

# Inventory Database Management System



By

Mahin Huda Malik

Miqdad Askari Khan

Muzammal Hussain

Usman Shafiq

Submitted to Faculty of Department of Computer Software Engineering National University of Sciences and Technology, Islamabad in partial fulfillment for the requirements of a B.E Degree in Computer Software Engineering, May 2017

## Inventory Database Management System

In the name of Allah, the Most Beneficent, the Most Merciful

## **ABSTRACT**

### **INVENTORY DATABASE MANAGEMENT SYSTEM**

The project will be used by the MCS management to keep track of the inventory system. Different access levels will be created for accessing database. The system will automatically generate reports on monthly basis. Separate reports can also be generated on the user's request. Data searching will be available on various attributes.

The main components of the system are a relational database on MySQL or another portable IDMS stores the essential data for subsequent analysis An Acquisitions Subsystem (AS) facilitates tracking of purchases, utilization and sale of inventory system. ASP. NET is being used for the management of the system CSS and Javascript will be used for the user interface of the system

Objective of the project is to develop the Inventory Management system for MCS:

To provide Inventory System access to all necessary personnel

To provide a full range of reports that will satisfy informational requirements

Develop an automated system that will be able to record, store, retrieve, and generate reports of inventory useful to management in decision-making

## **CERTIFICATE FOR CORRECTNESS AND APPROVAL**

It is certified that work contained in the thesis – Inventory Database Management System carried out by **Mahin Huda Malik, Miqdad Askari, Muzammal Hussain, Usman Shafiq** under supervision of **Dr. Naima Iltaf** for partial fulfillment of Degree of Bachelor of Software Engineering is correct and approved.

**Approved By**

---

**Dr. Naima Iltaf**

**Department of CSE, MCS**

## **DECLARATION**

No portion of the work presented in this dissertation has been submitted in support of another award or qualification either at this institution or elsewhere.

## **DEDICATION**

First of all, would like to Thank ALLAH Almighty for his blessings through which we are in this institute and presenting this landmark project.

Secondly Our Parents, without whose support and cooperation, a work of this magnitude would not have been possible. And to our supervisor, **Dr. Naima Iltaf**, who gave us great support and valuable suggestions throughout the implementation process.

Lastly to the creative minds of our group without whom this all wasn't possible.

## **ACKNOWLEDGEMENTS**

There is no success without the will of ALLAH Almighty. We are grateful to ALLAH, who has given us guidance, strength and enabled us to accomplish this task. Whatever we have achieved, we owe it to Him, in totality. We are also grateful to our parents and family and well-wishers for their admirable support and their critical reviews. We would like to thank our supervisor, **Dr. Naima Iltaf**, for her continuous guidance and motivation throughout the course of our project. Without their help, we would have not been able to accomplish anything.

## Contents

<b>List of Tables .....</b>	<b>xii</b>
<b>List of Figures .....</b>	<b>xiii</b>
<b>Chapter 1. Introduction.....</b>	<b>14</b>
1.1. Overview .....	14
1.2. Problem Statement .....	14
1.3. Approach .....	14
1.4. Scope.....	14
1.5. Objectives .....	14
1.6. Deliverables.....	15
<b>Chapter 2. Literature Review .....</b>	<b>16</b>
2.1. Introduction .....	16
2.2. What is Inventory Management System? .....	16
2.3. Advantages of Inventory Management System .....	16
2.3.1. Cost Savings .....	16
2.3.2. Increased Efficiency .....	16
2.3.3. Updated data .....	17
2.3.4. Data security.....	17
2.4. Similar Systems.....	17
2.4.1. Vendor Management Systems.....	17
2.4.2. Tracker Systems.....	17
<b>Chapter 3. Software Req. Specification (SRS) .....</b>	<b>18</b>
3.1. Introduction .....	18
3.1.1. Purpose .....	18
3.1.2. Document Conventions.....	18
3.1.3. Intended Audience and Reading Suggestions.....	18
For better understanding, the document is divided into sections:.....	18
This document is intended for:.....	19
3.1.4. Project Scope .....	19
3.1.5. References .....	19
3.2. Overall Description.....	20
3.2.1. Product Perspective .....	20
3.2.2. Product Function:.....	20
3.2.3. User Classes and Characteristics .....	20



3.2.4. Operating environment.....	20
3.2.5. Design and Implementation Constraints .....	21
3.2.6. User Documentation.....	21
3.2.7. Assumptions and Dependencies.....	21
3.3. System Features .....	22
3.3.1. Data Entry .....	22
3.3.2. Data Searching.....	22
3.3.3. Data Modification.....	23
3.3.4. Reports .....	23
3.3.5. Privilege Modes.....	24
3.4. External Interface Requirements .....	24
3.4.1. User Interfaces .....	24
3.4.2. Hardware Interfaces.....	24
3.4.3. Software Interfaces.....	24
3.5. Other Nonfunctional Requirements.....	25
3.5.1. Performance Requirements .....	25
3.5.2. Security Requirements.....	25
3.5.3. Software Quality Attributes.....	25
<b>Chapter 4. Design and Development .....</b>	<b>26</b>
4.1. Introduction .....	26
4.1.1. Purpose.....	26
4.1.2. Project Scope.....	26
4.1.3. References .....	26
4.1.4. Overview of document .....	26
4.2. System Architecture Description.....	27
4.2.1. Overview of modules / components .....	27
4.2.2. Structure and relationships .....	28
4.2.3. User interface issues .....	28
4.4. Detailed Description of Components.....	30
4.4.1. Component template description .....	30
4.4.2. UI Forms.....	30
4.4.3. Input Event Handler.....	32

4.4.4. Input Validator .....	33
4.4.5. Data Processing .....	33
4.5. Detailed Description of System .....	35
4.5.1. Use Cases .....	35
4.5.2. Activities .....	42
<b>Chapter 5. Project Test and Evaluation .....</b>	<b>45</b>
5.1. Introduction .....	45
5.2. Test Items .....	45
5.2.1. Features to be Tested .....	45
5.3. Approach .....	46
5.3.1. Unit Testing .....	46
5.3.2. Integration Testing .....	46
5.3.3. System Testing .....	46
5.4. Item Pass/Fail Criteria .....	46
5.5. Suspension Criteria and Resumption Requirements .....	46
5.6. Test Deliverables .....	47
5.6.1. Test Case 1 .....	47
5.6.2. Test Case 2 .....	49
5.6.3. Test Case 3 .....	51
5.6.4. Test Case 4 .....	53
5.6.5. Test Case 5 .....	55
5.6.6. Test Case 5 .....	55
5.6.7. Test Case 7 .....	57
5.6.8. Test Case 8 .....	58
5.6.9. Test Case 9 .....	60
5.7. Responsibilities, Staffing and Training Needs .....	63
5.7.1. Responsibilities .....	63
5.7.2. Staffing and Training Needs .....	63
5.8. Risk and Contingencies .....	63
5.8.1. Schedule Risk .....	63
5.8.2. Budget Risk .....	63
<b>Chapter 6. Future Work .....</b>	<b>64</b>

<b>Chapter 7. Conclusion .....</b>	<b>65</b>
<b>Bibliography.....</b>	<b>66</b>
<b>Appendix A. Glossary .....</b>	<b>67</b>
<b>Appendix B: Issues/Limitations.....</b>	<b>68</b>

## List of Tables

Table 1-1: Deliverables.....	15
Table 4-1: UI Forms.....	32
Table 4-2: Event Handler.....	33
Table 4-3: Input Validator .....	33
Table 4-4: Data Processing.....	34
Table 4-5: Use Case-Data Entry .....	37
Table 4-6 Use Case- Data Search .....	39
Table 4-7: Use Case-Data Update.....	39
Table 4-8: Use Case-Reports .....	41
Table 5-1: Test Case- Add Data.....	47
Table 5-2: Test Case-Add Disposal Data.....	49
Table 5-3: Test Case-Disposal Order Data .....	51
Table 5-4: Test Case-Disposal Order Quantity Data .....	53
Table 5-5: Test Case-Disposal Search .....	55
Table 5-6: Test Case-Disposal Order Details .....	56
Table 5-7: Test Case-Disposal Search Filters.....	57
Table 5-8: Test Case-Data Search Filters .....	58
Table 5-9: Test Case-Filter Selection .....	60

## List of Figures

Figure 1: Client Server Architecture.....	27
Figure 2: Components Model .....	30
Figure 3: Use Case Diagram.....	35
Figure 4: Use Case-Data Entry .....	36
Figure 5: Sequence Diagram-Data Entry .....	36
Figure 6: Use Case-Data Search .....	37
Figure 7: Sequence Diagram- Data Search.....	38
Figure 8: Use Case Diagram: Data Update.....	39
Figure 9: Use Case- Reports .....	40
Figure 10: Activity Diagram-Data Entry .....	42
Figure 11: Activity Diagram-Update Database .....	43
Figure 12: Activity Diagram-Data Search .....	44
Figure 13: Data Entry.....	48
Figure 14: Disposal Data Entry.....	50
Figure 15: Search Filters.....	52
Figure 16: Multiple Filters .....	52
Figure 17: Disposal Data .....	54
Figure 18: Database Search .....	59
Figure 19: Database Search Filters .....	61
Figure 20: Multiple Database Search Filters .....	62

# Chapter 1. Introduction

## 1.1. Overview

This project focuses on developing a sophisticated inventory database system for Military College of Signals to ease up the data entry work and provide the MIMS administration a separate database system to manage the data on a separate interface. This inventory system mainly comprises of a database system which is being used for data entry, data mining using different filters and for administrative purposes.

## 1.2. Problem Statement

*“Currently MS Excel spreadsheets are being used which consumes more time and labor to update.”*

## 1.3. Approach

The project involves implementation of database in MS SQL to store the inventory database of MCS. To access the database ASP .NET is used and C# is used in the backend.

## 1.4. Scope

The project is intended to automate the inventory database of Military College of Signals. It is intended to provide a reliable separate database of the inventory of MCS to manage it. The main components of the system will be as follows:

- A relational database on MySQL or another portable IDMS stores the essential data for subsequent analysis
- An Acquisitions Subsystem (AS) facilitates tracking of purchases, utilization and sale of inventory system
- ASP .NET, C# is used for the frontend of the system
- CSS and JavaScript will be used for the user interface of the system

## 1.5. Objectives

Objective of the project is to develop the Inventory Management system for MCS:

- To provide Inventory System access to all necessary personnel
- To provide a full range of reports that will satisfy informational requirements

Develop an automated system that will be able to record, store, retrieve, and generate reports of inventory useful to management in decision-making.

## 1.6. Deliverables

<b>Sr.</b>	<b>Tasks</b>	<b>Deliverables</b>
<b>1</b>	Literature Review	Literature Survey
<b>2</b>	Requirements Specification	Software Requirements Specification document (SRS)
<b>3</b>	Detailed Design	Software Design Specification document (SDS)
<b>4</b>	Implementation	Project demonstration
<b>5</b>	Testing	Evaluation plan and test document
<b>6</b>	Training	Deployment plan
<b>7</b>	Deployment	Complete application with necessary documentation

Table 1-1: Deliverables

## **Chapter 2. Literature Review**

### **2.1. Introduction**

In the previous chapter 1 discussed earlier the main objective of the senior project is to implement an inventory management system for reducing the errors and workload of employees in MCS .This chapter will be cover the content that will include the different types of inventory management systems, similar systems how is it used, different software's used for the inventory management etc.

