

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
Федерального государственного бюджетного образовательного учреждения
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»

Факультет _____ ИТР _____

Кафедра _____ ПИН _____

ЛАБОРАТОРНАЯ РАБОТА №2

По _____ Web - технологии _____

Тема _____ Простое веб-приложение. Верстка _____

Руководитель

Колпаков А.А.

(фамилия, инициалы)

(подпись)

(дата)

Студент _____ ПИНЗ - 122 _____

(группа)

Карев А.В.

(фамилия, инициалы)

(подпись)

(дата)

Муром 2025

Лабораторная работа №2

Тема: Простое веб-приложение. Верстка.

Цель работы: знакомство с node, npm, написание простого приложения на JavaScript. В ходе выполнения работа предстоит ознакомиться с кодом реализации простого интерфейса и вывода данных, и затем выполнить задания по варианту.

Ход работы:

1. Инициализация проета.

Создан проект, который состоит из файлов package.json и .gitignore.

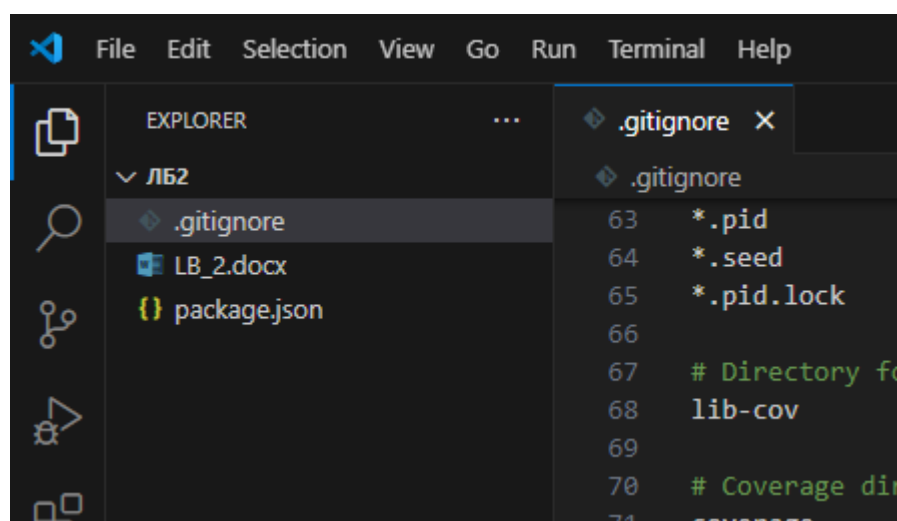


Рисунок 1 – созданные файлы

2. Создание главной страницы, подключение bootstrap.

Создание основной страницы index.html и ее открытие с помощью live сервера.

HTML код страницы index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Simple App</title>
</head>
<body>
<div>Hello world!</div>
</body>
```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 2 |

</html>

Hello world!

Рисунок 2 – Результат выполнения кода

Подключен bootstrap с помощью команды «npm I bootstrap» и добавлена кнопка из библиотеки компонентов в index.html.

Код HTML страницы:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Simple App</title>
<link rel="stylesheet"
href="node_modules/bootstrap/dist/css/bootstrap.min.css">
</head>
<body>
<button type="button" class="btn btn-primary">Hello world!</button>
<script
src="node_modules/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

Hello world!

Рисунок 3 – Результат добавления кнопки

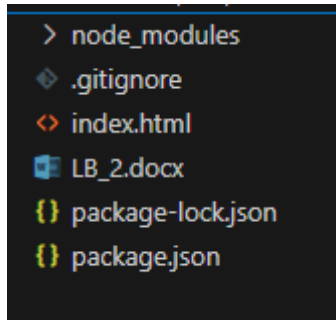


Рисунок 4 – Иерархия проекта

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 3 |

3. Простая кнопка на JavaScript

Создание аналогичной кнопки с помощью JavaScript, получив доступ к нашему HTML дереву.

HTML код:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Simple App</title>
<link rel="stylesheet"
href="node_modules/bootstrap/dist/css/bootstrap.min.css">
</head>
<body>
  <div id="root"></div>
  <script src="main.js" type="module"></script>
<script
src="node_modules/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

JavaScript код:

```
const root = document.getElementById('root');
root.insertAdjacentHTML('beforeend',
  '<button type="button" class="btn btn-primary">Hello world
2!</button>')
```

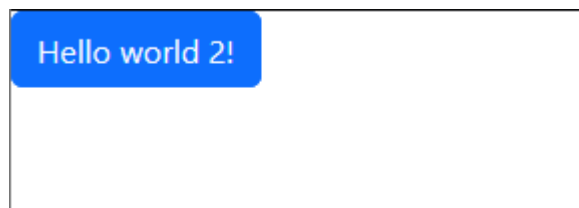


Рисунок 5 – Результат выполнения кода

4. Структурирование проекта

Создана страницы под новую архитектуру.

Код файла index.js:

```
export class MainPage {
  constructor(parent) {
    this.parent = parent;
  }
}
```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| | | | | | | 4 |
| Изм. | Лист | № докум. | Подпись | Дата | | |

```

render() {
    this.parent.insertAdjacentHTML('beforeend',
        '<button type="button" class="btn btn-primary">Hello world
3!</button>');
}
}

```

Код файла main.js:

```

import {MainPage} from "../pages/main/index.js";
const root = document.getElementById('root');
const mainPage = new MainPage(root);
mainPage.render();

```

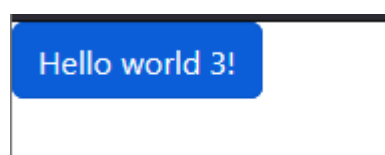


Рисунок 6 – Результат выполнения кода

Вынесем в компонент создание кнопки.

Код файла index.js с компонентом:

```

export class ButtonComponent {
    constructor(parent) {
        this.parent = parent;
    }
    render() {
        this.parent.insertAdjacentHTML('beforeend',
            '<button type="button" class="btn btn-primary">Hello world
4!</button>');
    }
}

```

Код файла index.js со страницей:

```

import {ButtonComponent} from "../../components/button/index.js";
export class MainPage {
    constructor(parent) {
        this.parent = parent;
    }
    render() {
        const button = new ButtonComponent(this.parent)
    }
}

```

```

        button.render()
    }
}

```

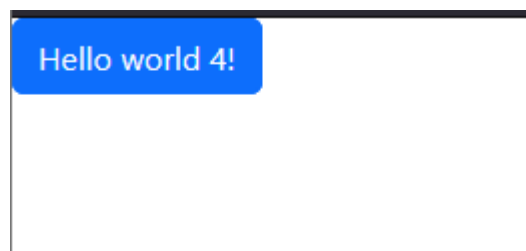


Рисунок 7 – Результат выполнения кода

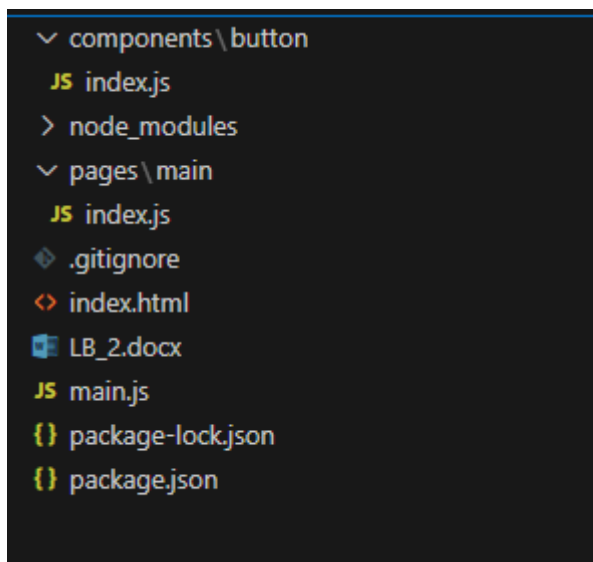


Рисунок 8 – Иерархия проекта

5. Верстка главной страницы

На главную страницу добавлен список карточек для отображения.

Код компонента карточки:

```

export class ProductCardComponent {
  constructor(parent) {
    this.parent = parent;
  }
  getHTML() {
    return (
      <div class="card" style="width: 300px;">
        

```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 6 |

```

    <div class="card-body">
    <h5 class="card-title">Акция</h5>
    <p class="card-text">Вот тут информация об акции</p>
    <button class="btn btn-primary">Нажми на меня</button>
    </div>
  </div>
  `
  )
  }

  render() {
    const html = this.getHTML()
    this.parent.insertAdjacentHTML('beforeend', html)
  }
}

```

Код отображения компонента на странице:

