Cue Time Systems - Project Overview

This document provides a detailed explanation of the functionality, structure, and purpose of the Cue Time Systems web application. Designed for managing customer relationships, tracking pool table rentals, generating reports, and sending promotional emails, this system leverages PHP, XAMPP, and an Azure-hosted SQL Server database.

# Overview of Application Features

- Customer Database Management

- Rental Reporting (Date Range, Rate Type, and Customer History)

- CSV Export and PDF Print Options

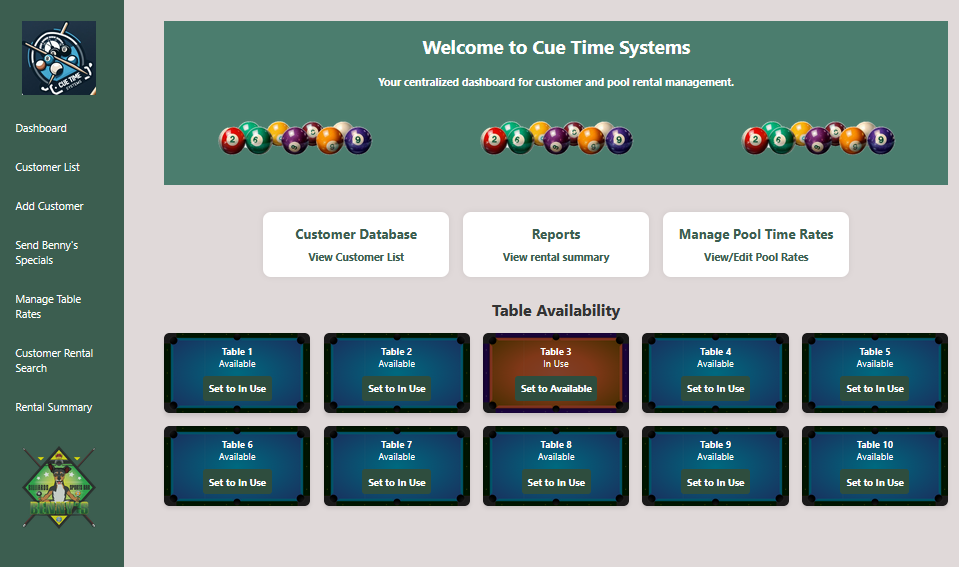
- Mock Email Campaigns and Individual Customer Messaging

-Manage Rental Rates

# Page-by-Page Breakdown

## 1. index.php (Dashboard)

The landing page for the Cue Time Systems application. It provides a summary of the system, quick access cards to Customer Database and Reports, as well as a Sidebar Menu for ease of navigation through all pages. Layout of tables showinf if available and allows manager to manually mark a table ‘In Use’ or ‘Available’ with button on each table, if a table is in service or special need. Displays a professional layout with branding.



Programming Code – Index.php

<?php

// File: index.php

// Description: This is the main dashboard page for Cue Time Systems, displaying customer and pool rental management options.

// Start the session

session\_start();

// Azure DB connection for dashboard

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

// Create a connection to the Azure SQL Database using PDO

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

} catch (PDOException $e) {

die("❌ Connection failed: " . $e->getMessage()); // Handle connection error

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

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

<title>Cue Time Systems - Dashboard</title>

<link rel="stylesheet" href="assets/css/style.css" />

</head>

<body>

<?php include 'sidebar.php'; // Include the sidebar for navigation ?>

<main class="main-content">

<header class="main-header" style="text-align: center;">

<div>

<h1>Welcome to Cue Time Systems</h1><br>

<p><strong>Your centralized dashboard for customer and pool rental management.</strong></p>

<div class="banner-container" style="display: flex; justify-content: center; margin-top: 20px;">

<img src="images/bannerballs.png" alt="Pool Balls Banner 1" style="width: 20%; height: auto;">

<img src="images/bannerballs.png" alt="Pool Balls Banner 2" style="width: 20%; height: auto;">

<img src="images/bannerballs.png" alt="Pool Balls Banner 3" style="width: 20%; height: auto;">

</div>

</header>

<section class="dashboard-cards">

<div class="card">

<h2>Customer Database</h2>

<p><a href="CustomerList.php">View Customer List</a></p>

</div>

<div class="card">

<h2>Reports</h2>

<p><a href="RentalSummaryReport.php">View rental summary</a></p>

</div>

<div class="card">

<h2>Manage Pool Time Rates</h2>

<p><a href="ManageRates.php">View/Edit Pool Rates</a></p>

</div>

</section>

<section class="pool-grid">

<h2 style="text-align: center; margin-top: 2rem;">Table Availability</h2>

<div class="pool-grid-container">

<?php

// Fetch table availability data

$sql = "SELECT TableID, IsAvailable FROM dbo.POOLTABLES ORDER BY TableID ASC";

$stmt = $conn->query($sql);

// Check if the query was successful

if ($stmt === false) {

die("❌ Query failed: " . print\_r($conn->errorInfo(), true));

}

// Initialize a flag to check if rows are returned

$rowsFound = false;

// Loop through each table and display it

while ($row = $stmt->fetch(PDO::FETCH\_ASSOC)) {

$rowsFound = true; // Set the flag to true when a row is found

$tableID = $row['TableID'];

$isAvailable = $row['IsAvailable'] ? 'Available' : 'In Use';

$statusClass = $row['IsAvailable'] ? 'available' : 'in-use'; // CSS class for styling

$newStatus = $row['IsAvailable'] ? 0 : 1; // Determine the new status to toggle

?>

<div class="pool-card <?= $statusClass ?>">

<img src="images/pool-table.png" alt="Pool Table <?= $tableID ?>" class="pool-table-image">

<div class="overlay">

<p><strong>Table <?= $tableID ?></strong></p>

<p><?= $isAvailable ?></p>

<!-- Toggle Button -->

<form method="POST" action="UpdateTableStatus.php" style="margin-top: 10px;">

<input type="hidden" name="TableID" value="<?= $tableID ?>">

<input type="hidden" name="IsAvailable" value="<?= $newStatus ?>">

<button type="submit" class="button">

Set to <?= $newStatus ? 'Available' : 'In Use' ?>

</button>

</form>

</div>

</div>

<?php

}

// If no rows were found, display a message

if (!$rowsFound) {

echo "<p style='text-align: center;'>No tables found in the database.</p>";

}

?>

</div>

</section>

</main>

</body>

</html>

Navigation Menu:

Sidebar.php

<!-- sidebar.php -->

<div class="sidebar">

<div>

<div class="logo-container">

<img src="images/CueTimelogo.png" alt="Cue Time Systems" class="logo" />

</div>

<nav>

<ul>

<li><a href="index.php">Dashboard</a></li>

<li><a href="CustomerList.php">Customer List</a></li>

<li><a href="CustomerAdd.php">Add Customer</a></li>

<li><a href="SendPromoEmail.php">Send Benny's Specials</a></li>

<li><a href="ManageRates.PHP">Manage Table Rates</a></li>

<li><a href="CustomerRentalSearch.php">Customer Rental Search</a></li>

<li><a href="RentalSummaryReport.php">Rental Summary</a></li>

</ul>

</nav>

</div>

<div class="benny-logo-container">

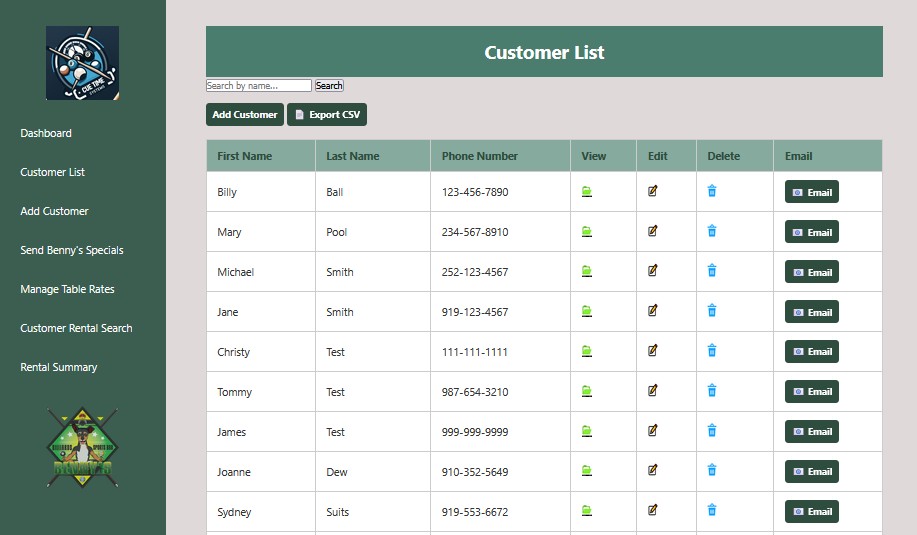
<img src="images/BennysLogo.png" alt="Benny's Logo" class="logo-benny" />

</div>

</div>

## 2. CustomerList.php

Displays all customers from the database in a table format. Allows searching by name or phone, and provides action buttons to View, Edit, Delete, or Email each customer. Also includes CSV export functionality.



CustomerList.php

<?php

// CustomerList.php - Cue Time Systems Customer Management

// Database connection

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

} catch (PDOException $e) {

die("❌ Connection failed: " . $e->getMessage());

}

?>

<!DOCTYPE HTML>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<title>Cue Time Systems - Customer List</title>

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.button {

background-color: #2e4d3e;

color: white;

padding: 5px 10px;

border-radius: 5px;

font-size: 0.9em;

text-decoration: none;

}

.button:hover {

background-color: #3e6a58;

}

table {

width: 100%;

border-collapse: collapse;

margin-top: 20px;

}

th, td {

border: 1px solid #ccc;

padding: 10px;

text-align: left;

}

th {

background-color: #87aa9f;

color: #2e4d3e;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<header class="main-header">

<h1>Customer List</h1>

</header>

<!-- Search Form -->

<form method="GET" action="CustomerList.php" style="margin-bottom: 1rem;">

<input type="text" name="search" placeholder="Search by name..." value="<?= htmlspecialchars($\_GET['search'] ?? '') ?>">

<input type="submit" value="Search">

</form>

<!-- Action Buttons -->

<a href="CustomerAdd.php" class="button">Add Customer</a>

<a class="button no-print" href="ExportCustomerCSV.php">📄 Export CSV</a>

<!-- Customer Table -->

<div class="box">

<table>

<tr>

<th>First Name</th>

<th>Last Name</th>

<th>Phone Number</th>

<th>View</th>

<th>Edit</th>

<th>Delete</th>

<th>Email</th>

</tr>

<?php

try {

$search = $\_GET['search'] ?? '';

if (!empty($search)) {

$sql = "SELECT CustomerID, FirstName, LastName, PhoneNumber FROM dbo.CUSTOMER

WHERE FirstName LIKE :first

OR LastName LIKE :last

OR PhoneNumber LIKE :phone

ORDER BY CustomerID ASC";

$stmt = $conn->prepare($sql);

$searchTerm = '%' . $search . '%';

$stmt->bindValue(':first', $searchTerm, PDO::PARAM\_STR);

$stmt->bindValue(':last', $searchTerm, PDO::PARAM\_STR);

$stmt->bindValue(':phone', $searchTerm, PDO::PARAM\_STR);

$stmt->execute();

} else {

$sql = "SELECT CustomerID, FirstName, LastName, PhoneNumber FROM dbo.CUSTOMER ORDER BY CustomerID ASC";

$stmt = $conn->query($sql);

