



Aula 05 – Menu Flyout em páginas Shell – Layout, Estilos e Cores

1. Atualize a *Solution* com as seguintes pastas (clique com o botão direito na pasta Views → add → new Folder) e *ContentPages* (clique com o botão direito nas pastas → add → new item → content page).



2. Abra a View AppShell.Xaml, remova a tag FlyoutBehaviour (1) e adicione o um template que exibirá no topo da página shell uma imagem e informações do App (2). O Layout (2) encontra-se copiável na sequência deste arquivo.

```
<Shell x:Class="AppRpgEtec.AppShell"
    xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    xmlns:local="clr-namespace:AppRpgEtec"
    FlyoutBehavior="Disabled"> 1
</Shell>

<Shell.FlyoutHeaderTemplate>
    <DataTemplate>
        <Grid>
            <Grid.RowDefinitions>
                <RowDefinition Height="{OnPlatform Android=130, iOS=200}"></RowDefinition>
                <RowDefinition Height="*"></RowDefinition>
                <RowDefinition Height="40"></RowDefinition>
            </Grid.RowDefinitions>
            <FlexLayout Grid.Row="0" Direction="Row" AlignItems="Center" >
                <Frame Padding="0" CornerRadius="40"
                    HeightRequest="80" WidthRequest="80" >
                    <Image Source="has_logo_transparente.svg"/>
                </Frame>
                <VerticalStackLayout Padding="10,0,0,0" Spacing="0">
                    <Label Text="App RPG Etec" FontAttributes="Bold" FontSize="Medium" />
                    <Label Text="Disciplina PAM-II"/>
                </VerticalStackLayout>
            </FlexLayout>
        </Grid>
    </DataTemplate>
</Shell.FlyoutHeaderTemplate>

<ShellContent Title="Home" ContentTemplate="{DataTemplate local:MainPage}" Route="MainPage" />
```



```
<Shell.FlyoutHeaderTemplate>
  <DataTemplate>
    <Grid>
      <Grid.RowDefinitions>
        <RowDefinition Height="{OnPlatform Android=130, iOS=200}"></RowDefinition>
        <RowDefinition Height="*"></RowDefinition>
        <RowDefinition Height="40"></RowDefinition>
      </Grid.RowDefinitions>
      <FlexLayout Grid.Row="0" Direction="Row" AlignItems="Center" >
        <Frame Padding="0" CornerRadius="40"
          HeightRequest="80" WidthRequest="80" >
          <Image/>
        </Frame>
        <VerticalStackLayout Padding="10,0,0,0" Spacing="0">
          <Label Text="App RPG Etec" FontAttributes="Bold" FontSize="Medium" />
          <Label Text="Disciplina PAM-II"/>
        </VerticalStackLayout>
      </FlexLayout>
    </Grid>
  </DataTemplate>
</Shell.FlyoutHeaderTemplate>
```

3. Adicione mais referências para as pastas das views que usaremos na próxima etapa.

```
<Shell x:Class="AppRpgEtec.AppShell"
  xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
  xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
  xmlns:viewsUsuarios="clr-namespace:AppRpgEtec.Views.Usuarios"
  xmlns:viewsPersonagens="clr-namespace:AppRpgEtec.Views.Personagens"
  xmlns:viewsDisputas="clr-namespace:AppRpgEtec.Views.Disputas"
  xmlns:local="clr-namespace:AppRpgEtec">

  <Shell.FlyoutHeaderTemplate>
    <DataTemplate>
```

4. Adicione uma imagem para o item de menu existe e adicione mais itens de menu para as páginas de listagem

```
  </DataTemplate>
</Shell.FlyoutHeaderTemplate>

  <ShellContent Title="Home" Icon="home.svg"
    ContentTemplate="{DataTemplate local:MainPage}" Route="MainPage" />

  <ShellContent Title="Usuários" Icon="menuusuarios.svg"
    ContentTemplate="{DataTemplate viewsUsuarios:ListagemView}" />

  <ShellContent Title="Personagens" Icon="menupersonagens.svg"
    ContentTemplate="{DataTemplate viewsPersonagens:ListagemView}" />

  <ShellContent Title="Disputas" Icon="menudisputas.svg"
    ContentTemplate="{DataTemplate viewsDisputas:ListagemView}" />

</Shell>
```



