Register view

@model ClassLibrary.WebModels.RegisterModel

@{

ViewBag.Title = "register";

Layout = "~/Views/Shared/\_AccountLayout.cshtml";

}

<section>

<div class="container">

<div class="row">

<div class="col-md-12">

<div class="main-title">

<img src="~/Content/images/icons/reg-icon.png" />

<span>Registration</span>

</div><!-- /main-title -->

</div><!-- /col-md-12 -->

</div><!-- /row -->

<div class="row">

@using (Ajax.BeginForm("register", "Account", null, new AjaxOptions

{

HttpMethod = "POST",

OnSuccess = "registerSuccess",

OnFailure = "ajaxFailure",

}, new { id = "accnt-register" }))

{

@Html.HiddenFor(m => m.UserLat, new { Value = 0 })

@Html.HiddenFor(m => m.UserLong, new { Value = 0 })

@Html.HiddenFor(m => m.PasswordMatch, new { Value = false })

@Html.HiddenFor(m => m.ValidEmail, new { Value = false })

@Html.HiddenFor(m => m.ProfilePic, new { Value = "" })

<div class="col-md-2 col-md-offset-1">

<div class="upload-photo">

<img id="profileImage" src="~/Content/images/icons/profile-icon.png" width="130" height="130" />

<input id="imageUpload" type="file" name="profile\_photo" placeholder="Photo" required capture>

<p>Click to Pick Image</p>

</div>

</div>

<div class="col-md-4 col-sm-6">

<div class="form-group">

@\*@Html.DropDownListFor(m => m.UserRole,

new SelectList(Enum.GetValues(typeof(ClassLibrary.Enum.UserRole)), "Key",

"Value"), new { @class = "form-control" })\*@

@Html.DropDownListFor(m => m.UserRole,

new List<SelectListItem> {

new SelectListItem { Value = "1" , Text = "User" },

new SelectListItem { Value = "2" , Text = "Cleaner" },

new SelectListItem { Value = "3" , Text = "Both" }

},

new { @class = "form-control" })

@\*<select class="form-control">

</select>\*@

</div>

<div class="form-group">

@\*@Html.DropDownListFor(m => m.ProfileType,

new SelectList(Enum.GetValues(typeof(ClassLibrary.Enum.ProfileType))), new { @class = "form-control" })\*@

@Html.DropDownListFor(m => m.ProfileType,

new List<SelectListItem> {

new SelectListItem { Value = "1" , Text = "Company" },

new SelectListItem { Value = "2" , Text = "Individual" }

},

new { @class = "form-control" })

@\*<select class="form-control">

</select>\*@

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Company Name" />\*@

@Html.TextBoxFor(m => m.CompanyName, new { autocomplete = "off", placeholder = "Company Name", @class = "form-control" })

<span id="spn-CompanyName" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="First Name" />\*@

@Html.TextBoxFor(m => m.FirstName, new { autocomplete = "off", placeholder = "First Name", @class = "form-control" })

<span id="spn-FirstName" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Last Name" />\*@

@Html.TextBoxFor(m => m.LastName, new { autocomplete = "off", placeholder = "Last Name", @class = "form-control" })

<span id="spn-LastName" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Email" />\*@

@Html.TextBoxFor(m => m.Email, new { autocomplete = "off", placeholder = "Email", @class = "form-control" })

<span id="spn-Email" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Password" />\*@

@Html.PasswordFor(m => m.Password, new { autocomplete = "off", placeholder = "Password", @class = "form-control" })

<span id="spn-Password" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Confirm Password" />\*@

@Html.PasswordFor(m => m.ConfirmPassword, new { autocomplete = "off", placeholder = "Confirm Password", @class = "form-control" })

<span id="spn-ConfirmPassword" style="color:red;display:none;"></span>

</div>

</div><!-- /col-md-4 -->

<div class="col-md-4 col-sm-6">

<div class="form-group">

@Html.DropDownListFor(m => m.Gender,

new SelectList(Enum.GetValues(typeof(ClassLibrary.Enum.Gender))), new { @class = "form-control" })

@\*<select class="form-control">

</select>\*@

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Phone Number" />\*@

@Html.TextBoxFor(m => m.Phone, new { autocomplete = "off", placeholder = "Phone Number", @class = "form-control" })

