Pharmacy Management System

18CSC209J - Database Management System and Cloud Integration Services

Mini Project Report

Submitted by

Papai Mondal [RA2111028010116]
B.Tech CSE Cloud Computing

Dhruv Deshmukh [RA2111028010125]
B.Tech CSE Cloud Computing

DEPARTMENT OF NETWORKING AND COMMUNICATION
SCHOOL OF COMPUTING
COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
(Under Section 3 of UGC Act, 1956)
S.R.M. NAGAR, KATTANKULATHUR – 603 203
KANCHEEPURAM DISTRICT
MAY 2023

BONFIDE

This is to certify that **18CSC209J - Database Management System and Cloud Integration Services** mini project report titled **"PHARMACY MANAGEMENT SYSTEM"** is the bonafide work of **PAPAI MONDAL (RA2111028010116), DHRUV DESHMUKH (RA2111028010125**) who undertook the task of completing the project within the allotted time.

SIGNATURE

Dr. B. Balakiruthiga

Associate Professor
Department of NWC,
SRM Institute of Science
and Technology

SIGNATURE

Dr. Annapurani Panaiyappan K

Professor and Head
Department of NWC,
SRM Institute of Science
and Technology

TABLE OF CONTENTS

ABSTRACT

The **Pharmacy Management System project abstract** must answer or address the needs of every issue that happens in the pharmacy. The issues could include the poor security of their record, manual handling of drug information, and others.

Pharmacists can use the Pharmacy Management System program to help them methodically manage their pharmacies. When a medicine’s name is input, the Pharmacy Management System can help by providing details about the medicine. A computer displays information about the medicine, such as its dosage and expiration date. In large medical stores, manually handling the specifics of all the drugs becomes very tough.

We can keep track of all the medicines by using this pharmacy management system. It is updated with new information as new medicines are introduced, and it includes an expiration date as well as a search option. When we complete the name of a medicine, it displays the medicine’s details.

To make this system, Visual Studio with PHP and ASP.NET was used. The SQL database was created with MySQL.

INTRODUCTION

The main goal of Pharmacy management system is to manage all records and transaction within the inventory and managing of sales. Pharmacy management System is a very effective tool for an organization to be efficient in business management. The traditional way of managing sales and inventory is performed by using a pen and a paper to write down the type and quantity of the stock. But errors in inventory records still exist even when the management uses IT systems and product data capturing technologies to improve the inventory systems. Inventory managers have to face inaccuracy of inventory records either at the store or at the warehouse level. In order to improve accuracy of inventory checking, people started using Auto ID technologies.

In EPC global Report, Auto ID technologies are defined as the host of technologies that are used to help machines to identify objects. It is about identifying items, capturing all information about the items, sending and storing those data into a computer with minimal human intervention.

Theoretical review.

Pharmacy Management system is a web base system that works as a website to manage and functioning all pharmacy activities through a web server (Apache). A web page is what you see on the screen when you type in a web address, click on a link, or put a query in a search engine. A web page can contain any type of information, and can include text, colour, graphics, animation and sound.

PROBLEM STATEMENT

Most pharmacies faced problems such as insufficient service promotions, lack of coherence of pharmacy services in hospitals, poor drug information systems, and the inconsistency of the pharmacy information management due to its manual processes. Now, these are the problems that must be solved with this Pharmacy Management System Project Proposal.

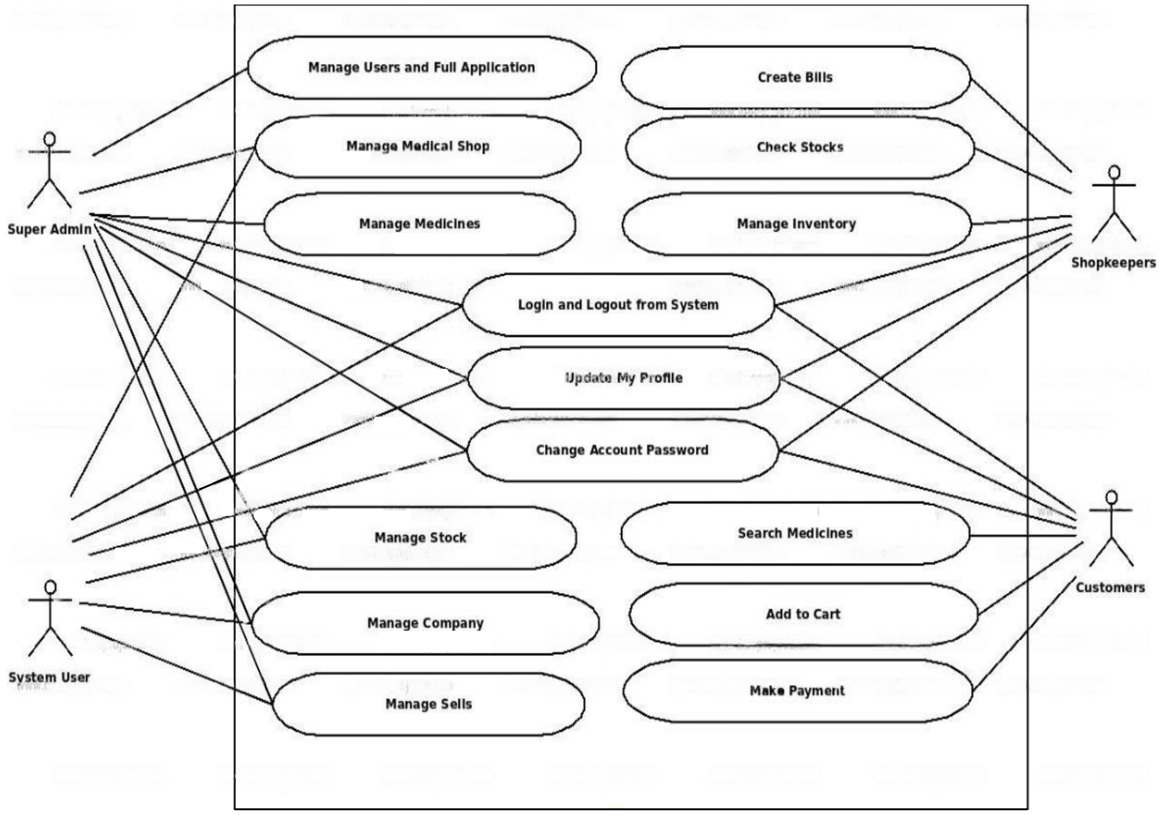
Solution: Pharmacy Management System Project is a great system for storing data, maintaining, and organizing the use and process of medications in the pharmacy. This computer software is programmed to perform the various tasks required in the operation of a pharmacy. The system will improve the efficiency of the pharmacy and enable the storing of digital records. Managing a system for pharmacy is the process of creating and implementing evidence-based pharmaceutical usage strategies to improve member and population health while maximizing healthcare resources.

Scope: One of the most important responsibilities of pharmacy management is to supervise and manage the pharmacy employees to ensure healthy working relationships and outcomes. Each of these functions is critical to the pharmacy's operation and should be improved.

