Pharmacy Management System

Mini Project Report -Database Lab (DSE 2260)

Department of Data Science & Computer Applications

B. Tech Data Science

4th Semester – Batch: B1/B2/B3/B4

Submitted By

Sahil Patil	(200968154)
Manthena Dinesh Varma	(200968164)
Pakalapati Sri Raam Tej	(200968170)
Sahil Mehul Bavishi	(200968174)

Mentored By

Vinayak M
Assistant Professor-Senior
DSCA, MIT

Archana H/ Shameem
Assistant Professor-Senior
DSCA, MIT

MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

Date:

CERTIFICATE

This is to certify that the Sahil Patil (200968154), Manthena Dinesh Varma (200968164), Pakalapati Sri Raam Tej (200968170), Sahil Mehul Bavishi (200968174), have successfully executed a mini project titled "Crypto Marketplace" rightly brining fore the competencies and skill sets they have gained during the course- Database Lab (DSE 2262 & DSE), thereby resulting in the culmination of this project.

Vinayak M
Assistant Professor-Senior
DSCA, MIT

Archana H / Shameem
Assistant Professor-Senior
DSCA, MIT

ABSTRACT

The main aim of the project is the management of the database of the pharmaceutical shop. This project is insight into the design and implementation of a Pharmacy Management System. This is done by creating a database of the available medicines in the shop. The primary aim of pharmacy management system is to improve accuracy and enhance safety and efficiency in the pharmaceutical store.

The aim of this project is to develop database for the effective management of a pharmaceutical store. We have developed this database for ensuring effective management by providing insights of the drugs in stock.

This application can be used in any pharmaceutical shops having a database to maintain. The application used can generate reports, as per the user's requirements. The application can show invoices, bills, receipts etc. It can also maintain the record of supplies sent in by the supplier.

Contents

Chapter 1

Introduction

The aim of the project is to create an effective application to help the pharmacist to maintain the records of the medicines, handle user details, generate invoice, check and renew validity. Pharmacy management system deals with the maintenance of drugs and consumables in the pharmacy unit. This pharmacy management system is user friendly.

Here, the admin who is handling the organization will be responsible to manage the record of the employee. Each employee will be given with a separate username and password.

Chapter 2

Synopsis

2.1 Proposed System

The aim of the project is to create an effective software to help the pharmacist to maintain the records of the medicines, handle user details, generate invoice, check and renew validity. Pharmacy management system deals with the maintenance of drugs and consumables in the pharmacy unit. This pharmacy management system is user friendly.

2.2 Objectives

•To develop an application that deals with the day to day requirement of any pharmacy.

•To develop the easy management of the medicines (drugs).

•To handle the inventory details like sales details, purchase details and stock expiry and quantity.

•To provide details information about the stock on details necessary and help locate it in shop easily.

•To make the stock manageable and simplify the use of inventory in the pharmacy.

Chapter 3

Functional Requirements

Briefly write overview of functionalities provided by the application in terms of different modules.

3.1 User Registering/Login module

Two lines about module briefly and it supports functionalities- New user registration, Login, Forgot password

3.1.1 New User Registration

The user must be able to cerate user id and password by supplying appropriate details.

INPUT	New username, Password, phone
Processing	The system must check availability of entered user name. Password must follow criteria- minimum 9 char, at least one capital, one number and one special character. Check for validity of phone number by prompting to enter OTP
OUTPUT	User created Successfully message / highlight the information entered which is wrong and allow to renter.

3.1.2 Login

The existing user must be able to login upon entering proper user name and password.

INPUT	username, Password
Processing	Check the user name and password against information stored in data storage
OUTPUT	If user entered correct user name & Password Login successful and open main application menu Else Display Login not successful, retry logging in

3.1.3 Forgot password

If existing user name is not bale to login, forgot password can be used to reset password.

