

AMDT TEST Assignment

~Documentation~

This is sample web application I have developed based on request of AMDT School of creativity,

I have used ASP.NET core with visual studio c# 2019 with following packages,

```
<PackageReference Include="Microsoft.AspNetCore.Authentication.Google" Version="5.0.9" />
  <PackageReference Include="Microsoft.AspNetCore.Authentication.OpenIdConnect"
Version="5.0.9" />
  <PackageReference Include="Microsoft.AspNetCore.Identity.EntityFrameworkCore"
Version="5.0.9" />
  <PackageReference Include="Microsoft.AspNetCore.Mvc.Razor.RuntimeCompilation"
Version="5.0.9" />
  <PackageReference Include="Microsoft.EntityFrameworkCore.Sqlite" Version="5.0.8" />
  <PackageReference Include="Microsoft.EntityFrameworkCore.SqlServer" Version="5.0.8" />
  <PackageReference Include="Microsoft.EntityFrameworkCore.Tools" Version="5.0.8">
```

First I have implemented this application by giving hard to get better understanding,

I have used cookie Authentication scheme, to create this,

Here is first method I have created using name identifier scheme,

```
//[HttpPost("login")]
//public async Task<IActionResult> validate(string username, string password,
string returnUrl)
//{
//    ViewData["ReturnUrl"] = returnUrl;
//    if (username == "AAA" && password == "123")
//    {

//        var claims = new List<Claim>();
//        claims.Add(new Claim("username", username));
//        claims.Add(new Claim(ClaimTypes.NameIdentifier, username));
//    }
//    //simply we can add any name here...
```

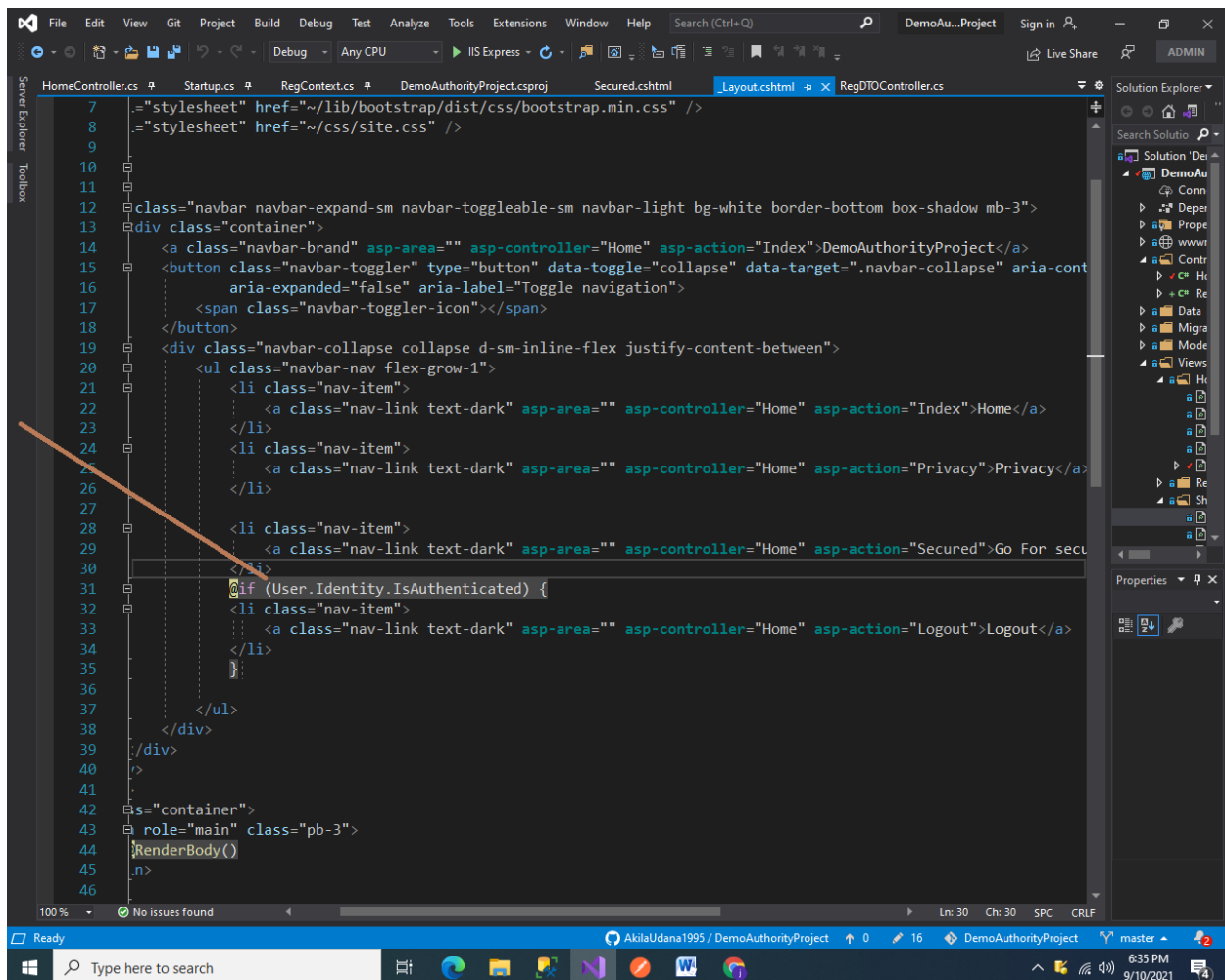
```
//      claims.Add(new Claim(ClaimTypes.Name, username));
//      //var ClaimsIdentity = new ClaimsIdentity(claims,
//      CookieAuthenticationDefaults.AuthenticationScheme);
//      var ClaimsPrincipal = new ClaimsPrincipal(ClaimsIdentity);
//      await HttpContext.SignInAsync(ClaimsPrincipal);
//      return Redirect(returnUrl);

//      // return View("Secured");
//  }

//      TempData["Error"] = "Error. username and/or Password is incorrect";
//      return View("login");// BadRequest();

//}
```

Here I have written small function to check user is authenticated or not,



Here I have just checked with hard coding by giving username as “AAA” and password as ‘123’,

I have made login method as Authorize using below code,

```
[HttpGet("login")]
public IActionResult login(string returnUrl)
{
    ViewData["ReturnUrl"] = returnUrl;
    return View();
}
```

Then I have created view for Login...

Then I have added this method to configure service. This method gets called by the runtime. Use this method to add services to the container.

```
//public void ConfigureServices(IServiceCollection services)
//{
//    services.AddControllersWithViews();
//    // services.AddAuthentication().AddCookie();
//    //
//    services.AddAuthentication(CookieAuthenticationDefaults.AuthenticationScheme).AddCookie(o
ptions =>
//    {

//        options.LoginPath = "/login";
//        options.AccessDeniedPath = "/denied";

//        options.Events = new CookieAuthenticationEvents()
//        {
//            OnSigningIn = async context =>
//            {
//                var principal = context.Principal;
//                if(principal.HasClaim(c=>c.Type==ClaimTypes.NameIdentifier))
//                {
//                    if(principal.Claims.FirstOrDefault(c=>c.Type==ClaimTypes.NameIdentifier).Value=="AAA")
//                    {
//                        var claimsIdentity = principal.Identity as
ClaimsIdentity;
//                        claimsIdentity.AddClaim(new Claim(ClaimTypes.Role,
"Admin"));
//                    }
//                }
//                await Task.CompletedTask;
//            },
//        },
//    }
```

```

//          OnSignedIn = async context =>
//          {
//              await Task.CompletedTask;
//          },

//          OnValidatePrincipal= async context=>
//          {
//              await Task.CompletedTask;
//          }

//      };

```

In here I have used cookie authentication events,

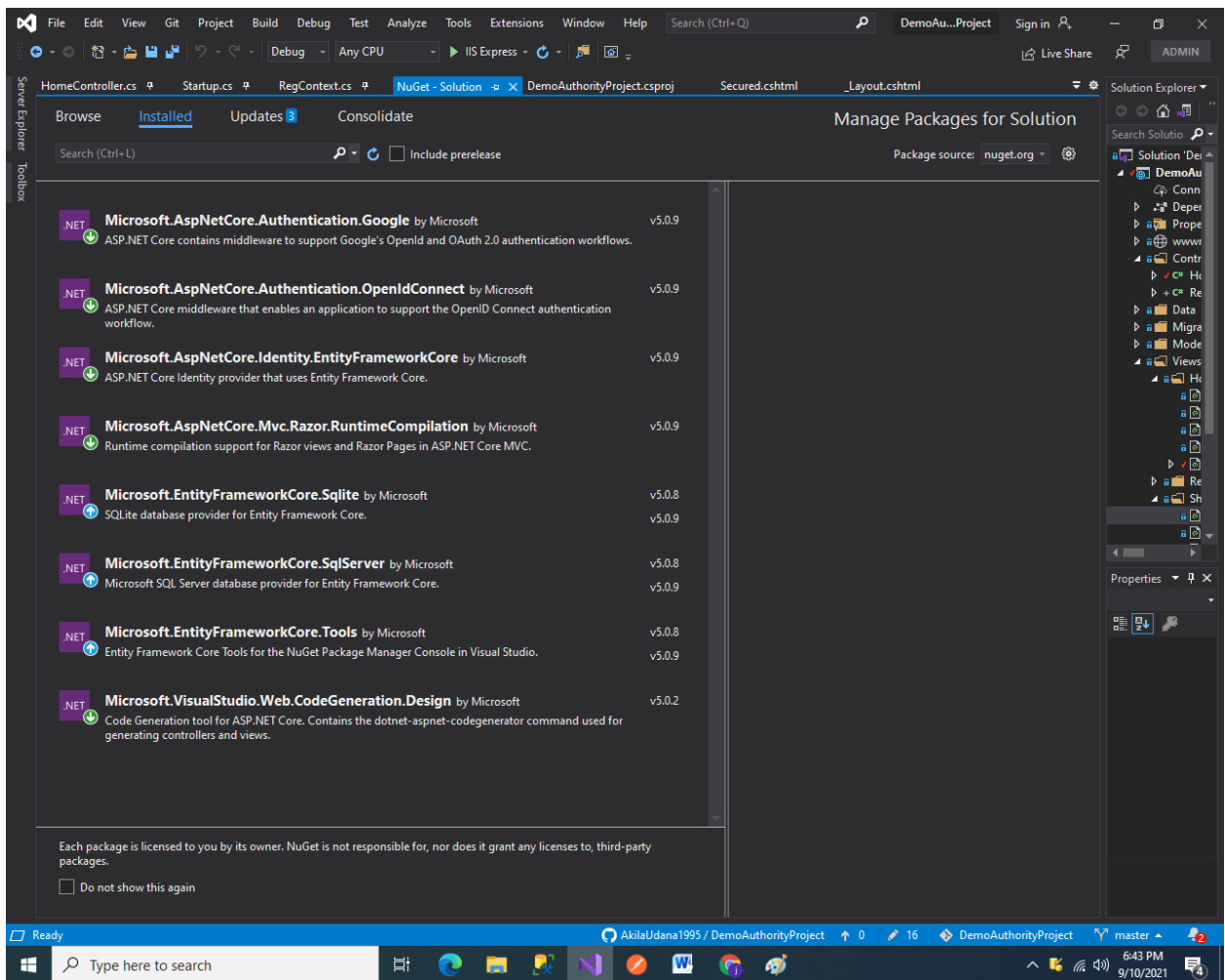
I have implemented them to below 3 states,

