**Film Production Management System**

**Insert, Update, Delete functionalities in the Film Production Management System**

The Film Production Management System is a comprehensive web-based application designed to manage critical aspects of film production, including financial planning, marketing strategies, and personnel management. The system streamlines operations by providing CRUD (Create, Read, Update, Delete) functionalities for managing data related to finance, marketing, and directors. It enhances productivity, ensures data accuracy, and supports decision-making processes in the film production pipeline.

### **CRUD Operations:**

**1. Create:**This operation is used to add new data into the database. In your system, it involves entering details for a finance team, marketing team, or director (such as phone numbers, budgets, or timeframes) through a form. Once submitted, the data is saved in the respective database table, creating a new record.

**2. Read:**The "Read" operation retrieves and displays existing data from the database. For example, it allows users to view all finance team records, including their phone numbers and budgets, or marketing team details. This is typically shown in a table format on the web interface.

**3. Update:**This operation modifies existing records in the database. Users can edit details like a finance team’s phone number or a director's information by filling out a form. When submitted, the updated data replaces the old record in the database.

**4. Delete:**The "Delete" operation removes unwanted records from the database. For instance, users can select and delete a specific finance team or marketing team record, permanently removing it from the system. This is useful for maintaining clean and accurate data.

**Index.php:**

<?php

// Database Configuration

$host = 'localhost';

$username = 'root';

$password = '';

$dbname = 'production\_db';

$port=3307;

// Create Connection

$conn = new mysqli($host, $username, $password, $dbname,$port);

// Check Connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

// Handle Create - Finance Team

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['create\_finance'])) {

$fi\_phno = $\_POST['fi\_phno'];

$assigned\_budget = $\_POST['assigned\_budget'];

$stmt = $conn->prepare("INSERT INTO finance\_team (fi\_phno, assigned\_budget) VALUES (?, ?)");

$stmt->bind\_param("sd", $fi\_phno, $assigned\_budget);

if ($stmt->execute()) {

header("Location: " . $\_SERVER['PHP\_SELF']);

exit();

} else {

echo "Error: " . $stmt->error;

}

$stmt->close();

}

// Handle Update - Finance Team

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['update\_finance'])) {

$fi\_id = $\_POST['fi\_id'];

$fi\_phno = $\_POST['fi\_phno'];

$assigned\_budget = $\_POST['assigned\_budget'];

$stmt = $conn->prepare("UPDATE finance\_team SET fi\_phno = ?, assigned\_budget = ? WHERE fi\_id = ?");

$stmt->bind\_param("sdi", $fi\_phno, $assigned\_budget, $fi\_id);

if ($stmt->execute()) {

header("Location: " . $\_SERVER['PHP\_SELF']);

exit();

} else {

echo "Error updating record: " . $stmt->error;

}

$stmt->close();

}

// Handle Delete - Finance Team

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['delete\_finance'])) {

$fi\_id = $\_POST['fi\_id'];

$stmt = $conn->prepare("DELETE FROM finance\_team WHERE fi\_id = ?");

$stmt->bind\_param("i", $fi\_id);

if ($stmt->execute()) {

header("Location: " . $\_SERVER['PHP\_SELF']);

exit();

} else {

echo "Error deleting record: " . $stmt->error;

}

$stmt->close();

}

// Handle Create - Director

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['create\_director'])) {

$d\_name = $\_POST['d\_name'];

$d\_age = $\_POST['d\_age'];

$d\_phno = $\_POST['d\_phno'];

$d\_email = $\_POST['d\_email'];

$stmt = $conn->prepare("INSERT INTO director (d\_name, d\_age, d\_phno, d\_email) VALUES (?, ?, ?, ?)");

$stmt->bind\_param("siss", $d\_name, $d\_age, $d\_phno, $d\_email);

if ($stmt->execute()) {

header("Location: " . $\_SERVER['PHP\_SELF']);

exit();

} else {

echo "Error: " . $stmt->error;

}

$stmt->close();

}

// Handle Update - Director

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['update\_director'])) {

$d\_id = $\_POST['d\_id'];

$d\_name = $\_POST['d\_name'];

$d\_age = $\_POST['d\_age'];

$d\_phno = $\_POST['d\_phno'];

$d\_email = $\_POST['d\_email'];

$stmt = $conn->prepare("UPDATE director SET d\_name = ?, d\_age = ?, d\_phno = ?, d\_email = ? WHERE d\_id = ?");

$stmt->bind\_param("sissi", $d\_name, $d\_age, $d\_phno, $d\_email, $d\_id);

if ($stmt->execute()) {

header("Location: " . $\_SERVER['PHP\_SELF']);

exit();

} else {

echo "Error updating record: " . $stmt->error;

}

$stmt->close();

}

// Handle Delete - Director

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['delete\_director'])) {

$d\_id = $\_POST['d\_id'];

$stmt = $conn->prepare("DELETE FROM director WHERE d\_id = ?");

$stmt->bind\_param("i", $d\_id);

if ($stmt->execute()) {

header("Location: " . $\_SERVER['PHP\_SELF']);

exit();

} else {

echo "Error deleting record: " . $stmt->error;

}

$stmt->close();

}

// Fetch Finance Team Data

$finance\_query = "SELECT \* FROM finance\_team";

$finance\_results = $conn->query($finance\_query);

// Fetch Director Data

$director\_query = "SELECT \* FROM director";

$director\_results = $conn->query($director\_query);

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Film Production System Dashboard</title>

<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;600&display=swap" rel="stylesheet">

<style>

/\* Modern, Clean Reset and Base Styles \*/

:root {

--primary-color: #2c3e50;

--secondary-color: #34495e;

--accent-color: #3498db;

--background-light: #ecf0f1;

--text-color: #333;

--white: #ffffff;

}

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: 'Poppins', sans-serif;

background-color: var(--background-light);

color: var(--text-color);

line-height: 1.6;

}

/\* Responsive Container \*/

.container {

width: 95%;

max-width: 1200px;

margin: 0 auto;

padding: 20px;

