Microsoft Partner
Silver Web Development
Silver Software Development
Gold Web Development

University of Pretoria – COS 301 Project Proposal

KinderFinder Specification

Prepared for

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

21 February 2014

Version 1.1

Prepared by

Hennie Lessing
Solutions Architect
DVT

Contents

1. Project Vision and Objectives 3

2. Requirements 4

2.1 Proposal Scope 4

Web Administration 4

Mobile Application 4

Web API 5

Basic Reporting 5

Embedded Hardware and prototyping 5

2.2 Architectural requirements 7

Algorithm 8

Technology preferences. 8

3. Skills Requirements 10

4. Project Deliverables 11

5. Glossary of Terms 12

1. Project Vision and Objectives

DVT is in need of a solution that uses indoor tracking of people, to assist with a common problem in public places. Many restaurants aim to give parents a worry free dining experience, by providing facilities to entertain, and safely look after their children. However, a parent still has the worry, and consistent need to ensure their children are still in the safe and designated areas. The solution will attempt to assist the parents in being able to make sure their children are still in these areas without the need to get up from their tables.

2. Requirements

2.1 Proposal Scope

The project will consist of 5 main components of which one is mostly a proof of concept (POC).

- Web Administration
- Mobile Application
- Web API
- Basic Reporting
- Embedded Hardware and prototyping (POC).

The requirement is to create a system, that will through the use of RFID technology in wristbands and matching readers / receivers, be able to track children throughout a restaurant or designated area. This information needs to be fed to an API on a server that will push information to the relevant devices to inform parents of the children’s whereabouts within the designated area. The following notification devices will be used:

- Hardware Electronic tracking device on a diner’s table with LED’s for each designated area in the restaurant. The LED will light up based on the child’s location in the tracked area (Not in scope for this proposal).
- Mobile application (Android or IOS or both). The mobile application must display a map, which was uploaded and configured through the Web Administration portal, and overlay the children’s location on this application.

Web Administration

The Web administration must:

- Be used to allow installers of the system to setup restaurants and their layouts (using maps).
- Enable restaurants to link wristbands to a Patron’s mobile application, or assigned electronic tracking device (device not in scope).
- Enable restaurants to clear and unlink wristbands from devices and patrons’ mobile apps.
- Allow access to basic reporting will also be done through the Web Administration.

Mobile Application

The mobile application needs to be developed for either Android or IOS. Ideally both, but for the scope of this project only one platform needs to be supported, unless the project team feels they would want to use some cross platform technology like for eg. Xamarin. This can be discussed with DVT. The mobile application will be one of the methods used by restaurant patrons to access tracking information on their children while having their dining experience.

The mobile implementation must:

- Allow for users to view a map (divided into zones) of the restaurant.
- Overlay all the wristbands registered on the app user’s name on the map to show where they are (some patrons might have multiple children to track).
- Allow for setting up of alarms and notifications based on movement of tracked wristbands.

- Allow for a user to select a restaurant, and a restaurant branch in order to get the map for the setup.
- Enable the user to link the wristband to their mobile application

Web API

The system must have a central web based API that concerns itself with gathering information and supplying information from the system. This API will be used by:

- Mobile Application
- Web Administration
- A Service, if needed, that can push information to devices.

Therefore this API must be loosely coupled from any implementation using it, as it needs to be re-used for multiple implementations, and possible future applications / implementations.

Examples of information to be sent and retrieved from the API:

- Wristband, with signal strength(RSSI) picked up with reader/receiver ID.(Only this information needs to be capture from the device, location calculations and algorithms can be implemented on the server side of the API which will allow for dumber and cheaper hardware devices)
- User Details
- Tracking information
- Any CRUD functionality must be channelled through this API

Basic Reporting

Basic reporting must be available through the web administration:

- Facility usage (zonebased) by children. This will enable a restaurant to see statistics on which areas are most popular for children.
- Usage statistic of mobile app users that could possibly enable restaurant chains to implement a loyalty program.
- Health reports, reports that will indicate the health of the hardware being used. This should pick up trends that are impossible with the restaurant layout. For e.g.
    - When the only path from zone A to zone B is through zone C, and Zone C was never picked up in the tracking stats (could indicate a broken reader / receiver).
    - When, for a specific wristband, the only path from zone A to zone B is through zone C, and Zone C was never picked up in the tracking stats. (Could indicate a broken wristband).
    - When a certain zone is not registering at all for a specific wristband or any wristband
    - When erratic zones readings are picked up.

