Miscellaneous Reports

# Test Report

The testing for this project was conducted on the web browser and on mobile phones. The Apache Cordova framework facilitates testing and deployment of applications because of the fact that it is written in HTML, JavaScript and CSS. For my testing phase, I used Google Chrome as my default web browser and a few android phones and tablets as my deployment device.

Based on the fact that the application is written in HTML, JavaScript and CSS, you can easily test a function without having to deploy the application to a virtual or physical device. For plain JavaScript functions, such as getting location, displaying a map and testing layouts, I simply used my web browser to test it. For the native mobile phone functions, such as accessing camera, displaying messages, accessing calendar, I had to deploy the app to a physical, or virtual, device in order to access the test results.

This approach was appropriate for this project. The reason why it was appropriate is due to the fact that I was using one of the best IDEs for developing Apache Cordova applications. The IDE is called JetBrains WebStorm. On this IDE, you can have direct access to virtual devices and also deployment options, making it easy to build, compile and run your code across several devices. The best useful feature about JetBrains WebStorm is the ability to create a Local Host in order to test your applications. This is how I was able to conduct testing on my Server – Client connections without always having to deploy it on a physical device.

For my PHP/MySQL scripts, I was using an IDE called JetBrains PhpStorm. This IDE is also great for writing PHP scripts, it provides code completion, refactoring and more. I chose this IDE because of its Version Control Systems integration, which enables the IDE to support remote deployment, databases/SQL, REST Clients and other tools. It also takes care of the Code Quality Analysis to help the developer write neat code that’s easy to support.

This approach was simpler, because I only had to test connections and queries to my remote Database, which is stored in a web server called 24Hosting. Once I knew the script was working, I deployed it to my web server and then accessed it from the Church Finder Ireland application.

The most interesting test happened when I sent the application to a friend over in Brazil. I needed to test the speed of the application when accessed from another country. Since my web server is located in England, the connections took longer than expected when compared to connections made from Ireland. This phase of the testing only used the *Date().getTime()* method to get the milliseconds that a function takes to execute.

The table below shows the difference between accessing my web server from two different locations:

|  |  |  |
| --- | --- | --- |
| **Name of function** | **Brazil (in seconds…)** | **Ireland (in seconds…)** |
| Accessing list of Churches | 3.1 | 1.2 |
| Getting details of a Church | 1.9 | 0.8 |
| Accessing map | 1.7 | 1.8 |
| Uploading a picture | 2.1 | 0.6 |
| Viewing an existing Picture | 2.6 | 1.3 |

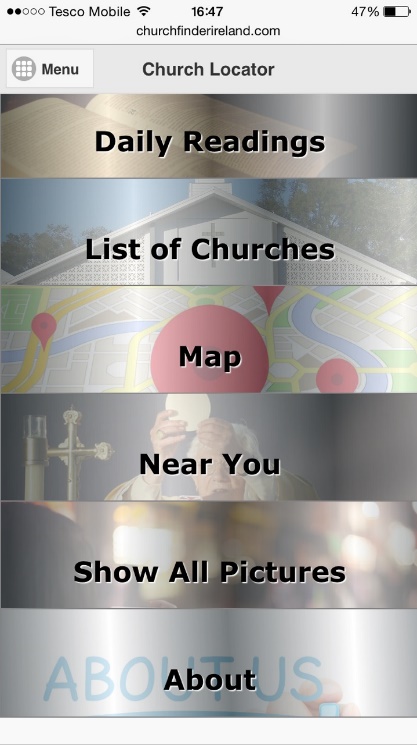
When the application was finished with the development, I initialized the system testing. This was an enjoyable approach because I was able to identify the main problem when developing Apache Cordova applications.

