

## CustomerTable.cshtml

```
@model IEnumerable<BankLoanProject.Models.Entities.Customer>

<link rel="stylesheet" href="~/css/admindashboard.css" />

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css" />

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></
script>

<!-- Dashboard Header -->

<div class="dashboard-header">

    <h1>

        <i class="fas fa-user-shield me-3"></i>

        Customer Management Dashboard

    </h1>

    @* <p>Manage customer registrations and KYC verification</p> *@

</div>


<!-- Table Controls -->

<div class="table-controls">

    <div class="rows-control">

        <label for="rowsPerPage">

            <i class="fas fa-list-ol"></i>

            Rows per page:
```

</label>

<select id="rowsPerPage" class="form-select" onchange="changeRowsPerPage()">

<option value="2" selected>2</option>

<option value="5">5</option>

<option value="10">10</option>

</select>

</div>

<div class="controls-right">

<!-- Bootstrap Search -->

<div class="input-group search-input-group">

<span class="input-group-text">

<i class="fas fa-search"></i>

</span>

<input type="text" class="form-control" id="searchInput" placeholder="Search customers..." oninput="searchCustomers()">

</div>

</div>

</div>

<!-- Table Container -->

<div class="table-container">

<div class="table-wrapper">

<table class="table">

<thead>

<tr>

<th> Customer ID</th>

```

        <th>Name</th>

        <th>Email</th>

        <th>Address</th>

        <th>Documents</th>

        <th>KYC Status</th>

        <th class="actions-column"> Actions</th>

    </tr>

</thead>

<tbody id="customerTableBody">

    @foreach (var customer in Model)

    {

        <tr class="customer-row" data-customer-id="@customer.CustomerId" data-
original-id="@customer.CustomerId" data-kyc-
status="@customer.KycStatus.ToLower()">

            <td><span class="customer-id">#@customer.CustomerId</span></td>

            <td><strong>@customer.Name</strong></td>

            <td>@customer.Email</td>

            <td>@customer.Address</td>

            <td>

                <a href="@customer.DocumentPath" target="_blank" class="document-
link">

                    documents/@(customer.Name.ToLower().Replace(" ", "_"))_kyc.pdf

                </a>

            </td>

            <td>

                @if (customer.KycStatus == "Approved" || customer.KycStatus ==
"Verified")

                {

                    <span class="status-badge status-verified">

```

```

        <i class="fas fa-check-circle"></i> Verified
    </span>
}
else
{
    <span class="status-badge status-pending">
        <i class="fas fa-clock"></i> @customer.KycStatus
    </span>
}
</td>

<td class="actions-cell">

    <div class="action-buttons">

        <button class="btn-action btn-edit"
onclick="openEditModal(@customer.CustomerId, '@customer.KycStatus')">
            <i class="fas fa-edit"></i>
            <span>Edit</span>
        </button>

        <form asp-action="Delete" method="post" style="display:inline;"
onsubmit="handleFormSubmit(event)">
            <input type="hidden" name="id" value="@customer.CustomerId" />
            <button type="submit" class="btn-action btn-delete"
                onclick="return confirm('Are you sure you want to delete this
customer?');">
                <i class="fas fa-trash-alt"></i>
                <span>Delete</span>
            </button>
        </form>

        <form asp-action="Verify" method="post" style="display:inline;"
onsubmit="handleFormSubmit(event)">

```

```

        <input type="hidden" name="id" value="@customer.CustomerId" />

        <button type="submit" class="btn-action btn-verify">

            <i class="fas fa-user-check"></i>

            <span>Verify</span>

        </button>

    </form>

</div>

</td>

</tr>

}

</tbody>

</table>

</div>

```

```

<!-- Pagination Section -->

<div class="pagination-section">

    <div class="pagination-info" id="paginationInfo">

        <!-- Pagination info will be generated by JavaScript -->

    </div>

    <nav aria-label="Disbursement pagination">

        <ul class="pagination" id="paginationContainer">

            <!-- Pagination buttons will be generated by JavaScript -->

        </ul>

    </nav>

</div>

</div>