}

/\* Elegant Header \*/

header {

background: linear-gradient(to right, var(--primary-color), var(--secondary-color));

color: var(--white);

text-align: center;

padding: 20px 0;

box-shadow: 0 4px 6px rgba(0,0,0,0.1);

}

header h1 {

font-weight: 600;

letter-spacing: 1px;

}

/\* Section Styling \*/

section {

background: var(--white);

border-radius: 8px;

box-shadow: 0 4px 6px rgba(0,0,0,0.1);

margin-bottom: 30px;

padding: 25px;

}

section h3 {

color: var(--primary-color);

border-bottom: 2px solid var(--accent-color);

padding-bottom: 10px;

margin-bottom: 20px;

font-weight: 600;

}

/\* Form Styling \*/

form {

display: grid;

grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));

gap: 15px;

background: var(--background-light);

padding: 20px;

border-radius: 8px;

margin-bottom: 20px;

}

form input {

width: 100%;

padding: 10px;

border: 1px solid #ddd;

border-radius: 4px;

font-family: 'Poppins', sans-serif;

transition: all 0.3s ease;

}

form input:focus {

outline: none;

border-color: var(--accent-color);

box-shadow: 0 0 0 2px rgba(52, 152, 219, 0.2);

}

form input[type="submit"] {

background-color: var(--accent-color);

color: var(--white);

border: none;

cursor: pointer;

font-weight: 600;

transition: background 0.3s ease;

}

form input[type="submit"]:hover {

background-color: var(--primary-color);

}

/\* Table Styling \*/

table {

width: 100%;

border-collapse: separate;

border-spacing: 0;

background: var(--white);

box-shadow: 0 4px 6px rgba(0,0,0,0.1);

border-radius: 8px;

overflow: hidden;

}

table th {

background: var(--primary-color);

color: var(--white);

font-weight: 600;

padding: 15px;

text-align: left;

}

table td {

padding: 15px;

border-bottom: 1px solid #e0e0e0;

transition: background-color 0.3s ease;

}

table tr:hover {

background-color: rgba(52, 152, 219, 0.05);

}

/\* Action Buttons \*/

.btn-update {

background-color: var(--accent-color);

color: var(--white);

}

.btn-delete {

background-color: #e74c3c;

color: var(--white);

}

.btn-update:hover,

.btn-delete:hover {

opacity: 0.9;

}

/\* Responsive Adjustments \*/

@media screen and (max-width: 768px) {

form {

grid-template-columns: 1fr;

}

table {

font-size: 14px;

}

table td, table th {

padding: 10px;

}

}

</style>

</head>

<body>

<header>

<h1>Film Production System Dashboard</h1>

</header>

<div class="container">

<!-- Finance Team Section -->

<section>

<h3>Finance Team Records</h3>

<form method="POST">

<input type="text" name="fi\_phno" placeholder="Phone Numbers" required>

<input type="number" name="assigned\_budget" step="0.01" placeholder="Assigned Budget" required>

<input type="submit" name="create\_finance" value="Add Finance Team Member">

</form>

<table>

<tr>

<th>ID</th>

<th>Phone Numbers</th>

<th>Assigned Budget</th>

<th>Actions</th>

</tr>

<?php while ($row = $finance\_results->fetch\_assoc()) { ?>

<tr>

<td><?php echo $row['fi\_id']; ?></td>

<td><?php echo $row['fi\_phno']; ?></td>

<td><?php echo $row['assigned\_budget']; ?></td>

<td>

<form method="POST" style="display:grid; grid-template-columns: 1fr;">

<input type="hidden" name="fi\_id" value="<?php echo $row['fi\_id']; ?>">

<input type="text" name="fi\_phno" value="<?php echo $row['fi\_phno']; ?>" required>

<input type="number" name="assigned\_budget" step="0.01" value="<?php echo $row['assigned\_budget']; ?>" required>

<input type="submit" name="update\_finance" value="Update" class="btn-update">

<input type="submit" name="delete\_finance" value="Delete" class="btn-delete">

</form>

</td>

</tr>

<?php } ?>

</table>

</section>

<!-- Director Section -->

<section>

<h3>Director Records</h3>

<form method="POST">

<input type="text" name="d\_name" placeholder="Name" required>

<input type="number" name="d\_age" placeholder="Age" required>

<input type="text" name="d\_phno" placeholder="Phone Numbers" required>

<input type="email" name="d\_email" placeholder="Email" required>

<input type="submit" name="create\_director" value="Add Director">

</form>

<table>

<tr>

<th>ID</th>

<th>Name</th>

<th>Age</th>

<th>Phone Numbers</th>

<th>Emails</th>

<th>Actions</th>

</tr>

<?php while ($row = $director\_results->fetch\_assoc()) { ?>

<tr>

<td><?php echo $row['d\_id']; ?></td>

<td><?php echo $row['d\_name']; ?></td>

<td><?php echo $row['d\_age']; ?></td>

<td><?php echo $row['d\_phno']; ?></td>

<td><?php echo $row['d\_email']; ?></td>

<td>

<form method="POST" style="display:grid; grid-template-columns: 1fr;">

<input type="hidden" name="d\_id" value="<?php echo $row['d\_id']; ?>">

<input type="text" name="d\_name" value="<?php echo $row['d\_name']; ?>" required>

<input type="number" name="d\_age" value="<?php echo $row['d\_age']; ?>" required>

<input type="text" name="d\_phno" value="<?php echo $row['d\_phno']; ?>" required>

<input type="email" name="d\_email" value="<?php echo $row['d\_email']; ?>" required>

<input type="submit" name="update\_director" value="Update" class="btn-update">

<input type="submit" name="delete\_director" value="Delete" class="btn-delete">

</form>

</td>

</tr>

<?php } ?>

</table>

</section>

</div>

</body>

</html>

**finace\_crud.php**

<?php

include 'db.php';

// Handle Create

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['create\_finance'])) {

$fi\_phno = $\_POST['fi\_phno'];

$assigned\_budget = $\_POST['assigned\_budget'];

$stmt = $conn->prepare("INSERT INTO finance\_team (fi\_phno, assigned\_budget) VALUES (?, ?)");

$stmt->bind\_param("ss", $fi\_phno, $assigned\_budget);

