

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ  
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»  
(БГТУ им. В.Г. Шухова)

Кафедра программного обеспечения вычислительной техники и  
автоматизированных систем

Лабораторная работа №2

Тема: «Клиентское программирование. Vue js»

дисциплина:

Технологии WEB-программирования

Выполнила:

ст. группы ПВ-41

Бойко Валерия Евгеньевна

Проверил: Картамышев С. В.

Белгород 2021 г.

**Цель работы:** познакомиться с основами разработки на языке JavaScript. Познакомится с основами разработки frontend приложения, в частности с использованием клиентских фреймворков (Vue js). Научиться переносить вёрстку в отдельные компоненты.

### **Задание к лабораторной работе:**

1. Изучить основы разработки на языке JavaScript.
2. Изучить основы разработки frontend-приложения.
3. Развернуть базовое приложение на фреймворке Vue js.
4. Добавить необходимые компоненты и перенести в них вёрстку, сделанную в прошлой лабораторной работе.
5. Продемонстрировать работу web-приложения.

### **Содержание**

Выполнение .....	3
Код программы.....	4
Catalog.vue.....	4
v-catalog.vue .....	5
v-catalog-item.vue.....	7
v-header.vue .....	10
Результат работы программы.....	13
Вывод:.....	14

## Выполнение

**Vue CLI** — полноценная система для быстрой разработки на Vue.js

- **CLI** (`@vue/cli`) — это npm-пакет, устанавливаемый глобально и предоставляющий команду `vue` в терминале. Он позволяет быстро создать новый проект командой `vue create`, или мгновенно прототипировать ваши новые идеи через `vue serve`. Также можно управлять проектами в графическом интерфейсе через `vue ui`. Мы рассмотрим, что он может делать, в следующих разделах руководства.
- **Vuex** — паттерн управления состоянием + библиотека для приложений на Vue.js. Он служит централизованным хранилищем данных для всех компонентов приложения с правилами, гарантирующими, что состояние может быть изменено только предсказуемым образом. Vuex интегрируется с официальным расширением `vue-devtools` (opens new window), предоставляя «из коробки» такие продвинутые возможности, как «машину времени» для отладки и экспорт/импорт слепков состояния данных.

Для начала работы на Vue необходимо выполнить следующий список действий:

### 1. Создать проект через консоль

```
vue create MY_project
```

### 2. Выбрать нужную сборку

### 3. Установить зависимости проекта с различными библиотеками

```
npm i axios -save
```

- `vuex` (Единое централ. хранилище)
- `axios` (Для получения и обработки данных через API)
- `node-sass/sass-loader` (Для работы с препроцессорами `scss`)

### 4. Установить фейковый api

```
npm install -g json-server
```

### 5. Добавляем db.json

```
npx json-server db.json
```

### 6. Далее создаем компоненты и формируем иерархию проекта

### 7. Получаем url

```
json-server --watch db.json
```

### 8. Запускаем проект

```
npm run serve
```

## Код программы

### Catalog.vue

```
<template>
  <div class="Catalog">
    <VHeader/>
    <v-catalog/>
    <v-cart
      v-if="CART.length"
      v-bind:cart_data="CART"
    />
    <v-footer/>
  </div>
</template>

<script>
  import VHeader from '../components/layouts/v-header'
  import vCatalog from '../components/catalog/v-catalog.vue'
  import VCart from '../components/catalog/v-cart.vue'
  import VFooter from '../components/layouts/v-footer.vue'
  import {mapGetters} from 'vuex'

  export default {
    name: 'Catalog',
    components: { VHeader, vCatalog, VCart, VFooter },
    computed: {
      ...mapGetters([
        'CART'
      ])
    }
  }
</script>

<style>
</style>
```

