

Міністерство освіти і науки України

Український державний університет науки і технологій

Факультет «Комп'ютерні технології і системи»

Кафедра «Комп'ютерні інформаційні технології»

### Пояснювальна записка


до кваліфікаційної

ОС Магістр

на тему: ВПЛИВ СТРАТЕГІЙ ПОПЕРЕДНЬОГО ЗАВАНТАЖЕННЯ РЕСУРСІВ  
НА ШВИДКІСТЬ ВІДОБРАЖЕННЯ ВЕБ-СТОРИНОК

за освітньою програмою: 12 Інженерія програмного забезпечення зі  
спеціальності: 121 Інженерія програмного забезпечення

Виконав: студент групи ПЗ2422:



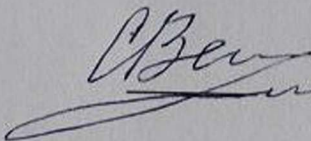
/ Андрій ПРИЙМИЧ /

Керівник:



/ Тетяна  
ГРИШЕЧКІНА /

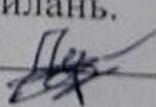
Нормоконтролер:



/ Світлана ВОЛКОВА /

Засвідчую, що у цій роботі немає запозичень з  
праць інших авторів без відповідних посилань.

Студент

Приймаг А.М. 

Ukrainian State University of Science and Technologies

Faculty «Computer technologies and systems»  
Department «Computer information technology»

### Explanatory Note

to Master's Thesis on

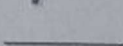
the topic: The impact of resource preloading strategies on web page rendering speed  
according to educational curriculum 12 Software engineering  
in the Speciality: 12 Software engineering

Done by the student of the group PZ2422:



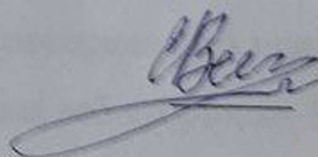
/ Andrii PRYIMYCH /

Scientific Supervisor:



/ Tetiana HRYSHECHKINA /

Normative controller:



/ Svitlana VOLKOVA /

Міністерство освіти і науки України  
Український державний університет науки і технологій

Факультет: «Комп'ютерні технології і системи»  
Кафедра: «Комп'ютерні інформаційні технології»  
Рівень вищої освіти: Магістр  
Освітня програма: «12 Інженерія програмного забезпечення»  
Спеціальність: «12 Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри КІТ

\_\_\_\_\_/Вадим ГОРЯЧКІН/

Дата 14.09.2025

ЗАВДАННЯ

на кваліфікаційну роботу Магістр студенту

Приймичу Андрію Михайловичу

1. Тема роботи: «Вплив стратегій попереднього завантаження ресурсів на швидкість відображення веб-сторінок» Керівник роботи: Тетяна Сергіївна Гришечкіна.
2. Строк подання студентом роботи: 11.01.2026
3. Вихідні дані до роботи:
4. Зміст пояснювальної записки (перелік питань, які потрібно опрацювати):  
вступ, збір вимог до програмного забезпечення, проектування тестування та налагодження, висновки, література.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): презентація, відео-демонстрація роботи програми

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз літературних джерел та предметної області. Постановка задачі.	01.09.2025 – 26.10.2025	
2	Обґрунтування вибору технологій та архітектури стенда.	27.10.2025 – 02.11.2025	
3	Проектування структури тестових веб-сторінок та сценаріїв завантаження.	03.11.2025 – 09.11.2025	
5	Програмна реалізація тестового ПЗ та веб-додатків для дослідження.	10.11.2025 – 14.11.2025	30 %
6	Вибір та обґрунтування метрик ефективності (Web Vitals).	17.11.2025 – 23.11.2025	
7	Фіналізація методики проведення експерименту.	24.11.2025 – 30.11.2025	
8	Написання перших розділів записки та документування методики.	08.12.2025 – 14.12.2025	60%
9	Проведення основного етапу експериментальних досліджень. Збір даних.	15.12.2025 – 21.12.2025	
10	Статистична обробка результатів, побудова графіків та порівняльний аналіз.	22.12.2025 – 28.12.2025	
	Формулювання висновків та рекомендацій.	29.12.2025 – 31.12.2025	
	Остаточне оформлення пояснювальної записки, підготовка презентації.	01.01.2026 – 04.01.2026	
	Подання кваліфікаційної роботи до кафедри	05.01.2026 – 11.01.2026	100%
11	Захист кваліфікаційної роботи на засіданні Екзаменаційної комісії	20.01.2026	

Студент

Андрій ПРИЙМИЧ

Керівник роботи

Т е т я н а Гришечкіна

## РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи магістра виконана на 47 с., 27 рис., 4 табл., 21 джерело.

Метою роботи є підвищення швидкості відображення контенту та покращення користувацького досвіду веб-додатків шляхом дослідження та впровадження оптимальних стратегій попереднього завантаження критичних ресурсів. Об'єктом дослідження є процес рендерингу веб-сторінок у сучасних браузерах. Предметом дослідження виступають методи та механізми оптимізації доставки контенту, зокрема стратегії попереднього завантаження попереднє завантаження, попередня вибірка та ліниве завантаження. Практичне значення одержаних результатів полягає у створенні алгоритму вибору ефективної стратегії завантаження залежно від типу контенту, що дозволяє суттєво зменшити час до першого відмальовування та покращити загальну чутливість інтерфейсу, його привабливість та зручність для користувача.

Робота складається з 8 розділів:

- вступ;
- огляд літератури та теоретичні відомості. Розкриває сучасні виклики у швидкодії веб-інтерфейсів, аналізує вплив швидкості завантаження на користувацький досвід та бізнес-показники, а також доводить необхідність оптимізації процесу рендерингу. Містить огляд принципів роботи браузерів та детальний аналіз технологій Resource Hints, зокрема Preload, Prefetch та Prerender, описуючи механізми їхньої дії та сфери застосування. Має 8 сторінок;
- розробка методології та визначення метрик дослідження. Обґрунтовує вибір ключових метрик ефективності Core Web Vitals та визначає набір інструментів для точного вимірювання швидкості. Має 2 сторінки;
- розробка програмного забезпечення необхідного для дослідження. Описує процес створення набору тестових веб-застосунків з реалізацією різних стратегій оптимізації, а також розробку спеціалізованого програмного забезпечення для автоматизації вимірювань та збору метрик

продуктивності. Має 13 сторінок;

- проведення дослідження та аналіз отриманих результатів. Описує процес виконання тестів продуктивності за затвердженим сценарієм, фіксацію показників при різних умовах мережі та збір статистичних даних для порівняння. Включає обробку зібраних даних, порівняльний аналіз ефективності досліджених стратегій, візуалізацію залежностей та інтерпретацію впливу оптимізації на час відображення сторінок. Має 6 сторінок;
- висновки. Підсумовує результати роботи, оцінює ступінь досягнення мети та надає конкретні рекомендації розробникам щодо вибору оптимальної стратегії завантаження ресурсів. Має 2 сторінки;

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ .....	8
ВСТУП.....	10
1. ОГЛЯД ЛІТЕРАТУРИ ТА ТЕОРЕТИЧНІ ВІДОМОСТІ .....	12
2. РОЗРОБКА МЕТОДОЛОГІЇ ТА ВИЗНАЧЕННЯ МЕТРИК ДОСЛІДЖЕННЯ	20
3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НЕОБХІДНОГО ДЛЯ	
ДОСЛІДЖЕННЯ .....	22
3.1. Опис програмного забезпечення що розробляється.....	22
3.2. Огляд аналогів .....	22
3.3. Опитування можливої аудиторії.....	26
3.4. Проектування ПЗ.....	32
3.5. Вибір мови програмування .....	32
3.6. Технологічна платформа.....	33
3.7. Розробка інтерфейсу користувача .....	33
3.8. Тестування .....	35
4. ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ	
.....	36
4.1. Опис використаного програмно-апаратного середовища.....	36
4.2. Пояснення експерименту .....	36
4.3. Дані, зібрані під час проведення експерименту .....	36
4.4. Аналіз отриманих даних .....	41
ВИСНОВКИ .....	43
ПЕРЕЛІК ЛІТЕРАТУРНИХ ДЖЕРЕЛ .....	45
ДОДАТКИ .....	45

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

API (Application Programming Interface) - інтерфейс прикладного програмування; набір визначень взаємодії між компонентами програмного забезпечення.

CDN (Content Delivery Network) - мережа доставки контенту; географічно розподілена мережа серверів для прискорення доставки контенту користувачам.

CLS (Cumulative Layout Shift) - сукупний зсув макета; метрика Core Web Vitals, що вимірює візуальну стабільність сторінки під час завантаження.

CRP (Critical Rendering Path) - критичний шлях рендерингу; послідовність кроків, які виконує браузер для перетворення HTML, CSS та JavaScript у пікселі на екрані.

CSS (Cascading Style Sheets) - каскадні таблиці стилів; формальна мова опису зовнішнього вигляду документа.

CSSOM (CSS Object Model) - об'єктна модель CSS; карта стилів, пов'язана з DOM, яку браузер використовує для відмальовування сторінки.

DNS (Domain Name System) - система доменних імен; ієрархічна система перетворення доменних імен в IP-адреси.

DOM (Document Object Model) - об'єктна модель документа; програмний інтерфейс для HTML-документів, що представляє сторінку у вигляді дерева об'єктів.

FCP (First Contentful Paint) - перше змістовне відмальовування; метрика, що фіксує час появи на екрані першого контенту (тексту або зображення).

HTML (HyperText Markup Language) - мова розмітки гіпертексту; стандартна мова розмітки для створення веб-сторінок.

HTTP (HyperText Transfer Protocol) - протокол передачі гіпертексту; протокол прикладного рівня для передачі даних у мережі Інтернет (включно з версіями HTTP/1.1, HTTP/2, HTTP/3).

JS (JavaScript) - мова програмування, що використовується для створення інтерактивності на веб-сторінках.

LCP (Largest Contentful Paint) - найбільше змістовне відмальовування; метрика Core Web Vitals, що фіксує час відображення найбільшого видимого елемента контенту.

RTT (Round Trip Time) - час кругового обертання; час, необхідний для відправлення сигналу (пакета даних) і отримання підтвердження про його отримання.

SEO (Search Engine Optimization) - пошукова оптимізація; комплекс заходів для підняття позицій сайту в результатах видачі пошукових систем.

SPA (Single Page Application) - однострінковий веб-застосунок; веб-сайт, який взаємодіє з користувачем шляхом динамічного переписування поточної сторінки, а не завантаження нових сторінок із сервера.

TBT (Total Blocking Time) - загальний час блокування; метрика, що вимірює загальний час, протягом якого головний потік був заблокований, що перешкоджало взаємодії з користувачем.

TLS (Transport Layer Security) - протокол захисту транспортного рівня; криптографічний протокол, що забезпечує захищений обмін даними в мережі.

TTFB (Time To First Byte) - час до отримання першого байта; час, що проходить від моменту відправлення запиту клієнтом до отримання першого байта даних від сервера.

TTI (Time to Interactive) - час до повної інтерактивності; час, необхідний сторінці для того, щоб стати повністю готовою до реакції на дії користувача.

UX (User Experience) - досвід користувача (користувацький досвід); сукупність відчуттів та реакцій користувача, що виникають у процесі або в результаті взаємодії з веб-ресурсом.

## ВСТУП

Мабуть, ні для кого не секрет, що веб-технології пройшли карколомний шлях еволюції та стали невід'ємною частиною нашого життя. Якщо ще десятиліття тому інтернет був переважно набором статичних сторінок, то сьогодні це середовище для складних інтерактивних додатків, де користувачі очікують миттєвого відгуку. Однак за цією зручністю ховається невпинне зростання «ваги» веб-ресурсів.

Згідно з даними HTTP Archive, середній розмір веб-сторінки за останні десять років збільшився більш ніж утричі. Якщо у 2011 році медіанний розмір сторінки становив близько 600 КБ, то у 2024 році цей показник на мобільних пристроях перевищує 2200 КБ, а на десктопах наближається до 2600 КБ. Кількість запитів до сервера (HTTP requests) для відображення однієї сторінки часто перевищує 70–80 одиниць, серед яких левову частку займають важкі JavaScript-скрипти, зображення високої роздільної здатності та сторонні трекери аналітики.

Таке зростання обсягу даних вступає у конфлікт із психологією користувача. Дослідження компанії Google свідчать про те, що 53% мобільних користувачів залишають сайт, якщо він завантажується довше 3 секунд. Ба більше, ймовірність відмови (bounce rate) зростає на 32%, якщо час завантаження збільшується з 1 до 3 секунд, і на катастрофічні 90%, якщо цей час сягає 5 секунд.

Ці технічні затримки мають прямий вплив на економічні показники бізнесу.

- гігант електронної комерції Amazon виявив, що кожні 100 мілісекунд затримки завантаження призводять до втрати 1% прибутку;
- компанія Akamai у своєму звіті зазначила, що затримка всього у 100 мілісекунд може знизити конверсію на 7%;
- медіа-ресурс BBC провів експеримент, який показав, що вони втрачають додатково 10% користувачів за кожну додаткову секунду завантаження сайту;
- дослідження Deloitte підтверджує: покращення швидкості мобільного сайту всього на 0,1 секунди може збільшити конверсію роздрібних продавців на 8,4%.

У сучасних умовах, коли Google впровадив метрики Core Web Vitals як фактор ранжування в пошуковій видачі, швидкість відображення (особливо LCP - Largest Contentful Paint) стала питанням виживання веб-ресурсу на ринку.

Незважаючи на усі плюси швидкісного інтернету (4G/5G), класичні методи оптимізації (стиснення, кешування, мініфікація) вже не є панацеєю. Браузери за замовчуванням працюють лінійно та реактивно: вони виявляють ресурси поступово, розбираючи HTML-код рядок за рядком. Це створює ефект «пляшкового горлечка» (network bottleneck), коли браузер «простоює», не знаючи про критичні ресурси, які знадобляться через кілька мілісекунд. Через це рендеринг сторінки затримується, і користувач бачить порожній екран.

Саме через це процес оптимізації продуктивності ускладнюється і потребує переходу від реактивного завантаження до проактивного. Тут на сцену виходять стратегії попереднього завантаження ресурсів (Resource Preloading). Це методики, що дозволяють «підказати» браузеру, які файли (шрифти, стилі, скрипти) є пріоритетними, і завантажити їх ще до того, як дійде черга обробки.

Ця магістерська робота присвячена аналізу впливу таких стратегій, як ліниве завантаження, попередня вибірка, попереднє завантаження медіа та сучасних механізмів Early Hints, на швидкість відображення веб-сторінок. Робота має на меті не лише дослідити теоретичні аспекти роботи браузерного рендерингу, але й експериментально довести, як правильна оркестрація завантаження ресурсів дозволяє суттєво покращити метрики Core Web Vitals, знизити показник відмов та підвищити загальну ефективність веб-додатків у сучасному перенасиченому інформацією просторі.

## 1 ОГЛЯД ЛІТЕРАТУРИ ТА ТЕОРЕТИЧНІ ВІДОМОСТІ

Швидкість завантаження є критичним фактором фінансової спроможності вебресурсу. Дослідження ринку підтверджують, що навіть мінімальні затримки призводять до відтоку клієнтів. У звіті Akamai зазначено:

«Затримка завантаження сторінки всього на 100 мілісекунд може знизити рівень конверсії на 7%» [3].

Сприйняття швидкості користувачем базується на візуальній стабільності контенту. Експерти Calibre підкреслюють:

«Психологія швидкості вчить нас: те, як користувач відчуває процес завантаження, є важливішим за технічні показники. Пріоритезація видимого контенту створює ілюзію миттєвої роботи» [9].

Тамі Евертс у праці «Time Is Money» наголошує:

«Продуктивність - це не просто функція; це фундаментальна складова користувацького досвіду, яка безпосередньо впливає на лояльність до бренду та дохід» [4].

Основна частина затримок припадає на фронтенд-частину завантаження. Стів Саундерс, засновник сучасних методик вебпродуктивності, стверджує:

«Більша частина часу, який користувач витрачає на очікування сторінки, припадає на завантаження компонентів: зображень, скриптів, стилів. Оптимізація фронтенду дає найбільший приріст швидкості» [2].

З огляду на вивчений матеріал, було здійснено пошук та аналіз актуальних стратегій підвищення швидкодії сайтів, реалізованих за допомогою мови програмування JavaScript та мови розмітки HTML. Було вирішено відкинути фреймворки та їх особисті методи оптимізації так як принципи їх роботи надзвичайно різні, використовують різні надбудови, їх надзвичайно багато і порівняння їх усіх неможливе. Серед джерел було розглянуто публікацію Дрю МакЛілана про Resource Hints, в ній розкажується про попереднє завантаження, попередню вибірку та ліниве завантаження, як про простий метод оптимізації, що може стати гарним фундаментом оптимізації вебсайту:

«Багато оптимізацій продуктивності можна зробити, коли ми можемо передбачити, що можуть робити користувачі, перш ніж вони це зроблять. Попереднє завантаження, попередня вибірка та ліниве завантаження це простий, але ефективний спосіб, за допомогою якого веб-розробники можуть допомогти браузеру бути на крок попереду користувача та підтримувати швидкість завантаження сторінок» [12]

Щодо різних методів оптимізації думки програмістів дещо відрізняються, комусь більше подобається ліниве завантаження, хтось любить `prefetch`, а дехто не признає нічого крім старого доброго попереднього завантаження.

Спеціалісти MDN Web Docs наприклад більше люблять саме ліниве завантаження:

«Ліниве завантаження дозволяє відкласти завантаження некритичних ресурсів (зображень, відео) до моменту, коли вони знадобляться користувачеві, що суттєво прискорює початковий рендеринг. Це найкращий метод оптимізувати ваш вебсайт» [5].

Адді Османі, відомий гуру оптимізації та автор багатьох книжок для початкових програмістів, додає:

«Нативне ліниве завантаження в браузерах дозволяє розробникам досягати високої продуктивності без використання складних JavaScript-бібліотек» [6].

З ним не згоден Віталій Фрідман, журналіст Smashing Magazine а також досвідчений програміст, у своїй статті він розказує про крутість попереднього завантаження:

«Використовуючи `preload` для шрифтів та ключових зображень, ми гарантуємо, що вони будуть готові до моменту, коли браузер почне малювати сторінку, уникаючи затримок, дуже гарний спосіб покращити швидкість вашого сайту» [8].

Його підтримує журналіст Сандер Хейлброн з інтернет видання Iron out:

«Попереднє завантаження, попередня вибірка та попереднє підключення – це важливі інструменти для розробника, які належать до так званих підказок ресурсів. Хоча вони називаються підказками, браузер обробляє попереднє

завантаження як директиву.» [10]

Програмісти Деміан Рензулі та Джеремі Вагнер у своїй статті “Prefetch resources to speed up future navigations” у свою чергу наголошують про те що цю методику також не варто списувати з рахунків:

«Netflix використовує час, який користувачі проводять на сторінках, які не ввійшли в систему, для попереднього завантаження React, який буде використовуватися після входу користувачів у систему. Завдяки цьому їм вдалося скоротити час до інтерактивності на 30% для майбутніх навігацій. Не варто недооцінювати цю методику» [11]

В той же час Метт Цойнерт, творець компанії DebugBear, у статті “Resource Prefetching May Slow Down Your Website” каже що цей метод може лише шкодити, якщо використовувати його неправильно:

«На жаль, переглядаючи каскади запитів, що показують завантаження ресурсів на нашому вебсайті, ми часто виявляли, що запити на попередню вибірку надходили на ранній стадії та конкурували за пропускну здатність з нашим основним зображенням контенту» [13].

Метт вирішив пройтись по кожній стратегії, у своїй статті “Common Problems With Preloading”, він написав про те як попереднє завантаження може лише погіршити оптимізацію:

«Попереднє завантаження може допомогти браузеру визначити пріоритети важливих запитів. Але попереднє завантаження великої кількості файлів може фактично погіршити пріоритезацію та уповільнити завантаження сторінки.» [14].

Наступним етапом було вирішино поглибитись в те як працюють стратегії попереднього завантаження, їх історію додавання і розвиток. Почав я з пошуків в технічній документації, щоб зрозуміти принцип їх роботи, знайшов що директива Preload ініціює запити до того, як браузер виявить ресурс у документі. У специфікації W3C зазначено:

«Resource Hints (Preload, Prefetch) дають можливість браузеру заздалегідь підготувати з'єднання або завантажити ресурси, що критично важливо для швидкого відображення першого екрана» [7].

Також була розглянута стаття на порталі для розробників Mozilla, в ній пояснено як працює ліниве завантаження:

«ліниве завантаження – це стратегія визначення ресурсів як неблокуючих (некритичних) та їх завантаження лише за потреби. Це спосіб скоротити довжину критичного шляху рендерингу, що призводить до зменшення часу завантаження сторінки» [15].

У іншій статті вони ж додають:

«Суть лінивого завантаження – дробління ресурсів та завантаження їх по черзі, за необхідності, тобто завантажуються лише ресурси що знаходяться в полі зору користувача та потрібні у цей момент» [16].

Щодо попередньої вибірки інформації було дещо менше, але я натрапив на статтю від Девіда Вольша, там він розказує про механізм роботи цієї стратегії:

«Попереднє завантаження посилань – це механізм браузера, який використовує час простою браузера для завантаження або попереднього завантаження документів, які користувач може відвідати найближчим часом. Веб-сторінка надає браузеру набір підказок щодо попереднього завантаження, і після того, як браузер завершить завантаження сторінки, він починає непомітно попереднє завантаження певних документів і зберігає їх у своєму кеші. Коли користувач відвідує один із попередньо завантажених документів, його можна швидко видалити з кешу браузера» [17].

Адді Османі додає:

«Завантаження веб-сторінок можна пришвидшити, якщо заздалегідь отримати наступну сторінку (або набір ресурсів для наступної сторінки), яку користувач, ймовірно, відвідає. Ми називаємо це попередньою вибіркою. У деяких випадках це може призвести до того, що наступні навігації завантажуються миттєво» [18].

Дізнавшись про стратегії і не знайшовши відповідь на моє питання що краще, я почав шукати про те як аналізується швидкість сайту, які метрики існують, як їх вимірюють і тд. Одразу знайшов статтю від українського хостинг провайдера City Host, яка розказувала про велику кількість існуючих метрик, в ній

вони розказують про підхід різних компаній до оцінки швидкості вебсайту, особливо цікаво було прочитати про підхід Google:

«Core Web Vitals — це три основні показники, які Google виділяє як найважливіші. До нього відноситься рендеринг більшої частини контенту, час першої інтерактивності та совокупне розміщення контенту (LCP, FID, CLS)» [19]. Очікувалось що інші компанії мають схожий набір метрик, але це виявилось не так, спеціалісти Mozilla наприклад покладаються на інші метрики:

«TTFB, LCP та CLS є одними з найважливіших показників швидкодії вебсайту, особливо важливо щоб ці показники завжди залишались у межах норми» [20].

Для детального вивчення термінології було використано матеріали ресурсу CityHost, де наведено роз'яснення основних понять та принципів їх роботи:

«TTFB (Time To First Byte) — проміжок часу від моменту надсилання запиту на хостинг, на якому розміщений сайт, до отримання першого байта відповіді. Між цими двома фазами відбувається чимало подій — перенаправлення, пошук DNS, узгодження TLS та інші. Усі ці процеси стисло називають «обробкою запиту».

FCP (First Contentful Paint) — час до моменту, коли користувач бачить на сайті перший контент; початковий, частковий рендеринг. Ця метрика не враховує завантаження iframe й передбачає текст (у тому числі той, для якого ще не завантажилися шрифти), зображення або елементи canvas (полотно, яке виглядає як плашка не білого кольору). Оптимальний час — до 1,8 секунди.

LCP (Largest Contentful Paint) — це час, коли більша частина контенту у видимій області екрана вже доступна для перегляду. Для розрахунку метрики LCP використовується час рендерингу найбільшого елемента (текстового блоку або зображення) від початку завантаження сайту.

FID (First Input Delay) — час від першої взаємодії користувача з інтерактивними елементами (натискання кнопки, введення тексту в поле, перехід за посиланням) до моменту, коли браузер починає на них реагувати. Час очікування відповіді також є важливою метрикою для репутації сайту, адже

користувачам не подобається, коли вони натискають кнопку, а нічого не відбувається.

TTI (Time To Interactive) — буквально перекладається як «час до інтерактивності», коли основний контент уже завантажився і користувач не лише бачить інтерактивні елементи, а й може повноцінно з ними взаємодіяти.

INP (Interaction to Next Paint) — показник швидкості відгуку сторінки на дії користувача, які відбуваються вже після повного завантаження сайту. Це проміжок часу між кліком по елементу та реакцією на нього. Оптимальною вважається швидкість реакції до 200 мілісекунд.

TTV (Total Blocking Time) — загальний час блокування, який триває від початку рендерингу контенту (FCP) до моменту інтерактивності сайту (TTI). У цей період користувач уже бачить вміст, але не може натискати кнопки та інші активні елементи.

CLS (Cumulative Layout Shift) — зміщення макета через підвантаження важчих елементів. Уявімо, що користувач відкрив сайт, побачив потрібне посилання й натиснув на нього, але в цей момент вище розгорнувся блок (відео або зображення), який довго завантажувався.

Speed Index — показує швидкість відображення контенту в процесі завантаження. Наприклад, основний контент на сайті завантажився за 4 секунди. Але в одному випадку протягом перших трьох секунд користувач бачить лише білий екран, а в другому вже з першої секунди починається рендеринг дрібних деталей, заголовків, блоків» [19].

Далі я поглибився в тему браузерного рендерингу та принципу роботи браузеру. Все починається з запиту інформації з серверу, нижче наведено етапи отримання інформації з серверу. Процес починається з можливого перенаправлення запиту, після чого, за наявності Service Worker, відбувається його ініціалізація та обробка fetch-події.

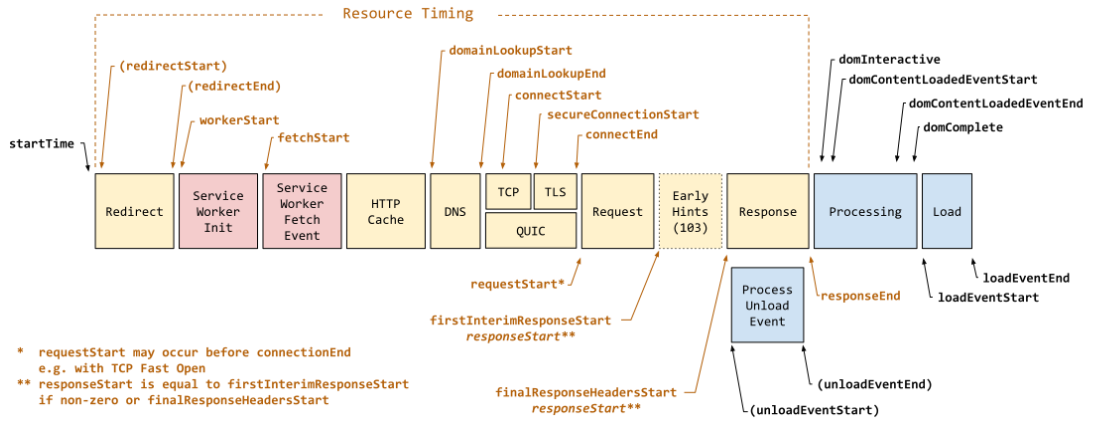


Рисунок 1.1 – Схема запиту інформації браузером

Процес починається з можливого перенаправлення запиту, після чого, за наявності Service Worker, відбувається його ініціалізація та обробка fetch-події. Далі браузер перевіряє HTTP-кеш і, якщо ресурс не знайдено, виконує DNS-запит для визначення IP-адреси домену. Після цього встановлюється мережеве з'єднання за допомогою TCP або QUIC, а у випадку HTTPS додатково виконується TLS-рукоштовання. Саме на цьому етапі вимірюється TTFB, це метрика що відображає проміжок часу від моменту надсилання запиту на хостинг, на якому розміщений сайт, до отримання першого байта відповіді. На перший погляд ця метрика залежить лише від швидкості інтернет з'єднання але це не так, у програмістів існує безліч методів покращення цієї метрики, основними є використання стратегій попереднього завантаження ресурсів, пріорітезація, передпідключення DNS, зміна CDN серверу. TTFB є одною із ключових метрик швидкодії сайту так як вона напряду впливає на усі інші, поокрачивши цю метрику покращаться і інші, тому що вона відповідає за ранній етап рендерингу, запит необхідної інформації з серверу. За документацією Google та Mozilla гарними вважаються значення до 800 мс, посередніми до1800 мс. Все що вище, дуже поганий результат.

