

VISVESVARAYA TECHNOLOGICAL UNIVERSITY  
“JNANA SANGAMA”, BELAGAVI - 590 018



A MINI PROJECT REPORT

on

“Nursery Management System”

*Submitted by*

A M Akhilesh

4SF19IS001

Jackson Lobo

4SF19IS031

*In partial fulfillment of the requirements for the V semester*

DBMS LABORATORY WITH MINI PROJECT

of

BACHELOR OF ENGINEERING

in

INFORMATION SCIENCE & ENGINEERING

*Under the Guidance of*

Ms. Jayapadmini Kanchan

Assistant Professor, Department of ISE

at



SAHYADRI

College of Engineering & Management

Adyar, Mangaluru - 575 007

2021 - 22

**SAHYADRI**  
**College of Engineering & Management**  
Adyar, Mangaluru - 575 007

**Department of Information Science & Engineering**



**CERTIFICATE**

This is to certify that the **Mini Project** entitled “**Nursery Management System**” has been carried out by **A M Akhilesh (4SF19IS001)** and **Jackson Lobo (4SF19IS031)**, the bonafide students of Sahyadri College of Engineering & Management in partial fulfillment of the requirements for the V semester **DBMS Laboratory with Mini Project (18CSL58)** of **Bachelor of Engineering in Information Science & Engineering** of Visvesvaraya Technological University, Belagavi during the year 2021 - 22. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements in respect of mini project work.

---

**Ms. Jayapadmini Kanchan**  
Assistant Professor  
Dept. of ISE, SCEM

---

**Mr. Ganaraj K**  
Assistant Professor  
Dept. of ISE, SCEM

---

**Dr. Shamanth Rai**  
HOD & Associate Professor  
Dept. of ISE, SCEM

**External Practical Examination:**

Examiner's Name

Signature with Date

1. ....

.....

2. ....

.....

**SAHYADRI**  
**College of Engineering & Management**  
**Adyar, Mangaluru - 575 007**

**Department of Information Science & Engineering**



**DECLARATION**

We hereby declare that the entire work embodied in this Mini Project Report titled “**Nursery Management System**” has been carried out by us at Sahyadri College of Engineering and Management, Mangaluru under the supervision of **Ms. Jayapadmini Kanchan** as the part of the V semester **DBMS Laboratory with Mini Project (18CSL58)** of **Bachelor of Engineering in Information Science & Engineering**. This report has not been submitted to this or any other University.

**A M Akhilesh(4SF19IS001)**

**Jackson Lobo(4SF19IS031)**

SCEM, Mangaluru

# Abstract

Nursery management system is a project which aims in developing a computerized system to maintain all the daily activities in a Nursery and also interact with the customers. This system helps both nursery owner and as well as its customers to keep a constant track of all the plants available in the Nursery. It allows the admin to maintain a record of his plants,customers and employees. The customers also can view their records by logging in and can buy or sell the plants to the nursery.This system will help the local nurseries to manage their work in a better way. It will also help the billing and invoicing to work in a better way. The meter reader takes the monthly reading and forwards the bill to respective consumer and to the respective ward. Afterwards the consumer can pay the bill through online or offline method and sometimes people will not remember of the payment of the water supply bills. At that time this application will remind you. And the one more advantage is you can give any complaints regarding water supply.

# Acknowledgement

It is with great satisfaction and euphoria that we are submitting the Mini Project Report on “**Nursery Management System**”. We have completed it as a part of the V semester **DBMS Laboratory with Mini Project (18CSL58)** of **Bachelor of Engineering in Information Science & Engineering** of Visvesvaraya Technological University, Belagavi.

We are profoundly indebted to our guide, **Ms. Jayapadmini Kanchan**, Assistant Professor, Department of Information Science & Engineering for innumerable acts of timely advice, encouragement and We sincerely express our gratitude.

We express our sincere gratitude to **Dr. Shamanth Rai**, Head & Associate Professor, Department of Information Science & Engineering for his invaluable support and guidance.

We sincerely thank **Dr. Rajesha S**, Principal, Sahyadri College of Engineering & Management and **Dr. D. L. Prabhakara**, Director, Sahyadri Educational Institutions, who have always been a great source of inspiration.

Finally, yet importantly, We express our heartfelt thanks to our family & friends for their wishes and encouragement throughout the work.

**A M Akhilesh**

4SF19IS001

V Sem, B.E., ISE

SCEM, Mangaluru

**Jackson Lobo**

4SF19IS031

V Sem, B.E., ISE

SCEM, Mangaluru

# Table of Contents

<b>Abstract</b>	<b>i</b>
<b>Acknowledgement</b>	<b>ii</b>
<b>Table of Contents</b>	<b>iii</b>
<b>List of Figures</b>	<b>iv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Purpose . . . . .	1
1.2 Scope . . . . .	2
1.3 Overview . . . . .	2
<b>2 Requirements Specification</b>	<b>3</b>
2.1 Hardware Specification . . . . .	3
2.2 Software Specification . . . . .	3
<b>3 System Design</b>	<b>4</b>
3.1 ER Diagram . . . . .	4
3.2 Mapping From ER Diagram to Schema Diagram . . . . .	5
3.3 Assumptions . . . . .	6
3.4 Schema Diagram . . . . .	7
<b>4 Implementation</b>	<b>9</b>
4.1 Pseudocode for Nursery Management System . . . . .	9
4.2 Tables used . . . . .	11
<b>5 Results and Discussion</b>	<b>13</b>
<b>6 Conclusion and Future work</b>	<b>19</b>
<b>References</b>	<b>20</b>

