A Mini Project Synopsis on

Online Lab Deadstock Management

SE-IT Engineering

Submitted By

Adarsh Singh	**(20104080)**
Pooja Sharma	**(20104090)**
Nainisha Sharma	**(20104042)**
Aditya Waingade	**(20104099)**

Under the Guidance of

Prof. Geetanjali Kalme

DEPARTMENT OF INFORMATION TECHNOLOGY

A.P. SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic year : 2021-22

CERTIFICATE

This is to certify that the Mini Project report on **Online Lab Deadstock Management** has been submitted by **Adarsh Singh (20104080)**, **Pooja Sharma (20104090)**, **Nainisha Sharma (20104042)** and **Aditya Waingade (20104099)** who are a Bonafide students of A.P. Shah Institute of Technology, Thane, Mumbai, as a partial fulfilment of the requirement for the degree in **Information Technology**, during the academic year **2021-2022** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Prof. Geetanjali Kalme
Guide

Prof. Kiran Deshpande	Dr. Uttam D Kolekar
HOD of Information Technology	Principle

External Examiner(s)
1.
2.

Place: A.P. Shah Institute of Technology, Thane
Date:

TABLE OF CONTENTS

Chapter 1

Introduction

Now a day, it is very difficult to manage dead stock information. Data record which is stored manually in books could cause many problems if even a small mistake is made and making changes later could be problematic too. Also it is hard to store the records manually for each material as there are hundreds of records to store. If the data records are lost than its lost forever since we cannot retrieve data from lost books. It also takes more time as well as efforts of user. Online Lab Deadstock Management System is a software used to store information related to specific labs present in colleges.

Essential function of Online Lab Deadstock Management system is viewing the recorded data, deleting the unwanted data and modifying the data as the admin want. The goal of this system is to manage the Deadstock efficiently and to be able to access the data easily.

1.1. Purpose

The purpose of Online Lab Deadstock Management System is to automate the existing manual system by the help of computerized equipment's and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with. The main purpose of Online Lab Deadstock Management system is to improve the efficiency of storing the data and managing them. With this system we can eliminate the possibility of misleading data and we can have an error free, reliable, secure, and a perfectly structural data.

The user can concentrate on their other activities rather than concentrating on the record keeping. Through this system we can maintain the data records without redundant entries. We can access the required data without being distracted by the irrelevant data. Basically the project describes how to manage for good performance and better services for the clients.

1.2. Objectives

- To create user friendly environment for management of labs deadstock element in database.
- To provide platform for permanent storage of data.
- To reduce the manual work for managing the lab deadstock.
- To track all the details about Lab Deadstock.
- The access to the records will be given to admin only as the project is built at administrative end.
- Improvement in control and performance : The system is developed to cope up with the current issues and problems of library. The system can add user, validate user and is also bug free.
- To develop an application that deals with the day to day requirement of management system.
- To develop the easy management of the inventory.

1.3. Scope

The user i.e. admin will be able to managed the data from past years to present. It will also reduce the cost of storage of data and the collection procedure of data will also go smoothly.

Our project aims at deadstock management, i.e. we have tried to computerize various processes of Deadstock Management System.

- In computer system, it is not necessary to create the manifest but we can directly print it, which saves our time.
- To assist the staff in capturing the effort spent on their respective working areas.
- To utilize resources in an efficient manner by increasing their productivity through automation.
- It satisfies the user requirement
- It is easy to understand by the user and operator

- It is easy to operate
- Have a good user interface

Our project can be used not only for lab deadstock management but also in other sectors.

- Can be applied in banking and retailing sectors.
- Can be applied for different types of warehouses.
- Can be useful for chemist/drug shop managers.
- Can be useful for electricity services providers.

Chapter 2

Problem Definition

The storing of data records was maintained manually with hands and the process of keeping, maintaining and retrieving the information was very tedious and lengthy. When wanting to retrieve a particular data, the admin had to search through all the data just to find that particular data. It always takes a long time to enter the data and retrieve them. It was also difficult to find errors and it was also very difficult to update them. There would always be unnecessary consumption of time while entering the data records and retrieving the records. One more problem was to keep the records symmetric order and it was also hard to store the data records in multiple records.

