|  |  |
| --- | --- |
| HÜ04 | Abgabetermin: 24.3.2023 23:59 |
| Name: | Aufwand in h: |
|  | Punkte: |

## Webapplikation (GET-Requests) 8 Punkte

Erstellen Sie eine Webapplikation mit folgenden Seiten (aus dem Navigationsmenü verlinkt):

* Index
* Zufallszahl
* Zeit bis zum Schulschluss 2023

Index: Soll eine Begrüßung in Textform und ein Bild anzeigen. Das Bild soll in Abhängigkeit der Uhrzeit angezeigt werden. Handelt es sich um eine ungerade Minute (z.B.: 12:23) soll ein anderes Bild angezeigt werden, als in einer geraden Minute. Wählen Sie selbst 2 verschiedene Bilder.

Zusätzlich soll die Seite folgende Links enthalten:

* Zufallszahl zw. 0 und 10
* Zufallszahl zw. 0 und 100
* Zufallszahl zw. 0 und 1000
* Zufallszahl zw. -10 und 10

Zufallszahl: Soll bei jedem Aufruf eine neue Zufallszahl generieren. Der Minimalwert- und der Maximalwert soll als URL-Parameter übergeben werden und zusätzlich zur Zufallszahl auf der Seite angezeigt werden. Erstellen Sie im Navigationsmenü einen Link und Verwenden Sie den Minimalwert 10 und den Maximalwert 100.

Zeit bis zum Schulschluss 2023: Zeigen Sie die Zeit bis zum Schulschluss 2023 in mehreren Varianten an. in Tagen, in Stunden, in Minuten, in Sekunden. Die Werte sollen sich bei jedem Neuaufruf der Seite aktualisieren.

Verwenden Sie ein einigermaßend ansprechendes Design für Ihre Seiten.

# Lösungsidee:

Ich habe die Aufgabe anhand der Angabe gelöst. Zuerst habe ich mich um die Zufallszahl gekümmert, dann um die Seite mit den Zeiten bis zum Schulschluss. Zum Schluss habe ich die Index-Seite noch fertig gemacht und alles ein wenig aufgehübscht. Hierbei habe ich mich an einem schlichten Design mit User Feedback orientiert.

Abgabe:

* Gesamtes Projekt als Zip-Archiv
* Ins Word-Dokument kopieren:

\_Layout.cshtml

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>@ViewData["Title"] - Hue04</title>

<**link** rel="stylesheet" href="~/lib/bootstrap/dist/css/bootstrap.min.css" />

<**link** rel="stylesheet" **href**="~/css/site.css" **asp-append-version**="true" />

<**link** rel="stylesheet" **href**="~/Hue04.styles.css" **asp-append-version**="true" />

</head>

<body>

<header>

<nav class="navbar navbar-expand-sm navbar-toggleable-sm navbar-light bg-white border-bottom box-shadow mb-3">

<div class="container">

<**a** class="navbar-brand" **asp-area**="" **asp-page**="/Index">Hue04</**a**>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target=".navbar-collapse" aria-controls="navbarSupportedContent"

aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="navbar-collapse collapse d-sm-inline-flex justify-content-between">

<ul class="navbar-nav flex-grow-1">

<li class="nav-item">

<button>

<**a** class="nav-link text-dark" **asp-area**="" **asp-page**="/Index">Home</**a**>

</button>

</li>

<li class="nav-item">

<button>

<**a** class="nav-link text-dark" **asp-area**="" **asp-page**="/TimeTillEndOfSchool">Time till End of School</**a**>

</button>

</li>

<li class="nav-item">

<button>

<**a** class="nav-link text-dark" **asp-area**="" **asp-page**="/RandomValue">Generate Randome Numbers</**a**>

</button>

</li>

<li class="nav-item">

<button>

<**a** class="nav-link text-dark" **asp-area**="" **asp-page**="/RandomValue" **asp-route-min**=10 **asp-route-max**=100>Random Number between 10 and 100</**a**>

</button>

</li>

<li class="nav-item">

<button>

<**a** class="nav-link text-dark" **asp-area**="" **asp-page**="/Privacy">Privacy</**a**>

</button>

</li>

</ul>

</div>

</div>

</nav>

</header>

<div class="container">

<main role="main" class="pb-3">

@RenderBody()

</main>

</div>

<footer class="border-top footer text-muted">

<div class="container">

&copy; 2023 - Hue04 - <**a** **asp-area**="" **asp-page**="/Privacy">Privacy</**a**>

</div>

</footer>

<**script** src="~/lib/jquery/dist/jquery.min.js"></**script**>

<**script** src="~/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></**script**>

<**script** **src**="~/js/site.js" **asp-append-version**="true"></**script**>

@await RenderSectionAsync("Scripts", required: false)

</body>

</html>

* + Alle selbst erstellten Razor-Pages (.cshtml und cshtml.cs)

RandomValue.cshtml

@page

@model Hue04.Pages.RandomValueModel

@{

ViewData["Title"] = "Generate Random Value";

