

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELGAVI, KARNATAKA -590 018



A Minor Project Report on

“INVENTORY MANAGEMENT SYSTEM”

Submitted in partial fulfillment of Bachelor of Engineering in Computer Science & Engineering during the academic year 2022-23.

By

Dharma Rakshith 4MN20CS014

Madanlal Chowdhary 4MN20CS025

Under the Guidance of

Prof. Hemanth C

Assistant Professor

Dept. of CS&E

MIT Thandavapura



2022-23



MIT THANDAVAPURA
COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

MAHARAJA INSTITUTE OF TECHNOLOGY THANDAVAPURA

NH 766, NANJANGUD TALUK, MYSURU – 571302

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
MAHARAJA INSTITUTE OF TECHNOLOGY THANDAVAPURA**



CERTIFICATE

*Certified that the minor project work entitled “**Inventory Management System**” is a bonafide work carried out by **Dharma Rakshith** (4MN20CS014) & **Madanlal Chowdhary** (4MN20CS025) for the course **DBMS Laboratory with Mini-Project** with course code **18CSL58** of Fifth Semester in Computer Science & Engineering under Visvesvaraya Technological University, Belagavi during academic year **2022-23**.*

It is certified that all corrections/suggestions indicated for Internal Assignment have been incorporated in the report. The report has been approved as it satisfies the course requirements.

Signature of Lab Staff In-Charge

Prof. Hemanth C
Assistant Professor
Dept. of CS&E
MIT Thandavapura

Signature of the HoD

Dr. Ranjit K N
Associate Professor & Head
Dept. of CS&E
MIT Thandavapura

External viva

Name of the Examiners

Signature with date

- 1).....
- 2).....

ACKNOWLEDGEMENT

It is the time to acknowledge all those who have extended their guidance, inspiration and their whole hearted co-operation all along our project work.

We are grateful to **Dr. Y T Krishne Gowda**, Principal, MIT Thandavapura, **Dr. H K Chethan**, Professor and Mentor, CS&E, MIT Thandavapura and also **Dr. Ranjit K N**, Associate Professor and Head, CS&E, MIT Thandavapura for having provided us academic environment which nurtured our practical skills contributing to the success of our project.

We would like to sincerely thank our project guide **Prof. Hemanth C**, Assistant Professor, Dept. of CS&E, MIT Thandavapura for providing relevant information, valuable guidance and encouragement to complete this project.

We wish to place a deep sense of gratitude to all Teaching and Non-Teaching staffs of Computer Science and Engineering Department for whole-hearted guidance and constant support without which this endeavor would not have been possible.

Our gratitude will not be complete without thanking our parents and also our friends, who have been a constant source of support and aspirations.

Dharma Rakshith

Madanlal Chowdhary

ABSTRACT

In this project, This project's main goal is to provide certain shops, stores, or any business with an online platform to manage their Inventory or Product Stock. This simple project is a Inventory Management System. This is a web-based application project developed in PHP and MySQL Database.

CONTENTS

SL. No.	Index	Page No.
1	INTRODUCTION	1-2
1.1	Project Description	1
1.2	Objectives of the project	1
1.3	Existing System	1
1.4	Proposed System	2
2	REQUIREMENTS AND ANALYSIS	3-6
2.1	Hardware Requirements	3
2.2	Software Requirements	3
2.3	Analysis	5
2.3.1	Use Case Diagram	5
2.3.2	Dataflow Diagram	6
3	SYSTEM DESIGN	7-12
3.1	E-R Diagram	7
3.2	MySQL Server	9
3.3	Normalization	11
4	IMPLEMENTATION AND RESULTS	13-16
4.1	Algorithm	13
4.2	Snapshots	14
	CONCLUSION AND FUTURE WORK	17
	REFERENCES	

LIST OF FIGURES

Fig No.	Index	Page No.
2.3.1	Use Case Diagram	5
2.3.2	Data flow Diagram	6
3.1	ER Diagram	7
3.1.1	Schema Diagram	8
3.3.1	Sales table in 1NF	12
3.3.2	Customer table in 2NF	12
4.2.1	Login	14
4.2.2	Item	14
4.2.3	Purchase	15
4.2.4	Vendor	15
4.2.5	Sales	16
4.2.6	customer	16

CHAPTER – 1

INTRODUCTION

1.1 Project Description

An inventory management system tracks inventory availability at all points of your supply chain — from purchasing, to production, and finally sales to consumers. An inventory management system is important for companies with large amounts of inventory in order to keep it at optimal levels.

1.2 Objective of the Project

This project's main goal is to provide certain shops, stores, or any business with an online platform to manage their Inventory or Product Stock. The application helps the business to easily store, retrieve, and monitor their product stock.

