Курсова робота. Щедровський Іван, РПЗ 19 2/9

Version: 0.3.0

Changelog:

## 0.1.0 - 24.09 23:30

Створений вступ та пункт 1.1 - основні поняття

## 0.2.0 - 25.09 00:30

Створений пункт 1.2

## 0.3.0 - 05.10 20:50

Створенні пункти 1.3 та 1.4

## 0.3.1 - 05.10 21:00

Додано в 1.1 “тематикою моєї роботи” є …  
Додані нові priority tasks

## 0.4.0 - 17.10 16:40

Створено розділ 2.1

====

Todo:

* [ ] В пункт 1.2 додати картинки
* [ ] Перечитати та дописати пункт 1.2
* [ ] В розділ 1.4 додати графік поширення браузерів
* [ ] Додати більше інформації в розділ 1.4 про Node.js

Priority:

* [ ] Рекомендую також додати вимоги до оформлення, вимоги до надійності, стадії й етапи розробки програмного проекту
* [ ] Огляд термінів - добре, але опису предметної області не достатньо - тут доречно було б додати огляд архітектур

Вступ

З розвитком прогресу і винайденням інтернету, об’єм доступної людині інформації зростає з кожним днем. Якщо раніше отримання певної інформації для більшості людей було не тільки важким, а іноді і неможливим завданням, сьогодні дістати з кишені смартфон та набрати в ньому запит в популярній пошуковій мережі не є проблемою: доступ до всесвітньої мережі даних, мільйонів документів є майже у кожної людини.

Інформаційні технології мають сьогодні пріоритетне значення в багатьох сферах діяльності й визначають майбутній розвиток суспільства.

Одним з таких напрямків є інтернет-бізнес, який починає привертати все більше значення в сучасному світі, і в нашій країні відповідно, у зв'язку з тенденцією до загальної глобалізації економіки. Інтернет і електронна торгівля грають в цьому процесі одну з найважливіших ролей. Поява абсолютного нового виду зв'язку зумовило перегляд підприємців процесу організації бізнесу.

Інтернет відкриває широкі можливості не тільки для великих підприємств. Багато початківців бізнесменів відразу відкривають свою справу в Інтернеті, так як для цього потрібна менша сума первинних інвестицій, ніж у сфері традиційної торгівлі. А компанії, які працюють за принципом прямих поставки, вкладають кошти виключно в веб-сайт і рекламу.

Західні покупці, а вслід за ними і українські, все частіше купують товари, не виходячи з дому, посилаючись на нестачу часу, невеликий асортимент товарів у традиційних магазинах і високі ціни. В Україні в початку 2000-х тільки інтузіасти зробили електронні покупки, сьогодні, за даними статистики, хоча б раз це зробили 4 з 5 чоловік. Розвиток бізнесу в інтернеті на сьогоднішній день крайне перспективне напрямок для підприємців, що і обумовлює актуальність даної роботи.

Розробка веб-сайтів електронної комерції – це нелегке завдання. У нього є свої проблеми. Оскільки велика кількість веб-користувачів відвідують сайт електронної комерції щодня, тому потрібно його підтримувати. Для цього може знадобитися залучити експертів веб-розробників електронної комерції від надійної та відомий компанії з розробки електронної комерції веб-сайтів, що забезпечує інтуїтивний процес веб-розробки електронної комерції з легкістю.

Темою моєї курсової роботи було обрано "Розробка багатошарової клієнт-серверної архітектури на прикладі створення сервісу онлайн продаж". Метою є створення зручного, легкого в використанні сервісу онлайн продаж.

Середою розробки сервісу було обрано сучасну інтегровану середу Visual Studio Code. Ця середа дозволяє швидко та ефективно працювати при написанні коду, налагодженні, компілюванні та складанні проекту.

1 Опис предметної області

1.1 Основні поняття

Предметна область — це частина реального світу, що розглядається в межах певного контексту.

Тематикою моєї роботи є створення сервісу онлайн продажів.