### **2.2. What is Inventory Management System?**

This is a computer based system for tracking inventory levels, sales , orders and deliveries. Companies use this system to avoid the overflowing of products and prevent product outage .This tool is used for organizing inventory data that was previously stored in excel sheet manually which was really cumbersome process and time consuming. This tool can be associated which is similar to distribution software, as distributors that can compete with less cash tied up in inventories have a distinct advantage over their competitors.

### **2.3. Advantages of Inventory Management System**

The some of the advantages of inventory system are thus summarized below which can prevent the organization /firm from suffering from big financial loss and other problems that may occur during the daily operations of the firm that can be viewed as the materials being out of stock or machine failures and many other operations happening on daily basis. There are several Advantages of using the inventory management in a business setting are:

#### **2.3.1. Cost Savings**

In many cases companies' inventory is one of the largest investments along with its workforce and locations. Inventory management systems help the companies to cut the expenses by minimizing the amount of unnecessary products and materials in storage. It also helps companies keep lost sales to a minimum by having enough stock to meet demand

#### **2.3.2. Increased Efficiency**

The inventory management system allows for may automated inventory task for example the system can automatically collect data, calculate costs .this also reduces in costs saving and time saving and thus consequently leading to increase in business efficiency.



### **2.3.3. Updated data**

Provides up to date and real time data on inventory levels is and advantage of inventory management system. Company executives can usually access the software through their mobile devices, laptop for checking current inventory numbers this automatic updating of inventory allows the business to make informed decisions.

### **2.3.4. Data security**

By accompanying with the restricted user rights, company managers can allow many employees to assist in inventory management. They can grant employees enough information access for tasks such as receiving products, taking orders, transfer products and perform other tasks without compromising company security. This can speed up the inventory management process and save managers' time.

## **2.4. Similar Systems**

### **2.4.1. Vendor Management Systems**

This system is a new trend for inventory management system. With this system the distributors can control inventory management for customers .The reports are calculated on daily basis and are transferred from the customer to the vendor. The data gathered from these reports lets the vendor knows when to recalculate the stock of the customers merchandise. This method significantly reduces the load of paper work /the cost and the labor .There are few drawbacks of using this method which is this will be difficult for the distributor/suppliers to keep up with demand of the product that is popular and is demanded by the customers from all around the world. Companies believe that the money saved in time and labor is worth the risks involved.

### **2.4.2. Tracker Systems**

Tracker system is an inventory management system allowing you to control your inventory with this software management system the you can you can track and control purchasing, invoicing, product recalls, shipping and receiving, ordering and all other aspects of inventory management. The software comes with full system of reports allowing you to have the status of your inventory available at your fingertips. Some of the forms available with Tracker Systems include packing slips, invoices and inbound receipts. Analysis reports include analysis summary, allocation reports, products lists, inventory summary and inventory history. The operating system required to use Tracker Systems is Windows 2000, Windows XP or Windows Vista

## **Chapter 3. Software Req. Specification (SRS)**

### **3.1. Introduction**

#### **3.1.1. Purpose**

This document describes the system study, proposed subsystems, data model of the system, and the modules implemented. This document briefly explains the functionality of each module that will be used in the system. Furthermore this document also describes the tasks that will be available to perform by the users of the system.

#### **3.1.2. Document Conventions**

- Heading are prioritized in a numbered fashion, the highest priority heading having a single digit and subsequent headings having more numbers, according to their level.
- Font used is Times New Roman
- All the main headings are of size 22 and bold.
- All the second level sub-headings are of size 16 and bold.
- All the further sub-headings are of size 14 and bold.
- All references in this document are provided where necessary, however where not present, the meaning is self-explanatory. All ambiguous terms have been clarified in the glossary at the end of this document.

#### **3.1.3. Intended Audience and Reading Suggestions**

This document is primarily intended for the developers, the project supervisor, and the evaluators. Furthermore, any person who needs to understand the system can found this document helpful.

**For better understanding, the document is divided into sections:**

- In section 2 an overall description of Application is provided. First product perspective is presented with product features and main functions. Then follow user classes and characteristics, operating environments that Application supports as well as design and implementation constraints. After all that, user documentation is presented and will provide you with more details about each feature's technology.
- In section 3 most important features are presented with detailed description, use cases and requirements.
- In section 4 user, hardware, software and communication interfaces are described.
- In section 5 requirements about security, safety and performance are presented along with the software quality attributes of the Application.

### **This document is intended for:**

- **Developers:** (Project Group)  
In order to be sure that they are developing the right project that fulfills the requirements provided in this document.
- **Testers:** (Project Group, Supervisor)  
In order to have an exact list of the features and functions that must respond according to requirements.
- **Users:**  
In order to get familiar with the idea of the project and how to use/respond in failure situations and suggest other features that would make it even more functional.
- **Documentation writers:** (Project Group)  
To know what features and in what way they have to explain. What technologies are required, how the system will respond in each user's action, what possible system failures may happen and what are the solutions to all those failures etc.
- **Project Supervisor: (Dr. Naima Iltaf)**  
This document will be used by the project supervisor to check whether all the requirements have been understood and in the end whether the requirements have been implemented properly and completely.
- **Project Evaluators:** (CSE Dept. MCS)  
To understand the scope of the project and evaluate it accordingly.

### **3.1.4. Project Scope**

This project is intended to provide a reliable separate database of the inventory of MCS to manage it. The scope of this project is to fully computerized the database.

### **3.1.5. References**

The reading material and some guides are available in the form of web links.

## **3.2. Overall Description**

### **3.2.1. Product Perspective**

The product is a new project to fully computerize the inventory database of Military College of Signals. Previously the database was operated on MS Excel spreadsheets and Ledger Books. This product will make the management of the database more easy and secure.

### **3.2.2. Product Function:**

Main features of the product are given below:

- Data entry
- Data view interface
- Data searching
- Updating existing data
- Reports
- User accounts

### **3.2.3. User Classes and Characteristics**

Following are user classes and their brief description.

#### **Tester (occasional user)**

Tester will use this project to check for bug finding. They will also use the project to check if it's in accordance to the Software Requirements Specification document.

#### **Project Supervisor (occasional user)**

Project supervisors will also use the product to evaluate. They will use this use this product to find the accuracy and error in the output.

#### **SMT Staff (Regular user)**

SMT Staff will use this product to manage the inventory database of MCS.

#### **Administration**

MIMS administrator will use this product to define access levels for the users of the software.

### **3.2.4. Operating environment**

Required operating environment for the application is listed below.

#### **Hardware Requirements:**

- **Computer:** To run the web browser for accessing the database.

- **Network Infrastructure:** To provide network connection to the software. Local Area Network will be used for accessing MIMS database.

**Software Requirements:**

- **Operating System:** Windows (XP, Vista, 7, 8, 8.1, 10), Linux.
- **Web Browser:** To access MIMS database.

### **3.2.5. Design and Implementation Constraints**

Constraints of the product are given below:

- Software will work with network connection

### **3.2.6. User Documentation**

A user manual will be provided to the users in which separate instructions will be given according to the user access level. It will include the details of the system's working. The project report will also be available for the users which will highlight the system features, working and procedures.

### **3.2.7. Assumptions and Dependencies**

- Overall performance of the product will depend on the hardware infrastructure and network speed.
- User must know about the Interface for the better performance of the product.
- Limitations of the product must be kept in mind by the user.

### **3.3. System Features**

System features are the most important part of our project as it is the basic outline of the included features of the database system. Following are the features included in the project.

#### **3.3.1. Data Entry**

The data entry process is the most important feature of this software. Database will be created by the following ways:

##### **Importing Data**

Data available in excel files in the form of spreadsheets will be imported to the database to create new database. Simple File Reader in ASP will be used for importing data from these excel files.

##### **Adding New Data**

New data can also be added through this option directly through the software. Data entry forms will be used to enter new data which will be saved in the database.

#### **3.3.2. Data Searching**

The user will be able to search the database in various ways. The following filters will be available in the database searching feature:

##### **Full Data**

Full database can be viewed in this feature. The user will be able to view the complete database and navigate through it.

##### **Ledger Page No**

The user will be able to search the database through this filter by selecting the option of L.P no.

##### **Nomenclature/Item Category**

The user will be able to search the database through this filter by selecting the option of Category. In the filter the type of the inventory will be described by item property i.e. if it's a computer, furniture, electrical equipment etc.

##### **Departments**

The user will be able to search the database for inventory through departments filter. In this filter user will be given a drop down menu of department names.

### **Labs**

The user will be able to search the database through the laboratories. In this filter the user will be asked to select one of the given options in the drop down menu.

### **Date & CRV No**

The user will be able to search the database through date & CRV (certificate receive voucher) no.

### **Vendor**

The user will be able to search the database for the inventory through this filter.

## **3.3.3. Data Modification**

Database will be able to modify. Updating the database consists of the following ways:

### **Add New Data**

New data can be added in the database. It will be available in the option of Modify Database. In this option the user will be prompted to enter data in the given form which will be saved in the existing database

### **Edit Previous Data**

Previous data can be added through this option. The software will ask the user to choose the data to be updated. The user will then be able to change the previous data.

### **Delete Data**

Data can be deleted from the database from this option. By selecting this option the user will be asked to choose the data which is to be deleted.

## **3.3.4. Reports**

The software will be capable of generating reports in the following types:

### **Daily Reports**

The reports will be generated on daily basis. These reports will be saved in the database and can be viewed and printed by searching.

### **Monthly Reports**

Monthly reports will be generated and can be viewed and printed by the user on request.

### **Requested Reports**

Reports will also be generated upon the request of the users. These reports will be generated by the choice of user and will use filters for selecting the data for the reports. These reports can be printed by the user.

### **3.3.5. Privilege Modes**

User privilege levels will be set on their unique ID on MIMS. Access level of the users can be changed by the database manager. The software can only be accessed by the user who has access privileges. Modification of the user accounts can be done by the database manager.

## **3.4. External Interface Requirements**

As far as our software engineering field is concerned, every project has three basic external interfaces which should be properly defined and designed while working on the project to make it more understandable. Following are the external interface requirements

### **3.4.1. User Interfaces**

Responsive graphical user interface will be provided to user to work with the software. Users with the access level will be provided the option to access the database. A separate interface will be available to navigate through features of the software. It will be designed using ASP .NET.

### **3.4.2. Hardware Interfaces**

The product requires a functional PC in order to work properly. This software requires an active connection to the Local Area Network to access the MIMS database.

### **3.4.3. Software Interfaces**

The software will be able to run on windows and Linux. Software will require a web browser to access the MIMS database.



### **3.5. Other Nonfunctional Requirements**

There are also some non-functional interface requirements. Some of them are briefly explained as follows:

#### **3.5.1. Performance Requirements**

Detailed database will be maintained. Due to various filters available the database searching/updating is very fast and accurate. The user will select one of the filters and manage the database as per requirement.

#### **3.5.2. Security Requirements**

User will be asked to enter login credentials to access the software. User access levels will be created. Access to the software will be given to limited users depending upon their access level only.