Після того як браузер отримав відповідь починається рендеринг сторінки. Спочатку HTML парситься у DOM, паралельно з цим завантажуються таблиці стилів(CSS) та скрипти. CSS будує CSSOM і блокує відображення доки не буде готовий, тоді як JavaScript може зупиняти парсинг HTML, якщо він не асинхронний або не відкладений. На основі DOM і CSSOM формується дерево

рендерингу, після чого браузер виконує розрахунок стилів і розмітки сторінки. Далі відбувається перше візуальне відображення контенту на екрані, і в цей момент фіксується метрика FCP, яка показує, коли користувач уперше бачить будь-який текст або графіку. Після цього браузер продовжує завантажувати ресурси, виконувати скрипти та домальовувати сторінку. Коли найбільший видимий елемент, наприклад головне зображення або великий блок тексту, повністю відрендерений, фіксується метрика LCP. У процесі подальшого завантаження та виконання JavaScript можуть відбуватися зміни макета через пізні завантаження зображень або вставку контенту. Сукупність таких несподіваних зсувів фіксується метрикою CLS, яка відображає візуальну стабільність сторінки з точки зору користувача.

## 2 РОЗРОБКА МЕТОДОЛОГІЇ ТА ВИЗНАЧЕННЯ МЕТРИК ДОСЛІДЖЕННЯ

Для підрахунку швидкодії вебсайтів та дослідження впливу стратегій попереднього завантаження на швидкість відображення веб сторінки було обрано TTFB, FCP, LCP, CLS. За допомогою цього набору метрик можливо покрити усі аспекти завантаження вебсайту, від запиту на сервер до відображення найбільшого елементу. TTFB покаже час від запиту до початку отримання даних, FCP розкаже про те коли користувач побачить перший елемент сайту, LCP про те коли користувач побачить найбільший елемент сторінки а CLS надасть дані про стабільність макета.

Дослідження має проводитись автоматизовано та у декілька етапів, потрібно дослідити сайт без оптимізації, ті, на яких використовується один з трьох методів оптимізації та усі три методи разом. Вебсайти мають бути розміщені на одному сервері для нівелювання затримки. Також буде налаштовано свій DNS сервер, для того щоб нівелювати навантаження на сторонні DNS сервери

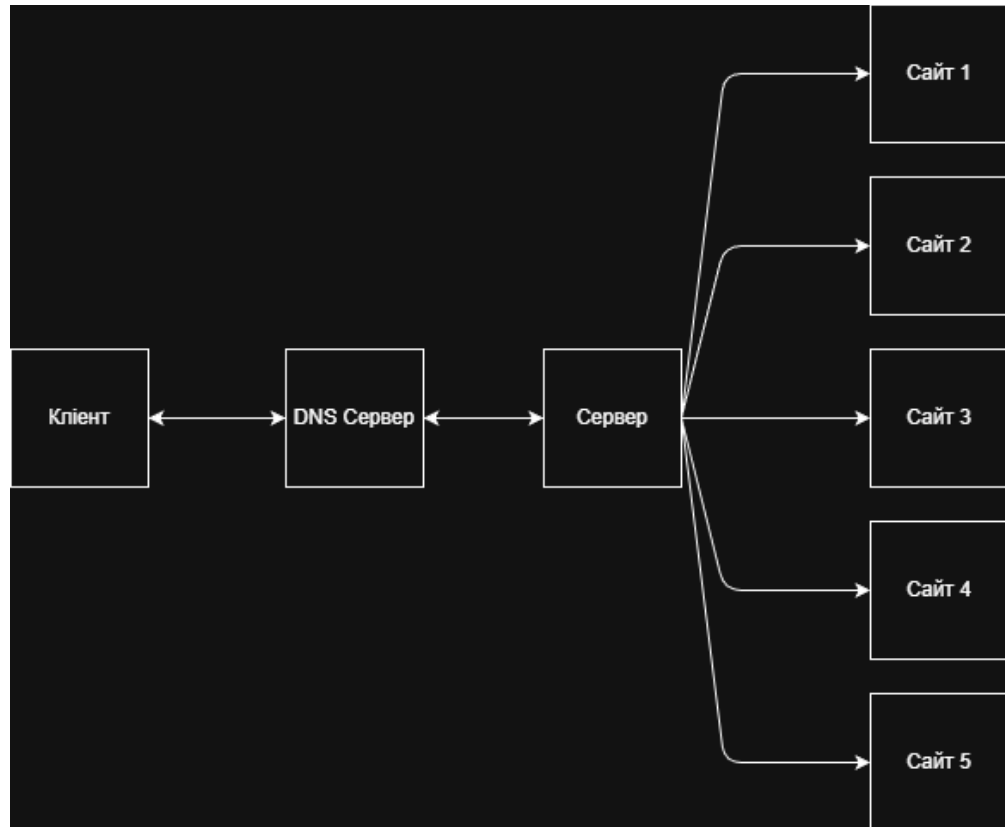


Рисунок 2.1 – Схема побудови архітектури дослідження

Так як інтернет підключення може бути не стабільним, кожен дослід буде повторюватись тричі. Тестові сайти будуть навантажені зображеннями великого розміру, великі таблиці стилів CSS, великі шрифтові файли. Для тестування буде розроблене спеціальне ПЗ, що буде виконувати аудит за допомогою відкриття headless браузеру( такого що не має інтерфейсу користувача) та подальшого збору метрик.

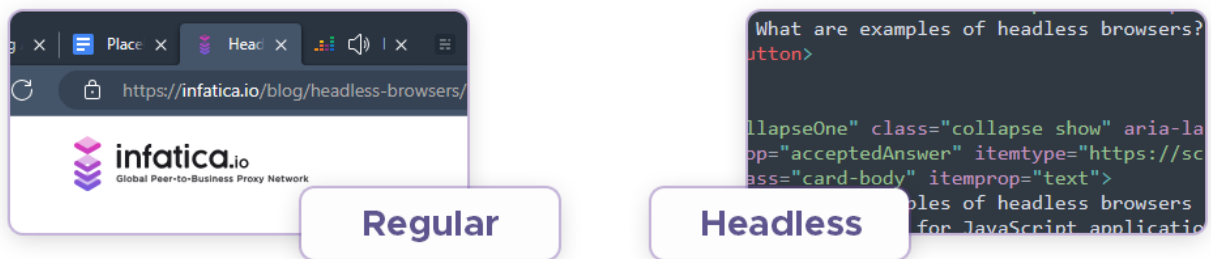


Рисунок 2.2 – Відмінність між звичайним та headless браузером

За допомогою відстеження часу запиту і відповіді, часу отримання необхідних для сайту файлів можна заміряти обрані метрики швидкодії. Таким чином буде проведено дослідження.

## 3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НЕОБХІДНОГО ДЛЯ ДОСЛІДЖЕННЯ

### 3.1 Опис програмного забезпечення що розробляється

Для проведення практичного дослідження стратегій оптимізації розробляється спеціалізоване програмне забезпечення — застосунок-аудитор. В основу роботи програми покладено принцип автоматизованого тестування продуктивності вебсторінок у середовищі безінтерфейсного браузера (*headless browser*). Такий підхід дозволяє імітувати реальну поведінку користувача та отримувати об'єктивні дані без впливу графічного інтерфейсу операційної системи на результати замірів.

Алгоритм роботи застосунку передбачає послідовне виконання таких етапів:

1. ініціалізація сесії: запуск екземпляра браузера та налаштування параметрів мережі та емуляції пристрою;
2. завантаження ресурсу: перехід за заданою URL-адресою та очікування повної побудови об'єктної моделі документа (DOM);
3. збір метрик: фіксація технічних.

Особлива увага при розробці приділяється автоматизації вимірювання чотирьох ключових характеристик швидкодії:

- час до отримання першого байта (TTFB): для оцінки затримки серверної частини та мережевого з'єднання;
- перша відмальовка контенту (FCP): як показник швидкості відображення перших візуальних елементів на екрані;
- найбільша відмальовка контенту (LCP): для визначення часу завантаження основного змісту сторінки;
- загальний час блокування (TVT): що характеризує інтерактивність та готовність сторінки до взаємодії з користувачем.

### 3.2 Огляд аналогів

#### 3.2.1 Огляд AMPLitude

AMPLitude – комплексне B2B рішення для аналізу вебсайтів, що включає в себе замір швидкодії сайту, аналіз сайту за допомогою III агентів та перевірка сайту з

точки зору маркетингу. Головним плюсом є те що застосунок пропонує All-in-one рішення, заплатив за один сервіс і отримуєш багато продуктів. Головним недоліком є вартість, продукт досить дорогий, навіть для B2B клієнтів а також малий набір функцій щодо заміру швидкодії сайту, сервіс не видає числові характеристики а підраховує overall рейтинг швидкодії та показує його користувачу. Такий підхід не дозволяє нормально аналізувати проблеми сайту і не дає зрозуміти етап на якому виникають проблеми.

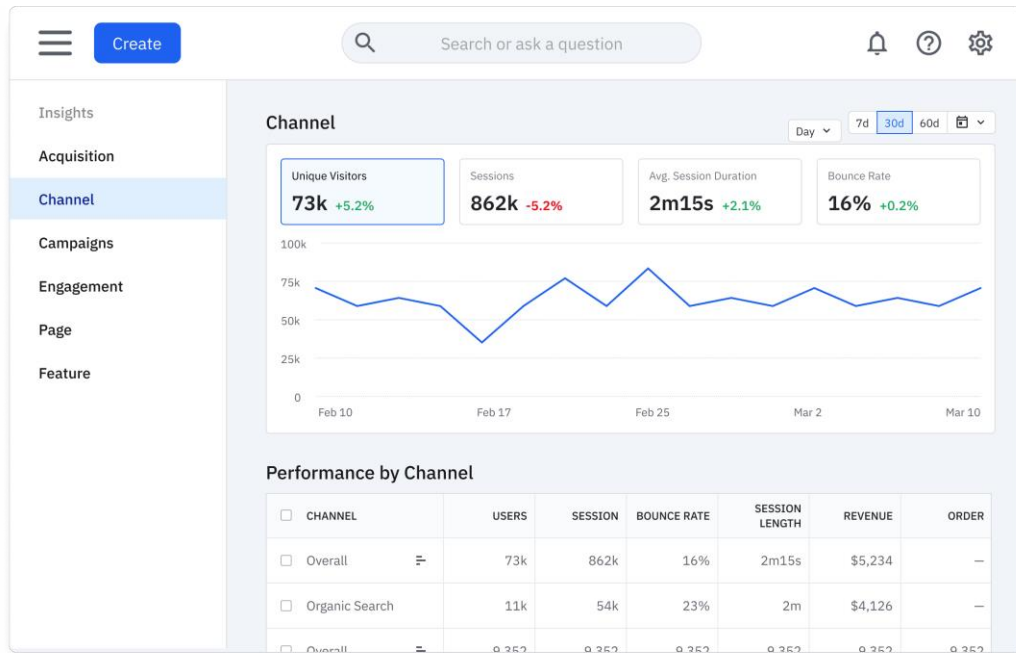


Рисунок 3.1 – Інтерфейс AMPLitude

### 3.2.2 Огляд Google PageSpeedInsights

Це вже спеціалізований інструмент від Google, що дає змогу оцінити ключові часові характеристики, має простий інтерфейс, має гарну візуалізацію цих метрик.

Отчет от 6 янв. 2026 г., 18:52:03

https://developers.google.com/

Анализировать

Мобильные устройства Компьютер

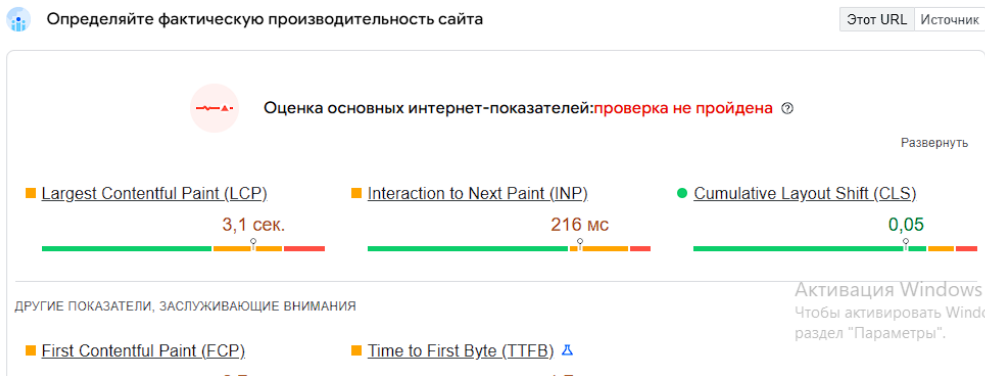


Рисунок 3.2 – Интерфейс PageSpeedInsights

Водночас цей проект має декілька особливостей та недоліків, які не дозволяють використовувати цей програмний продукт у дослідженні. Основними недоліками є часті помилки, інструмент від гугл постійно видає помилки замість аналізу веб сторінки, неможливо отримати чіткі часові характеристики, сервіс бере такі від усіх користувачів та показує середнє значення за останні 28 днів, причому якщо сайт не популярний то отримати метрики взагалі неможливо.

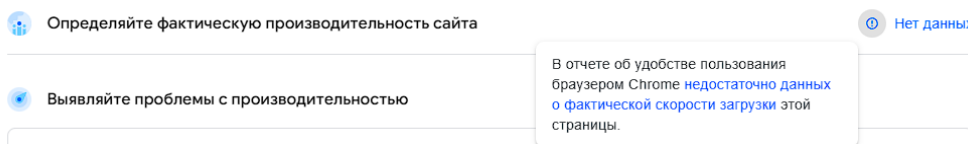


Рисунок 3.3 – Помилка недостатньої кількості даних через низьку кількість відвідувань

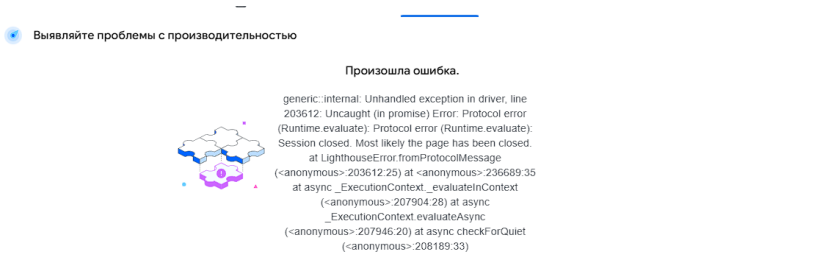


Рисунок 3.4 – Помилка проведення аудиту PageSpeedInsights

Загалом видно що проект являється безоплатним та некомерційним.

### 3.2.3 Огляд pingdom

B2B рішення що має безкоштовний тестовий період для аналізу вебсайту. Застосунок має гарний інтерфейс, можливість проведення тесту з декількох серверів. Також при проведенні аудиту сервіс аналізує код та пропонує поради для покращення.

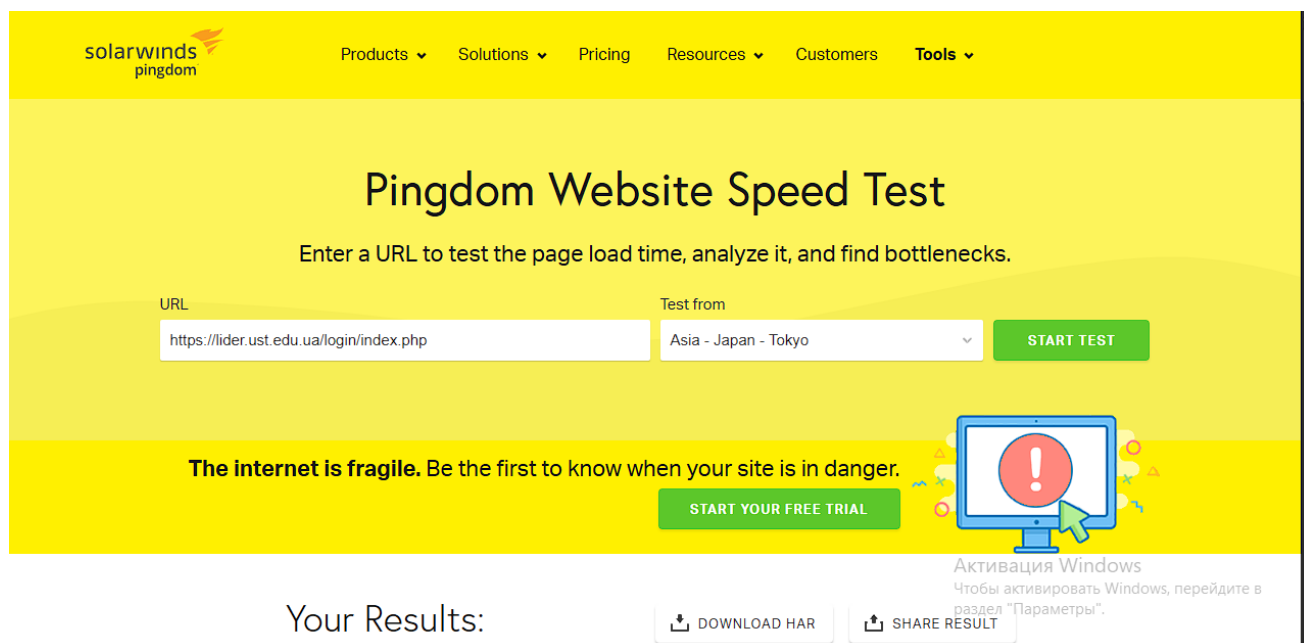


Рисунок 3.5 – Головна сторінка Pingdom

При всій зрозумілості інтерфейсу, сервіс не видає конкретних характеристик а лише формує так званий Performance grade, що не дає змоги нормально оцінити швидкодію та зрозуміти де саме проблем.

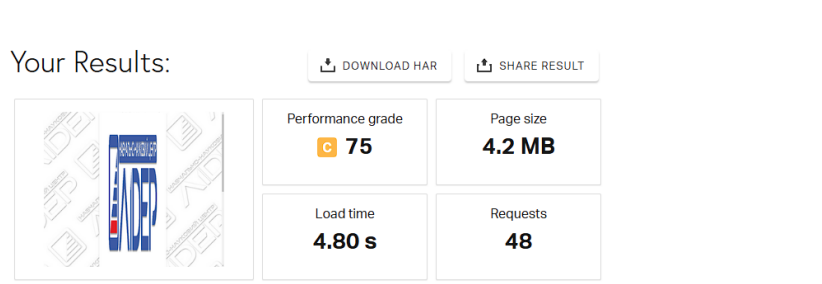


Рисунок 3.6 – Звіт Pingdom

### 3.2.4 Огляд SEOptimer

Комплексне рішення що має дуже великий набір інструментів для оптимізації сайтів, в тому числі при аудиті має секцію швидкодії, заміряє PLC, TTFB та PSC, при цьому ігнорує інші важливі метрики і не показує які саме

файли найдовше завантажувались.

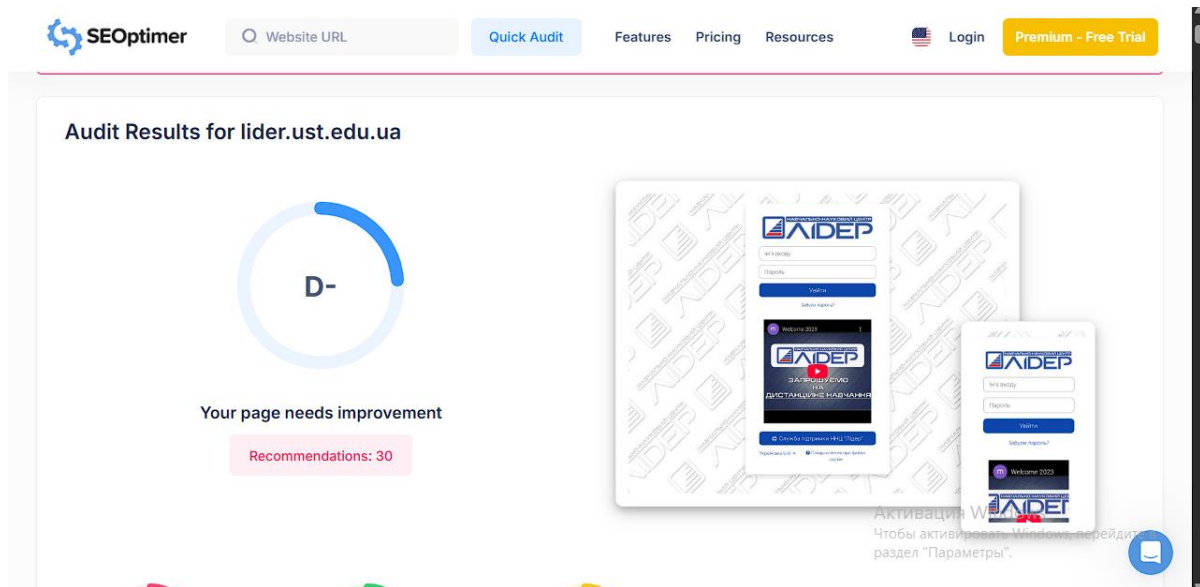


Рисунок 3.7 – Головний екран SEOptimer

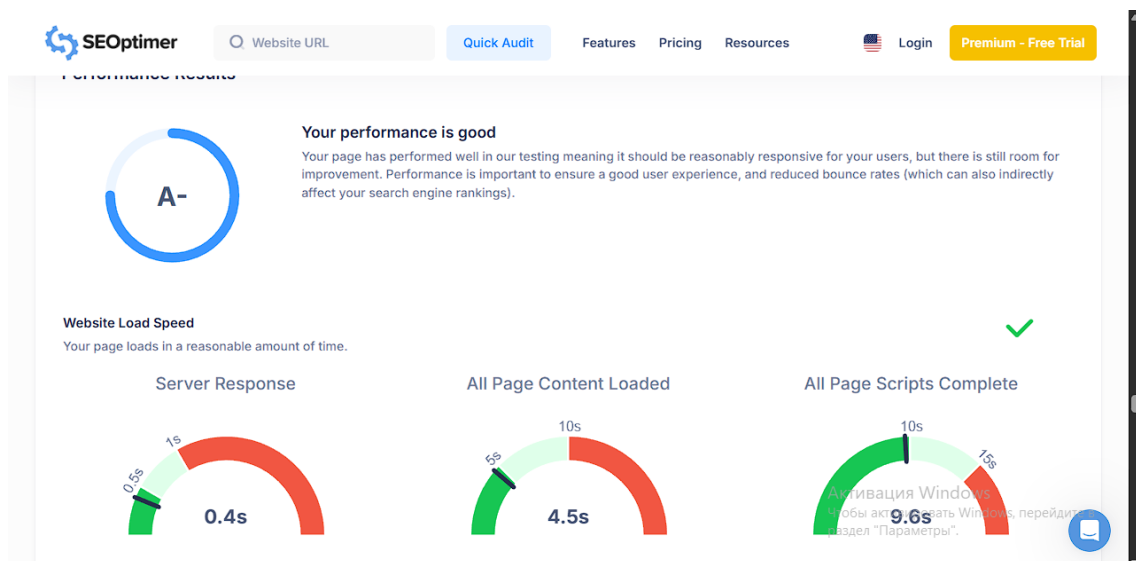


Рисунок 3.8 – Звіт SEOptimer

### 3.3 Опитування можливої аудиторії

Опитування проводилось серед студентів напрямлення програмна інженерія та розробників різного ступеня компетенції. Для опитування використовувалась платформа Google Forms через свою простоту та доступність. Всього було 7 питань з варіантами відповіді. Питання з відкритою відповіддю не використовувались через важкість обробки таких даних. В опитуванні взяло участь 31 людина. Перелік питань та отримані дані наведено нижче.

1. Яка ваша роль?

- Frontend-розробник;
  - Backend-розробник;
  - Project / Product Manager;
  - Студент;
  - Інше.
2. Ваш рівень технічної підготовки?
- Початковий;
  - Середній;
  - Просунутий.
3. Чи доводиться вам тестувати швидкодію сайту?
- Так, часто;
  - Так, рідко;
  - Ні.
4. Які інструменти для перевірки швидкості ви використовуєте зараз?
- PageSpeed Insights;
  - Lighthouse;
  - SEOptimer;
  - Інше.
5. Які метрики швидкості ви використовуєте для прийняття рішень частіше?
- LCP (Largest Contentful Paint);
  - INP (Interaction to Next Paint);
  - CLS (Cumulative Layout Shift);
  - FCP (First Contentful Paint);
  - TTFB (Time to First Byte);
  - Speed Index;
  - Загальний Performance Score;
  - Інше.
6. Чи розумієте ви значення всіх цих метрик?
- Так, повністю;
  - Частково;

- Ні.
7. Чи є якісь критичні для вас недоліки у інструменті що ви використовуєте?
- Ні, він ідеальний;
  - Незрозумілий або/та перевантажений інтерфейс користувача;
  - Надто багато технічних термінів;
  - Незрозуміло, з чого почати оптимізацію;
  - Важко оцінити вплив змін;
  - Інше.

### Яка ваша роль?

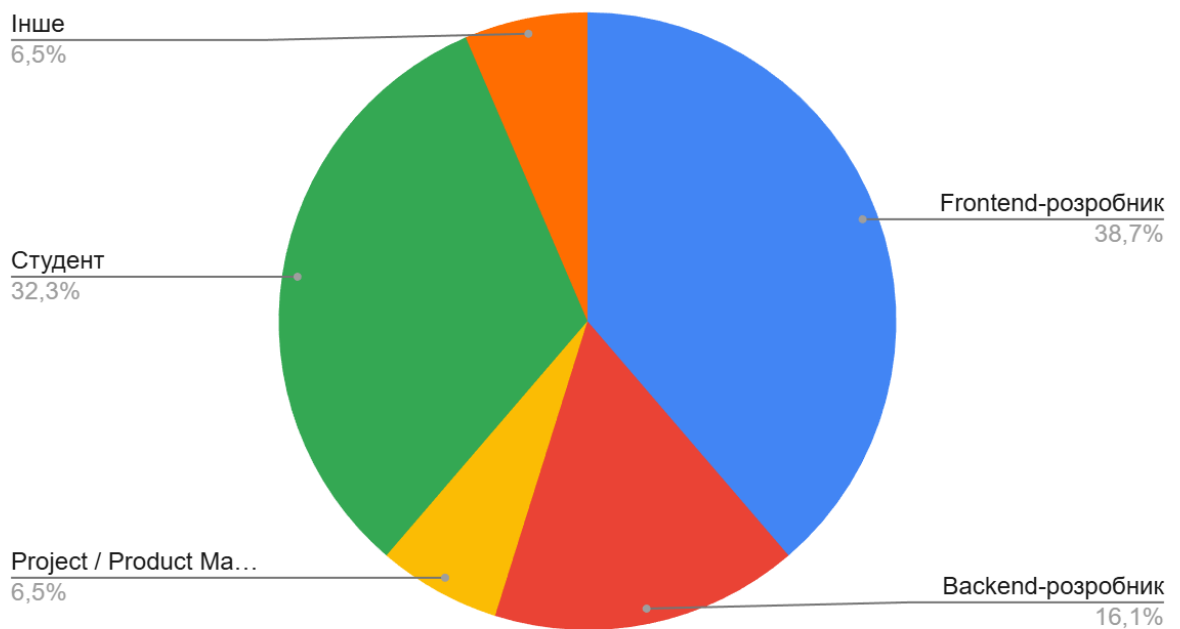


Рисунок 3.9 – Структура відповідей на запитання 1

Ваш рівень технічної підготовки?