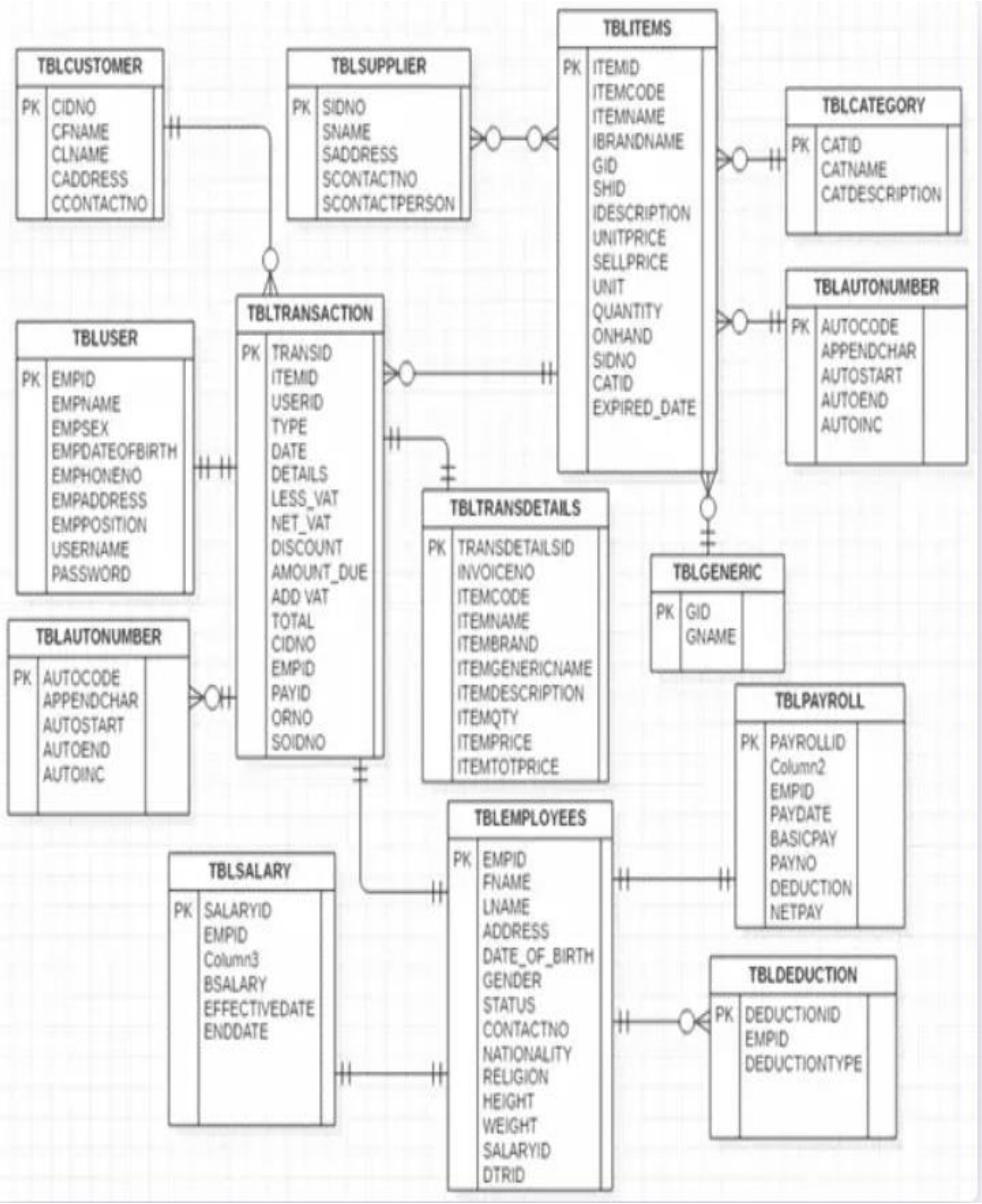
MODULE DESCRIPTION

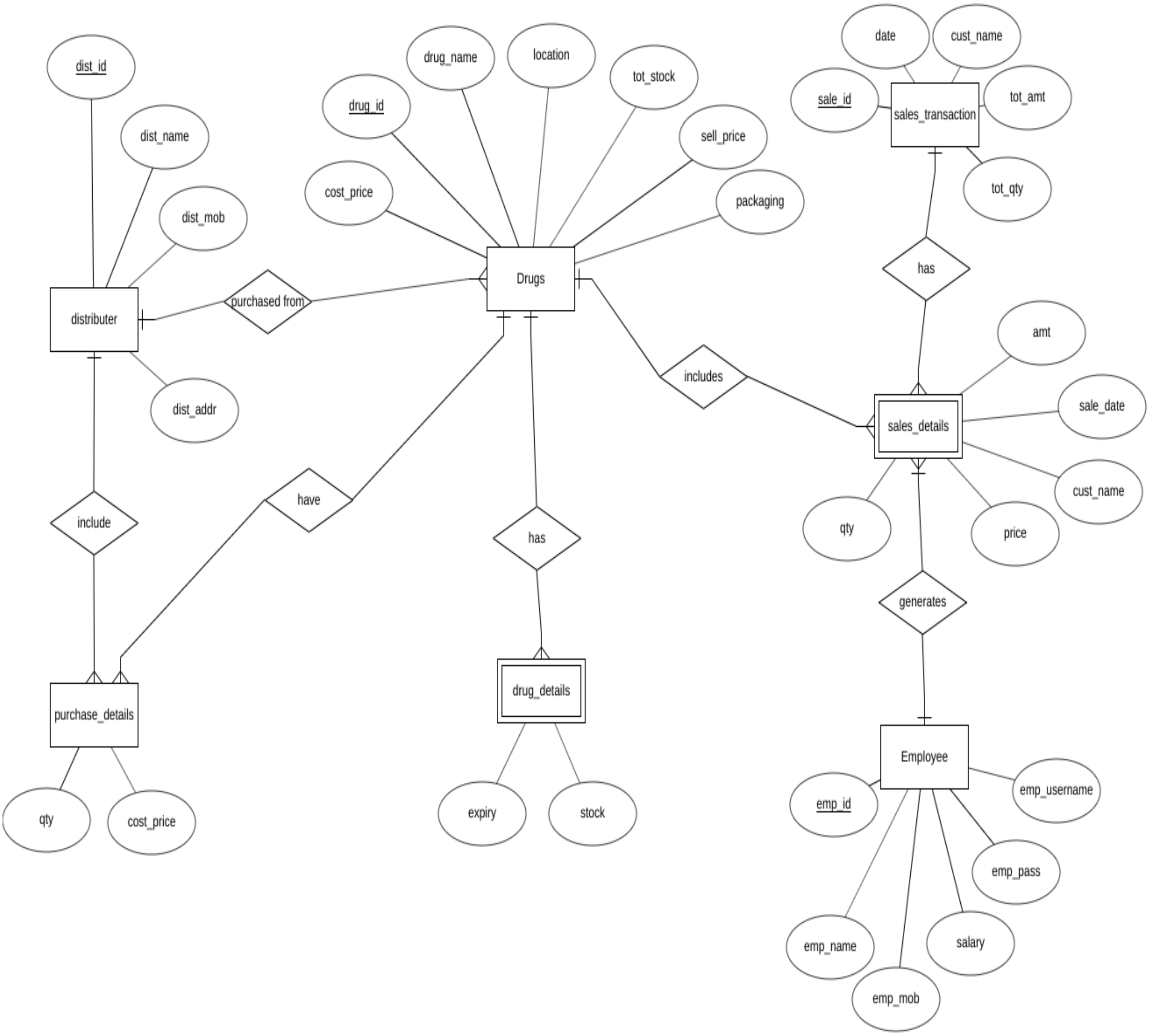
- **Pharmacy Information Management:** is a multi-functional system that helps pharmacists to keep track of medicine supplies and organize them. The modules aids in the reduction of medication errors, the improvement of patient safety, the reporting of drug usage, and the tracking of expenses.
- **Medicine Management:** To manage medicines modules will assess the need for and use of medication, the patient's response to medication, and the patient's level of understanding of the drug and how to take it with the patient.
- **Categorize Medicine Information:** Categorizing the drugs available in the pharmacy will be much easier for the admin through the help of this module. This will do the monitoring and checking of the medicine information to identify its category.
- **Monitor Medicine Orders:** is used to keep track of dates and events throughout the process chain, from placing an order with an external vendor to presenting goods in a store or receiving goods in a distribution center.
- **Manage Sales and Stocks:** This module will help the Pharmacist with the sales and stocks management that includes ordering, storing, tracking, and monitoring stock levels as well as monitoring their revenue.
- **Drug Inventory Management:** The drug inventory management module strives to reduce procurement and carrying expenses while maintaining a sufficient stock of products to meet the needs of customers and prescribers. This will also monitor the performance of the pharmacy and to know which are the most needed medicines.
- **Generate Processes Reports:** In all organization or business, reports are very essential. To help the admin in these matters, this module generates the transaction reports to keep track of the pharmacy activities.

USE CASE DIAGRAM



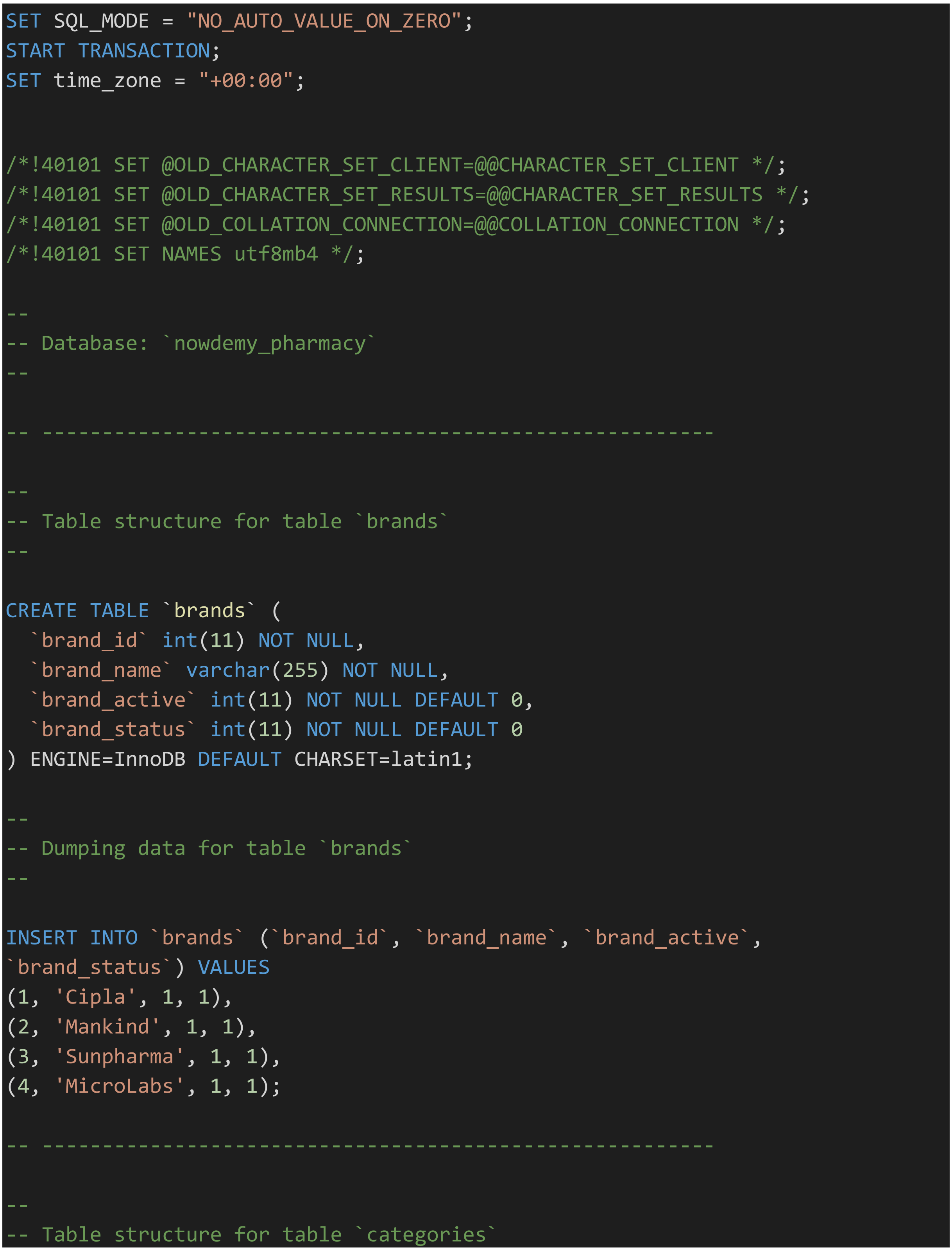
ER DIAGRAM


TBLCUSTOMER
PK CIDNO
CFNAME
CLNAME
CADDRESS
CCONTACTNO
TBLSUPPLIER
PK SIDNO
SNAME
SADDRESS
SCONTACTNO
SCONTACTPERSON
TBLITEMS
PK ITEMID
ITEMCODE
ITEMNAME
IBRANDNAME
GID
SHID
IDESCRIPTION
UNITPRICE
SELLPRICE
UNIT
QUANTITY
ONHAND
SIDNO
CATID
EXPIRED_DATE
TBLCATEGORY
PK CATID
CATNAME
CATDESCRIPTION
TBLAUTONUMBER
PK AUTOCODE
APPENDCHAR
AUTOSTART
AUTOEND
AUTOINC
TBLUSER
PK EMPID
EMPNAME
EMPSEX
EMPDATEOFBIRTH
EMPHONENO
EMPADDRESS
EMPPOSITION
USERNAME
PASSWORD
TBLTRANSACTION
PK TRANSID
ITEMID
USERID
TYPE
DATE
DETAILS
LESS_VAT
NET_VAT
DISCOUNT
AMOUNT_DUE
ADD VAT
TOTAL
CIDNO
EMPID
PAYID
ORNO
SOIDNO
TBLTRANSDETAILS
PK TRANSDETAILSID
INVOICENO
ITEMCODE
ITEMNAME
ITEMBRAND
ITEMGENERICNAME
ITEMDESCRIPTION
ITEMQTY
ITEMPRICE
ITEMTOTPRICE
TBLGENERIC
PK GID
GNAME
TBLAUTONUMBER
PK AUTOCODE
APPENDCHAR
AUTOSTART
AUTOEND
AUTOINC
TBLPAYROLL
PK PAYROLLID
Column2
EMPID
PAYDATE
BASICPAY
PAYNO
DEDUCTION
NETPAY
TBLSALARY
PK SALARYID
EMPID
Column3
BSALARY
EFFECTIVEDATE
ENDDATE
TBLEMPLOYEES
PK EMPID
FNAME
LNAME
ADDRESS
DATE_OF_BIRTH
GENDER
STATUS
CONTACTNO
NATIONALITY
RELIGION
HEIGHT
WEIGHT
SALARYID
DTRID
TBLDEDUCTION
PK DEDUCTIONID
EMPID
DEDUCTIONTYPE


dist_id
dist_name
dist_mob
distributer
dist_addr
purchased from
include
purchase_details
qty
cost_price
have
drug_id
drug_name
location
tot_stock
sell_price
packaging
cost_price
Drugs
has
drug_details
expiry
stock
includes
date
cust_name
sale_id
sales_transaction
tot_amt
tot_qty
has
amt
sales_details
sale_date
cust_name
qty
price
generates
Employee
emp_id
emp_username
emp_pass
salary
emp_name
emp_mob

