

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Threading.Tasks;
5 using Microsoft.AspNetCore.Builder;
6 using Microsoft.AspNetCore.Hosting;
7 using Microsoft.Extensions.Configuration;
8 using Microsoft.Extensions.DependencyInjection;
9 using DatabaseConnect;
10 using Microsoft.EntityFrameworkCore;
11 using Microsoft.AspNetCore.Authentication.JwtBearer;
12 using Microsoft.IdentityModel.Tokens;
13 using System.Text;
14 using Microsoft.AspNetCore.Rewrite;
15 using Microsoft.IdentityModel.Protocols.OpenIdConnect;
16 using Swashbuckle.AspNetCore.Swagger;
17
18 namespace LibraryAppMVC
19 {
20     public class Startup
21     {
22         public Startup(IConfiguration configuration)
23         {
24             Configuration = configuration;
25         }
26
27         public IConfiguration Configuration { get; }
28
29         // This method gets called by the runtime. Use this method to add
30         // services to the container.
31         public void ConfigureServices(IServiceCollection services)
32         {
33             /* JWT Resources
34              * https://auth0.com/blog/securing-asp-dot-net-core-2-applications-
35              \* https://blogs.msdn.microsoft.com/webdev/2017/04/06/jwt-validation-
36              \\* https://docs.microsoft.com/en-us/aspnet/core/migration/1x-to-2x/
37              \\\* identity-2x
38              \\\*/
39             services.AddAuthentication\\\(JwtBearerDefaults.AuthenticationScheme\\\)
40                 .AddJwtBearer\\\(opt =>
41                 {
42                     opt.Audience = "http://localhost:5001/";
43                     opt.Authority = "http://localhost:5000/";
44                     opt.Configuration = new OpenIdConnectConfiguration\\\(\\\);
45                     // Stackexchange https://stackoverflow.com/
46                     // questions/37693516/unable-to-obtain-configuration-from-well-
47                     // known-openid-configuration/37973711
48                     opt.TokenValidationParameters = new TokenValidationParameters
49                     {
50                         ValidateIssuer = true,
```

```
47         ValidateAudience = true,
48         ValidateLifetime = true,
49         ValidateIssuerSigningKey = true,
50         ValidIssuer = Configuration["Jwt:Issuer"],
51         ValidAudience = Configuration["Jwt:Issuer"],
52         IssuerSigningKey = new SymmetricSecurityKey           ↗
            (Encoding.UTF8.GetBytes(Configuration["Jwt:Key"])),
53         ClockSkew = TimeSpan.Zero // Lets tokens expire when   ↗
            they say they will
54     };
55 });
56
57 services.AddSingleton(Configuration);
58
59 services.AddDbContext<Context>(options => options.UseSqlServer   ↗
    (Configuration.GetConnectionString("TheDatabase")));
60 services.AddMvc()
61     .AddJsonOptions(options =>                                  ↗
        options.SerializerSettings.ReferenceLoopHandling =       ↗
            Newtonsoft.Json.ReferenceLoopHandling.Ignore);
62 services.AddSwaggerGen(c =>
63 {
64     c.SwaggerDoc("v1", new Info { Title = "FBLA Mobile App", Version   ↗
        = "v1" });
65 });
66 }
67
68 // This method gets called by the runtime. Use this method to configure   ↗
    the HTTP request pipeline.
69 public void Configure(IApplicationBuilder app, IHostingEnvironment env)
70 {
71     var redirOpt = new RewriteOptions();
72     if (env.IsDevelopment())
73     {
74         app.UseDeveloperExceptionPage();
75         app.UseBrowserLink();
76     }
77     else
78     {
79         app.UseExceptionHandler("/Home/Error");
80         // HTTPS redirector
81         redirOpt.AddRedirectToHttps();
82     }
83     app.UseRewriter(redirOpt);
84     app.UseAuthentication();
85     app.UseStaticFiles();
86     app.UseSwagger();
87     app.UseSwaggerUI(c =>
88     {
89         c.SwaggerEndpoint("/swagger/v1/swagger.json", "FBLA Mobile App   ↗
            v1");
90     });
```

```
91
92     app.UseMvc(routes =>
93     {
94         routes.MapRoute(
95             name: "default",
96             template: "{controller=Home}/{action=Index}/{id?}");
97     });
98     }
99 }
100 }
101
```