```

import {ProductCardComponent} from "../components/product-
card/index.js";
export class MainPage {
  constructor(parent) {
    this.parent = parent;
  }
  render() {
    const productCard = new ProductCardComponent(this.parent)
    productCard.render()
  }
}

```

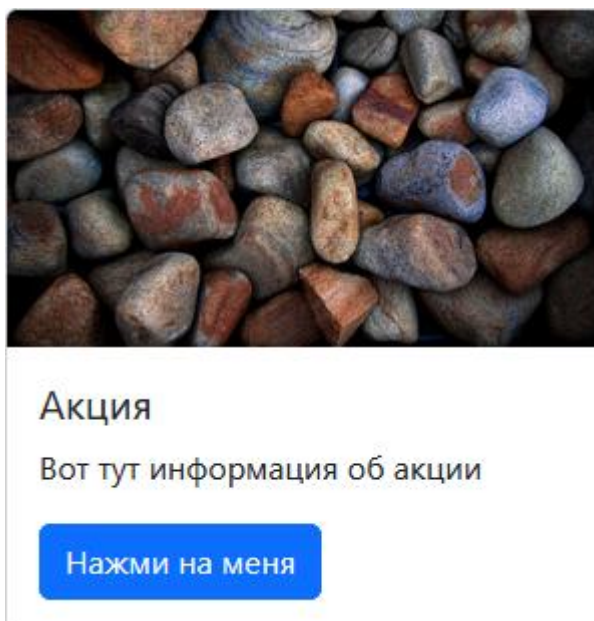


Рисунок 9 – Создание компонента

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 7 |

Данные были прокинуть в компонент и теперь функция `render` принимает данные, которые будут отрисованы. Так же добавлена логика отрисовки несколько компонентов сразу.

Код компонента:

```
export class ProductCardComponent {
  constructor(parent) {
    this.parent = parent;
  }
  getHTML(data) {
    return (
      <div class="card" style="width: 300px;">
        
        <div class="card-body">
          <h5 class="card-title">${data.title}</h5>
          <p class="card-text">${data.text}</p>
          <button class="btn btn-primary">Нажми на меня</button>
        </div>
      </div>
    )
  }
  render(data) {
    const html = this.getHTML(data)
    this.parent.insertAdjacentHTML('beforeend', html)
  }
}
```

Код главной страницы:

```
import {ProductCardComponent} from "../components/product-
card/index.js";
export class MainPage {
  constructor(parent) {
    this.parent = parent;
  }
  get pageRoot() {
    return document.getElementById('main-page')
  }
  getHTML() {
    return (
      <div class="card">
        <div class="card-body">
          <h5 class="card-title">Нажми на меня</h5>
          <p class="card-text">Нажми на меня</p>
          <button class="btn btn-primary">Нажми на меня</button>
        </div>
      </div>
    )
  }
}
```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| | | | | | | 8 |
| Изм. | Лист | № докум. | Подпись | Дата | | |


```

        <div id="main-page" class="d-flex flex-wrap"><div/>
        ,
    )
}
getData() {
    return [
        {
            id: 1,
            src:

"https://i.pinimg.com/originals/c9/ea/65/c9ea654eb3a7398b1f702c758c1c4206.jp
g",
            title: "Акция",
            text: "Такой акции вы еще не видели 1"
        },
        {
            id: 2,
            src:

"https://i.pinimg.com/originals/c9/ea/65/c9ea654eb3a7398b1f702c758c1c4206.jp
g",
            title: "Акция",
            text: "Такой акции вы еще не видели 2"
        },
        {
            id: 3,
            src:

"https://i.pinimg.com/originals/c9/ea/65/c9ea654eb3a7398b1f702c758c1c4206.jp
g",
            title: "Акция",
            text: "Такой акции вы еще не видели 3"
        }
    ]
}
render() {
    this.parent.innerHTML = ''
    const html = this.getHTML()
    this.parent.insertAdjacentHTML('beforeend', html)
    const data = this.getData()
    data.forEach((item) => {
        const productCard = new ProductCardComponent(this.pageRoot)
        productCard.render(item)
    })
}

```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| | | | | | | 9 |
| Изм. | Лист | № докум. | Подпись | Дата | | |

