Addis Ababa University

College of Natural Sciences

ITIL V3 based Service Desk System with Incident Classification and Technician Assignment

Melikte Paulos

A Project Submitted to the Department of Computer Science in Partial Fulfillment for the Degree of Master of Science in Computer Science

Addis Ababa, Ethiopia
September 2016

Addis Ababa University

College of Natural Sciences

ITIL V3 based Service Desk System with Incident Classification and Technician Assignment

Melikte Paulos

Advisor: Mesfin Kifle (PhD)

Name and Signature of members of the Examining Board

	Name	Signature	Date
Advisor:			
Examiner:			
Examiner:			

Acknowledgment

First and foremost I would like to thank God for giving me strength and hope during this whole process.

I would also like to thank my thesis advisor Dr. Mesfin Kifle, of the Department of Computer Science, for his excellent cooperation and great advices. In addition, I would like to extend my gratitude towards Nesredin Suleiman for his unceasing guidance and support.

I am grateful to my wonderful parents whose endless love and support made all this possible. I owe you everything.

I am also grateful to Miretab Teklay for his wise council as well as his remarkable willingness to help. Thank you for connecting me with the people I needed, making it easier to find resources and giving me guidance whenever I needed it.

I owe my deepest gratitude to Nathnael Gossaye for providing me with unfailing support and continuous encouragement throughout the process of researching and writing this thesis. This accomplishment would not have been possible without you.

Last but not least, many thanks to all my friends at the university and outside - who have made this journey a bit easier.

Abstract

Ethiopian Airlines is a company with abundant IT assets and services. It is the responsibility of the IT support department to resolve any issue related to IT services as soon as possible and to greatly minimize the impact it has on the business. Currently, the department is facing issues such as incidents not being assigned in time or assigned to the wrong teams, incidents not being correctly logged, incidents having no association with assets etc. In addition to this, other systems in the company follow international standards while the support team barely uses the in-house application developed to assist it.

The aim of this project is to design, develop and implement a system to improve the service management process of the IT support department. After collecting requirement and data from the company, several literature was reviewed to find an appropriate solution. Several data mining techniques were also examined to see which algorithms gave the best result. This has also helped in better understanding the current system and identifying important features that need to be developed.

We proposed a service desk system that complies with ITIL standards (IT Infrastructure Library), the most widely adopted approach for IT service management, with the added capability to automatically assign technicians just from the incident description.

The service desk system is implemented with selected processes that fit the current workflow of Ethiopian Airlines. In addition to that, data mining techniques are applied to the data collected from their current system in order to classify incidents (based on their description) to support teams. This will enable the system to automatically assign technicians without delay. The system was tested at the company and was found to accurately capture their workflow as well as introduce helpful ways to improve the service management process.

Keywords: ITIL, ITSM, Service Desk, Data mining, Text Classification

Table of Contents

List of Figures

List of Tables

Acronyms and Abbreviations

ACE: Achieving Competitive Excellence

CAB: Change Advisory Board

COBIT Control Objectives for Information and Related Technologies

ET: Ethiopian Airlines

IRT: Initial Response Time

ITIL: IT Infrastructure Library

ITSM: IT Service Management

MOF Microsoft Operations Framework

MPT: Maximum Processing Time

RFC: Request for Change

SLM Service Level Management

SLA: Service Level Agreement

SLP Service Level Performance

IT Information Technology

Chapter 1 – Introduction

1.1 Overview

It's undeniable that technology is an essential part of our day-to-day life. From the alarm that wakes us up to the different tools we use during the day, we are becoming more and more dependent on technology. Businesses – large or small – are benefiting greatly from various technological gadgets and greatly rely on them to function properly.

"Organizations are increasingly becoming aware of the essential role of Information Technology (IT) within their organizations, and are under pressure to account for costs, and to manage risks associated with the ever increasing vulnerability of their IT infrastructure" [1] . **IT Service Management** (ITSM) is therefore necessary to ensure the proper functioning of IT assets and services within an organization. "ITSM is concerned with the planning, designing, delivering, operating and controlling of IT services offered to customers" [2].

Large organizations like the Ethiopian Airlines (ET) have their own IT Support departments that play a very important role in helping the company serve its customers better. The Department is in charge of ensuring that the different systems (Software, Hardware and Network) of the company are running smoothly. The aim of the department is to provide stable, robust and reliable services to the users of the airline's systems and help these users to efficiently serve their customers. To achieve this, ET needs an ITSM system that helps in managing the tasks of the IT support staff allowing them to perform their duties proficiently. Such a system is called a **Service Desk**.

A **Service Desk** is a function within the discipline of IT service management intended to provide a single point of contact to meet the needs of both customers and IT employees [3]. It is where IT problems are fixed and preventive measures are taken. It helps in handling issues in a centralized and structured way.

The ET staff has done research on different frameworks for ITSM and arrived to the conclusion that the **IT Infrastructure Library** (ITIL) is the best one that fulfills their requirements. "ITIL is a collection of best practices" [4] and "the most widely adopted approach" [5] for ITSM. One of the main goals of ITIL is to transform IT departments into service oriented organizations. Service management as defined by ITIL includes: Service Strategy, Service Design, Service Transition, Service Operation, and Continual Service Improvement lifecycles which include several processes within them [6].

In addition to deploying a system that follows an international standard, the IT department can also benefit from having an effective Resource Management System that helps in managing the technical staff working on resolving issues. This will be implemented in the form of assigning technicians automatically to incidents, reducing the time incidents stay unassigned and/or get assigned to the wrong group.

For the system to automatically assign incidents to technicians, it was necessary to research how it's currently being done. It was seen that different support teams were assigned to incidents that seemed to be in the same incident category. Support teams were being assigned by the person at the central service desk after looking at their description. For the system to be able to do the automatically assign technicians, data mining techniques were applied to the data to help it learn from over 9,000 descriptions. This technique will enable the system to learn from new incidents and improve. It was selected after examining several methods and was found to be feasible and efficient.

1.2 Statement of the Problem

The problems with the current ITSM system of ET can be categorized into three. The first one is that there is no centralized way of handling incidents and/or service requests.

Most incidents regarding hardware, networking and other IT assets that are reported will not be recorded in the system; the maintenance staff will be informed of the issue through various means (phone, email etc.) and they handle it directly. This makes it difficult for tracking, reporting and learning from previous mistakes. Issues related with applications are resolved independently by each IT expert working on the respective application software.

Secondly, the system does not follow any international standard. Following an international standard ensures lower cost through tried and tested processes, quality systems, increased efficiency and minimized waste. It could also solve issues regarding missing functionalities such as controlling the time it takes to solve incidents, feedback collection from customers and integration between incident management and asset management.

Third problem involves the assignment of technicians to incidents. Technicians' workload and performance is not considered when assigning incidents and some incidents go unassigned for a while because they weren't seen in time. Incidents also get assigned to the wrong team or technician which unnecessarily increases the time it takes to resolve them.

The problem to be addressed by this project would be - how to solve the issues of ITSM in ET by following an international standard (ITIL) and utilize the data collected by the system to automatically assign incidents to technicians.

1.3 Objective

General objective

The general objective of this project is to design and develop an ITIL based service desk with automatic assigning of technicians in order to improve the current ITSM of the company.

Specific objective

The specific objectives are:

- Review literature to have a better understanding of standards, tools and techniques needed to implement the proposed system.
- Study the current system and identify user requirements
- Investigate tools for automatic assigning of resources
- Design and develop a system that follows the ITIL standard while fulfilling the requirements of the company
- Implement and test the proposed system.

1.4 Scope and Limitation

This project will mainly focus on developing an ITIL based service desk and the automatic assignment of incidents to technicians. The service desk will include Incident Management, Problem Management, Asset Management, Change Management and Service-level Management. The automatic assigning of support teams to incidents will be done using text classification.

1.5 Methods

Several methods will be implemented during the different lifecycles of this project. Each are described below.

Requirement Collection – Onsite visit will be made to observe how current incidents and service requests are being managed, interviews will be conducted with staff involved in IT service operation, and documents will be reviewed for a complete view on the development of the current system.

Literature review – Literature will be reviewed on the existing problems as well as other works related to the proposed solutions. Literature review is essential to have a better understanding, and to know the more efficient solution to the problems.

Creating Text Classification Model – Text classification processes will be applied to the data collected from the company. A classification model will then be created using the selected algorithm.

Design – a web based application will be designed using the following tools:

- ASP MVC along with C# will be used for developing the system which can be deployed on the company's server and accessed through the network.
- MSSQL will be used for the database.

Testing – Usability testing will be done by interviewing users while they're interacting with the system.

1.6 Significance of the study

Creating a single point of contact for dealing with the needs of users and the various IT support departments will improve the communication between them. This will help in avoiding misunderstandings and unwanted delays that occur during ITSM activities and effectively maximize the up-time of the company's IT assets and services. This in turn will help in increasing the productivity of the different departments of the organization which greatly benefits the company in reducing its cost in terms of time and money.

1.7 Organization of the Document

The remainder of this document includes seven chapters and is organized as follows. Chapter 2 and Chapter 3 will put forward different topics found during literature review and various works related to the project to give a clearer understanding of the project. Chapter 4 discusses the current and proposed systems, shows both functional and non-functional requirements that are identified and displays several models used to depict them. Chapter 5 discusses the system design and has diagrams that illustrate the chosen architecture. Chapter 6 will include the implementation of the system and testing done on the different components of the system. Finally, Chapter 7 will have conclusions made from the development process of the project and provide recommendations for future work.

Chapter 2 - Literature Review

This Chapter discusses literature reviews conducted regarding the topics that are useful for implementing the system. It explains the meaning of ITSM, examines the stages and processes of ITIL and describes the meaning and objective of a service desk. In addition to that, it shows reviews on topics concerning classifiers and decision trees. Finally it explains about MVC5 and ASP.Net.

2.1 The concept of ITSM

ITSM includes the planning, delivering and evaluating of services that are provided by IT organizations to their customers. It enables an organization to efficiently handle incidents, change and/or services requests or any new requirements. "It is the planned and controlled utilization of IT assets (including software, infrastructure and tools), people and processes to support operational needs of businesses as efficiently as possible" [7].

2.2 Frameworks of ITSM

2.2.1 COBIT

COBIT is a framework created by an organization called ISACA. The latest version called COBIT 5 consists of 5 main principles and 7 enablers. These components provide generic processes and define best practices for IT governance [8]. COBIT has a process model that divides ITSM into four domains: Plan and Organize, Acquire and Implement, Deliver and Support and Monitor and Evaluate.

2.2.2 MOF

MOF is Microsoft's approach to ITSM. Its documentation is licensed under Creative Commons and can be used by anyone. MOF groups all IT activities and processes into Service Management Functions (SMFs). And each SMF defines goals, outcomes and operational guidance for each phase it is linked to in the ITSM lifecycle [9].

2.2.3 ITIL

“ITIL provides a set of best practice processes for IT Service Management, promoting a quality approach to achieving business effectiveness and efficiency in the use of information systems” [7].

The ITIL V3 framework comprises of five stages: ITIL Service Strategy, ITIL Service Design, ITIL Service Transition, ITIL Service Operation and ITIL Continual Service Improvement.

These five stages contain the entire ITIL Service Lifecycle and are described as follows:

1. **Service Strategy:** The primary purpose of this stage is to identify the requirements of various stakeholders and documenting those needs. It focuses on “how to design, develop, and implement service management not only as an organizational capability but also as a strategic asset” [5].

   Processes under this stage: Strategy management for IT Services, Service portfolio management, Financial management for IT services, Demand management and Business relationship management.

2. **Service Design:** During this stage, services that are aligned with the business goals that are identified during the previous phase are designed and developed. “The processes that govern the management and delivery of services are also developed during this phase” [10].

   “Service design starts with new or changed business requirements and ends with the development of a service solution designed to meet the needs of the business [10]”.

   Processes under this stage: Design coordination, Service catalogue management, Service-level management, Availability management, Capacity management, IT service continuity management, Security management and Supplier management.

3. **Service Transition:** This phase “allows the service provider to align the new or changed service with the customer’s business requirements and operations as well as ensure that customers and users can use the new or changed service in a way that minimizes

cost to the business operations" [6]. This is important to make sure that new systems integrate well with existing systems, employees and processes.

Processes under this stage: Transition planning and support, Change management, Service asset and configuration management, Release and deployment management, Service validation and testing, Change evaluation and Knowledge management.

4. **Service Operation:** This stage of the service lifecycle is concerned with making sure activities regarding delivery and support of services keep running smoothly and efficiently. "Strategic objectives are ultimately realized through Service Operations, therefore making it a critical capability" [5].

Processes under this stage: Event Management, Access Management, Request Fulfillment, Problem Management and Incident Management.

5. **Continual Service Improvement:** This stage is concerned with regularly assessing and improving provided services. "The primary purpose of CSI is to continually align and realign IT services to the changing business needs by identifying and implementing improvements to IT services that support business processes" [6].

Among the processes mentioned above within the various stages, only the following 5 will be covered in this project and are described below. These processes were selected because they fulfilled the requirements of the company.

a. Incident Management

Incident Management focuses on restoring service disruptions as well as responding to service requests as quickly as possible, in order to minimize business impact. This process belongs to the service operation stage.

"It is the process for dealing with all incidents; this can include failures, questions or queries reported by the users (usually via a telephone call to the Service Desk), by technical staff, or automatically detected and reported by event monitoring tools" [5].

b. Problem Management

Problem Management is concerned with identifying the root-cause of incidents, and taking measures to give a proper solution. Afterwards, similar incidents can be prevented or they can find a quicker resolution [5]. This process is also defined by the service operation lifecycle.

c. Change Management

A request for change (RFC) is created with the aim of studying and implementing change on existing assets. Change management tries to ensure necessary changes are made without creating a major disruption to existing assets and processes. There needs to be a change advisory board (CAB) selected in order to approve or reject requested changes. “Potential members include [5]: Customer(s), Applications developers/maintainers, Specialists/technical consultants, Services and operations staff, e.g. Service Desk, test management, ITSCM, security, capacity, Facilities/office services staff (where changes may affect, moves/accommodation and vice versa), Contractor’s or third parties’ representatives, e.g. in outsourcing situations”

d. Asset Management

This is concerned with managing everything related to an asset’s lifecycle from planning its acquisition to its disposal in the end. It aims to “define and control the components of IT services and to maintain information about those components” [11].

e. Service Level Management

“Service Level Management (SLM) negotiates, agrees and documents appropriate IT service targets with representatives of the business, and then monitors and produces reports on the Service Provider’s ability to deliver the agreed level of service” [6]. SLA is defined in the service design phase under service level management (SLM).

2.3 Service Desk

“The Service Desk provides a point of communication to the users and a point of coordination for several IT groups and processes” [5].

One of the main objectives of an IT service desk is to provide a single point of contact between customers and IT personnel. The other is to streamline the restoration of normal service with as little impact on the users as possible and within agreed upon service levels.

Figure 1.1 shows how the different processes and functions interact with each other. Data collected from different sources such as the incident management process may be input to the problem management process for further analysis. Change management can then be initiated if necessary and after its implementation, the problem will have been solved and similar open incidents will get their resolution.

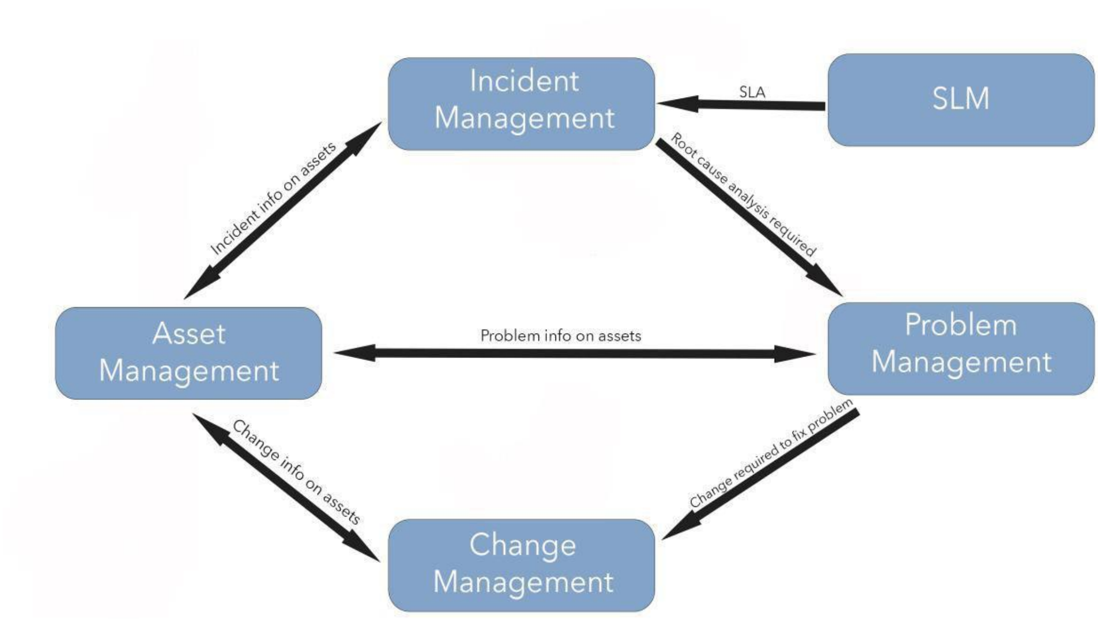

Figure 1.1: Process Flow for Solving Issues

2.4 Automatic Assignment

2.4.1 Full text search

Full text search and ranking was used in literature found in [12] to assign agents without machine learning. An organizational model is built with domain specific attributes; these attributes contain information about the departments within the organization, what they are in charge of and their employees. When a new task comes, a full text search is performed on the organizational model and the possible candidates are ranked. The task is then assigned to the topmost candidate.

2.4.2 Rule-based Induction

"Rule induction is an area of machine learning in which formal rules are extracted from a set of observations" [13]. Rules usually contain one or more conditions along with a conclusion and are represented as "IF condition Then conclusion" [13]. Rule based systems have 4 components namely – the working memory, the knowledge base, the inference engine and the user interface. The working memory contains facts about the domain that the inference engine takes to match with a rule that applies from the knowledge base. The user interface, as the name indicates, is used for the interaction between the system and the user.

2.4.3 Text Classification using Data mining

Text Classification, in data mining, is the assignment of text to predefined categories based on a training data provided to train the classifier. The training data is preprocessed during which each word is given a numeric value for the classification algorithms to classify it.

There are several classification algorithms as shown below:

Decision Trees

Decision tree builds classification models in the form of a tree structure. It has nodes, which represent different possibilities, connected with branches. The first node is called the root and the last nodes are called leaves and they represent the classification. "A root node (e.g., Outlook) has two or more branches (e.g., Sunny, Overcast and Rainy). Leaf node (e.g., Play) represents a classification or decision. In the case of regression the Leaf node (e.g., Hours Played) represents a decision on the numerical target" [14].

Bayesian Networks

Bayesian networks (BNs), are types of probabilistic graphical models [15]. They are used to model the probabilistic dependencies among different variables. They have nodes to represent the variables and these nodes are connected with edges which are the paths that show their dependencies. BNs also show "the quantitative strength of the connections between variables, allowing probabilistic beliefs about them to be updated automatically as new information becomes available” [16].

K-nearest Neighbor

K-nearest neighbor (k-NN) classification, is useful in the case of discrete set of classes. “An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small)” [17].

Summary

After reviewing several literature, the ITIL processes mentioned in this chapter were selected. These processes were found to be part of a problem solving workflow. When an incident is reported, it needs to be completed in a certain amount of time as defined by the SLA. When it’s found to have an underlying cause or if it’s difficult to solve then a problem is created in the Problem Management section for root cause analysis. After the analysis, if there needs to be a change in the system, a change is created in the Change Management section and reported to CAB and divided into tasks to be implemented.

The above mentioned processes will be made available through the Service Desk. In order for the service desk to be more effective, assigning technicians automatically was identified to be a useful feature. To achieve that, some techniques were reviewed and text classification was identified to be the best solution for the type of result we want to achieve.

Chapter 3 – Related Work

In this chapter, related works are explained. There are several studies done in automatic assignment of resources which will be discussed along with studies about ITIL based service desks. The first thesis implements an ITIL based service desk with automatic resolver group assigning and search knowledge base function.

"The IT service desk is a crucial function of incident management driven by alignment with the business objectives of an enterprise that requires IT support, balancing their operations and achieving desired service level targets" [18]. The thesis introduced in [18] shows the importance of knowledge management for a service desk system and develops a system for IT service desk outsourcing in the banking business called KMRCA IT service desk.

One of the focus areas of the thesis in [18] is developing an automatic resolver group assignment function. Using the classification function of data mining, a model is generated by using existing data. When a user reports an incident, the system does word extraction from the description and "implements the ID3-based method to generate a pattern and to identify a suitable resolver group" [18]. It then calculates the percentages of matching words in the assigned resolver group to check and see if the percentage matches the criteria set. If it does, it assigns the resolver group and if it doesn't it notifies the service desk agents for them to do the assigning.

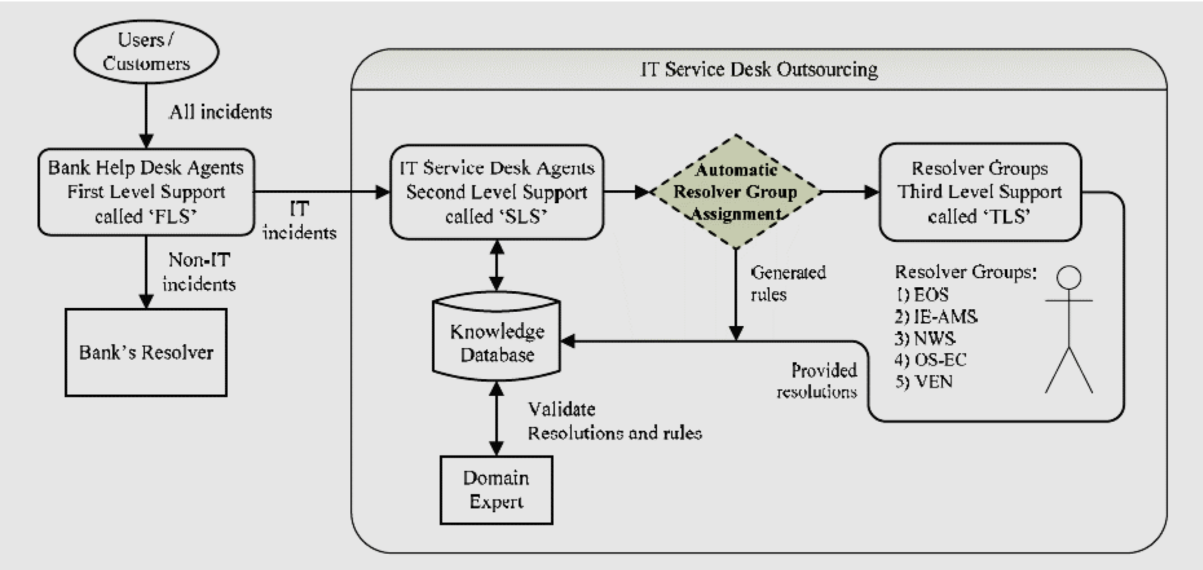

Figure 3.1: KMRCA IT Service Desk [18]

Figure 3.1 shows the process flow of KMRCA IT Service Desk. It displays the search knowledge base function as well as the automatic resolver group assignment.

Another study in [19] focuses on automatic ticket assignment and explains that the manual assignment of issues is highly error prone. It also states that making the end users select the category and sub category for their issues may be difficult especially for new users and also may be a tedious task to figure out how their issue should be categorized.

The system proposed has two text classification phases. The first one is to find the category for the issue while the second one tries to determine its subcategory. “If the prediction confidence is not greater than the predetermined threshold value, the ticket will be routed to an operator that can make the correct assignment” [19].

Four different classifications were examined: SVM, Naive Bayes, kNN and Decision Tree. The results found varied depending on the training data set used.

The automatic resolver group section of the first thesis is similar to this project, however it doesn’t include assigning a specific technician. In this project, technicians will be assigned to incidents by checking their workload and performances. The thesis discusses knowledge base searching functions which are not included in this project because it wasn’t identified as a requirement during the problem identification phase.

The second thesis explains why allowing the end user categorize the incident or manually assigning them by operators can be problematic. It discusses a two phased classification in order to assign the correct category to an incident.

Classification based on support team instead of multiple categories was more applicable to the data at hand since a second category did not exist.

Chapter 4 - Requirements Analysis

This chapter will discuss the current system and its problems. It will then describe the proposed solutions and display the diagrams that represent the requirements of the service desk system.

4.1 Current System

Currently, incidents in the IT department of ET can be classified into 3 categories:

- Computer Hardware Maintenance and Network related Incidents/Requests
- Application related Incidents/Requests
- Other IT related Incidents/Requests

The first ones regarding Computer Hardware and Networking maintenance are handled by a system called Mascot. It is a web application that allows users to register their incident, then a person at the central services desk assigns it to a technician, the technician can then assign it to a higher level technician and/or give closure. It also enables the staff to record various IT Assets to avoid misuse.

Application related incidents are resolved independently by each IT expert working on the respective application software accordingly. They are usually contacted through email or by phone calls and the incidents will not be logged.

Incidents that do not fall under the first two categories are handled by various support giving departments. They are mostly unregistered and are forgotten about after the solution is provided. In rare cases, these incidents may be registered on the Mascot system.

The registration form as displayed in Figure 4.1 provides different fields where a user can enter the details of the incident.

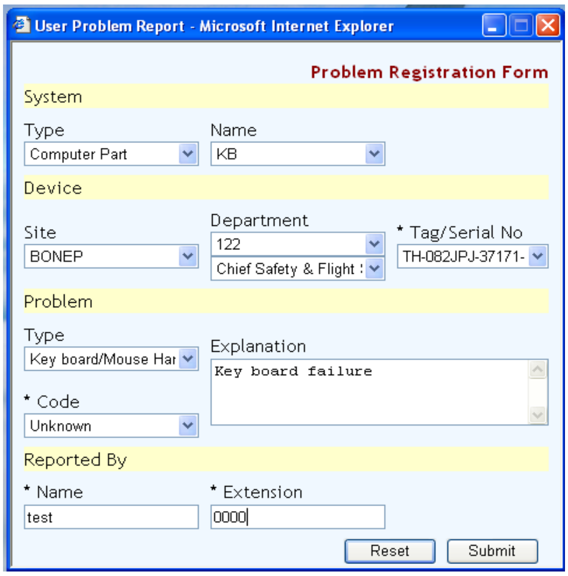

Figure 4.1: Incident Registration on the Mascot System

After a user submits the information about the incident, a person at the call center will be able to see it and check to see if it's a problem that he/she can solve. If it's possible, they can contact the user and solve the problem. However, if the person cannot fix it, he/she can assign it to technicians that can resolve the incident. Figure 4.2 shows how a technician can view and/or search through various.