#### **3.5.3. Software Quality Attributes**

- **Reliability:** Software should provide reliability to the user. The product will run stably with all the features aforementioned available and working. It should be tested and debugged completely. All exceptions should be well handled.
- **User Friendliness/Simplicity:** Software should have a graphically user interface with user friendly menus and options.
- **Reviewability:** Software will be provided to the SMT Staff through proper Military Protocols

## Chapter 4. Design and Development

### 4.1. Introduction

This project focuses on developing a sophisticated inventory database system for Military College of Signals to ease up the data entry work and provide the MIMS administration a separate database system to manage the data on a separate interface. This inventory system mainly comprises of a database system which is being used for data entry, data mining using different filters and for administrative purposes.

#### 4.1.1. Purpose

This document describes the system study, proposed subsystems, data model of the system, and the modules implemented. This document briefly explains the functionality of each module that will be used in the system. Furthermore this document also describes the tasks that will be available to perform by the users of the system.

#### 4.1.2. Project Scope

This project is intended to provide a reliable separate database of the inventory of MCS to manage it. The scope of this project is to fully computerized the database.

#### 4.1.3. References

The reading material and some guides are available in the form of web links.

1. **Sequence Diagram**  
[https://en.wikipedia.org/wiki/Sequence\\_diagram](https://en.wikipedia.org/wiki/Sequence_diagram)
2. **Block Diagram**  
[https://en.wikipedia.org/wiki/Block\\_diagram](https://en.wikipedia.org/wiki/Block_diagram)
3. **Use case modeling guidelines**  
<http://ieeexplore.ieee.org/document/787548/?reload=true&arnumber=787548>

#### 4.1.4. Overview of document

This section gives an overview of the remaining sections in this document.

Section 2 comprises of system architecture description. It contains the overview of working modules. Complete overview of the project is demonstrated using a complete diagrammatic view of the system. A high level design diagram is also used to visualize the structure within the system as well as individual modules. State machine diagrams are also used to show the basic workings in the project. Low level states using state machine diagrams are given with backend processing perspective as well as users' perspectives.

Section 3 has detailed description of each module in the system. Each sub-module is described as well. Diagrams are used wherever necessary for better understanding. Sketches for output screens are included in the subsection of web interface. It also covers

low level design diagrams like use cases, ER diagrams, sequence diagrams, and activity diagrams.

Section 4 has basic insight regarding the pseudo code of the system.

## 4.2. System Architecture Description

### 4.2.1. Overview of modules / components

This section introduces the high level design/system architecture (Fig 2.0). Using black box approach, we describe the main system modules and architecture. Our system is integrated in the existing MCS MIMS2 system. The main modules are UI and Business Logic layer while the third layer of Data Storage is being pre-implemented and is available in MIMS2 system.

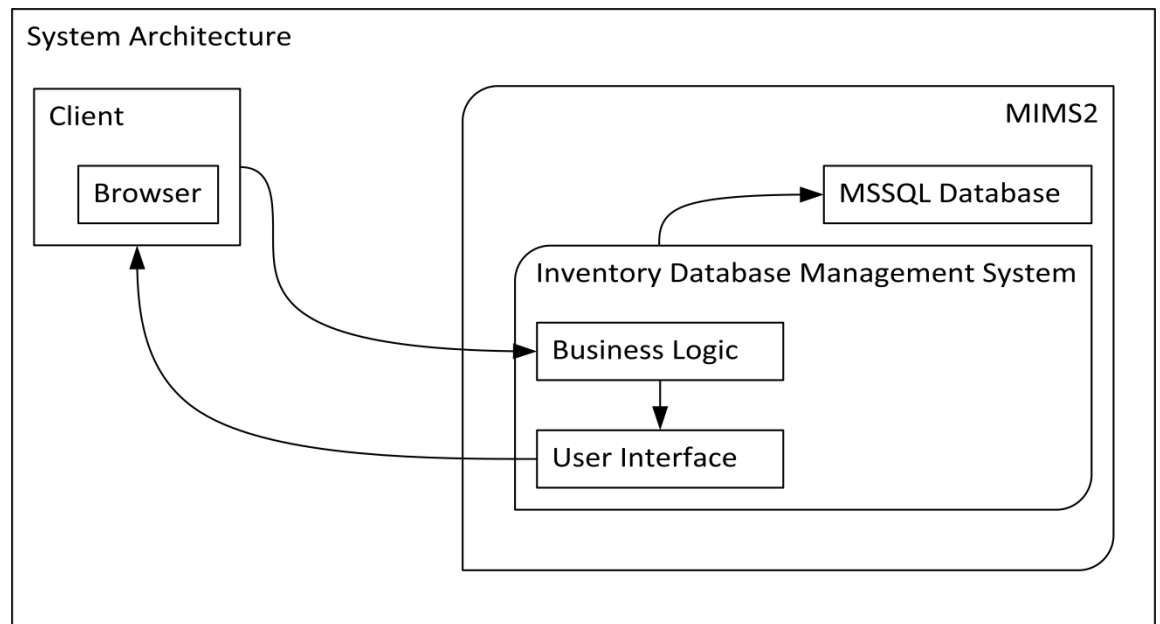


Figure 1: Client Server Architecture

#### Presentation Layer

Presentation layer is the user interface layer with .Net components for user interaction. Data can be entered and shown in proper formatting in presentation layer by the business layer.

#### Business Layer

The business layer is where all the logic i.e. sub-system (IDMS) level security, required functionalities and data processing is handled and data is fetched and saved to data layer.

## **Data Layer**

Data layer is the storage layer having MSSQL database already functional for existing MIMS2 functions (registrations etc.). The IDMS related tables will be implemented to this layer and as our system will be a part of MIMS2, this layer will be accessible as Data Layer.

### **4.2.2. Structure and relationships**

The IDMS is a 3-tier architecture based system and its architecture shown in Fig 2.0. The system has 3 modules. Here we discuss dependencies of the modules of the IDMS system.

#### **UI Module**

The user interface is the standalone module of IIDMS. It is interacted by user and depends on controller and model. Any data input such as login credentials are taken from UI and any data and information output is shown by this module such as reports generation depending upon the entered configuration.

#### **Business Logic**

The controller and model of our system are generally the same. Controller will handle data flow and user input and model will comprise on the functionalities made available by the IDMS. The business layer is dependent on the user input and data layer. This module will interact with data layer directly i.e. executing queries for data fetch/update.

#### **Data Module**

Pre implemented database module is the MIMS2 MSSQL database which will store IDMS data. This database is also responsible for user information for login purposes.

### **4.2.3. User interface issues**

The UI of IDMS is based on .net framework C# forms. These forms give us the mostly used UI components used in present day applications. Presently the UI of IDMS is quite simple and consists of following screens and no issues are expected as conflicts. All the possible errors will be properly handled and located via appropriate messages and location information to locate the error in a particular part of code.

#### **Login Screen**

Login Screen is the root directory of the UI tree of IDMS. It is simple screen with login credentials and proper hidden error messages to assist the user with login process. Login screen is only assessable when user is not logged in. The login button will log the user in if the credentials are correct. The forgot password button will let the MIMS2 password recovery service handle the request.

### **Search Screen**

Search screen is common screen for administrator and SMT user. Admin can view any items for checking purposes and SMT users will have access to searching the inventory for every day needs. Screen is comprised of following:

- Search Query
- Date Interval Selection
- Query Type Selection List
- Results Table

The query will hold the string for searching the database with respect to the query type selection. Both will be used under date credentials restriction to search and show if any data brings up.

Table view will be populated with all attributes that any particular item may have and the data entries. Entries are only for view purposes here and can't be edited/deleted.

### **Add Data to Inventory**

Adding new data to inventory is done via add data UI. This UI allows the SMT user to add new data by supplying all the sufficient data attributes and will help them selecting attribute values quickly if they have been used before. The add button will validate the data and then it will be submitted to MSSQL database.

### **Generate Reports**

Administrator can generate summary for printing purposes. This UI page will house the possible reports that Admin can get from IDMS. More report formats can be added later in future. For present requirements set the IDMS generate reports based on search and query type. The difference between Generate Reports UI page and Search UI page is that reports generated can be exported in any suitable format such as .pdf and only admin has the access to this page.

## 4.4. Detailed Description of Components

The IDMS has 3 modules out of which 1 module are available as MIMS2 database module. The remaining modules with respective components are shown in Fig 3.0.

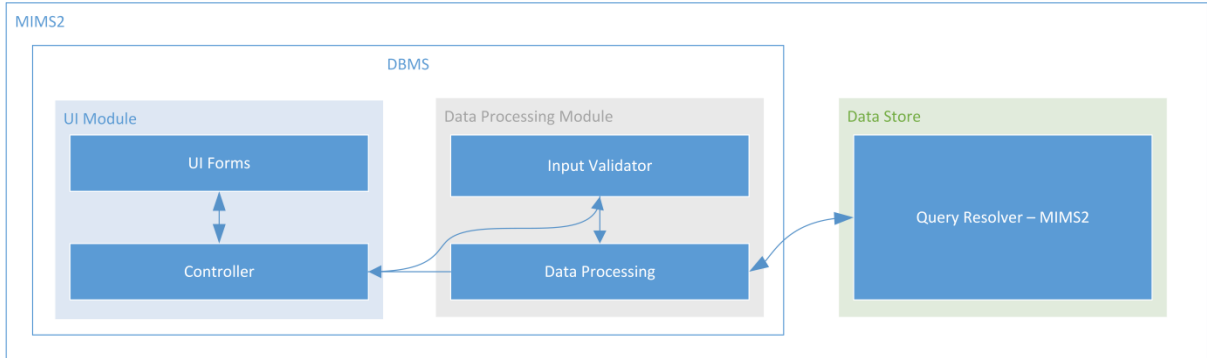


Figure 2: Components Model

### 4.4.1. Component template description

The components in IDMS are based on the concepts of .net framework and C# language. Following are the main components of IDMS

- **UI Forms:** the UI of the IDMS are the UI forms of .net and they contain any possible output and input GUI required by IDMS
- **Input Event Handler:** under the UI of IDMS is event handler which processes and redirects the user events such as mouse click on login button to the validator component.
- **Input Validator:** for best and smooth user experience input validation is necessary. The Input Validation component of IDMS will have all the required input validations and input processing.
- **Data Processing:** this component is the data handler component which is responsible to communicate with data layer/module to get and set the data in data store.

### 4.4.2. UI Forms

Identification	UI Forms is the component of UI Module of IDMS
Type	The UI Forms is sub system of UI Module which comprises of all UI classes
Purpose	The purpose of UI Forms is to separate all the processing from the View of UI Module and it is the GUI of IDMS and its function is to interact with the user visually for taking any input and showing any data.

