

What's new with **ABP 8.0?**



Engincan VESKE

Software Engineer at Volosoft

DELI E CODE CREATE CELEBRATE CODE CREAT DEVELOP

DESIGN

ABP Framework 8.0 Highlights

https://abp.io/b/v80

- Upgraded to .NET 8.0
- Upgraded to Angular 17
- Dynamic Claims
- CDN Support for Bundling & Minification system
- Read-Only Repositories: Don't track changes by default
- Account Module: Set username after social login



Upgraded to .NET 8.0

Migration Guides

- Microsoft's Migrate from ASP.NET Core 7.0 to 8.0 documentation
- ABP Framework 7.x to 8.0
- ABP Commercial 7.x to 8.0



Upgraded to Angular 17

- Angular 17 <u>has been released on November 8</u>
- ABP 8.0 Angular UI is based on Angular 17



Dynamic Claims

Dynamic Claims allows you to **get the latest user claims** without the need to re-login each time.

The **Dynamic Claims** feature is used to dynamically generate claims for the user in each request. This was needed because claims-based authentication is used in the ASP.NET Core, and because it stores the claims in the cookie or token, and they are static, they don't change until the next re-login.

It's already been documented: https://docs.abp.io/en/abp/8.0/Dynamic-Claims



Dynamic Claims

Configure the

AbpClaimsPrincipalFactoryOptions and set the IsDynamicClaimsEnabled option as true

public override void ConfigureServices(ServiceConfigurationContext context)
{
 context.Services.Configure<AbpClaimsPrincipalFactoryOptions>(options =>
 {
 options.IsDynamicClaimsEnabled = true;
 });
}

Add the **DynamicClaims** middleware into the request pipeline 👉

```
public override void OnApplicationInitialization(ApplicationInitializationContext context)
{
    // Add this line before UseAuthorization.
    app.UseDynamicClaims();
    app.UseAuthorization();
    //...
}
```



CDN Support for Bundling & Minification System

```
Configure<AbpBundlingOptions>(options =>
    options.StyleBundles
        .Add(bundleName: "MyStyleBundles", configureAction: config:BundleConfiguration =>
            config
                .AddFiles("/styles/my-style-1.css")
                .AddFiles("/styles/my-style-2.css")
                .AddFiles("https://cdn.abp.io/bootstrap.css");
    options.ScriptBundles
        .Add(bundleName: "MyScriptBundles", configureAction: config:BundleConfiguration =>
            config
                .AddFiles("/scripts/my-script-1.js")
                .AddFiles("/scripts/my-script-2.js")
                .AddFiles("https://cdn.abp.io/bootstrap.js");
```

In this version, ABP Framework's <u>Bundling System</u> provides **CDN Support** for MVC / Razor Pages.

The system automatically recognizes the **external/CDN files** and places them as script tags on the page along with the bundled CSS/JS files.

```
<link rel="stylesheet" href="/__bundles/MyStyleBundles.EA8C28419DCA43363E9670973D4C0D15.css?_v=638331889644609730" />
<link rel="stylesheet" href="https://cdn.abp.io/bootstrap.css" />
<script src="/__bundles/MyScriptBundles.C993366DF8840E08228F3EE685CB08E8.js?_v=638331889644937120"></script>
<script src="https://cdn.abp.io/bootstrap.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script><
```

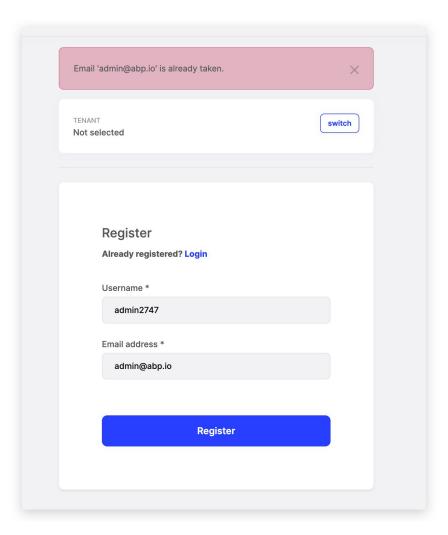


Read-Only Repositories: Don't track changes by default

In this version, ABP Framework provides read-only repository interfaces
(IReadOnlyRepository<T> Or
IReadOnlyBasicRepository<T>) to explicitly indicate that your purpose is to query data, but not change it.



Account Module: Set username after social login





Other News

- LDAP over SSL (LDAPS) setting has been added and recommended to establish a secure connection.
- **Object Mapping Enhancements** (supports mapping collection of objects for custom object mappers)

```
Once you implement IObjectMapper<User, UserDto>, ABP can automatically convert a collection of User objects to a collection of UserDto objects. The following generic collection types are supported:

• IEnumerable<T>
• ICollection<T>
• Collection<T>
• IList<T>
• List<T>
• List<T>
• T[] (array)
```

• Sending attachments with IEmailSender.QueueAsync() method



ABP Commercial 8.0 Highlights

- Suite: Creating Master/Detail Relationship
- Get profile picture from social/external logins
- Switch from Ocelot to YARP for the API Gateway (Microservice Solution Template)
- Password complexity indicators for MVC & Blazor Uls
- Export & Import Users as Excel/CSV



Suite: Creating Master/Detail Relationship