$stmt->execute();

echo "Finance Team created.";

}

// Read

$results = $conn->query("SELECT \* FROM finance\_team");

?>

<h1>Finance Team - CRUD</h1>

<!-- Form for Create -->

<form method="POST">

<h3>Create Finance Team Entry</h3>

Phone Numbers (comma-separated): <input type="text" name="fi\_phno" required><br>

Assigned Budget: <input type="number" step="0.01" name="assigned\_budget" required><br>

<input type="submit" name="create\_finance" value="Create Finance Team">

</form>

<h3>Finance Team Records</h3>

<table border="1">

<tr>

<th>ID</th>

<th>Phone Numbers</th>

<th>Assigned Budget</th>

</tr>

<?php while ($row = $results->fetch\_assoc()) { ?>

<tr>

<td><?php echo $row['fi\_id']; ?></td>

<td><?php echo $row['fi\_phno']; ?></td>

<td><?php echo $row['assigned\_budget']; ?></td>

</tr>

<?php } ?>

</table>

**marketing\_crud.php**

<?php

include 'db.php';

// Handle Create

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['create\_marketing'])) {

$m\_phno = $\_POST['m\_phno'];

$m\_budget = $\_POST['m\_budget'];

$m\_timeframe\_start = $\_POST['m\_timeframe\_start'];

$m\_timeframe\_end = $\_POST['m\_timeframe\_end'];

$stmt = $conn->prepare("INSERT INTO marketing\_team (m\_phno, m\_budget, m\_timeframe\_start, m\_timeframe\_end) VALUES (?, ?, ?, ?)");

$stmt->bind\_param("ssss", $m\_phno, $m\_budget, $m\_timeframe\_start, $m\_timeframe\_end);

$stmt->execute();

echo "Marketing Team created.";

}

// Handle Read

$results = $conn->query("SELECT \* FROM marketing\_team");

?>

<h1>Marketing Team - CRUD</h1>

<!-- Form for Create -->

<form method="POST">

<h3>Create Marketing Team Entry</h3>

Phone Numbers (comma-separated): <input type="text" name="m\_phno" required><br>

Budget: <input type="number" step="0.01" name="m\_budget" required><br>

Timeframe Start: <input type="date" name="m\_timeframe\_start" required><br>

Timeframe End: <input type="date" name="m\_timeframe\_end" required><br>

<input type="submit" name="create\_marketing" value="Create Marketing Team">

</form>

<h3>Current Marketing Teams</h3>

<table border="1">

<tr>

<th>ID</th>

<th>Phone Numbers</th>

<th>Budget</th>

<th>Timeframe Start</th>

<th>Timeframe End</th>

</tr>

<?php while ($row = $results->fetch\_assoc()) { ?>

<tr>

<td><?php echo $row['m\_id']; ?></td>

<td><?php echo $row['m\_phno']; ?></td>

<td><?php echo $row['m\_budget']; ?></td>

<td><?php echo $row['m\_timeframe\_start']; ?></td>

<td><?php echo $row['m\_timeframe\_end']; ?></td>

</tr>

<?php } ?>

</table>

**Production\_db.php**

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

START TRANSACTION;

SET time\_zone = "+00:00";

-- Database: `production\_db`

-- Table structure for table `director`

CREATE TABLE `director` (

`d\_id` int(11) NOT NULL,

`d\_name` varchar(255) DEFAULT NULL,

`d\_age` int(11) DEFAULT NULL,

`d\_phno` text DEFAULT NULL,

`d\_email` text DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

-- Table structure for table `finance\_team`

CREATE TABLE `finance\_team` (

`fi\_id` int(11) NOT NULL,

`fi\_phno` text DEFAULT NULL,

`assigned\_budget` decimal(10,2) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

-- Table structure for table `marketing\_team`

CREATE TABLE `marketing\_team` (

`m\_id` int(11) NOT NULL,

`m\_phno` text DEFAULT NULL,

`m\_budget` decimal(10,2) DEFAULT NULL,

`m\_timeframe\_start` date DEFAULT NULL,

`m\_timeframe\_end` date DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

-- Indexes for dumped tables

-- Indexes for table `director`

ALTER TABLE `director`

ADD PRIMARY KEY (`d\_id`);

-- Indexes for table `finance\_team`

ALTER TABLE `finance\_team`

ADD PRIMARY KEY (`fi\_id`);

-- Indexes for table `marketing\_team`

ALTER TABLE `marketing\_team`

ADD PRIMARY KEY (`m\_id`);

-- AUTO\_INCREMENT for dumped tables

-- AUTO\_INCREMENT for table `director`

ALTER TABLE `director`

MODIFY `d\_id` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=6;

-- AUTO\_INCREMENT for table `finance\_team`

ALTER TABLE `finance\_team`

MODIFY `fi\_id` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=6;

-- AUTO\_INCREMENT for table `marketing\_team`

ALTER TABLE `marketing\_team`

MODIFY `m\_id` int(11) NOT NULL AUTO\_INCREMENT;

COMMIT;

**db\_connnection.php**

<?php

$servername = "localhost";

$username = "root";

$password = "root";

$dbname = "production\_db";

// Create connection with exception handling

try {

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect\_error) {

throw new Exception("Database connection failed: " . $conn->connect\_error);

}

} catch (Exception $e) {

echo "Error: " . $e->getMessage();