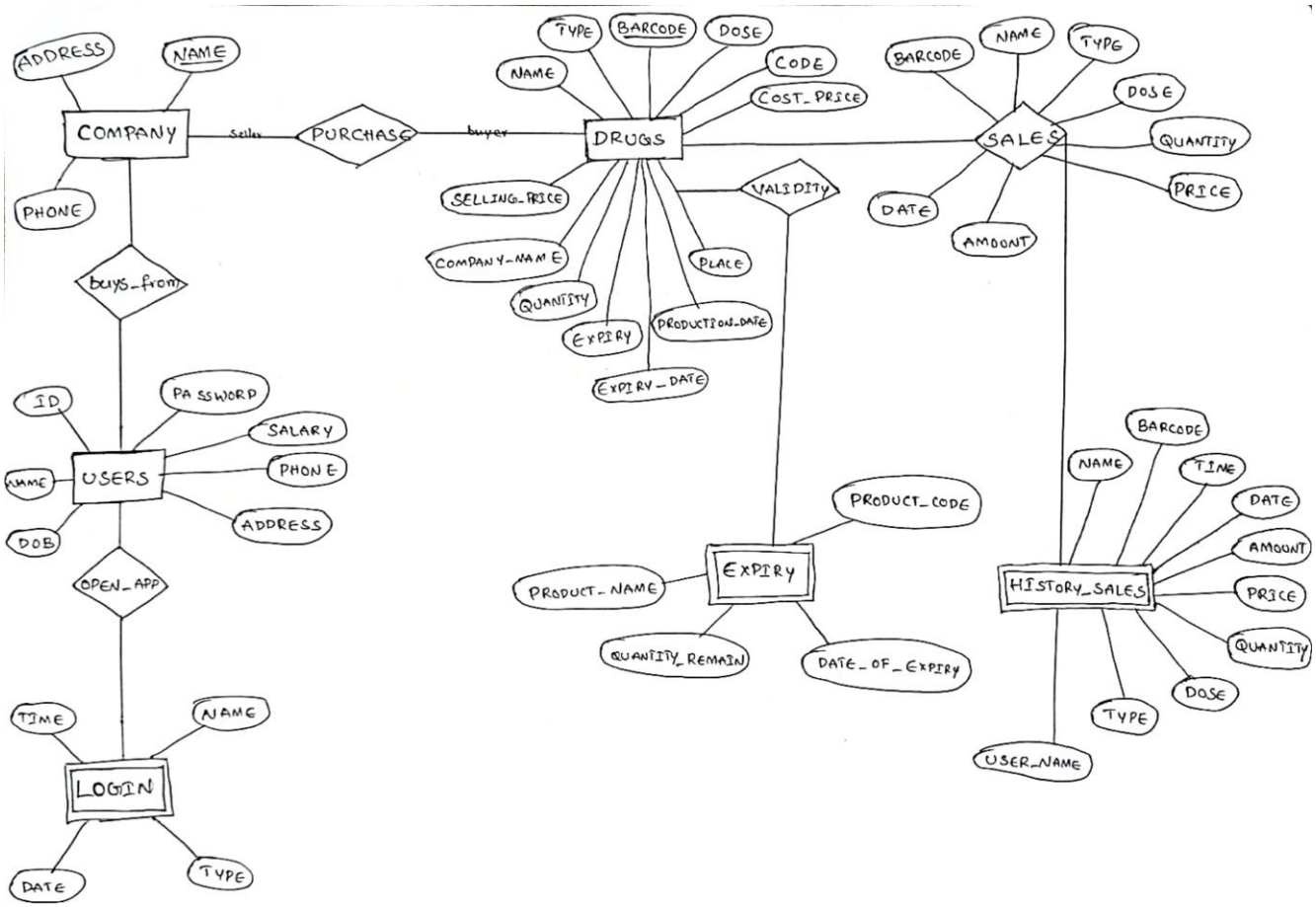
INPUT	Prompt user to enter username, Phone
Processing	If username and corresponding phone exist in the data storage Send OTP to Phone. Prompt the user to enter OTP If OTP matching Prompt user to change password according to criteria. Else OTP not matching. Else User name and corresponding Phone not existing in the storage

OUTPUT	Password successfully changed / User name, phone not matching

Chapter 4

Detailed Design

4.1 ER Diagram



4.2 Schema Diagram (Using MySQL)

Company(cname, address, phone)

Drugs(dname, dtype, barcode, dose, dcode, cost_price, selling_price, expiry, company_name, production_date, expiration_date, place, quantity)

company_name References Company

Expiry (product_name, product_code, expiry, quantity_remain)

History_sales(user_name, barcode,name_ ,type_ , dose , quantity , price , amount , date_ ,time_)

barcode References Drugs

Login(cname, type_, date_, time_)

