



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(национальный исследовательский университет)»

Институт №3 «Системы управления, информатика и электроэнергетика»

Кафедра 304 «Вычислительные машины, системы и сети»

Отчёт по лабораторной работе № 1

по учебной дисциплине «Разработка мобильных и web-приложений»

по теме Взаимодействие с web-клиентом

Выполнила студентка группы МЗО-211СВ-24:

Хамдан Шаза

Принял:

Титов Юрий Павлович

Москва, [2025]

1. Цель работы:

Разработать веб-приложение, работающее в локальной сети, с взаимодействием через браузер.

В рамках работы реализовать генератор случайных чисел с вводом диапазона пользователем, асинхронной передачей данных между фронтендом и бэкендом и адаптивной версткой для мобильных и десктопных устройств.

2. Теоретическая часть

Backend:

- Используется .NET Core для создания API
- Метод `rnd.Next(min, max + 1)` генерирует случайное число исключая `max` и `min`.
- Возвращает результат в формате JSON: {"number":42}

```
Controllers > RandomController.cs > ...
1  using Microsoft.AspNetCore.Mvc;
2
3  namespace RandomNumberGenerator.Controllers
4  {
5      [ApiController]
6      [Route("[controller]")]
7      public class RandomController : ControllerBase
8      {
9          [HttpGet]
10         public IActionResult GetRandomNumber(int min, int max)
11         {
12             if (min > max)
13                 return BadRequest("Min cannot be greater than Max.");
14
15             Random rnd = new Random();
16             int number = rnd.Next(min + 1, max);
17             return Ok(new { number });
18         }
19     }
20 }
```

Frontend:

- HTML: форма с двумя полями для чисел и кнопкой «Generate».

```
wwwroot > <> index.html > html > head > meta
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Random Number Generator</title>
7      <link rel="stylesheet" href="style.css">
8  </head>
9  <body>
10     <h1> Random Number Generator</h1>
11
12     <div class="container">
13         <label>Minimum:</label>
14         <input type="number" id="min" placeholder="Enter min value">
15
16         <label>Maximum:</label>
17         <input type="number" id="max" placeholder="Enter max value">
18
19         <button id="generate">Generate</button>
20
21         <p id="result"></p>
22     </div>
23
24     <script src="script.js"></script>
25 </body>
26 </html>
27
```

- CSS: стилизация и адаптивность через `@media (max-width: 600px)`.

```
49
50  /* Responsive for mobile */
51  @media (max-width: 600px) {
52      .container {
53          width: 90%;
54      }
55  }
56  @media (max-width: 270px) {
57      .container { width: 80%; font-size: 12px; }
58  }
```

- JS: Реализовать асинхронный вызов JS (`fetch + async/await`) для обращения к серверу.