Сервіс онлайн продажів - місце в інтернеті, де відбувається прямий продаж товарів споживачеві (юридичній або фізичній особі), враховуючи доставку. При цьому розміщення споживацької інформації, замовлення товару і угода відбуваються там само, всередині мережі (на сайті інтернет-магазину).

Розробка програмного забезпечення — це процес, спрямований на створення та підтримку працездатності, якості та надійності програмного забезпечення, використовуючи технології, методологію та практики з інформатики, керування проєктами, математики, інженерії та інших областей знання.

Архітектура програмного забезпечення — спосіб структурування програмної або обчислювальної системи, абстракція елементів системи на певній фазі її роботи. Система може складатись з кількох рівнів абстракції і мати багато фаз роботи, кожна з яких може мати окрему архітектуру.

Архітектура клієнт-сервер є одним із архітектурних шаблонів програмного забезпечення та є домінуючою концепцією у створенні розподілених мережних застосунків і передбачає взаємодію та обмін даними між ними.

Сервіс — це обслуговування населення, забезпечення його побутових потреб.

Електронна комерція — це сфера цифрової економіки, що включає всі фінансові та торгові транзакції, які проводяться за допомогою комп'ютерних мереж, та бізнес-процеси, пов'язані з проведенням цих транзакцій.

Елементами сервісу онлайн продаж є: каталог, товар, кошик, особистий кабінет користувача, замовлення

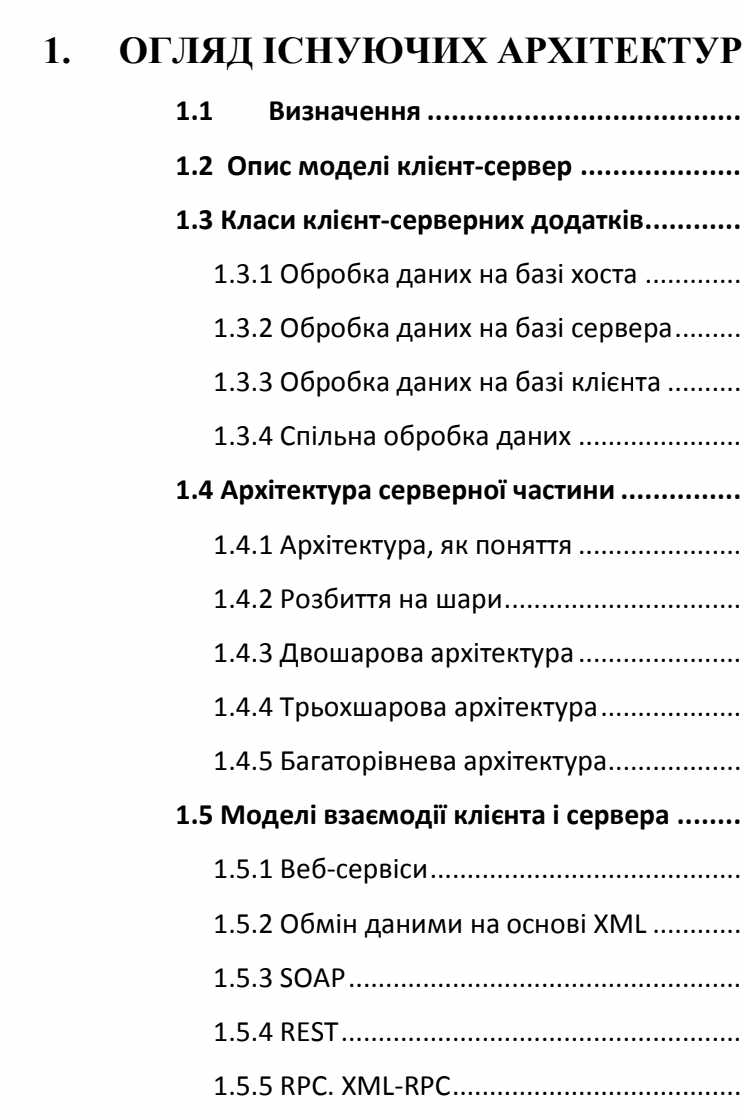
Каталог — список, перелік товарів, складений у певному порядку, щоб полегшити їхній пошук.

