Formal Project Proposal

Introduction

In recent years, especially in a major city like London, public transportation has become essential to commuters and tourists. That is why for my project, I've decided to create a cross-platform application that can be used to make travelling around London much easier. The app will include a way to find directions between two locations (including the cost of this), public transportation maps, and specially for tourists – locations of and directions to popular tourist locations.

Overview

The Problem

London is a major world city and one of the most famous cities in the world, it is definitely not one of the easiest to navigate around, especially without any help. Whilst London's public transportation is great, understanding how it works is not simple by any means. In the London Underground, there are multiple issues with multiple lines every single day that can effect the timing of trains' arrival, this means the timetable provided by TFL can be very unreliable. This will make knowing what time your train will be arriving difficult and will leave you unsure on what the best route to take to your destination is. This can cause inconvenience for commuters and tourists alike.

The Hypothesis and the Approach

The problem stated above could potentially be solved if travellers were able to conveniently access train arrivals in real time, this could be done within an app. Users could very conveniently access an app on their mobile phones that will allow them to see the train arrivals as well as additional functions within the app, such as; maps of the London Underground, a way to find directions between two locations and a list of popular tourist locations.

My approach will be to use Javascript and the Vue.js framework to create a cross-platform mobile app. I can retrieve data about London's public transportation by utilising TFL's (Transport for London) own API. TFL's API offers information such as current line statuses, live time arrivals and much more that I can implement into an app that is more user-friendly. I will be writing all of my code in the Visual Studio Code IDE and I will also be using GitHub to make my data easier to access and easier to keep track of.

Similar Apps

To help create the app, I have researched a couple similar and popular applications. The two that I have looked at are Google Maps and Citymapper. These are the two most commonly used applications by travellers in London to help navigate throughout the city.

Google Maps: Google Maps is definitely the most popular app used for travel worldwide. In terms of public transportation, Google Maps is capable of showing the user how to get from one location to another using public transportation. Google Maps is available on desktop platforms, as well as iOS and Android devices.

Citymapper: Citymapper is another example of a very popular app used for travelling, especially in London. In terms of public transport, Citymapper provides the most functionality. Within the Citymapper app, you are able to see the live arrivals for trains, see line statuses and much more.

I have researched these applications in particular to help develop how my own application will function and look like.

Evaluation

To ensure the app that I have decided to create works as intended and is appropriate for the target audience, the app will need to be evaluated and tested. There are a couple ways to do this that I have detailed below.

One way I can easily test the app myself is to actually use the app when I am travelling using London's public transportation and test the accuracy of the app's information, such as checking whether the time the app says the trains will arrive are accurate and checking whether the routes the app recommends me to take to my destination is indeed the quickest route I can take.

Other than the actual features of the app, the app will need to be tested to see if it has a good and appropriate UX and UI. I can achieve this by having my peers review the app, they can

test the app in the same way that I have stated above whilst also testing the UI of the app. With a larger sample size from all my peers and their user feedback, I'll have a better understanding of how good the UX and UI are, as well as how to improve these aspects of the app.

Required Resources

As the project I have chosen is purely software based, no hardware is required other than a computer that is capable of running an IDE such as Visual Studio Code. As previously mentioned, I will be using Javascript, the Vue.js framework and the API provided by TFL. I will also be using GitHub to make the whole process easier for me.

Deliverables

During the process of creating the app, I have decided on making three prototypes before the app is finalised. The first iteration will be a simple paper prototype to outline the appearance of the app, this means I will plan out how the app will look by drawing it out, I plan to complete the first iteration by Week 11. The second iteration will be a working prototype of the app with basic working functions, such as the ability to retrieve some information correctly from TFL's API, I plan to complete the second iteration by Week 16. The third iteration will be the final prototype of the app and should be fully functional, I plan to complete the third iteration by Week 20, this will give me enough time to make any last minute changes and complete the final report. All of the deliverables are listed in the table below.

Deadline	Milestone	Deliverable
Week 5	Submit project proposal	Project proposal
Week 8	Complete literature review	Literature review report
Week 11	Complete first iteration	Complete the paper prototype
Week 12	Submit interim report	Complete an interim report, outlining the progress made so far

Deadline	Milestone	Deliverable
Week 16	Complete second iteration	Complete a working prototype of the app with basic working functions
Week 20	Complete third/final iteration	Complete the final and fully-functional prototype of the app
Week 24	Submit final report	Submit the final report for the project and the fully-functional prototype of the app

References

[1] Sharma, A. (2018). Full-Stack Web Development with Vue.js and Node. Birmingham: Packt Publishing.

[2] Development, B. and Sheppard, D. (2019). Beginning Progressive Web App Development - Creating a Native App Experience on the Web | Dennis Sheppard | Apress.

[3] Chacon, S. (2019). Git - Book.

[4] Mardan, A. (2018). Practical Node.js: building real-world scalable web apps.

[5] Nelson, B. (2018). Getting to know Vue.js: learn to build single page applications in Vue from scratch.

[6] Vuejs.org. (2019). Guide — Vue.js. [online] Available at: https://vuejs.org/v2/guide/