DATABASE CREATION

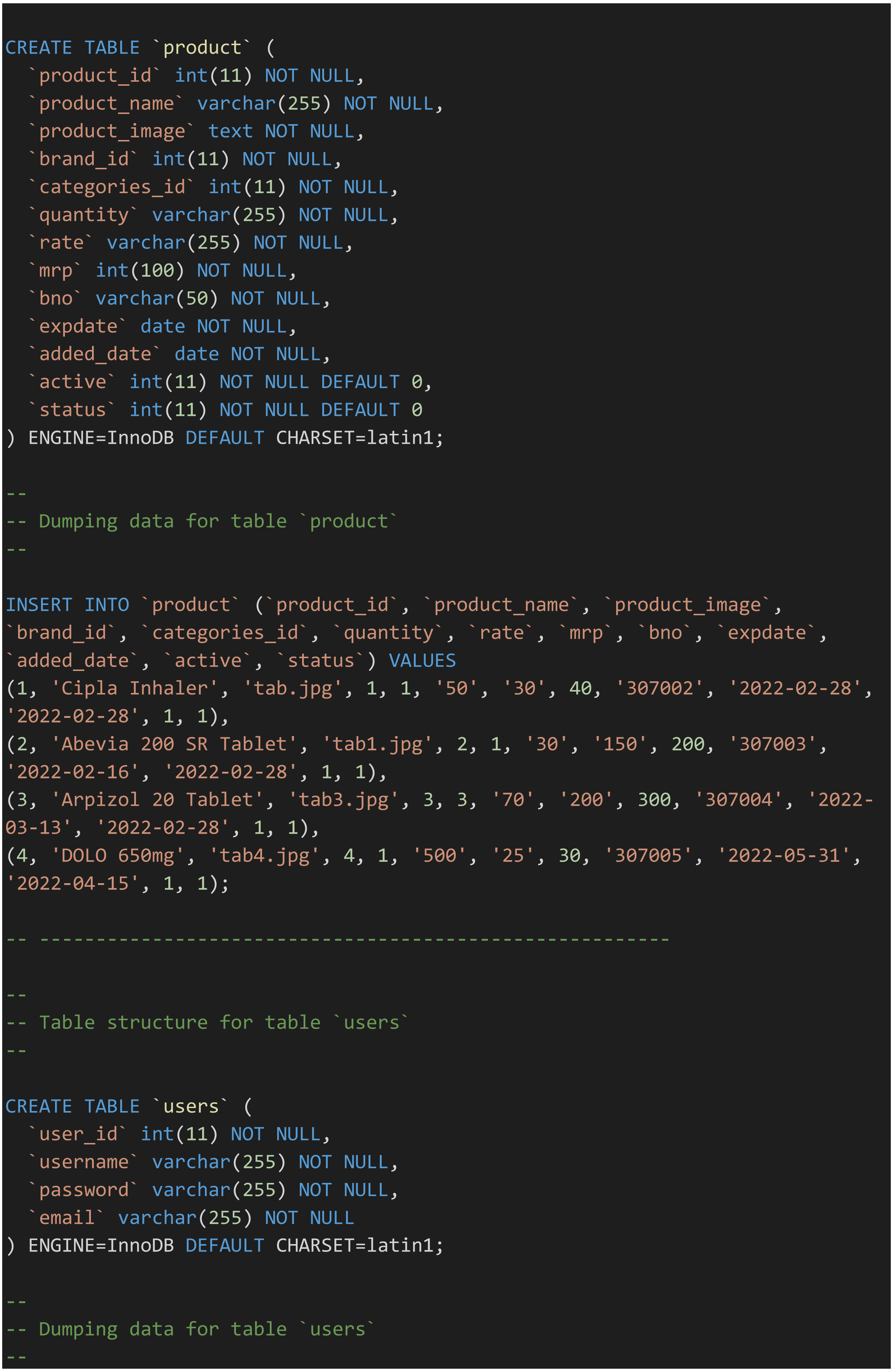
Below are the DDL and DML commands used for create the Database tables.



```sql
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `nowdemy_pharmacy`
--

-- --------------------------------------------------------

--
-- Table structure for table `brands`
--

CREATE TABLE `brands` (
  `brand_id` int(11) NOT NULL,
  `brand_name` varchar(255) NOT NULL,
  `brand_active` int(11) NOT NULL DEFAULT 0,
  `brand_status` int(11) NOT NULL DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `brands`
--

INSERT INTO `brands` (`brand_id`, `brand_name`, `brand_active`,
`brand_status`) VALUES
(1, 'Cipla', 1, 1),
(2, 'Mankind', 1, 1),
(3, 'Sunpharma', 1, 1),
(4, 'MicroLabs', 1, 1);

-- --------------------------------------------------------

--
-- Table structure for table `categories`
```

```
--

CREATE TABLE `categories` (
  `categories_id` int(11) NOT NULL,
  `categories_name` varchar(255) NOT NULL,
  `categories_active` int(11) NOT NULL DEFAULT 0,
  `categories_status` int(11) NOT NULL DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `categories`
--

INSERT INTO `categories` (`categories_id`, `categories_name`,
`categories_active`, `categories_status`) VALUES
(1, 'Tablets', 1, 1),
(2, 'Syrup', 1, 1),
(3, 'SkinLiquid', 1, 1),
(4, 'PainKiller', 1, 1);

-- --------------------------------------------------------

--
-- Table structure for table `orders`
--

CREATE TABLE `orders` (
  `id` int(15) NOT NULL,
  `uno` varchar(50) NOT NULL,
  `orderDate` date NOT NULL,
  `clientName` text NOT NULL,
  `projectName` varchar(30) NOT NULL,
  `clientContact` int(15) NOT NULL,
  `address` varchar(30) NOT NULL,
  `subTotal` int(100) NOT NULL,
  `totalAmount` int(100) NOT NULL,
  `discount` int(100) NOT NULL,
  `grandTotalValue` int(100) NOT NULL,
  `gstn` int(100) NOT NULL,
  `paid` int(100) NOT NULL,
  `dueValue` int(100) NOT NULL,
  `paymentType` int(15) NOT NULL,
  `paymentStatus` int(15) NOT NULL,
  `paymentPlace` int(5) NOT NULL,
  `delete_status` tinyint(5) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data for table `orders`
```

```sql
--

INSERT INTO `orders` (`id`, `uno`, `orderDate`, `clientName`, `projectName`,
`clientContact`, `address`, `subTotal`, `totalAmount`, `discount`,
`grandTotalValue`, `gstn`, `paid`, `dueValue`, `paymentType`, `paymentStatus`,
`paymentPlace`, `delete_status`) VALUES
(1, 'INV-0001', '2022-02-28', 'Santosh Kadam', '', 2147483647, '', 100, 10,
108, 49, 0, 49, 49, 2, 1, 0, 0),
(2, 'INV-0002', '2022-03-24', 'Aishwarya Joshi', '', 2147483647, '', 300, 0,
354, 0, 0, 354, 354, 3, 3, 1, 0),
(3, 'INV-0003', '2022-04-15', 'Saurabh Katkar', '', 2147483647, '', 860, 1015,
10, 1005, 155, 500, 505, 2, 2, 1, 0),
(4, 'INV-0004', '2022-04-15', 'Mayuri K', '', 2147483647, '', 60, 71, 0, 71,
11, 50, 21, 5, 2, 1, 0);

-- --------------------------------------------------------


--
-- Table structure for table `order_item`
--

CREATE TABLE `order_item` (
  `id` int(15) NOT NULL,
  `productName` int(100) NOT NULL,
  `quantity` varchar(255) NOT NULL,
  `rate` varchar(255) NOT NULL,
  `total` varchar(255) NOT NULL,
  `lastid` int(50) NOT NULL,
  `added_date` date NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `order_item`
--

INSERT INTO `order_item` (`id`, `productName`, `quantity`, `rate`, `total`,
`lastid`, `added_date`) VALUES
(5, 2, '1', '100', '100.00', 1, '0000-00-00'),
(6, 2, '2', '150', '300.00', 2, '0000-00-00'),
(7, 1, '2', '30', '60.00', 3, '2022-04-15'),
(8, 2, '4', '150', '600.00', 3, '2022-04-15'),
(9, 3, '1', '200', '200.00', 3, '2022-04-15'),
(10, 1, '2', '30', '60.00', 4, '2022-04-15');

-- --------------------------------------------------------


--
-- Table structure for table `product`
--
```



```sql
CREATE TABLE `product` (
  `product_id` int(11) NOT NULL,
  `product_name` varchar(255) NOT NULL,
  `product_image` text NOT NULL,
  `brand_id` int(11) NOT NULL,
  `categories_id` int(11) NOT NULL,
  `quantity` varchar(255) NOT NULL,
  `rate` varchar(255) NOT NULL,
  `mrp` int(100) NOT NULL,
  `bno` varchar(50) NOT NULL,
  `expdate` date NOT NULL,
  `added_date` date NOT NULL,
  `active` int(11) NOT NULL DEFAULT 0,
  `status` int(11) NOT NULL DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `product`
--

INSERT INTO `product` (`product_id`, `product_name`, `product_image`,
`brand_id`, `categories_id`, `quantity`, `rate`, `mrp`, `bno`, `expdate`,
`added_date`, `active`, `status`) VALUES
(1, 'Cipla Inhaler', 'tab.jpg', 1, 1, '50', '30', 40, '307002', '2022-02-28',
'2022-02-28', 1, 1),
(2, 'Abevia 200 SR Tablet', 'tab1.jpg', 2, 1, '30', '150', 200, '307003',
'2022-02-16', '2022-02-28', 1, 1),
(3, 'Arpizol 20 Tablet', 'tab3.jpg', 3, 3, '70', '200', 300, '307004', '2022-
03-13', '2022-02-28', 1, 1),
(4, 'DOLO 650mg', 'tab4.jpg', 4, 1, '500', '25', 30, '307005', '2022-05-31',
'2022-04-15', 1, 1);

-- --------------------------------------------------------

--
-- Table structure for table `users`
--

CREATE TABLE `users` (
  `user_id` int(11) NOT NULL,
  `username` varchar(255) NOT NULL,
  `password` varchar(255) NOT NULL,
  `email` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `users`
--
```

```sql
INSERT INTO `users` (`user_id`, `username`, `password`, `email`) VALUES
(1, 'admin', '7dd2259de9fef85fa6a0a04423a0dbc6', 'nowdemy@sample.com');

--
-- Indexes for dumped tables
--

--
-- Indexes for table `brands`
--
ALTER TABLE `brands`
  ADD PRIMARY KEY (`brand_id`);

--
-- Indexes for table `categories`
--
ALTER TABLE `categories`
  ADD PRIMARY KEY (`categories_id`);

--
-- Indexes for table `orders`
--
ALTER TABLE `orders`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `order_item`
--
ALTER TABLE `order_item`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `product`
--
ALTER TABLE `product`
  ADD PRIMARY KEY (`product_id`);

--
-- Indexes for table `users`
--
ALTER TABLE `users`
  ADD PRIMARY KEY (`user_id`);

--
-- AUTO_INCREMENT for dumped tables
--

--
```

```sql
-- AUTO_INCREMENT for table `brands`
--
ALTER TABLE `brands`
  MODIFY `brand_id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;

--
-- AUTO_INCREMENT for table `categories`
--
ALTER TABLE `categories`
  MODIFY `categories_id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;

--
-- AUTO_INCREMENT for table `orders`
--
ALTER TABLE `orders`
  MODIFY `id` int(15) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;

--
-- AUTO_INCREMENT for table `order_item`
--
ALTER TABLE `order_item`
  MODIFY `id` int(15) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=11;

--
-- AUTO_INCREMENT for table `product`
--
ALTER TABLE `product`
  MODIFY `product_id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;

--
-- AUTO_INCREMENT for table `users`
--
ALTER TABLE `users`
  MODIFY `user_id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
COMMIT;
```

NORMALISATION OF DATABASE

Normalization technique is used to break the large tables into smaller tables so that redundant data can be avoided. Normalization avoids different types of insertion, updation and deletion anomalies. Anomalies can create errors while interacting with databases.

Order_i8998+9d	User_id	User_name	Password	Email

Eliminate repeating groups and separate each order_id.

1st normal form
A table in the First Normal Form must meet the following requirements:

- Set atomic characteristics and columns.
- The data in the fields should all be from the similar data type.
- Each column in a table should have its own name.
- It also makes no difference in which order the data is stored.

Id	Product_name	Quatity	Rate	Total	Lastid

2nd normal form
A table in the Second Normal Form must meet the following requirements:

- It must be written in the first normal form; and
- It must be free of any partial dependency.

product_id	product_name	product_image	brand_id	categories_id	quantity	rate	mrp

The Product table fields are totally dependent on the primary keys, and we can establish more relationships from the Order_item table.

