|  |  |
| --- | --- |
| **эРоссийский университет транспорта (МИИТ)**  **Институт транспортной техники и систем управления**  **Кафедра «Управление и защита информации»** | |
| **Отчет**  **по практическому заданию**  **по теме «Разработка с использованием Vue.js. Задание 4»**  **по дисциплине**  **«Web-программирование»** | |
|  | Выполнил:  Cтудент группы ТКИ-541,  Николаев А.М.  Проверил:  Доцент кафедры УиЗИ к.т.н.,  Сафронов А. И. |
| г. Москва - 2023 | |

ОГЛАВЛЕНИЕ

[Цель работы 3](#_Toc147319431)

[Формулировка задачи 3](#_Toc147319432)

[Технология локального подключения фреймворка *Vue.js* 3](#_Toc147319433)

[Таблица соответствия переменных и методов 4](#_Toc147319434)

[ХОД РАБОТЫ 4](#_Toc147319435)

[Содержательная часть по параметрической вставке 4](#_Toc147319436)

[ВЫВОД 9](#_Toc147319437)

Цель работы

Разработать два локальных одностраничных веб-приложения (LSPWA) с использованием фреймворка Vue.js, демонстрирующих использование директив v-bind и v-on для реализации интерактивности на клиентской стороне без перезагрузки страницы.

Формулировка задачи

Приложение 1: Реализовать параметрическую подстановку в тег <ol>, установив начальное число и тип нумерации через v-bind.

Приложение 2: Создать интерфейс с кнопками, позволяющими изменять значение input типа range с использованием v-on для обработки событий клика.

**Не использовать:** *jQuerry*, «чистый» *JavaScript,* каскадные таблицы стилей (*CSS*), директиву *v-for*, директивы *v-if*, *v-else*, *v-else-if*, *v-model*, теги <*template*>.

**Реализовать:**

– локально (*LSPWA*), не прибегая к инструментарию *Node.js* и *npm* (*Node Package Manager*);

– в файле с именем *index\_Familiya\_I\_O.html*

# Технология локального подключения фреймворка *Vue.js*

Vue.js можно подключить локально, добавив в ваш HTML следующий тег script:

<script src="https://cdn.jsdelivr.net/npm/vue@2"></script>

Этот тег подключает Vue.js напрямую из CDN, и нет необходимости устанавливать что-либо через Node.js или npm.

# Таблица соответствия переменных и методов

|  |  |  |
| --- | --- | --- |
| Переменная/метод | Описание | Приложение |
| startNumber | Начальное число для нумерации списка | 1 |
| numberType | Тип нумерации списка (римские цифры) | 1 |
| value | Текущее значение ползунка | 2 |
| min | Минимальное значение ползунка | 2 |
| max | Максимальное значение ползунка | 2 |
| changeValue(step) | Метод изменения значения ползунка | 2 |
| resetValue() | Метод сброса значения ползунка | 2 |

# ХОД РАБОТЫ

## Содержательная часть по параметрической вставке

Код index\_Nikolaev\_A\_M\_1.html:

<!DOCTYPE html>

<html lang="ru">

<head>

    <meta charset="UTF-8">

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

    <title>Index\_Nikolaev\_A\_M\_1</title>

    <script src="https://cdn.jsdelivr.net/npm/vue@2"></script>

</head>

<body>

    <div id="app1">

        <ol :start="startNumber" :type="numberType">

            <li v-for="item in listItems">{{ item }}</li>

        </ol>

    </div>

    <script src="script1.js"></script>

</body>

</html>

Код script1.js:

new Vue({

    el: '#app1',

    data: {

        startNumber: 10,

        numberType: 'I',

        listItems: ['Пункт 1', 'Пункт 2', 'Пункт 3', 'Пункт 4', 'Пункт 5']

    }

});

|  |
| --- |
|  |
| Рисунок 1 – Вид страницы |

|  |
| --- |
|  |
| Рисунок 2 – Сеть Петрни |

Содержательная часть по методам и событиям

Код index\_Nikoilaev\_A\_M\_2.html:

<!DOCTYPE html>

<html lang="ru">

<head>

    <meta charset="UTF-8">

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

    <title>Index\_Nikolaev\_A\_M\_2</title>

    <script src="https://cdn.jsdelivr.net/npm/vue@2"></script>

</head>

<body>

    <div id="app2">

        <input type="range" :min="min" :max="max" v-bind:value="value" readonly>

        <p>{{ value }}</p>

        <button v-on:click="changeValue(0.1)">+0.1</button>

        <button v-on:click="changeValue(0.5)">+0.5</button>

        <button v-on:click="changeValue(1)">+1</button>

        <button v-on:click="resetValue">Сброс</button>

    </div>

    <script src="script2.js"></script>

</body>

</html>

Код script2.js:

new Vue({

    el: '#app2',

    data: {

        value: -5,

        min: -5,

        max: 7

    },

    methods: {

        changeValue: function(step) {

            let newValue = parseFloat((this.value + step).toFixed(1));

            if (newValue > this.max) {

                this.value = this.min;

            } else {

                this.value = newValue;

            }

        },

        resetValue: function() {

            this.value = this.min;

        }

    }

});

|  |
| --- |
|  |
| Рисунок 3 – Начальная страница сайта |

|  |
| --- |
|  |
| Рисунок 4 – Изменение на сайте |

|  |
| --- |
|  |
| Рисунок 5 – Сеть Петри |

ВЫВОД

Работа с фреймворком Vue.js позволяет эффективно и без значительных трудозатрат реализовывать динамическую интерактивность на клиентской стороне. Директивы v-bind и v-on предоставляют мощные инструменты для работы с атрибутами элементов и обработки событий соответственно, что демонстрируется в разработанных веб-приложениях.