This Inventory Management System was develop using PHP and MySQL Database. It is a simple system created using HTML, CSS design, JavaScript and Bootstrap. This Inventory management system is all about selling and buying products. It is a simple project that monitors the stock quantity. This system has a pleasant user interface and it is user-friendly.

1.3 Existing System

In most developing countries such as ours, inventory control are still managed using a pen and paper methodology even with the advent of the age of technology and internet.

Demerits of Existing System

This can sometimes leads to inefficient handling of day-to-day business matters and can also lead to security threats to the business due to the presence of physical records. Managing and retrieving all the records are time consuming.

1.4 Proposed System

This **Simple Inventory Management System Project** is only accessible to the shop administrator. It requires the administrator system user credential to gain access to the features and functionalities. The Administrator can manage the list of Customers, Sales, Purchases and Items. Each of these features contains **CRUD** (Create, Read, Update and Delete) operations or functionalities. The system automatically computes the available stock of each product on hand.

Merits of Proposed System

The application helps the business to easily store, retrieve, and monitor their product stock. It has a simple and pleasant user interface with the help of **Bootstrap 5 Framework** and **jQuery** that also gives the management or end-users a better experience while using the application. This project is consists of user-friendly features and functionalitiesThe system generates reports for the list of Items, Sales, Vendor, Customer, and Purchases. Each of these reports can be printed and exported to CSV, Excel, and PDF.

CHAPTER – 2

REQUIREMENTS AND ANALYSIS

2.1 Hardware Requirement

RAM:

2GB of RAM is a good amount of memory for a DBMS mini project. It will allow us to easily store and manage our data, as well as run queries quickly and efficiently. Additionally, 2GB of RAM will give us enough memory to run multiple database instances simultaneously for testing purposes. Finally, 2GB of RAM will also provide enough memory for caching data, which can improve the speed of database transactions.

Hard Disk:

For a PHP and MySQL DBMS mini project, 10GB of free hard disk space should be enough as this project is simple, we application. The main components of the project that would need disk space are the MySQL database files and the PHP source code files. The disk space needed for the MySQL database files is determined by the size of the data stored in the database, while the disk space needed for the PHP source code files will be determined by the size and complexity of the code.

Processor:

In terms of processor requirements, a sufficiently powerful processor is needed to handle the load of this project. For this PHP and MySQL DBMS mini project, a processor with at least an Intel Core i3 processor or equivalent should be sufficient.

2.2 Software Requirement

Operating System: Linux/ Windows

For the PHP and MySQL DBMS mini project. This can be done with any operating system that supports PHP and MySQL, such as Windows, Linux, or Mac OS X.

Tools:

1.MySQL:

Latest version of MySQL is needed to run this project and to create the queries and create tables.

2.Browser:

For this PHP and MySQL mini project, we will need a Web browser such as Chrome, Firefox, Safari or Microsoft edge. This helps to view how the web application will look after hosting.

3.HTML:

It used to create the user interface of the project. It is used to create webpages and display data from the MySQL database. HTML is also used to make the project look visually appealing and organized.

4.CSS and JS:

CSS and JS are used for a more interactive and dynamic user experience. CSS is used to style the webpages and make them more visually appealing. JavaScript is used to add interactive elements, such as animations and user interactions. This can help make the project more engaging and easier to use.

5.XAMPP server:

It provides a local development environment for web applications.

It is an easy to install package that contains the Apache web server, MySQL database, PHP, etc. With XAMPP, we can easily develop and test our web applications on our own computer without having to upload our files to a web server.

6.PHP scripting:

It is used to create dynamic webpages and applications. PHP is a scripting language that is used to access and manipulate databases, create user interfaces, and process data. It is also used to add functionality to webpages like forms, login and registration systems, search functions, and more.