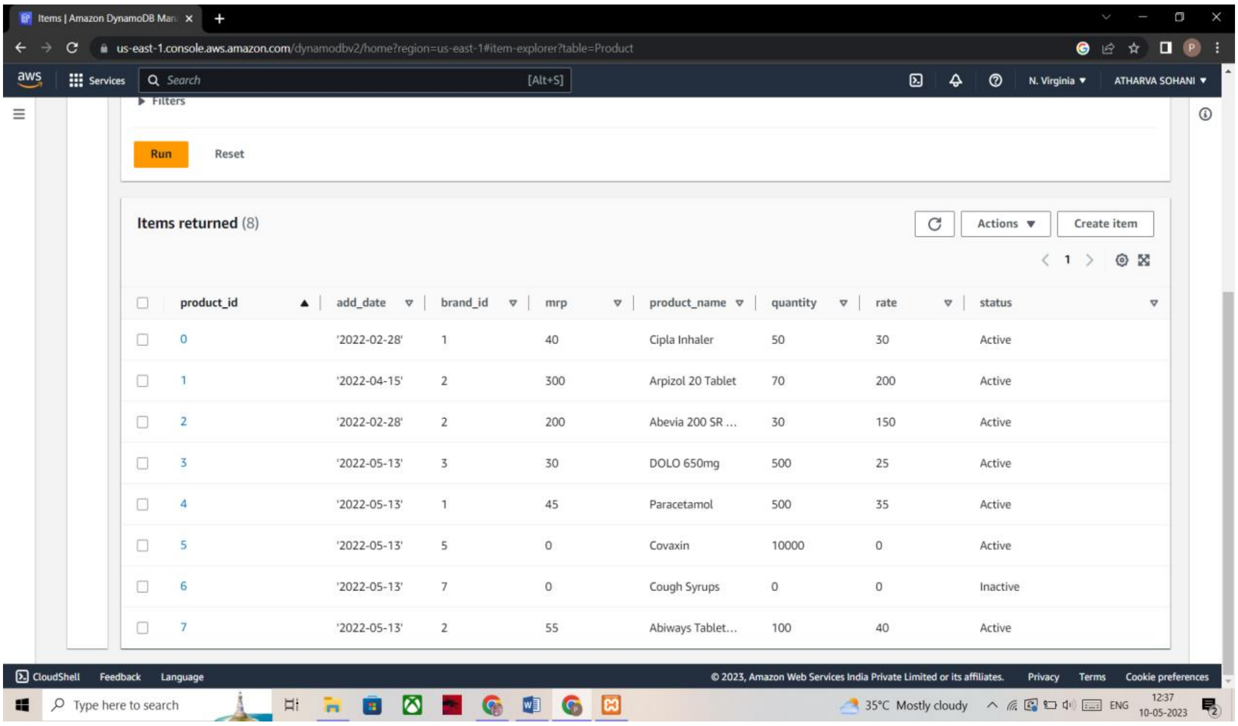
3rd normal form
The properties of a table in a Third Normal Form are as follows:

- The table must meet the rules of second normal form; and
- There should be no transitive dependencies.

Order_id	Order_date	ClientName	Address	Totalamt	gstn	paymentStatus

IMPLEMENTATION OF DYNAMODB

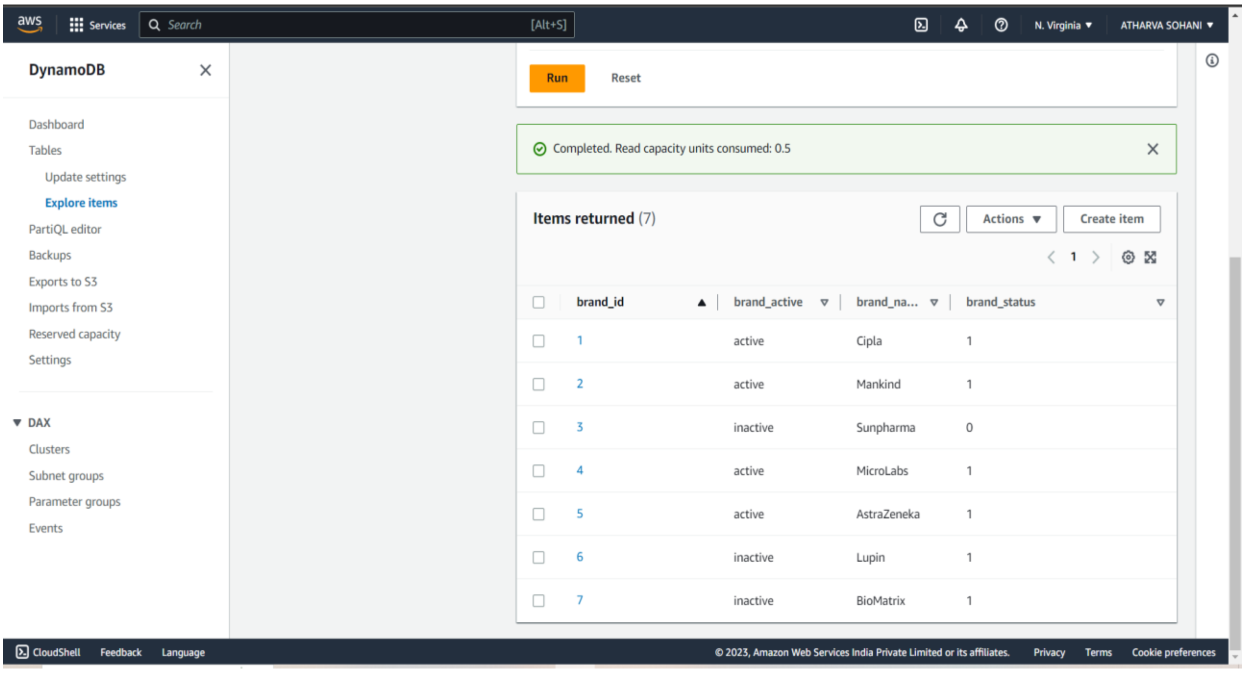
Products table:



Items returned (8)

product_id	add_date	brand_id	mrp	product_name	quantity	rate	status
0	'2022-02-28'	1	40	Cipla Inhaler	50	30	Active
1	'2022-04-15'	2	300	Arpizol 20 Tablet	70	200	Active
2	'2022-02-28'	2	200	Abevia 200 SR ...	30	150	Active
3	'2022-05-13'	3	30	DOLO 650mg	500	25	Active
4	'2022-05-13'	1	45	Paracetamol	500	35	Active
5	'2022-05-13'	5	0	Covaxin	10000	0	Active
6	'2022-05-13'	7	0	Cough Syrups	0	0	Inactive
7	'2022-05-13'	2	55	Abiways Tablet...	100	40	Active

Brands table:



Completed. Read capacity units consumed: 0.5

Items returned (7)

brand_id	brand_active	brand_na...	brand_status
1	active	Cipla	1
2	active	Mankind	1
3	inactive	Sunpharma	0
4	active	MicroLabs	1
5	active	AstraZeneka	1
6	inactive	Lupin	1
7	inactive	BioMatrix	1

DynamoDB:

Creating DynamoDb table using AWS CLI:

1. Open AWS CloudShell and select the region where you want to create the table. You can choose the region from the dropdown menu in the top-right corner of the CloudShell window.
2. Launch the AWS CLI terminal in CloudShell by clicking on the "Open Terminal" button in the top-right corner of the CloudShell window.
3. Create a JSON file with the table schema. For example, create a file named table_schema.json with the following contents:

JSON - Using a JSON document database, you can store each user's profile efficiently by storing only the attributes that are specific to each user.

EX:
```
echo '{
"TableName": "my_table",
"KeySchema": [
{
"AttributeName": "my_partition_key",
"KeyType": "HASH"
},
{
"AttributeName": "my_sort_key",
"KeyType": "RANGE"
}
],
"AttributeDefinitions": [
{
"AttributeName": "my_partition_key",
"AttributeType": "S"
},
{
"AttributeName": "my_sort_key",
"AttributeType": "N"
}
],
"BillingMode": "PAY_PER_REQUEST"
}' > table_schema.json
```

Create the table by running the following AWS CLI command:

```
aws dynamodb create-table --cli-input-json file://table_schema.json
```

To check the status of the table by running the following AWS CLI command:

```
aws dynamodb describe-table --table-name my_table
```

To add data:

```
{
"my_partition_key": {"S": "my_partition_key_value"},
"my_sort_key": {"N": "1"},
"attribute1": {"S": "value1"},
"attribute2": {"N": "2"}
}
```

Use the 'aws dynamodb put-item' command to add the item to the table. Run the following command:

```
aws dynamodb put-item --table-name my_table --item file://item.json
```

Verify that the item has been added by using the aws dynamodb get-item command. Run the following command:

```
aws dynamodb get-item --table-name my_table --key '{"my_partition_key": {"S":
"my_partition_key_value"}, "my_sort_key": {"N": "1"}}'
```

You can add more items by repeating the steps above with different data in the ' item.json' file.

Scan Operations:

Scanning a table for all items:

This command scans the my_table table and returns all items in the table

```
aws dynamodb scan --table-name my_table
```

If the table has many items, you may need to paginate the results to retrieve all items. To do this, add the --page-size parameter to the aws dynamodb scan command and specify the maximum number of items to retrieve per page. For example, to retrieve 100 items per page, run the following command:

```
aws dynamodb scan --table-name my_table --page-size 100
```

Scanning a table with filter expression:

First run this command to scan with a filter expression.

```
aws dynamodb scan --table-name my_table --filter-expression "attribute1 = :value" --
expression-attribute-values '{":value":{"S":"value1"}}'
```

Next verify the results by this command

```
aws dynamodb scan --table-name my_table --filter-expression "attribute1 = :value" --
expression-attribute-values '{":value":{"S":"value1"}}' | jq '.Items[].{attribute1:.attribute1.S,
attribute2:.attribute2.N}'
```

Scanning a table with a projection expression:

Run the aws dynamodb scan command with the --table-name parameter to specify the name of the table you want to scan, and the --projection-expression parameter to specify the attributes you want to include in the scan. For example, to scan a table named my_table and include only the attribute1 and attribute2 attributes in the results, run the following command:

```
aws dynamodb scan -- table-name  my_table --  projection-expression "attribute1, attribute2" -
-max-items 100
```

Scanning a table with a limit:

```
aws dynamodb scan --table-name my_table --limit 10
```

Scanning table with a parallel scan:

```
aws dynamodb scan --table-name my_table --total-segments 10
```

and use this for scanning the first segment:

```
aws dynamodb scan --table-name my_table --total-segments 10 --segment 0.
```

CONCLUSION

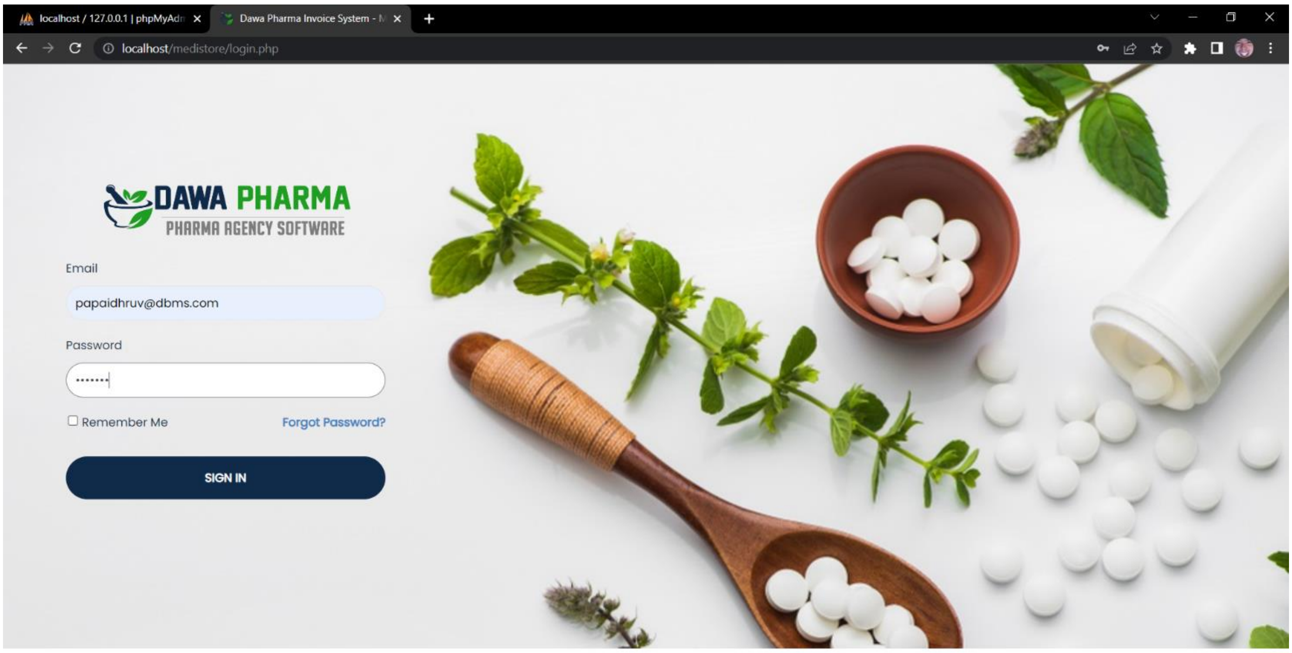
Pharmacy management system is actually a software which handle the essential data and save the data and actually about the database of a pharmacy and its management. This software helps in effectively management of the pharmaceutical store or shop. It provides the statistics about medicine or drugs which are in stocks which data can also be updated and edited. It works as per the requirement of the user and have options accordingly. It allow user to enter manufacturing as well as the expiry date of medicine placing in stock and for sales transaction. This software also has ability to print reports and receipts etc. There is other function available too. The main purpose is effectively and easily handling of pharmacy data and its management

The role of pharmacy management is to supervise and manage the pharmacy employees to preserve excellent working relationships and outcomes. To improve the services offered in every Pharmacy, the formulated proposal, abstract and modules of Pharmacy Management System Project management should be present in the PDF document.