Purchase(barcode, name_ , type_ , company_name , quantity , price , amount)

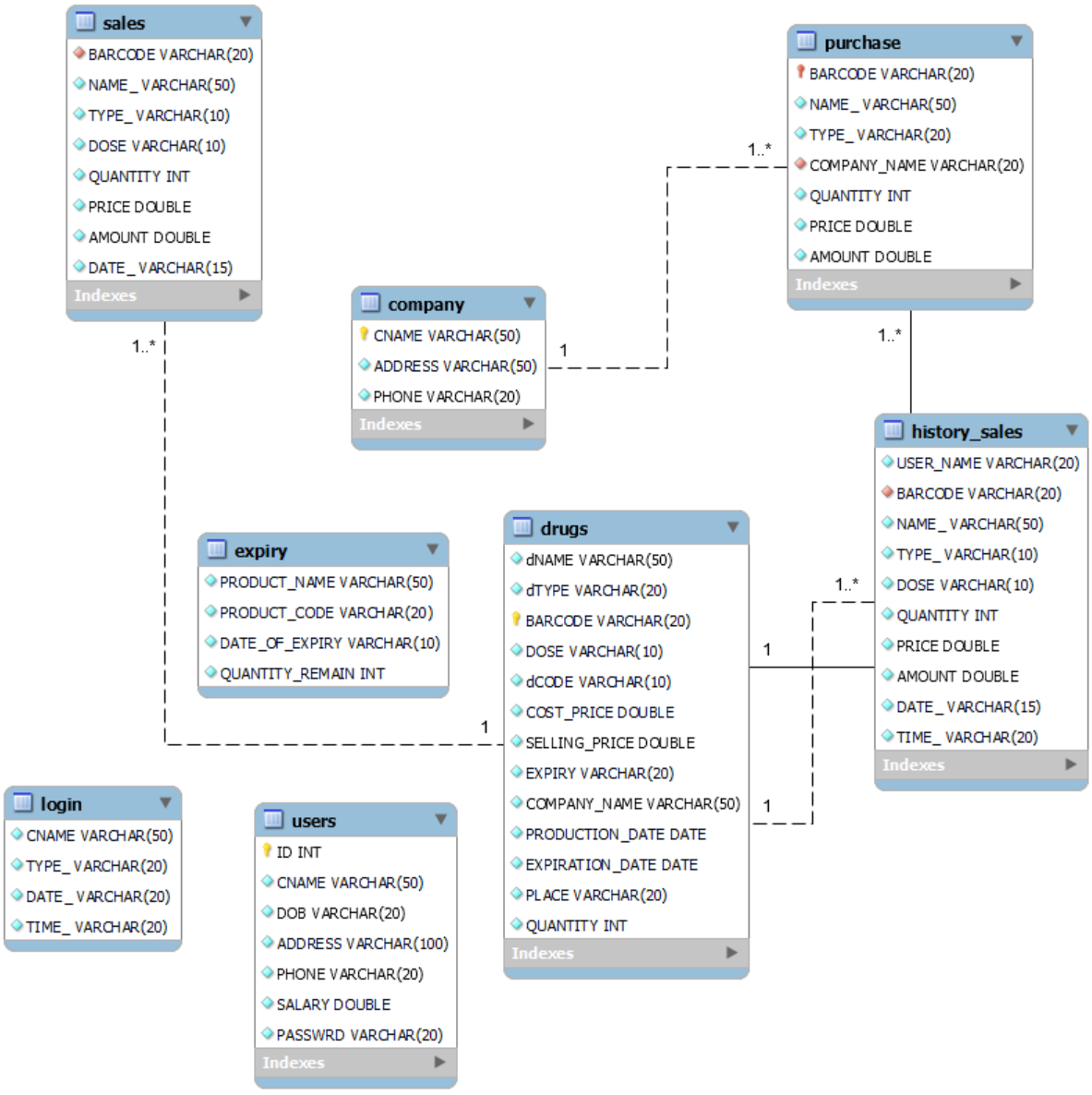
company_name References Company

barcode References Drugs

Sales(barcode, name_ , type_, dose , quantity, price , amount, date_)

barcode References Drugs

Users(id, cname, dob, address, phone, salary, passwrd**)**



4.3 Data Dictionary

COMPANY

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
CNAME	VARCHAR(50)	☑	☑	☐	☐	☐	☐	☐	☐
ADDRESS	VARCHAR(50)	☐	☑	☐	☐	☐	☐	☐	☐
PHONE	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
		☐	☐	☐	☐	☐	☐	☐	☐

