МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Информационные системы»

Дисциплина «Интернет-программирование»

**Лабораторная работа № 2**

Формирование макета (структуры) страниц сайта

Выполнил:

студент гр. ПИбд-23

Пазушкин И.П.

Проверил:

преподаватель

Истюков Т.А.

Ульяновск,   
2025 г.

**Задание лабораторной работы**

1. Разработать общий макет для страниц сайта. Макет должен обязательно включать следующие структурные элементы: шапка (header), блок навигации (navbar), содержимое (content), подвал (footer).

2. Шапка сайта должна содержать логотип и название сайта, а также включать блок навигации (navbar).

3. Блок навигации должен содержать выпадающее меню (как в примере).

4. Подвал сайта должен содержать контактные сведения, ссылки на социальные сети, время работы, адреса и т. д. в зависимости от варианта задания. Также в подвал необходимо добавить ссылки на соц. сети в виде значков (png, svg).

5. Встроить результаты ЛР № 1 в полученный макет. Также следует переделать структуру содержимого отдельных страниц для соответствия реальным сайтам.

6. При разработке макета следует использовать возможности flexbox.

7. Макет должен поддерживать адаптивную верстку для мобильных устройств, планшетов и ПК. Макеты для разных типов устройств должны отличаться.

8. Некоторые страницы могут иметь свои особенности макета.

9. Реализовать навигацию между страницами сайта.

10. Файлы проекта и отчет должны быть загружены в git-репозиторий на сервере http://student.git.athene.tech/

**Порядок выполнения работы**

Во-первых, был разработан общий макет для страниц. Мой макет состоит из:

* Header:
* Включает логотип онлайн-библиотеки "Elib".
* Содержит навигационное меню (navbar) с выпадающим списком для основных страниц («Профиль», «Подарочные сертификаты», «Выйти»), а также отдельную навигацию по книгам.
* Footer:
* Содержит контактную информацию: адрес, телефон и email.

Во-вторых, в «styles.css» была добавлена логика для адаптивной верстки страниц для мобильных устройств, планшетов и ПК.

**Различия между обычной и адаптивной вёрсткой**:

* **Обычная вёрстка** предполагает фиксированные размеры элементов (например, ширина блоков в пикселях), которые не изменяются при изменении размера экрана. Это может привести к некорректному отображению контента на устройствах с разными разрешениями, например, горизонтальной прокрутке на мобильных устройствах или пустым пространствам на больших экранах.
* **Адаптивная вёрстка** использует относительные единицы измерения (%, vw, vh, rem, em), медиа-запросы (@media) и гибкие макеты (например, Flexbox или Grid), чтобы адаптировать интерфейс под различные устройства.

В моем проекте:

* Для мобильных устройств (ширина экрана до 768px) навигационное меню сворачивается в гамбургер-меню, а элементы располагаются в один столбец для удобства чтения.
* Для планшетов (ширина экрана 769–1024px) меню остаётся видимым, но размеры шрифтов и отступов уменьшаются.
* Для ПК (ширина экрана более 1024px) макет раскрывается в полную ширину с сохранением всех элементов в горизонтальной ориентации.

**Скриншоты**

*Страница index.html:*

*Изображение выглядит как снимок экрана, текст, Прямоугольник, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.*

*Страница login.html:*

*Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, диаграмма

Контент, сгенерированный ИИ, может содержать ошибки.*

*Страница register.html:*

*Изображение выглядит как текст, программное обеспечение, снимок экрана, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.*

*Страница sertificates.html:*

*Изображение выглядит как текст, снимок экрана, программное обеспечение, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.*

*Страница user.html:*

*Изображение выглядит как текст, программное обеспечение, компьютер, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.*