And that ends our discussion about Pharmacy Management System Project Proposal.

APPENDIX 1

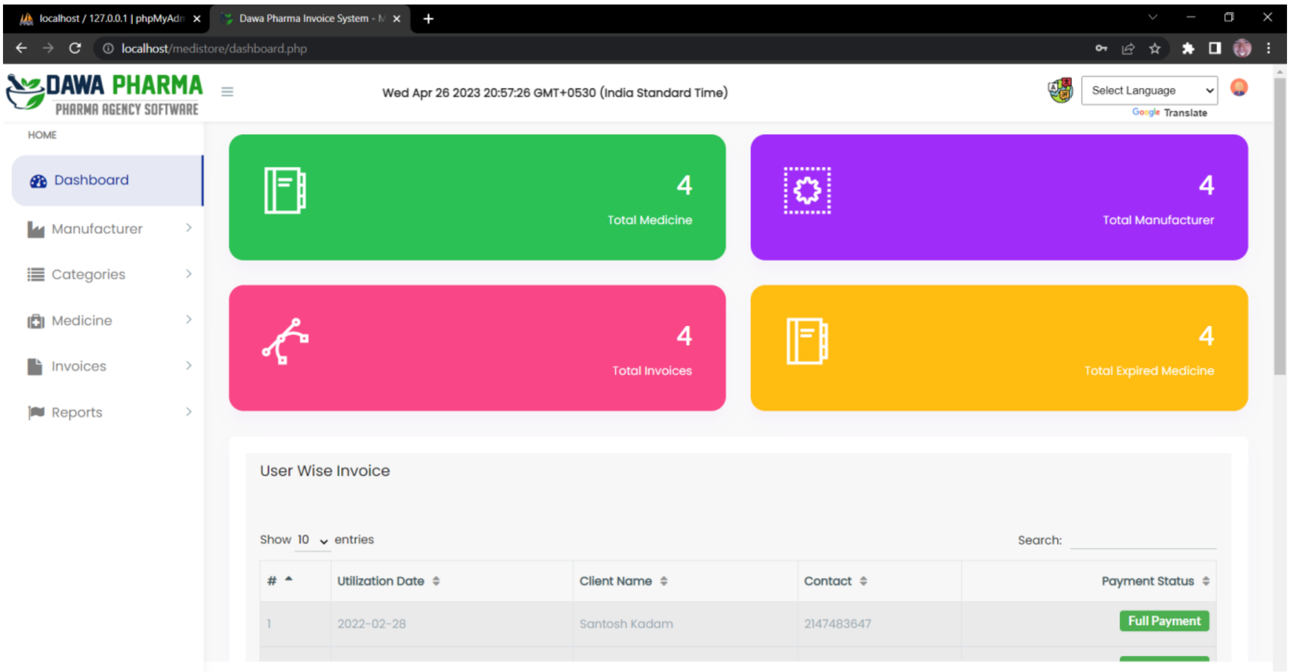
1. Login page interface


localhost / 127.0.0.1 | phpMyAdn
Dawa Pharma Invoice System - M
localhost/medistore/login.php
DAWA PHARMA
PHARMA AGENCY SOFTWARE
□ Remember Me
Forgot Password?
SIGN IN

Email: papaidhruv@dbms.com

Password: ••••••••

2. Dashboard/Home page interface



Wed Apr 26 2023 20:57:26 GMT+0530 (India Standard Time)

4 Total Medicine

4 Total Manufacturer

4 Total Invoices

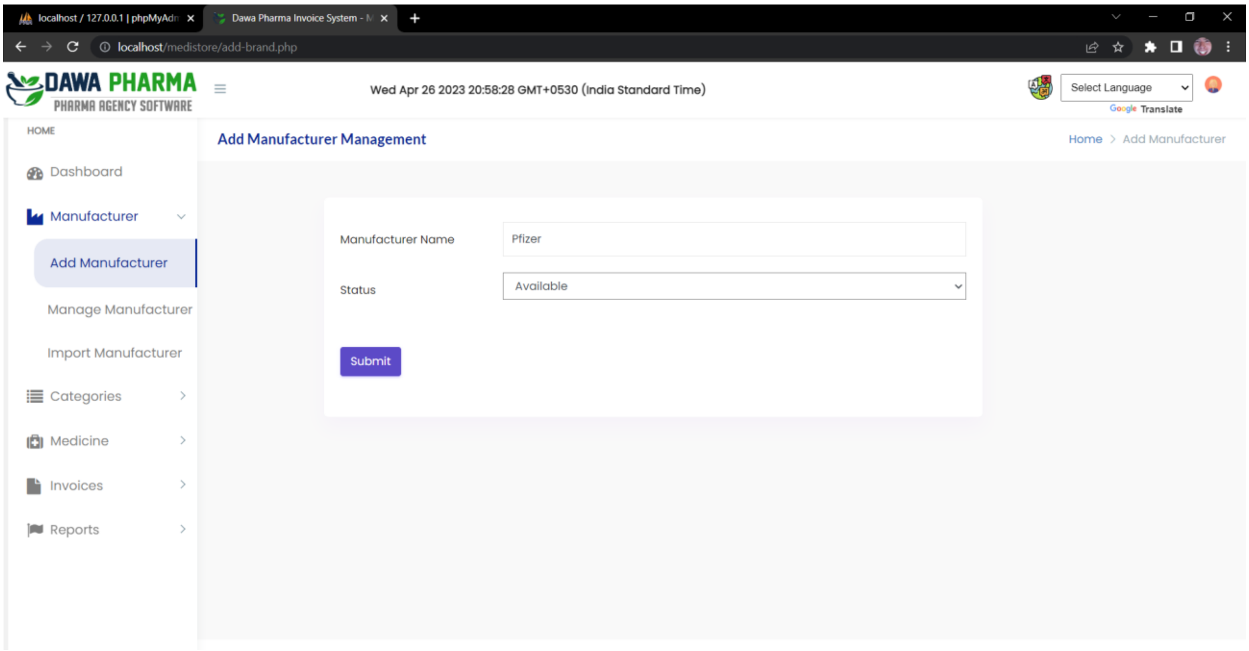
4 Total Expired Medicine

User Wise Invoice

Show 10 entries Search:

#	Utilization Date	Client Name	Contact	Payment Status
1	2022-02-28	Santosh Kadam	2147483647	Full Payment

2. Add Manufactures function



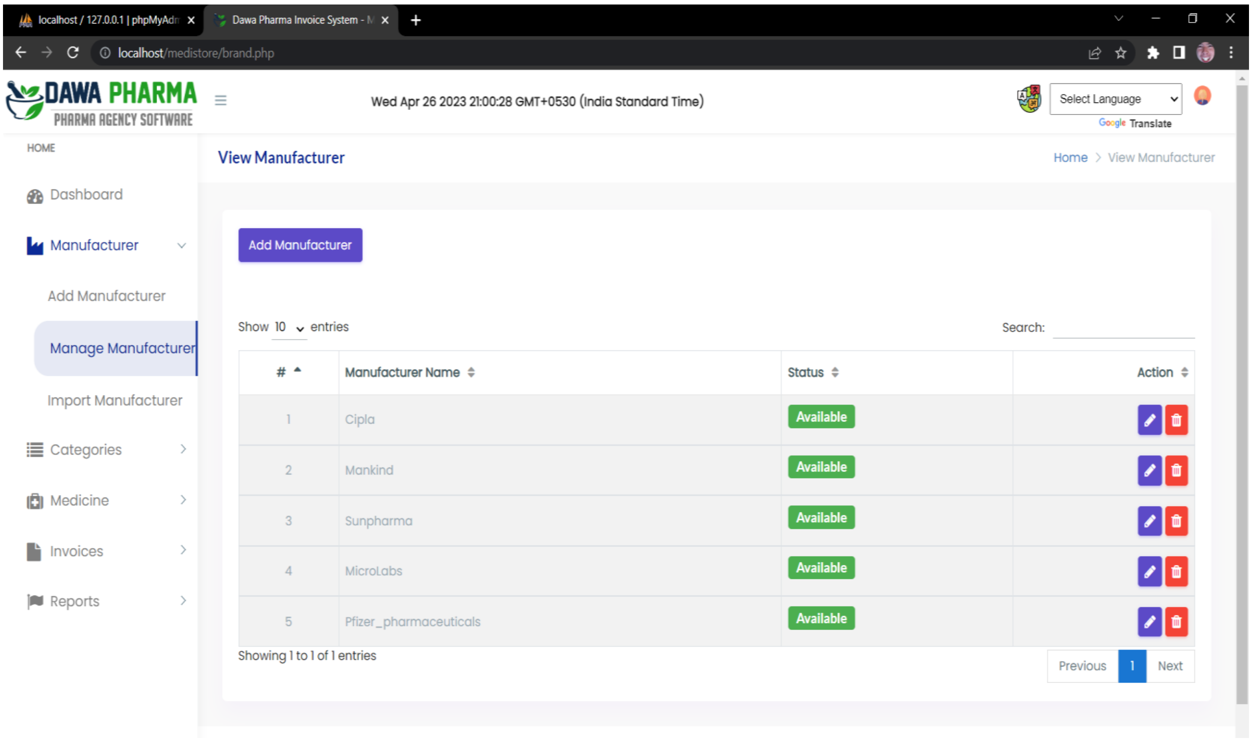
Wed Apr 26 2023 20:58:28 GMT+0530 (India Standard Time)

Add Manufacturer Management

Manufacturer Name	Pfizer
Status	Available

Submit

3. Manage Manufacturers function



Wed Apr 26 2023 21:00:28 GMT+0530 (India Standard Time)

