**Fi-Ware Social Proximity App**

Project Report

Joshua Eisikovits: C00156665

Project Supervisor: Dr Chris Meudec

17/04/2015

|  |
| --- |
| This document is the final document for the Fi-Ware Social Proximity Application. This document is a summary of the work that has been completed and is currently outstanding for the Fi-Ware Social Proximity App. This document will also aim to discuss what problems where encountered during the project and how they were tackled.  |

Table of Contents

[Introduction 3](#_Toc417036226)

[Purpose and Content 3](#_Toc417036227)

[Project Brief 3](#_Toc417036228)

[Submitted Project Description 3](#_Toc417036229)

[Mobile Client 3](#_Toc417036230)

[Web Application 4](#_Toc417036231)

[Register – Web Application 5](#_Toc417036232)

[Create Event – Web Application 5](#_Toc417036233)

[View Events Created – Web Application 6](#_Toc417036234)

[Final Product 7](#_Toc417036235)

[Fi-Ware Technology 7](#_Toc417036236)

[Final Product – Mobile Application 7](#_Toc417036237)

[Final Product – Web Application 7](#_Toc417036238)

[Description of Learning 8](#_Toc417036239)

[Conformance to Specification and Design 8](#_Toc417036240)

[Achieved from Project Specification and Design 9](#_Toc417036241)

[Unachieved from Project Specification and Design 10](#_Toc417036242)

[Questions posted to community. I did not disregard project for two iterations 11](#_Toc417036243)

[Technical Learning Skills 12](#_Toc417036244)

[Personal Learning Skills 13](#_Toc417036245)

[Acknowledgements 13](#_Toc417036246)

Introduction

Purpose and Content

The purpose and content of this document is to outline all work that has been completed to date on the Fi-Ware Social Proximity Application. The document will also cover in detail any work that is outstanding on the project in order to complete it to the final product version standard.

Also contained within this document are descriptions of all problems that were encountered while creating the Fi-Ware Social Proximity App as well as any solutions found.

Project Brief

Fi-Ware Social Proximity is a dual application consisting of a mobile application and a web application. The mobile application allows user to create a custom profile based on their current location and current interest. Whereas the web application allows an event organiser to login/register as well as create an event. Each event contains details such as name, position and even an event logo. Both applications use REST API techniques to transfer data to and from a MySQL database which is hosted on the Fi-Ware Cloud Virtual Machine.

Submitted Project Description

Mobile Client

The submitted application allows the user to login as a guest or using their social network account Facebook. The user is also allowed to create a profile which automatically locates their current position the user then has a list of current interests to choose from.



Once the user has installed SocialProx the application will be available for the user.

Create Profile - Mobile App Map View - Mobile App



Web Application

The submitted application allows the user to login to the web application if they are registered. If a user is not registered they can register to the web application. Registered users can also create events.

Login – Web Application

Register – Web Application



Create Event – Web Application



View Events Created – Web Application

(Show only events created by user that is logged in)

Final Product

The final release of Fi-Ware Social Proximity Application lacks some functionality which was mentioned in the functional specification and design documents.

However throughout the project I was able to complete the following;

Fi-Ware Technology

I was able to create a virtual machine using Fi-Lab. I also created security rules, a floating IP address as well as open network ports for incoming traffic to the virtual machine. However I was unable to implement the use of Generic Enablers in my project. I struggled a lot during the first iteration trying to come to terms with the Fi-Ware technology.

As I fell further behind throughout the duration of the project I was happy to have a working virtual machine in which I could install software (Java, Apache & MySQL). This was a major factor of the project if I was unable to establish connectivity between the mobile/web applications and Fi-Ware it would have been near impossible to proceed with the project as it was compulsory to use the Fi-Ware technology.

Final Product – Mobile Application

1. Sign in as a guest or using Facebook.
2. Create User Profile
	1. Automatically acquires GPS of phone, indoor accuracy is very accurate.
	2. Allows user to add current interest from a selected list.
3. View user current position of Google Map
4. Create profile is stored on remote server using MySQL database.

