Лекция 8
Внедрение React Router в приложение React

Узнайте, как создать целое React-приложение с нуля, с возможностью отображения контента, адаптирующегося в зависимости от URL-адреса, с помощью функциональности React Router.

1. Обзор многостраничного веб-сайта

В этом разделе мы отправимся в практическое путешествие по созданию корпоративного веб-сайта с использованием React Router. Этот проект даст вам практический опыт реализации маршрутизации в приложении React. Веб-сайт будет состоять из трех страниц: «Главная», «О нас» и «Контакты». Каждая страница будет иметь свой собственный URL, и мы рассмотрим, как настроить для них маршруты.

Прежде чем углубиться в детали, давайте рассмотрим проект в общих чертах:

Цель проекта : Мы стремимся создать простой корпоративный веб-сайт с тремя отдельными страницами, используя React Router для навигации;

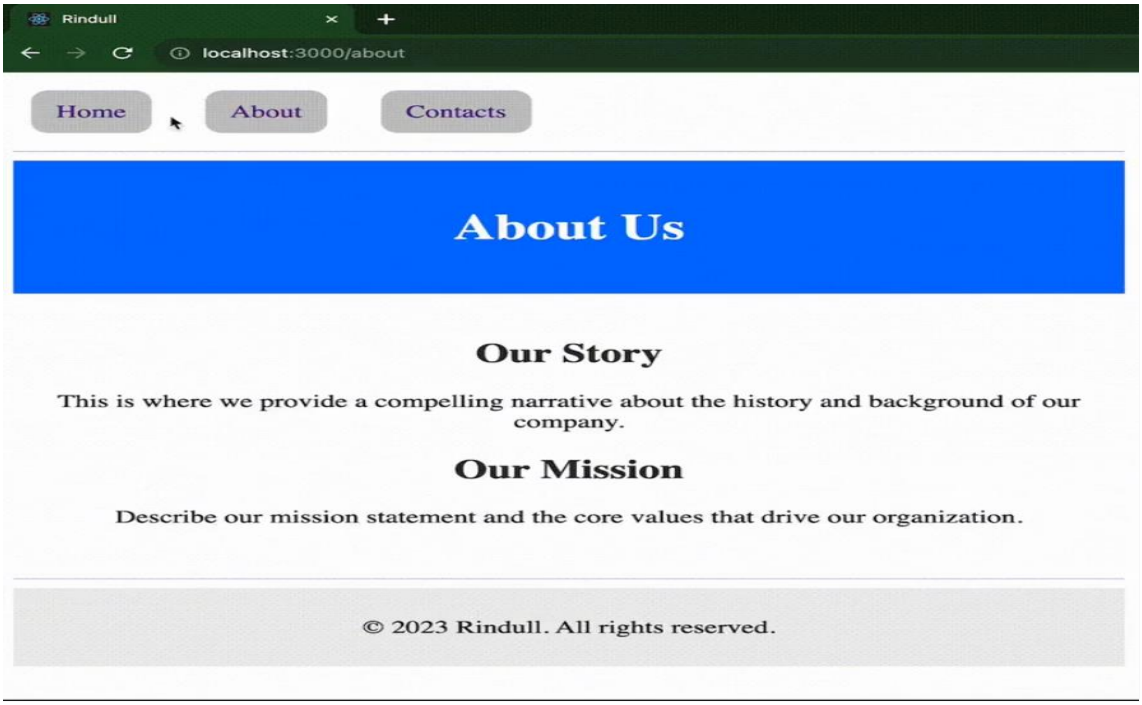
Структура страниц : Веб-сайт будет содержать следующие страницы:

- «Дом» - с указанием пути: /;
- "О программе" - с указанием пути: /about;
- «Контакты» - с указанием пути: /contacts.

Компоненты React: Все компоненты и стили React для этого проекта предопределены, что позволяет нам сосредоточиться исключительно на функциональности, предоставляемой React Router;

Песочница кода: Вам будет предоставлен доступ к песочнице кода, содержащей полный код приложения. Мы рекомендуем вам внимательно изучить код, чтобы понять основные концепции маршрутизации (рисунок 1).

На представленной ниже иллюстрации показана окончательная функциональность приложения.