```

    })
  }
}

```

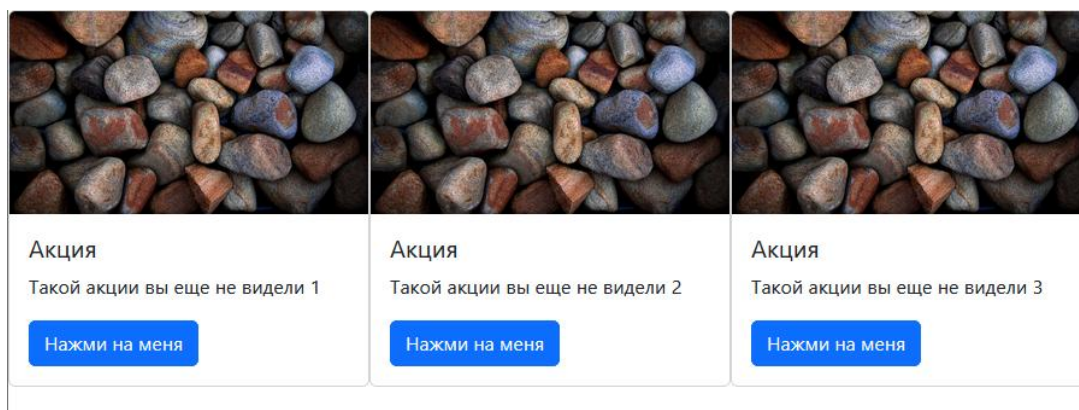


Рисунок 10 – Добавление компонентов

Добавлен обработчик нажатия на кнопку:

Добавленная функция для обработки кнопки в коде компонента:

```

addListeners(data, listener) {
  document
    .getElementById(`click-card-${data.id}`)
    .addEventListener("click", listener)
}

render(data, listener) {
  const html = this.getHTML(data)
  this.parent.insertAdjacentHTML('beforeend', html)
  this.addListeners(data, listener)
}

```

Измененный код для обработки кнопки на главной странице:

```

clickCard(e) {
  const cardId = e.target.dataset.id
}

render() {
  this.parent.innerHTML = ''
  const html = this.getHTML()
  this.parent.insertAdjacentHTML('beforeend', html)
  const data = this.getData()
  data.forEach((item) => {
    const productCard = new ProductCardComponent(this.pageRoot)
    productCard.render(item, this.clickCard.bind(this))
  })
}

```

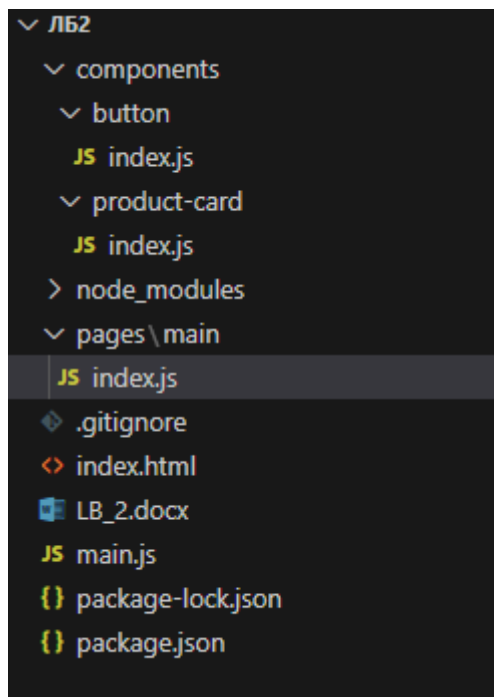


Рисунок 11 – Иерархия проекта

6. Верстка страницы продукта

Добавлена страница продукта. Код страницы:

```
import {ProductComponent} from "../../components/product/index.js";
export class ProductPage {
    constructor(parent, id) {
        this.parent = parent
        this.id = id
    }
    getData() {
        return {
            id: 1,
            src:
                "https://i.pinimg.com/originals/c9/ea/65/c9ea654eb3a7398b1f702c758c1c4206.j
                pg",
            title: `Акция ${this.id}`,
            text: "Такой акции вы еще не видели"
        }
    }
    get pageRoot() {
        return document.getElementById('product-page')
    }
    getHTML() {
        return (
            `
            <div class="product-page">
                <div class="product-card">
                    
                    <h3 class="product-title">${this.title}</h3>
                    <p class="product-text">${this.text}</p>
                </div>
            </div>
            `
        )
    }
}
```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 11 |