}

<div class="container-md text-center">

<h1 class="display-4">Random Number Generator</h1>

<**form** method="post" **asp-page-handler**="GenerateRandomValue">

<div class="row">

<div class="col-6">

Min:

<input type="number" id="min" name="min" value="@Model.Min"/>

</div>

<div class="col-6">

Max:

<input type="number" id="max" name="max" value="@Model.Max"/>

</div>

</div>

<br/>

<button type="submit">Generate random value</button>

</**form**>

<div class="row">

<p><h2 class="display-6">Random number: @Model.Result</h2></p>

</div>

</div>

RandomValue.cshtml.cs

using Microsoft.AspNetCore.Mvc;

using Microsoft.AspNetCore.Mvc.RazorPages;

namespace Hue04.Pages

{

public class RandomValueModel : PageModel

{

public string Result { get; set; } = "Try generating a random number by clicking on button.";

public int Min { get; set; }

public int Max { get; set; }

Random random = new Random();

public void OnGet(int? min, int? max)

{

SetValues(min, max);

GenerateRandomValue();

}

[HttpPost]

public void OnPostGenerateRandomValue(int? min, int? max)

{

SetValues(min, max);

GenerateRandomValue();

}

private void GenerateRandomValue()

{

this.Result = random.Next(Min, Max).ToString();

}

private void SetValues(int? min, int? max)

{

if (min.HasValue)

{

Min = min.Value;

}

else

{

Min = 0;

}

if (max.HasValue)

{

Max = max.Value;

}

else

{

Max = 0;

}

}

}

}

TimeTillEndOfSchool.cshtml

@page

@model Hue04.Pages.TimeTillEndOfSchoolModel

@{

ViewData["Title"] = "Time till End of School";

}

<div class="text-center">

<h1 class="display-4">Time till End of School</h1>

<p><h2 class="display-6">@Model.Days days</h2></p>

<p><h2 class="display-6">@Model.Hours hours</h2></p>

<p><h2 class="display-6">@Model.Minutes minutes</h2></p>

<p><h2 class="display-6">@Model.Seconds seconds</h2></p>

</div>

TimeTillEndOfSchool.cshtml.cs

using Microsoft.AspNetCore.Mvc;

using Microsoft.AspNetCore.Mvc.RazorPages;

namespace Hue04.Pages

{

public class TimeTillEndOfSchoolModel : PageModel

{

public String Days { get; set; }

public String Hours { get; set; }

public String Minutes { get; set; }

public String Seconds { get; set; }

public void OnGet()

{

TimeSpan timeTillEndOfSchool = new DateTime(2023, 06, 30, 9, 20, 0) - DateTime.Now;

Days = Math.Round(timeTillEndOfSchool.TotalDays, 2).ToString();

Hours = Math.Round(timeTillEndOfSchool.TotalHours, 2).ToString();

Minutes = Math.Round(timeTillEndOfSchool.TotalMinutes, 2).ToString();

Seconds = Math.Round(timeTillEndOfSchool.TotalSeconds, 2).ToString();

}

}

}

Index.cshtml

@page

@model IndexModel

@{

ViewData["Title"] = "Home page";

}

<div class="text-center">

<h1 class="display-4">Welcome!</h1>

<img src=@Model.Image style="width:100%; max-width:720px;" />

<p></p>

<div class="row">

<button class="col-3 text-lg-start">

<**a** class="nav-link text-dark" **asp-area**="" **asp-page**="/RandomValue" **asp-route-min**=0 **asp-route-max**=10>Random Number between 0 and 10</**a**>

</button>

<button class="col-3 text-lg-start">

<**a** class="nav-link text-dark" **asp-area**="" **asp-page**="/RandomValue" **asp-route-min**=0 **asp-route-max**=100>Random Number between 0 and 100</**a**>

</button>

<button class="col-3 text-lg-start">

<**a** class="nav-link text-dark" **asp-area**="" **asp-page**="/RandomValue" **asp-route-min**=0 **asp-route-max**=1000>Random Number between 0 and 1000</**a**>

</button>

<button class="col-3 text-lg-start">

<**a** class="nav-link text-dark" **asp-area**="" **asp-page**="/RandomValue" **asp-route-min**=-10 **asp-route-max**=10>Random Number between -10 and 10</**a**>

</button>

</div>

</div>

Index.cshtml.cs

using Microsoft.AspNetCore.Mvc;

using Microsoft.AspNetCore.Mvc.RazorPages;

namespace Hue04.Pages

{

public class IndexModel : PageModel

{

private readonly ILogger<IndexModel> \_logger;

public String Image { set; get; }

public IndexModel(ILogger<IndexModel> logger)

{

\_logger = logger;

}

public void OnGet()

{

if (DateTime.Now.Minute % 2 == 0)

{

Image = "img/mountain.jpg";

}

else

{

Image = "img/forest.jpg";

}

}

}

}

* + Screenshots

     