Some of the problem occurred while storing the data records manually are:

- Record Lost: If the record is lost then its lost forever since we cannot find a lost book.
- Difficult to search record: When wanting to search records for particular lab, we have to look through all the records just to find that particular record.
- Space Consuming: After the number of records become large the space for physical storage of file and records also increases.
- Cost consuming: To add each record paper and pen will be needed which will increase the cost for the management of lab deadstock.

In short, there is a lot of information to be maintained and it was hard for the user to keep up with all the information while running the business. Due to this reason we have made a system in such a way that it will help the user to maintain the data without any problem.

Chapter 3

Proposed System

To overcome the drawbacks and limitations of the existing style which means recording data manually, this Online Lab Deadstock Management System is proposed. It is a very efficient web application using Java as the Front End and MySQL and DB Derby as Back End. This application is more effective for lab deadstock data management; the data is more secured and can be accessed easily.

We have created our project in such a way that that the user has the rights to add, delete and update the records as the user wants. The user needs to sign up and then Signin to access the functions in online lab deadstock management system. After following the signup and Signin process the user will be able to perform the deadstock management work.

The system reduces manually work by

- Ensuring data accuracy.
- Minimize manual data entry
- Better service
- User friendly and interactive
- Data manipulation is possible
- Eliminating data error

3.1. BLOCK DIAGRAM:

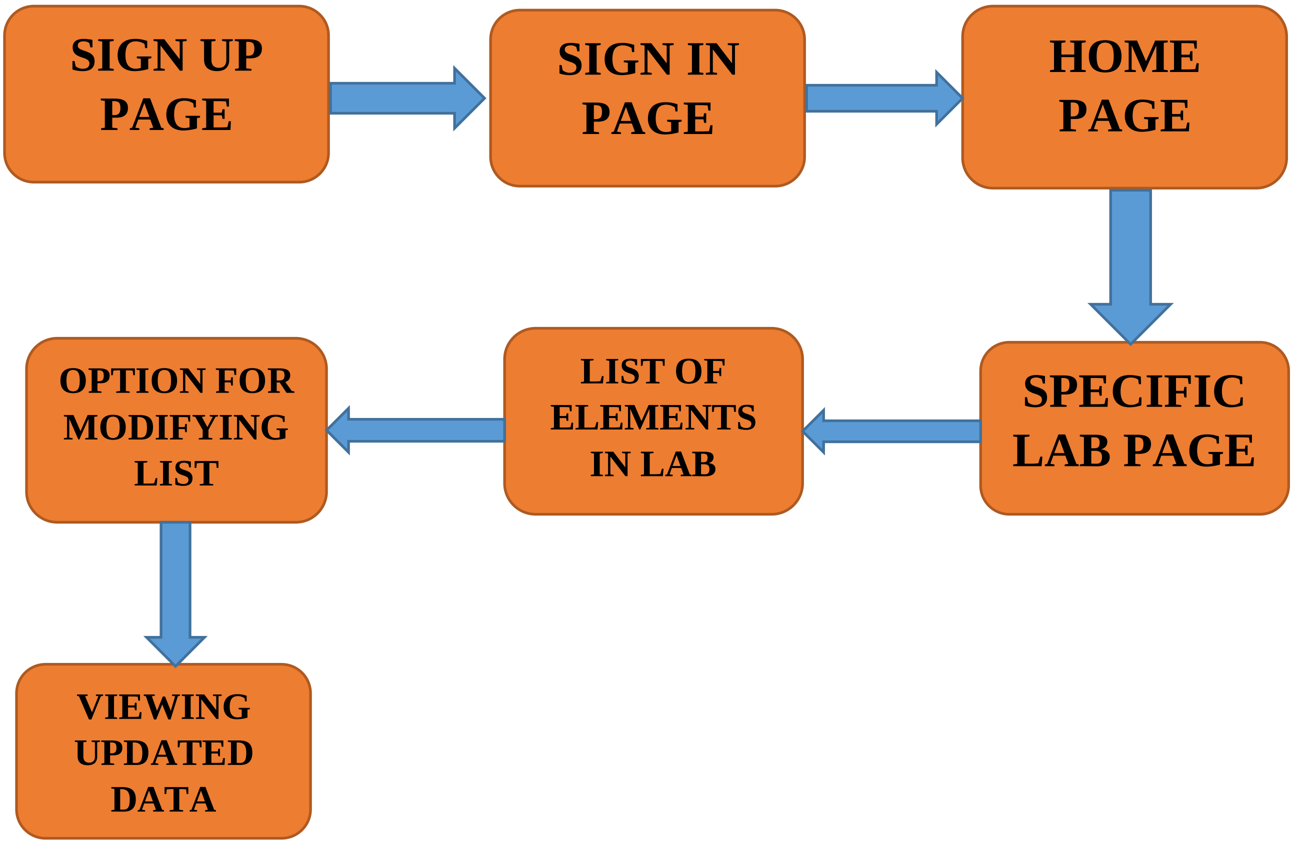

Fig 3.1. Block Diagram of Online Lab Deadstock Management

- The user will be first taken to the Signup page where he/she will be making an account for them. The input given by the user in userid and password will be save in Signup database as well as Signin database.
- After Signup process the user will see a Signin/Login page where the user need to give input which should be same as the input in Signup page otherwise an error will be shown
- After Signin page the user will be able to view different labs and by clicking a particular lab the user can enter in that particular lab deadstock page.
- After entering any lab page, the user will be able to add, delete and edit the deadstock element of that particular lab as they want.

- To go back to lab menu page, the user need to click on homepage and then the user can go to another lab from the home page.

3.1. Features and Functionality

- Laboratory deadstock elements tracking will be achieved.
- Data will be efficiently managed, stored and will also be accessed easily
- Data security will be achieved by providing access of database to user (with login ID and password)
- Only validated values will be taken and invalid will be ignored.
- Data will be kept in such a way that the happening of an error will be prevented.
- The Quantity input will only take numbers as an input and not alphabet (Varchar).
- Modification can be done if later the user wants to change something in the records.
- The user can also search for a record that they want.

CLASSES:

- Signup class: Use for managing signup details.