Function	<p>Each form will be dynamic in layout and appropriately placed in the panels. The main functions of UI Forms is to interact with user. This is the component which takes user input for login purposes, adding data purpose and updating data purpose. It is the first layer between data and the user. It will show user all the useful information about the process and issues if any. The modifiable components of UI Forms are tables which output the data and calendars which let user to choose the date.</p>																				
Subordinates	<p>UI Forms structure is based upon layouts of components. It satisfies the requirements of “ease of access” which is primary goal of IDMS to make it easy to use for non-technical workers of MCS SMT.</p>																				
Dependencies	<p>Many components can be disabled based on dependencies. It is to assure that no wrong input is accepted which may result in error. Major dependencies include UI page accessibility depending on login and user type. If a particular type of user (SMT) is logged in, he can’t assess the Administrative level functionalities.</p>																				
Interfaces	<p>UI Forms will be the main communicator to user and business layer hence it will have all the set of messages and error codes.</p> <table border="1" data-bbox="646 1010 1498 1623"> <thead> <tr> <th data-bbox="646 1010 1073 1066">Message</th> <th data-bbox="1081 1010 1498 1066">Error Code (if applied)</th> </tr> </thead> <tbody> <tr> <td data-bbox="646 1077 1073 1157">Username and/or password is invalid</td> <td data-bbox="1081 1077 1498 1157">n/a</td> </tr> <tr> <td data-bbox="646 1167 1073 1209">Login Successful</td> <td data-bbox="1081 1167 1498 1209">n/a</td> </tr> <tr> <td data-bbox="646 1220 1073 1299">Database connection was not established</td> <td data-bbox="1081 1220 1498 1299">100</td> </tr> <tr> <td data-bbox="646 1310 1073 1352">Request Timeout</td> <td data-bbox="1081 1310 1498 1352">101</td> </tr> <tr> <td data-bbox="646 1362 1073 1404">No entries found</td> <td data-bbox="1081 1362 1498 1404">n/a</td> </tr> <tr> <td data-bbox="646 1415 1073 1457">Update was unsuccessful</td> <td data-bbox="1081 1415 1498 1457">n/a</td> </tr> <tr> <td data-bbox="646 1467 1073 1509">Update done successfully</td> <td data-bbox="1081 1467 1498 1509">n/a</td> </tr> <tr> <td data-bbox="646 1520 1073 1562">Report generation failed</td> <td data-bbox="1081 1520 1498 1562">102</td> </tr> <tr> <td data-bbox="646 1572 1073 1614">Report generated and saved</td> <td data-bbox="1081 1572 1498 1614">n/a</td> </tr> </tbody> </table> <p>Text inputs: components related to text input will be used to take user input such as login username and search query string</p> <p>Tables: components related to tables data will be used to show reports and search data to user</p> <p>Calendar: to avoid date format issues calendar component will be used to ease input of date from users</p>	Message	Error Code (if applied)	Username and/or password is invalid	n/a	Login Successful	n/a	Database connection was not established	100	Request Timeout	101	No entries found	n/a	Update was unsuccessful	n/a	Update done successfully	n/a	Report generation failed	102	Report generated and saved	n/a
Message	Error Code (if applied)																				
Username and/or password is invalid	n/a																				
Login Successful	n/a																				
Database connection was not established	100																				
Request Timeout	101																				
No entries found	n/a																				
Update was unsuccessful	n/a																				
Update done successfully	n/a																				
Report generation failed	102																				
Report generated and saved	n/a																				

	Buttons: buttons will be used to process the user interactions
Resources	n/a
Processing	n/a
Data	n/a

Table 4-1: UI Forms

#### 4.4.3. Input Event Handler

Identification	Input Event Handler is the backend component of UI Module
Type	Input Event Handler is the backend sub-system of UI Module
Purpose	Purpose of Input Event Handler is to separate input handling from User Interface. This is responsible to deliver the events and user interactions from UI to validator which will in turn update the controller for further processing
Function	This component of UI system handles all the inputs from user. User inputs include data entry and button presses. This layer will deliver the events to validator
Subordinates	Input Event Handler is based upon the input events created by user. This part of UI Module is responsible to enable and disable certain components and retrieve data form the components and pass it to validation module for processing
Dependencies	This component is dependent on UI Events. Whenever user performs any action this component will receive some event (e.g. mouse click)
Interfaces	This component will receive events and pass them directly as data and function calls to the input validator and processing unit. This component has “has a” relation with input validator component.
Resources	n/a
Processing	Events have specific information about user input/interaction. Based upon the input location, this information will be extracted and might be used to enable/disable certain other components.



Data	Event data is the input data, hence all the user entered data is transferred from this component to validator component.
------	--

Table 4-2: Event Handler

#### 4.4.4. Input Validator

Identification	Input validator is the component and a sub-system of Business Logic module
Type	A component of Business Logic module
Purpose	Input validator component of Business Logic is related to validation of user input and passing the input after validation to appropriate layer
Function	This component of IDMS system is responsible for most of the possible user related errors. This module takes care of any input that is wrong or missing and lets the user know to correct their actions before further processing.
Subordinates	This component is based upon the methods specifically related to functionality of IDMS and validates input according to the processing required for that particular input
Dependencies	This component is dependent on user input coming from Event Handler component's raw input
Interfaces	Input validator is composed on simple checks and validation procedures which may include further processing which will be done at current level and once information is validated its passed to processing component
Resources	n/a
Processing	Processing may include string matching, parsing the input and checking against particular regular expression the format of input (e.g. email)
Data	Data will contain following: <ul style="list-style-type: none"> <li>• String</li> <li>• Date</li> <li>• Integers</li> <li>• Complex integral strings (further parsing required)</li> </ul>

Table 4-3: Input Validator

#### 4.4.5. Data Processing

Identification	Data Processing component is the controller of IDMS
----------------	---

Inventory Database Management System

Type	Data Processing is the sub system of Business logic layer of IDMS
Purpose	Data Processing layer is responsible to perform all the processing unique to IDMS. It is the data store communicator hence all data flows through this component.
Function	Input from input validator will be further processed for appropriate results in this component. This also handles if there is any validation that was not done on validation component level.
Subordinates	Component comprises with database libraries and connection objects. MSSQL interfaces are used to communicate with data store
Dependencies	Depends upon the successful connection to the database
Interfaces	MSSQL drives: libraries responsible for communication to the database/data store. Build in ADT and data structures.
Resources	n/a
Processing	All the data processing is done based on user input. Processing includes fetching data, updating data, parsing data and pushing the data and results to UI Module for user to view.
Data	Data includes: <ul style="list-style-type: none"> <li>• Database search results</li> <li>• Database update results</li> <li>• String</li> <li>• Complex data structures</li> <li>• Integers</li> </ul>

Table 4-4: Data Processing

## 4.5. Detailed Description of System

### 4.5.1. Use Cases

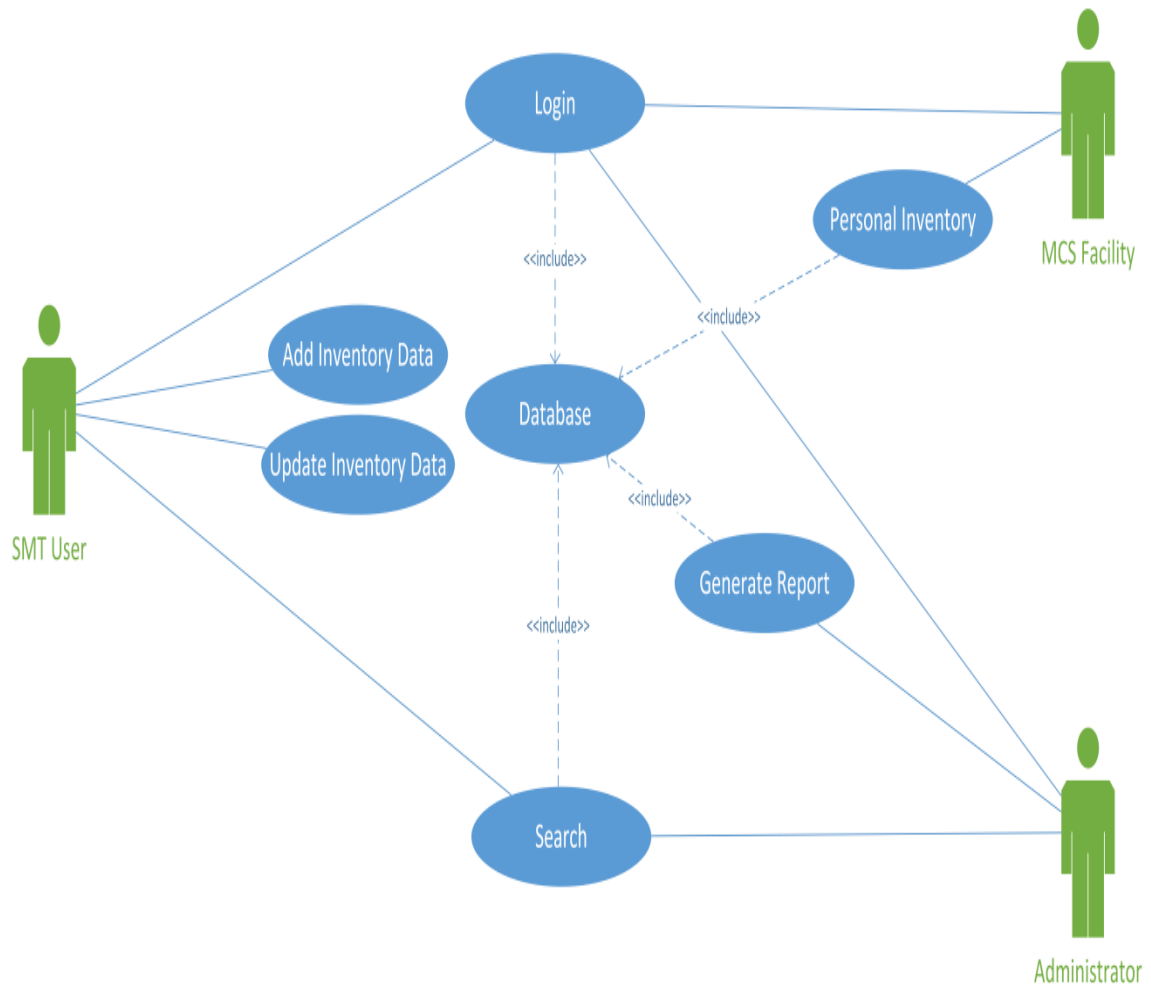


Figure 3: Use Case Diagram

**Data Entry**

This use case covers the first system feature of data entry into the data store/database shown in Fig 3.2.1. Following are the details related to this particular use case.