```

```

<!-- Edit Modal -->

```

```
<div class="modal fade" id="editModal" tabindex="-1" aria-  
labelledby="editModalLabel" aria-hidden="true">
```

```
  <div class="modal-dialog modal-lg">
```

```
    <div class="modal-content">
```

```
      <div class="modal-header">
```

```
        <h5 class="modal-title" id="editModalLabel">
```

```
          <i class="fas fa-edit"></i> Edit Customer
```

```
        </h5>
```

```
        <button type="button" class="btn-close" data-bs-dismiss="modal" aria-  
label="Close"></button>
```

```
      </div>
```

```
      <div class="modal-body" id="editModalBody">
```

```
        <!-- Form will be loaded here -->
```

```
      </div>
```

```
    </div>
```

```
  </div>
```

```
</div>
```

```
@section Scripts {
```

```
  <script>
```

```
    // Global variables for pagination and search
```

```
    let currentPage = 1;
```

```
    let rowsPerPage = 2; // Default
```

```
    let allRows = [];
```

```
    let filteredRows = [];
```

```
    let currentScrollPosition = 0;
```

```
    let tableHeight = 0;
```

```
    // Initialize on page load
```

```
document.addEventListener('DOMContentLoaded', function() {  
    // Restore saved state first  
    restoreSavedState();  
    initializeTable();  
});  
  
function restoreSavedState() {  
    const savedRowsPerPage = sessionStorage.getItem('rowsPerPage') ||  
localStorage.getItem('rowsPerPage');  
    const savedCurrentPage = sessionStorage.getItem('currentPage') ||  
localStorage.getItem('currentPage');  
  
    if (savedRowsPerPage) {  
        rowsPerPage = parseInt(savedRowsPerPage);  
        document.getElementById('rowsPerPage').value = rowsPerPage;  
    }  
  
    if (savedCurrentPage) {  
        currentPage = parseInt(savedCurrentPage);  
    }  
}  
  
function saveCurrentState() {  
    // Save to both sessionStorage and localStorage for persistence  
    sessionStorage.setItem('rowsPerPage', rowsPerPage);  
    sessionStorage.setItem('currentPage', currentPage);  
  
    localStorage.setItem('rowsPerPage', rowsPerPage);  
    localStorage.setItem('currentPage', currentPage);  
}
```

```
}
```

```
function initializeTable() {  
    allRows = Array.from(document.querySelectorAll('.customer-row'));  
    applyFilters();  
    updateTable();  
}
```

```
function applyFilters() {  
    let rows = [...allRows];  
  
    // Apply search filter  
    const searchTerm =  
document.getElementById('searchInput').value.toLowerCase();  
    if (searchTerm !== '') {  
        rows = rows.filter(row => {  
            const text = row.textContent.toLowerCase();  
            return text.includes(searchTerm);  
        });  
    }  
  
    filteredRows = rows;  
}
```

```
function updateTable() {  
    // Save current scroll position  
    currentScrollPosition = window.pageYOffset ||  
document.documentElement.scrollTop;
```



```
// Hide all rows first

allRows.forEach(row => row.style.display = 'none');

// Calculate pagination

const totalRows = filteredRows.length;

const totalPages = Math.ceil(totalRows / rowsPerPage) || 1;

const startIndex = (currentPage - 1) * rowsPerPage;

const endIndex = Math.min(startIndex + rowsPerPage, totalRows);

// Show current page rows

for (let i = startIndex; i < endIndex; i++) {

  if (filteredRows[i]) {

    filteredRows[i].style.display = "";

  }

}

// Update pagination controls

updatePagination(totalPages, totalRows, startIndex, endIndex);

// Save current state

saveCurrentState();

// Restore scroll position

setTimeout(() => {

  window.scrollTo(0, currentScrollPosition);

}, 10);

}
```

```
function updatePagination(totalPages, totalRows, startIndex, endIndex) {

    const paginationContainer = document.getElementById('paginationContainer');
    const paginationInfo = document.getElementById('paginationInfo');

    // Clear existing pagination
    paginationContainer.innerHTML = "";

    if (totalPages <= 1) {
        // Show single page button when only one page
        const singleBtn = createBootstrapPaginationButton('1', 1, false, true);
        paginationContainer.appendChild(singleBtn);
    } else {
        // First page button
        const firstBtn = createBootstrapPaginationButton('<', 1, currentPage === 1);
        paginationContainer.appendChild(firstBtn);

        // Previous page button
        const prevBtn = createBootstrapPaginationButton('<', currentPage - 1,
currentPage === 1);
        paginationContainer.appendChild(prevBtn);

        // Page number buttons
        const maxVisiblePages = 5;
        let startPage = Math.max(1, currentPage - Math.floor(maxVisiblePages / 2));
        let endPage = Math.min(totalPages, startPage + maxVisiblePages - 1);

        if (endPage - startPage + 1 < maxVisiblePages) {
            startPage = Math.max(1, endPage - maxVisiblePages + 1);
```

```

    }

    for (let i = startPage; i <= endPage; i++) {

        const pageBtn = createBootstrapPaginationButton(i, i, false, i ===
currentPage);

        paginationContainer.appendChild(pageBtn);

    }

    // Next page button

    const nextBtn = createBootstrapPaginationButton('»', currentPage + 1,
currentPage === totalPages);

    paginationContainer.appendChild(nextBtn);

    // Last page button

    const lastBtn = createBootstrapPaginationButton('«', totalPages, currentPage
=== totalPages);

    paginationContainer.appendChild(lastBtn);

}

// Update pagination info
if (totalRows > 0) {

    paginationInfo.innerHTML = `

        <i class="fas fa-info-circle"></i>

        Showing ${startIndex + 1} to ${endIndex} of ${totalRows} entries

    `;

} else {

    paginationInfo.innerHTML = `

        <i class="fas fa-info-circle"></i>

        No matching records found