```

<div id="product-page"></div>
,
)
}
render() {
  this.parent.innerHTML = ''
  const html = this.getHTML()
  this.parent.insertAdjacentHTML('beforeend', html)
  const data = this.getData()
  const product = new ProductComponent(this.pageRoot)
  product.render(data)
}
}

```

Создан компонент для отображения на странице продукта и добавлена отрисовка этого компонента на странице продукта:

```

export class ProductComponent {
  constructor(parent) {
    this.parent = parent
  }
  getHTML(data) {
    return (
      ,
      <div class="card mb-3" style="width: 540px;">
        <div class="row g-0">
          <div class="col-md-4">
            
          </div>
          <div class="col-md-8">
            <div class="card-body">
              <h5 class="card-title">${data.title}</h5>
              <p class="card-text">${data.text}</p>
            </div>
          </div>
        </div>
      </div>
      ,
    )
  }
  render(data) {
    const html = this.getHTML(data)
    this.parent.insertAdjacentHTML('beforeend', html)
  }
}

```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| | | | | | | 12 |
| Изм. | Лист | № докум. | Подпись | Дата | | |

```

}
}

```

Добавлен обработчик кнопки. При нажатии на кнопку открывается страница продукта с подгруженным компонентом:

```

clickCard(e) {
    const cardId = e.target.dataset.id
    const productPage = new ProductPage(this.parent, cardId)
    productPage.render()
}

```

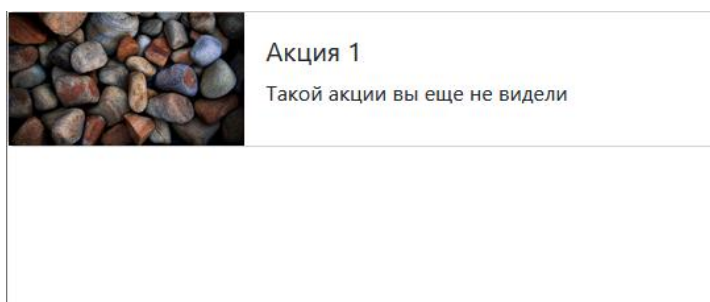


Рисунок 12 – Результат добавления карточки товара

Для удобства на страницу с продуктом добавлен компонент с кнопкой, возвращающий на главную страницу:

Код компонента:

```

export class BackButtonComponent {
    constructor(parent) {
        this.parent = parent;
    }
    addListeners(listener) {
        document
            .getElementById("back-button")
            .addEventListener("click", listener)
    }
    getHTML() {
        return (
            `
            <button id="back-button" class="btn btn-primary"
            type="button">Назад</button>
            `
        )
    }
}

```

```

render(listener) {
  const html = this.getHTML()
  this.parent.insertAdjacentHTML('beforeend', html)
  this.addListeners(listener)
}
}

```

Код страницы с товаром с добавленным компонентом:

```

import {BackButtonComponent} from "../../components/back-button/index.js";
import {MainPage} from "../main/index.js";
clickBack() {
  const mainPage = new MainPage(this.parent)
  mainPage.render()
}
render() {
  this.parent.innerHTML = ''
  const html = this.getHTML()
  this.parent.insertAdjacentHTML('beforeend', html)
  const backButton = new BackButtonComponent(this.pageRoot)
  backButton.render(this.clickBack.bind(this))
  const data = this.getData()
  const product = new ProductComponent(this.pageRoot)
  product.render(data)
}

```

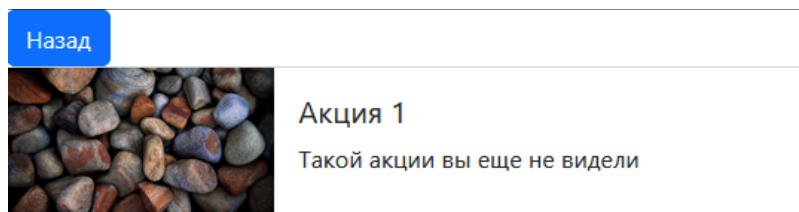


Рисунок 13 – Добавленная кнопка

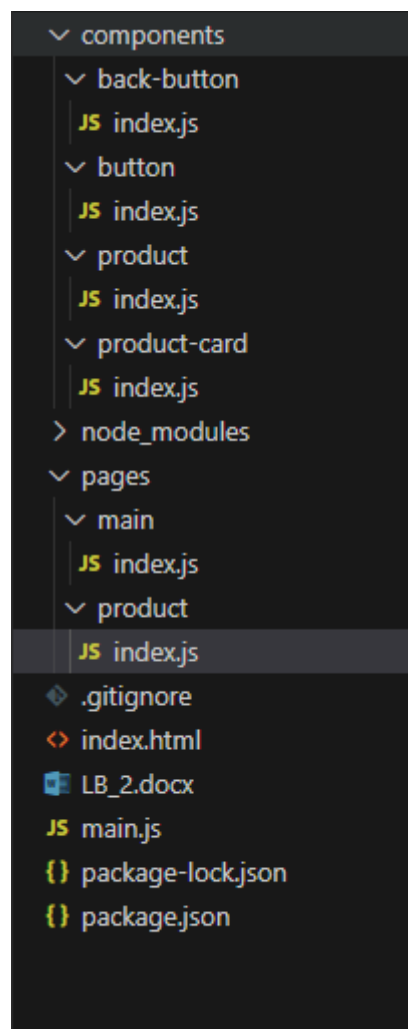


Рисунок 14 – Иерархия проекта

7. Задание

Создать двухстраничное приложение из примера по вариантам. Вариант состоит из темы и компонента, который необходимо использовать. Все данные должны соответствовать вашей теме. Компонент можно применить по своему усмотрению. Вариант: Тема - кошки, Компонент - уведомления.

JavaScript код главной страницы:

```
import {ProductCardComponent} from "../components/product-card/index.js";
import {ProductPage} from "../product/index.js";
export class MainPage {
  constructor(parent) {
    this.parent = parent;
  }
  get pageRoot() {
    return document.getElementById('main-page')
```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 15 |

```

    }
    getHTML() {
      return (
        <div id="main-page" class="d-flex flex-wrap"><div/>
      )
    }
    clickCard(e) {
      const cardId = e.target.dataset.id
      const productPage = new ProductPage(this.parent, cardId)
      productPage.render()
    }
    getData() {
      return [
        {
          id: 1,
          src:
            "https://static.wikia.nocookie.net/fictionrulezforever/images/f/f6/Profile_
            _Siamese_Cat.jpg",
          title: "Сиамская кошка",
          text: "Происхождение: Таиланд"
        },
        {
          id: 2,
          src:
            "https://w-dog.ru/wallpapers/5/17/293758380766135/kot-mejn-kun-
            ryzhij-pushistyj.jpg",
          title: "Мейн-кун",
          text: "Происхождение: США"
        },
        {
          id: 3,
          src:
            "https://santreyd.ru/upload/iblock/2ba/2bac2b4b388586bddff5b63e319732f8.jpg
            .b.jpg",
          title: "Персидская кошка",
          text: "Происхождение: Иран"
        }
      ]
    }

```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 16 |


```

render() {
    this.parent.innerHTML = ''
    const html = this.getHTML()
    this.parent.insertAdjacentHTML('beforeend', html)
    const data = this.getData()
    data.forEach((item) => {
        const productCard = new ProductCardComponent(this.pageRoot)
        productCard.render(item, this.clickCard.bind(this))
    })
}
}

```

Код страницы с элементами карточек:

```

import {ProductComponent} from "../../components/product/index.js";
import {BackButtonComponent} from "../../components/back-button/index.js";
import {MainPage} from "../main/index.js";
import { ButtonLiveAllertComponent } from "../../components/button_live_alert/index.js";
import { LiveButtonComponent } from "../../components/alert/index.js";
export class ProductPage {
    constructor(parent, id) {
        this.parent = parent
        this.id = id
    }
    clickBack() {
        const mainPage = new MainPage(this.parent)
        mainPage.render()
    }
    clickLive(){
        const LiveB = new LiveButtonComponent(this.parent)
        LiveB.render()
    }
    getData() {
        if (this.id == 1){
            return {
                id: 1,
                src:

"https://static.wikia.nocookie.net/fictionrulezforever/images/f/f6/Profile
_-_Siamese_Cat.jpg",
                title: `Кошка ${this.id}`,
                text: "Сиамская кошка"
            }
        }
    }
}

```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 17 |

