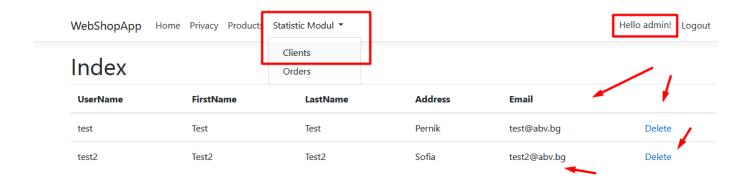
## WebShopApp - Част 6 - Работа с потребители

Добавяне на възможност за виждане на всички регистрирани потребители от администратора и възможност за изтриване на потребител/клиент/.



В папка Models Правим нова папка Client и в нея вю модел ClientIndexVM.

```
public class ClientIndexVM
{
   public string Id { get; set; } = null!;

   [Display(Name = "Username")]
   public string UserName { get; set; } = null!;

   [Display(Name = "First Name")]
   public string FirstName { get; set; } = null!;

   [Display(Name = "Last Name")]
   public string LastName { get; set; } = null!;

   public string Address { get; set; } = null!;

   public string Email { get; set; } = null!;

   public bool IsAdmin { get; set; }
}
```

#### Създаваме контролер ClientController

В него инжектираме UserManager и имплементираме екшъна Index

```
public class ClientController : Controller
   private readonly UserManager<ApplicationUser> _userManager;
   public ClientController(UserManager<ApplicationUser> userManager)
      this._userManager = userManager;
  // GET: ClientController
  public async Task<IActionResult> Index()
      var allUsers = this. userManager.Users
          .Select(u => new ClientIndexVM
              Id = u.Id,
              UserName = u.UserName,
              FirstName = u.FirstName,
              LastName = u.LastName,
              Address = u.Address,
              Email = u.Email,
          })
          .ToList();
      // Id на всички администратори
      var adminIds = (await userManager.GetUsersInRoleAsync("Administrator"))
           .Select(a => a.Id).ToList();
      //Ако потребителят е в списъка, то IsAdmin става true
      foreach (var user in allUsers)
          user.IsAdmin = adminIds.Contains(user.Id);
      //Вадим само клиентите без администратора и ги сортираме по username
      var users = allUsers.Where(x => x.IsAdmin == false)
          .OrderBy(x => x.UserName).ToList();
      //връщаме списъка
      return this. View (users);
```

```
@model IEnumerable<WebShopApp.Models.Client.ClientIndexVM>
@{
   ViewData["Title"] = "Index";
}
<h1>Clients</h1>
<thead>
      @Html.DisplayNameFor(model => model.UserName)
         @Html.DisplayNameFor(model => model.FirstName)
         @Html.DisplayNameFor(model => model.LastName)
         @Html.DisplayNameFor(model => model.Address)
         @Html.DisplayNameFor(model => model.Email)
         </thead>
   @foreach (var item in Model)
         @Html.DisplayFor(modelItem => item.UserName)
             @Html.DisplayFor(modelItem => item.FirstName)
             >
                @Html.DisplayFor(modelItem => item.LastName)
             @Html.DisplayFor(modelItem => item.Address)
             @Html.DisplayFor(modelItem => item.Email)
             <a asp-action="Delete" asp-route-id="@item.Id" class="btn-</pre>
danger">Delete</a>
             }
```

Създаваме вю за Index() на ClientController в папка Views/Client

В Layout.cshnml да запишем правилния контролер и екшън за обработка на клиентите.

Last Name

gogov

totov

Address

Pernik lulin 67

sofia Iulin 89

Email

test@abv.bg

test5@abv.bg

Тествайте, като влезете в администраторския профил.