DRUGS

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
CNAME	VARCHAR(50)	☑	☑	☐	☐	☐	☐	☐	☐
ADDRESS	VARCHAR(50)	☐	☑	☐	☐	☐	☐	☐	☐
PHONE	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
		☐	☐	☐	☐	☐	☐	☐	☐

EXPIRY

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
PRODUCT_NAME	VARCHAR(50)	☐	☑	☐	☐	☐	☐	☐	☐
PRODUCT_CODE	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
DATE_OF_EXPIRY	VARCHAR(10)	☐	☑	☐	☐	☐	☐	☐	☐
QUANTITY_REMAIN	INT	☐	☑	☐	☐	☐	☐	☐	☐
		☐	☐	☐	☐	☐	☐	☐	☐

HISTORY_SALES

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
USER_NAME	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
BARCODE	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
NAME_	VARCHAR(50)	☐	☑	☐	☐	☐	☐	☐	☐
TYPE_	VARCHAR(10)	☐	☑	☐	☐	☐	☐	☐	☐
DOSE	VARCHAR(10)	☐	☑	☐	☐	☐	☐	☐	☐
QUANTITY	INT	☐	☑	☐	☐	☐	☐	☐	☐
PRICE	DOUBLE	☐	☑	☐	☐	☐	☐	☐	☐
AMOUNT	DOUBLE	☐	☑	☐	☐	☐	☐	☐	☐
DATE_	VARCHAR(15)	☐	☑	☐	☐	☐	☐	☐	☐
TIME_	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
		☐	☐	☐	☐	☐	☐	☐	☐

LOGIN

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
CNAME	VARCHAR(50)	☐	☑	☐	☐	☐	☐	☐	☐
TYPE_	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
DATE_	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
TIME_	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
		☐	☐	☐	☐	☐	☐	☐	☐

PURCHASE

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
BARCODE	VARCHAR(20)	☑	☑	☐	☐	☐	☐	☐	☐
NAME_	VARCHAR(50)	☐	☑	☐	☐	☐	☐	☐	☐
TYPE_	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
COMPANY_NAME	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
QUANTITY	INT	☐	☑	☐	☐	☐	☐	☐	☐
PRICE	DOUBLE	☐	☑	☐	☐	☐	☐	☐	☐
AMOUNT	DOUBLE	☐	☑	☐	☐	☐	☐	☐	☐
		☐	☐	☐	☐	☐	☐	☐	☐

SALES

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
BARCODE	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
NAME_	VARCHAR(50)	☐	☑	☐	☐	☐	☐	☐	☐
TYPE_	VARCHAR(10)	☐	☑	☐	☐	☐	☐	☐	☐
DOSE	VARCHAR(10)	☐	☑	☐	☐	☐	☐	☐	☐
QUANTITY	INT	☐	☑	☐	☐	☐	☐	☐	☐
PRICE	DOUBLE	☐	☑	☐	☐	☐	☐	☐	☐
AMOUNT	DOUBLE	☐	☑	☐	☐	☐	☐	☐	☐
DATE_	VARCHAR(15)	☐	☑	☐	☐	☐	☐	☐	☐
		☐	☐	☐	☐	☐	☐	☐	☐

USERS

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
ID	INT	☑	☑	☐	☐	☐	☐	☐	☐
CNAME	VARCHAR(50)	☐	☑	☐	☐	☐	☐	☐	☐
DOB	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
ADDRESS	VARCHAR(100)	☐	☑	☐	☐	☐	☐	☐	☐
PHONE	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
SALARY	DOUBLE	☐	☑	☐	☐	☐	☐	☐	☐
PASSWRD	VARCHAR(20)	☐	☑	☐	☐	☐	☐	☐	☐
		☐	☐	☐	☐	☐	☐	☐	☐

PK – Primary key

NN – Not null

4.4 Relational Model Implementation

CREATE TABLE COMPANY

(CNAME VARCHAR(50) PRIMARY KEY,

ADDRESS VARCHAR(50) NOT NULL,

PHONE VARCHAR(20) NOT NULL);