2.3 Analysis

2.3.1 Use Case Diagram

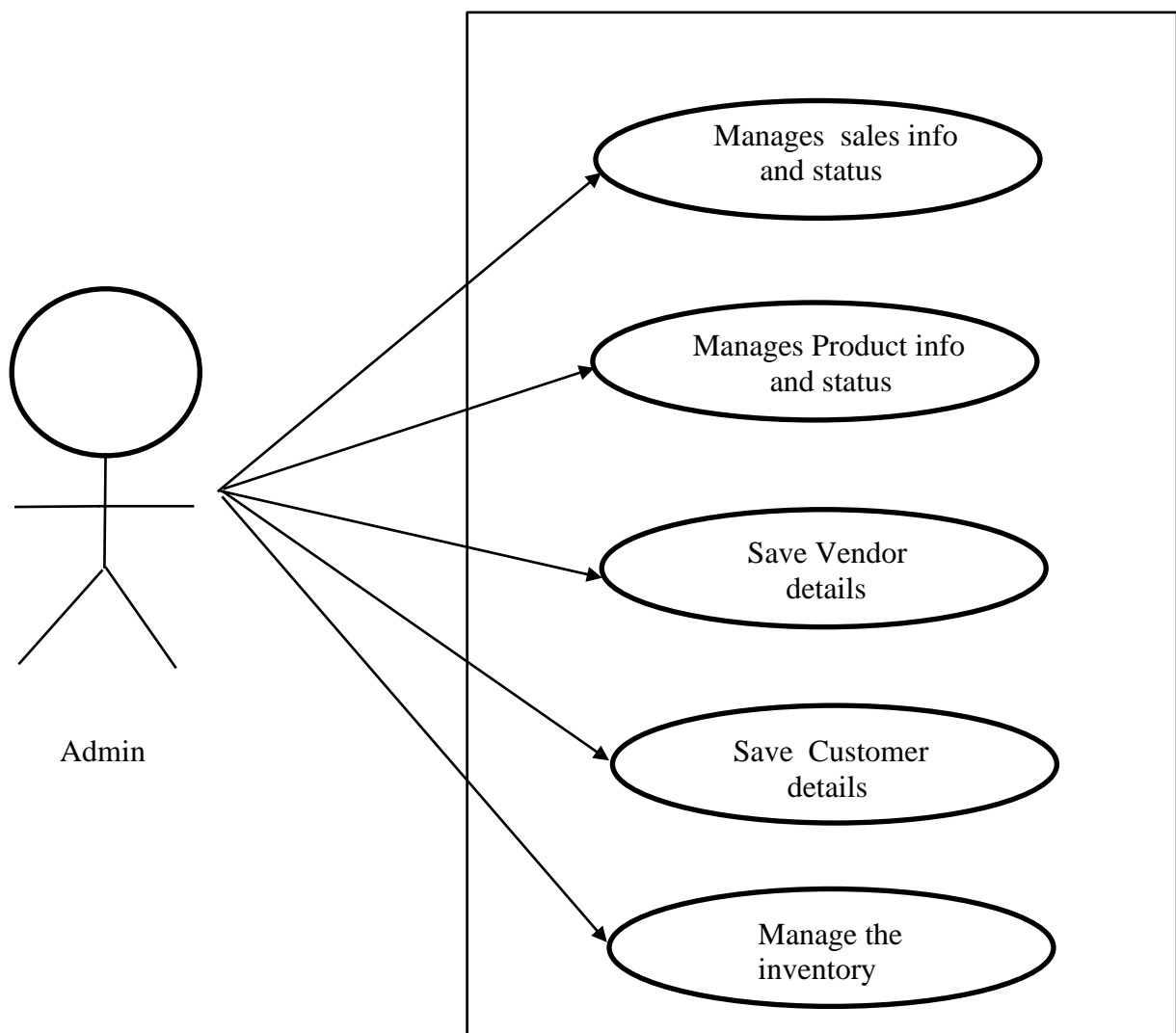


Fig 2.3.1:Use case Diagram

In the above diagram there is an actor named Admin.

Admin:

Admin can view the details of Purchase, Sales. He can add the Customer details, or remove the customer details. And he can add the details of Vendor or remove from the lists. Admin can manage the details of inventory.