1. `OnSigningIn`
2. `OnSignedIn`
3. `OnValidatePrincipal`

This procedure is I just only used to make me better understand and also to check how this work,

After that I have created Google API project to integrate my sample application with that, that help me to use Auth token,

So firstly I have added Google authentication for my project via Nugget packages,



After that I have added Google options to my sample application,

```
// .AddGoogle(options =>
// {
//     options.ClientId = "78010272082-
7ru4pa608258v71fskpfq273ca113es7.apps.googleusercontent.com";
//     options.ClientSecret = "Pg94giXkz_mLAFhi0F1GBBbP";
//     options.CallbackPath = "/auth";
//     options.AuthorizationEndpoint += "?prompt=consent"; //this for select
user account after logout from the current context page
// }); //This is Google open handler that has made with oauth 2.0
```

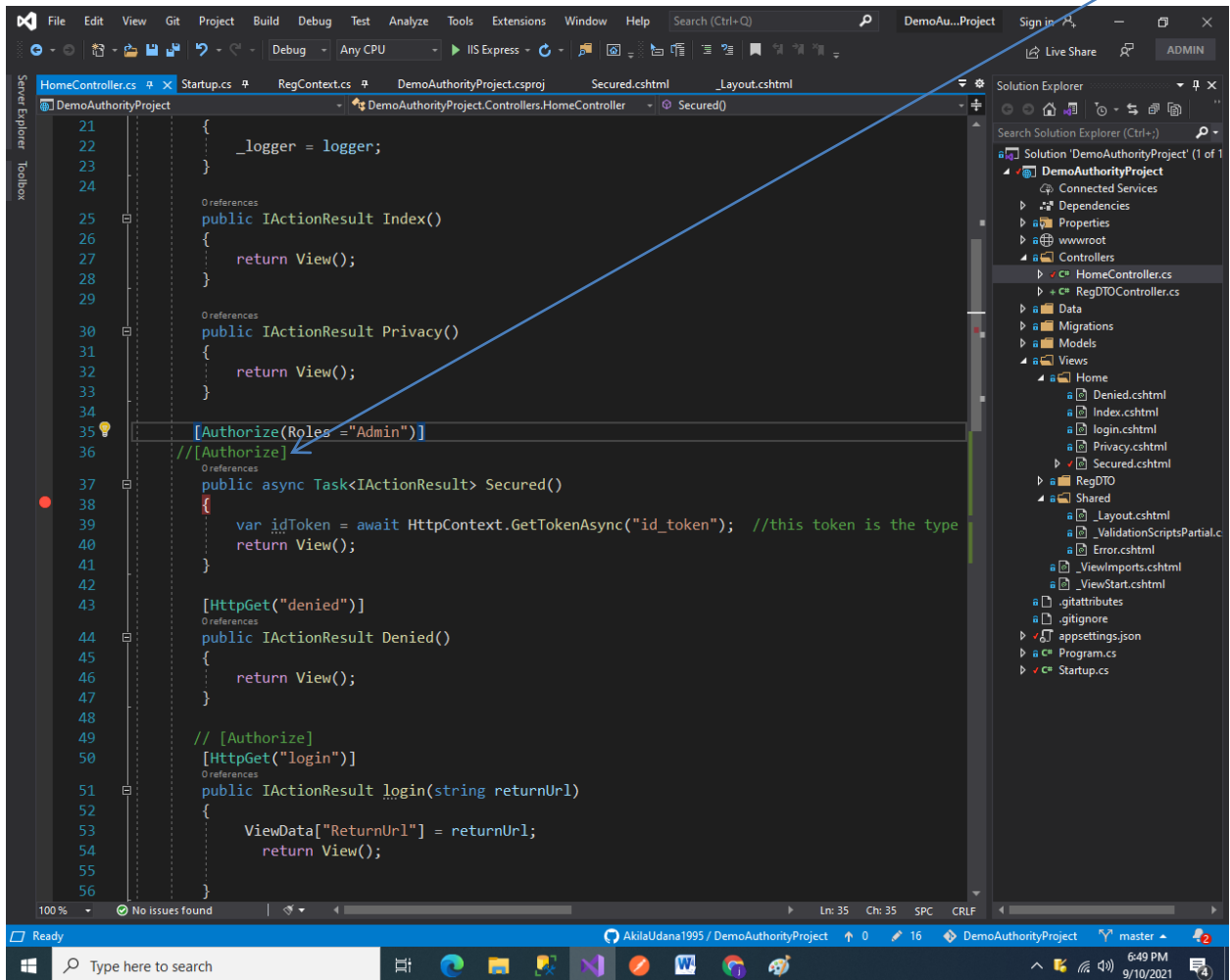
7ru4pa608258v71fskpfq273ca113es7.apps.googleusercontent.com is client id
Pg94giXkz_mLAFhi0F1GBBbP is client secret

After I have made application like that by adding this simple code,
If user logged through Google account is has to authorized,

```
[Authorize(Roles = "Admin")]
```

```
//[Authorize]
```

```
public async Task<IActionResult> Secured()  
{  
    var idToken = await HttpContext.GetTokenAsync("id_token"); //this token is  
the type of JWT or JOT  
    return View();  
}
```



That ensures that logged user is authorized or not,
For now I have commented “role based authorization”.

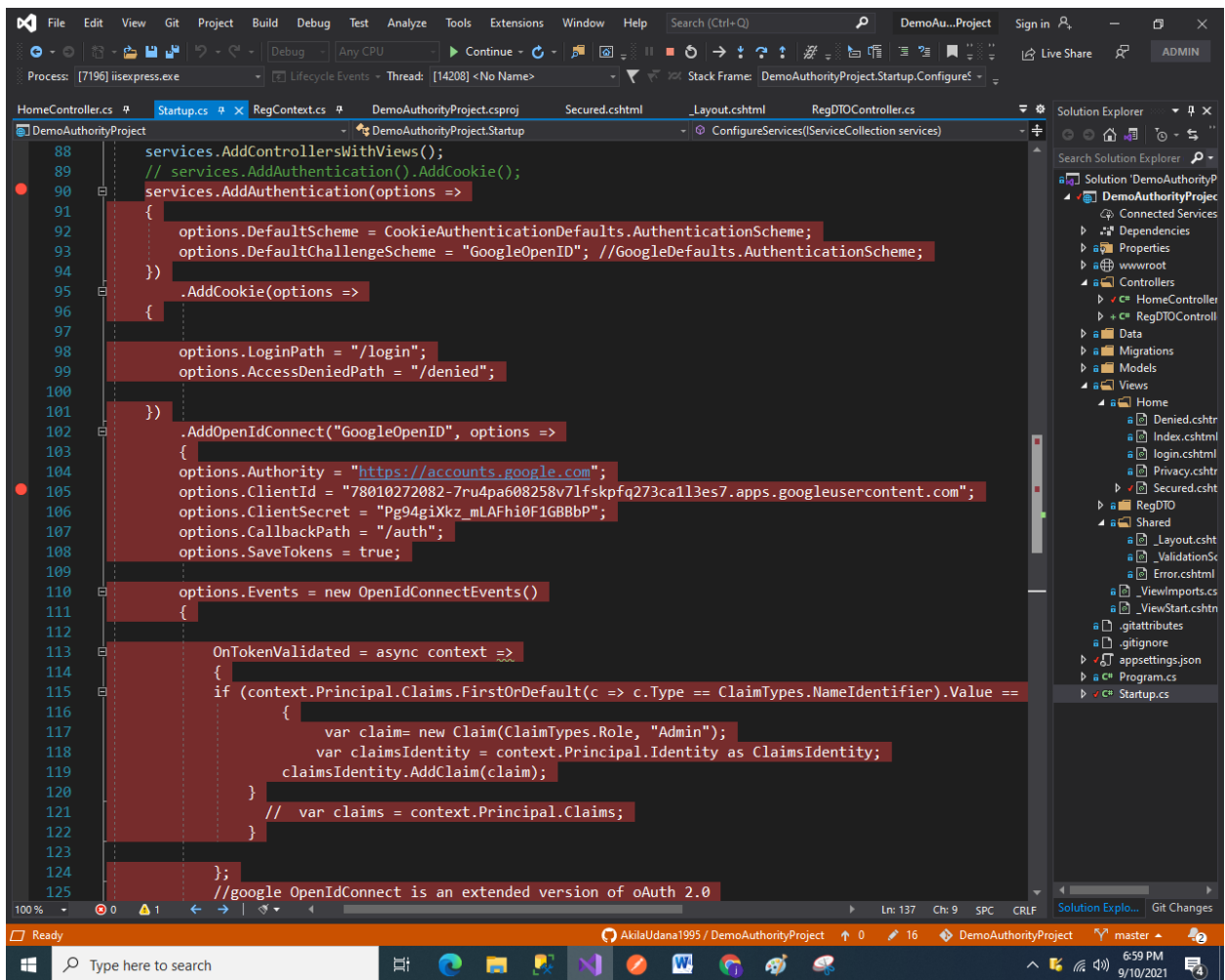
I created authentication handlers for this, I have used Google search for this,



The figure is also taken from Google.

