

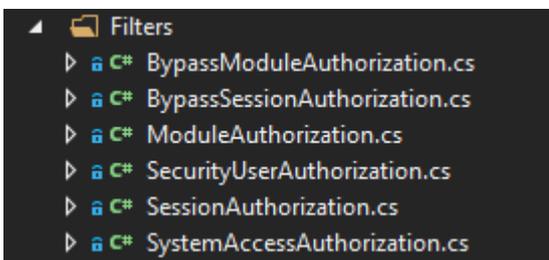
Attributes

[Go To Naming Conventions](#)

The ASP.NET MVC framework supports four different types of filters executed in the following order:

1. **Authorisation filters** - Implements the `IAuthorizationFilter` attribute.
2. **Action filters** - Implements the `IActionFilter` attribute.
3. **Result filters** - Implements the `IResultFilter` attribute.
4. **Exception filters** - Implements the `IExceptionFilter` attribute.

The common authorisation filters that is used in Signify are



- **BypassModuleAuthorisation:** The MVC authorization filter that enables one to by-pass module authentication.
- **BypassSessionAuthorisation:** The MVC authorization filter that enables one to by-pass session authentication.
- **ModuleAuthorisation:** Extended authorisation that determines if the user consists of a session.
- **SecurityUserAuthorisation:** Authorize the user against roles.
- **Session Authorisation:** Extended authorisation that determines if the user consists of a session, implemented in the base controller
- **SystemAccessAuthorisation:** Authorize the user on access to the module and the active status of the module.

Example of use

Applying a filter on a single method in the controller

```

[BypassSessionAuthorization]
0 references | 0 changes | 0 authors, 0 changes
public ActionResult Confirmation(Guid token)
{
    try
    {
        var decryptedToken = sysAccessRequest.TokenDataWrapper.Decrypt(token);

        if (decryptedToken != null)
        {
            return View(new AccessRequestConfirmationViewModel(decryptedToken.sysUserRegistrationId, decryptedToken.sysUserRegistrationId));
        }
        else
        {
            throw new Exception("The request token could not be deserialized.");
        }
    }
    catch (Exception ex)
    {
        ErrorUtilities.LogException(ex);
        throw new Exception("The request token could not be decrypted or deserialized.", ex.InnerException);
    }
}

```

Applying a filter on the controller as a whole

```

namespace SignifyHR.Areas.Security.Controllers
{
    [BypassSessionAuthorization]
    [BypassModuleAuthorization]
    0 references | Martinique Lourens, 62 days ago | 7 authors, 11 changes | 2 work items
    public class AccountController : SignifyHR.Controllers.BaseController
    {

```

Revision #9

Created 17 September 2020 01:54:50 by Theuns Pretorius

Updated 8 October 2020 05:17:35 by Theuns Pretorius