2.3.2 Data Flow Diagram

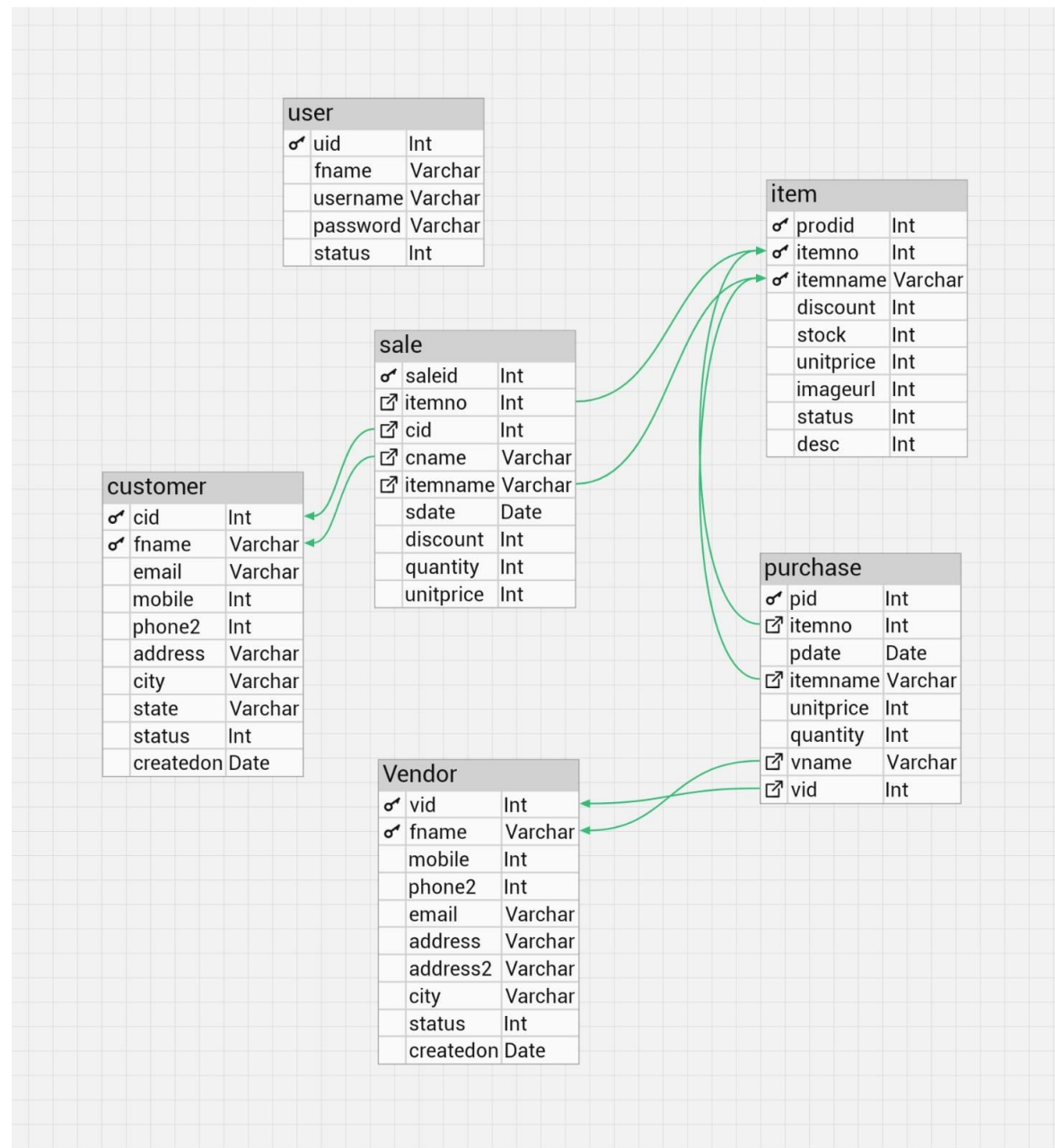


Fig 2.3.2:Data flow diagram