## v-catalog.vue

```
<template>
  <div class="v-catalog">
    <h1 style="text-align: left">Картини различных эпох</h1>
    <div class="v-catalog__list">
      <v-catalog-item
        v-for="(picture, index) in filteredPictures"
        :key="picture.name"
        v-bind:picture_data="picture"
        @addToCart="addToCart"
        @deleteFromCart="deleteFromCart(index)"
      />
    </div>
  </div>
</template>

<script>
import vCatalogItem from "../v-catalog-item.vue";
import { mapActions, mapGetters } from "vuex";

export default {
  name: "v-catalog",
  components: { vCatalogItem },
  data() {
    return {
      sortedPictures: [],
    };
  },
  methods: {
    ...mapActions(["GET_PICTURES_FROM_API", "ADD_TO_CART", "DELETE_FROM_CART"]),
    addToCart(data) {
      this.ADD_TO_CART(data);
    },
    deleteFromCart(index) {
      this.DELETE_FROM_CART(index);
    },
    sortPicturesBySearchValue(value) {
      this.sortedPictures = [...this.PICTURES];
      if (value) {
        this.sortedPictures = this.sortedPictures.filter(function (item) {
          return item.n_picture.toLowerCase().includes(value.toLowerCase());
        });
      } else {
        this.sortedPictures = this.PICTURES;
      }
    },
  },
  watch: {
    SEARCH_VALUE() {
      this.sortPicturesBySearchValue(this.SEARCH_VALUE);
    },
  },
  mounted() {
    this.GET_PICTURES_FROM_API().then((response) => {
      if (response.data) {
        this.sortPicturesBySearchValue(this.SEARCH_VALUE);
      }
    });
  },
  computed: {

```

```
...mapGetters(["PICTURES", "SEARCH_VALUE"]),
filteredPictures() {
  if (this.sortedPictures.length) {
    return this.sortedPictures;
  } else {
    return this.PICTURES;
  }
},
},
};
</script>

<style lang="scss">
.v-catalog {
  color: #221205;
  font-family: "Palatino Linotype", "Book Antiqua", Palatino, serif;
  font-weight: 700;

  padding-top: 150px;
  padding-left: 3.5%;
  padding-right: 4.5%;

  &__list {
    display: flex;
  }
}
</style>
```

## v-catalog-item.vue

```
<div class="v-catalog-item" v-if="picture_data.available">
  
  <p class="v-catalog-item__name">«{{picture_data.n_picture}}»<br/><br/>
    {{picture_data.author}}</p>

  <p class="v-catalog-item__price" v-if="picture_data.sale==0">
    <table height="40%" width="100%">
      <tr>
        <td style="vertical-align:top">
          {{picture_data.price}} $
          <button
            :id="get_id_in()"
            v-on:click="chpok(get_id_in(),get_id_out())"
            class="v-catalog-item__add_to_cart_btn btn"
            @click="addToCart"
            >Купить
          </button>
          <button
            :id="get_id_out()"
            style="display:none"
            v-on:click="chpok(get_id_out(),get_id_in())"
            class="v-catalog-item__add_to_cart_btn btn2"
            @click="deleteFromCart"
            >В корзине
          </button>
        </td>
      </tr>
    </table>
  </p>
<div class="v-catalog-item__price" v-else>
  <table height="40%" width="100%">
    <tr>
      <td style="vertical-align:top">
        <tr><del class="v-catalog-item__dell">{{picture_data.price}}
$</del></tr>

        <tr>{{picture_data.sale}} $</tr>
      </td>
      <td style="vertical-align:bottom">
        <button
          :id="get_id_in()"
          v-on:click="chpok(get_id_in(),get_id_out())"
          class="v-catalog-item__add_to_cart_btn btn"
          @click="addToCart"
          >Купить
        </button>
        <button
          :id="get_id_out()"
          style="display:none;"
          v-on:click="chpok(get_id_out(),get_id_in())"
          class="v-catalog-item__add_to_cart_btn btn2"
          @click="deleteFromCart"
          >В корзине
        </button>
      </td>
    </tr>
  </table>
</div>
</div>
<div class="v-catalog-item" v-else>
  <div class="v-catalog-item__blur">
    <img class="v-catalog-item__image">
```

```

        v-bind:src="require('../assets/images/' + picture_data.image)"
        alt="img">
    <p class="v-catalog-item__name">
        <{{picture_data.n_picture}}><br/><br/>
        {{picture_data.author}}
    </p>
    <p class="v-catalog-item__soldout">Продана на аукционе</p>
</div>
</div>
</template>

<script>
export default {
  name: "v-catalog-item",
  props: {
    picture_data: {
      type: Object,
      default() {
        return {};
      },
    },
  },
  data() {
    return {};
  },
  methods: {
    addToCart() {
      this.$emit("addToCart", this.picture_data);
    },
    deleteFromCart() {
      this.$emit("deleteFromCart");
    },
    chpok: function (self, id) {
      var elem1 = document.getElementById(self);
      var elem2 = document.getElementById(id);

      var state = elem2.style.display;

      if (state == "") {
        elem1.style.display = "";
        elem2.style.display = "none";
      } else {
        elem1.style.display = "none";
        elem2.style.display = "";
      }
    },
    get_id_in: function () {
      return "in_" + this.picture_data.name;
    },
    get_id_out: function () {
      return "out_" + this.picture_data.name;
    },
  },
};
</script>

<style lang="scss">
.v-catalog-item {
  box-shadow: 0 0 8px 0 #e0e0e0;

  width: calc(25% - 10px);
  margin-left: 1.5%;