First Name

gogo

toto

### Изтриване на потребител

Username

toto

Създаваме ClientDeleteVM е папка Models

```
public class ClientDeleteVM
{
   public string Id { get; set; } = null!;

   [Display(Name = "Username")]
   public string UserName { get; set; } = null!;

   [Display(Name = "First Name")]
   public string FirstName { get; set; } = null!;

   [Display(Name = "Last Name")]
   public string LastName { get; set; } = null!;

   public string Address { get; set; } = null!;

   public string Email { get; set; } = null!;
}
```

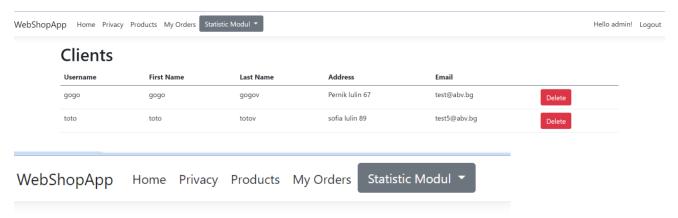
```
// GET: ClientController/Delete/5
   0 references
   public ActionResult Delete(string id)
       var user = this.userManager.Users.FirstOrDefault(x => x.Id == id);
       if (user == null)
          return NotFound();
       ClientDeleteVM userToDelete = new ClientDeleteVM()
           Id = user.Id,
           FirstName = user.FirstName,
           LastName = user.LastName,
           Address = user.Address,
           Email = user.Email.
           UserName = user.UserName
       };
       return View(userToDelete);
   }
// POST: ClientController/Delete/5
[HttpPost]
[ValidateAntiForgeryToken]
public async Task<IActionResult> Delete(ClientDeleteVM bidingModel)
    string id = bidingModel.Id;
    var user = await userManager.FindByIdAsync(id);
    if (user == null)
    {
        return NotFound();
    IdentityResult result = await userManager.DeleteAsync(user);
    if (result.Succeeded)
        return RedirectToAction("Success");
    return NotFound();
public ActionResult Success()
     return View();
```

#### Създаваме вюта в папка Views/Client с имена Delete и Success

```
@model WebShopApp.Models.Client.ClientDeleteVM
@{
    ViewData["Title"] = "Delete";
}
<h1>Delete</h1>
<h3>Are you sure you want to delete this?</h3>
<div>
    <h4>ClientDeleteVM</h4>
    <hr />
    <dl class="row">
        <dt class="col-sm-2">
            @Html.DisplayNameFor(model => model.UserName)
        <dd class="col-sm-10">
            @Html.DisplayFor(model => model.UserName)
        <dt class="col-sm-2">
            @Html.DisplayNameFor(model => model.FirstName)
        </dt>
        <dd class="col-sm-10">
            @Html.DisplayFor(model => model.FirstName)
        <dt class="col-sm-2">
            @Html.DisplayNameFor(model => model.LastName)
        </dt>
        <dd class="col-sm-10">
            @Html.DisplayFor(model => model.LastName)
        <dt class="col-sm-2">
            @Html.DisplayNameFor(model => model.Address)
        </dt>
        <dd class="col-sm-10">
            @Html.DisplayFor(model => model.Address)
        </dd>
        <dt class="col-sm-2">
            @Html.DisplayNameFor(model => model.Email)
        </dt>
        <dd class="col-sm-10">
            @Html.DisplayFor(model => model.Email)
        </dd>
    </dl>
    <form asp-action="Delete">
        <input type="hidden" asp-for="Id" />
        <input type="submit" value="Delete" class="btn btn-danger" /> |
        <a asp-action="Index" class="btn btn-warning">Back to List</a>
    </form>
</div>
```

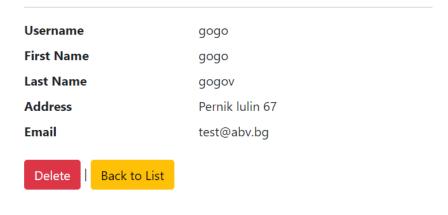


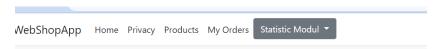
Тествайте, като влезете в администраторския профил.



# **Delete**

# Are you sure you want to delete this? ClientDeleteVM





You successfully remove a user!

Back to List