```

```

    `;
  }
}

```

```

function createBootstrapPaginationButton(text, page, disabled, active = false) {
  const li = document.createElement('li');

  li.className = `page-item ${active ? 'active' : ''} ${disabled ? 'disabled' : ''}`;

  const button = document.createElement('a');
  button.className = 'page-link';
  button.innerHTML = text;
  button.href = '#';

  if (!disabled) {
    button.onclick = (e) => {
      e.preventDefault();
      goToPage(page);
    };
  }

  li.appendChild(button);
  return li;
}

```

```

function goToPage(page) {
  currentPage = page;
  updateTable();
}

```

```
function changeRowsPerPage() {  
    rowsPerPage = parseInt(document.getElementById('rowsPerPage').value);  
    currentPage = 1;  
    updateTable();  
}
```

```
function searchCustomers() {  
    currentPage = 1;  
    applyFilters();  
    updateTable();  
}
```

```
function refreshData() {  
    const refreshBtn = document.querySelector('.btn-refresh i');  
    refreshBtn.classList.add('fa-spin');  
  
    // Clear search  
    document.getElementById('searchInput').value = "";  
  
    // Reset to original data  
    currentPage = 1;  
    applyFilters();  
  
    setTimeout(() => {  
        updateTable();  
        refreshBtn.classList.remove('fa-spin');  
    }, 800);
```

```
}
```

```
function openEditModal(id, kycStatus) {  
  $.get('/Customer/Edit/' + id, function (data) {  
    $('#editModalBody').html(data);  
  
    // Set the KYC status in the dropdown to match the customer's current status  
    setTimeout(() => {  
      const kycSelect = document.querySelector('#editForm  
select[name="KycStatus"]');  
      if (kycSelect) {  
        // Normalize the status - if it's "Approved", set it to "Verified"  
        const normalizedStatus = (kycStatus === "Approved") ? "Verified" :  
kycStatus;  
        kycSelect.value = normalizedStatus;  
      }  
    }, 100);  
  
    $('#editModal').modal('show');  
  });  
}
```

```
// Handle form submissions to prevent page scroll and maintain state  
function handleFormSubmit(event) {  
  // Save current scroll position and state before form submission  
  currentScrollPosition = window.pageYOffset ||  
document.documentElement.scrollTop;  
  saveCurrentState();  
  sessionStorage.setItem('scrollPosition', currentScrollPosition);
```

```

    }

    // Restore state after page reload
    window.addEventListener('load', function() {
        const savedScrollPosition = sessionStorage.getItem('scrollPosition');

        if (savedScrollPosition) {
            setTimeout(() => {
                window.scrollTo(0, parseInt(savedScrollPosition));

                // Clear saved scroll position
                sessionStorage.removeItem('scrollPosition');
            }, 100);
        }
    });
</script>
}

