Summary

Now that you have an architecture for TrojaNow, in this assignment, you will analyze that architecture and find an appropriate implementation platform for it.

No experience with mobile applications? Do not worry. Because you are going learn everything you need.

Then, you are going to implement skeleton of your application to be completed in the next phase.

Reading Chapter 9 of the textbook is an important starting point for this assignment.

Get the Tools

First we start with the tools. There are some frameworks out there that you can use to implement TrojaNow. Here is a brief list of them.

1- Android SDK:
The most popular development framework right now for developing Android apps is Android SDK. It only supports Java for classes and packages, XML notion for views and the UI, and the SQLite database system for storing data. It is supported by Android’s native operating system, thus, it seems to be the best option for people who want to achieve the best control of device infrastructure. On the other hand, it does not support a lot of architectural and programming patterns and styles. One would have to work around a little bit to find a good way to support architectures other than the ones Android SDK imposes on developers. This framework has great online community support and a lot of good online documentation.

2- PhoneGap (Cordova)
This platform is the most widely used non-Android-SDK framework. This a cross-platform, open source project aiming to make everyone able to write their applications using the familiar HTML, CSS, JavaScript combination, and to port it on three major mobile platforms. Adobe’s PhoneGap Build platform provides support for compiling the programs into native codes for each of the devices. Beware, however: compiling apps written in this platform has some difficulties sometimes.

Get started with PhoneGap at [http://phonegap.com](http://phonegap.com).

3- Titanium (Appcelerator)
Mostly used in enterprises and on cloud platforms, this framework supports JavaScript apps and translates them into native apps. It is not as popular as the other platforms, but more than 300 million application (on all platforms) are using Appcelerator today. Flexibility is the main strength of this framework.

For more information, go to [http://www.appcelerator.com](http://www.appcelerator.com).

4- Intel XDK
Intel XDK is built on top of Cordova, PhoneGap core, but it extends those tools with advanced capabilities and makes some performance issues related to PhoneGap disappear. It also doesn't share PhoneGap’s limitations like limited number of apps and lack of debugging mode. This is a brand new SDK with large potential benefits and ability to port applications to almost every platform (mobile and desktop) as apps or web apps.

5-Xamarin
Xamarin uses C# as the main programming language and is able to port applications on all major mobile operating systems. Unfortunately, most of its features are not supported in the non-commercial version of this platform which may make it a less-than-ideal choice. However, you may still find it informative and useful for TrojaNow.

For more information, go to http://xamarin.com.

Word of Caution
Remember, you should choose a framework that can provide you the means to implement your exact architecture. For example, if you have chosen MVC as your over-arching architectural pattern, you should be able to implement your architecture conforming to all MVC guidelines and its extensive separation of concerns.

So a good way to find the best framework for implementing your app is to consider your styles and patterns guidelines, look at the tools and languages supported by the framework, learn the underlying methodologies built into that framework, try to map your components and connectors to classes in that architecture and see if that whether a good fit for implementing your whole app’s architecture, or even a specific part of the architecture.
Implement the Architecture

After choosing an appropriate implementation platform among the above five—or other analogous tools you may find online or may already be familiar with—you are required to implement all your components/connectors and classes inside them as skeleton prototypes, without actual functionality. Take your architecture, make a place (package) for each of your components, and start writing your classes and methods inside them.

At this stage, you will just need to provide method stubs. Remember to put comments in your code for each component, class, their responsibility and their connection. You should specify how each class will actually perform its functionality. You should also include comments for each method and specify the method’s main task and responsibility. There is no need to try to identify and include all methods you may need in your eventual implementation, just the methods that are necessary to realize the application’s logic (for example, you may omit helper methods). Also, there is no need to write code for functions or any other thing related to the functionality of the app.

You only need to provide a skeleton of the whole application. However, your skeleton needs to compile and mock “run” in the programming language and on top of the platform of your choice. One of the most important things to keep in mind is having good comments about everything, because it will enable the reader to get to know how everything works.

Deliverables

List of deliverables is as follows:

1. A document containing your choice of platform. Explain why you picked that specific platform. What properties of your architecture and your chosen style is supported by the framework by default? What properties of your architecture is not supported by default? How are you going to implement those properties?

2. Project code containing your app’s skeleton. Remember to put sufficient comments and explanation in your code. If needed, mention how each package, class, or file is mapped to a components or class in the design artifact. You will be required to demonstrate that your app skeleton compiles and can run (without performing any actual functionality) on top of your chosen platform.