Hazardous Waste Finder: Hazardous Waste Tracking Management System

by

Karthi Kishore Jakkli Sounder

A PROJECT

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Master of Science

Major: Computer Science

Under the Supervision of Professor Jitender Deogun

Lincoln, Nebraska

May, 2014

Hazardous Waste Finder: Hazardous Waste Tracking Management System

KARTHI KISHORE JAKKLI SOUNDER, M.S.

University of Nebraska, 2014

Advisor: Jitender Deogun

Hazardous waste is a waste that is considered substantially harmful to health and environment. Hazardous waste includes explosives, compressed gas, poisons, oxidizers, corrosives, flammable solids/liquids, radioactive materials etc. It is important that hazardous waste needs to be recycled in a timely manner. Hazardous waste management comprises of three stages namely treatment, storage and disposal. Tracking the waste is a fundamental requirement at every stage in waste management. In this project, I developed an android mobile application - Hazardous Waste Tracker (HWT) using a hybrid mobile application Phonegap framework. HWT allows an inspector to conduct a weekly assessment about the type of hazardous waste collected and the recycled rate from an inspection site. The benefit of this project is to effectively manage time and manpower by eliminating the traditional use of paper reports used for tracking.

ACKNOWLEDGEMENTS

I would like to take this opportunity to sincerely thank my advisor Prof. Jitender Deogun for his constant support, motivation, guidance and encouragement during the project. Also, I would like to express my gratitude to Prof. Lisong Xu and Prof. David Swanson for agreeing to be on my committee. Last but not the least I place a deep sense of gratitude to my family members and friends who have been a constant source of inspiration throughout the project. My heart and soul will always be with them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 BACKGROUND

Chemicals are used in our day-to-day lives in the form of plastics, medicines, gasoline, and electronic devices etc. When chemicals/hazardous substances are thrown away, they become hazardous waste. Hazardous wastes are corrosive, ignitable, reactive and toxic substances [1] [2]. Examples of hazardous waste include batteries, aluminum cans, old corrugated containers, ferrous metals (copper, brass, lead, chrome-ore and nickel), rubber, transmission fluid etc. Hazardous wastes are meant to be reused or recycled. Disposal of hazardous waste becomes significant as it can create health risks for people and damage the environment [2].

The Environmental Protection Agency (EPA) regulates the management and disposal of hazardous waste. Under this mandate, the EPA developed strict requirements for all aspects of hazardous waste management including the treatment, storage, and disposal [3]. EPA has created three categories for hazardous waste namely solid waste, universal waste and household hazardous waste. Solid waste consists of ordinary waste that is generated in the normal business process such as paper, cardboard, plastic, glass and computer equipment. Though not regulated, recycling practices for solid waste have started to emerge in recent days. Universal waste is a special category of hazardous waste consisting of items such as batteries, oil,

fluorescent lamps and pesticides. They are subjected to regulatory requirements for proper disposal. Household hazardous waste generated from residential households, includes paints, solvents, automotive waste, aerosols, refrigerant containing applications have strict regulations for monitoring and waste disposal.

Standard EPA codes are assigned for different hazardous waste. The F-list includes wastes from common manufacturing and industrial processes such as soil treatment residues, metal treating, landfill leachate and spent solvents. The K-list includes waste from specific industries such as petroleum or pesticide manufacturing wastes. The P-list and U-list includes wastes from commercial chemical products such as chemical acids and pharmaceutical products [4].

Proper management and control can significantly reduce the dangers of hazardous waste. There are many rules for managing hazardous waste and preventing releases into the environment. A lot can go wrong when we try to contain hazardous waste like tanks used for storing petroleum products and other chemicals can leak and catch fire. When hazardous wastes are released in the air, water or on the land they can spread and contaminate the environment and poses great threats to our health. Hence tracking of hazardous waste is necessary.

Using a set of forms, an individual tracks hazardous waste from the time it leaves the generator facility where it is produced, until it reaches the off-site waste management facility that will store, treat or dispose the hazardous waste. This kind of tracking system ensures that hazardous waste is transported from the first place of

generation to the place of disposal without being tampered, dumped or illegally disposed along the way. The idea of this project is to eliminate the use of paper reports used in tracking hazardous waste thereby minimizing time and man hours.

1.2 STRUCTURE OF THE PROJECT

The rest of the project is structured as follows

- Chapter 2 reviews more on the mobile application development.
- Chapter 3 focusses on the requirement specification of the application.
- Chapter 4 focusses on the design and implementation aspect of the application.
- Chapter 5 covers the techniques employed in testing the application.
- Chapter 6 concludes the report with summary and suggestions for the future work.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

The advent of smartphones has led to enormous developments in mobile applications, used by people in their day-to-day activities. Mobile applications are software programs designed to run on smartphones, tablets and other mobile devices. According to Flurry Analytics, usage of mobile applications has increased at a tremendous rate in the last five years (115%) [5]. According to a report from comScore, mobile apps accounted for 47% of internet usage of the entire mobile internet usage when compared with mobile browsers that had a share of 8% [6]. According to a report from Neilson, mobile users download more apps and the average number of apps owned by a smartphone user lies at 41, a 28% increase since 2013 [7].

The development of mobile apps started initially for information retrieval purpose such as stock market, weather information, email, calendar etc. The growing consumer demand and introduction of new developer tools was the sole reason behind the expansion of mobile apps in domains such as banking, finance, mobile commerce, inventory control, gaming etc. Mobile apps are generally available through distribution platforms such as Apple Store, Google Play, Windows Store and Blackberry App world. As shown in Figure 2, mobile application development is classified into three categories namely native applications, mobile optimized web applications and hybrid/cross

platform applications. The categorization is based on the complexity involved, architectures, development methods practiced and cost involved in production.

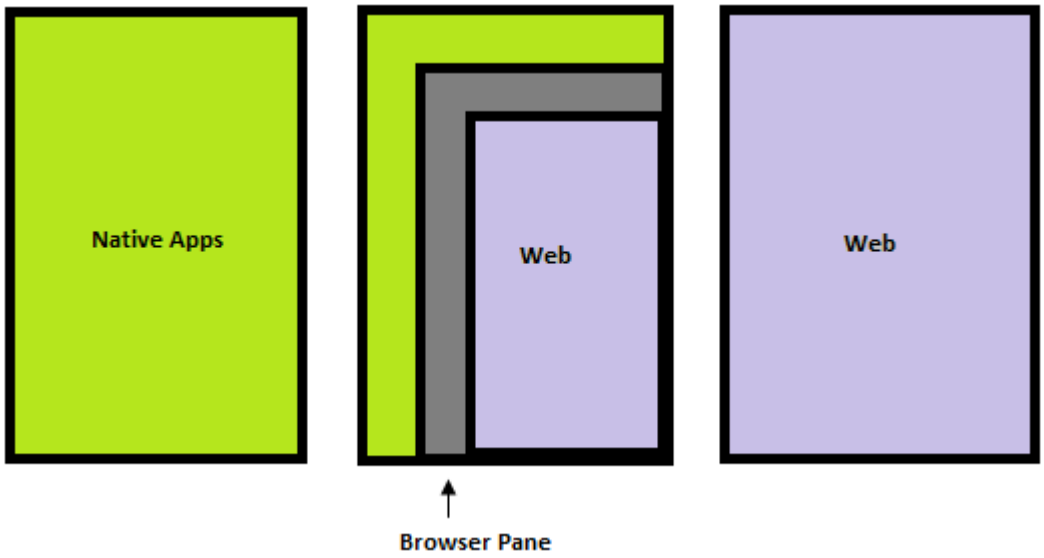

Figure 1 ***Native/ hybrid / web comparison***

2.2 NATIVE MOBILE APPLICATIONS

A native mobile app is a software application that is developed using a specific programming language such as Java, Objective C and C#. Native apps adhere to a specific platform (Android, iOS, Windows Mobile, Symbian, BADA, Blackberry), making use of device related features such as GPS, camera, accelerometer, compass etc. Since native apps make use of device features, excellent user experience and speedy performance is guaranteed. Native apps are available through an application stores making them safe and secured from the context of a mobile user. Some popular examples of native app include Facebook, Instagram, Weather Channel, Spotify etc. According to timetech.com, iOS and Android platforms dominate the native app market

with each having 800,000 applications [8]. The main drawbacks of native apps are the production cost involved in development as code cannot be re-used for another platform. Also, distribution of the native app code is complex as it involves standard code practices and needs approval from the concerned authorities related to the platform on top of which it is developed.

2.3 MOBILE OPTIMIZED WEB APPLICATIONS

HTML5, a latest standard of HTML is used to build diverse and powerful mobile websites and applications. Some prominent features of HTML5 include local storage, 2D/3D graphics, CSS3 support, Audio / Video and local SQL database [9]. Mobile optimized web applications are websites written using HTML5 that have the look and feel like native applications when run using mobile browsers. m.youtube.com, Groupon, scribd are well known websites recently developed as mobile optimized web applications using technologies such as Jquery, JSON, AJAX etc. Advantages of such applications include platform independency, cheaper build and maintenance costs. Also, it is possible to analyze the usage of mobile optimized web applications making use of web analytics tools. Low performance, inability to access device features (sensors, Bluetooth) and unattractive user interface are the key drawbacks of mobile optimized web applications.