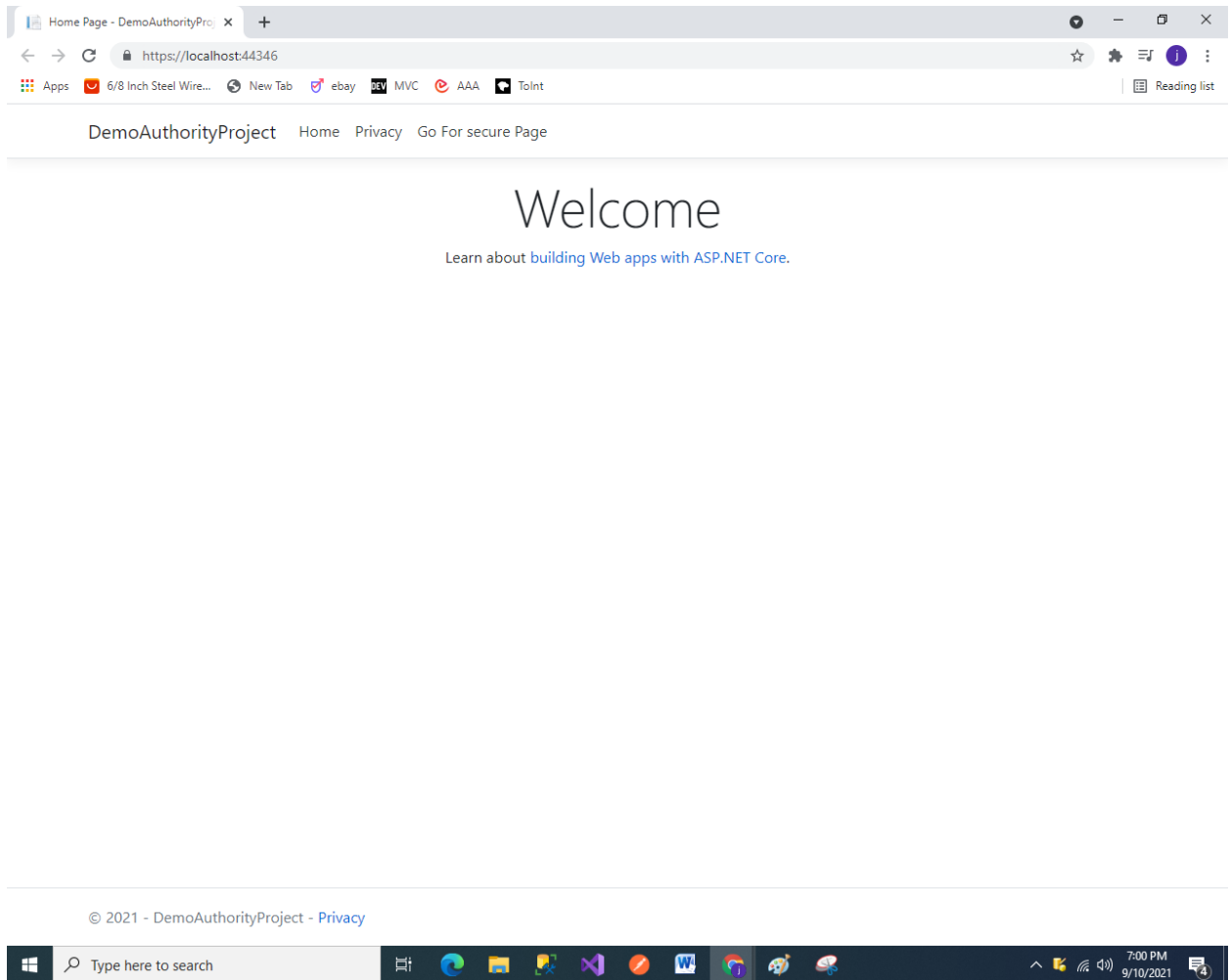
After that I also used generic openId connect for this, because that is extended version.

To that I have installed “Microsoft.AspNetCore.Authentication” then choose “OpenIdConnect.”

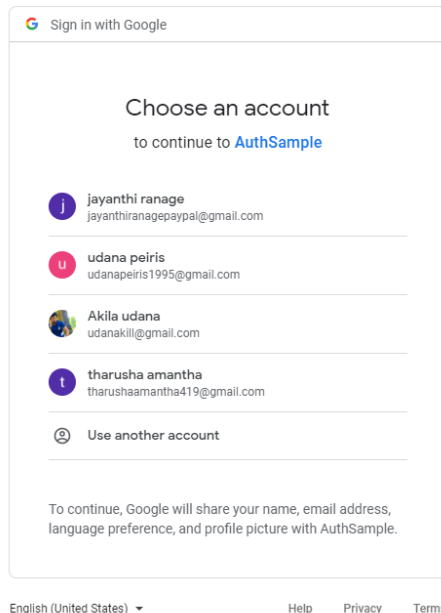
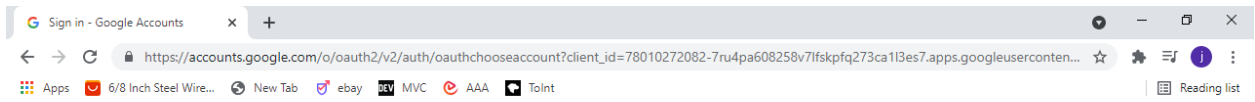


```
88 services.AddControllersWithViews();
89 // services.AddAuthentication().AddCookie();
90 services.AddAuthentication(options =>
91 {
92     options.DefaultScheme = CookieAuthenticationDefaults.AuthenticationScheme;
93     options.DefaultChallengeScheme = "GoogleOpenID"; //GoogleDefaults.AuthenticationScheme;
94 })
95     .AddCookie(options =>
96     {
97
98     options.LoginPath = "/login";
99     options.AccessDeniedPath = "/denied";
100
101     })
102     .AddOpenIdConnect("GoogleOpenID", options =>
103     {
104         options.Authority = "https://accounts.google.com";
105         options.ClientId = "78010272082-7ru4pa608258v71f5kpfq273call3es7.apps.googleusercontent.com";
106         options.ClientSecret = "Pg94giXkz_mLAFhi0F1GBBbP";
107         options.CallbackPath = "/auth";
108         options.SaveTokens = true;
109
110         options.Events = new OpenIdConnectEvents()
111         {
112             OnTokenValidated = async context =>
113             {
114                 if (context.Principal.Claims.FirstOrDefault(c => c.Type == ClaimTypes.NameIdentifier).Value ==
115                 {
116                     var claim= new Claim(ClaimTypes.Role, "Admin");
117                     var claimsIdentity = context.Principal.Identity as ClaimsIdentity;
118                     claimsIdentity.AddClaim(claim);
119                 }
120                 // var claims = context.Principal.Claims;
121             }
122         }
123     });
124     //google OpenIdConnect is an extended version of OAuth 2.0
125
```

After it walked me to this output.



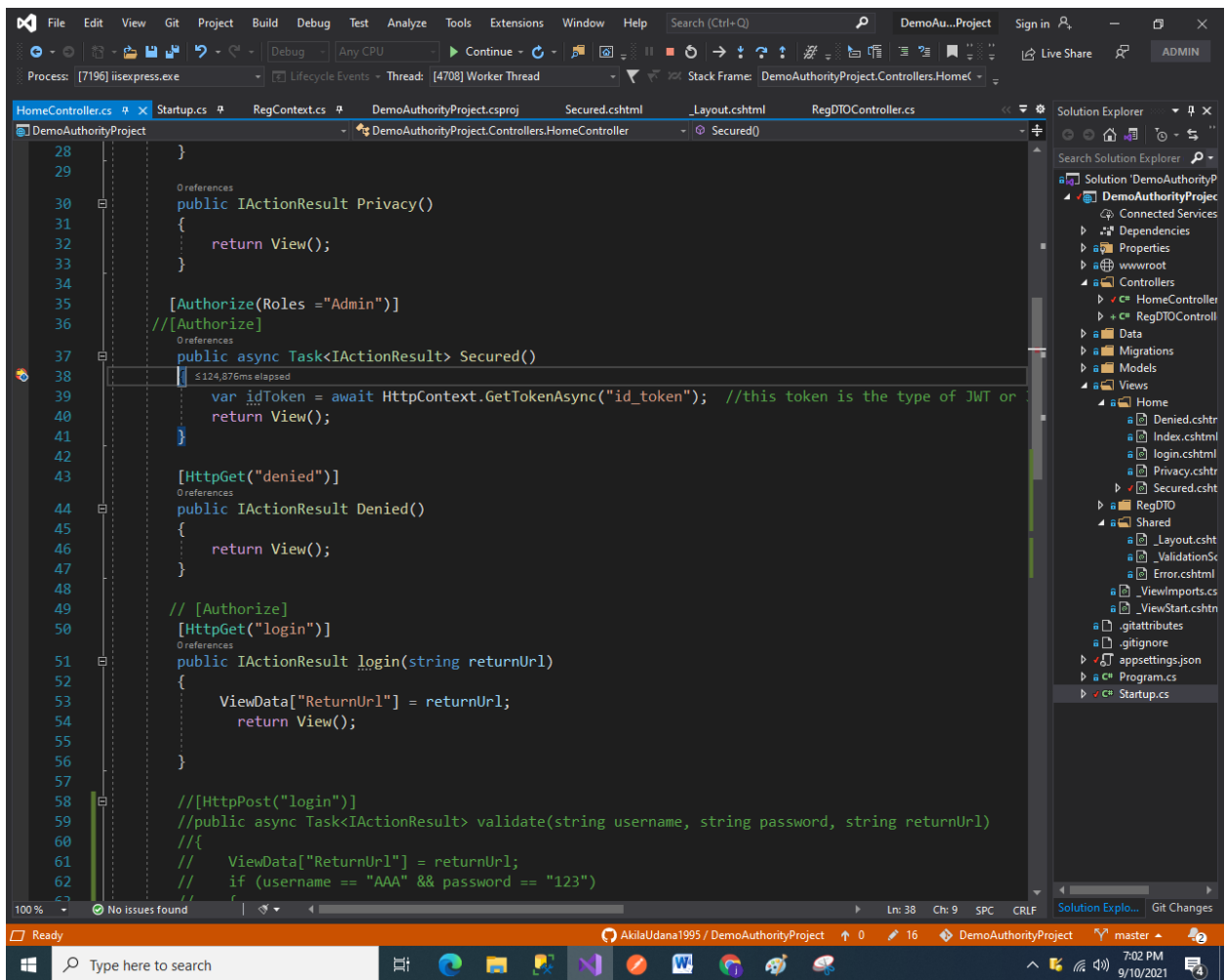
Now Google API is successfully connected and it asked me to select Google account to proceed,