}

$customers = $stmt->fetchAll(PDO::FETCH\_ASSOC);

if ($customers) {

foreach ($customers as $row) {

echo "<tr>";

echo "<td>" . htmlspecialchars($row['FirstName']) . "</td>";

echo "<td>" . htmlspecialchars($row['LastName']) . "</td>";

echo "<td>" . htmlspecialchars($row['PhoneNumber']) . "</td>";

echo "<td><a href='CustomerView.php?CustomerID=" . $row['CustomerID'] . "'><img src='images/view.png' alt='View' width='15' height='20'></a></td>";

echo "<td><a href='CustomerEdit.php?CustomerID=" . $row['CustomerID'] . "'><img src='images/edit-icon.png' alt='Edit' width='15' height='20'></a></td>";

echo "<td><a href='CustomerDelete.php?CustomerID=" . $row['CustomerID'] . "'><img src='images/delete-icon.png' alt='Delete' width='15' height='20'></a></td>";

echo "<td><a class='button no-print' href='EmailSingleCustomer.php?id=" . $row['CustomerID'] . "'>📧 Email</a></td>";

echo "</tr>";

}

} else {

echo "<tr><td colspan='7'>No customers found.</td></tr>";

}

} catch (PDOException $e) {

echo "<tr><td colspan='6'>❌ Error: " . $e->getMessage() . "</td></tr>";

}

?>

</table>

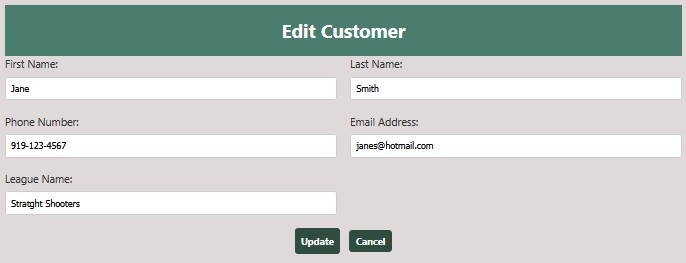
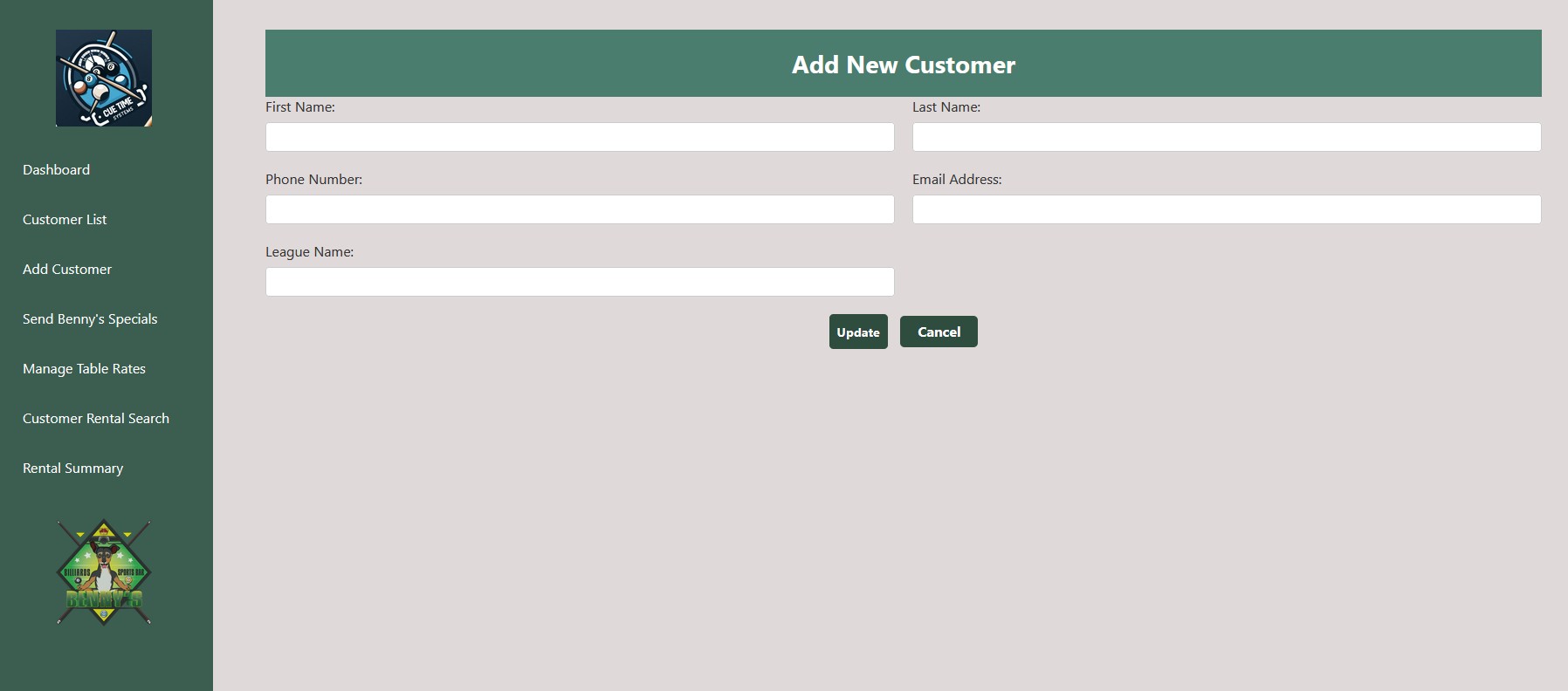
</div>

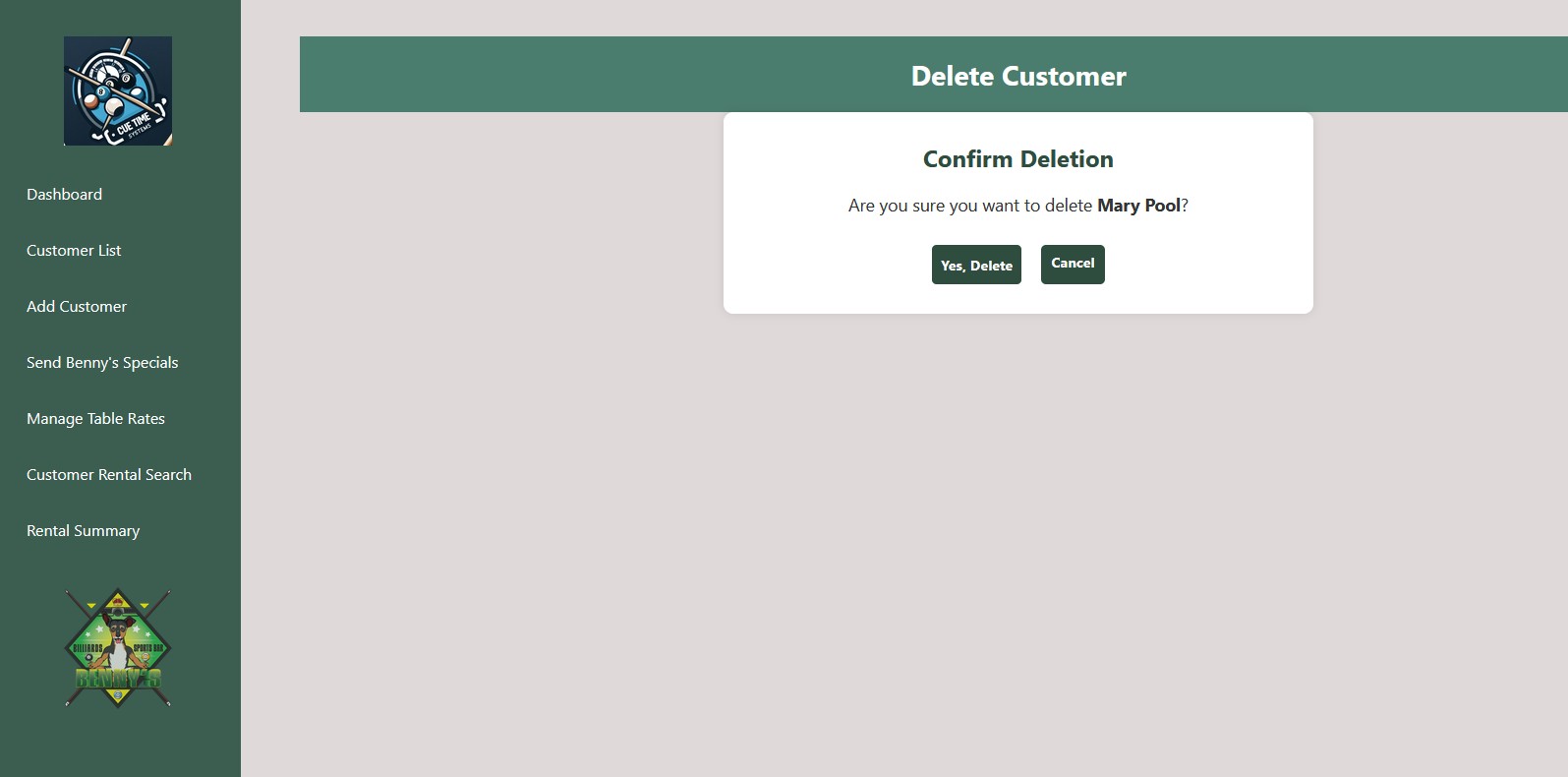
</main>

</body>

</html>

## 3. CustomerAdd.php / CustomerEdit.php / CustomerDelete.php / CustomerView.php

These pages allow adding a new customer, updating existing customer details, confirming deletion, and viewing full customer details respectively. Each interacts directly with the Azure SQL database.



CustomerAdd.php

<!DOCTYPE HTML>

<html>

<head>

<title>Cue Time Systems - Add Customer</title>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.form-grid {

display: grid;

grid-template-columns: 1fr 1fr;

gap: 20px;

margin-bottom: 20px;

}

.form-grid label {

display: block;

margin-bottom: 8px;

font-weight: 500;

}

.form-grid input[type="text"],

.form-grid input[type="email"] {

width: 100%;

padding: 8px;

border-radius: 4px;

border: 1px solid #ccc;

}

.form-actions {

text-align: center;

margin-top: 30px;

}

.form-actions .button {

font-family: 'Segoe UI', sans-serif;

font-size: 1rem;

padding: 10px 20px;

background-color: #2e4d3e;

border: none;

color: white;

border-radius: 5px;

cursor: pointer;

transition: background-color 0.3s ease;

}

.form-actions .button:hover {

background-color: #6e978a;

}

.main-header h1 {

text-align: center;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; // Sidbar navigation call to be on page?>

<main class="main-content">

<header class="main-header">

<h1>Add New Customer</h1>

</header>

<div class="box">

<form action="CustomerAddProcessing.php" method="post">

<div class="form-grid">

<div>

<label for="FirstName">First Name:</label>

<input type="text" id="FirstName" name="FirstName" required>

</div>

<div>

<label for="LastName">Last Name:</label>

<input type="text" id="LastName" name="LastName" required>

</div>

<div>

<label for="PhoneNumber">Phone Number:</label>

<input type="text" id="PhoneNumber" name="PhoneNumber" required>

</div>

<div>

<label for="EmailAddress">Email Address:</label>

<input type="email" id="EmailAddress" name="EmailAddress" required>

</div>

<div>

<label for="LeagueName">League Name:</label>

<input type="text" id="LeagueName" name="LeagueName">

</div>

</div>

<div class="form-actions" style="margin-top: 1rem;">

<input type="submit" value="Update" class="button">

<a href="CustomerList.php" class="button" style="margin-left: 10px;">Cancel</a>

</div>

</form>

</div>

</main>

</body>

</html>

CustomerAddProcessing.php

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$FirstName = $\_POST['FirstName'];

$LastName = $\_POST['LastName'];

$PhoneNumber = $\_POST['PhoneNumber'];

$EmailAddress = $\_POST['EmailAddress'];

$LeagueName = $\_POST['LeagueName'];

// Azure SQL connection

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

$sql = "INSERT INTO customer (FirstName, LastName, PhoneNumber, EmailAddress, LeagueName)

VALUES (:FirstName, :LastName, :PhoneNumber, :EmailAddress, :LeagueName)";