- Signin/Login class: Use for managing Signin/Login details.
- Menu class: Use for going to lab pages.
- AI AND MACHINE LEARNING LAB class: Use for managing deadstock related to al and machining learning.
- CLOUD COMPUTING & VIRTUALIZATION class: Use for managing deadstock related cloud computing.
- COMPUTER PROGRAMMING class: Use for managing deadstock related computer programming.
- IMAGE PROCESSING & COMPUTER VISION class: Use for managing deadstock related to image processing and computer vision.
- WEB TECHNOLOGY class: Use for managing deadstock related to web technology.

<u>Working of online lab deadstock management system:</u>

<u>1. REGISTER NEW USER</u>

Description of feature:

This feature can be performed by admin to create account.

Functional requirements

- System must be able to store information that the user gave as input.
- System must allow only if correct input is given by the user.

2. USER LOGIN:

Description of feature:

This feature is used by the user to login into system. They are required to enter user id and password before they are allowed to enter the system. The user id and password will be verified and if invalid id is there user is allowed to not enter the system.

Functional requirements:

- user id and password is used to login.
- The system must only allow user with valid id and password to enter the system. If any of the input given by the user is wrong, then the user will not be able to access the system.

3. ADDING RECORD:

Description of feature:

This feature is used by the user to add records regarding the lab.

Functional Requirement:

- User will only be allowed to enter valid data i.e. in place of integer only integer values are allowed and same for the Varchar values.
- The recorded data can later be viewed by the records.

4. DELETE RECORD:

Description of features:

This feature is used by the user to delete records regarding the lab

Functional Requirement:

- User will be allowed to delete the data by clicking on the row they want to delete and after that the user will click on delete button for deleting records.
- The records will be deleted permanently.

5. EDIT RECORD:

Description of features:

This feature is used by the user to edit records regarding the lab.

Functional Requirement:

- The system will allow the user to edit the records as the user want.
- The system will allow date input in date input and will not take any other input and same goes for the other inputs.

6. SEARCH RECORD:

Description of features:

This feature is used to search a particular record available in the deadstock elements.

Functional Requirement:

- The system will allow the user to search a particular record as the user wants.
- The system will only allow the input for search item that is available in the records otherwise it will ask the user to input the correct value.

Chapter 4

Project Outcome

- User will be able to login in database following the signup and sign in pages.
- User will be able to view, add, delete and update the data records.
- There will be only one user that will have the rights to manage the data records and that is admin.
- The user will be able to manage lab deadstock elements list easily, efficiently and economically.
- User will be able to track all the data records.
- User will be able to modify the data as he/she wants.
- For each lab user will be able to manage the data as they want.

