



XAML

Vincent Uhlmann
IT-Akademie Dr. Heuer GmbH

WAS IST XAML

- Abkürzung für eXtensible Application Markup Language
- Deklarative XML-basierte Sprache zum Definieren von UI
- Trennt das UI-Design von der Geschäftslogik (C# Code-Behind)
- Sorgt für ein übersichtliches und strukturiertes Layout

SYNTAX

- XAML verwendet XML-Syntax, um das User Interface zu beschreiben
- Jedes UI-Element wird durch ein XML-Element repräsentiert
- Attribute innerhalb der Elemente setzen
Eigenschaften wie Größe, Farbe und Verhalten
- Elemente bilden UI-Komponenten wie Button, Label, StackLayout
- Eigenschaften werden als Attribute innerhalb der Elemente definiert, z.B. Text="Hello World"

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
  xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
  x:Class="MauiApp1.MainPage">
  <ScrollView>
    <VerticalStackLayout
      Padding="30,0"
      Spacing="25">
      <Label
        Text="Hello, World!"
        Style="{StaticResource Headline}">
      </Label>
    </VerticalStackLayout>
  </ScrollView>
</ContentPage>
```

SYNTAX

- Elemente werden verschachtelt, um die Hierarchie der UI zu definieren
- Eltern-Kind-Beziehungen in der Hierarchie bestimmen das Layout und die Anordnung der Komponenten
- Jedes XAML-Element wird durch ein Öffnungs-Tag `<ElementName>` eingeleitet und durch ein Schließungs-Tag `</ElementName>` beendet
- Die Eigenschaften des Elements werden innerhalb des Öffnungs-Tags definiert, zwischen den `<` und `>` Zeichen
- Selbstschließende Tags `<Button ... />` für Elemente ohne Kinder

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="MauiApp1.MainPage">
    <ScrollView>
        <VerticalStackLayout
            Padding="30,0"
            Spacing="25">
            <Label
                Text="Hello, World!"
                Style="{StaticResource Headline}">
            </Label>
        </VerticalStackLayout>
    </ScrollView>
</ContentPage>
```

AUFBAU

- **<?xml version="1.0" encoding="utf-8" ?>:** Definiert die XML-Version und die Kodierung
- **ContentPage:** Root-Element für eine Seite in .NET MAUI, die den Hauptinhalt der Ansicht definiert
- **XML-Namespaces:** Namespaces für zusätzliche XAML-Features, wie z.B. die Zuordnung der XAML-Seite zur C#-Klasse
- **ScrollView:** Erlaubt das Scrollen des Inhalts, falls dieser größer als der Bildschirm ist
- **VerticalStackLayout:** Ordnet die Untergeordneten-Elemente vertikal an
- **Label:** Zeigt Text an

