|  |  |
| --- | --- |
| **Российский университет транспорта (МИИТ)**  **Институт транспортной техники и систем управления**  **Кафедра «Управление и защита информации»** | |
| **Отчет**  **по лабораторной работе №3**  **по дисциплине «Web-программирование» Вариант 12** | |
|  | Выполнили:  Студенты группы ТКИ-541  Кузнецов К. К.  Степанов Д. Е. |
|  | Проверил:  Доцент кафедры УиЗИ к.т.н. с.н.с.  Сафронов А. И. |
| Москва 2024 | |

**Оглавление**

[1 Цель работы 3](#_Toc177327927)

[2 Формулировка задачи 4](#_Toc177327928)

[3 Детализированное текстовое описание ситуации 5](#_Toc177327929)

[4 Сеть Петри – схема ситуации 6](#_Toc177327930)

[4.1 Полная 6](#_Toc177327931)

[4.2 Краткая 7](#_Toc177327932)

[4.3 Описание краткой сети Петри 8](#_Toc177327933)

[5 Вывод 9](#_Toc177327934)

1. Цель работы

Разработать четыре простых локальных одностраничных web-приложение (LSPWA) под управлением фреймворка Vue.js на языке JavaScript в соответствии с указаниями вариантов индивидуального задания (покрываются три раздела последовательного вычислительного процесса и один раздел разветвляющегося вычислительного процесса).

1. Формулировка задачи

**Вариант 12.** Реализовать:

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

– в файле с именем index01\_Familiya\_I\_O1\_Familiya\_I\_O2.html параметрическую вставку;

– в файле с именем index02\_Familiya\_I\_O1\_Familiya\_I\_O2.html параметрическую связку;

– в файле с именем index03\_Familiya\_I\_O1\_Familiya\_I\_O2.html работу с методами и событиями;

– в файле с именем index04\_Familiya\_I\_O1\_Familiya\_I\_O2.html условную отрисовку.

**v-bind:**

Настроить при помощи параметрической подстановки тег <ol></ol>. Предусмотреть пять позиций в списке. Нумерация позиций (type) в списке выполняется в заглавном римском стиле, начиная с 10 (start).

**v-on:**

Предусмотреть три экранных кнопки <button></button>, позволяющие выполнять различный шаг (0.1, 0.5, 1.0) по заданному диапазону (input range) [-5; 7]. Предусмотреть сброс в начальное положение по нажатии по четвёртой экранной кнопке <button></button>. Выполнять округление значения до одного знака после плавающей точки. Если при очередном шаге должен произойти переход за границы установленного диапазона значений – принудительно выставлять минимальное значение диапазона.

**v-model:**

В обрамлённой области масштабируемой векторной графики <svg></svg> размером 800 на 600 пикселей по ширине и высоте, соответственно, в псевдослучайно сгенерированной позиции изображена окружность радиуса 10 пикселей. Областью символьных клавиш «a», «w», «s», «d», аналогичных области стрелок курсора («», «», «», «») на клавиатуре менять с шагом в 10 пикселей положение окружности в соответствующем нажатой клавише направлении (реализовать шаг по отпускании клавиши). Не позволять окружности подступать краями (не центром) к границам <svg></svg> ближе, чем на 10 пикселей даже при инициализации её начального положения.

**v-if:**

Предусмотреть 5 элементов «checkbox». Каждый предназначен для активации и отображения на веб-странице своего и только своего шаблона <template></template>. Первый шаблон содержит информацию о библиографической ссылке на русском языке, второй шаблон аналогичную информацию на английском языке, третий шаблон демонстрирует библиографическую ссылку в формате ГОСТ (пример: 1. Баранов, Л. А. Интеллектуальное централизованное управление движением внеуличного городского железнодорожного транспорта в условиях интенсивного движения / Л. А. Баранов // Надежность. – 2021. – Т. 21, № 2. – С. 17-23), четвёртый – библиографическую ссылку в формате Scopus (пример: [1] Baranov L A 2021 Intelligent centralized control of off-street urban rail transport in heavy traffic conditions (Reliability vol 21) pp 17–23), пятый – транслитерацию русскоязычной информации в формат Scopus. Учитывать при вводе: «Номер по порядку», «Фамилию автора», «Имя автора», «Отчество автора», «Название статьи», «Название журнала», «Год выпуска», «Номер тома», «Номер выпуска», «Номера страниц».

1. Детализированное текстовое описание ситуации

Аренда автомобиля начинается с составления маршрута, а также выбора агрегатора, предоставляющего соответствующие услуги. Необходимо удостовериться в том, что сам маршрут и конечная его точка находятся в разрешенной для поездки и парковки зоне у конкретного агрегатора.

