Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Институт информационных технологий

Факультет компьютерных технологий

Лабораторная работа № 11

WEB-СЕРВИСЫ

Выполнил: студент гр. 881062 Кузьменко Е.В.

Проверил: Потоцкий Д.С.

Минск 2020

## Задания

## 1*Индивидуальные задания*

**Вариант 1.**

Создать web-службу для авторизации доступа к базе данных.

**Вариант 2**.

Создать web-службу для получения переводов английских слов.

**Вариант 3**.

Создать web-службу для получения курсов валют.

**Вариант 4.**

Создать web-службу для проверки того, что переданное число является простым.

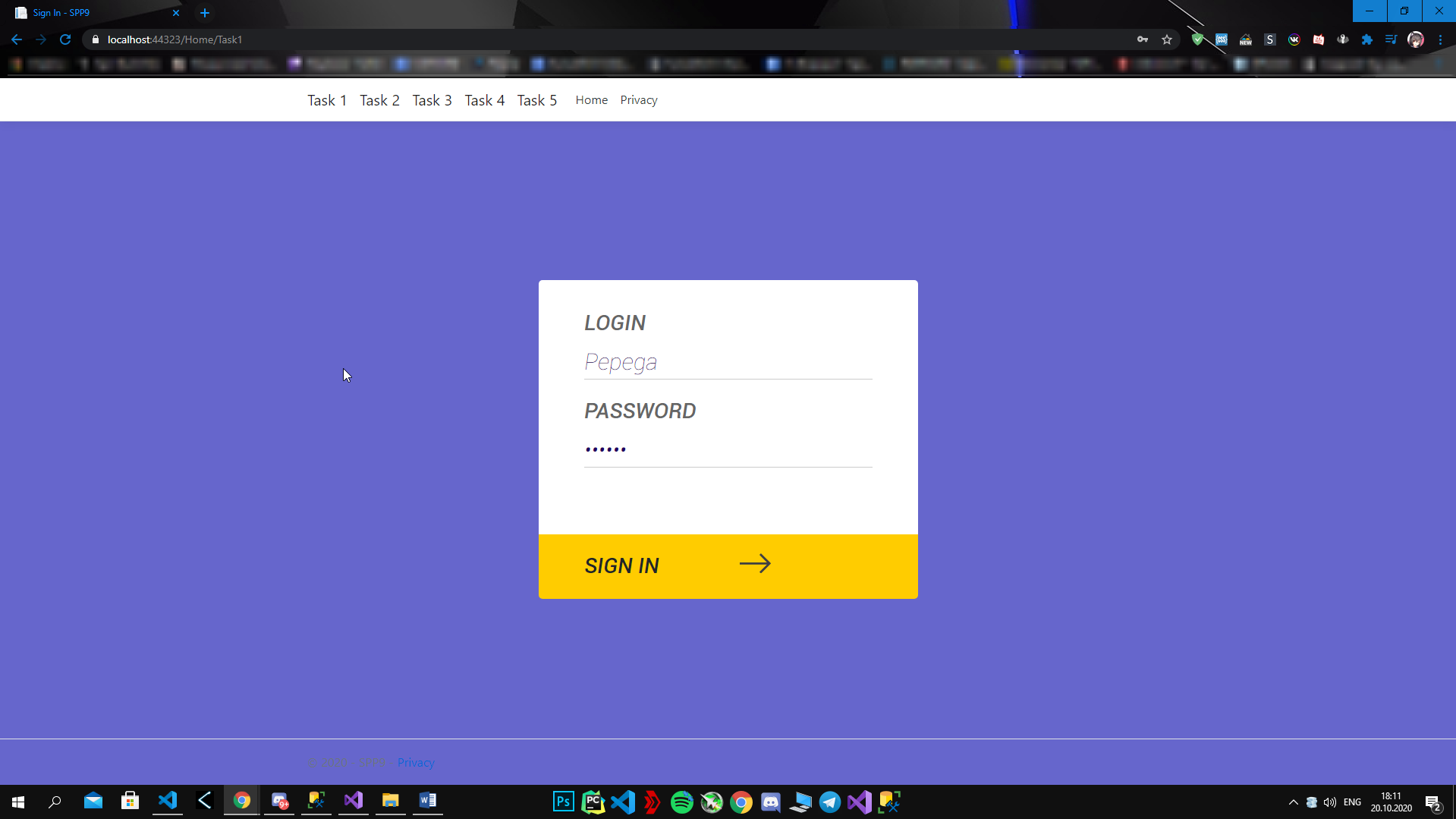
**Вариант 5.**

Создать web-службу для шифрования строки.