View Manufacturer

Add Manufacturer

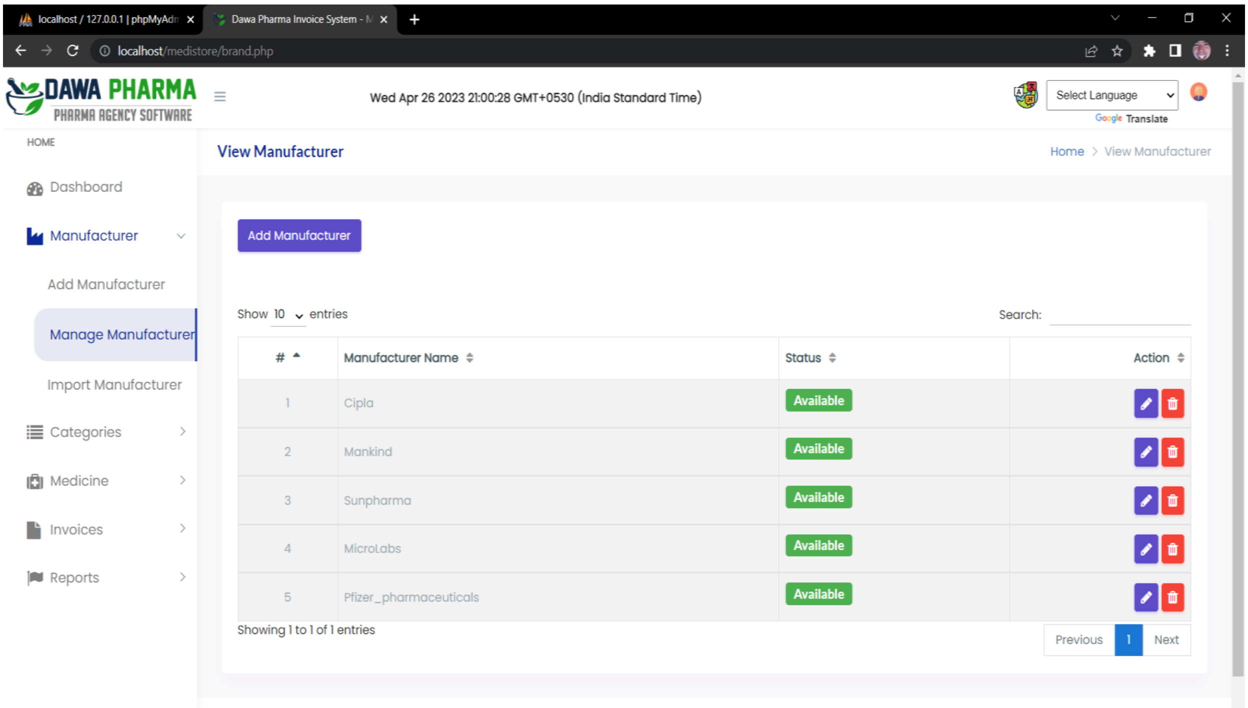
Show 10 entries Search:

#	Manufacturer Name	Status	Action
1	Cipla	Available	
2	Mankind	Available	
3	Sunpharma	Available	
4	MicroLabs	Available	
5	Pfizer_pharmaceuticals	Available	

Showing 1 to 1 of 1 entries

Previous 1 Next

4. Add medicine category



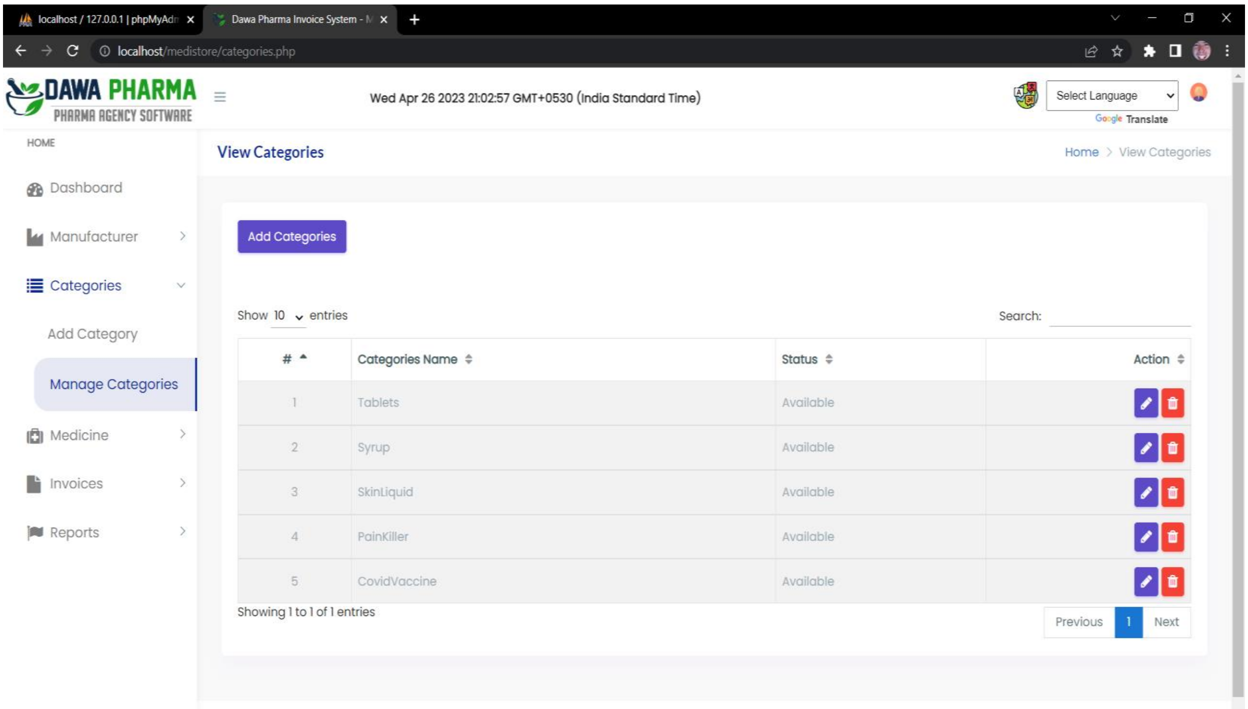
Wed Apr 26 2023 21:00:28 GMT+0530 (India Standard Time)

View Manufacturer

#	Manufacturer Name	Status	Action
1	Cipla	Available	
2	Mankind	Available	
3	Sunpharma	Available	
4	MicroLabs	Available	
5	Pfizer_pharmaceuticals	Available	

Showing 1 to 1 of 1 entries

5. Manage Medicine Category



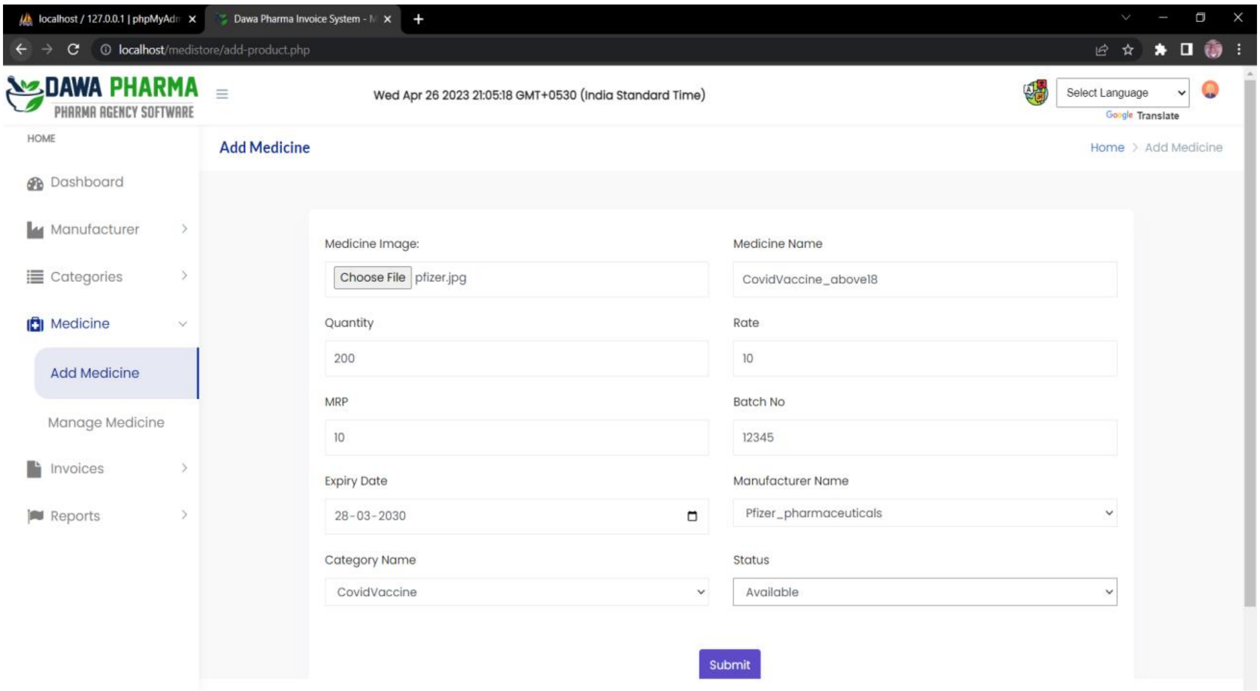
Wed Apr 26 2023 21:02:57 GMT+0530 (India Standard Time)

View Categories

#	Categories Name	Status	Action
1	Tablets	Available	
2	Syrup	Available	
3	SkinLiquid	Available	
4	PainKiller	Available	
5	CovidVaccine	Available	

Showing 1 to 1 of 1 entries

6. Add new Medicine page



Wed Apr 26 2023 21:05:18 GMT+0530 (India Standard Time)

Add Medicine

Medicine Image: Choose File pfizer.jpg

Medicine Name: CovidVaccine_above18

Quantity: 200

Rate: 10

MRP: 10

Batch No: 12345

Expiry Date: 28-03-2030

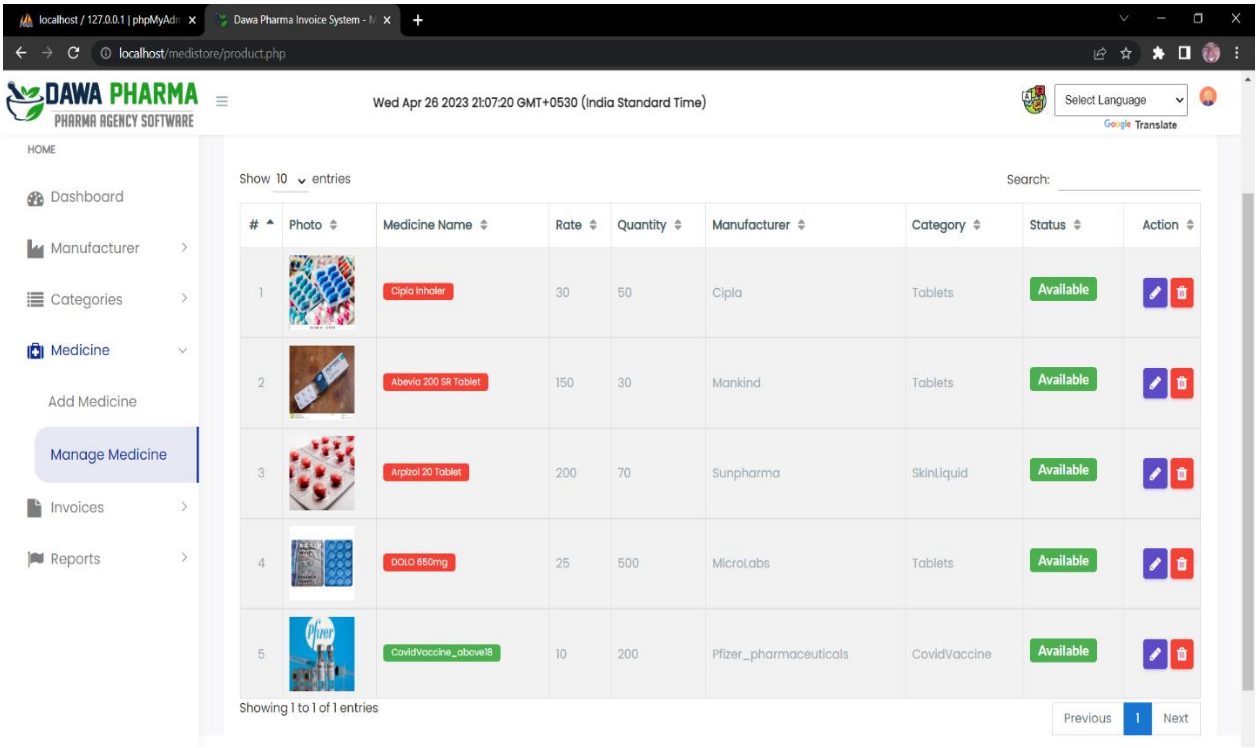
Manufacturer Name: Pfizer_pharmaceuticals

Category Name: CovidVaccine

Status: Available

Submit

7. Manage Medicines page



Wed Apr 26 2023 21:07:20 GMT+0530 (India Standard Time)