```

### **admindashboard.css**

```

/* Admin Dashboard Styling */
:root {
    --primary: #2c3e50;
    --secondary: #3498db;
    --accent: #16a085;
    --light: #f8f9fa;
    --dark: #343a40;
    --success: #27ae60;
    --warning: #f39c12;
}

```

```
--danger: #e74c3c;

--info: #1abc9c;

--table-header-bg: linear-gradient(to right, #2c3e50, #3498db);

--pagination-border: #495057;

--pagination-bg: #f8f9fa;

--pagination-active-bg: #3498db;

}
```

```
body{

    background-color: #f5f7fa;

    font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;

    padding: 20px 0;

    margin: 0;

}
```

```
/* Header Container */
```

```
.dashboard-header {

    background: linear-gradient(135deg, var(--primary), var(--secondary));

    color: white;

    border-radius: 15px;

    padding: 10px 25px;

    margin: 70px auto 30px auto;

    box-shadow: 0 5px 15px rgba(0, 0, 0, 0.1);

    text-align: center;

    max-width: 92%;

}
```

```
.dashboard-header h1 {
```



```
margin: 0;
font-size: 1.6rem;
font-weight: 700;
text-shadow: 0 2px 4px rgba(0, 0, 0, 0.3);
display: flex;
align-items: center;
justify-content: center;
gap: 15px;
}
```

```
.dashboard-header h2 {
margin: 0;
font-size: 1.5rem;
font-weight: 700;
text-shadow: 0 2px 4px rgba(0, 0, 0, 0.3);
display: flex;
align-items: center;
justify-content: center;
gap: 15px;
}
```

```
.dashboard-header p {
margin: 8px 0 0 0;
opacity: 0.9;
font-size: 1rem;
}
```

```
/* Table Controls */
```

```
.table-controls {  
  background-color: white;  
  border-radius: 12px;  
  padding: 20px;  
  box-shadow: 0 5px 15px rgba(0, 0, 0, 0.05);  
  margin: 20px auto;  
  max-width: 96%;  
  display: flex;  
  justify-content: space-between;  
  align-items: center;  
  flex-wrap: wrap;  
  gap: 15px;  
  min-height: 70px;  
}
```

```
.rows-control {  
  display: flex;  
  align-items: center;  
  gap: 10px;  
}
```

```
.rows-control label {  
  color: var(--primary);  
  font-weight: 600;  
  font-size: 0.95rem;  
  display: flex;  
  align-items: center;  
  gap: 8px;
```

```
    white-space: nowrap;
}
```

```
.rows-control.form-select {
    min-width: 80px;
    border: 2px solid #e0e0e0;
    border-radius: 6px;
    font-size: 0.9rem;
}
```

```
.rows-control.form-select:focus {
    border-color: var(--secondary);
    box-shadow: 0 0 0 0.25rem rgba(52, 152, 219, 0.25);
}
```

/\* Controls Right Side \*/

```
.controls-right {
    display: flex;
    align-items: center;
    gap: 15px;
}
```

/\* Bootstrap Search Input Group \*/

```
.search-input-group {
    width: 300px;
}
```

```
.search-input-group.input-group-text {
```

```
background-color: var(--secondary);  
color: white;  
border: 2px solid var(--secondary);  
}
```

```
.search-input-group.form-control {  
border: 2px solid #e0e0e0;  
border-left: none;  
}
```

```
.search-input-group.form-control:focus {  
border-color: var(--secondary);  
box-shadow: 0 0 0 0.25rem rgba(52, 152, 219, 0.25);  
}
```

