

Rasmus Lystrøm
External Associate Professor
ITU
rnie@itu.dk

Agenda

Course Evaluation

Xamarin.Forms

Consuming a REST API

Security in ASP.NET, ~~UWP~~, and ~~Xamarin.Forms~~

Final lecture

Course Evaluation I/II

Universal Windows Platform and latest bits

Java

Mandatory exercises

The Book

Microsoft propaganda

Work load

Course Evaluation II/II

Condescending, intimidating, arrogant,
know-it-all, mind-blind

Next year:

Swap lecture days

New book?

Reduce GUI part?

Exercises slightly smaller

Upgrade Xamarin.Forms I/II

Move from PCL to .NET Standard

1. Close Visual Studio
2. Open folder with PCL project
3. Delete the following folders and files:
 - bin
 - obj
 - Properties
 - packages.config

Upgrade Xamarin.Forms I/II

4. Open the .csproj file and replace the contents with:

```
<Project Sdk="Microsoft.NET.Sdk">

  <PropertyGroup>
    <TargetFramework>netstandard2.0</TargetFramework>
  </PropertyGroup>

  <ItemGroup>
    <PackageReference Include="Xamarin.Forms"
                      Version="2.4.0.74863" />
  </ItemGroup>

</Project>
```

5. Open Visual Studio

6. Update packages and build

Swagger I/II

PS> Install-Package Swashbuckle.AspNetCore

```
services.AddRouting(options =>
{
    options.LowercaseUrls = true;
});
```

```
services.AddSwaggerGen(c =>
{
    c.SwaggerDoc("v1", new Info { Title = "My API", Version =
"v1" });
    c.DocumentFilter<LowerCaseDocumentFilter>();
});
```


Swagger II/II

```
app.UseSwagger();
```

```
app.UseSwaggerUI(c =>
{
    c.SwaggerEndpoint("/swagger/v1/swagger.json", "My API");
});
```

```
public class LowerCaseDocumentFilter : IDocumentFilter
{
    public void Apply(SwaggerDocument swaggerDoc,
DocumentFilterContext context)
    {
        swaggerDoc.Paths = swaggerDoc.Paths.ToDictionary(d =>
            d.Key.ToLower(), d => d.Value);
    }
}
```

"launchUrl": [http://localhost:\[port\]/swagger/](http://localhost:[port]/swagger/)

Security in ASP.NET Core

File -> New Project

(Copy setup to exiting project)

Enable SSL

```
services.Configure<MvcOptions>(options =>
{
    options.Filters.Add(new RequireHttpsAttribute());
});
```

```
var options = new
RewriteOptions().AddRedirectToHttps();
app.UseRewriter(options);
```

Final Lecture

23 November 2017

Recap / repetition – what do you need?

Final Lecture II

Topics:

Publishing apps to the cloud

Exception handling

IDisposable

Settings and secrets

Logging

Security in UWP and Xamarin.Forms

Entity Framework Core – what?