PRF Number: [ ] Show Detail Edit

1 - 25 of 365 << < > >>

No.	PRF Number	Report Date	System Type	Tag/Serial No.	Department	Ext.	Status	Action By	Action Date
101	58828	12/6/2010 12:15:02 PM	SU	SU0137	General Accounting	8875	Fixed On Site	14321 - Biniam	12/6/2010 1:16:50 PM
102	58827	12/6/2010 12:13:55 PM	SU	SU2049	Distribution	8875	Fixed On Site	14321 - Biniam	12/6/2010 2:46:57 PM
103	58826	12/6/2010 12:12:22 PM	SU	SU1037	Adminstration & Svcs Avi	8482	Fixed On Site	14020 - Tilahun	12/7/2010 8:45:22 AM
104	58825	12/6/2010 12:09:46 PM	MN	MN1311	Appearance Control	8715	Escalated	14339 - Biruk	12/6/2010 2:36:12 PM
105	58824	12/6/2010 12:08:36 PM	Extention	8188	Appearance Control	8715	Fixed On Site	14339 - Biruk	12/7/2010 10:49:58 AM
106	58823	12/6/2010 12:07:25 PM	Extention	8650	Administeration & Finance Cat	8168	Unchecked	14020 - Tilahun	12/6/2010 12:07:25 PM
107	58822	12/6/2010 12:04:32 PM	OP	OP0024	Passenger Revenue & Interline	8029	Escalated	14634 - Dawit	12/6/2010 2:08:52 PM
108	58821	12/6/2010 12:01:07 PM	OP	OP134	HR & Infn & Employees Aff	8037	Fixed On Site	14321 - Biniam	12/6/2010 12:21:50 PM
109	58820	12/6/2010 11:59:32 AM	OP	VNBVB8RGHG	Commercial Operations Traning	8372	Fixed On Site	14634 - Dawit	12/6/2010 2:39:54 PM
110	58819	12/6/2010 11:56:11 AM	SC	CN29KS06 1 8	Engineering	8838	Checked	14339 - Biruk	12/6/2010 3:19:05 PM
111	58818	12/6/2010 11:30:14 AM	KB	KB0298	Centeral Reservations	8547	Gate Pass	111 - Default	12/6/2010 1:57:40 PM
112	58817	12/6/2010 11:05:22 AM	Extention	8401	Comm Admin & Interline	8401	Fixed On Site	14321 - Biniam	12/6/2010 11:10:56 AM
113	58816	12/6/2010 11:02:41 AM	Extention	8393	Centeral & West Africa Admin	8393	Fixed On Site	14321 - Biniam	12/6/2010 11:12:46 AM
114	58815	12/6/2010 10:37:03 AM	NEWOFFICE	newooficemon	Training Methods & Standards	8430	Fixed On Site	14634 - Dawit	12/6/2010 2:02:30 PM
115	58814	12/6/2010 10:34:00 AM	OP	OP9006	Appearance Control	8299	Unchecked	14339 - Biruk	12/6/2010 10:34:00 AM
116	58813	12/6/2010 10:27:18 AM	Direct	0116630618	Area Mgr N,S & W	0	Unchecked	14321 - Biniam	12/6/2010 10:27:18 AM
117	58812	12/6/2010 10:23:05 AM	LP	CNU7361BKK	MOCC & Outstations	8448	Escalated	14339 - Biruk	12/6/2010 1:18:15 PM

Figure 4.2: List of Incidents on the Mascot System

4.2 Proposed system

Ethiopian Airlines is an internationally competing company with the majority of its systems following international standards. The importance of following international standards is to ensure that business operations are as efficient as possible and that quality service is being provided to their customers. For this reason, it's necessary that the Service Desk System of the company follows an international standard as well.

When developing the ITIL Service Desk System for ET, processes that fulfill their requirements were selected. These processes are SLM, incident management, problem management, change management and asset management from Service Design, Service Transition and Service Operation phases.

Service Level Agreements (SLAs) are defined in the system to ensure that tasks are handled within a defined period of time agreed upon between customers and service providers. When an incident occurs, it needs to be handled by the service provider within the time stated in the SLA.

During incident management, users enter free text to describe the incident they are reporting. Text classification will then be used to assign the incident to a support team based on the description. After the support team is identified, the system retrieves technicians within that support team and assigns the incident to a technician with less workload. Since a technician needs to finish his task within the time specified in the SLA, workload will be calculated based how much time is left for him/her to finish their work. However, if the new incident is set to the highest priority, the technician can "pause" his work and resume to it when the new task is done.

When incidents occur repeatedly a root cause analysis needs to be done and this is where the problem management comes into focus. When a problem is created, several tasks are created which may be assigned to different technicians. Each tasks have deadlines and the problem can only be closed when these tasks are completed.

During the problem management, if the team decides that a change needs to be made to a system and/or a process, they will submit a Request for Change in the change management process so that it gets studied and approved. If the change is approved by the CAB, tasks will be created and assigned similar to the problem management and the change will be implemented.

Asset Management is used to define and keep the history of various assets. These assets can be either IT related or non IT related.

4.3 Functional requirements

Functional requirements describe the interactions between the system and its users. Users include people who interact with the system as well as other external systems. We have identified the following as functional requirements of the proposed Service Desk system based on ITIL V3.

1. **Service Level Management**

FR-01: The system should allow the defining and editing of SLA rules

FR-02: The system should allow the deletion of SLA rules

FR-03: The system should automatically update Incidents to apply SLA rules to determine resolution time.

FR-04: The system should help to track SLA compliance using automatic escalations (up to 3 levels i.e. 1st level technician, 2nd level technician etc.) in case of SLA violations at pre-defined time intervals

FR-05: The system should measure Service level performance using SLA violation reports.

2. **Incident Management**

FR-06: The system should be able to register incidents

FR-07: The system should enable automatically assigning the Priority to an Incident based on the Impact and Urgency levels. It should also provide the capability for technicians to assign the priority manually to override the one made by the system

FR-08: The system should enable all level escalations manually and automatically. It should also take in consideration of Service Level Agreement which helps in prioritizing the incident and enabling timely escalations

FR-09: The system should support users to find all current and past incidents which are equal or similar to the search parameters. It should support finding all incidents for a specific Configuration Item

FR-10: After identifying the issue and resolving the incident, the system should automatically add the resolution as a part of the FAQ

FR-11: The system should provide the privilege for the user to close his incident. At times the user might also not respond back even if the issue is resolved, in this case technicians should be able to give closure

FR-12: The system should provide a satisfaction survey to users. Managers should be able to see the results.

FR-13: The system should get data daily from the attendance system of the company to be able to assign incidents to available technicians.

3. Problem Management

FR-14: The system should allow raising new Problems independently and linking Incidents to Problems.

FR-15: The system should support problem management staff to assign a person who takes responsibility for the coordination and the planning of a solution of the problem.

FR-16: The system should allow users to search through and view existing problems with ongoing status as well as completed.

FR-17: The system should allow users to create different tasks under a problem and assign it to different technicians

FR-18: The system should allow users to log action taken on the problems as well as close problems

4. Change Management

FR-20: The system should allow initial submission of request for change “RFC”

FR-21: The system should enable CAB Members to make a decision to **accept or reject** a Change Plan based on information submitted on the change submission form

FR-22: The system should enable creating CABs (change Advisory board). Based on the defined change type, the system enables technicians choose to send the change plan for approval to CAB members

FR-23: The system should help Change Managers to assign, schedule and track tasks.

FR-24: The system should allow users to close RFC

5. **Assets Management**

FR-26: The system should maintain all assets and employees information. Each asset is uniquely identified and tracked

FR-27: The system should help users to get detail information such asset state, license, warranty expiry date etc.

6. **Automatically Assigning Technicians**

FR-28: The system should automatically assign incidents to a support team

FR-29: The system should assign an incident to a specific technician within the support team based on the workload and availability of a technician

4.4 Non Functional Requirement

Documentation: The system shall have a manual which describes the workflow of the system.

User Interface: The user interface shall be responsive so that users are able to access it from devices with different screen sizes (eg – Desktop PCs, Laptops, Mobiles, Tablets etc.)

Portability: Users shall access the system from any operating system with an internet browser.

Security

Authentication

The system shall incorporate the following tasks regarding to authentication:

- Authenticate users through credentials, such as username and password.
- Enforce the usage of diverse character sets; in this case the password shall have a minimum of six characters which is the combination of alphabets, digits and special characters.

Role based Access Control

The system should provide administrators the ability to determine who can perform what actions, from where, in what order, and in some cases under what relational circumstances.

Robustness: The system shall be able to handle error conditions gracefully, without failure and shall continue its smooth operation to the end users with them not knowing that an error occurred.

Flexibility: The system shall be designed assuming that features will be added to the system.

4.5 System models

4.5.1 Use Cases

A use case is a list of actions that define the interactions between an actor and a system, in order to accomplish a task. The actor can be a human or other external system. The use cases for the Service Desk system are described below.

Actors:

Customers – a person reporting incidents/requesting services and tracking resolution process

Technicians – Includes all levels of support technicians: 1st level technician, 2nd level technician, 3rd level technician. They get assigned to tasks in Incident management, problem management and change management systems and are able to see the progress of tasks they are assigned to. They also register assets and are in charge of updating their status.

Managers – Their role is to define SLA and overlook the tasks that are assigned to technicians as well as view reports. Unresolved issues that are beyond the capacity of the 3rd level technicians will be viewed by them for decision making.

CABs – people that are in charge of accepting or rejecting requests for change.

Track SLA system – component of the system that escalates incidents to higher level technicians when an SLA is violated

Service Level Management

Use case 1.1 - Define SLA

Name	Define SLA
Primary Actors	Managers
Description	An actor should be able to set SLA (Service Level Agreement) for each category and priority level
Preconditions	An actor is logged in and has access to the SLM
Basic course of Actions	1. The actor selects “SLA Management” 2. The system displays a list of SLAs 3. The actor selects category and priority Level 4. The actor sets Days, Hours and Minutes 5. The actor clicks on the “Save” button 6. The system saves the information on the database
Post conditions	SLA will be saved

Use case 1.2 - View SLA

Name	View SLA
Primary Actors	Technicians, Managers
Description	A user should be able to view SLA that was previously set
Preconditions	A user is logged in and has access to SLA management
Basic course of Actions	1. The actor selects “SLA Management” 2. The system displays a list of SLAs 3. The actor views set SLA
Post conditions	Users will view information

Use case 1.3 - Edit SLA

Name	Edit SLA
Primary Actors	Managers
Description	A user should be able to edit SLA that was previously set
Preconditions	A user is logged in, has access to SLA management and SLA was previously set

Basic course of Actions	1. The actor will select “SLA Management” 2. The system will display the list of SLAs 3. The actor will select “Edit” on a specific SLA 4. The system will display the content of the SLA for editing 5. The actor will make changes and click on the save button 6. The system will save the changes
Post conditions	SLA will be edited

Use case 1.4 – Track SLA Violation

Name	Track SLA Violation
Primary Actors	SLM
Description	The system should track incidents to make sure they are being handled according to the SLA. When the agreed upon SLA is violated, the system should initiate the Assign Technician task
Preconditions	An SLA is defined
Basic course of Actions	1. The system will check if an incident violated the SLA or not 2. If not, the system will do nothing 3. If so, the system will initiate the Assign Technician component
Post conditions	SLA will be enforced

Use case 1.5 – View SLP

Name	View SLP
Primary Actors	Manager
Description	The system should display the Service Level Performance based on SLA violations
Preconditions	SLA is defined and Incidents are registered
Basic course of Actions	1. Manager clicks on “View SLP” 2. The system display a list with SLP information 3. Manager views detail of SLP
Post conditions	SLP will be viewed

Incident Management

Use case 2.1 – Register Incident

Name	Register Incident
Primary Actors	Customers, Technicians, Managers
Description	An actor should be able to register incidents
Preconditions	An actor is logged in
Basic course of Actions	1. The actor Select “Incidents” 2. The system displays a list of incidents with options to edit, delete or add new incidents 3. The actor selects “Add New” 4. The system displays a form 5. The actor fills in the form 6. The actor clicks on the “Save” button 7. The system saves the information on the database
Post conditions	Incident will be registered

Use case 2.2 – Assign Support Team

Name	Assign Support Team
Primary Actors	Incident Management System
Description	The system should assign a technician based on the description of the incident.
Preconditions	An incident is registered and the classification model is created
Basic course of Actions	1. The system will take the incident description and the words will be lower cased and tokenized then stop words will be removed. 2. The system will then use the classification model to identify the support team 3. The system initiates the “Assign Technician” task passing the support team information
Post conditions	Support Team will be identified

Use case 2.3 – Update Support Team Model

Name	Update Support Team Model

Primary Actors	Windows Scheduler
Description	The system should update the support team model weekly in order to get up-to-date information for classifying support teams
Preconditions	A scheduler is set up to initiate the process periodically
Basic course of Actions	4. The scheduler initiates the update model subsystem 5. The system will perform data cleaning tasks 6. The system will perform data transformation tasks 7. The system will perform data reduction tasks 8. The system will build and update the classification model
Post conditions	Support Team Model will be updated

Use case 2.4 – Get Attendance Information

Name	Get Attendance Information
Primary Actors	Incident Management system
Description	The system gets attendance information from the company's attendance system. The data will be imported from XML files.
Preconditions	A user is logged in
Basic course of Actions	1. The system checks for an XML file at a specified hour 2. If file is found, the system imports the data and saves it to the database.
Post conditions	Employee's attendance information will be available

Use case 2.5 - Assign Technicians

Name	Assign Technicians
Primary Actors	Technicians, Managers, Incident Management System
Secondary Actors	Technicians
Description	The system should assign tasks based on the level, availability and workload of a technician.

Preconditions	Incidents are registered on the system
Basic course of Actions	Case 1: A new incident is registered 1. The system uses the classification model to identify a Support Team and initiates the "Assign Technician" task 2. Within the identified Support team, the system checks for availability and workload of technicians 3. The system assigns a technician Case 2: A technician assigns another technician 1. The technician/manager selects "Incident Management" 2. The system displays a list of incidents 3. The technician/manager clicks on "Edit" 4. The system displays the edit page 5. The technician/manager selects another technician and saves the form 6. The system saves the update Case 3: SLA is Violated 1. The system checks if an incident has exceeded the processing time according to the SLA 2. The system checks for availability and workload of technicians 3. The system assigns a higher level technician Case 4: Technician on the highest level (Level 3) is unable to resolve the incident 1. The system checks if the SLA time is exceeded while a level 3 technician is trying to resolve it 2. The system updates the status to "Unresolved" and displays the information to the Manager of the department for decision.
Post conditions	Technicians will be assigned to incidents

Use case 2.6 – Set Priority

Name	Set Priority
Primary Actors	Technicians, Managers, Incident Management System
Description	When the incident is registered, the system automatically gives it a priority based on Urgency and Business Impact.

	Then a user should be able to change the priority of an incident.
Preconditions	A user is logged in and an incident is registered
Basic course of Actions	System: 1. The actor saves an incident 2. The system checks the values for urgency and business impact and calculates priority 3. The system sets the priority for the incident Users: 1. The actor selects “Incident Management” 2. The actor selects a specific incident 3. The actor updates priority level 4. The actor clicks on the “Save” button 5. The system updates the information
Post conditions	Priority will be set

Use case 2.7 – View Incident

Name	View Incident
Primary Actors	Customers, Technicians, Managers
Description	A user should be able to view registered incidents
Preconditions	Customer: can view incident that he/she has submitted, Technician: can view incidents that are assigned to him/her Managers: can view all incidents
Basic course of Actions	1. The actor will select “Incidents” 2. The system will display list of incidents 3. The actor will view the list 4. The actor will select one to view the details of the incident 5. The system will display the details on an incident
Post conditions	Users will view information

Use case 2.8 – Search Incidents

Name	Search Incidents
Primary Actors	Customers, Technicians, Managers

Description	A user should be able to search through registered incidents
Preconditions	Customer: can search for an incident that he/she has submitted, Technician: can search for incidents that are assigned to him/her Managers: can search for any incident
Basic course of Actions	1. The actor will select "Incidents" 2. The system will display list of incidents 3. The actor will view the list 4. The actor will type in a keyword on the search field 5. The system will search for results related to the keyword 6. The actor will view the search results
Post conditions	Users will view information

Use case 2.9 – Close Incident

Name	Close Incident
Primary Actors	Customers, Incident Management
Description	An incident should be closed by the customer if they're satisfied with the resolution or by the system if there's no response from the customer after a certain period of time.
Preconditions	An incident is registered
Basic course of Actions	1. The actor will select "Incidents" 2. The system will show list of incidents 3. The actor will select one of the incidents to view the details 4. The actor will select "Closed" from status 5. The system will save the changes
Post conditions	Incident will be closed

Use case 2.10 – Give Resolution

Name	Give Resolution
Primary Actors	Technician

Description	A user should be able to write down a resolution and update status of incident
Preconditions	A user is logged in
Basic course of Actions	1. The actor will select “Incidents” 2. The system will show list of incidents 3. The actor will select a specific incident 4. The actor will write resolution 5. The actor will update Status 6. The actor will click on the Save button 7. The system will save the changes
Post conditions	A resolution will be given to an incident

Use case 2.11 – Fill Satisfaction Survey

Name	Fill Satisfaction Survey
Primary Actors	Customers
Description	A user should be able to fill satisfaction survey
Preconditions	A user is logged in
Basic course of Actions	1. The actor selects “Satisfaction Survey” 2. The system displays the satisfaction survey form 3. The actor fills in the necessary information and clicks on “Save” 4. The system saves the form
Post conditions	Satisfaction Survey will be filled

Use case 2.12 – View Filled Satisfaction Survey

Name	View Filled Satisfaction Survey
Primary Actors	Managers
Description	A user should be able to view filled satisfaction survey
Preconditions	A user is logged in and has access to the page
Basic course of Actions	1. The actor will select “View Filled Satisfaction Survey” 2. The system will display a page which consists of the satisfaction survey 3. The actor will view a page with a filled out satisfaction survey

Post conditions	Users will view filled satisfaction surveys

Problem Management

Use case 3.1 – Register Problem

Name	Register Problem
Primary Actors	Technicians, Managers
Description	A user should be able to register problems
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects "Problem Management" 2. The system displays a list of problems 3. The actor clicks on "Add New" 4. The system displays the new problem registering form 5. The actor fills in the form 6. The actor clicks the "Save" button 7. The system saves the problem on the database
Post conditions	Problem will be registered

Use case 3.2 – Create Tasks

Name	Create Tasks
Primary Actors	Technician, Managers
Description	A user should be able to create a task and assign technicians to various tasks within a problem
Preconditions	A user is logged in as a manager
Basic course of Actions	1. The actor selects the “Tasks” tab under “Problem Management" 2. The system displays a list of tasks 3. The actor clicks on "Add New" 4. The system displays the task form 5. The actor fills in the form 6. The actor clicks the "Save" button

	7. The system saves the task on the database
Post conditions	Users will assign a technician to a task

Use case 3.3 – Assign Tasks

Name	Assign Tasks
Primary Actors	Technicians, Managers
Secondary Actors	Technicians
Description	A user should be able to assign different technicians to various tasks within a problem
Preconditions	A user is logged in
Basic course of Actions	1. The actor selects “Problem Management” 2. The system displays a list of all registered problems 3. The actor selects the detail for a specific problem and go to the “Tasks” section. 4. The actor displays a list of tasks for a problem, the user can add new or edit an existing one. 5. The actor can enter task information such as description, priority level and assign it to a technician. 6. The actor clicks on the “Save” button 7. The system updates the database with the new information
Post conditions	Users will assign a technician to a task

Use case 3.4 – View Problem

Name	View Problem
Primary Actors	Technicians, Managers
Description	A user should be able to view registered problems
Preconditions	A user is logged in as a technician or a manager

Basic course of Actions	1. The actor selects "Problem Management" 2. The system displays a list of all registered problems 3. The actor select one to view details 4. The system renders the problem's details
Post conditions	Users will view problem information

Use case 3.5 – Search Problems

Name	Search Problems
Primary Actors	Technicians, Managers
Description	A user should be able to search through registered problems
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects "Problem Management" 2. The system displays a list of all registered problems 3. Type a search keyword in the search input and click search 4. The system searches the database and renders the results
Post conditions	Users will view information

Use case 3.6 – Close Problem

Name	Close Problem
Primary Actors	Technician, Problem Management
Description	A problem should be closed by the technician if they're satisfied with the resolution
Preconditions	A problem is registered

Basic course of Actions	1. The actor selects “Problem Management” 2. The system displays a list of all registered problems 3. The actor selects one to view details 4. The actor selects the “Problem Status” tab 5. The actor clicks on “Add New” 6. The actor select “Closed” from status and writes the reason for closure 7. The actor clicks on “Save Changes” 8. The system updates the problem’s status on the database
Post conditions	Problem will be closed

Use case 3.7 – Schedule Problem Solving

Name	Schedule Problem Solving
Primary Actors	Technicians, Managers
Description	A technician or a manger should be able to set a start date and due date for tasks of a problem to be resolved
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects “Problem Management” 2. The system displays a list of all registered problems 3. The actor selects a specific problem and clicks “edit” 4. The actor picks start date and end date and clicks “save” 5. The system updates the start and end dates for the problem on the database
Post conditions	Users will set schedule for a registered problem

Use case 3.8 – Log Action Taken

Name	Log Action Taken
Primary Actors	Technicians, Managers
Description	A user should be able to log action taken on a problem

Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects “Problem Management” 2. The system displays a list of all unsolved problems 3. The actor selects a specific problem and goes to the “Problem Status” section 4. The actor enters information such as the action taken and findings 5. The actor clicks on “Save Changes” 6. The system saves the action taken
Post conditions	Action taken will be registered

Change Management

Use case 4.1 – Create RFC

Name	Create RFC
Primary Actors	Technicians, Managers
Description	A user should be able to create requests for change
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects “Change Management" 2. The system displays a list of requests for change 3. The actor clicks on "Add New" 4. The system displays the request for change form 5. The actor fills in the form 6. The actor clicks the "Save" button 7. The system saves the RFC on the database
Post conditions	RFC will be registered

Use case 4.2 – View RFC

Name	View RFC
Primary Actors	Technicians, Managers

Description	A user should be able to view registered RFCs
Preconditions	Customer: can view problem that he/she has submitted, Technician: can view problem that are assigned to him/her Managers: can view all problems
Basic course of Actions	1. The actor selects “Change Management” 2. The system displays a list of ongoing changes 3. The actor select one to view details 4. The system renders the change’s details
Post conditions	Users will view change information

Use case 4.3 – Search for RFC

Name	Search for RFC
Primary Actors	Technicians, Managers
Description	A user should be able to search through requests for change
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects “Change Management" 2. The system displays a list of RFCs 3. Type a search keyword in the search input and click search 4. The system searches the database and renders the results
Post conditions	RFC search result will be shown

Use case 4.4 – Close RFC

Name	Close RFC
Primary Actors	Technician, Managers
Description	A user can close an RFC

Preconditions	A RFC is registered
Basic course of Actions	1. The actor selects “Change Management” 2. The system displays a list of all RFCs 3. The actor selects one to view details 4. The actor select “Closed” from status 5. The actor clicks on “Save Changes” 6. The system updates the RFC’s status on the database
Post conditions	RFC will be closed

Use case 4.5 – Create RFC Tasks

Name	Create RFC Tasks
Primary Actors	Technicians, Managers
Description	A user should be able to create tasks for RFCs
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects the “Tasks” tab under “Change Management" 2. The system displays a list of tasks 3. The actor clicks on "Add New" 4. The system displays the task form 5. The actor fills in the form 6. The actor clicks the "Save" button 7. The system saves the RFC task on the database
Post conditions	RFC task will be registered

Use case 4.6 – Assign Tasks

Name	Assign Tasks
Primary Actors	Technicians, Managers
Secondary Actors	Technicians

Description	A user should be able to assign different technicians to various tasks within a change request
Preconditions	A user is logged in
Basic course of Actions	1. The actor selects “Change Management” 2. The system displays a list of all registered change requests 3. The actor selects the detail for a specific one and goes to the “Tasks” section. 4. The system displays a list of tasks and the user can add new or edit an existing one. 5. The actor can enter task information such as description, priority level and assign it to a technician. 6. The actor clicks on the “Save” button 7. The system updates the database with the new information
Post conditions	Users will assign a technician to a task

Use case 4.7 – Schedule Task Resolution

Name	Schedule Task Resolution
Primary Actors	Technicians, Managers
Description	A technician or a manger should be able to set a start date and due date for a task to be completed.
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects “Change Management” 2. The system displays a list of change requests 3. The actor selects a specific change request and clicks “edit” 4. The actor picks start date and end date and clicks “save” 5. The system updates the start and end dates for the change request task on the database
Post conditions	Users will set schedule for a registered change request

Use case 4.8 – Log Action Taken

Name	Log Action Taken
Primary Actors	Technicians, Managers
Description	A user should be able to log action taken on a change
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects "Change Management" 2. The system displays a list of all ongoing changes 3. The actor selects a specific change and goes to the "Change Status" section 4. The actor enters information such as the action taken and findings 5. The actor clicks on "Save Changes" 6. The system saves the action taken
Post conditions	Action taken will be registered

Use case 4.9 – Accept/Reject Change

Name	Accept/Reject Change
Primary Actors	CAB Member
Description	A Change advisory board member should be able to accept or reject a change
Preconditions	A user is logged in as a change advisory board member
Basic course of Actions	1. The actor selects "Change Management" 2. The system displays a list of all RFCs 3. The actor selects one to view details 4. The actor clicks on accept or reject change 5. The system saves the action taken
Post conditions	Action taken will be registered

Asset Management

Use case 5.1 –Register Assets

Name	Register Assets
Primary Actors	Technicians, Managers
Description	A user should be able to create assets
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects "Asset Management" 2. The system displays a list of assets 3. The actor clicks on "Add New" 4. The system displays the new asset registering form 5. The actor fills in the form 6. The actor clicks the "Save" button 7. The system saves the asset on the database
Post conditions	Asset will be registered

Use case 5.2 – View Asset

Name	View Asset
Primary Actors	Technicians, Managers
Description	A user should be able to view registered assets
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects “Asset Management” 2. The system displays a list of all registered assets 3. The actor select one to view details 4. The system renders the asset’s details
Post conditions	Users will view asset information

Use case 5.3 –Search for Assets

Name	Search for Assets

Primary Actors	Technicians, Managers
Description	A user should be able to search for an asset
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects “Asset Management" 2. The system displays a list of assets 3. Type a search keyword in the search input and click search 4. The system searches the database and renders the results
Post conditions	Asset search result will be shown to user

Use case 5.4 –Edit Asset

Name	Edit Asset
Primary Actors	Technicians, Managers
Description	A user should be able to edit an asset
Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects “Asset Management” 2. The system displays a list of all registered assets 3. The actor picks an asset and clicks “Edit” 4. The actor edits the asset info 5. The actor clicks “Save” 6. The system updates the asset information
Post conditions	Asset will be edited

Use case 5.1 –Delete Asset

Name	Delete Asset
Primary Actors	Technicians, Managers
Description	A user should be able to delete an asset

Preconditions	A user is logged in as a technician or a manager
Basic course of Actions	1. The actor selects "Asset Management" 2. The system displays a list of all registered assets 3. The actor picks an asset and clicks "Delete" 4. The system deletes the asset from the database
Post conditions	Asset will be deleted

4.5.2 Use Case Model

A use case model represents the interaction between actors and the system. Figure 4.4 shows the SLA use case with the tasks that the different actors perform. Technicians and Managers enter Maximum Processing Time for each incident categories. After an SLA is set, the system can use it to track and report incidents that violated it.