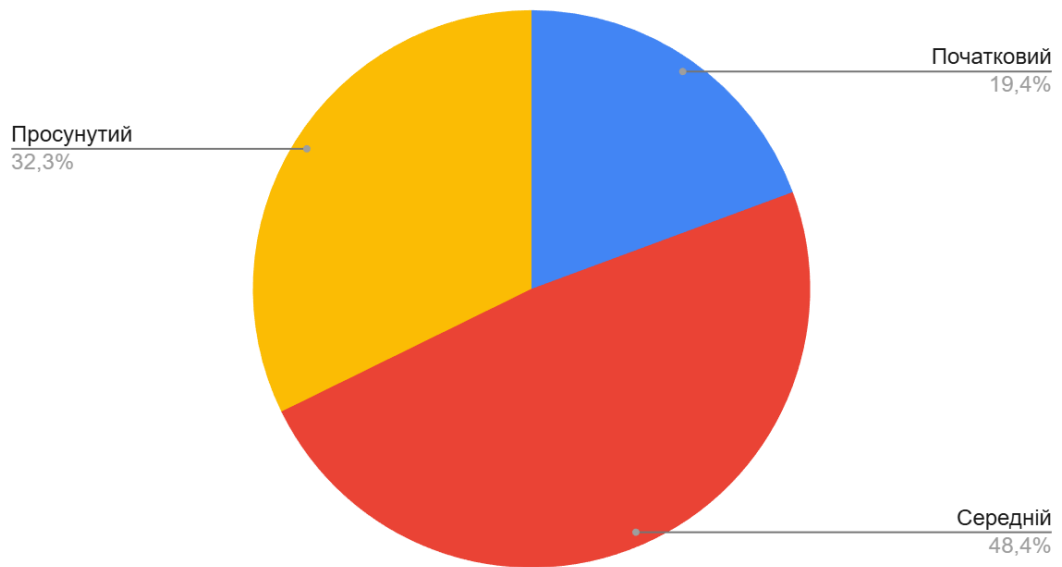


Рисунок 3.10 – Структура відповідей на запитання 2

Чи доводиться вам тестувати швидкодію сайту?

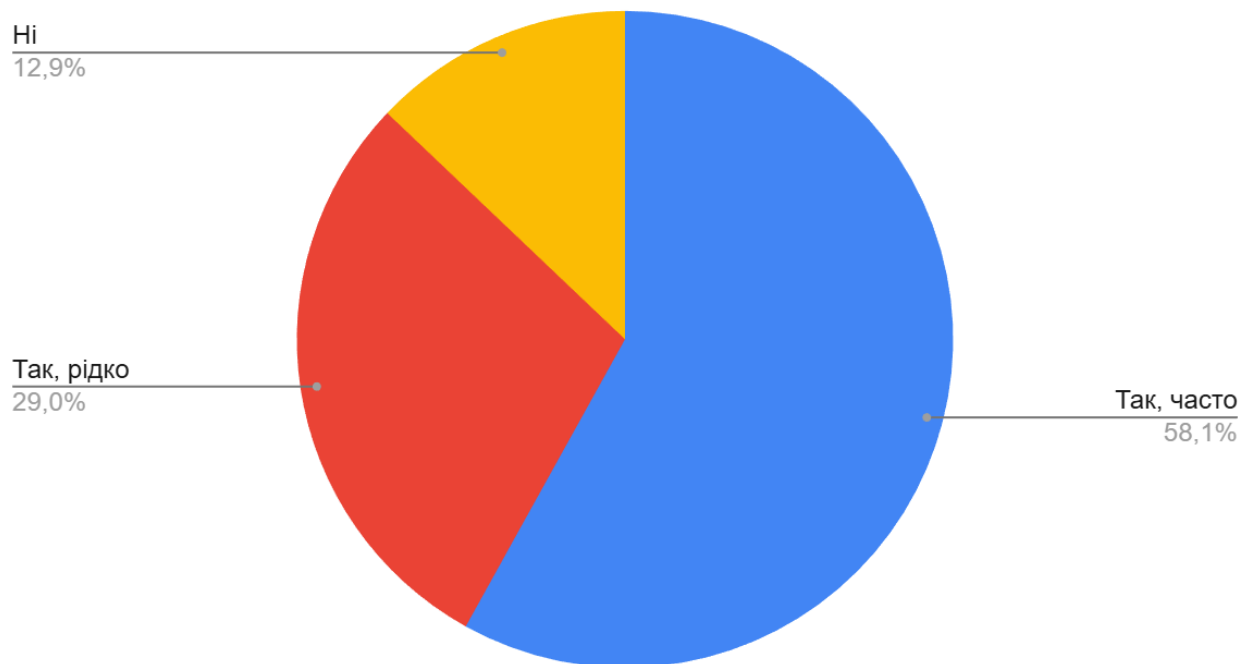


Рисунок 3.11 – Структура відповідей на запитання 3

Які інструменти для перевірки швидкості ви використовуєте зараз?

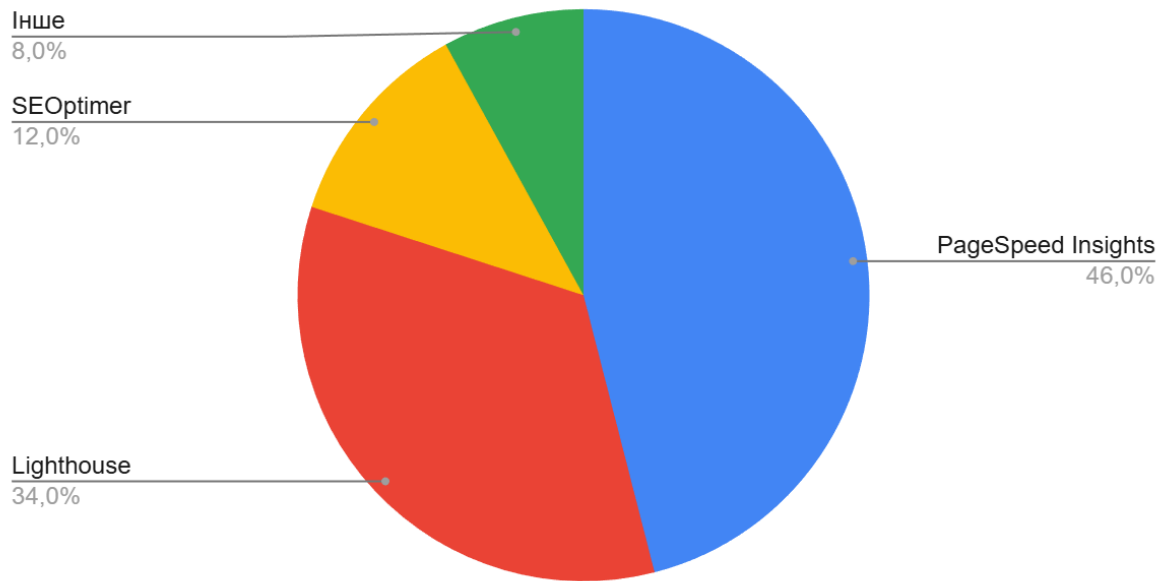


Рисунок 3.12 – Структура відповідей на запитання 4

Які метрики швидкості ви використовуєте для прийняття рішень частіше?

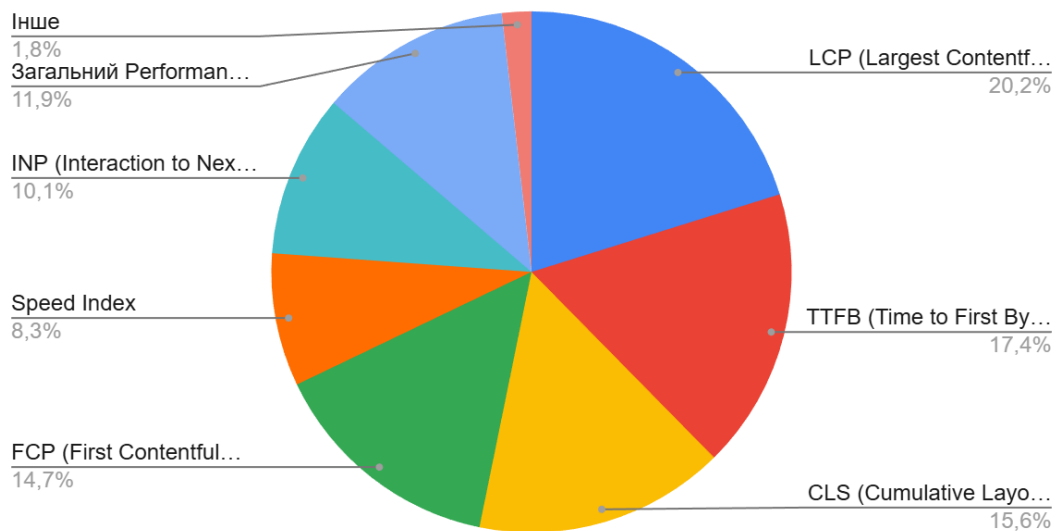


Рисунок 3.13 – Структура відповідей на запитання 5

Чи розумієте ви значення всіх цих метрик?

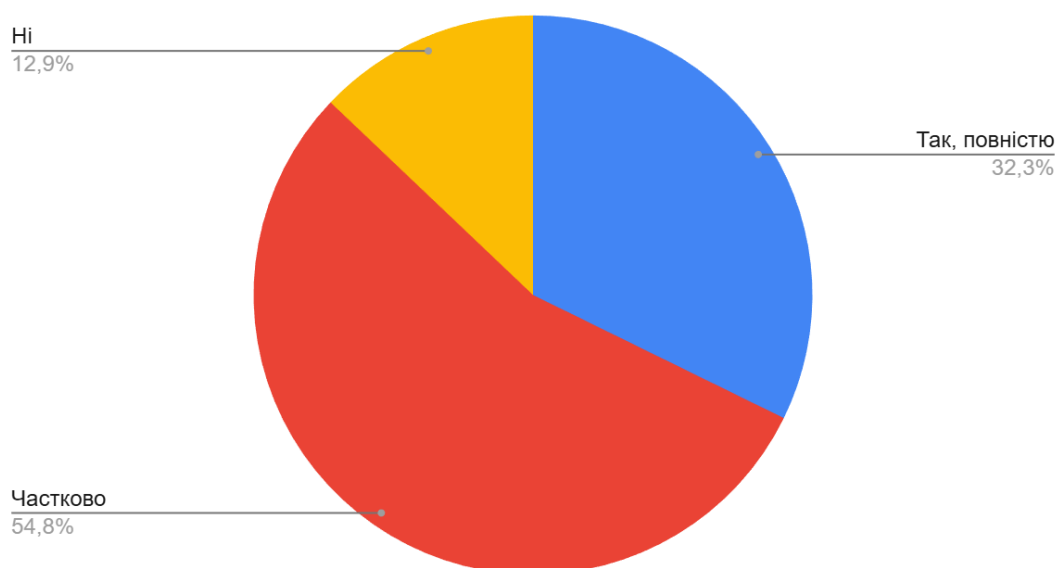


Рисунок 3.14 – Структура відповідей на запитання 6

Чи є критичні недоліки в інструменті, який ви використовуєте?

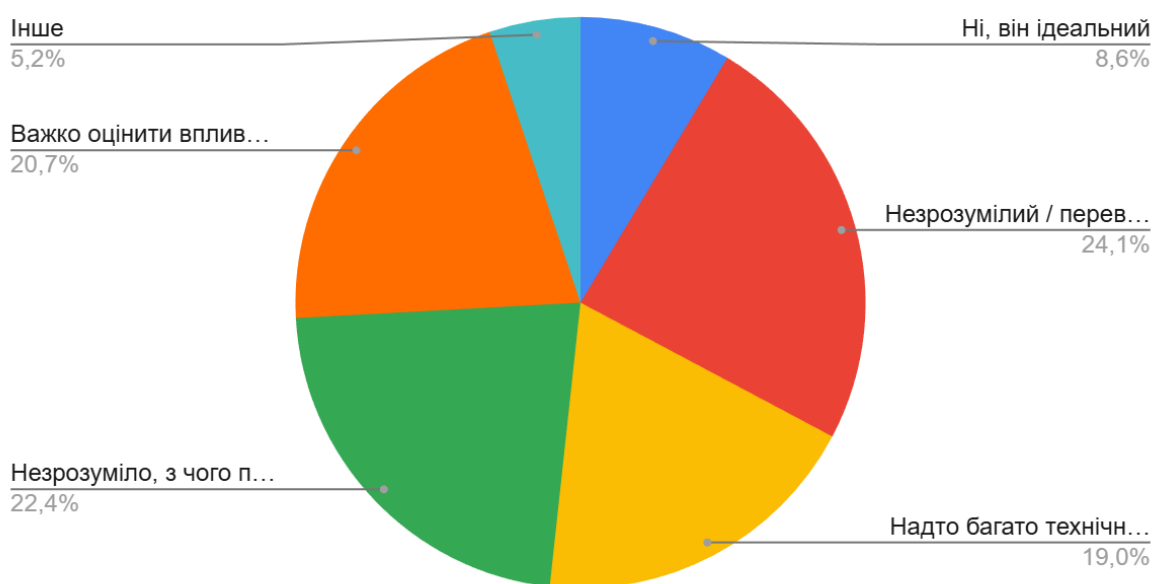


Рисунок 3.15 – Структура відповідей на запитання 7

Результати опитування показали те що можливі користувачі опираються в основному на метрики TTFB, LCP, CLS, FCP. Також користувачі цінують простий та зручний інтерфейс, який поєднує функціональність. Застосунок має підкреслювати зони для оптимізації.

### 3.4 Проектування ПЗ

За архітектуру було вирішено брати класичний клієнт-серверний застосунок. Така архітектура дозволяє досить просто масштабувати продукт, додавати в нього нові функції в майбутньому. Проект не матиме своєї бази даних і буде кешувати дані у користувача. Таке рішення було прийняте для того щоб не підпадати під законодавство по збереженню та обробці даних користувача GDPR. Нижче наведено схему застосунку:

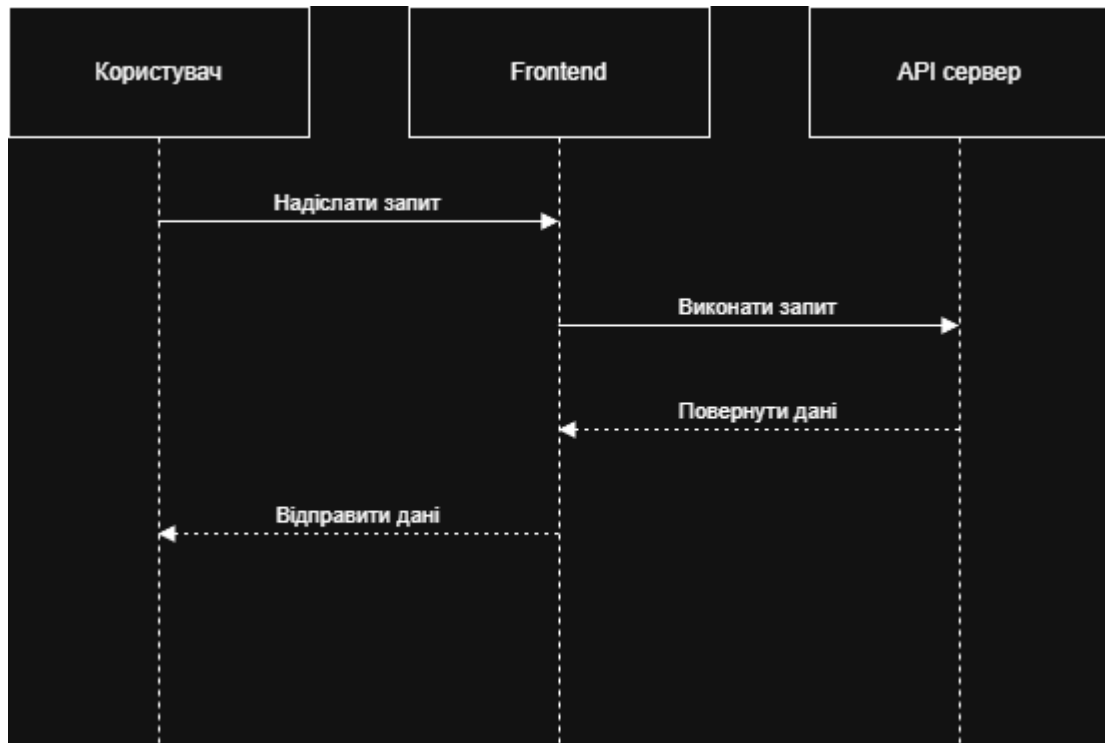


Рисунок 3.15 – Архітектура аудитора

### 3.5 Вибір мови програмування

Основою ПЗ було обрано Javascript, і бекенд і фронтенд частини застосунку будуть написані на ньому а саме на TypeScript. Для бекенду буде використано Node.js в обгортці з Express.js, це найпопулярніша зв'язка яка дає змогу написати якісний бекенд API сервер, підключити необхідні API та headless браузер. Фронтенд буде написано за допомогою Bootstrap, це популярна бібліотека стилів, що є дуже оптимізованою, відповідає усім сучасним стандартам стилю та є легко інтегрованою. Також використано Vite у ролі складальника сайту. У якості headless браузера для збору даних було обрано Playwright, це сучасний, ефективний, зручний та безкоштовний headless браузер на базі Chromium. Його

було обрано через простоту використання, підтримку великого набору функцій, гарного API для розробки, що сильно полегшило інтеграцію.

### 3.6 Технологічна платформа

Для розробки було використано середовище розробки VS Code, воно нативно підтримує усі технології що використовуються в аудиторі, має систему надзвичайно зручних плагінів та тем, вказує на помилки. Для тестування використовувався Postman.

«Postman - це платформа для розробки та тестування API, яка дозволяє створювати та ділитися колекціями, автоматизувати тести та контролювати продуктивність інтерфейсів прикладного програмування.» [21].

Це лідер ринку, з великим набором інструментів тестування аудитора.

### 3.7 Розробка інтерфейсу користувача

Для frontend частини застосунку було створено простий та мінімалістичний UI, що поєднує в собі простоту та інформативність. Для розробки дизайну було використано Figma, в якому за допомогою каталогу елементів Bootstrap було створено наступний UI. Нижче наведений перший екран та екран звіту.

#### Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

Рисунок 3.17 – Перший екран аудитора

## Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

Аудит Помилка: HTTP 500

Не вдалося виконати аудит. Перевір URL або спробуй ще раз.

Рисунок 3.18 – Головний екран аудитора(помилка проведення аудиту, посилання недоступне)

## Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

Аудит Готово

Потрібні суттєві покращення продуктивності.

### Ключові метрики

Добре
Слід покращити
Погано

Метрика	Значення	Оцінка	Пояснення
TTFB	113 ms	Добре	Час до першого байта від сервера (responseStart – requestStart).
FCP	3812 ms	Погано	Перший видимий контент на екрані.
LCP	3812 ms	Слід покращити	Найбільший елемент стає видимим (герой, великий текст).
CLS	0.000	Добре	Стабільність макета (накопичене зміщення).

▶ Що означають ці метрики?

\* Обчислено в headless Chromium через Performance API та спостерігачі.

### Ресурси (топ за тривалістю)

Тип	URL
css	...e.php/1/theme_moove/loginbgimg/17661
iframe	https://www.youtube.com/embed/zwZtvWJ
script	...s://lider.ust.edu.ua/lib/requirejs.php/1766
script	...s://lider.ust.edu.ua/lib/requirejs.php/1766
script	...der.ust.edu.ua/lib/requirejs.php/17661280
img	...php/1/theme_moove/logo/1766128032/Li
script	...du.ua/theme/yui_combo.php?rollup/3.17.2
link	https://lider.ust.edu.ua/theme/styles.php/m
script	...edu.ua/lib/javascript.php/1766128032/lib
script	...u.ua/lib/javascript.php/1766128032/lib/rev
css	...ont.php/moove/core/1766128032/fontaw
script	...t.edu.ua/lib/javascript.php/1766128032/liit
xmlhttprequest	...map%22%2C%22args%22%3A%7B%22then
link	...ly=Raleway:ital,wght@0,300;0,400;0,500;0,
xmlhttprequest	...C%22themenamename%22%3A%22moove%22%
xmlhttprequest	.../service.php?sesskey=DRL1GaogRI&info=
xmlhttprequest	...component%22%3A%22core_form%22%2
script	...ua/lib/javascript.php/1766128032/lib/jque
link	...u.ua/theme/yui_combo.php?rollup/3.17.2/
script	...in.js&m/1766128032/filter_glossary/autoli
script	...ange/event-valuechange-min.js&3.17.2/ev
link	...t.edu.ua/theme/yui_combo.php?3.17.2/css

Натисни на заголовок стовпця для сортування.

Рисунок 3.19 – Екран звіту

### 3.8. Тестування

Тестування складалось з декількох етапів, спочатку виконувалось ручне тестування запитів, відбувалась перевірка чи усі дані запитує headless браузер та як він це робить, я посилав запити до відомих вебресурсів за допомогою Postman, а також мого API і звіряв набір отриманих даних. Також я перевіряв точність підрахунку метрик, я вираховував показники вручну (за допомогою дерева запитів) та звіряв їх з розробленою програмою. Відхилення вийшло мінімальним (меншим за загальноприйняті 5%). Тести я повторював 3 рази для того, щоб мінімізувати вплив випадкових обставин чи помилок розрахунку.

Таблиця 3.1 – Похибка вимірювання метрик

Метрика	Показник вручну	Показник програми	Відхилення
LCP (Largest Contentful Paint)	1.2 сек	1.224 сек	+2%
TTFB (Time to First Byte)	0.15 сек	0.1485 сек	-1%
CLS (Cumulative Layout Shift)	0.002	0.002	0%

Тестування підтвердило точність виконання API запитів а також те що похибка знаходиться в межах норми(<5%).

## 4 ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

### 4.1 Опис використаного програмно-апаратного середовища

Для дослідження у роботі було застосовано комплексний підхід до тестування, що включає три основні компоненти. Першим елементом виступає розроблений застосунок-аудитор, який в автоматичному режимі фіксує метрики продуктивності в середовищі безінтерфейсного браузера. Другим компонентом є п'ять тестових сайтів: еталонна версія без оптимізації, варіанти з відкладеним завантаженням, попереднім завантаженням і вибіркою ресурсів, а також сайт, що поєднує всі ці стратегії. Замикає програмний комплекс платформа Postman, яка використовується для контрольної верифікації мережових запитів та аналізу серверних відповідей, що гарантує точність отриманих статистичних результатів.

### 4.2 Пояснення експерименту

Усі тестові сайти мали в коді підвантаження великих таблиць стилів CSS, скрипт файли, великий набір зображень гарної якості. Всі 5 сайтів реалізовували різний підхід о оптимізації веб застосунків. Коротко про кожен:

- відсутність оптимізації;
- ліниве завантаження;
- попереднє завантаження;
- попередня вибірка;
- усі стратегії разом.

Усі вебсайти були розміщені на одному сервері, для точності розрахунків та нівелювання показника затримки. Кожне тестування було проведено 3 рази, в таблиці результатів наведено середні значення. Наведено скріншоти перших звітів.

### 4.3 Дані, зібрані під час проведення експерименту

Таблиця 4.1 – Дані отримані під час дослідження

№	Сайт без оптимізації	Lazy loading	Попереднє завантаження	Попередня вибірка	Усі стратегії
TTFB	3.2 мс	3.8 мс	2.6 мс	3.0 мс	20.8

Таблиця 4.1 – Дані отримані під час дослідження (продовження)

FCP	3797.2 мс	277 мс	542.4 мс	205.0 мс	198.0
LCP	3822.8 мс	3170 мс	557.8 мс	206.0 мс	241.4
CLS	0.0002 мс	0.0008 мс	0.0006 мс	0.0006 мс	0.006 мс

## Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

Аудит

Готово

Потрібні суттєві покращення продуктивності.

### Ключові метрики

Добре Слід покращити Погано

Метрика	Значення	Оцінка	Пояснення
<b>TTFB</b>	2 ms	Добре	Час до першого байта від сервера (responseStart – requestStart).
<b>FCP</b>	3791 ms	Погано	Перший видимий контент на екрані.
<b>LCP</b>	3812 ms	Слід покращити	Найбільший елемент стає видимим (герой, великий текст).
<b>CLS</b>	0.000	Добре	Стабільність макета (накопичене зміщення).

► Що означають ці метрики?

\* Обчислено в headless Chromium через Performance API та спостерігачі.

### Ресурси (топ за тривалістю)

Усі типи ▾

Тип	URL
-----	-----

Рисунок 4.1 – Результат 1 звіту (без оптимізації)

## Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

https://relifemc.com/lazy-loading

Аудит

Готово

Є простір для оптимізації відмальовування.

### Ключові метрики

Добре

Слід покращити

Погано

Метрика	Значення	Оцінка	Пояснення
TTFB	4 ms	Добре	Час до першого байта від сервера (responseStart – requestStart).
FCP	272 ms	Добре	Перший видимий контент на екрані.
LCP	3185 ms	Слід покращити	Найбільший елемент стає видимим (герой, великий текст).
CLS	0.001	Добре	Стабільність макета (накопичене зміщення).

► Що означають ці метрики?

\* Обчислено в headless Chromium через Performance API та спостерігачі.

### Ресурси (топ за тривалістю)

Усі типи



Тип	URL	Тривалість	Розмір
img	https://picsum.photos/1000/1000?random=6	3185 ms	21367 B
img	https://picsum.photos/1000/1000?random=7	405 ms	1967 B
img	https://picsum.photos/1200/1200?random=2	337 ms	1836 B
img	https://picsum.photos/300/300?random=4	337 ms	1893 B
img	https://picsum.photos/300/300?random=3	336 ms	480 B
img	https://picsum.photos/300/300?random=5	336 ms	1397 B
img	https://picsum.photos/300/300?random=1	318 ms	459 B
img	https://picsum.photos/300/300?random=0	318 ms	944 B

Натисни на заголовок стовпця для сортування.

Рисунок 4.2 – Результат 2 звіту (ліниве завантаження)

## Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

https://relifemc.com/preloading

Аудит

Готово

Сторінка загалом швидко відмальовується.

### Ключові метрики

Добре

Слід покращити

Погано

Метрика	Значення	Оцінка	Пояснення
---------	----------	--------	-----------

<b>TTFB</b>	2 ms	Добре	Час до першого байта від сервера (responseStart – requestStart).
-------------	------	-------	--

<b>FCP</b>	532 ms	Добре	Перший видимий контент на екрані.
------------	--------	-------	-----------------------------------

<b>LCP</b>	544 ms	Добре	Найбільший елемент стає видимим (герой, великий текст).
------------	--------	-------	---

<b>CLS</b>	0.001	Добре	Стабільність макета (накопичене зміщення).
------------	-------	-------	--

Рисунок 4.3 – Результат 3 звіту (попереднє завантаження)

## Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

https://relifemc.com/prefetching

Аудит

Готово

Сторінка загалом швидко відмальовується.