Chapter 5

Software Requirement

The whole Project is divided in two parts the front end and the back end

- Front End: - Java (NetBeans.)
- Back End: - MYSQL, DB Derby.

I. FRONT END:

JAVA (Apache NetBeans):

Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages. As of 2019, Java was one of the most popular programming languages in use according to GitHub, particularly for client–server web applications, with a reported 9 million developers

NetBeans is an Integrated Development Environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, it has extensions for other languages like PHP, C, C++, HTML5 and JavaScript. Applications based on NetBeans, including the NetBeans IDE, can be extended by third party developers.

We have used Java (Apache NetBeans) for our project, we created a project in NetBeans in which we created a package and in package different classes has been created. For the project to run we have linked all the classes in such a way that the admin will be able to first signup and then login and after login the admin will be able to manage the deadstock as he/she want.

II. BACK END:

MYSQL:

MySQL is an open-source relational database management (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structures Query Language. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database.

In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

We have used MySQL connector to connect the java page to the database and use sql commands to connect the tables to java page

rs2xml:

rs2xml.jar is a jar file and a java library that makes JTable manipulation a bit easier. We have added this jar file for making our project more efficient.

DB Derby:

Apache Derby (previously distributed as IBM Cloudscape) is a relational database management system (RDBMS) developed by the Apache Software Foundation that can be embedded in Java programs and used for online transaction processing. Apache Derby is developed as an open source project under the Apache 2.0 license. For a time, Oracle distributed the same binaries under the name Java DB. The Java DB database is Oracle's supported distribution of Apache Derby.

The DB Derby serves as the intermediary between the user and the database. The database structure itself is stored as a collection of files, So, we can access the data in those files through the DB Derby.

We have downloaded DB Derby additionally and later use to start our database server and we have also added it in our project library.

Chapter 6

Project Design

In this phase, a logical system is built which fulfils the given requirements. Design phase of software development deals with transforming the admin's requirements into a logically working system. Normally, design is performed in the following in the following two steps:

1. Primary Design Phase:

In this phase, the system is designed at block level. The blocks are created on the basis of analysis done in the problem identification phase. Different blocks are created for different functions emphasis is put on minimising the information flow between blocks. Thus, all activities which require more interaction are kept in one block.

2. Secondary Design Phase:

In the secondary phase the detailed design of every block is performed.

The general tasks involved in the design process are the following:

1. Design various blocks for overall system processes.

2. Design various database structures.

3. Specify details of programs to achieve desired functionality.

4. Design the form of inputs, and outputs of the system.

5. Perform documentation of the design.

6. System reviews.

User Interface Design

User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overall flow of screens and messages is called a dialogue.

The following steps are various guidelines for User Interface Design:

1. The system user should always be aware of what to do next.

2. The screen should be formatted so that various types of information, instructions and messages always appear in the same general display area.

3. Message, instructions or information should be displayed long enough to allow the system user to read them.

4. Use display attributes sparingly.

5. A user should not be allowed to proceed without correcting an error.

6. The system user should never get an operating system message or fatal error.

<u>Various Table are made in database to maintain records</u>:

- SIGNUP Table:
  In this table the information regarding userid, password and confirm password are stored.
- SIGNIN/LOGIN Table:
  In this table, the information we store in signup table will also be stored here and to go further we need to give the input of userid and password same as the input in signup table.
- AI AND MACHINE LEARNING LAB Table:
  In this table, we store the deadstock records as we want. The columns in the table are item, name of supplier, bill. No, bill date, quantity, rate/unit, cost, date of delivery, date of installation and identity no. We have kept bill.no as primary key and the format of date will be "yyyy-mm-dd".
  The Tables for other Labs are the same as the Al and Machine learning lab.

<u>Various Pages are made in project:</u>

- SIGNUP PAGE:

  In this page the user will give input in text field (userid, password and confirm password) and later the input given by the user will be stored in the database.