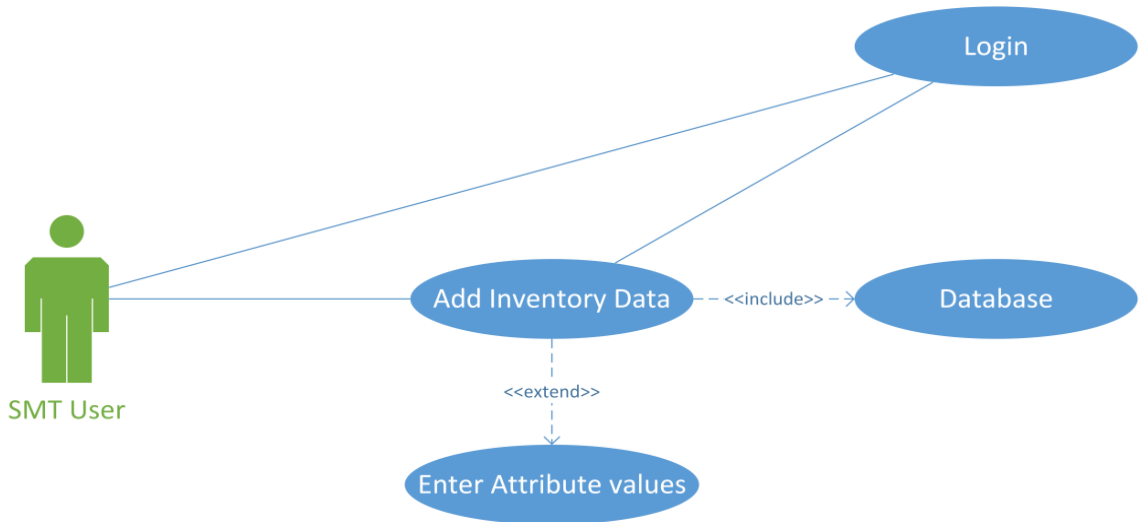


Figure 4: Use Case-Data Entry

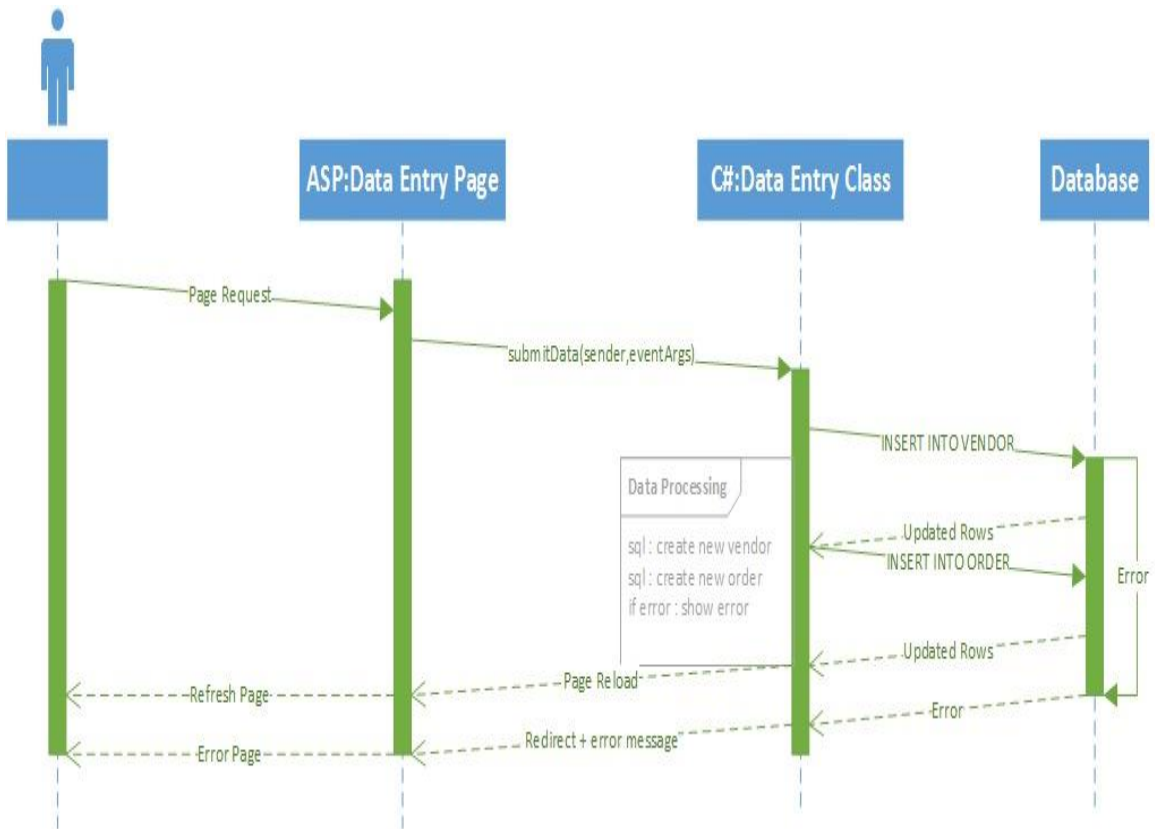


Figure 5: Sequence Diagram-Data Entry

Use Case Name	Data Entry
Actor(s)	SMT User
Pre-Condition	User has a stable network connection and User is logged in as SMT user
Normal Course	User enters order information and item information that belong to this particular order and presses the Add button. User is notified that entry has been added successfully
Post-Condition	User is taken to main menu
Alternate Course	If user doesn't enter any required field value or database connection is not online error message is shown.
Post-Condition	User is taken back to entry form
Assumptions	Nil
Priority	High
Frequency	Low

Table 4-5: Use Case-Data Entry

**Database search**

This use case covers the second feature of database with general search. This search can be done by SMT user and/admin as well shown in Fig 3.2.2. The detailed description is in the table below.

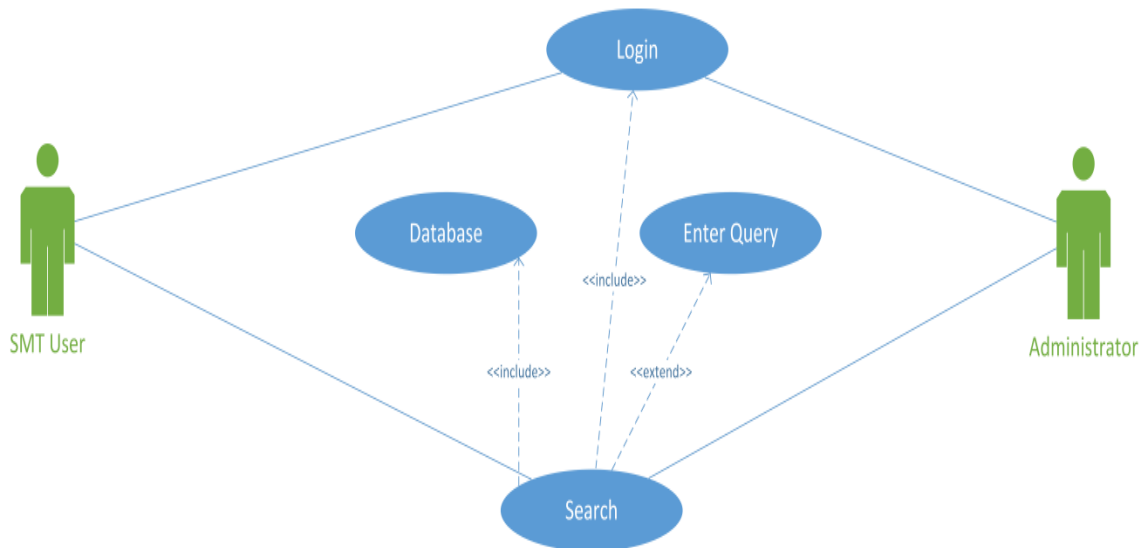


Figure 6: Use Case-Data Search

## Inventory Database Management System

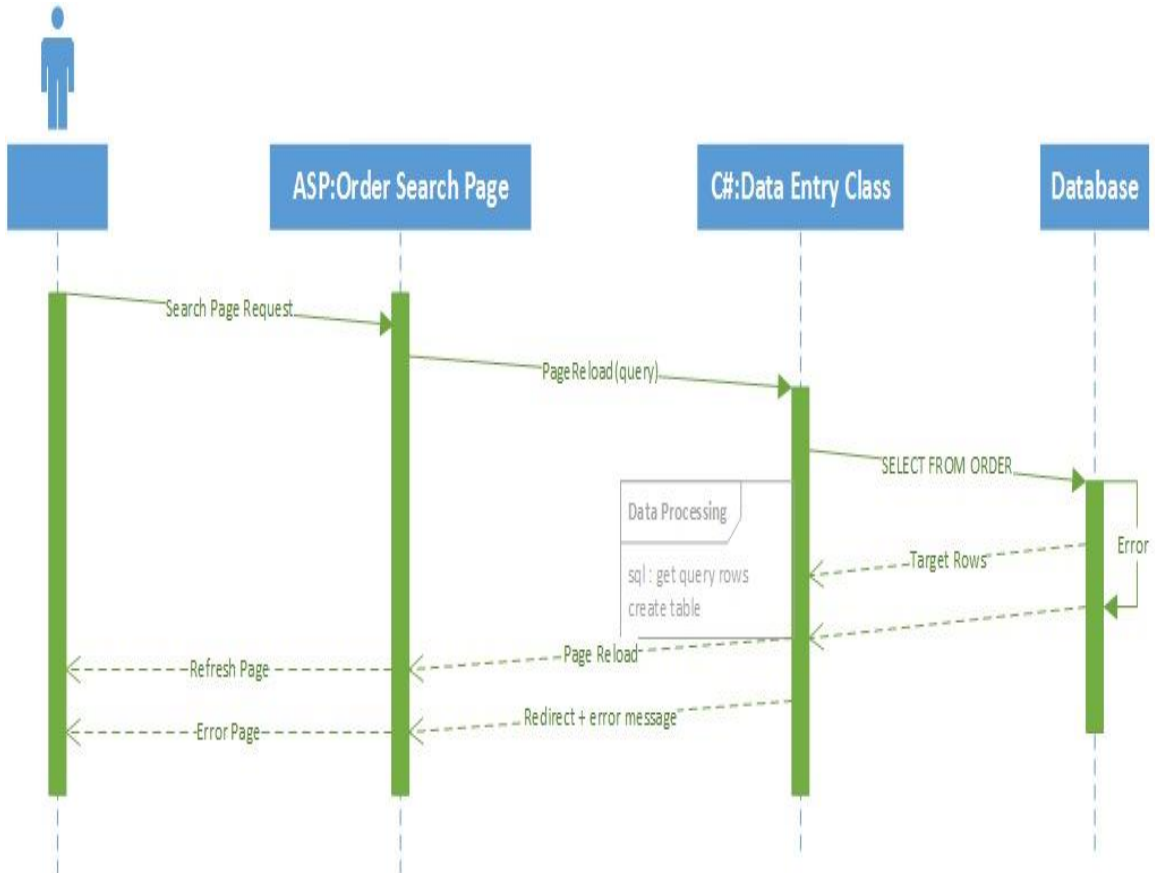


Figure 7: Sequence Diagram- Data Search

Use Case Name	Database Search
Actor(s)	SMT User Administrator
Pre-Condition	User has a stable network connection and User is logged in as SMT user or Admin
Normal Course	User enters the search query, then selects the date if needed, then selects the search type and presses the search button
Post-Condition	User is shown the data via a table that resulted with given query
Alternate Course	If no corresponding data exists or database is not connected an error message is shown indicating the issue
Post-Condition	User is taken back to search screen
Assumptions	Nil
Priority	Medium

Frequency	Medium
-----------	--------

Table 4-6 Use Case- Data Search

**Data Update**

This use case covers the third feature of IDMS which is to be able to update the inventory data. This case is particularly important for facility when some items of inventory are taken back from some place and transferred to other place/persons. Case is shown in Fig 3.2.3. Details are in following table.