Следующий шаг для выбора агрегатора - выбор наиболее подходящего варианта автомобиля. В этом шаге необходимо ответить на следующие вопросы:

* какие(/ой) из агрегаторов в данный момент предоставляют автомобили в ближайшей территориальной доступности (или на необходимой территории)?
* какие(/ой) из агрегаторов в данный момент предоставляют автомобили необходимого класса?
* какие(/ой) из агрегаторов в данный момент предлагают наиболее выгодные цены на аренду?

Впоследствии пользователь выбирает понравившийся автомобиль и бронирует его. Следующий этап – проверка состояния автомобиля, который включает в себя фиксирование новых (тех, о которых агрегатор не в курсе) повреждений, повреждений и грязи в салоне, а также проверка исправности автомобиля. На основе этих критериев, пользователем принимается решение начать поездку. В случае неудовлетворения пользователя (например, состоянием салона автомобиля), выбирается следующий автомобиль. Если автомобили у одного агрегатора закончились, имеет смысл перейти к следующему и начать поиск автомобиля заново.

1. Технология локального подключения фреймворка vue.js

<script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>

1. Таблица соответствия переменных и методов, используемых в приложениях

|  |  |  |
| --- | --- | --- |
| Задание | Переменная/метод | Описание |
| 1 | listType | Тип списка |
| 1 | listStart | Номер начального элемента |
| 1 | items | Массив с элементами списка |
| 2 | value | Значение input |
| 2 | min | Минимальное значение input |
| 2 | max | Максимальное значение input |
| 2 | changeValue(step) | Изменение значения в input |
| 2 | resetValue() | Установка минимального значения в input |
| 3 | x | Позиция круга по оси абсцисс |
| 3 | y | Позиция круга по оси ординат |
| 3 | getRandomPosition() | Генерация случайного значения координаты в заданных пределах |
| 3 | moveCircle(step) | Перемещение круга |
| 3 | mounted() | Добавление обработчика события keyup сразу после инициализации приложения |
| 3 | beforeUnmount() | Удаление обработчика события keyup непосредственно перед удалением компонента |
| 4 | selectedTemplate | Массив выбранных чекбоксов |

1. Содержательная часть по параметрической вставке
   1. Код web-приложения

<html>

<head>

<script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>

</head>

<body>

<div id="app">

<ol :type="listType" :start="listStart">

<li>{{ items[0] }}</li>

<li>{{ items[1] }}</li>

<li>{{ items[2] }}</li>

<li>{{ items[3] }}</li>

<li>{{ items[4] }}</li>

</ol>

</div>

<script>

const app = Vue.createApp({

data() {

return {

listType: 'I',

listStart: 10,

items: ['Item 1', 'Item 2', 'Item 3', 'Item 4', 'Item 5']

};

}

});

app.mount('#app');

</script>

</body>

</html>

* 1. Отображение в браузере

|  |
| --- |
|  |
| 1. – Состояние при listStart == 10 |

|  |
| --- |
|  |
| 1. – Состояние при listStart == 666 |

1. Содержательная часть по параметрической связке
   1. Код web-приложения

<html></html>

<head>

<script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>

</head>

<body>

<div id="app">

<input type="number" :value="value" :min="min" :max="max" step="0.1" readonly>

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

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

<button v-on:click="changeValue(1.0)">Step 1.0</button>

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

</div>

<script>

const app = Vue.createApp({

data() {

return {

value: -5,

min: -5,

max: 7

};

},

methods: {

changeValue(step) {

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

if (newValue > this.max) {

newValue = this.min;

}

this.value = newValue;

},

resetValue() {

this.value = this.min;

}

}

});

app.mount('#app');

</script>

</body>

</html>

* 1. Отображение в браузере

|  |
| --- |
|  |
| 1. – Начальное состояние |

|  |
| --- |
|  |
| 1. – Состояние при нажатии Step 0.1 |

|  |
| --- |
|  |
| 1. – Состояние при нажатии Step 0.5 |
|  |
| 1. – Состояние при нажатии Step 1.0 |

|  |
| --- |
|  |
| 1. – Состояние при выходе за границы |
|  |
| 1. – Состояние при нажатии кнопки Reset |

1. Содержательная часть по методам и событиям
   1. Код web-приложения

<html></html>

<head>

<script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>

</head>

<body>

<div id="app">

<svg width="800px" height="600px" style="border:1px solid black;">

<circle :cx="x" :cy="y" r="10px" fill="blue"></circle>

</svg>

</div>

<script>

const app = Vue.createApp({

data() {

return {

x: this.getRandomPosition(20, 780),

y: this.getRandomPosition(20, 580)