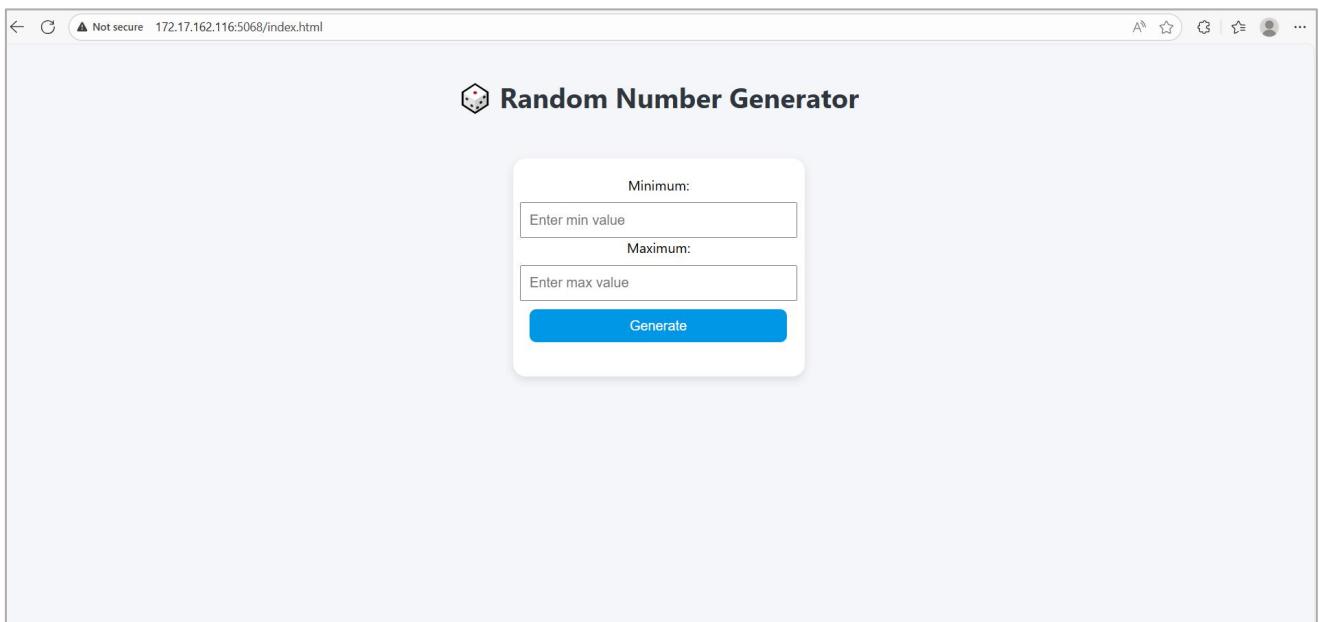
```

wwwroot > JS script.js > ...
1  document.getElementById('generate').addEventListener('click', async () => {
2    const min = Number(document.getElementById('min').value);
3    const max = Number(document.getElementById('max').value);
4    const result = document.getElementById('result');
5
6    if (isNaN(min) || isNaN(max)) {
7      result.textContent = "⚠ Please enter both numbers.";
8      return;
9    }
10
11   if (min > max) {
12     result.textContent = "⚠ Min cannot be greater than Max.";
13     return;
14   }
15
16   try {
17     // Get the current host IP address dynamically
18     const baseUrl = window.location.origin;
19     const response = await fetch(` ${baseUrl}/Random?min=${min}&max=${max}`);
20
21     if (!response.ok) throw new Error("Server error");
22
23     const data = await response.json();
24     result.textContent = ` Random Number: ${data.number}`;
25   } catch (error) {
26     result.textContent = ` ✘ Error: ${error.message}`;
27   }
28 });

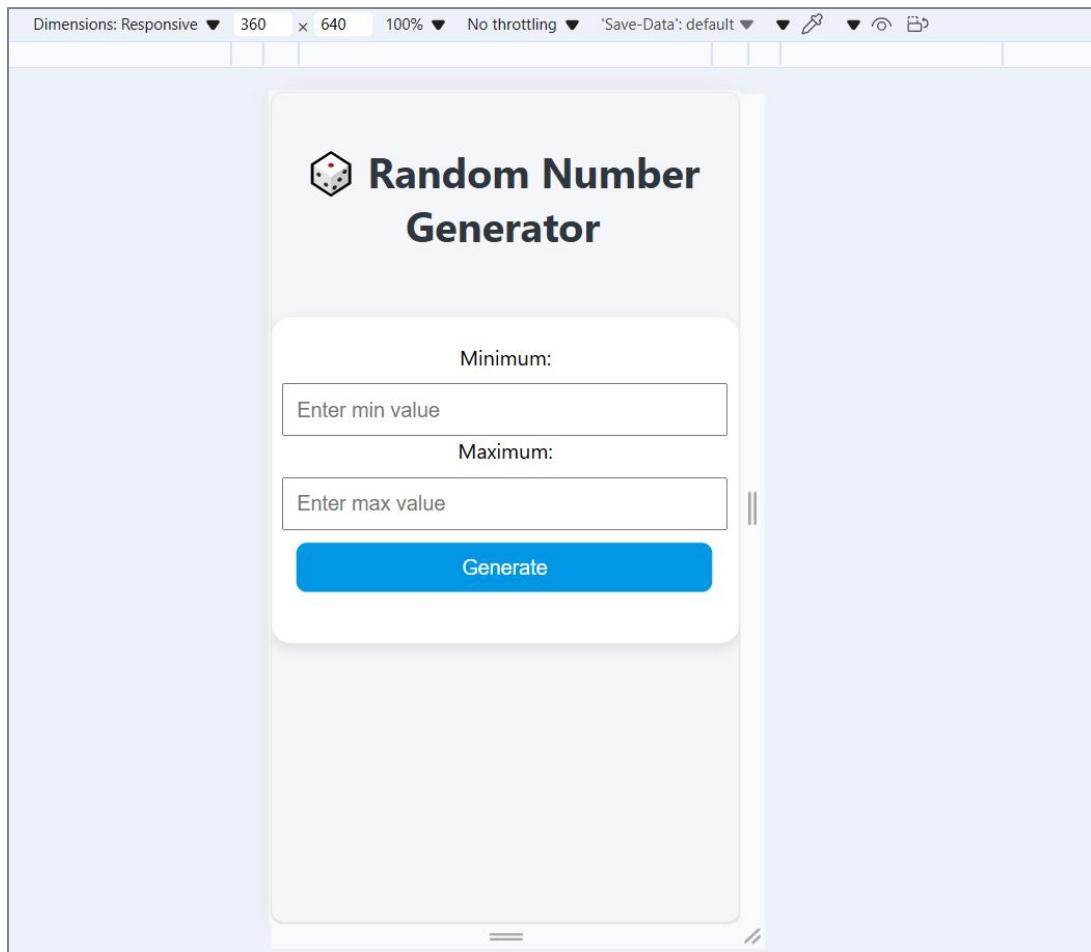
```

