SRTW'S MANAGEMENT SYSTEM

By
Maj M Kashif Majeed
Maj Ahmed Ali
Capt Hafiz-ul-Asad

Submitted to the Faculty of Computer Science,
Military College of Signals, National University of Sciences and Technology,
Rawalpindi
in partial fulfillment for the requirements of a B.E Degree in
Computer Software Engineering
JULY 2017

TABLE OF CONTENS

Analysis & Evaluation

Future Work

LIST OF FIGURES

ABSTRACT

SRTW’s MANAGEMENT SYSTEM

Role of SRTWs(Signal repair technical workshops) has increased manifold in view of additional responsibility entrusted to Signals for field repair of all radio communication equipment. The current inventory system in place in all the workshops in Pakistan Army is manual register entries. Therefore, an application needs to be developed for keeping record of all of formations equipment along with repair profile at each SRTW. The same should also be available at Signals Directorate for showing fused picture of all communication equipment date and repair activities undertaken at each SRTW to keep an analytical eye on the ongoing fitness of different types of signal equipment in use in the Army and to keep track of which equipment is mal functioning more than usual. Application should also allow generation of reports in different formats.

This web application will enable the user to quickly and easily check the progress of their corresponding signal equipment. Moreover, the user at the general headquarters will be able to generate different forms of reports for their information and subsequent analysis.

1. Introduction

The purpose of this document is to give the user a clear and precise description of the functionality and design details of SRTWs management system, a web based software application that would be built for digitalizing the inventory of Signal Repair Technical Workshops of Pakistan Army. Role of SRTWs (Signal repair technical workshops) has increased manifold in view of additional responsibility entrusted to Signals for field repair of all radio communication equipment. An application needs to be developed for keeping record of all of formations equipment along with repair profile at each SRTW. The same should also be available at Signals Directorate for showing fused picture of all communication equipment date and repair activities undertaken at each SRTW. Application should also allow generation of reports in different formats.

1.1 Purpose

This web application's goal is to enable the user to be able to quickly and easily check the progress of their corresponding signal equipment. Moreover, the user at the general headquarters will be able to generate different forms of reports for their information and subsequent analysis.

1.2 Document Conventions

- SRS: Software Requirement Specification.
- DB : Data Base.
- GHQ: General Headquarters
- SRTWs: Signal repair technical workshops
- OAS: Office automation system
- Fmn: Formation
- SMS: Signal Repair Technical Workshops Management System
- 2IC : Second in Command
- NCO : Non Commissioned Officer

1.3 Intended Audience and Reading Suggestions

- **Project Supervisor**: It will help to supervise the project and guide the team in a better way.
- **Development Team**: It will help the developer to develop the product and to trace back the functional requirements.
- **Testing Team**: It will help the testers to understand the constraints.

- **Users**: The potential stakeholders of the system, who are interested in submitting, repairing and maintaining states of the equipment.
- **UG Project Evaluation Team:** Evaluation committee which will evaluate the progress of UG Projects.

1.4 Scope

Using SRTWs management system, clients will be able to

- Register themselves by their unit/HQ IDs
- Perform login by their user name and password.
- Check Progress
- Update progressGenerate reports

1.5 Product Vision

For users, who wish to get their equipment repaired and the staff who are responsible for analyzing and information sharing with higher headquarters can keep track of the problems through SMS. **What** The SMS is an a web application that provides the users with interface to enter the submitted quipment, and while allowing the staff to get to know about it and respond to each submission. **Unlike** In the existing practice of physical visit to the concerned SRTW or make telephone call to the Office. **Our Product** By using SMS, user can not only save time by swift viewing of the status of their submitted equipment and onward resolution followed by the feedback; but also bring their status into the notice of the higher officials at GHQ signals directorate.

LITERATURE REVIEW

2. Overall Description

2.1 Product Perspective

This software project is in the context of a military product for digitalizing the inventory of SRTWs. It will enable the SRTWs users to digitalize their record and update it. The product will facilitate the users at the GHQ signals directorate by allowing them to generate reports of different categories of signal equipment submitted for repair. Moreover, it will also facilitate the end unit users to view the progress of their submitted equipment.

2.2 Product Features

2.2.1 Equipment Submission

The equipment will be submitted by unit technical parties manually at local SRTW.

2.2.2 Equipment Entry

The user at SRTW will enter the equipment details in the SRTWs management system.

2.2.3 Login/Access Right

The system will allow users to login, based on their role in Army (i.e. units, SRTWs, GHQ, Administrator), each user will have different features visible.

2.2.4 Record maintenance and Report generation

This feature will allow Administrator to maintain records of equipment submitted and repaired in SRTWs by different units/fmns. Different users will have different access level to database repository. The database will be designed in such a way to protect data integrity and confidentiality.

2.3 User Classes and Characteristics

The software has four types of users, System administrator, SRTW user, Unit user and GHQ Signal directorate user. All the three types of users have different access level to the system and its data and can perform functions assigned to their respective roles.

2.3.1 System Administrator

The admin user can create new user with prior permission of Organization user. It can also deem a user as active or inactive based on some policy. It will have access to all the data records.

2.3.2 SRTW User

The SRTW user will use the SMS in order to enter the details of submitted equipment, update its status and send message to the relevant unit/fmn to collect the repaired equipment.

2.3.3 Unit User

The unit user can only view the details and status of their submitted equipment.

2.3.4 GHQ Signal Directorate User

This user will be able to view and generate reports in different formats regarding the submitted and repaired equipment of different units and fmns.

2.4 Operating Environment

The Web based system of SMS shall run on the computer system with following specifications:

- OAS think client
- At least 1 GB of RAM
- At least 160 GB of free disk space
- Linux based operating system
- Pentium 4 or higher
- Color monitor and OAS subnet.

SMS should be managed with ORACLE database management system.

2.5 Design and Implementation Constraints

Constraints of the product are given below:-

- SMS will not work without OAS subnet connection.
- If the local data centre is offline then communication will not be possible.

- Communication and speed will be dependent on network infrastructure's specifications.
- Communication outside the OAS network will not be possible.

2.6 User Documentation

Final release will be accompanied with a user guide to inform users how to use SMS. User documentation that would be delivered along with the final product

2.7 Assumptions and Dependencies

- Basic assumption for development of SMS is that system should be available 24x7 since a report by GHQ user may be required at any time.
- The server will be able to handle a large number of requests especially when initially entered, as there will be a lot of record entries.
- Overall performance of the product will depend on the hardware infrastructure and network speed.
- User must know the language and User Interface for the better performance of the product.
- Limitations of the product must be kept in mind by the user.

3. External Interface Requirements

3.1 User Interface

SMS consists of a web based interface that will be used by all the users. The learning curve for the interface should be gradual, so as to make the users of the application feel at ease while also enabling the Administration authority to interact with the system conveniently.

3.2 Hardware Interfaces

3.2.1 Computer System

- System shall have keyboard input.

- System shall have mouse input.
- System shall have a monitor.
- System shall have a working OAS connection and the hardware requirements that come with it.

3.2.2 Web and Database Server

- To process requests and retrieve/store data.

3.3 Software Interfaces

- SMS should be able to run on any version of the following Web browsers: Microsoft Internet Explorer, Mozilla Firefox.
- Primary Operating System supported by SMS Interface will be LINUX/WINDOWS.
- SMS should be able to run on Websphere/Tomcat configured in a stable Linux machine.
- SMS should be work with ORACLE database management system.

3.4 Communications Interfaces

- System shall be connected to the OAS services.
- Communication between the Web Interface and the server will be through OAS subnet over a web browser.

4. System Features

4.1 Functional Requirements

4.1.1 The users shall need to use OAS to access the web application.

4.1.2 The users shall get a Login option when he/she visits the system.

4.1.3 System shall verify the login credentials from user table for logging in user must enter his/her credentials.

4.1.4 User will get logged in and home page will get displayed if user enters valid information.

4.1.5 System shall allow user to select register page option and register as a new member, if the person is new.

4.1.6 The user shall fill a form to register and become a member, some of the entries in the form are mandatory in order to authenticate the user. Entries of the user form are under the authority of the administrator only.

4.1.7 System shall authenticate newly created member against the given unit ID.

4.1.8 System shall provide password to members to get forgotten password through administrator only.

4.1.9 An administrator shall delete an account/member from the system's databases on request by the member.

4.1.10 System shall allow the unit members to view the progress of their submitted equipment.

4.1.11 System shall allow the SRTWs members to update their accounts. They are not allowed to keep any mandatory fields blank while updating.

4.1.12 System shall allow the GHQ signals directorate users to generate a report indicating total equipment being submitted and repaired in SRTWs monthly.

4.1.13 System shall allow the GHQ signals directorate users to generate a report indicating which category of equipment is being repaired most in SRTWs monthly.

4.1.14 System shall allow the GHQ signals directorate users to generate a report indicating which unit/formation equipment is being repaired most in SRTWs monthly.

4.1.15 System shall allow the GHQ signals directorate users to generate a report indicating the current state of total equipment being repaired in SRTWs.

4.1.16 System shall allow the GHQ signals directorate users to generate a report indicating the current state of any specific equipment being repaired in any specific SRTW.

4.2 Specific Requirements

4.2.1 The user shall have the ability to register using his unit ID.

4.2.2 The user shall have the capability to view progress of their equipment.

4.2.3 The system shall have the capability of sending update of repaired equipment to user units.

4.2.4 The user shall have the ability to generate reports of different categories of equipment submitted for repair.

5. Non Functional Requirements

5.1 The system's web interface will use the JSF framework.

5.2 The code of the system will be stored on local repository.

5.3 The search will be handled by Jasper or direct query. The database backend system in use will be ORACLE.

5.4 Development environment will be the IDE Netbeans.

5.5 The messages will be handled using 'Java messaging service/Netbeans'.

5.6 The system will use Iterative model (Scrum Framework) and Agile Methodology.

5.7 IDE inherent debugging will be used for bug tracking.

5.8 The system will offer a simple, attractive, user friendly user interface that a layman can understand.

5.9 Password to be used must be at-least 8 characters.

5.10 The system should take care of hacks like SQL injection. It should also not expose any sensitive information to normal users.