$stmt = $conn->prepare($sql);

$stmt->bindParam(':FirstName', $FirstName);

$stmt->bindParam(':LastName', $LastName);

$stmt->bindParam(':PhoneNumber', $PhoneNumber);

$stmt->bindParam(':EmailAddress', $EmailAddress);

$stmt->bindParam(':LeagueName', $LeagueName);

if ($stmt->execute()) {

header("Location: CustomerList.php");

exit();

} else {

echo "<h2>Error adding customer.</h2>";

}

} catch (PDOException $e) {

echo "<h2>Connection failed: " . $e->getMessage() . "</h2>";

}

} else {

echo "<h2>Invalid request.</h2>";

}

?>

CustomerEdit.php

<?php

if (!isset($\_GET['CustomerID'])) {

echo "<h2>No Customer ID Provided</h2>";

exit();

}

$CustomerID = intval($\_GET['CustomerID']);

// Azure SQL DB connection

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

$stmt = $conn->prepare("SELECT \* FROM customer WHERE CustomerID = :CustomerID");

$stmt->bindParam(':CustomerID', $CustomerID, PDO::PARAM\_INT);

$stmt->execute();

$row = $stmt->fetch(PDO::FETCH\_ASSOC);

if (!$row) {

echo "<h2>Customer not found.</h2>";

exit();

}

} catch (PDOException $e) {

die("Connection failed: " . $e->getMessage());

}

?>

<!DOCTYPE HTML>

<html>

<head>

<title>Cue Time Systems - Edit Customer</title>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.form-grid {

display: grid;

grid-template-columns: 1fr 1fr;

gap: 20px;

margin-bottom: 20px;

}

.form-grid label {

display: block;

margin-bottom: 8px;

font-weight: 500;

}

.form-grid input[type="text"],

.form-grid input[type="email"] {

width: 100%;

padding: 8px;

border-radius: 4px;

border: 1px solid #ccc;

}

.form-actions {

text-align: center;

margin-top: 30px;

}

.main-header h1 {

text-align: center;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<header class="main-header">

<h1>Edit Customer</h1>

</header>

<div class="box">

<form action="CustomerUpdateProcessing.php" method="post">

<input type="hidden" name="CustomerID" value="<?= htmlspecialchars($CustomerID) ?>">

<div class="form-grid">

<div>

<label for="FirstName">First Name:</label>

<input type="text" id="FirstName" name="FirstName" value="<?= htmlspecialchars($row['FirstName']) ?>" required>

</div>

<div>

<label for="LastName">Last Name:</label>

<input type="text" id="LastName" name="LastName" value="<?= htmlspecialchars($row['LastName']) ?>" required>

</div>

<div>

<label for="PhoneNumber">Phone Number:</label>

<input type="text" id="PhoneNumber" name="PhoneNumber" value="<?= htmlspecialchars($row['PhoneNumber']) ?>" required>

</div>

<div>

<label for="EmailAddress">Email Address:</label>

<input type="email" id="EmailAddress" name="EmailAddress" value="<?= htmlspecialchars($row['EmailAddress']) ?>" required>

</div>

<div>

<label for="LeagueName">League Name:</label>

<input type="text" id="LeagueName" name="LeagueName" value="<?= htmlspecialchars($row['LeagueName']) ?>">

</div>

</div>

<div class="form-actions" style="margin-top: 1rem;">

<input type="submit" value="Update" class="button">

<a href="CustomerList.php" class="button" style="margin-left: 10px;">Cancel</a>

</div>

</form>

</div>

</main>

</body>

</html>

CustomerUpdateProcessing.php

<?php

if (!isset($\_GET['CustomerID'])) {

echo "<h2>No Customer ID Provided</h2>";

exit();

}

$CustomerID = intval($\_GET['CustomerID']);

// Azure SQL DB connection

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

$stmt = $conn->prepare("SELECT \* FROM customer WHERE CustomerID = :CustomerID");

$stmt->bindParam(':CustomerID', $CustomerID, PDO::PARAM\_INT);

$stmt->execute();

$row = $stmt->fetch(PDO::FETCH\_ASSOC);

if (!$row) {

echo "<h2>Customer not found.</h2>";

exit();

}

} catch (PDOException $e) {

die("Connection failed: " . $e->getMessage());

}

?>

<!DOCTYPE HTML>

<html>

<head>

<title>Cue Time Systems - Edit Customer</title>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.form-grid {

display: grid;

grid-template-columns: 1fr 1fr;

gap: 20px;

margin-bottom: 20px;

}

.form-grid label {

display: block;

margin-bottom: 8px;

font-weight: 500;

}

.form-grid input[type="text"],

.form-grid input[type="email"] {

width: 100%;

padding: 8px;

border-radius: 4px;

border: 1px solid #ccc;

}

.form-actions {

text-align: center;

margin-top: 30px;

}

.main-header h1 {

text-align: center;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<header class="main-header">

<h1>Edit Customer</h1>

</header>

<div class="box">

<form action="CustomerUpdateProcessing.php" method="post">

<input type="hidden" name="CustomerID" value="<?= htmlspecialchars($CustomerID) ?>">

<div class="form-grid">

<div>

<label for="FirstName">First Name:</label>

<input type="text" id="FirstName" name="FirstName" value="<?= htmlspecialchars($row['FirstName']) ?>" required>

</div>

<div>

<label for="LastName">Last Name:</label>

<input type="text" id="LastName" name="LastName" value="<?= htmlspecialchars($row['LastName']) ?>" required>

</div>

<div>

<label for="PhoneNumber">Phone Number:</label>

<input type="text" id="PhoneNumber" name="PhoneNumber" value="<?= htmlspecialchars($row['PhoneNumber']) ?>" required>

</div>

<div>

<label for="EmailAddress">Email Address:</label>

<input type="email" id="EmailAddress" name="EmailAddress" value="<?= htmlspecialchars($row['EmailAddress']) ?>" required>

</div>

<div>

<label for="LeagueName">League Name:</label>

<input type="text" id="LeagueName" name="LeagueName" value="<?= htmlspecialchars($row['LeagueName']) ?>">

</div>

</div>

<div class="form-actions" style="margin-top: 1rem;">

<input type="submit" value="Update" class="button">

<a href="CustomerList.php" class="button" style="margin-left: 10px;">Cancel</a>

</div>

</form>

</div>

</main>

</body>

</html>

CustomerDelete.php

<?php

$CustomerID = intval($\_GET['CustomerID'] ?? $\_POST['CustomerID'] ?? 0);

$FirstName = $LastName = "";

$deleted = false;

$message = "";

// Azure SQL connection settings

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

// If form was submitted via POST, perform delete

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && $CustomerID > 0) {

$stmt = $conn->prepare("DELETE FROM customer WHERE CustomerID = :CustomerID");

$stmt->bindParam(':CustomerID', $CustomerID, PDO::PARAM\_INT);

if ($stmt->execute()) {

header("Location: CustomerList.php");

exit();

} else {

$message = "Failed to delete customer.";

}

}

// Preload customer name for confirmation

if ($CustomerID > 0) {

$stmt = $conn->prepare("SELECT FirstName, LastName FROM customer WHERE CustomerID = :CustomerID");

$stmt->bindParam(':CustomerID', $CustomerID, PDO::PARAM\_INT);

$stmt->execute();

$row = $stmt->fetch(PDO::FETCH\_ASSOC);

if ($row) {

$FirstName = $row['FirstName'];

$LastName = $row['LastName'];

}

}

} catch (PDOException $e) {

$message = "Connection failed: " . $e->getMessage();

}

?>

<!DOCTYPE HTML>

<html>

<head>

<title>Cue Time Systems - Delete Customer</title>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.confirm-box {

background-color: #fff;

padding: 30px;

border-radius: 10px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

max-width: 600px;

margin: 0 auto;

}

.confirm-box h2 {

text-align: center;

color: #2e4d3e;

margin-bottom: 20px;

}

.confirm-box p {

text-align: center;

font-size: 1.1rem;

margin-bottom: 30px;

}

.confirm-buttons {

display: flex;

justify-content: center;

gap: 20px;

}

.main-header h1 {

text-align: center;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<header class="main-header">

<h1>Delete Customer</h1>

</header>

<div class="confirm-box">

<h2>Confirm Deletion</h2>

<?php if (!empty($FirstName)): ?>

<p>Are you sure you want to delete <strong><?= htmlspecialchars($FirstName . ' ' . $LastName) ?></strong>?</p>

<form action="CustomerDelete.php" method="post">

<input type="hidden" name="CustomerID" value="<?= $CustomerID ?>">

<div class="confirm-buttons">

<input type="submit" value="Yes, Delete" class="button">

<a href="CustomerList.php" class="button">Cancel</a>

</div>

</form>

<?php else: ?>

<p>No customer found to delete.</p>

<div class="confirm-buttons">

<a href="CustomerList.php" class="button">Back to List</a>

</div>

<?php endif; ?>

</div>

</main>

</body>

</html>

CustomerView.php

<?php

$CustomerID = intval($\_GET['CustomerID'] ?? $\_POST['CustomerID'] ?? 0);

$FirstName = $LastName = "";

$deleted = false;

$message = "";

// Azure SQL connection settings

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

// If form was submitted via POST, perform delete

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && $CustomerID > 0) {

$stmt = $conn->prepare("DELETE FROM customer WHERE CustomerID = :CustomerID");

$stmt->bindParam(':CustomerID', $CustomerID, PDO::PARAM\_INT);

if ($stmt->execute()) {

header("Location: CustomerList.php");

exit();

} else {

$message = "Failed to delete customer.";

}

}

// Preload customer name for confirmation

if ($CustomerID > 0) {

$stmt = $conn->prepare("SELECT FirstName, LastName FROM customer WHERE CustomerID = :CustomerID");

$stmt->bindParam(':CustomerID', $CustomerID, PDO::PARAM\_INT);

$stmt->execute();

$row = $stmt->fetch(PDO::FETCH\_ASSOC);

if ($row) {

$FirstName = $row['FirstName'];

$LastName = $row['LastName'];

}

}

} catch (PDOException $e) {

$message = "Connection failed: " . $e->getMessage();

}

?>

<!DOCTYPE HTML>

<html>

<head>

<title>Cue Time Systems - Delete Customer</title>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.confirm-box {

background-color: #fff;

padding: 30px;

border-radius: 10px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

max-width: 600px;

margin: 0 auto;

}

.confirm-box h2 {

text-align: center;

color: #2e4d3e;

margin-bottom: 20px;

}

.confirm-box p {

text-align: center;

font-size: 1.1rem;

margin-bottom: 30px;

}

.confirm-buttons {

display: flex;

justify-content: center;

gap: 20px;

}

.main-header h1 {

text-align: center;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<header class="main-header">

<h1>Delete Customer</h1>

</header>

<div class="confirm-box">

<h2>Confirm Deletion</h2>

<?php if (!empty($FirstName)): ?>

<p>Are you sure you want to delete <strong><?= htmlspecialchars($FirstName . ' ' . $LastName) ?></strong>?</p>