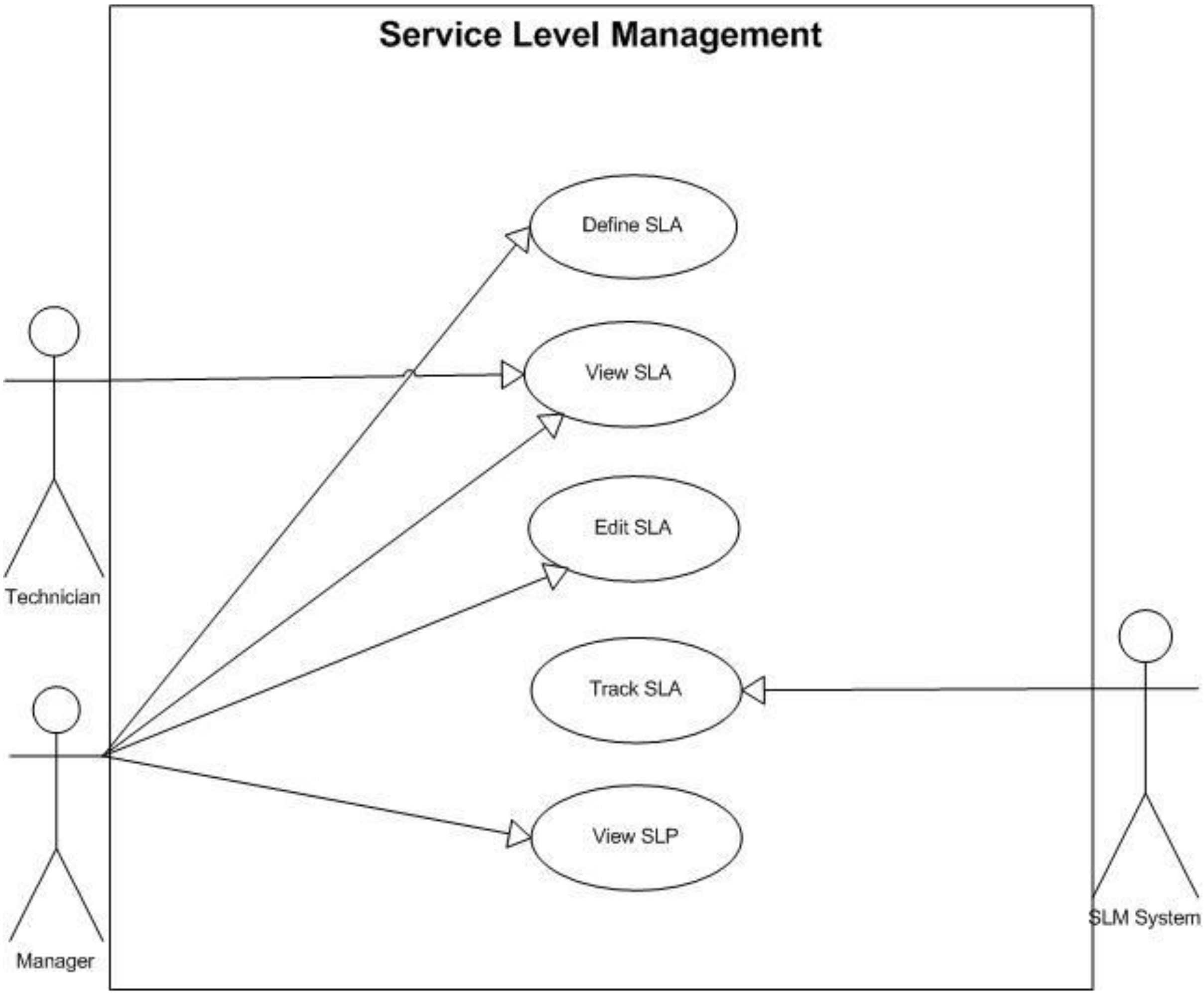

Figure 4.4: Use Case Model for SLA

Figure 4.5 shows the Incident Management use case, it involves tasks related to incidents such as registering, viewing, closing etc. It is part of the system that customers have access to.

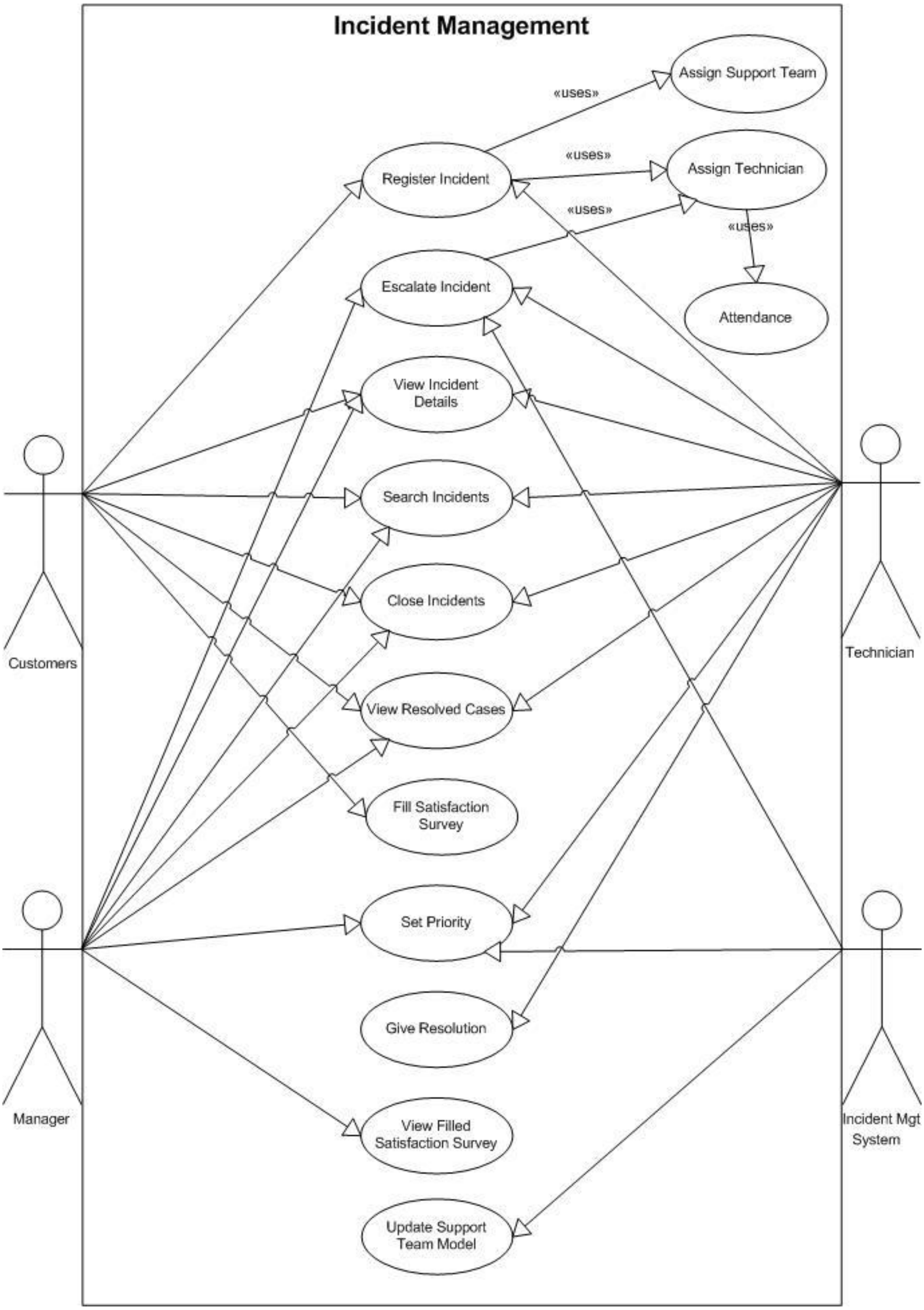

Figure 4.5: Use Case Model for Incident Management

Figure 4.6 shows the use case model for updating the Support Team model.

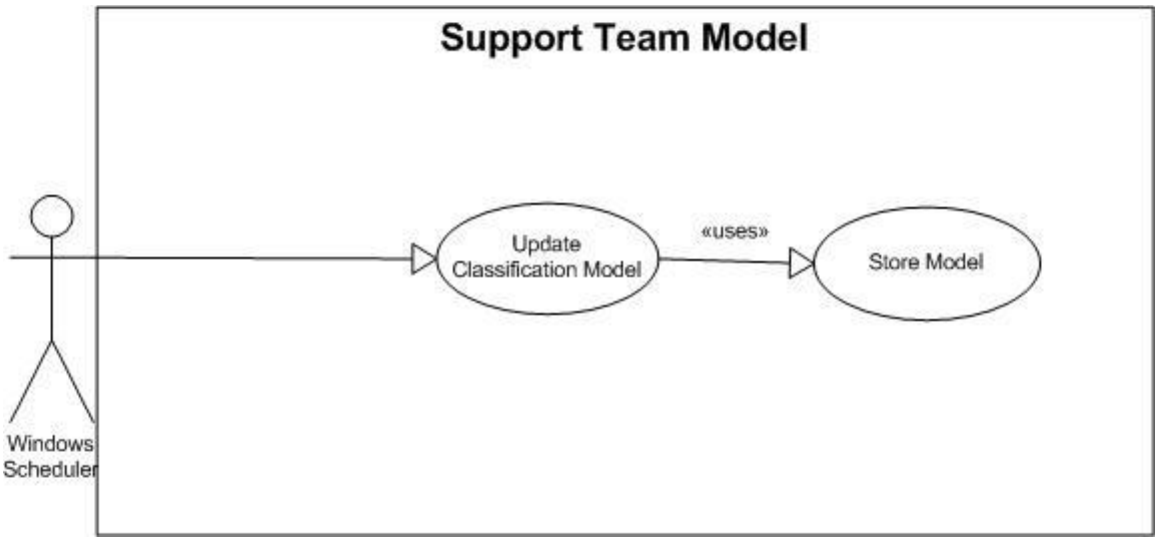

Figure 4.6: Use Case Model for Updating Support Team Model

Figure 4.7 shows the Problem Management use case with the tasks that the different actors perform. It involves tasks related to problems and it is part of the system that the company's staff use for root cause analysis on incidents.

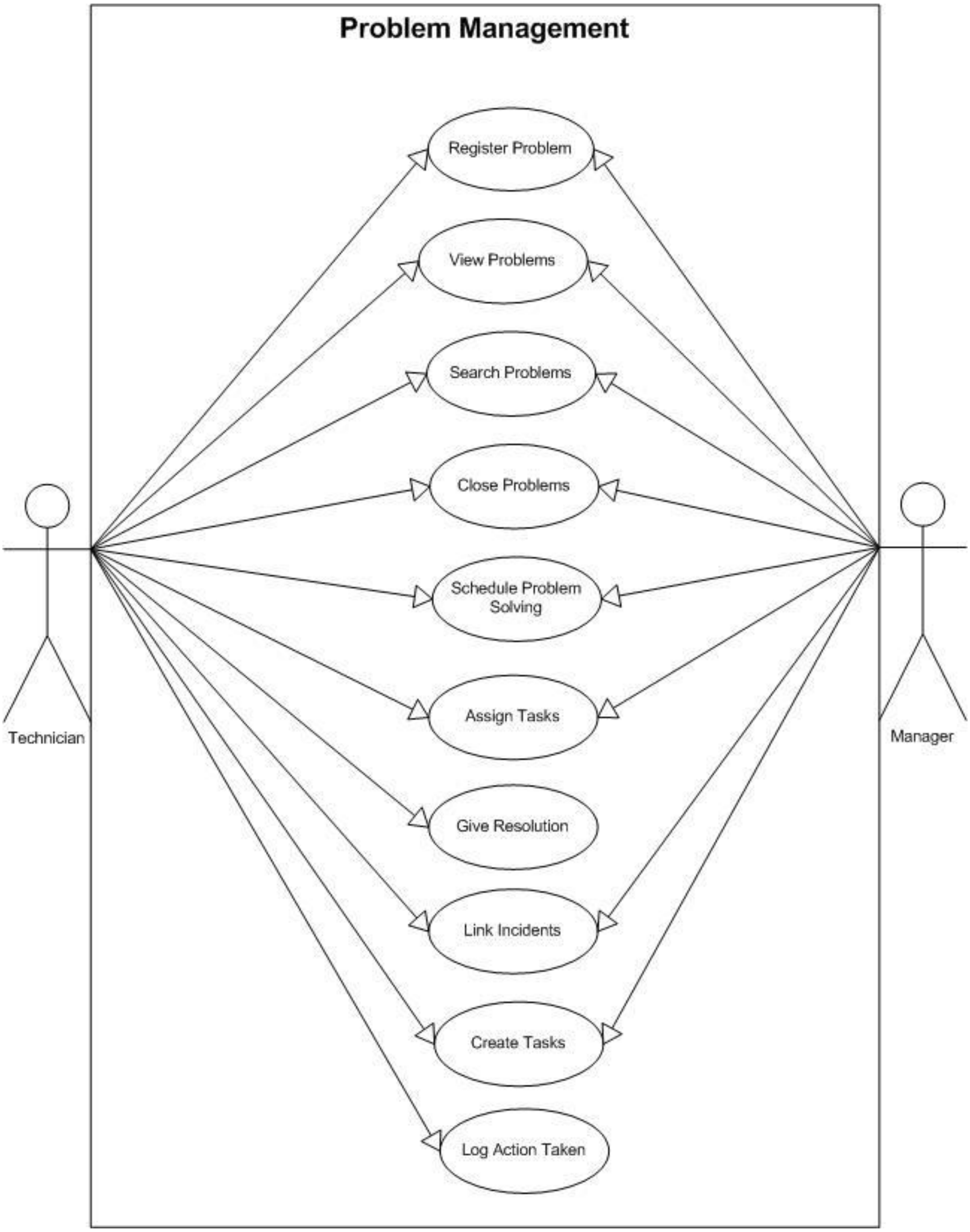

Figure 4.7: Use Case Model for Problem Management

Figure 4.8 shows the Change Management use case which involves tasks related to change management and it is part of the system that the company's staff use for managing change related tasks including accepting or rejecting it.

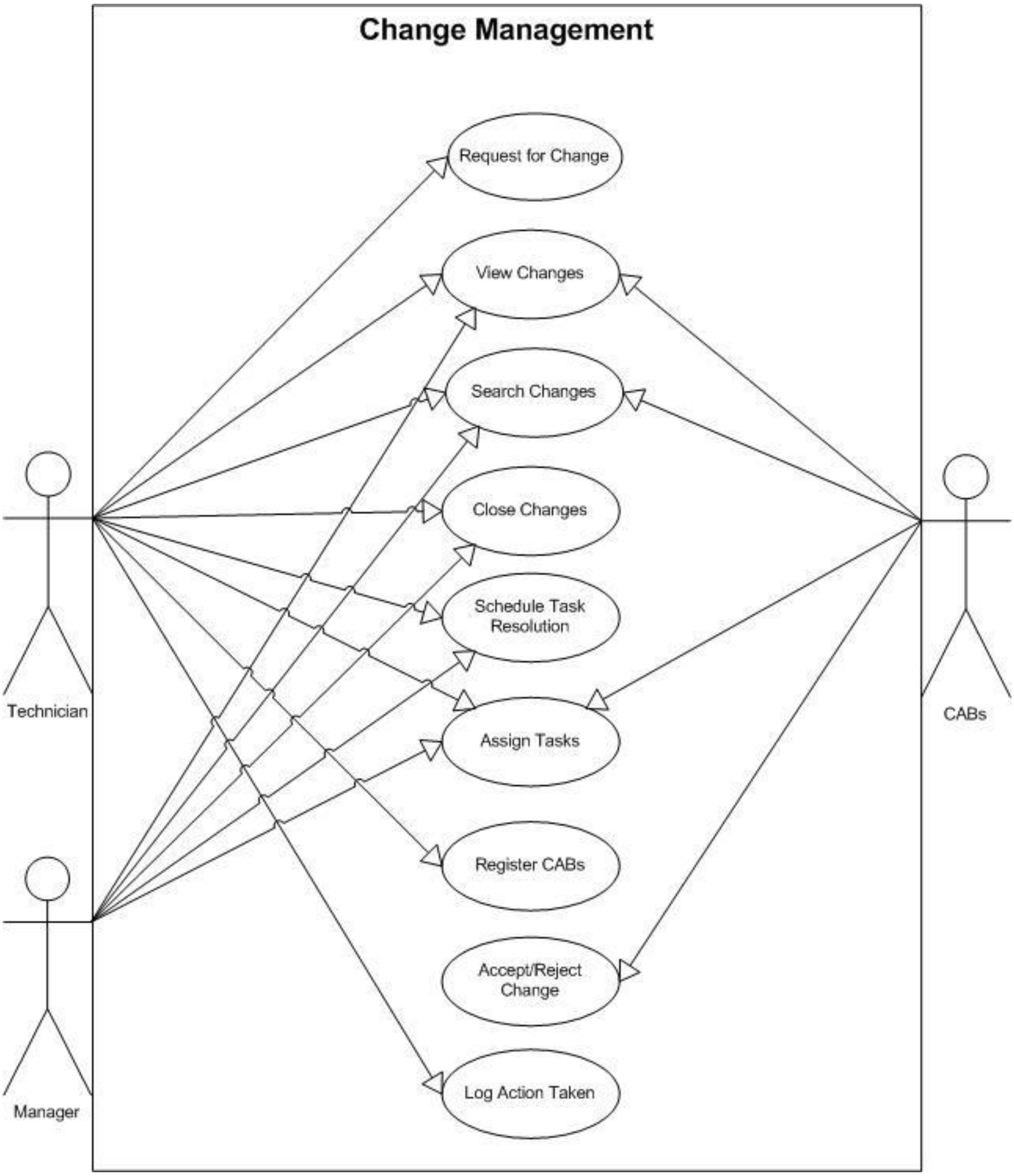

Figure 4.8: Use Case Model for Change Management

Figure 4.9 shows the Asset Management use case with the tasks that the different actors perform. It involves tasks related to Assets and it is part of the system that the company's staff use for registering their different types of assets in the system.

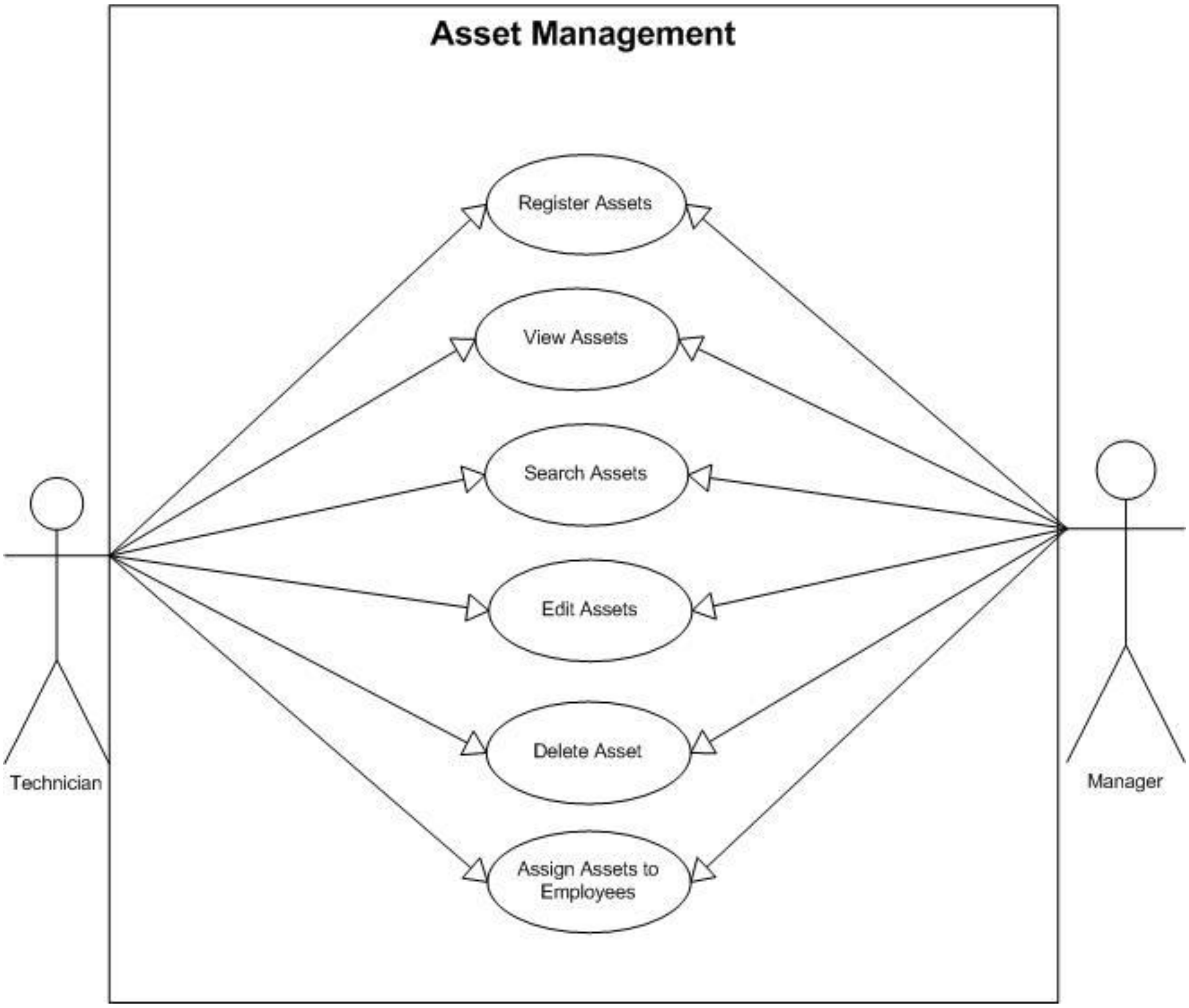

Figure 4.9: Use Case Model for Asset Management

4.5.3 Class Diagram

A Class diagram displays the classes that exist in a system along with their attributes and operations and relationships. The classes identified for the Service Desk system are displayed below.

Figure 4.10 displays the class diagram for SLM.

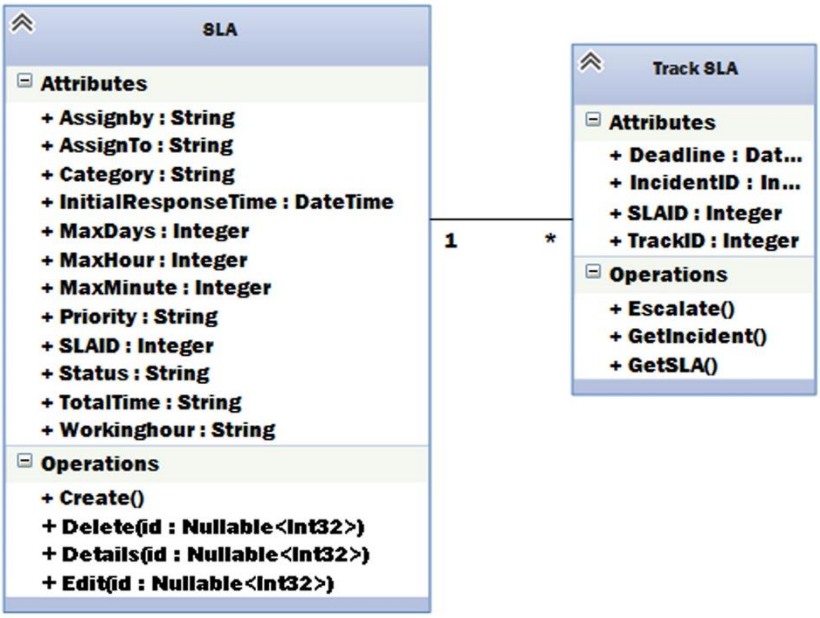

Figure 4.10: Class Diagram for SLM

Figure 4.11 shows the classes, associations, attributes and operations for Incident Management System.

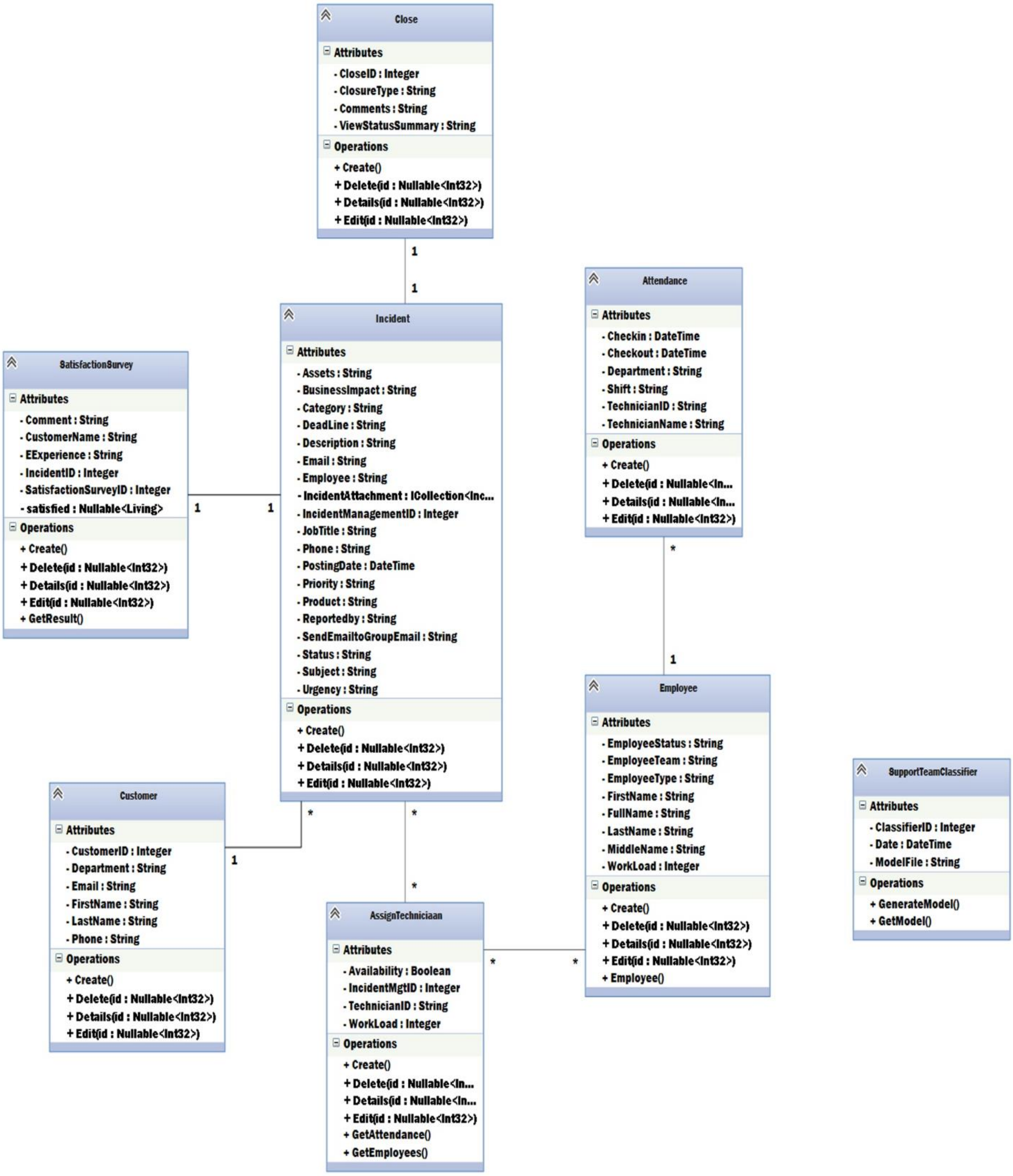

Figure 4.11: Class Diagram for Incident Management

Figure 4.12 shows the classes, associations, attributes and operations for Problem Management system.