Вот исходный код в виде CodeSandbox:
https://codesandbox.io/p/sandbox/187d4e57-8b06-4d95-941b-2e2e20835fcc-mvsy5f?file=%2Fsrc%2Findex.js&from-embed

2. Установка и настройка среды разработки React

Шаг 1: Установка Node.js и npm

Необходимо установить Node.js и npm на локальный компьютер.

Node.js — это среда выполнения с открытым исходным кодом, позволяющая запускать код JavaScript на стороне сервера.

npm расшифровывается как «Node Package Manager» (менеджер пакетов Node). Это менеджер пакетов для Node.js, который позволяет разработчикам легко устанавливать, управлять и обмениваться повторно используемыми пакетами или модулями кода JavaScript.

– Перейдите на официальный сайт Node.js: https://nodejs.org;

– Скачайте установщик для вашей операционной системы (Windows, macOS или Linux). Нас интересует версия LTS (Long Term Support), так как она отличается стабильностью и надежностью;

– Запустите установщик и следуйте инструкциям по установке;

– После завершения установки Node.js и npm (Node Package Manager) будут установлены на вашем компьютере.

Как проверить, установлены ли Node.js и npm?

Чтобы проверить, установлены ли на компьютере Node.js и npm, можно открыть терминал (командную строку или эмулятор терминала) и выполнить соответствующие команды. Порядок действий:

1. Откройте терминал;

2. Проверьте версию Node.js;

Выполните в терминале следующую команду.

```
Node - v
```

Эта команда выведет версию Node.js, установленную на компьютере. Если Node.js установлен, мы увидим номер версии.

3. Проверьте версию version.

Выполните следующую команду в терминале.

```
npm -v
```

Если обе команды возвращают номера версий, значит, Node.js и npm успешно установлены и доступны в системе. Если команды не распознаются или приводят к ошибке, это означает, что Node.js и/или npm не установлены либо пути к ним настроены неверно. В таком случае может потребоваться повторно выполнить шаги по установке Node.js и убедиться, что процесс прошел успешно. Иногда также рекомендуется перезагрузить компьютер.

4. Установка Node.js и npm

1) Скачайте Node.js. Откройте официальный сайт: https://nodejs.org/en/download. Выберите версию LTS (не Current).

2) Установите Node.js. После скачивания запустите файл .msi.

Вместе с Node.js автоматически устанавливается и **npm**, отдельно npm устанавливать не нужно.

3) Перезапустите терминал. Если у вас открыт VS Code или Command Prompt, **закройте его и откройте заново**. Затем выполните команду

```
node -v
```

5. Выбор редактора кода для разработки на React

5.1 Вы можете выбрать любой редактор, с которым вам удобно работать.

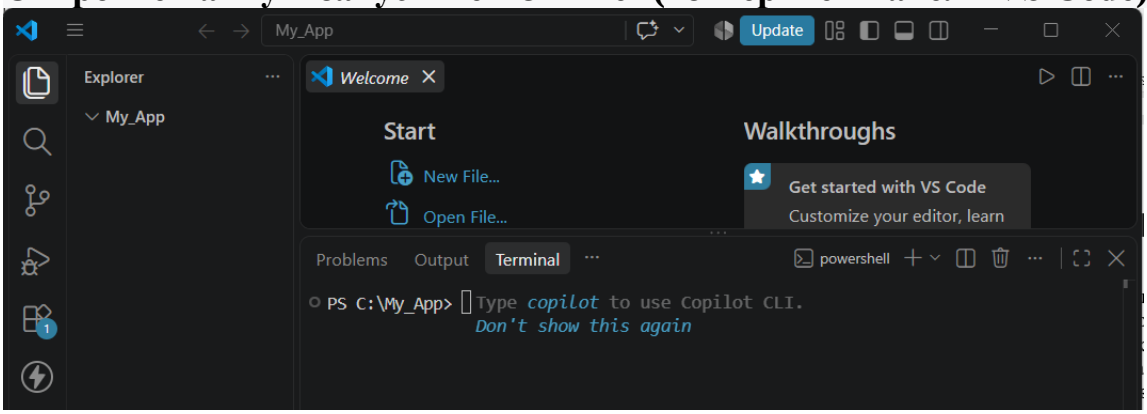
- Visual Studio Code - https://code.visualstudio.com/;

На протяжении всего курса все примеры будут демонстрироваться с использованием Visual Studio Code. Важно отметить, что выбор редактора не оказывает существенного влияния на содержание материала, однако внешний вид интерфейса может отличаться.