**Исходный код**

**eslint.config.js**

import perfectionistPlugin from 'eslint-plugin-perfectionist'; import stylisticJsPlugin from '@stylistic/eslint-plugin-js'; import prettierPlugin from 'eslint-plugin-prettier'; import jsPlugin from '@eslint/js'; import globals from 'globals'; export default [ jsPlugin.configs.recommended, perfectionistPlugin.configs['recommended-line-length'], { rules: { 'stylistic/padding-line-between-statements': [ 'error', { blankLine: 'always', prev: 'import', next: '\*' }, { blankLine: 'any', prev: 'import', next: 'import' }, { blankLine: 'always', next: 'return', prev: '\*' }, { blankLine: 'always', prev: 'block-like', next: '\*' }, { blankLine: 'always', next: 'block-like', prev: '\*' }, { blankLine: 'always', prev: 'export', next: '\*' }, { blankLine: 'always', next: 'export', prev: '\*' }, { prev: 'singleline-const', blankLine: 'always', next: '\*' }, { prev: 'singleline-let', blankLine: 'always', next: '\*' }, { next: 'singleline-const', blankLine: 'always', prev: '\*' }, { next: 'singleline-let', blankLine: 'always', prev: '\*' }, { prev: 'singleline-let', next: 'singleline-let', blankLine: 'any' }, { prev: 'singleline-const', next: 'singleline-const', blankLine: 'any', }, ], 'prettier/prettier': [ 'error', { jsxSingleQuote: false, endOfLine: 'auto', singleQuote: true, }, ], 'no-promise-executor-return': 'error', 'array-callback-return': 'error', 'no-unused-expressions': 'error', 'prefer-arrow-callback': 'error', 'prefer-destructuring': 'error', 'consistent-return': 'error', 'arrow-body-style': 'error', 'object-shorthand': 'error', 'no-return-assign': 'error', 'no-await-in-loop': 'error', 'no-throw-literal': 'error', 'no-extend-native': 'error', 'no-return-await': 'error', 'prefer-template': 'error', 'no-else-return': 'error', 'accessor-pairs': 'error', 'no-lone-blocks': 'error', 'require-await': 'error', 'prefer-const': 'error', 'dot-notation': 'error', 'no-multi-str': 'error', 'camelcase': 'error', 'no-proto': 'error', 'curly': 'error', }, languageOptions: { globals: { ...globals.browser, ...globals.es2025, ...globals.jest, }, }, plugins: { stylistic: stylisticJsPlugin, prettier: prettierPlugin, }, files: ['\*\*/\*.js'], }, { ignores: ['node\_modules', 'vite.config.js'], }, ];

**package.json**

{ "name": "elib", "version": "0.0.0", "devDependencies": { "@eslint/js": "^9.20.0", "@stylistic/eslint-plugin-js": "^3.1.0", "eslint": "^9.20.0", "eslint-plugin-perfectionist": "^4.8.0", "eslint-plugin-prettier": "^5.2.3", "globals": "^15.14.0", "prettier": "^3.5.0", "sass-embedded": "^1.83.4", "vite": "^6.1.0" }, "private": true, "scripts": { "dev": "vite", "build": "vite build", "preview": "vite preview", "lint:fix": "eslint --fix" }, "type": "module", "dependencies": { "epubjs": "^0.3.93" } }

**src\book.html**

<!doctype html> <html lang="ru"> <head> <meta charset="UTF-8" /> <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no" /> <link rel="icon" type="image/svg" href="/icon.svg" /> <title>Книга</title> <script type="module" src="./scripts/book.js" defer></script> <script type="module" src="./styles/pages/book.scss" defer></script> </head> <body> <main class="book-reader"> <button id="prev" type="button"> <img src="/images/icons/arrow-back.svg" /> </button> <div id="area"></div> <button id="next" type="button"> <img src="/images/icons/arrow-forward.svg" /> </button> </main> </body> </html>

