

Dokumentation

Lavet af Elias Kouly

Jeg har lavet et project hvor man kan vælge computer og give feedback derefter

Contents

Hvorfor er der wpf.....	3
- Hvad er forskellen på WPF og Windows Forms?.....	3
- Hvad er fordelene med WPF?	3
- Forklar i hvilken sammenhæng man bruger WPF?	3
- Hvad er XAML?	4
- Hvad er fordelene med XAML?	4
Begyndt at kode.....	4
Dokumentation på video nummer 1	4
Video nummer 2 dokumentation	6
Vindue properties.....	6
Window XAML	6
Tilføj en database string	6
Dokumentation for opgave	7
Mockup UML	7
Hvad skal programmet gøre?	8
Kode.....	9
Konklusion	10

Hvorfor er der wpf

Fordi den kan bedre håndtere animation, og de andre f.eks. winforms er gammelt i forhold. Med wpf kan siden også køre med grafikortet det gør den ikke med winforms. På den måde kan man lave 3D interface, og gøre det hele mere spændende. Wpf er lidt sværere end winforms

- Hvad er forskellen på WPF og Windows Forms?

WPF er nyere og bruger nyere teknologi, f.eks. bruger WPF grafikort hvilket sørger for at der kan komme meget mere visuelt, end med winforms. WPF kan også håndtere mere med animation. Man kan drag n drop lige som i Winforms. Men det er dårligt da den laver hard coded positions.

- Hvad er fordelene med WPF?

Den gør det hele nemmere, den er stærkere det vil sige at den kan håndtere mere.

- Forklar i hvilken sammenhæng man bruger WPF?

Man kan bruge det til klient programmering (Web) den sørger for at hjemmesider f.eks. kan nemt skaleres i forhold til skærm størrelsen

- Hvad er XAML?

XAML (en forkortelse for eXtensible Application Markup Language) Det er et kodesprog som der er blevet udviklet af microsoft

- Hvad er fordelene med XAML?

Det er nemt at læse da, det ligner meget HTML og CSS blandet sammen. Selv første gang jeg kiggede på noget kode syntes jeg at det virkede overskueligt. Den har også et nyere og mere forståeligt layout i forhold til html og css ifølge min mening

Begyndt at kode

1. Åben c#
2. find eller søg på wpf app (Framework)
3. Lav det og giv den et navn
4. Hvis man kigger til højre i (Solution explore) kan man højre klikke mainWindows og så view code. På den måde kan man både se og ændre den kode der er.

Hvad betyder den boks der er nedenunder så? Det er XAML

Dokumentation på video nummer 1

Jeg fulgte videoen og jeg forstår hvad han gjorde og hvorfor, her er noget af det første der skete i videoen

```
<Grid>
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="20" />
    <ColumnDefinition Width="auto" />
    <ColumnDefinition Width="auto" />
    <ColumnDefinition Width="auto" />
    <ColumnDefinition Width="*" />
    <ColumnDefinition Width="20" />
  </Grid.ColumnDefinitions>
  <Grid.RowDefinitions>
    <RowDefinition Height="20" />
    <RowDefinition Height="auto" />
    <RowDefinition Height="auto" />
    <RowDefinition Height="auto" />
    <RowDefinition Height="auto" />
    <RowDefinition Height="auto" />
    <RowDefinition Height="*" />
    <RowDefinition Height="20" />
  </Grid.RowDefinitions>
```

Her laver man nogle colloner som man kan komme ting inde i, lige som grid i HTML og CSS. Auto sørger for at når man kommer noget ind i boksen vil boksen automatisk skalere i forhold til hvad der er puttet der ind

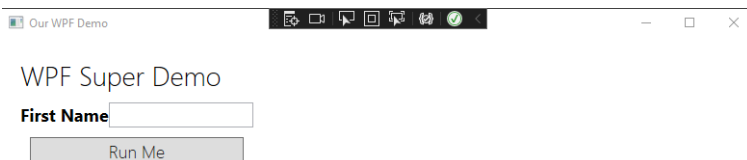
```
<TextBlock Grid.Column="1"
           Grid.Row="1"
           FontSize="28"
           Text="WPF Super Demo"
           Grid.ColumnSpan="3"
           Margin="0,0,0,10"/>
```

Her er har man først valgt i hvilket grid man ønsker teksten skal være i, og ColumnSpan, sørger for at den går over 3 column er. Margin 0,0,0,10 siger at alle sider bortset fra ned af skal have margin 10

```
<TextBlock Grid.Column="1"
           Grid.Row="2"
           FontWeight="Bold"
           Text="First Name"
           FontFamily="Segoe UI" />

<TextBox x:Name="firstNameText"
          Grid.Column="2"
          Grid.Row="2"
          Width="150"/>

<Button x:Name="submitButton"
        Content="Run Me"
        Grid.Column="1"
        Grid.Row="3"
        Grid.ColumnSpan="2"
        Margin="10"
        Click="submitButton_Click"/>
```



Video nummer 2 dokumentation

Vindue properties

Stack Panel. Er en container. Det sørger for at alle elementerne bliver lagt oven på hinanden.