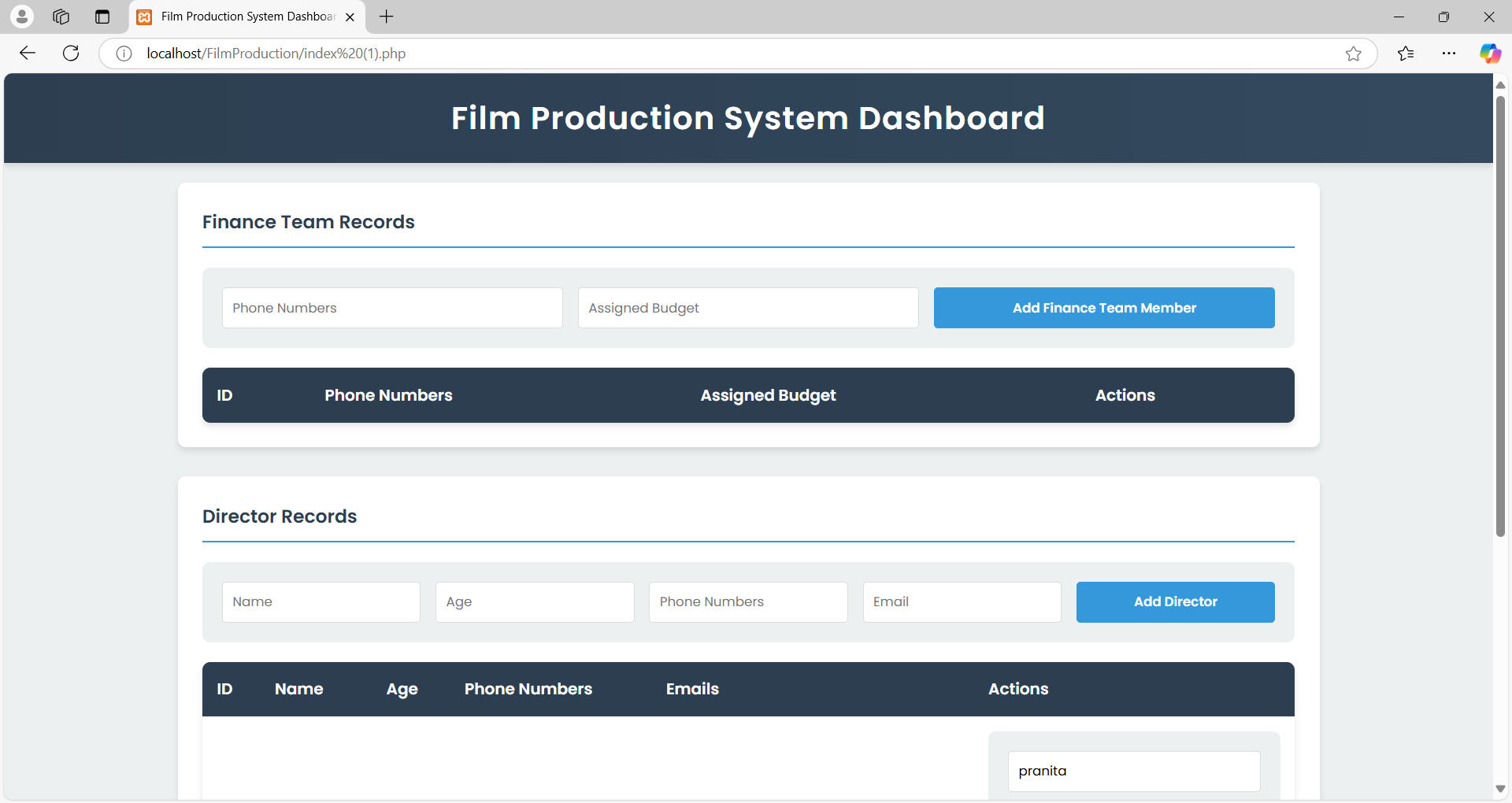
exit();

