Analysis of Alternatives

The main criteria for the programming language is that all of the team members must at least have a basic understanding of it and are comfortable with that language. There are multiple alternatives for programming languages such as java, python, html etc. Considering html or javascript might be the best choice among all the programming languages for a web-based application. However, after some discussions, we realized that most of us do not have a strong foundation for html or javascript due to some of us being from Computer Science courses and some of us from Software Engineering courses but most of us are confident in our python programming language which we had all learnt from some of the units. Hence, python was ultimately chosen as the programming language for this project as all the team members are familiar and comfortable with python.

Following that, our team had to perform some research for some platform or architecture that supports web development using python. The criterias that we mainly focus on are the user-friendliness, libraries and the variety of functionalities of the platform. Since we are building a MVC model, we will need to consider all 3 aspects of the model which are the Model, View and Controller. After some research, we found that Django, Flask and CherryPy might be our possible options as all of them are python-supported web development frameworks.

Firstly, we made a comparison between Django and Flask. The main differences between flask and django is that Flask does not support dynamic HTML pages and Django offers dynamic HTML pages. Moreover, Flask is built for rapid development while Django is for easy and simple project development. From this, our team has decided to use Django as dynamic HTML pages are essential for our project.

Moving on, Django is compared with CherryPy. CherryPy is a flexible built-in plugin system which offers simplicity for running multiple HTTP servers simultaneously with a powerful configuration system while Django has a plethora of ready-to- use libraries that the users can utilise straightaway. From this, we know that CherryPy is more flexible compared to Django because the user can import a high variety of libraries but it is also more time consuming to find a suitable library that the user wants but Django has already provided them which makes it more time-efficient.

Hence, Django has been chosen as the final platform that our team will implement for this project due to multiple criterias that we have considered.