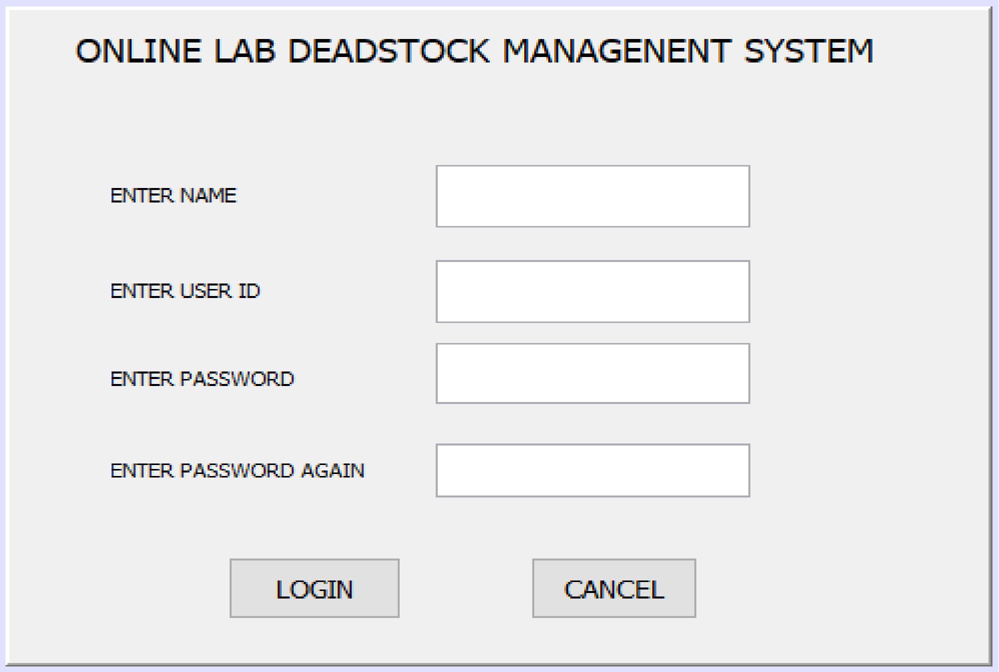

Fig.6.1. Signup Page

- SIGNIN/LOGIN PAGE:

  In this page the input user has given in signup page will be use. The user will give the input in userid and password and the input should be same as the input given in signup page by the user

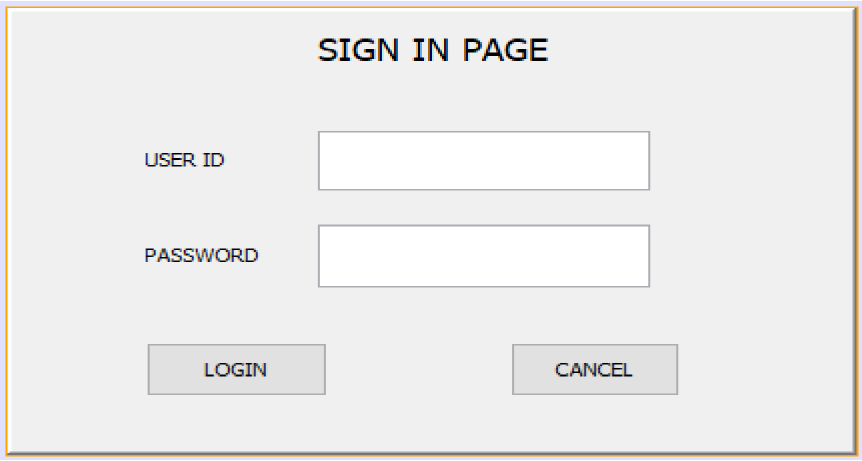

Fig.6.2. Signin Page

- MENU PAGE:

  In this page the user will be clicking on the lab that the user wants. After clicking on any lab the user will be transfer into that particular lab page.