Embedded Hardware and prototyping

Physical hardware devices used in this project includes:

- Receiver/Reader prototype
    - These are devices that are placed at predetermined locations within the designated area where tracking will occur in order to pick up wristbands and

  - their relative signal strengths. These devices will use wireless communication will to communicate this information to an access point with internet access.
- Wristband prototype(doesn’t need to be a wristband yet for the project)
  - These can be active or passive RFID tags, or wireless transponders/transceivers. They will be used to communicate its existence in range of a receiver/reader and its signal strength (RSSI).
- Electronic tracking device(not in scope)
  - This device, when developed, will be a piece of hardware the patron can put on their table with LED’s representing zones on the restaurant map. Not to be of concern for this project.

2.2 Architectural requirements

Devices in this implementation needs to be as cheap as possible, and easy to install. Therefore it must be low powered, wireless devices and its functionality must be limited to gathering information and passing it to the server that will process it further in order to save battery power.

The diagram in Figure 1 depicts a suggested architecture.

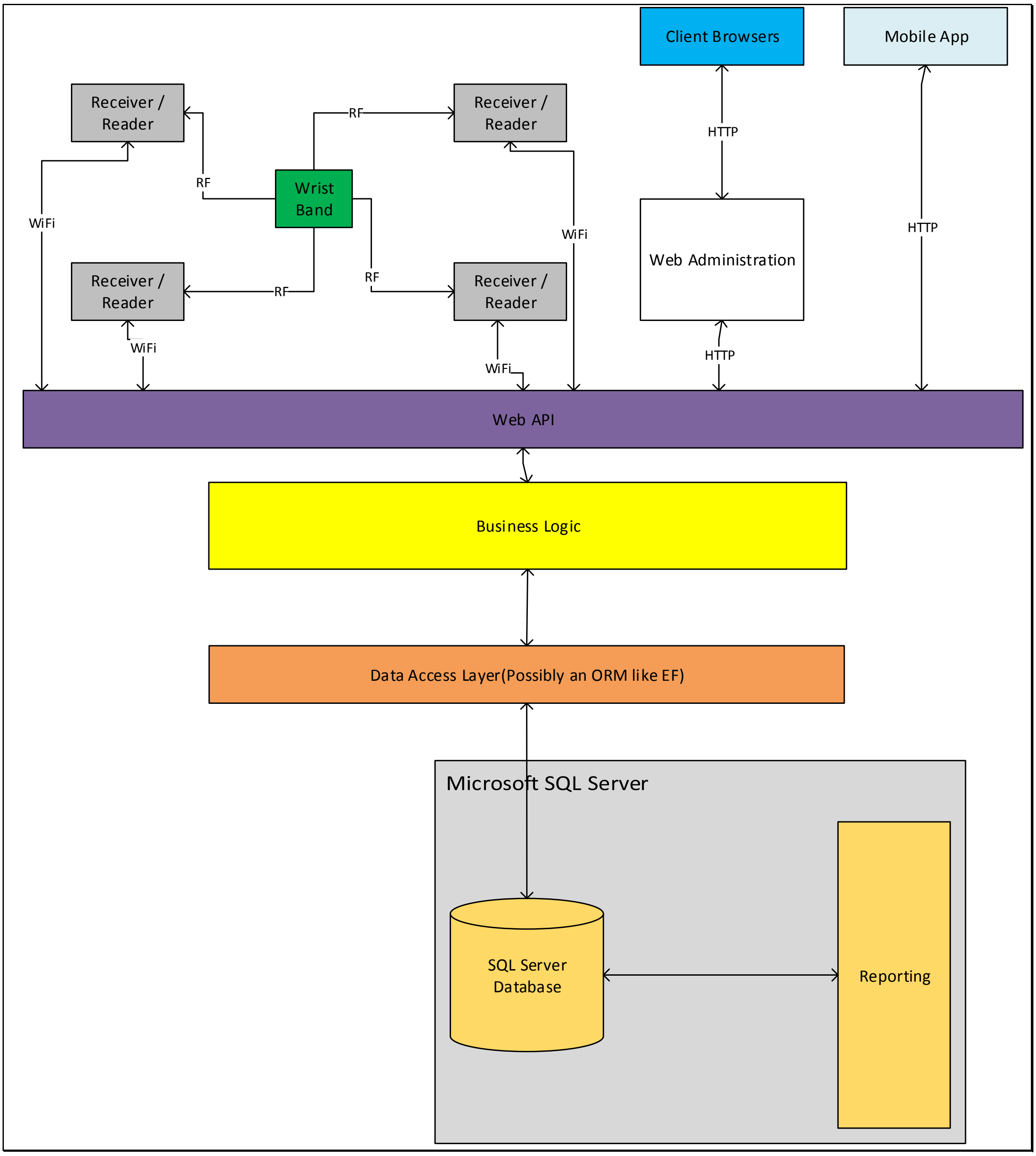

Figure 1

In this diagram:

Readers and receivers will pick up a wristband in its range (Active RFID or any other technology the project team can suggest) and report the wristband's identifier together with RSSI to the Web API which will pass it to the business logic layer to run an algorithm to determine the position.

This position will then get logged in the database, from where it can be reported on, and pushed to other devices that needs to know this information.

Algorithm

The project team must play with algorithms to determine which algorithm, paired with the hardware will give us the most accurate results. Algorithms to consider could be zoning, triangulation, or a combination of both. As a result of how RF waves behave, triangulation alone can prove to be tricky so a combination of zoning, decent placement of multiple readers and receivers might need to be used.

Technology preferences.

This project lends opportunity to work with a wide variety of technologies.

However, for the Web administration and Database, DVT will provide Microsoft tools to deliver the solution. Therefore, the Web API, Web Administration, and Business Logic layer will need to be done in .Net, using:

- ASP.Net MVC 5 (HTML 5, JavaScript, JQuery)
- ASP.NET WebApi
- C#
- Microsoft SQL Server 2012

The mobile application needs to be built for either IOS or android. Consideration can be given to write once deploy everywhere frameworks to target both, like Xamarin. The mobile app will need to use the Web API to communicate with the application backend.