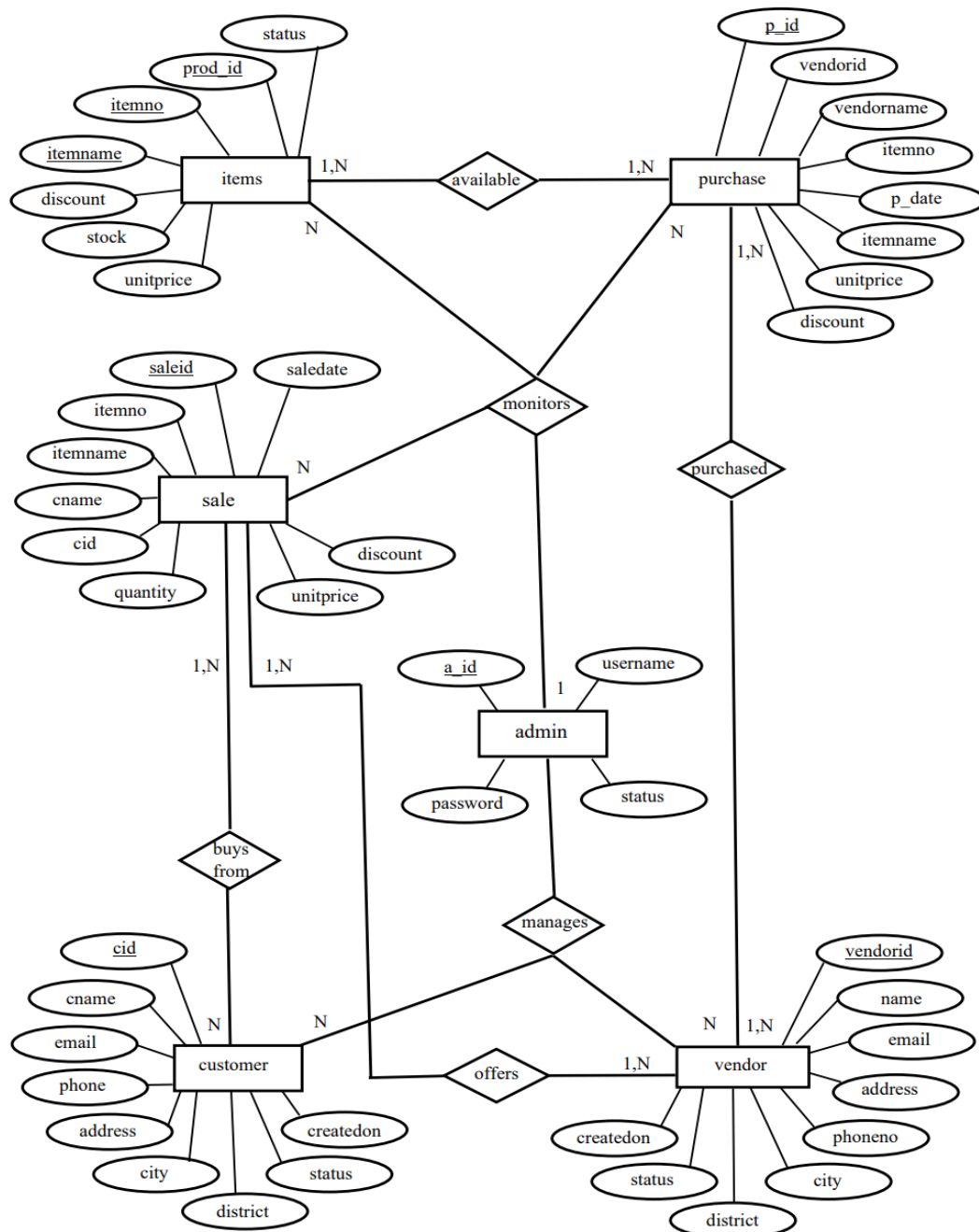
CHAPTER – 3**SYSTEM DESIGN****3.1 ER Diagram**