2.4 HYBRID APPLICATIONS

Hybrid applications have gained significance recently as it stands middle ground between native applications and mobile optimized web applications. Though mobile

optimized web applications are platform independent, they cannot access device specific features. The hybrid approach has evolved to deliver platform independence incorporating an application to access the device specific features and providing offline support. Hybrid applications are built using HTML5 pages coupled with JavaScript and CSS that runs in a browser packaged inside a native container app making it accessible to native API's. Since the hybrid application is embedded as a native app, it is easily distributed through app stores.

2.5 HYBRID APPLICATION FRAMEWORK: PHONEGAP

Phonegap (Adobe Cordova) is an open source hybrid application framework produced by Nitobi which was later purchased by Adobe Systems. Phonegap helps to create native applications for mobile devices using web APIs HTML5 (user interface), CSS (layout and styling) and JavaScript (interactivity with the native OS functionality) facilitating integration with HTTP services [14]. Phonegap application development supports desktop and mobile operating platforms that include iOS, Android, Windows Phone, Blackberry, Bada, Symbian and HP Webos. Phonegap framework supports native device features such as accelerometer, camera, compass, geo-location, contacts, media, local storage and notifications offering a lot of flexibility for different platforms as shown in Table 1. BBC Olympics, My Heart Camera, exfm, Yoga + Travel are some of the popular applications that are available in app stores developed using Phonegap [11].

Figure 2 shows the basic architecture of Phonegap framework. The application is developed with web standards, wrapped up using Phonegap framework that has access

to native APIs and deployed across multiple platforms. The main advantage of using Phonegap framework is the availability of single code base across platforms and the developer doesn't have to worry about programming languages such as Objective C or

Figure 2 Phonegap Architecture

Java. For these reasons, it has become easier to develop and maintain native mobile applications at relatively low cost.

	iPhone 3G	iPhone4 & greater	Android	Blackberry OS 6.C+	Blackberry OS 10	Web OS	Symbian	Bada	Windows Phone 7 and 8
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓	✓
Compass		✓	✓		✓	✓		✓	✓
Contacts	✓	✓	✓	✓	✓		✓	✓	✓
File	✓	✓	✓	✓	✓	✓			✓
Geo Location	✓	✓	✓	✓	✓	✓	✓	✓	✓
Media	✓	✓	✓		✓				✓
Network	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification(Alert)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification(Sound)	✓	✓	✓	✓	✓	✓	✓	✓	✓

Notification(Vibration)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Storage	✓	✓	✓	✓	✓	✓			✓

Table 1 Phonegap Supported features and unsupported features

CHAPTER 3

REQUIREMENTS SPECIFICATION

3.1 INTRODUCTION

3.1.1 APPLICATION OVERVIEW

Hazardous waste finder is a mobile application intended to make an inspector to collect and transmit the hazardous waste data from the inspection area. The container that stores the hazardous waste and the waste area where the container is located will be inspected weekly and the collected data is transmitted to the web component. Assuming the web component already exists, a substantial amount of work is done on the mobile side to measure and track the amount of hazardous waste recycled by eliminating the use of paper reports. A minimal interface provides only the essential functionality and keeps the process simple, allowing inspectors to remain calm and free of stress. The application is implemented for Android devices and includes HTML/CSS, JavaScript, and the Phonegap framework. The data that is collected from the mobile component is stored using the MYSQL database.

3.1.2 SCOPE

Hazardous waste finder will be targeted towards an inspector, who does the inspection on the hazardous waste with the help of mobile devices.

3.1.3 DEFINITIONS

3.1.3.1 SOLID/UNIVERSAL HAZARDOUS WASTE TRACKING

This module will make it easier to track the amount of solid or universal waste and the recycled information. The inspector will select the location, waste type from a predefined list and will be able to filter the waste type group, waste type and waste sub type individually based on the attribute specified. Extra information such as waste source, waste quantity, measuring unit, container size and notes about the waste, photo capabilities adds to the waste details that need to be tracked. The amount of recycled information is gathered making use of the percent full rate and percent recycled rate.

3.1.3.2 CONTAINER INSPECTION

This module will make it easier to monitor the container that stores the hazardous waste by listing its corresponding attributes. The inspector will select the container id, container size, container type, container contents and the percent full rate of the container. Also, container attributes such as grounded, creased, rusted, leaked, label visibility adds to the vital information needed when examining a container. The container is certified to be fine only if the percent full rate is greater than 90% and all its attributes converge to be true.

3.1.3.3 WORK ORDERS

This module will make it easier to track the shipment status of the container. The inspector will select the container id that displays the details about the container such as container weight, shipment date etc.

3.1.3.4 AREA INSPECTION

This module will make it make it easier to monitor the storage areas of the hazardous waste. The inspector will select the location that needs to be monitored and safety aspects of a location such as warning signs, emergency contact lists, communications available, aisle space maintained etc. are recorded.

3.1.3.5 NEW INSPECTION REQUEST

This module will make to easier to track any deviations encountered in the inspection process. The request will allow an inspector to take photos of the abnormalities, take down notes using the devices keypad and rate the issue in level of urgency to the appropriate party by embedding information such as date, time, location latitude and location longitude.

3.2 OVERALL DESCRIPTION

3.2.1 PRODUCT PRESPECTIVE

This application is completely independent for Android devices. The application relies on information to be processed and sent to the database where the web component makes us of the recorded information thereby minimizing paper reports.

3.2.2 PRODUCT FUNCTIONS

The application focusses on the following actions that an inspector can perform:

- Solid/Universal Hazardous Waste Tracking.
- Work Orders.

- Container Inspection.
- Area Inspection.
- New Inspection Request.

3.2.3 OPERATING ENVIRONMENT

The application is built using Phonegap version 2.9.0 and will run on all current versions of Android operating systems from SDK version 9.0 and above.

Rationale: Previous versions of Android did not offer many features integral to the application such as swipe buttons.

CHAPTER 4

DESIGN AND IMPLEMENTATION

4.1 STRUCTURAL (CLASS) DIAGRAM

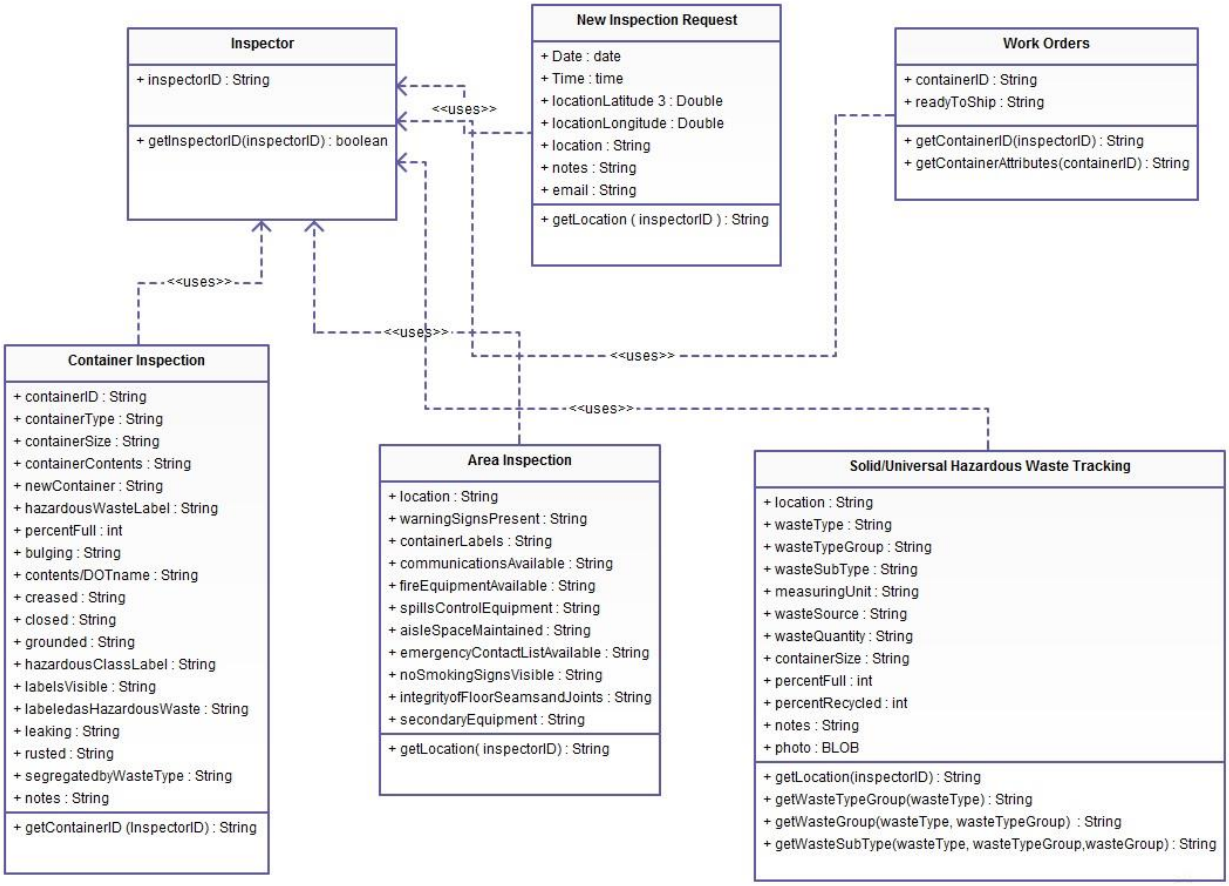

Figure 3 Class Diagram

4.2 BEHAVIOURAL (USE CASE) DIAGRAM

4.2.1 LOGIN

Description: The inspector should be able to login into the application.

