ROUGH DRAFT TECH REVIEW

Cross-Platform Implementation

1.1 Overview

When considering Android or IOS mobile development, we would think of Java and Objective C. However, there are other ways to build effective and user-friendly applications. Xamarin, Meteor and Electron are a few IDE's that allow cross-platforming. Below we will conduct an in-depth review and comparison of the various IDE's.

1.2 Criteria

1.3 Potential Choices

1.3.1 Xamarin

Xamarin is a unique tool that integrates the language C# and Visual Studio, to create applications for all mobile platforms (IOS, Android and Windows). The cross platform looks 100% identical on any device, which provides better user experience. There are two major products, Xamarin.IOS and Xamarin.Android. Both of these products allow the source code to compile and handle such issues as garbage collection, memory allocation and platform interoperability. Due to Xamarin's cross-platforming, maintenance and updates can be handled a lot easier. With a simple update to the source file, it will be applied to both Android and IOS.

1.3.2 Meteor

Meteor is a model view controller framework for Node.js. According to Upwork, "It uses a mix of front-end JavaScript that runs in the browser, back-end JavaScript that runs on a Meteor server within a Node.js container, and any other HTML, CSS, and static assets to create highly reactive user interfaces." Meteor is also unique in its ways of integrating into any text editor or IDE. The use of Sublime Text editor requires no specific Meteor packages and an IDE such as WebStorm is also considered to be one of the best for Meteor out of the box. MongoDB is the only database at this time it can communicate with.

1.3.3 PhoneGap

PhoneGap is a mobile application development framework that integrates HTML, CSS and JavaScript. According to PhoneGap, "is the cloud based developer tool built on top of the framework, which offers cloud based mobile app development without the need for SDKs, compilers and hardware."

1.4 Discussion (Using comparison terms like "in contrast" and "compared to")

1.5 Conclusion (We chose Choice X because…); can include a simplified table

- Spoke with client and will not end up using a cross platform

- We are going to create a android application first
- For our stretch goal we will create an IOS application

Work Cited

https://www.upwork.com/hiring/development/meteorjs-node-js-framework/

https://forums.meteor.com/t/best-ides-for-meteor-what-do-you-use-and-why/8023

User action - verification (geolocation)

2.1 Overview

Retrieving the geolocation (current location) for our application is a crucial tool for us to complete our assignment. By obtaining one's current location will allow the user to see various events that are happening around them. After reviewing a few potential choices I came up with the Google Map's API, Cordova-Plugin-Geolocation API and Android Location API.

2.2 Criteria

2.3 Potential Choices

2.3.1 Google Map API

Google Map's API is a very useful tool on retrieving the current location of an individual. On the google maps API website, it provides a tutorial on how to display the geographic location of a user or device on google maps using the browser's HTML5 Geolocation feature along with the Google Maps JavaScript API. Within this website it provides example code and comments on how it works, making this a very useful tool. Both Android and IOS Google Maps API's requires the app to prompt the user for consent to use location services. They have very similar functionalites which determine the accuracy of the location return by the API's.

2.3.2 Cordova - Plugin - Geolocation API

This API provides the location information of any device in the form of its latitude and longitude. According to cordova's website, " Common sources of location information include Global Position System (GPS) and location inferred from network signals such as IP address, WiFi and Bluetooth MAC addresses. A quick not states that the API is based on the W3C geolocation API specification and only execute on devices that don't already provide an implementation.

2.3.3 Android Location API

Within the android API you can call android.location.LocationListener. There are three main components of this API. location class gives geographic location. It contains latitude and

longitude. LocationManager provides access to the system location services. And lastly LocationListener used for receiving notifications from the LocationManager when the location has changed.

FourSquare

- API provides programmatic access to the Foursquare places database, merchant platform and venues service.
- Use the Foursquare API to incorporate the venues platform into the application, which is then used as the app's location database.

2.4 Discussion (Using comparison terms like "in contrast" and "compared to")

After reviewing Google Map's API, Cordova-Plugin-Geolocation API and Android Location API, I noticed they all have their similarities and differences.

- Google maps api simple to use and provides example code
- Cordova-Plugin-Geolocation API includes other information that we don't need
    - IP address, WiFI and Bluetooth MAC addresses
- Android Location API
    - Very similar to Google Maps API and will be a toss up between the two

2.5 Conclusion (We chose Choice X because…); can include a simplified table

- Google's Map API will be used
- Simple to use
- Example code is provide
- Works excellent with android

Event Data Display

3.1 Overview

Similar to the geolocation section, uploading and viewing events on a map is another key concept we will be implementing into our application for users to easily view events that are happening around the community.

3.2 Criteria

3.3 Potential Choices

3.3.1 Google Map API

Under the title adding a Map with a marker in google explains the process on how to implement markers or events to the map. The explanation of it is very straightforward and understandable.

3.3.2 MapBox

Is a very similar to google maps on displaying various icons on a map. However, you can customize every aspect of the map from the colors, hiding or showing specific layers, to choosing which information to present on the map, all while the users are interacting with the map.

3.3.3 Microsoft Bing Map

- Multiple Map API's
- Bing Maps V8 Web Control:
- Bing Maps in Windows UWP
- Bing Maps REST Services - gets the geolocation
- Bing Spatial Data Services
- Bing Maps WPF Control
- https://msdn.microsoft.com/en-us/library/ff428643.aspx

3.4 Discussion (Using comparison terms like "in contrast" and "compared to")

3.5 Conclusion (We chose Choice X because…); can include a simplified table

- Similar to the previous statements to regards of geolocation we will use Google Maps api