5.11 Data should not become corrupted in case of system crash or power failure.

5.12 Central server of the system must be able to handle all the incoming requests simultaneously i.e. around 1000 requests per second per host.

5.13 Speed of the system should be responsive i.e. response to a particular action should be available in short period of time. For e.g. on registering as a new user the notification about the approval is sent immediately to the respective user unit through OAS.

5.14 The web application will be available in english language.

5.15 Certain functionalities will be required, based on e the performance and response of SMS. SMS has to be efficient in its response and operation. The product domain requires that the software is optimized in terms of performance. The data flow should happen in the most efficient way.

5.16 In case of data loss, system will back up the data in GHQ data centre and will restore it as per demand

5.17 Users shall be required to log in to the SMS for their own credential information.

5.18 The system shall permit only authorized members to do administrator's task.

5.19 The system shall permit users to view only their own profile and data that are intended for them.

5.20 The System will provide confidentiality and integrity.

5.21 SMS should be available 24x7 as the data entry of report generation can be lodged at any time. If at all system is down so the servers will take about 15 minutes to start the SMS again.

5.22 SMS must support modifiability so any further improvements or features are easy to incorporate.

5.23 Different quality tests should be performed so that SMS is free from faults and perform according to requirements

DESIGN & DEVELOPMENT

1. Overview

The System architecture description section is the main focus of this chapter. It provides an overview of the system's major components and architecture, as well as specifications on the interaction between the system and the user.

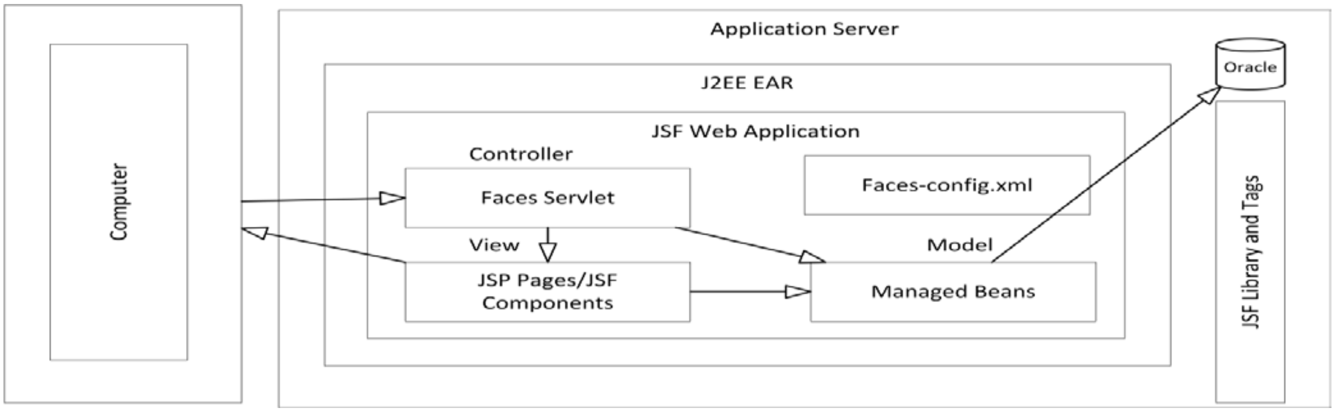
This document is organized in such a way that the detailed architecture of the system is provided initially. The design is further elaborated using diagrammatical representation of system components, classes, states, sequence of events, flow of events and their relationships. Furthermore user interface constraints have been discussed along with issues like product reuse, design decision and tradeoffs, pseudo code for components and appendices.

2. System architecture description

2.1 Overview of modules / components

Our system is designed with extensibility and scalability in mind. We are taking great care in designing a framework which can be updated easily. Many of the anticipated changes to our system in future phases will only require adding new types of data and changing the user presentation code to make use of these new data. The framework we have designed will only require "plugging in" these new types of data without refactoring the logic that passes the data over the network, retrieves and updates the database, etc. There are four basic, logical components of the system: the Database Engine, JSF Application, and the reporting Application.

2.2 Structure and relationships



2.2.1 Database Engine

- Existing open source software: ORACLE
- Hosts the backend database which is used for central data storage.
- Will provide all information to JSF application

2.2.2 JSF Application

- JSF holds all Java classes that of web application used to represent the data in memory.
- JSF application will be deployed in TOMCAT (container)
- Provides a uniform representation on both sides of the network connection.
- Client applications can use these objects to represent data, separating it completely from the server-side representation.
- Provides methods and procedures that can be invoked remotely by a client application via java server faces.
- Retrieve inventory data.
- Update inventory data.
- Generate reports.
- Every client request will be controlled by JSF beans.
- Reporting Application
- Uses Jasper Reports to present xhtml-based, web-accessible reports based on the data found in the database.

2.3 User interface issues

The user interface of the SMS will be divided into three main sections: The "Unit User " application, the "SRTWs User" application, and the "GHQ User" Application. These tools will allow the various users to accomplish the management of the software, and the management of the rest of their inventory. The personas described in the SRS represent the types of people who will make use of the SMS - and the purpose of this section is to describe how the user interfaces allow those people to do their tasks.

2.3.1 The Unit User Application

This application will reside on the OAS Server, and will be used mostly by the Unit User. It will have the following basic layout:

Login Screen - This first screen will allow *unit users* to access to the application.

Unit Equipment Search Screen - This screen allows *unit user* to search and view status/progress of submitted equipment.

2.3.2 The SRTW User Application

This application will reside on the OAS server, and will be used by the SRTW User. It will have the following layout:

Login Screen - This lets SRTW User to log in to the application.

Forum Screen - After login, the application will present this screen and will list of the available functionalities to the current user. The functionalities are :-

- Add new equipment data
  - A new screen will appear in which the SRTW user will be able to :-
  - Set attributes and values related to new equipment
- Search inventory
  - A new screen will appear in which the SRTW user will be able to :-
  - Search equipment by Category/name

- Update status of submitted equipment
  - A new screen will appear in which the SRTW user will be able to :
  - Update the status of the equipment based on unique ID

2.3.3 The GHQ User Application

This application will reside on the OAS server, and will be used by the GHQ User. It will have administrative rights with following layout:

Login Screen - This lets GHQ User to log in to the application.

Forum Screen - After login, the application will present this screen and will list of the available functionalities to the current user. The functionalities are :-

- Managing User Accounts
  - A new screen will appear in which the GHQ user will be able to :-
  - Add SRTW/Unit User accounts
  - Remove SRTW/Unit User accounts
- Generate Reports
  - A new screen will appear in which the GHQ user will be able to :-
  - Select a report type and view.

3. Detailed description of components

3.1 Component Overview

The following bulleted outline provides a basic overview of the purpose and architecture of our system's major components. The tables in the remainder of section 3 will provide more details on each component:

- **Database**: We will be using JDBC as our database software. It will contain tables to represent various Equipment data.

- **JSF Application**: The JSF will be a package of classes common to web browser application. The server software will use these class definitions to know how to store the objects in the database. Our server side will directly manipulate the contents of the database, based on commands from the SRTW and GHQ users. The Client side will be a simple web browser connected to the OAS network. Its purpose is to present data to the user as requested, and provide an interface so that the user can easily view and update. JSF application will be deployed in TOMCAT container.
- **Reporting Application** Jasper reports which is a separate module in JSF will be used to generate messages to units regarding the updation of status.

3.2 Database

Identification	Database Software
Type	Module
Purpose	Provides means of data management and storage for the server software.
Function	Takes ORACLE QUERY and UPDATE commands from the server software, and stores the data according to those commands.
Subordinates	The database will store data for each type of item in the class library in tables. inventory SRTW Equipment Category Equipment Name Work order Number Unit Fmn Date In Fault Progress Date Out Remarks User User Id Type Username Password

Dependencies	An administrative user must perform database setup functions, adding and modifying the structure of tables so that the server software can store the data as appropriate.
Interfaces	Apart from administrative setup, all modification of the database items will be performed via JDBC by the server application. The Reporting application will read the database through Jasper Reports to generate reports.
Resources	The database we are using is ORACLE, which will run on Army OAS. All tables will be created by an administrator (GHQ User). We will create a GUI which will allow manual entry of initial data in the following tables: Inventory, Users. Once these are entered, we will use the data held in these fields as a baseline for our user interface. Our management system will connect to the database using JDBC connectivity. Once the server gets a request for a certain data type from a client it will look up the table that is associated with the requested object, then parse the restriction clause and create a valid ORACLE statement. Once it has generated a valid ORACLE statement, the server will connect to the database, execute the query and get a java ResultSet back. At this point, it will convert the ResultSet to the Object that was requested by the client and send it back to the client via Sockets and ObjectStreams over the network. Calls to the database should not take longer than (but should be much less than) 30 seconds and transfers of the each Object over the network should not take longer than 1 minute. Conversion of a ResultSet to a Java Object should take less than 1 second. Data retrievals in general should be less than 2MB.
Processing	Following types of **ORACLE QUERRIES** will come from the client:- Select Update Create Table
Data	The data in the database will be filled by the SRTW user manually.