Fig 3.1:ER Diagram

There are 6 entities in Inventory management database,

1.Admin

- 2.Item
- 3.Purchase
- 4.Vendor
- 5.Sales
- 6.Customer

Relation between the entities:

- 1.Item are available for the Purchase to Purchase the items from vendor
- 2.The Purchase table has details of vendors from whom the items are Purchased.
- 3.Vendor offer Sales to the Customer to buy the items.
- 4.Customer buys from the Sales offered by the Vendor.
- 5.Admin manages the details of Customers and Vendors, and it also monitors the Sales, Items and Purchase.

Schema Diagram:

The six tables with their primary keys in below schema diagram we have mentioned as:

- 1.admin: u_id
- 2.items: itemno,itemname,prod_id
- 3.purchase: p_id
- 4.vendor: vendorid
- 5.sale: saleid
- 6.customer: cid

- In purchase table, there are two references itemno and itemname which are refering the item table.
- In sale table, there are three references itemno, itemname, and cid ,where cid is refering to the customer table and remaining two are refering item table.

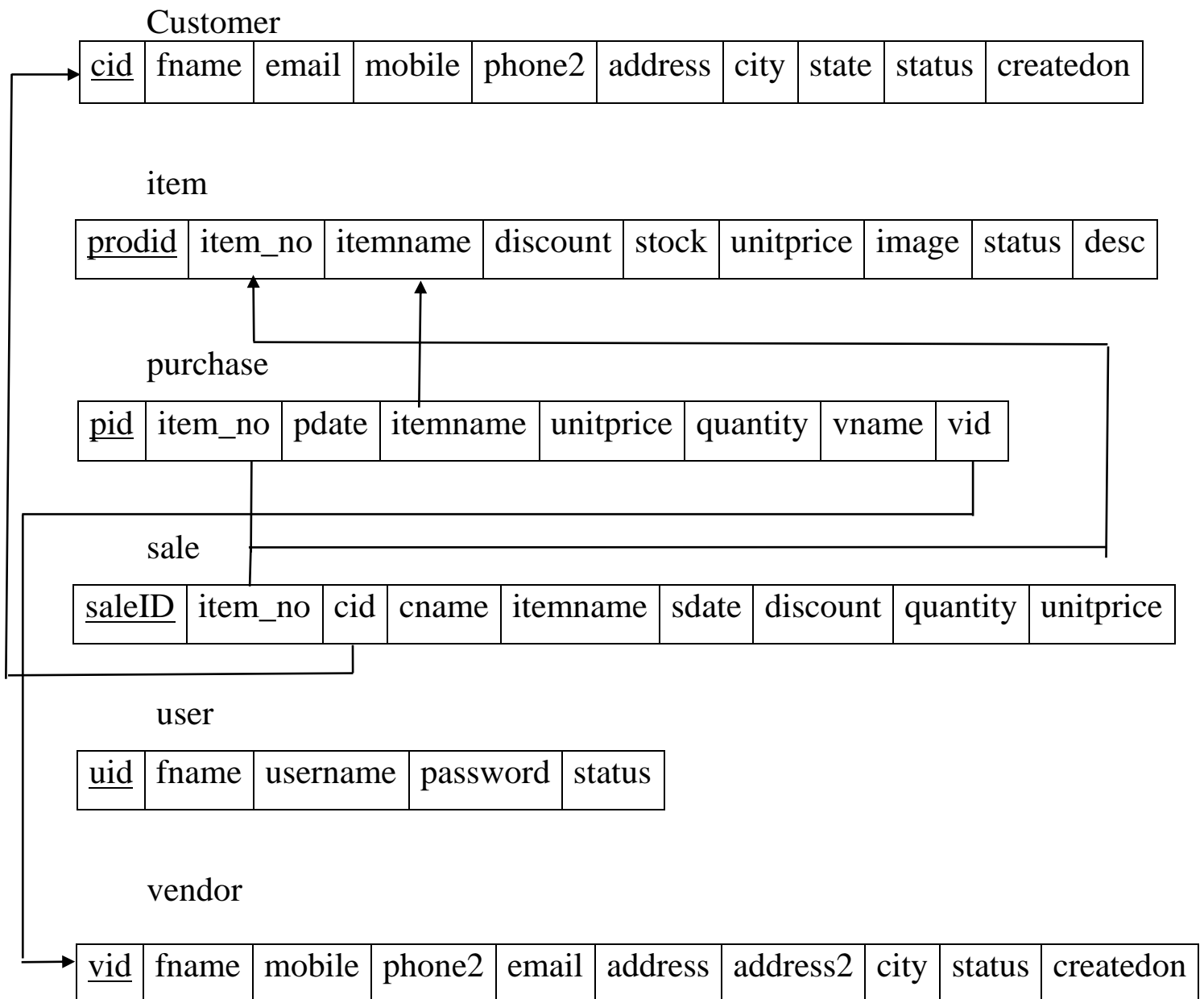


Fig 3.1.1:Schema Diagram

3.2 MySQL Server

SQL (Structured Query Language) is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS). In comparison to older read/write APIs like ISAM or VSAM, SQL offers two main advantages: first, it introduced the concept of accessing many records with one single command; and second, it eliminates the need to specify how to reach a record, e.g. with or without an index.

Originally based upon relational algebra and tuple relational calculus, SQL consists of a data definition language, data manipulation language, and data control language. The scope of SQL includes data insert, query, update and delete, schema creation and modification, and data access control. Although SQL is often described as, and to a great extent is, a declarative language (4GL), it also includes procedural elements. SQL was initially developed at IBM by Donald D. Chamberlin and Raymond F. Boyce in the early 1970s. This version, initially called SEQUEL (Structured English Query Language), was designed to manipulate and retrieve data stored in IBM's original quasirelational database management system, System R, which a group at IBM San Jose Research Laboratory had developed during the 1970s.

The SQL language is subdivided into several language elements, including:

- Clauses, which are constituent components of statements and queries. (In some cases, these are optional.)
- Expressions, which can produce either scalar values, or tables consisting of columns and rows of data
- Predicates, which specify conditions that can be evaluated to SQL three-valued logic (3VL)(true/false/unknown) or Boolean truth values and are used to limit the effects of statements and queries, or to change program flow.
- Queries, which retrieve the data based on specific criteria. This is an important element of SQL.
- Statements, which make diagnostics. you have a persistent effect on schemata and data, or may control transactions, program flow, connections, sessions.

3.3 Normalization

Normalization is the process of efficiently organizing data in a database. There are two goals of the normalization process: eliminating redundant data (for example, storing the same data in more than one table) and ensuring data dependencies make sense (only storing related data in a table). Both of these are worthy goals as they reduce the amount of space a database consumes and ensure that data is logically stored. There are several benefits for using Normalization in Database.

Benefits:

- Eliminate data redundancy
- Improve performance
- Query optimization
- Faster update due to less number of columns in one table Index improvement

There are different types of normalizations form available in the database.

1.First Normal Form (1NF):

First normal form (1NF) sets the very basic rules for an organized database:

- Eliminate duplicative columns from the same table.
- Create separate tables for each group of related data and identify each row with unique column or set of columns (the primary key).
- Remove repetitive groups
- Create Primary Key

Before we proceed let's understand a few things -- A KEY is a value used to identify a record in a table uniquely. A KEY could be a single column or combination of multiple columns Note: Columns in a table that are NOT used to identify a record uniquely are called non-key columns.

