**Web API check list**

* **Projects**: [Project].Models, .Data, .Web, .WCF
* **DbContext**: [Project]DbContext.cs with all DbSets
* **Migrations**:
  + package manager > enable\_migrations (for [Project].Data)
  + Configuration() > this.AutomaticMigrationDataLossAllowed = true;
  + In ApplicationDbContext :

Database.SetInitializer<ApplicationDbContext>(new MigrateDatabaseToLatestVersion<ApplicationDbContext, Configuration>());

* **Repositories**
  + All, Find, Add, Update, Delete(T), Delete(id), int SaveChanges
  + ChangeState(T entity, EntityState state)
* **UoW**
  + private context, dictionary, all repos
  + GetRepository<T>()
  + Use the ApplicationUser from WebAPI
* **BaseController**

protected IApplicationData data;

public BaseApiController(IApplicationData data)

{

this.data = data;

}

* **IUserIdProvider, and AspNetUserIdProvider in Infrastructure folder with method:**

public string GetUserId()

{

return Thread.CurrentPrincipal.Identity.GetUserId();

}

For unit testing.

Otherwise to get the method in User.Identity, use using Microsoft.AspNet.Identity;

* **IoC**: Ninject.web.webapi OWIN
  + GitHub link: <https://github.com/ninject/Ninject.Web.Common/wiki/Setting-up-a-OWIN-WebApi-application>
  + Add namespaces:

using Ninject;

using Ninject.Web.Common.OwinHost;

using Ninject.Web.WebApi.OwinHost;

* + Startup.cs >

app.UseNinjectMiddleware(CreateKernel)

.UseNinjectWebApi(GlobalConfiguration.Configuration);

* + static StandardKernel CreateKernel()

{

var kernel = new StandardKernel();

kernel.Load(Assembly.GetExecutingAssembly());

RegisterMappings(kernel);

return kernel;

}

* + static RegisterMappings(kernel)

kernel.Bind<IApplicationData>().To<ApplicationData>().WithConstructorArgument("context", c => new ApplicationDbContext());

kernel.Bind<IUserIdProvider>().To<AspNetUserIdProvider>();

* **Routing**: WebApiConfig.cs > for routings
* **Fix connection strings**
* **OData** – OData Web API 2.2, v4

Open the file App\_Start/WebApiConfig.cs. Add the following **using** statements:

using System.Web.OData.Builder;

using System.Web.OData.Extensions;

Then add : config.AddODataQueryFilter();

* **CORS** for Web API 2.2
  + WebApiConfig.cd > config.EnableCors(new EnableCorsAttribute("\*", "\*", "\*"));
  + Global.asax >

protected void Application\_BeginRequest(object sender, EventArgs e)

{

Response.Headers.Add("Access-Control-Allow-Origin", "\*");

}

**WCF Check list**

* **[Service].svc**

Factory="System.ServiceModel.Activation.WebServiceHostFactory"

* **[ServiceInterface]**

[WebGet(UriTemplate="")]

* **Web.config**

<system.serviceModel>

<standardEndpoints>

<webHttpEndpoint>

<standardEndpoint helpEnabled="true" defaultOutgoingResponseFormat="Json" />

</webHttpEndpoint>

</standardEndpoints>

</system.serviceModel>