**Software Requirements**

**Specification**

**for**

**Asset Management with Barcode Tagging System**

**Version 1.0 approved**

**Prepared by Joanna De Guzman, Adrian Tobias, Miguel Mayor**

**Nov. 11, 2016**

**Table of Contents**

**Table of Contents .......................................................................................................................... ii**

**Revision History ............................................................................................................................ ii**

**1. Introduction..............................................................................................................................1**

1.1 Purpose ............................................................................................................................................ 1

1.2 Document Conventions.................................................................................................................... 1

1.3 Intended Audience and Reading Suggestions.................................................................................. 1

1.4 Product Scope .................................................................................................................................. 1

1.5 References........................................................................................................................................ 2

**2. Overall Description ..................................................................................................................3**

2.1 Product Perspective ......................................................................................................................... 2

2.2 Product Functions ............................................................................................................................ 2

2.3 User Classes and Characteristics ..................................................................................................... 2

2.4 Operating Environment.................................................................................................................... 2

2.5 Design and Implementation Constraints .......................................................................................... 2

2.6 User Documentation ........................................................................................................................ 2

2.7 Assumptions and Dependencies ...................................................................................................... 4

**3. External Interface Requirements ...........................................................................................4**

3.1 User Interfaces ................................................................................................................................. 3

3.2 Hardware Interfaces ......................................................................................................................... 8

3.3 Software Interfaces .......................................................................................................................... 8

3.4 Communications Interfaces ............................................................................................................. 8

**4. System Features........................................................................................................................9**

4.1 Event Scheduling.............................................................................................................................. 9

4.2 CRUD Operations........................................................................................................................... 9

**5. Other Nonfunctional Requirements .......................................................................................10**

5.1 Performance Requirements.............................................................................................................. 10

5.2 Safety Requirements ........................................................................................................................ 10

5.3 Security Requirements ..................................................................................................................... 10

5.4 Software Quality Attributes ............................................................................................................. 10

5.5 Business Rules ................................................................................................................................. 11

**6. Other Requirements ................................................................................................................5**

**Appendix A: Glossary....................................................................................................................5**

**Appendix B: Analysis Models .......................................................................................................5**

**Appendix C: To Be Determined List............................................................................................6**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

**1. Introduction**

**1.1 Purpose**

The purpose of this Software Requirement Specification, or SRS, is to give an in-depth and detailed description of what is needed for the Asset Management System for Maintenance Department of Weather.ph. This will show what the purpose of the software, as well as the development of the system software itself. This will also reveal constraints of the system, Interface Requirements, and System Features.

**1.2 Document Conventions**

* CRUD – stands for Create, Read, Update, Delete. Create means to make a record, read lets the user view the record, update allows the user to modify the record, and delete, as the name implies, lets you remove the record.
* MySQL Database - an open source RDBMS, or relational database management system, that is a software which holds the data and any information that comes from users.

* Attributes – The headers of the tables that label the different data. E.g. Employee Table has an attribute called *firstName* which is the first name of the employee.
* Tables – Specific group that holds information about a subject. E.g. Employee Table contains the records of the employees.

**1.3 Intended Audience and Reading Suggestions**

This SRS is intended for the developers of the Asset Management System, specifically the project manager, the systems analyst/systems developer, and the project researcher. Also, it will be relevant for the Head Technician of the Operations and Maintenance department of Weather.ph to read this SRS.

**1.4 Product Scope**

The Asset Management System for the Operations and Maintenance department of Weather.ph is a web-based application that implements a CRUD module to allow the creation, readability, update, and deletion of service reports and information of the weather stations. This Asset Management System also includes a barcode tagging function that allows easy access of the information of the weather station itself. As stated earlier, this software system is limited to only the Operations and Maintenance Department of Weather.ph.

The Asset Management System will include a web application that consists of a frontend and a backend. The frontend will display basic website information such as a home screen, about page, and a login page. The backend will consist of functions only accessible to those who have an Administrator account. These functions include CRUD modules for service reports, weather stations, and privilege management.

