# 

# 

**The Green Git Inspector**

**Analysis of Alternatives**

**FIT2100 Semester 2, 2019**

**Min Hee Cho, Jack Thomas Whelan, Jun Yoong Ooi, Abhijeet Mondal**

**TEAM green**

# 

# Analysis of Alternative

For ‘The Green Git Inspector’, three options surfaced about the platform that the app will run on. The options are desktop, mobile and web. Analysis of these options considering ‘The Green Git Inspector’ are shown in the table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **desktop** | **mobile** | **web** |
| **OS dependent** | Yes | Yes | No |
| **Hardware dependent (customer device)** | Yes | Yes | No |
| **Installation** | Yes | Yes | No |
| **Manual update** | Yes | Yes | No |
| **Requires internet (for this app)** | Yes | Yes | Yes |
| **Use device resources** | Yes | Yes | No |
| **Access to files in device** | Yes | Yes | No |
| **Cost(time, effort)** | High | High | Low |

After analysing all platforms and criteria, web app seemed to be the most preferred option for the following reasons:

1. This app can be used from any operating system, because there is no certainty that all users will be android or ios or windows user.
2. As web apps are mainly websites that can be accessed easily from devices, so it can be used from any device which has an internet connection.

## 

## **Terms of Reference**

Language Choice: Python, JavaScript

Criteria:

1. Familiarity
2. Compatible with Platform choice
3. UI customizability
4. Element Creation
5. GitHub API compatibility

Code Editor Choice: Brackets, Atom, WebStorm, VSCode

Criteria:

1. Cost
2. Speed
3. Language Support
4. Git Controls
5. Extensions

Web Hosting Server Choice: Netlify, GitHub Pages, Third Party

Criteria:

1. Cost
2. Git Integration
3. Downtime

## **Body**

### Language

|  |  |  |
| --- | --- | --- |
|  | **Python** | **JavaScript** |
| **Familiarity** | Familiar | Somewhat Familiar |
| **Native Platform** | Desktop App | Web App |
| **UI customizability** | Will need libraries to build a UI | Highly customizable with JavaScript |
| **Element Creation** | Will need libraries to create tables/pie charts | Pie Charts and Tables native |
| **GitHub API compatibility** | Yes | Yes |

Python is a familiar programming language for our development team, so Python was in strong consideration for our language of choice. However, the demand for a web app for this specific project outweighed the experience we had with Python. The development team is also reasonably familiar with JavaScript. Creating a UI including the required pie charts is also intuitive with HTML/CSS rather than using libraries in Python.

### Code Editor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Brackets** | **Atom** | **WebStorm** | **VSCode** |
| **Cost** | Free | Free | 30-Day Free Trial, $12.90 per month | Free |
| **Speed** | Moderate | Fast | Fast | Fastest |
| **Language** | JavaScript, HTML/CSS | JavaScript, HTML/CSS | JavaScript, HTML/CSS | JavaScript, HTML/CSS |
| **Git Controls** | With Extensions | Yes | With Extensions | Yes |
| **Extensions** | Yes | Yes | Yes | Yes |

The choice of a code editor is quite important as having the team use all the same editor would be useful for troubleshooting, pair programming and simply avoiding any compatibility issues that may arise. Brackets, Atom and VSCode are all free while WebStorm requires a subscription with a free 30-day trial. VSCode has better performance as a code editor when compared to Brackets, Atom and WebStorm. All options can code in JavaScript and HTML/CSS. Atom and VSCode have inbuilt Git controls while Brackets and WebStorm require extensions to do the same thing. All options have extensions for increased functionality, however VSCode seems to have a larger variety of choices when compared with the other options.

### Web Hosting Server

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Netlify** | **GitHub Pages** | **Third Party Web Hosting** |
| **Cost** | Free | Free | $15 a month |
| **Git Integration** | Yes | Yes | Some |
| **Server Downtime** | Short | Short | Shortest/None |

Choosing a web hosting server is also important as testing must happen while the web app is being hosted. There are 3 main choices: Netlify is a third-party website that connects to a Git repository to host a website from those files. Github Pages also connects to a Git repository. Third party web hosting is a paid service. However, Netlify and GitHub Pages are free and with git integration. However, the server downtime for Netlify and GitHub Pages might be higher than a third party web Hosting service.

## **Recommendations**

For language, JavaScript would be better as a web app is better than a desktop app for this specific project. JavaScript also allows the development team to use HTML/CSS to customize the UI rather than using other libraries in Python to do the same thing. VSCode would be the best choice for the code editor, however, Atom would also yield a very similar experience. GitHub Pages would also serve its purpose as a deployment option for the web app. Deployment is easy and streamlined as it simply uses the code from the connected GitHub project.