Товар — продукт природи і людської праці або тільки людської праці у матеріальній і нематеріальній субстанції та у формі послуг, який завдяки своїм властивостям здатний задовольняти наявні чи передбачувані суспільні потреби.

Кошик - список, перелік товарів які користувач планує купити. Створений щоб полегшити перегляд, видалення або редагування кількості товарів

Особистий кабінет - це особливий розділ, який дозволяє користувачу отримати доступ до даних про стан та деталей замовлення.

// in todo

Огляд термінів - добре, але опису предметної області не достатньо - тут доречно було б додати огляд архітектур  
https://classroom.google.com/c/NDg5NDczMTg4ODkx/a/NDg5NDczMTg4ODk4/details  


1.2 Постанова завдання

Метою курсової роботи є створення системи онлайн продаж. Дана система здійснює продаж аксесуарів для домашніх тварин.

Система повинна дозволити покупцям здійснювати швидкий та детальний перегляд продукції.

Сайт повинен включати в себе функцію реєстрації користувача, залишення відгуків й коментарів та здійснення замовлення.

Після вивчення предметної області і створення відповідної бази даних необхідно створити сайт, який має наступні функції:

* відображення даних про категорії товарів в меню (`тут буде рисунок`);
* відображення основної інформації про всі товари у вигляді карток (назва, ціна, зображення);
* відображення детальної інформації про товар (назва, зображення, ціна, опис, кількість на складі) (`тут буде рисунок`);
* відображення списку нових (`тут буде рисунок`) та найпопулярніших (`тут буде рисунок`) товарів на головній сторінці сайту;
* зручне та швидке маніпулювання даними адміністратором (редагування, видалення, додавання інформації);
* зручне та швидке маніпулювання даними модератором (редагування, видалення коментарів);
* обчислення вартості всіх замовлених товарів;
* вибір способу оплати;
* вибір способу доставки;
* фільтр для товарів по категоріям;
* фільтр для товарів по ціні;
* сортування товарів по ціні (від найменшої до найбільшої та навпаки);
* зручний перехід між категоріями;
* реєстрація на сайті(`тут буде рисунок`);
* авторизація на сайті(`тут буде рисунок`);
* зручне та швидке оформлення замовлення.

// in todo  
Рекомендую також додати вимоги до оформлення, вимоги до надійності, стадії й етапи розробки програмного проекту

1.3 Обґрунтування вибору середовища розробки системи

Visual Studio Code - інтегрована середа розробки програмних продуктів, розроблена Microsoft та доступна на всіх операційних системах.

Visual Studio Code надає можливість швидко розробляти продукти завдяки вбудованому функціоналу та багатьом розширенням.

Також ця середа розробки надає можливість запускати та відлагувати програму завдяки точкам зупинки, стеку викликів та інтерактивній консолі

Visual Studio Code має вбудований графічний інтерфейс для роботи с Git, який надає можливість переглядати відмінності, проміжні файли, робити знімки змін, відправляти знімки на сервер та отримувати їх.

Редактор має багато розширень, які дозволяють налаштувати його як візуальну складову, так і функціональну [1]

Git — це безкоштовна розподілена система контролю версій із відкритим вихідним кодом, призначена для швидкої та ефективної обробки будь-яких проектів, від малих до дуже великих.

Git простий у вивченні та має невеликий розмір з блискавичною продуктивністю. Він перевершує такі інструменти SCM, як Subversion, CVS, Perforce і ClearCase, завдяки таким функціям, як дешеве локальне розгалуження, зручні робочі області та численні робочі процеси. [2]

Postman — це платформа API для створення та використання API. Postman спрощує та оптимізує кожен крок життєвого циклу API щоб створювати кращі API — швидше.