# List of Figures

3.1	ER Diagram of Nursery Management System . . . . .	4
3.2	Schema Diagram for Nursery Management System . . . . .	8
4.1	Pseudocode for connection to database . . . . .	9
4.2	Pseudocode for Inserting values into the database . . . . .	10
4.3	Pseudocode for Updation . . . . .	10
4.4	Structure of Nursery table . . . . .	11
4.5	Structure of Plants table . . . . .	11
4.6	Structure of Customers table . . . . .	12
4.7	Structure of Employees table . . . . .	12
5.1	Register Page . . . . .	13
5.2	Login Page . . . . .	13
5.3	Customer homepage . . . . .	14
5.4	Admin homepage . . . . .	14
5.5	Customer views all the Plants . . . . .	15
5.6	Customer providing details while selling a plant . . . . .	15
5.7	Admin views all the Plants present in his Nursery . . . . .	16
5.8	Admin views all the Employees working in his Nursery . . . . .	16
5.9	Admin views all the Customers who have visited his Nursery . . . . .	17
5.10	Customer's cart containing plants to buy . . . . .	17
5.11	Invoice after the Customer checks out . . . . .	18
5.12	Out of Stock . . . . .	18

# Chapter 1

## Introduction

Nursery management system is a project which aims in developing a computerized system to maintain all the daily activities in a Nursery and also interact with the customers. This system helps both nursery owner and as well as its customers to keep a constant track of all the plants available in the Nursery. It allows the admin to maintain a record of his plants, customers and employees. The customers also can view their records by logging in and can buy or sell the plants to the nursery. This system will help the local nurseries to manage their work in a better way. s. This system will provide the detailed description of the plants to customers so that they can view the different plant types that are available in the nursery and will buy the one which is more suitable to them . It will also help the billing and invoicing to work in a better way.

### 1.1 Purpose

The main aim of Nursery Management Sysytem is used to keep and maintain a record of plants in a nursery and allow customers to buy plants as well as sell their plants to the Nursery. We aim to demonstrate the use of create, read, update and delete (CRUD) MySQL operations through this project. The project starts by creating a Nursery and adding details of employee into it. The role of an employee is to manage daily activities in a Nursery i.e to look after the plants etc. The customer can register to the website by providing required details. This system will provide the detailed description of the plants to customers so that they can view the different plant types that are available in the nursery and will buy the one which is more suitable to them .



## 1.2 Scope

To mitigate the issues of conventional and manual method of keeping a record of transactions by writing it in a book, the Nursery management system has been created. This web-application can be used by local nurseries to manage the daily activities and keep a record of transactions with the customer. The owner of the nursery can make any modifications to the application like add a new plant or employee and update the existing information, which makes this system user friendly.

## 1.3 Overview

The project starts by creating a Nursery and adding details of employee into it. The role of an employee is to manage daily activities in a Nursery i.e to look after the plants etc. The customer can register to the website by providing required details. This system will provide the detailed description of the plants to customers so that they can view the different plant types that are available in the nursery and will buy the one which is more suitable to them.

# Chapter 2

## Requirements Specification

### 2.1 Hardware Specification

- Processor : Intel(R) Core(TM) i3-1005G1 CPU @ 1.20GHz 1.19 GHz
- RAM : 8GB
- Hard Disk : 1TB
- Input Device : Standard keyboard and Mouse
- Output Device : Monitor

### 2.2 Software Specification

- Database : MySQL 5.5
- Markup Language : HTML5
- Scripting Language: PHP 7.0.1
- IDE :Visual Studio Code

# Chapter 3

## System Design

### 3.1 ER Diagram

The project starts by creating a Nursery entity with attributes like N\_id, Name etc and then Plants, Customer and Employee entities are created. The Nursery, Plants, Customer and Employee are the strong entity types. The nursery has plants and the customer visits the nursery. The role of an employee is to manage daily activities in a Nursery i.e to look after the plants etc. The customer can register to the website by providing required details. This system will provide the detailed description of the plants to customers so that they can view the different plant types that are available in the nursery and will buy the one which is more suitable to them .

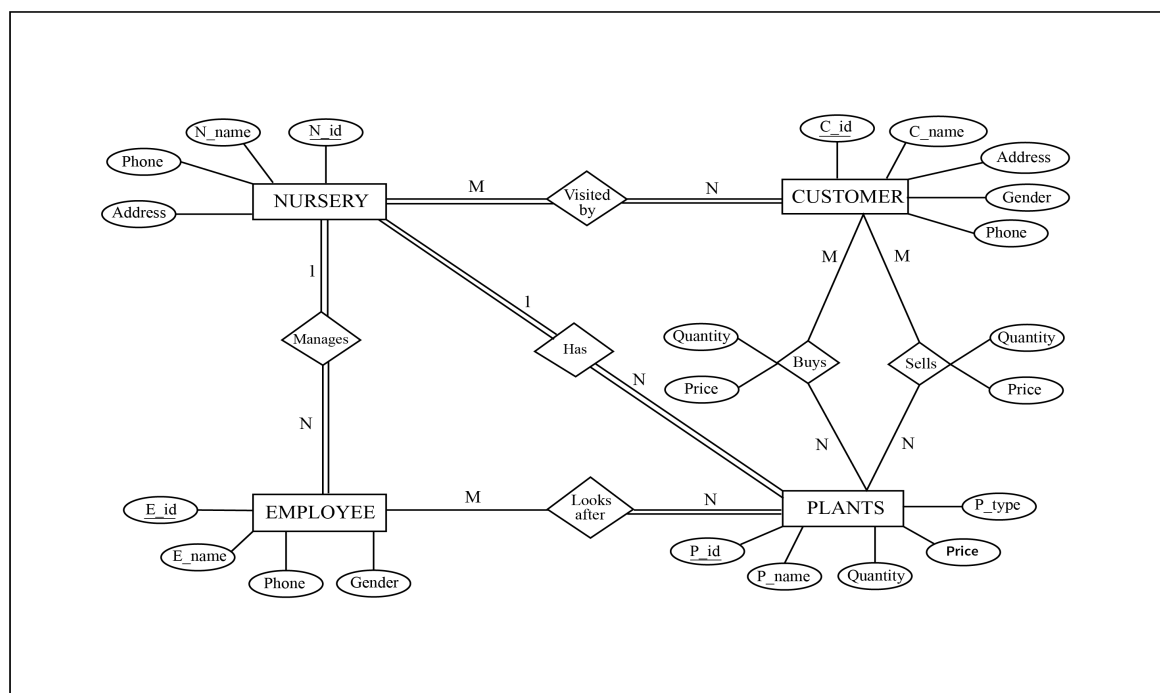


Figure 3.1: ER Diagram of Nursery Management System

## 3.2 Mapping From ER Diagram to Schema Diagram

**1.Mapping of Regular Entities:**This step involves mapping all the regular entity types to tabular format by identifying their primary keys.