/\* Table Container \*/

```
.table-container {  
background-color: white;  
border-radius: 12px;  
padding: 0;  
box-shadow: 0 5px 15px rgba(0, 0, 0, 0.05);  
margin: 20px auto;  
max-width: 96%;  
overflow: hidden;  
border: 4px solid var(--primary);  
min-height: auto;  
display: flex;  
flex-direction: column;
```

```
}
```

```
/* Table wrapper to ensure consistent height */
```

```
.table-wrapper {  
    flex: 1;  
    min-height: auto;  
    display: flex;  
    flex-direction: column;  
}
```

```
/* Table Styling */
```

```
.table {  
    margin-bottom: 0;  
    border-collapse: separate;  
    border-spacing: 0;  
    width: 100%;  
    flex: 1;  
}
```

```
/* Table Header */
```

```
.table thead {  
    background: var(--table-header-bg);  
}
```

```
.table thead th {  
    background: var(--table-header-bg);  
    color: white;  
    font-weight: 600;
```

```
border: 1px solid rgba(255, 255, 255, 0.2);
border-bottom: 2px solid rgba(255, 255, 255, 0.3);
padding: 8px 6px;
text-transform: uppercase;
font-size: 0.7rem;
letter-spacing: 1px;
vertical-align: middle;
position: relative;
text-align: center; /* Changed from left to center */
white-space: nowrap; /* Prevent header text wrapping */
}
```

```
.table thead th:first-child {
    border-left: none;
}
```

```
.table thead th:last-child {
    border-right: none;
}
```

```
.table thead th i {
    margin-right: 8px;
    font-size: 1rem;
}
```

```
.table thead th.actions-column {
    width: 220px;
    min-width: 220px;
```

```
}
```

```
.table tbody td {  
    padding: 8px 6px;  
    border: 1px solid #e0e0e0;  
    border-top: none;  
    vertical-align: middle;  
    word-break: break-word; /* Allow text to wrap within cells */  
    font-size: 0.8rem;  
    text-align: center; /* Added to center table body content */  
}
```

```
.table tbody td:first-child {  
    border-left: none;  
}
```

```
.table tbody td:last-child {  
    border-right: none;  
}
```

```
.table tbody tr:hover {  
    background-color: rgba(52, 152, 219, 0.05);  
    transition: background-color 0.2s ease;  
}
```

```
.table tbody tr:last-child td {  
    border-bottom: none;  
}
```

```
/* Customer ID Styling */
```

```
.customer-id {  
  font-weight: 700;  
  color: var(--primary);  
  font-size: 1rem;  
}
```

```
/* Status Badge */
```

```
.status-badge {  
  padding: 6px 12px;  
  border-radius: 20px;  
  font-weight: 500;  
  display: inline-flex;  
  align-items: center;  
  gap: 5px;  
  font-size: 0.8rem;  
  text-align: center;  
  min-width: 90px;  
  justify-content: center;  
}
```

```
.status-verified {  
  background-color: rgba(39, 174, 96, 0.2);  
  color: var(--success);  
  border: 1px solid var(--success);  
}
```



```
.status-pending {  
  background-color: rgba(243, 156, 18, 0.2);  
  color: var(--warning);  
  border: 1px solid var(--warning);  
}
```

```
.status-approved {  
  background-color: rgba(39, 174, 96, 0.2);  
  color: var(--success);  
  border: 1px solid var(--success);  
}
```

/\* Document Link \*/

```
.document-link {  
  color: var(--secondary);  
  text-decoration: none;  
  font-weight: 500;  
  transition: all 0.3s ease;  
}
```

```
.document-link:hover {  
  color: var(--primary);  
  text-decoration: underline;  
}
```