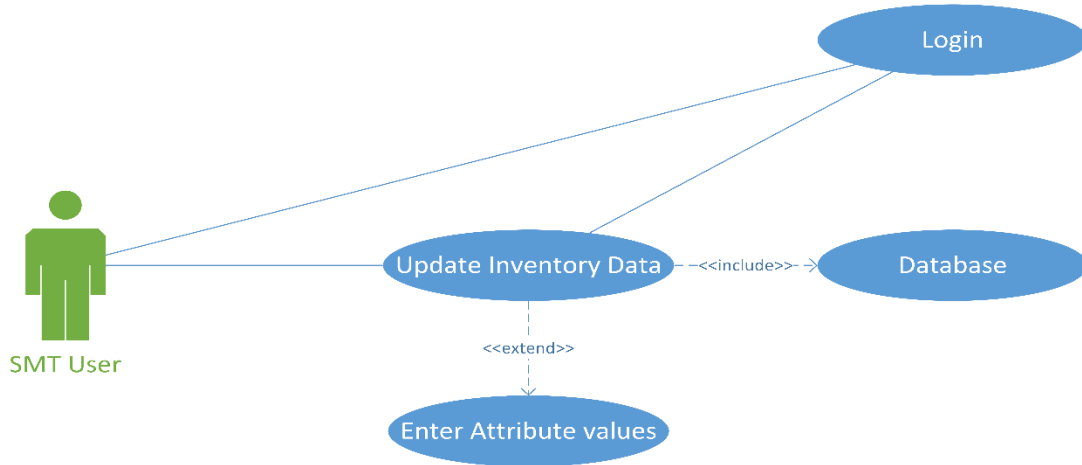


Figure 8: Use Case Diagram: Data Update

Use Case Name	Data Update
Actor(s)	SMT User
Pre-Condition	User has a stable network connection and User is logged in as SMT user
Normal Course	User enters the correct id of target data entry and update information in the forum and hits the update button
Post-Condition	User is shown the success message and returns to main menu
Alternate Course	User enters incorrect id or is not connected to database, is shown the error message indicating the issue
Post-Condition	User is taken back to update screen
Assumptions	Nil
Priority	High
Frequency	Low

Table 4-7: Use Case-Data Update

### Generate Report

One of the most important features of our system is to be able to generate reports. Reports are a critical part of operation of IDMS which help to visualize and export certain information from IDMS. Case is shown in Fig 3.2.4. Details are in following table.

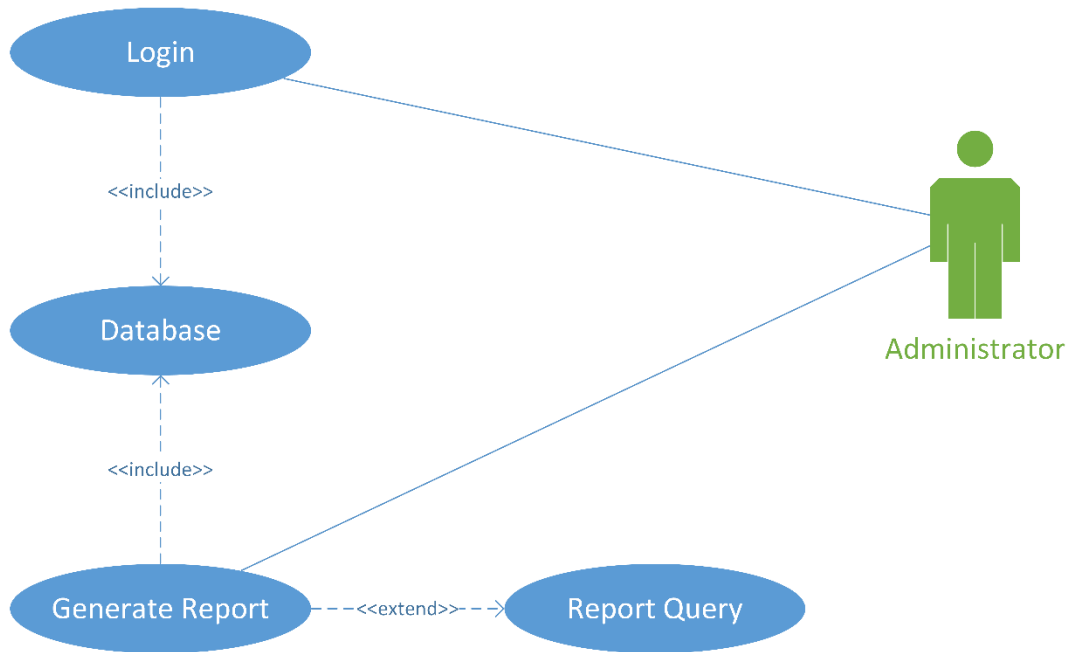


Figure 9: Use Case- Reports

Use Case Name	Generate Report
Actor(s)	Admin
Pre-Condition	User is logged in as administrator and has a stable network connection
Normal Course	User enters the search query, selects the search type, sets the date interval and hits the generate report button
Post-Condition	Report is generated, exported as file and shown to the user in table format
Alternate Course	User enters invalid/null query and/or selects invalid report type and/or selects invalid date interval and hits the search button and/or no data is found, a message regarding the issue is shown to user
Post-Condition	User is taken back to report generation screen