<form action="CustomerDelete.php" method="post">

<input type="hidden" name="CustomerID" value="<?= $CustomerID ?>">

<div class="confirm-buttons">

<input type="submit" value="Yes, Delete" class="button">

<a href="CustomerList.php" class="button">Cancel</a>

</div>

</form>

<?php else: ?>

<p>No customer found to delete.</p>

<div class="confirm-buttons">

<a href="CustomerList.php" class="button">Back to List</a>

</div>

<?php endif; ?>

</div>

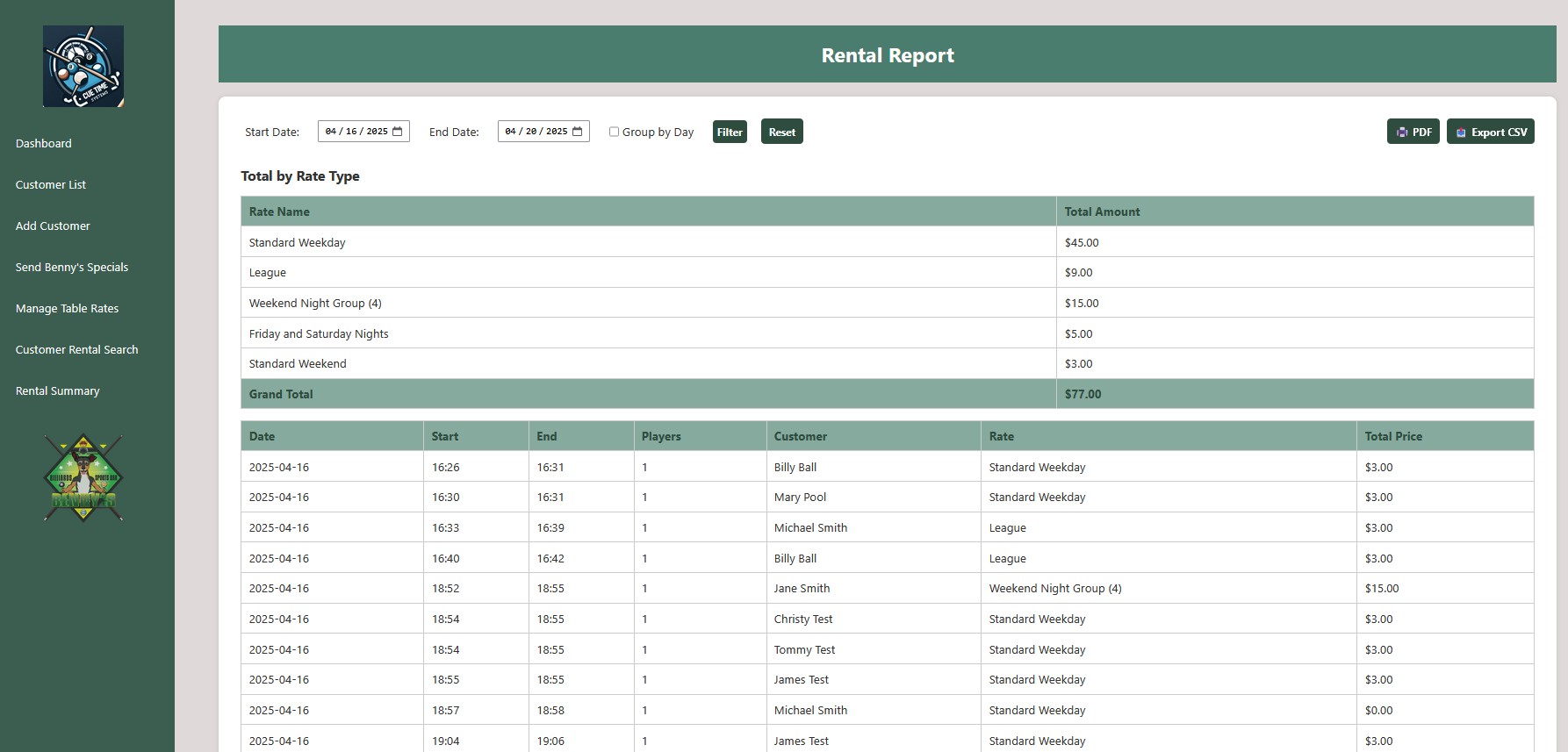
</main>

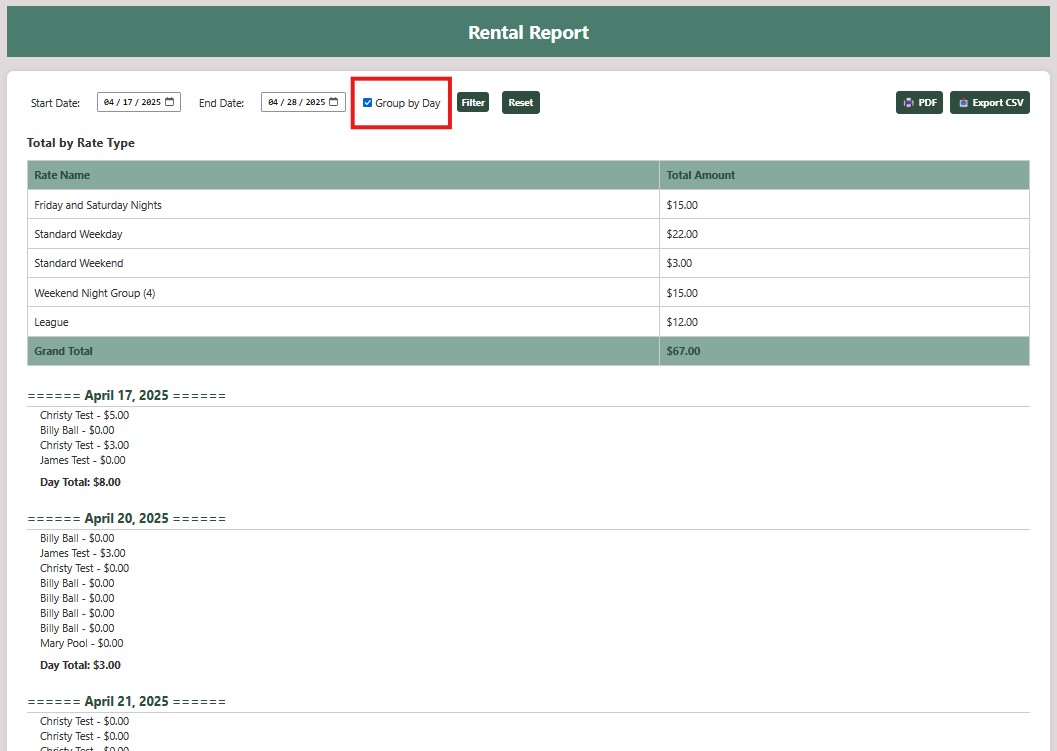
</body>

</html>

## 4. RentalSummaryReport.php

Generates a date-range-based summary of all pool table rentals, showing date, times, rate type, customer, and total price. Includes filters, export to CSV, and 'Print to PDF' using dompdf. Totals are summarized by rate type. Added a ‘Group by day’ view





RentalSummaryReport.php

<?php

// RentalSummaryReport.php

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

$startDate = $\_GET['start'] ?? date('Y-m-01');

$endDate = $\_GET['end'] ?? date('Y-m-d');

$groupByDay = isset($\_GET['groupByDay']);

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

$sql = "

SELECT

pr.RentalDate,

pr.RentalStart,

pr.RentalEnd,

pr.TotalPlayers,

pr.TotalPrice,

CONCAT(c.FirstName, ' ', c.LastName) AS CustomerName,

r.RateName

FROM POOLRENTAL pr

JOIN CUSTOMER c ON pr.CustomerID = c.CustomerID

JOIN RATES r ON pr.RateID = r.RateID

WHERE pr.RentalDate BETWEEN :start AND :end

ORDER BY pr.RentalDate ASC, pr.RentalStart

";

$stmt = $conn->prepare($sql);

$stmt->bindParam(':start', $startDate);

$stmt->bindParam(':end', $endDate);

$stmt->execute();

$rentals = $stmt->fetchAll(PDO::FETCH\_ASSOC);

$totalsByRate = [];

$grandTotal = 0;

foreach ($rentals as $row) {

$rate = $row['RateName'];

$totalsByRate[$rate] = ($totalsByRate[$rate] ?? 0) + $row['TotalPrice'];

$grandTotal += $row['TotalPrice'];

}

} catch (PDOException $e) {

die("Connection failed: " . $e->getMessage());

}

?>

<!DOCTYPE HTML>

<html>

<head>

<title>Cue Time Systems - Rental Summary Report</title>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.report-box {

background-color: #fff;

padding: 30px;

border-radius: 10px;

box-shadow: 0 0 8px rgba(0, 0, 0, 0.1);

margin: 20px auto;

}

.report-filters {

margin-bottom: 20px;

}

.report-filters input[type="date"],

.report-filters input[type="submit"],

.report-filters label {

padding: 6px;

margin-right: 10px;

}

.report-table {

width: 100%;

border-collapse: collapse;

margin-top: 15px;

}

.report-table th, .report-table td {

border: 1px solid #ccc;

padding: 10px;

}

.report-table th {

background-color: #87aa9f;

color: #2e4d3e;

}

.report-summary {

margin-top: 30px;

}

.daily-header {

margin-top: 30px;

font-weight: bold;

font-size: 1.2em;

color: #2e4d3e;

border-bottom: 2px solid #ccc;

padding-bottom: 5px;

}

.customer-line {

margin-left: 20px;

font-size: 1em;

}

.day-total-line {

margin-left: 20px;

font-weight: bold;

color: #333;

margin-top: 10px;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<header class="main-header">

<h1>Rental Report</h1>

</header>

<div class="report-box">

<form method="get" class="report-filters" style="display: flex; justify-content: space-between; align-items: center; flex-wrap: wrap; gap: 10px;">

<div style="display: flex; align-items: center; flex-wrap: wrap; gap: 10px;">

<label for="start">Start Date:</label>

<input type="date" id="start" name="start" value="<?= htmlspecialchars($startDate) ?>">

<label for="end">End Date:</label>

<input type="date" id="end" name="end" value="<?= htmlspecialchars($endDate) ?>">

<label>

<input type="checkbox" name="groupByDay" value="1" <?= $groupByDay ? 'checked' : '' ?>>

Group by Day

</label>

<input type="submit" value="Filter" class="button">

<a class="button no-print" href="RentalSummaryReport.php">Reset</a>

</div>

<div style="display: flex; gap: 10px;">

<a class="button no-print" href="RentalSummaryPDF.php?start=<?= urlencode($startDate) ?>&end=<?= urlencode($endDate) ?>" target="\_blank">🖨️ PDF</a>

<a class="button no-print" href="ExportRentalSummaryCSV.php?start=<?= urlencode($startDate) ?>&end=<?= urlencode($endDate) ?>">📤 Export CSV</a>

</div>

</form>

<div class="report-summary">

<h3>Total by Rate Type</h3>

<table class="report-table">

<thead>

<tr>

<th>Rate Name</th>

<th>Total Amount</th>

</tr>

</thead>

<tbody>

<?php foreach ($totalsByRate as $rate => $total): ?>

<tr>

<td><?= htmlspecialchars($rate) ?></td>

<td>$<?= number\_format($total, 2) ?></td>

</tr>

<?php endforeach; ?>

<tr>

<th>Grand Total</th>

<th>$<?= number\_format($grandTotal, 2) ?></th>

</tr>

</tbody>

</table>

</div>

<?php if ($groupByDay): ?>

<?php

$dailyData = [];

foreach ($rentals as $rental) {

$date = date('Y-m-d', strtotime($rental['RentalDate']));

$dailyData[$date][] = $rental;

}

?>

<?php foreach ($dailyData as $date => $dayRentals): ?>

<div class="daily-header">====== <?= htmlspecialchars(date('F j, Y', strtotime($date))) ?> ======</div>

<?php

$dayTotal = 0;

foreach ($dayRentals as $rental):

$dayTotal += $rental['TotalPrice'];

?>

<div class="customer-line"><?= htmlspecialchars($rental['CustomerName']) ?> - $<?= number\_format($rental['TotalPrice'], 2) ?></div>

<?php endforeach; ?>

<div class="day-total-line">Day Total: $<?= number\_format($dayTotal, 2) ?></div>

<?php endforeach; ?>

<?php else: ?>

<table class="report-table">

<thead>

<tr>

<th>Date</th>

<th>Start</th>

<th>End</th>

<th>Players</th>

<th>Customer</th>

<th>Rate</th>

<th>Total Price</th>

</tr>

</thead>

<tbody>

<?php foreach ($rentals as $rental): ?>

<tr>

<td><?= htmlspecialchars(date('Y-m-d', strtotime($rental['RentalDate']))) ?></td>

<td><?= htmlspecialchars(substr($rental['RentalStart'], 0, 5)) ?></td>

<td><?= htmlspecialchars(substr($rental['RentalEnd'], 0, 5)) ?></td>

<td><?= htmlspecialchars($rental['TotalPlayers']) ?></td>

<td><?= htmlspecialchars($rental['CustomerName']) ?></td>

<td><?= htmlspecialchars($rental['RateName']) ?></td>

<td>$<?= number\_format($rental['TotalPrice'], 2) ?></td>

</tr>

<?php endforeach; ?>

</tbody>

</table>

<?php endif; ?>

</div>

</main>

</body>

</html>

## 5. RentalSummaryPDF.php

Creates a downloadable/printable PDF version of the Rental Summary using dompdf. Accepts start and end date filters via URL.