**src\index.html**

<!doctype html> <html lang="ru"> <head> <meta charset="UTF-8" /> <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no" /> <link rel="icon" type="image/svg" href="/icon.svg" /> <title>Elib - онлайн библиотека</title> <script src="./index.js" type="module"></script> </head> <body> <header class="header"> <a href="/" class="header\_\_logo"> <img src="/icon.svg" alt="Elib logo" /> </a> <section class="header\_\_controls"> <form class="header\_\_form"> <input name="search" class="header\_\_input" type="text" placeholder="Поиск" spellcheck required /> <button type="submit" class="header\_\_form\_\_button"> <img src="/images/icons/search.svg" alt="search icon" /> </button> </form> <nav class="header\_\_navs"> <div class="user-profile"> <span class="user-profile-toggle"> <p class="user-icon">user</p> <!-- <img src="/path/to/user-icon.svg" alt="User" class="user-icon" /> --> </span> <div class="user-profile-dropdown"> <a href="/profile" class="link">user</a> <a href="/sertificates" class="link">Сертификаты</a> <hr /> <a href="/logout" class="link">Выйти</a> </div> </div> </nav> </section> </header> <main class="main-content"> <section> <div class="main-content\_\_heading-block"> <h2>Фантастика</h2> <hr /> </div> <div class="main-content\_\_books"> <product-card img="/images/covers/harry-potter.jpg" link="/" name="Гарри Поттер и философский камень" ></product-card> <product-card img="/images/covers/harry-potter.jpg" link="/" name="Гарри Поттер и философский камень" ></product-card> <product-card img="/images/covers/harry-potter.jpg" link="/" name="Гарри Поттер и философский камень" ></product-card> <product-card img="/images/covers/harry-potter.jpg" link="/" name="Гарри Поттер и философский камень" ></product-card> <product-card img="/images/covers/harry-potter.jpg" link="/" name="Гарри Поттер и философский камень" ></product-card> <product-card img="/images/covers/harry-potter.jpg" link="/" name="Гарри Поттер и философский камень" ></product-card> <product-card img="/images/covers/harry-potter.jpg" link="/" name="Гарри Поттер и философский камень" ></product-card> </div> </section> <section> <div class="main-content\_\_heading-block"> <h2>Детективы</h2> <hr /> </div> <div class="main-content\_\_books"> <product-card img="/images/covers/sherlock-holmes.jpg" link="/" name="Шерлок Холмс" ></product-card> <product-card img="/images/covers/sherlock-holmes.jpg" link="/" name="Шерлок Холмс" ></product-card> </div> </section> </main> <footer class="footer"> <div class="footer\_\_img"> <img class="footer\_\_img--content" src="/icon.svg" alt="logo" /> </div> <div class="footer\_\_content"> <div class="footer\_\_content\_\_links"> <a class="footer\_\_content\_\_links-link" href="/">Главная</a> <a class="footer\_\_content\_\_links-link" href="/register" >Регистрация</a > <a class="footer\_\_content\_\_links-link" href="/sertificates" >Купоны и бонусы</a > </div> <div>2025-2025 &#128218;</div> </div> </footer> </body> </html>

**src\index.js**

import './styles/style.scss'; import { Card } from './scripts/components/card'; customElements.define('product-card', Card);

**src\login.html**

<!doctype html> <html lang="ru"> <head> <meta charset="UTF-8" /> <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no" /> <link rel="icon" type="image/svg" href="/icon.svg" /> <title>Вход</title> <script src="./index.js" type="module"></script> </head> <body> <header class="header"> <a href="/" class="header\_\_logo"> <img src="/icon.svg" alt="Elib logo" /> </a> <section class="header\_\_controls"> <form class="header\_\_form"> <input name="search" class="header\_\_input" type="text" placeholder="Поиск" spellcheck required /> <button type="submit" class="header\_\_form\_\_button"> <img src="/images/icons/search.svg" alt="search icon" /> </button> </form> <nav class="header\_\_navs"> <div class="user-profile"> <span class="user-profile-toggle"> <p class="user-icon">user</p> <!-- <img src="/path/to/user-icon.svg" alt="User" class="user-icon" /> --> </span> <div class="user-profile-dropdown"> <a href="/profile" class="link">user</a> <a href="/sertificates" class="link">Сертификаты</a> <hr /> <a href="/logout" class="link">Выйти</a> </div> </div> </nav> </section> </header> <main class="main-auth"> <section class="main-auth\_\_form-section"> <div class="main-auth\_\_form-block"> <h1 class="main-auth\_\_heading">С возвращением!</h1> <form class="main-auth\_\_form"> <!-- <label for="email"></label> --> <input id="email" name="email" type="text" placeholder="Email" required /> <!-- <label for="password"></label> --> <input id="password" name="password" type="password" placeholder="Пароль" required /> <button type="submit" class="primary-button primary-button\_\_large"> Войти </button> </form> <a class="link" href="/register">зарегистрироваться</a> </div> </section> <section class="main-auth\_\_bg-image"> <img src="/images/posters/book-shkaf.svg" /> </section> </main> <footer class="footer"> <div class="footer\_\_img"> <img class="footer\_\_img--content" src="/icon.svg" alt="logo" /> </div> <div class="footer\_\_content"> <div class="footer\_\_content\_\_links"> <a class="footer\_\_content\_\_links-link" href="/">Главная</a> <a class="footer\_\_content\_\_links-link" href="/register" >Регистрация</a > <a class="footer\_\_content\_\_links-link" href="/sertificates" >Купоны и бонусы</a > </div> <div>2025-2025 &#128218;</div> </div> </footer> </body> </html>