Alle ting skal starte og slutte et sted, det starter sådan her **<Button>** og slutter sådan **</button>** præcis ligesom i HTML

WindowState Betyder bare om det skal starte i fuld skærm eller ikke i fuld skærm

WindowStartupLocation Det er kordinaterne for hvor vinduet er

Topmost Hvis den er sat til True vil den side i toppen hele tiden

SizeToContent Så vil den automatisk prøve at skalere til størrelsen

ShowInTaskbar Hvis den er sat til false vil man ikke kunne se den inde i sin taskbar

Icon Det er vinduets ikon

Window XAML

X:Class Den fortæller hvilken fil den skal bruge. (Filen er lavet fra start det er default)

Partial class Det er en class som der er en del af en anden class. Fordi den er en del af xaml file.

X:Class og Partial class er kombineret og vil derfor køre i fuld skærm

WPF Designer xmlns:d d står for designer det sørger for at vi kan se det visuelt det vi gør

xmlns:mc Den sørger for at man kan åbne word filer

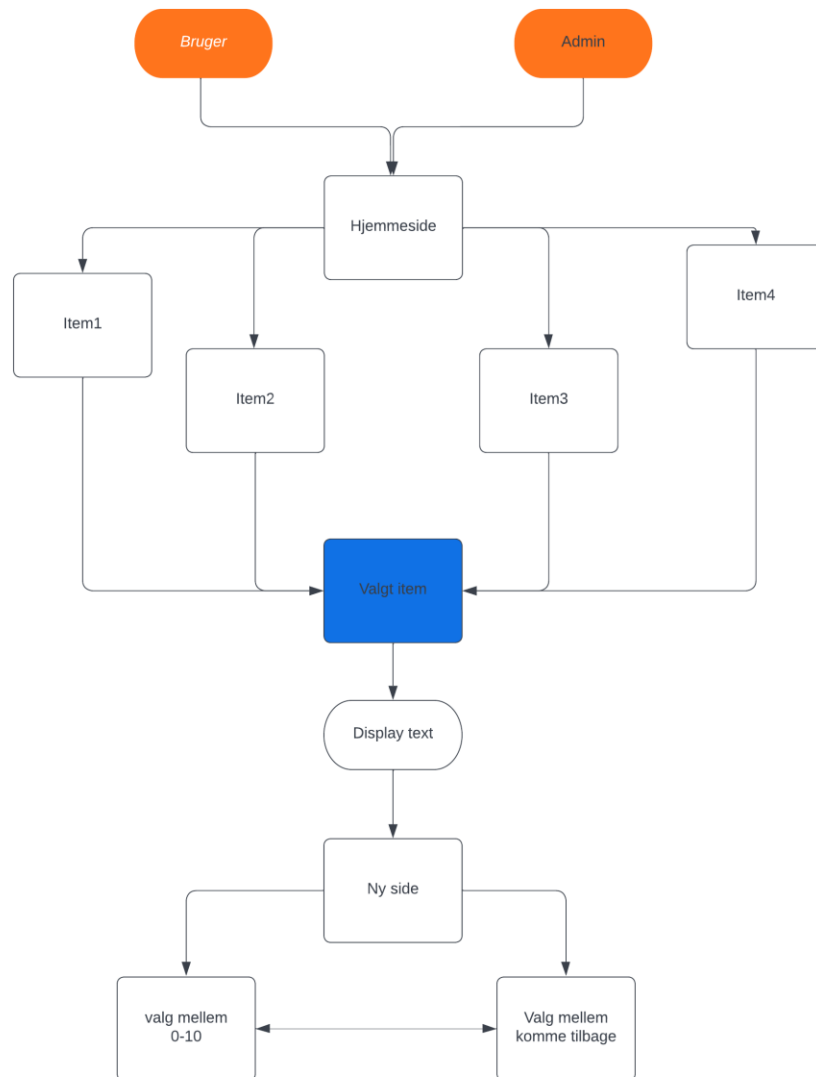
Tilføj en database string

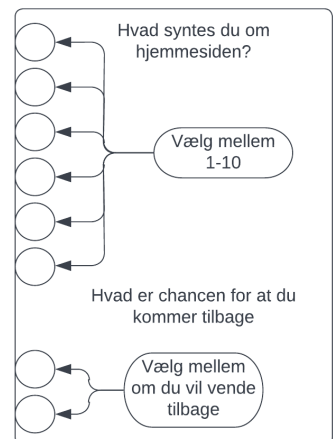
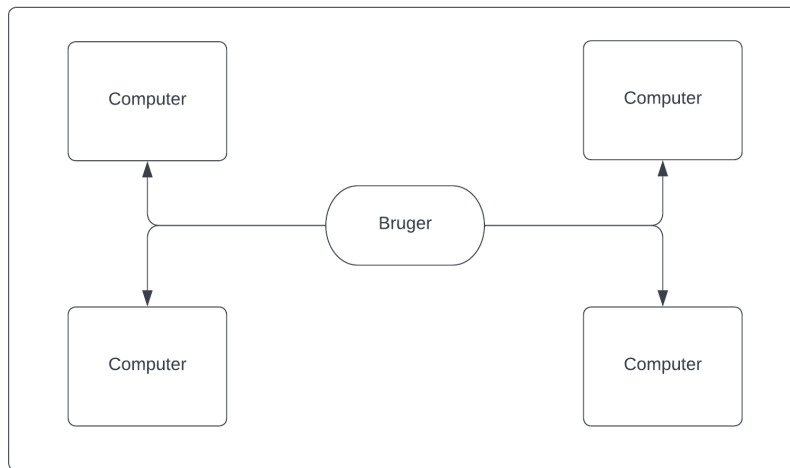
1. Først højre klik på references > Add references
2. vælg System.Configuration fra Assemblies
3. Når de er tilføjet, kan man starte med at konfigurere

Sådan får du adgang til det

```
String dbConnStr = ConfigurationManager.ConnectionStrings["DefaultConnectionString"].ConnectionString;
```

Dokumentation for opgave Mockup UML





Lige så snart man kommer ind skal man se 3 mega gode og 1 dårlig computer. Brugeren kan så vælge imellem hvilke en han eller hun vil vælge, og når de så har valgt skal de kunne give feedback

Jeg tænker at når man laver en button hvor man ligger billeder på og tilføjer et click event. Så den vil tage en hen til den anden side. Derefter skal man kunne vælge hvad man syntes på en skala fra 0-10 og hvor stor chancen er for at man kommer tilbage. Det bedste ville være at dele de 2 spørgsmål op i grupper så man kan vælge imellem to ting

Hvad skal programmet gøre?

Når brugeren først kommer ind, kan han vælge imellem de 4 computere, når han har klikket bliver han sendt hen til et andet sted, hvor han kan sige hvor godt at syntes det var. Koden er ikke så avanceret, så det skulle være ret lige til at begynde på

Kode

På første side, også kaldet **MainWindow** kan man se at jeg har tilføjet en hel del Grids. Det sørger for at jeg kan dele computerne op i dele så nogle af dem er i venstre og andre i højre side. Grunden til at jeg har sat dem til auto, det er fordi så kommer størrelsen på gridden an på hvor stort billedet er

```
<Grid.ColumnDefinitions>
  <ColumnDefinition Width="20" />
  <ColumnDefinition Width="auto"/>
  <ColumnDefinition Width="auto" />
  <ColumnDefinition Width="auto" />
  <ColumnDefinition Width="auto" />
  <ColumnDefinition Width="*" />
  <ColumnDefinition Width="20" />
</Grid.ColumnDefinitions>
```

Når man skriver * betyder det at man vil fylde resten, så hvis auto var tomme så vil stjerne fylde dem ud

Neden under er der 4 knapper. De har alle sammen PCer som billede. Jeg har lagt dem i hver deres grid

```
<Button
  x:Name="myButton"
  Background="Gray"
  Click="Button_Click"
  Grid.Column="1"
  Grid.Row="1">
  <Image
    Width="150"
    Source="expensive pc.png"></Image>
</Button>
```

Næsten alle sammen ser sådan ud. Der er blevet lagt et klik event på den, så når man klikker på den, kommer man ind på næste side, men det vil jeg skrive om senere. Først tager vi lige alt det der er på denne side

Nedeunder alle billederne kan man se, de her textblocks. De er også blevet lagt ind i hver deres, grid. De er bare blevet lagt lige ved siden af billederne, og så margin til at sørge for at de står i midten

```
<TextBlock
  TextWrapping="Wrap"
  FontSize="30"
  Text="very good"
  Grid.Column="2"
  Grid.Row="3"
  Margin="0,50,0,50"></TextBlock>
```

I cs filen

neden under er der metoder som der sørger for at når man har klikket på en computer bliver man både sendt over til hjemmesiden og så vil den skrive i en tekst, tak for det beløb computeren nu kostede

```
private void Button_Click(object sender, RoutedEventArgs e)
{
    firstpc.Foreground = Brushes.Black;
    purchase.Text = "Thanks for the 20K";

    Reveiw pg = new Reveiw();
    pg.Show();
}
```

Det er sådan set koden for første side, nu til næste side som også er kaldt **Review.xaml**

I stedet for grid har jeg brugt StackPanel, Det er nemmere at bruge hvis man bare ønsker et enkelt design. Det sørger bare for at alle ting bliver lagt oven på hinanden. Derefter har jeg lavet noget tekst med Label, og så brugte jeg RadioButton. RadioButton sørger for at der kommer de her cirkler hvor man kan vælge imellem en a tingene. Jeg har givet dem et GroupName, da det sørger for at man kan tjekke flere cirkler, men ikke flere i den samme gruppe, derfor er der 2

Det var alt koden der er skrevet

Konklusion

Nu virker det hele. Der er 4 PCer hvor man kan se hvad de koster og hvad de hedder. Når man klikker på dem kan man give feedback på 2 ting ved at klikke på nogle skala bokse