**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**

**(МИИТ)**

|  |
| --- |
| **ИНСТИТУТ ТРАНСПОРТНОЙ ТЕХНИКИ И СИСТЕМ УПРАВЛЕНИЯ**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Кафедра «Управление и защита информации»

Лабораторная работа №6

**«Веб-приложения (LSPWA) под управлением фреймворка Vue.js на языке JavaScript»**

по дисциплине

**«Web программирование»**

|  |  |
| --- | --- |
|  | Выполнил: студент группы ТКИ-541  Куминов В. П.  Проверил: к.т.н., доц.  Сафронов А. И. |

**Москва – 2023 г.**

**1. Цель:**

Построить два одностраничных веб-приложения (LSPWA) под управлением фреймворка Vue.js на языке JavaScript.

**2. Постановка задачи:**

Разработать два простых локальных одностраничных веб-приложения (LSPWA) под управлением фреймворка Vue.js на языке JavaScript в соответствии с указаниями вариантов индивидуального задания.

Базовая структура отчёта по работе:

1. Цель работы.

2. Формулировка задачи.

3. Технология локального подключения фреймворка Vue.js.

4. Таблица соответствия переменных и методов, используемых в веб-приложениях.

5. Содержательная часть по параметрической связке (код веб-приложения + отображение в браузере (с учётом изменяющихся состояний) + Сеть Петри).

6. Содержательная часть по условной отрисовке (код веб-приложения + отображение в браузере (с учётом изменяющихся состояний) + Сеть Петри).

7. Вывод.

Не использовать: jQuerry, «чистый» JavaScript, каскадные таблицы стилей (CSS), директиву v-for, Vue.component при условной отрисовке.

Реализовать:

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

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

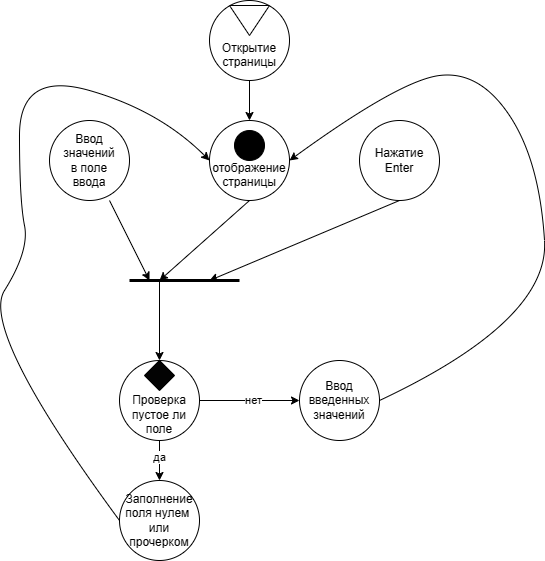
Списки, перебор v-for: 9. В обрамлённую таблицу <table>, состоящую из четырёх столбцов выводится список товаров в формате [Наименование] [Производитель] [Количество] [Цена] (указать как названия столбцов). Под таблицей размещены поля для ввода <input> соответствующих значений наименования товара, производителя, количества и цены их размер соответствует размеру названия столбцов. Нажатие на «Enter» в любом из полей – сигнал для ввода данных. При этом каждое из полей контролируется на отсутствие пустого значения. Для наименования и производителя пустоту менять на серию из пяти подряд идущих дефисов «-». Для количества и цены пустоту менять на нули. Ввод сопровождается отображением нового товара в таблице. Содержимое полей <input> после ввода сбрасывается.

Наблюдатели watch: 9. На страницу выводится обрамлённый элемент масштабируемой векторной графики с значениями ширины и высоты в пропорции 16:9. Начальные приближения ширины и высоты записаны в переменные, значения которых выводятся в соответствующие элементы <input> Синхронно подгонять под пропорции 16:9 любые изменения, связанные как с шириной, так и с высотой.

**3. Ход работы**

Построим сеть Петри первого приложения:

Рисунок 1. Сеть Петри приложения 1



Построим сеть Петри второго приложения:

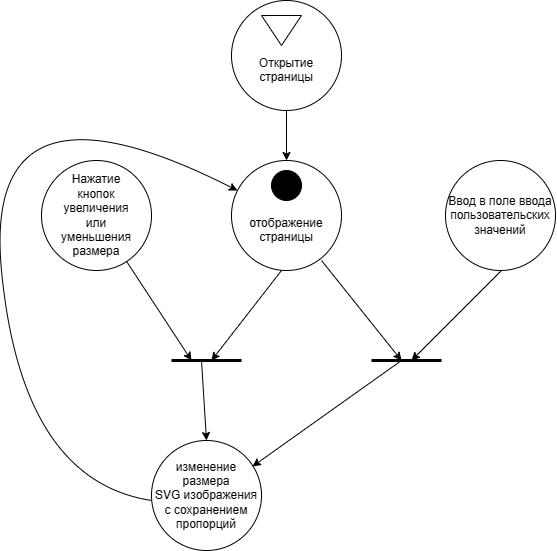


Рисунок 2 Сеть Петри второго приложения

**Код программы приложения 1:**

<!DOCTYPE html>

<html>

<head>

<title>app1\_v-for</title>

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

<style>

table {

width: 100%;

table-layout: fixed;

}

th, td {

width: 20%;

}

input {

width: 24.35%;

margin-right: 0%;

}

</style>

</head>

<body>

<div id="app">

<table border="1">

<tr>

<th>Наименование</th>

<th>Производитель</th>

<th>Количество</th>

<th>Цена</th>

</tr>

<tr v-for="item in items">

<td>{{ item.name }}</td>

<td>{{ item.manufacturer }}</td>

<td>{{ item.quantity }}</td>

<td>{{ item.price }}</td>

</tr>

</table>

<input v-model="newItem.name" @keyup.enter="addItem" placeholder="Наименование">

<input v-model="newItem.manufacturer" @keyup.enter="addItem" placeholder="Производитель">

<input v-model="newItem.quantity" @keyup.enter="addItem" placeholder="Количество">

<input v-model="newItem.price" @keyup.enter="addItem" placeholder="Цена">

</div>

<script>

new Vue({

el: '#app',

data: {

newItem: {name: '', manufacturer: '', quantity: '', price: ''},

items: []

},

methods: {

addItem() {

this.newItem.name = this.newItem.name || '-----';

this.newItem.manufacturer = this.newItem.manufacturer || '-----';

this.newItem.quantity = this.newItem.quantity || '0';

this.newItem.price = this.newItem.price || '0';

this.items.push({...this.newItem});

this.newItem = {name: '', manufacturer: '', quantity: '', price: ''};

}

}

});