Порядок действий:

1. Выберите редактор кода;
2. Перейдите на сайт выбранного редактора и скачайте установщик для вашей операционной системы;
3. Запустите установщик и следуйте инструкциям по установке.

Откройте папку и запустите Terminal (из верхней панели VS Code)

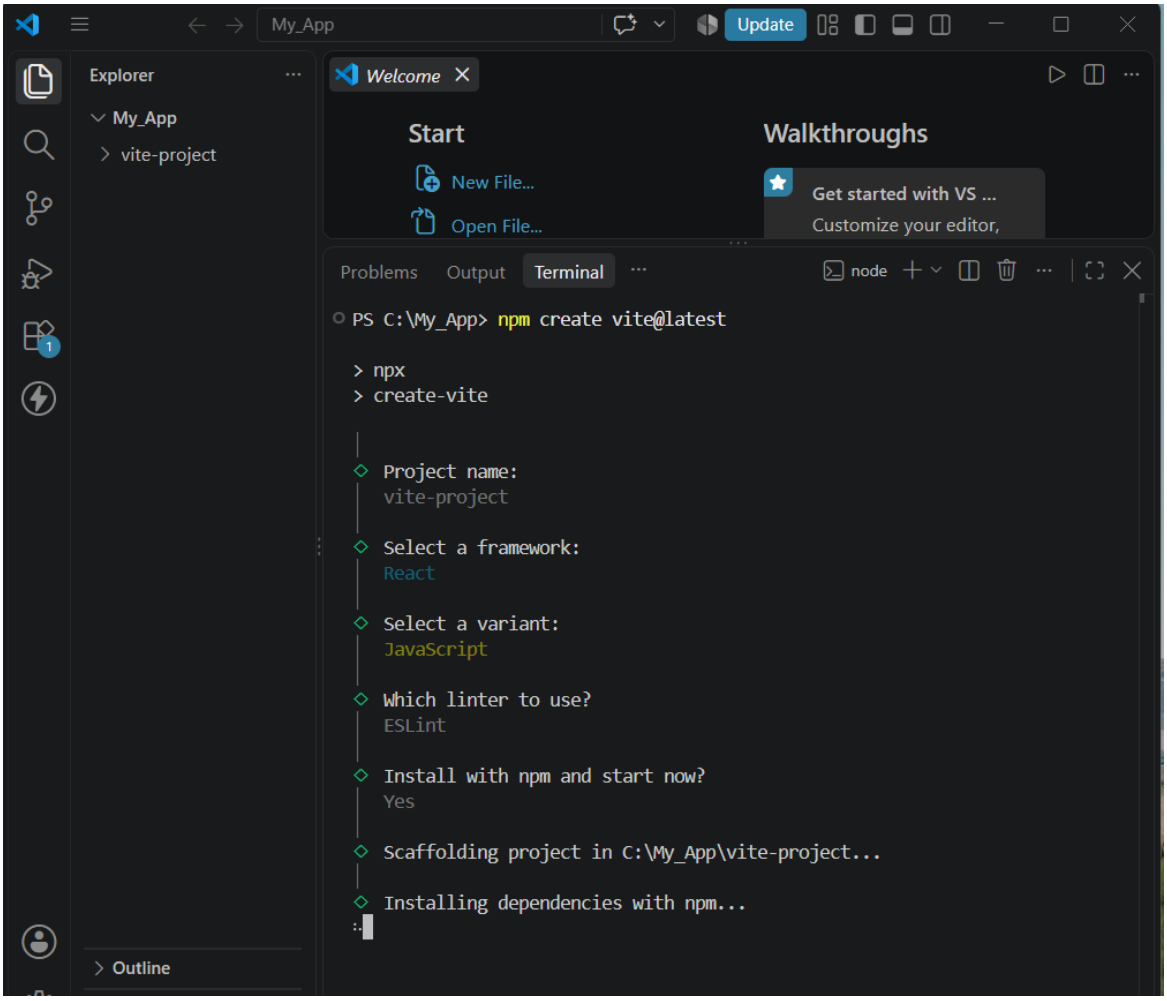