RentalSummaryPDF.php

<?php

require\_once 'dompdf/autoload.inc.php';

use Dompdf\Dompdf;

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

$startDate = $\_GET['start'] ?? date('Y-m-01');

$endDate = $\_GET['end'] ?? date('Y-m-d');

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

// Fetch data

$sql = "

SELECT pr.RentalDate, pr.RentalStart, pr.RentalEnd, pr.TotalPlayers, pr.TotalPrice,

CONCAT(c.FirstName, ' ', c.LastName) AS CustomerName, r.RateName

FROM POOLRENTAL pr

JOIN CUSTOMER c ON pr.CustomerID = c.CustomerID

JOIN RATES r ON pr.RateID = r.RateID

WHERE pr.RentalDate BETWEEN :start AND :end

ORDER BY pr.RentalDate DESC, pr.RentalStart

";

$stmt = $conn->prepare($sql);

$stmt->bindParam(':start', $startDate);

$stmt->bindParam(':end', $endDate);

$stmt->execute();

$rentals = $stmt->fetchAll(PDO::FETCH\_ASSOC);

// Calculate totals by rate

$totalsByRate = [];

foreach ($rentals as $row) {

$rate = $row['RateName'];

$totalsByRate[$rate] = ($totalsByRate[$rate] ?? 0) + $row['TotalPrice'];

}

// Build HTML

$html = "<h2>Rental Summary Report</h2>

<p>From: $startDate to $endDate</p>

<table border='1' cellspacing='0' cellpadding='5' width='100%'>

<tr><th>Date</th><th>Start</th><th>End</th><th>Players</th><th>Customer</th><th>Rate</th><th>Total Price</th></tr>";

foreach ($rentals as $r) {

$html .= "<tr>

<td>{$r['RentalDate']}</td>

<td>" . substr($r['RentalStart'], 0, 5) . "</td>

<td>" . substr($r['RentalEnd'], 0, 5) . "</td>

<td>{$r['TotalPlayers']}</td>

<td>" . htmlspecialchars($r['CustomerName']) . "</td>

<td>" . htmlspecialchars($r['RateName']) . "</td>

<td>$" . number\_format($r['TotalPrice'], 2) . "</td>

</tr>";

}

$html .= "</table><br><h3>Total by Rate Type</h3>

<table border='1' cellspacing='0' cellpadding='5' width='50%'>

<tr><th>Rate Name</th><th>Total</th></tr>";

foreach ($totalsByRate as $rate => $total) {

$html .= "<tr><td>" . htmlspecialchars($rate) . "</td><td>$" . number\_format($total, 2) . "</td></tr>";

}

$html .= "</table>";

// Generate PDF

$dompdf = new Dompdf();

$dompdf->loadHtml($html);

$dompdf->setPaper('A4', 'portrait');

$dompdf->render();

$dompdf->stream("RentalSummaryReport.pdf", array("Attachment" => true));

?>

ExportRentalSummaryCSV.php

<?php

$startDate = $\_GET['start'] ?? date('Y-m-01');

$endDate = $\_GET['end'] ?? date('Y-m-d');

header('Content-Type: text/csv');

header('Content-Disposition: attachment;filename=RentalSummaryReport.csv');

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

$sql = "

SELECT

pr.RentalDate,

pr.RentalStart,

pr.RentalEnd,

pr.TotalPlayers,

pr.TotalPrice,

CONCAT(c.FirstName, ' ', c.LastName) AS CustomerName,

r.RateName

FROM POOLRENTAL pr

JOIN CUSTOMER c ON pr.CustomerID = c.CustomerID

JOIN RATES r ON pr.RateID = r.RateID

WHERE pr.RentalDate BETWEEN :start AND :end

ORDER BY pr.RentalDate DESC, pr.RentalStart

";

$stmt = $conn->prepare($sql);

$stmt->bindParam(':start', $startDate);

$stmt->bindParam(':end', $endDate);

$stmt->execute();

$rentals = $stmt->fetchAll(PDO::FETCH\_ASSOC);

$output = fopen('php://output', 'w');

fputcsv($output, ['Date', 'Start Time', 'End Time', 'Players', 'Customer', 'Rate', 'Total Price']);

foreach ($rentals as $rental) {

fputcsv($output, [

$rental['RentalDate'],

substr($rental['RentalStart'], 0, 5),

substr($rental['RentalEnd'], 0, 5),

$rental['TotalPlayers'],

$rental['CustomerName'],

$rental['RateName'],

number\_format($rental['TotalPrice'], 2)

]);

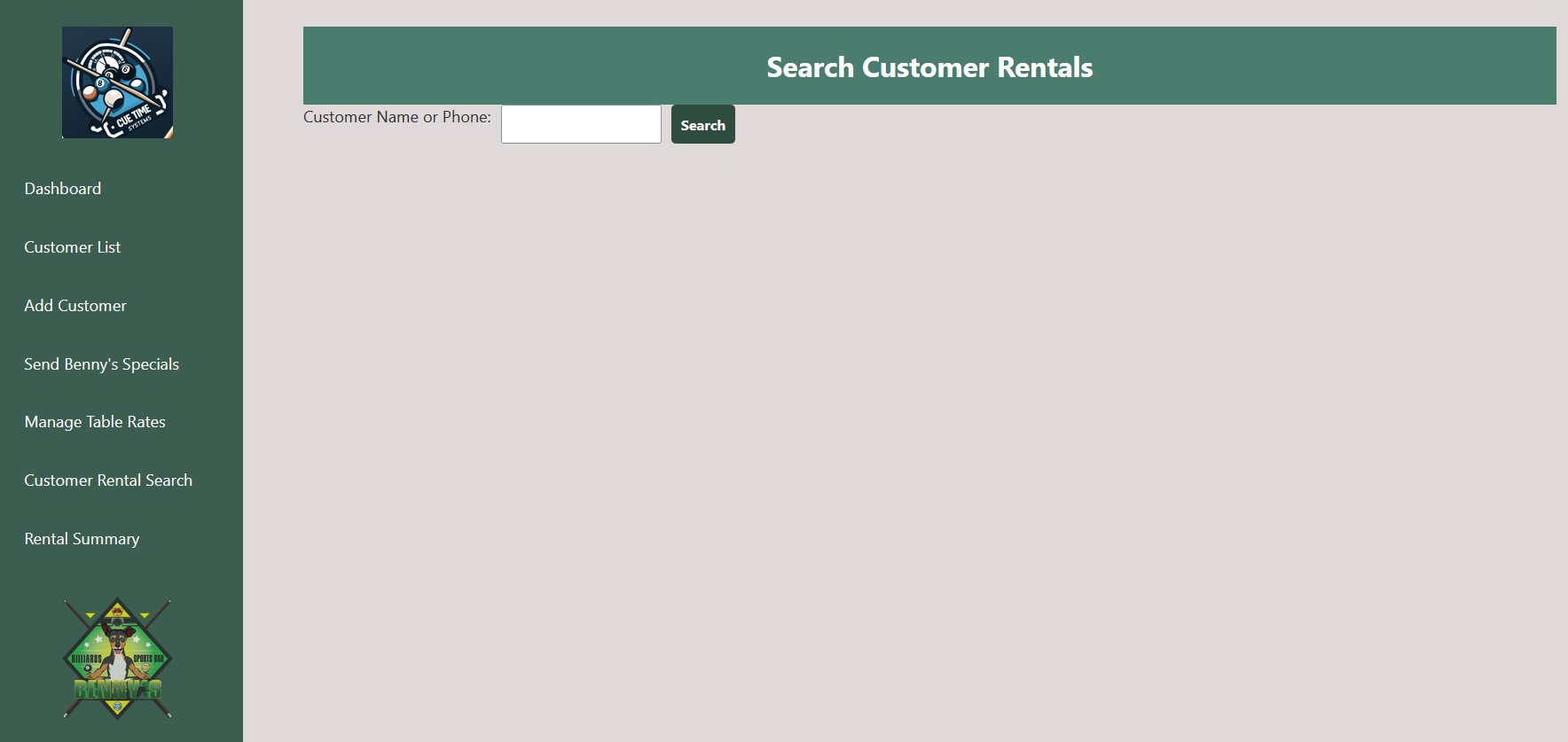
}

fclose($output);

exit;

?>

## 6. CustomerRentalReport.php

Searches for rentals by customer first name, last name or phone number and returns a history of rentals for matching customers. Displays rental date, rate, total players, and total price. Includes 'Print' functionality.

CustomerRentalSearch.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<title>Customer Rental Search</title>

<link rel="stylesheet" href="assets/css/style.css" />

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<header class="main-header">

<h1>Search Customer Rentals</h1>

</header>

<div class="report-box">

<form action="CustomerRentalReport.php" method="post" class="report-filters" style="display: flex; gap: 10px;">