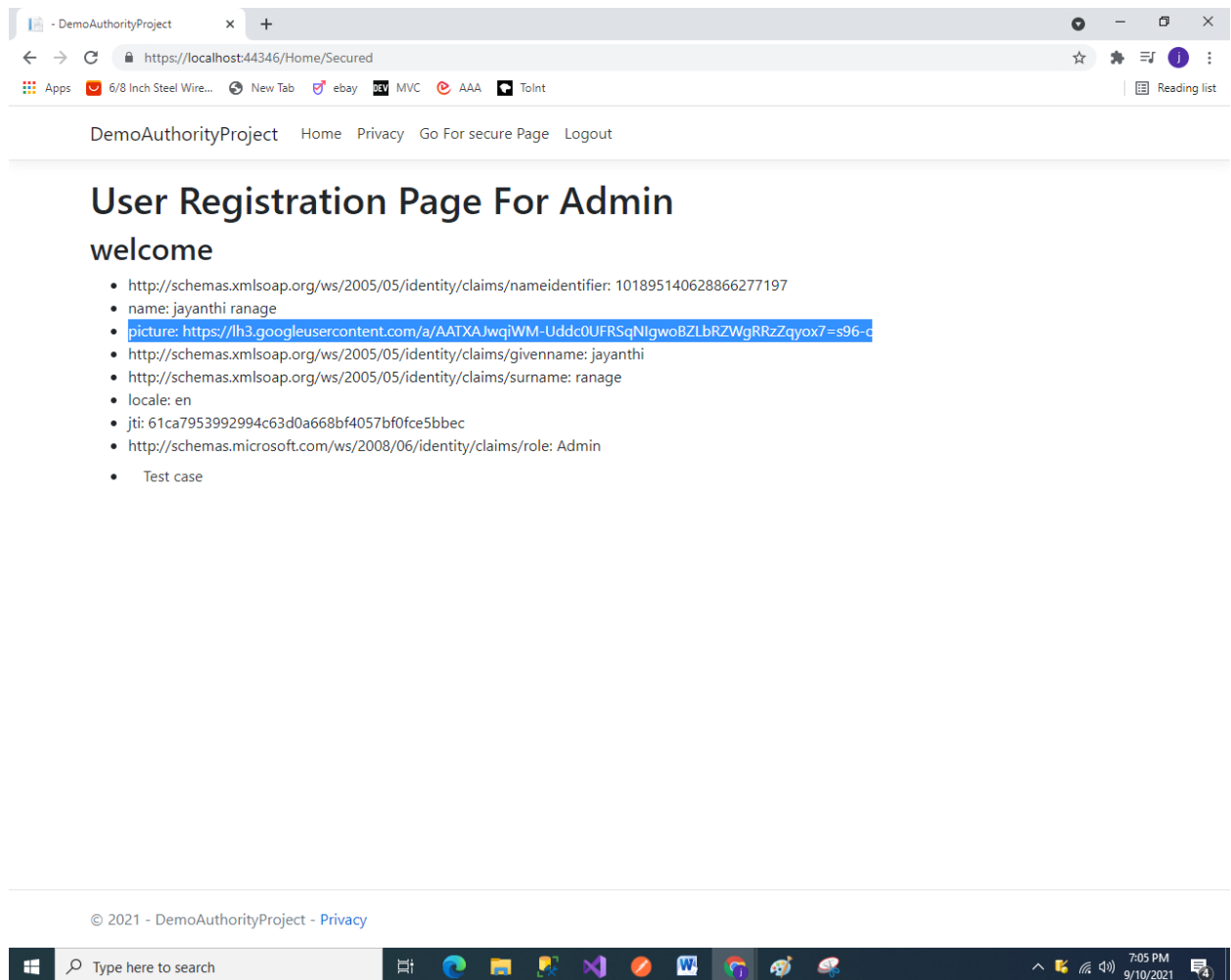
After I select Google account, the break point what I put at

```
[Authorize(Roles = "Admin")]
//[Authorize]
public async Task<IActionResult> Secured()
{
    var idToken = await HttpContext.GetTokenAsync("id_token"); //this token is
the type of JWT or JOT
    return View();
}
```

Is fired, that means my application is working for now.



Another advantage is openID is that shows our account picture as well,



picture: https://lh3.googleusercontent.com/a/AATXAJwqiWM-Uddc0UFRSqNlgwoBZLbRZWgRRzZqyox7=s96-c

here it shows as link,

if I do a brief explain about the content what I see here is

http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier:
101895140628866277197 this is name identifier for my case.

After that I have added a JWT token,

```
var idToken = await HttpContext.GetTokenAsync("id_token"); //this token is the type of JWT or JOT
```

I have searched this in internet and found,
That is used JSON notation to embedded Authentication and Authorization Claims in applications...

By further I have figure out that chat is made with major 3 parts,

1. Header
2. PayLoad
3. Signature

1.Header : this is containing information about the type of token. It could be JWT or JOT and also regarding the algorithm used to encrypt the signature.

2.Payload: This is contain the auth claims

3.Signature: this is an encrypted block of data which will be used to verify the authenticity of the token.

By further I have learned that these 3 parts are separated by (.).

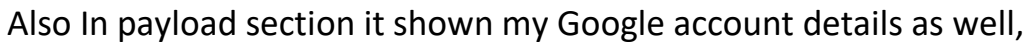
F33R8LwfO3yEGGfPoQKBICBvGjfiWw41K1ZXMS2u4ysjjK89tkFrXhcxUo3GEEpwVo2Rb8NXLHkxa
oTww8y6b7qvUDsfOalj2if2GJ9pm9MWliYBYbq8SAUqAB6h-
pnrs3EWwWelFTZTK6PiZ_M0tCHulbFaU-
KAAUzkmHLhxKanLGk0wGbSeGquAzr3s6oQX94xfPq0fxAoG--KM9AtRhxl_FsA”

This token is in the form of JWT or JOT.

Even these details might not be irrelevant for the task what you gave me, but before that I have not implement this , So I am documenting what I have learned,

These things were all new things for me...

So have gone to <https://jwt.io/> and pasted my encrypted token and got decrypted details,



Also In payload section it shown my Google account details as well,

PAYLOAD: DATA

```
{
  "iss": "https://accounts.google.com",
  "azp": "78010272082-7ru4pa608258v7lfskpfq273ca1l3es7.apps.googleusercontent.com",
  "aud": "78010272082-7ru4pa608258v7lfskpfq273ca1l3es7.apps.googleusercontent.com",
  "sub": "101895140628866277197",
  "nonce": "637668783842845333.NTJiMTR10GQtNzEwZi00ZDAyLWlzMjAtNWMzNzh1M2ZkM2M1YjVmY2E2YjYtMGI2Ny00NDI1LTg3MWItZDRhZDdmM2IzNGUw",
  "name": "jayanthi ranage",
  "picture": "https://lh3.googleusercontent.com/a/AATXA.lwqiWM-Uddc0UFRSqNIgwoBZLbRZWgRRzZqyox7=s96-c",
  "given_name": "jayanthi",
  "family_name": "ranage",
  "locale": "en",
  "iat": 1631281560,
  "exp": 1631285160,
  "jti": "6b9048d7915e807e32f9cb0041238e47eb6850b6"
}
```

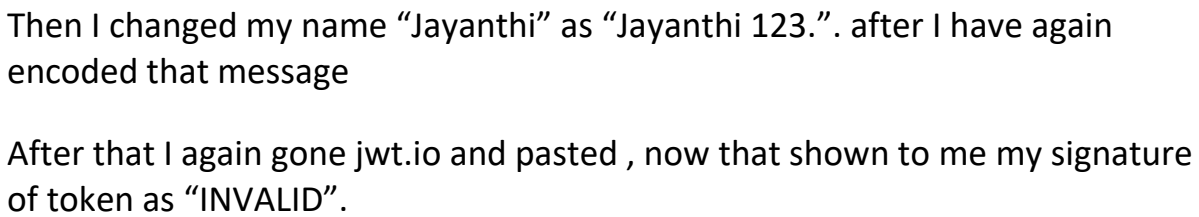


Also it has shown me as signature is verified now.

The screenshot shows the JWT.io website interface. At the top, there's a navigation bar with links like 'Debugger', 'Libraries', 'Introduction', 'Ask', and 'Get a T-shirt!'. The main content area is divided into two sections. The left section displays a decoded JWT token, showing its three parts separated by dots: a header, a payload, and a signature. The right section shows the token's structure, including the header, payload, and signature, with a text box for the private key. Below these sections, there's a 'Signature Verified' status and a 'SHARE JWT' button. At the bottom, there's a section titled 'Libraries for Token Signing/Verification' with a filter dropdown set to 'All'. A warning message is displayed: 'Warning: Learn more about critical vulnerabilities in JSON Web Token libraries with asymmetric keys.' The Windows taskbar is visible at the bottom of the browser window.

