


UTM DASHBOARD

Tasarım dersinde arařtırmalarımı tamamladıđımız ve bařlangıcını yaptıđımız UTM dashboard üzerine alıřmalarımız devam etti ve geliřtirdik ve bu srete gvenli yazılım geliřtirme kriterlerine dikkat ettik. Bařlıca řu zamana dek yaptıklarımız oturum sresi belirleme, giriř yanlıř denemesinde hesap kilitlenmesi, arayzde gncellemeler ve Super Admin rolne zel kullanıcıları ynetebileceđi panel eklenmiřtir. Buna ek olarak Asp.Net'in kullanıcı bilgilerine ek olarak İsim ve Soyisim bilgileri de eklenmiřtir.

ncelikle dashboard aılır aılmaz giriř ekranı bizi karřılamaktadır.



Giriř Yap

Giriř yapmak iin email & řifre girin.

Email

řifre

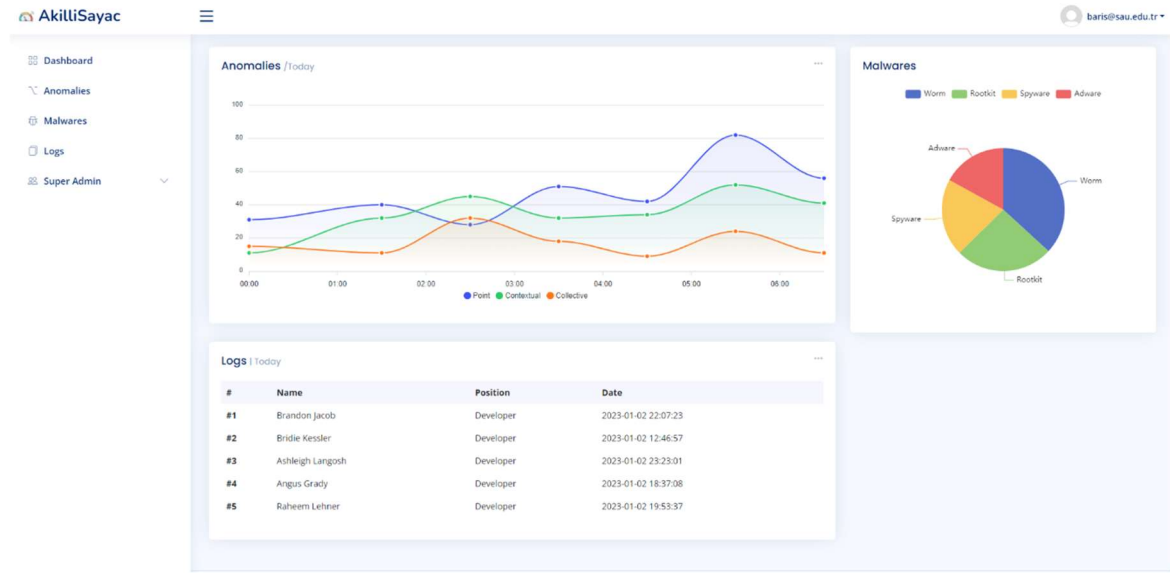
☐ Beni Unutma

Giriř Yap

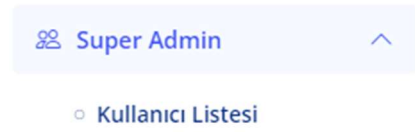
[řifremi Unuttum](#)

Designed by AkıllıSayac

Giriş yapıldıktan sonra anasayfa görünümü bu şekildedir.



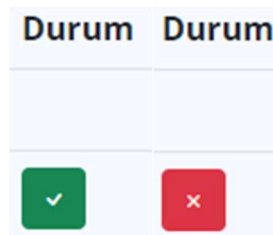
Sol panelde bulunan Super Admin seçeneği yalnızca Super Admin rolüne sahip kullanıcılara görülmekte olup Kullanıcı Listesi bölümünde kullanıcılarla ilgili işlemler yapılabilmektedir.



Kullanıcı listesi sayfasını görünümü bu şekildedir ve buradan yeni bir kullanıcı ekleme, var olan kullanıcının durumunu aktif veya inaktif yapabilme ve kullanıcı silme işlemleri yapılabilmektedir.

Kullanıcı Listesi				
Durum	İsim	Soyisim	Email	Kullanıcı Rolü
	Super	Admin	baris@sau.edu.tr	SuperAdmin
✓	Barış	Yılmaz	baris1@sau.edu.tr	Admin
Yeni Kayıt Oluştur				

Hesap durumu aktif işe yeşil, inaktif ise kırmızı görünmektedir. Butona basıldığında statü değişir.