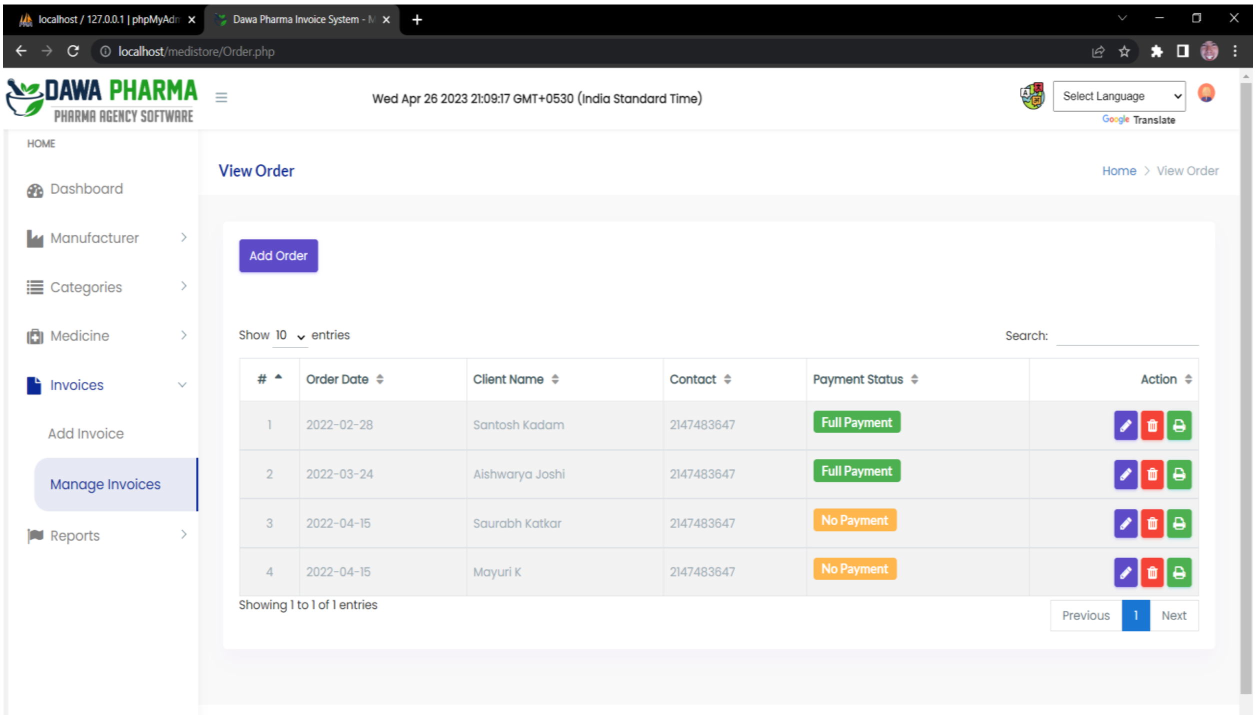
Show 10 entries Search:

#	Photo	Medicine Name	Rate	Quantity	Manufacturer	Category	Status	Action
1		Cipla Inhaler	30	50	Cipla	Tablets	Available	
2		Abevia 200 SR Tablet	150	30	Mankind	Tablets	Available	
3		Arpizol 20 Tablet	200	70	Sunpharma	SkinLiquid	Available	
4		DOLO 650mg	25	500	MicroLabs	Tablets	Available	
5		CovidVaccine_above18	10	200	Pfizer_pharmaceuticals	CovidVaccine	Available	

Showing 1 to 1 of 1 entries

Previous 1 Next

8. Manage orders



Wed Apr 26 2023 21:09:17 GMT+0530 (India Standard Time)

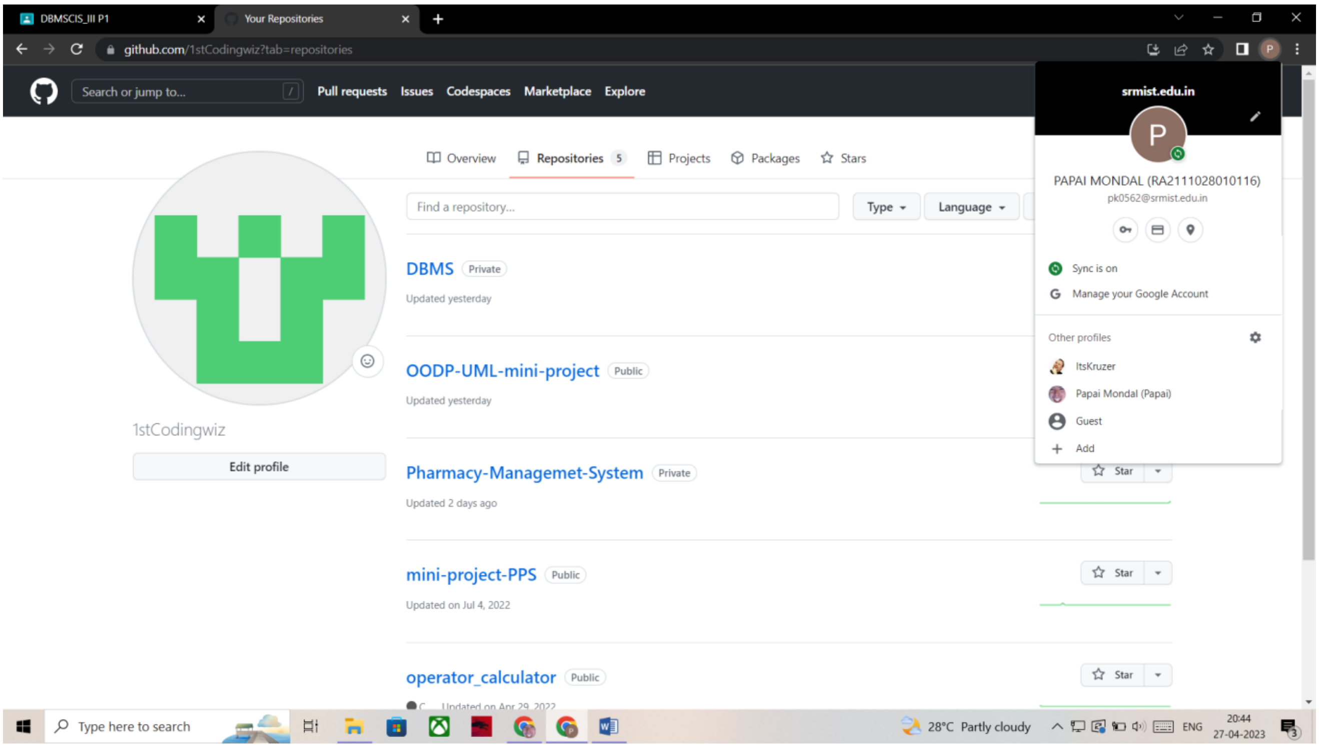
View Order

#	Order Date	Client Name	Contact	Payment Status	Action
1	2022-02-28	Santosh Kadam	2147483647	Full Payment	
2	2022-03-24	Aishwarya Joshi	2147483647	Full Payment	
3	2022-04-15	Saurabh Katkar	2147483647	No Payment	
4	2022-04-15	Mayuri K	2147483647	No Payment	

Showing 1 to 1 of 1 entries

APPENDIX 2

Papai Mondal (RA2111028010116)
Github Profile name- 1stCodingwiz
Github Link - https://github.com/1stCodingwiz



PAPAI MONDAL (RA2111028010116)
pk0562@srmist.edu.in

1stCodingwiz

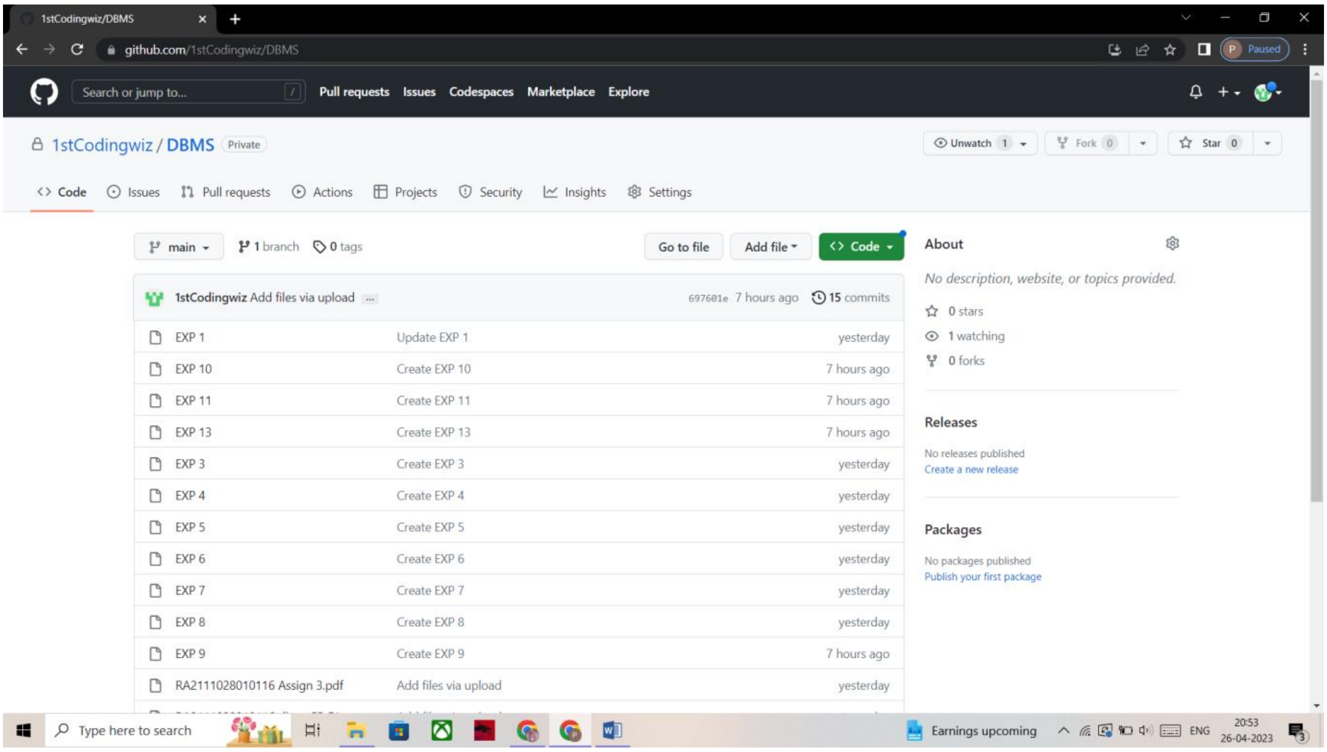
DBMS Private
Updated yesterday

OODP-UML-mini-project Public
Updated yesterday

Pharmacy-Managemet-System Private
Updated 2 days ago

mini-project-PPS Public
Updated on Jul 4, 2022

operator_calculator Public
Updated on Apr 29, 2022



1stCodingwiz / DBMS Private

1stCodingwiz Add files via upload ... 697601e 7 hours ago 15 commits

EXP 1	Update EXP 1	yesterday
EXP 10	Create EXP 10	7 hours ago
EXP 11	Create EXP 11	7 hours ago
EXP 13	Create EXP 13	7 hours ago
EXP 3	Create EXP 3	yesterday
EXP 4	Create EXP 4	yesterday
EXP 5	Create EXP 5	yesterday
EXP 6	Create EXP 6	yesterday
EXP 7	Create EXP 7	yesterday
EXP 8	Create EXP 8	yesterday
EXP 9	Create EXP 9	7 hours ago
RA2111028010116 Assign 3.pdf	Add files via upload	yesterday

About

No description, website, or topics provided.