**2.Mapping of 1:1 Relation:**In this step foreign keys are assigned using foreign key approach.The primary key of the participating relation R or S is added as primary key to second entity types by looking at the participating constraints.

**3.Mapping of 1:N Relation:**Foreign key approach is used to add one sided primary key to the n sided entity at foreign key.

**4.Mapping of M:N Relation:**Here we use the cross reference approach where the relationship is converted to a new relation within attributes on primary keys of both participating relation.

**5.Mapping of Weak Entity:**When mapping weak entity types along with other attributes the partial key and primary key of parent entity together will form their primary key of the new relation.

**6.Mapping of N-ary Relation:**For mapping N array relationship we create a new relation with a relationship name in its attribute and primary keys of all participating entity types.

**7.Mapping of Multivalued Relation:**For multivalued attributes a separate relation has to be created along with primary key of parent relation.

This database has the following mappings:

### Step – 1 : Mapping of Regular Entities.

From the ER diagram we identify all the strong entities E and create a relation R that includes all it's simple attributes and primary keys.

The following are the strong entities from our schema diagram :

- 1.NURSERY (N\_id, N\_name , Address , Phone )
- 2.PLANTS ( P\_id, P\_name , P\_type , Quantity , Price )
- 3.CUSTOMER (C\_id, C\_name , Address , Phone , Gender)
- 4.EMPLOYEE (E\_id , E\_name , Address , Phone , Gender )

### Step – 2 : Mapping of binary 1:1 Relation Types.

None of the entities are participating in the 1:1 relation type. In this step foreign keys are assigned using foreign key approach.The primary key of the participating relation R or S is added as primary key to second entity types by looking at the participating

constraints.

### Step – 3 : Mapping of binary 1:N Relation Types.

The NURSERY and EMPLOYEE entities are participating in the 1:N relation type. Since EMPLOYEE is the nth side of the relation, we include the primary key of NURSERY entity as the Foreign key in EMPLOYEE entity.

The NURSERY and PLANTS entities are participating in the 1:N relation type. Since PLANTS is the nth side of the relation, we include the primary key of NURSERY entity as the Foreign key in PLANTS entity.

- 1.NURSERY (N\_id, N\_name , Address , Phone )
- 2.PLANTS ( P\_id, P\_name , P\_type , Quantity , Price , N\_id )
- 3.CUSTOMER (C\_id, C\_name , Address , Phone , Gender)
- 4.EMPLOYEE (E\_id , E\_name , Address , Phone , Gender , N\_id )

### Step – 4 : Mapping of binary M:N Relation Types.

The relationship between the NURSERY and the CUSTOMER is M:N .So we create a new relation VISITED\_BY which includes the primary key of NURSERY and CUSTOMER entity. The combination of the two primary keys will form the primary key of the VISITED\_BY relation.

The relationship between the CUSTOMER and the PLANTS is M:N .So we create a new relation BUY which includes the primary key of CUSTOMER and PLANTS entity. The combination of the two primary keys will form the primary key of BUY relation.

The relationship between the CUSTOMER and the PLANTS is M:N .So we create a new relation SELL which includes the primary key of CUSTOMER and PLANTS entity. The combination of the two primary keys will form the primary key of SELL relation.

The relationship between the EMPLOYEE and the PLANTS is M:N .So we create a new relation LOOKS\_AFTER which includes the primary key of EMPLOYEE and PLANTS entity. The combination of the two primary keys will form the primary key of the LOOKS\_AFTER relation.

## 3.3 Assumptions

- There are multiple nurseries in the database, one nursery has many plants. Each plant name has different plant id's, same plant name will have different plant id's

in different nurseries.

- A customer can buy plants from the nursery with the required quantity as well as the customer can sell the plants to a nursery if nursery owner accepts it. Each customer will have different customer id's.
- One nursery has many employees, one employee works for only one nursery. Each employee will have different employee id's. The employee will be assigned a work by the nursery owner.
- The primary keys are N\_id of Nursery table, P\_id of Plants table, C\_id of customer table and E\_id of Employee table.

## 3.4 Schema Diagram

A Schema is a pictorial representation of the relationship between the database tables in the database that is created. The database schema of a database system is its structure described in a formal language supported by the database management system (DBMS). The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database. These integrity constraints ensure compatibility between parts of the schema. All constraints are expressible in the same language. A database can be considered a structure in realization of the database language. The states of a created conceptual schema are transformed into an explicit mapping, the database schema. This describes how real-world entities are modelled in the database.