Precondition: The inspector must be a registered.

Main flow of events:

1. The inspector enters the id in the textbox for the inspector id.
2. The inspector selects the login button.

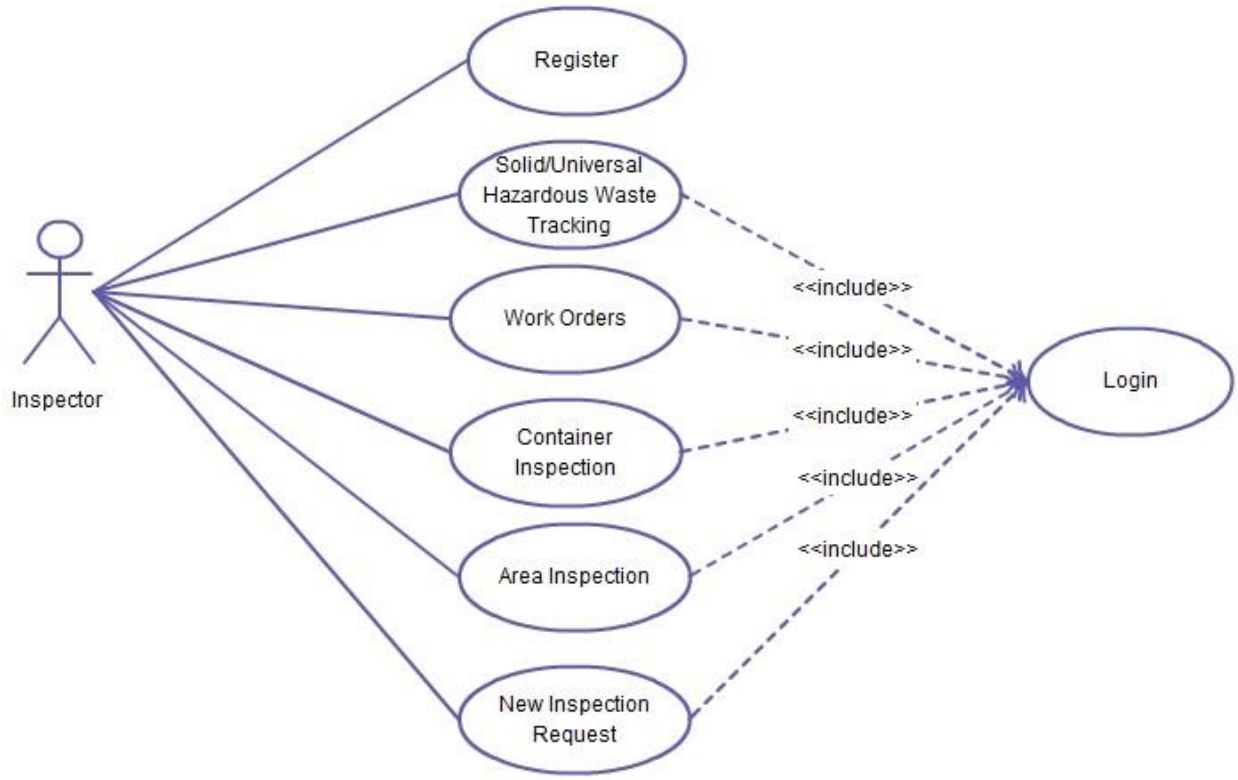

Figure 4 Use case Diagram

Post condition: The inspector will be logged into the application.

Exceptions:

1a. If an inspector selects the login button without entering the id, the systems prompts the inspector to enter a valid id.

4.2.2 SOLID/UNIVERSAL HAZARDOUS WASTE TRACKING

Description: The inspector enters the details about the type of hazardous waste that needs to be tracked.

Precondition: The inspector has logged in.

Main flow of events:

1. The inspector navigates to the solid/hazardous page.
2. The inspector enters the location and type of hazardous waste to be tracked.
3. The inspector selects the waste type group, waste group and waste sub type one by one dynamically based on the previous selections.
4. The inspector selects the measuring unit (kilograms/pounds), enters the waste source and the waste quantity/weight.
5. The inspector selects the container size, enters the full rate and recycle rate of the container.
6. The inspector enters the description about the waste and uploads photo, if any.
7. The inspector submits the type of hazardous waste.

Post condition: The inspector cannot make changes once submitted.

Extensions:

3. a: The waste type group, waste group and waste sub type are retrieved from the database.

4.2.3 CONTAINER INSPECTION

Description: The inspector enters the attributes about the container that stores the waste.

Precondition: The inspector has logged in.

Main flow of events:

1. The inspector navigates to the container inspection page.
2. The inspector selects the container id and new container.
3. The inspector selects the container type, container size and container contents.
4. The inspector enters the percent full rate of the container.
5. The inspector enters the container attributes such as new container, bulging, contents/DOT name, creased, grounded, closed, hazardous class label, labels visible, labeled as hazardous waste, leaking, rusted and segregated by waste type.
6. The inspector enters the notes about the container, if any.
7. The inspector submits the container attributes.

Post condition: The inspector cannot make changes once submitted.

Extensions:

5.a : The container attributes are simple yes/no buttons that makes it easy to monitor the container.

4.2.4 AREA INSPECTION

Description: The inspector enters the storage information with respect to the inspection point.

Precondition: The inspector has logged in.

Main flow of events:

1. The inspector selects the location of the inspection point.
2. The inspects selects the storage information about the inspection point such as warning signs present, container labels, communication available, fire equipment available, spills control equipment, aisle space maintained, emergency contact list available, no smoking signs available, integrity of floor seams and joints and secondary equipment
3. The inspector submits the information about the inspection point.

Post condition: The inspector cannot make changes once submitted.

Extensions:

2.a : The storage information about the inspection point are simple yes/no buttons that makes it easy to monitor the storage areas.

4.2.5 NEW INSPECTION REQUEST

Description: The inspector generates a new inspection request if they discover something wrong during the inspection process.

Precondition: The inspector has logged in.

Main flow of events:

1. The inspector enters the current date and time.
2. The inspector selects the location and location co-ordinates (latitude and longitude).
3. The inspector enters the notes about the waste/inspection point and the email address of the concerned person to which the issue needs to be reported.
4. The inspector uploads photo about the inspection point, if any.

Post condition: The inspector cannot make changes once submitted.

Extensions:

2.a: The location co-ordinates such as latitude and longitude are displayed using geo-location.

4.2.6 WORK ORDERS

Description: The inspector tracks the waste that is open and requires action on whether it can be disposed or not.

Precondition: The inspector has logged in.

Main flow of events:

1. The inspector selects the container id of the open item that requires action.
2. The inspector gets to see the container attributes such as location of the container, shipment date, material type and barrel type.
3. The inspector selects the ready to ship status based on the previous step.

Post condition: The inspector cannot make changes once submitted.

Extensions:

2.a : The container attributes are entered in the web component side and displayed on the mobile side.

4.3 SYSTEM ARCHITECHTURE

The hazardous waste finder application is based on three-tier architecture. By introducing the middle layer, the client side is handling only the presentation logic. Scalabilty becomes easier in three tier architecture. Since the middle layer protects the database layer, security aspect is not comprised. The data sent from the client side using AJAX, initializes XMLHttpRequest Object. The XMLHttpRequest object along with the serialized data and request parameter makes asynchronous communication with the server, where the request is handled and the data is retrieved. Finally, the XMLHttpRequest Object receives the XML data using a callback, processes it and updates the HTML DOM.