};

},

methods: {

getRandomPosition(min, max) {

return Math.ceil(Math.random() \* (max - min + 1) / 10) \* 10 + min;

},

moveCircle(event) {

const step = 10;

switch (event.key) {

case 'a':

case 'ArrowLeft':

if (this.x - step >= 20) this.x -= step;

break;

case 'w':

case 'ArrowUp':

if (this.y - step >= 20) this.y -= step;

break;

case 's':

case 'ArrowDown':

if (this.y + step <= 580) this.y += step;

break;

case 'd':

case 'ArrowRight':

if (this.x + step <= 780) this.x += step;

break;

}

}

},

mounted() {

window.addEventListener('keyup', this.moveCircle);

},

beforeUnmount() {

window.removeEventListener('keyup', this.moveCircle);

}

});

app.mount('#app');

</script>

</body>

</html>

* 1. Отображение в браузере

|  |
| --- |
|  |
| 1. – Начальное состояние |

|  |
| --- |
|  |
| 1. – Состояние на границах |

* 1. Сеть Петри

|  |
| --- |
|  |
| 1. – Сеть Петри |

1. Содержательная часть по условной отрисовке
   1. Код web-приложения

<html>

<head>

<script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>

</head>

<body>

<div id="app">

<div>

<label><input type="checkbox" v-model="selectedTemplate" value="1"> Шаблон 1 (Русский)</label>

<label><input type="checkbox" v-model="selectedTemplate" value="2"> Шаблон 2 (Английский)</label>

<label><input type="checkbox" v-model="selectedTemplate" value="3"> Шаблон 3 (ГОСТ)</label>

<label><input type="checkbox" v-model="selectedTemplate" value="4"> Шаблон 4 (Scopus)</label>

<label><input type="checkbox" v-model="selectedTemplate" value="5"> Шаблон 5 (Транслитерация)</label>

</div>

<template v-if="selectedTemplate.includes('1')">

<div>

<h3>Библиографическая ссылка на русском языке</h3>

1. Баранов, Л. А. Интеллектуальное централизованное управление движением внеуличного городского железнодорожного транспорта в условиях интенсивного движения | Надежность - 2021 | Том 21 | Выпуск № 2. | Страницы 17-23

</div>

</template>

<template v-if="selectedTemplate.includes('2')">

<div>

<h3>Библиографическая ссылка на английском языке</h3>

1. Baranov L A Intelligent centralized control of off-street urban rail transport in heavy traffic conditions | Reliability - 2021 | Volume 21 | Release № 2. | Pages 17-23

</div>

</template>

<template v-if="selectedTemplate.includes('3')">

<div>

<h3>Библиографическая ссылка в формате ГОСТ</h3>

1. Баранов, Л. А. Интеллектуальное централизованное управление движением внеуличного городского железнодорожного транспорта в условиях интенсивного движения / Л. А. Баранов // Надежность. – 2021. – Т. 21, № 2. – С. 17-23

</div>

</template>

<template v-if="selectedTemplate.includes('4')">

<div>

<h3>Библиографическая ссылка в формате Scopus</h3>

[1] Baranov L A 2021 Intelligent centralized control of off-street urban rail transport in heavy traffic conditions (Reliability vol 21) pp 17–23

</div>

</template>

<template v-if="selectedTemplate.includes('5')">

<div>

<h3>Транслитерация русскоязычной информации в формат Scopus</h3>

[1] Baranov L A 2021 Intellektualnoe centralizovannoe upravlenie dvizheniem vneulichnogo gorodskogo zheleznodorozhnogo transporta v usloviyah intensivnogo dvizheniya (Nadezhnost vol 21) pp 17–23

</div>

</template>

</div>

<script>

const app = Vue.createApp({

data() {

return {

selectedTemplate: []

};

}

});

app.mount('#app');

</script>

</body>

</html>

* 1. Отображение в браузере

|  |
| --- |
|  |
| 1. – Начальное состояние |

|  |
| --- |
|  |
| 1. – Отображение всех шаблонов |

|  |
| --- |
|  |
| 1. – Частичное отображение шаблонов |

* 1. Сеть Петри

|  |
| --- |
|  |
| 1. – Сеть Петри |

1. Вывод

В результате проведенной работы были разработаны четыре простых локальных одностраничных web-приложение (LSPWA) под управлением фреймворка Vue.js на языке JavaScript.