### Ключові метрики

Добре

Слід покращити

Погано

Метрика	Значення	Оцінка	Пояснення
<b>TTFB</b>	3 ms	Добре	Час до першого байта від сервера (responseStart – requestStart).
<b>FCP</b>	200 ms	Добре	Перший видимий контент на екрані.
<b>LCP</b>	200 ms	Добре	Найбільший елемент стає видимим (герой, великий текст).
<b>CLS</b>	0.001	Добре	Стабільність макета (накопичене зміщення).

► Що означають ці метрики?

Рисунок 4.4 – Результат 4 звіту(попередня вибірка)

## Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

Аудит

Готово

Сторінка загалом швидко відмалюється.

**Ключові метрики** 
Добре
Слід покращити
Погано

Метрика	Значення	Оцінка	Пояснення
<b>TTFB</b>	19 ms	Добре	Час до першого байта від сервера (responseStart – requestStart).
<b>FCP</b>	190 ms	Добре	Перший видимий контент на екрані.
<b>LCP</b>	237 ms	Добре	Найбільший елемент стає видимим (герой, великий текст).
<b>CLS</b>	0.006	Добре	Стабільність макета (накопичене зміщення).

▶ Що означають ці метрики?

\* Обчислено в headless Chromium через Performance API та спостерігачі.

Рисунок 4.5 – Результат 5 звіту(повна оптимізація)

### 4.4 Аналіз отриманих даних

Аналіз отриманих даних підтверджує, що розумне використання методів попереднього завантаження - це не просто технічне налаштування, а фундамент комфорту користувача. Скорочення середнього часу очікування у 18 разів кардинально змінює сприйняття сайту: замість роздратування від очікування людина отримує миттєвий відгук. У цифровому світі швидкість наряду корелює з довірою до бренду. Коли сторінки відкриваються швидко, користувач підсвідомо сприймає ресурс як надійний та професійний. Для бізнесу це має прямий економічний ефект, оскільки кожна зекономлена секунда зменшує ризик того, що відвідувач закриє вкладку, так і не здійснивши покупку чи реєстрацію.

Особливу роль у дослідженні відіграє відкладене завантаження (lazy loading). Хоча технічні заміри першої відмалювки (FCP) показують невелику

затримку, з точки зору реального досвіду користувача це є виправданим кроком. Система свідомо «притримує» важкі елементи, які наразі не потрібні, звільняючи ресурси браузера для обробки того, що знаходиться безпосередньо перед очима людини. Це робить старт роботи із сайтом легшим та плавнішим, дозволяючи користувачеві почати взаємодію з контентом, не чекаючи завантаження всієї сторінки цілком.

Найбільшу стабільність та найкращі враження забезпечує поєднання всіх стратегій одночасно. Такий підхід дозволяє згладити навантаження на пристрій та мережу, роблячи роботу сайту передбачуваною навіть при нестабільному з'єднанні. Для власника продукту впровадження такого комплексного рішення означає створення сучасного та конкурентоспроможного сервісу, який однаково добре працює для кожного клієнта. В кінцевому підсумку, така турбота про продуктивність перетворюється на лояльність аудиторії та стабільне зростання бізнес-показників.

## ВИСНОВКИ

У межах дипломної роботи проведено всебічне дослідження стратегій оптимізації вебзастосунків, що розпочалося з ґрунтовного аналізу друкованих джерел та наукових публікацій для формування теоретичної бази. На етапі проектування було враховано результати опитування зацікавлених сторін та проведено критичний огляд існуючих аналогів, що дозволило чітко визначити функціональні вимоги до розроблюваного спеціалізованого програмного забезпечення. Практична частина роботи полягала у створенні авторського застосунку-аудитора та п'яти тестових сайтів, на яких було апробовано різні моделі завантаження ресурсів для збору об'єктивних метрик швидкодії.

Результати експерименту, зафіксовані в ході дослідження, підтвердили надзвичайну ефективність превентивних методів. Зокрема, використання стратегій попередньої вибірки та завантаження дозволило скоротити час першої відмальовки контенту (FCP) з базових 3797.2 мс до мінімальних 198.0 мс у комбінованому сценарії, що свідчить про прискорення роботи майже у 19 разів. Хоча механізм лінивого завантаження очікувано продемонстрував вищий показник LCP (3170 мс) через специфіку відтермінування обробки об'єктів, він забезпечив швидку появу первинних елементів на рівні 277 мс. Це підтверджує, що грамотна пріоритезація критичного шляху рендерингу є вирішальним фактором для створення якісного користувацького досвіду.

Підсумковий аналіз проведеного дослідження дозволяє стверджувати, що найвищий рівень стабільності та продуктивності системи досягається саме за умови глибокої та одночасної інтеграції всіх розглянутих стратегій. Такий комплексний підхід дозволяє нівелювати індивідуальні недоліки окремих методів, створюючи синергетичний ефект, за якого технічні метрики сайту залишаються стабільно високими незалежно від складності контенту чи якості мережевого з'єднання користувача. Для сучасного бізнесу впровадження подібного технічного стека є стратегічно важливим кроком, адже воно прямо конвертується у зростання лояльності аудиторії, підвищення глибини переглядів та суттєве покращення показників конверсії завдяки відсутності технічних бар'єрів при

взаємодії з інтерфейсом. Розроблене в межах диплома програмне забезпечення разом із отриманими аналітичними даними слугують надійною методичною базою, яку можна використовувати для аудиту та масштабування реальних високонавантажених вебпроектів, забезпечуючи їм конкурентну перевагу на ринку.

## ПЕРЕЛІК ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Горячкін, В. М. Навчальний посібник для підготовки кваліфікаційної роботи на здобуття ОС Бакалавр [Текст] / В. М. Горячкін, О. В. Горбова, О. С. Куроп'ятник. – Дніпро : Український державний університет науки і технологій, 2022. – 136 - 137с;
2. Souders, S. High Performance Web Sites [Текст] : монографія / S. Souders. – USA : O'Reilly Media, 2007. – 170 с.
3. The State of Online Retail Performance [Електронний ресурс] / Akamai Report. – 2017. – Режим доступу [URL]<https://www.akamai.com/blog/news/akamai-releases-spring-2017-state-of-online-retail-performance-report>. – Назва з екрана. – Дата звернення: 4.01.2026.
4. Everts, T. Time Is Money: The Business Value of Web Performance [Текст] / T. Everts. – USA : O'Reilly Media, 2016. – 102 с.
5. Lazy loading [Електронний ресурс] / MDN Web Docs. – 2023. – Режим доступу [URL][https://developer.mozilla.org/en-US/docs/Web/Performance/Lazy\\_loading](https://developer.mozilla.org/en-US/docs/Web/Performance/Lazy_loading). – Назва з екрана. – Дата звернення: 4.01.2026.
6. Osmani, A. Native lazy-loading for the web [Електронний ресурс] / Addy Osmani // web.dev. – 2019. – Режим доступу [URL]<https://web.dev/native-lazy-loading/>. – Назва з екрана. – Дата звернення: 4.01.2026.
7. Resource Hints [Електронний ресурс] / W3C Working Draft. – 2022. – Режим доступу [URL]<https://www.w3.org/TR/resource-hints/>. – Назва з екрана. – Дата звернення: 4.01.2026.
8. Friedman, V. Everything You Ever Wanted To Know About Preloading [Електронний ресурс] / Vitaly Friedman // Smashing Magazine. – 2021. – Режим доступу [URL] <https://www.smashingmagazine.com/2016/02/preload-what-is-it-good-for/>. – Назва з екрана. – Дата звернення: 4.01.2026.
9. The Science of Visual Perception and Web Performance [Електронний ресурс] / Calibre App Blog. – 2021. – Режим доступу [URL] <https://calibreapp.com/blog/perceived-performance>. – Назва з екрана. – Дата звернення: 4.01.2026.

10. Heilbron, S. Accelerate your website's loading speed: why using a Link Preload Tag is crucial [Электронный ресурс] / Sander Heilbron // IronOut Blog. – 2023. – Режим доступа : URL : <https://www.iron-out.io/blog/accelerate-your-websites-loading-speed-why-using-a-link-preload-tag-is-crucial/>. – Назва з екрана. – Дата звернення: 4.01.2026.
11. Renzulli, D. Prefetch resources to speed up future navigations [Электронный ресурс] / Demián Renzulli, Jeremy Wagner // web.dev. – 2025. – Режим доступа : URL : <https://web.dev/articles/link-prefetch?hl=en>. – Назва з екрана. – Дата звернення: 4.01.2026.
12. McLellan, D. Optimizing Performance With Resource Hints [Электронный ресурс] / Drew McLellan // Smashing Magazine. – 2025. – Режим доступа : URL : <https://www.smashingmagazine.com/2019/04/optimization-performance-resource-hints/>. – Назва з екрана. – Дата звернення: 4.01.2026.
13. Zeunert, M. Resource Prefetching May Slow Down Your Website [Электронный ресурс] / Matt Zeunert // DebugBear Blog. – 2025. – Режим доступа : URL : <https://www.debugbear.com/blog/prefetch-slower-website>. – Назва з екрана. – Дата звернення: 4.01.2026.
14. Zeunert, M. Common Problems With rel="preload" [Электронный ресурс] / Matt Zeunert // DebugBear Blog. – 2023. – Режим доступа : URL : <https://www.debugbear.com/blog/rel-preload-problems>. – Назва з екрана. – Дата звернення: 4.01.2026.
15. Lazy loading [Электронный ресурс] / Mozilla Developer Docs. – 2026. – Режим доступа : URL : [https://developer.mozilla.org/en-US/docs/Web/Performance/Guides/Lazy\\_loading](https://developer.mozilla.org/en-US/docs/Web/Performance/Guides/Lazy_loading). – Назва з екрана. – Дата звернення: 4.01.2026.
16. Critical rendering path [Электронный ресурс] / Mozilla Developer Docs. – 2026. – Режим доступа : URL : [https://developer.mozilla.org/en-US/docs/Web/Performance/Guides/Critical\\_rendering\\_path](https://developer.mozilla.org/en-US/docs/Web/Performance/Guides/Critical_rendering_path). – Назва з екрана. – Дата звернення: 4.01.2026.

17. Walsh, D. HTML5 Link Prefetching [Електронний ресурс] / David Walsh. – 2010. – Режим доступу : URL : <https://davidwalsh.name/html5-prefetch>. – Назва з екрана. – Дата звернення: 4.01.2026.
18. Osmani, A. Speed up next-page navigations with prefetching [Електронний ресурс] / Addy Osmani. – Режим доступу : URL : <https://addyosmani.com/blog/prefetching/>. – Назва з екрана. – Дата звернення: 4.01.2026.
19. Основні метрики швидкості завантаження сайту: TTFB, FCP, LCP та інші [Електронний ресурс] / Cityhost Blog. – 2022. – Режим доступу : URL : <https://cityhost.ua/blog/osnovnye-metriki-skorosti-zagruzki-sayta-ttfb-fcp-lcp-i-drugie.html>. – Назва з екрана. – Дата звернення: 6.01.2026.
20. LargestContentfulPaint [Електронний ресурс] / Mozilla Developer Docs. – 2026. – Режим доступу : URL : <https://developer.mozilla.org/en-US/docs/Web/API/LargestContentfulPaint>. – Назва з екрана. – Дата звернення: 6.01.2026.
21. What is Postman? [Електронний ресурс] / OpenAPI Blog. – 2026. – Режим доступу : URL : <https://openapi.com/blog/postman-what-it-is-and-how-it-works>. – Назва з екрана. – Дата звернення: 6.01.20

44165850.1551

Додаток А

Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЗАТВЕРДЖУЮ

Перший проректор Українського

державного університету

науки і технологій

Анатолій РАДКЕВИЧ

Web Performance Auditor

Технічне завдання

ЛИСТ ЗАТВЕРДЖЕННЯ

44165850.1551-01-ЛЗ

Завідувач кафедри КІТ

Вадим ГОРЯЧКІН

Керівник розробки

Тетяна Гришечкіна

Виконавець

Андрій ПРИЙМИЧ

Нормоконтролер

Світлана ВОЛКОВА

44165850.1551

ЗАТВЕРДЖЕНО

Додаток А

44165850.1551-01-ЛЗ

Web Performance Auditor

Технічне завдання

44165850.1551-01

Листів 14

2025

44165850.1551

## ЗМІСТ

1. ВВЕДЕННЯ.....	<b>Error! Bookmark not defined.</b>
2. ПІДСТАВИ ДЛЯ РОЗРОБКИ .....	<b>Error! Bookmark not defined.</b>
3. ПРИЗНАЧЕННЯ РОЗРОБКИ.....	<b>Error! Bookmark not defined.</b>
4. ВИМОГИ ДО ПРОГРАМНОГО ПРОДУКТУ .....	7
4.1. Вимоги до функціональних характеристик .....	7
4.2. Вимоги до надійності .....	8
4.3. Умови експлуатації.....	<b>Error! Bookmark not defined.</b>
4.4. Вимоги до складу і параметрів технічних засобів .....	9
4.5. Вимоги до інформаційної і програмної сумісності..	<b>Error! Bookmark not defined.</b>
4.6. Вимоги до маркування і упаковки .....	<b>Error! Bookmark not defined.</b>
4.7. Вимоги до транспортування і зберігання ...	<b>Error! Bookmark not defined.</b>
5. ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ .....	<b>Error! Bookmark not defined.</b>
6. СТАДІЇ ТА ЕТАПИ РОЗРОБКИ.....	<b>Error! Bookmark not defined.</b>
7. ПОРЯДОК І КОНТРОЛЬ ПРИЙМАННЯ .....	<b>Error! Bookmark not defined.</b>
8. БІБЛІОГРАФІЧНИЙ СПИСОК .....	<b>Error! Bookmark not defined.</b>

44165850.1551

## ВВЕДЕННЯ

У сучасному цифровому просторі швидкість та якість взаємодії користувача з веб-ресурсами є критичними факторами успіху будь-якого онлайн-проекту. Зростання складності веб-технологій, зокрема поширення односторінкових додатків (SPA), висуває нові вимоги до аналізу ефективності їх роботи. У цьому контексті актуальним завданням стає не лише моніторинг базової доступності, але й глибоке дослідження стратегій оптимізації завантаження ресурсів та рендерингу контенту.

Даний звіт присвячений опису програмного продукту "Web Performance Auditor" - інструменту, розробленого для комплексного аудиту продуктивності веб-сторінок та веб-додатків, необхідного для виконання дослідження.

Назва програми: Web Performance Auditor

Область застосування: Аналіз продуктивності веб-сторінок, дослідження стратегій оптимізації завантаження ресурсів для таких категорій користувачів: Дослідники веб-технологій, Веб-розробники, Студенти і викладачі технічних спеціальностей

Об'єкт використання: Веб-сторінки, веб-додатки, SPA (Single Page Applications)

Ключові слова і терміни: Веб-метрики, продуктивність, TTFB, FCP, LCP, CLS, попереднє завантаження, оптимізація, аудит, рендеринг, ресурси, швидкість завантаження.

44165850.1551

## 1 ПІДСТАВИ ДЛЯ РОЗРОБКИ

Підставою для розробки є наказ від 02.10.25 №1401 ст ректора Українського державного університету науки і технологій "Про призначення наукових керівників та затвердження тем магістерських робіт" за спеціальністю 121 "Інженерія програмного забезпечення" факультету "Комп'ютерних технологій і систем" по кафедрі "Комп'ютерні інформаційні технології".

Тема дипломної роботи - "РОЗРОБКА ПЗ ДЛЯ ДОСЛІДЖЕННЯ ВПЛИВУ СТРАТЕГІЙ ПОПЕРЕДНЬОГО ЗАВАНТАЖЕННЯ НА ШВИДКІСТЬ ВІДОБРАЖЕННЯ ВЕБ-СТОРИНОК". Керівник - Гришечкіна Тетяна Сергіївна, доцент

44165850.1551

## 2 ПРИЗНАЧЕННЯ РОЗРОБКИ

Функціональне призначення:

Web Performance Auditor розробляється з метою забезпечення дослідників та розробників потужним набором функцій для аналізу та оптимізації продуктивності веб-сторінок:

- збір ключових метрик продуктивності (TTFB, FCP, LCP, CLS) для точного вимірювання швидкості завантаження та відображення веб-сторінок;
- детальний аналіз ресурсів, що дозволяє ідентифікувати файли, які найбільше впливають на час завантаження сторінки (JavaScript, CSS, зображення, шрифти);
- оцінка якості веб-сторінок за стандартами Google Web Vitals.

Експлуатаційне призначення:

Web Performance Auditor покликаний вирішити проблему складних аудиторів ефективності оптимізації для рядових користувачів або junior спеціалістів. Він заміряє ключові метрики продуктивності, не потребує великого набору знань та показує чіткий простір для оптимізації, демонструючи які елементи сайту сповільнюють його роботу та завантаження. Наявний набір фільтрів дозволяє легко орієнтуватись по даним та відстежувати прогрес по оптимізації тих чи інших файлів.

## 3 ВИМОГИ ДО ПРОГРАМНОГО ПРОДУКТУ

### 3.1 Вимоги до функціональних характеристик

Програма повинна виконувати наступні функції:

- збір ключових веб-метрик продуктивності: TTFB (Time to First Byte), FCP (First Contentful Paint), LCP (Largest Contentful Paint), CLS (Cumulative Layout Shift) для заданих веб-сторінок;
- відображення оцінки отриманих метрик із поясненнями для користувача згідно зі стандартами Google Web Vitals;
- автоматичний аналіз переліку ресурсів сторінки та визначення тих, що найбільше впливають на час завантаження;
- можливість фільтрації та сортування ресурсів за типом та тривалістю завантаження.

Організація вхідних та вихідних даних:

- вхідні дані: URL веб-сторінки для аналізу, що вводиться користувачем через веб-інтерфейс;
- вихідні дані: таблиці з метриками продуктивності, таблиці з переліком ресурсів, оцінки якості ("Добре", "Слід покращити", "Погано"), візуальні індикатори стану.

Часові характеристики:

- час виконання аудиту однієї веб-сторінки не повинен перевищувати 60 секунд;
- затримка відображення результатів після завершення аналізу не повинна перевищувати 2 секунди;
- система повинна забезпечувати відображення статусу виконання аудиту в режимі реального часу.

Інші вимоги:

- ПЗ не повинно впливати на контент досліджуваних веб-сторінок;
- забезпечення коректної роботи з сучасними веб-технологіями (React, Vue, Angular, тощо);
- підтримка аналізу як статичних сторінок, так і SPA (Single Page Applications).

### 3.2 Вимоги до надійності

44165850.1551

Вимоги до надійності наступні:

- забезпечення стабільної роботи програми при аналізі різних типів веб-сторінок, включаючи ті, що містять помилки або повільно завантажуються;
- наявність механізмів обробки помилок та виведення зрозумілих повідомлень користувачу у разі невдалого аудиту;
- коректне завершення роботи headless браузера після кожного аудиту для запобігання витoku ресурсів;
- наявність timeout механізмів для уникнення безкінечного очікування завантаження сторінок;
- наявність резервної копії тексту програми на зовнішньому носії;

### 3.3 Умови експлуатації

- температура навколишнього повітря 15-30°C, відносна вологість 30-70%;
- доступ до мережі інтернет зі стабільним з'єднанням (мінімальна швидкість 5 Мбіт/с);
- наявність на пристрої сучасного веб-браузера (Google Chrome версії 90+, Mozilla Firefox 88+, Safari 14+, або аналогічний);
- для використання Web Performance Auditor користувач повинен мати базові навички роботи з веб-інтерфейсами та розуміти основи веб-технологій;
- для адміністрування та технічної підтримки рекомендується мати знання Node.js, Express, та основ роботи з Playwright.

### 3.4 Вимоги до складу і параметрів технічних засобів

Склад технічних засобів:

- веб-сервер для хостингу клієнтської частини програми (Vite dev server або статичний хостинг);
- API-сервер на базі Node.js для обробки запитів на аудит;
- Headless Chromium браузер (через Playwright) для проведення аналізу веб-сторінок.

Мінімальні системні вимоги для серверної частини:

- процесор: 2-ядерний CPU з тактовою частотою 2.0 GHz або вище;

44165850.1551

- оперативна пам'ять: мінімум 2 ГБ (рекомендовано 4 ГБ);
- вільне місце на диску: мінімум 500 МБ для встановлення Playwright та залежностей;
- операційна система: Linux, macOS, або Windows 10/11.

Мінімальні вимоги для клієнтської частини:

- будь-який пристрій з сучасним веб-браузером;
- роздільна здатність екрану: мінімум 1024x768 пікселів (рекомендовано 1366x768 або вище);

### 3.5 Вимоги до інформаційної і програмної сумісності

Інформаційні структури:

- обмін даними між клієнтом та сервером здійснюється через REST API у форматі JSON;
- всі метрики зберігаються у структурованому форматі для можливості подальшого аналізу;

Методи рішення та мови програмування:

Фронтенд (Front-end):

- HTML5, CSS3 для розмітки та стилізації;
- JavaScript (ES6+) для клієнтської логіки;
- Bootstrap 5.3+ для адаптивного дизайну та UI-компонентів;
- Vite як bundler та dev server.

Бекенд (Back-end):

- Node.js (версія 18+ або 20+) для серверної логіки;
- Express.js 4+ як веб-фреймворк для створення API;
- Playwright 1.47+ для автоматизації браузера та збору метрик.

API та інтеграції:

- Performance API (браузерний) для збору метрик продуктивності;
- PerformanceObserver API для моніторингу LCP та CLS;
- Navigation Timing API для вимірювання TTFB;
- Resource Timing API для аналізу завантаження ресурсів.

44165850.1551

### 3.6 Вимоги до маркування і упаковки

Приклад маркування програми (назва програми, розробник, контакти, рік розробки):

Web Performance Auditor

Розробник: Приймич Андрій

УДУНТ, кафедра КІТ

2026

Версія: 1.0.0

Упаковка програмного продукту включаючи документацію повинна бути захищена від механічних та кліматичних пошкоджень (наприклад, пластикова коробка для CD дисків або захищений USB-накопичувач).

### 3.7 Вимоги до транспортування і зберігання

Транспортування повинно забезпечити збереження програмного продукту та його цілісність, та запобігати несанкціонованому доступу до нього. Програмний виріб міститься на оптичному носії даних типу CD/DVD або флеш-носії та повинен мати відповідну упаковку, що має захист від механічних ушкоджень та атмосферного впливу (пластиковий футляр).

Зберігання програмного продукту повинно здійснюватися в приміщенні з температурою 15-25°C та відносною вологістю не більше 70%.

44165850.1551

#### 4 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

До складу програмної документації мають входити:

1. Технічне завдання (цей документ)
2. Керівництво користувача з описом:
  - призначення програми
  - інструкції з встановлення та запуску
  - опис інтерфейсу користувача
  - опис повідомлень та можливих помилок

Вся документація до програмного додатку повинна задовольняти вимогам до програмної документації згідно ДСТУ.

44165850.1551

## 5 СТАДІЇ ТА ЕТАПИ РОЗРОБКИ

№ з/п	Назва етапу розробки	Строк виконання	Примітка
1	Огляд літератури та аналіз існуючих інструментів аудиту	01.09.2025 - 05.09.2025	
2	Розробка структур вхідних і вихідних даних, вимог до системи	06.09.2025 - 18.09.2025	
3	Дослідження метрик продуктивності та методів їх збору	19.09.2025 - 30.09.2025	
4	Узгодження та затвердження ТЗ, постановка задачі	01.10.2025 - 10.10.2025	30%
5	Розробка серверної частини (API, інтеграція з Playwright)	11.10.2025 - 17.10.2025	
6	Розробка логіки збору та обробки метрик	18.10.2025 - 27.10.2025	
7	Розробка і реалізація клієнтського інтерфейсу	28.10.2025 - 03.11.2025	
8	Інтеграція фронтенду та бекенду	04.11.2025 - 08.11.2025	60%
9	Тестування та відлагодження програми	09.11.2025 - 11.11.2025	100%

44165850.1551

## 6 ПОРЯДОК І КОНТРОЛЬ ПРИЙМАННЯ

Контроль за виконанням роботи здійснює керівник розробки Грiшечкіна

Тетяна Сергiївна.

Прийм програмного продукту здійснюється комісією у складі:

- завідувач кафедри КІТ;
- керівник розробки;
- нормоконтролер.

Критерії приймання:

Відповідність функціоналу програми технічному завданню

- коректність роботи всіх функцій програми;
- відповідність інтерфейсу вимогам зручності використання;
- наявність повної програмної документації;
- успішне проходження тестування на різних типах веб-сторінок;
- відповідність часових характеристик заявленим вимогам.

## 7 БІБЛІОГРАФІЧНИЙ СПИСОК

- Горячкін, В. М. Навчальний посібник для підготовки кваліфікаційної роботи на здобуття ОС Бакалавр [Текст] / В. М. Горячкін, О. В. Горбова, О. С. Куроп'ятник. -- Дніпро : Український державний університет науки і технологій, 2022. -- 136-137с.
- Івченко, Ю.М. Основи стандартизації програмних систем: методичні вказівки до дипломного проектування та лабораторних робіт / уклад.: Ю.М. Івченко, В. І. Шинкаренко, В. Г. Івченко; Дніпропетр. нац. ун-т залізн. трансп. ім. акад. В. Лазаряна. -- Д.: Вид-во Дніпропетр. нац. ун-ту залізн. трансп. ім. акад. В. Лазаряна, 2009. - 38 с.
- Web Performance Optimization [Електронний ресурс] / Google Developers, Режим доступу  
[URL]<https://developers.google.com/web/fundamentals/performance>
- Performance API [Електронний ресурс] / MDN Web Docs, Режим доступу  
[URL][https://developer.mozilla.org/en-US/docs/Web/API/Performance\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Performance_API)

44165850.1551

Додаток Б  
Текст програми  
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЗАТВЕРДЖУЮ  
Перший проректор  
Українського  
державного  
університету науки і  
технологій  
Анатолій РАДКЕВИЧ

Web Perfomance Auditor

ЛИСТ  
ЗАТВЕРДЖЕННЯ  
Текст програми  
44165850.1551-01 12-01  
ЛЗ

Завідувач кафедри КІТ  
Вадим ГОРЯЧКІН  
Керівник розробки  
Тетяна ГРИШЕЧКІНА  
Виконавець  
Андрій ПРИЙМИЧ  
Нормоконтролер  
Світлана ВОЛКОВА

44165850.1551

ЗАТВЕРДЖЕНО

44165850.1551-01-ЛЗ

Web Perfomance Auditor

ТЕКСТ ПРОГРАМИ

44165850.1551-01 12-01

Листів 5

44165850.1551

## АННОТАЦІЯ

Документ 44165850.1551-01 12-01 “ Web Perfomance Auditor. Додаток Б. Текст програми” входить до складу програмної документації на застосунок Web Perfomance Auditor.