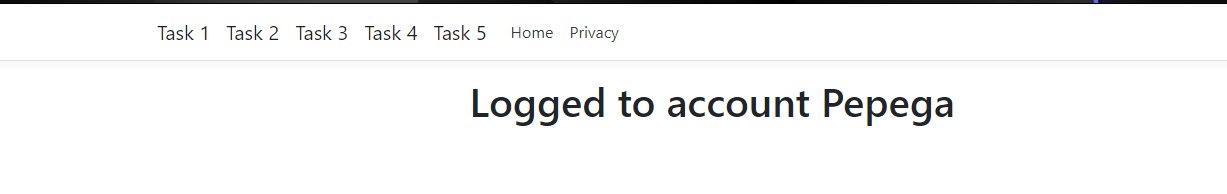
## Скриншот выполнения программы:

**Задание 1**

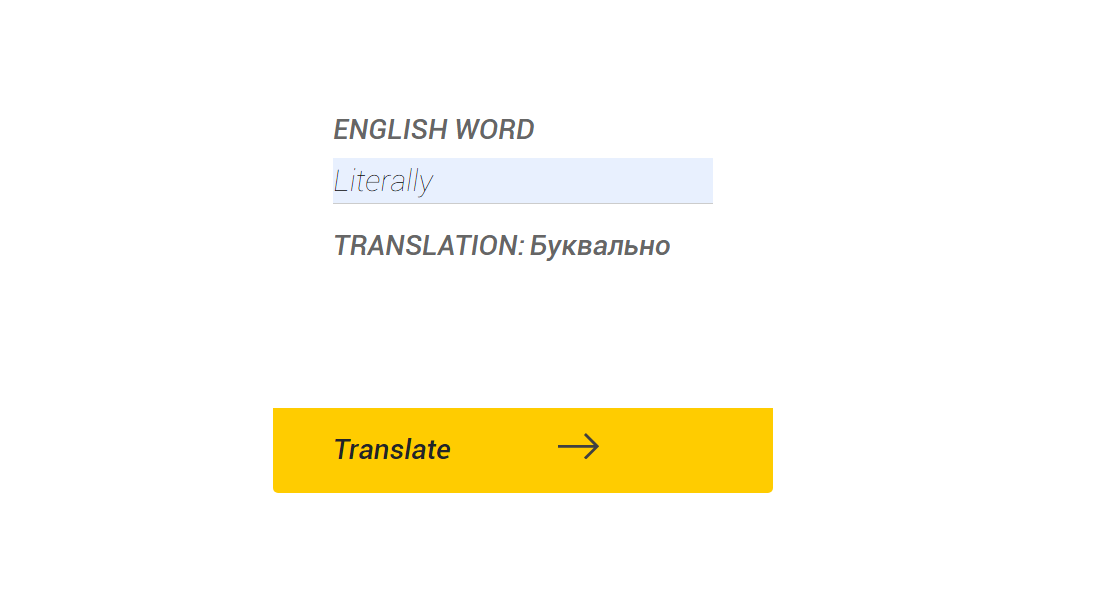
Форма авторизации



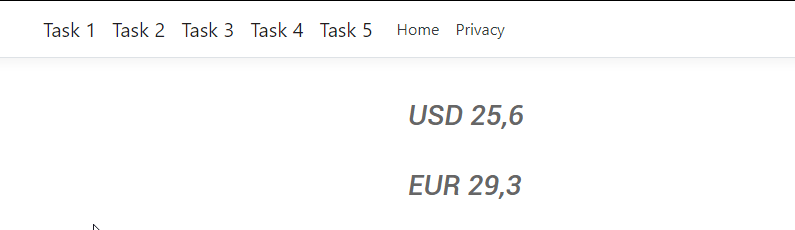
## Сообщение о успешной авторизации



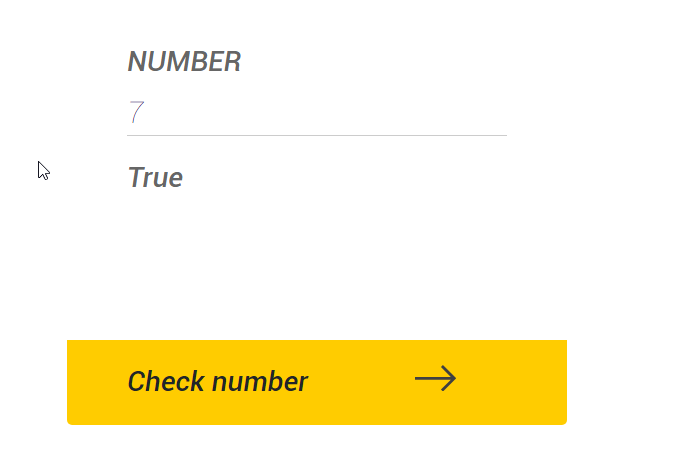
**Задание 2**



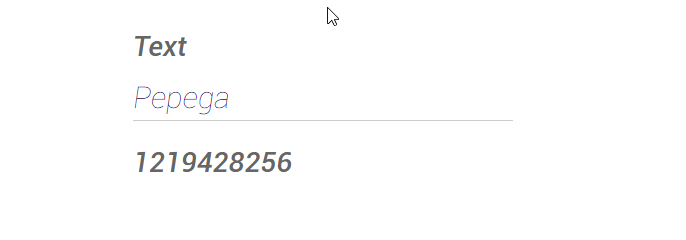
**Задание 3**



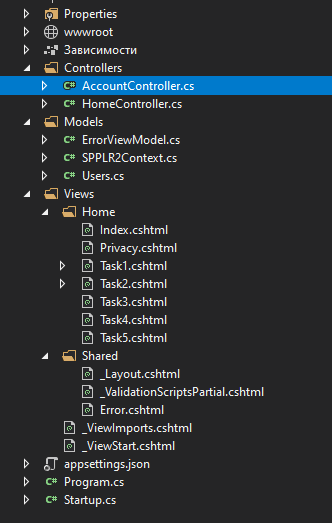
**Задание 4**



**Задание 5**



Структура проекта



## Код программы:

Задание 1

@{

ViewData["Title"] = "Sign In";