Primary Key



saleID	itemNumber	customerID
2	1	39
3	4	18
4	5	25
5	6	24
6	7	14
9	6	26
10	5	25
11	10	26
12	1	14
13	3	38

Fig 3.3.1:sales table in 1NF

2.Second Normal Form(2NF):

Second normal form(2NF) further address the concept of removing duplicative data:

- Meet all the requirements of the first Normal form.
- Remove the subset of data that apply to multiple rows of a table and place them in

separate tables.

customerID	fullName	email	mobile	phone2	address	address2	city	district	status	createdOn
25	Abhishek	abhi12@gmail.com	893783488	777833737	12, Downing Street	Bangalore	Bangalore	Bangalore	Active	2022-11-03 07:58:07
26	Sachin Tendulkar	sachintendulkar@icc.com	444958303	84792838	789-4, Apartment 3,	Cric Complex	New Delhi	Puttalam	Active	2018-05-03 07:58:38
38	Nuwan Perera	nuwan@yahoo.com	839378202	0	Nuwan Villa, Lower Street,	North Mulativu	Mullaitivu	Mullaitivu	Active	2018-05-05 16:47:49
39	Amal Silverton	amals452@yahoo.com	232345676	0	Amal's House, Amal's Street,	Amal Road	Ambalangoda	Galle	Active	2018-05-05 18:57:06
40	Andrew Symonds	symonds@cricket.au.com	123	0	23, Oak View Avenue	Western Australia	Melbourne	Colombo	Disabled	2018-05-13 06:50:23
41	Mark Taylor		111	0 111				Colombo	Active	2018-05-13 06:54:54
42	Nelson Mandela	sjobs@apple.com	333829832	0	1st Floor, Apple House,	Las Vegas Street	Las Vegas	Kalutara	Disabled	2018-05-13 08:09:41

Fig 3.3.2:Customer table has 2NF

3.Third Normal Form (3NF):

Third normal form (3NF) goes one large step further:

- Meet all the requirements of the second normal form.
- Remove columns that is not dependent upon the primary key.

This Inventory Management System will achieve 3NF.

CHAPTER – 4

IMPLEMENTATION AND RESULTS

4.1 Algorithms

Index:

1. Display options of register and admin login.
2. Link corresponding pages to them.

Register:

1. Connect to database.
2. Read customers' name, email, phone number, password, address.
3. Check if any of the field is null.
4. If any of the field is null then,
5. Display warning message.
6. Else insert the values into the corresponding table and show successfully registered.

Admin login:

1. Connect to database.
2. Read admin name and password using post method.
3. If the entered values are correct then,
4. Redirect to admin operation.
5. Else show error message.

Admin operations:

1. Show admin operations such as,
2. View customer, view sales, view Purchase, view Reports.

View customers:

1. Connect to database.
2. Retrieve the customers' name, email, phone number, password, address from the corresponding table.
3. Display the details in the form of table.

View Sales:

1. Connect to database.
2. Retrieve the salesID, customerID, Item number from the corresponding table.
3. Display the details in the form of table (Excel).

View Purchase:

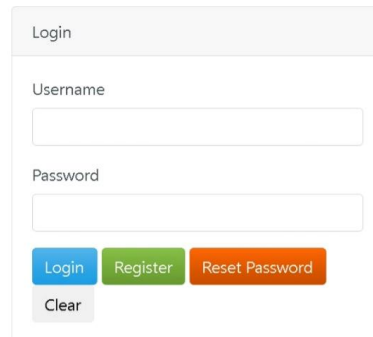
1. Connect to database.
2. Retrieve the purchaseID, purchaseDate, and other details from the table.
3. Display the details in the form of table.

View Vendor:

1. Connect to database.
2. Retrieve the vendorID, email, status, and other details from the table.
3. Display the details in the form of table.