Inventory Database Management System

Assumptions	Nil
Priority	High
Frequency	Medium

Table 4-8: Use Case-Reports

### 4.5.2. Activities

#### Add New

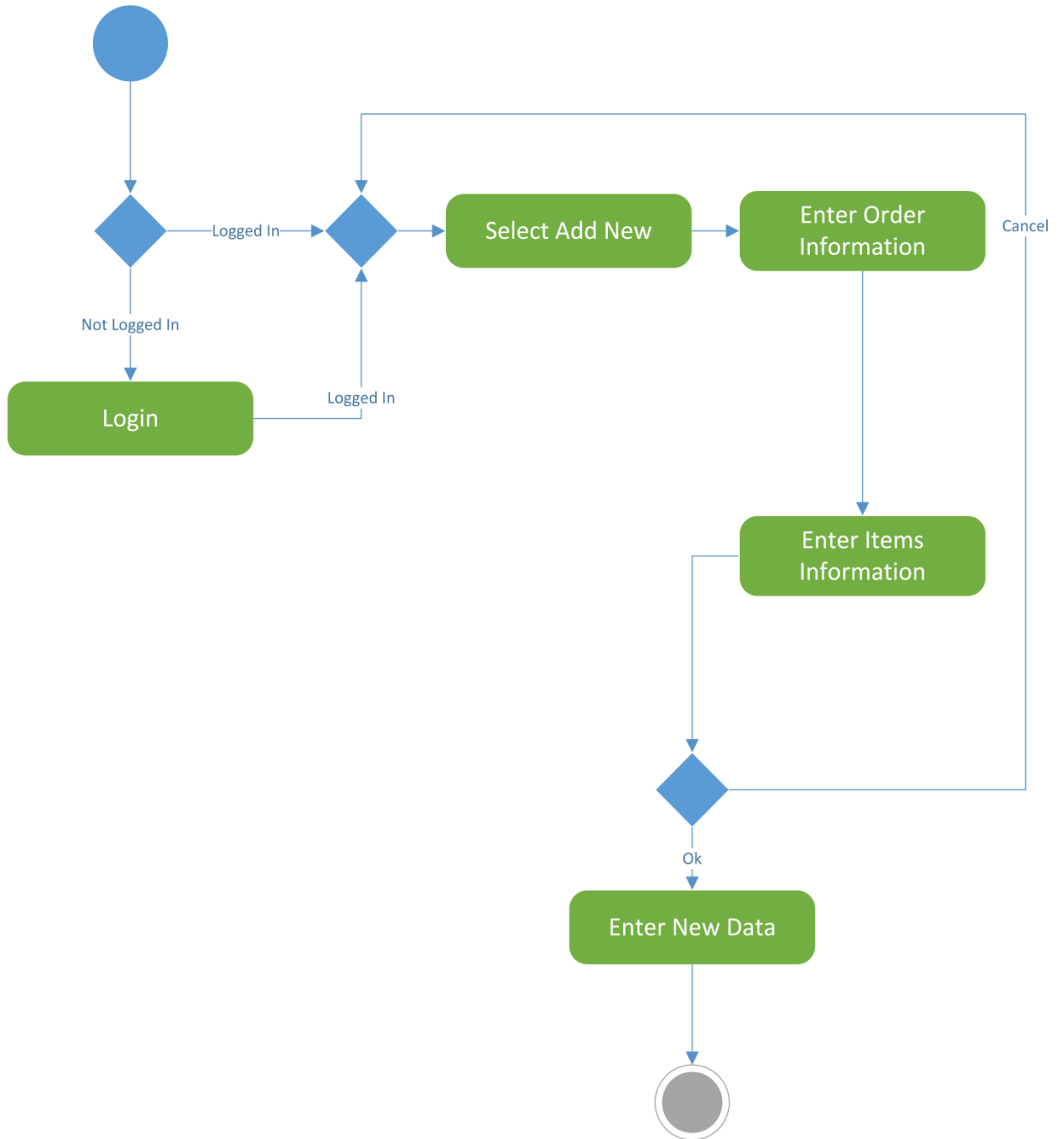


Figure 10: Activity Diagram-Data Entry

**Update**

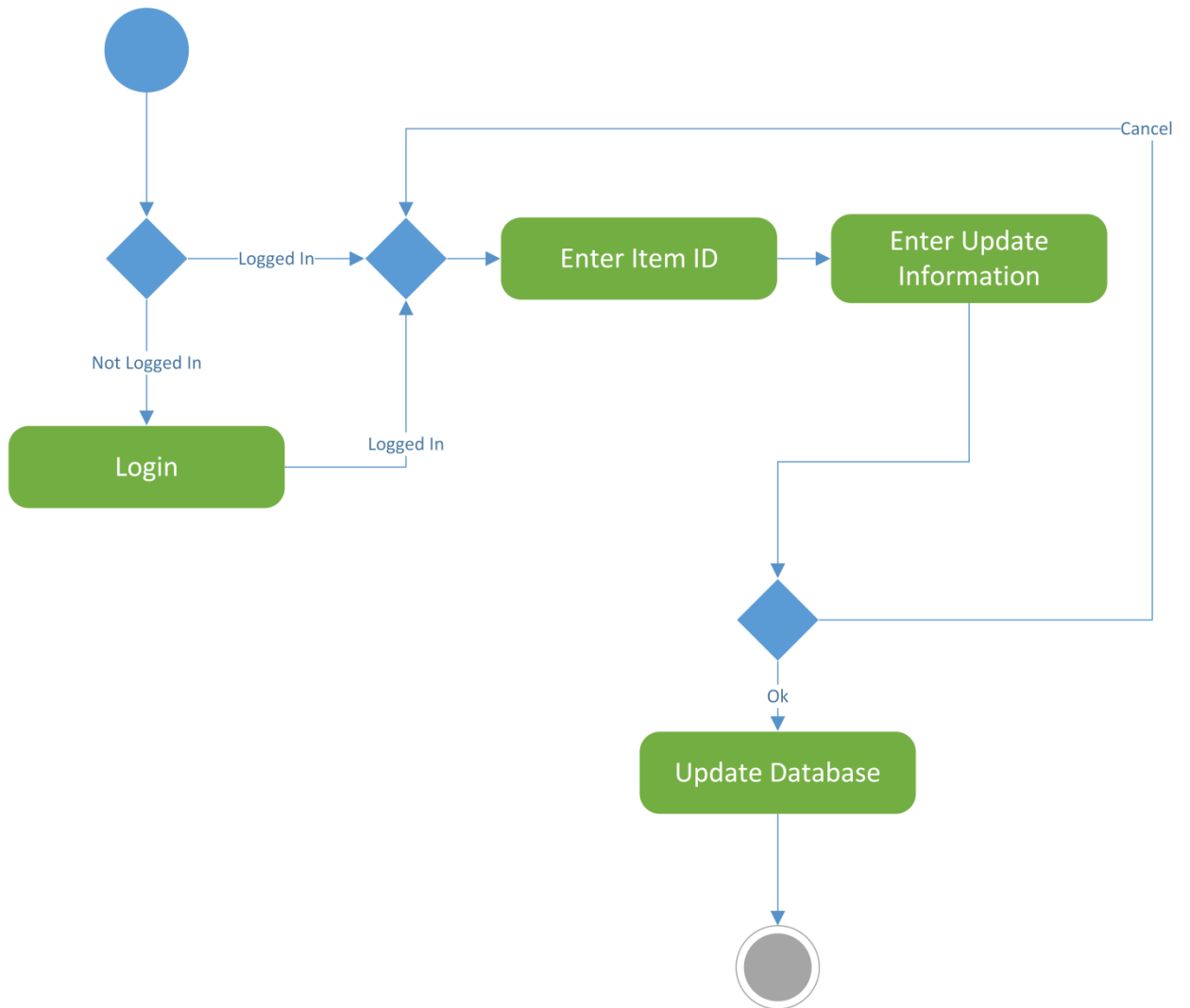


Figure 11: Activity Diagram-Update Database

**Search**

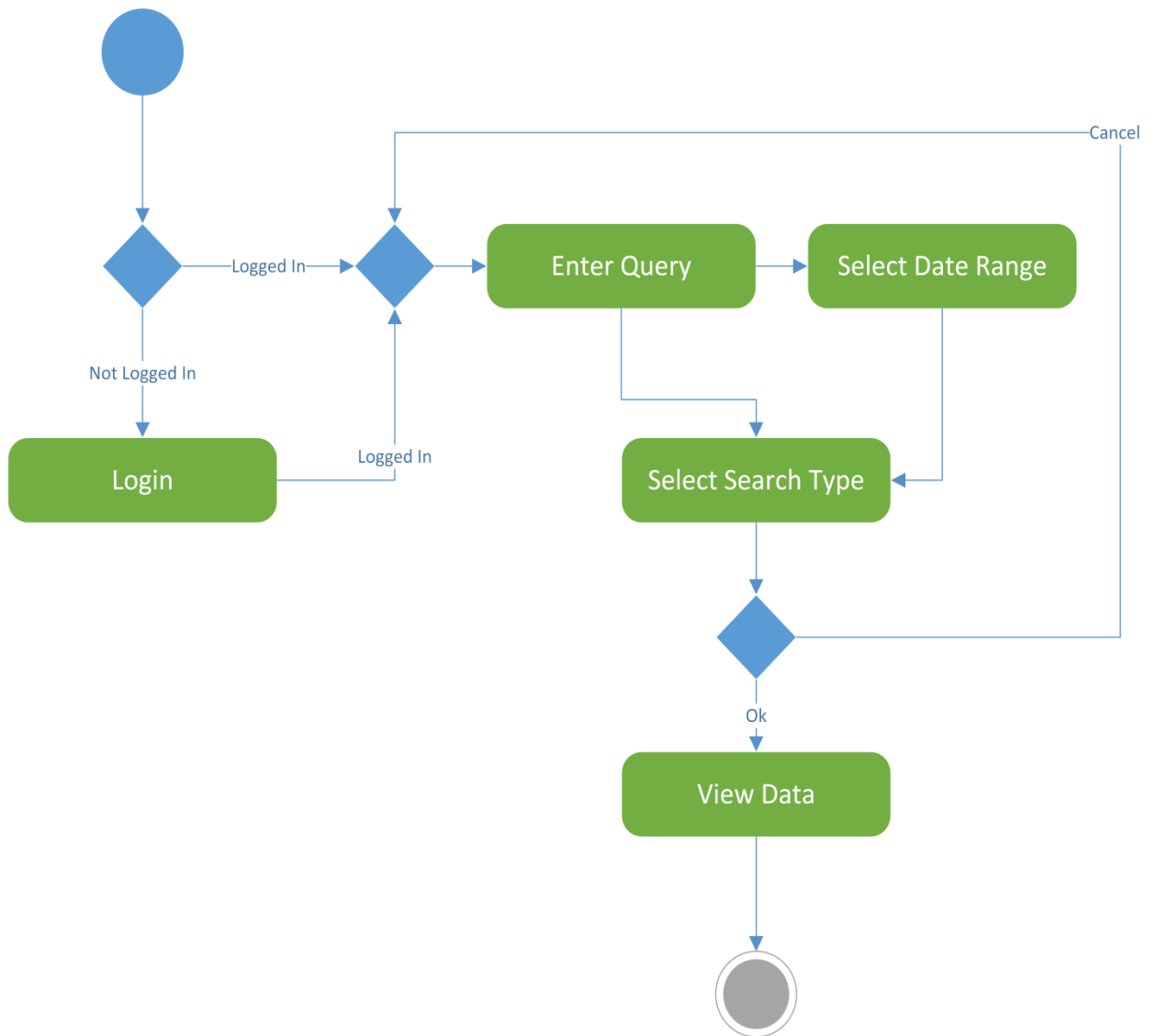


Figure 12: Activity Diagram-Data Search

## **Chapter 5. Project Test and Evaluation**

### **5.1. Introduction**

This test plan document describes the appropriate process and methods used to plan testing of the Inventory Database Management System. The test plan will ensure that Inventory Database Management System works as intended without any failure.

We are performing manual testing of the project i.e.; no tool or script is being utilized for testing purpose. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behavior or bug. First of all, each unit (module) will be tested separately and then whole project will be tested after integration of all the units.

Software testing, depending on the testing method employed, can be implemented at any time in the development process. However, most of the test effort occurs after the requirements have been defined and the coding process has been completed.

This document includes the plan, scope, approach and procedure of Inventory Database Management System. The pass/fail criteria of the test items are also defined. The Test Plan document documents and tracks the necessary information required to effectively define the approach to be used in the testing of the product.

### **5.2. Test Items**

Based on the requirements of Inventory Database Management System, design description, modules of web application, sql database and non-functional scenario will be tested. The requirements defined in Software Requirements Specification and the design entities as explained in Software Design Document will be tested.

#### **5.2.1. Features to be Tested**

Following features are to be tested:

- Ability to Add inventory data in database
- Ability to Add disposal data in database
- Ability to Filter disposal data in database
- Ability to Search database through various filters
- Ability to Select multiple filters for searching database

### **5.3. Approach**

The project is using waterfall approach, with producing modules and integrating them. All the modules are tested individually and then integrated with system and integration tests are applied.

Black Box testing technique will be used for testing functionality of each module.

#### **5.3.1. Unit Testing**

Unit testing is that part of testing which requires a thorough check of each module of the project. In our project, there are 6 modules which we have to check if they are functioning normally or not. For this we will start from the module which is least dependent on other modules for its functions and then work our way through to the module which is dependent on other modules to function and test.

#### **5.3.2. Integration Testing**

Integration testing is the part where we will test all the previous tested modules in a way that they are functioning normally when they are combined together.

#### **5.3.3. System Testing**

In the end, system testing will ensure that all the modules are working, separately and together combined. Then only the final outcome of the program will decide the correctness of whole system.

### **5.4. Item Pass/Fail Criteria**

Details of the test cases are specified in the section Test Deliverables. Following the principles outlined below, a test item would be judged as pass or fail.

- Preconditions are met
- Inputs are carried out as specified
- The actual output is as specified in the expected output for the test to be passed
- The system does not work or the actual output is different from the expected output for the test to be failed

### **5.5. Suspension Criteria and Resumption Requirements**

Testing will be suspended on the occurrence of any defect/bug in system which cannot allow further testing of the system. Testing will be resumed when the defect/bug is removed.

## 5.6. Test Deliverables

Following are the test cases performed on the system for testing purpose:

### 5.6.1. Test Case 1

Test Case Name	Add Inventory Data
Test case no	1
Description	This module will add inventory data to the database. All the fields should be filled. If a field is missed, system will not add data to the database and will ask to fill the empty fields.
Testing Technique Used	Black Box testing
Preconditions	Database should be running
Input Values	Fill all the empty fields
Steps	<ul style="list-style-type: none"> <li>• Fill all the empty fields</li> <li>• Click “Submit” button</li> </ul>
Expected Output	Data will be stored in the database
Actual Output	Data is stored in the database
Status	Pass.

Table 5-1: Test Case- Add Data

## Add Inventory Data.

### Add New Inventory Order

Enter order information below

**Details**

Hp i7 CPU

**Unit Price**

50000

**Vendor**

Alpha Computers

**Account Head**

HOD

**Minute Sheet Number**

12

Submit

**Category**

Computer

**Department**

CS

**Vendor Contact**

0321-1234567

**Ledger Reference**

22

**CRV Reference**

18

**Quantity**

50

**Location**

OS Lab

**A/U**

1

**PO Reference**

3

**CRV Date [dd-mm-yyyy]**

06-08-2016

Figure 13: Data Entry



**5.6.2. Test Case 2**

Test Case Name	Add Disposal Data
Test case no	2
Description	This module will store the data of the disposed items. CIV No and CIV Date are required fields. Remarks is optional field. Required fields must be filled. Error is shown if required fields are not filled
Testing Technique Used	White box testing
Preconditions	Data should be present in the database
Input Values	<ul style="list-style-type: none"> <li>• CIV No will be entered in numeric format</li> <li>• CIV Date will be entered in date format (dd-mm-yy)</li> </ul>
Steps	<ul style="list-style-type: none"> <li>• All the required fields are filled</li> <li>• Click “Add Disposal” button</li> </ul>
Expected Output	Data with entered CIV No and CIV Date will be selected for disposal
Actual Output	Data with entered CIV No and CIV Date is selected for disposal
Status	Pass.

Table 5-2: Test Case-Add Disposal Data

Inventory Management   Add Purchase Order   Add Disposal Order   Search   Register   Log in

### Add Disposal

#### Add New Disposal

Total selected items

**Enter CIV Number**

**Enter CIV Date**

**Remarks (optional)**

**Add Disposal**

#### Add New Filter

#### Filters

**Select filter type**

**Enter value to filter**

Figure 14: Disposal Data Entry

**5.6.3. Test Case 3**

Test Case Name	Disposal Order Data Filter
Test case no	3
Description	This module will select the filters for the data from the disposal order table. Multiple filters can be added. If a filter is selected value for the filter is required to be entered.
Testing Technique Used	Black Box testing
Preconditions	Data should be present in the database to be disposed
Input Values	<ul style="list-style-type: none"> <li>• Select filter/s</li> <li>• Enter value for each filter selected</li> </ul>
Steps	<ul style="list-style-type: none"> <li>• Select filter/s</li> <li>• Enter value for each filter selected</li> </ul>
Expected Output	Data will be filtered according to the selected filters
Actual Output	Data is filtered according to the selected filters
Status	Pass.

