## Objectives

* Create an ASP.NET core MVC application (Create an ASP.NET core project & add the MVC Framework)
* Secure the application using Identity server,
* OWASP (<https://cheatsheetseries.owasp.org/cheatsheets/DotNet_Security_Cheat_Sheet.html>)
* Add the \_\_ database.
* Interact with the database using the Entity Framework ORM
* Apply dependency injection & MVC pattern
* Expose data using a JSON web API
* (misc) logging

## Frontend

For the front-end the goal is to set up a clean and responsive user interface that scales well on most screen sizes; The UI should be intuitive, display relevant form fields, error/success messages and use Ajax requests where necessary for better UX.

* Component Library: [Bootstrap 4](https://getbootstrap.com/)
* Datatable: [Kendo UI Grid](https://demos.telerik.com/kendo-ui/grid/index)
* JS Libraries: [JQuery](https://jquery.com/)

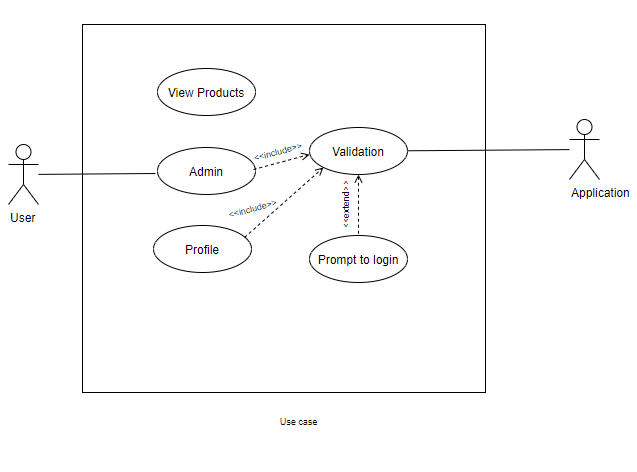
Backend

For the back-end, the goal is to build up a secure server that serves the front-end securely; It should be secure and allow for CRUD operations by exposing a minimal API to authenticated and authorized admin users.

* Allow only GET requests via ajax
* Use logging
* authN & authZ middleware
* Data Access

## Routing Table

|  |  |  |
| --- | --- | --- |
| **Controller** | **Action** | **Route** |
| Home | Index | /home/index |
| Home | Error | /home/error |
| Admin | Index | /admin/index |
| Product | Create | /product/create |
| Product | Read | /product/:pid |
| Product | Update | /product/update/:pid |



to-do

[Setup]

1. ~~Setup ASP.NET core (3.1) MVC project + Identity(in-app)~~
2. ~~Do db migrations~~
3. Setup testing library
4. ~~Configure console logger~~
5. ~~Setup kendo UI Grid~~

[Design]

1. ~~Rework templates and partials~~
2. ~~Minor scaffolding re-design for Areas > Identity~~
3. ~~Add 404 status page~~
4. ~~Design & implement landing page~~
5. ~~Design & implement products manager page~~
6. Design & implement shop/products display page

[Security]

1. ~~Basic authentication with Identity~~
2. ~~SImple authorization for pages & CRUD~~

[Enhancements]

1. Unit tests
2. Role based auth