5.2 Запуск проекта на React с использованием Vite

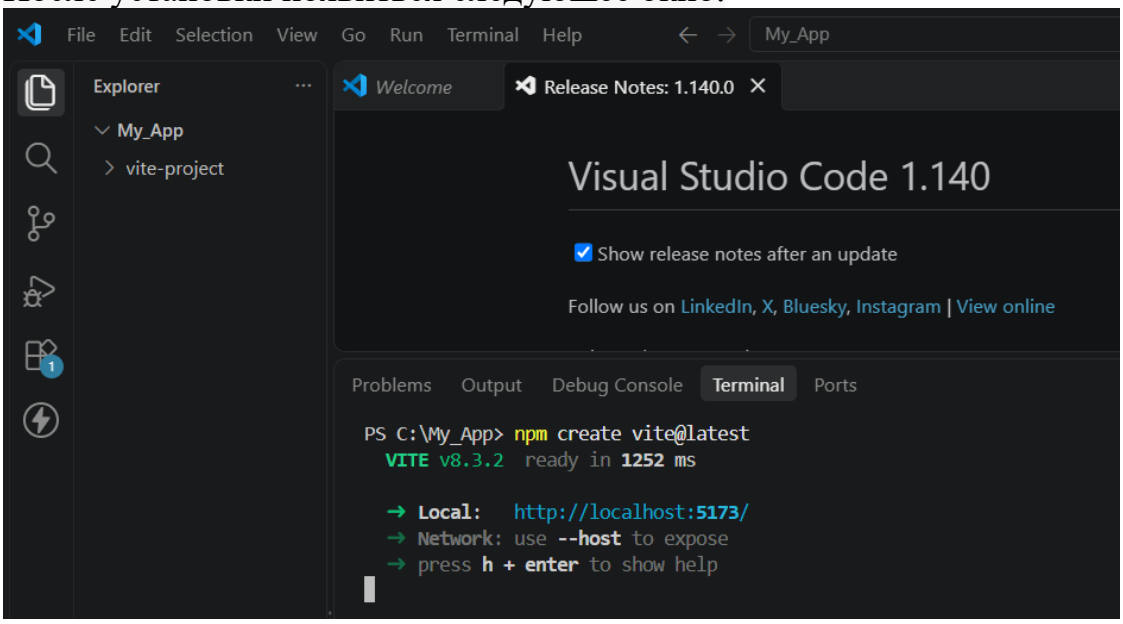
Инициализация проекта. Для настройки проекта воспользуйтесь терминалом. Откройте терминал в редакторе кода (если он еще не открыт), выполните следующую команду и нажмите Enter:

```
npm create vite@latest
```