**1.5 References**

Yii 2 Documentation – Accessible through the internet; Link: http://www.yiiframework.com/doc-2.0/guide-index.html

Main Paper Documentation - Accessible through the internet; Link: http://projects2.apc.edu.ph/wiki/index.php/Project\_-\_Barcode\_Info\_-\_113

**2. Overall Description**

**2.1 Product Perspective**

The origin of the web application came from the suggestion of the client to replace the existing system that is being used today. The current system, an excel file holding all the data, is already obsolete in terms of availability and consistency. The web application, which would replace the existing system, contains specific functionalities that the current system cannot provide.

**2.2 Product Functions**

The functionalities of the web application are the following:

* Service Report, Weather Station, Event Create, Read, Update, Delete (CRUD)
* Event calendar and schedule
* Barcode implementation
* User level management

**2.3 User Classes and Characteristics**

There are two types of user level, Advance and Normal. These two types of user levels separate the privilege use of the CRUD operation on the data provided. The Advance user has full CRUD functionality on the Service Report and Weather Station entries while the Normal user cannot use the update and delete functionality. These user levels are granted by the admin and the admin only holds the power to change, create and delete a user’s level. A user without any type of user level would not be able to access any page unless that user is given a user level.

**2.4 Operating Environment**

The web application is currently being hosted by XAMPP Apache’s server and the database being used is a MySQL database. If the construction of the functionalities reach 80% (currently 60%), the web application would be integrated to Bluemix for hosting purposes. The hardware needed to access the web application is a normal computer with access to the internet.

**2.5 Design and Implementation Constraints**

Security considerations and implementations on the web application is to be added after the web application is finished. The functionalities of the web application are prioritized first per request of the client. The barcode phone scanner implementation would also be affect due to the barcode phone scanner is another system entirely.

**2.6 User Documentation**

A user manual will be created to guide the new users on how to use the web application. This would cover steps on how to create weather station entries as well as service report entries. Other functions like event creation and view is also covered in the user manual

**2.7 Assumptions and Dependencies**

The issues around the operating environment is that it cannot be accessed through the internet. The web application is still hosted locally in a computer. This would limit the access of the proposed users to use the system.

**3. External Interface Requirements**

**3.1 User Interfaces**

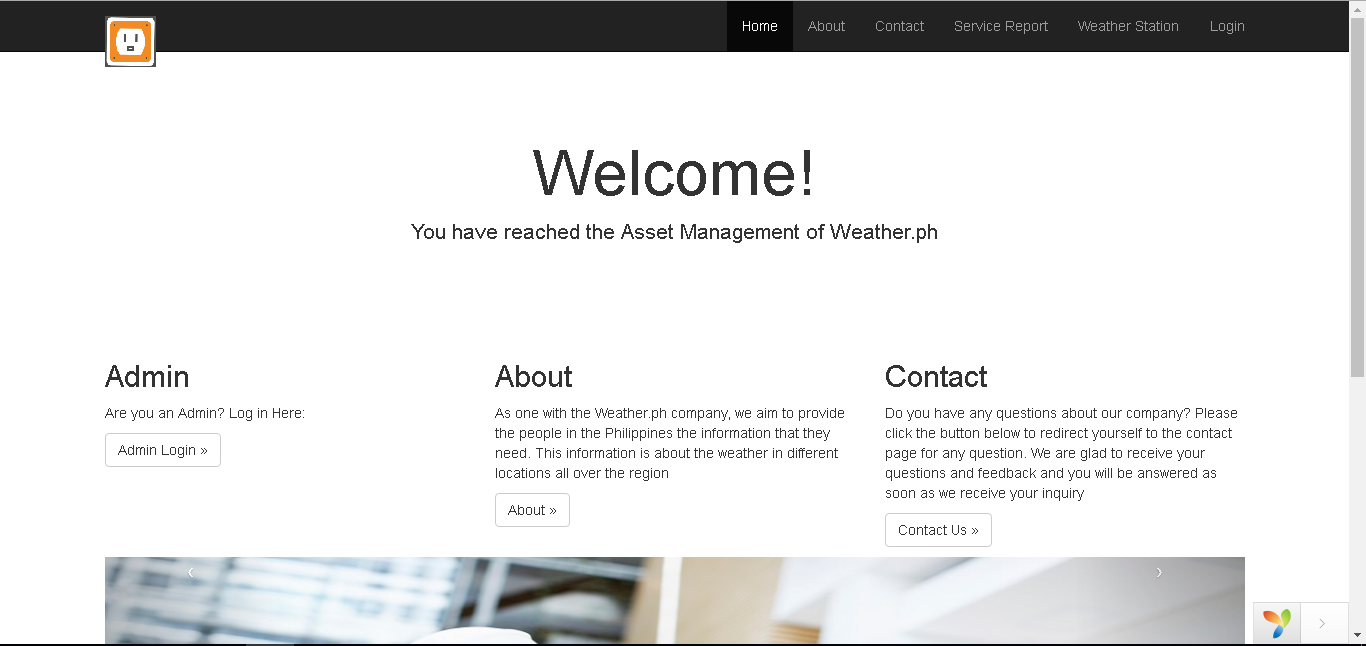
The first screen that should be visible is the Home Page which displays a “Welcome!” Header that includes several links leading to Admin Login page, About page, and the Contact Page. There is also a navigation bar at the top of the webpage (Figure 1).