CREATE TABLE DRUGS

(DNAME VARCHAR(50) NOT NULL,

DTYPE VARCHAR(20) NOT NULL,

BARCODE VARCHAR(20) PRIMARY KEY,

DOSE VARCHAR(10) NOT NULL,

DCODE VARCHAR(10) NOT NULL,

COST_PRICE DOUBLE NOT NULL,

SELLING_PRICE DOUBLE NOT NULL,

EXPIRY VARCHAR(20) NOT NULL,

COMPANY_NAME VARCHAR(50) NOT NULL,

PRODUCTION_DATE DATE NOT NULL,

EXPIRATION_DATE DATE NOT NULL,

PLACE VARCHAR(20) NOT NULL,

QUANTITY INT NOT NULL);

CREATE TABLE EXPIRY

(PRODUCT_NAME VARCHAR(50) NOT NULL,

PRODUCT_CODE VARCHAR(20) NOT NULL,

DATE_OF_EXPIRY VARCHAR(10) NOT NULL,

QUANTITY_REMAIN INT NOT NULL);

CREATE TABLE HISTORY_SALES

(USER_NAME VARCHAR(20) NOT NULL,

BARCODE VARCHAR(20) NOT NULL,

NAME_ VARCHAR(50) NOT NULL,

TYPE_ VARCHAR(10) NOT NULL,

DOSE VARCHAR(10) NOT NULL,

QUANTITY INT NOT NULL,

PRICE DOUBLE NOT NULL,

AMOUNT DOUBLE NOT NULL,

DATE_ VARCHAR(15) NOT NULL,

TIME_ VARCHAR(20) NOT NULL,

FOREIGN KEY(BARCODE) REFERENCES DRUGS(BARCODE));

CREATE TABLE LOGIN

(CNAME VARCHAR(50) NOT NULL,

TYPE_ VARCHAR(20) NOT NULL,

DATE_ VARCHAR(20) NOT NULL,

TIME_ VARCHAR(20) NOT NULL);

CREATE TABLE PURCHASE

(BARCODE VARCHAR(20) PRIMARY KEY,

NAME_ VARCHAR(50) NOT NULL,

TYPE_ VARCHAR(20) NOT NULL,

COMPANY_NAME VARCHAR(20) NOT NULL,

QUANTITY INT NOT NULL,

PRICE DOUBLE NOT NULL,

AMOUNT DOUBLE NOT NULL,

FOREIGN KEY(BARCODE) REFERENCES DRUGS(BARCODE));

CREATE TABLE SALES

(BARCODE VARCHAR(20) NOT NULL,

NAME_ VARCHAR(50) NOT NULL,

TYPE_ VARCHAR(10) NOT NULL,

DOSE VARCHAR(10) NOT NULL,

QUANTITY INT NOT NULL,

PRICE DOUBLE NOT NULL,

AMOUNT DOUBLE NOT NULL,

DATE_ VARCHAR(15) NOT NULL,

FOREIGN KEY(BARCODE) REFERENCES DRUGS(BARCODE));

CREATE TABLE USERS

(ID INT PRIMARY KEY,

CNAME VARCHAR(50) NOT NULL,

DOB VARCHAR(20) NOT NULL,

ADDRESS VARCHAR(100) NOT NULL,

PHONE VARCHAR(20) NOT NULL,

SALARY DOUBLE NOT NULL,

PASSWRD VARCHAR(20) NOT NULL);

Inserting records

INSERT INTO drugs (dNAME, dTYPE, BARCODE, DOSE, dCODE, COST_PRICE, SELLING_PRICE, EXPIRY, COMPANY_NAME, PRODUCTION_DATE, EXPIRATION_DATE, PLACE, QUANTITY) VALUES

('Novalo', 'Bills', 'fsdgjfihjorodsf', 'normal', '3d00', 2, 3, 'Available for use', 'Med_City', '2017-03-03', '2019-03-03', 'N-Right', 40),

('novafol', 'Bills', 'ftrkl432432md', 'normal', '2xaa', 33, 40, 'Available for use', 'Med_City', '2016-01-01', '2017-01-01', 'N-Left', 27),

('Declofien', 'Bills', 'ftwerqanjfmd', 'normal', '2xaa', 31, 37, 'Available for use', 'Med_City', '2016-01-01', '2017-01-01', 'N-Left', 27);

