Introduction

This is a report whereby I will track my learning of a particular computing tool. I will be discussing the steps I undertook to learn the given tool, and my progression of knowledge and application of the tool.

Furthermore, I will investigate the extent of the usefullness of the tool within a non-computing domain of application, focussing on the areas in which it can benefit the given domain, and areas in which it could present as a potential drawback over other tools, practices or methods.

Jira

Jira is an agile project management platform developed by the Australian company *Atlassian* in 2002. Described as a "flexible issue tracking tool that helps teams plan, manage, and report on their work" (Cprime, 2021), Jira was originally targeted at software developers for use in issue management. But has since been widely adopted by a wider range of computing domains and non-computing domains (for example, in large corporations such as Qantas). However, it still remains most prevalent in the Software Development domain (which is where it is most applicable to myself).

Justification

As I have outlined in my **Self-Assessment** and my **Self Learning Proposal**, my biggest weaknesses relevant to the professional computing industry are my poor written/typed communication, lack of ability to multitask and lack of programming skills.

Jira, being a project management tool, should allow me plenty of opportunity to work on improving my communication skills, as I work towards learning how it allows for ease of communication between computing teams.
Furthermore, in learning how to effectively use Jira, I will hopefully be able to improve my multitasking skills. Through being able to plan projects and concurrent tasks within those given projects (which will allow me to better manage my time, making it easier to work on multiple tasks at once). Finally Jira should be able to provide me with more programming experience through the use of Git repositories for Python code. Being relatively inexperienced with Python (and just programming in general), my debugging skills are rather poor. Thus as I begin to learn Jira, interpreting the depth of its efficiency to aid computing teams in Issue Management, I should be able to get more hands-on experience with debugging. Bringing my overall programming skills to a higher level than I would otherwise be at.

In order for me to best teach myself how to efficiently use Jira and improve these skills, I will need to have some basic understanding of programming and the use of Git. Since Jira revolves around the movement of code between team members in a computing environment, it is vital that I have at least a basic understanding of programming and how it works. Moreover, it is also greatly important that I be familiar with Git (the creation of Git repositories, and the use of Github to push and pull files from/to repositories), as it is the primary tool used to share code through Jira. Fortunately, as I do have basic programming skills and understanding of Git, I should be able to effectively teach myself Jira, and thus improve on my skills that have been marked for improvment.

Resource Search

In order to teach myself how to utilize Jira to a high standard, I will need to view and interpret various resources. In terms of learning Jira, there are a vast array of resources, spanning from articles, blog posts, websites, youtube videos (in particular the videos created by *Atlassian* on their channel), and scholarly articles (through Google Scholar). This depth of resources will likely be sufficient for me to potentially teach myself Jira in detail and thus demonstrate such subsequent knowledge.

Furthermore, as stated in my **Self Learning Proposal**, the *Level 1 - Basic Application* of my learning of Jira will cover a lot of the general functions and use of the tool. However, further application of my knowledge will require a understanding of how to ***effectively*** use Jira. Which therefore means that there is a large scope for high level understanding of this tool, should I use appropriate resources.

Logistics - Staff Management and Team Communication

The domain of application that I have to chosen to represent is Logistics. I feel that Jira, while still being primarily targeted towards Issue Management within Software Development, has great potential to be used in the non-computing domain of Logistics, and in particular, Staff Management and Team Communication. Logistics, in my case here, is referring to a given corporation's ability to effectively manage teams and staff members and to provide clear chains of command that allow for ease of communication.

Jira will be specifically applicable to Logistics since it allows for such management of teams and staff members, and efficient facilitation of clear chains of command. Moreover, it has the potential to greatly improve team communication and overall communication at all levels within any large corporation.

> Products and apps built on top of the Jira platform help teams plan, assign, track, report, and manage work. The Jira platform brings teams together for everything from agile software development and customer support to start-ups and enterprises
>
> — (Atlassian, 2021)

To this extent, in this report I will discuss the extent to which Jira can benefit Logistics within large corporations, and the potential limits Jira may have in maximizing communication and efficiency.

References

1. Atlassian. (2021). Jira Overview | Products, Projects and Hosting. Retrieved March 27, 2021, from Atlassian website: https://www.atlassian.com/software/jira/guides/getting-started/overview

2. Cprime. (2021). What is Jira? Learn about the tool to plan, manage and report by Atlassian. Retrieved March 27, 2021, from Cprime website: https://www.cprime.com/what-is-jira/