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

</configSections>

<connectionStrings>

<add name="DefaultConnection" connectionString="Data Source=.;Initial Catalog=Articles;Integrated Security=True" providerName="System.Data.SqlClient" />

</connectionStrings>

* + <appSettings></appSettings>
* **BaseController**

protected IArticlesData data;

public BaseApiController(IArticlesData data)

{

this.data = data;

}

* **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 >

public void Configuration(IAppBuilder app)

{

ConfigureAuth(app);

app.UseNinjectMiddleware(CreateKernel).UseNinjectWebApi(GlobalConfiguration.Configuration);

}

private static StandardKernel CreateKernel()

{

var kernel = new StandardKernel();

kernel.Load(Assembly.GetExecutingAssembly());

RegisterMappings(kernel);

return kernel;

}

private static void RegisterMappings(StandardKernel kernel)

{

kernel

.Bind<IMusicStoreData>()

.To<MusicStoreData>()

.WithConstructorArgument("context", c => new MusicStoreDbContext());

}

* + RegisterMappings(kernel)
    - kernel.Bind<IData>().To<Data>().WithConstructorArgument("context", c => new DbContext());
* **Routing**: WebApiConfig.cs > for routings
* **CORS** for Web API 2.2
  + WebApiConfig.cd > config.EnableCors(new EnableCorsAttribute("\*", "\*", "\*"));

config.Formatters.JsonFormatter.SerializerSettings.ReferenceLoopHandling = Newtonsoft.Json.ReferenceLoopHandling.Ignore; config.Formatters.Remove(config.Formatters.XmlFormatter);

* + Global.asax >

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

{

if (this.Response.Headers.GetValues("Access-Control-Allow-Origin")==null)

{

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

}

}

* **Files to change for Template project**

Rename Exam’s folders and projects files to Project Name

Add existing projects with new names

Change in projects in Application/Assembly & Namespace & Assembly Information

Solution refactor/adjust namespaces

remove reference to other projects and add with resharper

WebApi\Views\Web.config

<add namespace="Exam.WebApi" />

</namespaces>

WebApi\Areas\HelpPage\App\_Start\HelpPageConfig.cs

on line 26 : MessageId = "Exam.WebApi.Areas.HelpPage………

replace templatename with project name in cshtml’s on WebApi

**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>