}

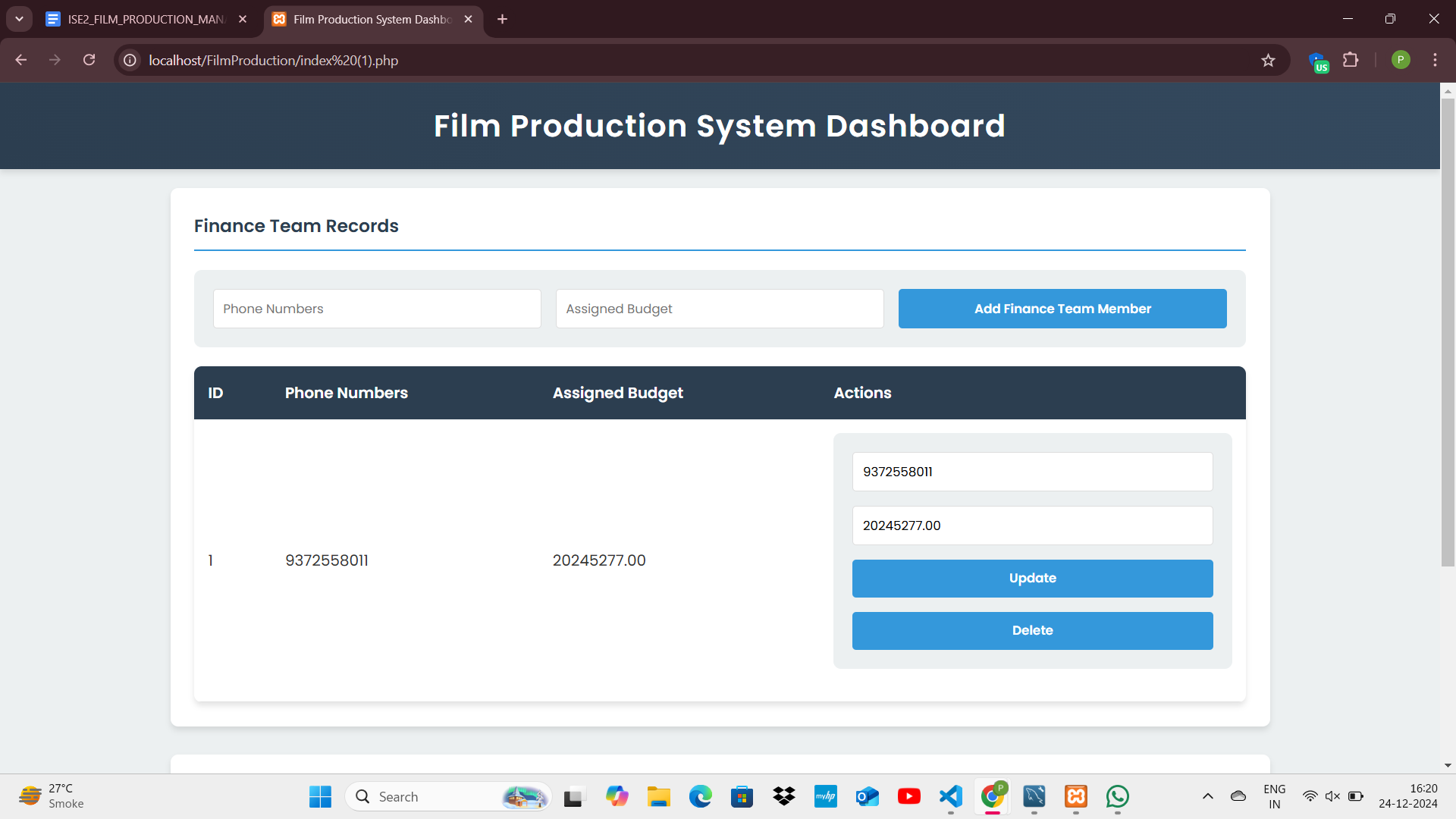
?>

**Output:**

**Dashboard of film Production Management:**

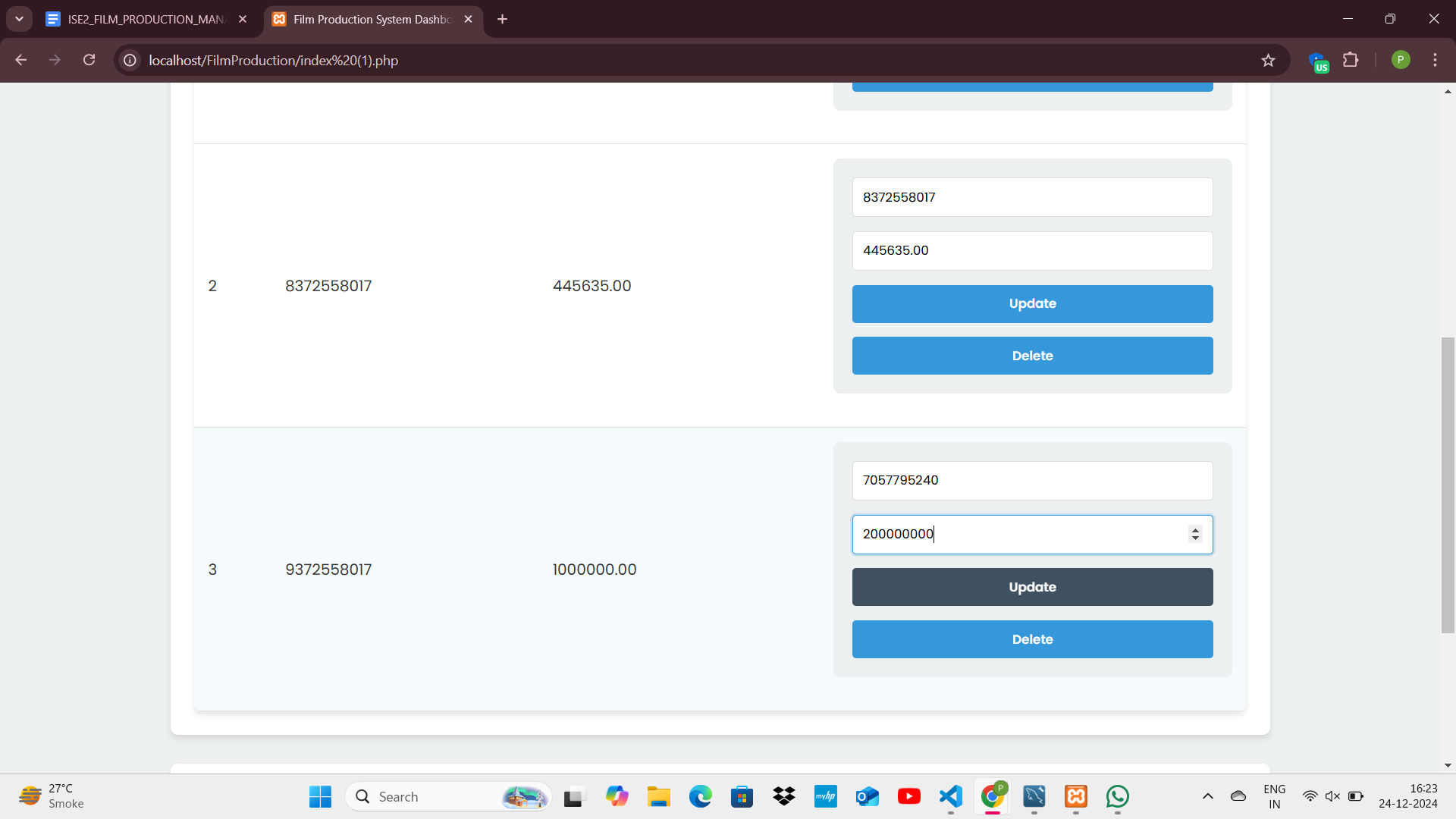
****

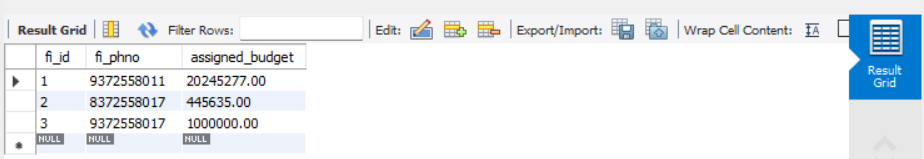
**Add finance Team Record:**

****

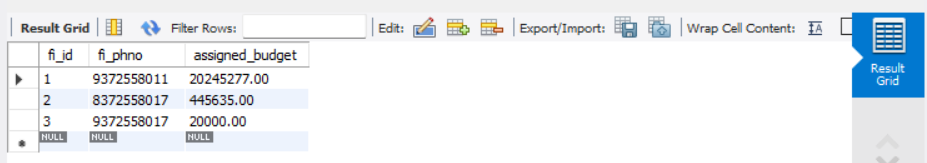
****

**Update Finance Record:**

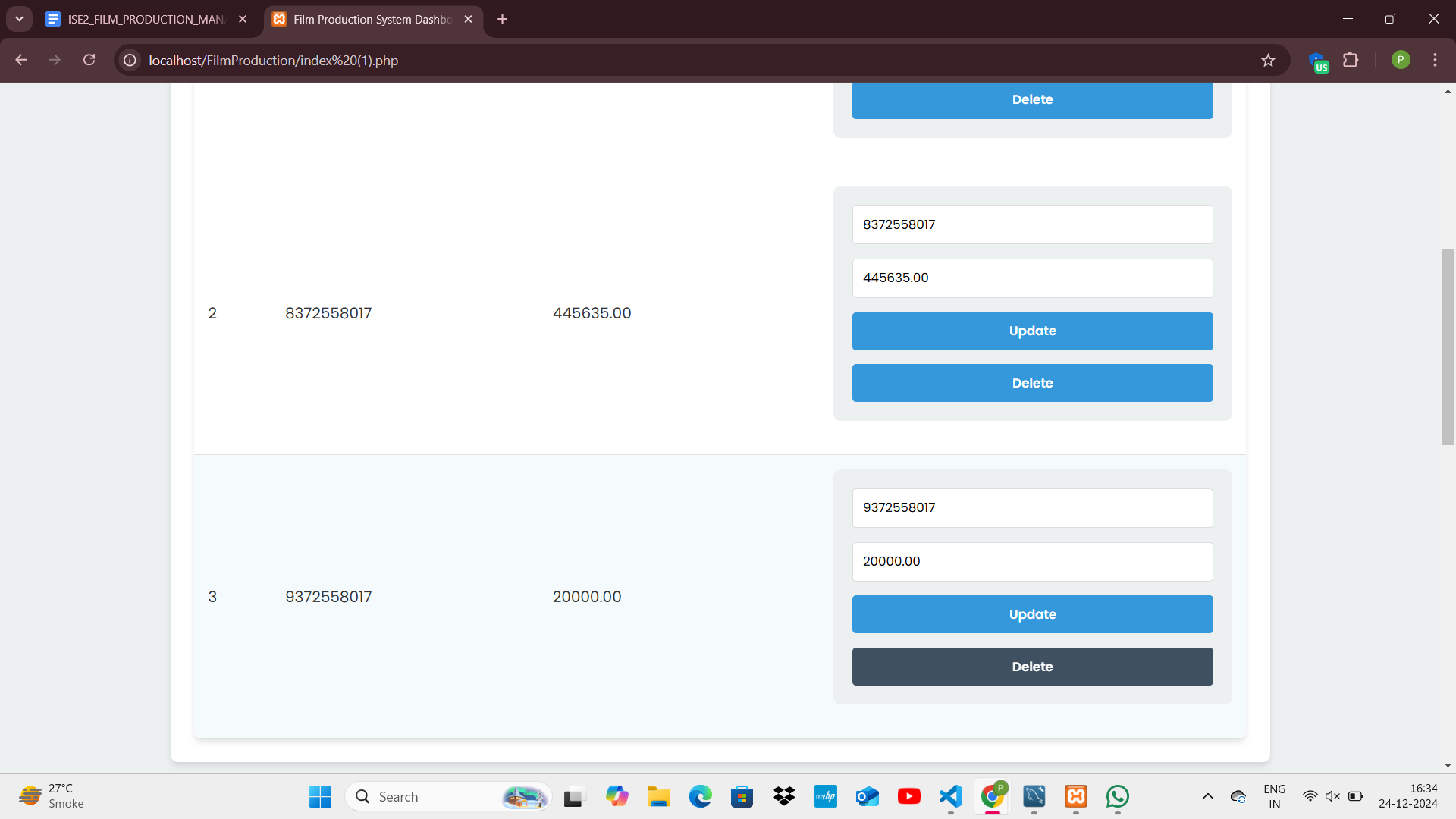
**Finance record before**

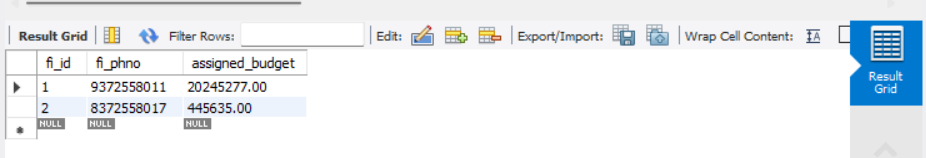