Table 5-3: Test Case-Disposal Order Data

# Inventory Database Management System

Inventory Management   Add Purchase Order   Add Disposal Order   Search   Register   Log in

### Add New Filter

Select filter type

Account Head

Enter value to filter

### Filters

Account Head

	Details	Category	Unit Price	Quantity	CRV Number	CRV Date	Department	Location
<input type="button" value="Add"/> <input type="button" value="Exclude"/> <input type="text" value="0"/>	Hp i7 Laptop	Computer	70000	1	21	8/13/2017 12:00:00 AM	CS	HOD Office
<input type="button" value="Add"/> <input type="button" value="Exclude"/> <input type="text" value="0"/>	Hp i7 CPU	Computer	50000	50	18	6/8/2016 12:00:00 AM	CS	Os Lab

Figure 15: Search Filters

Inventory Management   Add Purchase Order   Add Disposal Order   Search   Register   Log in

### Add New Filter

Select filter type

Unit Price

Enter value to filter

### Filters

Account Head    Unit Price

	Details	Category	Unit Price	Quantity	CRV Number	CRV Date	Department	Location	Vendor
<input type="button" value="Add"/> <input type="button" value="Exclude"/> <input type="text" value="0"/>	Hp i7 Laptop	Computer	70000	1	21	8/13/2017 12:00:00 AM	CS	HOD Office	A tech

© 2017 - My ASP.NET Application

Figure 16: Multiple Filters

**5.6.4. Test Case 4**

Test Case Name	Disposal Order Data Quantity
Test case no	4
Description	This module will select the quantity of the data to be disposed. The quantity must be less than or equal to the total amount of data quantity. It cannot be zero and it is a required field
Testing Technique Used	Black Box testing
Preconditions	Data should be present in the database
Input Values	Numeric value of the quantity to be disposed
Steps	Enter value of the data to be disposed from the database
Expected Output	Data will be added in the disposal table
Actual Output	Data is added in the disposal table
Status	Pass

Table 5-4: Test Case-Disposal Order Quantity Data

# Inventory Database Management System

	Details	Category	Unit Price	Quantity	CRV Number	CRV Date	Department	Location
<input type="button" value="Add"/> <input type="button" value="Exclude"/> <input type="text" value="0"/>	Hp i7 Laptop	Computer	70000	1	21	8/13/2017 12:00:00 AM	CS	HOD Office
<input type="button" value="Add"/> <input type="button" value="Exclude"/> <input type="text" value="3"/>	Hp i7 CPU	Computer	50000	50	18	6/8/2016 12:00:00 AM	CS	Os Lab
<input type="button" value="Add"/> <input type="button" value="Exclude"/> <input type="text" value="0"/>	Tables	Furniture	5000	16	20	4/3/2017 12:00:00 AM	EE 2	CR- 40

Figure 17: Disposal Data

**5.6.5. Test Case 5**

Test Case Name	Disposal Order Search
Test Case No	5
Description	This module allows searching of the data selected for disposal purpose
Testing Technique Used	Black Box testing
Preconditions	Disposal data should be added in the database
Input Values	Click “Search Disposal Order” button
Steps	Click “Search Disposal Order” button
Expected Output	Disposal Order table will be shown with the option of searching using various filters
Actual Output	Disposal Order table is shown with the option of searching using various filters
Status	Pass

Table 5-5: Test Case-Disposal Search

**5.6.6. Test Case 5**

Test Case Name	Disposal Order Details
Test Case No	6
Description	This module allows viewing the details of the disposal order. All the inventory selected for disposal in a specific order will be shown
Testing Technique Used	Black Box testing
Precondition	Disposal order data should be present in the database
Input Values	Click “View Details” in Disposal order search table

Inventory Database Management System

Steps	Click “View Details” in Disposal order search table of the specific order to view its details
Expected Output	Details of the disposal order will be shown in a new page
Actual Output	Details of the disposal order are shown in a new page
Status	Pass

Table 5-6: Test Case-Disposal Order Details



**5.6.7. Test Case 7**

Test Case Name	Disposal Order Search Filters
Test Case No	7
Description	This feature allows searching the disposal order table through various filters. Multiple filters can be selected at the same time
Testing Technique Used	Black Box testing
Preconditions	Disposal data should be present in the disposal search table
Input Values	<ul style="list-style-type: none"> <li>• Select filter/s</li> <li>• Enter value for the filter/s</li> </ul>
Steps	<ul style="list-style-type: none"> <li>• Select filter/s</li> <li>• Enter value for the filter/s</li> </ul>
Expected Output	Data will be filtered according to selected filter/s
Actual Output	Data is filtered according to selected filter/s
Status	Pass

Table 5-7: Test Case-Disposal Search Filters

**5.6.8. Test Case 8**

Test Case Name	Data Search Filters
Test case no	8
Description	This feature allows searching the database through details. Multiple filters can be selected for data search
Testing Technique Used	Black Box testing
Preconditions	Data should be present in the database
Input Values	<ul style="list-style-type: none"> <li>• Select filter/s</li> <li>• Enter value for each selected filter</li> </ul>
Steps	<ul style="list-style-type: none"> <li>• Select filter/s</li> <li>• Enter value for each selected filter</li> </ul>
Expected Output	Data will be filtered through the database according to the selected filter/s
Actual Output	Data is filtered through the database according to the selected filter/s
Status	Pass.

Table 5-8: Test Case-Data Search Filters

# Inventory Database Management System

## Search

### Add New Filter

Select filter type

Select filter

Enter value to filter

Add Filter

### Filters

Details	Category	Department	Vendor_Name	Vendor_Contact	Leasing	Price	Quantity	Equipment_Purchase_Date	Account_Head	CRV_No	CRV_Date	Ledger	Minutesheet
Hp i7 Laptop	Computer	CS	A tech	0304-1234567	HOD Office	70000	1	3/16/2017 12:00:00 AM	HOD	21	8/13/2017 12:00:00 AM	23	1
Hp i7 CPU	Computer	CS	Alpha Computers	0321-1234567	Os Lab	50000	50	3/16/2017 12:00:00 AM	HOD	18	6/8/2016 12:00:00 AM	22	12
Tables	Furniture	EE 2	AM Furnitures	0331-1234567	CR- 40	5000	16	3/16/2017 12:00:00 AM	HOD	20	4/3/2017 12:00:00 AM	34	2
Benches	Furniture	EE 2	AM Furnitures	0331-1234567	CR- 40	4000	16	3/16/2017 12:00:00 AM	HOD	20	4/3/2017 12:00:00 AM	34	2
Acer			Pacific		Computer						5/9/2016		

Figure 18: Database Search

**5.6.9. Test Case 9**

Test Case Name	Search filter selection
Test case no	9
Description	This feature allows the user to select multiple filters for searching the database. Value for the selected filter is required to search
Testing Technique Used	Black Box testing
Preconditions	Data should be present in the database
Input Values	<ul style="list-style-type: none"> <li>• Select filter type</li> <li>• Enter value for the selected filter</li> <li>• If needed select another filter and follow the same procedure</li> </ul>
Steps	<ul style="list-style-type: none"> <li>• Select filter type</li> <li>• Enter value for the selected filter</li> <li>• If needed select another filter and follow the same procedure</li> </ul>
Expected Output	Filter/s will be selected with its value for searching through the database. Filtered search result will be shown
Actual Output	Filter/s will be selected with its value for searching through the database. Filtered search result will be shown
Status	Pass.

Table 5-9: Test Case-Filter Selection

# Inventory Database Management System

## Search

### Add New Filter

Select filter type

Category

Enter value to filter

Computer

Add Filter

### Filters

Category
x

Details	Category	Department	Vendor_Name	Vendor_Contact	Leasing	Price	Quantity	Equipment_Purchase_Date	Account_Head	CRV_No	CRV_Date	Ledger	Minutesheet
Hp i7 Laptop	Computer	CS	A tech	0304-1234567	HOD Office	70000	1	3/16/2017 12:00:00 AM	HOD	21	8/13/2017 12:00:00 AM	23	1
Hp i7 CPU	Computer	CS	Alpha Computers	0321-1234567	Os Lab	50000	50	3/16/2017 12:00:00 AM	HOD	18	6/8/2016 12:00:00 AM	22	12
Acer Monitors	Computer	EE	Pacific Computers	0325-1234567	Computer Lab 2	7500	50	3/16/2017 12:00:00 AM	HOD	9	5/9/2016 12:00:00 AM	28	8
A4 tech Mouse	Computer	EE	Pacific Computers	0325-1234567	Computer Lab 2	250	50	3/16/2017 12:00:00 AM	HOD	9	5/9/2016 12:00:00 AM	28	8
Keyboard	Computer	EE	Pacific Computers	0325-1234567	Computer Lab 2	500	50	3/16/2017 12:00:00 AM	HOD	9	5/9/2016 12:00:00 AM	28	8

Figure 19: Database Search Filters

## Search

### Add New Filter

Select filter types

Details

Enter value to filter

Add Filter

### Filters

Category x

details x

Details	Category	Department	Vendor_Name	Vendor_Contact	leasing	Price	Quantity	Equipment_Purchase_Date	Account_Head	CRV_No	CRV_Date	Ledger	Minutesheet
Hp i7 CPU	Computer	CS	alpha Computers	0321-1234567	bone Lab	50000	50	3/16/2017 12:00:00 AM	HOD	18	8/6/2016 12:00:00 AM	22	12

Figure 20: Multiple Database Search Filters

## **5.7. Responsibilities, Staffing and Training Needs**

### **5.7.1. Responsibilities**

All developers of the project are responsible for the completion of all units testing and integration testing tasks.

### **5.7.2. Staffing and Training Needs**

Basic knowledge of testing strategies and techniques is needed for the testing of project.

Techniques such as Black Box testing, integration testing should be known to developers.

All the developers will be testing each other's work and will be actively participating in the development and testing of the project simultaneously.

## **5.8. Risk and Contingencies**

Efforts have been made to remove all and every chance of failure but there are certain unpredictable factors such as network issues, corrupt input data, or system failure that may lead to some issues. Error handling will be applied more deeply to cover all these issues but unforeseen circumstances may happen.

### **5.8.1. Schedule Risk**

The project might get behind schedule. So, in order to complete the project on time, we will need to increase the hours/day.

### **5.8.2. Budget Risk**

The budget is compensated by using less costly alternatives to fit the budget requirements.

## **Chapter 6. Future Work**

This project will be used as a basis to understand and add features to make it into an even bigger and complex system, which can be modified per the institution's needs. Or it could continue growing as generic software for other universities, companies and warehouses, it can be commercialized, but first, it would need to be deployed at least here at MCS. Additional features like complex inventory details can also be included according to the needs of MCS.



## **Chapter 7. Conclusion**

To conclude, an automated inventory database management system is an essential replacement for a manual management system. The main purpose of automating the inventory system is to control the movement and storage of the products, together with the benefit of enhanced security and quicker handling. The newly created software upgrades the capabilities of the inventory management system. It can provide detailed information of the inventory available and provide the user to search through the inventory easily unlike before. All in all, the inventory database management system has become more reliable and efficient after the automation, simplifying the process for the operators, the supplier and the dealers.

## Bibliography

- <http://ieeexplore.ieee.org/document/741940/?tp=&isnumber=16016&arnumber=741940&punumber=5982>
- [https://en.wikipedia.org/wiki/Sequence\\_diagram](https://en.wikipedia.org/wiki/Sequence_diagram)
- [https://en.wikipedia.org/wiki/Block\\_diagram](https://en.wikipedia.org/wiki/Block_diagram)
- [https://en.wikipedia.org/wiki/Internet\\_of\\_things](https://en.wikipedia.org/wiki/Internet_of_things)

## **Appendix A. Glossary**

IDMS: Inventory Database Management System

MCS: Military College of Signals

MIMS2: Database of MCS

OS: Operating System

## **Appendix B: Issues/Limitations**

All possible issues have already been mentioned where required in the SRS. Any remaining ones are listed below:

1. The group shall try to match the features and NFRs as best as possible, however, like all software projects, any discrepancies are apologized for at this stage.
2. Feedback on requirements is expected from the users to help the group in improving the design and implementation of the project.