Make domain models also persistence models? – dilemma.

At start it seemed the best to put models representing entities in Core of the application but then the dilemma appeared. Since they were only ‘thin’ objects with no logic whatsoever they also fit well into persistence. There were two choices, either make separate domain and persistence models and use mapper to transform them as needed or just put models into persistence part of application. Considering small size of application and no further plans to scale it the obvious choice was to put them into persistence layer.

Authentication/Authorization by Bearer Token – it limits Cross-Site Request Forgery attacks since it has expiration date and doesn’t require client app to store user credentials. Browser sends x-www-urlencoded request with username and password and then the Backend OWIN middleware takes control sending back a response with token for further Authorization

Sources: <https://docs.microsoft.com/en-us/aspnet/web-api/overview/security/individual-accounts-in-web-api>

http://bitoftech.net/2014/06/01/token-based-authentication-asp-net-web-api-2-owin-asp-net-identity/