Postman може зберігати та керувати специфікаціями API, документацією, рецептами робочих процесів, тестами та результатами, показниками та всім іншим, що стосується API.

Платформа Postman містить повний набір інструментів, які допомагають прискорити життєвий цикл API — від проектування, тестування, документування та моделювання до спільного використання та видимості ваших API.

Підхід Postman до управління, що базується на повному життєвому циклі, дає змогу відмовитись від розробки свойого сервісу, створюючи API кращої якості та сприяючи співпраці між командами розробників і командами дизайнерів API. [3]

Swagger - сервіс, який дозволяє спростити розробку API за допомогою професійного набору інструментів з відкритим кодом.

Swagger дозволяє спроектувати та змоделювати API відповідно до стандартів на основі специфікацій.

Покращує та спрощує роботу завдяки інтерактивної документації API.

Також Swagger виконує прості функціональні тести API без додаткових витрат [4]

1.4 Обґрунтування вибору середовища функціонування системи

Середовищем функціонування сайту є браузер.

Браузер — програмне забезпечення для комп'ютера або іншого електронного пристрою, як правило, під'єднаного до Інтернету, що дає можливість користувачеві взаємодіяти з текстом, малюнками або іншою інформацією на гіпертекстовій веб сторінці. Тексти та малюнки можуть містити посилання на інші веб сторінки, розташовані на тому ж веб сайті або на інших веб сайтах. Веб Переглядач з допомогою гіперпосилань дозволяє користувачеві швидко та просто отримувати інформацію, розміщену на багатьох веб сторінках.

Різні браузери мають різні реалізації одних і тих-же речей, а також різну підтримку нових функцій. Сайт буде коректно працювати в будь-якому сучасному браузері, а саме в:

* Google Chrome версії 84 та вище
* Microsoft Edge версії 84 та вище
* Safari версії 14.1 та вище
* Firefox версії 63 та вище
* Opera версії 70 та вище

Сайт не буде працювати в Internet Explorer будь якої версії.

Середовищем функціонування сайту є Node.js.

Node.js - це JavaScript–оточення побудоване на JavaScript–рушієві Chrome V8. Було обрано саме Node.js через швидкість та зручність розробки, а також за можливість використовувати один і той самий код, як на сайті, так і не сервері. [5]

2 Спеціальний розділ

2.1 Основні рішення щодо реалізації компонентів системи

JavaScript – це легковажна (легковагова або легка, якщо від англ lightweight), інтерпретована або JIT-компілюванна, об'єктно-орієнтована мова з функціями першого класу. Найширше застосування знаходить мову сценаріїв веб-сторінок, але також використовується і в інших програмних продуктах, наприклад, Node.js або Apache CouchDB.

JavaScript це прототипно-орієнтована, мультипарадигменна мова з динамічною типізацією, яка підтримує об'єктно-орієнтований, імперативний та декларативний (наприклад, функціональне програмування) стилі програмування. [6]

TypeScript — це строго типізована мова програмування, яка ґрунтується на JavaScript і надає кращі інструменти будь-якого масштабу.

TypeScript додає до JavaScript додатковий синтаксис для підтримки більш тісної інтеграції з редактором. Дозволяє ловити помилки на ранніх стадіях розробки.

Код TypeScript перетворюється на JavaScript, який виконується будь-де, де працює JavaScript: у браузері, на Node.js або Deno.