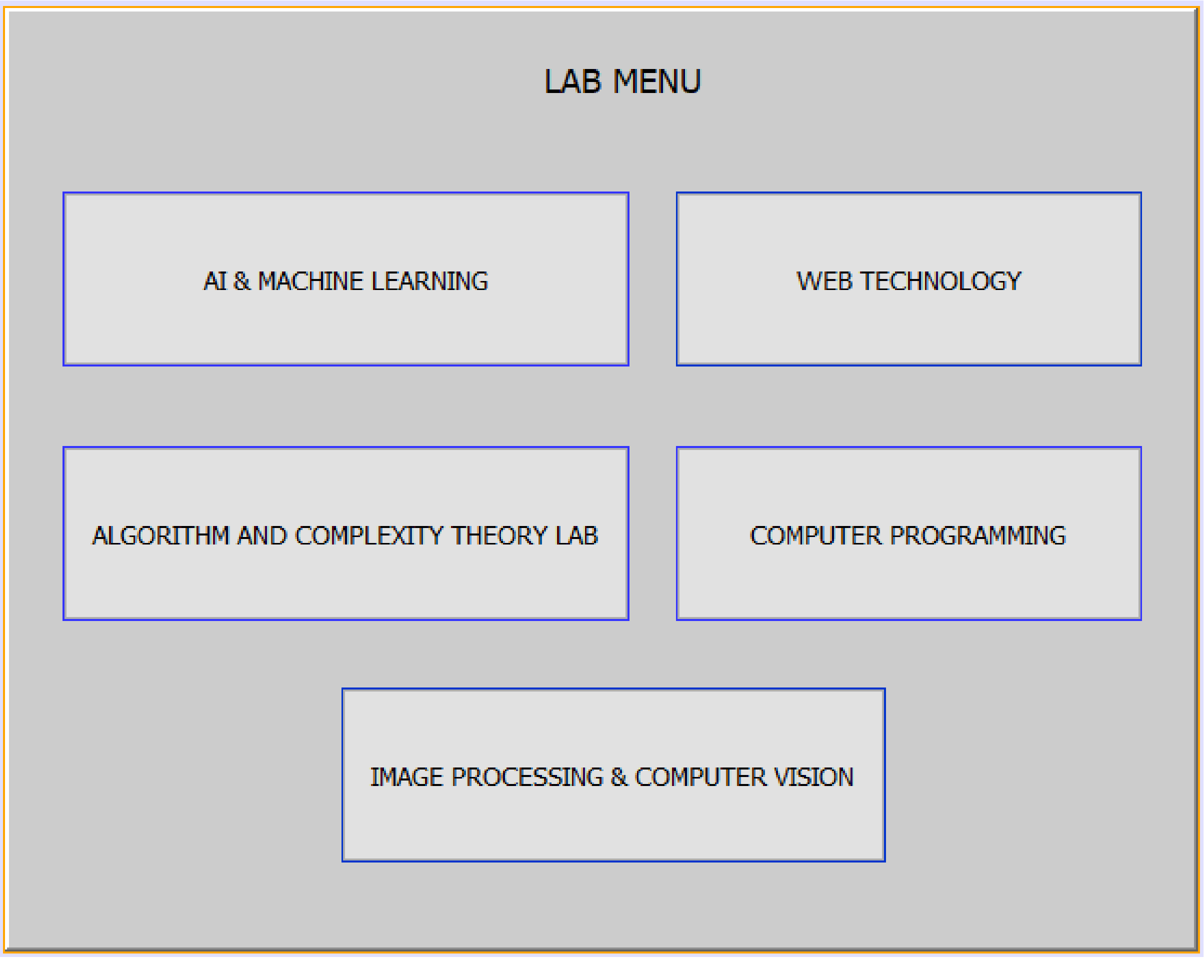
LAB MENU
AI & MACHINE LEARNING
WEB TECHNOLOGY
ALGORITHM AND COMPLEXITY THEORY LAB
COMPUTER PROGRAMMING
IMAGE PROCESSING & COMPUTER VISION

Fig.6.3. Menu Page

- AI AND MACHINE LEARNING LAB PAGE:

  In this page the user will be able to view the deadstock of the labs. The user will be able to add, delete and edit the records as the user wants. The user can manipulate the records as they want.

