**مدیریت جلسه کاربران در نرم‌افزارهای ویندوزی با C#**

**مقدمه**

مدیریت جلسه کاربران (User Session Management) یکی از مهم‌ترین جنبه‌های طراحی نرم‌افزارهای مدرن است. این فرآیند شامل شناسایی کاربران، نگهداری اطلاعات آن‌ها در طول استفاده از برنامه و پایان دادن به جلسات در زمان مناسب می‌شود. این مقاله به بررسی مفاهیم، پیاده‌سازی و نکات مرتبط با مدیریت جلسه در نرم‌افزارهای ویندوزی با استفاده از زبان C# می‌پردازد.

**ساختار اصلی سیستم مدیریت جلسه**

در این پروژه، ما از سه بخش اصلی استفاده کرده‌ایم:

1. **کلاس UserSession**: این کلاس برای نگهداری اطلاعات مربوط به کاربران ایجاد شده است. این اطلاعات شامل شناسه کاربر، نام کاربری و نقش کاربر است.
2. **کلاس SessionManager**: این کلاس یک مدیریت‌کننده استاتیک برای نگهداری و مدیریت جلسات کاربران است. از یک دیکشنری داخلی برای نگهداری اطلاعات جلسات استفاده می‌کند.
3. **کلاس UserContext**: این کلاس برای نگهداری اطلاعات کاربر جاری در سطح برنامه استفاده می‌شود و توسط Dependency Injection به فرم‌ها ارسال می‌شود.

**کدهای پیاده‌سازی**

**کلاس UserSession**

این کلاس اطلاعات اولیه هر کاربر را ذخیره می‌کند:

namespace AWMS.app.Forms.RibbonUser  
{  
 public class UserSession  
 {  
 public int UserID { get; set; }  
 public string Username { get; set; }  
 public int RoleID { get; set; }  
 }  
}

**کلاس SessionManager**

این کلاس مدیریت تمام جلسات کاربران را بر عهده دارد:

namespace AWMS.app.Forms.RibbonUser  
{  
 public static class SessionManager  
 {  
 private static readonly Dictionary<int, UserSession> \_sessions = new();

public static void AddSession(int userId, UserSession session)  
 {  
 \_sessions[userId] = session;  
 }

public static UserSession GetSession(int userId)  
 {  
 return \_sessions.ContainsKey(userId) ? \_sessions[userId] : null;  
 }

public static void RemoveSession(int userId)  
 {  
 \_sessions.Remove(userId);  
 }  
 }  
}

**کلاس UserContext**

برای نگهداری اطلاعات کاربر جاری، از این کلاس استفاده می‌شود:

namespace AWMS.app.Forms.RibbonUser  
{  
 public class UserContext  
 {  
 public int UserId { get; set; }  
 public string Username { get; set; }  
 public int RoleID { get; set; }  
 }  
}

**فرم ورود (frmLogin)**

**وظیفه فرم**

فرم ورود کاربران مسئول احراز هویت و شروع جلسات است. در اینجا کاربران نام کاربری و رمز عبور خود را وارد می‌کنند. پس از تأیید اعتبار، اطلاعات جلسه ایجاد شده و به سیستم اضافه می‌شود.

**کد فرم ورود**

using AWMS.dapper.Repositories;  
using DevExpress.XtraEditors;  
using Microsoft.Extensions.DependencyInjection;

namespace AWMS.app.Forms.RibbonUser  
{  
 public partial class frmLogin : XtraForm  
 {  
 private readonly IServiceProvider \_serviceProvider;

public frmLogin(IServiceProvider serviceProvider)  
 {  
 InitializeComponent();  
 \_serviceProvider = serviceProvider;

this.KeyDown += new KeyEventHandler(frmLogin\_KeyDown);  
 this.AcceptButton = btnEnter;  
 }

private async void btnEnter\_Click(object sender, EventArgs e)  
 {  
 string username = txtUserName.Text;  
 string passwordHash = txtPassword.Text; // فرض بر این است که تابع HashPassword وجود دارد

var user = await \_serviceProvider.GetService<IUserDapperRepository>()!.GetUserByUsernameAsync(username);

if (user != null && user.PasswordHash == passwordHash)  
 {  
 var session = new UserSession  
 {  
 UserID = user.UserID,  
 Username = user.Username,  
 RoleID = user.RoleID  
 };

SessionManager.AddSession(user.UserID, session);

var scope = \_serviceProvider.CreateScope();  
 var serviceProvider = scope.ServiceProvider;  
 var userContext = serviceProvider.GetRequiredService<UserContext>();  
 userContext.UserId = user.UserID;  
 userContext.Username = user.Username;  
 userContext.RoleID = user.RoleID;

var mainForm = serviceProvider.GetRequiredService<frmMain>();  
 mainForm.MdiParent = this.MdiParent; // تنظیم فرم اصلی به عنوان والد  
 mainForm.Show();  
 this.Hide();  
 }  
 else  
 {  
 error.Text = "The username or password is incorrect.";  
 }  
 }

private void simpleButton1\_Click(object sender, EventArgs e)  
 {  
 txtPassword.Text = "";  
 txtUserName.Text = "";  
 txtUserName.Focus();  
 }

private void frmLogin\_KeyDown(object sender, KeyEventArgs e)  
 {  
 if (e.KeyCode == Keys.Enter)  
 {  
 btnEnter\_Click(sender, e);  
 }  
 }

private void button1\_Click(object sender, EventArgs e)  
 {  
 txtPassword.Text = "123";  
 txtUserName.Text = "admin";  
 }  
 }  
}

**فرم اصلی (frmMain)**

**وظیفه فرم**

فرم اصلی برنامه محلی است که کاربران پس از ورود به برنامه با آن کار می‌کنند. هنگام بسته شدن این فرم، جلسه کاربر باید از سیستم حذف شود.

**کد فرم اصلی**

using AWMS.app.Forms.RibbonUser;  
using System;  
using System.Windows.Forms;

namespace AWMS.app.Forms  
{  
 public partial class frmMain : Form  
 {  
 private readonly UserContext \_userContext;

public frmMain(UserContext userContext)  
 {  
 InitializeComponent();  
 \_userContext = userContext;

this.FormClosing += frmMain\_FormClosing;  
 }

private void frmMain\_FormClosing(object sender, FormClosingEventArgs e)  
 {  
 if (\_userContext != null)  
 {  
 SessionManager.RemoveSession(\_userContext.UserId);  
 }

Application.Exit();  
 }  
 }  
}

**نتیجه‌گیری**

این سیستم مدیریت جلسه کاربران به ما کمک می‌کند تا اطلاعات کاربران را به صورت امن و ساختاریافته نگهداری کنیم. با پیاده‌سازی این ساختار، می‌توان تجربه کاربری بهتری فراهم کرد و امنیت داده‌ها را ارتقا داد. این روش به‌ویژه در نرم‌افزارهای سازمانی که مدیریت کاربران حیاتی است، بسیار کارآمد است.

برای بهبود بیشتر این سیستم، می‌توان از روش‌های رمزنگاری پیشرفته برای حفاظت از اطلاعات استفاده کرد و همچنین گزارش‌گیری دقیق از جلسات کاربران را به سیستم اضافه نمود.