/\* Action Buttons - Modern Card Style \*/

```
.actions-cell {  
  width: 220px;
```

```
    min-width: 220px;
    padding: 12px 8px;
}
```

```
.action-buttons {
    display: flex;
    gap: 4px;
    flex-wrap: nowrap;
    justify-content: center;
    align-items: center;
}
```

```
.btn-action {
    border-radius: 8px;
    padding: 4px 6px;
    font-size: 0.6rem;
    font-weight: 600;
    transition: all 0.3s;
    border: none;
    display: flex;
    flex-direction: column;
    align-items: center;
    justify-content: center;
    min-width: 35px;
    height: 35px;
    cursor: pointer;
    box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
    gap: 2px;
}
```

```
}
```

```
.btn-action i {  
    font-size: 0.7rem;  
    margin-bottom: 2px;  
}
```

```
.btn-action span {  
    font-size: 0.55rem;  
    font-weight: 600;  
    text-transform: uppercase;  
    letter-spacing: 0.5px;  
}
```

```
.btn-edit {  
    background-color: #ff9500;  
    color: white;  
}
```

```
.btn-edit:hover {  
    background-color: #e6850e;  
    color: white;  
    transform: translateY(-2px);  
    box-shadow: 0 4px 8px rgba(255, 149, 0, 0.3);  
}
```

```
.btn-delete {  
    background-color: #ff4757;
```

```
color: white;
}
```

```
.btn-delete:hover {
    background-color: #ff3742;
    color: white;
    transform: translateY(-2px);
    box-shadow: 0 4px 8px rgba(255, 71, 87, 0.3);
}
```

```
.btn-verify {
    background-color: #2ed573;
    color: white;
}
```

```
.btn-verify:hover {
    background-color: #26d065;
    color: white;
    transform: translateY(-2px);
    box-shadow: 0 4px 8px rgba(46, 213, 115, 0.3);
}
```

```
/* Pagination section - Minimal Gap */
```

```
.pagination-section {
    background-color: #f8f9fa;
    border-top: 1px solid #e0e0e0;
    padding: 8px 15px;
    margin-top: 0;
```

```
}
```

```
/* Enhanced Bootstrap Pagination Styling */
```

```
.pagination {  
    margin: 4px 0 0 0;  
    gap: 6px;  
    justify-content: center;  
}
```

```
/* Pagination Info */
```

```
.pagination-info {  
    color: var(--primary);  
    font-weight: 500;  
    font-size: 0.85rem;  
    display: flex;  
    align-items: center;  
    justify-content: center;  
    gap: 6px;  
    margin-bottom: 4px;  
}
```

```
.pagination .page-item .page-link {  
    color: #2c3e50;  
    background-color: var(--pagination-bg);  
    border: 3px solid #2c3e50;  
    padding: 8px 12px;  
    font-weight: 700;  
    font-size: 0.9rem;
```

```
min-width: 40px;
min-height: 40px;
text-align: center;
display: flex;
align-items: center;
justify-content: center;
border-radius: 6px;
transition: all 0.3s ease;
text-decoration: none;
box-shadow: 0 1px 3px rgba(0, 0, 0, 0.1);
}
```

```
.pagination .page-item.disabled .page-link {
  opacity: 0.5;
  cursor: not-allowed;
  background-color: #e9ecef;
  border-color: #6c757d;
  color: #495057;
  transform: none;
  box-shadow: 0 1px 2px rgba(0, 0, 0, 0.05);
}
```

```
.pagination .page-item.disabled .page-link:hover {
  background-color: #e9ecef;
  border-color: #6c757d;
  color: #495057;
  transform: none;
  box-shadow: 0 1px 2px rgba(0, 0, 0, 0.05);
}
```

```
}
```

```
.pagination .page-item.active .page-link {  
  background-color: var(--pagination-active-bg);  
  color: white;  
  border-color: var(--secondary);  
  transform: translateY(-2px);  
  box-shadow: 0 3px 6px rgba(52, 152, 219, 0.4);  
  font-weight: 800;  
}
```

```
.pagination .page-item.active .page-link:hover {  
  background-color: var(--secondary);  
  color: white;  
  border-color: var(--primary);  
}
```

```
/* Modal Styling */
```

```
.modal-content {  
  border-radius: 15px;  
  border: none;  
  box-shadow: 0 10px 30px rgba(0, 0, 0, 0.1);  
  overflow: hidden;  
}
```

```
.modal-header {  
  background: linear-gradient(to right, var(--primary), var(--secondary));  
  color: white;
```

```
border-bottom: none;
padding: 20px 25px;
}
```

```
.modal-title {
  font-weight: 600;
  letter-spacing: 0.5px;
  font-size: 1.3rem;
  display: flex;
  align-items: center;
  gap: 10px;
}
```

```
.modal-body {
  padding: 25px;
  background-color: #fafafa;
}
```

```
/* Form Styling */
```

```
.form-group {
  margin-bottom: 20px;
  position: relative;
}
```

```
.form-group label {
  font-weight: 600;
  color: var(--primary);
  margin-bottom: 8px;
```



```
display: flex;
align-items: center;
gap: 8px;
font-size: 0.95rem;
}
```

```
.form-control {
border: 2px solid #e0e0e0;
border-radius: 8px;
padding: 12px 15px;
transition: all 0.3s;
font-size: 0.95rem;
width: 100%;
box-sizing: border-box;
}
```

```
.form-control:focus {
border-color: var(--secondary);
box-shadow: 0 0 0 0.25rem rgba(52, 152, 219, 0.25);
outline: none;
}
```