When you click the Login webpage, a login page will appear. If you are an admin, you will be able to enter the required information for you to access the Service Report page, Privilege Management page, and the Weather Station page. Otherwise, the only accessible pages are the Home page, Contact page, Signup page, and the Login page (Figure 2).

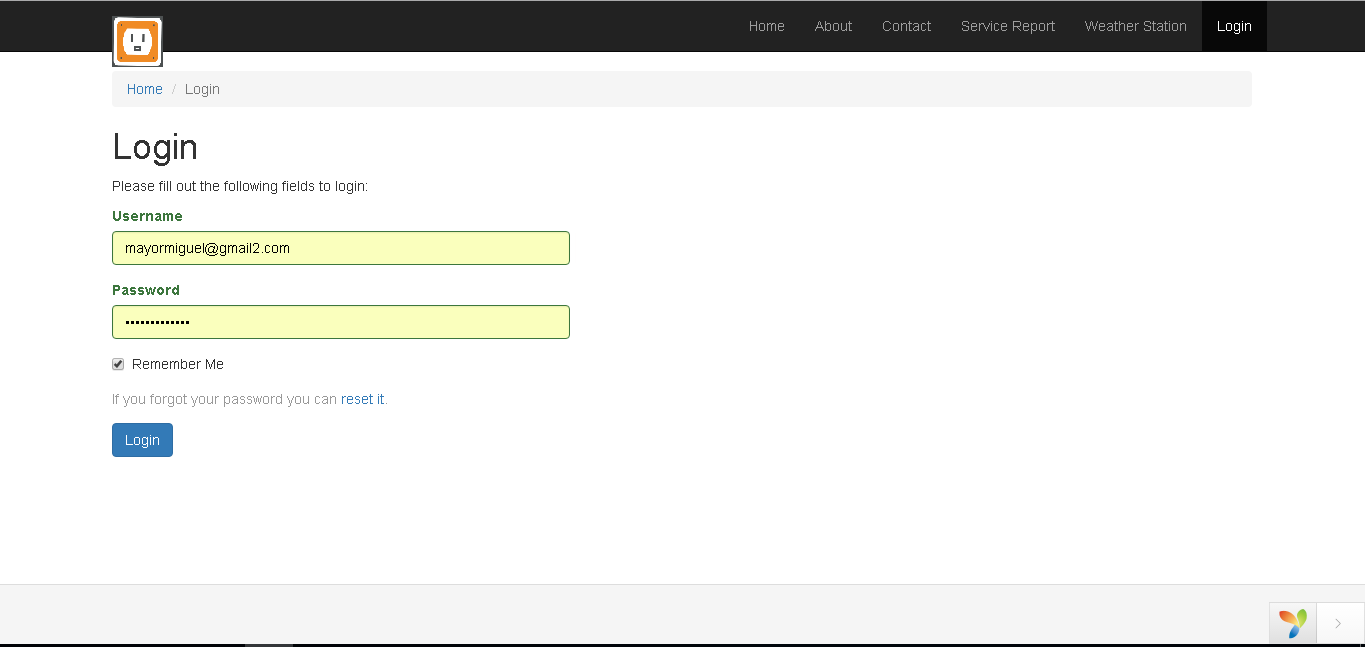
The About page contains brief information about the company, Weather.ph, as well as some logos of the partners (Figure 3).

