Startup Class:s

1. app.UseAuthentication();

2. services.AddAuthentication(CookieAuthenticationDefaults.AuthenticationScheme)

.AddCookie(options => {

options.LoginPath = "/Auth/Login/";

});

3. If need to disable redirect:

services.AddAuthentication(CookieAuthenticationDefaults.AuthenticationScheme)

.AddCookie(options => {

options.Events.OnRedirectToLogin = (context) =>

{

context.Response.StatusCode = 401;

return Task.CompletedTask;

};

});

AuthController:

//Authenticate the user

[HttpPost]

public async Task<IActionResult> Login(AuthUser user)

{

//Call the AuthManager to authenticate the user!!

if (user.UserName == "jdoe" && user.Password == "password")

{

var claims = new List<Claim>

{

new Claim(ClaimTypes.Name, user.UserName),

new Claim(ClaimTypes.Role, "admin")

};

var userIdentity = new ClaimsIdentity(claims, "login");

ClaimsPrincipal principal = new ClaimsPrincipal(userIdentity);

await HttpContext.SignInAsync(principal);

//Just redirect to our index after logging in.

return Redirect("/");

}

return View();

}

//Remove authentication:

public async Task<IActionResult> Logout()

{

await HttpContext.SignOutAsync();

return Redirect("/");

}