Конечный интерфейс:

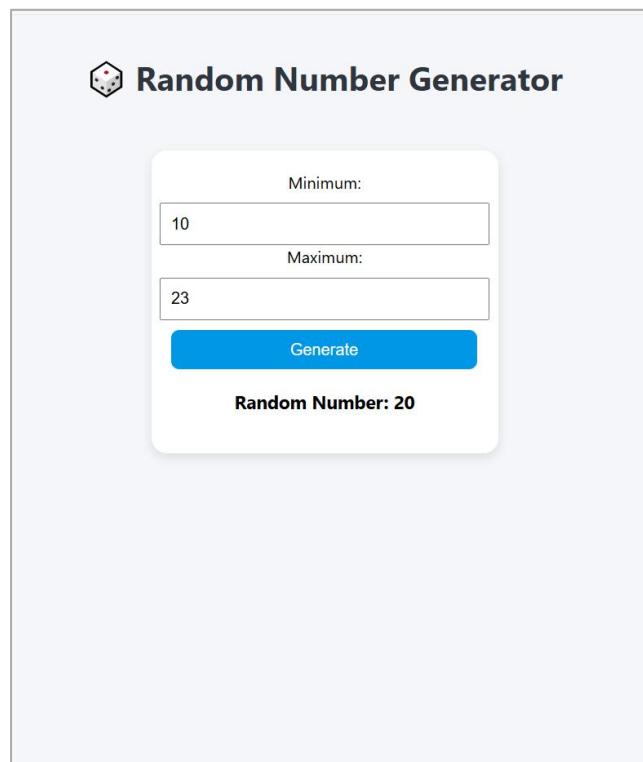
- В браузере:



➤ На интерфейсе размером (360 X 640):



Пример применения:



20:40

58%



Random Numb...

▲ 192.168.90.44:5068



Random Number Generator

Minimum:

Enter min value

Maximum:

Enter max value

Generate



Выводы

В ходе лабораторной работы было создано веб-приложение генератора случайных чисел, которое корректно работает как на десктопном браузере, так и на мобильном (Aurora OS).

Реализована асинхронная работа JS с backend, проверена обработка некорректных данных, обеспечена адаптивная верстка для разных экранов.