The Contact page is where the user can send business inquiries or other questions by filling up a form and sending it to the company (Figure 4).

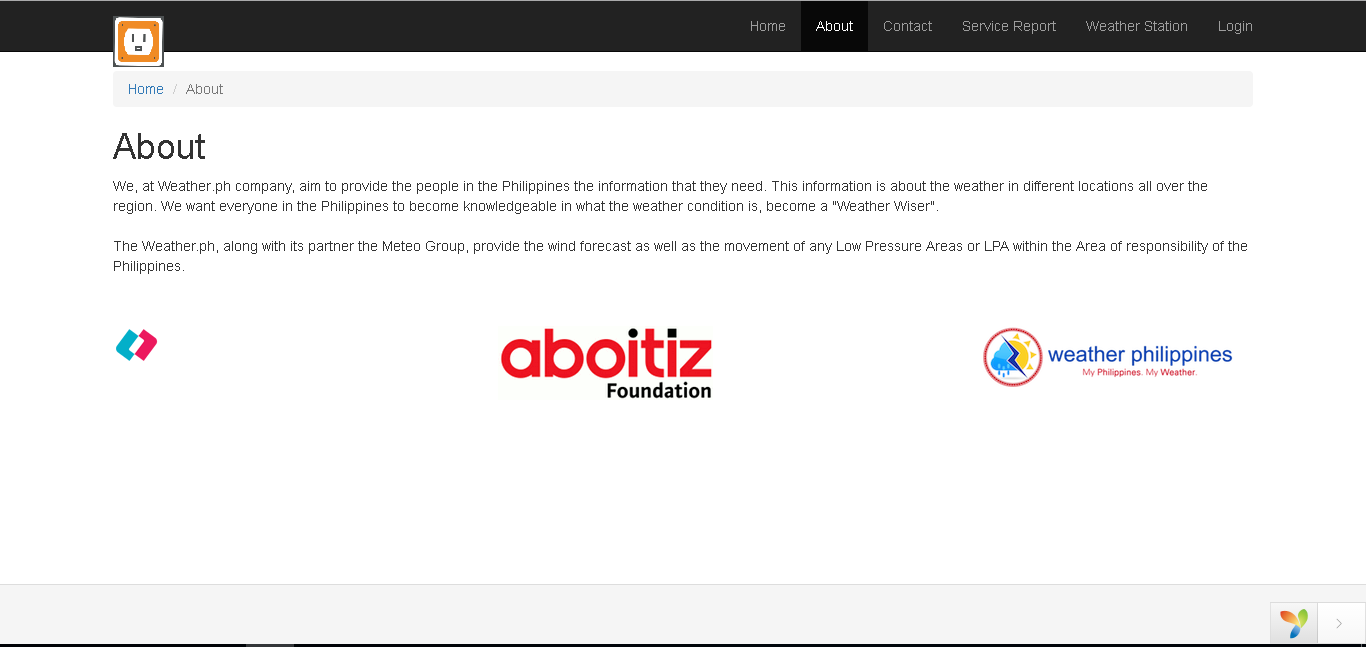
*Figure 1. Home page*



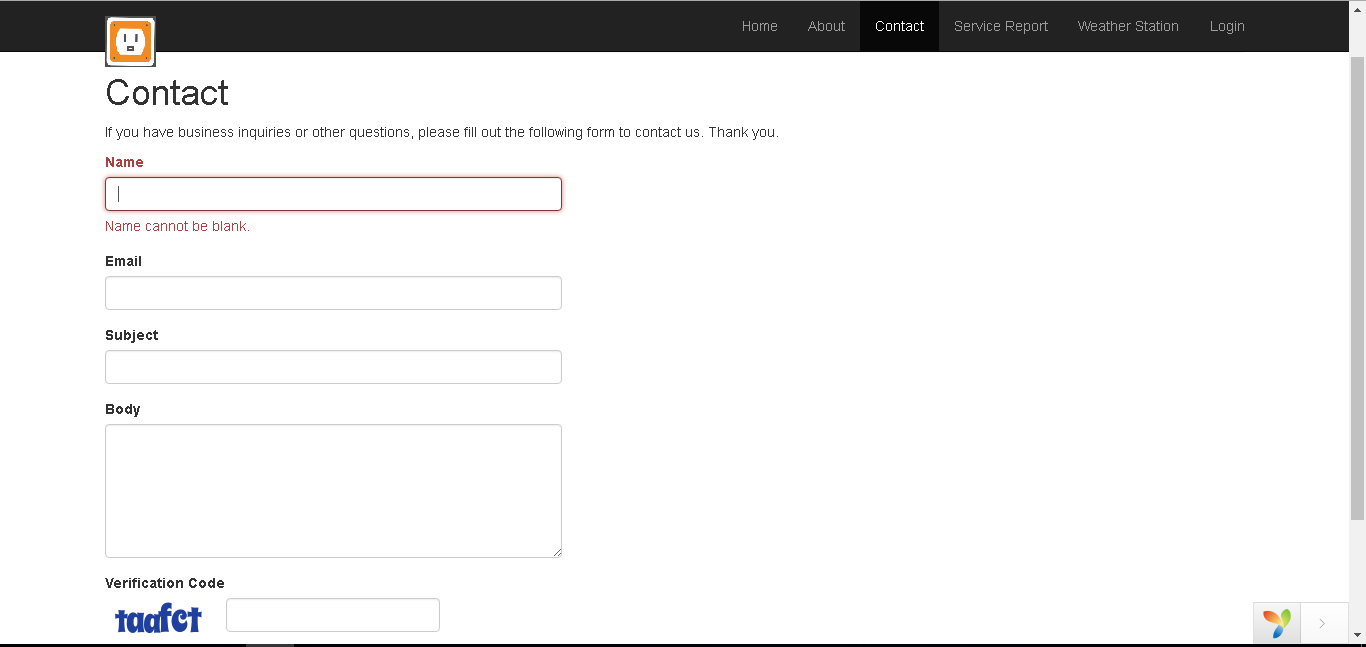
*Figure 2. Login Page*

**

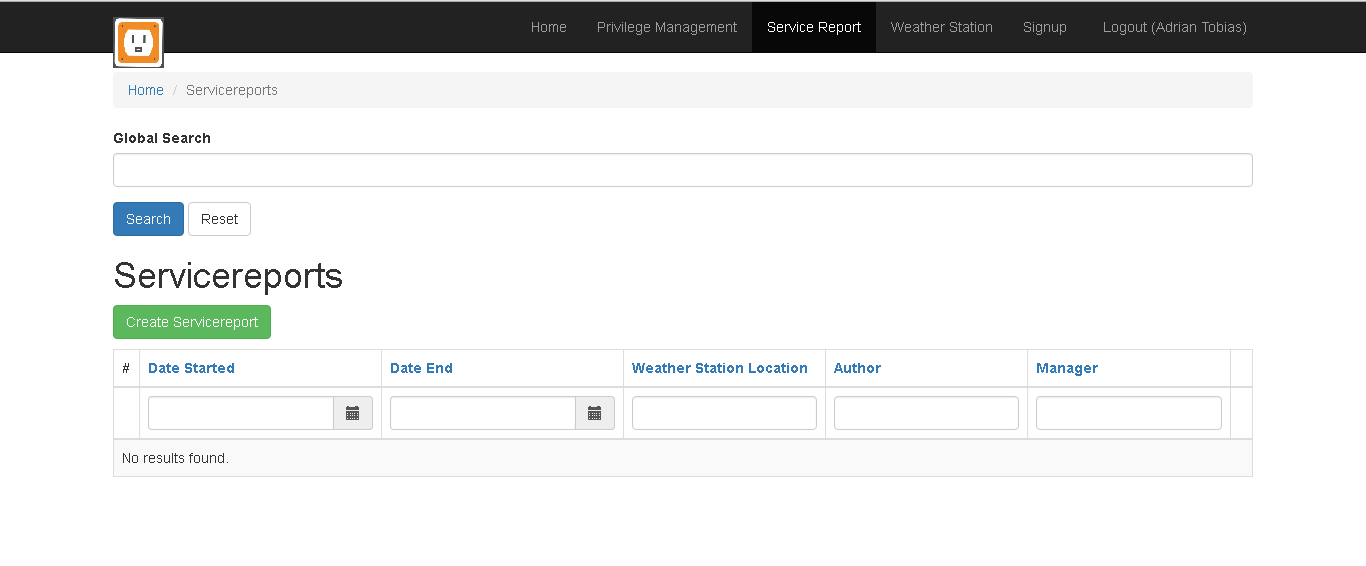
*Figure 3. About Page*

**

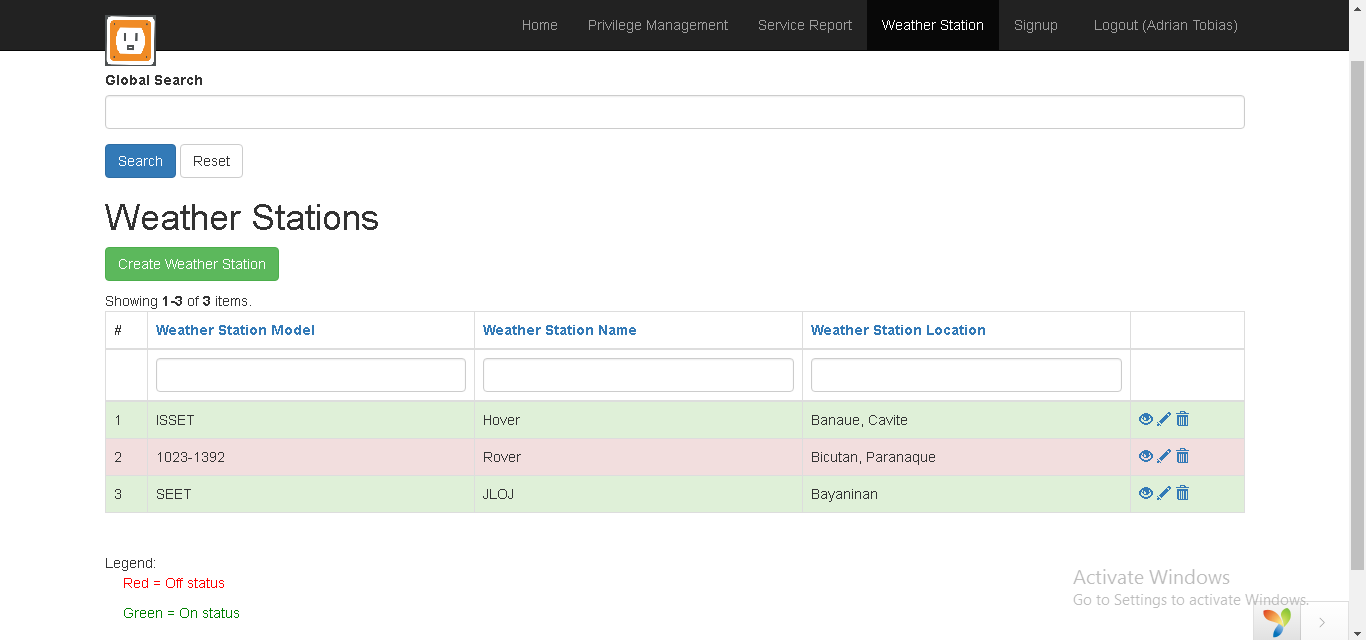
*Figure 4. Contact Page*

