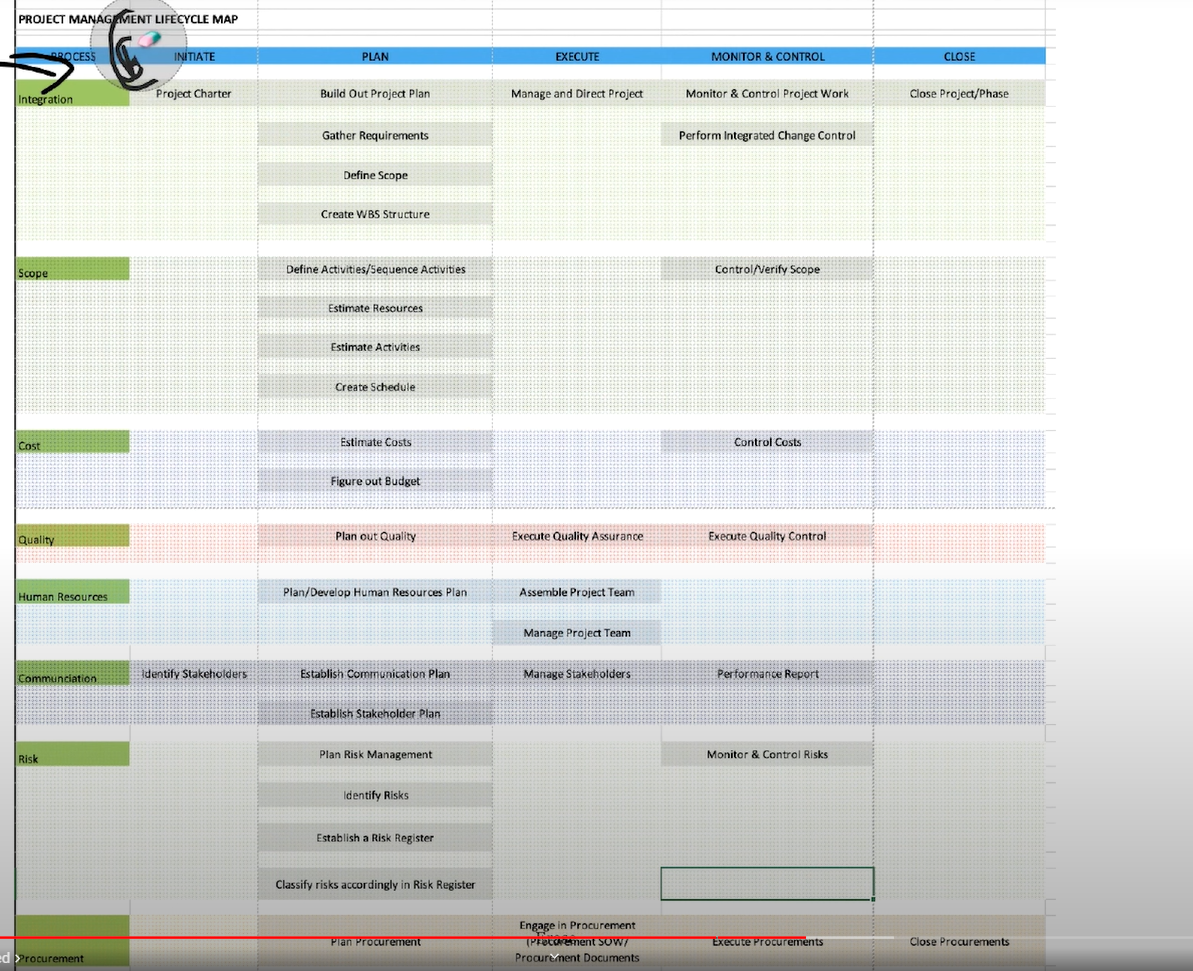
I can confirm that I have agreed to be Alan Heslop’s supervisor for their MSc project

|  |
| --- |
|  |
| Gavin McClary  (Project Supervisor)  Date: \_\_\_\_\_\_\_\_\_\_\_\_\_ |

Table of reference

|  |  |  |  |
| --- | --- | --- | --- |
| Appendix | | Item | What is it? |
| 1 |  | | Project Plan |
| 1 |  | | Project Charter |
| 2 |  | | Stakeholder register |
| 3 |  | | Stakeholder Communications Plan |
| 4 |  | | Power V Interest |
| 5 |  | | Work Breakdown Structure. |
| 6 |  | | Project Scope |
| 7 |  | | RACI Chart |
|  |  | | GANTT Chart |
|  | Delete this after, but WBS will be the risk register | | Risk Register |
| 8 | Graphical user interface, text, application, email  Description automatically generated | | Sprint Plan - JIRA |

**

**

*https://www.youtube.com/watch?v=ABPqUjyvapU*