TypeScript розуміє JavaScript і використовує визначення типу, щоб надати чудові інструменти без додаткового коду. [7]

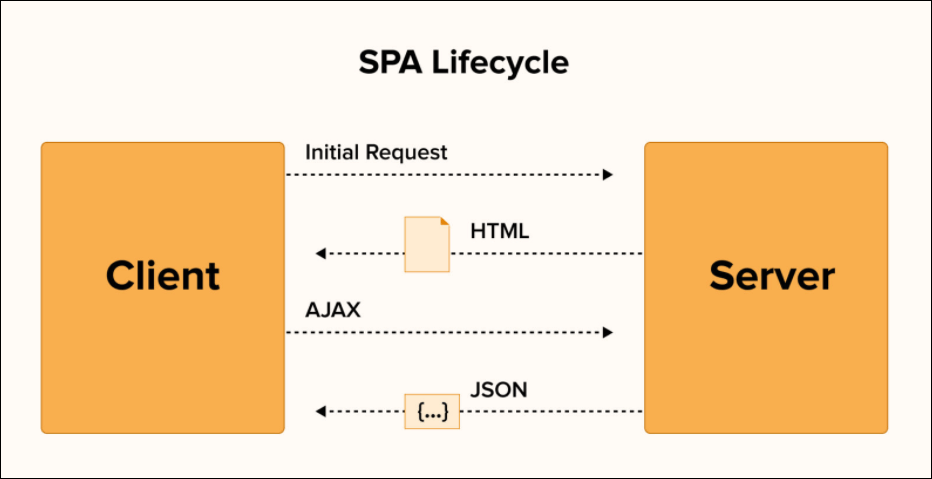
в тебе занадто різкі переходи до означень. Потрібно написати, щось типу “клієнт реалізовано у вигляді SPA додатку” і потім вже означення. Щоб не було просто набору означень, а читалося як суцільний опис

SPA (односторінкова програма) — це реалізація веб-програми, яка завантажує лише один веб-документ, а потім оновлює основний вміст цього єдиного документа за допомогою JavaScript API, таких як XMLHttpRequest і Fetch, коли має бути показано інший вміст.

Таким чином, це дозволяє користувачам використовувати веб-сайти без завантаження цілих нових сторінок із сервера, що може призвести до підвищення продуктивності та більш динамічного досвіду, з деякими компромісними недоліками, такими як SEO, більше зусиль, необхідних для підтримки стану, реалізації навігації та досягнення значущої продуктивності моніторинг.[8]

Візуалізація на стороні сервера (SSR) — це здатність програми перетворювати файли HTML на сервері на повністю відтворену сторінку HTML для клієнта. Веб-браузер надсилає запит на інформацію від сервера, який миттєво відповідає, надсилаючи повністю відтворену сторінку клієнту. Пошукові системи можуть сканувати та індексувати вміст перед доставкою, що корисно для цілей пошукової оптимізації.[9]

Схема роботи програми:



Картинка: “Схема роботи програми”

Основні маршрути програми:

* “/” - Домашня сторінка
* “/catalogs” - Сторінка каталогу товарів
* “\*” - Сторінка помилки(404)
* “/signin” - Сторінка авторизації
* “/signup” - Сторінка реєстрації
* “/account” - Сторінка особистого акаунту

Multiple page application - це веб-програма, яка складається з кількох веб-сторінок, які завантажуються, коли користувач відвідує різні частини сторінки. Це традиційний шаблон розробки веб-додатків, який підходить для веб-сайтів, які мають працювати з великою кількістю вмісту. [10]

SEO (Search Engine Optimization) — це процес, який робить веб-сайт більш помітним у результатах пошуку, що також називається покращенням рейтингу пошуку. [11]

Було обрано саме SPA, тому-що односторінкові програми ідеально підходять, якщо потрібно створити динамічне рішення з обмеженим обсягом даних. Незважаючи на те, що вони працюють у браузері, односторінкові програми виглядають як настільні чи мобільні програми, тому вони такі популярні.

В порівнянні з MPA, SPA виграє в усьому, окрім SEO. Але ця проблема вирішується за допомогою використання SSR

/\* Тут буде секція SSR \*/

Webpack — це інструмент збірки на основі бандлера. Це означає, що для обслуговування програми йому потрібно просканувати, обробити та об’єднати весь JavaScript програми в один файл. Це стосується як встановлених залежностей, так і коду.