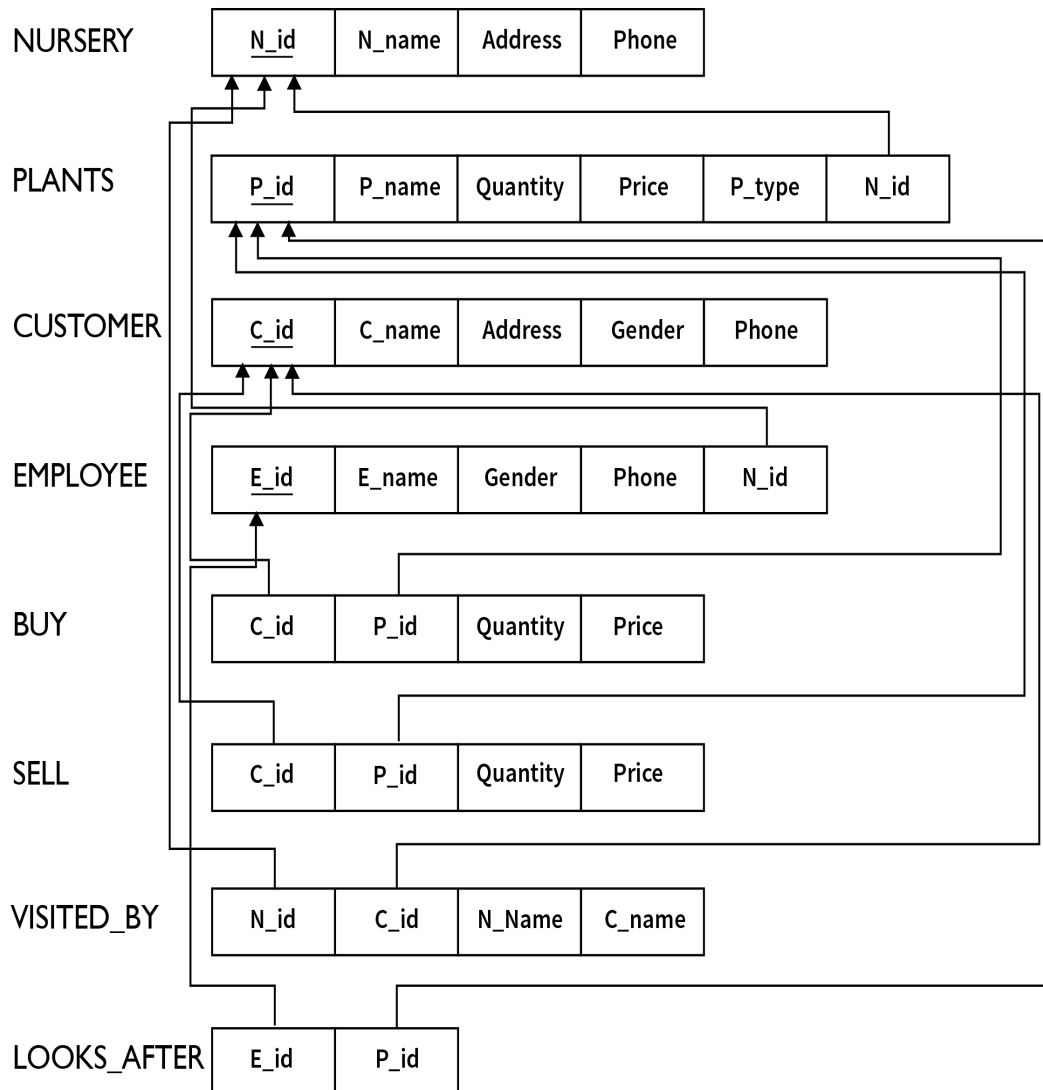
SCHEMA DIAGRAM :

Figure 3.2: Schema Diagram for Nursery Management System

# Chapter 4

## Implementation

### 4.1 Pseudocode for Nursery Management System

#### Pseudocode to Connection of Database:

In order to store or access the data inside a MySQL database, you first need to connect to the MySQL database server. In PHP you can easily do this using the `mysqli_connect()` function. All communication between PHP and the MySQL database server takes place through this connection. The hostname parameter in the above syntax specifies the host name (e.g. localhost), whereas the username and password parameters specify the credentials to access MySQL server, and the database parameter, if provided will specify the default MySQL database to be used when performing queries. The default username for MySQL database server is root and there is no password.

A screenshot of a code editor window titled 'config.php'. The code is as follows:

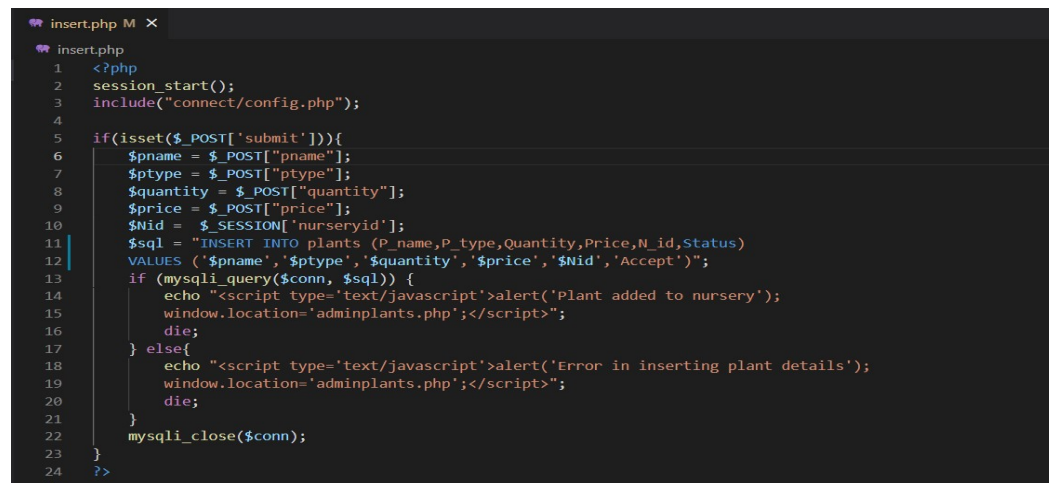
```
connect > config.php
1  <?php
2  $servername = "localhost";
3  $username = "root";
4  $password = "";
5  $database = "nursery";
6  $conn = mysqli_connect($servername, $username, $password, $database);
7
8  if($conn == false){
9      die('Error: Cannot connect');
10 }
11
12 ?>
13
```

Figure 4.1: Pseudocode for connection to database



## Pseudocode for Inserting values into the database:

Insert statement is a DML (Data modification language) statement which is used to insert data in the MySQL table. PHP \$\_POST is a PHP super global variable which is used to collect form data after submitting an HTML form with method="post". The PHP mysqli\_query() function accepts a string value representing a query as one of the parameters and, executes/performs the given query on the database.



```

insert.php M X
insert.php
1  <?php
2  session_start();
3  include("connect/config.php");
4
5  if(isset($_POST['submit'])){
6      $pname = $_POST["pname"];
7      $ptype = $_POST["ptype"];
8      $quantity = $_POST["quantity"];
9      $price = $_POST["price"];
10     $Nid = $_SESSION['nurseryid'];
11     $sql = "INSERT INTO plants (P_name,P_type,Quantity,Price,N_id,Status)
12     VALUES ('$pname','$ptype','$quantity','$price','$Nid','Accept')";
13     if (mysqli_query($conn, $sql)) {
14         echo "<script type='text/javascript'>alert('Plant added to nursery');
15         window.location='adminplants.php';</script>";
16         die;
17     } else{
18         echo "<script type='text/javascript'>alert('Error in inserting plant details');
19         window.location='adminplants.php';</script>";
20         die;
21     }
22     mysqli_close($conn);
23 }
24 ?>