Results

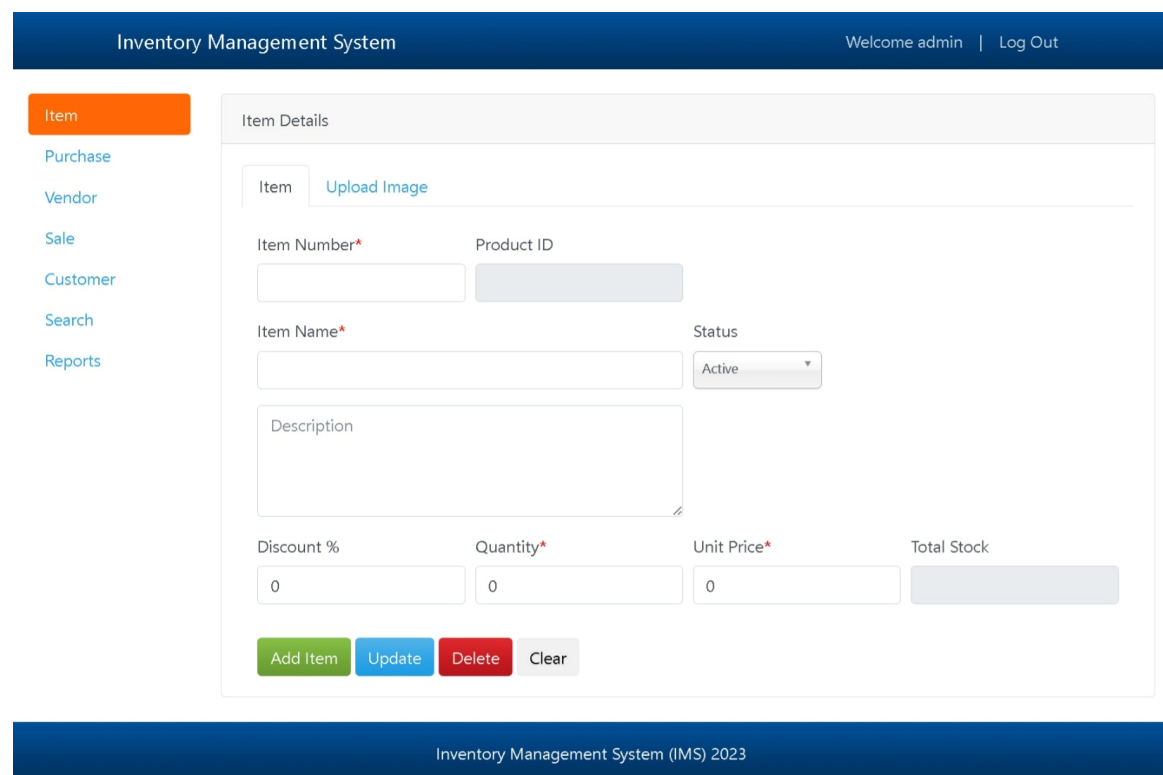
4.2 Snapshots



A screenshot of the login page. It features a light gray header with the word "Login". Below the header, there are two input fields: "Username" and "Password". At the bottom, there are four buttons: "Login" (blue), "Register" (green), "Reset Password" (orange), and "Clear" (gray).

Inventory Management System (IMS) 2023

Fig 4.2.1:Login Page



A screenshot of the "Item Details" page. The page has a dark blue header with "Inventory Management System" on the left and "Welcome admin | Log Out" on the right. On the left side, there is a sidebar with a list of menu items: "Item" (highlighted in orange), "Purchase", "Vendor", "Sale", "Customer", "Search", and "Reports". The main content area is titled "Item Details" and contains a form. The form has two tabs: "Item" (selected) and "Upload Image". The form fields include: "Item Number*" (text input), "Product ID" (text input), "Item Name*" (text input), "Status" (dropdown menu with "Active" selected), "Description" (text area), "Discount %" (text input with value "0"), "Quantity*" (text input with value "0"), "Unit Price*" (text input with value "0"), and "Total Stock" (text input). At the bottom of the form, there are four buttons: "Add Item" (green), "Update" (blue), "Delete" (red), and "Clear" (gray).

Inventory Management System (IMS) 2023

Fig 4.2.2:Item

Inventory Management System

Welcome admin | Log Out

Item

Purchase

Vendor

Sale

Customer

Search

Reports

Purchase Details

Item Number*

Purchase Date*

Purchase ID

Item Name*

Current Stock

Vendor Name*

Quantity*

Unit Price*

Total Cost

Add Purchase

Update

Clear

Inventory Management System (IMS) 2023

Fig 4.2.3:Purchase

Inventory Management System

Welcome admin | Log Out

Item

Purchase

Vendor

Sale

Customer

Search

Reports

Vendor Details

Full Name*

Status

Vendor ID

Phone (mobile)*

Phone 2

Email

Address*

Address 2

City

District

Add Vendor

Update

Delete

Clear

Inventory Management System (IMS) 2023

Fig 4.2.4:Vendor

Inventory Management System

Welcome admin | Log Out

Item

Purchase

Vendor

Sale

Customer

Search

Reports

Sale Details

Item Number*

Customer ID*

Customer Name

Sale ID

Item Name

Sale Date*

Total Stock

Discount %

Quantity*

Unit Price*

Total

Add Sale

Update

Clear

Inventory Management System (IMS) 2023

Fig 4.2.5:Sales

Inventory Management System

Welcome admin | Log Out

Item

Purchase

Vendor

Sale

Customer

Search

Reports

Customer Details

Full Name*

Status

Customer ID

Phone (mobile)*

Phone 2

Email

Address*

Address 2

City

District

Add Customer

Update

Delete

Clear

Inventory Management System (IMS) 2023

Fig 4.2.6:Customer

CONCLUSION AND FUTURE WORK

- Inventory management is a very complex but essential part of the supply chain. An effective inventory management system helps to reduce stock-related costs such as warehousing, carrying, and ordering costs.
- Since this project was started with very little knowledge about the InventoryManagement System, we came to know about the enhancement capability during the process of building it. Some of the scope we can increase for the betterment and effectiveness are listed below:

Interactive user interface design.

Online payment system can be added.

Making the system flexible in any type.

REFERENCES

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017.
2. Head First PHP & MySQL, Lynn Beighley and Michael Morrison.
3. Murach's PHP and MySQL, Joel Murach & Ray Harris, 3rd Edition.