*/\* Form Buttons \*/*

```
.btn-primary {
background: linear-gradient(135deg, var(--secondary), var(--primary));
color: white;
border: none;
border-radius: 8px;
```

```
padding: 12px 25px;
font-weight: 600;
transition: all 0.3s;
cursor: pointer;
display: flex;
align-items: center;
gap: 8px;
}
```

```
.btn-primary:hover {
  background: linear-gradient(135deg, var(--primary), var(--secondary));
  color: white;
  transform: translateY(-2px);
  box-shadow: 0 5px 15px rgba(52, 152, 219, 0.3);
}
```

```
.btn-secondary {
  background-color: #6c757d;
  color: white;
  border: none;
  border-radius: 8px;
  padding: 12px 25px;
  font-weight: 600;
  transition: all 0.3s;
  cursor: pointer;
  display: flex;
  align-items: center;
  gap: 8px;
}
```

```
}
```

```
.btn-secondary:hover {  
    background-color: #5a6268;  
    color: white;  
    transform: translateY(-2px);  
}
```

```
/* Close button styling */
```

```
.btn-close {  
    background: none;  
    border: none;  
    font-size: 1.5rem;  
    color: white;  
    opacity: 0.8;  
}
```

```
.btn-close:hover {  
    opacity: 1;  
}
```

```
/* Row styling */
```

```
.row {  
    display: flex;  
    margin: 0 -10px;  
    flex-wrap: wrap;  
}
```

```
.col-md-6 {  
    flex: 0 0 50%;  
    max-width: 50%;  
    padding: 0 10px;  
    box-sizing: border-box;  
}
```

```
/* Responsive Design */
```

```
@media (max-width: 768px) {  
    .table-controls {  
        flex-direction: column;  
        align-items: stretch;  
        gap: 15px;  
        min-height: 140px;  
    }  
}
```

```
.controls-right {  
    flex-direction: column;  
    gap: 10px;  
}
```

```
.search-input-group {  
    width: 100%;  
}
```

```
.table-container {  
    overflow-x: auto;  
    max-width: 98%;  
}
```

```
}
```

```
.table {  
  min-width: 900px;  
}
```

```
.dashboard-header {  
  margin: 15px auto;  
  padding: 20px;  
  max-width: 98%;  
}
```

```
.dashboard-header h1 {  
  font-size: 1.5rem;  
}
```

```
.action-buttons {  
  flex-direction: column;  
  gap: 2px;  
}
```

```
.btn-action {  
  font-size: 0.55rem;  
  padding: 2px 4px;  
  min-width: 30px;  
  height: 30px;  
}
```

```
.col-md-6 {  
  flex: 0 0 100%;  
  max-width: 100%;  
}
```

```
.refresh-container {  
  max-width: 98%;  
}
```

```
.pagination {  
  gap: 4px;  
}
```

```
.pagination.page-item.page-link {  
  padding: 6px 10px;  
  min-width: 35px;  
  min-height: 35px;  
  font-size: 0.8rem;  
}  
}
```

```
@media (max-width: 576px) {  
  .dashboard-header h1 {  
    font-size: 1.3rem;  
    flex-direction: column;  
    gap: 10px;  
  }  
}
```

```
.dashboard-header p {  
    font-size: 1rem;  
}
```

```
.table thead th {  
    padding: 4px 2px;  
    font-size: 0.6rem;  
}
```

```
.table tbody td {  
    padding: 4px 2px;  
    font-size: 0.7rem;  
}
```

```
.actions-cell {  
    width: 120px;  
    min-width: 120px;  
}
```

```
.btn-action {  
    font-size: 0.5rem;  
    padding: 2px 3px;  