3.3 Reporting Application

Identification	Jasper Report Package
Type	Package
Purpose	The Report Package will reside on the server and generate reports requested by the client side.
Function	The Server will get a request from the client in the form of Reports.run*ReportName*(Arguments). The server will keep the incoming socket open so the generated report can be sent back. After a request is made, the ReportPackage will call our JasperInterface, which will be created so a programmer doesn't have to be familiar with the JasperReports package in order to write reports. The run*ReportName* call will know already know the ORACLE query and have the definition of that specific report contained within. This will then call JasperInterface.generateReportPDF/HTML*(ReportName,outputfile)* Once report is generated, the server will send the resulting output file back to the client over the same connection that the request was made. The report output will be stored on the server in a folder called reports.
Subordinates	This will take a JasperReports report specification, and will access the data to fill that report.
Dependencies	This reporting application is just a package of pre-defined reports that the server can call once a request is made by a client computer. A report should be generated and sent over the network in less than 5 minutes time but should be around 30-60 seconds from request to transfer.
Interfaces	Report requested from client: Reports.run*ReportName*(Arguments), Server will respond to client a keepalive signal as long as the connection is kept and the report is still being generated. If the report has a problem or cannot run, the Exception that is thrown will be passed to the server and it will send an error code back to the client. Server creating report with our defined interface. JasperInterface.generateReportPDF/HTML*(ReportName*, *outputfile)*
Resources	Our application will run on OAS network and use Java predefined sockets. The Reports Package requires that our server is running. It will also communicate with an ORACLE database which will be running and accepting JDBC connections. ORACLE must have the users who will login to our client application to have their name and password setup the same as in their client application. ORACLE may not need to be running on the same machine the server is running on. But it must be accessible over the network. To run a report and transfer it to the client, it should take less than 5 minutes but should be more like 30-60 seconds if good network conditions permit. The

	file sent will be of the type requested by the client.
Processing	Reports.run*ReportName*(Arguments) Parse argument list that may include a date-range, users (or clients) to run the report on, depending on what type of report is being requested. Make Report Call to interface class: JasperInterface.generateReportPDF/HTML*(ReportName*, Args, *outputfile)* Take arguments stated above and pack them into a HashMap. Make the above call to Jasper Assistant sending it also a list of the reports arguments. If report is generated: Send File back over the same socket, if report didn't work. Return a proper error code back to the client. Close socket. If client knows about a certain report it can use the getFile(filepath) which is a part of server's implementation.
Data	Arguments will be contained in a HashMap, which can be sent to JasperReports and retrieved from report specification. The report specification file will be held in a folder on a /ReportSource file on the server. All other data is held in the Database or contained in other classes or methods.

3.4 JSF Application

Identification	Framework
Type	program, package
Purpose	To represent data on a GUI To control the business logic Database interaction Class packages
Function	The server program will run on a seperate machine as the database engine The server continuously listens on a known port for client connections. It creates a new thread to handle each connection.

	Retrieve inventory data Update inventory data Logs a user into the system, and attempts to connect to server. Allows the user to see a list of their equipment, Allows the user to update the status. Generate reports
Subordinates	The server will rely on the class library of JSF.
Dependencies	The server depends on the database engine. It relies on the implementation of the Data Object classes in the JSF. The client side rely on the server to retrieve and synchronize inventory data and to run reports.
Interfaces	The server will listen on a predefined OAS intranet. It will use Java's Object Stream along with the serializable Data Objects in the JSF to pass information across the network The Client browser application uses controller class for interaction with the user. Uses the database class for interaction with a server.
Resources	As previously mentioned, the server will use the OAS network, relying on Java's Socket and Server Socket classes to do this. It also relies on Java's Object Stream class and Java's serialization mechanism.
Processing	Login filter class will control all the logins and security Database class will deal with the communication with database Controller class is the all functionality class
Data	The server uses the ORACLE database engine to store data, and is ignorant to the specific logic utilized to load, store, and manipulate the data. The server acts only as a communication device between the clients and the database.

4. Reuse and relationship to other products

4.1 Java and Java tools

From the beginning we set a goal to make use of any existing code to avoid wasting time duplicating other’s work. We also decided to use open source or freeware solutions wherever possible. Because we are creating software for a defined application, it is possible to use open source technologies in each area. First, we decided to implement our code in Java because it is free and allows us to run on a wide range of operating systems. Next, we chose a development environment that would allow us to edit our Java code. Eclipse was chosen because it is free and has many plug-ins which allow us to use already existing applications.

4.2 Database and Reporting

With these decisions in place, we needed a database on which to store our records. We chose ORACLE, because it is open source and above all it's the database used with the OAS in Army. We chose JasperReports as the means to generate reports because it is a freeware/open source package written in Java that generates reports in XML, PDF, or HTML files.

5. Design decisions and tradeoffs

5.1 Java/.NET Debate

The first issue we dealt with as a group was whether to develop in Java or .NET. Two of our members wanted to develop in .NET because they are familiar with it and enjoy working with it more. Also, PDA development is easier with .NET. However, while some members of the group do not have .NET experience, all have some Java coding experience. Another important issue we considered in selecting a code was the fact that .NET does not work in all standard operating environments. Above all once we approached GHQ then we came to know that in OAS all applications are implemented in JAVA.

5.2 Three Tier Design

Deciding how to judiciously divide the project between all team members was another design issue. We finally decided on a 3 tier design, which is an application program organized into three major parts, each of which is distributed to different places in a network.

The three parts are:

1. The JSF application
2. The reporting application
3. The database and programming related to managing it

A 3-tier application uses the client/server computing model. With three tiers or parts, each part can be developed concurrently by a different team of programmers. Because the programming for a tier can be changed or relocated without affecting the other tiers, the 3-tier model makes it easier for an enterprise or software packager to continually evolve an application as new needs and opportunities arise. Existing applications or critical parts can be permanently or temporarily retained and encapsulated within the new tier of which it becomes a component. This design idea was very appealing to our team, especially for portability purposes.

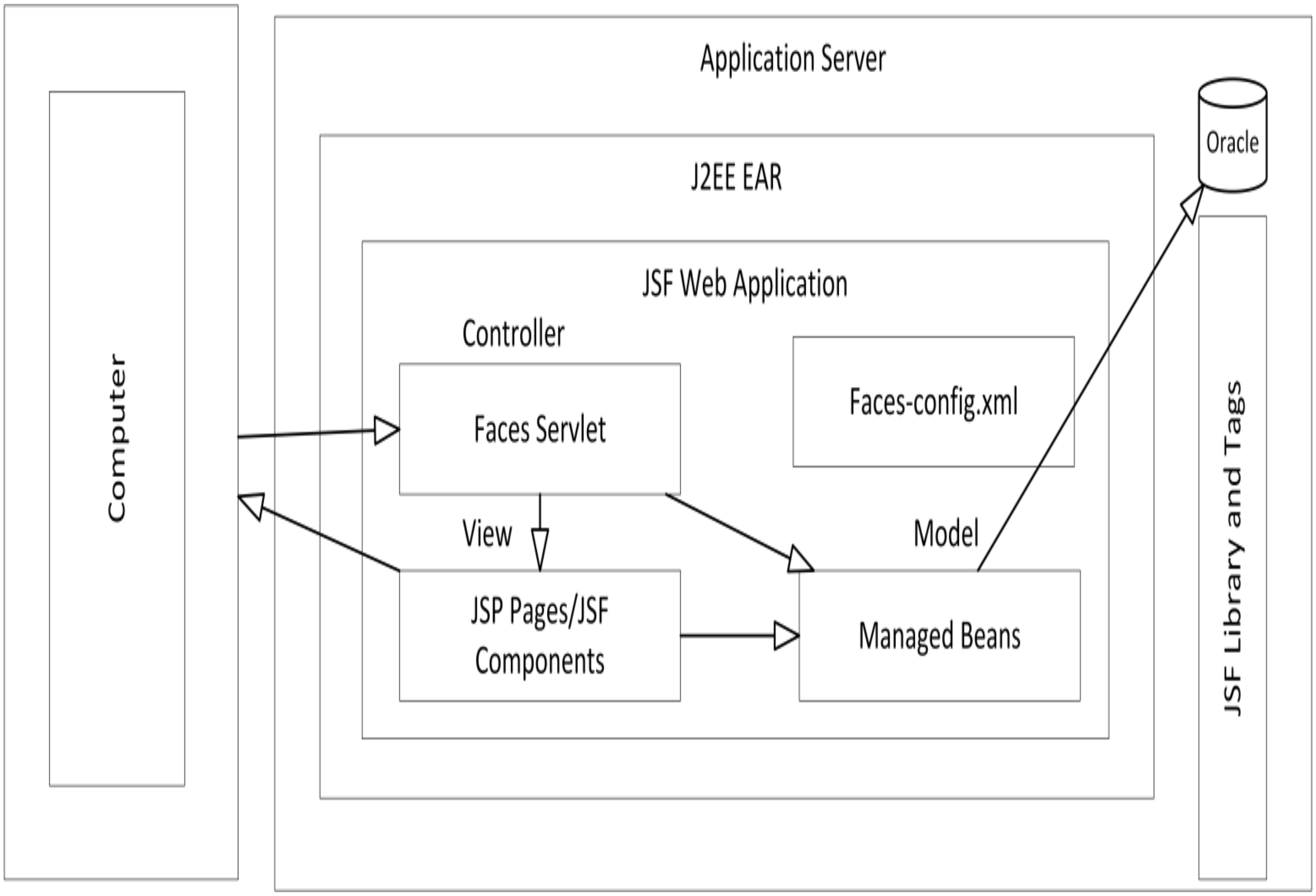
5.3 Schedule

As a team we also chose not to include a scheduling feature in our software. There are several existing software products that help you plan for the future but few that let you track things done in the past. We also recognized that we would not have enough time to properly implement a scheduling feature and decided to exclude it from our plan.