<label for="search">Customer Name or Phone:</label>

<input type="text" id="search" name="search" required>

<button type="submit" class="button">Search</button>

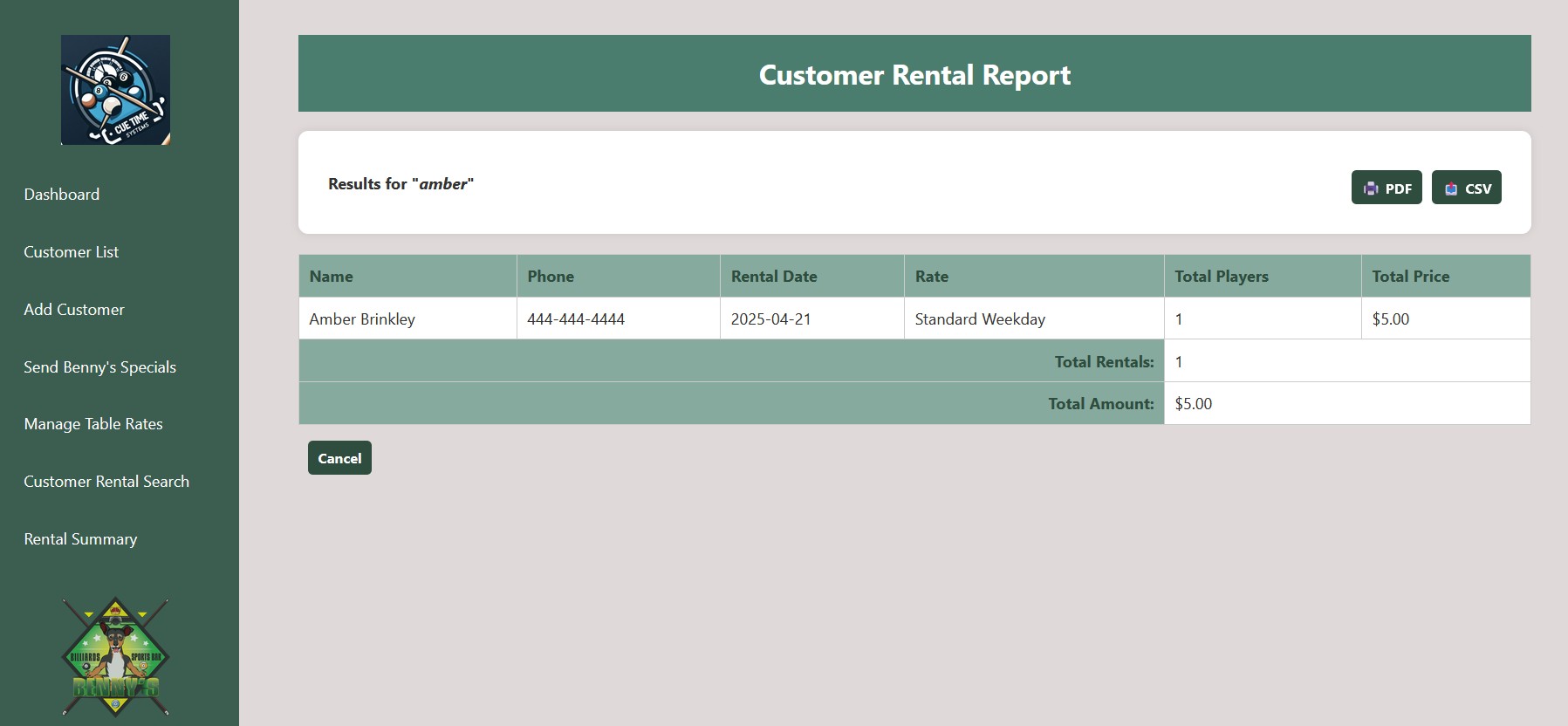
</form>

</div>

</main>

</body>

</html>



CustomerRentalReport.php

<?php

require\_once 'db\_connect.php';

$search = $\_POST['search'] ?? '';

$sql = "

SELECT

c.FirstName,

c.LastName,

c.PhoneNumber,

r.RateName,

pr.RentalDate,

pr.TotalPlayers,

pr.TotalPrice

FROM poolrental pr

JOIN customer c ON pr.CustomerID = c.CustomerID

JOIN rates r ON pr.RateID = r.RateID

WHERE c.FirstName LIKE :first

OR c.LastName LIKE :last

OR c.PhoneNumber LIKE :phone

ORDER BY pr.RentalDate DESC

";

$stmt = $conn->prepare($sql);

$searchParam = '%' . $search . '%';

$stmt->bindValue(':first', $searchParam);

$stmt->bindValue(':last', $searchParam);

$stmt->bindValue(':phone', $searchParam);

$stmt->execute();

$result = $stmt->fetchAll(PDO::FETCH\_ASSOC);

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Customer Rental Report</title>

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.report-box {

background-color: #fff;

padding: 30px;

border-radius: 10px;

box-shadow: 0 0 8px rgba(0, 0, 0, 0.1);

margin: 20px auto;

}

.report-table {

width: 100%;

border-collapse: collapse;

margin-top: 15px;

}

.report-table th, .report-table td {

border: 1px solid #ccc;

padding: 10px;

}

.report-table th {

background-color: #87aa9f;

color: #2e4d3e;

}

.no-print {

margin-top: 10px;

display: inline-block;

}

@media print {

.no-print {

display: none;

}

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<header class="main-header">

<h1>Customer Rental Report</h1>

</header>

<div class="report-box">

<?php if (!empty($search)): ?>

<div style="display: flex; justify-content: space-between; align-items: center;">

<p style="margin: 0; font-weight: bold;">

Results for "<em><?= htmlspecialchars($search) ?></em>"

</p>

<div style="display: flex; gap: 10px;">

<a class="button no-print"

href="CustomerRentalPDF.php?search=<?= urlencode($search) ?>"

target="\_blank">

🖨️ PDF

</a>

<a class="button no-print"

href="ExportCustomerRentalCSV.php?search=<?= urlencode($search) ?>">

📤 CSV

</a>

</div>

</div>

<?php endif; ?>

</div>

<table class="report-table">

<thead>

<tr>

<th>Name</th>

<th>Phone</th>

<th>Rental Date</th>

<th>Rate</th>

<th>Total Players</th>

<th>Total Price</th>

</tr>

</thead>

<tbody>

<?php

$totalRentals = 0;

$totalAmount = 0.00;

foreach ($result as $row):

$totalRentals++;

$totalAmount += $row['TotalPrice'];

?>

<tr>

<td><?= htmlspecialchars($row['FirstName'] . ' ' . $row['LastName']) ?></td>

<td><?= htmlspecialchars($row['PhoneNumber']) ?></td>

<td><?= htmlspecialchars($row['RentalDate']) ?></td>

<td><?= htmlspecialchars($row['RateName']) ?></td>

<td><?= htmlspecialchars($row['TotalPlayers']) ?></td>

<td>$<?= number\_format($row['TotalPrice'], 2) ?></td>

</tr>

<?php endforeach; ?>

</tbody>

<tfoot>

<tr>

<th colspan="4" style="text-align:right;">Total Rentals:</th>

<td colspan="2"><?= $totalRentals ?></td>

</tr>

<tr>

<th colspan="4" style="text-align:right;">Total Amount:</th>

<td colspan="2">$<?= number\_format($totalAmount, 2) ?></td>

</tr>

</tfoot>

</table>

</div>

<div class="form-actions" style="margin-top: 1rem;">

<a href="CustomerRentalSearch.php" class="button" style="margin-left: 10px;">Cancel</a>

</div>

</main>

</body>

</html>

## 7. CustomerRentalPDF.php

Generates a printable PDF report for a customer’s rental history using the same filters.

CustomerRentalPDF.php

<?php

require\_once 'dompdf/autoload.inc.php';

use Dompdf\Dompdf;

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

$search = $\_GET['search'] ?? '';

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

$sql = "

SELECT

c.FirstName,

c.LastName,

c.PhoneNumber,

r.RateName,

pr.RentalDate,

pr.TotalPlayers,

pr.TotalPrice

FROM poolrental pr

JOIN customer c ON pr.CustomerID = c.CustomerID

JOIN rates r ON pr.RateID = r.RateID

WHERE c.FirstName LIKE :first

OR c.LastName LIKE :last

OR c.PhoneNumber LIKE :phone

ORDER BY pr.RentalDate DESC

";

$stmt = $conn->prepare($sql);

$searchParam = '%' . $search . '%';

$stmt->bindValue(':first', $searchParam);

$stmt->bindValue(':last', $searchParam);

$stmt->bindValue(':phone', $searchParam);

$stmt->execute();

$result = $stmt->fetchAll(PDO::FETCH\_ASSOC);

// Build the PDF

$totalRentals = 0;

$totalAmount = 0.00;

$html = "<h2>Customer Rental Report</h2>";

$html .= "<p>Search Term: <strong>" . htmlspecialchars($search) . "</strong></p>";

$html .= "<table border='1' cellspacing='0' cellpadding='5' width='100%'>

<tr>

<th>Name</th>

<th>Phone</th>

<th>Rental Date</th>

<th>Rate</th>

<th>Players</th>

<th>Total Price</th>

</tr>";

foreach ($result as $row) {

$totalRentals++;

$totalAmount += $row['TotalPrice'];

$html .= "<tr>

<td>" . htmlspecialchars($row['FirstName'] . ' ' . $row['LastName']) . "</td>

<td>" . htmlspecialchars($row['PhoneNumber']) . "</td>

<td>" . htmlspecialchars($row['RentalDate']) . "</td>

<td>" . htmlspecialchars($row['RateName']) . "</td>

<td>" . htmlspecialchars($row['TotalPlayers']) . "</td>

<td>$" . number\_format($row['TotalPrice'], 2) . "</td>

</tr>";

}

$html .= "</table><br>";

$html .= "<h4>Total Rentals: $totalRentals</h4>";

$html .= "<h4>Total Amount: $" . number\_format($totalAmount, 2) . "</h4>";

$dompdf = new Dompdf();

$dompdf->loadHtml($html);

$dompdf->setPaper('A4', 'portrait');

$dompdf->render();

$dompdf->stream("CustomerRentalReport.pdf", array("Attachment" => true));

?>

ExportCustomerRentalCSV.php

<?php

$search = $\_GET['search'] ?? '';

header('Content-Type: text/csv');

header('Content-Disposition: attachment;filename=CustomerRentalReport.csv');

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

$sql = "

SELECT

c.FirstName,

c.LastName,

c.PhoneNumber,

r.RateName,

pr.RentalDate,

pr.TotalPlayers,

pr.TotalPrice

FROM poolrental pr

JOIN customer c ON pr.CustomerID = c.CustomerID

JOIN rates r ON pr.RateID = r.RateID

WHERE c.FirstName LIKE :first

OR c.LastName LIKE :last

OR c.PhoneNumber LIKE :phone

ORDER BY pr.RentalDate DESC

";

$stmt = $conn->prepare($sql);

$searchParam = '%' . $search . '%';

$stmt->bindValue(':first', $searchParam);

$stmt->bindValue(':last', $searchParam);

$stmt->bindValue(':phone', $searchParam);

$stmt->execute();

$results = $stmt->fetchAll(PDO::FETCH\_ASSOC);

$output = fopen('php://output', 'w');

fputcsv($output, ['Name', 'Phone', 'Rental Date', 'Rate', 'Total Players', 'Total Price']);

foreach ($results as $row) {

fputcsv($output, [

$row['FirstName'] . ' ' . $row['LastName'],

$row['PhoneNumber'],

$row['RentalDate'],

$row['RateName'],

$row['TotalPlayers'],

number\_format($row['TotalPrice'], 2)

]);