```

Figure 4.2: Pseudocode for Inserting values into the database

## Pseudocode for Updation:

The UPDATE statement is used to modify the existing records in a table. The WHERE clause specifies which record(s) that should be updated.



```

update.php M X
update.php
1  <?php
2  session_start();
3  include("connect/config.php");
4  $pid = $_GET["id"];
5  $sql = "SELECT * FROM plants WHERE P_id = $pid";
6  $result = mysqli_query($conn,$sql);
7  while($row = mysqli_fetch_array($result)){
8      $pname = $row['P_name'];
9      $ptype = $row['P_type'];
10     $quantity = $row['Quantity'];
11     $price = $row['Price'];
12 }
13 $db_fields = array('P_name','P_type','Quantity','Price');
14 $sqlfield = '';
15 if ($_SERVER['REQUEST_METHOD'] == 'POST') {
16     foreach($db_fields as $fieldname) {
17         if (!empty($_POST[$fieldname])) {
18             $sqlfield .= "$fieldname = '{$_POST[$fieldname]}',";
19         }
20     }
21     $sqlfield = rtrim($sqlfield,',');
22     $sql = "UPDATE plants SET $sqlfield WHERE P_id = $pid";
23     if (mysqli_query($conn, $sql)) {
24         echo "<script type='text/javascript'>alert('Details updates successfully');
25         window.location='adminplants.php';</script>";
26         die;
27     } else{
28         echo "<script type='text/javascript'>alert('Oops! Something went wrong. Please try again later.');"
29         window.location='adminplants.php';</script>";
30         die;
31     }
32     mysqli_close($conn);
33 }
34 ?>

```

Figure 4.3: Pseudocode for Updation

## 4.2 Tables used

Nursery table :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>N_id</b>	int(11)			No	None		AUTO_INCREMENT
2	<b>N_name</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
3	<b>Address</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
4	<b>Gmail</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
5	<b>Phone</b>	varchar(10)	utf8mb4_general_ci		Yes	NULL		
6	<b>username</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
7	<b>password</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		

Figure 4.4: Structure of Nursery table

Plants table :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>P_id</b>	int(11)			No	None		AUTO_INCREMENT
2	<b>P_name</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
3	<b>P_type</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
4	<b>Quantity</b>	varchar(10)	utf8mb4_general_ci		Yes	NULL		
5	<b>Price</b>	varchar(10)	utf8mb4_general_ci		Yes	NULL		
6	<b>N_id</b>	int(11)			Yes	NULL		
7	<b>Status</b>	varchar(20)	utf8mb4_general_ci		No	None		

Figure 4.5: Structure of Plants table

## Customers table :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>C_id</b>	int(11)			No	None		AUTO_INCREMENT
2	<b>C_name</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
3	<b>Address</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
4	<b>Phone</b>	varchar(10)	utf8mb4_general_ci		Yes	NULL		
5	<b>gender</b>	varchar(10)	utf8mb4_general_ci		Yes	NULL		
6	<b>gmail</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
7	<b>password</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
8	<b>password1</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
9	<b>username</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		

Figure 4.6: Structure of Customers table

## Employees table :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>E_id</b>	int(11)			No	None		AUTO_INCREMENT
2	<b>E_name</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
3	<b>Address</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
4	<b>Phone</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
5	<b>Gender</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
6	<b>Gmail</b>	varchar(20)	utf8mb4_general_ci		Yes	NULL		
7	<b>N_id</b>	int(11)			Yes	NULL		

Figure 4.7: Structure of Employees table

# Chapter 5

## Results and Discussion

### Register Page:

This is the Sign-up page. Here the customer can register a new account.

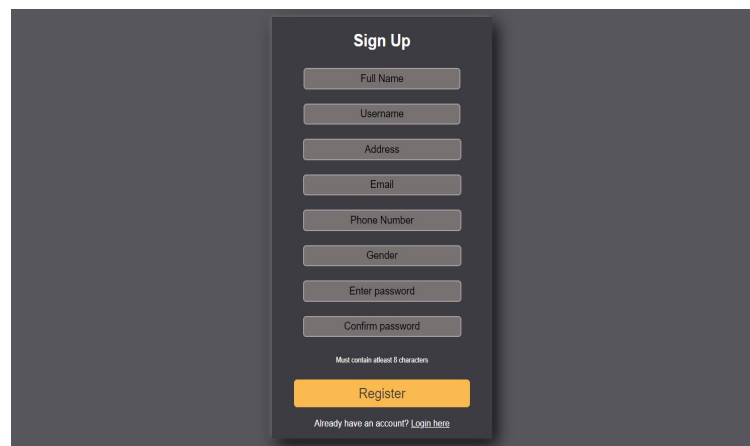
A dark-themed registration form titled "Sign Up". It contains input fields for "Full Name", "Username", "Address", "Email", "Phone Number", "Gender", "Enter password", and "Confirm password". Below the password fields is a small text requirement: "Must contain atleast 8 characters". At the bottom is a yellow "Register" button and a link that says "Already have an account? [Login here](#)".

Figure 5.1: Register Page

### Login Page:

This is the login page. If the Customer has already registered the account, to validate the user like authenticating the username and password is done in login page.