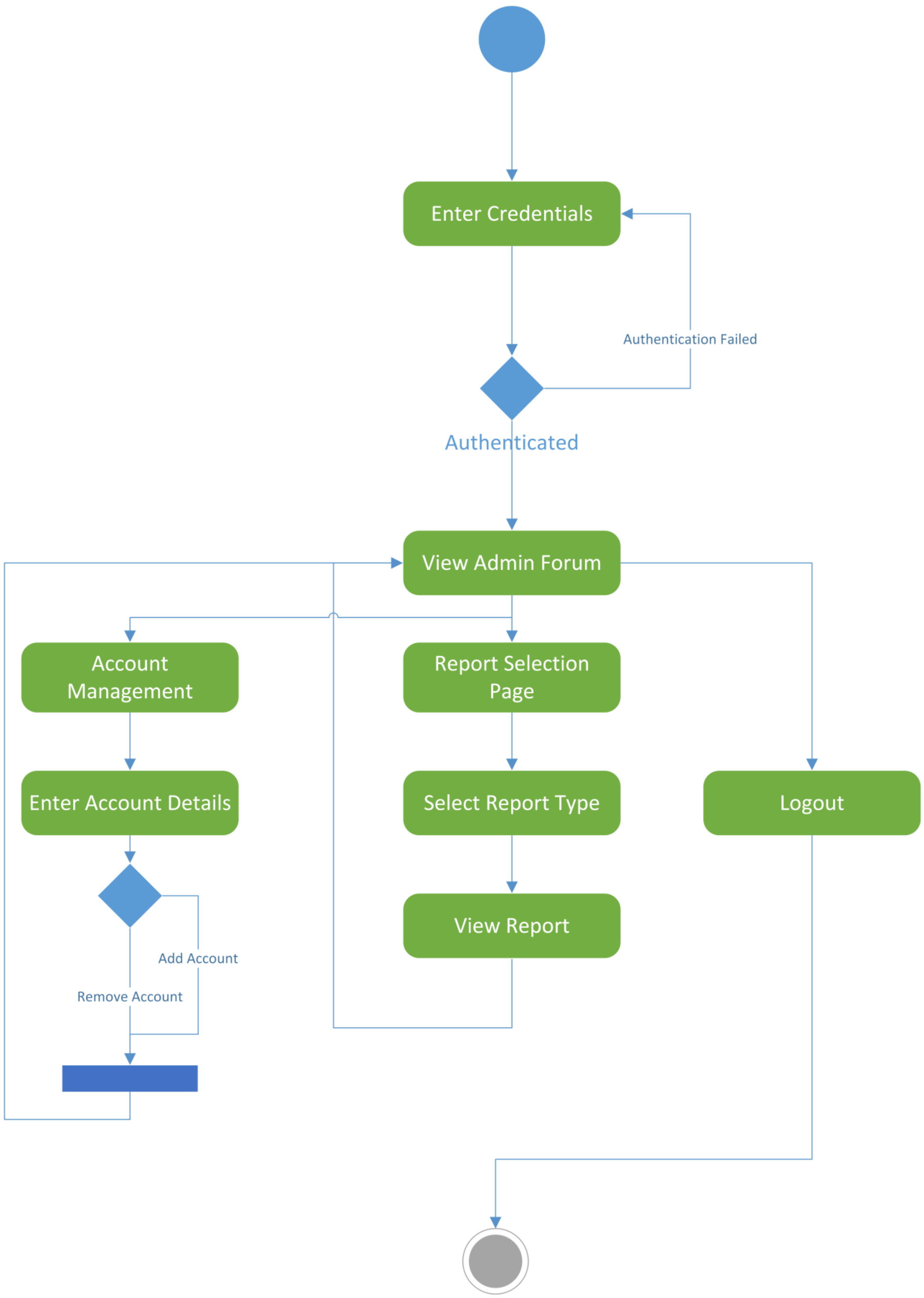
6. Appendices

6.1 Diagrams

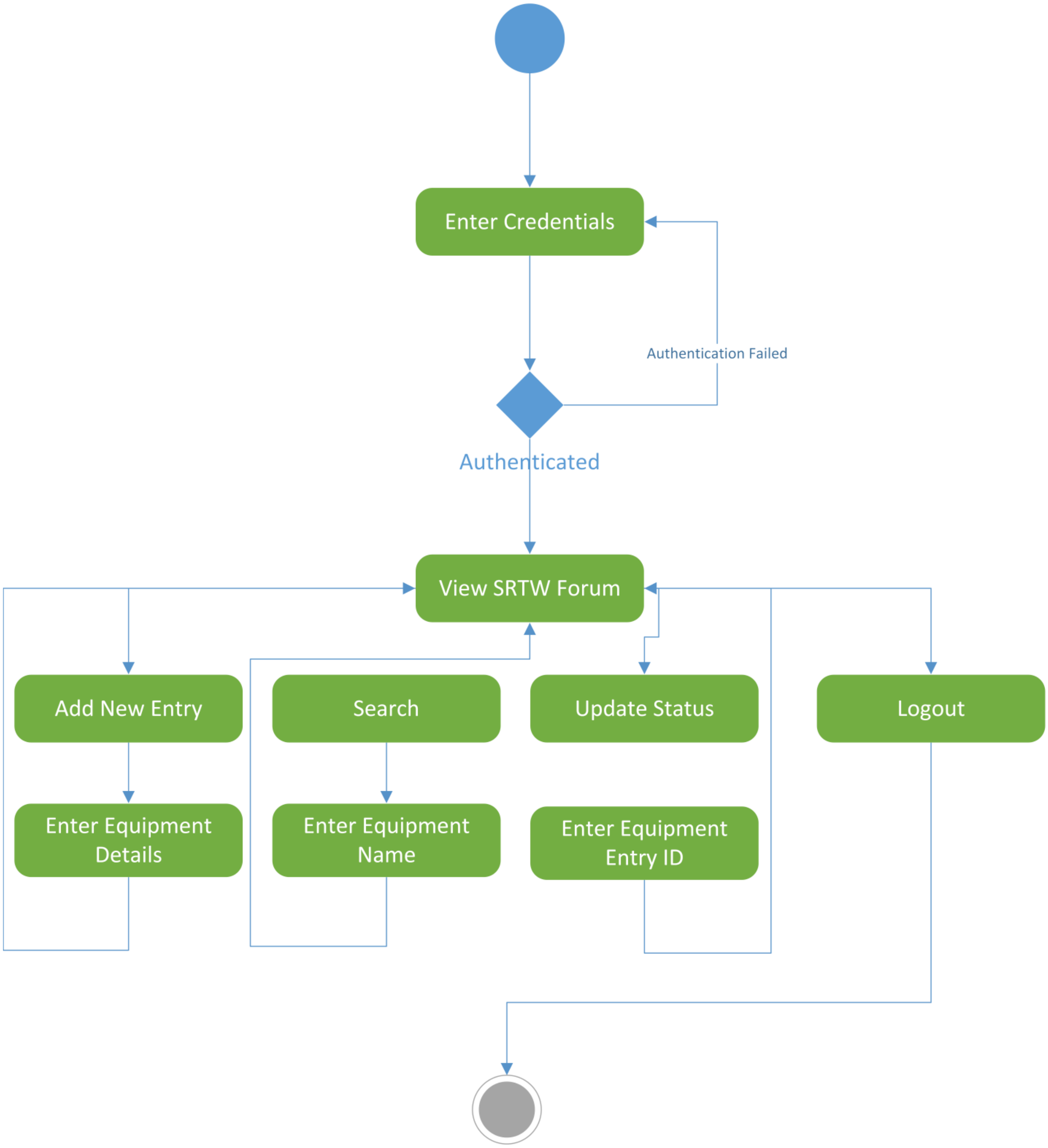
6.1.1 System Structure Diagram



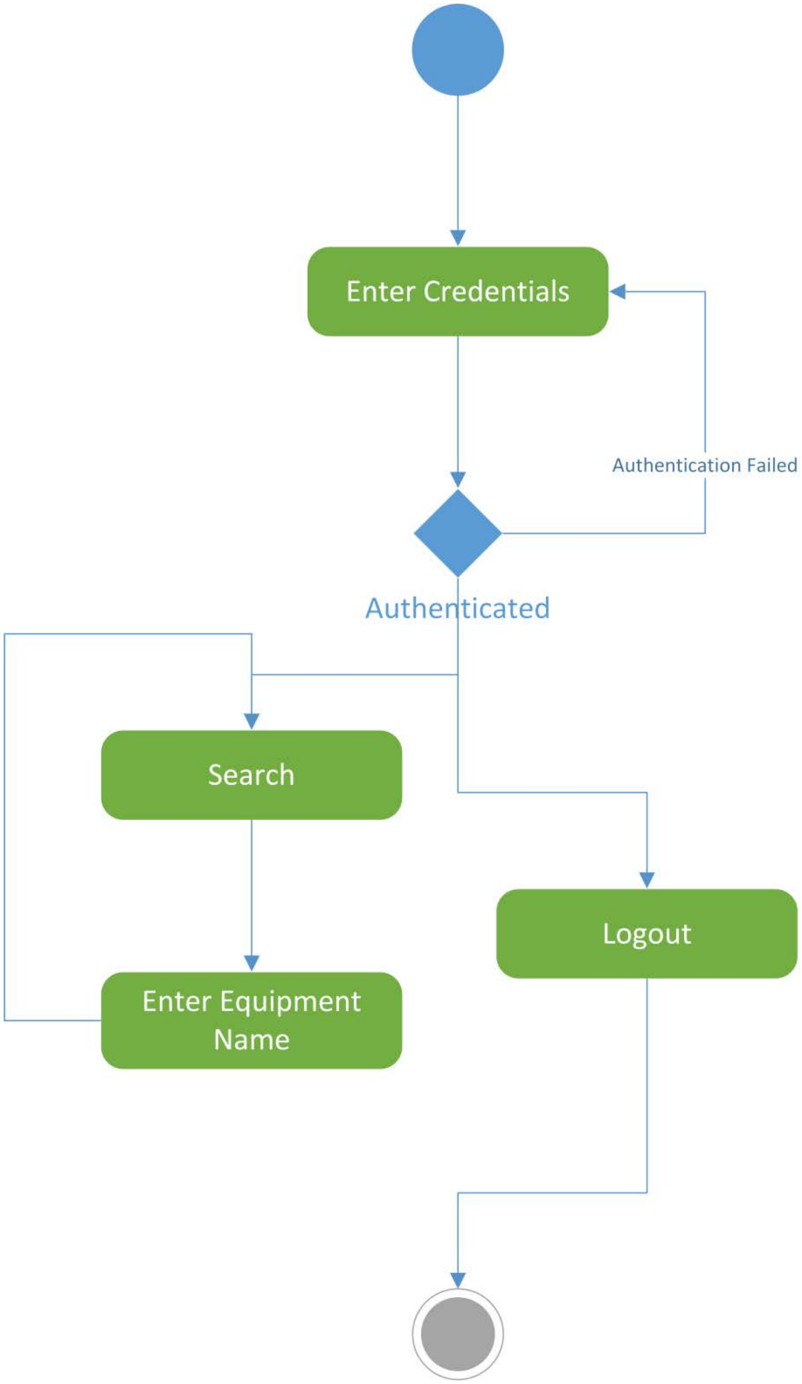
6.1.2 Admin Activity Diagram



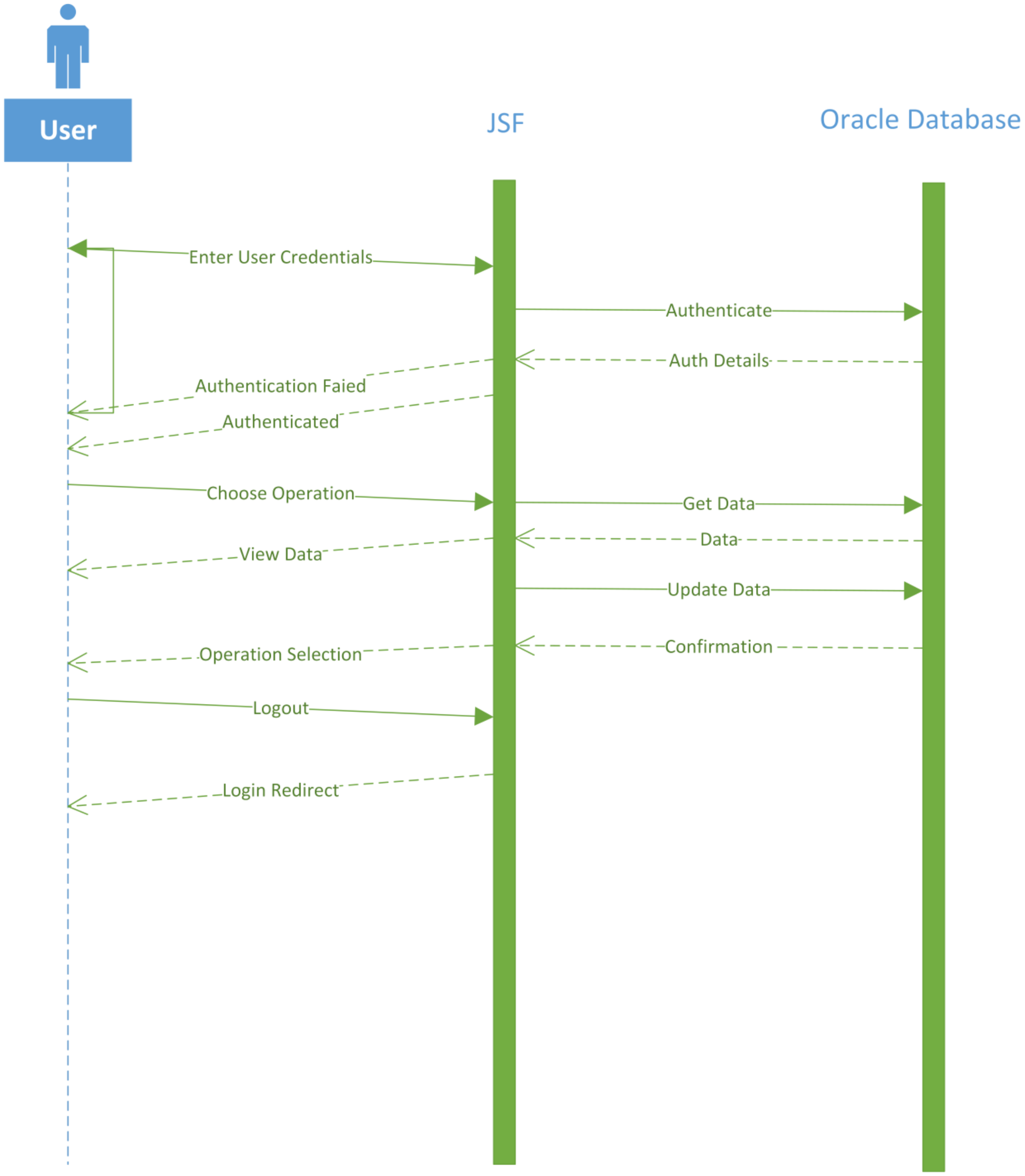
6.1.3 SRTW User Activity Diagram



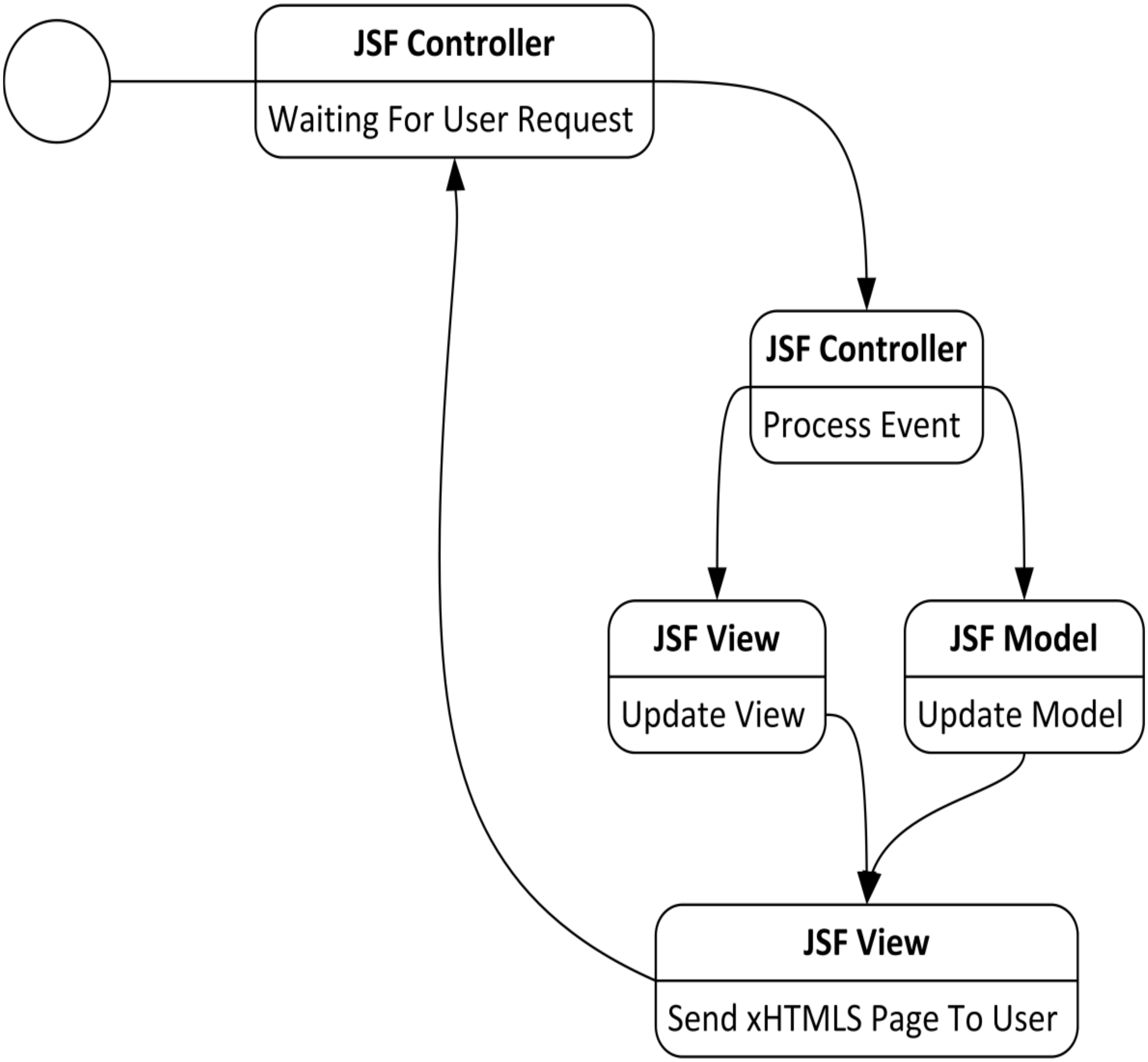
6.1.4 Unit User Activity Diagram



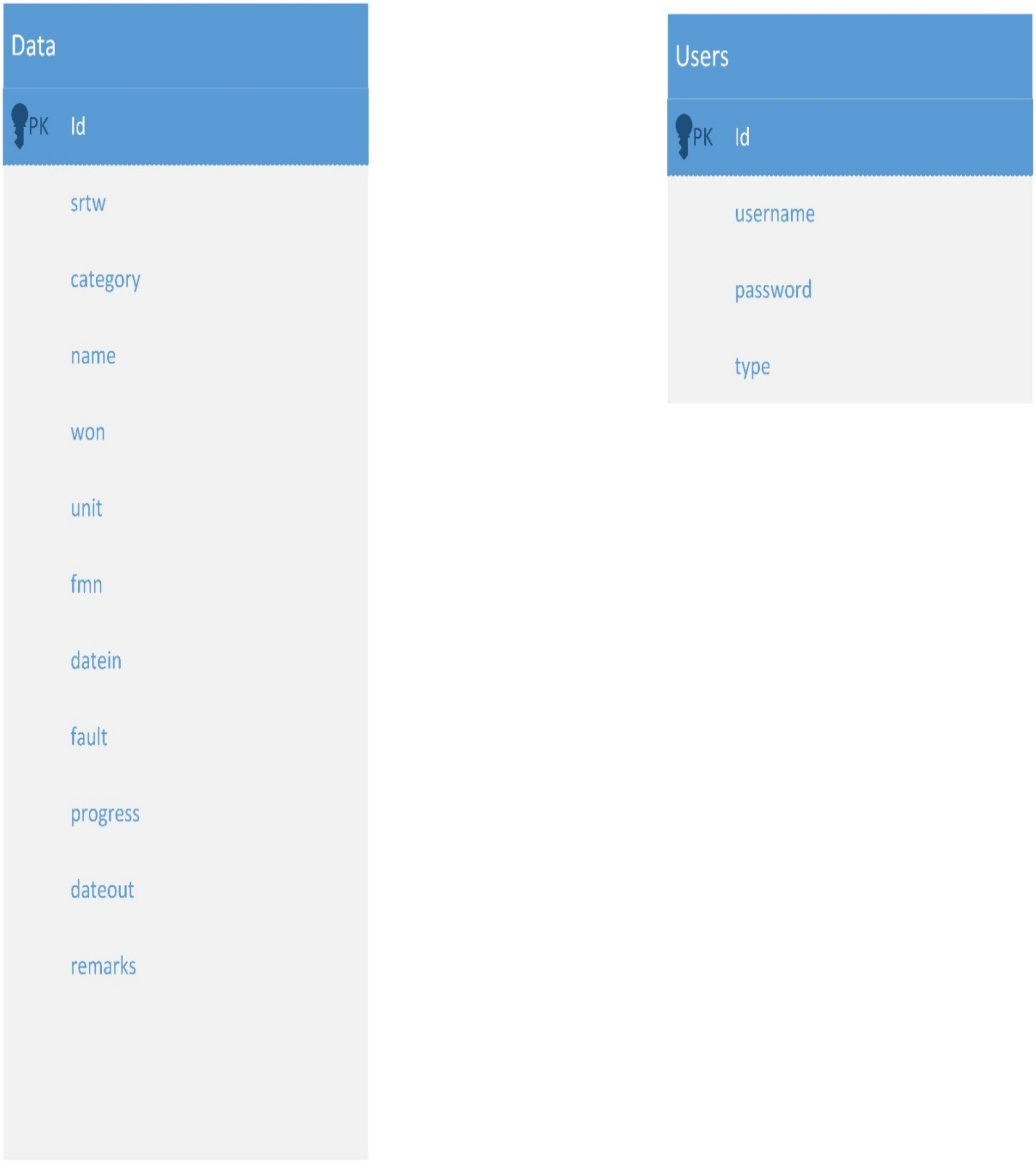