```

    }
    else if (this.id == 2){
        return {
            id: 2,
            src:
                "https://w-dog.ru/wallpapers/5/17/293758380766135/kot-mejn-
kun-ryzhij-pushistyj.jpg",
            title: `Кошка ${this.id}`,
            text: "Мейн-кун"
        }
    }
    else {
        return {
            id: 3,
            src:

"https://santreyd.ru/upload/iblock/2ba/2bac2b4b388586bddff5b63e319732f8.jp
g.b.jpg",

            title: `Кошка ${this.id}`,
            text: "Персидская кошка"
        }
    }
}

get pageRoot() {
return document.getElementById('product-page')
}

getHTML() {
return (
`
<div id="product-page"></div>
`
)
}

render() {
this.parent.innerHTML = ''
const html = this.getHTML()
this.parent.insertAdjacentHTML('beforeend', html)

const backButton = new BackButtonComponent(this.pageRoot)
backButton.render(this.clickBack.bind(this))

const LiveButton = new ButtonLiveAllertComponent(this.pageRoot)
LiveButton.render(this.clickLive.bind(this))

```

| | | | | | | |
|------|------|----------|---------|------|-----------------------|------|
| | | | | | МИВУ 09.03.04 – 2.009 | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 18 |

```

        const data = this.getData()
        const product = new ProductComponent(this.pageRoot)
        product.render(data)
    }
}

```

Код компонента уведомления:

```

export class LiveButtonComponent {
    constructor(parent) {
        this.parent = parent;
    }
    getHTML() {
        return (
            `<div      class="alert      alert-success      alert-dismissible"
role="alert">
                <div>Бронь успешна выполнена!</div>
                <button      type="button"      class="btn-close"      data-bs-
dismiss="alert" aria-label="Закрыть"></button>
            </div>
            `
        )
    }
    render() {
        const html = this.getHTML()
        this.parent.insertAdjacentHTML('beforeend', html)
    }
}

```

Код компонента дополнительной кнопки:

```

export class ButtonLiveAllertComponent {
    constructor(parent) {
        this.parent = parent;
    }
    addListeners(listener) {
        document
            .getElementById("liveAlertBtn")
            .addEventListener("click", listener)
    }
    getHTML(data) {
        return (
            `
            <button id="liveAlertBtn" class="btn btn-primary"

```