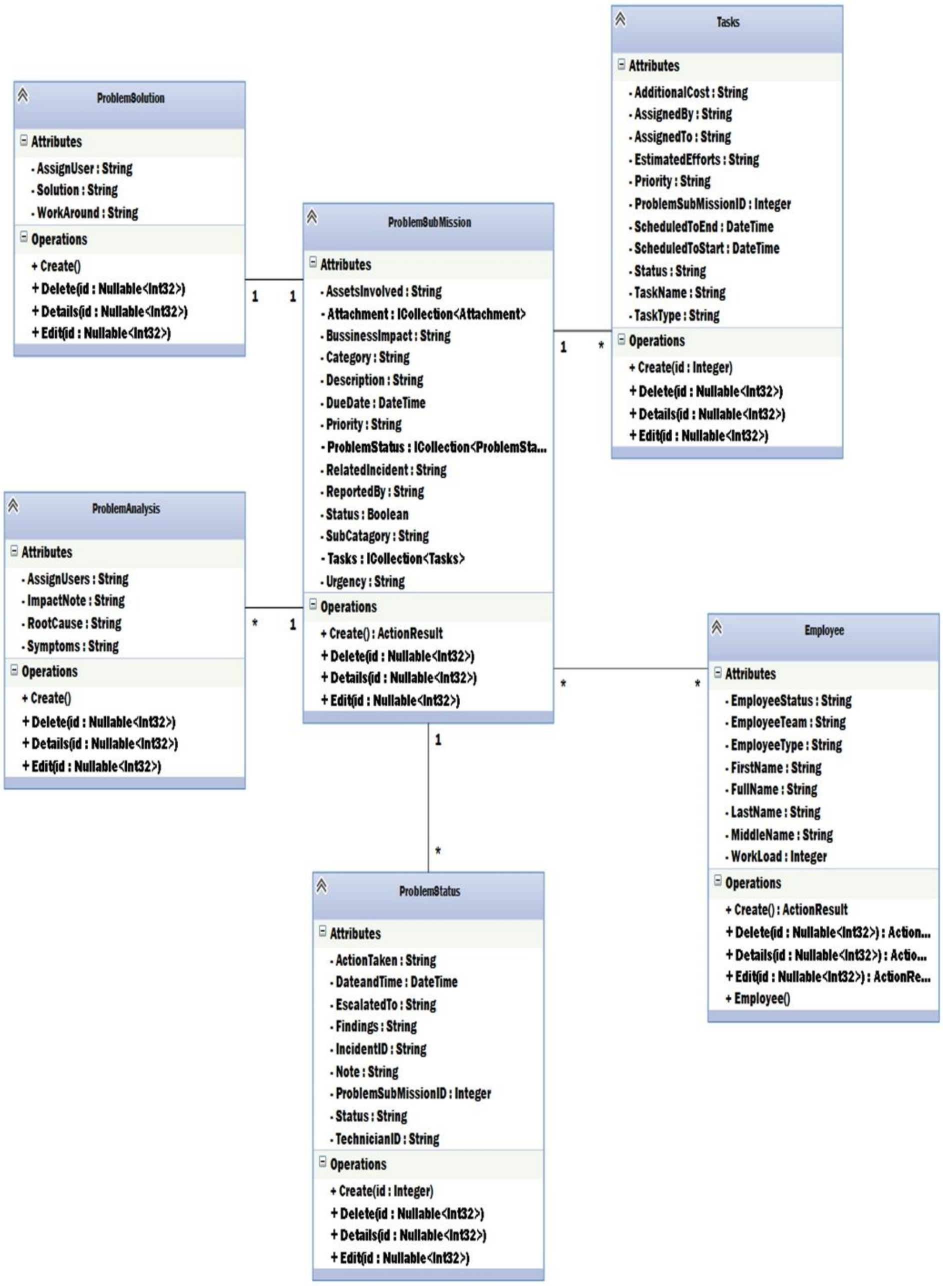

Figure 4.12: Class Diagram for Problem Management

Figure 4.13 shows the classes, associations, attributes and operations for Change Management System.

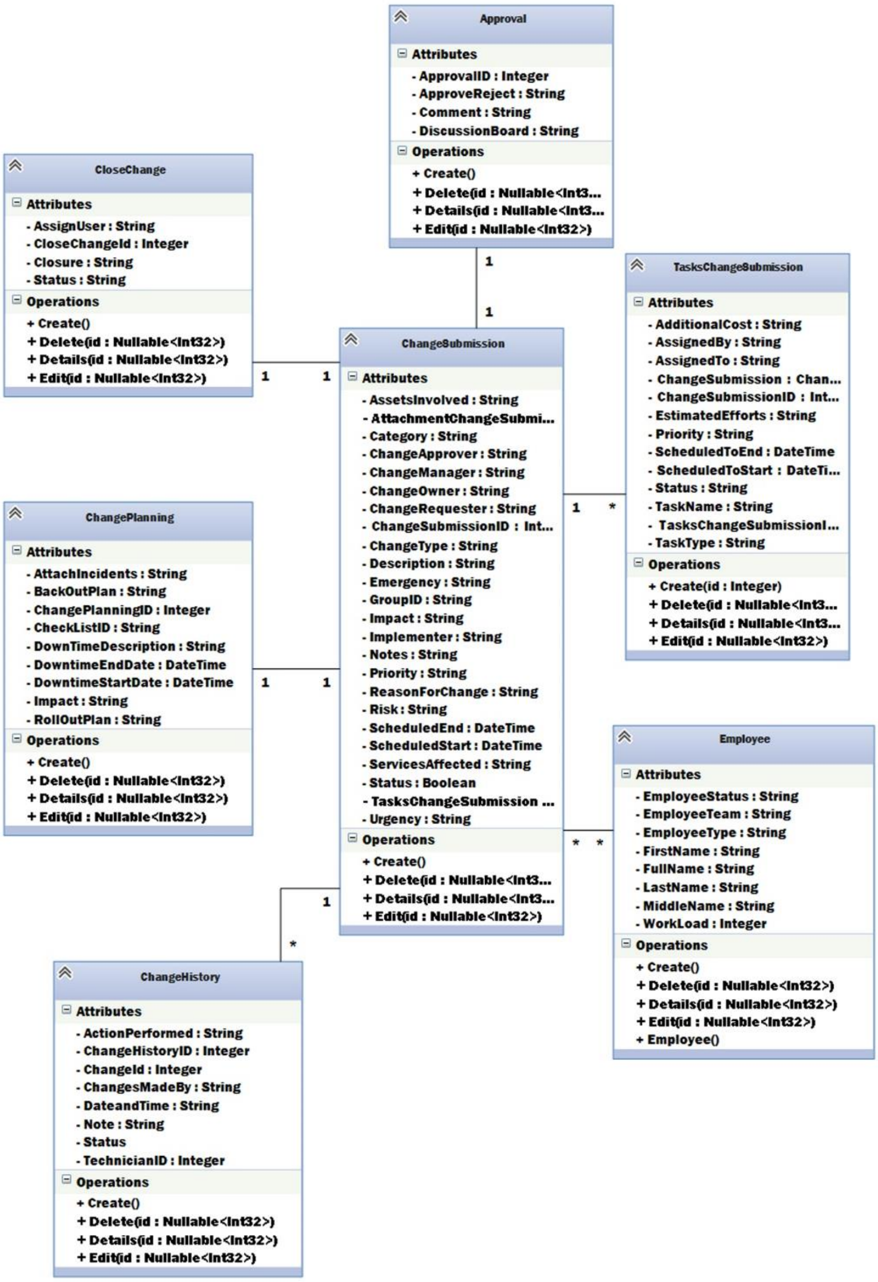

Figure 4.13: Class Diagram for Change Management

Figure 4.14 shows the classes, associations, attributes and operations for Asset Management System.

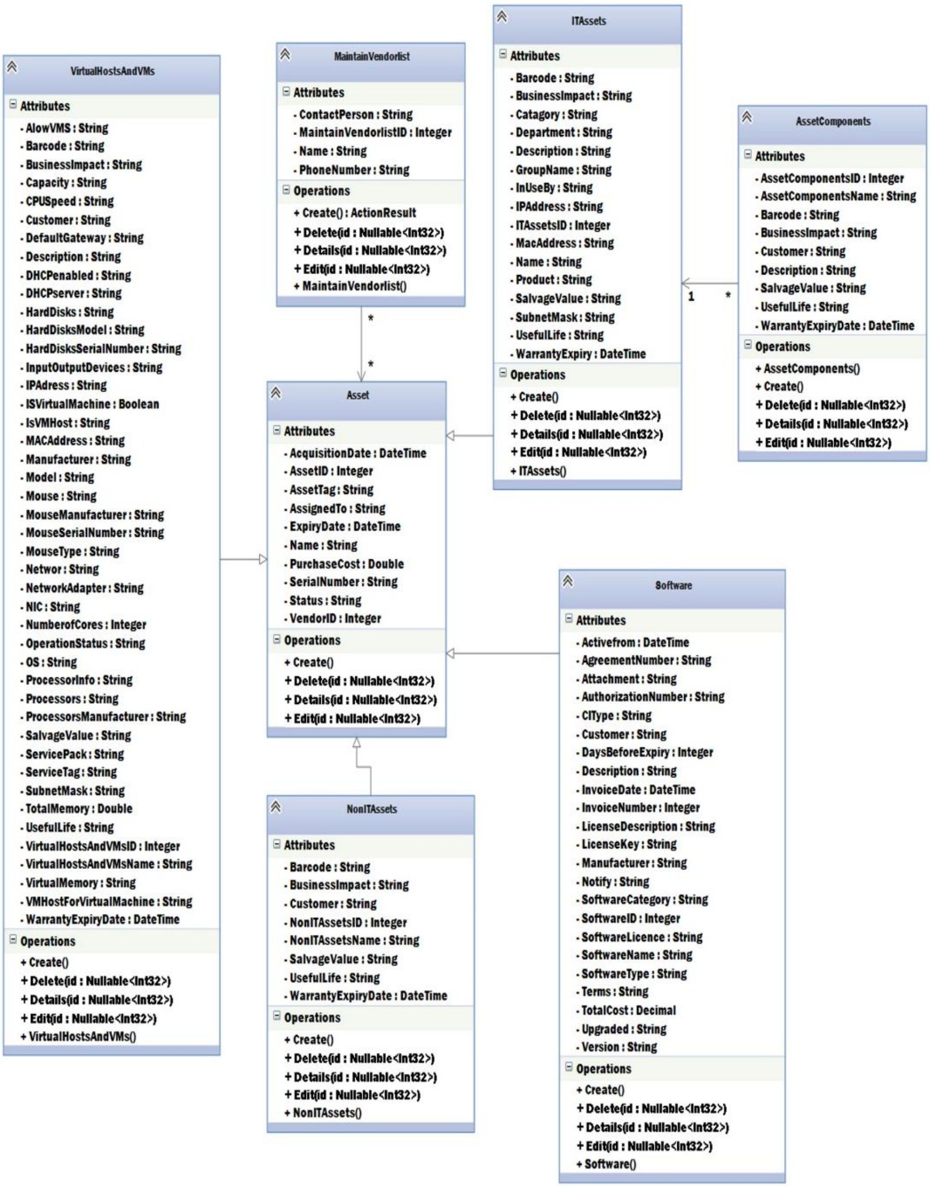

Figure 4.14: Class Diagram for Asset Management

4.5.4 Sequence Diagrams

Sequence diagrams are used to show the communication between the different objects of the system. Below are sequence diagrams for the different components of the Service Desk system. Each have three objects: the UI which is the interface that the user interacts with, the Controller which processes the information and communicates with the Model when necessary and the Model which handles database related actions.

SLA Management

Figure 4.15 shows how a user interacts with the system when defining a new SLA.

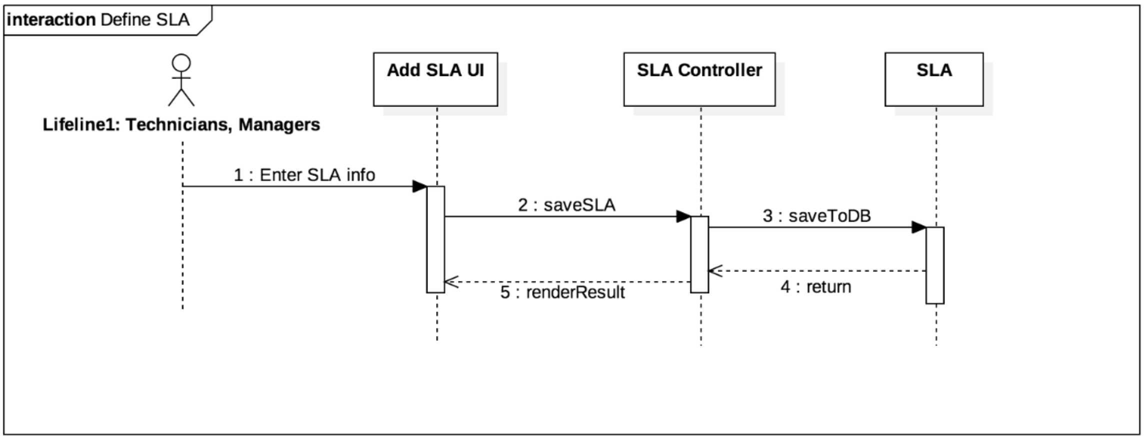

Figure 4.15: Define SLA Sequence Diagram

Figure 4.16 shows the interaction when displaying already defined SLAs.

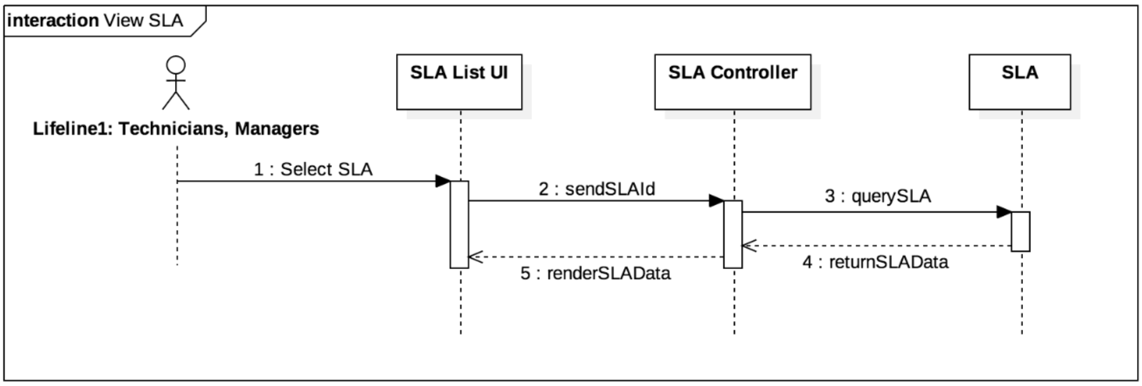

Figure 4.16: View SLA Sequence Diagram

Fig 4.17 displays the sequence when editing an SLA.

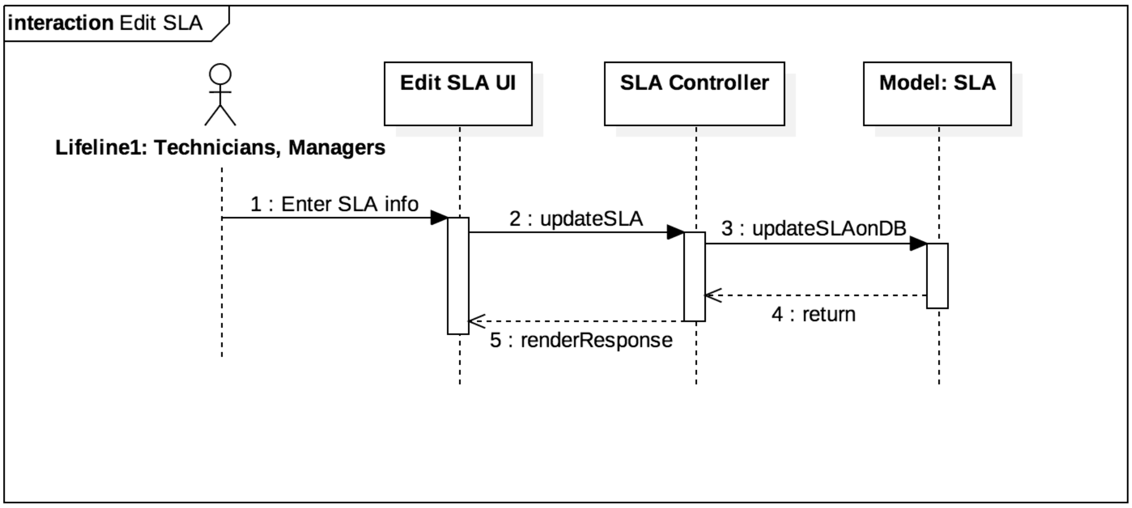

Figure 4.17: Edit SLA Sequence Diagram

Figure 4.18 shows the interaction that occurs when the system escalates an incident when an SLA is violated.

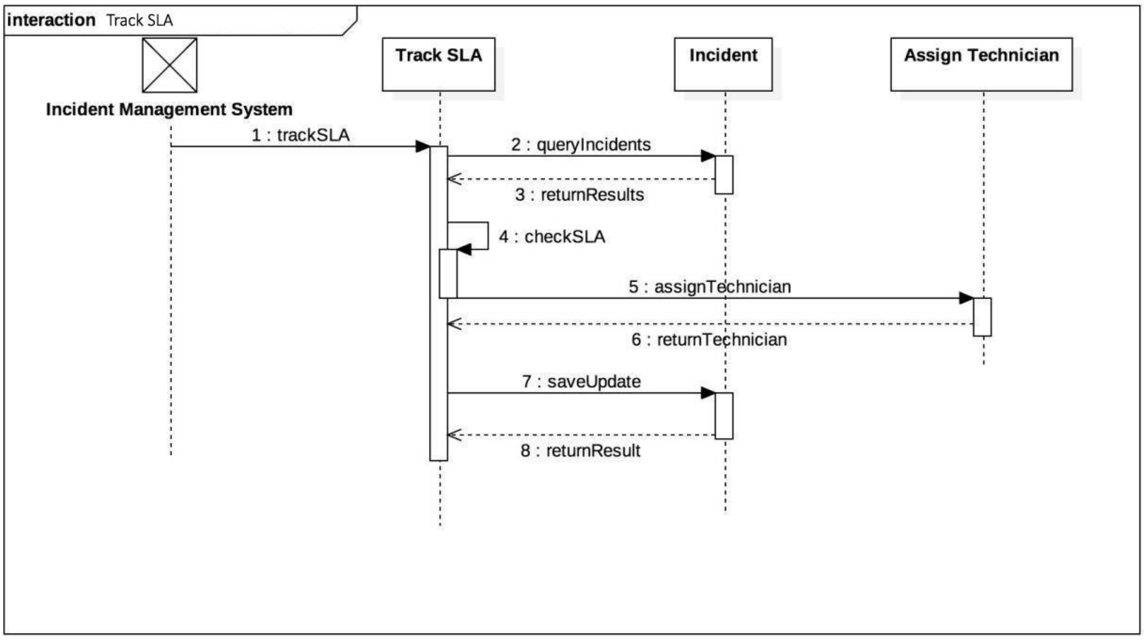

Figure 4.18: Track SLA

Incident Management

Figure 4.19 shows the interaction between objects when a user registers an incident

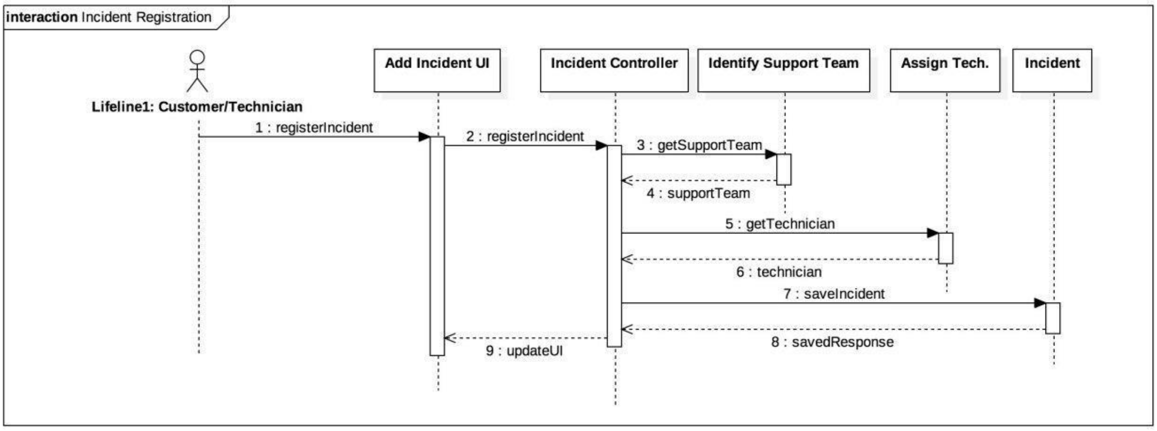

Figure 4.19: Register Incident Sequence Diagram

Figure 4.20 illustrates the communication between objects when searching for incidents.

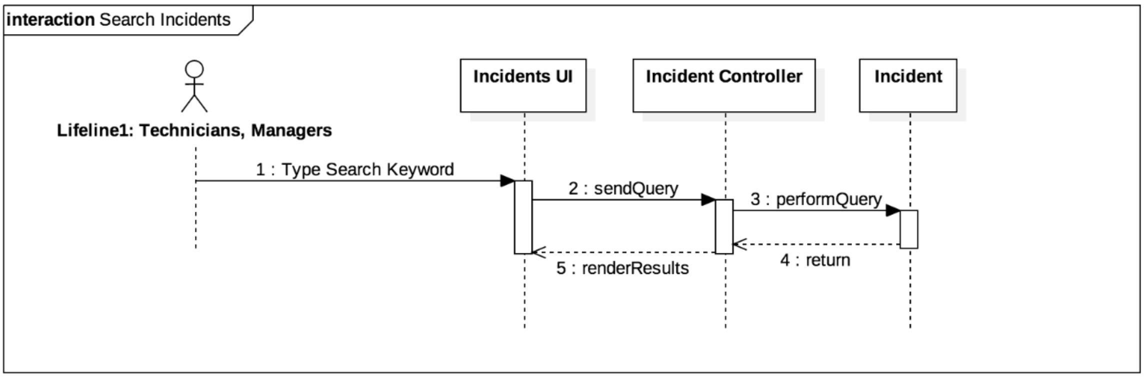

Figure 4.20: Search Incidents Sequence Diagram

Figure 4.21 shows the sequence diagram when viewing incidents. Incidents are displayed according to the role and privilege of the user.

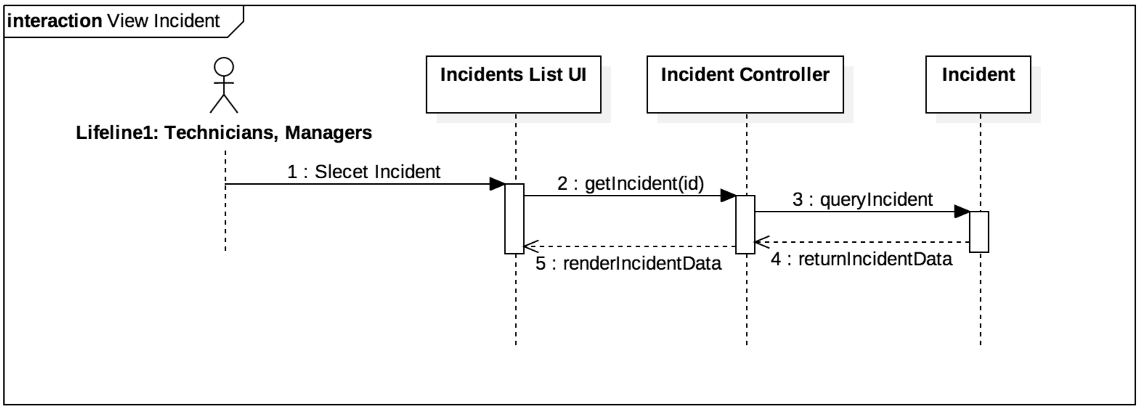

Figure 4.21: View Incident Sequence Diagram

Figure 4.22 shows the sequence diagram during the closing of an incident.

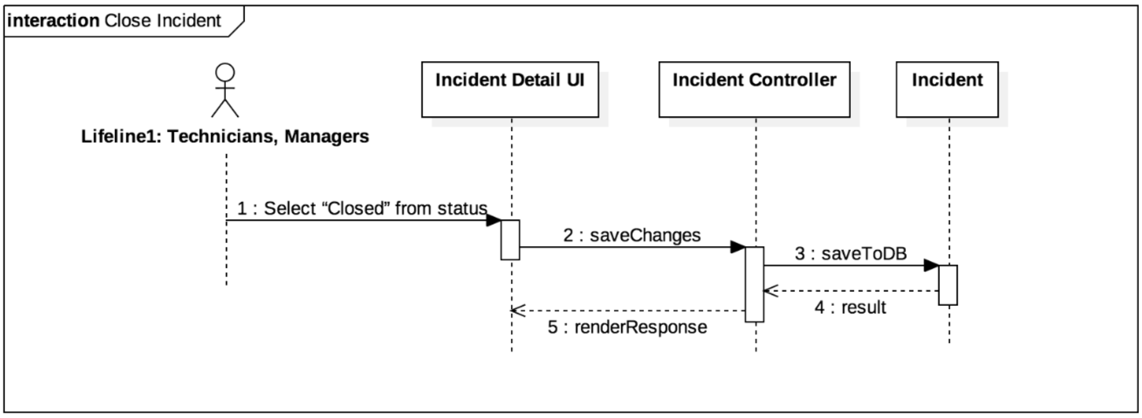

Figure 4.22: Close Incident Sequence Diagram

Figure 4.23 shows the sequence diagram when displaying resolved cases for users to go through to answer any questions they may have.

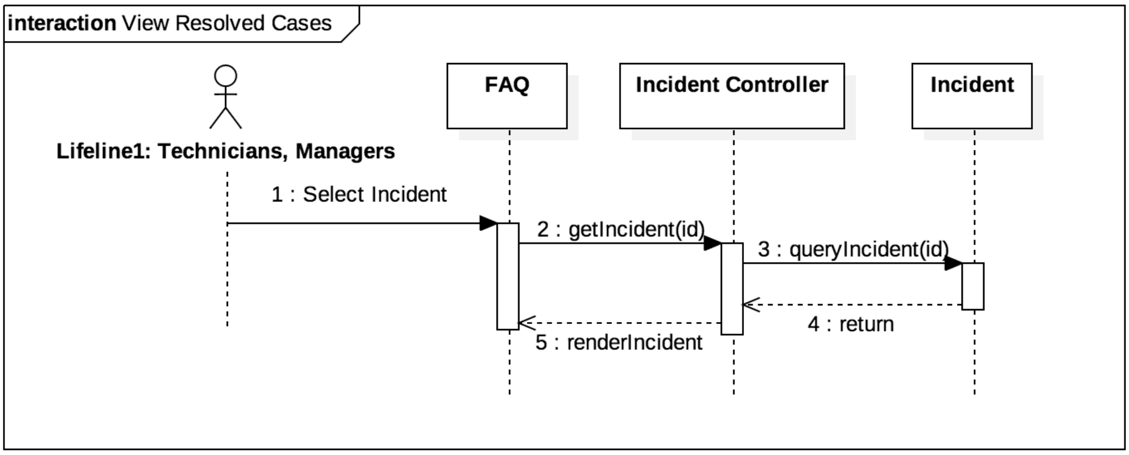

Figure 4.23: View Resolved Incidents Sequence Diagram

Figure 4.24 shows the interaction when customers fill in the satisfaction survey for the handling of the incident they reported.

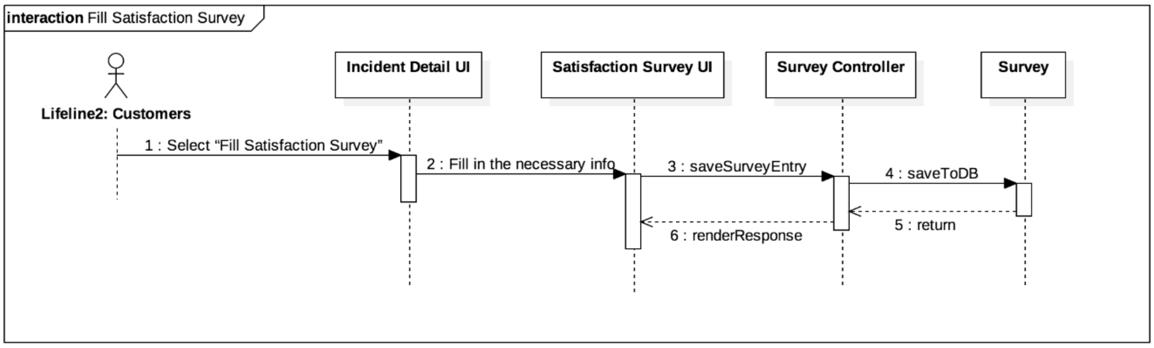

Figure 4.24: Fill Satisfaction Survey Sequence Diagram

Figure 4.25 shows the sequence diagram for setting the priority of an incident. The technician assigns priority to an incident based on the business impact and urgency values.