**src\register.html**

<!doctype html> <html lang="ru"> <head> <meta charset="UTF-8" /> <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no" /> <link rel="icon" type="image/svg" href="/icon.svg" /> <title>Регистрация</title> <script src="./index.js" type="module"></script> </head> <body> <header class="header"> <a href="/" class="header\_\_logo"> <img src="/icon.svg" alt="Elib logo" /> </a> <section class="header\_\_controls"> <form class="header\_\_form"> <input name="search" class="header\_\_input" type="text" placeholder="Поиск" spellcheck required /> <button type="submit" class="header\_\_form\_\_button"> <img src="/images/icons/search.svg" alt="search icon" /> </button> </form> <nav class="header\_\_navs"> <div class="user-profile"> <span class="user-profile-toggle"> <p class="user-icon">user</p> <!-- <img src="/path/to/user-icon.svg" alt="User" class="user-icon" /> --> </span> <div class="user-profile-dropdown"> <a href="/profile" class="link">user</a> <a href="/sertificates" class="link">Сертификаты</a> <hr /> <a href="/logout" class="link">Выйти</a> </div> </div> </nav> </section> </header> <main class="main-auth"> <section class="main-auth\_\_form-section"> <div class="main-auth\_\_form-block"> <h1 class="main-auth\_\_heading">Добро пожаловать!</h1> <form class="main-auth\_\_form"> <label for="email">введите email</label> <input id="email" name="email" type="email" placeholder="Email" required /> <label for="password">придумайте пароль</label> <input id="password" name="password" type="password" placeholder="Пароль" required /> <label for="cc-number">и отдайте все ваши деньги</label> <input id="cc-number" type="number" autocomplete="cc-number" placeholder="xxxx xxxx xxxx xxxx" /> <button type="submit" class="primary-button primary-button\_\_large"> Зарегистрироваться </button> </form> <a class="link" href="./login">войти</a> </div> </section> <section class="main-auth\_\_bg-image"> <img src="/images/posters/book-shkaf.svg" /> </section> </main> <footer class="footer"> <div class="footer\_\_img"> <img class="footer\_\_img--content" src="/icon.svg" alt="logo" /> </div> <div class="footer\_\_content"> <div class="footer\_\_content\_\_links"> <a class="footer\_\_content\_\_links-link" href="/">Главная</a> <a class="footer\_\_content\_\_links-link" href="/register" >Регистрация</a > <a class="footer\_\_content\_\_links-link" href="/sertificates" >Купоны и бонусы</a > </div> <div>2025-2025 &#128218;</div> </div> </footer> </body> </html>

**src\scripts\book.js**

import ePub from 'epubjs'; class BookReader { constructor(bookPath) { console.log('[Constructor] Initializing with book path:', bookPath); this.book = ePub(bookPath); this.rendition = null; this.isReady = false; this.book.on('error', error => { console.error('[Book Error]', error); }); } async init() { try { console.log('[Init] Starting initialization...'); await this.book.ready; console.log('[Init] Book metadata:', this.book.metadata); console.log('[Init] Creating rendition...'); this.rendition = this.book.renderTo('area', { allowScriptedContent: true, height: '100%', spread: 'none', width: '100%', }); console.log('[Init] Displaying content...'); await this.rendition.display(); this.isReady = true; console.log('[Init] Rendition ready:', this.rendition); console.log('[Init] Current location:', this.rendition.currentLocation()); } catch (error) { console.error('[Init Error]', error); throw error; } } async prevPage() { console.log('[Navigation] Attempting previous page...'); if (!this.isReady) { console.warn('[Navigation] Reader not ready!'); return; } try { await this.rendition.prev(); console.log( '[Navigation] New location:', this.rendition.currentLocation(), ); } catch (error) { console.error('[Navigation Error]', error); } } async nextPage() { console.log('[Navigation] Attempting next page...'); if (!this.isReady) { console.warn('[Navigation] Reader not ready!'); return; } try { await this.rendition.next(); console.log( '[Navigation] New location:', this.rendition.currentLocation(), ); } catch (error) { console.error('[Navigation Error]', error); } } } // Инициализация let reader = null; document.addEventListener('DOMContentLoaded', async () => { console.log('[DOM] Content loaded'); try { const params = new URLSearchParams(window.location.search); const bookPath = params.get('book'); console.log('[DOM] Book path from URL:', bookPath); if (!bookPath) { throw new Error('Add ?book= parameter to URL'); } reader = new BookReader(bookPath.replace('-', '').concat('.epub')); console.log('[DOM] Reader instance created'); await reader.init(); console.log('[DOM] Reader initialization completed'); } catch (error) { console.error('[DOM Error]', error); alert(`Fatal error: ${error.message}`); } }); const [next, prev] = document.querySelectorAll('button'); next.addEventListener('click', async () => { console.log('[UI] Prev button clicked'); await reader?.prevPage(); }); prev.addEventListener('click', async () => { console.log('[UI] Next button clicked'); await reader?.nextPage(); });

