**Projekt nr 2 - Struktura dokumentu XAML**

1. Wybieramy projekt zgodnie z numerem dziennika.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NRD** | | **Zadanie** *(wiersz)* | | | |
| **17** | **1** | 1 | 6 | 11 | 16 |
| **18** | **2** | 2 | 7 | 12 | 17 |
| **19** | **3** | 3 | 8 | 13 | 18 |
| **20** | **4** | 4 | 9 | 14 | 19 |
| **21** | **5** | 5 | 10 | 15 | 16 |
| **22** | **6** | 1 | 7 | 13 | 18 |
| **23** | **7** | 2 | 8 | 14 | 19 |
| **24** | **8** | 3 | 9 | 15 | 20 |
| **25** | **9** | 4 | 10 | 11 | 15 |
| **26** | **10** | 5 | 6 | 12 | 16 |
| **27** | **11** | 1 | 8 | 13 | 17 |
| **28** | **12** | 2 | 9 | 14 | 18 |
| **29** | **13** | 3 | 10 | 15 | 19 |
| **30** | **14** | 4 | 6 | 11 | 20 |
| **31** | **15** | 5 | 7 | 12 | 15 |
| **32** | **16** | 1 | 8 | 13 | 16 |
| **33** |  | 2 | 9 | 14 | 17 |

1. Do każdego projektu dokładamy kontrolkę z zaokrąglonymi rogami w którym umieszczamy nazwisko i imię, a kolory ustawiamy wg tabeli.

|  |  |  |  |
| --- | --- | --- | --- |
| kolory | NRD dla Koloru tła | NRD dla Koloru Trzcionki | NRD dla Koloru obramowania |
| Barwy podstawowe RGB HTML | 18 | 16 | 17 |
| 19 | 17 | 18 |
| 20 | 18 | 19 |
| 21 | 19 | 20 |
| 22 | 20 | 21 |
| 23 | 21 | 22 |
| 24 | 22 | 23 |
| 25 | 23 | 24 |
| 26 | 24 | 25 |
| 27 | 25 | 26 |
| 28 | 26 | 27 |
| 29 | 27 | 28 |
| 30 | 28 | 29 |
| 16 | 29 | 30 |
| 17 | 30 | 15 |
| 18 | 15 | 16 |
|  |  |  |  |

1. Realizujemy 3 widoki kontrolek wg
   1. VerticalStackLayout + HorizontalStackLayout (Łączenie). Możemy łączyć oba układy, np. pionowy layout z poziomym układem przycisków.
   2. 2.5.2 Grid – układ w siatce (wiersze/kolumny)
   3. Dowolny
2. Jako pracę wysyłamy email zawierający:
   1. Temat: 3p AM Kowalski Jan projekt nr 2
   2. 3 zrzuty ekranu z uruchomionych aplikacji - w treści emaila
   3. 3 kody źródłowe \*.xaml zawierający zdefiniowane kontrolki – w postaci tekstu, jako treść e-maila

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 |  | | | |
| 2 |  | | | |
| 3 |  | | | |
| 4 |  | | | |
| 5 |  | | | |
| 6 | Opcja -1 | Opcja -2 | Opcja -3 | Opcja -4 |
| 7 |  | *~~Kolory z tabeli~~* |  |  |
| 8 | Opcja -2 |  |  |  |
| 9 |  |  | Opcja -2 |  |
| 10 |  | *~~Kolory z tabeli~~* | *~~Kolory z tabeli~~* |  |
| 11 | Godzina od | | Godzina do | |
|  | |  | |
| 12 |  | |  | |
| 13 |  | |  | |
| 14 |  | |  | |
| 15 | *~~Kolory z tabeli~~* | |  | |
| 16 |  |  | *~~Kolory z tabeli~~* | |
| 17 |  | Wersja 2 |  | |
| 18 |  |  |  | |
| 19 |  |  | *~~Kolory z tabeli~~* | |
| 20 |  |  | *~~Kolory z tabeli~~* | |

Przykłady

## VerticalStackLayout – układa elementy jeden pod drugim.  Idealny do formularzy, list, ekranów logowania.

<VerticalStackLayout Padding="20" Spacing="10">

    <Label Text="Zaloguj się" FontSize="24" FontAttributes="Bold" HorizontalOptions="Center"/>

    <Entry Placeholder="E-mail" WidthRequest="250"/>

    <Entry Placeholder="Hasło" IsPassword="True" WidthRequest="250"/>

    <Button Text="Zaloguj" BackgroundColor="Red" TextColor="White"/>

</VerticalStackLayout>

**Elementy są ułożone w pionie** – jeden pod drugim.  
 **Padding=”20″** – odstęp od krawędzi ekranu.  
 **Spacing=”10″** – odstęp między elementami.  
 **Dobrze sprawdza się w formularzach!**

## HorizontalStackLayout (Układ poziomy) – układa elementy obok siebie (poziomo). Idealny do przycisków, ikon, paska nawigacji.