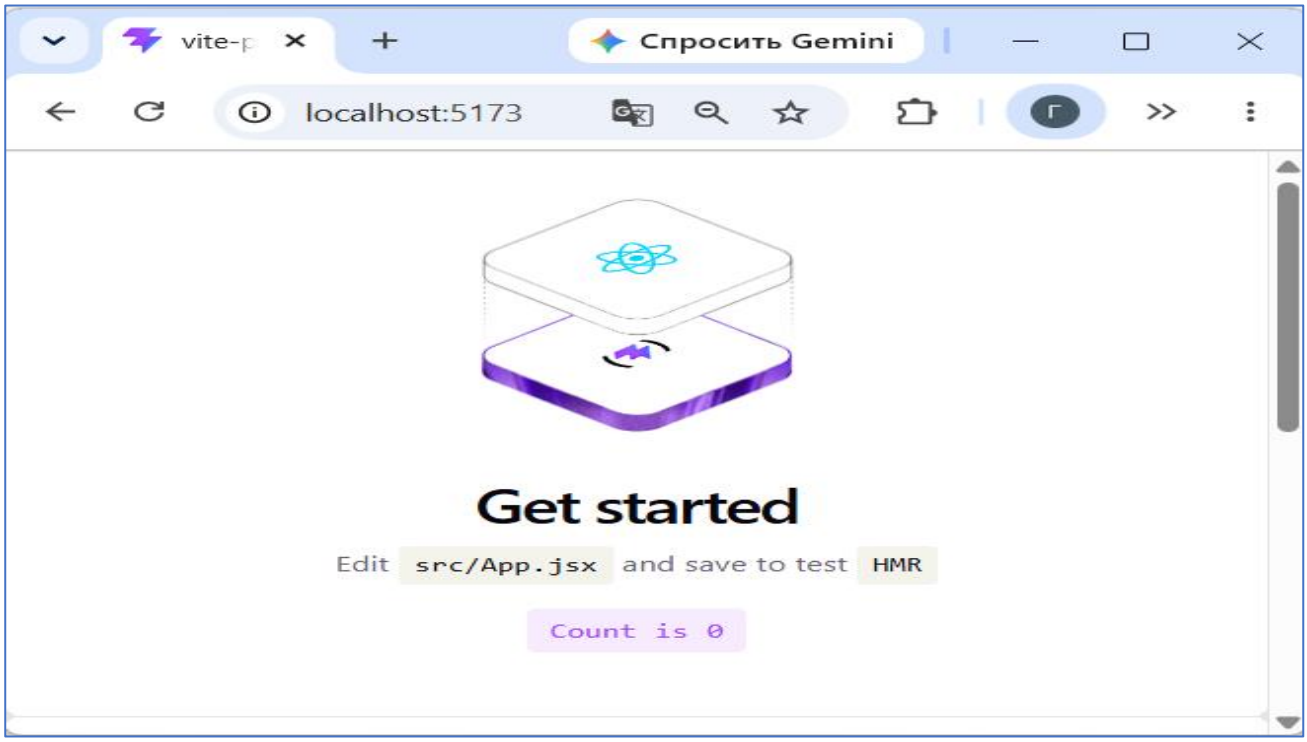
После завершения настройки Vite предложит установить зависимости. установить команду nmp.



После установки появиться следуюшее окно.



После выполнения этих команд вы увидите локальный URL для разработки — как правило: http://localhost:5173/. Вы можете открыть эту ссылку в браузере, чтобы увидеть работающее приложение.



3. Установка React Router

Установка. Чтобы начать использовать библиотеку React Router, сначала необходимо установить её в проект с помощью менеджера пакетов, такого как npm или yarn. Откройте терминал в директории проекта и выполните следующую команду (при использовании npm):

```
npm install react-router-dom
```

Проверка установки. Выполните команду в терминале, находясь в папке проекта:

```
npm list react-router-dom
```

Если установлен, вы увидите дерево с версией.

Проверка папки **node_modules.** Перейдите в папку node_modules/ внутри вашего проекта и проверьте наличие директорий:

- react-router-dom
- react-router

4. Включение маршрутизации в приложении React

После установки необходимых зависимостей мы можем приступить к интеграции библиотеки в наше приложение.

Для интеграции функциональности маршрутизации в наше приложение необходимо обернуть корневой компонент B**rowserRouter** библиотекой **react-router-dom.** Этот шаг крайне важен, поскольку он инициализирует систему маршрутизации и позволяет управлять навигацией на основе URL-адреса.

Пример.

- Откройте **index.js файл** в вашем проекте;

- Импортируйте **BrowserRouter** компонент из react-router-domбиблиотеки:
- Оберните **App** компонент в другой **BrowserRouter** компонент, как показано ниже

```
import { BrowserRouter } from "react-router-dom";
```

Оберните App компонент в другой **BrowserRouter** компонент, как показано ниже.

index.js файл:

```
import React from "react";
import { BrowserRouter } from "react-router-dom";
import ReactDOM from "react-dom/client";
import App from "./App/App";

const root = ReactDOM.createRoot(document.getElementById("root"));
root.render(
  <React.StrictMode>
    <BrowserRouter>
      <App />
    </BrowserRouter>
  </React.StrictMode>
);
```

Обернув **App** компонент в тег `<div>` **BrowserRouter**, мы делаем функциональность маршрутизации доступной в нашем приложении.