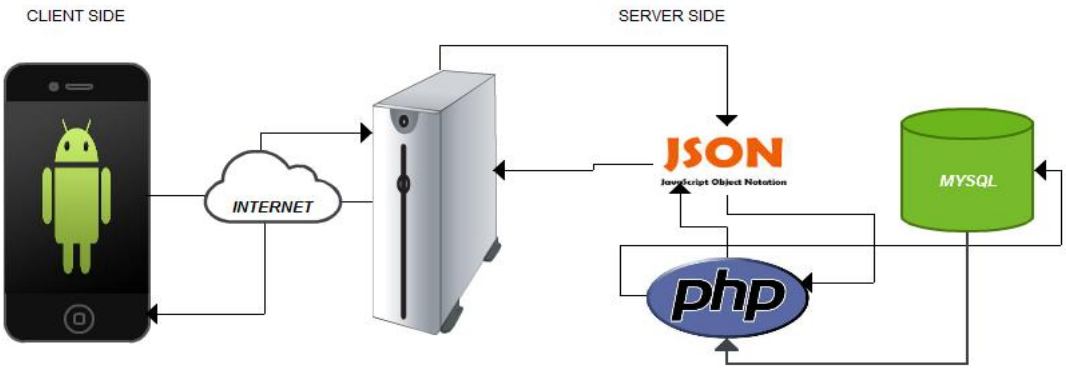

Figure 5 System Architecture

4.4 DATABASE SCHEMA

For the main centralized database, MYSQL was chosen for the reason being open and free, yet powerful and secure enough to warrant for enterprise class applications. The following schema represents the mobile component of the Hazardous Waste tracker database.