У цьому документі представлено текст програми яку було розроблено для дослідження

44165850.1551

## Index.html

```
<!DOCTYPE html>
<html lang="uk">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <title>Аудит URL – метрики рендерингу</title>
    <link rel="icon" href="data:," />
    <script type="module">
      import "bootstrap/dist/css/bootstrap.min.css";
      import "./src/styles.css";
    </script>
  </head>
  <body>
    <div class="container">
      <header class="mb-4">
        <h1 class="h3">Аудит сайту за URL</h1>
        <p class="text-muted small">
          Введи адресу сторінки й натисни <em>Аудит</em>. Отримаєш TTFB, FCP,
          LCP, CLS і список ресурсів.
        </p>
      </header>

      <form id="audit-form" class="row gy-2 gx-2 align-items-center">
        <div class="col-lg-7">
          <input
            id="url"
            type="url"
            class="form-control"
            placeholder="https://example.com"
            required
          />
        </div>
        <div class="col-lg-auto">
          <button class="btn btn-primary" type="submit">
            <span class="spinner-border spinner-border-sm me-2 d-none" id="spin" aria-hidden="true"></span>
            Аудит
          </button>
        </div>
        <div class="col-lg-auto">
          <span id="status" class="text-muted small"></span>
        </div>
      </form>

      <hr class="my-4" />

      <div id="summary" class="alert d-none" role="alert"></div>

      <div id="results" class="d-none">
        <div class="row g-3">
          <div class="col-lg-6">
            <div class="card shadow-sm">
```

44165850.1551

```

<div class="card-body">
  <div class="d-flex justify-content-between align-items-center mb-2">
    <h2 class="h5 mb-0">Ключові метрики</h2>
    <div class="small">
      <span class="badge bg-success">Добре</span>
      <span class="badge bg-warning text-dark">Слід покращити</span>
      <span class="badge bg-danger">Погано</span>
    </div>
  </div>
  <div class="table-wrap">
    <table class="table table-sm align-middle" id="metrics-table">
      <thead>
        <tr>
          <th>Метрика</th>
          <th>Значення</th>
          <th>Оцінка</th>
          <th>Пояснення</th>
        </tr>
      </thead>
      <tbody></tbody>
    </table>
  </div>
  <details class="mt-2">
    <summary class="small text-muted">Що означають ці метрики?</summary>
    <ul class="small mt-2 mb-0">
      <li><strong>TTFB</strong> – наскільки швидко сервер віддає перший байт відповіді.</li>
      <li><strong>FCP</strong> – коли користувач вперше бачить будь-який контент (текст/фон/зображення).</li>
      <li><strong>LCP</strong> – коли з'являється найбільший елемент (герой-зображення чи великий блок тексту)
      – критично для сприйняття швидкості.</li>
      <li><strong>CLS</strong> – стабільність макета; скільки «стрибає» сторінка під час завантаження.</li>
    </ul>
  </details>
  <p class="small text-muted mb-0 mt-2">
    * Обчислено в headless Chromium через Performance API та спостерігачі.
  </p>
</div>
</div>
</div>

<div class="col-lg-6">
  <div class="card shadow-sm">
    <div class="card-body">
      <div class="d-flex justify-content-between align-items-center mb-2">
        <h2 class="h5 mb-0">Ресурси (топ за тривалістю)</h2>
        <select id="res-filter" class="form-select form-select-sm" style="max-width:180px">
          <option value="">Усі типи</option>
          <option value="script">script</option>
          <option value="link">style</option>
          <option value="img">img</option>
          <option value="fetch">fetch</option>
          <option value="css">css</option>
        </select>
      </div>
    </div>
  </div>
</div>

```

44165850.1551

```

</div>
<div class="table-wrap">
  <table class="table table-sm align-middle" id="resources-table">
    <thead>
      <tr>
        <th>Тип</th>
        <th>URL</th>
        <th>Тривалість</th>
        <th>Розмір</th>
      </tr>
    </thead>
    <tbody></tbody>
  </table>
</div>
<p class="small text-muted mb-0">Натисни на заголовок стовпця для сортування.</p>
</div>
</div>
</div>
</div>
</div>
</div>
</div>

<script type="module" src="/src/main.js"></script>
</body>
</html>

```

Src/main.js

```

const form = document.getElementById('audit-form');
const statusEl = document.getElementById('status');
const results = document.getElementById('results');
const summary = document.getElementById('summary');
const spin = document.getElementById('spin');

const metricsBody = document.querySelector('#metrics-table tbody');
const resBody = document.querySelector('#resources-table tbody');
const resFilter = document.getElementById('res-filter');

let lastResources = [];
let currentSort = { key: 'duration', dir: 'desc' };

function fmtMs(ms) {
  if (ms == null || Number.isNaN(ms)) return '-';
  return `${Math.round(ms)} ms`;
}

function fmtBytes(b) {
  if (!b || b <= 0) return '0 B';
  const units = ['B', 'KB', 'MB', 'GB'];
  let i = 0, n = b;
  while (n >= 1024 && i < units.length - 1) { n /= 1024; i++; }
  return `${n.toFixed(n < 10 && i > 0 ? 1 : 0)} ${units[i]}`;
}

```

44165850.1551

```

const THRESHOLDS = {
  TTFB: { good: 200, meh: 500 },
  FCP: { good: 1800, meh: 3000 },
  LCP: { good: 2500, meh: 4000 },
  CLS: { good: 0.1, meh: 0.25 }
};

function rateMetric(name, value) {
  if (value == null) return { badge: '<span class="badge bg-secondary">--</span>', score: 2 };
  const t = THRESHOLDS[name];
  if (!t) return { badge: '<span class="badge bg-secondary">--</span>', score: 2 };

  const v = value;
  if (v <= t.good) return { badge: '<span class="badge bg-success">Добре</span>', score: 0 };
  if (v <= t.meh) return { badge: '<span class="badge bg-warning text-dark">Слід покращити</span>', score: 1 };
  return { badge: '<span class="badge bg-danger">Погано</span>', score: 2 };
}

function buildSummary(metrics) {
  const keys = ['FCP', 'LCP', 'CLS'];
  const worst = Math.max(...keys.map(k => rateMetric(k, metrics[k]).score));
  const classes = ['alert-success', 'alert-warning', 'alert-danger'];
  const texts = [
    'Сторінка загалом швидко відмальовується.',
    'Є простір для оптимізації відмальовування.',
    'Потрібні суттєві покращення продуктивності.'
  ];
  summary.className = `alert ${classes[worst]}`;
  summary.textContent = texts[worst];
  summary.classList.remove('d-none');
}

function renderMetrics(m) {
  const explain = {
    TTFB: 'Час до першого байта від сервера (responseStart - requestStart).',
    FCP: 'Перший видимий контент на екрані.',
    LCP: 'Найбільший елемент стає видимим (герой, великий текст).',
    CLS: 'Стабільність макета (накопичене зміщення).'
  };
};

const row = (name, value, pretty) => {
  const val = name === 'CLS'
    ? (value == null ? '-' : String(Number(value).toFixed(3)))
    : fmtMs(value);
  const rating = rateMetric(name, value).badge;
  return `<tr>
  <td><strong>${name}</strong></td>
  <td>${val}</td>
  <td>${rating}</td>
  <td class="small">${pretty}</td>
</tr>`;
}

```

44165850.1551

```

};

metricsBody.innerHTML = [
  row('TTFB', m.TTFB, explain.TTFB),
  row('FCP', m.FCP, explain.FCP),
  row('LCP', m.LCP, explain.LCP),
  row('CLS', m.CLS, explain.CLS)
].join('');

buildSummary(m);
}

function renderResources(resources) {
  const type = resFilter.value;
  let list = type ? resources.filter(r => r.initiatorType === type || (type === 'css' && r.initiatorType === 'link')) : resources.slice();

  const key = currentSort.key;
  const dir = currentSort.dir === 'asc' ? 1 : -1;
  list.sort((a, b) => {
    const va = key === 'name' ? a.shortName : a[key] ?? 0;
    const vb = key === 'name' ? b.shortName : b[key] ?? 0;
    if (va < vb) return -1 * dir;
    if (va > vb) return 1 * dir;
    return 0;
  });

  resBody.innerHTML = list.map(r => `
  <tr>
    <td>${r.initiatorType}</td>
    <td title="${r.name}">${r.shortName}</td>
    <td>${fmtMs(r.duration)}</td>
    <td>${fmtBytes(r.transferSize)}</td>
  </tr>
  `).join('') || '<tr><td colspan="4">--</td></tr>';
}

document.querySelectorAll('#resources-table thead th').forEach((th, idx) => {
  th.style.cursor = 'pointer';
  th.addEventListener('click', () => {
    const map = { 0: 'initiatorType', 1: 'name', 2: 'duration', 3: 'transferSize' };
    const key = map[idx];
    if (currentSort.key === key) {
      currentSort.dir = currentSort.dir === 'asc' ? 'desc' : 'asc';
    } else {
      currentSort.key = key;
      currentSort.dir = key === 'name' ? 'asc' : 'desc';
    }
    renderResources(lastResources);
  });
});
};

```

44165850.1551

```

resFilter.addEventListener('change', () => renderResources(lastResources));

form.addEventListener('submit', async (e) => {
  e.preventDefault();
  const url = document.getElementById('url').value.trim();
  if (!url) return;

  statusEl.textContent = 'Запуск headless-аудиту...';
  results.classList.add('d-none');
  summary.classList.add('d-none');
  spin.classList.remove('d-none');

  try {
    const res = await fetch(`/api/audit?url=${encodeURIComponent(url)}`);
    if (!res.ok) throw new Error(`HTTP ${res.status}`);
    const data = await res.json();

    statusEl.textContent = 'Готово';
    spin.classList.add('d-none');

    const m = data.metrics || {};
    renderMetrics(m);

    lastResources = Array.isArray(data.resources) ? data.resources : [];
    renderResources(lastResources);

    results.classList.remove('d-none');
  } catch (err) {
    statusEl.textContent = 'Помилка: ' + err.message;
    summary.className = 'alert alert-danger';
    summary.textContent = 'Не вдалося виконати аудит. Перевір URL або спробуй ще раз.';
    summary.classList.remove('d-none');
    spin.classList.add('d-none');
    console.error(err);
  }
});

```

Src/main.css

```

body { padding: 1.5rem; }
.table-wrap { overflow: auto; }
.small { font-size: .9rem; }
code { background: #f8f9fa; padding: .125rem .375rem; border-radius: .25rem; }

```

Package.json

```

{
  "name": "prymich-diploma",
  "private": true,
  "version": "0.1.0",
  "type": "module",
  "scripts": {

```

44165850.1551

```

    "dev": "vite",
    "build": "vite build",
    "preview": "vite preview",
    "serve": "node server.js"
  },
  "devDependencies": {
    "vite": "^5.4.0"
  },
  "dependencies": {
    "bootstrap": "^5.3.3",
    "express": "^4.19.2",
    "playwright": "^1.47.2"
  }
}

```

Server.js

```

import express from 'express';
import { chromium } from 'playwright';

const app = express();
const PORT = process.env.PORT || 3001;

app.get('/api/health', (_, res) => res.json({ ok: true }));

app.get('/api/audit', async (req, res) => {
  const target = req.query.url;
  if (!target || typeof target !== 'string') {
    return res.status(400).json({ error: 'Missing ?url' });
  }

  const browser = await chromium.launch({ headless: true });
  const context = await browser.newContext({
    viewport: { width: 1366, height: 768 }
  });
  const page = await context.newPage();

  await page.addInitScript(() => {

    window.__audit = { LCP: null, CLS: 0 };
    try {
      const lcpObs = new PerformanceObserver((list) => {
        const entries = list.getEntries();
        const last = entries[entries.length - 1];
        if (last) {
          window.__audit.LCP = last.startTime;
        }
      });
      lcpObs.observe({ type: 'largest-contentful-paint', buffered: true });
    } catch {}

    try {

```

44165850.1551

```

let cls = 0;
const clsObs = new PerformanceObserver((list) => {
  for (const e of list.getEntries()) {
    if (!e.hadRecentInput) cls += e.value;
  }
  window.__audit.CLS = +cls.toFixed(4);
});
clsObs.observe({ type: 'layout-shift', buffered: true });
} catch {}
});

try {
const resp = await page.goto(target, { waitUntil: 'load', timeout: 45000 });
await page.waitForLoadState('networkidle', { timeout: 15000 });
await page.waitForTimeout(2500);

const data = await page.evaluate(() => {
  const nav = performance.getEntriesByType('navigation')[0];
  const ttfb = nav ? nav.responseStart - nav.requestStart : null;

  const paints = performance.getEntriesByType('paint');
  const fcpEntry = paints.find(p => p.name === 'first-contentful-paint');
  const FCP = fcpEntry ? fcpEntry.startTime : null;

  const LCP = (window.__audit && window.__audit.LCP) || null;
  const CLS = (window.__audit && window.__audit.CLS) ?? null;

  const resources = performance.getEntriesByType('resource').map(r => ({
    name: r.name,
    shortName: r.name.length > 64 ? '...' + r.name.slice(-63) : r.name,
    initiatorType: r.initiatorType,
    duration: r.duration,
    transferSize: r.transferSize
  })));

  resources.sort((a,b) => b.duration - a.duration);

  return {
    metrics: { TTFB: ttfb, FCP, LCP, CLS },
    resources: resources.slice(0, 30)
  };
});

res.json({ url: target, ...data, status: resp?.status() });
} catch (err) {
  console.error(err);
  res.status(500).json({ error: String(err) });
} finally {
  await context.close();
  await browser.close();
}
});

```

44165850.1551

```
app.listen(PORT, () => {  
  console.log('API listening on :' + PORT);  
});
```

Vite.config.js

```
import { defineConfig } from 'vite';  
  
export default defineConfig({  
  server: {  
    port: 5173,  
    proxy: {  
      '/api': 'http://localhost:3001'  
    }  
  }  
});
```

44165850.1551

Додаток В  
Керівництво користувача  
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЗАТВЕРДЖУЮ  
Перший проректор  
Українського  
державного університету  
науки і технологій  
Анатолій РАДКЕВИЧ

Web Perfomance Auditor

Керівництво користувача  
ЛИСТ ЗАТВЕРДЖЕННЯ  
44165850.1551-01 ІЗ 01-ЛЗ

Завідувач кафедри КІТ  
Вадим ГОРЯЧКІН  
Керівник розробки  
Тетяна ГРИШЕЧКІНА  
Виконавець  
Андрій ПРИЙМИЧ  
Нормоконтролер Світлана  
ВОЛКОВА

44165850.1551

ЗАТВЕРДЖЕНО  
44165850.1551-01 ІЗ 01-ЛЗ

Web Perfomance Auditor

Керівництво користувача

44165850.1551-01 ІЗ 01

Листів 12

2026

44165850.1551

## АННОТАЦІЯ

Документ 44165850.1551-01 ІЗ 01 “ВПЛИВ СТРАТЕГІЙ ПОПЕРЕДНЬОГО ЗАВАНТАЖЕННЯ РЕСУРСІВ НА ШВИДКІСТЬ ВІДОБРАЖЕННЯ ВЕБ-СТОРІНОК. Додаток В. Керівництво користувача” входить до складу програмної документації Web Perfomance Auditor.

У даному документі представлено керівництво користувача.

44165850.1551

ЗМІСТ  
ВВЕДЕННЯ

.....**Error!**

**Bookmark not defined.**

1.УМОВИ ЗАСТОСУВАННЯ.....

**Error! Bookmark not defined.**

1.1.Вимоги до складу і параметрів технічних засобів  
.....**Error! Bookmark not defined.**

1.2. Вимоги до інформаційного та програмного середовища  
.....**Error! Bookmark not defined.**

2.ПІДГОТОВКА ДО ВИКОРИСТАННЯ.....

**Error! Bookmark not defined.**

3.ОПИС ЕКРАНІВ ТА ОПЕРАЦІЙ.....

**Error! Bookmark not defined.**

4.АВАРІЙНІ СИТУАЦІЇ.....

**Error! Bookmark not defined.**

5.СТРУКТУРА ПОВІДОМЛЕНЬ

.....**Error! Bookmark not defined.**

44165850.1551

## ВВЕДЕННЯ

Цей застосунок покликаний стати зручним та ефективним інструментом для аналізу технічного стану веб-сторінок та застосунків, орієнтуючись на максимальну простоту та доступність. Основна проблема, яку він вирішує, полягає у високому бар'єрі входу до сфери професійної оптимізації, де зазвичай панує складна термінологія та заплутані звіти. Програма автоматично збирає найважливіші дані про швидкість роботи ресурсу та перетворює їх на зрозумілу картину, що дозволяє легко оцінити якість завантаження, що може допомогти зрозуміти чи потрібно залучати профільних експертів.

Користувач може використовувати усі функції застосунку без додаткових навичок, оскільки інтерфейс та логіка роботи побудовані на принципі інтуїтивності. Програма не вимагає від людини знання мов програмування чи глибокого розуміння архітектури серверів.

44165850.1551

## 1.УМОВИ ЗАСТОСУВАННЯ

### 1.1. Вимоги до складу і параметрів технічних засобів

Вимоги при доступі з мобільного пристрою(смартфону):

- процесор - MediaTek MT6580;
- оперативна пам'ять - не менше 2 ГБ;
- підтримка мереж - Підтримка 4g та Wifi мереж.

Вимоги при доступі з персонального комп'ютера:

- процесор - Intel Core I3 2<sup>nd</sup> gen;
- оперативна пам'ять - не менше 4 гб;
- підтримка мереж - підтримка Ethernet/ WiFi.

### 1.2. Вимоги до інформаційного та програмного середовища

Вимоги до програмного середовища при доступі з мобільного пристрою:

- операційна система - Android 8 Nougat або вище, IOS 12.0 та вище;
- браузер з підтримкою Javascript та localStorage.

Вимоги до програмного середовища при доступі з ПК:

- операційна система - Windows 10, MacOS Mojave чи Linux;
- браузер з підтримкою Javascript та localStorage.

44165850.1551

## 2. ПІДГОТОВКА ДО ВИКОРИСТАННЯ

Встановлення додаткового ПЗ перед використанням не потрібне. Носій даних з програмою містить керівництво користувача та текстовий файл формату txt app.txt із URL посиланням на застосунок. Перед використанням необхідно ознайомитись із керівництвом користувача.

Для початку роботи з вебзастосунком необхідно відкрити браузер та набрати в пошуковому рядку посилання на додаток, або вставити його до цього скопіювавши із файлу app.txt.

44165850.1551

### 3. ОПИС ЕКРАНІВ ТА ОПЕРАЦІЙ

Web Performance Auditor Має лише два екрани . Нижче наведено детальний опис екранів та їх демонстрацію.

#### Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

Рисунок 3.1 – Головний екран застосунку

На цей екран користувач потрапляє одразу після завантаження програми. Є короткий опис функцій а також можна розпочати аудит, для цього потрібно ввести посилання на потрібну сторінку та натиснути “Аудит”.

Після запуску аудиту екран змінює стан та відображає етапи проведення аудиту. Це виглядає наступним чином:

#### Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

  Запуск headless-аудиту...

Рисунок 3.2 – Головний екран застосунку(аудит запущено)

Останнім є функціональний екран, який користувач отримує після завершення аудиту . Головна частина інтерфейсу розділена на дві інформативні зони. Зліва розташована таблиця з ключовими метриками, такими як час до першого байта (TTFB), швидкість появи першого контенту (FCP), час завантаження найбільшого елемента (LCP) та стабільність макета (CLS). Справа розташована зона з файлами, що відображає назву файлу, його категорію (скрипт, шрифт, зображення, тощо), розмір та час, що пішов на завантаження. Також є підказки щодо отриманих значень та невелика пам’ятка про метрики та їх призначення.

44165850.1551

Аудит сайту за URL

Введи адресу сторінки й натисни *Аудит*. Отримаєш TTFB, FCP, LCP, CLS і список ресурсів.

https://lider.ust.edu.ua/login/index.php Аудит Готово

Потрібні суттєві покращення продуктивності.

### Ключові метрики

Добре Слід покращити Погано

Метрика	Значення	Оцінка	Пояснення
TTFB	131 ms	Добре	Час до першого байта від сервера (responseStart – requestStart).
FCP	3672 ms	Погано	Перший видимий контент на екрані.
LCP	3672 ms	Слід покращити	Найбільший елемент стає видимим (герой, великий текст).
CLS	0.000	Добре	Стабільність макета (накопичене зміщення).

► Що означають ці метрики?

\* Обчислено в headless Chromium через Performance API та спостережувачі

### Ресурси (топ за тривалістю)

Усі типи

Тип	URL
iframe	https://www.youtube.com/embed/zwZtvWJx5zw?si=Y1LI
script	...s://lider.ust.edu.ua/lib/requirejs.php/1766128032/core/
script	...s://lider.ust.edu.ua/lib/requirejs.php/1766128032/core/
css	...e.php/1/theme_moove/loginbgimg/1766128032/Backç
script	...der.ust.edu.ua/lib/requirejs.php/1766128032/core_form
img	...php/1/theme_moove/logo/1766128032/LOGO%20-%2
script	...du.ua/theme/yui_combo.php?rollup/3.17.2/yui-moodle
script	...edu.ua/lib/javascript.php/1766128032/lib/polyfills/po

Рисунок 3.3 – Функціональний екран програми

44165850.1551

#### 4.АВАРІЙНІ СИТУАЦІЇ

Додаток спроектований так, щоб не містити аварійних ситуацій. Усі аварійні ситуації, що можуть статись пов'язані із недотриманням вимог до технічних параметрів та програмного середовища. Вебзастосунок не зберігає дані локально на пристрої користувача, тому в разі аварійної ситуації даним застосунку нічого не загрожує.

44165850.1551

## 5. СТРУКТУРА ПОВІДОМЛЕНЬ

Таблиця 5.1 – Повідомлення які отримує користувач

Текст повідомлення	Опис ситуації в якій користувач отримує повідомлення	Рекомендовані дії
Введіть URL	Користувач вводить некоректне значення в поле для введення посилання	Перевірити правильність введеного посилання
Не вдалося виконати аудит. Перевір URL або спробуй ще раз.	Сайт який користувач пробує проаналізувати недоступний або має обмежений доступ	Перевірити правильність введеного посилання. Перевірити доступність сайту. Спробувати пізніше.
Потрібні суттєві покращення продуктивності.	Сайт котрий перевіряє користувач має серйозні проблеми в оптимізації	Це результат аудиту, необхідно проаналізувати показники які видає програма та вжити заходи по оптимізації за потреби
Є простір для оптимізації відмальовування	Сайт котрий перевіряє користувач має деякі проблеми в оптимізації	Це результат аудиту, необхідно проаналізувати показники які видає програма та вжити заходи по оптимізації за потреби

44165850.1551

Сторінка загалом швидко відмальовується.	Сайт або веб-застосунок, котрий перевіряється не має серйозних проблем з оптимізацією	Це результат аудиту, серйозних проблем не виявлено хоча ми(я) рекомендую переглянути числові характеристики, можливо вдасться покращити результати
--	---	--

44165850.1551

Додаток Г  
Лістинг веб-сайтів на котрих проводиться тестування  
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЗАТВЕРДЖУЮ  
Перший проректор  
Українського  
державного  
університету науки і  
технологій  
Анатолій РАДКЕВИЧ

Web Perfomance Auditor

Лістинг веб-сайтів на котрих проводиться тестування

ЛИСТ ЗАТВЕРДЖЕННЯ  
44165850.1551-01 90-01 ЛЗ

Завідувач кафедри КІТ  
Вадим ГОРЯЧКІН  
Керівник розробки  
Тетяна ГРИШЕЧКІНА  
Виконавець  
Андрій ПРИЙМИЧ  
Нормоконтролер  
Світлана ВОЛКОВА

44165850.1551

ЗАТВЕРДЖЕНО

44165850.1551-01-ЛЗ

Web Performance Auditor

Лістинг веб-сайтів на  
котрих проводиться  
тестування  
44165850.1551-01 90-01  
Листів 66

44165850.1551

### АННОТАЦІЯ

Документ 44165850.1551-01 90-01 “Вплив стратегій попереднього завантаження ресурсів на швидкість відображення веб-сторінок. Додаток Г. Лістинг веб-сайтів на котрих проводиться тестування ” входить до складу програмної документації на застосунок Web Perfomance Auditor.