    min-width: 25px;  
    height: 25px;  
}
```

```
.pagination.page-item.page-link {  
    padding: 4px 8px;
```

```
    min-width: 30px;
    min-height: 30px;
    font-size: 0.75rem;
  }
}
```

```
/* Loading states */
```

```
.loading{
  opacity: 0.6;
  pointer-events: none;
}
```

```
/* Smooth transitions */
```

```
*{
  box-sizing: border-box;
}
```

```
/* Re-enabled smooth transitions */
```

```
.table-container,
.table-controls{
  transition: all 0.3s ease;
}
```

```
/* Column width adjustments to prevent overflow */
```

```
.table th:nth-child(1){
  width: 8%;
}
```



```
/* Customer ID */  
.table th:nth-child(2) {  
    width: 15%;  
}
```

```
/* Name */  
.table th:nth-child(3) {  
    width: 15%;  
}
```

```
/* Email */  
.table th:nth-child(4) {  
    width: 17%;  
}
```

```
/* Address */  
.table th:nth-child(5) {  
    width: 15%;  
}
```

```
/* Documents */  
.table th:nth-child(6) {  
    width: 10%;  
}
```

```
/* KYC Status */  
.table th:nth-child(7) {  
    width: 20%;  
}
```

```
}
```

```
/* Actions */
```

```
/* Ensure table cells don't overflow */
```

```
.table td {  
    max-width: 0;  
    overflow: hidden;  
    text-overflow: ellipsis;  
}
```

```
.pagination.page-item.page-link {  
    color: #2c3e50; /* Darker text color */  
    background-color: var(--pagination-bg);  
    border: 3px solid #2c3e50; /* Darker border */  
    padding: 8px 12px;  
    font-weight: 700;  
    font-size: 0.9rem;  
    min-width: 40px;  
    min-height: 40px;  
    text-align: center;  
    display: flex;  
    align-items: center;  
    justify-content: center;  
    border-radius: 8px;  
    transition: all 0.3s ease;  
    text-decoration: none;  
    box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);  
}
```

```
.pagination.page-item.disabled.page-link {  
    opacity: 0.5;  
    cursor: not-allowed;  
    background-color: #e9ecef;  
    border-color: #6c757d; /* Darker border for disabled state */  
    color: #495057; /* Darker text for disabled state */  
    transform: none;  
    box-shadow: 0 1px 2px rgba(0, 0, 0, 0.05);  
}
```

```
.pagination.page-item.disabled.page-link:hover {  
    background-color: #e9ecef;  
    border-color: #6c757d; /* Darker border for disabled hover state */  
    color: #495057; /* Darker text for disabled hover state */  
    transform: none;  
    box-shadow: 0 1px 2px rgba(0, 0, 0, 0.05);  
}
```

/\* Add this CSS rule after the existing pagination styles to highlight the active page button \*/

```
.pagination.page-item.active.page-link {  
    background-color: var(--pagination-active-bg);  
    color: white;  
    border-color: var(--secondary);  
    transform: translateY(-2px);  
    box-shadow: 0 4px 8px rgba(52, 152, 219, 0.4);  
    font-weight: 800;
```

```
}
```

```
.pagination.page-item.active.page-link:hover {  
  background-color: var(--secondary);  
  color: white;  
  border-color: var(--primary);  
}
```