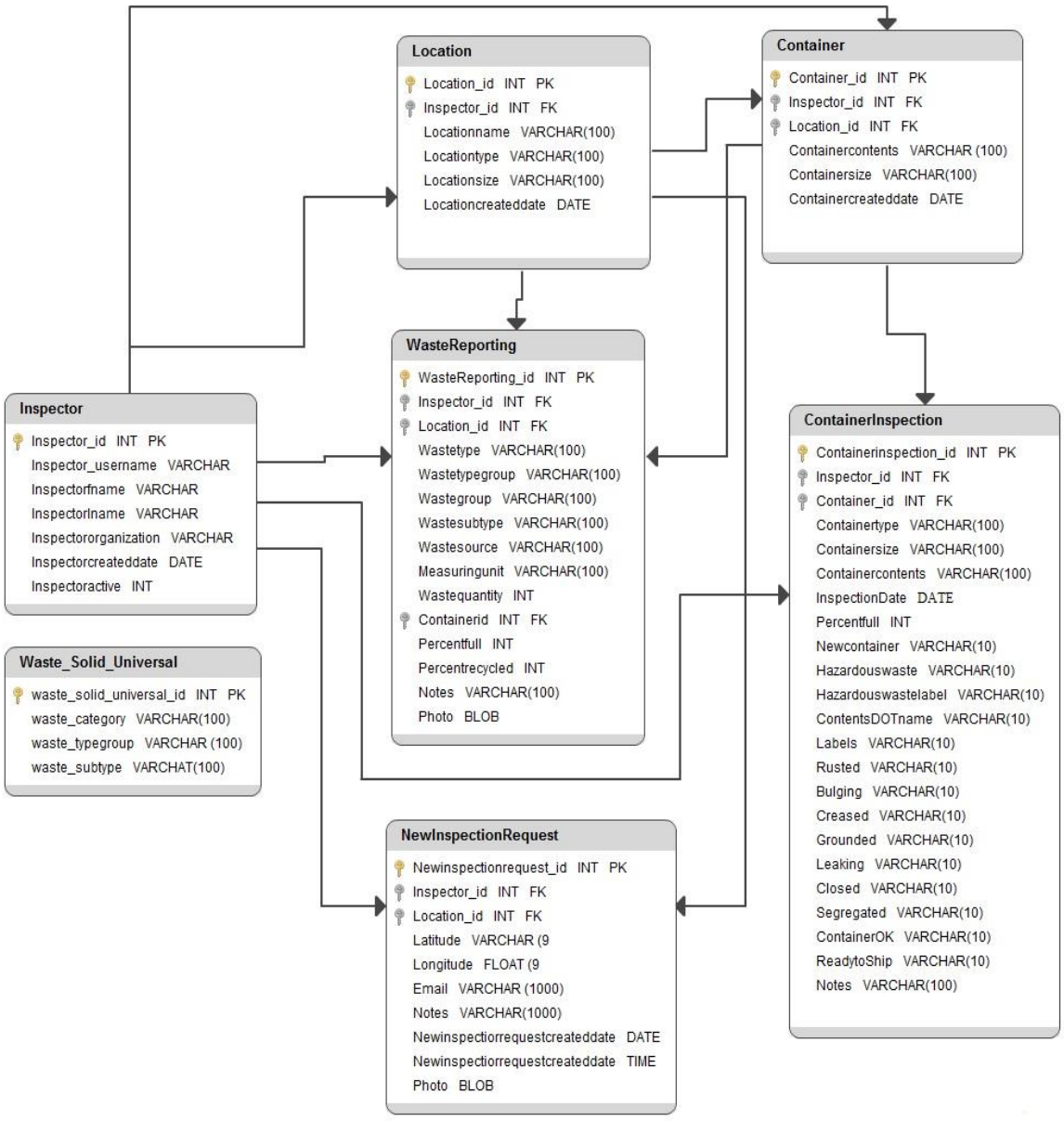

Figure 6 Database Schema for Hazardous Waste Finder

4.5 IMPLEMENTATION SCREEN CAPTURES

4.5.1 LOGIN MODULE SCREEN

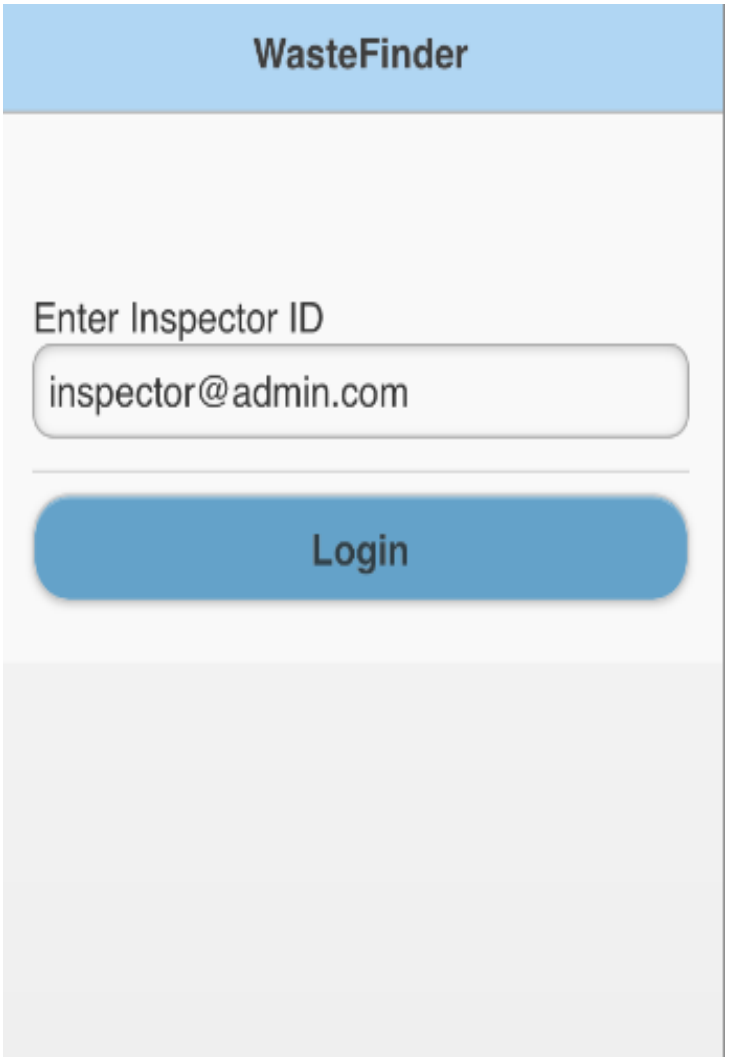

Figure 7 Login Screen

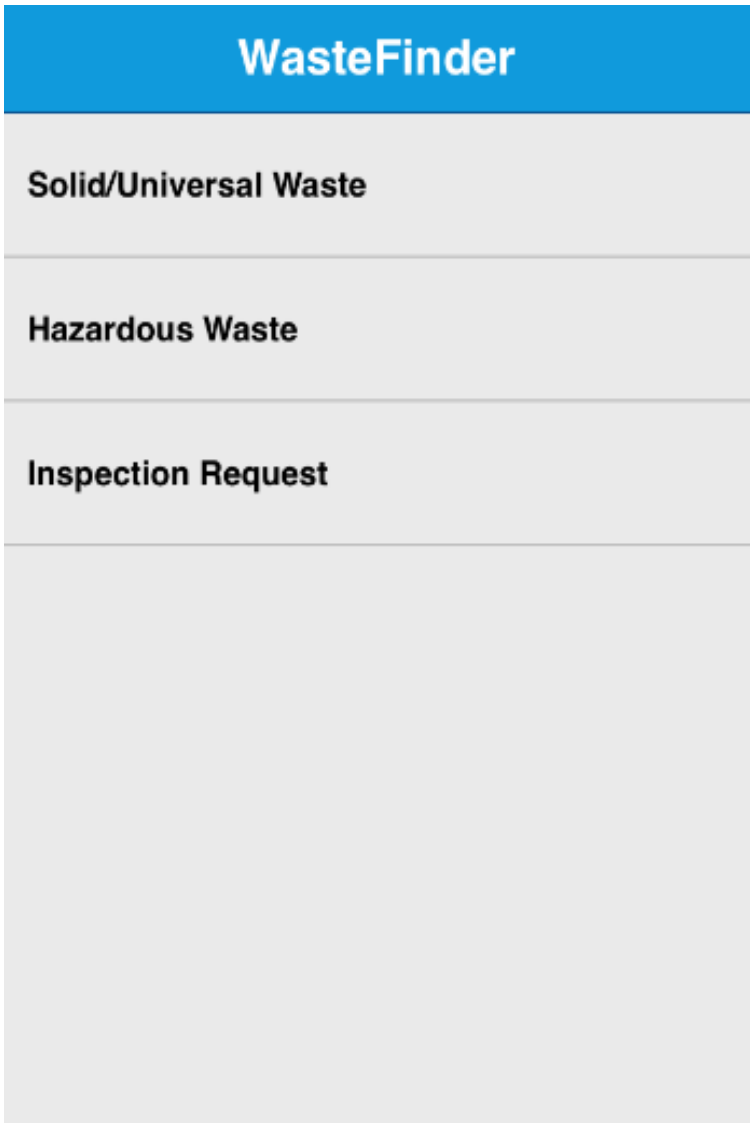

Figure 8 Menubar Screen

4.5.2 SOLID/UNIVERSAL WASTE TRACKING SCREEN

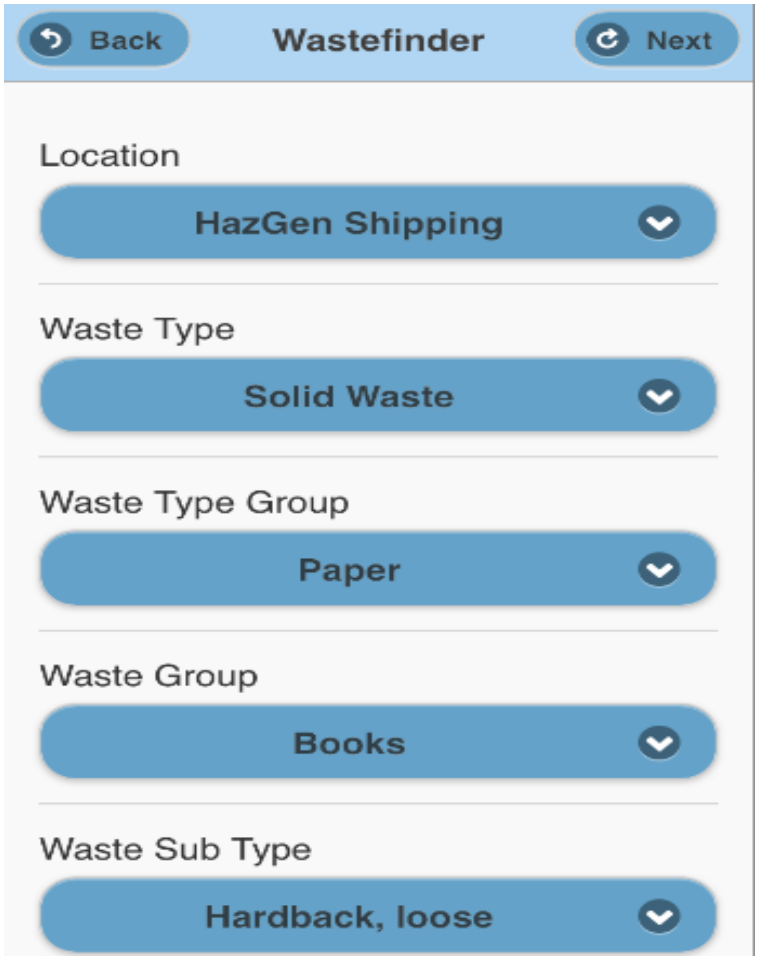

Figure 9 Solid/Universal Waste Tracking Screen - Part 1

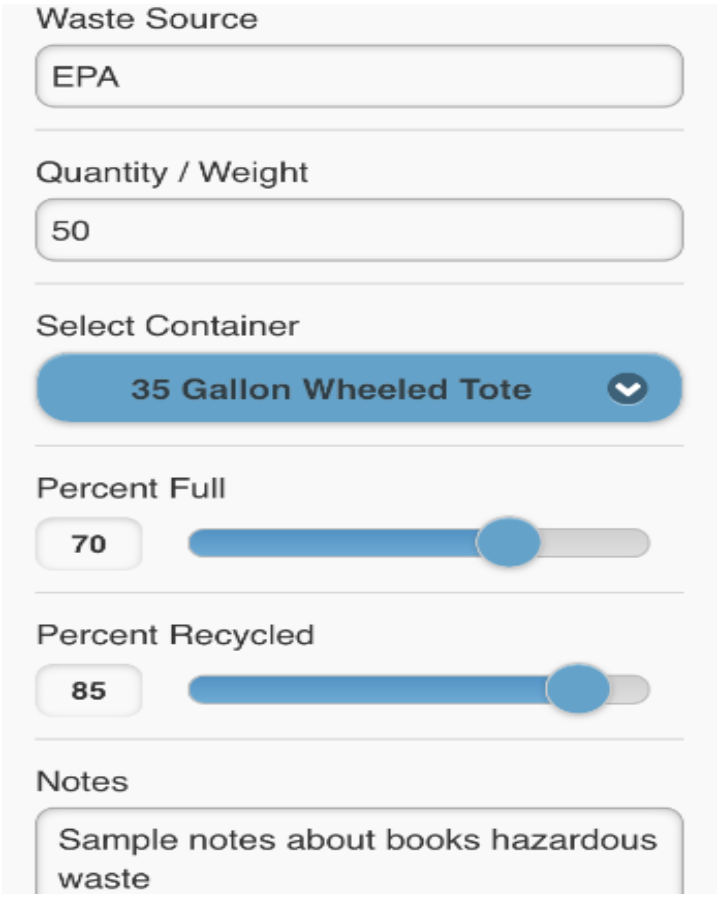

Figure 10 Solid/Universal Waste Tracking Screen - Part 2

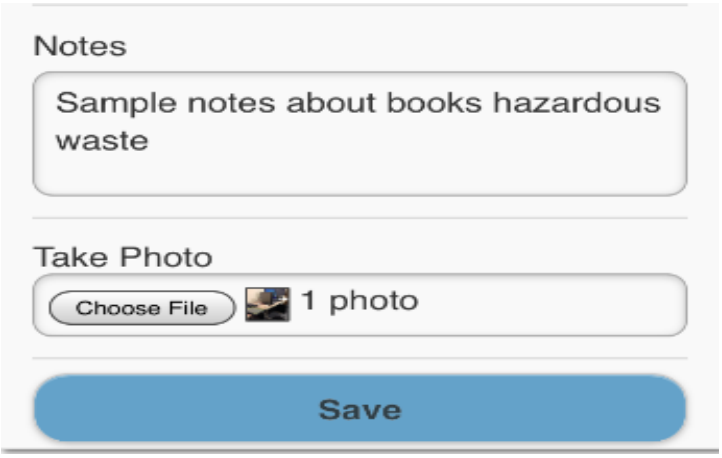

Figure 11 Solid Universal Waste Tracking Screen - Part 3

4.5.3 CONTAINER INSPECTION SCREEN

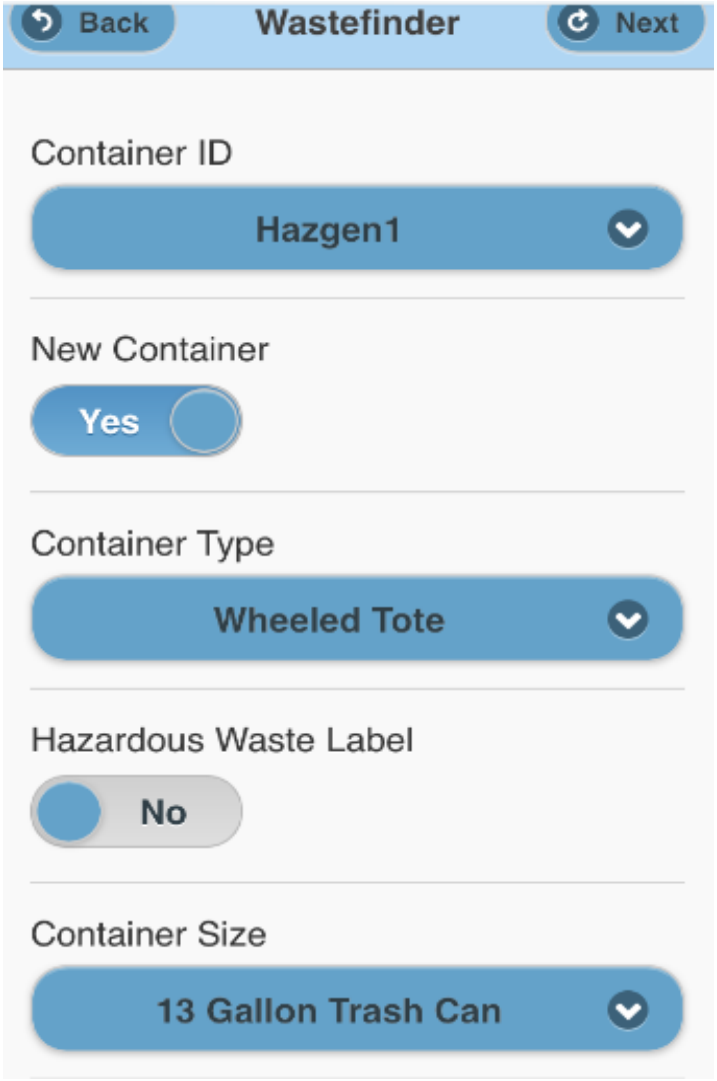

Figure 12 Container Inspection Screen - Part 1

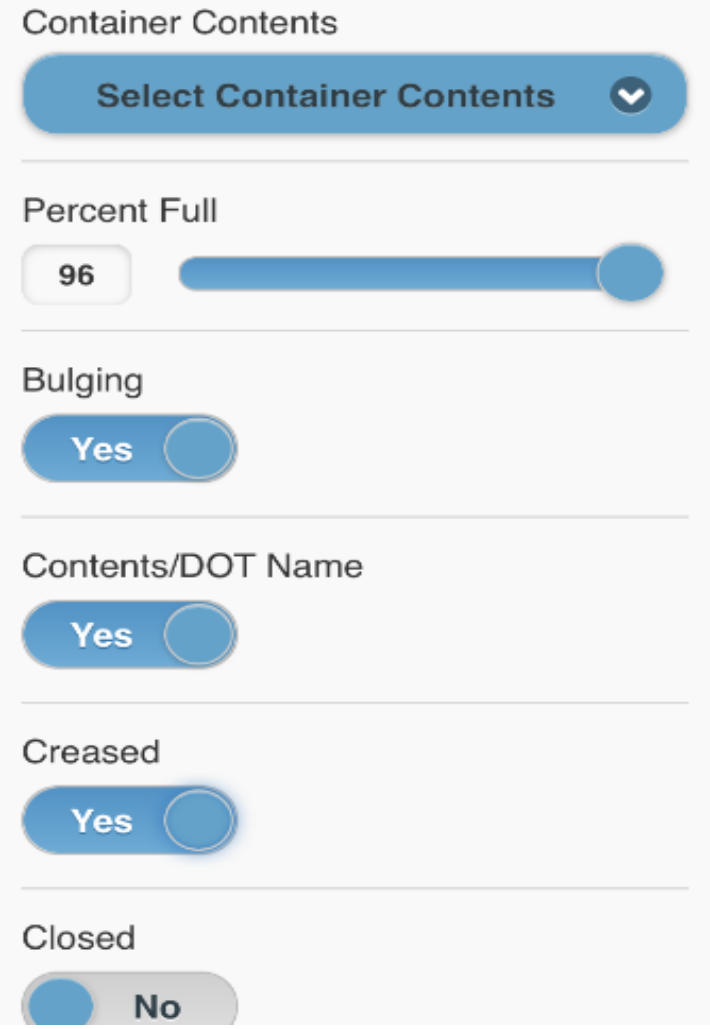

Figure 13 Container Inspection Screen - Part 2

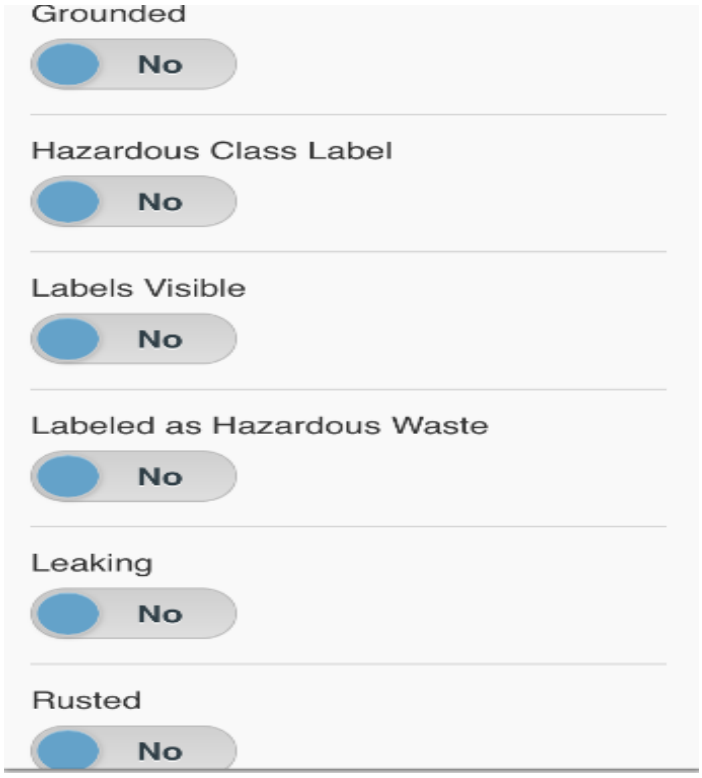

Figure 14 Container Inspection Screen - Part 3

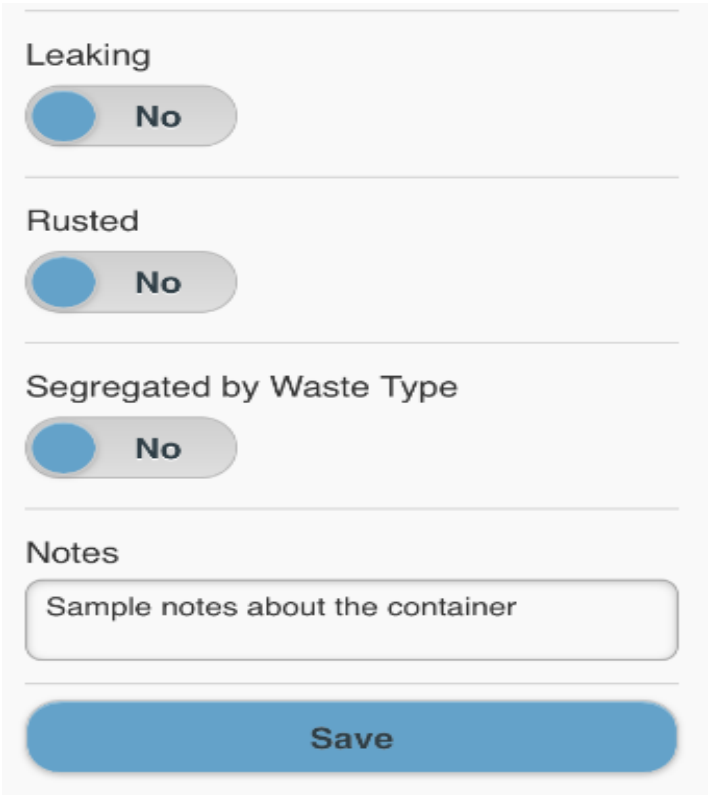

Figure 15 Container Inspection Screen- Part 4

4.5.4 AREA INSPECTION SCREEN

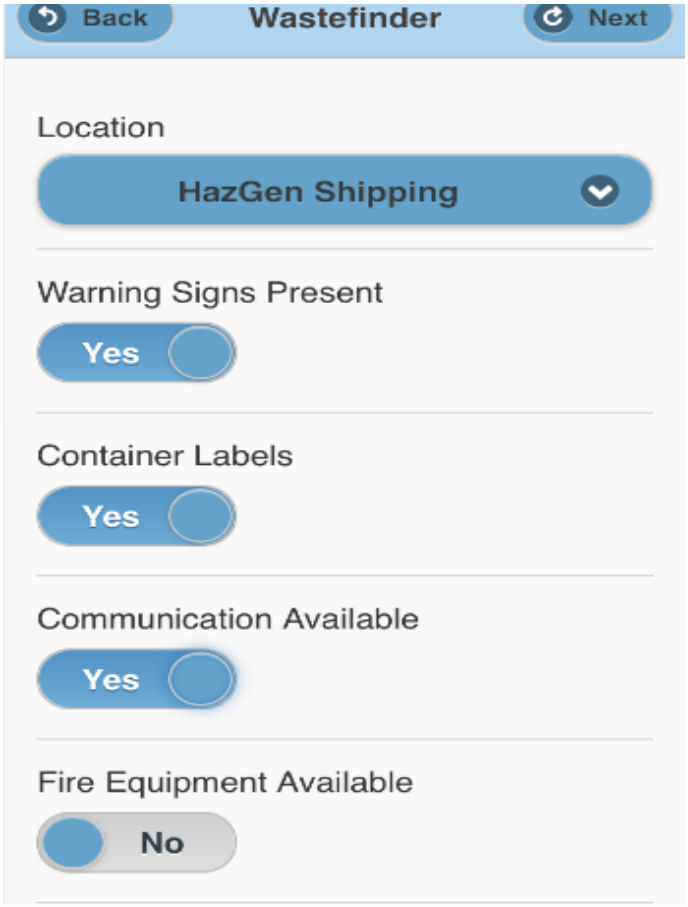

Figure 16 Area Inspection Screen - Part 1

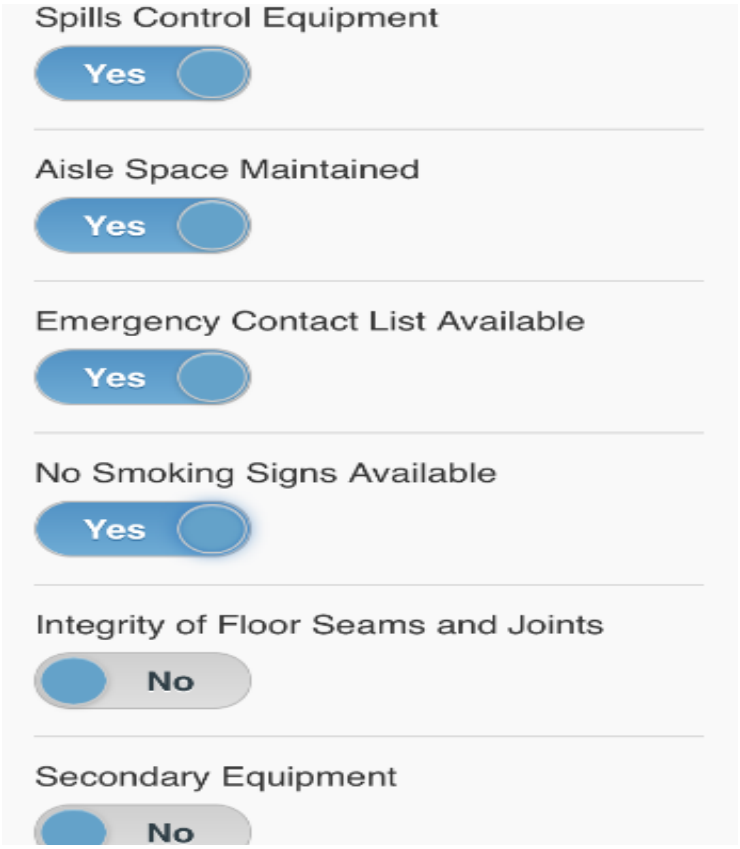

Figure 17 Area Inspection Screen - Part 2

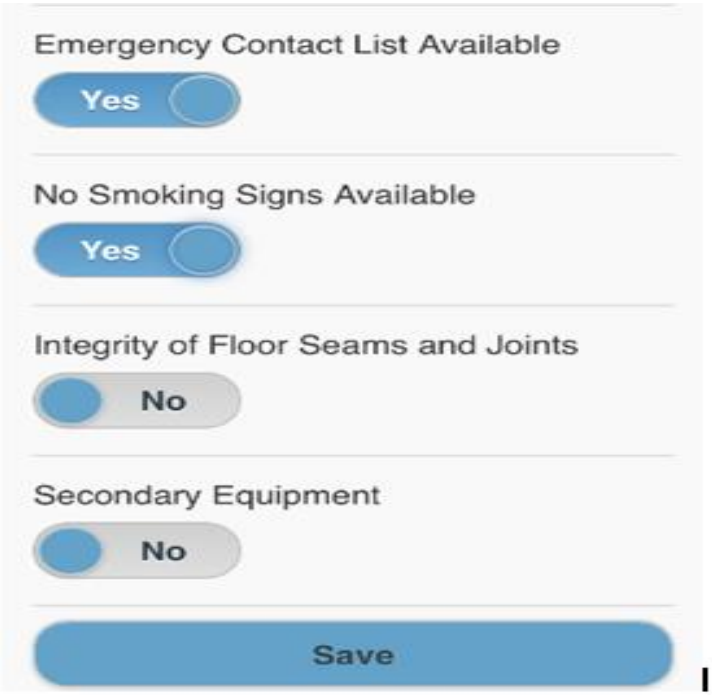

Figure 18 Area Inspection Screen - Part 3

4.5.5 NEW INSPECTION REQUEST SCREEN

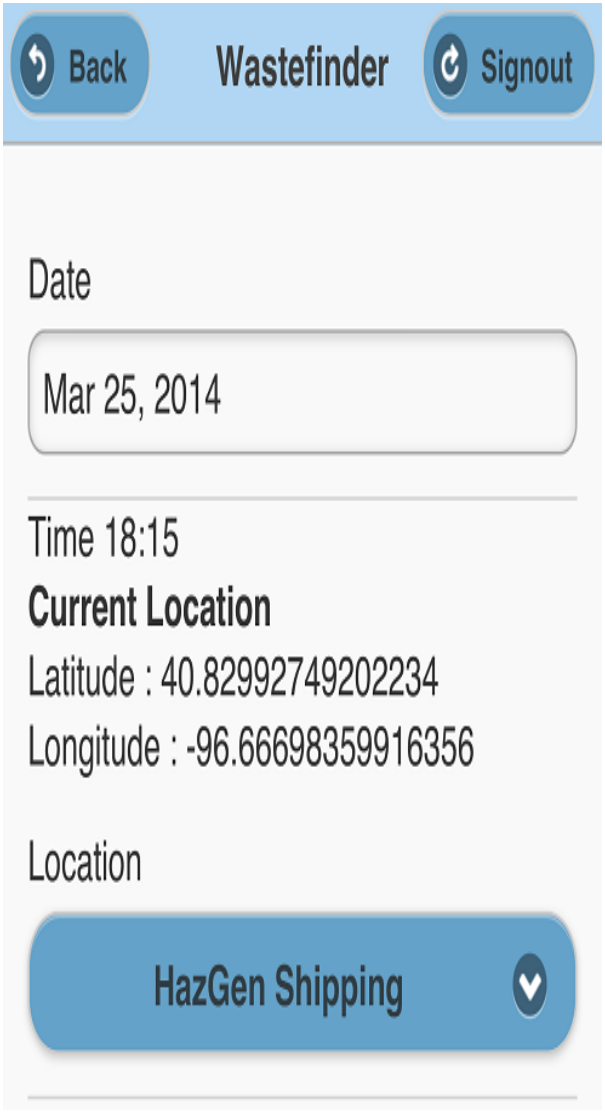

Figure 19 New Inspection Request Screen - Part 1

Figure 20 New Inspection Request - Part 2

4.5.6 WORK ORDERS SCREEN

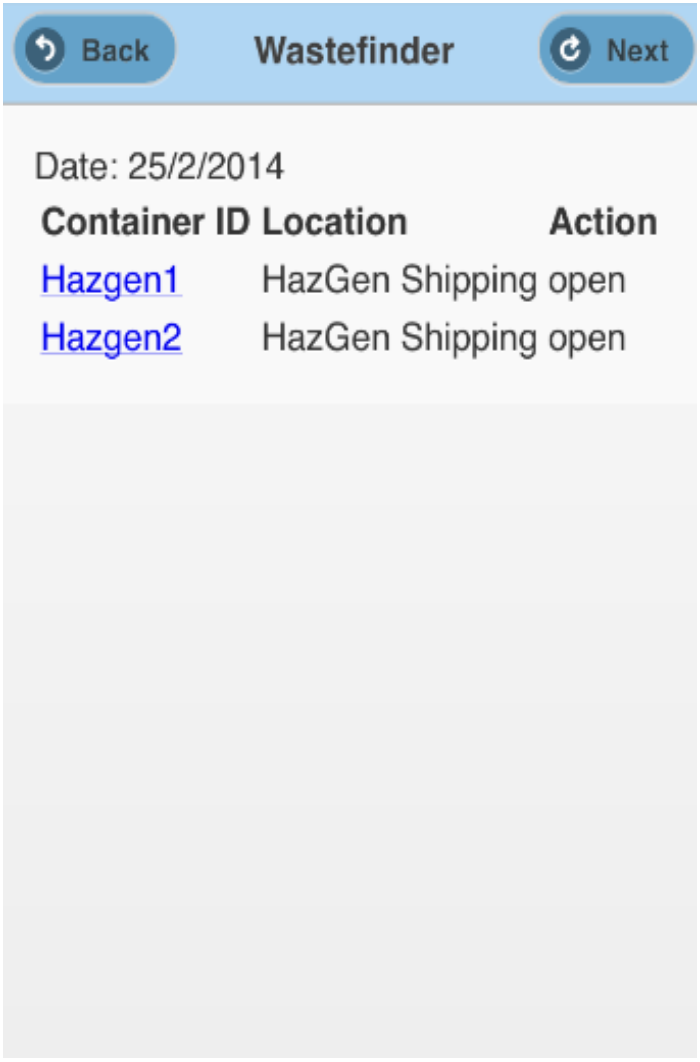

Figure 21 Work Orders Screen - Part 1

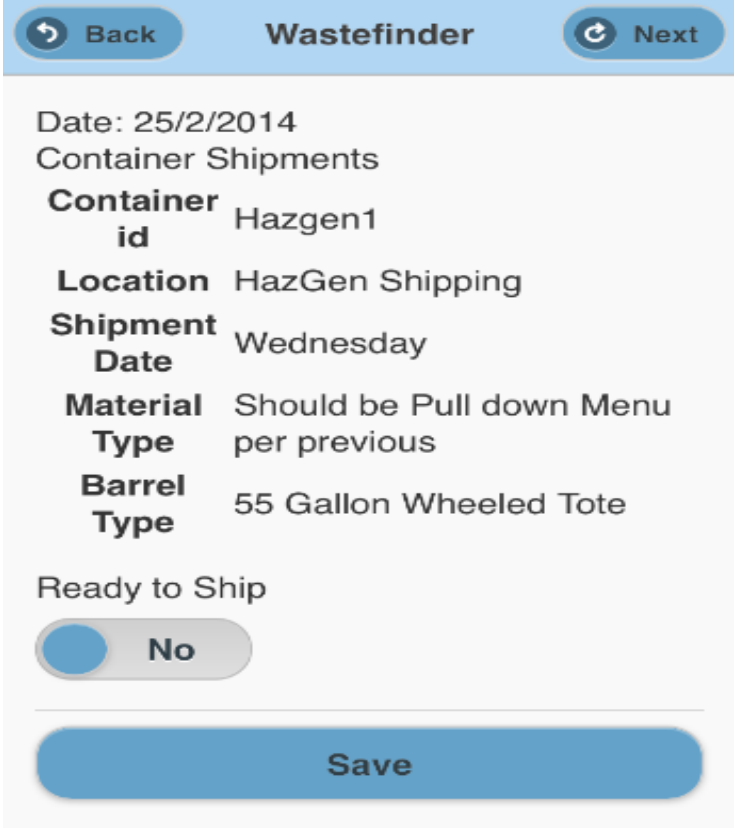

Figure 22 Work Orders Screen - Part 2

Figure 23 Work Orders Screen - Part 3

CHAPTER 5

TESTING

5.1 UNIT TESTING

Various modules have been tested individually and manually to check whether if the expected result is visible on the screen. Below is the listed table shows the test cases employed in testing the application.

S.No	Test Case Description	Expected Result	Actual Result
1.	On submitting a valid inspector user id	Login and show the home screen	Pass
2.	On submitting an invalid inspector user id	Prompt to enter valid user id	Pass