<span id="spn-Phone" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Address" />\*@

@Html.TextBoxFor(m => m.Address, new { autocomplete = "off", placeholder = "Address", @class = "form-control" })

<span id="spn-Address" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Suburb" />\*@

@Html.TextBoxFor(m => m.Suburb, new { autocomplete = "off", placeholder = "Suburb", @class = "form-control" })

<span id="spn-Suburb" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="State" />\*@

@Html.TextBoxFor(m => m.State, new { autocomplete = "off", placeholder = "State", @class = "form-control" })

<span id="spn-State" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Country" />\*@

@Html.TextBoxFor(m => m.Country, new { autocomplete = "off", placeholder = "Country", @class = "form-control" })

<span id="spn-Country" style="color:red;display:none;"></span>

</div>

<div class="form-group">

@\*<input type="text" class="form-control" placeholder="Postal code / ZIP" />\*@

@Html.TextBoxFor(m => m.Zip, new { autocomplete = "off", placeholder = "Postal code / ZIP", @class = "form-control" })

<span id="spn-Zip" style="color:red;display:none;"></span>

</div>

<div class="form-group">

<div class="terms-cond-txt">

<div class="terms-conditions">

<input type="checkbox" class="hidden-checkbox" id="accnt-checkbox-register-terms" />

<span class="checkbox-css" id="foo"></span>

<span>By registering you agree to the Terms & Conditions</span>

</div>

</div>

</div>

</div><!-- /col-md-4 -->

<div class="col-md-4 col-md-offset-4">

<div class="form-group">

<input type="submit" style="display:none;" id="btn-register-submit" value="Register" class="login-blck-btn" />

<input type="button" disabled value="Register" style="background: #d6d6d6;" id="btn-dummy-register-submit" class="login-blck-btn" />

<br />

</div>

</div>

}

</div><!-- /row -->

</div><!-- /container -->

</section>

<br><br><br>

<script src="https://maps.googleapis.com/maps/api/js?libraries=places&key=AIzaSyD57-Hx6gnYLUHPqPVd9-\_Bg2-Z25Dkzp8"></script>

<!-- Profile image upload -->

<script>

$("#profileImage").click(function (e) {

$("#imageUpload").click();

});

function previewProfileImage(uploader) {

//ensure a file was selected

if (uploader.files && uploader.files[0]) {

var imageFile = uploader.files[0];

var fileType = imageFile["type"];

var ValidImageTypes = ["image/jpeg", "image/png"];

if ($.inArray(fileType, ValidImageTypes) < 0) {

toastr.error("Invalid file type");

}

else {

var reader = new FileReader();

reader.onload = function (e) {

//set the image data as source

$('#profileImage').attr('src', e.target.result);

$('#ProfilePic').val(e.target.result);

}

reader.readAsDataURL(imageFile);

}

}

}

$("#imageUpload").change(function () {

previewProfileImage(this);

});

</script>

<!-- checkbox tick js -->

<script>

$('.checkbox-css').click(function () {

$(this).toggleClass('active fa fa-check');

if ($(this).siblings('.hidden-checkbox').prop('checked')) {

$(this).siblings('.hidden-checkbox').prop('checked', false);

}

else { $(this).siblings('.hidden-checkbox').prop('checked', true); }

});

</script>

<script type="text/javascript">

function registerSuccess(data) {

$(".preloder").fadeOut();

if (data.Status) {

toastr.success(data.Message);

setTimeout(function () {

if (data.Type == 3) {

//window.location.href = "/Account/login?type=3";

swal({

title: "Registered!",

text: "Now you can log in as User or Cleaner.",

type: "success",

timer: 4000

},

function () {

setTimeout(function () {

window.location.href = "/Account/home";

}, 100);

});

}

else if (data.Type == 2) {

window.location.href = "/Cleaner/home";

}

else if (data.Type == 1) {

window.location.href = "/User/home";

}

}, 500);

}

else {

$('#btn-dummy-register-submit').prop('disabled', false);

toastr.error(data.Message);

}

}

$(document).ready(function () {

});

$(document).on('change', '#ProfileType', function () {

var value = $(this).val();

if (value == 2) {

$('#CompanyName').parent().hide();