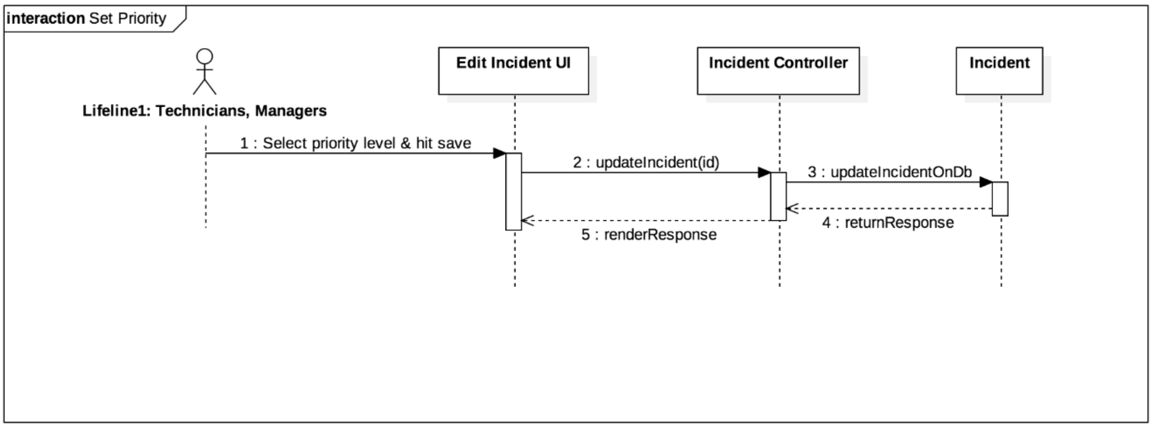

Figure 4.25: Set Priority Sequence Diagram

Figure 4.26 shows the sequence diagram for a technician giving the resolution for an incident.

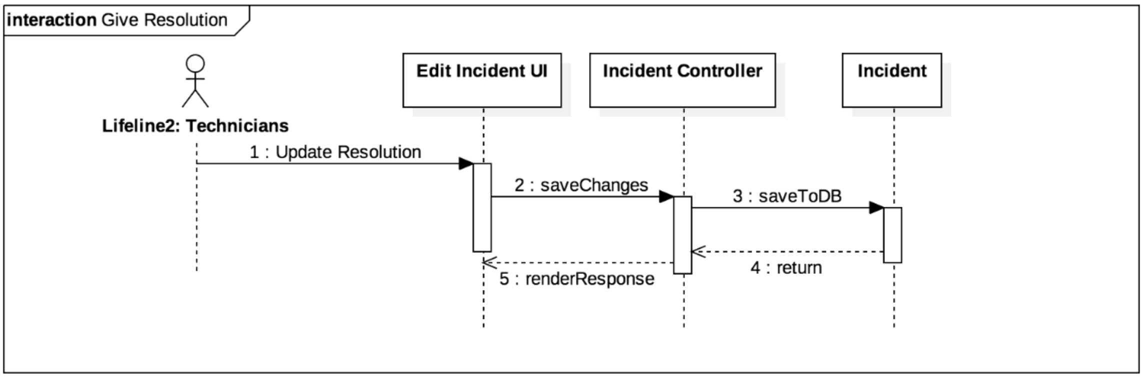

Figure 4.26: Give Resolution Sequence Diagram

Figure 4.27 shows the sequence diagram for viewing the satisfaction survey filled by customers.

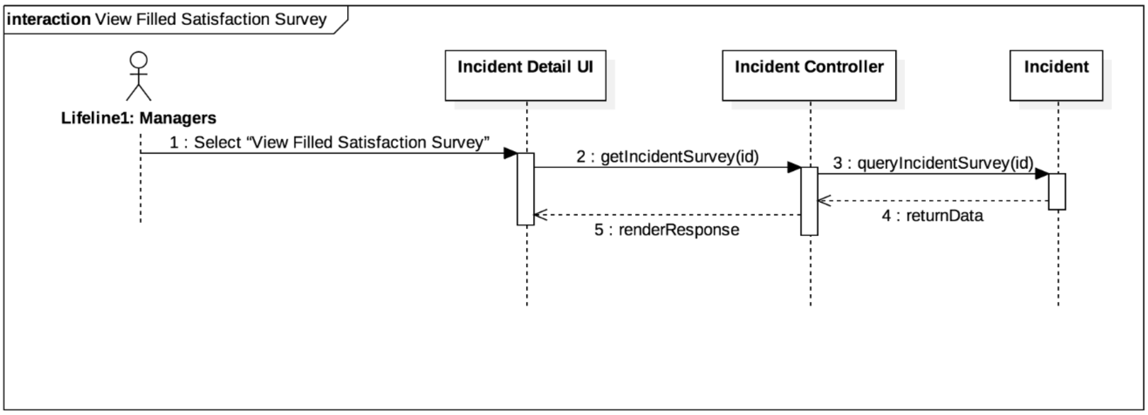

Figure 4.27: View Filled Satisfaction Survey Sequence Diagram

Figure 4.28 shows how the system periodically updates the support team model used during support team identification.

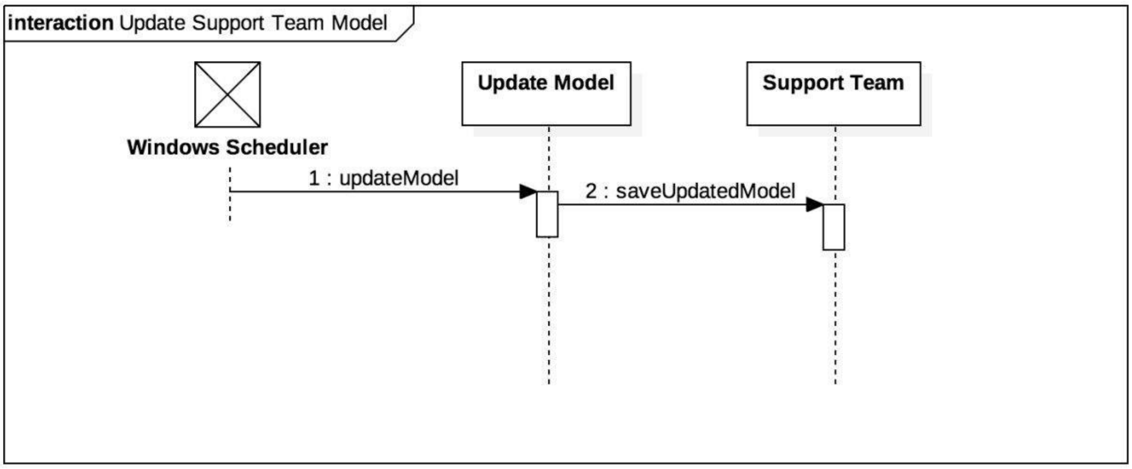

Figure 4.28: Update Support Team Model Sequence Diagram

Problem Management

Figure 4.29 shows how a user registers a problem. A problem is registered when the same incident keeps recurring and root cause analysis is needed.

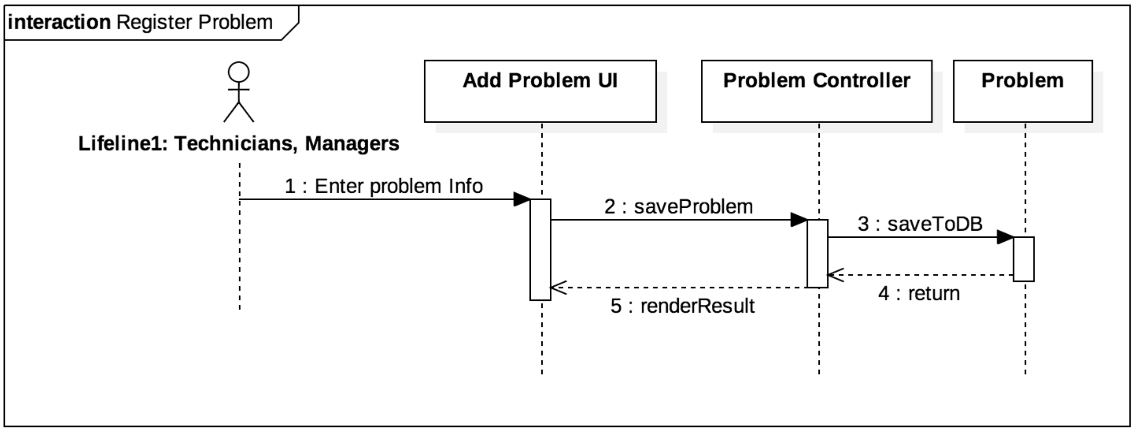

Figure 4.29: Register Problem

Figure 4.30 shows the sequence diagram for displaying registered problems.

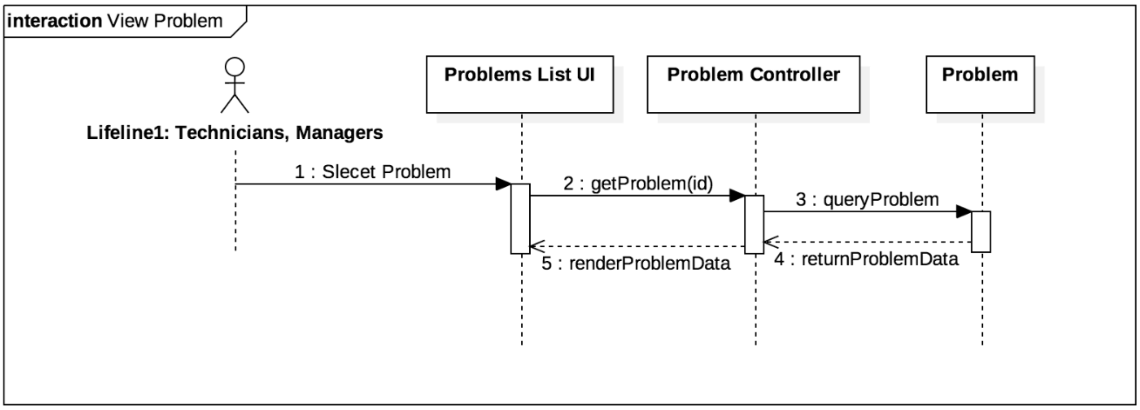

Figure 4.30: View Problem

Figure 4.31 shows the sequence diagram for searching through registered problems.

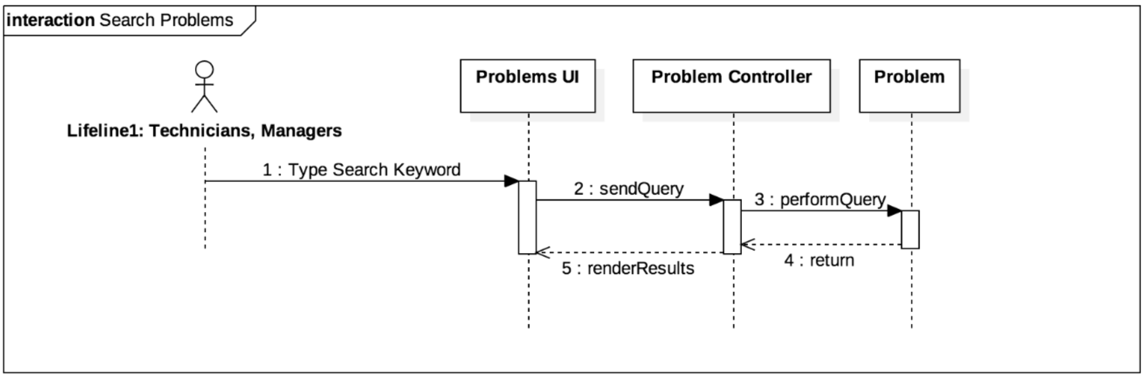

Figure 4.31: Search Problems

Figure 4.32 shows the sequence diagram for closing a problem.

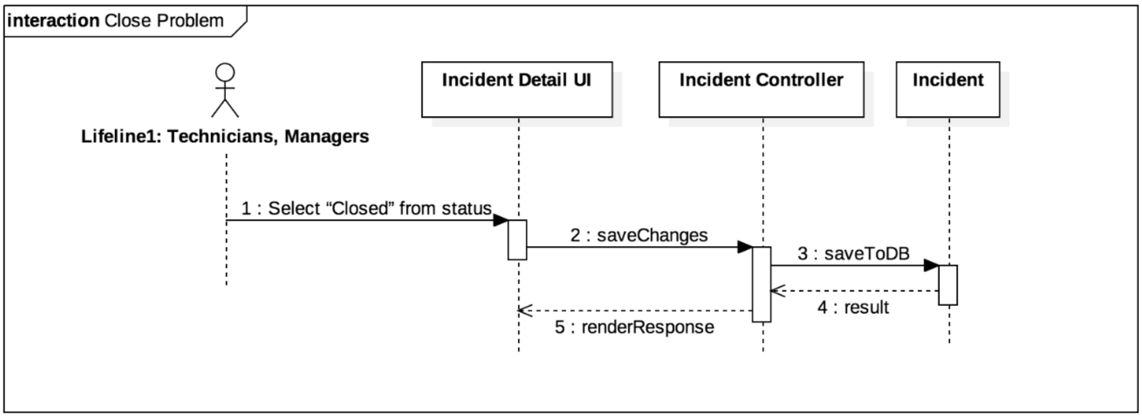

Figure 4.32: Close Problem

Figure 4.33 shows the sequence diagram for assigning tasks. Under one problem, several tasks may be created and assigned to different technicians. A problem can be closed when all these tasks are completed.

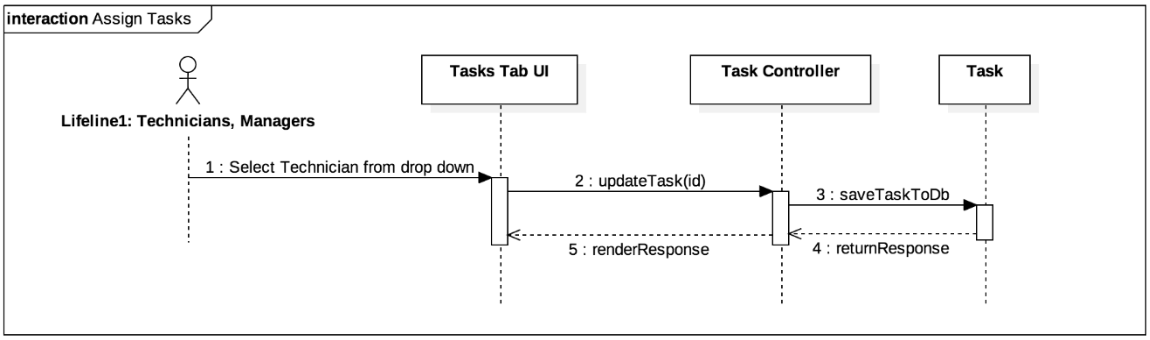

Figure 4.33: Assign Tasks

Figure 4.34 shows the sequence diagram for logging action taken. Technicians that are assigned to different tasks will log the actions they have taken.

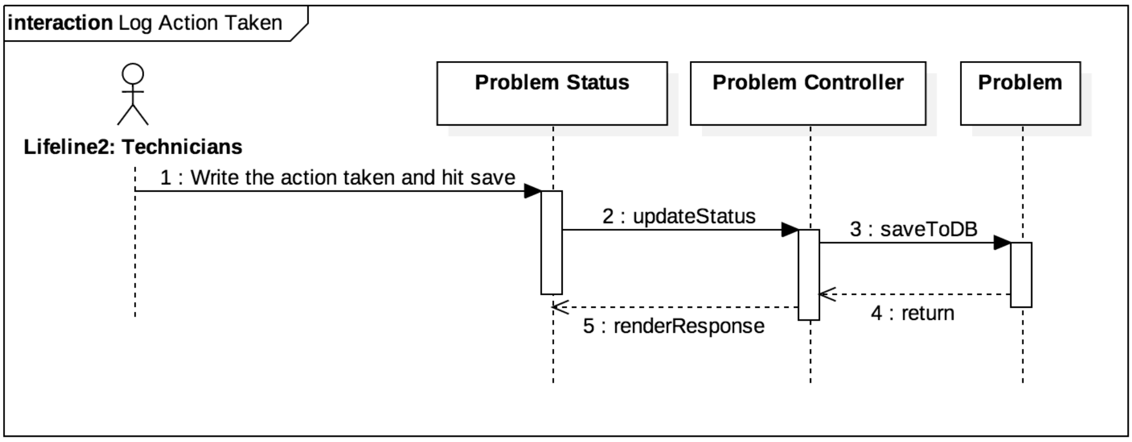

Figure 4.34: Log Action Taken

Change Management

Figure 4.35 shows how a user registers a request for change. An RFC is registered after deciding a change is needed to solve a recurring incident (problem).

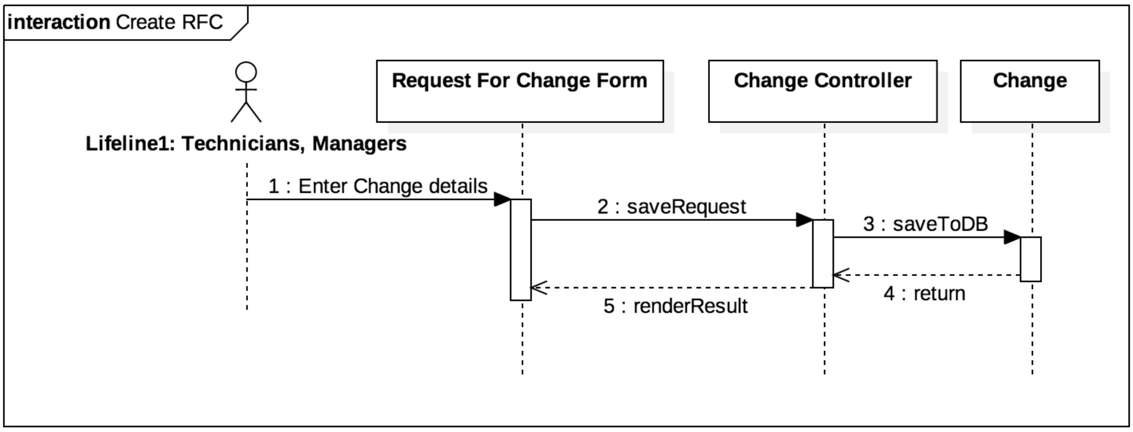

Figure 4.35: Create RFC

Figure 4.36 shows the sequence diagram for displaying registered request for changes.

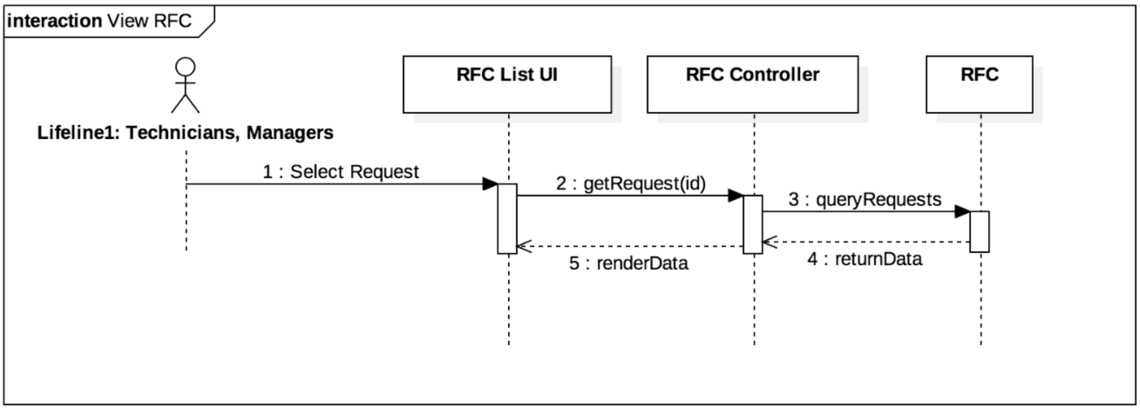

Figure 4.36: View RFC

Figure 4.37 shows the sequence diagram for searching through request for changes.

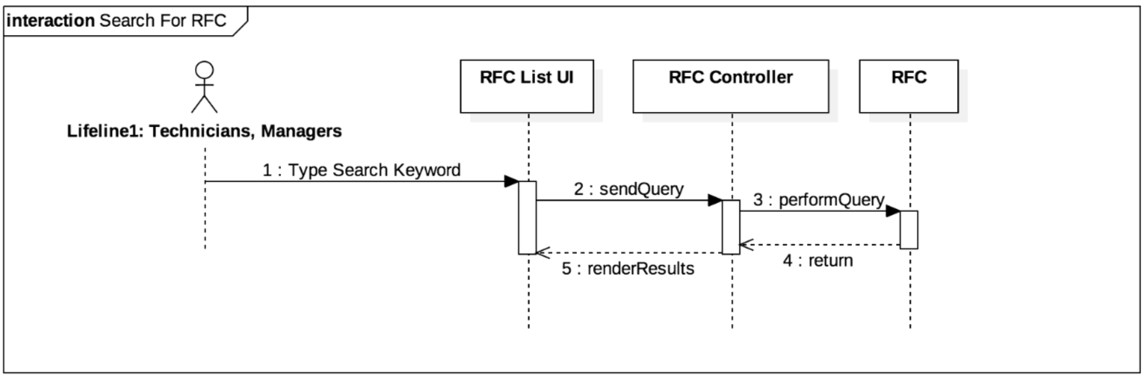

Figure 4.37: Search for RFC

Figure 4.38 shows the sequence diagram for closing opened request for changes.

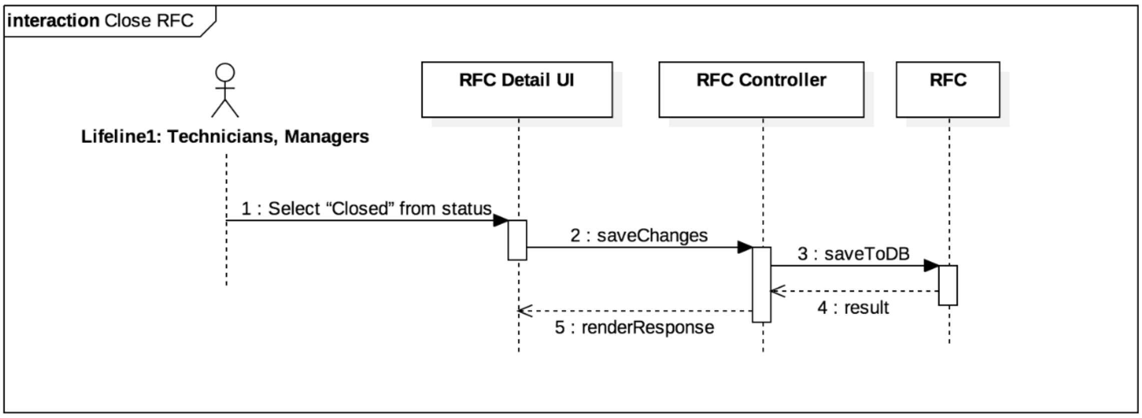

Figure 4.38: Close RFC

Figure 4.39 shows the sequence diagram for creating several tasks under one request for change. These tasks may be assigned to different technicians and may have different schedules.

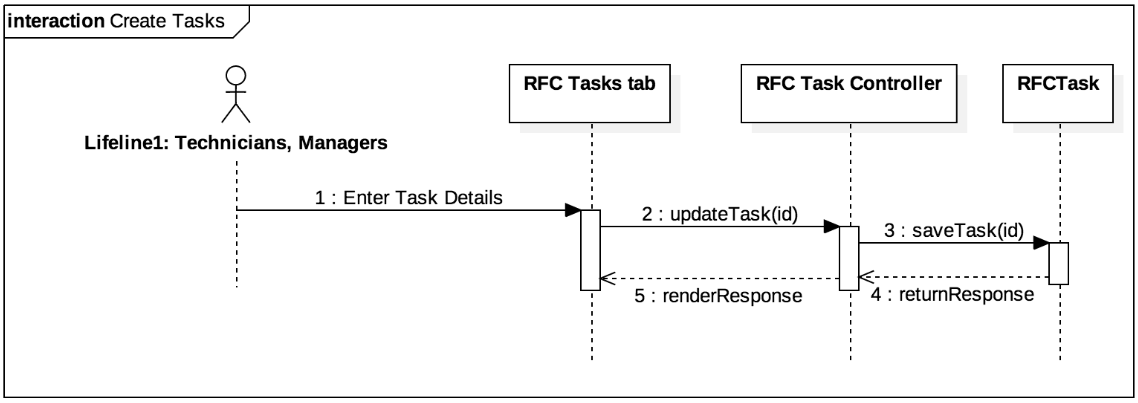

Figure 4.39: Create Tasks

Figure 4.40 shows the sequence diagram for logging action taken for request for changes.

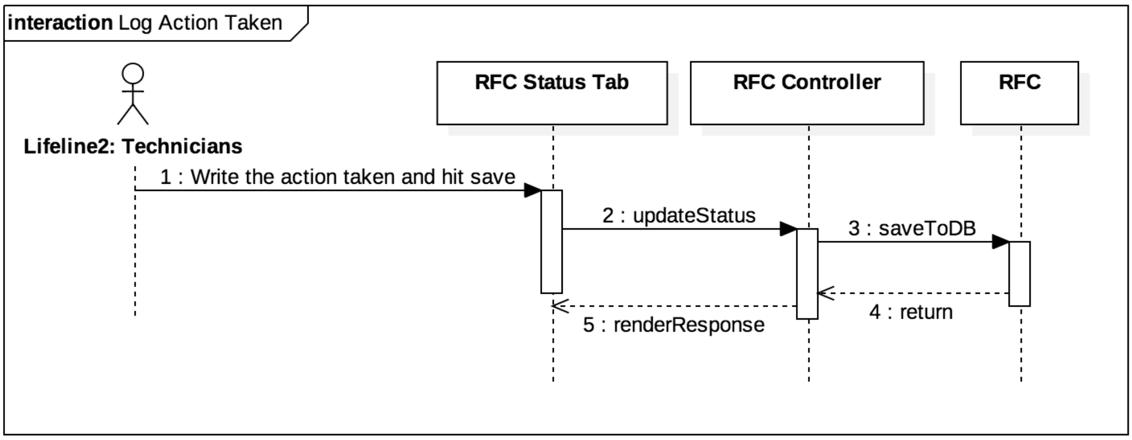

Figure 4.40: Log Action Taken

Figure 4.41 shows the sequence diagram for accepting or rejecting change. The change advisory board is incharge of this action and only approved request for changes can be viewed by technicians.

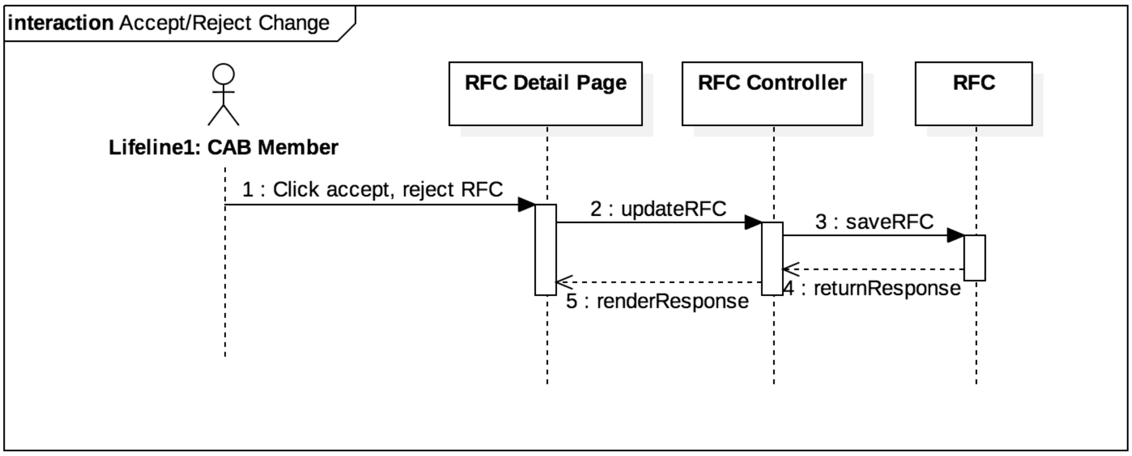

Figure 4.41: Accept/Reject Change

Asset Management

Figure 4.42 shows how a user registers an asset

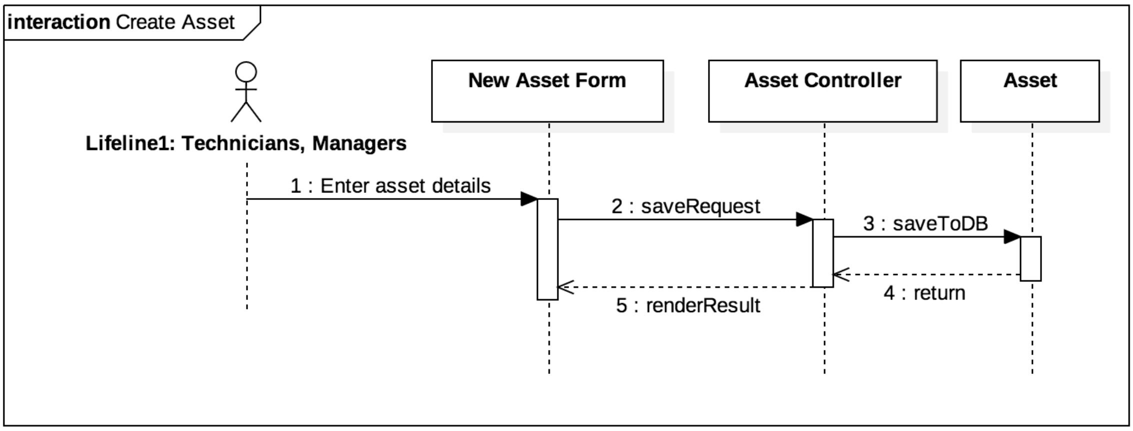

Figure 4.42: Create Asset

Figue 4.43 shows the sequence diagram for displaying registered assets.