3.	On Clicking the solid/universal button without any user input in the home screen	Displays the solid universal waste page	Pass
4.	On selecting the location in the solid/universal screen	Displays the locations where solid/universal waste needs inspection	Pass
5.	On selecting the dropdowns for waste type group, waste sub type and waste group in solid/universal page	Displays the corresponding defined list pulled from the database	Pass
6.	On selecting the photo compatibility button in	Opens the camera and can take the picture	Pass

	solid/universal page		
7.	On clicking the save button in solid/universal page	Saves all the details to the database	Pass
8.	On clicking the hazardous waste button without any user input in the home screen	Displays the menu which has the work orders, container and area inspection menu	Pass
9.	On clicking the container inspection button	Displays the container inspection page	Pass
10.	On selecting the container id in the container inspection page	Displays the container name that needs an inspection	Pass
11.	On clicking the save button in container inspection page	Saves all the details to the database	Pass
12.	On clicking the area inspection button	Displays the area inspection page	Pass
13.	On selecting the location in the area inspection page	Displays the location that needs an inspection	Pass
14.	On clicking the save button in container inspection page	Saves all the details to the database	Pass
15.	On clicking the work orders button	Displays the list of container that needs action	Pass
16.	On selecting the ready to ship status	Updates the changes in the database	Pass

17.	On Clicking the new inspection request button without any user input	Displays the new inspection request form With date, time and location coordinates	Pass
18.	On selecting the location in the new inspection request form	Displays the locations that needs to be inspected	Pass
19.	On selecting the photo compatibility button	Opens the camera and can take the picture	Pass
20.	On clicking the logout button	Logs out of the application and goes back to the login page	Pass

Table 2 Test Cases for the application

5.2 COMPATIBILITY TESTING