<HorizontalStackLayout Padding="10" Spacing="15" HorizontalOptions="Center">

    <Button Text="Strona Główna" BackgroundColor="Gray" TextColor="White"/>

    <Button Text="O nas" BackgroundColor="Gray" TextColor="White"/>

    <Button Text="Kontakt" BackgroundColor="Gray" TextColor="White"/>

</HorizontalStackLayout>

**Elementy są ułożone w poziomie** – obok siebie.  
 **Spacing=”15″** – większy odstęp między przyciskami.  
 **Idealny do przycisków nawigacyjnych!**

## VerticalStackLayout + HorizontalStackLayout (Łączenie) Możemy łączyć oba układy, np. pionowy layout z poziomym układem przycisków.

**Formularz rejestracji z przyciskami akcji**

<VerticalStackLayout Padding="20" Spacing="15">

    <Label Text="Rejestracja" FontSize="24" FontAttributes="Bold" HorizontalOptions="Center"/>

    <Entry Placeholder="E-mail"/>

    <Entry Placeholder="Hasło" IsPassword="True"/>

    <!-- Poziome przyciski -->

    <HorizontalStackLayout Spacing="10">

        <Button Text="Anuluj" BackgroundColor="Gray" TextColor="White" WidthRequest="100"/>

        <Button Text="Zarejestruj" BackgroundColor="Red" TextColor="White" WidthRequest="100"/>

    </HorizontalStackLayout>

</VerticalStackLayout>

**Formularz rejestracyjny (VerticalStackLayout)**.  
 **Przyciski są w jednej linii (HorizontalStackLayout)**.  
 **Dobrze sprawdza się w formularzach i dialogach!**

## 2.5.2 Grid – układ w siatce (wiersze/kolumny)

**Opis**

* Najbardziej precyzyjny układ: definiujesz **wiersze** i **kolumny**, a potem ustawiasz, w której „komórce” ląduje kontrolka.
* Idealny do formularzy, kart, paneli z ikoną i opisem.

**Formularz logowania w Grid (2 kolumny, 2 wiersze + przycisk przez całą szerokość)**

<Grid Padding="10" RowSpacing="8" ColumnSpacing="8">

<!-- Definicje 3 wierszy i 2 kolumn -->

<Grid.RowDefinitions>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="\*"/>

<ColumnDefinition Width="\*"/>

</Grid.ColumnDefinitions>

<!-- Wiersz 1 -->

<Label Grid.Row="0" Grid.Column="0"

Text="Etykieta"

BackgroundColor="LightBlue"

HorizontalOptions="Center"

VerticalOptions="Center"/>

<Entry Grid.Row="0" Grid.Column="1"

Placeholder="Wpisz tekst"

BackgroundColor="LightYellow"/>

<!-- Wiersz 2 -->

<Button Grid.Row="1" Grid.Column="0"

Text="Kliknij mnie"

BackgroundColor="LightGreen"/>

<Switch Grid.Row="1" Grid.Column="1"

OnColor="Orange"/>

<!-- Wiersz 3 -->

<DatePicker Grid.Row="2" Grid.Column="0"

BackgroundColor="LightPink"/>

<TimePicker Grid.Row="2" Grid.Column="1"

BackgroundColor="Lavender"/>

</Grid>



 RowDefinitions/ColumnDefinitions – ustawiasz **szkielet** siatki  
 Auto – tyle miejsca, ile potrzebuje zawartość  
 \* – elastycznie wypełnij resztę (może być też 2\*, 3\*)  
 Grid.Row, Grid.Column, Grid.ColumnSpan – pozycjonowanie elementów

**Tip INF.04:** Grid to najczęstszy wybór w zadaniach egzaminacyjnych z formularzami.

## 2.5.3 FlexLayout – elastyczny jak CSS Flexbox

**Opis**

* Układa elementy w **wierszu** lub **kolumnie**, z kontrolą rozmieszczenia i wyrównania.
* Świetny do pasków przycisków, „kart” i responsywnych paneli.