Це причина, чому створення програми для розробки та запуск сервера розробки за допомогою Webpack може зайняти дуже багато часу

Після зберігання файлу, весь пакет JavaScript буде побудовано за допомогою Webpack, тому зміна може відображатися у браузері протягом 10 секунд, навіть якщо HMR увімкнено. Повільний цикл зворотного зв’язку, викликаний Webpack, створює поганий досвід для розробників(DX). [12]

Гаряча заміна модулів (HMR) обмінює, додає або видаляє модулі під час роботи програми без повного перезавантаження.[13]

Досвід розробника (DX) описує досвід розробників під час використання або роботи над вашим продуктом. Це пакет як позитивних, так і негативних почуттів.[14]

Vite — це інструмент для створення JavaScript нового покоління, який має на меті покращити досвід розробників, які раніше працювали з Webpack, коли йдеться про створення програм JavaScript.

Інструмент використовує два вдосконалення, нещодавно внесені в екосистему JavaScript, щоб запропонувати кращий досвід розробника: доступність модулів ES у веб-переглядачі та інструменти збирання компіляції до рідних, як-от esbuild.

Основна ідея Vite проста: трансформування та обслуговування частину коду вашої програми за допомогою модулів ES, коли це запитує браузер.

У той час як робочий процес на основі бандлера, як-от Webpack, повинен буде обробити всі модулі JavaScript перед одним запитом браузера, Vite обробляє лише потрібні модулі залежностей перед одним запитом браузера.

Ось чому Vite може обробити збірку для розробки швидше, ніж Webpack. По мірі зростання програми, час, який необхідний щоб обробити збірку для розробки, зростатиме повільно. [15]

Виходячи з вище перерахованої різниці webpack та vite, для розробки сервісу було обрано vite.

Node.js - це JavaScript–оточення побудоване на JavaScript–рушієві Chrome V8. Було обрано саме Node.js через швидкість та зручність розробки, а також за можливість використовувати один і той самий код, як на сайті, так і не сервері. [5]

npm(Node Package Manager) - менеджер пакетів, що входить до складу Node.js, найбільший у світі реєстр програмного забезпечення. Розробники пакетів з відкритим кодом з усіх континентів використовують npm для спільного використання та запозичення пакетів, а багато організацій також використовують npm для керування приватною розробкою. [16]

pnpm - це новий менеджер пакунків для JavaScript, створений на основі npm для спрощення процесу інсталяції пакетів у програмах вузла. PNPM є альтернативою NPM. Він дотримується тих же принципів, що й NPM, але має деякі додаткові функції, які роблять його потужнішим, ніж його попередник. [17]

Vue.js - Доступна, ефективна та універсальна структура для створення веб-інтерфейсів користувача. [18]

В порівнянні з React, Vue є більш швидшим, як зі сторони розробки, так і зі сторони виконання коду. Vue має реактивність та двосвязаність, якої немає в React. Також Vue має офіційну реалізацію роутеру, глобального сховища та тестових утіліт.

vitest - Сверхшвидка середа модульного тестування. В порівнянні з найбільш популярною середою запусків тестів jest - vitest є набагато більш швидшим при виконанні тестів без включених потоків. При використанні vitest разом з vite різниця з jest стає більш замітною. [19]

SCSS - найдосконаліша, стабільна та найпотужніша професійна мова розширення CSS у світі. [20]

SCSS має багато переваг над звичайним css, а саме:

* Вкладеність css властивостей одна в одну, що робить код більш зрозумілим.
* Змінні. SCSS має свою реалізацію зміних, які можна використовувати в циклах та генерувати значення динамічно
* SCSS має більш досконалу реалізацію операторів: можливо додавати, віднімати та перемоножувати значення CSS.
* Функції - SCSS дозволяє багатократно використовувати стилі css як функції
* Міксини - дозволяють один раз створити набір правил css, та потім використовувати їх багатократно, або змішувати з другими правилами.