Final Product – Web Application

1. Register Event Organiser
2. Login Event Organiser
3. Create Event
	1. Allows event organiser to upload image of event
	2. Acquires current position (longitude and latitude depending on browser)
4. Create event is stored on remote server using MySQL database.
5. Able to use sessions to only allow registered users to create an event.
6. Also using session variable to query all event databases to retrieve events created by the current user who is logged in.

Description of Learning

Conformance to Specification and Design

The final release product of Fi-Ware Social Proximity Application matches the ‘External Interface Requirements ‘ under ‘Specific Requirements’ in my Functional Specification Document as I think my application really does grasp simplicity for the user.

Nevertheless it also matches the ‘Communication Interface’ mentioned in the sub section. I was also able to implement communication with Facebook and Google Maps API.

Originally the projected specification of the application was to build a mobile application with a separate web application which would have features for other users such as event organisers. The initial plan from my research was to create a Hybrid Application using Titanium Appcelerator which would work on multiple devices (Android/iOS/Windows Phone etc.) using one code base JavaScript. This was achieved as the mobile application was developed using this development tool throughout the project.

The submitted application is a combination of a hybrid mobile application and a web application. The hybrid mobile app was developed using the proposed technologies that I came across during my research. The technologies implemented were;

1. JavaScript – the core work of the application was written in JavaScript.
2. HTML5 – Used for web based form applications.
3. CSS – styling and designing of HTML pages.
4. REST API (HTTP methods, POST, GET, PUT etc.) to transfer HTML form data to a remote server.

The web application was developed using Java, Java Servlet Pages and HTML. Both applications communicate via Java Servlet pages (hosted on Apache Tomcat) using REST API methods to store and retrieve information on a MySQL database, both Apache Tomcat and MySQL were manually installed on the Fi-Ware Virtual Machine.

MySQL was the only relational database I covered in my Research Document as I knew during the second iteration I was running out of time and couldn’t afford to fall further behind by researching other databases. I came to the conclusion that I would use technologies that I had experience with prior to the project so that I would not fall further behind.

Achieved from Project Specification and Design

The above diagrams were the requirements given at the start of the project. Throughout the project I was able to complete the following.

* Make use of the Fi-Ware backend. However I did not take full advantage of the Infrastructure such as using Generic Enabler APIs. This was due to the fact that I found the documentation on Fi-Ware and Generic Enablers extremely difficult to understand. Other infrastructures such as Google App Engine and Amazon Web Services provide users with easy to read documentation along with video tutorials etc. I understand Fi-Ware is relatively new but I was pleased to be able to implement the Fi-Ware technology.
* The ability to send the phones Geolocation to the web app. This was accomplished by using JavaScript on the mobile side along with HTTP (POST) to communicate over to the web app.
* The ability to select and add interests from and to the web application. The mobile application has a pre-selected list (to prevent invalid or mistyped events)
* Handling GPS difficulties indoors. Using JavaScript and Titanium Appcelerator I was able to select

Titanium**.**Geolocation**.**accuracy **=** Titanium**.**Geolocation**.**ACCURACY\_HIGH**;**

Titanium**.**Geolocation**.**distanceFilter **=** 10**; //in metres**

This allowed the devices GPS of the device to gain a greater position.

* Further Exploration – Facebook Login API. I was able to implement a Facebook login with the mobile application. Although there are no other additional Facebook features such as post to timeline, share with friends etc. personally felt that for the projects point of you the more functionality the better.
* Web front end that allows event organisers to create event and place logos. Throughout the duration of the project I didn’t think I was going to get anywhere near implementing this as I struggled so much with connectivity (between mobile application and Fi-Ware backend)in the early iterations. However as you can see from above ‘Finished Product Web Application’ I have developed a (not so great GUI) a functional web application that meets the heading ‘insane’ requirements. Unfortunately there are some validation errors (date on different browsers with MySQL database format) but I feel that I have achieved the requirements.
	+ Event organiser can register to the application.
	+ Event organiser can create events.
		- HTML5 Geolocation API implemented for longitude and latitude
	+ The use of HTTP sessions to prevent unregistered users from creating or viewing created events.

Unachieved from Project Specification and Design

* For the web application to send push notifications to mobile users when two users are within the same vicinity.
* The ability for mobile users to interact with one another (chat) if they are within the same proximity.