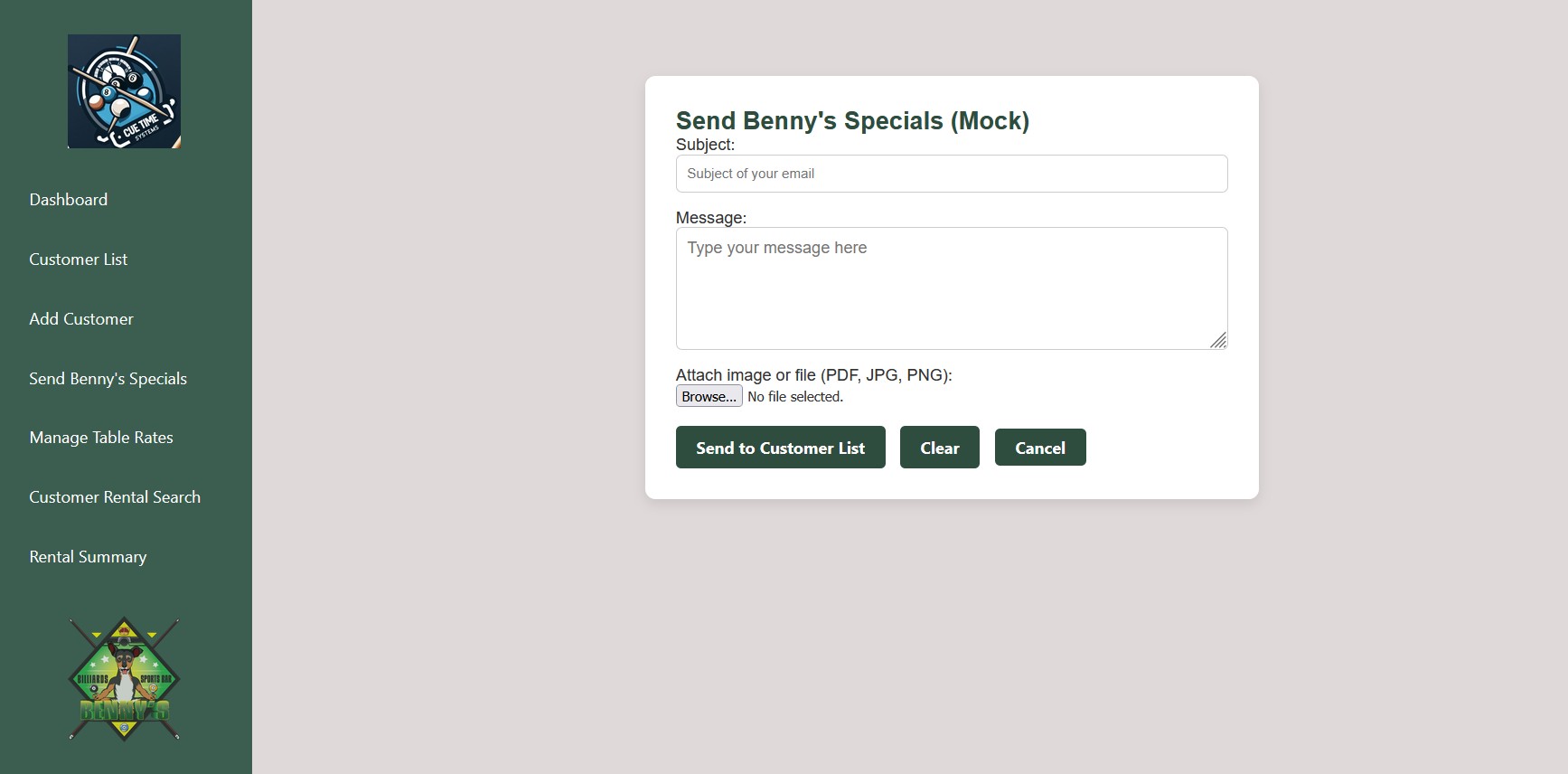
}

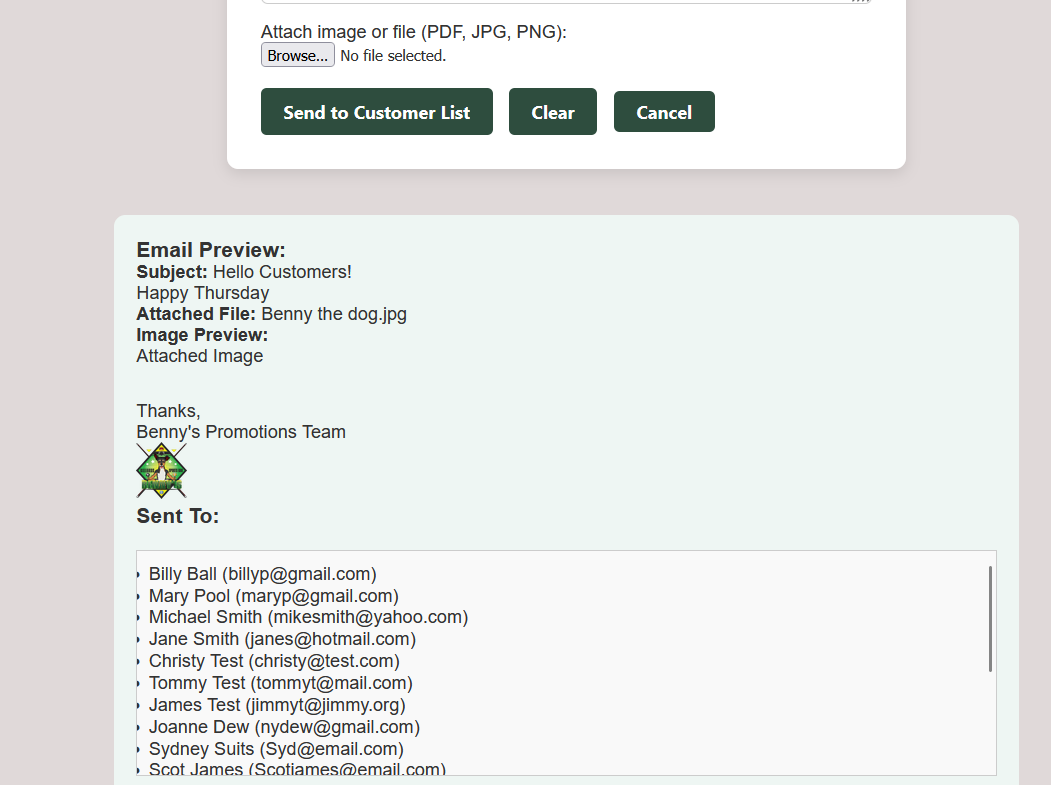
fclose($output);

exit;

?>

## 8. SendPromoEmail.php

A mock bulk email tool for simulating email campaigns. Includes fields for Subject and Message, file attachment (PDF or image), and shows preview of sent message and attached image or file.



SendPromoEmail.php

<?php

$submitted = false;

$subject = $\_POST['subject'] ?? '';

$message = $\_POST['message'] ?? '';

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && !empty($subject) && !empty($message)) {

$submitted = true;

$uploadedFileName = '';

if (!empty($\_FILES['attachment']['name'])) {

$uploadedFileName = basename($\_FILES['attachment']['name']);

$uploadedTempPath = $\_FILES['attachment']['tmp\_name'];

// Create uploads folder if it doesn't exist

$targetDir = "uploads/";

if (!is\_dir($targetDir)) {

mkdir($targetDir, 0755);

}

$targetPath = $targetDir . $uploadedFileName;

move\_uploaded\_file($uploadedTempPath, $targetPath);

}

}

function getCustomers() {

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

$stmt = $conn->prepare("SELECT FirstName, LastName, EmailAddress FROM customer WHERE EmailAddress IS NOT NULL AND EmailAddress <> ''");

$stmt->execute();

return $stmt->fetchAll(PDO::FETCH\_ASSOC);

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Mock Email to Customer List</title>

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.promo-form {

max-width: 600px;

margin: 40px auto;

background: #fff;

padding: 30px;

border-radius: 10px;

box-shadow: 0 4px 12px rgba(0,0,0,0.1);

font-family: Arial, sans-serif;

}

.promo-form h2 {

margin-top: 0;

color: #2e4d3e;

}

.promo-form input[type="text"],

.promo-form textarea {

width: 100%;

padding: 10px;

margin-bottom: 15px;

border: 1px solid #ccc;

border-radius: 6px;

font-family: Arial, sans-serif;

}

.promo-form textarea {

height: 120px;

}

.promo-form input[type="submit"],

.promo-form input[type="reset"],

.promo-form a.button {

background: #2e4d3e;

color: white;

padding: 10px 20px;

border: none;

font-weight: bold;

border-radius: 5px;

cursor: pointer;

text-decoration: none;

font-size: 1rem;

margin-right: 10px;

}

.promo-form input[type="submit"]:hover,

.promo-form input[type="reset"]:hover,

.promo-form a.button:hover {

background: #3e6a58;

}

.mock-email-preview {

background: #eef6f3;

padding: 20px;

border-radius: 10px;

margin: 30px auto;

max-width: 800px;

font-family: Arial, sans-serif;

}

.mock-email-preview h3 {

margin-top: 0;

}

.recipient-box {

max-height: 200px;

overflow-y: auto;

border: 1px solid #ccc;

padding: 10px;

background-color: #f9f9f9;

margin-top: 20px;

}

.logo-sig {

margin-top: 30px;

}

.logo-sig img {

max-height: 50px;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<div class="promo-form">

<h2>Send Benny's Specials (Mock)</h2>

<form method="post" enctype="multipart/form-data">

<label for="subject">Subject:</label>

<input type="text" id="subject" name="subject" required placeholder="Subject of your email" value="<?= htmlspecialchars($subject) ?>">

<label for="message">Message:</label>

<textarea id="message" name="message" required placeholder="Type your message here"><?= htmlspecialchars($message) ?></textarea>

<label for="attachment">Attach image or file (PDF, JPG, PNG):</label><br>

<input type="file" name="attachment" id="attachment" accept=".pdf,.jpg,.jpeg,.png"> <br><br>

<input type="submit" value="Send to Customer List" class="button">

<input type="reset" value="Clear" class="button">

<a href="index.php" class="button">Cancel</a>

</form>

</div>

<?php if ($submitted): ?>

<div class="mock-email-preview">

<h3>Email Preview:</h3>

<p><strong>Subject:</strong> <?= htmlspecialchars($subject) ?></p>

<p><?= nl2br(htmlspecialchars($message)) ?></p>

<?php if (!empty($uploadedFileName)): ?>

<p><strong>Attached File:</strong> <?= htmlspecialchars($uploadedFileName) ?></p>

<?php if (preg\_match('/\.(jpg|jpeg|png)$/i', $uploadedFileName)): ?>

<p><strong>Image Preview:</strong></p>

<img src="uploads/<?= urlencode($uploadedFileName) ?>" style="max-width: 100%; height: auto; margin-top: 10px;" alt="Attached Image">

<?php endif; ?>

<?php endif; ?>

<div class="logo-sig">

<p>Thanks,<br>Benny's Promotions Team</p>

<img src="images/BennysLogo.png" alt="Benny's Logo">

</div>

<h3>Sent To:</h3>

<div class="recipient-box">

<ul>

<?php

$customers = getCustomers();

foreach ($customers as $cust) {

echo "<li>" . htmlspecialchars($cust['FirstName'] . ' ' . $cust['LastName']) . " (" . htmlspecialchars($cust['EmailAddress']) . ")</li>";