**

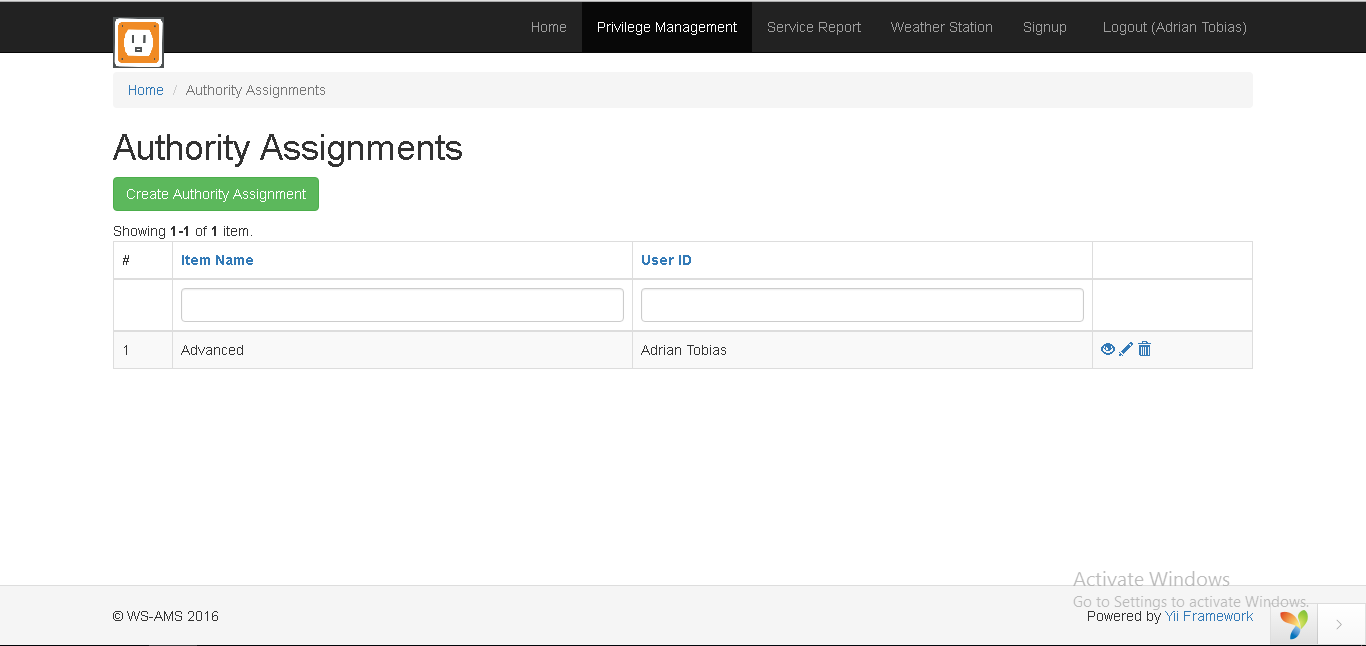
*Figure 5. Service Report Page*

**

*Figure 6. Weather Station Page*

**

*Figure 7. Privilege Management Page*

**

**3.2 Hardware Interfaces**

The devices that can run the web application is any tablet device, cellphone, and computer device that has internet access. The database server connection is managed through the hosting of the Apache server in the localhost computer

**3.3 Software Interfaces**

The web application connects to the database for the manipulation of data. Each session of the web application could view and manipulate the data in the database, either using phone, tablet, or computer as a medium. All functionalities are presented per the screen size and everything is adjusted after opening the web application itself

**3.4 Communications Interfaces**

To access or to communicate with the web application, the user needs to access the web application through the web browser. The communication standards that it would use is the HTTP, TCP and the IP standard. The data transfer rates from the user to the server depends on the speed of the internet. If 1Mbps speed is attainable, it is possible for the transfer rate upload to be at 100kb/s. One security risk that the web application is cross site scripting and unauthorized access to the web application.