I have learned that 3 sections are separated by (.), then I have taken Payload section and walked through, <https://revoltjs.org/utilities> to get encode it again and see more,

I have decoded the encoded text and made simple change to my name in there,



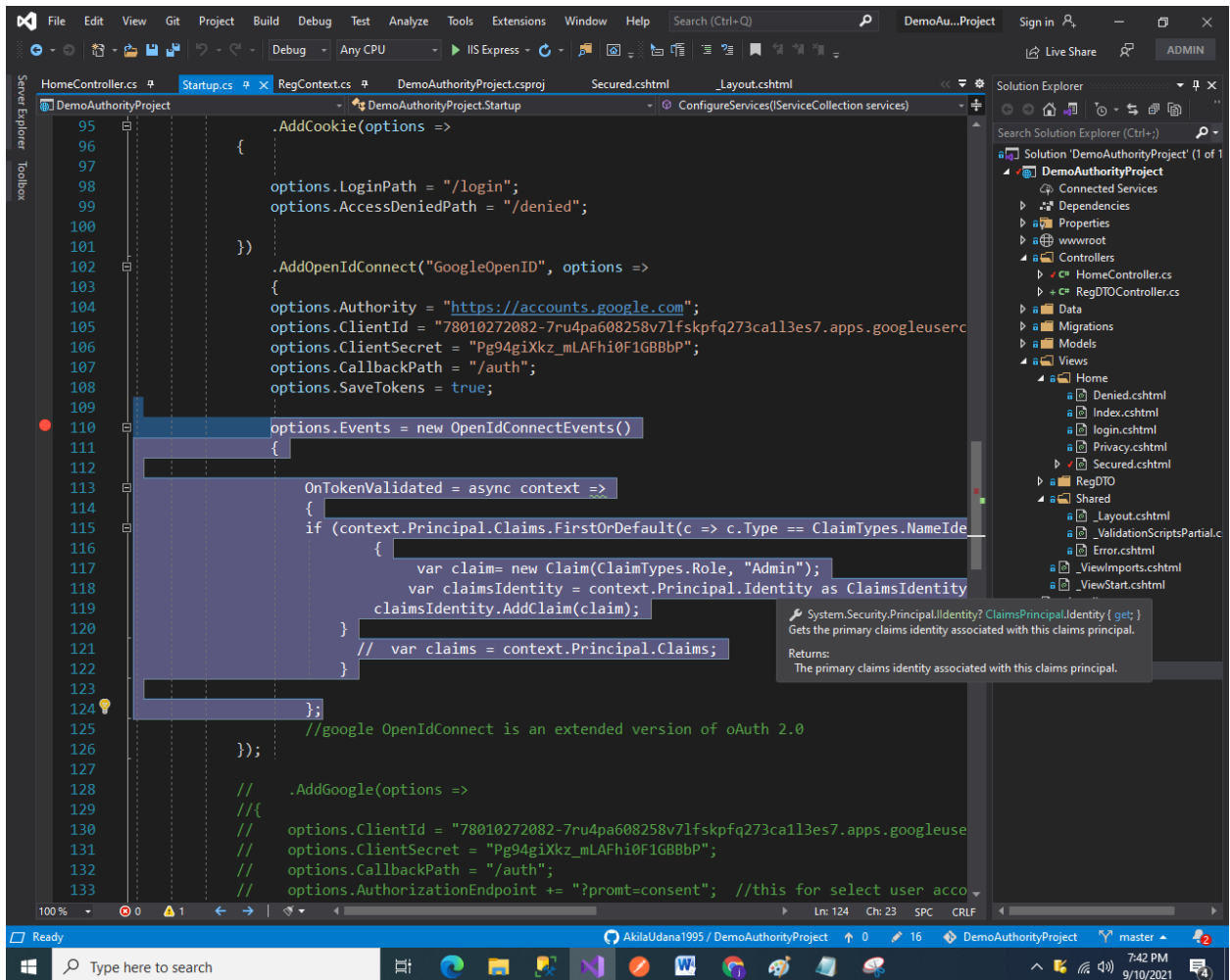
After that I again gone jwt.io and pasted , now that shown to me my signature of token as "INVALID".

The screenshot shows the JWT.io website interface. On the left, a long JWT token is displayed in a pink box. On the right, the 'VERIFY SIGNATURE' section shows the RSASHA256 algorithm and a public key. A red error message 'Invalid Signature' is shown below the token. The 'SHARE JWT' button is visible at the bottom right.

Finally in home controller I have added role is [Authentication] and modified it also in startup.cs like blow,

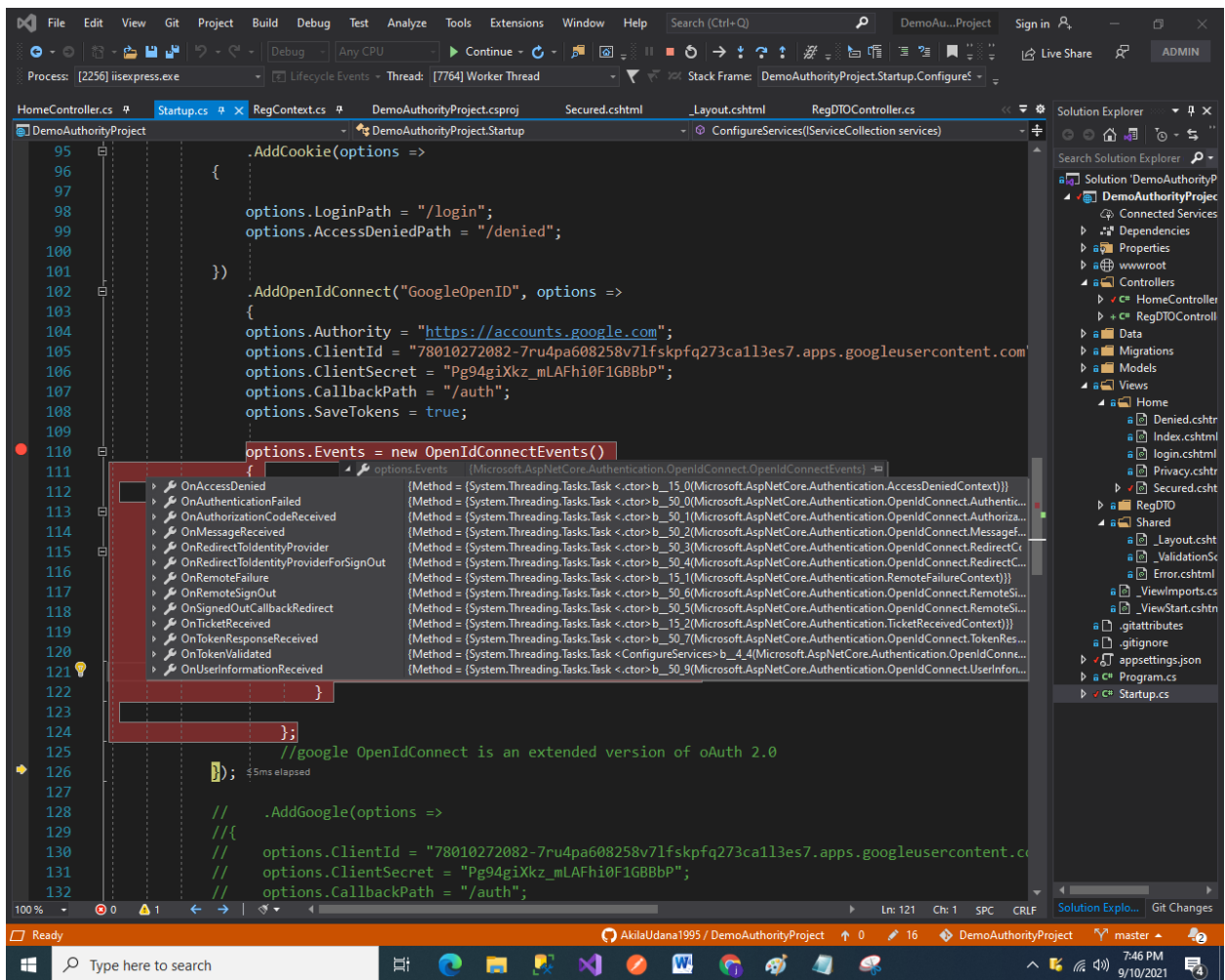
```
options.Events = new OpenIdConnectEvents()
{
    OnTokenValidated = async context =>
    {
        if (context.Principal.Claims.FirstOrDefault(c => c.Type ==
ClaimTypes.NameIdentifier).Value == "101895140628866277197")
        {
            var claim= new Claim(ClaimTypes.Role, "Admin");
            var claimsIdentity = context.Principal.Identity as
ClaimsIdentity;
            claimsIdentity.AddClaim(claim);
        }
        // var claims = context.Principal.Claims;
    }
}
```

```
};
```

