Analysis of Alternatives: Platform

Summary

This analysis will discuss the reasoning behind our decision in choosing an appropriate platform to develop our COVID Manager application. It will go through the criteria that were taken into consideration in this process, rank the importance of each of them as well as listing the platform options we are considering. These options will be compared against each identified criterion to find the most suitable platform.

Terms of reference

To identify an appropriate platform for our project, the first and most important factor to consider is the cost to develop, run and deploy the application since we are all students with a very limited budget. The next two factors are the complexity of the code base on these platforms and their adaptability to different operating systems. Making the right decision based on these criteria would help saving lots of developing time and leaving more time for quality assurance. Team members' experience with the platforms is another significant factor to consider. Lastly, applications' testability and performance on each platform are looked at to help ensure the practicality of the project.

Since our application's purpose is to provide users with quick updates on the current COVID situation and some interesting data, users would most likely use it during their free time. Hence, it should be easily accessed on mobile devices, which means a desktop application would not be an appropriate choice. The two other options to consider are web application and mobile application

Body

PLATFORM	Web application	Mobile application
Cost	Arranging a server can cost some fees, but since our application is relatively small, we can find free server hosting services such as Netlify and GITHub.	A mobile app does not require a server. But publishing applications in app stores will involve some costs such as annual developer account fees.
Code complexity	There are many frameworks that can be used to simplify the developing process. For example, React framework can be used to create small, reusable files instead of large, dense files. This therefore reduces the	Can use a cross-platform framework like React Native to avoid repeating the front-end developing process. But React Native still has many limitations and sometimes still requires knowledge about targeted platforms.

	complexity of the developing process.	
Flexibility	Since this app is front-end heavy, using a web app means our application can run on any operating system and therefore helps saving time.	iOS and Android require different programming languages for developing the front-end, so we would need to build the front-end twice.
Team members' proficiency	Everyone is familiar of developing a web application	Much more complex to do and many team members have never developed a mobile application before
Performance	Web apps written in JavaScript need to be interpreted before running and therefore takes longer to run.	Once a mobile app is compiled, it runs much faster than a web app.
Testability	Testing a web application can be done easily on any device.	Team members who don't own the right type of device would need to download an emulator to run and test the application.

Recommendations

Based on the evaluations of both platforms, web application is chosen over mobile application due to its wide range of supporting frameworks, its flexibility with operating systems, the ease of the testing process and team members' proficiency. It can be slower than a mobile application but this can be overlooked since performance is not the most important factor for this application.