}

?>

</ul>

</div>

</div>

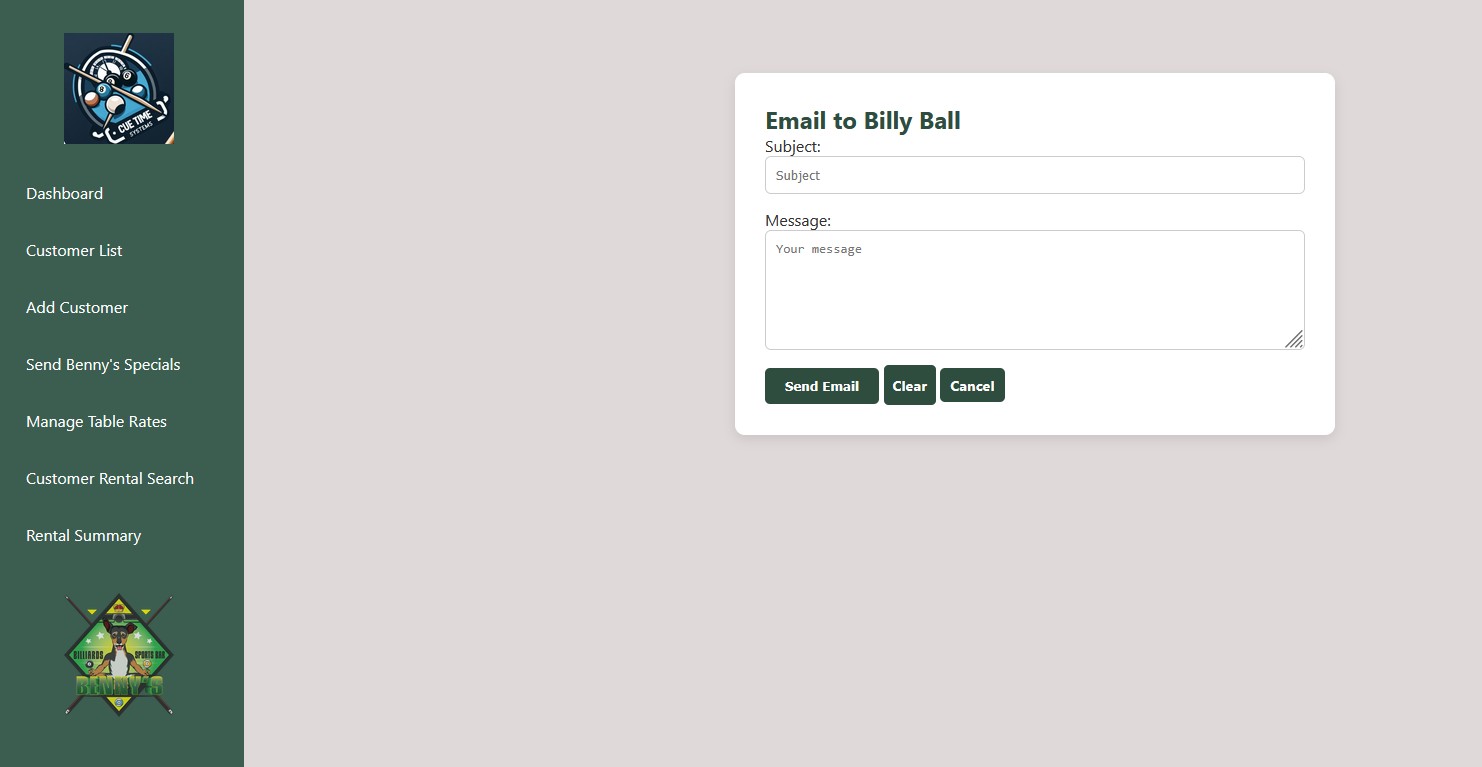
<?php endif; ?>

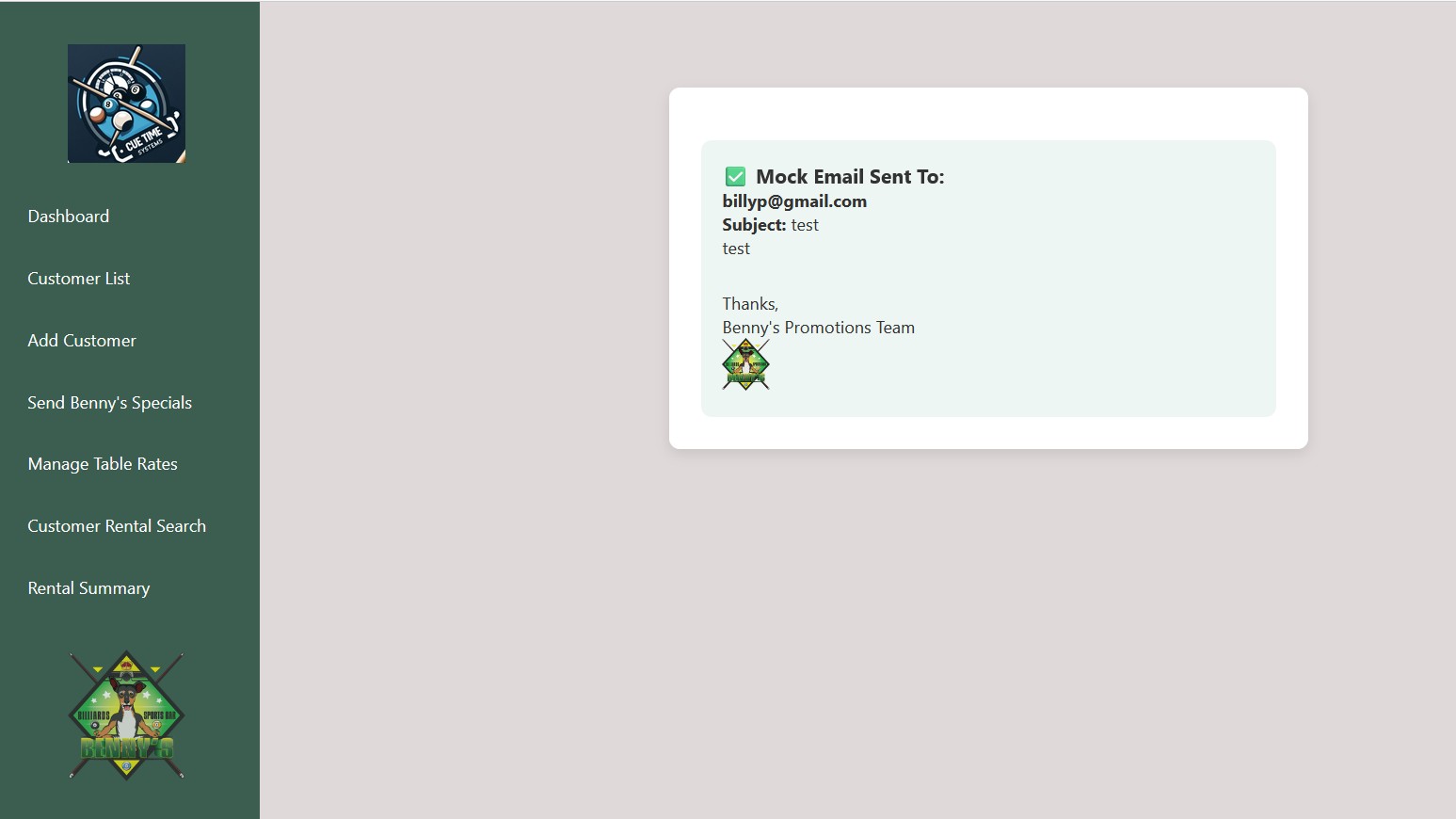
</main>

</body>

</html>

## 9. EmailSingleCustomer.php

Allows the user to send a mock email to an individual customer. Pre-fills their email address from the database. Supports subject and message customization and includes a branded signature with the Benny's logo.



EmailSingleCustomer.php

<?php

$submitted = false;

$subject = $\_POST['subject'] ?? '';

$message = $\_POST['message'] ?? '';

$customer = null;

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

if (isset($\_GET['id'])) {

$stmt = $conn->prepare("SELECT FirstName, LastName, EmailAddress FROM customer WHERE CustomerID = :id");

$stmt->bindValue(':id', $\_GET['id'], PDO::PARAM\_INT);

$stmt->execute();

$customer = $stmt->fetch(PDO::FETCH\_ASSOC);

}

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && !empty($subject) && !empty($message) && isset($\_POST['email'])) {

$submitted = true;

}

} catch (PDOException $e) {

die("Connection failed: " . $e->getMessage());

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Email Individual Customer</title>

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.email-box {

max-width: 600px;

background: #fff;

padding: 30px;

margin: 40px auto;

border-radius: 10px;

box-shadow: 0 4px 12px rgba(0,0,0,0.1);

}

.email-box h2 {

color: #2e4d3e;

}

.email-box input[type="text"],

.email-box textarea {

width: 100%;

padding: 10px;

margin-bottom: 15px;

border-radius: 6px;

border: 1px solid #ccc;

}

.email-box textarea {

height: 120px;

}

.email-box input[type="submit"] {

background: #2e4d3e;

color: white;

padding: 10px 20px;

font-weight: bold;

border: none;

border-radius: 5px;

cursor: pointer;

}

.email-box input[type="submit"]:hover {

background: #3e6a58;

}

.preview {

background: #eef6f3;

padding: 20px;

border-radius: 10px;

margin-top: 20px;

}

.logo-sig {

margin-top: 30px;

}

.logo-sig img {

max-height: 50px;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<div class="email-box">

<?php if ($customer && !$submitted): ?>

<h2>Email to <?= htmlspecialchars($customer['FirstName'] . ' ' . $customer['LastName']) ?></h2>

<form method="post">

<input type="hidden" name="email" value="<?= htmlspecialchars($customer['EmailAddress']) ?>">

<label for="subject">Subject:</label>

<input type="text" name="subject" id="subject" required placeholder="Subject" value="<?= htmlspecialchars($subject) ?>">

<label for="message">Message:</label>

<textarea name="message" id="message" required placeholder="Your message"><?= htmlspecialchars($message) ?></textarea>

<input type="submit" value="Send Email">

<input type="reset" value="Clear" class="button">

<a href="index.php" class="button">Cancel</a>

</form>

<?php elseif ($submitted): ?>

<div class="preview">

<h3>✅ Mock Email Sent To:</h3>

<p><strong><?= htmlspecialchars($\_POST['email']) ?></strong></p>

<p><strong>Subject:</strong> <?= htmlspecialchars($subject) ?></p>

<p><?= nl2br(htmlspecialchars($message)) ?></p>

<div class="logo-sig">

<p>Thanks,<br>Benny's Promotions Team</p>

<img src="images/BennysLogo.png" alt="Benny's Logo">

</div>

</div>

<?php else: ?>

<p style="color: red;">Customer not found or ID missing.</p>

<?php endif; ?>

</div>

</main>

</body>

</html>

## 10. dompdf Integration

Used for generating PDF versions of reports. Included in the Bennys\_clean/dompdf directory and referenced by report export pages.

## 11. Manage Rental Rates

This page allows the manager to simply adjust the pool rates per hour



ManageRates.php

<?php

// ManageRates.php

$server = 'tcp:mis4173.database.windows.net,1433';

$database = 'bennys';

$username = 'bennysadmin';

$password = 'Poolhall1!';

try {

$conn = new PDO("sqlsrv:Server=$server;Database=$database", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

} catch (PDOException $e) {

die("❌ Connection failed: " . $e->getMessage());

}

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

$rateID = $\_POST['rate\_id'];

$rateName = $\_POST['rate\_name'];

$rateValue = $\_POST['rate'];

$stmt = $conn->prepare("UPDATE rates SET RateName = :rateName, Rate = :rateValue WHERE RateID = :rateID");

$stmt->bindParam(':rateName', $rateName);

$stmt->bindParam(':rateValue', $rateValue);

$stmt->bindParam(':rateID', $rateID);

$stmt->execute();

}

$stmt = $conn->query("SELECT \* FROM rates ORDER BY RateID ASC");

$rates = $stmt->fetchAll(PDO::FETCH\_ASSOC);

?>

<!DOCTYPE HTML>

<html>

<head>

<title>Cue Time Systems - Manage Rates</title>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<link rel="stylesheet" href="assets/css/style.css" />

<style>

.button {

background-color: #2e4d3e;

color: white;

padding: 5px 10px;

border-radius: 5px;

font-size: 0.9em;

text-decoration: none;

}

.button:hover {

background-color: #3e6a58;

}

table {

width: 100%;

border-collapse: collapse;

margin-top: 20px;

}

th, td {

border: 1px solid #ccc;

padding: 10px;

text-align: left;

}

th {

background-color: #87aa9f;

color: #2e4d3e;

}

input[type="text"],

input[type="number"] {

width: 100%;

padding: 5px;

}

.save-btn {

padding: 5px 10px;

background-color: #2e4d3e;

color: white;

border: none;

border-radius: 5px;

cursor: pointer;

}

.save-btn:hover {

background-color: #3e6a58;

}

</style>

</head>

<body>

<?php include 'sidebar.php'; ?>

<main class="main-content">

<header class="main-header">

<h1>Manage Pool Table Rental Rates</h1>

</header>

<div class="box">

<table>

<tr>

<th>Rate ID</th>

<th>Rate Name</th>

<th>Rate ($)</th>

<th>Action</th>

</tr>

<?php foreach ($rates as $row) { ?>

<tr>

<form method="post" action="ManageRates.php">

<td><?= $row['RateID']; ?></td>

<td>

<input type="text" name="rate\_name" value="<?= htmlspecialchars($row['RateName']); ?>">

</td>

<td>

<input type="number" name="rate" step="0.01" value="<?= number\_format($row['Rate'], 2); ?>">

</td>

<td>

<input type="hidden" name="rate\_id" value="<?= $row['RateID']; ?>">

<button type="submit" name="save" class="save-btn">Save</button>

</td>

</form>

</tr>

<?php } ?>

</table>

</div>

</main>

</body>

</html>

# Conclusion

Cue Time Systems provides a comprehensive customer and rental management experience for small businesses, complete with clean design, professional reporting tools, and marketing communication features. The system is modular, extensible, and fully integrated with a cloud-hosted database.