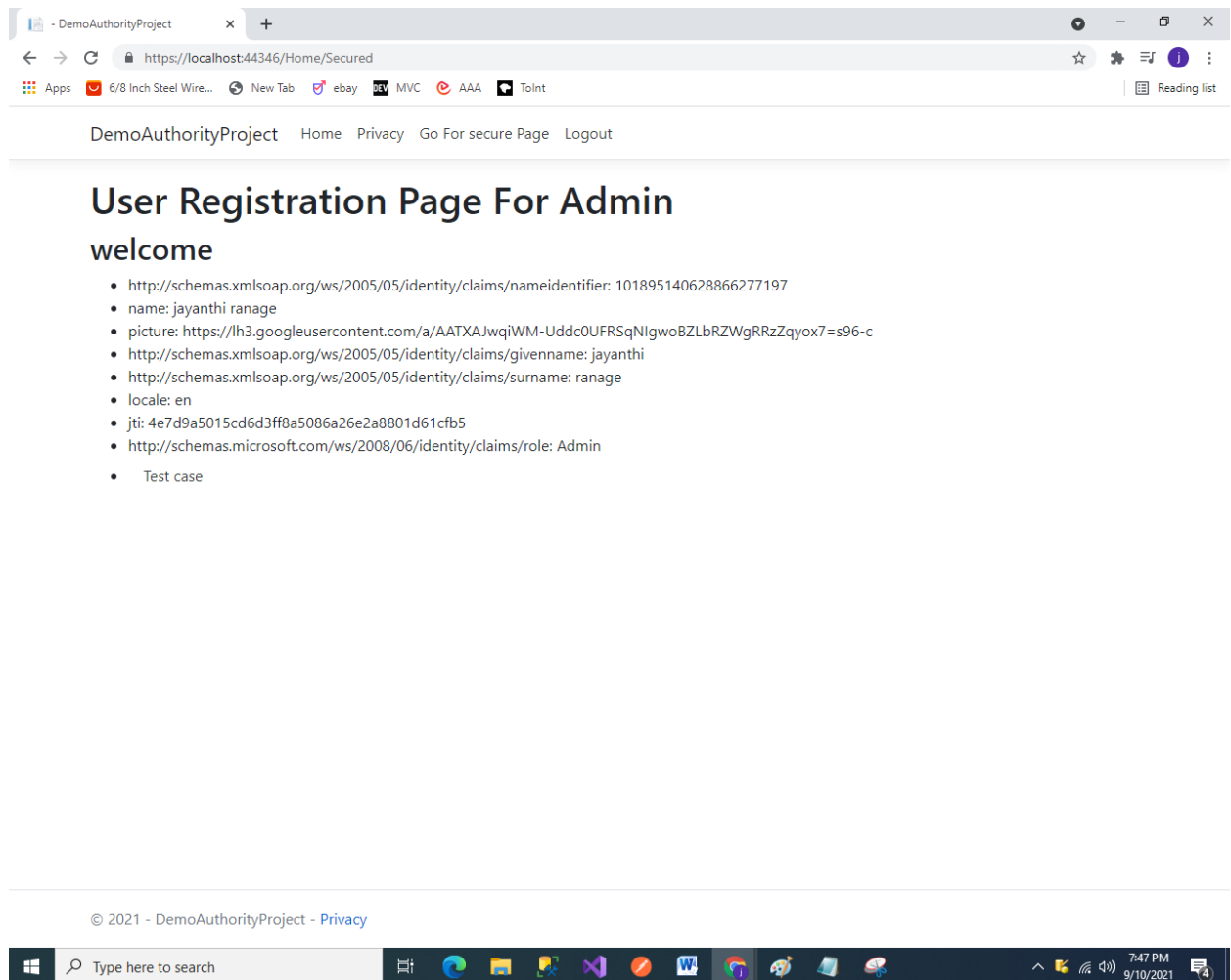


Since I have already know my name identifier for Google account named JAYANTHIRANAGE@PAYPAL.COM that is 101895140628866277197

I have putted it there,

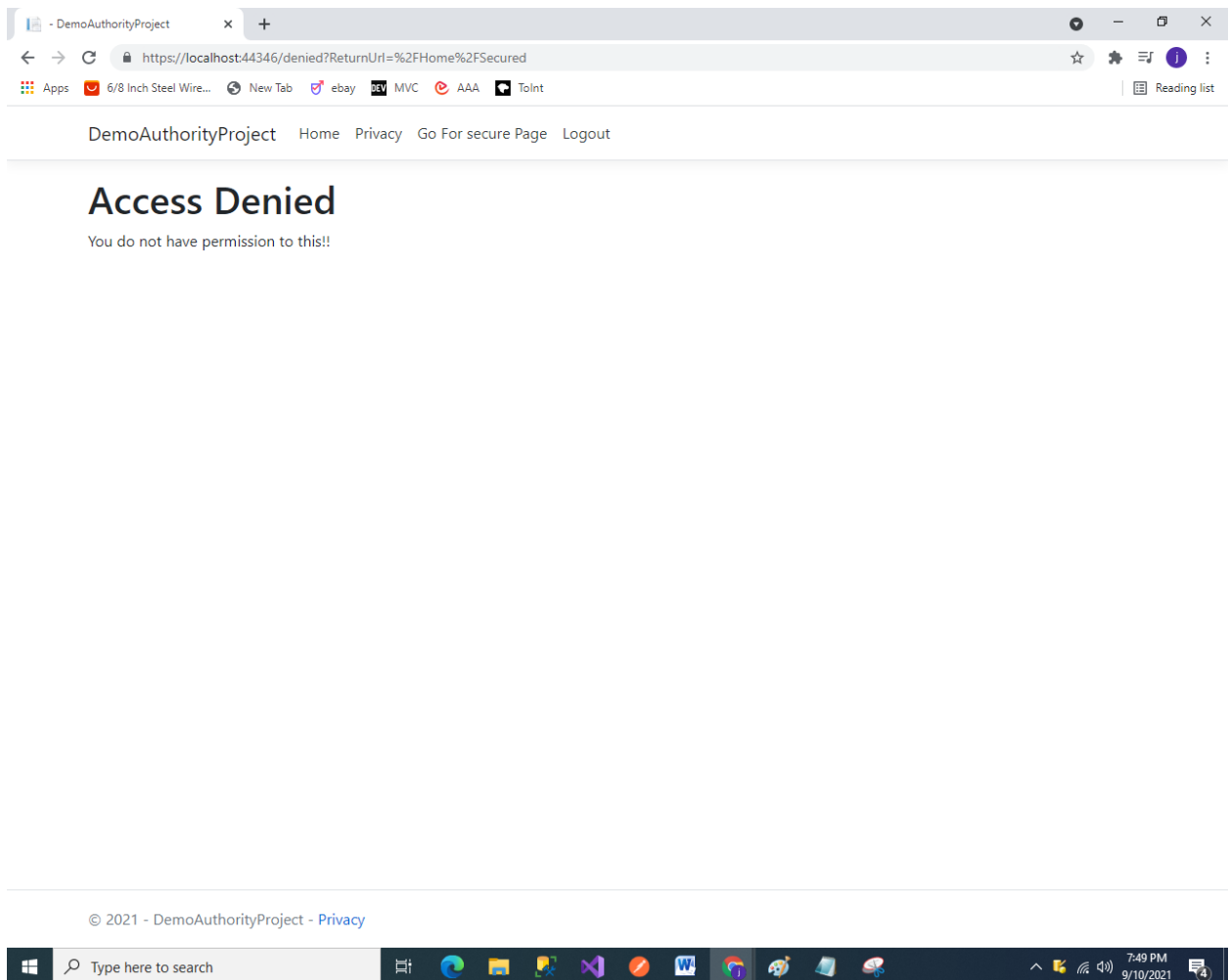


Since I have configured admin rights only for Google account called jayanthiranaage@paypal.com is allow admin access only for that.



Once I have logged out from that and tried to logged to system using another account to admin panel,

Since I have not given admin privileges to other accounts it showed me as below.



```
.AddCookie(options =>
{
    options.LoginPath = "/login";
    options.AccessDeniedPath = "/denied";
})
```

So my all functions are working..

Now I have implanted user authentication and authorization for admin, So now user is able to access system,

Now I have followed this Procedure to add/Update/View and Delete New User.

First I have created Model class to catch up details from front end application.


```

public class RegDTO
{
    [Key]
    public int UserID { get; set; }
    [Required(ErrorMessage = "This Filed is Required")]
    public string FirstName { get; set; }
    [Required(ErrorMessage = "This Filed is Required")]
    public string LastName { get; set; }
    // public int Email { get; set; }
    // public int Password { get; set; }
    public DateTime Bithdate { get; set; }

    public int Status { get; set; }
}

```

here I have used [key] attribute using ASP.NET core notations and also marked some fields as required as well,

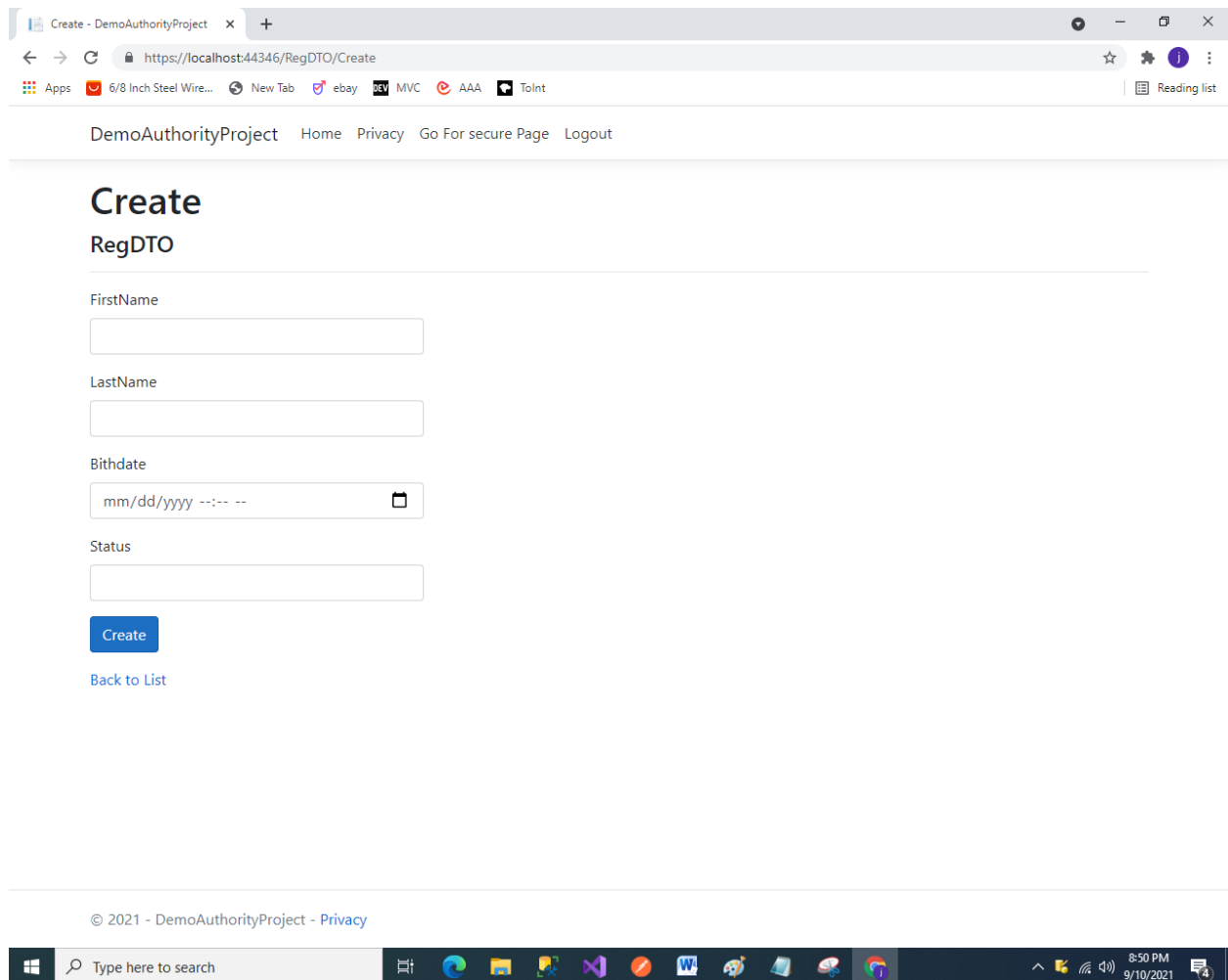
Then After I have created Context class for data base related queries and DB Set,