У цьому документі представлено лістинг сайтів які використовуються для тестування

44165850.1551

## Package.json

```
{
  "name": "url-audit-demo",
  "version": "1.0.0",
  "description": "Demo websites for testing web performance",
  "type": "commonjs",
  "main": "index.js",
  "scripts": {
    "start": "node index.js",
    "dev": "nodemon index.js"
  },
  "dependencies": {
    "express": "^4.18.2"
  },
  "devDependencies": {
    "nodemon": "^3.0.2"
  },
  "engines": {
    "node": ">=16.0.0"
  }
}
```

## Package-lock.json

```
{
  "name": "url-audit-demo",
  "version": "1.0.0",
  "lockfileVersion": 3,
  "requires": true,
  "packages": {
    "": {
      "name": "url-audit-demo",
      "version": "1.0.0",
      "dependencies": {
        "express": "^4.18.2"
      },
      "devDependencies": {
        "nodemon": "^3.0.2"
      },
      "engines": {
        "node": ">=16.0.0"
      }
    },
    "node_modules/accepts": {
      "version": "1.3.8",
      "resolved": "https://registry.npmjs.org/accepts/-/accepts-1.3.8.tgz",
      "integrity": "sha512-LwW2AFLUGPVEKe9vwNS/d9Xg/z6kQ/zQDHJ/nvH9DybX9pZQwEjxtMI8bVdAe/H+WDgiR/nBR2GVX4U04qUw==",
      "dependencies": {
        "mime-types": "2.1.34",
        "negotiator": "0.6.2"
      }
    }
  }
}
```

44165850.1551

```

    "license": "MIT",
    "dependencies": {
      "mime-types": "~2.1.34",
      "negotiator": "0.6.3"
    },
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/anymatch": {
    "version": "3.1.3",
    "resolved": "https://registry.npmjs.org/anymatch/-/anymatch-3.1.3.tgz",
    "integrity": "sha512-KMReFUr0B4t+D+OBkjR3KYqvocp2XaSz055UcB6mgQMd3KbcE+mWTyvVV7D/zsdEbNnV6acZUutkiHQXvTr1Rw==",
    "dev": true,
    "license": "ISC",
    "dependencies": {
      "normalize-path": "^3.0.0",
      "picomatch": "^2.0.4"
    },
    "engines": {
      "node": ">= 8"
    }
  },
  "node_modules/array-flatten": {
    "version": "1.1.1",
    "resolved": "https://registry.npmjs.org/array-flatten/-/array-flatten-1.1.1.tgz",
    "integrity": "sha512-PCVAQswWemu6UdxdDFFX/+gVeYqKAod3D3UVm91jHwyngu0wAvYPhx8nNlM++NqRcK6CxxpUafjmhIdKiHibqg==",
    "license": "MIT"
  },
  "node_modules/balanced-match": {
    "version": "1.0.2",
    "resolved": "https://registry.npmjs.org/balanced-match/-/balanced-match-1.0.2.tgz",
    "integrity": "sha512-3oSeU00TMV67hN1AmbXsK4yaqU7tjiHl1bxRDZ0pH0KW9+CeX4bRAaX0Anxt0tx2MrpRpWwQaPwIlISEJhYU5Pw==",
    "dev": true,
    "license": "MIT"
  },
  "node_modules/binary-extensions": {
    "version": "2.3.0",
    "resolved": "https://registry.npmjs.org/binary-extensions/-/binary-extensions-2.3.0.tgz",

```

44165850.1551

```

    "integrity": "sha512-
Ceh+7ox5qe7LJuLHoY0feh3pHuUDHAcRUeyL2VYghZwfpkNIy/+80cg0a3UuSoYzavmylwuLWQOf3h10jjMM
Iw==",
    "dev": true,
    "license": "MIT",
    "engines": {
      "node": ">=8"
    },
    "funding": {
      "url": "https://github.com/sponsors/sindresorhus"
    }
  },
  "node_modules/body-parser": {
    "version": "1.20.4",
    "resolved": "https://registry.npmjs.org/body-parser/-/body-parser-1.20.4.tgz",
    "integrity": "sha512-
ZTgYYLMOXY9qKU/57FAo8F+HA2dGX7bqGc71txDRC1rS4frdFI5R7NhlUxHxH6M0YItAP0sHB4uqAOcYKxO6u
GA==",
    "license": "MIT",
    "dependencies": {
      "bytes": "~3.1.2",
      "content-type": "~1.0.5",
      "debug": "2.6.9",
      "depd": "2.0.0",
      "destroy": "~1.2.0",
      "http-errors": "~2.0.1",
      "iconv-lite": "~0.4.24",
      "on-finished": "~2.4.1",
      "qs": "~6.14.0",
      "raw-body": "~2.5.3",
      "type-is": "~1.6.18",
      "unpipe": "~1.0.0"
    },
    "engines": {
      "node": ">= 0.8",
      "npm": "1.2.8000 || >= 1.4.16"
    }
  },
  "node_modules/brace-expansion": {
    "version": "1.1.12",
    "resolved": "https://registry.npmjs.org/brace-expansion/-/brace-expansion-
1.1.12.tgz",
    "integrity": "sha512-
9T9UjW3r0UW5c1Q7GTW1lptXwhvYmEzFhzMfZ9H7FQWt+uZePjZPjBP/W1ZEyZ1twGWom5/56TF41PcqjndH
cg==",
    "dev": true,
    "license": "MIT",
    "dependencies": {
      "balanced-match": "^1.0.0",

```

44165850.1551

```
    "concat-map": "0.0.1"
  },
  "node_modules/braces": {
    "version": "3.0.3",
    "resolved": "https://registry.npmjs.org/braces/-/braces-3.0.3.tgz",
    "integrity": "sha512-xQVXQd9m5v47j0DOPu6Jx/WmkURP7YHtMBCy5pzQ1vbn6w9yK1j+P+hvntewYqu4L09lY4d8l6A4+hkKYog=",
    "dev": true,
    "license": "MIT",
    "dependencies": {
      "fill-range": "^7.1.1"
    },
    "engines": {
      "node": ">=8"
    }
  },
  "node_modules/bytes": {
    "version": "3.1.2",
    "resolved": "https://registry.npmjs.org/bytes/-/bytes-3.1.2.tgz",
    "integrity": "sha512-/Zb3j+PW6djS332q57b64eB1k2FBhkl5g87x8f9f4B3vPkYc085zPBj+5lk8+8RS93JV8tSP8gPg6Ev4zQ==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.8"
    }
  },
  "node_modules/call-bind-apply-helpers": {
    "version": "1.0.2",
    "resolved": "https://registry.npmjs.org/call-bind-apply-helpers/-/call-bind-apply-helpers-1.0.2.tgz",
    "integrity": "sha512-803J35RvE6G0A5hCV/1ldGgqOy0468eSgqf0e36F0ZgHA0wNQcncw0lM90uiv0kzO9bd31zr9gp5nErQQ=",
    "license": "MIT",
    "dependencies": {
      "es-errors": "^1.3.0",
      "function-bind": "^1.1.2"
    },
    "engines": {
      "node": ">= 0.4"
    }
  },
  "node_modules/call-bound": {
    "version": "1.0.4",
    "resolved": "https://registry.npmjs.org/call-bound/-/call-bound-1.0.4.tgz",
```

44165850.1551

```

    "integrity": "sha512-
+ys997U96po4Kx/ABpBCqhA9EuxJaQWDQg7295H4hBphv3IZg0boBKuWYpt4YXp6MZ5AmZQnU/tyMT1RpaSe
jg==",
    "license": "MIT",
    "dependencies": {
      "call-bind-apply-helpers": "^1.0.2",
      "get-intrinsic": "^1.3.0"
    },
    "engines": {
      "node": ">= 0.4"
    },
    "funding": {
      "url": "https://github.com/sponsors/ljharb"
    }
  },
  "node_modules/chokidar": {
    "version": "3.6.0",
    "resolved": "https://registry.npmjs.org/chokidar/-/chokidar-3.6.0.tgz",
    "integrity": "sha512-
7VT13fmjotKpGipCW9JEQAusEPE+Ei8n16/g4FBAmIm0GOOLMua9NDDo/DWp0ZAXCr3cPq5ZpBqmPAQgDda2
Pw==",
    "dev": true,
    "license": "MIT",
    "dependencies": {
      "anymatch": "~3.1.2",
      "braces": "~3.0.2",
      "glob-parent": "~5.1.2",
      "is-binary-path": "~2.1.0",
      "is-glob": "~4.0.1",
      "normalize-path": "~3.0.0",
      "readdirp": "~3.6.0"
    },
    "engines": {
      "node": ">= 8.10.0"
    },
    "funding": {
      "url": "https://paulmillr.com/funding/"
    },
    "optionalDependencies": {
      "fsevents": "~2.3.2"
    }
  },
  "node_modules/concat-map": {
    "version": "0.0.1",
    "resolved": "https://registry.npmjs.org/concat-map/-/concat-map-0.0.1.tgz",
    "integrity": "sha512-
/Srv4dswyQNBfohGpz9o6Yb3Gz3SrUDqBH5rTuhGR7aht1bYKnVxw2bCFMR1jaA7EXHaXZ8wsHdodFvbkhKm
qg==",
    "dev": true,

```

44165850.1551

```

    "license": "MIT"
  },
  "node_modules/content-disposition": {
    "version": "0.5.4",
    "resolved": "https://registry.npmjs.org/content-disposition/-/content-disposition-0.5.4.tgz",
    "integrity": "sha512-FveZTNuGw04cx1AiWbzi6zTAL/lhehaWbTtgluJh4/E95DqMwTmha3KZN1aAWA8cFIhHzMZUvLevkw5Rqk+tSQ==",
    "license": "MIT",
    "dependencies": {
      "safe-buffer": "5.2.1"
    },
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/content-type": {
    "version": "1.0.5",
    "resolved": "https://registry.npmjs.org/content-type/-/content-type-1.0.5.tgz",
    "integrity": "sha512-nTjqfcBFEipKdXCv4YDQWcFmcLZKm81ldf0pAopTvyrFGVbcR6P/VAA5G7N+0tTr8QqiU0tFadD6FK4NtJwOA==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/cookie": {
    "version": "0.7.2",
    "resolved": "https://registry.npmjs.org/cookie/-/cookie-0.7.2.tgz",
    "integrity": "sha512-yki5XnKuf750150uGTllt6kKILY4nQ1eNIQatoXEByZ5dWgnKqbnqmTrBE5B4N71rMJKQ2ytWMiT02o0v6Ew/w==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/cookie-signature": {
    "version": "1.0.7",
    "resolved": "https://registry.npmjs.org/cookie-signature/-/cookie-signature-1.0.7.tgz",
    "integrity": "sha512-BVJCN4WpR0ZO8e9VgMkilTYzqHTFa71Dvv9cb+4j7BD6YdZa8RjAWOn3pmn9dN3I6E8N/3k3oXx4iVH9cg==",
    "license": "MIT"
  },

```

44165850.1551

```

"node_modules/debug": {
  "version": "2.6.9",
  "resolved": "https://registry.npmjs.org/debug/-/debug-2.6.9.tgz",
  "integrity": "sha512-bC7ElrdJaJnPbAP+1EotYvqZsb3ecl15wi6Bfi6BJTUcNowp6cvspg0jXznRTKDjm/E7AdgFBVeAPVMNcKGsH
MA==",
  "license": "MIT",
  "dependencies": {
    "ms": "2.0.0"
  }
},
"node_modules/depd": {
  "version": "2.0.0",
  "resolved": "https://registry.npmjs.org/depd/-/depd-2.0.0.tgz",
  "integrity": "sha512-g7nH6P6dyDioJogAAGprGpCtVImJhpPk/roCzdb3fIh61/s/nPsfR6onyMwkCAR/0lC3yBC0lESvUoQEAssI
rw==",
  "license": "MIT",
  "engines": {
    "node": ">= 0.8"
  }
},
"node_modules/destroy": {
  "version": "1.2.0",
  "resolved": "https://registry.npmjs.org/destroy/-/destroy-1.2.0.tgz",
  "integrity": "sha512-g7JgTaXIIaR1w4iJ5SNoN0hnMY7Gpc/n8D4qSCJw8QqFWXf7cuAgnEHxBpweaVcPevC2l3KpjYCx3NypQQga
Jg==",
  "license": "MIT",
  "engines": {
    "node": ">= 0.8",
    "npm": "1.2.8000 || >= 1.4.16"
  }
},
"node_modules/dunder-proto": {
  "version": "1.0.1",
  "resolved": "https://registry.npmjs.org/dunder-proto/-/dunder-proto-
1.0.1.tgz",
  "integrity": "sha512-KIN/nDJBQRcXw0MLVhZE9iQHmG68qAVIBg9CqmUYjmQIhgij9U5MFvrqkUL5FbtyyzZuOe0t0zdeRe4UY7ct
+A==",
  "license": "MIT",
  "dependencies": {
    "call-bind-apply-helpers": "^1.0.1",
    "es-errors": "^1.3.0",
    "gopd": "^1.2.0"
  },
  "engines": {
    "node": ">= 0.4"
  }
}

```

44165850.1551

```

    }
  },
  "node_modules/ee-first": {
    "version": "1.1.1",
    "resolved": "https://registry.npmjs.org/ee-first/-/ee-first-1.1.1.tgz",
    "integrity": "sha512-WMwm9LhRUo+WUaRN+vRuETqG89IgzphVSNkdfgeb6sS/E40rDIN7t48CAewSHXc6C81efD8KKfr5vY61brQ1
ow==",
    "license": "MIT"
  },
  "node_modules/encodeurl": {
    "version": "2.0.0",
    "resolved": "https://registry.npmjs.org/encodeurl/-/encodeurl-2.0.0.tgz",
    "integrity": "sha512-Q0n9HRi4m6JuGIV1eFlmvJB7ZEvXu93IrMyiMsGC0lrMJMwzRgx6WGquyfQgZVb31vhGgXnfmPNNXmxnOkRB
rg==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.8"
    }
  },
  "node_modules/es-define-property": {
    "version": "1.0.1",
    "resolved": "https://registry.npmjs.org/es-define-property/-/es-define-property-1.0.1.tgz",
    "integrity": "sha512-04493Q4J43jaGJ5PhLjRmWw9LNKWY5C6xYpqd6p1dH439a4wTAW6C1kpAE5c6+Zz4fDfyzvevVt4W
e3nRfgfUZ4rNGL232gUgX06QNYyez04KdjFrF+LTRoOXmrOgFKDg4BCdsjW8EnT69eqdYGmRpJwiPVYNrCaW
3g==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.4"
    }
  },
  "node_modules/es-errors": {
    "version": "1.3.0",
    "resolved": "https://registry.npmjs.org/es-errors/-/es-errors-1.3.0.tgz",
    "integrity": "sha512-UwLJ+cAAw7l2nKheMcBqo7d79/4x3H+f0H/XB36n1J6o7dR8yo7EEXhSu3zPn2khyOvF682Qd2N
Zf5H2Kxt2xjTvbJvP2ZWLEICxA6j+hAmMzIlypy4xcBg1vKVnx89Wy0GbS+kf5cwCVFFzdCFh2XSCFNULS6c
sw==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.4"
    }
  },
  "node_modules/es-object-atoms": {
    "version": "1.1.1",
    "resolved": "https://registry.npmjs.org/es-object-atoms/-/es-object-atoms-1.1.1.tgz",

```

44165850.1551

```

    "integrity": "sha512-
FGgH2h8zKNim9lj7dankFPcICIK9Cp5bm+c2gQSYePhpaG5+esrL0DihIorn+Pe6FGJzWhXQotPv73jTald
XA==",
    "license": "MIT",
    "dependencies": {
      "es-errors": "^1.3.0"
    },
    "engines": {
      "node": ">= 0.4"
    }
  },
  "node_modules/escape-html": {
    "version": "1.0.3",
    "resolved": "https://registry.npmjs.org/escape-html/-/escape-html-1.0.3.tgz",
    "integrity": "sha512-
NiSupZ40euGwr68lGIeym/ksIZMJodUGOSCZ/FSnTxcrekbvqrgdUx1J0MpijaKZVjAJrWrGs/6Jy80Muyj9
ow==",
    "license": "MIT"
  },
  "node_modules/etag": {
    "version": "1.8.1",
    "resolved": "https://registry.npmjs.org/etag/-/etag-1.8.1.tgz",
    "integrity": "sha512-
aIL5Fx7mawVa300a12BnEE4iNvo1qETxLrPI/o05L7z6go7fCw1J6EQmbK4FmJ2AS7kgVF/KEZWufBfdC1Mc
Pg==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/express": {
    "version": "4.22.1",
    "resolved": "https://registry.npmjs.org/express/-/express-4.22.1.tgz",
    "integrity": "sha512-
F2X8g9P1X7uCPZMA3Mvf9wcTqlyNp7IhH5qPCI0izha0IYXaW9L535tGA3qmjRzpH+bZczqq7hVKxTR4Nwnu
+g==",
    "license": "MIT",
    "dependencies": {
      "accepts": "~1.3.8",
      "array-flatten": "1.1.1",
      "body-parser": "~1.20.3",
      "content-disposition": "~0.5.4",
      "content-type": "~1.0.4",
      "cookie": "~0.7.1",
      "cookie-signature": "~1.0.6",
      "debug": "2.6.9",
      "depd": "2.0.0",
      "encodeurl": "~2.0.0",
      "escape-html": "~1.0.3",

```

44165850.1551

```

    "etag": "~1.8.1",
    "finalhandler": "~1.3.1",
    "fresh": "~0.5.2",
    "http-errors": "~2.0.0",
    "merge-descriptors": "1.0.3",
    "methods": "~1.1.2",
    "on-finished": "~2.4.1",
    "parseurl": "~1.3.3",
    "path-to-regexp": "~0.1.12",
    "proxy-addr": "~2.0.7",
    "qs": "~6.14.0",
    "range-parser": "~1.2.1",
    "safe-buffer": "5.2.1",
    "send": "~0.19.0",
    "serve-static": "~1.16.2",
    "setprototypeof": "1.2.0",
    "statuses": "~2.0.1",
    "type-is": "~1.6.18",
    "utils-merge": "1.0.1",
    "vary": "~1.1.2"
  },
  "engines": {
    "node": ">= 0.10.0"
  },
  "funding": {
    "type": "opencollective",
    "url": "https://opencollective.com/express"
  }
},
"node_modules/fill-range": {
  "version": "7.1.1",
  "resolved": "https://registry.npmjs.org/fill-range/-/fill-range-7.1.1.tgz",
  "integrity": "sha512-YgGK3WYi4tWdg2Jy3ebRz2rXowDxnld4bkQB00cc/1Zw9AWnC0i9ztDJitvtQvaI9KaLyKrc+hBW0
yg==",
  "dev": true,
  "license": "MIT",
  "dependencies": {
    "to-regex-range": "^5.0.1"
  },
  "engines": {
    "node": ">=8"
  }
},
"node_modules/finalhandler": {
  "version": "1.3.2",
  "resolved": "https://registry.npmjs.org/finalhandler/-/finalhandler-
1.3.2.tgz",

```

44165850.1551

```

    "integrity": "sha512-
aA4RyPcd3badbdABGDuTXCMTtOneUCAYH/gxoYRTZlIJdF0YPWuGqiAsIrhNnnqdXGswYk6dGujem4w80UJF
hg==",
    "license": "MIT",
    "dependencies": {
      "debug": "2.6.9",
      "encodeurl": "~2.0.0",
      "escape-html": "~1.0.3",
      "on-finished": "~2.4.1",
      "parseurl": "~1.3.3",
      "statuses": "~2.0.2",
      "unpipe": "~1.0.0"
    },
    "engines": {
      "node": ">= 0.8"
    }
  },
  "node_modules/forwarded": {
    "version": "0.2.0",
    "resolved": "https://registry.npmjs.org/forwarded/-/forwarded-0.2.0.tgz",
    "integrity": "sha512-
buRG0fpBtRHSTCOASe6hD258tEubFoRLb4ZNA6NxMVHNw2gOcwHo9wyablzMzOA5z9xA9L1KNjk/Nt6MT9aY
ow==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/fresh": {
    "version": "0.5.2",
    "resolved": "https://registry.npmjs.org/fresh/-/fresh-0.5.2.tgz",
    "integrity": "sha512-
zJ2mQYM18rEF0udeV4GShTGIQ7Rbza7OzbU9I/XBpm7kqgMywgmylMwXHxZJmkVoYkna9d2pVXVPdYTP9ej
8Q==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/fsevents": {
    "version": "2.3.3",
    "resolved": "https://registry.npmjs.org/fsevents/-/fsevents-2.3.3.tgz",
    "integrity": "sha512-
5xoDfX+fL7faATnagmWPpbFtwh/R77WmMMqqHGS65C3vvB0YHrgF+B1YmZ3441tMj5n63k0212XNoJwzlhff
Qw==",
    "dev": true,
    "hasInstallScript": true,
    "license": "MIT",
    "optional": true,

```

44165850.1551

```

"os": [
  "darwin"
],
"engines": {
  "node": "^8.16.0 || ^10.6.0 || >=11.0.0"
}
},
"node_modules/function-bind": {
  "version": "1.1.2",
  "resolved": "https://registry.npmjs.org/function-bind/-/function-bind-1.1.2.tgz",
  "integrity": "sha512-7XHNxH7qX9xG5mIwxkhumTox/MIRNcOgDrxWsMt2pAr23WHP6MrR1N7FBSFPcPr+oV00F744iUgR82nJMFg2SA==",
  "license": "MIT",
  "funding": {
    "url": "https://github.com/sponsors/ljharb"
  }
},
"node_modules/get-intrinsic": {
  "version": "1.3.0",
  "resolved": "https://registry.npmjs.org/get-intrinsic/-/get-intrinsic-1.3.0.tgz",
  "integrity": "sha512-9fSjSaos/fRIVIp+xSjLE6lfwhES7LNtKaCBIamHsjr2na1BiABJPo0m0jjz8GJDURarmCPGqaiVg5mfjb98CQ==",
  "license": "MIT",
  "dependencies": {
    "call-bind-apply-helpers": "^1.0.2",
    "es-define-property": "^1.0.1",
    "es-errors": "^1.3.0",
    "es-object-atoms": "^1.1.1",
    "function-bind": "^1.1.2",
    "get-proto": "^1.0.1",
    "gopd": "^1.2.0",
    "has-symbols": "^1.1.0",
    "hasown": "^2.0.2",
    "math-intrinsics": "^1.1.0"
  }
},
"engines": {
  "node": ">= 0.4"
},
"funding": {
  "url": "https://github.com/sponsors/ljharb"
}
},
"node_modules/get-proto": {
  "version": "1.0.1",
  "resolved": "https://registry.npmjs.org/get-proto/-/get-proto-1.0.1.tgz",

```

44165850.1551

```

    "integrity": "sha512-
sTSfBjoXBp89JvIKIefqw7U2CCebSc74kiY6awiGogKtoSGbgjYE/G/+19sF3MWFpNc9Ico0C40DfKHfxFmp
0g==",
    "license": "MIT",
    "dependencies": {
      "dunder-proto": "^1.0.1",
      "es-object-atoms": "^1.0.0"
    },
    "engines": {
      "node": ">= 0.4"
    }
  },
  "node_modules/glob-parent": {
    "version": "5.1.2",
    "resolved": "https://registry.npmjs.org/glob-parent/-/glob-parent-5.1.2.tgz",
    "integrity": "sha512-
AOIgSQcepiJYwP3ARnGx+5VnTu2HBYdzbGP45eLw1vr3zB3vZLeyed1sC9hnbc0c9/SrMyM5RPQrkGz4aS9Z
ow==",
    "dev": true,
    "license": "ISC",
    "dependencies": {
      "is-glob": "^4.0.1"
    },
    "engines": {
      "node": ">= 6"
    }
  },
  "node_modules/gopd": {
    "version": "1.2.0",
    "resolved": "https://registry.npmjs.org/gopd/-/gopd-1.2.0.tgz",
    "integrity": "sha512-
ZUKRh6/kUFoAiAtTYPZJ3hw9wNxx+BIB0ijnlg9PnrJsCcSjs1wyD6vJpaYtgnzDrKYRSqf3006Rfa93xs
Rg==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.4"
    },
    "funding": {
      "url": "https://github.com/sponsors/ljharb"
    }
  },
  "node_modules/has-flag": {
    "version": "3.0.0",
    "resolved": "https://registry.npmjs.org/has-flag/-/has-flag-3.0.0.tgz",
    "integrity": "sha512-
sKJf1+ceQBr4SMkvQnBDNDtf4TXpVhVGateu0t918bl30FnBE2m4vNLX+VWe/dpjlB+HugGYzW7uQXH98HPE
Yw==",
    "dev": true,
    "license": "MIT",

```

44165850.1551

```

    "engines": {
      "node": ">=4"
    }
  },
  "node_modules/has-symbols": {
    "version": "1.1.0",
    "resolved": "https://registry.npmjs.org/has-symbols/-/has-symbols-1.1.0.tgz",
    "integrity": "sha512-1cDNdwJ2Jaohmb3sg40mKaMBwuC48sYni5HUw2DvsC8LjGTLK9h+eb1X6Ryu0He4hT0ULCW68iomhjUoKUq1
PQ==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.4"
    },
    "funding": {
      "url": "https://github.com/sponsors/ljharb"
    }
  },
  "node_modules/hasown": {
    "version": "2.0.2",
    "resolved": "https://registry.npmjs.org/hasown/-/hasown-2.0.2.tgz",
    "integrity": "sha512-186Q035/4u07oN5X0Uy6jKLfzCa4toD5zFvw91Yz6z6zQ81gO9M/ygzZqFoVCS79zYz1dPz7z5z1eZQ==",
    "license": "MIT",
    "dependencies": {
      "function-bind": "^1.1.2"
    },
    "engines": {
      "node": ">= 0.4"
    }
  },
  "node_modules/http-errors": {
    "version": "2.0.1",
    "resolved": "https://registry.npmjs.org/http-errors/-/http-errors-2.0.1.tgz",
    "integrity": "sha512-38y68/493Q0X6O0Wb4aD973u18032558611F108417013b62B726889930110600f1f212278P8L
mQ==",
    "license": "MIT",
    "dependencies": {
      "depd": "~2.0.0",
      "inherits": "~2.0.4",
      "setprototypeof": "~1.2.0",
      "statuses": "~2.0.2",
      "toidentifier": "~1.0.1"
    },
    "engines": {
      "node": ">= 0.8"
    }
  },

```

44165850.1551

```

    "funding": {
      "type": "opencollective",
      "url": "https://opencollective.com/express"
    }
  },
  "node_modules/iconv-lite": {
    "version": "0.4.24",
    "resolved": "https://registry.npmjs.org/iconv-lite/-/iconv-lite-0.4.24.tgz",
    "integrity": "sha512-v3MXnZAcvnywkTUEZomIActle7RXXeedOR31wwl7VlyoX04Qi9arvSenNQWne1TcRwhCL1HwLI21bEqdpj8/rA==",
    "license": "MIT",
    "dependencies": {
      "safer-buffer": ">= 2.1.2 < 3"
    },
    "engines": {
      "node": ">=0.10.0"
    }
  },
  "node_modules/ignore-by-default": {
    "version": "1.0.1",
    "resolved": "https://registry.npmjs.org/ignore-by-default/-/ignore-by-default-1.0.1.tgz",
    "integrity": "sha512-Ius2VYcGNk7T90CppJqcIkS5ooHUZyIQK+ClZfMfMNF9VSE73Fq+906u/CWu92x4gzZMW0wFFYckPObzdEbA==",
    "dev": true,
    "license": "ISC"
  },
  "node_modules/inherits": {
    "version": "2.0.4",
    "resolved": "https://registry.npmjs.org/inherits/-/inherits-2.0.4.tgz",
    "integrity": "sha512-k/vGaX4/Yla3WzyMCvTQOXYeIHvqOKtngBduzTHpzpQZzAskKMhZ2K+EnBiSM9zGSoIFeMpXKxa4dYeZIQqewQ==",
    "license": "ISC"
  },
  "node_modules/ipaddr.js": {
    "version": "1.9.1",
    "resolved": "https://registry.npmjs.org/ipaddr.js/-/ipaddr.js-1.9.1.tgz",
    "integrity": "sha512-0KI/607xoxSTOH7GjN1FfSbLou0+btTicjsQSWQlh/hZyKN8KpmMf7uYwPW3R+akZ6R/w18Z1XSHBYXiYUPO3g==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.10"
    }
  },
  "node_modules/is-binary-path": {

```

44165850.1551

```

    "version": "2.1.0",
    "resolved": "https://registry.npmjs.org/is-binary-path/-/is-binary-path-
2.1.0.tgz",
    "integrity": "sha512-
ZMERYes6pDydyGidse70sHxtbI7WVeUEozgR/g7rd0xUimYNlvZRE/K2MgZTjWy725IfelLeVcEM97mmtRG
Xw==",
    "dev": true,
    "license": "MIT",
    "dependencies": {
      "binary-extensions": "^2.0.0"
    },
    "engines": {
      "node": ">=8"
    }
  },
  "node_modules/is-extglob": {
    "version": "2.1.1",
    "resolved": "https://registry.npmjs.org/is-extglob/-/is-extglob-2.1.1.tgz",
    "integrity": "sha512-
SbKbANkN603Vi4jEZv49LeVJMn4yGwsbzZworEoyEiutsN3nJYdb036zfhGJ6QEDpOZIFkDtnq5JRxmvl3js
oQ==",
    "dev": true,
    "license": "MIT",
    "engines": {
      "node": ">=0.10.0"
    }
  },
  "node_modules/is-glob": {
    "version": "4.0.3",
    "resolved": "https://registry.npmjs.org/is-glob/-/is-glob-4.0.3.tgz",
    "integrity": "sha512-
xelSayHH36ZgE7ZWhli7pw34hNbNl80jv5KVmkJD4hBdD3th8Tfk9vYasLM+mXW0ZhFkgZfxhLSnrwRr4e1S
Sg==",
    "dev": true,
    "license": "MIT",
    "dependencies": {
      "is-extglob": "^2.1.1"
    },
    "engines": {
      "node": ">=0.10.0"
    }
  },
  "node_modules/is-number": {
    "version": "7.0.0",
    "resolved": "https://registry.npmjs.org/is-number/-/is-number-7.0.0.tgz",
    "integrity": "sha512-
41Cifkg6e8Ty1SpdtTpeLVMqvSBEVzTttHvERD741+pnZ8ANv0004MRL43QKPD1K9cGvNp6NZWZUB1bGXYxx
ng==",
    "dev": true,

```

44165850.1551

```

    "license": "MIT",
    "engines": {
      "node": ">=0.12.0"
    }
  },
  "node_modules/math-intrinsics": {
    "version": "1.1.0",
    "resolved": "https://registry.npmjs.org/math-intrinsics/-/math-intrinsics-1.1.0.tgz",
    "integrity": "sha512-IXtbwEk5HTPyEwyKX6hGkYXxM9nbj64B+ilVJnC/R6B0pH5G4V3b0pVbL7DBj4tkhBAppbQUlf6F6X19LHu1g==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.4"
    }
  },
  "node_modules/media-typer": {
    "version": "0.3.0",
    "resolved": "https://registry.npmjs.org/media-typer/-/media-typer-0.3.0.tgz",
    "integrity": "sha512-dq+qe1Q9akHpcO1/gUVRTxVIOkAJ1wR3QAvb4RsVjS8oVoFjDGTc679wJYmUmknUF5HwMLOgb50+a3KxfWapPQ==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/merge-descriptors": {
    "version": "1.0.3",
    "resolved": "https://registry.npmjs.org/merge-descriptors/-/merge-descriptors-1.0.3.tgz",
    "integrity": "sha512-gaNvAS7TZ897/rVaZ0nMtAyxNyi/pdbjbAwUpFQpN70GqnVf0iXpeUUMKRBmzXaSQ8DdTX4/0ms62r2K+hE6mQ==",
    "license": "MIT",
    "funding": {
      "url": "https://github.com/sponsors/sindresorhus"
    }
  },
  "node_modules/methods": {
    "version": "1.1.2",
    "resolved": "https://registry.npmjs.org/methods/-/methods-1.1.2.tgz",
    "integrity": "sha512-cvF3j4r00xWzVojTUp2X7z+Xns5LmR0F08YYo3Ej4jZz3x37+6V0Vqo5wq0y6o3w5+4zPz4BvRZz8w==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  }