**Pasek akcji (trzy przyciski w jednym rzędzie, równo rozstawione)**

<FlexLayout Direction="Row"

JustifyContent="SpaceBetween"

AlignItems="Center"

Padding="16">

<Button Text="Anuluj" WidthRequest="110"/>

<Button Text="Zapisz" WidthRequest="110"/>

<Button Text="Wyślij" WidthRequest="110"/>

</FlexLayout>

 Direction=”Row” albo „Column”  
 JustifyContent (Start, Center, SpaceBetween, SpaceAround) – **rozmieszczenie w osi głównej**  
 AlignItems (Start, Center, End, Stretch) – **wyrównanie w osi poprzecznej**

**Karta z obrazkiem i opisem (Row + zawijanie)**

<FlexLayout Direction="Row" Wrap="Wrap" AlignItems="Center">

<Image Source="avatar.png" WidthRequest="64" HeightRequest="64" Margin="0,0,12,0"/>

<Label Text="Jan Kowalski – krótki opis użytkownika" WidthRequest="220" Margin="0,0,12,0"/>

<Button Text="Profil"/>

</FlexLayout>

 Wrap=”Wrap” – gdy zabraknie miejsca, elementy **zawijają** do nowej linii  
 Idealny do „pływających” układów

## 2.5.4 AbsoluteLayout – pozycje „na sztywno” lub procentowo

**Opis**

* Umożliwia ustawianie elementów **w konkretnych współrzędnych** lub **proporcjonalnie** do rozmiaru.

**Nagłówek na górze + napis wyśrodkowany procentowo**

<AbsoluteLayout>

<!-- Pasek nagłówka (20% wysokości, pełna szerokość) -->

<BoxView Color="#222"

AbsoluteLayout.LayoutBounds="0,0,1,0.2"

AbsoluteLayout.LayoutFlags="All"/>

<!-- Tytuł wycentrowany (pozycja 50%/10%) -->

<Label Text="Panel ustawień" TextColor="White" FontSize="24"

AbsoluteLayout.LayoutBounds="0.5,0.1,AutoSize,AutoSize"

AbsoluteLayout.LayoutFlags="PositionProportional"

</AbsoluteLayout>

 LayoutBounds=”x,y,width,height” + LayoutFlags (All / PositionProportional / SizeProportional)  
 Używaj **rozsądnie** – trudniej utrzymać responsywność niż w Grid/Flex

## 2.5.5 ScrollView – zawartość przewijalna

**Opis**

* Zawija **jeden** element potomny, który zwykle jest layoutem z wieloma kontrolkami.
* Gdy formularz jest długi → **must have** na telefonie.

<ScrollView>

<VerticalStackLayout Padding="20" Spacing="12">

<Label Text="Długi formularz" FontAttributes="Bold" FontSize="22"/>

<!-- …dużo pól… -->

<Entry Placeholder="Adres e-mail"/>

<Entry Placeholder="Telefon"/>

<Entry Placeholder="Ulica i numer"/>

<Entry Placeholder="Miasto"/>

<Entry Placeholder="Kod pocztowy"/>

<!-- …itd. -->

<Button Text="Zapisz" BackgroundColor="Red" TextColor="White"/>

</VerticalStackLayout>

</ScrollView>

 Tylko **jeden** bezpośredni „dziecko” – w środku daj VerticalStackLayout/Grid  
 Dzięki temu zawartość **przewija się**, zamiast „ucinać” na małym ekranie

## 2.5.6 ContentView – prosty kontener / blok wielokrotnego użytku

**Opis**

* Opakowanie na fragment interfejsu, który chcesz **wydzielić** lub **wielokrotnie użyć**.

<ContentView Padding="12" BackgroundColor="#EEF6FF">

<HorizontalStackLayout Spacing="10">

<Image Source="info.png" WidthRequest="20" HeightRequest="20"/>

<Label Text="Twoje zmiany zostały zapisane." TextColor="#0A3D91"/>

</HorizontalStackLayout>

</ContentView>

 Dobre do „paneli informacyjnych”, kart, sekcji, które pojawiają się w kilku miejscach