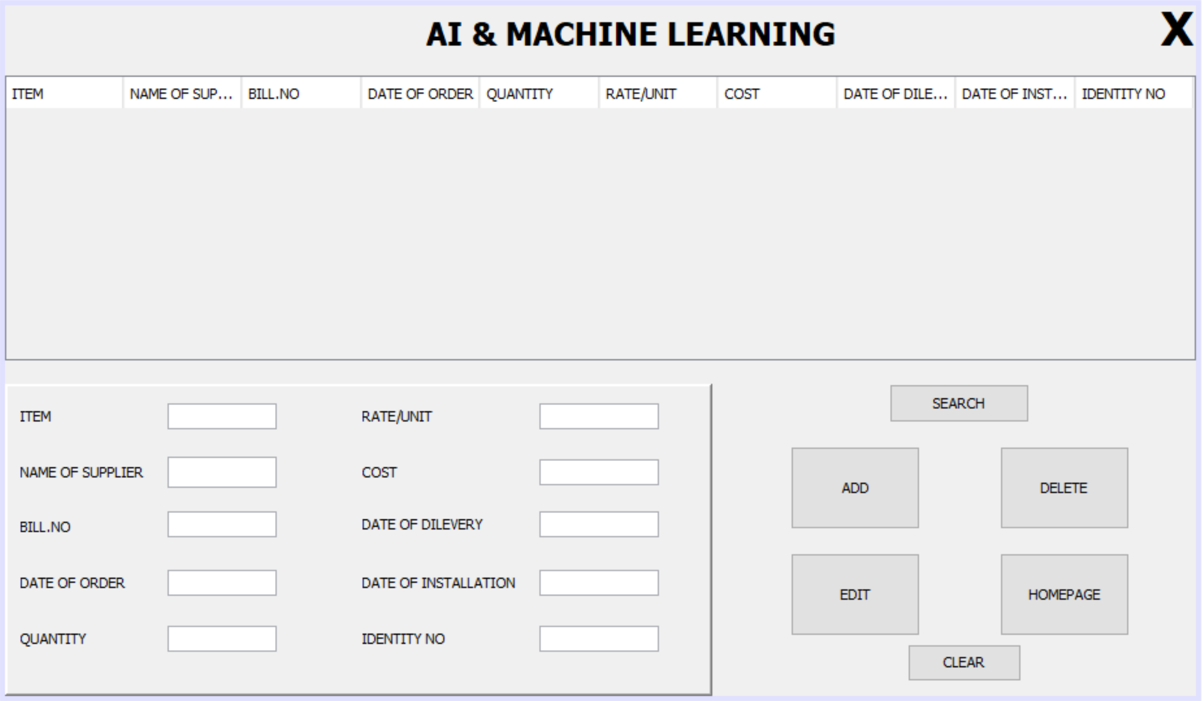

Fig.6.4. Lab Page

Chapter 7

Project Scheduling Template

Sr. No	Group Member	Time Duration	Work to be done
1	Adarsh Singh Pooja Sharma Nainisha Sharma Aditya Waingade	1st week of January	Implementing 1st module/functionality (Designing the Signup and Sign in page and after login the admin will enter the main page)
2	Adarsh Singh Pooja Sharma Nainisha Sharma Aditya Waingade	3rd week of January	Testing 1st module In main menu there will be different labs to enter. The admin will be selecting from the following options ○ Lab 1 ○ Lab 2 ○ Lab 3
3	Adarsh Singh Pooja Sharma Nainisha Sharma Aditya Waingade	1sd week of February	Implementing 2nd module/functionality (Designing the pages for individual labs and feeding the information related to the labs)

4	Adarsh Singh Pooja Sharma Nainisha Sharma Aditya Waingade	By the end of March month	Implementing 3rd module/functionality (Testing the system to have the admin view the list of recorded data related to labs.)

Chapter 8

Conclusion

Thus we have created our system in such a way that the admin can manage the data records easily and efficiently. The system is created in such a way so that it can satisfy all the requirement of the labs present in colleges. The admin will be able to manage the records in less time as compared to before when the work was done manually.

At the end it is concluded that we have made effort on following points…

- We define the problem on which we are working in the project.
- We describe the requirement Specifications of the system and the actions that can be done on these things.
- We understand the problem domain and produce a model of the system, which describes operations that can be performed on the system.
- We included features and operations in detail, including screen layouts.

.

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Ms. Geetanjali Kalme** Expressing gratitude towards our HoD, **Prof. Kiran Deshpande**, and the Department of Information Technology for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our teacher **Ms. Vidya Shet** who gave us her valuable suggestions and ideas when we were in need of them. We would also like to thank our peers for their helpful suggestions.

Reference

[1] International Journal of Advanced Research in Computer and Communication Engineering IJARCCE 10.pdf Vol. 7, Issue 2, February 2018

[2] Working with the Java DB (Derby) Database Working with the Java DB (Derby) Database (apache.org)

[3] Create Swing GUI Project and Design Jframe in Java NetBeans, Create Swing GUI Project and Design jframe in Java NetBeans - SKOTechLearn Tips, May 26, 2019

[4] Inventory Management System https://youtu.be/HfSQL2H7_mE, Nov 2020.