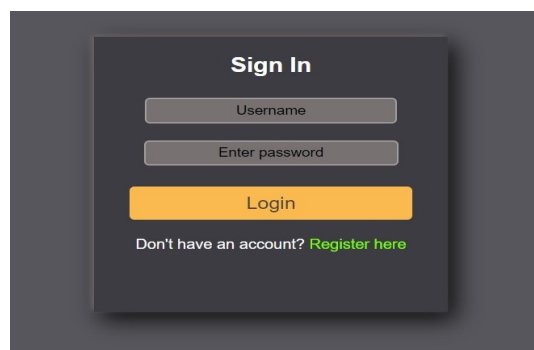
A dark-themed login form titled "Sign In". It contains input fields for "Username" and "Enter password". Below these is a yellow "Login" button. At the bottom is a link that says "Don't have an account? [Register here](#)".

Figure 5.2: Login Page

### Customer Homepage:

This page shows Customer details to the logged in Customer and it also displays the Plants that the Customer has sold to any Nursery.

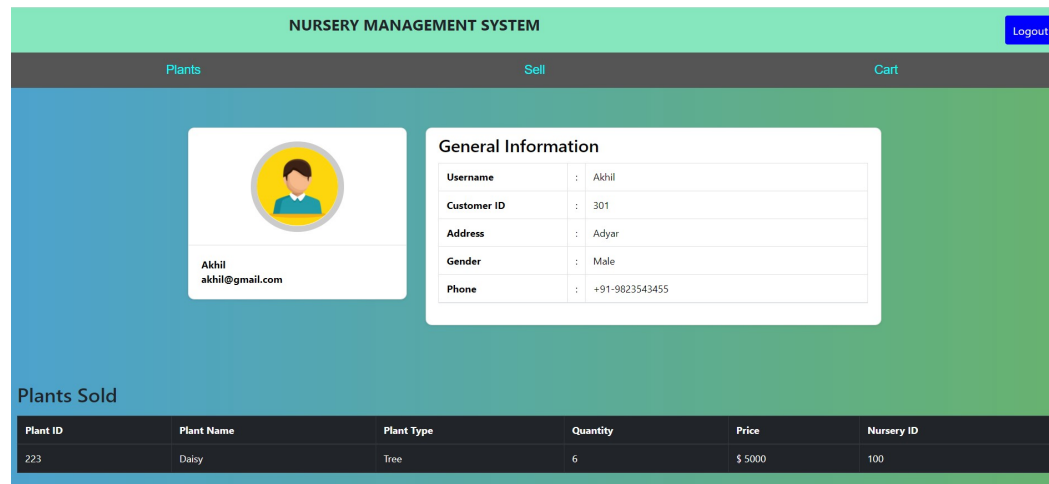


Figure 5.3: Customer homepage

### Admin Homepage:

This page shows Admin details to the logged in Admin. One Nursery has only one admin, this admin can make changes in his Nursery.

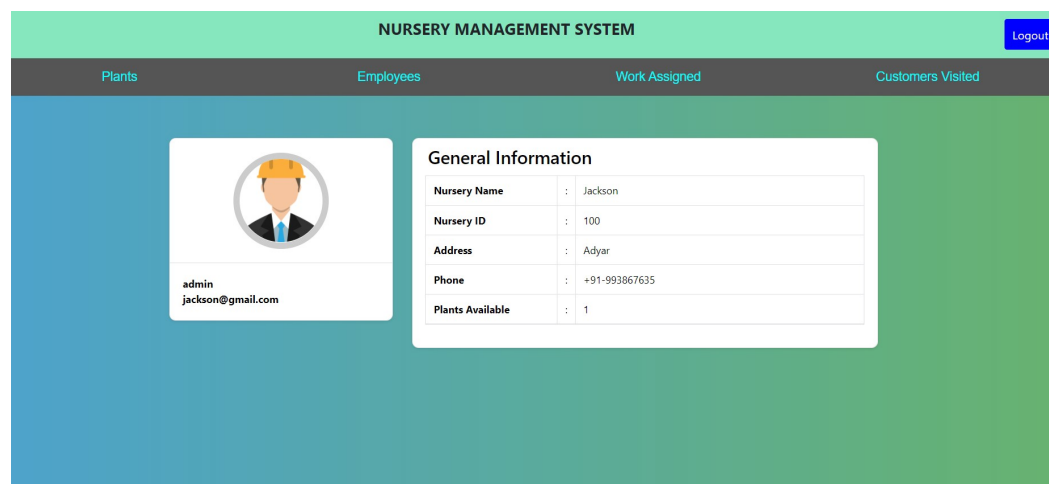


Figure 5.4: Admin homepage

## Display Plants(Customer):

The Customer can view all the Plants that are available in different Nurseries. The Customer can buy the Plants by adding the plant to the cart that he wishes to buy.



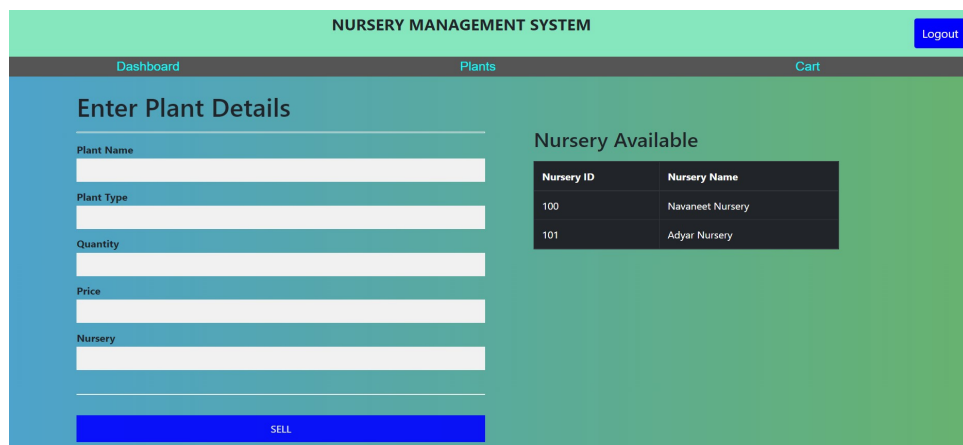
The screenshot shows the 'NURSERY MANAGEMENT SYSTEM' interface. At the top, there is a green header with the system name and a 'Logout' button. Below the header is a dark navigation bar with three options: 'Profile', 'Sell', and 'Cart'. The main content area has a light blue background and displays a table titled 'Plant Details'. The table lists various plants with their IDs, names, types, quantities, prices, and the nursery they belong to. Each row has an 'Add to cart' button.