```



```
border-radius: 5px;

&__image {
  border-radius: 5px;
  object-fit: cover;
  width: 100%;
  height: 50%;
}

&__name,
&__price,
&__soldout {
  padding: 1px 15px 2px;
  text-align: left;
  white-space: nowrap;
}

&__dell {
  color: #555c6d;
  font-size: 13px;
  text-decoration: line-through;
}

&__blur {
  position: relative;
  z-index: -1;
  filter: opacity(50%);
}
}
</style>
```

## v-header.vue

```
<template>
  <div class="v-header">
    <table>
      <tr>
        <td>
          <router-link :to="{ name: 'mainPage' }">
            
          </router-link>
        </td>
        <td>
          <div class="pages">
            <button><a href="#/catalog">Каталог</a></button>
            <button><a href="#/delivery">Доставка</a></button>
            <button><a href="#/payment">Оплата</a></button>
            <button><a href="#/contacts">Контакты</a></button>
            <button><a href="#/about_gallery">О галерее</a></button>
          </div>
        </td>
        <td>
          <div class="search_fields">
            <input
              type="text"
              class="search_field"
              v-model="searchValue"
              placeholder="Поиск по названию картины"
            />
            <span
              class="close"
              onclick="this.previousElementSibling.value = ''"
              @click="search('') "
            >
              &times;
            </span>
            <button class="search_btn" @click="search(searchValue)">
              Найти
            </button>
          </div>
        </td>
      </tr>
    </table>
  </div>
</template>

<script>
import { mapActions, mapGetters } from "vuex";

export default {
  name: "v-header",
  data() {
    return {
      searchValue: "",
    };
  },
  computed: {
    ...mapGetters(["SEARCH_VALUE"]),
  },
  methods: {
    ...mapActions(["GET_SEARCH_VALUE_TO_VUEX"]),
    search(value) {
      this.GET_SEARCH_VALUE_TO_VUEX(value);
      this.$router.push("/catalog");
    }
  }
}
```

```
    },  
  },  
};  
</script>
```