MongoDB - система управління базами даних, яка працює з документоорієнтованою моделлю даних. На відміну від реляційних СУБД, MongoDB не потребує таблиці, схеми або окремої мови запитів. Інформація зберігається як документів чи колекцій.

Розробники позиціонують продукт як проміжну ланку між класичними СУБД та NoSQL. MongoDB не використовує схеми, як це роблять реляційні бази даних, що підвищує продуктивність усієї системи. [21]

Express - це мінімалістичний та гнучкий веб-фреймворк для програм Node.js, що надає широкий набір функцій для мобільних та веб-додатків.[22]

Протокол передачі гіпертексту (HTTP) — це протокол прикладного рівня для передачі гіпермедійних документів, наприклад HTML. Його розроблено для зв’язку між веб-браузерами та веб-серверами, але його також можна використовувати для інших цілей. HTTP відповідає класичній моделі клієнт-сервер, коли клієнт відкриває з’єднання, щоб зробити запит, а потім чекає, поки не отримає відповідь. HTTP є протоколом без стану, тобто сервер не зберігає жодних даних (стану) між двома запитами. [23]

REST (Representational state transfer) – це стиль архітектури програмного забезпечення для розподілених систем, таких як World Wide Web, який зазвичай використовується для побудови веб-служб. [24]

У випадку REST є дуже простим інтерфейсом управління інформацією без використання якихось додаткових внутрішніх прошарків. Кожна одиниця інформації однозначно визначається глобальним ідентифікатором, наприклад URL. Кожна URL, у свою чергу, має строго заданий формат.

Відсутність додаткових внутрішніх прошарків означає передачу даних у тому вигляді, як і самі дані. Тобто. ми не загортаємо дані в XML, як це робить SOAP та XML-RPC, не використовуємо AMF, як це робить Flash і т.д. Просто надаємо самі дані.

Кожна одиниця інформації однозначно визначається URL – це означає, що URL є первинним ключем для одиниці даних.

Як відбувається управління інформацією сервісу – це повністю й грунтується на протоколі передачі. Найбільш поширений протокол звичайно ж є HTTP. Так ось, для HTTP дія над даними визначається за допомогою методів: GET (отримати), PUT (додати, замінити), POST (додати, змінити, видалити), DELETE (видалити). Таким чином, дії CRUD (Create-Read-Update-Delete) можуть виконуватися як з усіма чотирма методами, так і лише за допомогою GET та POST.

Fetch API надає інтерфейс для отримання ресурсів (включно з мережею). [25]

Список літератури:

[1] - <https://code.visualstudio.com/#meet-intellisense>

[2] - <https://git-scm.com/>

[3] - <https://www.postman.com/>

[4] - <https://swagger.io/>

[5] - <https://nodejs.org>

[6] - <https://developer.mozilla.org/ru/docs/Web/JavaScript>

[7] - <https://www.typescriptlang.org/>

[8] - https://developer.mozilla.org/en-US/docs/Glossary/SPA

[9] - <https://www.heavy.ai/technical-glossary/server-side-rendering>

[10] - https://yojji.io/blog/spa-vs-mpa

[11] - <https://developer.mozilla.org/en-US/docs/Glossary/SEO>

[12] - <https://webpack.js.org/>

[13] - <https://webpack.js.org/concepts/hot-module-replacement/>

[14] - <https://developerexperience.io/practices/good-developer-experience>

[15] - <https://vitejs.dev/>

[16] - <https://docs.npmjs.com/about-npm>

[17] - <https://pnpm.io/>

[18] - <https://vuejs.org/>

[19] - <https://vitejs.dev/>

[20] - <https://www.freecodecamp.org/news/give-your-css-superpowers-by-learning-sass>

[21] - <https://itglobal.com/ru-ru/company/glossary/mongodb/>

[22] - https://expressjs.com/

[23] - https://developer.mozilla.org/en-US/docs/Web/HTTP

[24] - <https://habr.com/ru/post/38730/>

[25] - https://developer.mozilla.org/en-US/docs/Web/API/Fetch\_API