The application was mainly designed for android phones as it helps the inspectors to carry out the hazardous waste inspection process. Different smartphones have various screen size and resolution; hence the application has been tested for its compatibility with different sizes on the emulator.

CHAPTER 6

SUMMARY AND FUTURE DEVELOPMENT

6.1 SUITABILTY OF PURPOSE AND EVALUATION

The application has been designed with the inspector in mind, who is aware of the hazardous waste inspection process. In that respect and also given the pace at which our environment is growing tremendously, it is believed that this application can serve as an ideal component minimizing the cost involved in hazardous waste management practices by generating useful waste records based on the tracking information. The purpose of the project was to understand the process involved in the inspection of hazardous waste management and come up with a solution that can make the whole job easier. With that in mind, the whole project was determined to be a success.

The choice of android as the initial client platform turns out to be positive one as it has become flexible with a lot of features. While iOS provides a similar set of features, the combination of android and Phonegap proved to be a more economical and budgeted platform to develop an application initially. In terms of the application code, it should be noted that the development followed standard practices in design and configuration and therefore it can be easily extended, if any need arises.

The Client code was built using Phonegap version 2.9 and android versions above 9 and given the fact that Phonegap is still in the early stages; it might become a hindrance for future expansion, if Adobe decides to add, drop or modify any features. However, a developer can easily find and fix the code related issues, if Adobe comes with proper documentation of the same.

6.2 ACHIEVEMENTS

6.2.1 TECHNOLOGICAL

The following technologies were learned to some depth during the course of development.

i. Client side
    a. HTML5
    b. Jquery - Mobile
    c. CSS
    d. AJAX
ii. Server-side and transfer
    a. Tomcat server
    b. JSON
    c. HTTP
    d. MySQL
iii. Phonegap
    a. Basics

b. Geo-location

c. Camera/Capture

6.2.2 PROJECT MANAGEMENT