6.1.5 Sequence Diagram



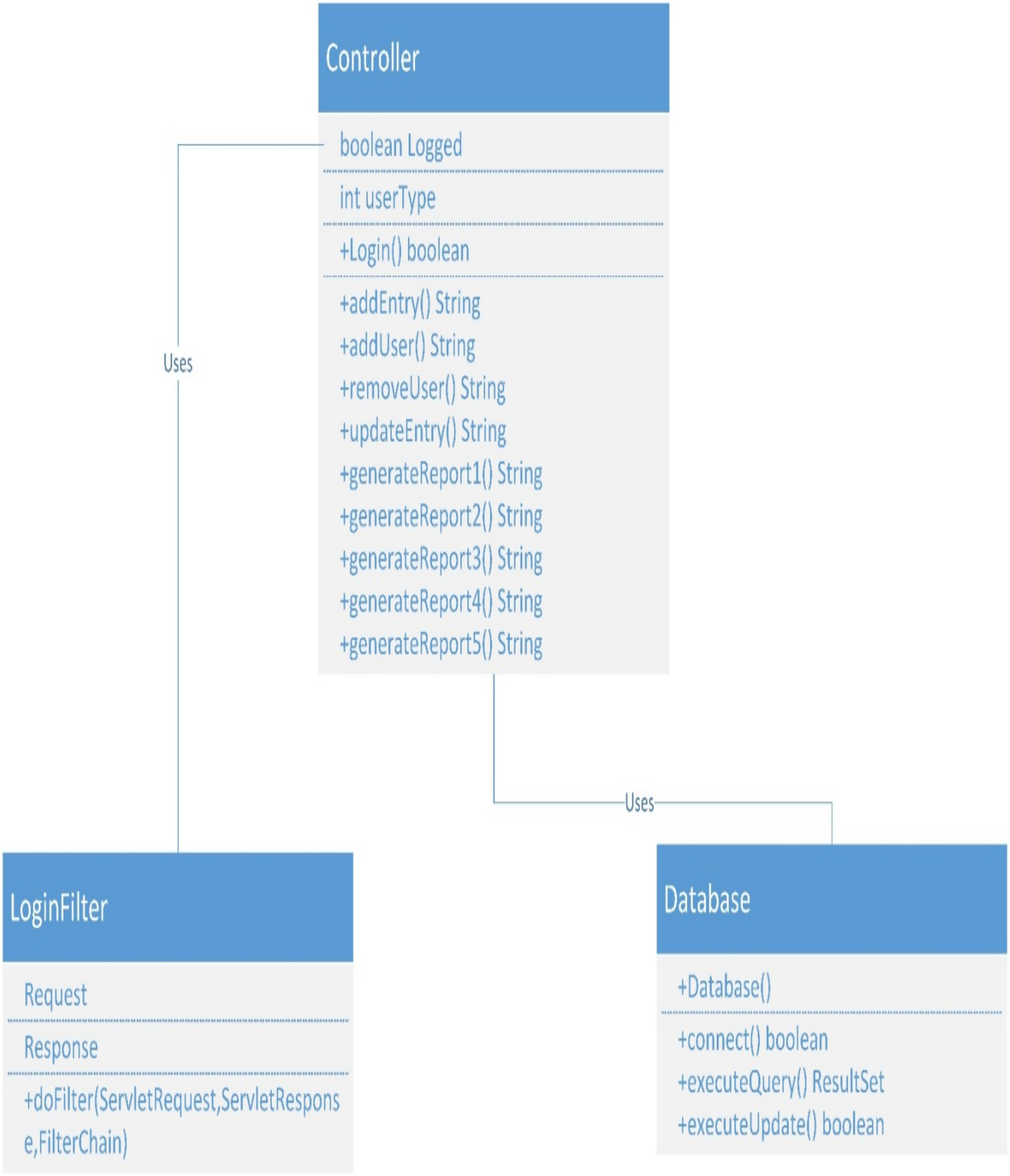
6.1.6 Server State Machine Diagram



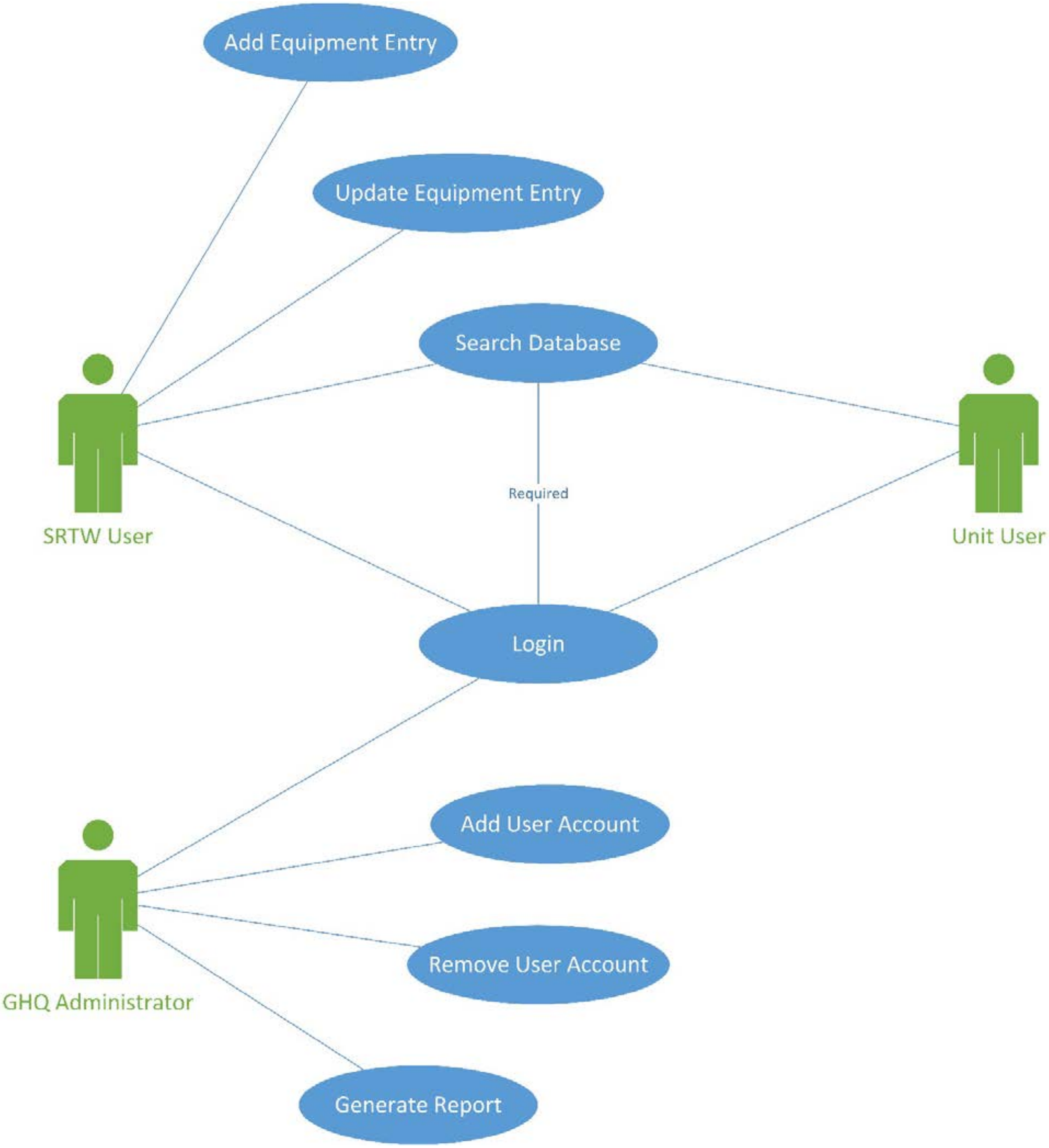
6.1.7 ER Diagram



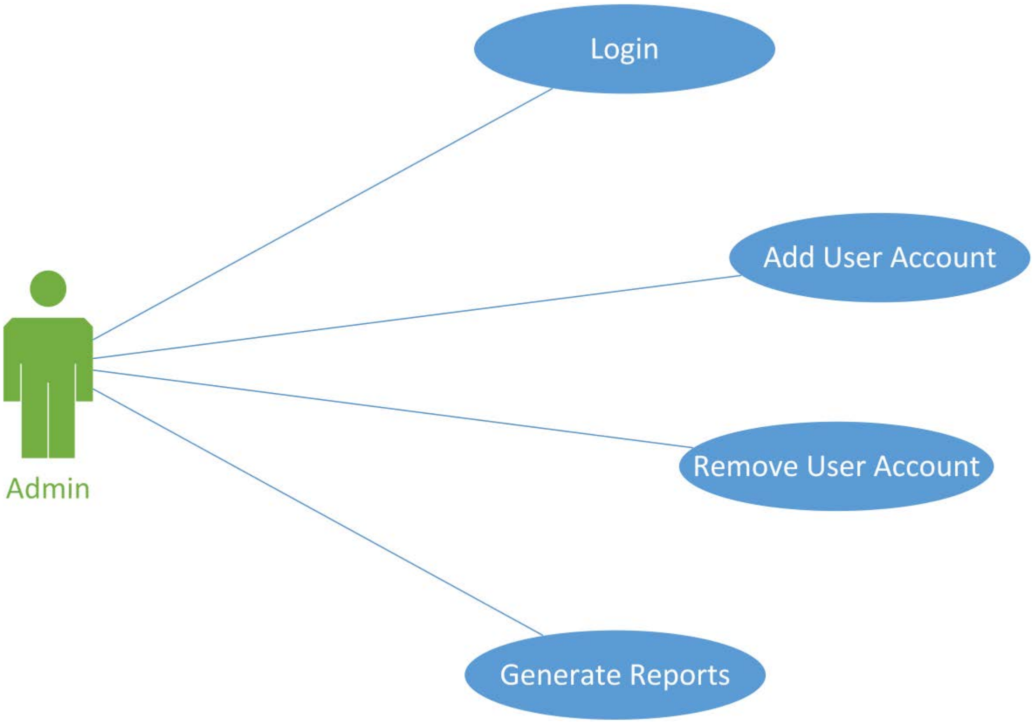
6.1.8 Class Diagram



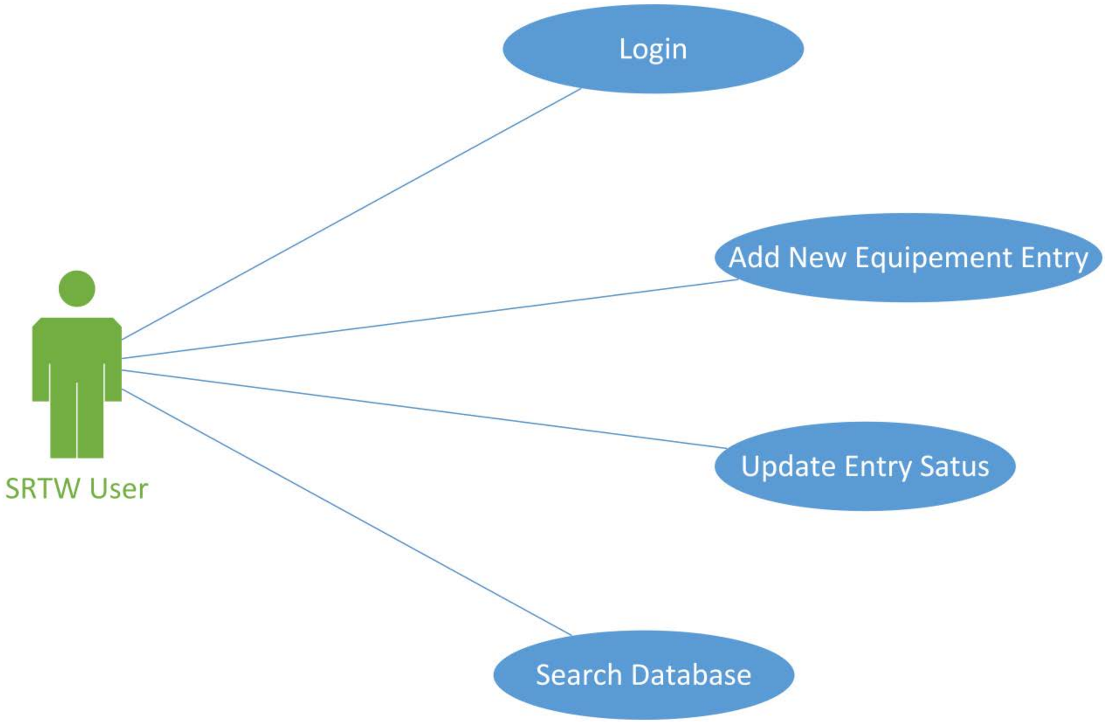
6.2 Use Case Diagram



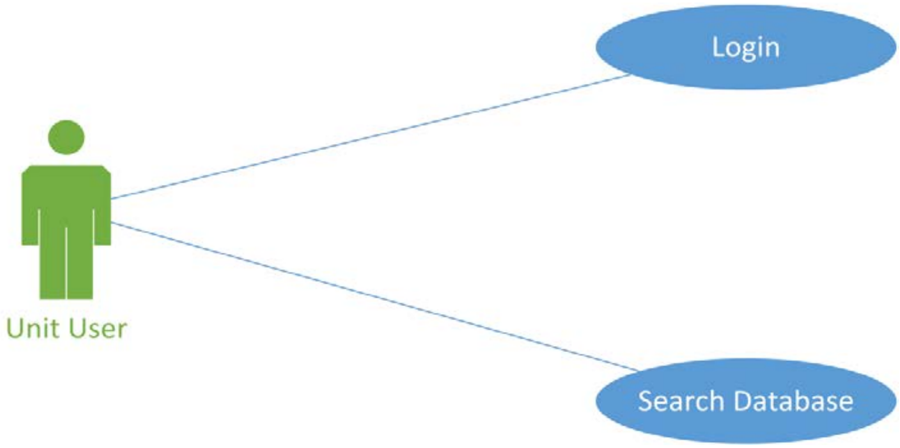
6.2.1 GHQ User (Admin) Use case Diagram



6.2.2 SRTW User Use Case Diagram



6.2.3 Unit User Use Case Diagram



6.3 Use Cases

Use Case ID:	01
Use Case Name:	Login
Actor:	Unit user, SRTW user, GHQ user
Description:	This use case scenario enables the user to login in to its personal forum screen
Pre-conditions:	The user must have valid username and password
Post-conditions:	The user will move to his forum screen
Priority:	This use case holds high priority
Frequency of Use:	Generally, this test will be performed once when the application is turned on but it can be performed as many times as the user wishes as per his choice.
Normal Course of Events:	1. Turn on the application 2. Enter the user type 3. Enter the password 4. Login to Forum screen

Alternative Courses:	1. The user chooses to exit the application
Exceptions:	-
Includes:	-
Special Requirements:	
	-