****

**After Updating Finance Record**

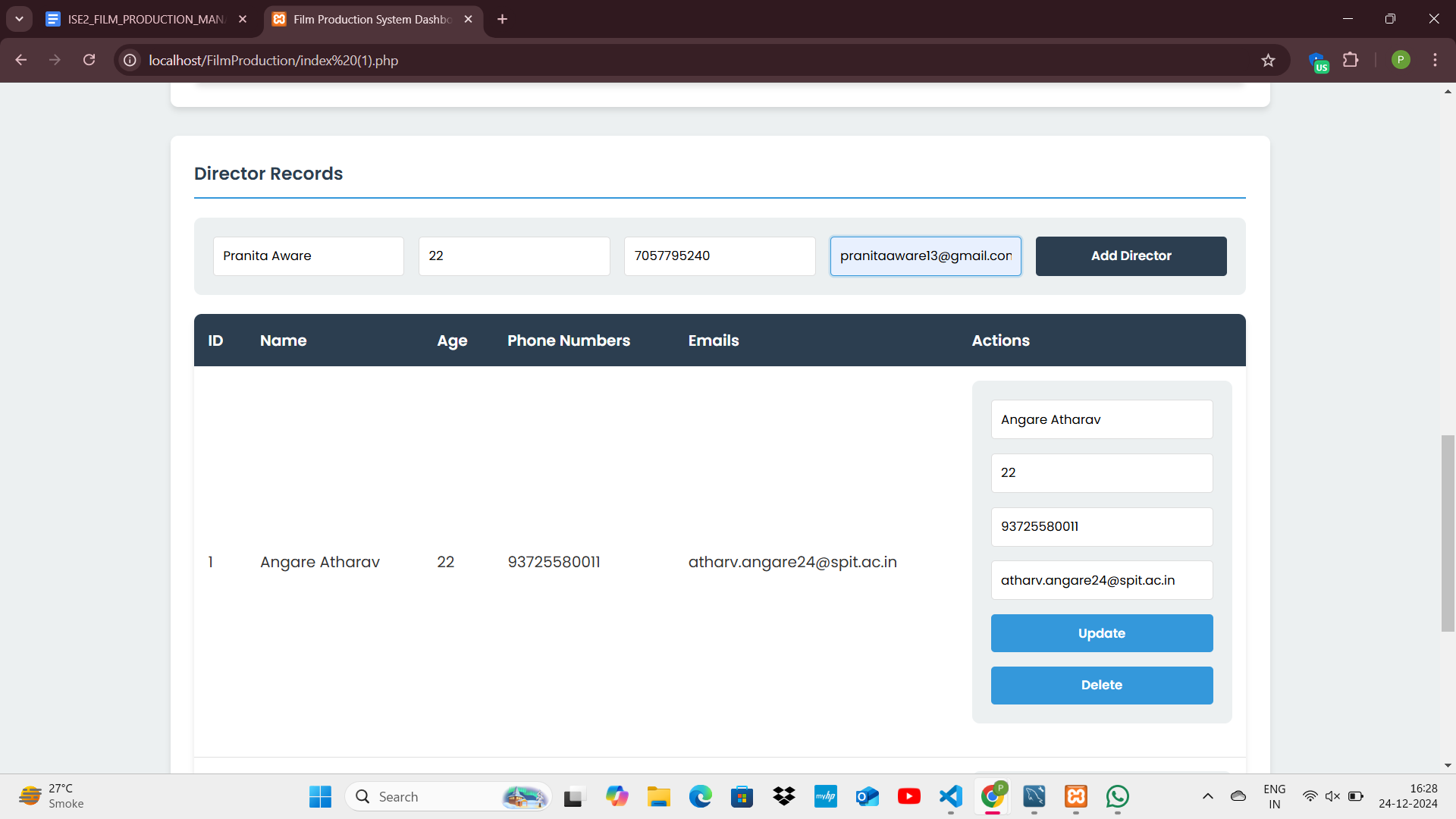
****

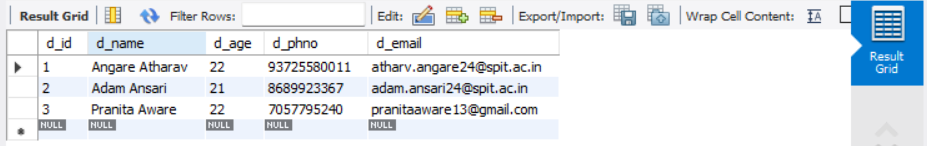
**Delete 3rd record:**

****

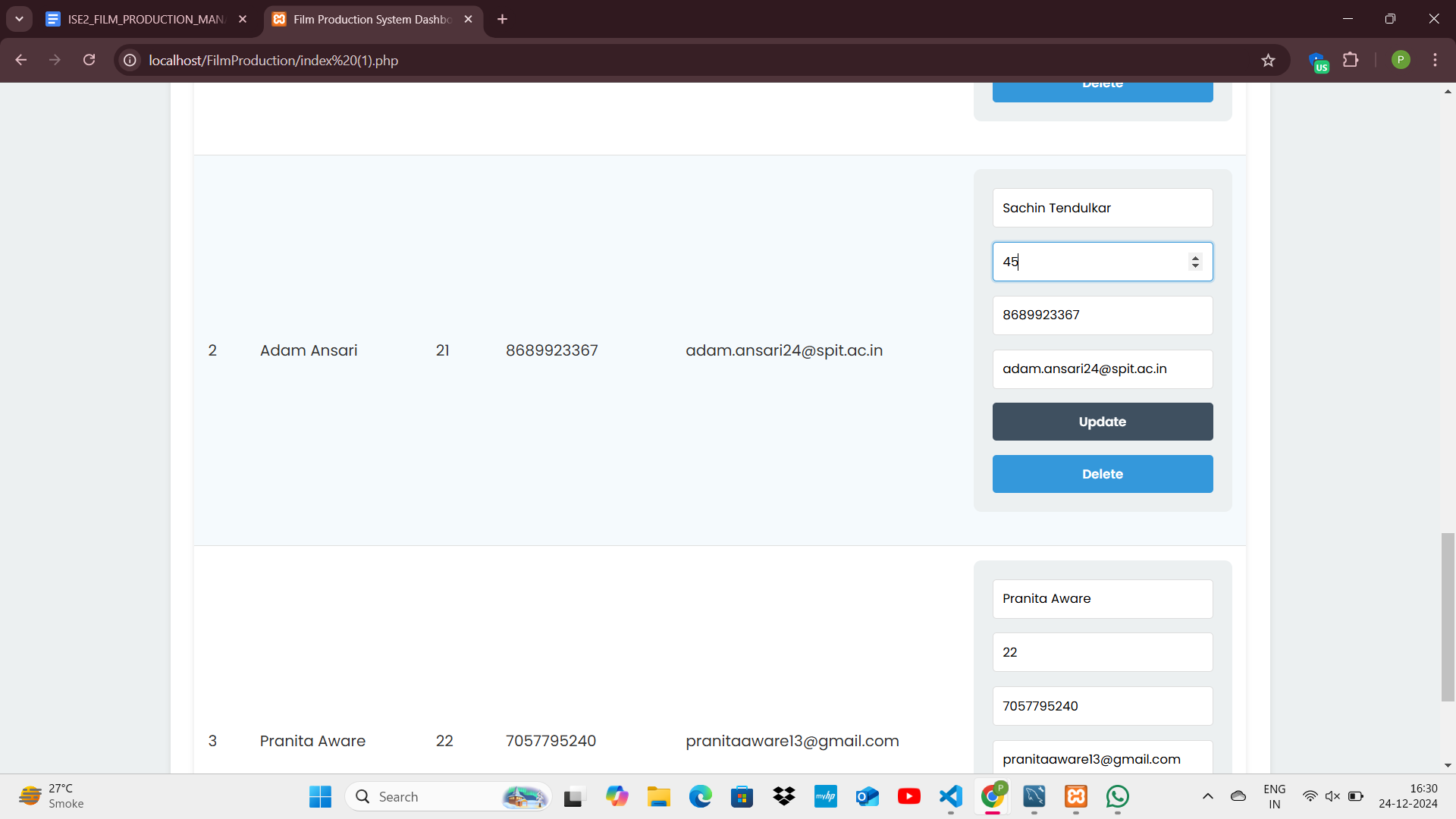
****

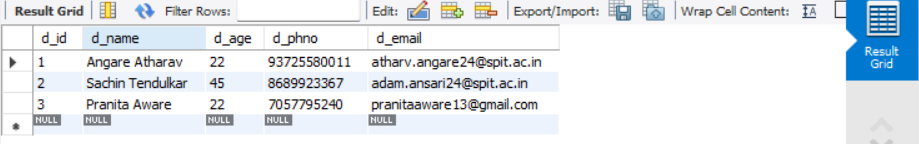
**Add director records:**

****

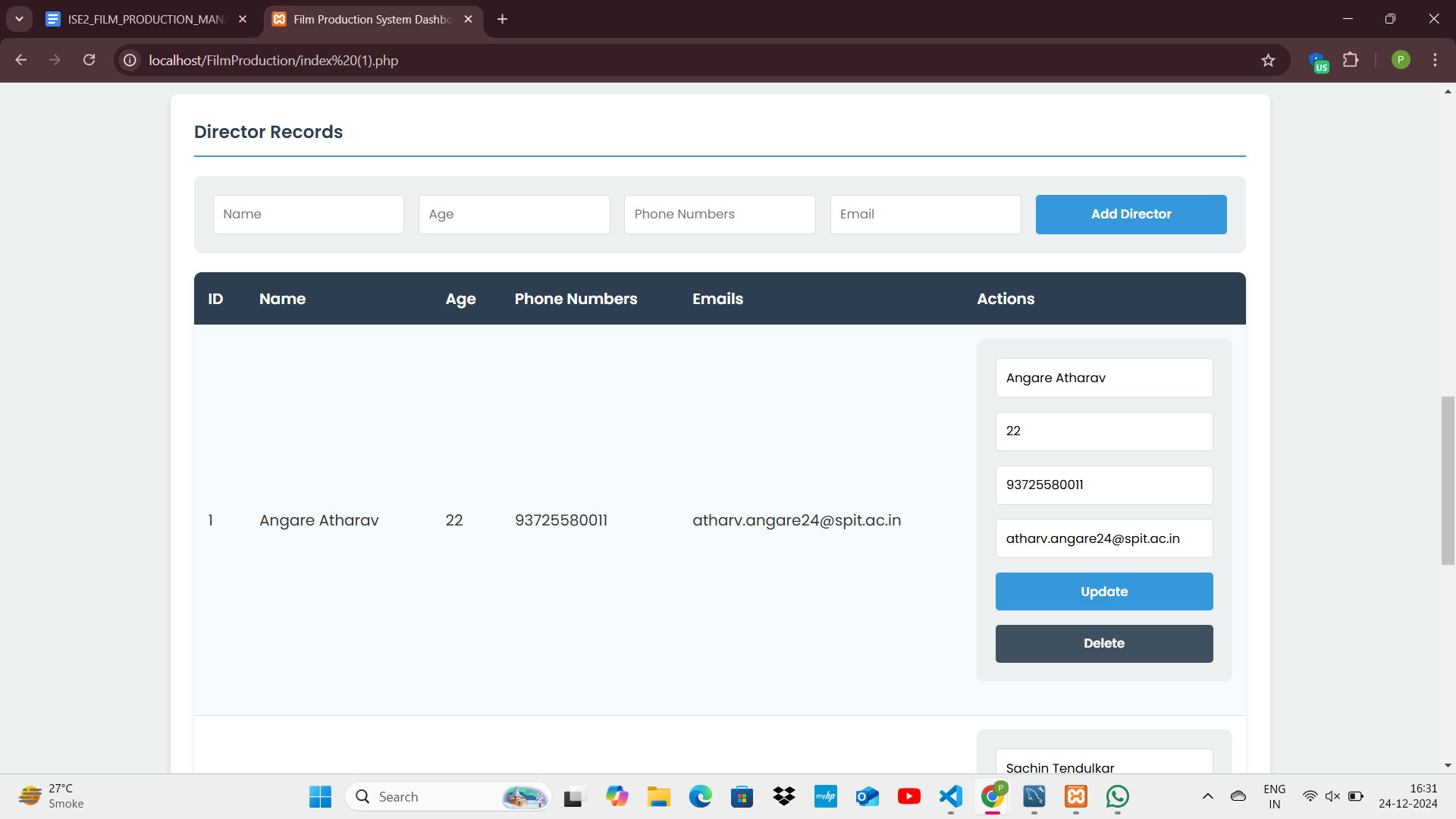
****

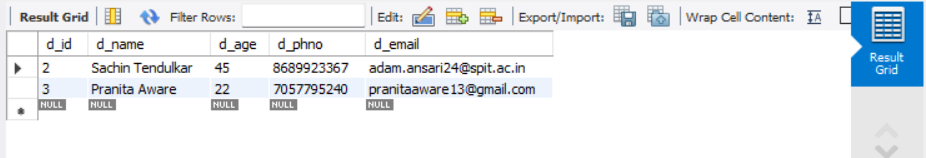
**Update 2nd Record, Change Name and Age:**

****

****

**Delete 1 st record:**

****

****

**Observation:**

While working with the Film Production Management System, I found the CRUD operations fairly straightforward, but I faced challenges in managing the database connections efficiently. The most difficult part was understanding how to properly handle the form inputs and ensure they were correctly passed to the SQL queries, especially when dealing with multiple tables like finance, marketing, and directors. I had to learn how to use prepared statements to prevent SQL injection and ensure smooth operations. Debugging the forms and aligning them with the database structure was also a bit tricky, but once I understood the relationships between the tables, I was able to streamline the process.