The project served not only as a platform to try out a new emerging technology in the area of mobile application development such as Phonegap, but also to expand an array of skills including software engineering, time management, planning and some agile principles.

6.3 FUTURE DEVELOPMENT

From the application point of view, embedding a voice recognition system to insert notes about a waste can prove to be decisive in future. The application can be extended to other operating platforms such as iOS and Windows Phone with the use of Phonegap framework. The performance and security aspect of the application is one key area which be looked at as new versions of operating system are getting released every year.

REFERENCES

[1]. Hazardous waste - *http://www.epa.gov/wastes/hazard*

[2]. EPA Hazardous waste

http://www.epa.gov/superfund/students/clas_act/haz-ed/ff_01.htm

[3]. Wiki Hazardous waste -

http://en.wikipedia.org/wiki/Hazardous_waste_in_the_United_States

[4]. Listed wastes - *http://www.epa.gov/osw/hazard/wastetypes/listed.htm*

[5]. Flurry - http://blog.flurry.com/?Tag=Usage%20Statistics

[6]. CNN Money -

http://money.cnn.com/2014/02/28/technology/mobile/mobile-apps-internet/

[7]. Mobile Apps - *What customers really want*

http://offers2.compuware.com/rs/compuware/images/Mobile_App_Survey_Report.pdf

[8]. iOS vs Android - techland.time.com

http://techland.time.com/2013/04/16/ios-vs-android/

[9]. Breaking and Fixing Origin based Access Control in Hybrid Web/Mobile Application Frameworks by Martin Georgiev, Suma Jana and Vitaly Shimatikov.

[10]. Mobile Client Architecture - Tata Consultancy Services

http://www.tcs.com/SiteCollectionDocuments/White%20Papers/Mobility_Whitepaper_Client-Architecture_1012-1.pdf

[11]. Phonegap applications - http://phonegap.com/app/

[12]. Trace - An android software and client architecture.

[13]. Restaurant Finder - An android application, Spoorthy Pane.

[14]. PhoneGap Docs - http://phonegap.com/

[15]. Jquery Mobile - http://jquerymobile.com/

[16]. JSON - http://www.json.org/

[17]. HTML5 local Storage - http://diveintohtml5.info/storage.html

[18]. Building a Phonegap Application - http://coenraets.org/blog/phonegap-tutorial/