```

44165850.1551

```

    }
  },
  "node_modules/mime": {
    "version": "1.6.0",
    "resolved": "https://registry.npmjs.org/mime/-/mime-1.6.0.tgz",
    "integrity": "sha512-x0Vn8spI+wuJ106S7gnbaQg8Pxb4NNHb7KSINmEWKiPE4RKOp1vij+nNkmYmmRgP68mc70j2EbeTFRsrswaQeg==",
    "license": "MIT",
    "bin": {
      "mime": "cli.js"
    },
    "engines": {
      "node": ">=4"
    }
  },
  "node_modules/mime-db": {
    "version": "1.52.0",
    "resolved": "https://registry.npmjs.org/mime-db/-/mime-db-1.52.0.tgz",
    "integrity": "sha512-sPU4uV7dY1vtWJxwwxHD0PuihVNiE7TyAbQ5SWxDCB9mUYv0groQ0wYQQ0KPJ8CIbE+1ETVl0oK1UC2nU3gYvg==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/mime-types": {
    "version": "2.1.35",
    "resolved": "https://registry.npmjs.org/mime-types/-/mime-types-2.1.35.tgz",
    "integrity": "sha512-lNkzj+HEudJaqmP20kj9oZtGjZ/z6eS1V5UZjPg9IY3GwP48GQ3KQWk8bzZ1pHsJQ2Ih3XWJWzW88T5w==",
    "license": "MIT",
    "dependencies": {
      "mime-db": "1.52.0"
    },
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/minimatch": {
    "version": "3.1.2",
    "resolved": "https://registry.npmjs.org/minimatch/-/minimatch-3.1.2.tgz",
    "integrity": "sha512-J7p63hRiAjw1NDEww1W7i37+ByIrOW05XQQAzZ3V0cL0PnybwpfmV/N05zFAzwQ9USyEcX6t3U0+K5aqBQOIHw==",
    "dev": true,
    "license": "ISC",

```

44165850.1551

```

    "dependencies": {
      "brace-expansion": "^1.1.7"
    },
    "engines": {
      "node": "*"
    }
  },
  "node_modules/ms": {
    "version": "2.0.0",
    "resolved": "https://registry.npmjs.org/ms/-/ms-2.0.0.tgz",
    "integrity": "sha512-Tpp60P6IUJDTuOq/5Z8cdskzJujfwqfOTkrwIwj7IRISpnkJnT6SyJ4PCPnGMoFjC9ddhal5KVIYtAt97ix05A==",
    "license": "MIT"
  },
  "node_modules/negotiator": {
    "version": "0.6.3",
    "resolved": "https://registry.npmjs.org/negotiator/-/negotiator-0.6.3.tgz",
    "integrity": "sha512-+EUsqGPLsM+j/zdChZjsnX51g4XrHF0IXwfnCVPGLQk/k5giakcKsuxCObBRu6DSm9opw/O6s1WbJdghQM4bBg==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/nodemon": {
    "version": "3.1.11",
    "resolved": "https://registry.npmjs.org/nodemon/-/nodemon-3.1.11.tgz",
    "integrity": "sha512-0g==",
    "dev": true,
    "license": "MIT",
    "dependencies": {
      "chokidar": "^3.5.2",
      "debug": "^4",
      "ignore-by-default": "^1.0.1",
      "minimatch": "^3.1.2",
      "pstree.remy": "^1.1.8",
      "semver": "^7.5.3",
      "simple-update-notifier": "^2.0.0",
      "supports-color": "^5.5.0",
      "touch": "^3.1.0",
      "undefsafe": "^2.0.5"
    },
    "bin": {
      "nodemon": "bin/nodemon.js"
    }
  },

```

44165850.1551

```

"engines": {
  "node": ">=10"
},
"funding": {
  "type": "opencollective",
  "url": "https://opencollective.com/nodemon"
}
},
"node_modules/nodemon/node_modules/debug": {
  "version": "4.4.3",
  "resolved": "https://registry.npmjs.org/debug/-/debug-4.4.3.tgz",
  "integrity": "sha512-
RGwwWnwQvkVfavKVt22FGLw+xYSdzARwm0ru6DhTVA3umU5hZc28V3k04stgYryrTlLpuv9I9GiiJltAjNbc
qA==",
  "dev": true,
  "license": "MIT",
  "dependencies": {
    "ms": "^2.1.3"
  },
  "engines": {
    "node": ">=6.0"
  },
  "peerDependenciesMeta": {
    "supports-color": {
      "optional": true
    }
  }
},
"node_modules/nodemon/node_modules/ms": {
  "version": "2.1.3",
  "resolved": "https://registry.npmjs.org/ms/-/ms-2.1.3.tgz",
  "integrity": "sha512-
6FlzubTLZG3J2a/NVCA1eEhjzq50xgHyaCU9yYXvcLsvoVaHJq/s5xXI6/XXP6tz7R9xA0tHnSO/tXtF3WRT
lA==",
  "dev": true,
  "license": "MIT"
},
"node_modules/normalize-path": {
  "version": "3.0.0",
  "resolved": "https://registry.npmjs.org/normalize-path/-/normalize-path-
3.0.0.tgz",
  "integrity": "sha512-
6eZs5Ls3WtCisHwP9S2GUy8dqkpGi4BVSz3GaqiE6ezub0512ESztXUwUB6C6IKbQkY2Pnb/mD4WYojCRwcw
lA==",
  "dev": true,
  "license": "MIT",
  "engines": {
    "node": ">=0.10.0"
  }
}

```

44165850.1551

```

    },
    "node_modules/object-inspect": {
      "version": "1.13.4",
      "resolved": "https://registry.npmjs.org/object-inspect/-/object-inspect-1.13.4.tgz",
      "integrity": "sha512-W67iLl4J2EXEgTbfeHCffrjDfitvLANg0U1X3wFUUSTx92KXRFegMHUVgSqE+wwhAbi4WqjGg9czysTV2Epbew==",
      "license": "MIT",
      "engines": {
        "node": ">= 0.4"
      },
      "funding": {
        "url": "https://github.com/sponsors/ljharb"
      }
    },
    "node_modules/on-finished": {
      "version": "2.4.1",
      "resolved": "https://registry.npmjs.org/on-finished/-/on-finished-2.4.1.tgz",
      "integrity": "sha512-OVSSJ7t+UOu9kFNSjF4j0faHIyLhLazW0X34B4I6tqGVy/ooOvLxH7PZ4dW7Kz7XZDlW61Oz82V61Yw==",
      "license": "MIT",
      "dependencies": {
        "ee-first": "1.1.1"
      },
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/parseurl": {
      "version": "1.3.3",
      "resolved": "https://registry.npmjs.org/parseurl/-/parseurl-1.3.3.tgz",
      "integrity": "sha512-CiyeOxFT/JZyN5m0z9PfXw4SCBJ6Sygz1Dp10wqjlhDEGGBP1GnsUVEL0p63hoG1fcj3fHynXi9NY04nWOL+qQ==",
      "license": "MIT",
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/path-to-regexp": {
      "version": "0.1.12",
      "resolved": "https://registry.npmjs.org/path-to-regexp/-/path-to-regexp-0.1.12.tgz",
      "integrity": "sha512-01v3c0/6X7xagEcXupv739ZogEfYpVn8Z06/qP4EgQk86I94uY3nAjv6h+2X0IuOjCuP3E7v5VqNtYg==",
      "license": "MIT"
    }
  }
}

```

44165850.1551

```
  },
  "node_modules/picomatch": {
    "version": "2.3.1",
    "resolved": "https://registry.npmjs.org/picomatch/-/picomatch-2.3.1.tgz",
    "integrity": "sha512-JU3teHTNjmE2VCGFzuY8EXzCDVwEqB2a8fsIvwaStHhAWJEeVd1o1QD80CU6+ZdEXXSLbSsuLwJjkCBWqRQUVA==",
    "dev": true,
    "license": "MIT",
    "engines": {
      "node": ">=8.6"
    },
    },
    "funding": {
      "url": "https://github.com/sponsors/jonschlinkert"
    }
  },
  "node_modules/proxy-addr": {
    "version": "2.0.7",
    "resolved": "https://registry.npmjs.org/proxy-addr/-/proxy-addr-2.0.7.tgz",
    "integrity": "sha512-89jOxkk7xUz1Pdq6SIXkYmb7DKYUoqQSSjUy3lhUJxUhdPoeEUl5NFMDG9Zsh7S3qfWpW37InGBod8QWOqRQAg==",
    "license": "MIT",
    "dependencies": {
      "forwarded": "0.2.0",
      "ipaddr.js": "1.9.1"
    },
    "engines": {
      "node": ">= 0.10"
    }
  },
  "node_modules/pstree.remy": {
    "version": "1.1.8",
    "resolved": "https://registry.npmjs.org/pstree.remy/-/pstree.remy-1.1.8.tgz",
    "integrity": "sha512-7DZwQmxKnu3aR542U+X8FypNzbfJ+C5XQDk3uWjWxn6151aIMGthWYRXTqT1E5oJvg+ljaa20Ji+VfvCOQ8w==",
    "dev": true,
    "license": "MIT"
  },
  "node_modules/qs": {
    "version": "6.14.1",
    "resolved": "https://registry.npmjs.org/qs/-/qs-6.14.1.tgz",
    "integrity": "sha512-71872bA9ZG9Z9zG1Dy2Y/1e3teLzFUB0YQ6F0K4FQoXQ8Q5VUdN3XG1Cg1Dk+yKtDnI8lDn+Vz7uYqQ==",
    "license": "BSD-3-Clause",
    "dependencies": {
      "side-channel": "^1.1.0"
    }
  }
}
```

44165850.1551

```

    },
    "engines": {
      "node": ">=0.6"
    },
    "funding": {
      "url": "https://github.com/sponsors/ljharb"
    }
  },
  "node_modules/range-parser": {
    "version": "1.2.1",
    "resolved": "https://registry.npmjs.org/range-parser/-/range-parser-1.2.1.tgz",
    "integrity": "sha512-Hrgsx+orqoygnmhFbKaHE6c296J+HTAQXoxEF6gNupR0mmGJRozzFG3ccAveqCBwrw/2yxQ5BVd/GT15ag0wSg==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/raw-body": {
    "version": "2.5.3",
    "resolved": "https://registry.npmjs.org/raw-body/-/raw-body-2.5.3.tgz",
    "integrity": "sha512-8GkRrGxUwtaI3UgW2A0xU0LhV13D4Ij0Np0ZLZv0x4O3CNZUGkVd0QeU9A0YgcP0ZG4Mj3Q4vX1zCQ==",
    "license": "MIT",
    "dependencies": {
      "bytes": "~3.1.2",
      "http-errors": "~2.0.1",
      "iconv-lite": "~0.4.24",
      "unpipe": "~1.0.0"
    }
  },
  "engines": {
    "node": ">= 0.8"
  }
},
"node_modules/readdirp": {
  "version": "3.6.0",
  "resolved": "https://registry.npmjs.org/readdirp/-/readdirp-3.6.0.tgz",
  "integrity": "sha512-083300236980988d40209e2d192148a08a5049547c06997856097032c79165e",
  "dev": true,
  "license": "MIT",
  "dependencies": {
    "picomatch": "^2.2.1"
  }
},
  "engines": {

```

44165850.1551

```

      "node": ">=8.10.0"
    }
  },
  "node_modules/safe-buffer": {
    "version": "5.2.1",
    "resolved": "https://registry.npmjs.org/safe-buffer/-/safe-buffer-5.2.1.tgz",
    "integrity": "sha512-rp3So07KcdmmKbGvgaNxQ5Jr7bGVSvk5S9Eq1F+ppbRo70+YeaDxkw5Dd8NPN+GD6bjnYm2VuPuCXmpuYvmCXQ==",
    "funding": [
      {
        "type": "github",
        "url": "https://github.com/sponsors/feross"
      },
      {
        "type": "patreon",
        "url": "https://www.patreon.com/feross"
      },
      {
        "type": "consulting",
        "url": "https://feross.org/support"
      }
    ],
    "license": "MIT"
  },
  "node_modules/safer-buffer": {
    "version": "2.1.2",
    "resolved": "https://registry.npmjs.org/safer-buffer/-/safer-buffer-2.1.2.tgz",
    "integrity": "sha512-YZo3K82SD7Riyi0E1EQPojLz7kpepnSQI9IyPbHHg1XXXevb5dJI7tpyN2ADxGcQbHG7vcyRHk0cbwqcQriUtg==",
    "license": "MIT"
  },
  "node_modules/semver": {
    "version": "7.7.3",
    "resolved": "https://registry.npmjs.org/semver/-/semver-7.7.3.tgz",
    "integrity": "sha512-Yo3K82SD7Riyi0E1EQPojLz7kpepnSQI9IyPbHHg1XXXevb5dJI7tpyN2ADxGcQbHG7vcyRHk0cbwqcQriU0Q==",
    "dev": true,
    "license": "ISC",
    "bin": {
      "semver": "bin/semver.js"
    },
    "engines": {
      "node": ">=10"
    }
  },

```

44165850.1551

```

"node_modules/send": {
  "version": "0.19.2",
  "resolved": "https://registry.npmjs.org/send/-/send-0.19.2.tgz",
  "integrity": "sha512-VMBMxbDeehAxpOtWJXlcUS5E8iXh6QmN+BkRX1GARS3wRaXEEgzCcB10gTQazO42tpNIya8xIyNx8f1110FP
rg==",
  "license": "MIT",
  "dependencies": {
    "debug": "2.6.9",
    "depd": "2.0.0",
    "destroy": "1.2.0",
    "encodeurl": "~2.0.0",
    "escape-html": "~1.0.3",
    "etag": "~1.8.1",
    "fresh": "~0.5.2",
    "http-errors": "~2.0.1",
    "mime": "1.6.0",
    "ms": "2.1.3",
    "on-finished": "~2.4.1",
    "range-parser": "~1.2.1",
    "statuses": "~2.0.2"
  },
  "engines": {
    "node": ">= 0.8.0"
  }
},
"node_modules/send/node_modules/ms": {
  "version": "2.1.3",
  "resolved": "https://registry.npmjs.org/ms/-/ms-2.1.3.tgz",
  "integrity": "sha512-6FlzubTLZG3J2a/NVCAleEhJzq5oxgHyaCU9yYXvcLsvoVaHJq/s5xXI6/XP6tz7R9xA0tHnSO/tXtF3WRT
lA==",
  "license": "MIT"
},
"node_modules/serve-static": {
  "version": "1.16.3",
  "resolved": "https://registry.npmjs.org/serve-static/-/serve-static-
1.16.3.tgz",
  "integrity": "sha512-x0RTqQel6g5SY7Lg6ZreMms0zncHFU7nhnRWkKgWuMTu5NN0DR5oruckMqRvacAN9d5w6ARnRBX19xhDCgfm
eA==",
  "license": "MIT",
  "dependencies": {
    "encodeurl": "~2.0.0",
    "escape-html": "~1.0.3",
    "parseurl": "~1.3.3",
    "send": "~0.19.1"
  },
  "engines": {

```

44165850.1551

```

    "node": ">= 0.8.0"
  }
},
"node_modules/setprototypeof": {
  "version": "1.2.0",
  "resolved": "https://registry.npmjs.org/setprototypeof/-/setprototypeof-
1.2.0.tgz",
  "integrity": "sha512-
E5LDX7Wrp85Kil5bhZv46j8j0eboKq5JMmYM3gVGdGH8xFpPWXUMsNr10DCrkoXMEeNi/XZIwuRvY4XNwYMJ
pw==",
  "license": "ISC"
},
"node_modules/side-channel": {
  "version": "1.1.0",
  "resolved": "https://registry.npmjs.org/side-channel/-/side-channel-
1.1.0.tgz",
  "integrity": "sha512-
ZX99e6tRweoUXqR+VBrs1hda51Nh5MTQwou5tnUDgbtyM0dBgmhEDtWGP/xbKn6hqfPRHujUNwz5fy/wbbhn
pw==",
  "license": "MIT",
  "dependencies": {
    "es-errors": "^1.3.0",
    "object-inspect": "^1.13.3",
    "side-channel-list": "^1.0.0",
    "side-channel-map": "^1.0.1",
    "side-channel-weakmap": "^1.0.2"
  },
  "engines": {
    "node": ">= 0.4"
  },
  "funding": {
    "url": "https://github.com/sponsors/ljharb"
  }
},
"node_modules/side-channel-list": {
  "version": "1.0.0",
  "resolved": "https://registry.npmjs.org/side-channel-list/-/side-channel-list-
1.0.0.tgz",
  "integrity": "sha512-
FCLHtRD/gnpCiCHEiJL0wdmFP+wzCmDEkc9y7NsYxeF4u7Btsn1ZuwgwJGxImImHicJArLP4R0yX4c2KCrMr
TA==",
  "license": "MIT",
  "dependencies": {
    "es-errors": "^1.3.0",
    "object-inspect": "^1.13.3"
  },
  "engines": {
    "node": ">= 0.4"
  },
},

```

44165850.1551

```

    "funding": {
      "url": "https://github.com/sponsors/ljharb"
    }
  },
  "node_modules/side-channel-map": {
    "version": "1.0.1",
    "resolved": "https://registry.npmjs.org/side-channel-map/-/side-channel-map-1.0.1.tgz",
    "integrity": "sha512-VcJCNfMsbY3tTdo02nbjtM/ewra6jPHmpThenkTYh8pG9ucZ/1P8So4u4FGBek/BjpOVsDCMoLA/iuBKIFXRA==",
    "license": "MIT",
    "dependencies": {
      "call-bound": "^1.0.2",
      "es-errors": "^1.3.0",
      "get-intrinsic": "^1.2.5",
      "object-inspect": "^1.13.3"
    },
    "engines": {
      "node": ">= 0.4"
    },
    "funding": {
      "url": "https://github.com/sponsors/ljharb"
    }
  },
  "node_modules/side-channel-weakmap": {
    "version": "1.0.2",
    "resolved": "https://registry.npmjs.org/side-channel-weakmap/-/side-channel-weakmap-1.0.2.tgz",
    "integrity": "sha512-WPS/HvHQTYnHisLo9McqBH0Jk2FkH0/tlpvldyrnem4aeQp4hai3gythswg6p01oSoTl58rcpiFAjF2br2Ak2A==",
    "license": "MIT",
    "dependencies": {
      "call-bound": "^1.0.2",
      "es-errors": "^1.3.0",
      "get-intrinsic": "^1.2.5",
      "object-inspect": "^1.13.3",
      "side-channel-map": "^1.0.1"
    },
    "engines": {
      "node": ">= 0.4"
    },
    "funding": {
      "url": "https://github.com/sponsors/ljharb"
    }
  },
  "node_modules/simple-update-notifier": {
    "version": "2.0.0",

```

44165850.1551

```

    "resolved": "https://registry.npmjs.org/simple-update-notifier/-/simple-
update-notifier-2.0.0.tgz",
    "integrity": "sha512-
a2B9Y0K1NX19u/vsW6sTIu9vGEpfKu2wRV6l1H3XEas/0gUIzGzBoP/IouTcUQbm9JWZLH3COxyn03TY1Fax
6w==",
    "dev": true,
    "license": "MIT",
    "dependencies": {
      "semver": "^7.5.3"
    },
    "engines": {
      "node": ">=10"
    }
  },
  "node_modules/statuses": {
    "version": "2.0.2",
    "resolved": "https://registry.npmjs.org/statuses/-/statuses-2.0.2.tgz",
    "integrity": "sha512-
DvEy55V3DB7uknRo+4i0GT5fP1s1R8wQohVdknigZPMpMstaKJQWhwiYBACJE3U12pTnATihhBYnRhZQHGBi
Rw==",
    "license": "MIT",
    "engines": {
      "node": ">= 0.8"
    }
  },
  "node_modules/supports-color": {
    "version": "5.5.0",
    "resolved": "https://registry.npmjs.org/supports-color/-/supports-color-
5.5.0.tgz",
    "integrity": "sha512-
QjVjwdXIt408MIiAqCX4oUKsgU2EqAGzs2Ppkm4aQYbjm+ZEWECw4SfFNTr4uMNZma0ey4f51gLRk0aX0QM
ow==",
    "dev": true,
    "license": "MIT",
    "dependencies": {
      "has-flag": "^3.0.0"
    },
    "engines": {
      "node": ">=4"
    }
  },
  "node_modules/to-regex-range": {
    "version": "5.0.1",
    "resolved": "https://registry.npmjs.org/to-regex-range/-/to-regex-range-
5.0.1.tgz",
    "integrity": "sha512-
65P7iz6X5yEr1cwcgvQxbbIw7Uk3gOy5dIdtZ4rDveLqhrdJP+Li/Hx6tyK0NEb+2GCyneCMJiGqrADCSNk8
sQ==",
    "dev": true,

```

44165850.1551

```

    "license": "MIT",
    "dependencies": {
      "is-number": "^7.0.0"
    },
    "engines": {
      "node": ">=8.0"
    }
  },
  "node_modules/toidentifier": {
    "version": "1.0.1",
    "resolved": "https://registry.npmjs.org/toidentifier/-/toidentifier-1.0.1.tgz",
    "integrity": "sha512-o5sSPKEkg/DIQNmH43V0/uerLrpzVedkUh8tGNvaeXpfpuwjKenlSox/20/BTlZUtEe+JG7s5YhEz608P1AHRA==",
    "license": "MIT",
    "engines": {
      "node": ">=0.6"
    }
  },
  "node_modules/touch": {
    "version": "3.1.1",
    "resolved": "https://registry.npmjs.org/touch/-/touch-3.1.1.tgz",
    "integrity": "sha512-r0eojU4bI8MnHr8c5bNo7lJDDdI2qXlWJk6a9EAFG7vbhTjE1YhBVS3/miuE0uOuoLdb8Mc/rVfsmm6eo5o9GA==",
    "dev": true,
    "license": "ISC",
    "bin": {
      "nodetouch": "bin/nodetouch.js"
    }
  },
  "node_modules/type-is": {
    "version": "1.6.18",
    "resolved": "https://registry.npmjs.org/type-is/-/type-is-1.6.18.tgz",
    "integrity": "sha512-TkRKr9sUTxEH8MdfuCSP7VizJyzRNMjj2J2do2Jr3Kym598JVdEksuzPQCn1FPW4ky9Q+iA+ma9BGm06XQBy8g==",
    "license": "MIT",
    "dependencies": {
      "media-typer": "0.3.0",
      "mime-types": "~2.1.24"
    },
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/undefsafe": {
    "version": "2.0.5",

```



44165850.1551

```

const PORT = 3002;

app.use(express.json());
app.use(express.static("public"));

// API endpoints
app.get("/api/products", (req, res) => {
  const page = req.query.page || 1;
  const itemsPerPage = 30;

  const products = Array(itemsPerPage)
    .fill(0)
    .map( (_, i) => ({
      id: parseInt(page) * itemsPerPage + i,
      name: `Premium Product ${parseInt(page) * itemsPerPage + i + 1}`,
      description: "High-quality item with excellent reviews",
      price: Math.floor(Math.random() * 200) + 20,
      category: ["Electronics", "Fashion", "Home",
        "Sports"][Math.floor(Math.random() * 4)],
      image: `https://picsum.photos/300/300?random=${parseInt(page) * itemsPerPage +
        i}`,
      rating: (Math.random() * 2 + 3).toFixed(1),
      inStock: Math.random() > 0.3,
    }));

  res.json({
    page: parseInt(page),
    items: products,
    total: 500,
    hasMore: parseInt(page) < 20,
  });
});

app.get("/api/articles", (req, res) => {
  const page = req.query.page || 1;
  const itemsPerPage = 12;

  const articles = Array(itemsPerPage)
    .fill(0)
    .map( (_, i) => ({
      id: parseInt(page) * itemsPerPage + i,
      title: `Technical Deep Dive: Web Performance Article ${parseInt(page) *
        itemsPerPage + i + 1}`,
      excerpt:
        "Comprehensive guide to optimizing web application performance across modern
        browsers and network conditions.",
      author: `Tech Author ${Math.floor(Math.random() * 10) + 1}`,
      date: new Date(Date.now() - Math.random() * 30 * 24 * 60 * 60 *
        1000).toLocaleDateString(),
    }));

```

44165850.1551

```

        readTime: Math.floor(Math.random() * 10) + 3,
        category: ["Performance", "Frontend", "DevOps", "Security", "Best
Practices"][Math.floor(Math.random() * 5)],
        views: Math.floor(Math.random() * 50000),
        image: `https://picsum.photos/400/250?random=${parseInt(page) * itemsPerPage +
i}` ,
    }));

    res.json({
        page: parseInt(page),
        articles,
        total: 150,
    });
});

app.get("/api/team", (req, res) => {
    const roles = [
        "Frontend Developer",
        "Backend Developer",
        "UI/UX Designer",
        "DevOps Engineer",
        "Product Manager",
        "QA Engineer",
        "Scrum Master",
    ];

    const team = Array(15)
        .fill(0)
        .map((_, i) => ({
            id: i,
            name: `Team Member ${i + 1}`,
            role: roles[i % roles.length],
            email: `member${i}@example.com`,
            experience: Math.floor(Math.random() * 10) + 1,
            specialties: ["React", "TypeScript", "Node.js", "AWS",
"Docker"][Math.floor(Math.random() * 5)],
            avatar: `https://i.pravatar.cc/150?img=${i}`,
        }));

    res.json({ team });
});

app.get("/api/metrics", (req, res) => {
    res.json({
        timestamp: new Date().toISOString(),
        metrics: {
            ttfb: Math.floor(Math.random() * 200) + 100,
            fcp: Math.floor(Math.random() * 1000) + 800,
            lcp: Math.floor(Math.random() * 1500) + 1200,
        }
    });
});

```

44165850.1551

```

    cls: (Math.random() * 0.2).toFixed(3),
    tti: Math.floor(Math.random() * 2000) + 1500,
    totalSize: Math.floor(Math.random() * 2000) + 500,
  },
});
});

// Serve HTML files
app.get("/", (req, res) => {
  res.sendFile(path.join(__dirname, "public", "lazy-loading.html"));
});

app.get("/lazy-loading", (req, res) => {
  res.sendFile(path.join(__dirname, "public", "lazy-loading.html"));
});

app.get("/preloading", (req, res) => {
  res.sendFile(path.join(__dirname, "public", "preloading.html"));
});

app.get("/prefetching", (req, res) => {
  res.sendFile(path.join(__dirname, "public", "prefetching.html"));
});

app.listen(PORT, () => {
  console.log(`
  ┌────────────────────────────────────────────────────────────────────────────────┐
  │ URL Audit Demo Server Running                                             │
  └────────────────────────────────────────────────────────────────────────────────┘

Server started on http://localhost:${PORT}

Available routes:
  http://localhost:${PORT}/lazy-loading   (Lazy Loading Demo)
  http://localhost:${PORT}/preloading    (Preloading Demo)
  http://localhost:${PORT}/prefetching   (Prefetching Demo)

API Endpoints:
  GET /api/products?page=1   (Product catalog)
  GET /api/articles?page=1   (Blog articles)
  GET /api/team              (Team members)
  GET /api/metrics           (Performance metrics)
  `);
});

module.exports = app;