I failed to implement the above because of the following reasons.

As I have expressed many times through this document, I fell behind from the first iteration due to connectivity issues between the mobile client and Fi-Ware Infrastructure. Connectivity wasn’t accomplished until the end of the 2nd iteration (the 23rd February – Recorded on Project Diary). The reason for this is because I was trying to communicate from the mobile client and Java Servlet using JSON.

Throughout the first and second iteration (screen grabs below of reaching out to the Titanium Community and Stack Overflow Community) there was many a time I wanted to throw in the towel and give up out of utter frustration.

Questions posted to community. I did not disregard project for two iterations



Nevertheless more research about Java Servlets and Apache Tomcat lead to RESTful architecture and implementing form based applications both via mobile and web applications.

In conclusion HTTP requests (GET, POST, PUT, etc.) were the most appropriate for the communication between both mobile and web application to the Fi-Ware backend. I can now only look back and be even more frustrated with myself that it took the majority of two iterations for something as straightforward as using HTTP requests.

I can only imagine if the two iterations that were lost what other functionality could I have been implemented or the GUI may look more professional. I may have been able to research further into WebSocket for chat or other techniques as well as how to send push notifications from the web app to mobile users.

This really concludes why one of the mandatory requirements was not implemented. Personally in the final iteration I looked at the Project Spec to see what functionality I could do in the final weeks. Therefore I thought that it was a strategic move to work towards implementing a web application rather than researching further into chat and push notifications which may never have been implemented after the further research.

Technical Learning Skills

Throughout the duration of the Fi-Ware Social Proximity project I have learned a lot about different technologies and platforms.

 Firstly was setting up a virtual machine on the Fi-Ware Infrastructure. I was able to create a virtual machine (CentOS Linux operating system) assign a floating IP address to this instance so that I could access the virtual machine from an external computer. I was then able to open TCP/UDP ports to allow traffic in and out of the virtual machine. After the virtual machine was set up I was able to download certain software (Java, Apache Tomcat and MySQL) via command line using Xubuntu operating system. I feel that I have come quiet competent and comfortable directing m self around the virtual machine using just the command line interface.

Secondly was the ability to develop an Android application using the Titanium Appcelerator IDE. I was able to use web technologies such as JavaScript, HTML and CSS to produce an Android application that is able to locate the user position as well as communicate with a remote server to store information on a MySQL database.

Furthermore, I also learned how to make beneficial use of Java Servlets and Apache Tomcat for this project. I was able to write multiple Java Servlets programs which acted as middle layer between the Tomcat Web Server and Mobile Client. Using Java Servlets I was able to collect input from the mobile client through web paged forms and store in a MySQL database using JDBC API (Java Database Connectivity).

Throughout the project I learned a lot about JDBC such as how to;

1. Connect to the MySQL database on the virtual machine.
2. Create SQL statements to query tables on the MySQL database.
3. Executing the SQL statements.

Learning how to create Java Servlets with the JDBC API connectivity for MySQL database also allowed me to further develop the project by creating a web based application. Using Java Servlets I was able to take advantage and allow users to upload images to the web application which could be stored on the MySQL database. If I was to use Java and Java Servlets again in the future to work with a relational database, thanks to this project I feel that I would not have a problem working with these technologies again.

Personal Learning Skills

Throughout the project there was many technical skills developed and improved in terms of programming and networking. However there were also many personal skills developed on such as;

* Communication skills between students and project supervisors. I had the feeling the consistent communication can prepare us for when we are working and could be communicating with potential customers and employers.
* Presentations in front of class and project supervisors. Many companies work in agile environments (small presentations after each iteration). Therefore improving on presentational skills I thought was very beneficial. Rehearsing presentations is something I’ve never done before, I found it quiet unusual at first but it is a good personal skill to improve on.
* A very common personal skill is time management. Managing work load with other class assignments and projects around project was very challenging but manageable.

Acknowledgements

I would like to thank my Project Supervisor Dr. Chris Meudec for his help and participation throughout the duration of the project.

I would also like to thank the remaining Fourth year lectures for their roles in extending class assignments and projects to fit this project deadlines.

|  |
| --- |
|  |