0 stars
1 watching
0 forks

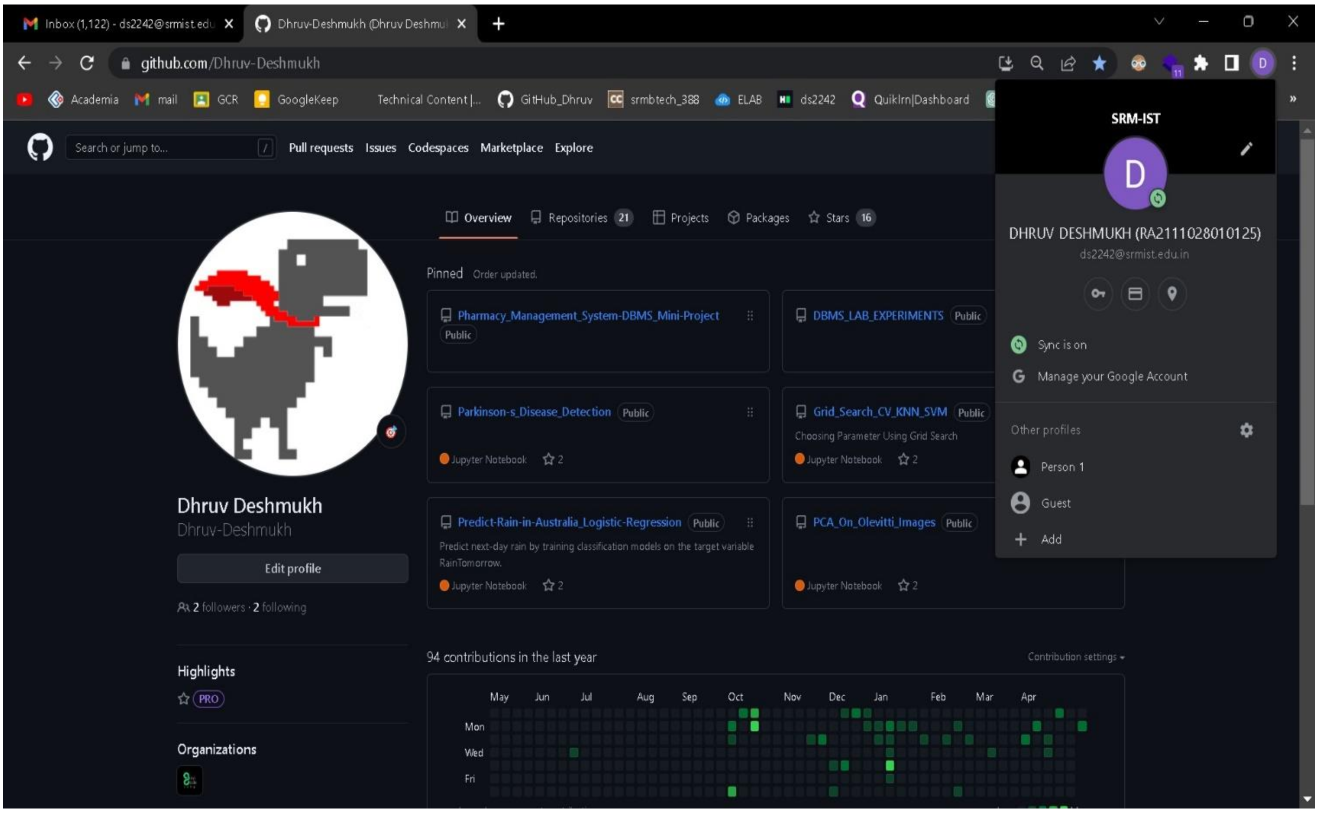
Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Dhruv Deshmukh (RA2111028010125)
Github Profile name – Dhruv-Deshmukh
Github link - https://github.com/Dhruv-Deshmukh



DHRUV DESHMUKH (RA2111028010125)
ds2242@srmist.edu.in

Dhruv Deshmukh
Dhruv-Deshmukh

2 followers · 2 following

Pinned Order updated.

Pharmacy_Management_System-DBMS_Mini-Project Public

DBMS_LAB_EXPERIMENTS Public

Parkinson-s_Disease_Detection Public
Jupyter Notebook ☆ 2

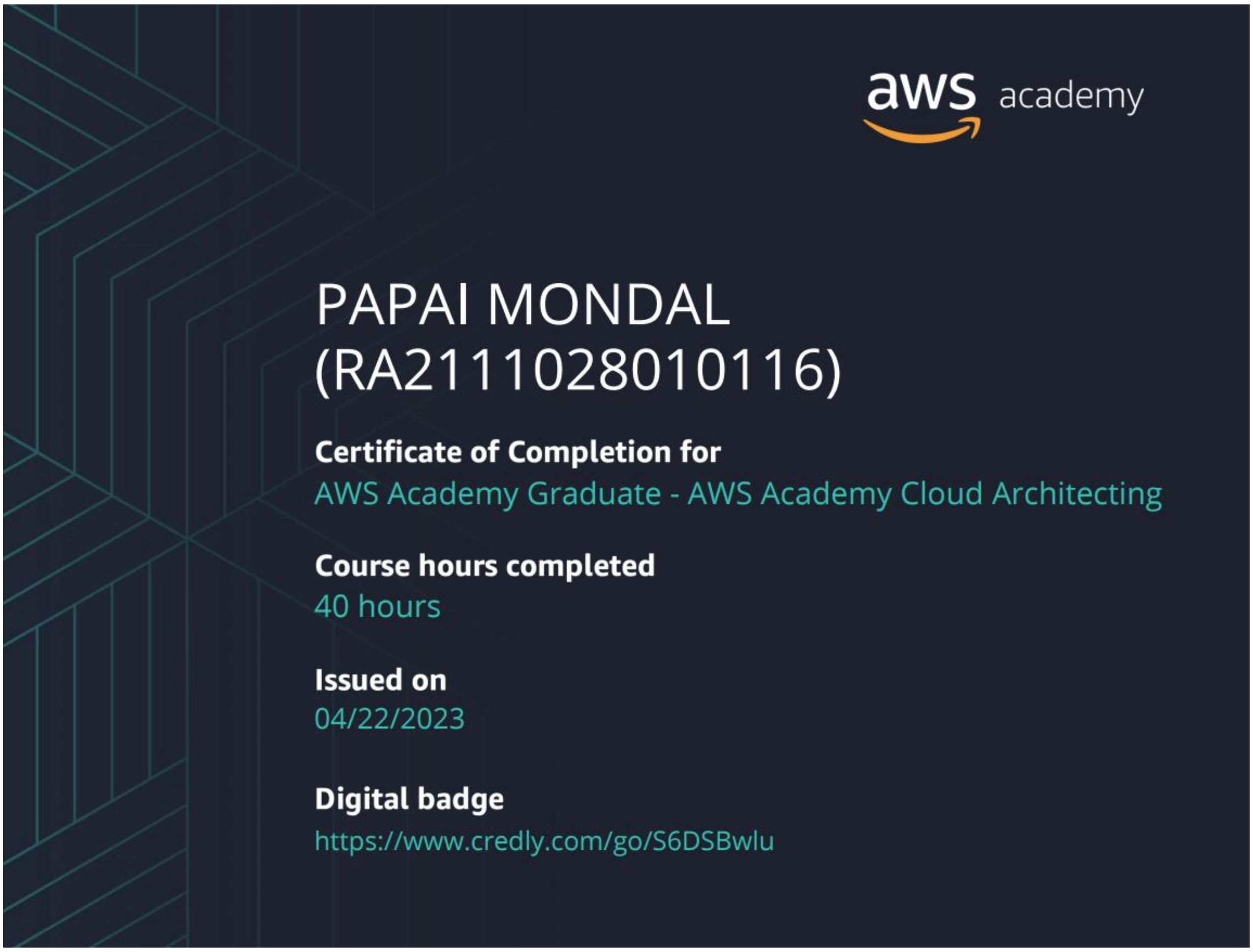
Grid_Search_CV_KNN_SVM Public
Choosing Parameter Using Grid Search
Jupyter Notebook ☆ 2

Predict-Rain-in-Australia_Logistic-Regression Public
Predict next-day rain by training classification models on the target variable RainTomorrow.
Jupyter Notebook ☆ 2

PCA_On_Olevitti_Images Public
Jupyter Notebook ☆ 2

94 contributions in the last year

APPENDIX 3



aws academy

PAPAI MONDAL (RA2111028010116)

Certificate of Completion for

AWS Academy Graduate - AWS Academy Cloud Architecting

Course hours completed

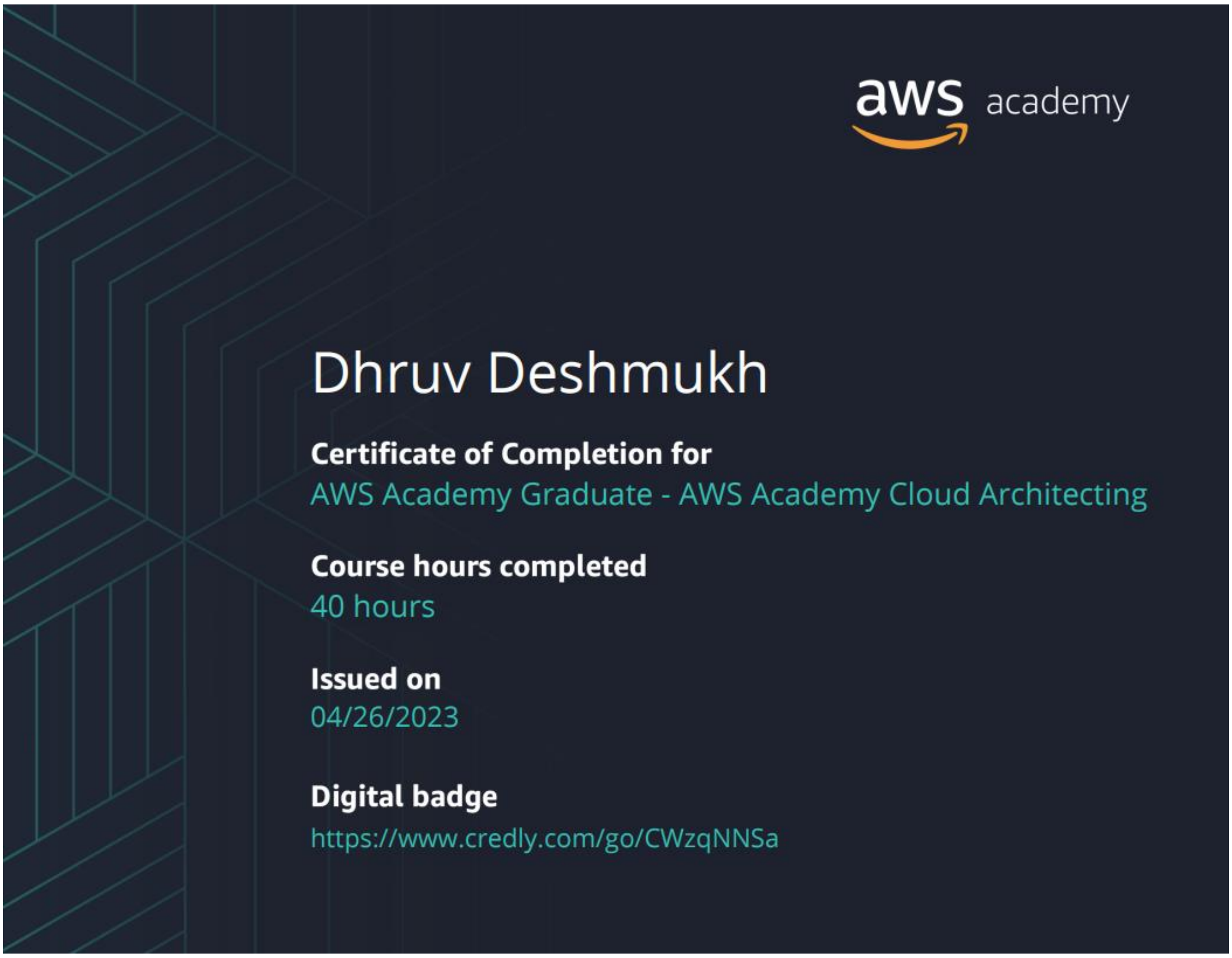
40 hours

Issued on

04/22/2023

Digital badge

https://www.credly.com/go/S6DSBwlu



aws academy

Dhruv Deshmukh

Certificate of Completion for

AWS Academy Graduate - AWS Academy Cloud Architecting

Course hours completed

40 hours

Issued on

04/26/2023

Digital badge

https://www.credly.com/go/CWzqNNSa