Plant ID	Plant Name	Plant Type	Quantity	Price	Nursery	Action
223	Daisy	Tree	8	800	Navaneet Nursery	<a href="#">Add to cart</a>
230	Bluestem	Flower	9	999	Navaneet Nursery	<a href="#">Add to cart</a>
231	Caladium	Flower	2	199	Navaneet Nursery	<a href="#">Add to cart</a>
232	Cactus	Flower	7	999	Navaneet Nursery	<a href="#">Add to cart</a>
233	Lily	Flower	0	99	Adyar Nursery	<a href="#">Add to cart</a>
234	Rose	Flower	5	900	Navaneet Nursery	<a href="#">Add to cart</a>
235	Bluestem	Flower	4	299	Navaneet Nursery	<a href="#">Add to cart</a>

Figure 5.5: Customer views all the Plants

## Sell Plants(Customer):

The Customer can sell plants to the Nursery by providing the necessary details. The details of the plant will be sent to the Admin of that Nursery.



The screenshot shows the 'NURSERY MANAGEMENT SYSTEM' interface. At the top, there is a green header with the system name and a 'Logout' button. Below the header is a dark navigation bar with three options: 'Dashboard', 'Plants', and 'Cart'. The main content area has a light blue background and displays a form titled 'Enter Plant Details' and a table titled 'Nursery Available'. The form has input fields for Plant Name, Plant Type, Quantity, Price, and Nursery, followed by a 'SELL' button. The table lists available nurseries with their IDs and names.

Plant Name
<input type="text"/>

Plant Type
<input type="text"/>

Quantity
<input type="text"/>

Price
<input type="text"/>

Nursery
<input type="text"/>

[SELL](#)

Nursery ID	Nursery Name
100	Navaneet Nursery
101	Adyar Nursery

Figure 5.6: Customer providing details while selling a plant

### Display Plants(Admin):

The admin views all the Plants that are available in his Nursery only. The admin can add a new plant, delete a plant or even update the details of a plant. The plants which the Customer wants to sell will be received by the admin and the admin can either accept or reject the plants. After accepting the plant, it gets displayed in Customer's plant display.

NURSERY MANAGEMENT SYSTEM

Logout

Profile

Employees

Work Assigned

Customers Visited

Plant Details

Add New Plants

Plant ID	Plant Name	Plant Type	Quantity	Price	Action
237	Chikku	Fruit	10	50	<div>UpdateDelete</div>
238	Mango	Fruit	25	25	<div>UpdateDelete</div>
239	Betel Leaf Plant	Leaf	75	10	<div>UpdateDelete</div>

Customers Plants

Customer	Plant Name	Plant Type	Quantity	Price	Action
Jackson	Lily	Flower	10	99	<div>AcceptReject</div>

Figure 5.7: Admin views all the Plants present in his Nursery

### Display Employees(Admin):

The admin views all the Employees who are working in his Nursery. The admin can add a new Employee, delete an Employee or even update the details of an Employee.

NURSERY MANAGEMENT SYSTEM

Logout

Profile

Plants

Work Assigned

Customers Visited

Employee Details

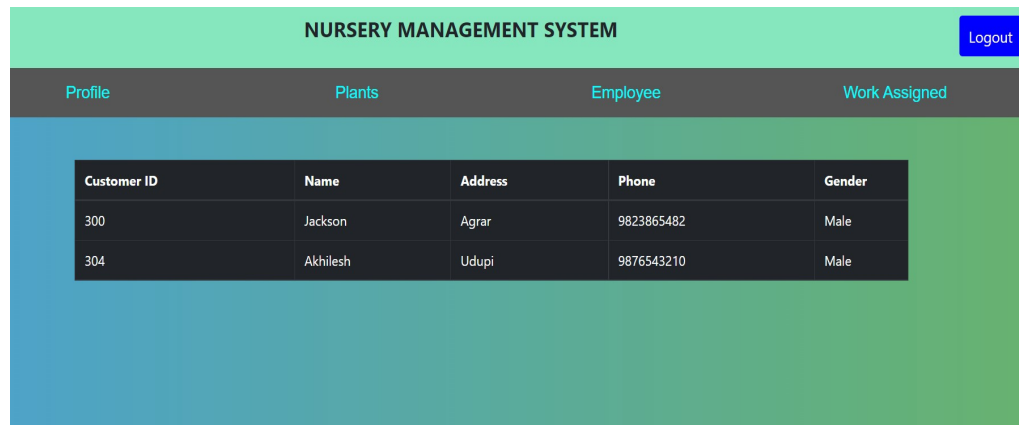
Add New Employee

Employee ID	Employee Name	Address	Phone	Gender	Action
443	Raj	Mangalore	8976543221	Male	<div>UpdateDelete</div>
444	Yasin	Bhatkal	9877865665	Male	<div>UpdateDelete</div>
445	Ramappa	Mysore	6758678874	Male	<div>UpdateDelete</div>

Figure 5.8: Admin views all the Employees working in his Nursery

## Display Customers(Admin):

The admin views all the Customers who have visited his Nursery. The admin can add a new Employee,delete an Employee or even update the details of an Customer.