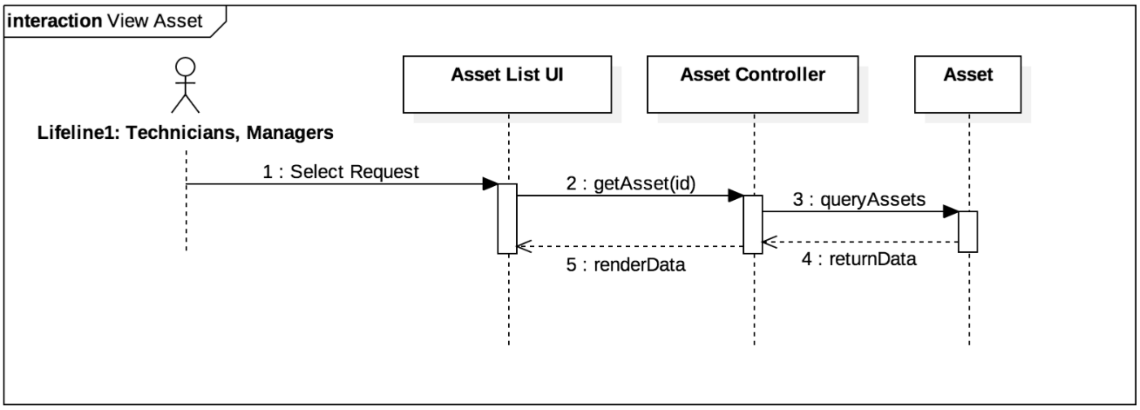

Figure 4.43: View Asset

Figure 4.44 shows the sequence diagram for searching for registered assets.

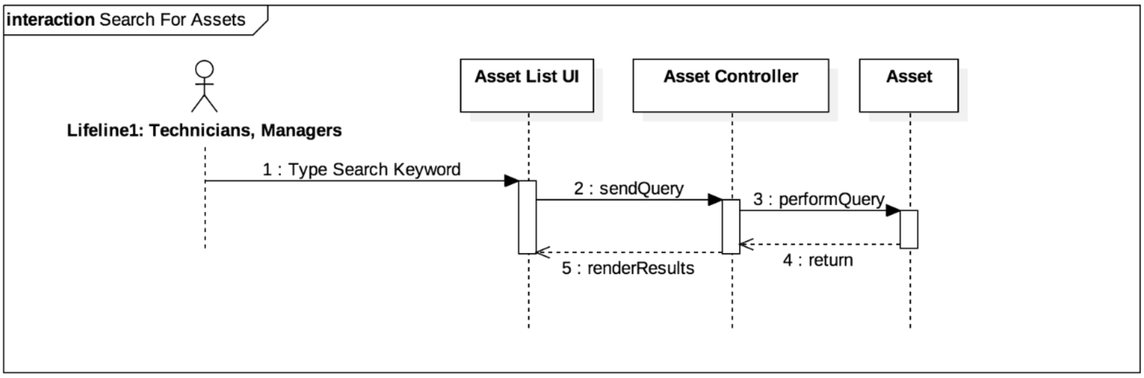

Figure 4.44: Search for Assets

Figure 4.45 shows the sequence diagram for deleting assets.

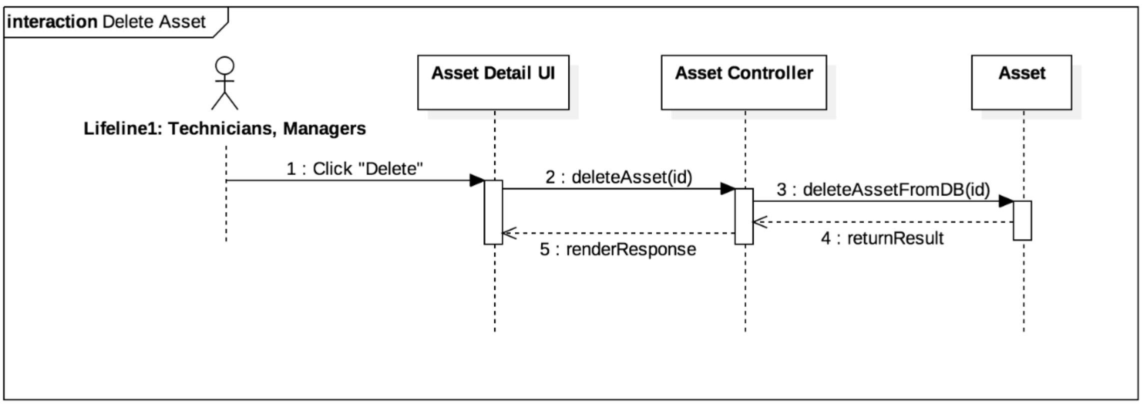

Figure 4.45: Delete Asset

Figure 4.46 shows the sequence diagram for editing assets.

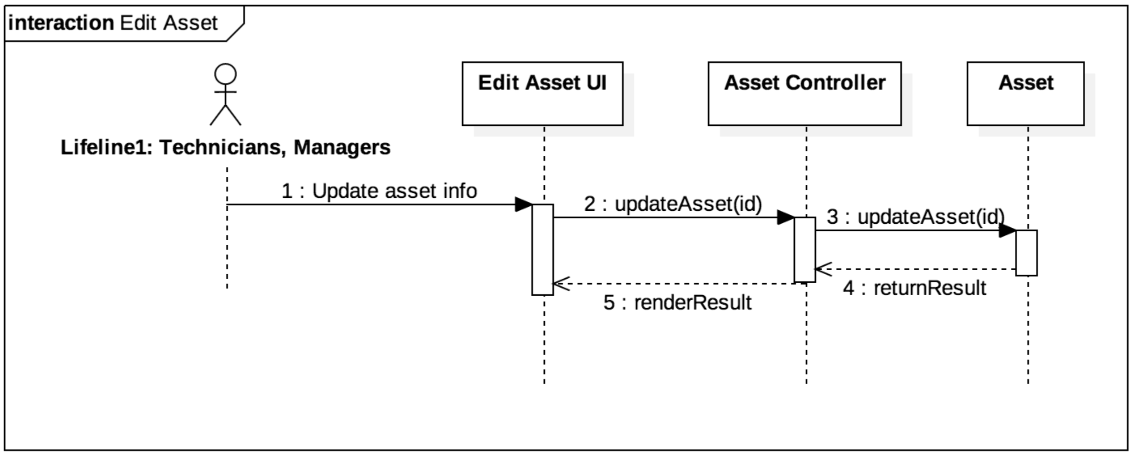

Figure 4.46: Edit Asset

Chapter 5 - System Design

5.1 Introduction

The System Design Document (SDD) is a document that records the results of the system design process and describes how the system will satisfy the Requirements Specification. It interprets the requirements into a description of the architecture, components, interfaces, and data necessary to support the build process.

5.2 Design Goals

The design goals identify the qualities that the system should focus on. Based on the non-functional requirements described in the analysis phase the following design goals are identified:

User Interface: The user interface should be designed using Bootstrap in order to conveniently achieve responsiveness.

Portability: The system should be built by following web browser standards in order to be used by any browser on any platform.

Security

- The system shall use Authentication in order to prevent unauthorized access.
- Role based Access Control should be there to determine the access level of the users of the system.

Robustness: The system shall have be able to handle error conditions gracefully, without failure and shall continue its smooth operation to the end users with them not knowing that an error occurred. This includes a tolerance of invalid files, software defects, and unexpected operating conditions.

Flexibility: The system shall have an MVC architecture that allows modifications and feature additions easily.

5.3 Proposed Software Architecture

The software architecture models the software design. The architectural layering and the subsystem decomposition are used to obtain high cohesion and low coupling, the Hardware/Software mapping indicates the various hardware devices and equipment used in the system and their interaction with the software components is explained. Access Control techniques used in the system will also be included.

5.3.1 System Architecture Diagram

The system architecture diagram shows the layered structure of the system and the different components involved in the designed system. A layered structure is chosen because it makes the system easily maintainable for further modification or fault correction, flexible for integrating additional components and scalable to adjust to changing load.

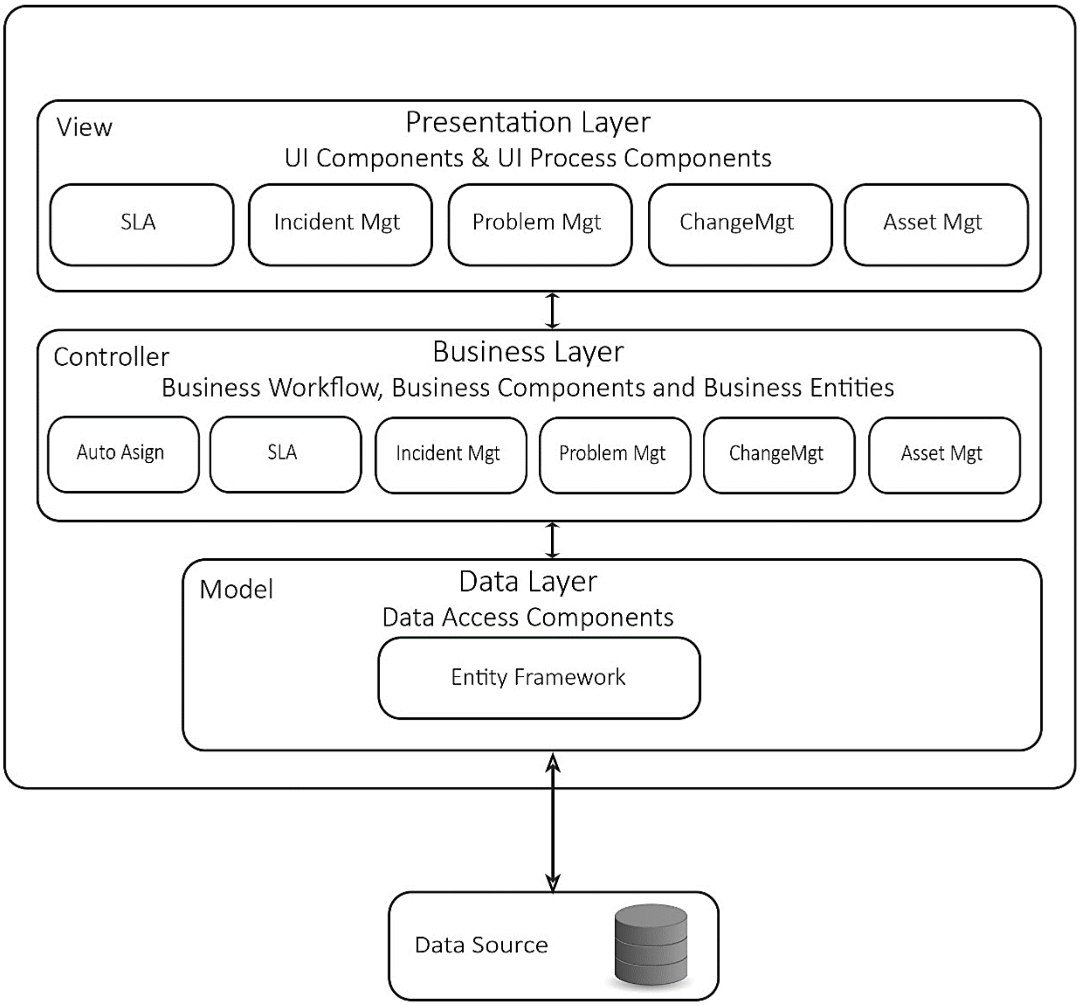

Figure 5.1: System Architecture

This system has a three-layered architecture comprised of presentation, business, and data layers. The presentation layer is found in the View, the business layer is in the Controller and the data layer is in the Model of the system.

Presentation layer contains the user oriented functionality responsible for managing user interaction with the system, and generally consists of components that provide a common bridge into the core business logic encapsulated in the business layer.

In this project, several pages are designed in a responsive manner within the Incident, Problem, Change and Asset Management components. The ASP.NET MVC Framework uses a view engine, which is the component responsible for processing a view in order to generate a response for the browser, this is done in the presentation layer.

Business layer contains any application logic that is concerned with the retrieval, processing, transformation, and management of application data; application of business rules and policies; and ensuring data consistency and validity.

This layer processes incoming requests from each component we have on the presentation layer (Incident, Problem, Change and Asset Management components) and performs operations on the model, and select views to render to the user. It also has the component that interacts with the classifier model in order to automatically assign technicians to incidents.

Data layer provides access to data hosted within the boundaries of the system, and data exposed by other networked systems; perhaps accessed through services. This layer has Data Access Components that abstract the logic required to access the underlying data stores.

For this project, entity framework is used which is a conceptual design of the database. It generates business objects and entities according to the database tables and provides a means for easily performing database related operations.

Each of the above mentioned components will be further explained in the Subsystem Decomposition section.

Weka Interface is a component that will be developed to create an interface between the database and the Weka tool. It will select attributes and their values from the database and uses Weka to execute data mining operations. The result is then stored in the database.

5.4 Sub-System Decomposition

Subsystem decomposition describes the decomposition into subsystems and the responsibilities of each. This is the main product of the system design. Accordingly, we have identified and classified the subsystems.

1 User Interface
2 SLM
3 Incident Management
   3.1. Incident Management
   3.2. Support Team Identification
   3.3. Attendance
   3.4. Auto-assign and Escalation
4 Problem Management
5 Change Management
6 Asset Management
7 Account Management
8 Storage

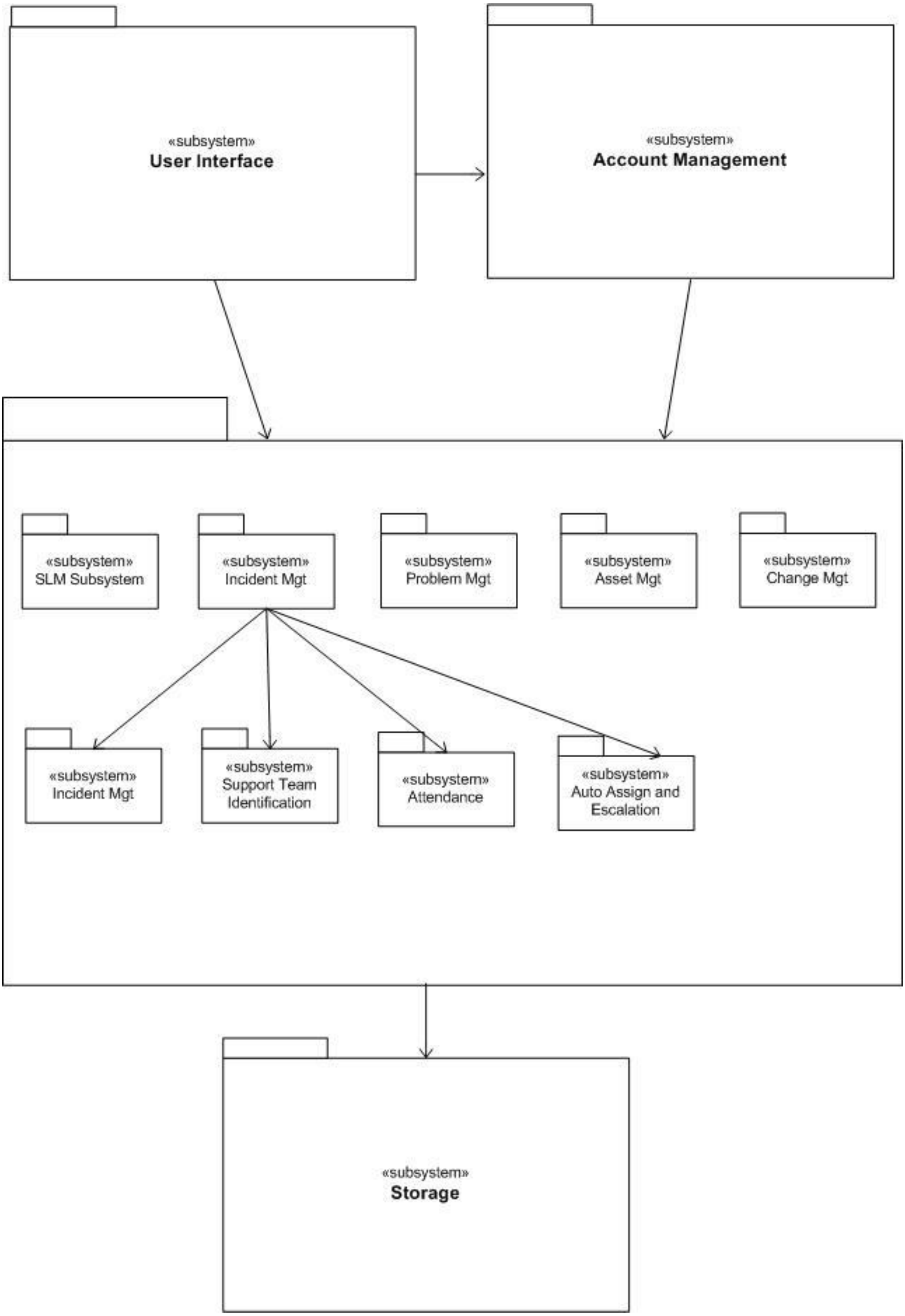

Figure 5.2: Sub-system Decomposition

User Interface: It implements the interface for users to interact with the system based on their privilege

Account Management: Account Management subsystem will be responsible for creating accounts, assigning and denying roles, editing user names and passwords, and deleting accounts.

SLM: This subsystem handles issues related to SLA including defining and updating it.

Incident Management: Users will be able to perform all tasks related to incident management such as registering, viewing, prioritizing, categorizing and resolving incidents.

Support Team Identification: This subsystem is used to identify the support team during incident registration. To be able to do that, it builds a classification model based on previous incidents and updates it periodically to enable learning from new incidents. When a user enters a free text description of the incident, this subsystem performs data cleaning and uses the classification model created to identify the support team that is responsible for the resolution.

Attendance: The Attendance Subsystem is used to import data daily from ET's attendance system. It gets the data in an xml format and uses employee id number to match it with the technicians registered on the service desk. This will allow the system to check availability of technicians before assigning an incident to them.

Auto-assign: This is used to allocate incident resolution tasks to IT support staff. The default task allocation mechanism is automatic, where the system picks IT support staff based on availability and current load information of the support team members. If necessary, technicians and managers are allowed to assign tasks manually as well.

Escalation: There are different levels to solve a problem based on the type of incident and corresponding time limit to solve the problem. If the problem fails to be solved by a technician on a certain level, based on the agreement, it will be passed to a technician on the next level. The SLA detail will be set by the manager or a technician with access.

Problem Management: Users will be able to perform all tasks related to problem management such as registering, viewing, prioritizing, categorizing and resolving problems.

Change Management: This subsystem enables users to perform all tasks related to change management such as registering, viewing, prioritizing, categorizing and implementing changes. Creating CABs (Change Advisory Boards) and approval/denial of changes are also handled by this subsystem.

Asset Management: Technicians and Managers will be able to perform all tasks related to Asset management such as registering, viewing and updating their status.

Database: This is where data from the various components is stored. It allows inserting, reading, updating and deleting data within the system. Each class from within the Subsystems described above are responsible for inserting, updating and modifying their own data by utilizing the corresponding methods belonging to the Database Class within the Database Subsystem.

5.5 Hardware/Software Mapping

The system will have three machines; the User device will provide access to the user interface through a browser, the Server Machine will run the application logic while the database server handles the storage and provide all data necessary for the Service Desk services.

For the realization of the Service Desk, .Net framework has been selected which is a collection of interfaces and standards developed by Microsoft that enables the development of portable web-based systems in several languages.

Microsoft SQL Server 2014 is chosen as the database, and it has many features which are ideal for the .Net based application. Support for Multiple Platforms, Integration with Microsoft .Net Enterprise Servers, scalability, replication, centralized Management and reliability are important qualities for this project.

Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed. The deployment diagram for this system can be seen in Figure 5.2 below.

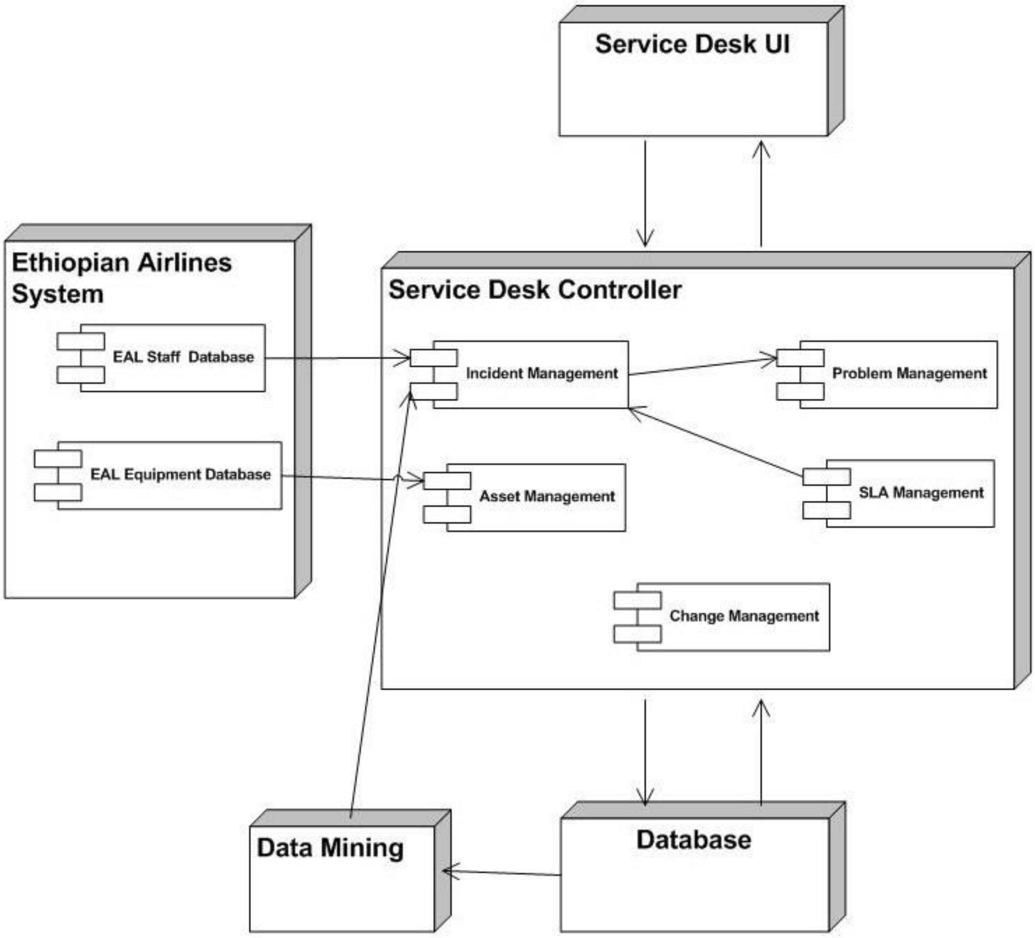

Figure 5.3: Software/Hardware Mapping

5.6 Persistent Data Management

Persistent data management deals with how persistent data of the system are stored and managed. Several Entity objects were identified as persistent data objects for all the components of the system. Crow's Foot notation has been used which represents the entities as boxes, relationships with lines and cardinality with chicken feet at the end of the lines.

The system will use the Microsoft SQL database engine for storing data. It is very reliable and used by some of the biggest corporations in the world.

Figure 5.4 shows the ER Diagram for Incident Management.

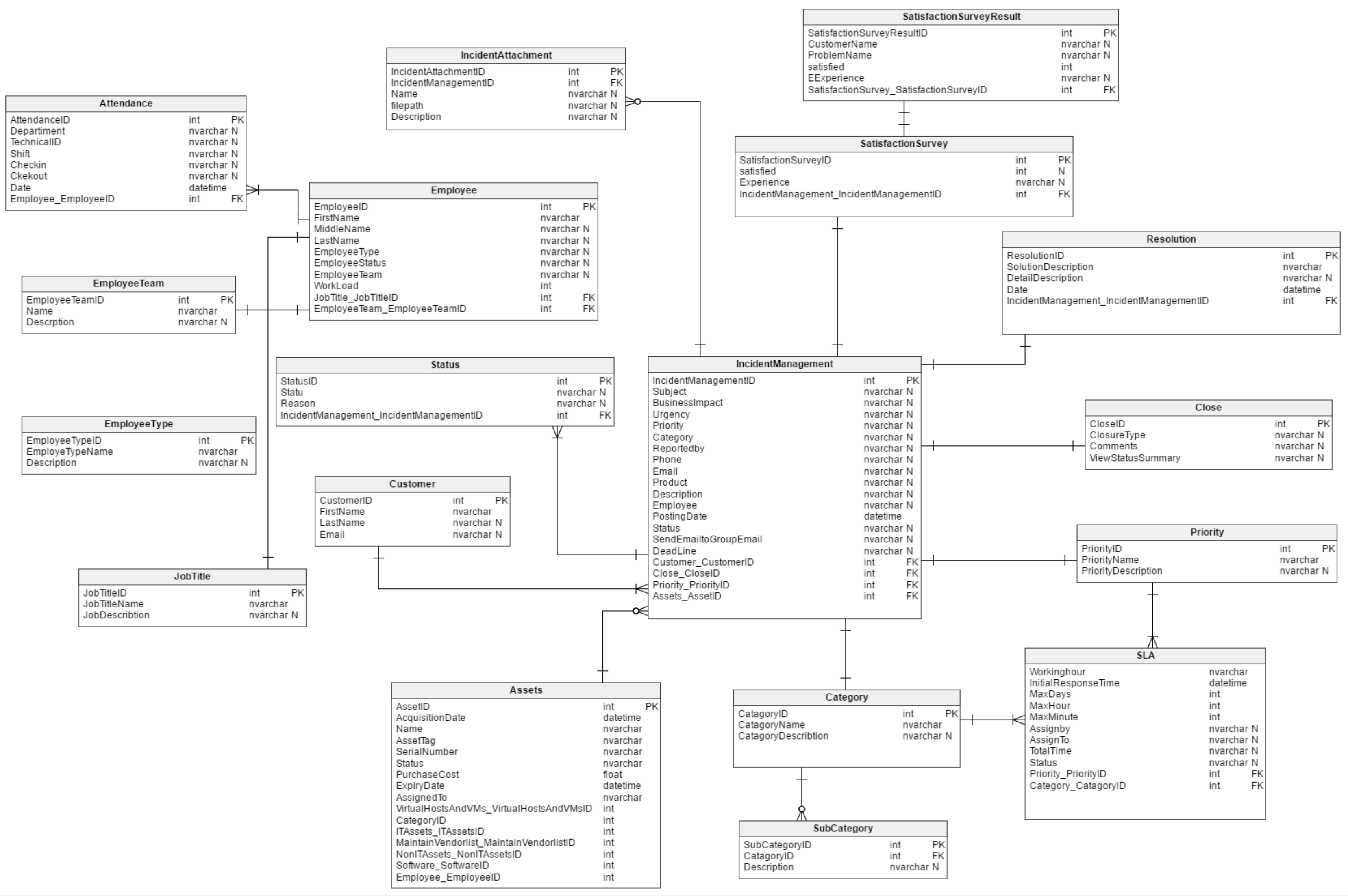

Figure 5.4: Persistent Diagram for Incident Management

Figure 5.5 shows the ER Diagram for Problem Management.