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
  xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
  x:Class="MauiApp1.MainPage">
  <ScrollView>
    <VerticalStackLayout
      Padding="30,0"
      Spacing="25">
      <Label
        Text="Hello, World!"
        Style="{StaticResource Headline}">
      </Label>
    </VerticalStackLayout>
  </ScrollView>
</ContentPage>
```

SEITEN

- Seitentypen, die spezifische Strukturen und Verhaltensweisen für unterschiedliche Anwendungsanforderungen anbieten
- **ContentPage:** Die grundlegendste Seite, die für die Darstellung einzelner Inhalte verwendet wird
- **NavigationPage:** Ermöglicht die hierarchische Navigation zwischen Seiten mit einem integrierten Zurück-Button
- **TabbedPage:** Zeigt Tabs am oberen oder unteren Rand des Bildschirms an, zwischen denen Benutzer wechseln können
- **FlyoutPage:** Erlaubt die Erstellung von Seiten mit einem Flyout-Menü, auch bekannt als Hamburger-Menü

SEITEN



ContentPage



FlyoutPage



NavigationPage

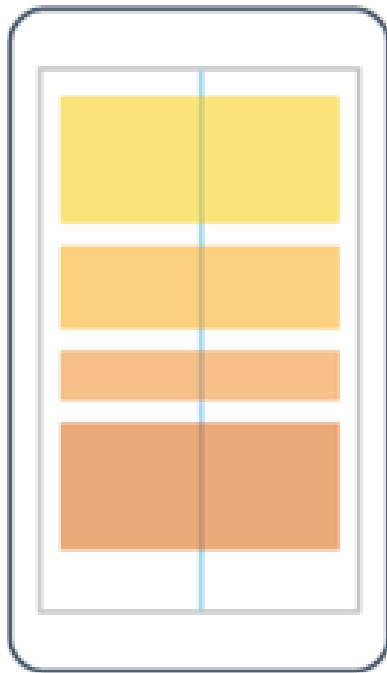


TabbedPage

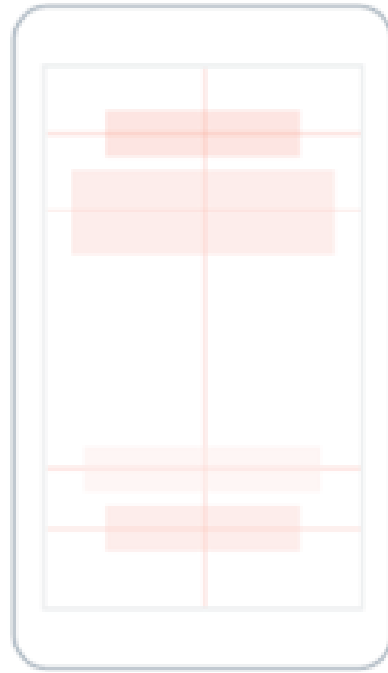
LAYOUTS

- Layouts organisieren visuelle Elemente
- **StackLayout:** Einfache sequentielle Anordnung von Elementen, entweder vertikal oder horizontal
- **GridLayout:** Flexibles Layout, das Elemente in einem Raster aus Zeilen und Spalten anordnet
- **FlexLayout:** Bietet eine flexible Anordnung von Elementen basierend auf dem Flexbox-Layoutmodell
- **AbsoluteLayout:** Ermöglicht die exakte Platzierung von Elementen an spezifischen Koordinaten
- ...

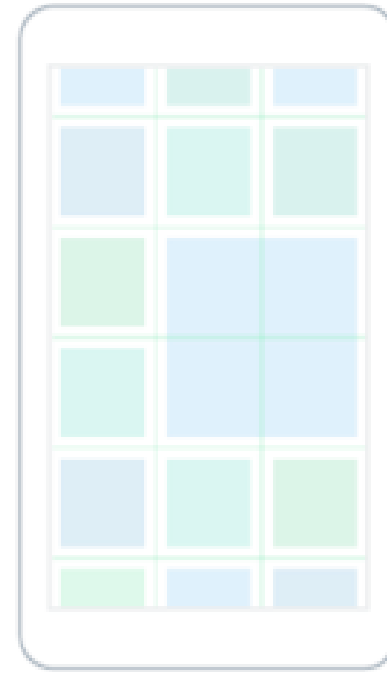
LAYOUTS



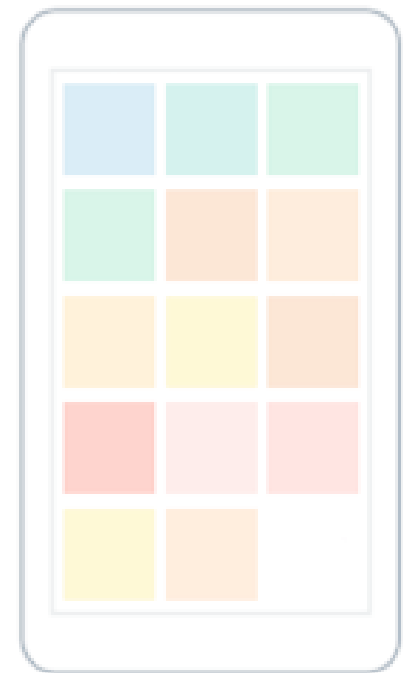
StackLayout



AbsoluteLayout



Grid



FlexLayout

ANSICHTEN

- Ansichten (Views) sind die Bausteine der Benutzeroberfläche
- Sie repräsentieren sichtbare Elemente, mit denen Benutzer interagieren können, wie Buttons, Labels, Textfelder und Bilder
- **Button:** Ermöglicht Benutzerinteraktionen durch Klicks
- **Label:** Zeigt Text auf dem Bildschirm an
- **Entry:** Einzeiliges Textfeld für die Benutzereingabe
- **Editor:** Mehrzeiliges Textfeld für umfangreichere Eingaben
- **Image:** Stellt Bilder aus lokalen Ressourcen oder URLs dar
- ...

KOMPLEXE UI-KOMPONENTEN

- **ListView und CollectionView:** Für die Darstellung von Listen oder Sammlungen von Daten
- **CarouselView:** Ermöglicht das horizontale Durchblättern von Inhalten
- **SwipeView:** Fügt Wischgesten hinzu, um Aktionen wie Löschen oder Bearbeiten zu ermöglichen
- ...