

# Navigation entre pages avec MAUI

---



Il existe plusieurs manières de gérer la navigation entre pages dans MAUI (Multi-platform App UI) dont voici un résumé.

## 1. Navigation de base avec Shell

```
// Dans App.xaml.cs
public App()
{
    InitializeComponent();
    MainPage = new AppShell();
}

// Dans AppShell.xaml.cs
public AppShell()
{
    InitializeComponent();
    Routing.RegisterRoute(nameof(DetailPage), typeof(DetailPage));
}

// Navigation vers une page
await Shell.Current.GoToAsync(nameof(DetailPage));

// Retour en arrière
await Shell.Current.GoToAsync("..");
```

## 2. Navigation avec paramètres

```
// Enregistrement de la route
Routing.RegisterRoute("detailpage", typeof(DetailPage));

// Navigation avec paramètres
await Shell.Current.GoToAsync($"detailpage?id={item.Id}&name={item.Name}");

// Dans DetailPage.xaml.cs (réception des paramètres)
[QueryProperty(nameof(Id), "id")]
[QueryProperty(nameof(Name), "name")]
public partial class DetailPage : ContentPage
{
    private string _id;
    private string _name;

    public string Id { get => _id; set { _id = value; OnPropertyChanged(); } }
    public string Name { get => _name; set { _name = value; OnPropertyChanged(); } }
}
```

### 3. Navigation modale (Push)

```
// Afficher une page de manière modale
await Navigation.PushModalAsync(new ModalPage());

// Fermer une page modale
await Navigation.PopModalAsync();
```

### 4. Navigation par pile (Push/Pop)

```
// Ajouter une page à la pile
await Navigation.PushAsync(new SecondPage());

// Retirer la page actuelle de la pile
await Navigation.PopAsync();

// Aller à la racine de la pile
await Navigation.PopToRootAsync();
```

### 5. Navigation avec animation personnalisée

```
// Navigation avec une animation personnalisée
await Navigation.PushAsync(new SecondPage(), false); // false désactive
l'animation par défaut

// Pour une animation personnalisée
```

```
var secondPage = new SecondPage();
secondPage.Opacity = 0;
await Navigation.PushAsync(secondPage, false);
await secondPage.FadeTo(1, 500);
```

## 6. Navigation tabulaire (TabBar)

```
<!-- Dans AppShell.xaml -->
<TabBar>
  <Tab Title="Accueil" Icon="home.png">
    <ShellContent ContentTemplate="{DataTemplate local:HomePage}" />
  </Tab>
  <Tab Title="Profil" Icon="profile.png">
    <ShellContent ContentTemplate="{DataTemplate local:ProfilePage}" />
  </Tab>
</TabBar>
```

## 7. Navigation programmée vers un onglet

```
// Naviguer vers un onglet spécifique
await Shell.Current.GoToAsync("//profile");
```