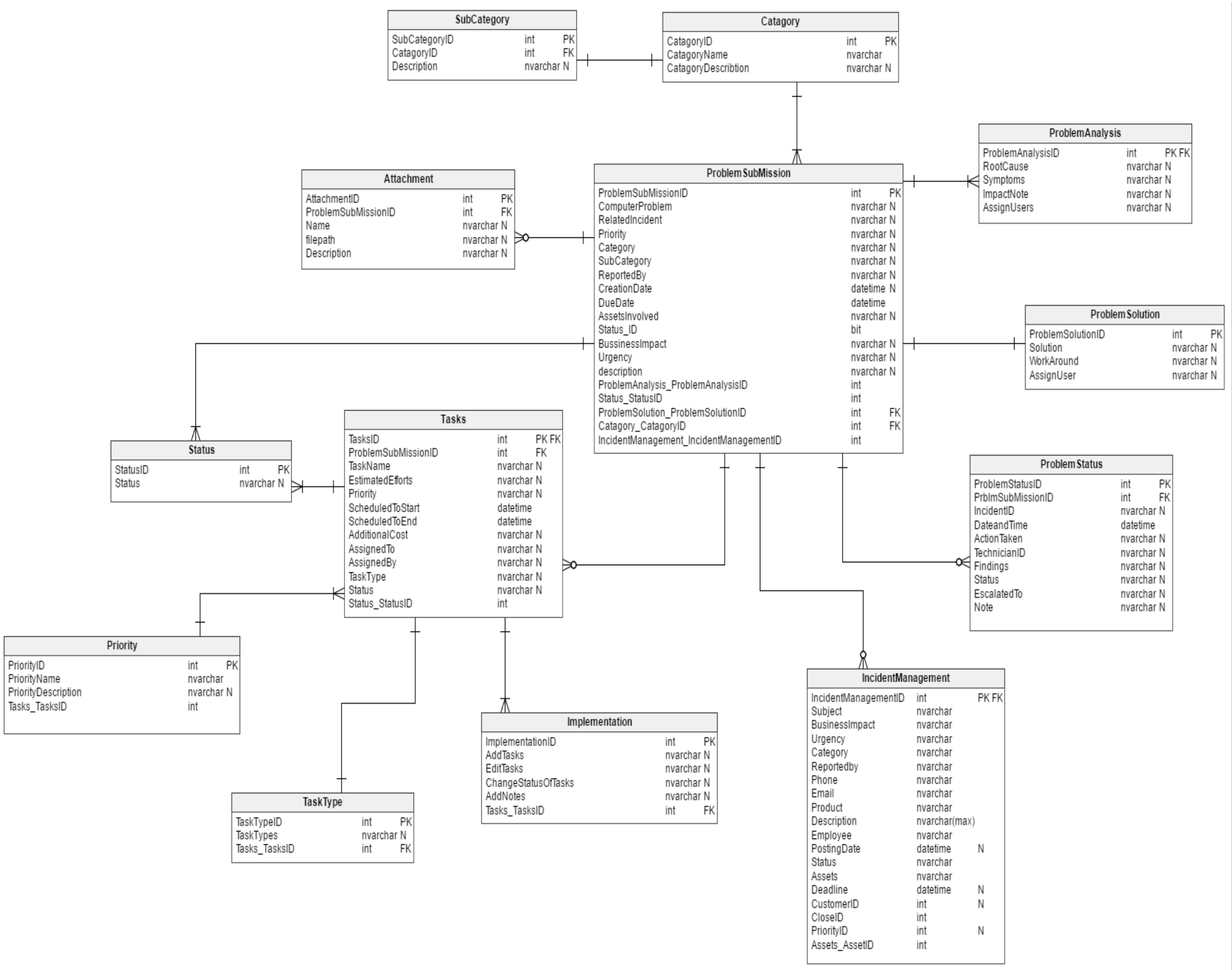

Figure 5.5: Persistent Diagram for Problem Management

Figure 5.6 shows the ER Diagram for Change Management.

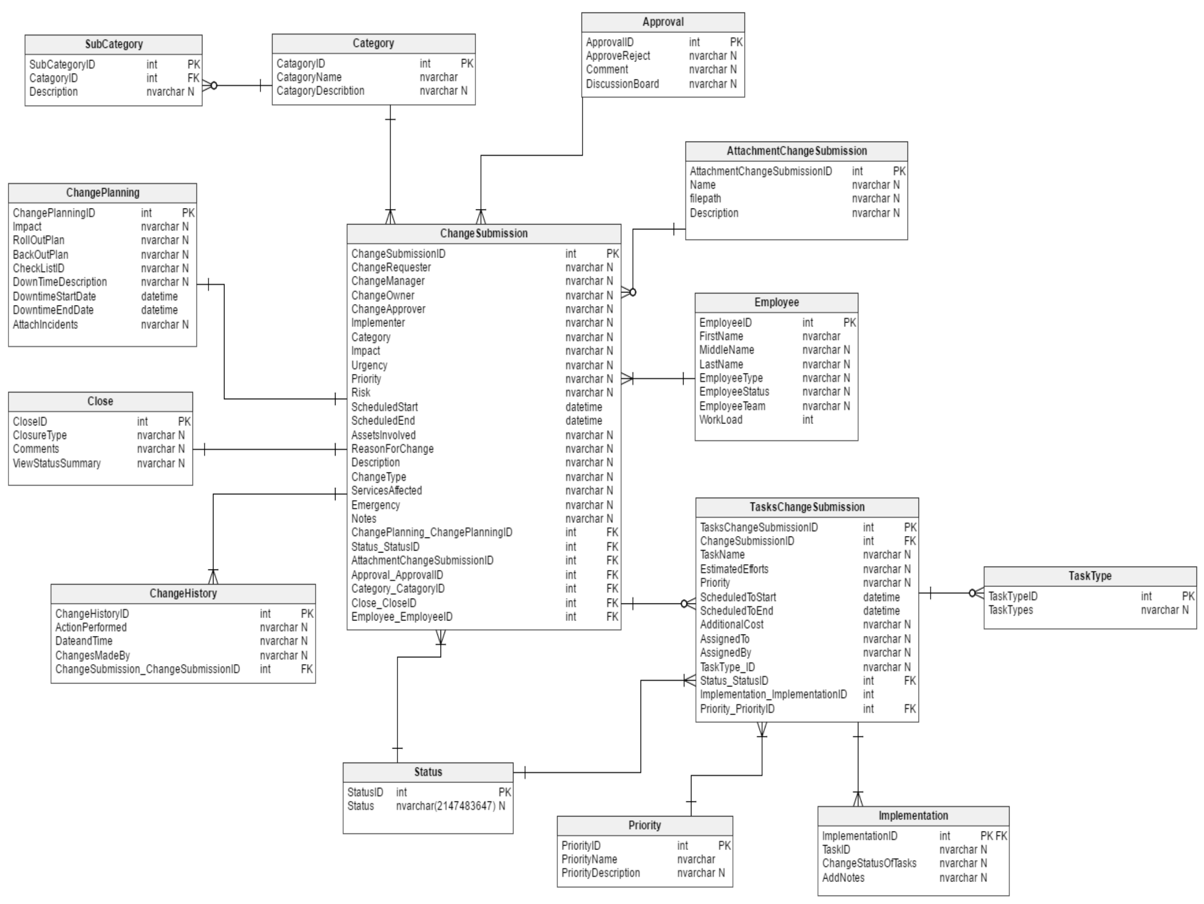

Figure 5.6: Persistent Diagram for Change Management

Figure 5.7 shows the ER Diagram for Asset Management.

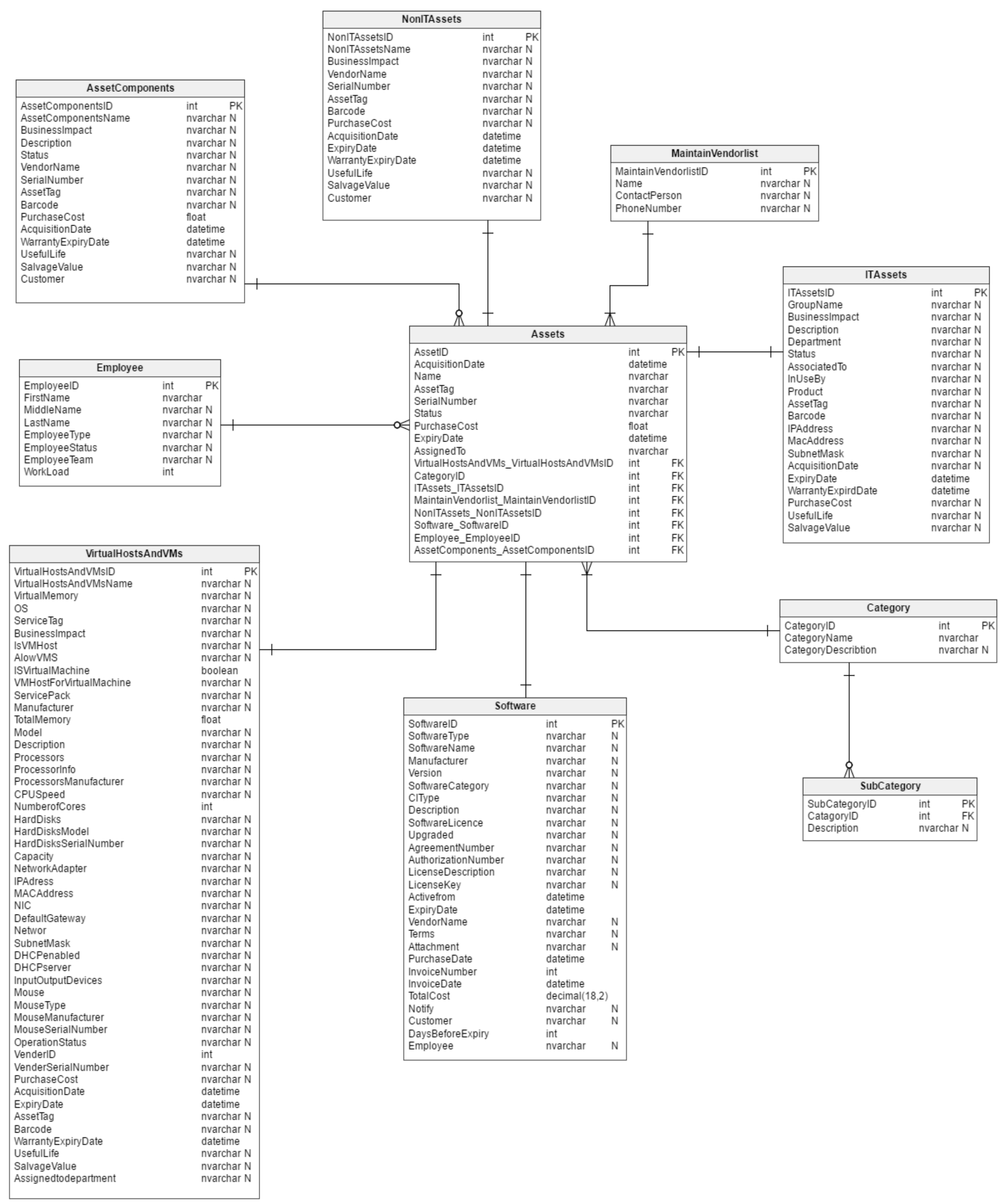

Figure 5.7: Persistent Diagram for Asset Management

Chapter 6 - Implementation

This chapter explains the developed system and its features along with screenshots. The development environment that was used to implement the system is also described.

6.1 Development Environment and Tools

Development Environment

System development as well as data mining using Weka was done in the following environment:

- Lenovo G50-80, Intel® Core™ i5 – 5200U CPU Processor 2.20GHz
- Installed Memory (RAM) – 8GB
- Windows 10 Home 64-bit Operating System

Tools and Techniques

- Visual Studio 2013 for developing the system
- Weka for data mining techniques (Preprocessing and Modeling)
- Eclipse for implementing Weka functions
- Microsoft Excel for data cleaning
- Microsoft Word for documentation
- Microsoft Visio for UML diagrams

Infrastructure Requirements

To implement the system the following resources were required:

- Web server (IIS).
- Microsoft.NET Framework 4.5
- MSSQL Server 2014
- Server computer with at least 4 GB RAM.
- Any device with a browser

6.2 System Implementation

The system implementation was divided into two phases: First is developing a system that has Incident Management, SLA configuration, Problem Management, Change Management and Asset Management that follow ITIL standards. The second is applying data mining techniques to data collected from existing system for better accuracy in assigning support team for incidents. Each will be described below.

When any user tries to access the system, the first thing he/she sees is a page as in Figure 6.1 where they can search through previously solved incidents and a Register/Login link. If they find a related issue that can help them solve their incident then they can read the resolution from this page and fix their own issue.

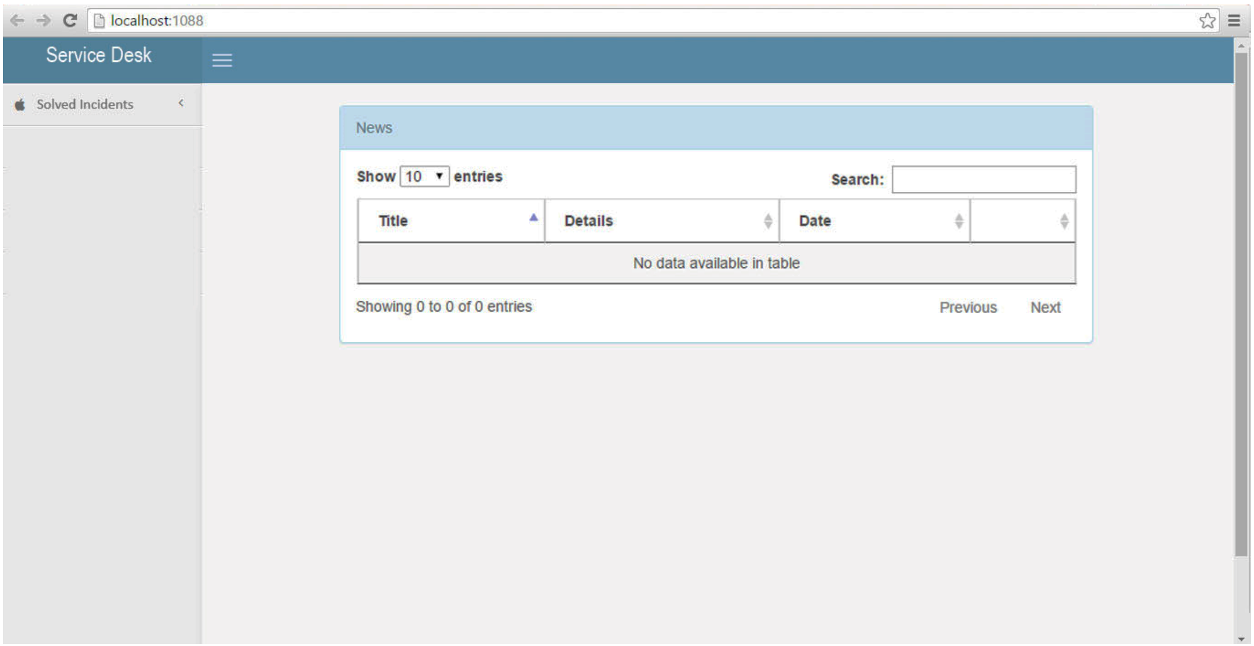

Figure 6.1: Home Page

For any action other than a simple search of incidents, a user will have to use their credentials to login to the system. Users of the system are Customers, technicians (with different levels), managers and CABs. Therefore, after logging in each of these users will see different pages according to their privileges.

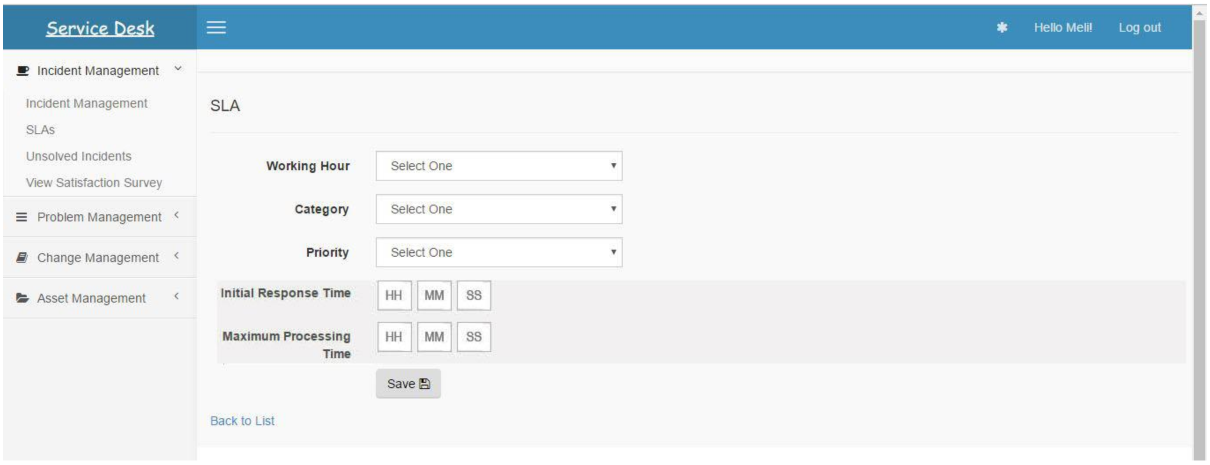

Figure 6.2: SLA Management Page

The SLA page in Figure 6.2 allows users to set the time for Initial Response Time (IRT) and Maximum Processing Time (MPT) in terms of days, hours and minutes. Each incident category will have an SLA to ensure that they are being handled according to the agreement. After an incident is reported, the system checks its category and assigns a deadline to it based on the MPT. When the MPT is reached, the incident is escalated to a higher level technician.

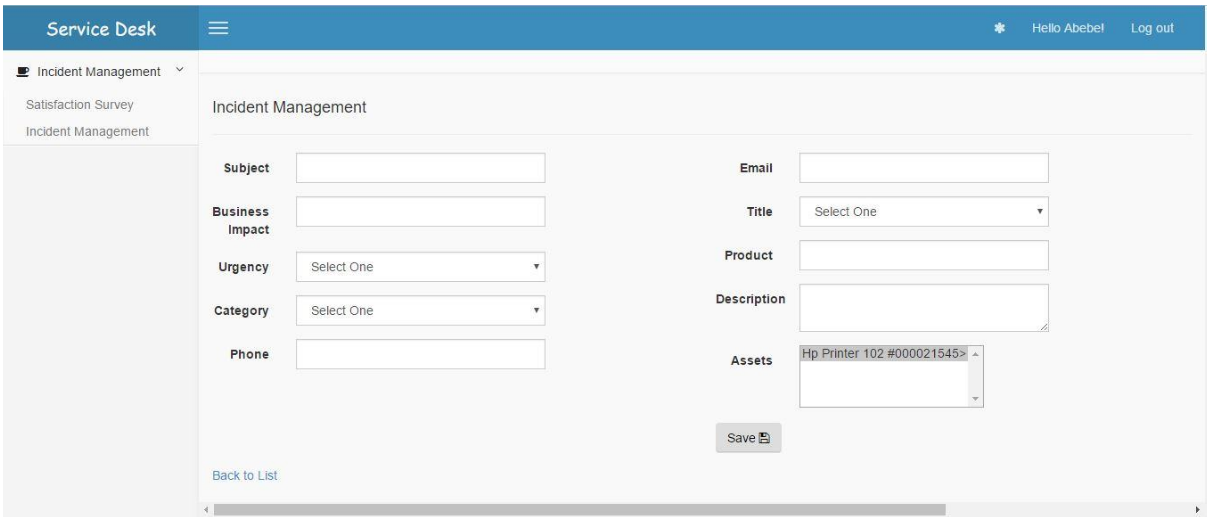

Figure 6.3: Incident Creation Page

Figure 6.3 shows a page where a customer can enter the information regarding the incident they encountered. After the customer submits an incident, the system automatically finds a support team based on the model created from the classification algorithm. After that it checks for available technicians and workload of each team member within that support team and assigns a technician to work on the incident. Priority is also assigned automatically based on the values of “Business Impact” and “Urgency” the users enter. This can later be changed by the technician.

If the technician doesn’t complete the task within the specified amount of time in the SLA, the system automatically escalates it to a technician on a higher level.

The customer can keep checking the status of the incident by logging in to the system. When an incident is resolved, the customer can submit a satisfaction survey as displayed in Figure 6.4 rating the performance of the technician on a scale of 1-10 and write a review as well. This can later be viewed by the manager and can give insight about the performance of the technicians.

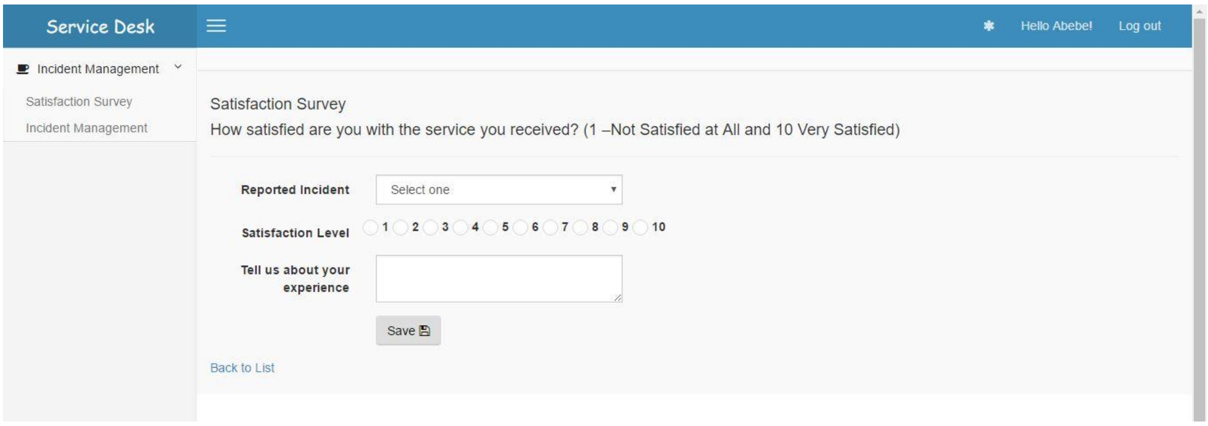

Figure 6.4: Satisfaction Survey Page

When technicians are assigned to incidents, he/she will be able to see them as in Fig. below. They can then edit fields such as priority, escalate it to a technician on a higher level or give their resolution after solving.

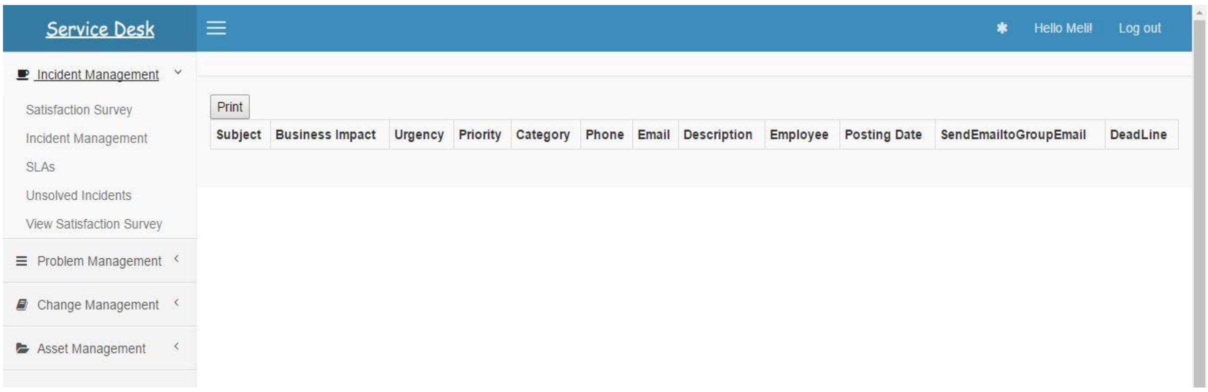

Figure 6.5: View Incidents

To get a list of technicians that are available in the company, it was necessary to find a way to get data from their attendance system. To accomplish that, it was decided to implement an xml file importer that imports data from their system once a day (in the morning, a few minutes past their morning check-in hour). Figure 6.5 shows a screenshot of the sample xml code that gets imported.

```xml
<Attendance>
<Department>Cargo System Support</Department>
<TechnicianID>ETIT00012</TechnicianID>
<Name> Solomon Abebe</Name>
<Shift>Normal Shift</Shift>
<Checkin>9/18/2016  2:30</Checkin>
<Checkout>9/18/2016  11:30</Checkout>
<Date>9/18/2016</Date>
</Attendance>
```

Figure 6.6: Screenshot of sample xml file

After importing the file, the attendance list is displayed in the system as in Figure 6.6.

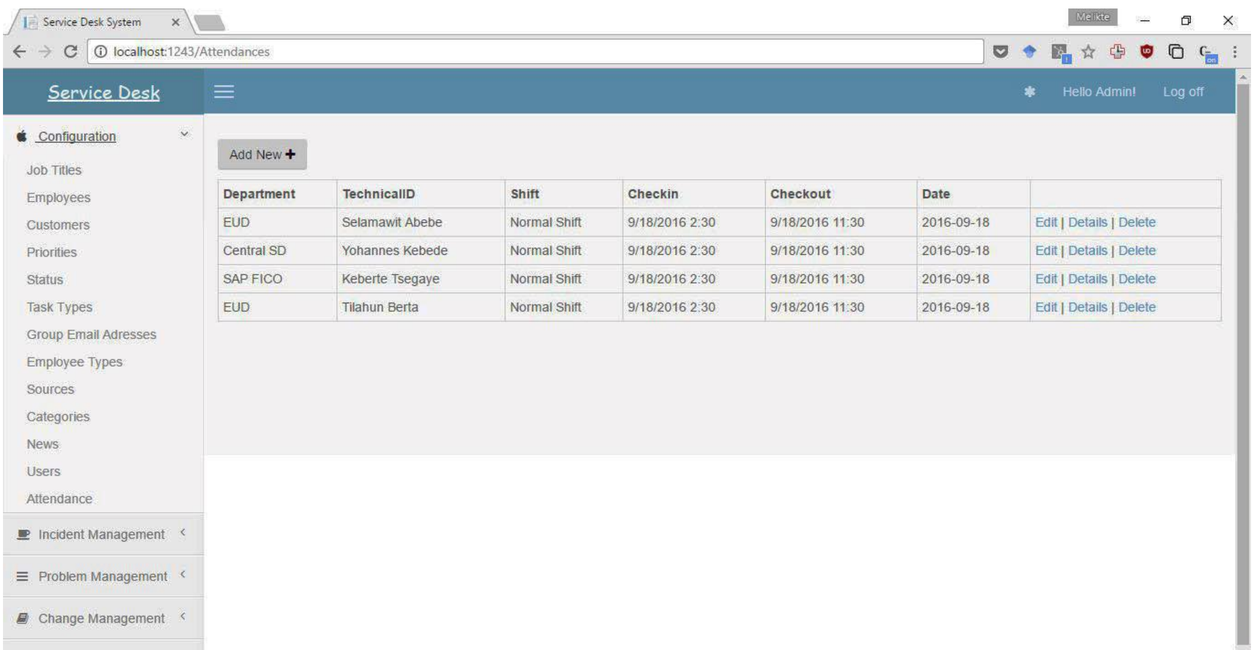

Figure 6.7: Attendance Page

Problem Management

When a problem is submitted, the technician and/or manager reporting it will enter the basic information and save it. After saving it, the user has more options to add detailed information as well as assigning it to a responsible technician. Individual tasks can be created within a problem and assigned to different technicians with scheduled start and end time. In addition to that technicians can log their findings and status on a separate tab for the responsible body to track.

