Analysis of Alternatives: Programming Language

# Glossary/Reference

* JavaScript: a client-side language, and refers to HTML/CSS as well in this document
* Flask: a micro-framework for web apps based on Python
* Stack Overflow 2021 Developers Survey: <https://insights.stackoverflow.com/survey/2021>

# Terms of Reference

* Decision: Which programming language/technologies to use in our project
* Options we have: JavaScript or Python/Flask
* Criteria: Learning curve, Support of technology, Simplicity, Flexibility, Experience of developers with the technology

# Body

Python/Flask

Advantages:

* Python is a popular language (ranked third in Stack Overflow Developers survey), which means that there will be a lot of resources for learning python
* It’s lightweight, as Flask is micro-framework, which means lower levels of abstraction.
* As it’s based on Python, it’s simple to use due to Python’s simple syntax.
* It has unit testing, which allows for fast and easy debugging

Disadvantages:

* Handles requests one at a time, which can slow performance
* Using 3rd party modules can pose a security risk
* There is no common standard, which can make it hard to learn or pick up
* Hard to scale for large projects
* None of the developers have used the flask framework
* Not as popular as JavaScript (16.14 % of developers use flask, Stack Overflow Developers’ Survey 2021)

JavaScript

Advantages

* Most popular language in Stack Overflow Developers’ survey for nine years in a row, which means there will be lots of support, both in terms of APIs and educational resources
* All the developers have used JavaScript before, which means the learning curve will be short or none
* Is flexible as it can do both front-end and back end
* It’s more popular than flask (link: <https://stackshare.io/stackups/flask-vs-javascript>)
* It’s syntax is simple as it was inspired by Java and is similar to other popular languages like C++

Disadvantages

* As it’s primarily a client-side language, there can be a security risk on the client side
* Different browsers interpret JavaScript differently, however this was an issue on older browsers
* It can be hard to debug, as one error can completely change how JavaScript is interpreted

# Recommendation

JavaScript

* Every developer has used it
* Lots of API support, as well as educational resources
* Is flexible with client and server-side support
* Has simple syntax

Analysis of Alternatives: Platform

# Glossary/Reference

* Browser market share: <https://gs.statcounter.com/browser-market-share>

# Terms of Reference

* Decision: which platform to choose for our project ?
* Options: mobile app or web app, and for browsers Chrome or Firefox
* Criteria: availability for audience, cost of maintenance, customer requirements, compatibility

# Body

Native Mobile App

Advantages

* Better performance on mobile
* Can add additional functionality by accessing device features (camera etc.)
* Low setup cost

Disadvantages

* Cost of maintenance is high due to mobile upgrades, OS updates etc.
* There are multiple platforms for which we must target the audience
* May not be compatible with older devices
* Need approval from App stores

Web App

Advantages

* Easy to maintain as there is only one codebase
* Compatible with older devices, thus can target a wider audience
* Can be optimized for mobile
* No need for approval from App stores

Disadvantages

* There is a regular cost of hosting the web app
* Can’t interact as strongly with device features (such as camera etc.)
* Relies on internet connection

Browser: Chrome

Advantages

* Is the most popular browser, therefore it serves as the benchmark for where most users will interact with web app
* Has good developer tools with which the developers have experience

Disadvantages

* May require a license cost if we need to add more features in the future

Browser: Firefox

Advantages

* Open source, which means no cost for looking at the browser’s code

Disadvantages

* Isn’t as popular as Chrome, so won’t serve as a good benchmark

# Decision

Platform

* Web App: as it will have a lower maintenance cost, and can target a wider audience
* Browser: will target for Chrome as it’s the most popular browser, but shouldn’t make too much of a difference