}

else {

$('#CompanyName').parent().show();

}

});

$(document).on('keydown', "#Phone , #Zip ", function (e) {

// Allow: backspace, delete, tab, escape, enter and .

if ($.inArray(e.keyCode, [46, 8, 9, 27, 13, 110, 190]) !== -1 ||

// Allow: Ctrl+A, Command+A

(e.keyCode === 65 && (e.ctrlKey === true || e.metaKey === true)) ||

// Allow: home, end, left, right, down, up

(e.keyCode >= 35 && e.keyCode <= 40)) {

// let it happen, don't do anything

return;

}

// Ensure that it is a number and stop the keypress

if ((e.shiftKey || (e.keyCode < 48 || e.keyCode > 57)) && (e.keyCode < 96 || e.keyCode > 105)) {

e.preventDefault();

}

});

$(document).on('click', '#btn-dummy-register-submit', function () {

var status = true;

$('#accnt-register input[type="text"],input[type="password"]').each(function () {

if ($(this).val() != null && $(this).val() != "") {

if (this.id == "Password" || this.id == "ConfirmPassword") {

if ($('#PasswordMatch').val() == "false") {

$('#spn-ConfirmPassword').text("Password does not match").show();

status = false;

}

else {

$(this).parent().find('#spn-' + this.id).hide();

}

}

else {

if (this.id == "Email") {

if ($('#ValidEmail').val() == "false") {

status = false;

$('#spn-Email').text("Invalid email").show();

}

else {

$(this).parent().find('#spn-' + this.id).hide();

}

}

else {

$(this).parent().find('#spn-' + this.id).hide();

}

}

}

else {

if (this.id == "CompanyName") {

if ($('#ProfileType').val() == 1) {

$(this).parent().find('#spn-' + this.id).text("Required").show();

status = false;

}

else {

$(this).parent().find('#spn-' + this.id).hide();

}

}

else {

$(this).parent().find('#spn-' + this.id).text("Required").show();

status = false;

}

}

});

if ($('#ProfilePic').val() != "") {

if (status == true) {

$('#btn-dummy-register-submit').prop('disabled', true);

$('#btn-register-submit').trigger('click');

$(".preloder").show();

}

}

else {

toastr.error("Upload image");

}

});

$(document).on('keyup', '#Email', function () {

var email = $(this).val();

if (email != null && email != "") {

var status = emailValidator(email);

setTimeout(function () {

if (status) {

$('#ValidEmail').val("true");

$('#spn-Email').hide();

}

else {

$('#ValidEmail').val("false");

$('#spn-Email').text("Invalid email").show();

}

}, 250);

}

else {

$('#spn-Email').text("Required").show();

}

});

$(document).on('click', '#foo', function () {

if ($('#accnt-checkbox-register-terms').is(":checked")) {

$('#btn-dummy-register-submit').prop('disabled', false);

$('#btn-dummy-register-submit').css('background', '#000');

}

else {

$('#btn-dummy-register-submit').prop('disabled', true);

$('#btn-dummy-register-submit').css('background', '#d6d6d6');

}

});

$(document).on('keyup', '#Password', function () {

var password = $('#Password').val();

var confirmpassword = $('#ConfirmPassword').val();

if (password != null && password != "" && confirmpassword != null && confirmpassword != "") {

$('#spn-Password').hide();

if (password != confirmpassword) {

$('#spn-ConfirmPassword').text("Password does not match").show();

$('#PasswordMatch').val("false");

}

else {

$('#spn-ConfirmPassword').hide();

$('#spn-Password').hide();

$('#PasswordMatch').val("true");

}

}

else {

if (password != "" && confirmpassword == "") {

$('#spn-Password').hide();

$('#PasswordMatch').val("false");

}

else if (password == "") {

$('#spn-Password').text("Required").show();

$('#PasswordMatch').val("false");

}

}

});

$(document).on('keyup', '#ConfirmPassword', function () {

var password = $('#Password').val();

var confirmpassword = $('#ConfirmPassword').val();

if (password != null && password != "" && confirmpassword != null && confirmpassword != "") {

if (password != confirmpassword) {

$('#spn-ConfirmPassword').text("Password does not match").show();