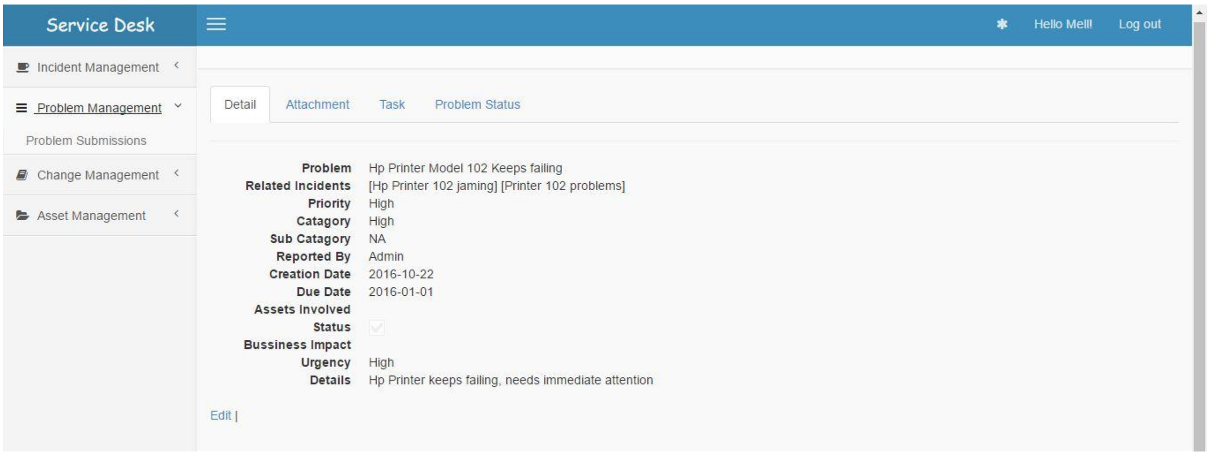

Figure 6.8: Problem Management Page

Technicians can view a list of problems assigned to them and Managers can view a list of problems that are currently created in order to see progress and take action.

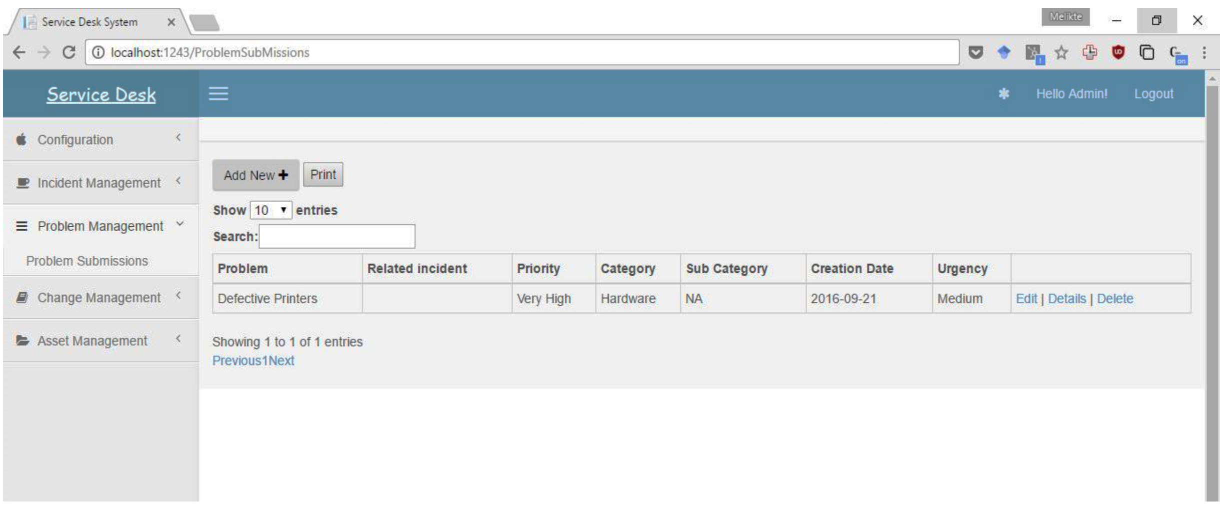

Figure 6.9: View Problems Page

Change Management

A technician can initiate change and submit a form with the descriptions. The CABs will then be able to view a list with submitted RFCs and view the details to either approve or reject. If approved, individual tasks can be assigned to different technicians to implement the change.

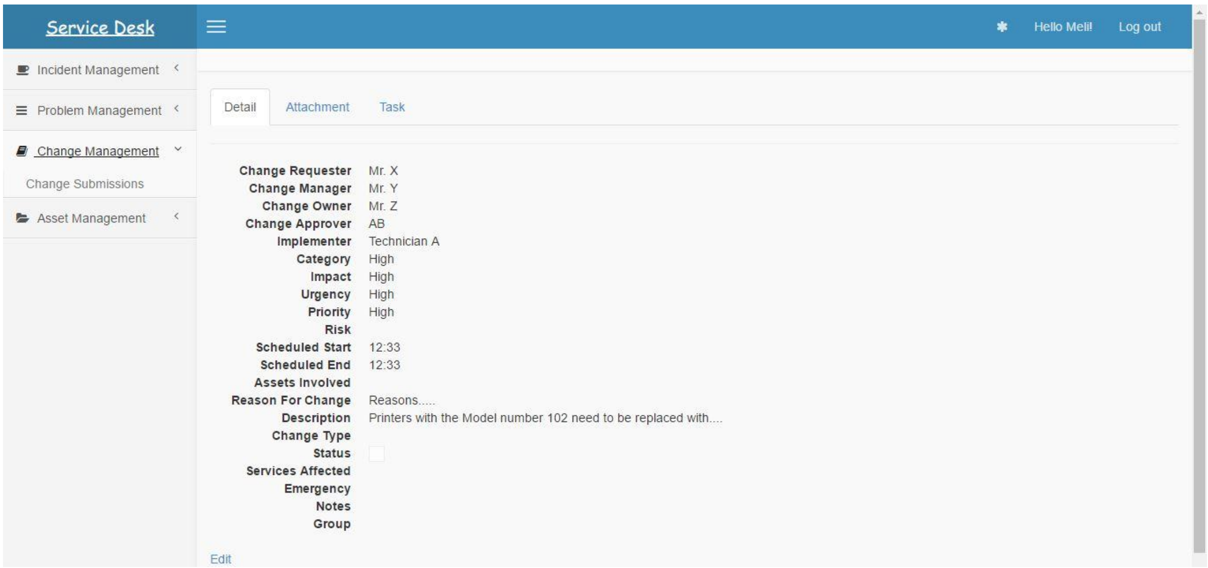

Figure 6.10 : Change Management Page

Technicians can view a list of changes assigned to them and Managers can view a list of changes that are currently going on in order to see progress and take action. Figure 6.11 shows the View Changes page when logged in as a CAB.

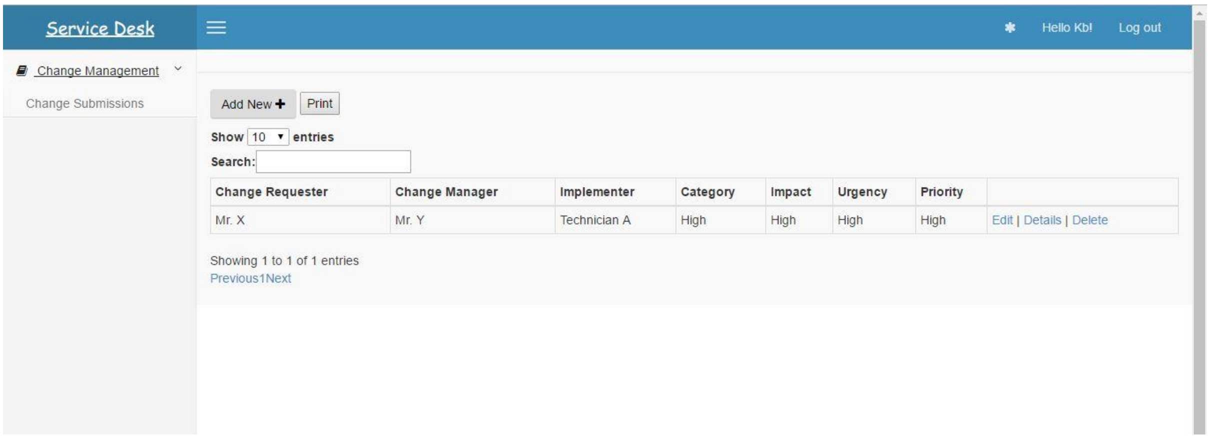

Figure 6.11: View Changes Page

Assigning Support Team

Currently, support teams are being assigned by the person at the central service desk after looking at their description. This method led to incidents being assigned to the wrong support teams. Therefore, for the system to be able to correctly classify incidents from the free text descriptions entered by users, we applied data mining techniques to the data received from ET.

The different data mining processes are applied to the data to help the system learn from over 9,000 descriptions. The process for assigning support team starts with Data Understanding and Preprocessing. Steps taken are described as follows.

Data Understanding

Understanding the data is useful in identifying which models can be applied to get results, finding useful patterns and figuring out which attributes can be unnecessary. Experts in the domain were interviewed to clarify the meaning of each attribute in the data received from their system. The following table displays the attributes and their description.

Table 6.1 Data Description

Id	Automatically generated id (eg. 1000000039, 1000000040)
Description	Description of the incident (eg. E-mail configuration issues, Damaged UTP cable)
Priority	Priority given to each incident (eg. 1: Very High, 2: High)
Status	Status of the incident (eg. In progress, Forwarded)
Created On	Date the incident was reported (eg. 08.12.2015
Message Processor	Full Name of the person working to fix the incident.
Support Team	Team responsible for fixing the incident (eg. 12_SAP_HCM)
Technician Level	The level of the technician that handled the incident. (eg. Incident - 04_Desktop Support Services - Remote Support - Data Migration L1)
Central SD Team Member	Full name of the person that initially accepted the incident message.
Team Supervisor	Full name of the person that supervises the incident resolving.
Reported By	Full name of the person that reported the incident.

Data Cleaning

Missing values of the support team column were filled by referring to the technician level column. In few cases it wasn't clear enough to infer from the column in which case the entire row was removed.

The data included records with "test" in the incident description which were recorded during the testing phase of the system. These rows have also been removed.

Some values contained characters such as "/" and "|" which was noise that had no relevance and was also cleaned.

Rows with descriptions such as "Please help" or "problem" have been removed because they don't contain useful information.

Abbreviations such as "pwd" were changed into "password" because the algorithm will consider them as two different words.

Data Transformation

After the Data Cleaning process, the rest of the data preprocessing was done on Weka. Weka has powerful preprocessing tools and settings that are able to perform very well for the requirements of this project.

The data was loaded on Weka on the first tab (Preprocessing tab) and a list of attributes are displayed with information about each attribute such as count, number of unique values, graphs etc. Here, attributes that are deemed unnecessary can easily be removed by selecting the attributes and clicking the "Remove" button. For this project, "Support Team" and "Description" are selected and the rest are removed.

The goal is to construct a model so that when a user reports an incident, the system will identify the type of incident from the description and assign it to the appropriate support team. To achieve this, the system needs to understand the words used in the description and know which team they belong to.

After the two attributes are selected "String to Word Vector" filter is applied to the description data to change it to numerical data. This is done because classifiers usually cannot process string data well. The values of the description attribute now became attributes themselves and are listed in the attributes column with frequency count as their values. String to vector filter has many settings that are essential for this process.

The ones that were used are: “Lower Case Tokens” which changes every word into lower case so that all words with different capitalization are considered as the same word. “Stop Words” - a list of stop words were downloaded and words like “problem”, “working”, “request” etc. were added to it. These words are seen many times in various descriptions but they provide no value since they cannot help in differentiating the various incidents.

Next was the Word Tokenizer which helped in dividing words that were hyphenated, joined by characters such as “&”, “-” etc.

After applying “String to Vector” with the following settings, we get a list of attributes with words used in the description; the number of attributes became 3652. However, not all of these attributes are useful which means data reduction needs to be done.

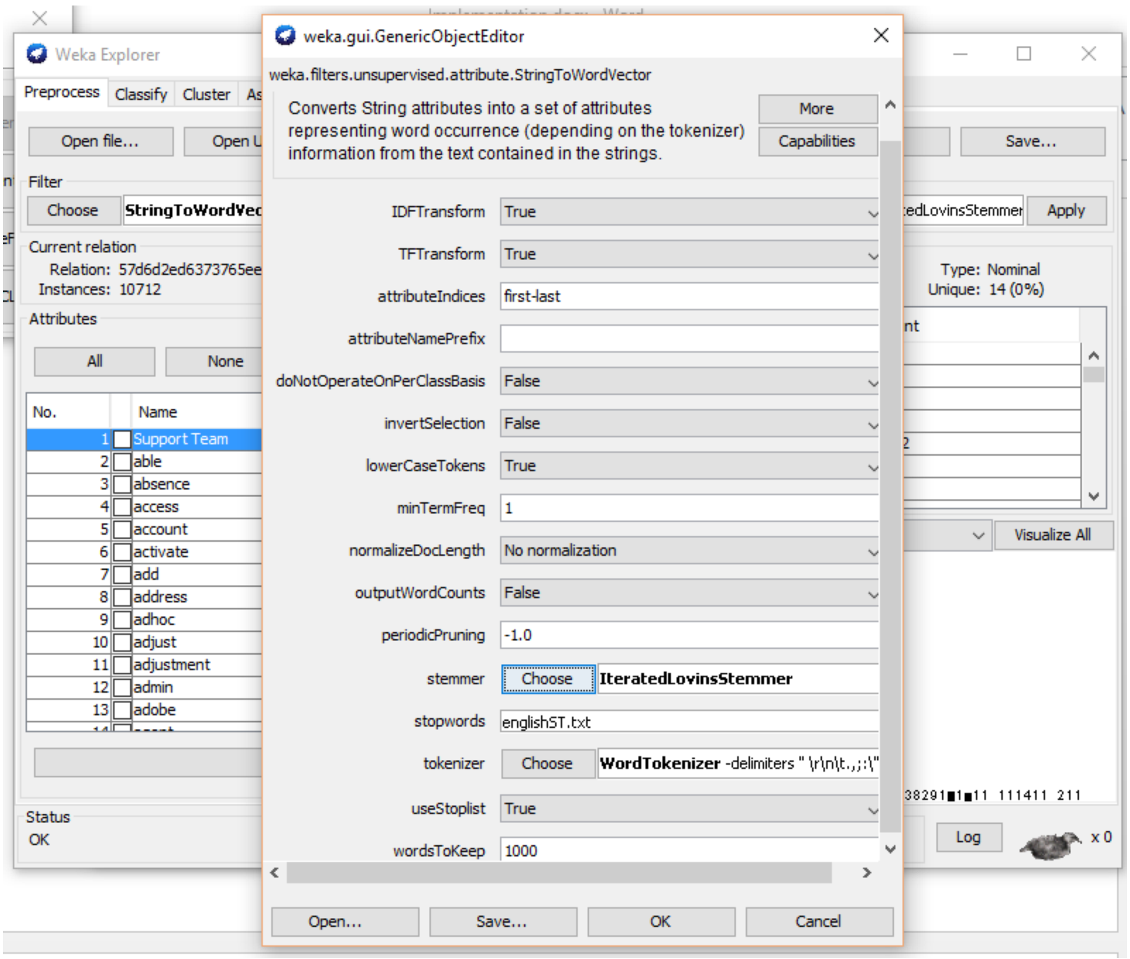

Figure 6.12: Weka String to Vector Settings

Data Reduction

For data reduction, “Attribute Selection” is selected from Weka which is a supervised attribute filter that can be used to select attributes. “InfoGainAttributeEval” is selected as an evaluator which identifies attributes based on their information gain. We then select the “Ranker” option which ranks attributes by their individual evaluations. By setting its threshold to 0 we can remove attributes which rank below zero. After applying this, the number of attributes drastically went down to 147 which means only these words have good information gain.

Classification

The next step is to find a fitting classifier algorithm by trying the different ones provided by Weka. After trying different classifiers, the results from J48, Random Forest, Random Tree, REP Tree and PART were found to have a higher percentage of correctly classified instances. The classification was done by a Percentage split of 66%.

At first, it was difficult to get a percentage above 74.5%. Figure 6.13 below shows the result obtained during the first time J48 classification algorithm was applied to the data.

```
=== Summary ===

Correctly Classified Instances        6924               73.9033 %
Incorrectly Classified Instances      2445               26.0967 %
Kappa statistic                          0.6826
Mean absolute error                      0.0376
Root mean squared error                  0.1518
Relative absolute error                 41.2119 %
Root relative squared error             71.3645 %
Total Number of Instances             9369
```

Figure 6.13: Summary of classification

After seeing this result, we proceeded with identifying the incorrectly classified attributes with the help of Weka. In order to improve the accuracy of the classifying algorithm, the preprocessing steps mentioned in sections above were applied in order to fix errors in the data.

Table 6.2 Performance of the different classifiers

Evaluation Criteria	J48	Random Forest	REP Tree	Random Tree	PART
Correctly Classified Instances	89.76 %	89.21 %	88.94 %	90.8091 %	89.0775 %
Time taken to build model	15.68s	71.87s	5.82s	0.87s	65.93
Size of Tree/ Number of Rules	265	10	253	2101	138

As displayed in Table 6.2 five of the top performing classifying algorithms are selected and compared. It can be seen that each of them have around 89% of correctly classified instances even though the time taken to build the model and size of tree varies. From this we have selected J48 to be the better classifier since it has the higher percentage while taking less time and having an acceptable size of tree.

When more and more incidents are logged and the data becomes large, the model created will enable the system to assign technicians to incidents faster and with better accuracy.

Figure 6.12 displays the results from J48 classifier done on a 66% Percentage split of the data.

```
=== Summary ===

Correctly Classified Instances        8435               89.7627 %
Incorrectly Classified Instances       962               10.2373 %
Kappa statistic                          0.8792
Mean absolute error                      0.0204
Root mean squared error                  0.101
Relative absolute error                 20.1803 %
Root relative squared error             44.9264 %
Total Number of Instances             9397
```

Figure 6.14: Results from REPTree

Confusion Matrix

A confusion matrix shows the number of correctly and incorrectly classified items. It displays how each items from various classes listed at the top are classified as the list of classes displayed at right.

It can be seen from the confusion matrix on Fig. 6.13 that the majority of the incidents are being classified in the team “EUD”.

By looking at the data it can be seen that some incidents with the same description or very similar description are assigned to different teams. By talking to experts in the company, it was realized that this is happening due to human error. Incidents are assigned to the wrong team and even though these incidents are eventually handled by the correct team, it is not logged in the system. In the future, this can be corrected by enforcing a rule that Service Desk technicians need to make the edit when an incident is assigned to a wrong technician and then re-assigned to a correct one.

In addition to this, encouraging the customers both on the system and at their offices, to write a more descriptive text when reporting an incident will significantly increase the results. Also, when the data is collected over a longer period of time and when the types of incidents get repeated, the system will be able to identify them better.

=== Confusion Matrix ===

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	<-- classified as
111	0	0	13	1	1	1	2	0	0	0	0	0	0	0	3	0	a = PSS
0	579	0	3	0	0	2	1	0	0	0	0	0	0	0	1	0	b = EUD OUTST
0	0	404	55	1	1	2	0	0	0	0	0	1	2	0	0	0	c = NET COMN
6	0	40	2766	3	4	7	17	0	4	0	0	4	1	12	1	2	d = EUD
0	0	0	44	367	2	4	1	0	12	0	0	1	0	2	3	0	e = SAP HCM
0	0	0	49	10	343	3	1	0	0	0	0	5	0	3	3	0	f = SAP LOG
1	0	15	139	6	4	1261	1	0	0	0	0	0	0	1	3	0	g = EUD RMT
1	0	5	84	0	1	20	574	0	0	0	0	0	0	2	0	0	h = SYS ADMIN
0	0	0	3	1	0	0	0	50	0	0	0	0	0	0	0	0	i = REV ACT
0	0	1	8	3	0	2	12	0	202	0	0	0	0	1	1	0	j = CENTRAL SD
0	1	0	1	1	0	0	0	0	0	27	0	0	0	0	1	0	k = DB ADMIN
0	0	4	13	0	0	0	0	0	0	0	467	0	0	1	0	0	l = NET ADMIN
1	0	0	38	7	22	4	1	1	0	0	0	208	0	3	0	0	m = SAP FICO
0	0	47	23	0	0	1	1	0	0	0	1	0	273	0	0	0	n = NET TECH
1	1	0	17	7	24	2	1	0	1	1	0	6	0	373	1	0	o = SAP PORT
1	0	0	58	2	0	1	1	0	0	0	0	0	0	1	222	0	p = CARGO SYS Cargo Systems
0	0	0	7	0	0	8	0	0	0	0	0	0	0	0	0	208	q = OS AVTN

Figure 6.15: Confusion Matrix

6.3 Testing and Results

In this section, we will be discussing results found from doing usability test on the ITIL Service Desk System.

Usability Testing

Usability testing tests whether the system fulfills the requirement or not and if it's effective in allowing users use the system to perform the expected tasks. It tests whether an application or a product is user-friendly or not.

Usability testing also reveals whether users feel comfortable with the application according to different parameters – the flow, navigation and layout, and content. Usability testing also tests the following features of the system: How easy it is to use the system? How easy it is to learn the system? How convenient the system is to end user? and Do the features meet their needs?

According to ISO/IEC 25010, "quality in use is the degree to which a product or system can be used by specific users to meet their needs to achieve specific goals" [20]. It categorizes the properties of quality in use as effectiveness, efficiency, freedom from risk and satisfaction in specific contexts of use. Thus, testers for the ITIL Service Desk system were made to test these properties.

The test was conducted by identifying targets which are suitable to take the test. These targets were made to use the system according to the roles they would understand and afterwards a questionnaire was provided to them for gathering feedback. A total of five participants were chosen for this study comprised of users with different roles in regards to the Service Desk. The evaluators were:

Table 7.1 Testers Profile

No.	Profession	Age Group	Gender	Role Tested
1.	Financial Officer	25 – 30	Male	Customer
2.	Ticketing	25 – 30	Female	Customer

3.	Support Team Head	35 – 40	Male	Administrator
4.	Manager	25 – 30	Male	Manager + CAB
5.	IT Support Staff	25 - 30	Male	Technician

The participants performed different tasks and their respective responses were recorded. While conducting the test, each participant was made to complete the task without any guidance, in order to prevent bias.

After all the participants completed the study, the data was compiled to determine the outcome. The questionnaires which were given out with the respective responses are as tabulated below:

Table 7.2 Testing Results

No	**Questions**	**Responses (in Percentage)**				
		Disagree Strongly	**Disagree**	**No Response**	**Agree**	**Agree Strongly**
1	I was able to understand the functionalities of the system without difficulty.				2	3
2	I could easily find what I am looking for, everything is organized well and set out as expected.				2	3
3	It accurately captured the workflow of the organization and has features that will improve the current service management process in the company.				0	5

4	Functionalities I need are included in the system and there aren’t any missing features that I want included.				4	1
5	The navigation system is intuitive and helpful.				2	3
6	I liked the user interface, it was easy to use with good design.				1	4

Based on this testing we can conclude that the users were able to navigate through the system easily and found the different functionalities to be very helpful. We can also see that the requirements for quality in use were met.

Chapter 7 – Conclusion and Recommendation

The IT support department of Ethiopian Airlines can greatly benefit from a well-organized service management and an effective resource management system. Without this system, not only will the staff of the department be unproductive, the people who are requesting for the service will not get solutions in time and maintenance cost will be high.

This project implements a Service Desk introducing selected processes of the ITIL standard. These processes aim to enable the IT support team to address issues effectively and in an organized manner. Recurring incidents that need further root cause analysis could also be logged in the system as problems and assigned to various technicians. When change is necessary to fix a certain problem, the system allows change management procedures to be handled with ease of communication. Assets are also registered according to their type and linked with incidents so that faulty assets could easily be identified.

In addition to this, the Service Desk automatically assigns incidents to technicians. This is done by creating a classification model using Weka tool with data from their existing system. Integration with WEKA is also implemented so that the classification model gets updated as the department uses the system more.

Automatically assigning incidents to technicians will minimize the number of incidents that are not assigned in time as well as those that get assigned to the wrong technicians. While this improves resolution time, other components of the Service Desk will allow the technician address incidents, problems and changes in a structured way which will increase effective communication among the different stakeholders.

Future work and Recommendation

In the future, we recommend that ET implements other ITIL processes as well in order to benefit greatly from the system. The other processes may include Availability Management, Release and Deployment Management, Capacity Management, Supplier Management etc. which can be integrated with this system.

In addition to that, since the number of mobile device users is increasing, a native mobile application can also be developed and integrated with the system for convenience. It will allow users to access certain things offline once they download them e.g. a list of tasks to be done by technicians. Push notifications can also be enabled so that technicians can get notified on their phone when they are assigned to a task and customers can be notified when the incident they reported is solved or when a technician is asking for clarification.

Since native mobile applications have access to native APIs, users can benefit from things such as integration with the mobile phone calendar for scheduling their tasks and getting reminders as well.

References

[1] Alastair Bovim, Kevin Johnston, Salah Kabanda, Maureen Tanner, and Adrie Stander, ITIL adoption in South African: A Capability Maturity view, Proceedings of the e-Skills for Knowledge Production and Innovation Conference 2014

[2] FitSM – Standards for lightweight IT Service Management (Part 0: Overview and vocabulary), Edition 2016 – Version 2.3, June 14, 2016

[3] Service Desk, http://en.wikipedia.org/wiki/Service_desk_(ITSM), last accessed on September 2016

[4] Matthew L. Helm, Information Technology Infrastructure Library (ITIL) Overview, September 2010

[5] Service operation, Office of Government Commerce, London 2007

[6] The Official Introduction to the ITIL Service Lifecycle, 2007

[7] Mathias Sallé, IT Service Management and IT Governance: Review, Comparative Analysis and their Impact on Utility Computing, Trusted Systems Laboratory, HP Laboratories Palo Alto, June 2004

[8] ISACA, Cobit 5 Introduction, February 28 -2012

[9] Microsoft Corporation, Microsoft® Operations Framework - Cross Reference ITIL® V3 and MOF 4.0, May 2009

[10] S.C. Hui*, G. Jha Nanyang, Application Data mining for improving customer service support, Technological University, School of Applied Science, Nanyang Avenue, Singapore 639798, Singapore; 1999

[11] Donna Knapp, A Guide to Service Desk Concepts, Course Technology, Cengage Learning, 2010

[12] Dominik Reichelt, Alexander Lawall and Thomas Schaller, Inter-Organizational Case Assignment Based on Agent Attributes and Functions, University of Applied Sciences Hof - Institute for Information Systems, 2015

[13] Rule Induction https://en.wikipedia.org/wiki/Rule_induction, last accessed on November 5

[14] Decision Tree, http://www.saedsayad.com/decision_tree.htm, last accessed on September, 2016

[15] Ben-Gal I., Bayesian Networks, Encyclopedia of Statistics in Quality & Reliability, Wiley & Sons (2007)

[16] Kevin B. Korb, Ann E. Nicholson, Bayesian Artificial Intelligence: Second Edition, 2011

[17] Altman, N. S. (1992). "An introduction to kernel and nearest-neighbor nonparametric regression". The American Statistician. 46 (3): 175–185.

[18] Mr Padej Phomasakha Na Sakolnakorn, Knowledge Management System Improvement Towards Service Desk Of Outsourcing In Banking Business, 2007

[19] Mucahit Altintas, A. Cuneyd Tantug, Machine Learning Based Ticket Classification In Issue Tracking Systems, Proceeding of the International Conference on Artificial Intelligence and Computer Science (AICS 2014), 15 - 16 September 2014, Bandung, INDONESIA.

[20] ISO/IEC 25010, Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models, https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en, last accessed on January 1, 2017

DECLARATION

I, the undersigned, declare that this project is my original work and has not been presented for a degree in any other university, and that all source of materials used for the project have been properly approved.

Declared by:

Name: ________________________________

Signature: ____________________________

Date: ________________________________

Confirmed by Advisor:

Name: ________________________________

Signature: ____________________________

Date: ________________________________