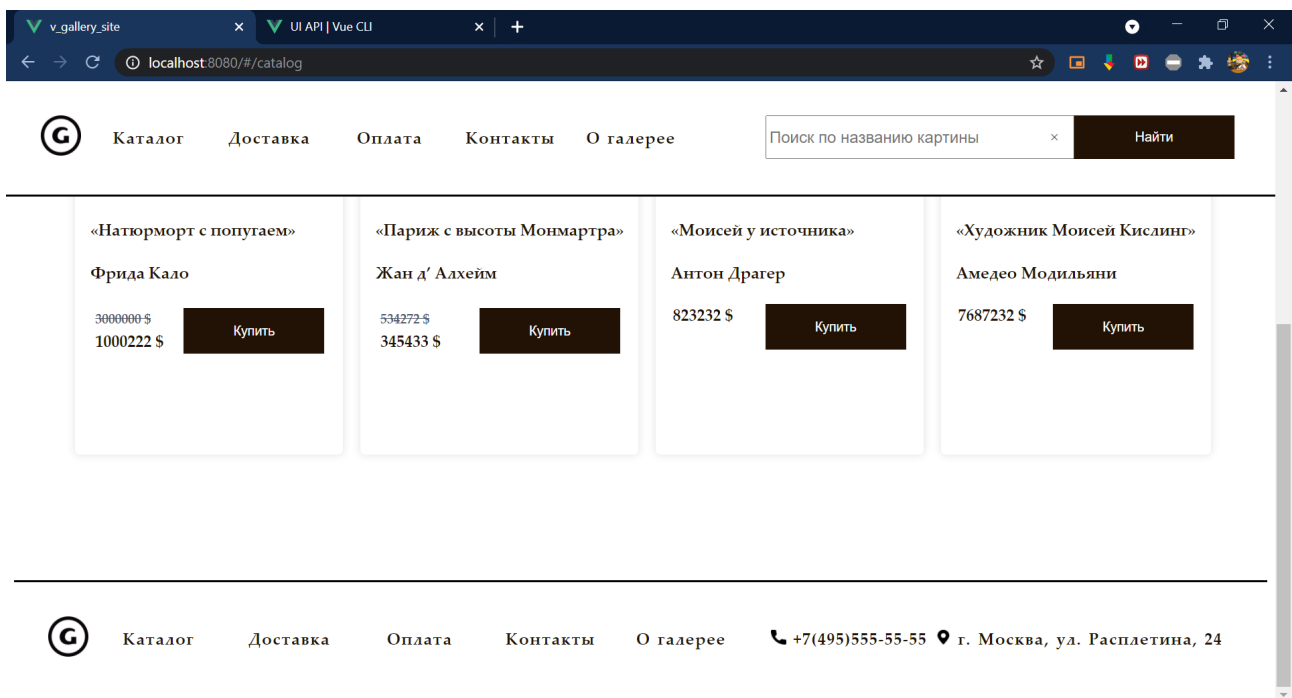
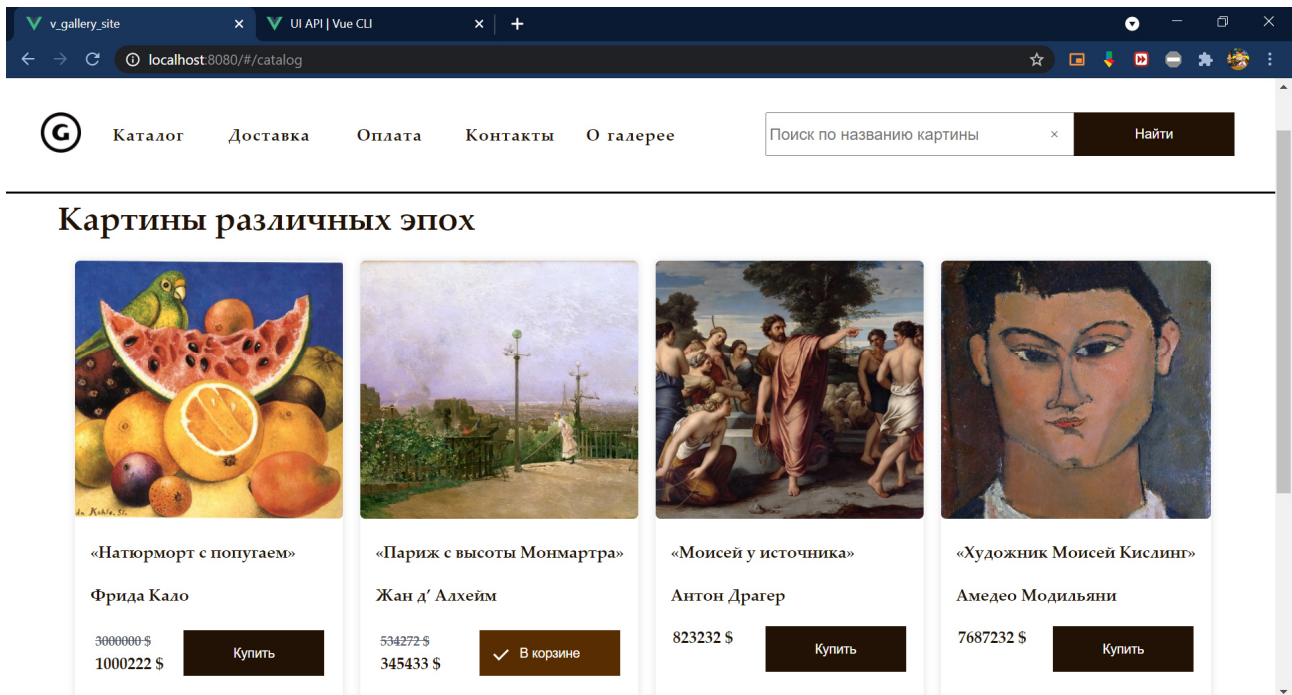
```
<style lang="scss">  
.v-header {  
  background-color: white;  
  padding: 2.5%;  
  left: 0;  
  top: 0;  
  
  border-bottom: solid;  
  position: fixed;  
  width: 100%;  
  
  table {  
    width: 97%;  
  }  
  
  img {  
    width: 40px;  
  }  
  
  .pages button {  
    width: 120px;  
    height: 40px;  
    font-size: 16px;  
    background: white;  
    border: 0;  
    outline: none;  
    cursor: pointer;  
  
    &:hover {  
      background: wheat;  
    }  
  }  
  
  .pages a {  
    color: #221205;  
    font-family: "Palatino Linotype", "Book Antiqua", Palatino, serif;  
    font-weight: 700;  
    letter-spacing: 1px;  
    word-spacing: 2px;  
    text-decoration: none;  
  }  
  
  .search_fields {  
    display: flex;  
    justify-content: center;  
    align-items: center;  
  }  
  
  .search_fields input {  
    width: 300px;  
    height: 37px;  
    font-size: 16px;  
  }  
  
  .search_fields .search_btn {  
    width: 160px;  
    height: 43px;  
    background: #221205;  
    color: #ffffff;
```