INSERT INTO history_sales (USER_NAME, BARCODE, NAME_, TYPE_, DOSE, QUANTITY, PRICE, AMOUNT, DATE_, TIME_) VALUES

('admin', 'fsdgjfihjorodsf', 'Novalo', 'Bills', ' Free used', 2, 6, 12, '12-02-2017', '05:02:06'),

('admin', 'fsdgjfihjorodsf', 'Novalo', 'Bills', ' Free used', 2, 6, 12, '12-02-2017', '05:02:26'),

('admin', 'fsdgjfihjorodsf', 'Novalo', 'Bills', ' Free used', 4, 6, 24, '12-02-2017', '05:02:40'),

('admin', 'ftrkl432432md', 'novafol', 'Injection', '1 (Day)', 2, 14, 28, '13-02-2017', '01:38:00'),

('admin', 'ftrkl432432md', 'novafol', 'Injection', '1 (Day)', 2, 14, 28, '13-02-2017', '01:38:10'),

('admin', 'ftrkl432432md', 'novafol', 'Injection', '1 (Day)', 7, 14, 98, '13-02-2017', '01:38:28'),

('admin', 'ftrkl432432md', 'novafol', 'Injection', '1 (Day)', 1, 14, 14, '13-02-2017', '01:38:46');

INSERT INTO login (CNAME, TYPE_, DATE_, TIME_) VALUES

('admin', 'Admin', '17-02-2017', '10:30:24'),

('admin', 'Admin', '17-02-2017', '10:32:48'),

('mark', 'Employee', '17-02-2017', '10:32:56'),

('admin', 'Admin', '17-02-2017', '10:33:10'),

('mark', 'Employee', '17-02-2017', '10:33:37'),

('admin', 'Admin', '17-02-2017', '10:36:21'),

('admin', 'Admin', '17-02-2017', '10:36:53'),

('admin', 'Admin', '17-02-2017', '10:49:27'),

('admin', 'Admin', '17-02-2017', '11:02:23'),

('admin', 'Admin', '17-02-2017', '01:40:08'),

('admin', 'Admin', '18-02-2017', '10:50:29'),

('admin', 'Admin', '18-02-2017', '10:51:50'),

('admin', 'Admin', '18-02-2017', '10:53:33');

INSERT INTO purchase (BARCODE, NAME_, TYPE_, COMPANY_NAME, PRICE, QUANTITY, AMOUNT) VALUES

('fsdgjfihjorodsf', 'Novalo', 'Bills', 'Med_City', 40, 2, 80),

('ftwerqanjfmd', 'novafol', 'Bills', 'Med_City', 40, 1, 40),

('ftrkl432432md', 'Declofien', 'Bills', 'Med_City', 65, 5, 325);

INSERT INTO users (ID, CNAME, DOB, ADDRESS, PHONE, SALARY,PASSWRD) VALUES

(1, 'admin', '23-12-1995', 'Someplace India', '9800000000', 50000, 'admin'),

(2, 'mark', '3-2-1972', 'Bangalore India', '01290789432', 2000, 'mark'),

(3, 'clark', '3-2-1971', 'Nowhere Earth-616', '01147893423', 4000, 'rootaccess'),

(4, 'Tony Stark', '7-8-1977', ' 10880 Malibu Point, Malibu, California', '011804368743', 3000, 'rootaccess');

4.5 Queries

List of queries used to retrieve data

4.5.1 Searching employee details and department name in which they are working Empno entered by the user.

[Assume that you have a user interface for searching Employee information]

SELECT ENAME, DEPNAME from Emp,Dept where emp.deptno=dept.deptno and empno=100;

…..

4.7 Triggers

[if applicable/ if implemented then show the code for Triggers]

….

4.8 Stored Procedures

[if applicable/ if implemented then show the code for Triggers]

…

4.9 Stored Functions

[if applicable/ if implemented then show the code for Triggers]

…..

5. Functional Requirement Implementation

[code related to different function implementation may be added here]

~~6. Testing~~

~~[set of testcases passed/failed]~~

[No need to write these]

7. Result

[screenshots of applications with 1-2-line explanation]

8. Conclusion and Future Work

8.1 Conclusion

….

8.2 Scope for future work

….

~~References~~