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>()
- **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 >

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

  - 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("*", "*", "*"));
  - 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>
  ```