Use Case ID:	**02**
Use Case Name:	Search Database
Actor:	Unit user, SRTW user, GHQ user
Description:	This use case scenario enables the user to search the database as per his authorized rights
Pre-conditions:	The user must have logged in to the application
Post-conditions:	The user will search the database as per his authorized rights.
Priority:	This use case holds high priority
Frequency of Use:	Generally, this test will be performed once when the user is logged in but it can be performed as many times as the user wishes as per his choice.
Normal Course of Events:	1. Turn on the application 2. Login 3. Search Database 4. View results
Alternative Courses:	1. The user chooses to exit the application 2. User logs out
Exceptions:	1. User enters incorrect equipment
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**03**
Use Case Name:	Add Equipment entry
Actor:	SRTW user
Description:	This use case scenario enables the user to enter new equipment into the database.
Pre-conditions:	The user must have logged in to the application
Post-conditions:	The user will be able to add new equipment details into the database
Priority:	This use case holds high priority
Frequency of Use:	Generally, this task will be performed once when the user is logged in but it can be performed as many times as the user wishes as per the new submitted equipment
Normal Course of Events:	1. Turn on the application 2. Login 3. Select Add new Equipment button 4. Enter new Equipment details
Alternative Courses:	1. The user chooses not to enter the new equipment 2. User logs out
Exceptions:	1. User enters incorrect equipment details
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**04**
Use Case Name:	Update Equipment entry
Actor:	SRTW user
Description:	This use case scenario enables the user to update status of equipment in the database.
Pre-conditions:	The user must have logged in to the application
Post-conditions:	The user will be able to update equipment details in the database
Priority:	This use case holds medium priority
Frequency of Use:	Generally, this task will be performed once when the user is logged in and the submitted equipment has been repaired.
Normal Course of Events:	1. Turn on the application 2. Login 3. Select Update Equipment button 4. Update Equipment Status
Alternative Courses:	1. The user chooses not to update equipment status 2. User logs out
Exceptions:	1. User enters incorrect equipment details
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**05**
Use Case Name:	Add User Account
Actor:	GHQ user
Description:	This use case scenario enables the user to Add new user account into the database.
Pre-conditions:	The user must have logged in to the application The user must be at the managing user account option
Post-conditions:	The user will be able to add new account details into the database
Priority:	This use case holds high priority
Frequency of Use:	Generally, this task will be performed once when the user is logged in and a new user request has been received.
Normal Course of Events:	1. Turn on the application 2. Login 3. Select Managing user accounts 4. Select Add new user account 4. Enter new user account details
Alternative Courses:	1. The user chooses not to add new user account 2. User logs out
Exceptions:	1. User enters incorrect user account details
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**06**
Use Case Name:	Remove User Account
Actor:	GHQ user
Description:	This use case scenario enables the user to remove a user account from the database.
Pre-conditions:	The user must have logged in to the application The user must be at the managing user account option
Post-conditions:	The user will be able to remove user account from the database
Priority:	This use case holds medium priority
Frequency of Use:	Generally, this task will be performed once when the user is logged in and a request to remove a particular account has been made
Normal Course of Events:	1. Turn on the application 2. Login 3. Select Managing user accounts 4. Select Remove user account 4. Remove any user account
Alternative Courses:	1. The user chooses not to remove a particular user account 2. User logs out
Exceptions:	1. User enters incorrect accounts details
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**07**
Use Case Name:	Generate reports
Actor:	GHQ user
Description:	This use case scenario enables the user to generate a particular type of report regarding the submitted or repaired equipment
Pre-conditions:	The user must have logged in to the application The user must be at the generate reports option
Post-conditions:	The user will be able to generate any particular type of report
Priority:	This use case holds high priority
Frequency of Use:	Generally, this task will be performed once when the user is logged in and any type of report is required by higher authorities
Normal Course of Events:	1. Turn on the application 2. Login 3. Select Generate reports option 4. Select a particular type of report 5. View or print the report
Alternative Courses:	1. The user chooses not to generate a report 2. User logs out
Exceptions:	-
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

ANALYSIS & EVALUATION

1. Overview

The purpose of this chapter is to give the user a clear physical demonstration of the functionalities of SRTWs management system, a web based software application that would be built for digitalizing the inventory of Signal Repair Technical Workshops of Pakistan Army.

This chapter is aimed to eliminate ambiguities and misunderstandings that may exist. It will explain all functions that the software should perform.

2. Test Cases

Use Case ID:	**01**
Use Case Name:	Login
Actor:	Unit user, SRTW user, GHQ user
Description:	This use case scenario enables the user to login in to its personal forum screen
Pre-conditions:	The user must have valid username and password
Post-conditions:	The user will move to his forum screen according to the type of use
Priority:	This use case holds high priority
Expected Result	When the user enters the username and password, it checks whether the username and password provided are correct. If they are correct then the user can switch to its forum screen If the username and password provided are incorrect then again login screen appears
Actual Result	The user was able to login to its required status when the correct username and password provided. Login screen appeared again when incorrect username or password was entered
Alternative Courses:	1. The user chooses to exit the application

Exceptions:	-
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**02**
Use Case Name:	Search Database
Actor:	Unit user, SRTW user, GHQ user
Description:	This use case scenario enables the user to search the database as per his authorized rights
Pre-conditions:	The user must have logged in to the application
Post-conditions:	The user will search the database as per his authorized rights.
Priority:	This use case holds high priority
Expected Result:	The unit user gets the table of its submitted eqpt. The Srtw user gets the table of any eqpt by entering the name or category of eqpt The GHQ user gets the table of all eqpt by generating the monthly reports
Actual Result:	All the users got the desired results When incorrect name was entered no result was shown
Alternative Courses:	1. The user chooses to exit the application 2. User logs out
Exceptions:	1. User enters incorrect equipment
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**03**
Use Case Name:	Add Equipment entry
Actor:	SRTW user
Description:	This use case scenario enables the user to enter new equipment into the database.
Pre-conditions:	The user must have logged in as SRTW user to the application
Post-conditions:	The user will be able to add new equipment details into the database
Priority:	This use case holds high priority
Expected Result:	New eqpt detailed successfully added to the database
Actual Result:	New eqt details were successfully added to the database If any field is left blank or incorrect format used then the eqpt entry form will appear again and details will not be added to the database.
Alternative Courses:	1. The user chooses not to enter the new equipment 2. User logs out
Exceptions:	1. User enters incorrect equipment details
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**04**
Use Case Name:	Update Equipment entry
Actor:	SRTW user
Description:	This use case scenario enables the user to update status of equipment in the database.
Pre-conditions:	The user must have logged in to the application as SRTW user
Post-conditions:	The user will be able to update equipment details in the database
Priority:	This use case holds medium priority
Expected Result:	Eqpt status successfully updated
Actual Result:	Eqpt status successfully updated When incorrect unique eqpt ID was entered, status was not updated
Alternative Courses:	1. The user chooses not to update equipment status 2. User logs out
Exceptions:	1. User enters incorrect equipment details
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**05**
Use Case Name:	Add User Account
Actor:	GHQ user
Description:	This use case scenario enables the user to Add new user account into the database.
Pre-conditions:	The user must have logged in to the application as an ADMIN user
Post-conditions:	The user will be able to add new account details into the database
Priority:	This use case holds high priority
Expected Result:	New user successfully added to the database i.e. both unit user and SRTW user.
Actual Result:	New user successfully added to the database i.e. both unit user and SRTW user. If a username or password field was left empty then the add new user page appears again
Alternative Courses:	1. The user chooses not to add new user account 2. User logs out
Exceptions:	1. User enters incorrect user account details
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**06**
Use Case Name:	Remove User Account
Actor:	GHQ user
Description:	This use case scenario enables the user to remove a user account from the database.
Pre-conditions:	The user must have logged in to the application as an ADMIN user
Post-conditions:	The user will be able to remove user account from the database
Priority:	This use case holds medium priority
Expected Result:	The entered account successfully removed from database
Actual Result:	The entered account was successfully deleted from the data base If the account name field is left empty then same screen appears again.
Alternative Courses:	1. The user chooses not to remove a particular user account 2. User logs out
Exceptions:	1. User enters incorrect accounts details
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	**07**
Use Case Name:	Generate reports
Actor:	GHQ user
Description:	This use case scenario enables the user to generate a particular type of report regarding the submitted or repaired equipment
Pre-conditions:	The user must have logged in to the application as an ADMIN user
Post-conditions:	The user will be able to generate any particular type of report
Priority:	This use case holds high priority
Expected Result:	User successfully generates monthly, highest category monthly,highest unit monthly,all under repaired eqpt and specific eqpt reports.
Actual Result:	All reports were generated successfully If wrong month is entered in which no ept is submitted then no report is generated
Alternative Courses:	1. The user chooses not to generate a report 2. User logs out
Exceptions:	-
Includes:	-
Special Requirements:	No special requirements.
Assumptions:	-
Notes and Issues:	-

FUTURE WORK

1. Overview

The purpose of this chapter is to elaborate what will be the future aspects of this project, what are our intentions in conjunction with army's requirements and feasibility.

2. New Functionalities

Additional functionalities may be used in this management system like:-

- Ability to generate work order of a specific equipment from the unit user
- Ability to make a query or suggestion by the unit user to the SRTW user
- Ability to accept a suggestion by SRTW user from unit user
- Ability to make a suggestion by SRTW user to GHQ user
- Ability to accept a suggestion by GHQ user from SRTW user
- Add new formats of reports by GHQ user

CONCLUSION

The increased Role of SRTWs in view of additional responsibility entrusted to Signals for field repair of all radio communication equipment necessitated the need for an application to digitalize the record of all of formations equipment along with repair profile at each SRTW which we have developed. SMS can be effectively used throughout Pakistan Army for this purpose. SMS will also facilitate the requirement for analysis of different types of signal equipment at Signals Directorate for showing fused picture of all communication equipment date and repair activities undertaken at each SRTW.

Moreover, after deploying it on OAS it may be further developed for more efficient reporting and analysis.

BIBLIOGRAPHY

1. Visit to SRTW Rawalpindi

A visit was made to SRTW Rawalpindi in the month of October 2016 with a view to study and understand the existing system and determine the flaws therein. In that context, the interview with 2ic and concerned NCO was followed by the overview and visit of the facility. It was found that most of the sections weren’t having any online reporting system. So we decided to bring all the sections on the same platform using our SMS.

2. References

- http://searchdatabase.techtarget.coment
- Signal inventory management system (book)
- https://www.youtube.com/playlist?list=PLEAQNNR8IIB588DQvb2wbKFQh2DviPApl