Buna ek olarak kullanıcı 5 defa parolasını yanlış girdiği durumda da hesap otomatik olarak inaktif duruma geçer ve kapanır. Kullanıcının durumunun inaktif olduğu durumda giriş yapmaya çalışırsa aşağıdaki ekranla karşılaşmaktadır.



```
builder.Services.AddIdentity<AkilliSayacUser, IdentityRole>(options =>
{
    options.SignIn.RequireConfirmedAccount = false;
    options.SignIn.RequireConfirmedEmail = false;
    options.Lockout.MaxFailedAccessAttempts = 5;
    options.Lockout.AllowedForNewUsers = true;
    options.Lockout.DefaultLockoutTimeSpan = TimeSpan.FromDays(1000);
});
```


Bunlara ek olarak oturum açıldıktan sonra herhangi bir işlem yapılmadığı durumda oturum 5 dakika sonra kapanmakta ve auth cookie kendini imha etmektedir.

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly
1P_JAR	2023-03-23-15	.gstatic.com	/	2023-04-22T15:25:07.274Z	19	
AspNetCore.Identity.Application	CFDJ8NZDUVRSDhGwDBb0cY2gcOzGu7nVbVl8_kQwK-9PaZVEUnfOp3tmelDddpM7...	localhost	/	2023-04-01T15:34:33.410Z	890	✓
AspNetCore.Antiforgery.n-0DvDIMgro	CFDJ8NZDUVRSDhGwDBb0cY2gcOzGu7nVbVl8_kQwK-9PaZVEUnfOp3tmelDddpM7...	localhost	/	Session	190	✓

İlgili işlevi yerine getiren kod aşağıdaki gibidir. Cookie'nin MaxAge'i de belirlediğimiz expire zamanı kadar olacak şekilde düzenlenmiştir ki oturumdan atıp cookie hala aktif durumda bulunup güvenlik zaafiyetine sebep olmasın. Bunun haricinde HttpOnly bayrağı da set edilmiştir.

```
builder.Services.ConfigureApplicationCookie(options =>
{
    options.Cookie.Name = "AspNetCore.Identity.Application";
    options.ExpireTimeSpan = TimeSpan.FromMinutes(5);
    options.Cookie.MaxAge = options.ExpireTimeSpan;
    options.SlidingExpiration = true;
    options.Cookie.HttpOnly = true;
});
```

SuperAdmin yeni bir kullanıcı eklenmek istediğinde ise aşağıdaki ekranla karşılaşılır.



Kayıt Oluştur

Üye kaydı oluştur.

İsim

Soyisim

Email

Şifre

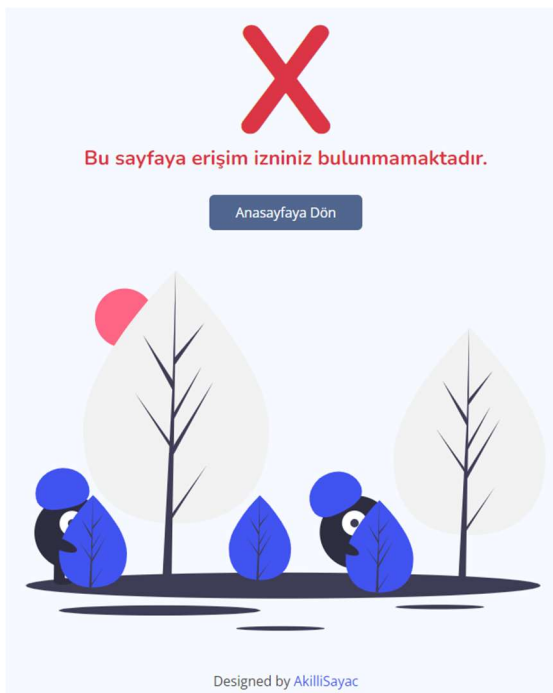
Şifreyi Doğrula

Rol

Kayıt Oluştur

Designed by AkilliSayac

Ayrıca Admin rolüne sahip kullanıcının veya Anonymous birinin yetkili bir sayfaya erişmeye çalıştığında karşılaçağı ekran şu şekildedir.



Bunlara ek olarak direkt olarak yetkisiz bir biçimde kayıt olma sayfası bulunmadığından dolayı bu işlemi ilk uygulama açıldığında yerine getirmekteyiz. Öncelikle SuperAdmin ve Admin rolleri sonrasında ise SuperAdmin rolüne sahip, bilgilerini manuel girdiğimiz bir kullanıcı oluşturularak default bir hesaba sahip olmuş oluyoruz. Bu işlemi de ilgili kod ile yerine getiriyoruz.

```
public static void SeedUsers(UserManager<AkilliSayacUser> userManager)
{
    if(userManager.FindByEmailAsync("baris@sau.edu.tr").Result == null)
    {
        var user = new AkilliSayacUser
        {
            FirstName = "Super",
            LastName = "Admin",
            UserName = "baris@sau.edu.tr",
            Email = "baris@sau.edu.tr",
            RoleName = "SuperAdmin",
            EmailConfirmed = true
        };
        var result = userManager.CreateAsync(user, "P@ssword1").Result;
        if (result.Succeeded)
        {
            userManager.AddToRoleAsync(user, "SuperAdmin").Wait();
        }
    }
}
```