**src\scripts\components\card.js**

import stylesString from './card.scss?inline'; export class Card extends HTMLElement { static get observedAttributes() { return ['img', 'name', 'link', 'icon']; } #link; #name; /\* захардкоженная иконка на кнопке \*/ #icon; /\* обложка \*/ #img; constructor() { super(); this.#icon = '/images/icons/add.svg'; } render() { if (this.#link !== '/') { this.#icon = '/images/icons/resume.svg'; } return ` <style> ${stylesString} </style> <div class="card"> <div class="card\_\_cover"> <img class="card\_\_cover\_\_img" src="${ this.#img }" alt="${this.#name} cover" lazy /> </div> <div class="card\_\_description"> <span class="card\_\_description\_\_text"> ${this.#name} </span> <button class="card\_\_button"> <a href="${this.#link}"> <img src="${this.#icon}" class="card\_\_button\_\_img" alt="add icon" /> </a> </button> </div> </div> `; } attributeChangedCallback(name, oldValue, newValue) { if (name === 'img') { this.#img = newValue; } else if (name === 'href') { this.#link = newValue; } else if (name === 'name') { this.#name = newValue; } else if (name === 'icon') { if (newValue) { this.#icon = newValue; } } } updateData() { this.#img = this.getAttribute('img'); this.#link = this.getAttribute('link'); this.#name = this.getAttribute('name'); // this.#icon = this.getAttribute("icon"); } connectedCallback() { this.attachShadow({ mode: 'open' }); this.updateData(); this.shadowRoot.innerHTML = this.render(); } }