</script>

</body>

</html>

Результат работы кода можно увидеть на рисунке 3

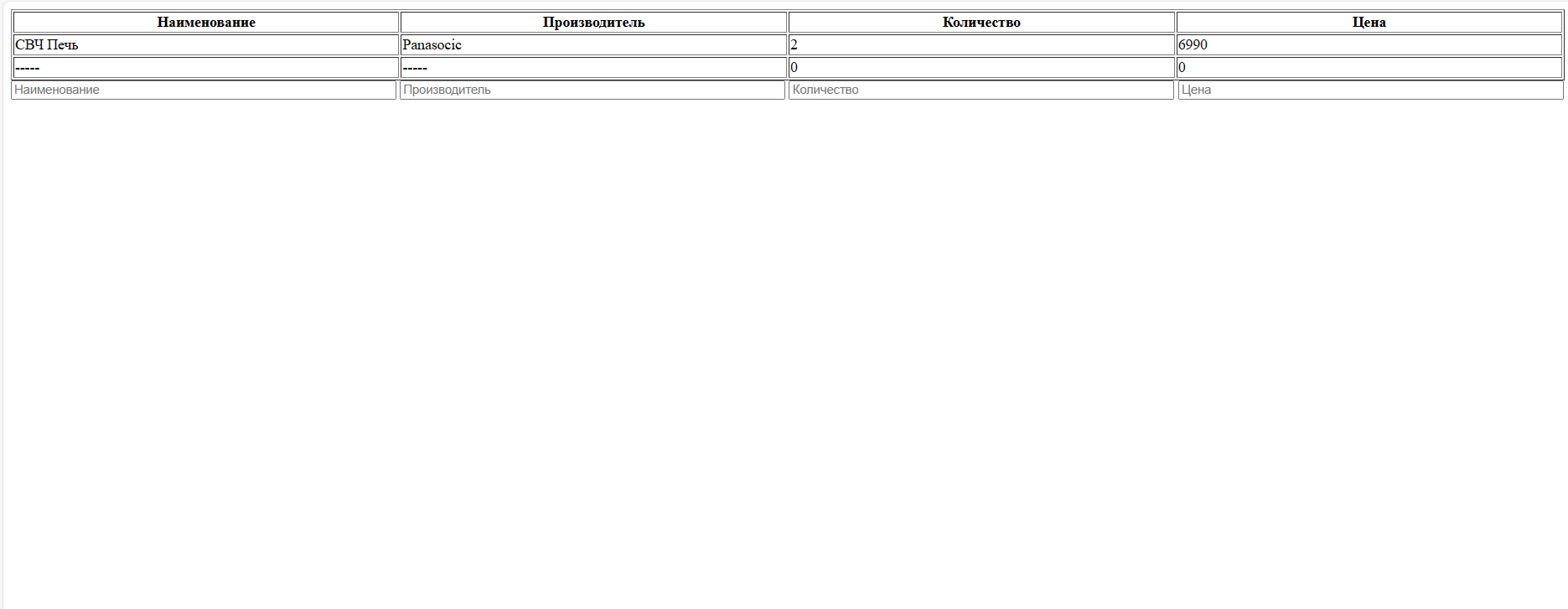


Рисунок 3. Отображение приложения 1.

**Код программы приложения 2:**

<!DOCTYPE html>

<html>

<head>

<title>app2\_watch</title>

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

<style>

#app {

width: 100%;

height: 100%;

display: flex;

justify-content: center;

align-items: center;

}

svg {

border: 1px solid black;

}

</style>

</head>

<body>

<div id="app">

<div>

<label>Ширина: <input v-model.number="width" type="number" min="0"></label><br>

<label>Высота: <input v-model.number="height" type="number" min="0"></label><br>

<svg :width="width" :height="height">

<image :href="'160559.svg'" :width="width" :height="height"/>

</svg>

</div>

</div>

<script>

new Vue({

el: '#app',

data: {

width: 160,

height: 90

},

watch: {

width(val) {

this.height = val \* (9 / 16);

},

height(val) {

this.width = val \* (16 / 9);

}

}

});

</script>

</body>

</html>

Результат работы кода можно увидеть на рисунках 4-5

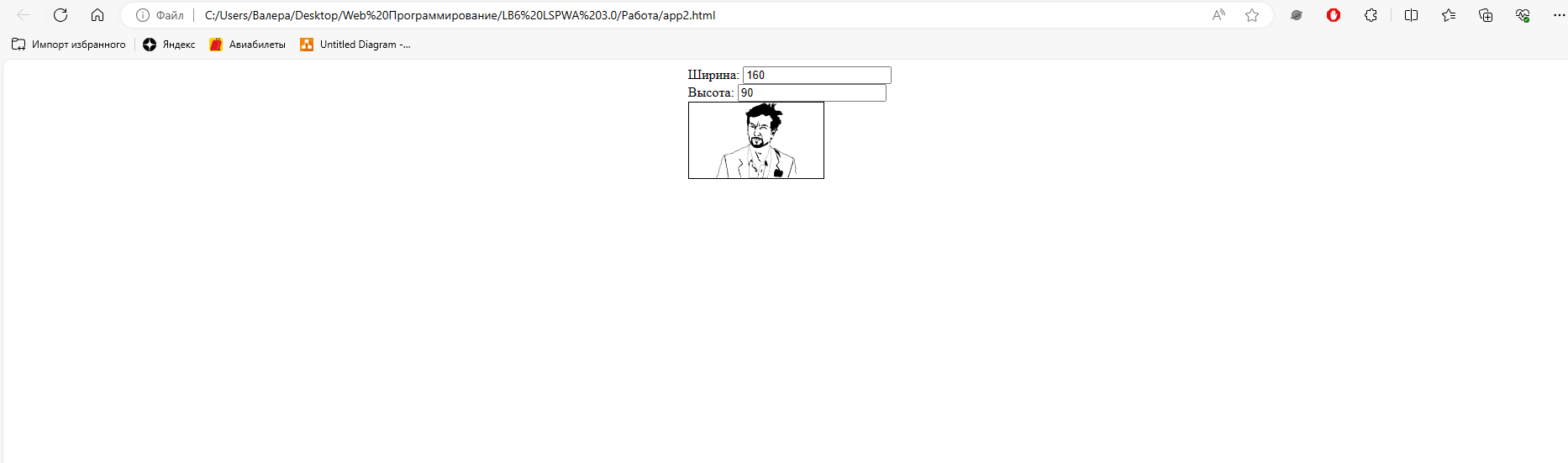


Рисунок 4. Открытие приложения.



Рисунок 5. Результат при изменении параметров размера

**4. Вывод**

В ходе выполнения данного задания были разработаны два одностраничных веб-приложения на фреймворке Vue.js. В процессе работы были использованы такие возможности Vue.js, как v-for и наблюдатели (watch).

В целом, работа над заданием позволила углубить знания и навыки работы с фреймворком Vue.js и его возможностями. Это был интересный и полезный опыт.