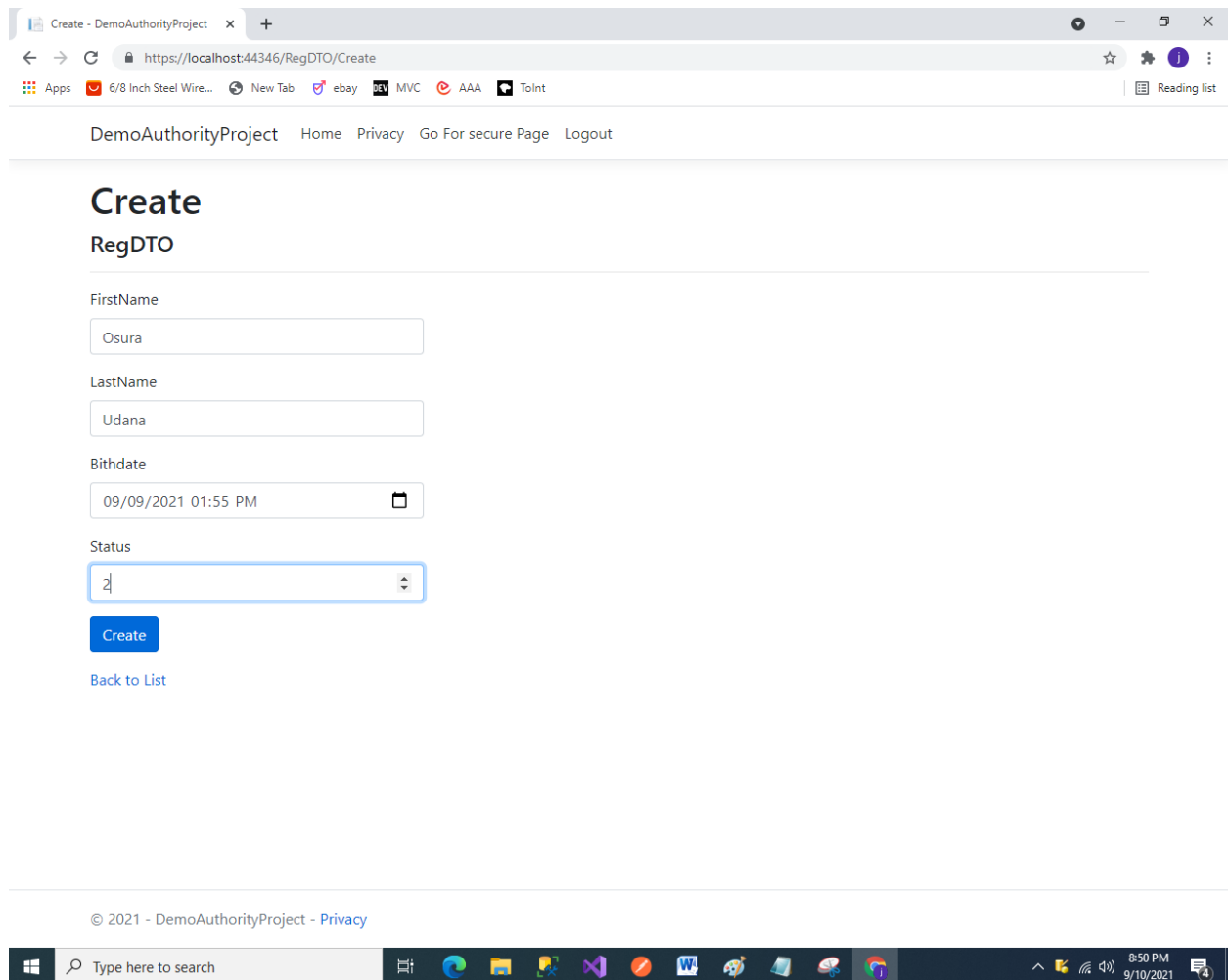
I have used MSEF Core for this,

Then I have made migration and updated data bases,

After I have created another controller with READ/WRITE actions using entity framework then scaffolds it with Views,

After Debug...for POST Method,





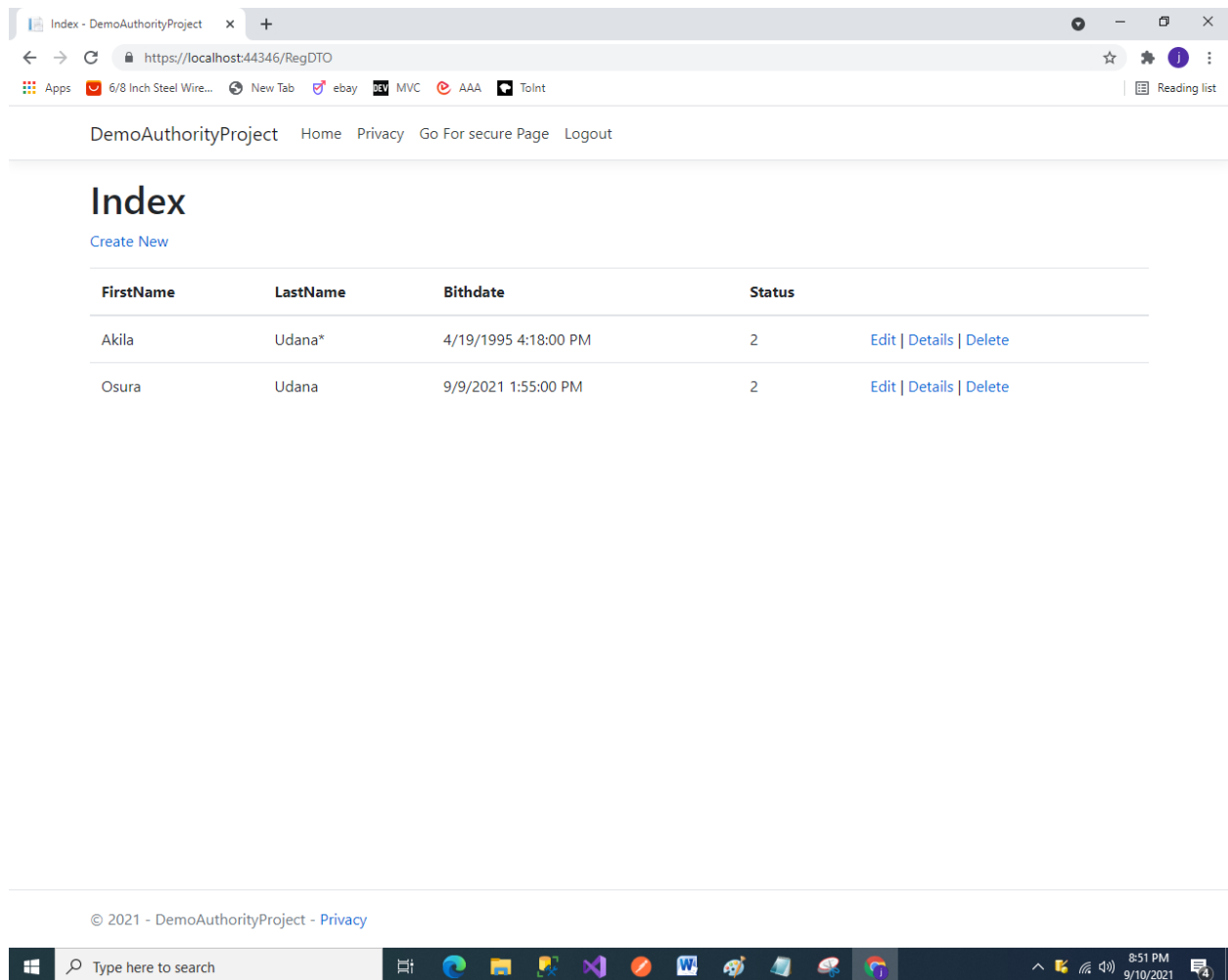
Visual Studio IDE showing the RegDTOController.cs file in the DemoAuthorityProject. The code is in C# and implements the Create and Edit methods for the RegDTO model.

```
51
52 // POST: RegDTO/Create
53 // To protect from overposting attacks, enable the specific properties you want to bind to.
54 // For more details, see http://go.microsoft.com/fwlink/?LinkId=317598.
55 [HttpPost]
56 [ValidateAntiForgeryToken]
57 public async Task<IActionResult> Create([Bind("UserID,FirstName,LastName,Bithdate,Status")] RegDTO regDTO)
58 {
59     if (ModelState.IsValid)
60     {
61         _context.Add(regDTO);
62         await _context.SaveChangesAsync();
63         return RedirectToAction("Index");
64     }
65     return View(regDTO);
66 }
67
68 // GET: RegDTO/Edit/5
69 public async Task<IActionResult> Edit(int? id)
70 {
71     if (id == null)
72     {
73         return NotFound();
74     }
75
76     var regDTO = await _context.iProducts.FindAsync(id);
77     if (regDTO == null)
78     {
79         return NotFound();
80     }
81     return View(regDTO);
82 }
83
84 // POST: RegDTO/Edit/5
85 // To protect from overposting attacks, enable the specific properties you want to bind to.
86 // For more details, see http://go.microsoft.com/fwlink/?LinkId=317598.
87 [HttpPost]
```