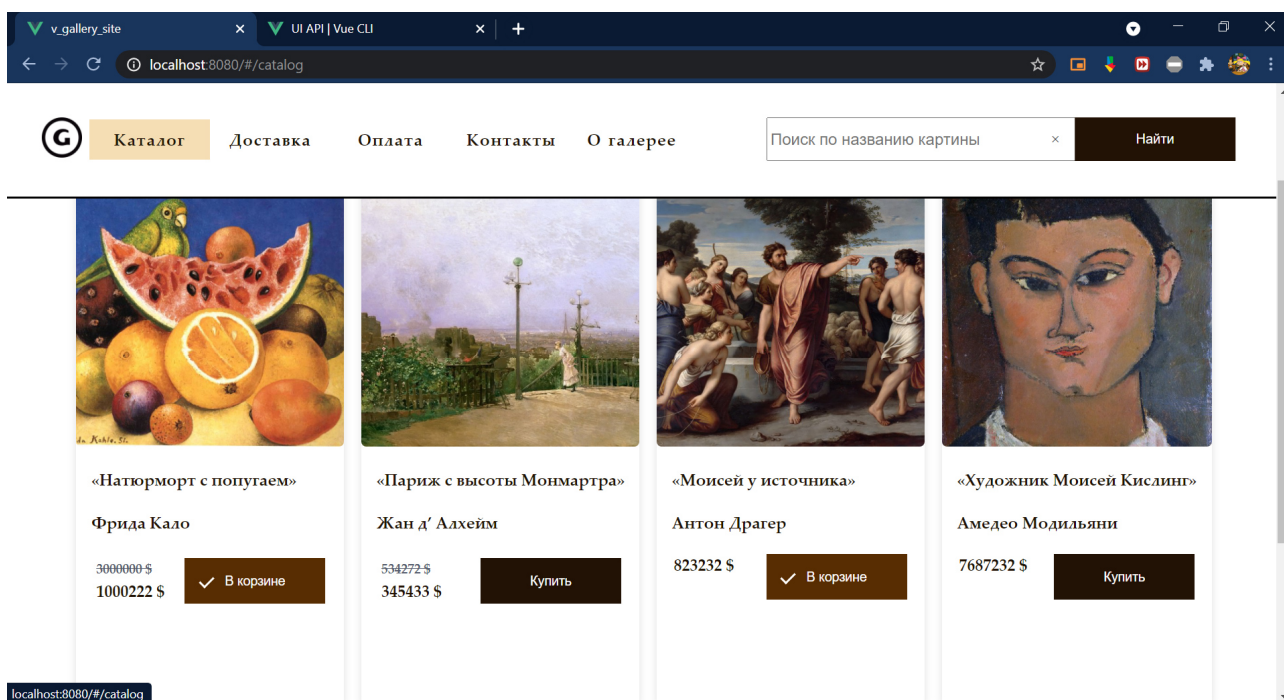
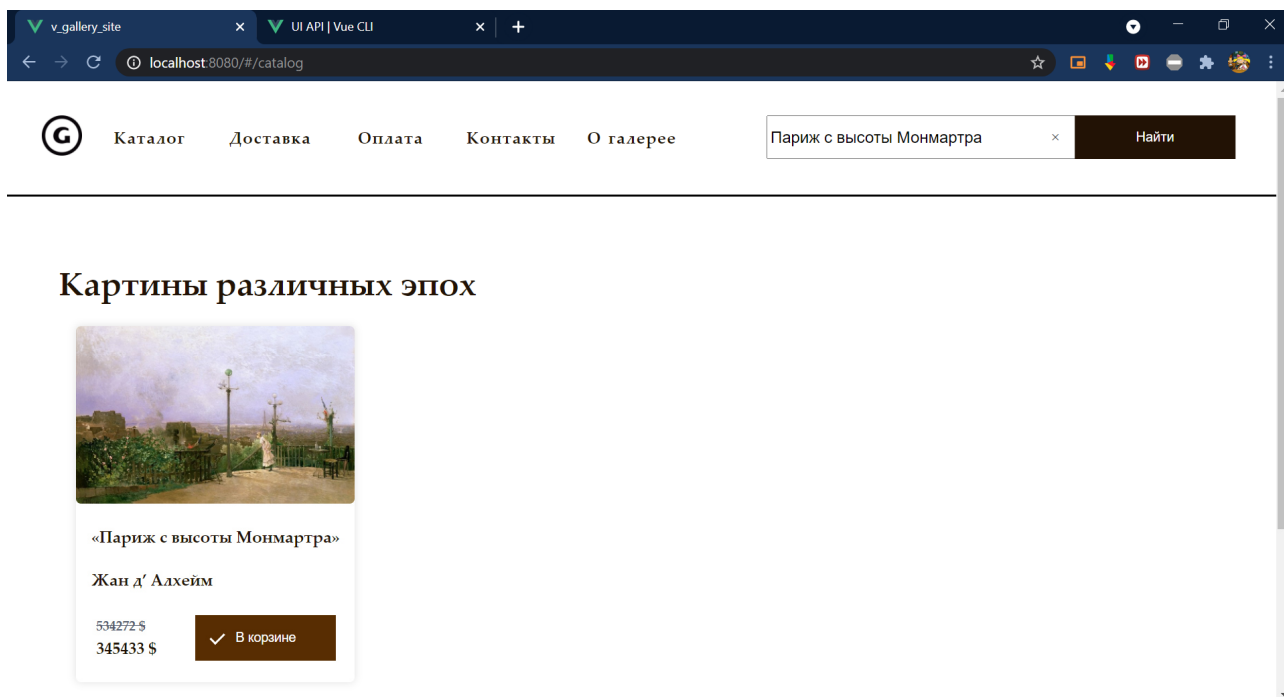
```
border: 0;
transform: translateX(-7%);
cursor: pointer;

&:hover {
  background: #592d02;
}
}

.close {
  position: relative;
  right: 25px;
  width: 10px;
  cursor: pointer;
}
}
</style>
```

# Результат работы программы





## Вывод:

В ходе выполнения данной лабораторной работы познакомились с основами разработки на языке JavaScript. Изучили основы разработки frontend приложения, в частности с использованием клиентских фреймворков (Vue js). Научились переносить вёрстку в отдельные компоненты.