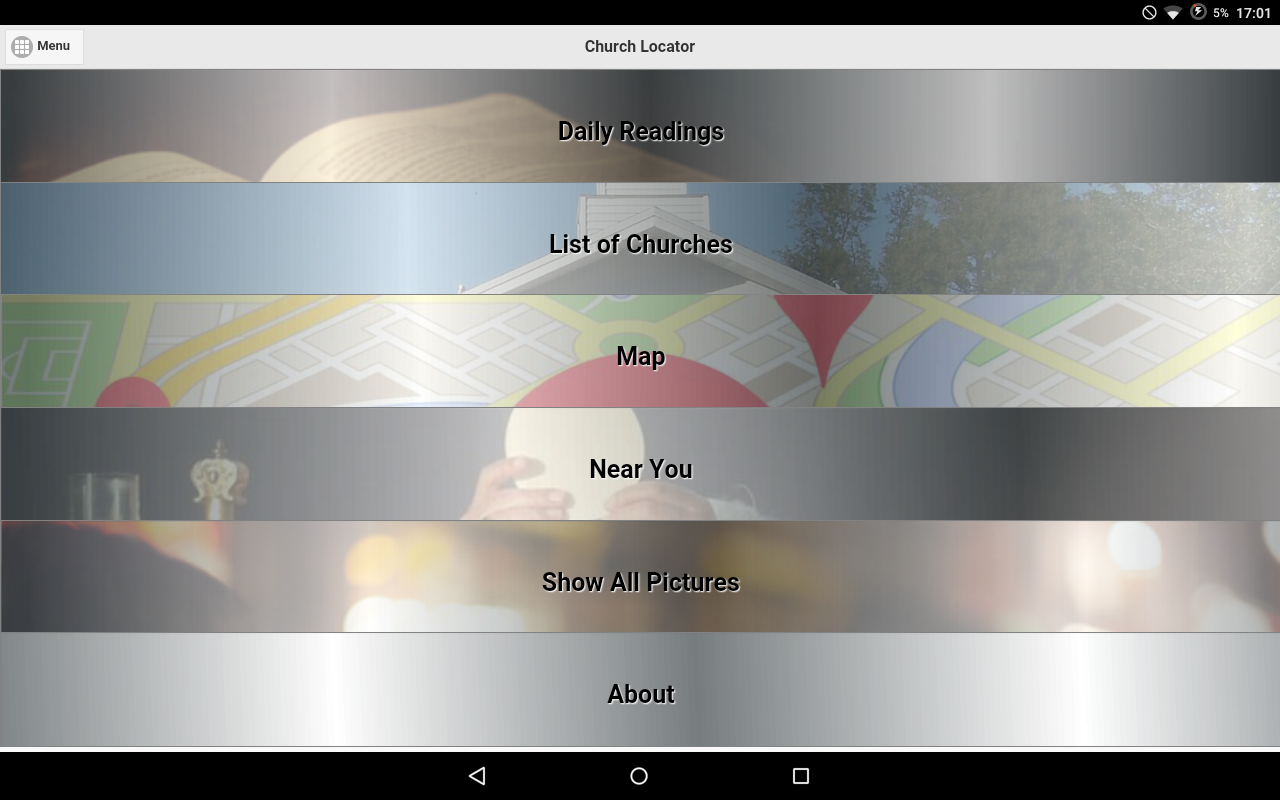
The table below shows a list of devices, and an indication of whether the application was successfully tested on the device:

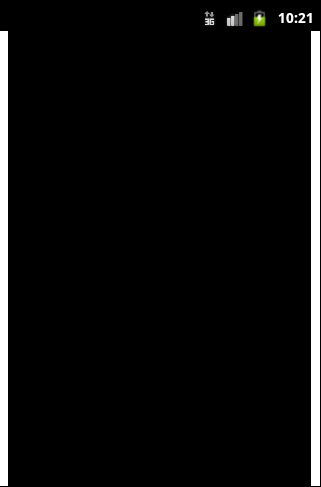
|  |  |  |  |
| --- | --- | --- | --- |
| **Device Name** | **OS Version** | **Successful** | **Comments** |
| Samsung Galaxy S5 | Android 5.1 | Yes |  |
| Google Nexus 5 | Android 5.1 | Yes |  |
| Motorola Moto G | Android 5.1 | Yes |  |
| HTC One | Android 5 | Yes with errors | CSS content not being correctly rendered, making the layout of the application really messy |
| Sony Xperia SP C5303 | Android 4.2 | No | The application was unable to start. Unknown problem |
| Samsung Galaxy S4 | Android 4.4 | Yes with errors | CSS content not being correctly rendered, making the layout of the application really messy |
| Samsung Galaxy S3 | Android 4.4 | No | The application was unable to start. Unknown problem |
| Samsung Galaxy Tab 2 10.1 | Android 4.4 | Yes with errors | CSS content not being correctly rendered, making the layout of the application really messy |
| Samsung Galaxy Pro 4 10.1 | Android 5.1 | Yes |  |
| iPhone 6 | iOS 8 | Yes |  |
| iPhone 5S | iOS 8 | Yes |  |
| Desktop | Windows 8.1 | Yes with errors | Mobile native functions do not work |

The images below are an illustration of the system testing made on physical devices:

**Samsung Galaxy S5 iPhone 6**

****

**Sony Xperia SP C5303 Galaxy Tab Pro 4 10.1**

****

# Installation Manual

# User Manual

# Post-Project Review