}

<body class="background">

<div class="Login-div align-content-center">

@using (Html.BeginForm("LoginToAccount", "Account", FormMethod.Post))

{

<div class="register-content">

<label class="black-caps">LOGIN</label>

<input id="Login" class="Register-input" name="Login" placeholder="Enter your login here" type="text" />

<label class="black-caps">PASSWORD</label>

<input id="pass" class="Register-input" name="Password" placeholder="Enter your password here" type="password" />

<label class="Exeption">@ViewBag.msg</label>

</div>

<div class="create-account loggin\_to\_account" style="cursor:pointer">

<text> SIGN IN </text>

<svg style="fill: rgb(64, 64, 65); width: 41px; height: 26px;">

<polygon points="27.93 0 25.95 1.98 35.85 11.88 0 11.88 0 14.68 35.85 14.68 25.95 24.59 27.93 26.57 41.21 13.28 27.93 0"></polygon>

</svg>

<input class="Create-acc-button" type="submit" value="" />

</div>

}

</div>

</body>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Http;

using Microsoft.AspNetCore.Mvc;

namespace SPP9.Controllers

{

//контроллер логина в аккаунт

public class AccountController : Controller

{

//метод установки куки

public void setCookie(string Key, string Value, int expires)

{

CookieOptions cookieOptions = new CookieOptions();

cookieOptions.Expires = DateTime.Now.AddDays(expires);

Response.Cookies.Append(Key, Value, cookieOptions);

}

//логин в аккаунт

public ActionResult LoginToAccount(string Login, String Password)

{

if (Login != null && Password != null)

{

using (Models.SPPLR2Context context = new Models.SPPLR2Context())

{

//поиск аккаунта в базе данных

var user = context.Users.FirstOrDefault(r => r.Username == Login && r.Password == Password);

if (user != null)

{

//если аккант найден установка куки с именем юзера

setCookie("Login\_data", user.Username, 7);

//редирект на страницу с информацеей о логине

return RedirectToRoute(new { controller = "Home", action = "Logged" });

}

else

{

//редирект на окно логина если аккаунт не найден

return RedirectToRoute(new { controller = "Home", action = "Task1" });

}

}

}

else

{

//редирект на страницу логина если не все поля заполнены

return RedirectToRoute(new { controller = "Home", action = "Task1" });

}

}

}

}

Задание 2-5 бэкэнд

using System;

using System.Collections;

using System.Collections.Generic;

using System.Diagnostics;

using Microsoft.AspNetCore.Mvc;

using Microsoft.Extensions.Logging;

using SPP9.Models;

namespace SPP9.Controllers

{

public class HomeController : Controller

{

private readonly ILogger<HomeController> \_logger;

public HomeController(ILogger<HomeController> logger)

{

\_logger = logger;

}

//проверка простое ли число

public static bool IsPrimeNumber(uint n)

{

var result = true;

if (n > 1)

{

for (var i = 2u; i < n; i++)

{

if (n % i == 0)

{

result = false;

break;

}

}

}

else

{

result = false;

}

return result;

}

public IActionResult Index()

{

return View();

}

public IActionResult Logged()

{

//получение куки

ViewBag.msg = string.Format("Logged to account {0}" , Request.Cookies["Login\_data"]);

return View("Index");

}

//Task1

public IActionResult Task1()

{

return View();

}

//Task2

public IActionResult Task2()

{

return View();

}

public IActionResult Translate(string Login)

{

switch(Login){

case "Literally":

ViewBag.Translated = "Буквально";

break;

case "Dog":

ViewBag.Translated = "Собака";

break;

case "Cat":

ViewBag.Translated = "Кот";

break;

default:

ViewBag.Translated = "Not Found";

break;

}

return View("Task2");

}

public IActionResult Task3()

{

ViewBag.USD = "25,6";

ViewBag.EUR = "29,3";

return View();

}

public IActionResult Task4()

{

return View();

}

public IActionResult Task5()

{

return View();

}

public IActionResult CheckNumber(uint Login)

{

ViewBag.Result = IsPrimeNumber(Login);

return View("Task4");

}

public IActionResult Encrypt(string Login)

{

ViewBag.Result = Login.GetHashCode();

return View("Task5");

}

public IActionResult Privacy()

{

return View();

}

[ResponseCache(Duration = 0, Location = ResponseCacheLocation.None, NoStore = true)]

public IActionResult Error()

{

return View(new ErrorViewModel { RequestId = Activity.Current?.Id ?? HttpContext.TraceIdentifier });

}

}

}