@model IEnumerable<dsBusTravelTicketSystem.Models.Bus>

@{

ViewData["Title"] = "Bus Management";

}

<div class="control-section">

<ejs-grid id="Grid" dataSource="@Model" allowSorting="true" allowFiltering="true" allowPdfExport="true" allowExcelExport="true" allowGrouping="true"

actionComplete="actionComplete" actionBegin="actionBegin" actionFailure="actionFailure"

toolbarClick="toolbarClick" toolbar="@(new List<string>() { "Add", "Edit", "Delete", "ExcelExport", "PdfExport", "Search" })"

allowPaging="true">

<e-grid-editSettings allowAdding="true" allowDeleting="true" allowEditing="true" mode="Dialog" template='#dialogtemplate' showDeleteConfirmDialog="true"></e-grid-editSettings>

<e-grid-pagesettings pageCount="5"></e-grid-pagesettings>

<e-grid-filterSettings type="Excel"></e-grid-filterSettings>

<e-grid-columns>

<e-grid-column field="BusId" headerText="Bus ID" isPrimaryKey="true" validationRules="@(new { required = true })" textAlign="Right" width="120"></e-grid-column>

<e-grid-column field="BusNumber" headerText="Bus Number" validationRules="@(new { required = true })" width="150"></e-grid-column>

<e-grid-column field="Capacity" headerText="Capacity" width="150"></e-grid-column>

<e-grid-column field="BusTypeId" headerText="Bus Type ID" width="150"></e-grid-column>

<e-grid-column field="BusRegistrationValidityDate" headerText="Registration Validity Date" width="150" format="yMd"></e-grid-column>

<e-grid-column field="BusOdometer" headerText="Odometer" width="150"></e-grid-column>

<e-grid-column field="BusInsuranceValidityDate" headerText="Insurance Validity Date" width="150" format="yMd"></e-grid-column>

<e-grid-column field="CreatedAt" headerText="Created At" width="150" format="yMd"></e-grid-column>

<e-grid-column field="CreatedByUserId" headerText="Created By User ID" width="150"></e-grid-column>

<e-grid-column field="LastModifiedByUserId" headerText="Last Modified By User ID" width="150"></e-grid-column>

<e-grid-column field="LastModifiedAt" headerText="Last Modified At" width="150" format="yMd"></e-grid-column>

</e-grid-columns>

</ejs-grid>

</div>

<script id='dialogtemplate' type="text/x-template">

<div id="dialogTemp"></div>

</script>

<script>

function actionBegin(args) {

// Validate edit inputs

if (args.requestType === 'save') {

if (!args.data.BusNumber || !args.data.Capacity) {

args.cancel = true; // Prevent save if invalid

alert('Invalid input: Please fill in the required fields.');

}

}

}

function actionFailure(args) {

// If the action fails, re-open the dialog and display a message

if (args.requestType === 'save') {

alert('Edit failed: Please correct the errors and try again.');

args.cancel = true; // Keep the dialog open

}

}

function actionComplete(args) {

if (args.requestType === 'beginEdit' || args.requestType === 'add') {

let spinner = ej.popups.createSpinner({ target: args.dialog.element });

ej.popups.showSpinner(args.dialog.element);

var url = "@Url.Action("DetailsPartial", "Bus")";

var ajax = new ej.base.Ajax({

url: url,

type: "POST",

contentType: "application/json",

data: JSON.stringify({ value: args.rowData })

});

ajax.send().then(function (data) {

appendElement(data, args.form);

args.form.elements.namedItem('BusNumber').focus();

if (args.dialog && typeof args.dialog.setTitle === 'function') {

let title = args.requestType === 'beginEdit' ? `Details of Bus ${args.rowData.BusId} - ${args.rowData.BusNumber}` : "Add New Bus";

args.dialog.setTitle(title);

}

ej.popups.hideSpinner(args.dialog.element);

}).catch(function (xhr) {

console.log(xhr);

ej.popups.hideSpinner(args.dialog.element);

});

} else if (args.requestType === 'save') {

alert("Record saved successfully.");

var url = "@Url.Action("Edit", "Bus")"

var ajax = new ej.base.Ajax({

url: url,

type: "POST",

data: JSON.stringify(args.data),

contentType: "application/json"

});

ajax.send().then(function (response) {

var result = JSON.parse(response);

if (result.success) {

alert(result.message);

args.grid.refresh();

} else {

alert('Save failed: ' + result.message);

}

}).catch(function (xhr) {

console.log(xhr);

alert('An error occurred while saving the record.');

args.grid.refresh();

});

} else if (args.requestType === 'delete') {

if (confirm(`Do you want to delete the selected bus with ID ${args.data[0].BusId} and number ${args.data[0].BusNumber}?`)) {

var url = "@Url.Action("DeleteConfirmed", "Bus")";

var ajax = new ej.base.Ajax({

url: url,

type: "POST",

data: JSON.stringify(args.data[0].BusId),

contentType: "application/json"

});

ajax.send().then(function (response) {

var result = JSON.parse(response);

if (result.success) {

alert(result.message);

} else {

alert('Deletion failed: ' + result.message);

args.grid.refresh();

}

}).catch(function (xhr) {

console.log(xhr);

alert('An error occurred while deleting the record.');

args.grid.refresh();

});

} else {

args.cancel = true;

}

}

}

function appendElement(elementString, form) {

form.querySelector("#dialogTemp").innerHTML = elementString;

var script = document.createElement('script');

script.type = "text/javascript";

var serverScript = form.querySelector("#dialogTemp").querySelector('script');

script.textContent = serverScript.innerHTML;

document.head.appendChild(script);

serverScript.remove();

}

function toolbarClick(args) {

var gridObj = document.getElementById("Grid").ej2\_instances[0];

if (args.item.id === 'Grid\_pdfexport') {

gridObj.pdfExport();

} else if (args.item.id === 'Grid\_excelexport') {

gridObj.excelExport();

}

}

</script>

<style>

.form-group.col-md-6 {

width: 250px;

}

:-ms-fullscreen, .e-dialog {

max-width: 552px;

}

@@media only screen and (max-width: 991px) {

.e-dialog.e-control.e-popup {

min-width: 300px !important;

}

}

@@media only screen and (min-width: 992px) {

.e-edit-dialog.e-dialog.e-control.e-popup {

min-width: 550px !important;

min-height: 415px;

}

.e-bigger.e-edit-dialog.e-dialog.e-control.e-popup,

.e-bigger .e-edit-dialog.e-dialog.e-control.e-popup {

min-height: 496px;

}

}

</style>