5. Insira as imagens da pasta da aula dentro da pasta Resources → Images.
6. Abra a classe UsuarioViewModel dentro da pasta ViewModels/Usuarios e confirme se a página que será chamada após o login será a página que contém o menu.

```
public async Task AutenticarUsuario()//Método para autenticar um usuário
{
    try
    {
        Usuario u = new Usuario();
        u.Username = Login;
        u.PasswordString = Senha;

        Usuario uAutenticado = await uService.PostAutenticarUsuarioAsync(u);

        if (!string.IsNullOrEmpty(uAutenticado.Token))
        {
            string mensagem = $"Bem-vindo(a) {uAutenticado.Username}.";

            //Guardando dados do usuário para uso futuro
            Preferences.Set("UsuarioId", uAutenticado.Id);
            Preferences.Set("UsuarioUsername", uAutenticado.Username);
            Preferences.Set("UsuarioPerfil", uAutenticado.Perfil);
            Preferences.Set("UsuarioToken", uAutenticado.Token);

            await Application.Current.MainPage
                .DisplayAlert("Informação", mensagem, "Ok");

            Application.Current.MainPage = new AppShell();
        }
        else
        {
            await Application.Current.MainPage
                .DisplayAlert("Informação", "Dados incorretos :(", "Ok");
        }
    }
}
```

- Execute o aplicativo para realizar os testes necessários.



7. Clique com o direito no projeto (C#) e adicione duas *content pages*: **AboutView.xaml** e **DescriptionView.xaml**.
8. Faça as melhorias abaixo para que possamos criar ícones dos menus no rodapé da página Shell (FlyOutItem - 1), para criar uma página de menu que contenha duas abas (2 - Tab), para criar um item de menu para sair do app que aparecerá apenas no menu lateral (3) e um rodapé para o menu (4 - Shell.FlyOutFooter).

```
</Shell.FlyoutHeaderTemplate>
1 <FlyoutItem FlyoutDisplayOptions="AsMultipleItems">
    <ShellContent Title="Home" Icon="home.svg"
        ContentTemplate="{DataTemplate local:MainPage}" Route="MainPage" />
    <ShellContent Title="Usuários" Icon="menuusuarios.svg"
        ContentTemplate="{DataTemplate viewsUsuarios:ListagemView}" />
    <ShellContent Title="Personagens" Icon="menupersonagens.svg"
        ContentTemplate="{DataTemplate viewsPersonagens:ListagemView}" />
    <ShellContent Title="Disputas" Icon="menudisputas.svg"
        ContentTemplate="{DataTemplate viewsDisputas:ListagemView}" />
2 <Tab Title="Info" Route="info" Icon="info.svg">
    <ShellContent Title="Sobre" Route="sobre" ContentTemplate="{DataTemplate local>AboutView}" />
    <ShellContent Title="Descrição" Route="desc" ContentTemplate="{DataTemplate local:DescriptionView}" />
</Tab>
3 </FlyoutItem>
<ShellContent Title="Sair" Icon="Exit.png" ContentTemplate="{DataTemplate viewsUsuarios:LoginView}"
    Shell.FlyoutBehavior="Disabled"/>
4 <Shell.FlyoutFooter>
    <Label x:Name="lblLogin" TextColor="Black" FontAttributes="Bold" HorizontalOptions="Center" />
</Shell.FlyoutFooter>
</Shell>
```

- Observe que o item de menu “sair” possui uma propriedade Shell.FlyoutBehaviour com valor false, para que quando o usuário clique em sair o menu não mais fique disponível.



9. Entre na parte de código da página AppShell.xaml (F7) para recuperar dados do usuário da Preference e atribuir ao label que adicionamos no design

```
using AppRpgEtec.Views.Personagens;  
  
namespace AppRpgEtec  
{  
    5 references  
    public partial class AppShell : Shell  
    {  
        1 reference  
        public AppShell()  
        {  
            InitializeComponent();  
  
            string login = Preferences.Get("UsuarioUsername", string.Empty);  
            lblLogin.Text = login;  
  
            Routing.RegisterRoute("cadPersonagemView", typeof(CadastroPersonagemView));  
        }  
    }  
}
```

Resultados esperados:

