

Министерство образования и молодежной политики Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по учебной практике

УП 01.01

Выполнил: Молотова В.А.

Группа: ПР-21

Преподаватель: Мирошниченко Г.В.

2023

Содержание

[1. Задание №6.1 Индивидуальное задание (вариант 4) 3](#_Toc138325047)

[**1.1** **Описание задачи** 3](#_Toc138325048)

[**welcome.xaml** 3](#_Toc138325049)

[**welcome.xaml.cs** 3](#_Toc138325050)

[**Pade1.xaml** 4](#_Toc138325051)

[**Pade1.xaml.cs** 7](#_Toc138325052)

[**mystyle.css** 8](#_Toc138325053)

[**disain.css** 9](#_Toc138325054)

[**1.3 Описание разработанных функций** 11](#_Toc138325055)

[**1.4 Алгоритм решения** 11](#_Toc138325056)

[**1.5 Используемые библиотеки** 13](#_Toc138325057)

[**1.6 Тестовые случаи** 13](#_Toc138325058)

[**1.7 Используемые инструменты** 15](#_Toc138325059)

[**1.8 Описание пользовательского интерфейса** 15](#_Toc138325060)

[**Экран welcome:** 15](#_Toc138325061)

[**Экран Page1:** 15](#_Toc138325062)

[**1.9 Приложение (pr screen экранов)** 16](#_Toc138325063)

# 1. Задание №6.1 Индивидуальное задание (вариант 4)

## **Описание задачи**

1.Необходимо с помощью CSS оформить командные элементы в

соответствии с вариантом:

2. Сделать проверку на заполнение полей во вкладке Welcome

3. Переход на второй экран при нажатии на кнопку Sign in (с передачей

данных фамилии пользователя со стилизацией как у Sign in)

4. При нажатии на кнопку Static (4

вариант) на 3 экране показать значение из выбранного списка с

расшифровкой информации и передачей максимального значения из

слайдера

### **welcome.xaml**

<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="zadanie4Molotova.welcome"

NavigationPage.HasNavigationBar="False">

<ContentPage.Resources>

<StyleSheet Source="/mystyles.css" />

</ContentPage.Resources>

<StackLayout Orientation="Vertical " >

<StackLayout Orientation="Vertical" HorizontalOptions="Center" Margin="0, 120">

<Label Text="Welcome" FontSize="23" HorizontalOptions="Center" ></Label>

<StackLayout Margin="50">

<Entry x:Name="login" WidthRequest="300" Placeholder="Login" ></Entry>

<Entry x:Name="password" WidthRequest="300" Placeholder = "Password" IsPassword ="True" ></Entry>

</StackLayout >

<Button x:Name="signIn" Text="Sign in" Clicked="signIn\_Clicked" Margin="50, -30"></Button>

</StackLayout>

<StackLayout Orientation="Horizontal" Margin="0 , -70">

<CheckBox Margin="55 , 1"></CheckBox>

<Label Text="Remember me" FontSize="16" Margin="-60, 4" ></Label>

<Label Text="I forgor!" FontSize="16" Margin="105, 5" ></Label>

</StackLayout>

</StackLayout>

</ContentPage>

### **welcome.xaml.cs**

using Xamarin.Forms;

using Xamarin.Forms.Xaml;

namespace zadanie4Molotova

{

[XamlCompilation(XamlCompilationOptions.Compile)]

public partial class welcome : ContentPage

{

public welcome()

{

InitializeComponent();

}

private async void signIn\_Clicked(System.Object sender, System.EventArgs e)

{

try

{

if (login.Text.Length == 0)

{

await DisplayAlert("Ошибка", "Необходимо ввести логин", "Ок");

}

else if (login.Text.Length < 3)

{

await DisplayAlert("Ошибка", "Логин не может быть короче 3 символов", "Ок");

}

else if(password.Text.Length == 0)

{

await DisplayAlert("Ошибка", "Необходимо ввести пароль", "Ок");

}

else if(password.Text.Length < 6)

{

await DisplayAlert("Ошибка", "Пароль не может быть короче 6 символов", "Ок");

}

else

{

await Navigation.PushAsync(new Page1());

}

}

catch

{

await DisplayAlert("Ошибка", "Необходимо ввести логин и пароль", "Ок");

}

}

}

}

### **Pade1.xaml**

<?xml version="1.0" encoding="utf-8" ?>

<CarouselPage xmlns="http://xamarin.com/schemas/2014/forms"

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="zadanie4Molotova.Page1">

<CarouselPage.Resources>

<StyleSheet Source="/disain.css" />

</CarouselPage.Resources>

<ContentPage BackgroundColor="#818181">

<StackLayout Orientation="Vertical">

<StackLayout Orientation="Horizontal">

<Button StyleClass="but1" Text="STATIC" Clicked="Button\_Clicked"

></Button>

<Button StyleClass="but4" Text="STATIC" Clicked="Button\_Clicked"

x:Name="but4"

></Button>

<Button StyleClass="but7" Text="STATIC" Clicked="Button\_Clicked"

x:Name="but7"

></Button>

</StackLayout>

<StackLayout Orientation="Horizontal">

<Button StyleClass="but2" Text="HOVER"

></Button>

<Button StyleClass="but5" Text="HOVER"

></Button>

<Button StyleClass="but8" Text="HOVER"

></Button>

</StackLayout>

<StackLayout Orientation="Horizontal">

<Button StyleClass="but3" Text="PRESSED"

></Button>

<Button StyleClass="but6" Text="PRESSED"

></Button>

<Button StyleClass="but9" Text="PRESSED"

></Button>

</StackLayout>

<StackLayout Orientation="Horizontal">

<Picker x:Name="vid" WidthRequest="150" FontSize="25" Title="DROPDOWN" BackgroundColor="#439e44" TextColor="White">

<Picker.Items>

<x:String>Select One</x:String>

<x:String>Select Two</x:String>

<x:String>Select Three</x:String>

<x:String>Select Four</x:String>

</Picker.Items>

</Picker>

<StackLayout Orientation="Horizontal" >

<StackLayout Orientation="Vertical">

<Switch IsToggled="false"

HorizontalOptions="Center" x:Name="r1" StyleClass="r1" ThumbColor="White"

VerticalOptions="CenterAndExpand"/>

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r2" StyleClass="r2" ThumbColor="White"/>

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r3" StyleClass="r3" ThumbColor="White"/>

</StackLayout>

<StackLayout Orientation="Vertical">

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r4" StyleClass="r4" ThumbColor="White"/>

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r5" StyleClass="r5" ThumbColor="White"/>

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r6" StyleClass="r6" ThumbColor="White"/>

</StackLayout>

<StackLayout Orientation="Vertical">

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r7" StyleClass="r7" ThumbColor="#f84453"/>

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r8" StyleClass="r8" ThumbColor="#f84453"/>

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r9" StyleClass="r9" ThumbColor="#f84453"/>

</StackLayout>

<StackLayout Orientation="Vertical">

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r10" StyleClass="r10" ThumbColor="White"/>

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r11" StyleClass="r11" ThumbColor="White"/>

<Switch IsToggled="false"

HorizontalOptions="Center"

VerticalOptions="CenterAndExpand" x:Name="r12" StyleClass="r12" ThumbColor="White"/>

</StackLayout>

</StackLayout>

</StackLayout>

<StackLayout Orientation="Vertical" Margin="10, 10">

<StackLayout Orientation="Horizontal">

<Slider WidthRequest="180" x:Name="sl1" StyleClass="sl1" MinimumTrackColor="#499d4a" MaximumTrackColor ="#E3E3E3" ThumbColor="#fffcfd"

Minimum="0" Maximum="100" Value="50">

</Slider>

<Slider WidthRequest="180" Minimum="0" Maximum="100" Value="50" x:Name="sl4" StyleClass="sl4" MinimumTrackColor="#499d4a" MaximumTrackColor ="#E3E3E3" ThumbColor="#fffcfd">

</Slider>

</StackLayout>

<StackLayout Orientation="Horizontal">

<Slider WidthRequest="180" x:Name="sl2" Minimum="0" Maximum="100" Value="50" StyleClass="sl2" MinimumTrackColor="#499d4a" MaximumTrackColor ="#E3E3E3" ThumbColor="#fffcfd">

</Slider>

<Slider WidthRequest="180" Minimum="0" Maximum="100" Value="50" x:Name="sl5" StyleClass="sl5" MinimumTrackColor="#499d4a" MaximumTrackColor ="#E3E3E3" ThumbColor="#fffcfd">

</Slider>

</StackLayout>

<StackLayout Orientation="Horizontal">

<Slider WidthRequest="180" Minimum="0" Maximum="100" Value="50" x:Name="sl3" StyleClass="sl3" MinimumTrackColor="#e25646" MaximumTrackColor ="#E3E3E3" ThumbColor="#E42946">

</Slider>

<Slider WidthRequest="180" Minimum="0" Maximum="100" Value="50" x:Name="sl6" StyleClass="sl6" MinimumTrackColor="#e25646" MaximumTrackColor ="#E3E3E3" ThumbColor="#E42946" >

</Slider>

</StackLayout>

<StackLayout Orientation="Horizontal">

<Slider WidthRequest="180" Minimum="0" Maximum="100" Value="50" Margin="100, 10" x:Name="sl7" StyleClass="sl7" MinimumTrackColor="#e25646" MaximumTrackColor ="#E3E3E3" ThumbColor="#E42946">

</Slider>

</StackLayout>

</StackLayout>

<StackLayout Orientation="Horizontal">

<StackLayout Orientation="Vertical">

<RadioButton Value="1" x:Name="rad1" StyleClass="rad33"> </RadioButton>

<RadioButton Value="2" x:Name="rad2" StyleClass="rad333"> </RadioButton>

</StackLayout>

<StackLayout Orientation="Vertical">

<CheckBox x:Name="ch1" StyleClass="ch1"> </CheckBox>

<CheckBox x:Name="ch2" StyleClass="ch1"> </CheckBox>

</StackLayout>

</StackLayout>

</StackLayout>

</ContentPage>

<ContentPage>

<StackLayout>

<Label x:Name="itog" FontSize="25" HorizontalOptions="Center" VerticalOptions="Center" TextColor="White" Text="Здесь будет показываться, какой вариант списка выбран">

</Label>

<Label x:Name="slider" FontSize="25" HorizontalOptions="Center" VerticalOptions="Center" TextColor="White" Text="Здесь будет показываться, максимальное значение из слайдера">

</Label>

</StackLayout>

</ContentPage>

</CarouselPage>

### **Pade1.xaml.cs**

using System;

using System.Linq;

using Xamarin.Forms;

using Xamarin.Forms.Xaml;

namespace zadanie4Molotova

{

[XamlCompilation(XamlCompilationOptions.Compile)]

public partial class Page1 : CarouselPage

{

public Page1()

{

InitializeComponent();

}

void Button\_Clicked(System.Object sender, System.EventArgs e)

{

try

{

string text = "Выбран " + vid.Items[vid.SelectedIndex];

itog.Text = text;

}

catch

{

}

try

{

double[] array = new double[7];

array[0] = sl1.Value;

array[1] = sl2.Value;

array[2] = sl3.Value;

array[3] = sl4.Value;

array[4] = sl5.Value;

array[5] = sl6.Value;

array[6] = sl7.Value;

slider.Text = "Максимальное значение из слайдера:" + Convert.ToString(array.Max());

}

catch

{

itog.Text = "Здесь будет показываться, какой вариант списка выбран";

slider.Text = "Здесь будет показываться, максимальное значение из слайдера";

}

}

}

}

### **mystyle.css**

^contentpage {

background-color: #2D2F36;

}

label {

color: white;

}

Entry {

background-color: white;

color: black;

}

Button {

border-color: #DD4F3D;

background-color: #DD4F3D;

color: white;

}

CheckBox {

border-color: white;

color: #DD4F3D;

}

Picker {

color: white

}

.print {

color: #DD4F3D;

}

DatePicker {

color: #DD4F3D;

}

### **disain.css**

^contentpage {

background-color: #2d2f36;

}

.but1 {

background-color: #449c47;

color: white;

border-radius: 25;

margin: 10, 10;

padding: 14 30;

}

.but2 {

background-color: #3a8038;

color: white;

border-radius: 25;

margin: 10, 10;

padding: 14 30;

}

.but3 {

background-color: #2f6d2e;

color: white;

border-radius: 25;

margin: 10, 10;

padding: 14 20;

}

.but4 {

background-color: #efefef;

color: black;

border-radius: 25;

margin: 10, 10;

padding: 14 20;

}

.but5 {

background-color: #eeeeee;

color: black;

border-radius: 25;

margin: 10, 10;

padding: 14 20;

}

.but6 {

background-color: #e0e0e0;

color: black;

border-radius: 25;

margin: 10, 10;

padding: 14 20;

}

.but7 {

background-color: #efefef;

color: black;

border-radius: 25;

margin: 10, 10;

padding: 14 20;

border-color: black;

}

.but8 {

background-color: #eeeeee;

color: black;

border-radius: 25;

margin: 10, 10;

padding: 14 20;

border-color:black;

}

.but9 {

background-color: #e0e0e0;

color: black;

border-radius: 25;

margin: 10, 10;

padding: 14 20;

border-color: black;

}

.vid {

background-color: #439e44;

border-block-color: black;

border-color: black;

padding: 14 30;

}

.r1 {

color: #f84e49;

}

.r2 {

color: #f84e49;

border-color: white;

}

.r3 {

color: #f84e49;

}

.r7 {

color: #312930;

flood-color: #f63c58;

}

.r8 {

color: #312930;

flood-color: #f63c58;

}

.r9 {

color: #312930;

flood-color: #f63c58;

}

.r4 {

color: #f2f2f2;

}

.r5 {

color: #f2f2f2;

}

.r6 {

color: #f2f2f2;

}

.r10 {

color: #2b2a2f;

}

.r11 {

color: #2b2a2f;

}

.r12 {

color: #2b2a2f;

text-shadow: initial;

}

.sl1 {

color: #499d4a;

}

.sl2 {

color: #499d4a;

}

.sl3 {

color: #499d4a;

}

.rad1 {

background-color: #d65155;

}

.rad2 {

color: #ffedf7;

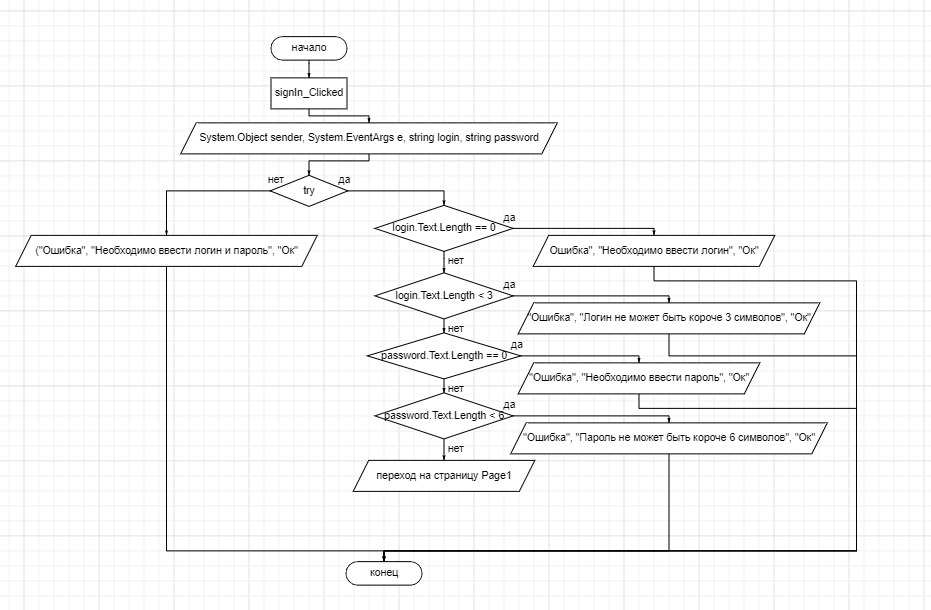
}

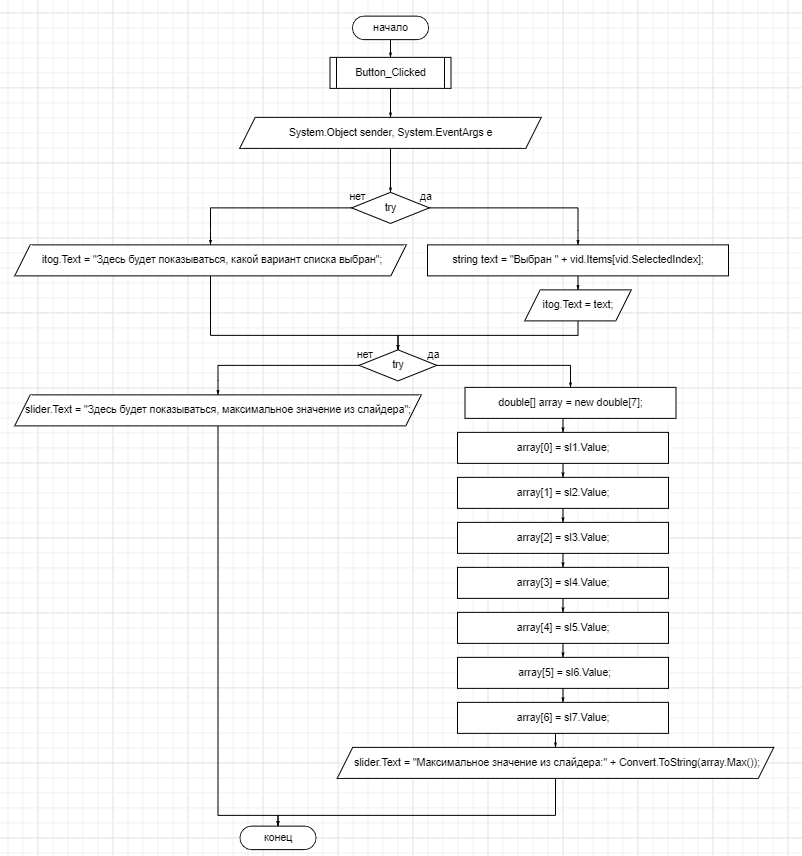
## **1.3 Описание разработанных функций**

signIn\_Clicked – при нажатии на кнопку “sign in” сделан переход на другую страницу

Button\_Clicked – при нажатии на кнопку “ Static” на другой страницу показывается выбраный элемент списка и максимальное значение из слайдеров

## **1.4 Алгоритм решения**





## **1.5 Используемые библиотеки**

using System;

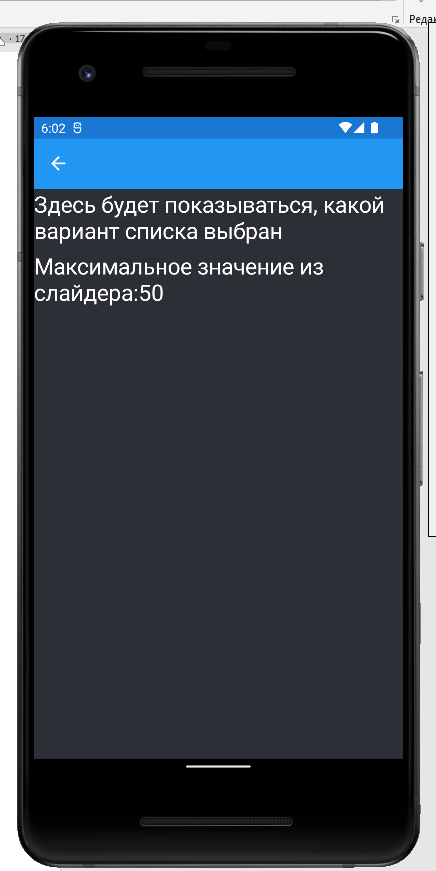
using System.Linq;

using Xamarin.Forms;

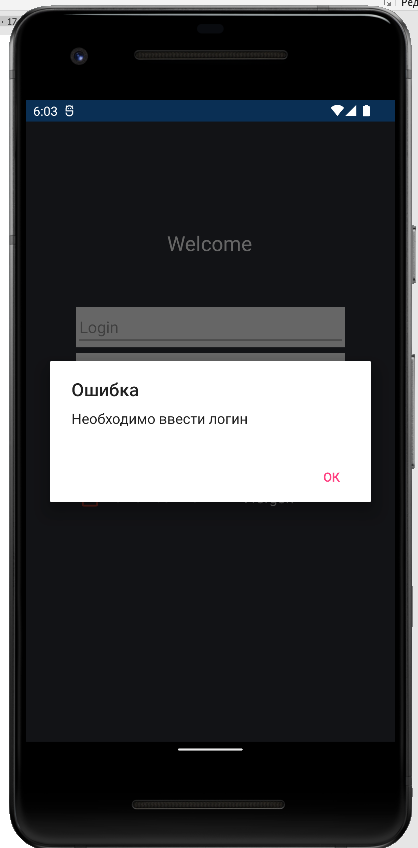
using Xamarin.Forms.Xaml;

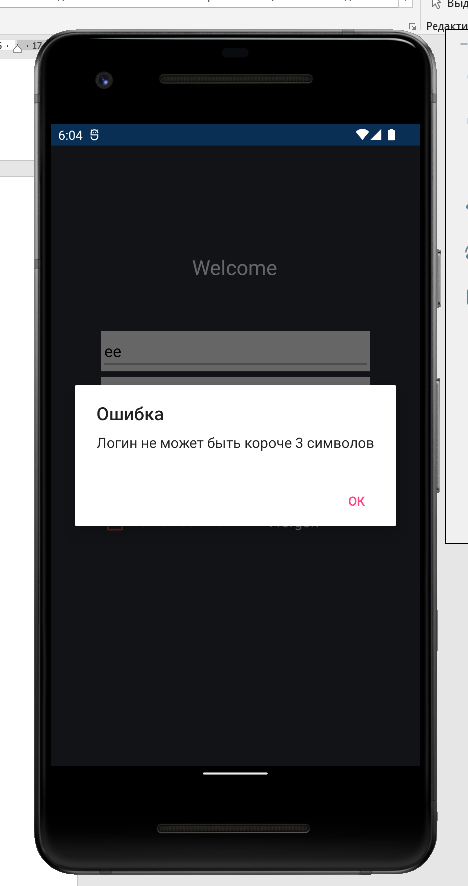
## **1.6 Тестовые случаи**

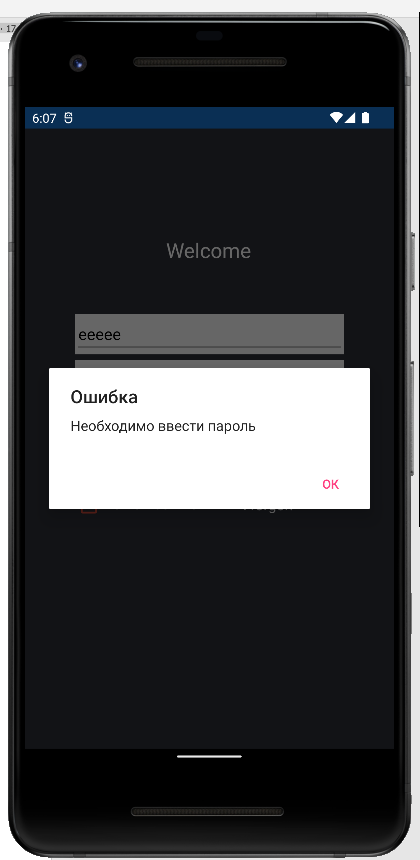
* Если пользователь не выберет элемент из списка и нажмет “ Static” то программа просто покажет строку, где будет показываться выбранный элемент.

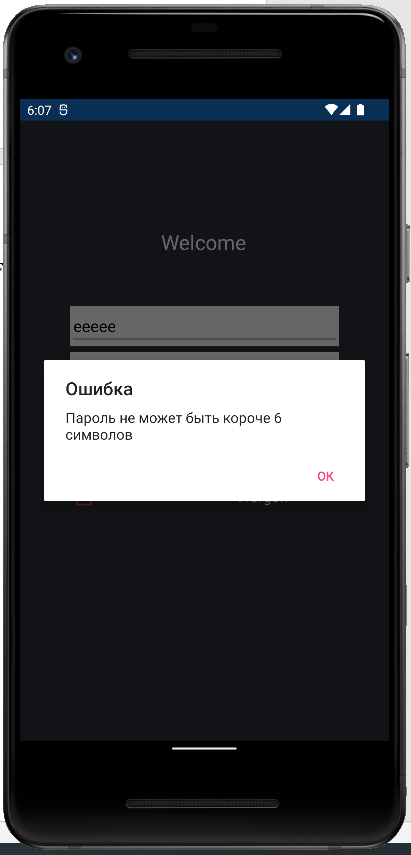


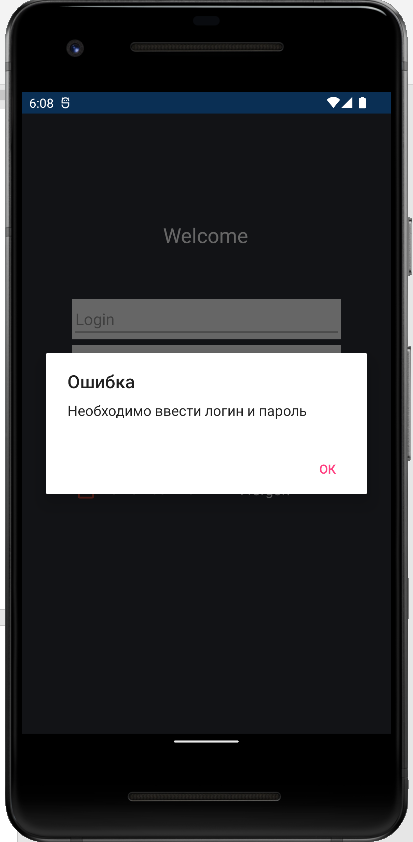
* Есть проверки на правильный ввод пароля и логина











## **1.7 Используемые инструменты**

Язык C# и css. Среда разработки Visual Studio, .NET Framework 4.7.2

## **1.8 Описание пользовательского интерфейса**

### **Экран welcome:**

* Label (3 шт)
* Entry (2 шт)
* Button (1шт)
* CheckBox (1 шт)

### **Экран Page1:**

* Label (2 шт)
* Button (9шт)
* Picker (1 шт)
* Swich (12 шт)
* Slider (7 шт)
* radioButton (2 шт)
* cheakBox (2 шт)

## **1.9 Приложение (pr screen экранов)**