```

        type="button">Забронировать</button>
        ,
    )
}
render(listener) {
    const html = this.getHTML()
    this.parent.insertAdjacentHTML('beforeend', html)
    this.addListeners(listener)
}
}

```

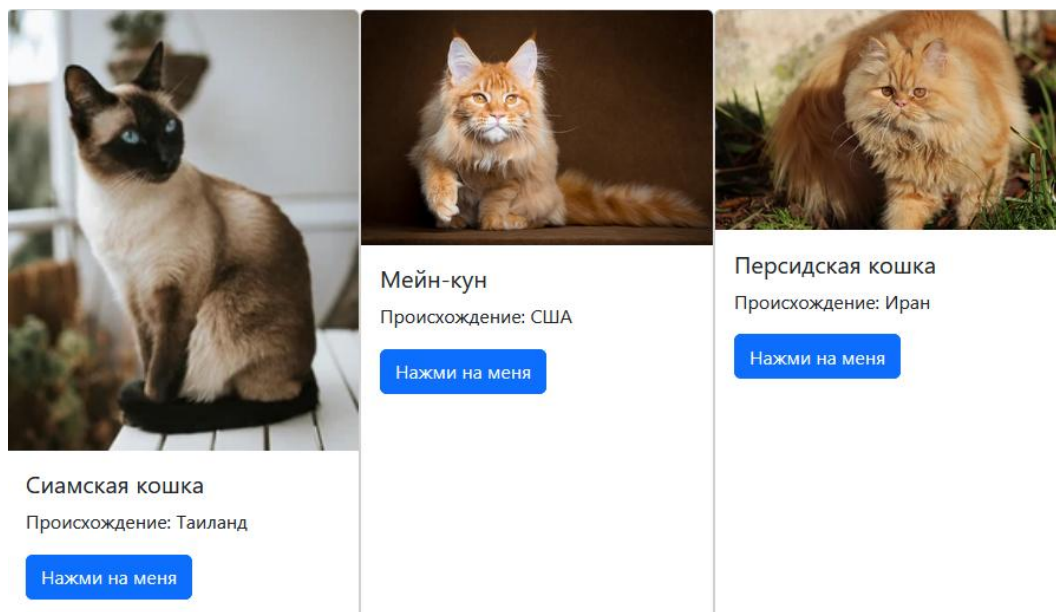


Рисунок 15 – Созданная главная страница

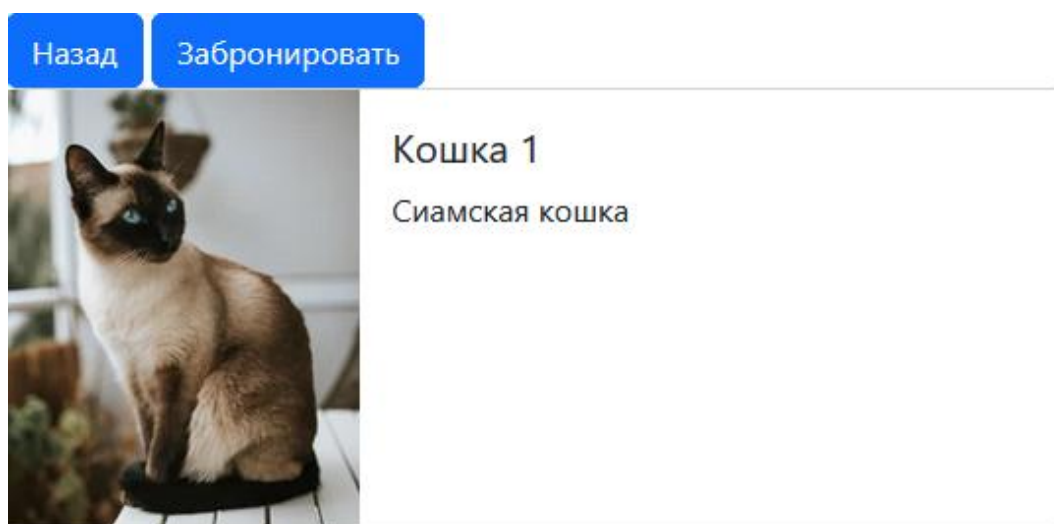


Рисунок 16 – Страница с карточкой

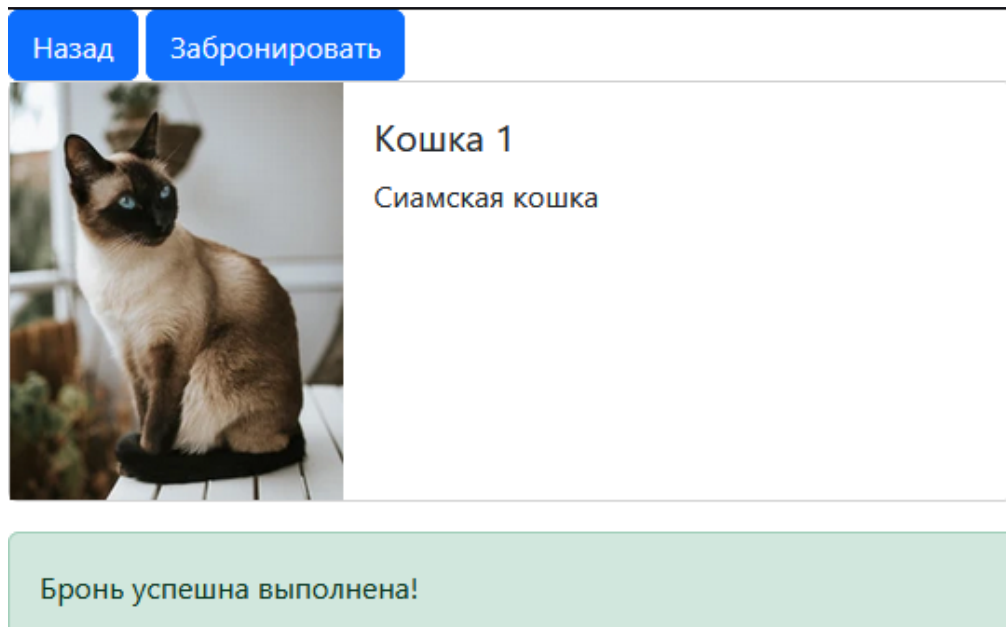


Рисунок 17 – Вывод уведомления

Вывод: ознакомились с node, npm, написано простое приложение на JavaScript. В ходе выполнения работы написан код простого интерфейса и вывода данных, и выполнено задание по варианту.