As the hardware part of this project is purely POC, multiple suggested technologies for these components of the project can be used.

A few possible suggestions could be:

- Arduino
  - Possibly for readers/receivers or wristbands, together with an active or Passive RFID's and readers
  - C++ Syntax, and easy to deploy code.
  - See http://www.arduino.cc/
- .Net Gadgeteer
  - Possibly for readers/receivers, wristbands, together with active or passive RF Id's and readers
  - Uses .net Micro Framework and IDE, easy to deploy code.
  - See https://www.ghielectronics.com/technologies/gadgeteer)
- Arduino based Jeenode
  - Possibly for wristbands together with an active or passive RF Id
  - See http://jeelabs.net/projects/hardware/wiki/JeeNode

- Raspberry Pi
  - Possibly for Receivers, together with active or passive RFID Readers
  - Runs Linux, java, node.js, or any other programming language can be used to pick up readings and pass it over Wi-Fi to the Web API.
- Any other technology the project team can find that will satisfy the physical and low cost requirements.

3. Skills Requirements

General development skills will be needed for this project. But a wide range of different programming languages will be used.

Technical skills this project will require and teach:

- Research and development
- Introduction to the Internet of Things and the maker movement
- .Net Development skills
- HTML 5 and JavaScript development skills
- Possibly Java development skills, depending on choices made for hardware.
- C++ Development Skills
- API Integration

For the hardware part, basic know how of circuits is not of such paramount importance as the hardware section of this project is deemed as a POC.

DVT will assist in giving guidance with regard to this project, and many tutorials and open source libraries that was posted by the maker community can be found online also.

The key is that the WebApi must be able to receive RSSI and Wristband Identifier data, and that the business logic that runs the algorithm to determine positioning must be easily configurable or interchangeable, as this might need to be changed depending on what hardware or techniques are being used (Dependency injection can possibly be used for this).

4. Project Deliverables

Project deliverables expected are:

1. Source Code
2. Technical specifications
3. User Requirements Specifications
4. Architecture Design
5. Build and deployment scripts or instructions.
6. Release Notes.

5. Glossary of Terms

Acronym	Definition
RFID	Radio-Frequency identifier
API	Application Programming interface.
RSSI	Radio Signal Strength Indication
POC	Proof of Concept
MVC	Model View Controller
ASP.NET MVC	A Project type in Visual Studio implementing the MVC pattern