

## Adding Custom User Data Fields Into a User Details Table

We need to allow users to enter their custom details, such as first and last name, when they are registering, thus we need to modify the registration functionality to allow for this.

In the **RegisterViewModel** class of the **Models/AccountViewModel.cs** file we simply add the desired properties, such as **FirstName** and **LastName**, with the appropriate data attributes.

```
1
2 public class RegisterViewModel
3 {
4     [Required]
5     public string FirstName { get; set; }
6     [Required]
7     public string LastName { get; set; }
8     // Code Omitted for Brevity
9 }
```

We now need to add the form fields to the **Views/Account/Register.cshtml** file, such as below.

```
1
2 <div class="form-group">
3     @Html.LabelFor(m => m.FirstName, new { @class = "col-md-2 control-label" })
4     <div class="col-md-10">
5         @Html.TextBoxFor(m => m.FirstName, new { @class = "form-control" })</div>
6     </div>
7     <div class="form-group">
8         @Html.LabelFor(m => m.LastName, new { @class = "col-md-2 control-label" })
9         <div class="col-md-10">
10             @Html.TextBoxFor(m => m.LastName, new { @class = "form-control" })</div>
11         </div>
12 </div>
```

Finally, in the HttpPost version of the Register method in the Controllers/accountController.cs file, we add pull the custom properties in the RegisterViewModel into a new UserDetail object and save it as a new record in the database.

```

[HttpPost]
[AllowAnonymous]
[ValidateAntiForgeryToken]
public async Task<ActionResult> Register(RegisterViewModel model)
{
    if (ModelState.IsValid)
    {
1      var user = new ApplicationUser { UserName = model.Email, Email = model.Email };
2      var result = await UserManager.CreateAsync(user, model.Password);
3      if (result.Succeeded)
4      {
5          #region Dealing with custom user details
6              UserDetail newUserDeets = new UserDetail();
7              newUserDeets.UserID = user.Id;
8              newUserDeets.FirstName = model.FirstName;
9              newUserDeets.LastName = model.LastName;
10             newUserDeets.ResumeFile = model.ResumeFile;/--TODO: handle file upload
11             FsdEntities db = new FsdEntities();
12             db.UserDetails.Add(newUserDeets);
13             db.SaveChanges();

            #endregion

            var code = await UserManager.GenerateEmailConfirmationTokenAsync(user.Id);

            // Code Omitted for Brevity

```

At this point we can run the application and register a new user. The form will allow us to enter our first and last name, and will insert those values into the UserDetails table.