public async Task<IActionResult> OnPostAsync(string returnUrl = null)

{

returnUrl ??= Url.Content("~/");

ExternalLogins = (await \_signInManager.GetExternalAuthenticationSchemesAsync()).ToList();

if (ModelState.IsValid)

{

// This doesn't count login failures towards account lockout

// To enable password failures to trigger account lockout, set lockoutOnFailure: true

var result = await \_signInManager.PasswordSignInAsync(Input.Email, Input.Password, Input.RememberMe, lockoutOnFailure: false);

if (result.Succeeded)

{

\_logger.LogInformation("User logged in.");

return LocalRedirect(returnUrl);

}

if (result.RequiresTwoFactor)

{

return RedirectToPage("./LoginWith2fa", new { ReturnUrl = returnUrl, RememberMe = Input.RememberMe });

}

if (result.IsLockedOut)

{

\_logger.LogWarning("User account locked out.");

return RedirectToPage("./Lockout");

}

else

{

ModelState.AddModelError(string.Empty, "Invalid login attempt.");

return Page();

}

}

// If we got this far, something failed, redisplay form

return Page();

}