The Solution Explorer on the right shows the project structure, including the RegDTOController.cs file. A tooltip for the regDTO object is visible, showing the following values:

Property	Value
Bithdate	(9/9/2021 1:55:00 PM)
FirstName	"Osura"
LastName	"Udana"
Status	2
UserID	0

The status bar at the bottom indicates the file is at Line 59, Column 13, with 16 characters and a CRLF line ending. The system tray shows the time as 8:50 PM on 9/10/2021.



POST is Working now.

Also In SQL DB side it has also updated too.

I have added screen shots for other methods too.

Index - DemoAuthorityProject

+

https://localhost:44346/RegDTO

☆ ⚙ ⓘ ⋮

Apps 6/8 Inch Steel Wire... New Tab ebay MVC AAA Toint

Reading list

DemoAuthorityProject Home Privacy Go For secure Page Logout

Index

[Create New](#)

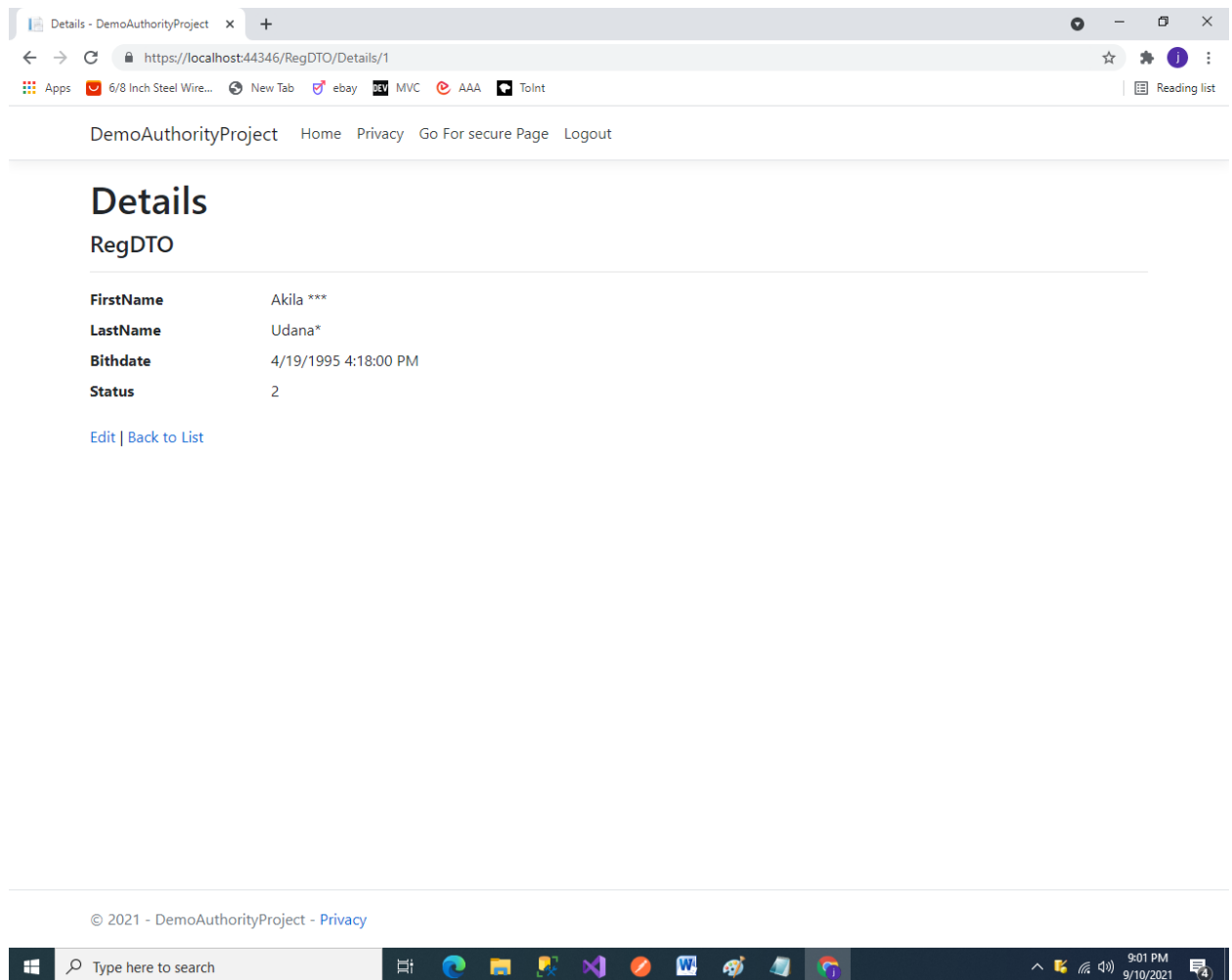
FirstName	LastName	Bitthdate	Status	
Akila ***	Udana*	4/19/1995 4:18:00 PM	2	Edit Details Delete
Osura	Udana	9/9/2021 1:55:00 PM	2	Edit Details Delete

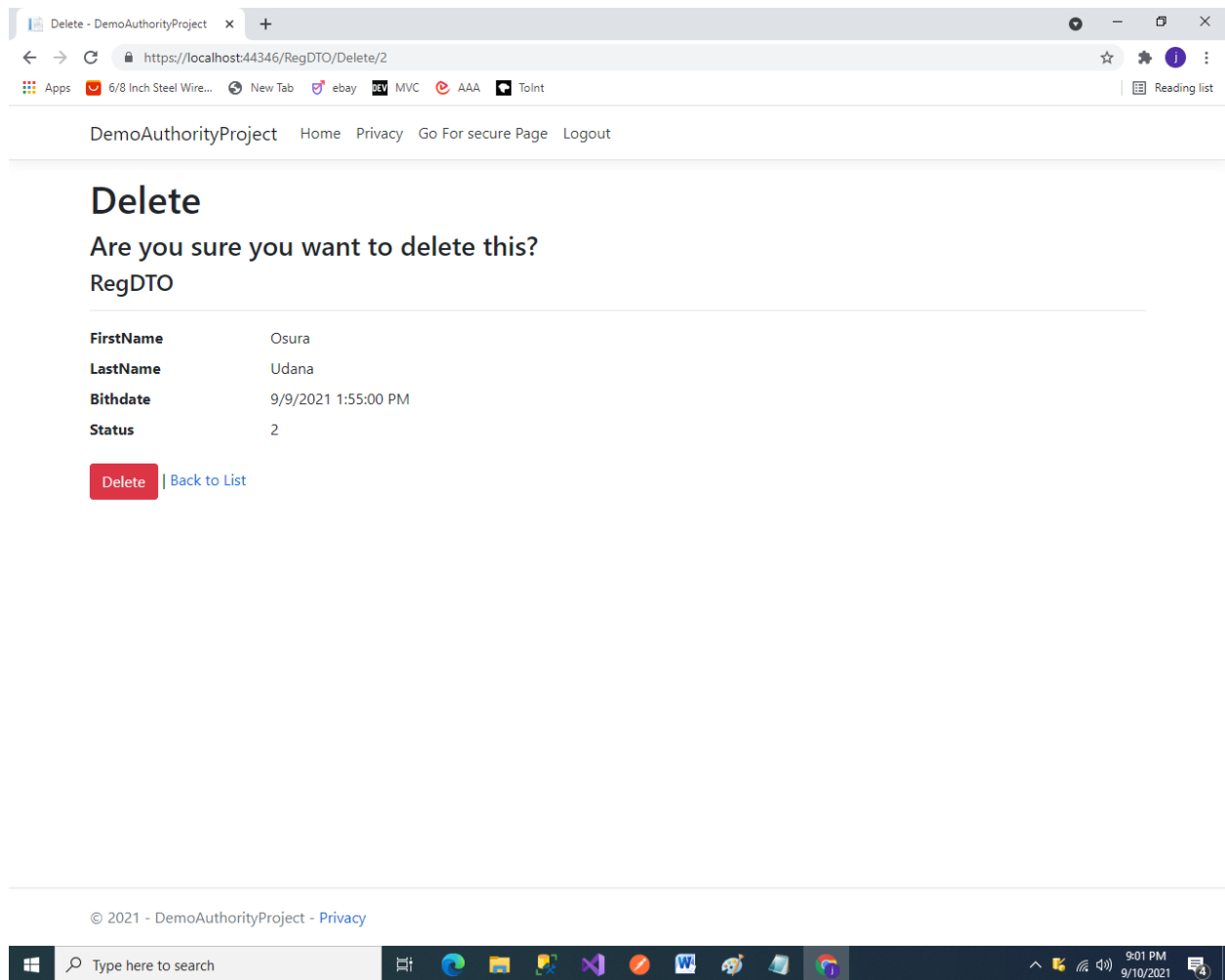
© 2021 - DemoAuthorityProject - [Privacy](#)

Windows Search Type here to search

9:00 PM
9/10/2021

4





Thank you!!