**src\sertificates.html**

<!doctype html> <html lang="ru"> <head> <meta charset="UTF-8" /> <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no" /> <link rel="icon" type="image/svg" href="/icon.svg" /> <title>Сертификаты</title> <script src="./index.js" type="module"></script> </head> <body> <header class="header"> <a href="/" class="header\_\_logo"> <img src="/icon.svg" alt="Elib logo" /> </a> <section class="header\_\_controls"> <form class="header\_\_form"> <input name="search" class="header\_\_input" type="text" placeholder="Поиск" spellcheck required /> <button type="submit" class="header\_\_form\_\_button"> <img src="/images/icons/search.svg" alt="search icon" /> </button> </form> <nav class="header\_\_navs"> <div class="user-profile"> <span class="user-profile-toggle"> <p class="user-icon">user</p> <!-- <img src="/path/to/user-icon.svg" alt="User" class="user-icon" /> --> </span> <div class="user-profile-dropdown"> <a href="/profile" class="link">user</a> <a href="/sertificates" class="link">Сертификаты</a> <hr /> <a href="/logout" class="link">Выйти</a> </div> </div> </nav> </section> </header> <main class="main-sertificates"> <div class="heading"> <div class="main-sertificates\_\_head"> <h2 class="main-sertificates\_\_head\_\_info-text">Дарите знания!</h2> <div class="main-sertificates\_\_head\_\_logo"> <img class="main-sertificates\_\_head\_\_logo-img" src="/images/posters/book-present.svg" alt="banner logo" /> </div> </div> </div> <div class="main-sertificates\_\_suggestions"> <section class="main-sertificates\_\_banner"> <div class="main-sertificates\_\_banner\_\_logo"> <img class="main-sertificates\_\_banner\_\_logo-img" src="/images/posters/book-3-m.svg" alt="banner logo" /> </div> <div class="main-sertificates\_\_banner\_\_info"> <h3 class="main-sertificates\_\_banner\_\_info-text"> Подписка на 3 месяца </h3> <span class="main-sertificates\_\_banner\_\_info-description" >Всё, от Гарри Поттера до Незнайки. Открывайте для себя новые удивительные миры и занимательные истории!</span > <button type="button" class="primary-button">Купить</button> </div> </section> <section class="main-sertificates\_\_banner"> <div class="main-sertificates\_\_banner\_\_logo"> <img class="main-sertificates\_\_banner\_\_logo-img" src="/images/posters/book-3-m.svg" alt="banner logo" /> </div> <div class="main-sertificates\_\_banner\_\_info"> <h3 class="main-sertificates\_\_banner\_\_info-text"> Подписка на 3 месяца </h3> <span class="main-sertificates\_\_banner\_\_info-description" >Всё, от Гарри Поттера до Незнайки. Открывайте для себя новые удивительные миры и занимательные истории!</span > <button type="button" class="primary-button">Купить</button> </div> </section> </div> </main> <footer class="footer"> <div class="footer\_\_img"> <img class="footer\_\_img--content" src="/icon.svg" alt="logo" /> </div> <div class="footer\_\_content"> <div class="footer\_\_content\_\_links"> <a class="footer\_\_content\_\_links-link" href="/">Главная</a> <a class="footer\_\_content\_\_links-link" href="/register" >Регистрация</a > <a class="footer\_\_content\_\_links-link" href="/sertificates" >Купоны и бонусы</a > </div> <div>2025-2025 &#128218;</div> </div> </footer> </body> </html>

**src\user.html**

<!doctype html> <html lang="ru"> <head> <meta charset="UTF-8" /> <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no" /> <title>Личный кабинет</title> <link rel="icon" type="image/svg" href="/icon.svg" /> <script type="module" src="./index.js"></script> </head> <body> <header class="header"> <a href="/" class="header\_\_logo"> <img src="/icon.svg" alt="Elib logo" /> </a> <section class="header\_\_controls"> <form class="header\_\_form"> <input name="search" class="header\_\_input" type="text" placeholder="Поиск" spellcheck required /> <button type="submit" class="header\_\_form\_\_button"> <img src="/images/icons/search.svg" alt="search icon" /> </button> </form> <nav class="header\_\_navs"> <div class="user-profile"> <span class="user-profile-toggle"> <p class="user-icon">user</p> <!-- <img src="/path/to/user-icon.svg" alt="User" class="user-icon" /> --> </span> <div class="user-profile-dropdown"> <a href="/profile" class="link">user</a> <a href="/sertificates" class="link">Сертификаты</a> <hr /> <a href="/logout" class="link">Выйти</a> </div> </div> </nav> </section> </header> <main class="main-content"> <section> <div class="main-content\_\_heading-block"> <h2>Ваши книги, <span style="color: red">user</span></h2> <hr /> </div> <div class="main-content\_\_books"> <product-card img="/images/covers/harry-potter.jpg" link="/book.html?book=harry-potter" name="Гарри Поттер и философский камень" ></product-card> <product-card img="/images/covers/harry-potter.jpg" link="/book.html?book=harry-potter" name="Гарри Поттер и философский камень" ></product-card> </div> </section> </main> <footer class="footer"> <div class="footer\_\_img"> <img class="footer\_\_img--content" src="/icon.svg" alt="logo" /> </div> <div class="footer\_\_content"> <div class="footer\_\_content\_\_links"> <a class="footer\_\_content\_\_links-link" href="/">Главная</a> <a class="footer\_\_content\_\_links-link" href="/register" >Регистрация</a > <a class="footer\_\_content\_\_links-link" href="/sertificates" >Купоны и бонусы</a > </div> <div>2025-2025 &#128218;</div> </div> </footer> </body> </html>

**vite.config.js**

import { defineConfig } from 'vite'; export default defineConfig({ root: 'src', build: { outDir: '../dist', }, });