**4. System Features**

**4.1 Event Scheduling**

4.1.1 Description and Priority

This feature has a high priority since the client stated that they want this type of feature because if this feature is a part of the system, it would be much easier to handle the people who will execute the maintenance on site. The benefit of this feature is that when implemented, the client and the other workers would have an easier access to the schedule of the event coming up this week or this month. The risk of not completing this feature is that the client would stick to the current system of scheduling which is assigning jobs through email

4.1.2 Stimulus/Response Sequences

The feature starts with the head technician creating a maintenance event date or any type of event. This would let the system prompt the head technician when is the start date and end date as well as the people who are to be assigned to the event. The people assigned are then notified that they have a specific task coming up. The system would display all the event either by month or by week

4.1.3 Functional Requirements

*For Normal User:*

REQ-1: Should have an account to view the event schedule

*For Administrator:*

REQ-1: Should have an admin account. If error occurs, the event would be cancelled and the form is needed to be filled up again for the event to be created.

**4.2 CRUD Operations**

4.2.1 Description and Priority

This feature has a high priority since all the operations on the data is on this one operation. The benefit of implementing this feature is that there would be an easier way to check and edit the data and information given. The risk of not having this feature would greatly delay the overall project since the whole project depends on this feature

4.2.2 Stimulus/Response Sequences

The feature starts with the head technician or normal user creating either weather station entry or service report entry. The normal user has the power to view and create functionality while the admin has four functionalities: Create, Read, Update, Delete. The system would depend on the user what to do next.

4.2.3 Functional Requirements

*For Normal User:*

REQ-1: Should have an account to view the event schedule and create an entry. If error occurs, present error 403.

*For Administrator:*

REQ-1: Should have an admin account to use all the functionalities of CRUD.

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

The web application of the system must run 24/7, for the administrators to be able to access the website at any given time, in case a need for change is required, or in case a problem has occurred.

The web application itself is not intensive with internet bandwidth, as each Web page’s size is about 40-60KB. The only aspect that would be intensive with internet usage is the feature that the data is real-time, which means that servers have to constantly update the data of the web application.

The web application could be accessed outside the local area network of the company, so that the site administrators could access the website anywhere. No VPN would be required to access the web application. The website must also not be indexed so that it would be unsearchable in search engines (e.g. Google).

**5.2 Safety Requirements**

To mitigate the chances of having data leakage, the user must be aware of his surroundings and keep his password to himself or herself. By not spreading the user’s password, the chance of having unauthorized access could be minimized

**5.3 Security Requirements**

Since the creation of the web application is at a rush, there is security holes involve in the database and the accessing of the pages. Although access control is implemented, there are ways on how to access the pages without being a user. The users could be signed up by the admin and those users need to log on to the system via the frontend part of the page. The security is needed to be upgrade to a standard which is defined by having possible defenses against XSS attacks as well as MySQL Injection attacks

**5.4 Software Quality Attributes**

The application should have the following attributes:

* Availability – to have constant access to the data within the web application
* Maintainability – to have a constant run of the web application and to keep it running and avoid it breaking down
* Portability – accessible through any device
* Reliability – having data which is constant and current

These following traits must be observed so that user satisfaction is achieved. Ease of use over ease of learning is observed during the creation of the web application

**5.5 Business Rules**

A normal user could only read and create entry in the service report, weather station and the user could only view the event schedule while the admin has full access to the pages, have the full CRUD functionalities, user creation, and event creation.

**6. Other Requirements**

The requirements for establishing a database is to have the company apply the tables of the web application’s database migrated to their database.

**Appendix A: Glossary**

* CRUD – Create, Read, Update, Delete; standard for any web application
* Error 403 – Common HTTP error. Forbidden access to a certain page
* HTTP - The Hypertext Transfer Protocol (HTTP) is an application protocol for distributed, collaborative, hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web.
* IP - The Internet Protocol (IP) is the method or protocol by which data is sent from one computer to another on the Internet.
* KBps – Kilobytes per second. Used as a unit of speed in terms of data transfer
* MySQL - is an open-source relational database management system
* TCP - The Transmission Control Protocol (TCP) provides a communication service at an intermediate level between an application program and the Internet Protocol
* VPN - A virtual private network (VPN) extends a private network across a public network, such as the Internet. It enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network.
* XAMPP - is a free and open source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P).
* XSS - Cross-Site Scripting (XSS) attacks are a type of injection, in which malicious scripts are injected into otherwise benign and trusted web sites.