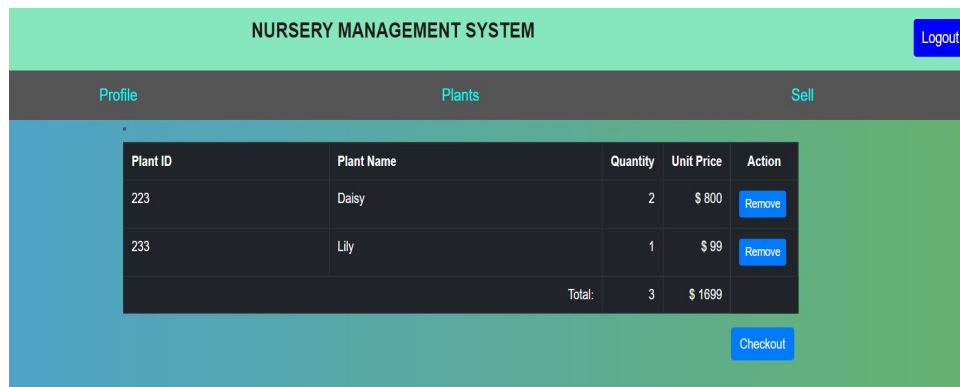


NURSERY MANAGEMENT SYSTEM					Logout
Profile	Plants	Employee	Work Assigned		
Customer ID	Name	Address	Phone	Gender	
300	Jackson	Agrar	9823865482	Male	
304	Akhilesh	Udupi	9876543210	Male	

Figure 5.9: Admin views all the Customers who have visited his Nursery

## Customer Cart:

The Customer can pile up the Plants they want to buy to the cart and then checkout which produces Invoice or a bill.The Customer can add as well as remove the plants from their cart.



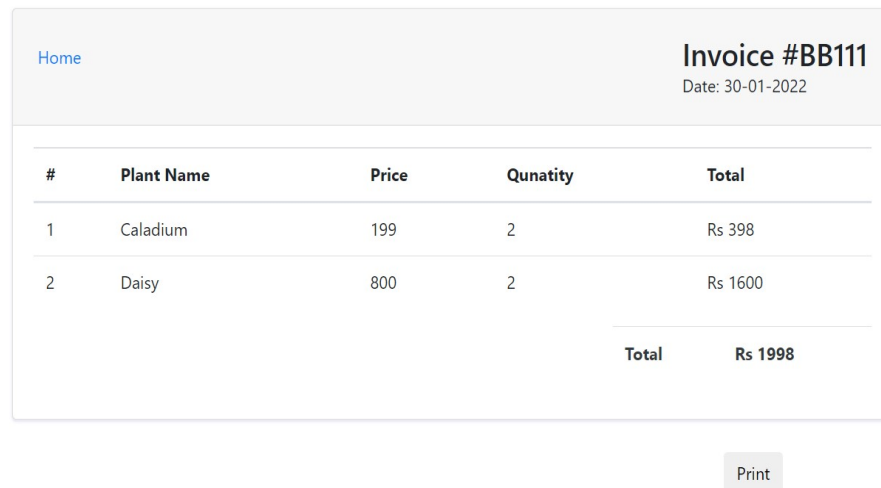
NURSERY MANAGEMENT SYSTEM					Logout
Profile	Plants	Sell			
Plant ID	Plant Name	Quantity	Unit Price	Action	
223	Daisy	2	\$ 800	Remove	
233	Lily	1	\$ 99	Remove	
Total:		3	\$ 1699		
					Checkout

Figure 5.10: Customer's cart containing plants to buy



### Customer Checkout:

The checkout process is the series of steps a customer follows to purchase the items in their cart. After the Customer checks out, the Invoice or a Bill is generated and simultaneously the total available plants quantity is updated in the Plants table.



The screenshot shows a web interface for an invoice. At the top left is a 'Home' link. At the top right is the invoice title 'Invoice #BB111' and the date 'Date: 30-01-2022'. Below this is a table with the following data:

#	Plant Name	Price	Qunatity	Total
1	Caladium	199	2	Rs 398
2	Daisy	800	2	Rs 1600
Total				Rs 1998

Below the table is a 'Print' button.

Figure 5.11: Invoice after the Customer checks out

### Out of stock :

Out of stock occurs every time a plant is not available when a customer would actually be ready to buy it. When the quantity of the plants available is zero, if the customer tries to add that plant to the cart, an out of stock trigger is displayed.

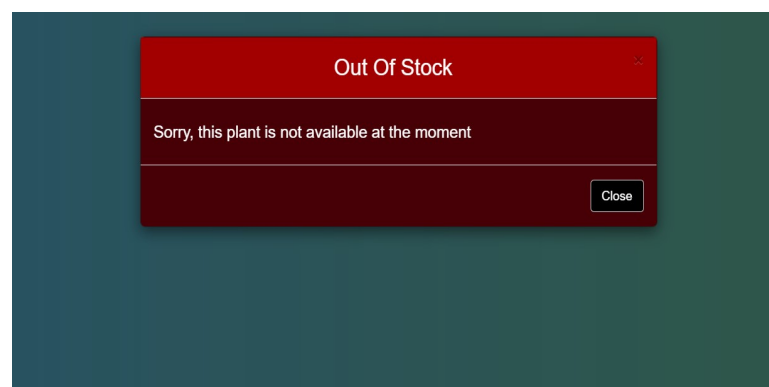


Figure 5.12: Out of Stock

# Chapter 6

## Conclusion and Future work

This website provides a computerized version of nursery management system which will benefit the nursery owners as well as the customers who wish to visit and communicate with the nursery. It has a facility for customer login where customer can login and can see the plants available to buy or sell. It also has admin login where nursery owner can add new plants, employees and update their details or delete records which aren't necessary. Once the customer buys the plants, the available quantity of plants get updated automatically and also price is calculated with bill or invoice generation as well.

This project can be further improved by developing a better experience for the customers to sell their plants to the nursery and also allowing customers to pay through Google Pay or UPI based on their convenience. Further improvements can be made by validating phone numbers provided by customers through OTP. Some features like displaying plants along with their images, allowing nursery owners to add videos of their nursery or grafting a plant, storing the fertilizers used for different plants can be developed in future.

# References

- [1] Database systems Models, Languages, Design and Application Programming, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, Pearson.
- [2] Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill.
- [3] Silberschatz Korth and Sudharshan: Database System Concepts, 6th Edition, McGraw Hill, 2013.
- [4] Coronel, Morris, and Rob, Database Principles Fundamentals of Design, Implementation and Management, Cengage Learning 2012.