$('#PasswordMatch').val("false");

}

else {

$('#spn-ConfirmPassword').hide();

$('#spn-Password').hide();

$('#PasswordMatch').val("true");

}

}

else {

if (password != "" && confirmpassword == "") {

$('#spn-ConfirmPassword').text("Required").show();

$('#spn-Password').hide();

$('#PasswordMatch').val("false");

}

else {

if (confirmpassword != "" && password == "") {

$('#spn-Password').text("Required").show();

$('#PasswordMatch').val("false");

}

}

}

});

</script>

<script>

var componentForm = {

street\_number: 'long\_name',

route: 'long\_name',

locality: 'long\_name',

administrative\_area\_level\_1: 'long\_name',

country: 'long\_name',

postal\_code: 'long\_name',

neighborhood: 'long\_name',

sublocality\_level\_3: 'long\_name',

sublocality\_level\_2: 'long\_name',

sublocality\_level\_1: 'long\_name'

};

google.maps.event.addDomListener(window, 'load', function () {

var places = new google.maps.places.Autocomplete(document.getElementById('Address'), { componentRestrictions: { country: 'au' } });

var cities = new google.maps.places.Autocomplete(document.getElementById('Suburb'), { types: ['(cities)'], componentRestrictions: { country: 'au' } });

var state = new google.maps.places.Autocomplete(document.getElementById('State'), { types: ['(regions)'], componentRestrictions: { country: 'au' } });

var country = new google.maps.places.Autocomplete(document.getElementById('Country'), { types: ['(regions)'], componentRestrictions: { country: 'au' } });

google.maps.event.addListener(cities, 'place\_changed', function () {

var cityFieldValue = $('#Suburb').val();

cityFieldValue = cityFieldValue.substring(0, cityFieldValue.indexOf(','));

$('#Suburb').val(cityFieldValue);

});

google.maps.event.addListener(state, 'place\_changed', function () {

var stateFieldValue = $('#State').val();

stateFieldValue = stateFieldValue.substring(0, stateFieldValue.indexOf(','));

$('#State').val(stateFieldValue);

});

google.maps.event.addListener(country, 'place\_changed', function () {

var countryFieldValue = $('#Country').val();

$('#Country').val(countryFieldValue);

});

google.maps.event.addListener(places, 'place\_changed', function () {

var addressFieldValue = $('#Address').val();

//$('#city').val('');

//$('#StreetNumber').val('');

//$('#country').val('');

//$('#postalCode').val('');

//$('#Route').val('');

//$('#UserLat').val('');

//$('#UserLong').val('');

var place = places.getPlace();

var address = place.formatted\_address;

var latitude = place.geometry.location.lat();

var longitude = place.geometry.location.lng();

addressFieldValue = addressFieldValue.substring(0, addressFieldValue.indexOf(','));

for (var i = 0; i < place.address\_components.length; i++) {

var addressType = place.address\_components[i].types[0];

//console.log(addressType);

if (addressType == "locality") {

var val = place.address\_components[i][componentForm[addressType]];

//console.log(val)

$('#Suburb').val(val);

}

if (addressType == "route") {

var val = place.address\_components[i][componentForm[addressType]];

//console.log(val)

}

if (addressType == "postal\_code") {

var val = place.address\_components[i][componentForm[addressType]];

$('#Zip').val(val);

//console.log(val)

}

if (addressType == "neighborhood") {

var val = place.address\_components[i][componentForm[addressType]];

//console.log(val)

}

if (addressType == "administrative\_area\_level\_1") {

var val = place.address\_components[i][componentForm[addressType]];

//console.log(val)

$('#State').val(val);

}

if (addressType == "country") {

var val = place.address\_components[i][componentForm[addressType]];

//console.log(val)

$('#Country').val(val);

}

if (addressType == "street\_number") {

var val = place.address\_components[i][componentForm[addressType]];

//console.log(val)

//$('#StreetNumber').val(val);

}

}

$('#Address').val(addressFieldValue);

$('#UserLat').val(latitude);

$('#UserLong').val(longitude);

//var mesg = "Address: " + address;

//mesg += "\nLatitude: " + latitude;

//mesg += "\nLongitude: " + longitude;

});

});

</script>