```

44165850.1551

## public/lazy-loading.html

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Lazy Loading - E-Commerce Gallery</title>
    <style>
      * {
        margin: 0;
        padding: 0;
        box-sizing: border-box;
      }

      body {
        font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
        background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
        min-height: 100vh;
        padding: 40px 20px;
      }

      header {
        text-align: center;
        color: white;
        margin-bottom: 50px;
      }

      h1 {
        font-size: 2.5rem;
        margin-bottom: 10px;
        text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.3);
      }

      .strategy-badge {
        display: inline-block;
        background: rgba(255, 255, 255, 0.2);
        padding: 8px 16px;
        border-radius: 20px;
        font-size: 0.9rem;
        backdrop-filter: blur(10px);
      }

      .metrics {
        background: rgba(255, 255, 255, 0.1);
        color: white;
        padding: 15px 20px;
        border-radius: 10px;
        margin-bottom: 30px;
        max-width: 600px;

```

44165850.1551

```
margin-left: auto;
margin-right: auto;
}

.metrics p {
margin: 5px 0;
font-size: 0.9rem;
}

.gallery {
display: grid;
grid-template-columns: repeat(auto-fill, minmax(250px, 1fr));
gap: 25px;
max-width: 1200px;
margin: 0 auto;
}

.product-card {
background: white;
border-radius: 12px;
overflow: hidden;
box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
transition: transform 0.3s ease, box-shadow 0.3s ease;
cursor: pointer;
}

.product-card:hover {
transform: translateY(-10px);
box-shadow: 0 15px 40px rgba(0, 0, 0, 0.3);
}

.product-image {
width: 100%;
height: 250px;
background: linear-gradient(135deg, #e0e0e0 0%, #f5f5f5 100%);
display: flex;
align-items: center;
justify-content: center;
overflow: hidden;
position: relative;
}

.product-image img {
width: 100%;
height: 100%;
object-fit: cover;
transition: opacity 0.3s ease;
}
```

44165850.1551

```
.placeholder {
  position: absolute;
  width: 100%;
  height: 100%;
  background: linear-gradient(90deg, #e0e0e0 25%, #f0f0f0 50%, #e0e0e0 75%);
  background-size: 200% 100%;
  animation: shimmer 1.5s infinite;
}

@keyframes shimmer {
  0% {
    background-position: 200% 0;
  }
  100% {
    background-position: -200% 0;
  }
}

.product-info {
  padding: 20px;
}

.product-name {
  font-size: 1.1rem;
  font-weight: 600;
  color: #333;
  margin-bottom: 8px;
}

.product-description {
  font-size: 0.85rem;
  color: #666;
  line-height: 1.4;
  margin-bottom: 12px;
}

.product-price {
  font-size: 1.3rem;
  color: #667eea;
  font-weight: 700;
}

.loading-indicator {
  text-align: center;
  padding: 40px;
  color: white;
  font-size: 1.1rem;
}
```

44165850.1551

```

.spinner {
  display: inline-block;
  width: 40px;
  height: 40px;
  border: 4px solid rgba(255, 255, 255, 0.3);
  border-top: 4px solid white;
  border-radius: 50%;
  animation: spin 1s linear infinite;
  margin-bottom: 20px;
}

@keyframes spin {
  to {
    transform: rotate(360deg);
  }
}

footer {
  text-align: center;
  color: rgba(255, 255, 255, 0.7);
  margin-top: 60px;
  padding-top: 30px;
  border-top: 1px solid rgba(255, 255, 255, 0.2);
}
</style>
</head>
<body>
<header>
<h1>Lazy Loading Demo</h1>
<span class="strategy-badge">Intersection Observer API</span>
<div class="metrics">
  <p><strong>Strategy:</strong> Images load only when entering viewport</p>
  <p><strong>Initial Load:</strong> No image resources loaded</p>
  <p><strong>On Scroll:</strong> Images requested dynamically</p>
</div>
</header>

<main class="gallery" id="gallery"></main>

<div class="loading-indicator" id="loading" style="display: none">
  <div class="spinner"></div>
  <p>Loading more products...</p>
</div>

<footer>
  <p>© 2025 URL Audit Demo - Lazy Loading Strategy</p>
</footer>

<script>

```

44165850.1551

```
const PRODUCTS_PER_PAGE = 30;
const API_URL = "/api/products";
let currentPage = 1;
let isLoading = false;

const generateProducts = (page) => {
  const products = [];
  const startId = (page - 1) * PRODUCTS_PER_PAGE;

  for (let i = 0; i < PRODUCTS_PER_PAGE; i++) {
    products.push({
      id: startId + i,
      name: `Premium Product ${startId + i + 1}`,
      description: `High-quality item with excellent reviews and features`,
      price: Math.floor(Math.random() * 200) + 20,
      image: `https://picsum.photos/300/300?random=${startId + i}`,
    });
  }
  return products;
};

const createProductCard = (product) => {
  const card = document.createElement("div");
  card.className = "product-card";

  card.innerHTML = `
<div class="product-image">
  <div class="placeholder"></div>
  
</div>
<div class="product-info">
  <div class="product-name">${product.name}</div>
  <div class="product-description">${product.description}</div>
  <div class="product-price">${product.price}</div>
</div>
`;

  return card;
};

const lazyLoadImages = () => {
  const images = document.querySelectorAll("img[data-src]");

  const imageObserver = new IntersectionObserver(
    (entries, observer) => {
```

44165850.1551

```

    entries.forEach((entry) => {
      if (entry.isIntersecting) {
        const img = entry.target;
        img.src = img.dataset.src;
        img.removeAttribute("data-src");

        img.onload = () => {
          const placeholder =
img.parentElement.querySelector(".placeholder");
          if (placeholder) placeholder.style.display = "none";
        };

        observer.unobserve(img);
      }
    });
  },
  { rootMargin: "50px" }
);

images.forEach((img) => imageObserver.observe(img));
};

const loadProducts = async () => {
  const gallery = document.getElementById("gallery");
  const products = generateProducts(currentPage);

  products.forEach((product) => {
    gallery.appendChild(createProductCard(product));
  });

  lazyLoadImages();
  currentPage++;
};

const loadMoreOnScroll = () => {
  if (window.innerHeight + window.scrollY >= document.body.offsetHeight - 500)
{
  if (!isLoading) {
    isLoading = true;
    document.getElementById("loading").style.display = "block";

    setTimeout(() => {
      loadProducts();
      isLoading = false;
      document.getElementById("loading").style.display = "none";
    }, 500);
  }
}
};

```

44165850.1551

```

        loadProducts();
        window.addEventListener("scroll", loadMoreOnScroll);
    </script>
</body>
</html>

```

## Prefetching.html

```

<!DOCTYPE html>
<html lang="en">
  <head>

    <link
href="https://fonts.googleapis.com/css2?family=JetBrains+Mono:wght@400;600&display=swap" rel="stylesheet" />
    <link
      href="https://fonts.googleapis.com/css2?family=Roboto:wght@100;300;400;500;700;900&family=Roboto+Condensed:wght@300;400;700&family=Roboto+Slab:wght@100;300;400;500;700;900&display=swap"
      rel="stylesheet"
    />
    <link
      href="https://fonts.googleapis.com/css2?family=Open+Sans:wght@300;400;500;600;700;800&family=Open+Sans+Condensed:wght@300;700&display=swap"
      rel="stylesheet"
    />
    <link
      href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@400;500;600;700;800;900&family=Playfair+Display+SC:wght@400;700;900&display=swap"
      rel="stylesheet"
    />
    <link
      href="https://fonts.googleapis.com/css2?family=Montserrat:wght@100;200;300;400;500;600;700;800;900&display=swap"
      rel="stylesheet"
    />
    <link
href="https://fonts.googleapis.com/css2?family=Lato:wght@100;300;400;700;900&display=swap" rel="stylesheet" />
    <link
      href="https://fonts.googleapis.com/css2?family=Merriweather:wght@300;400;700;900&family=Merriweather+Sans:wght@300;400;700;800&display=swap"
      rel="stylesheet"
    />
    <link
href="https://fonts.googleapis.com/css2?family=Oswald:wght@200;300;400;500;600;700&display=swap" rel="stylesheet" />
    <link

```

44165850.1551

```
    href="https://fonts.googleapis.com/css2?family=Source+Sans+Pro:wght@200;300;400;600;700;900&family=Source+Serif+Pro:wght@400;600;700&display=swap"
    rel="stylesheet"
  />
  <link
    href="https://fonts.googleapis.com/css2?family=Raleway:wght@100;200;300;400;500;600;700;800;900&display=swap"
    rel="stylesheet"
  />
  <link
    href="https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;500;600;700;800;900&display=swap"
    rel="stylesheet"
  />
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Prefetching - Multi-Page App</title>

  <style>
    * {
      margin: 0;
      padding: 0;
      box-sizing: border-box;
    }

    body {
      font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
      background: #f5f5f5;
      color: #333;
    }

    header {
      background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
      color: white;
      padding: 20px 40px;
      box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
      position: sticky;
      top: 0;
      z-index: 1000;
    }

    header h1 {
      font-size: 1.8rem;
      margin-bottom: 10px;
    }

    .strategy-badge {
      display: inline-block;
      background: rgba(255, 255, 255, 0.2);
```

44165850.1551

```
padding: 6px 12px;
border-radius: 20px;
font-size: 0.85rem;
backdrop-filter: blur(10px);
}

nav {
  display: flex;
  gap: 20px;
  margin-top: 15px;
  flex-wrap: wrap;
}

nav a,
nav button {
  background: rgba(255, 255, 255, 0.15);
  color: white;
  border: 1px solid rgba(255, 255, 255, 0.3);
  padding: 10px 20px;
  border-radius: 6px;
  cursor: pointer;
  text-decoration: none;
  transition: all 0.3s;
  font-size: 0.9rem;
  font-weight: 500;
}

nav a:hover,
nav button:hover,
nav a.active,
nav button.active {
  background: rgba(255, 255, 255, 0.25);
  transform: translateY(-2px);
  box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);
}

.container {
  max-width: 1200px;
  margin: 0 auto;
  padding: 40px 20px;
}

.info-box {
  background: white;
  padding: 20px;
  border-radius: 8px;
  margin-bottom: 30px;
  border-left: 4px solid #667eea;
  box-shadow: 0 2px 8px rgba(0, 0, 0, 0.05);
}
```

44165850.1551

```
}

.info-box h3 {
  color: #667eea;
  margin-bottom: 10px;
}

.page-section {
  display: none;
  animation: fadeIn 0.3s ease-in;
}

.page-section.active {
  display: block;
}

@keyframes fadeIn {
  from {
    opacity: 0;
    transform: translateY(10px);
  }
  to {
    opacity: 1;
    transform: translateY(0);
  }
}

.grid {
  display: grid;
  grid-template-columns: repeat(auto-fill, minmax(300px, 1fr));
  gap: 25px;
  margin-bottom: 30px;
}

.card {
  background: white;
  border-radius: 8px;
  padding: 25px;
  box-shadow: 0 2px 8px rgba(0, 0, 0, 0.1);
  transition: transform 0.3s, box-shadow 0.3s;
}

.card:hover {
  transform: translateY(-5px);
  box-shadow: 0 5px 15px rgba(0, 0, 0, 0.15);
}

.card h3 {
  color: #667eea;
}
```

44165850.1551

```
    margin-bottom: 12px;
    font-size: 1.3rem;
}

.card p {
    color: #666;
    line-height: 1.6;
    margin-bottom: 15px;
}

.card-meta {
    display: flex;
    gap: 10px;
    font-size: 0.85rem;
    color: #999;
}

.badge {
    background: #f0f0f0;
    padding: 4px 8px;
    border-radius: 4px;
}

h2 {
    color: #333;
    margin-bottom: 20px;
    font-size: 2rem;
    border-bottom: 2px solid #667eea;
    padding-bottom: 10px;
}

.loading {
    text-align: center;
    padding: 40px;
    color: #666;
}

.spinner {
    display: inline-block;
    width: 40px;
    height: 40px;
    border: 4px solid #f0f0f0;
    border-top: 4px solid #667eea;
    border-radius: 50%;
    animation: spin 1s linear infinite;
}

@keyframes spin {
    to {
```

44165850.1551

```
    transform: rotate(360deg);
  }
}

.cache-indicator {
  background: #e8f5e9;
  border-left: 4px solid #4caf50;
  padding: 12px 15px;
  margin: 15px 0;
  border-radius: 4px;
  color: #2e7d32;
  font-size: 0.9rem;
}

.cache-indicator.fetched {
  background: #fff3e0;
  border-left-color: #ff9800;
  color: #e65100;
}

footer {
  text-align: center;
  padding: 40px 20px;
  color: #999;
  border-top: 1px solid #ddd;
  margin-top: 60px;
}

@media (max-width: 768px) {
  header h1 {
    font-size: 1.3rem;
  }

  nav {
    gap: 10px;
  }

  nav a,
  nav button {
    padding: 8px 12px;
    font-size: 0.8rem;
  }

  .grid {
    grid-template-columns: 1fr;
  }
}
</style>
</head>
```

44165850.1551

```

<body>
  <header>
    <h1>Prefetching Demo</h1>
    <span class="strategy-badge">Link Rel Prefetch</span>
    <nav>
      <button class="nav-btn active" data-page="home"> Home</button>
      <button class="nav-btn" data-page="products">Products</button>
      <button class="nav-btn" data-page="articles">Articles</button>
      <button class="nav-btn" data-page="team">Team</button>
      <button class="nav-btn" data-page="contact">Contact</button>
    </nav>
  </header>

  <div class="container">
    <div class="info-box">
      <h3>Prefetching Strategy</h3>
      <p>
        Resources for likely next pages are prefetched during idle time using
        <code>link rel="prefetch"</code>. This
        improves perceived performance when users navigate between pages.
      </p>
      <p><strong>Cache Status:</strong> <span id="cache-
status">Monitoring...</span></p>
    </div>

    <!-- HOME PAGE -->
    <section class="page-section active" id="home">
      <h2>Welcome to Prefetch Demo</h2>

      <div class="cache-indicator">✓ Home page cached (current)</div>

      <div class="info-box">
        <h3>What is Prefetching?</h3>
        <p>
          Prefetching is a low-priority hint that tells the browser about
          resources the user might need in the
          future. Unlike preload, prefetch loads resources during browser idle
          time, so it doesn't compete with
          critical resources.
        </p>
        <p>This demo uses ResourceTiming API to detect which pages have been
        prefetched into the browser cache.</p>
      </div>

      <h3 style="margin-top: 30px; color: #667eea">Key Advantages of
      Prefetching</h3>

      <div class="grid">
        <div class="card">

```

44165850.1551

```

    <h3>Non-Blocking</h3>
    <p>Prefetch requests have the lowest priority and don't block critical
resources or user interactions.</p>
  </div>

  <div class="card">
    <h3>Smart Navigation</h3>
    <p>Predict user behavior and prefetch next page resources based on link
hover or user patterns.</p>
  </div>

  <div class="card">
    <h3>Cache-Aware</h3>
    <p>Browser respects HTTP cache headers, so prefetch won't re-download
recently cached resources.</p>
  </div>

  <div class="card">
    <h3>Analytics-Driven</h3>
    <p>Use analytics to identify most-visited next pages and prefetch those
resources intelligently.</p>
  </div>

  <div class="card">
    <h3>Cross-Domain</h3>
    <p>Prefetch works across domains, allowing CDN resources and third-party
APIs to be prepped.</p>
  </div>

  <div class="card">
    <h3>Bandwidth-Safe</h3>
    <p>Check user's connection speed with Network Information API before
prefetching large assets.</p>
  </div>
</div>

<h3 style="margin-top: 30px; color: #667eea">Implementation Details</h3>

<div class="info-box">
  <h3>Hover-Based Prefetching</h3>
  <p>
    This demo implements smart prefetching: when you hover over a navigation
link, it prefetches all resources
    for that page (HTML, images, stylesheets). If you click the link
quickly, the resources are already in the
    cache!
  </p>
</div>

```

44165850.1551

```

<div class="info-box">
  <h3>Dynamic Prefetch Example</h3>
  <pre
    style="background: #f5f5f5; padding: 12px; border-radius: 4px; overflow-
x: auto; font-size: 0.85rem"
    ><code>
document.querySelectorAll('nav a').forEach(link => {
  link.addEventListener('mouseenter', () => {
    fetch(link.href, { priority: 'low' })
      .then(r => r.text());
  });
});</code></pre>
</div>
</section>

<section class="page-section" id="products">
  <h2>Products Catalog</h2>

  <div class="cache-indicator fetched" id="products-cache">Loading
products...</div>

  <div class="grid" id="products-grid">
    <div class="loading">
      <div class="spinner"></div>
      <p>Simulating product data fetch...</p>
    </div>
  </div>
</section>

<section class="page-section" id="articles">
  <h2>Blog Articles</h2>

  <div class="cache-indicator fetched" id="articles-cache">Loading
articles...</div>

  <div class="grid" id="articles-grid">
    <div class="loading">
      <div class="spinner"></div>
      <p>Simulating article data fetch...</p>
    </div>
  </div>
</section>

<section class="page-section" id="team">
  <h2>Our Team</h2>

  <div class="cache-indicator fetched" id="team-cache">Loading team
members...</div>

```

44165850.1551

```

<div class="grid" id="team-grid">
  <div class="loading">
    <div class="spinner"></div>
    <p>Simulating team data fetch...</p>
  </div>
</div>
</section>

<section class="page-section" id="contact">
  <h2> Get In Touch</h2>

  <div class="cache-indicator fetched" id="contact-cache"> Loading contact
form...</div>

  <div style="max-width: 600px">
    <div class="card">
      <h3>Contact Information</h3>
      <p><strong>Email:</strong> demo@ust.com</p>
      <p><strong>Phone:</strong> +380991283123</p>
      <p><strong>Address:</strong> вул. Лазаряна, 2, Дніпро</p>
    </div>

    <form style="margin-top: 30px" onsubmit="e => e.preventDefault()">
      <div style="margin-bottom: 15px">
        <label style="display: block; margin-bottom: 5px; font-weight:
500">Name</label>
        <input
          type="text"
          placeholder="Your name"
          style="width: 100%; padding: 10px; border: 1px solid #ddd; border-
radius: 4px"
        />
      </div>
      <div style="margin-bottom: 15px">
        <label style="display: block; margin-bottom: 5px; font-weight:
500">Email</label>
        <input
          type="email"
          placeholder="your@email.com"
          style="width: 100%; padding: 10px; border: 1px solid #ddd; border-
radius: 4px"
        />
      </div>
      <div style="margin-bottom: 15px">
        <label style="display: block; margin-bottom: 5px; font-weight:
500">Message</label>
        <textarea
          placeholder="Your message..."
          rows="5"

```

44165850.1551

```

        style="width: 100%; padding: 10px; border: 1px solid #ddd; border-
radius: 4px"
        ></textarea>
    </div>
    <button
    type="submit"
    style="
        background: #667eea;
        color: white;
        padding: 12px 30px;
        border: none;
        border-radius: 4px;
        cursor: pointer;
        font-weight: 500;
    "
    >
        Send Message
    </button>
</form>
</div>
</section>
</div>

<footer>
    <p>© 2025 URL Audit Demo - Prefetching Strategy</p>
    <p style="margin-top: 10px; font-size: 0.9rem">Hover over navigation links to
trigger prefetching during idle time</p>
</footer>

<link rel="prefetch" href="/?page=products" />
<link rel="prefetch" href="/?page=articles" />
<link rel="prefetch" href="/?page=team" />
<link rel="prefetch" href="/?page=contact" />

<script>
    const pages = {
        home: {
            title: "Home",
            loadTime: 0,
        },
        products: {
            title: "Products",
            items: () => generateProducts(15),
        },
        articles: {
            title: "Articles",
            items: () => generateArticles(12),
        },
        team: {

```

44165850.1551

```

    title: "Team",
    items: () => generateTeam(8),
  },
  contact: {
    title: "Contact",
    items: () => [],
  },
};

function generateProducts(count) {
  return Array(count)
    .fill(0)
    .map((_, i) => ({
      id: i,
      title: `Premium Product ${i + 1}`,
      description: "High-quality item with excellent features and customer
reviews",
      price: `$$${(Math.random() * 300 + 20).toFixed(2)}`,
      badge: ["New", "Sale", "Popular", "Limited"][Math.floor(Math.random() *
4)],
    }));
}

function generateArticles(count) {
  return Array(count)
    .fill(0)
    .map((_, i) => ({
      id: i,
      title: `Article: Performance Tips ${i + 1}`,
      excerpt: "Discover best practices for optimizing web application
performance and user experience...",
      author: "Tech Team",
      date: "2 days ago",
      readTime: `${Math.floor(Math.random() * 10) + 3} min read`,
    }));
}

function generateTeam(count) {
  const roles = [
    "Frontend Developer",
    "Backend Developer",
    "UI Designer",
    "Product Manager",
    "DevOps",
    "QA Engineer",
  ];
  return Array(count)
    .fill(0)
    .map((_, i) => ({

```

44165850.1551

```

        id: i,
        name: `Team Member ${i + 1}`,
        role: roles[i % roles.length],
        bio: "Passionate about building amazing web experiences",
    }));
}

function renderPage(pageName) {
    document.querySelectorAll(".page-section").forEach((s) =>
s.classList.remove("active"));

    const section = document.getElementById(pageName);
    section.classList.add("active");

    if (pageName !== "home") {
        loadPageContent(pageName);
    }

    document.querySelectorAll(".nav-btn").forEach((btn) => {
        btn.classList.remove("active");
        if (btn.dataset.page === pageName) btn.classList.add("active");
    });
}

function loadPageContent(pageName) {
    const page = pages[pageName];
    let gridId = `${pageName}-grid`;
    const grid = document.getElementById(gridId);
    const cacheStatus = document.getElementById(`${pageName}-cache`);

    if (!grid) return;

    const startTime = performance.now();

    setTimeout(() => {
        const items = page.items();
        const endTime = performance.now();
        const loadTime = Math.round(endTime - startTime);

        const isCached = loadTime < 200;

        cacheStatus.textContent = isCached
        ? `✓ ${page.title} loaded from cache (${loadTime}ms)`
        : `✓ ${page.title} fetched fresh (${loadTime}ms)`;
        cacheStatus.className = isCached ? "cache-indicator" : "cache-indicator
fetched";

        grid.innerHTML = items

```

44165850.1551

```

    .map(
      (item) => `
<div class="card">
  <h3>${item.title || item.name}</h3>
  <p>${item.description || item.excerpt || item.bio || ""}</p>
  <div class="card-meta">
    ${item.badge ? `<span class="badge">${item.badge}</span>` : ""}
    ${item.role ? `<span class="badge">${item.role}</span>` : ""}
    ${
      item.price
        ? `<span class="badge" style="color: #667eea; font-weight:
600;">${item.price}</span>`
        : ""
    }
    ${
      item.readTime ? `<span class="badge">${item.readTime}</span>` : ""
    }
    ${item.date ? `<span class="badge">${item.date}</span>` : ""}
  </div>
</div>
`
    )
    .join("");
  }, Math.random() * 500 + 300);
}

document.querySelectorAll(".nav-btn").forEach((btn) => {
  btn.addEventListener("click", () => {
    renderPage(btn.dataset.page);
  });
  btn.addEventListener("mouseenter", () => {
    const page = btn.dataset.page;
    if (page !== "home") {
      console.log(`Prefetching ${page}...`);

      fetch(`/?page=${page}`, { priority: "low" }).catch(() => {});
    }
  });
});

function updateCacheStatus() {
  const cached = Object.keys(pages).filter((page) => {
    return page === "home" ||
document.getElementById(page).classList.contains("active");
  });

  document.getElementById("cache-status").textContent = `${cached.length} of
${
  Object.keys(pages).length
} pages in memory`;
}

```

44165850.1551

```

    setInterval(updateCacheStatus, 1000);

    updateCacheStatus();
  </script>
</body>
</html>

```

## Preloading.html

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Preloading - Technical Documentation</title>

    <link
      rel="preload"
      as="font"
      href="https://fonts.googleapis.com/css2?family=JetBrains+Mono:wght@400;600&dis
play=swap"
      crossorigin
    />
    <link rel="preload" as="image" href="https://picsum.photos/1200/600?random=1" />
    <link
      rel="preload"
      as="font"
      href="https://fonts.googleapis.com/css2?family=Roboto:wght@100;300;400;500;700
;900&family=Roboto+Condensed:wght@300;400;700&family=Roboto+Slab:wght@100;300;400;50
0;700;900&display=swap"
      crossorigin
    />
    <link
      rel="preload"
      as="font"
      href="https://fonts.googleapis.com/css2?family=Open+Sans:wght@300;400;500;600;
700;800&family=Open+Sans+Condensed:wght@300;700&display=swap"
      crossorigin
    />
    <link
      rel="preload"
      as="font"
      href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@400;500;6
00;700;800;900&family=Playfair+Display+SC:wght@400;700;900&display=swap"
      crossorigin
    />
    <link
      rel="preload"
      as="font"

```

44165850.1551

```
    href="https://fonts.googleapis.com/css2?family=Montserrat:wght@100;200;300;400;500;600;700;800;900&display=swap"
    crossorigin
  />
  <link
    rel="preload"
    as="font"
    href="https://fonts.googleapis.com/css2?family=Lato:wght@100;300;400;700;900&family=Lato:wght@100;300;400;700;900&display=swap"
    crossorigin
  />
  <link
    rel="preload"
    as="font"
    href="https://fonts.googleapis.com/css2?family=Merriweather:wght@300;400;700;900&family=Merriweather+Sans:wght@300;400;700;800&display=swap"
    crossorigin
  />
  <link
    rel="preload"
    as="font"
    href="https://fonts.googleapis.com/css2?family=Oswald:wght@200;300;400;500;600;700&family=Oswald:wght@200;300;400;500;600;700&display=swap"
    crossorigin
  />
  <link
    rel="preload"
    as="font"
    href="https://fonts.googleapis.com/css2?family=Source+Sans+Pro:wght@200;300;400;600;700;900&family=Source+Serif+Pro:wght@400;600;700&display=swap"
    crossorigin
  />
  <link
    rel="preload"
    as="font"
    href="https://fonts.googleapis.com/css2?family=Raleway:wght@100;200;300;400;500;600;700;800;900&display=swap"
    crossorigin
  />
  <link
    rel="preload"
    as="font"
    href="https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;500;600;700;800;900&display=swap"
    crossorigin
  />
```

44165850.1551

```
<link
href="https://fonts.googleapis.com/css2?family=JetBrains+Mono:wght@400;600&display=swap" rel="stylesheet" />
<link
href="https://fonts.googleapis.com/css2?family=Roboto:wght@100;300;400;500;700;900&family=Roboto+Condensed:wght@300;400;700&family=Roboto+Slab:wght@100;300;400;500;700;900&display=swap"
rel="stylesheet"
/>
<link
href="https://fonts.googleapis.com/css2?family=Open+Sans:wght@300;400;500;600;700;800&family=Open+Sans+Condensed:wght@300;700&display=swap"
rel="stylesheet"
/>
<link
href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@400;500;600;700;800;900&family=Playfair+Display+SC:wght@400;700;900&display=swap"
rel="stylesheet"
/>
<link
href="https://fonts.googleapis.com/css2?family=Montserrat:wght@100;200;300;400;500;600;700;800;900&display=swap"
rel="stylesheet"
/>
<link
href="https://fonts.googleapis.com/css2?family=Lato:wght@100;300;400;700;900&display=swap" rel="stylesheet" />
<link
href="https://fonts.googleapis.com/css2?family=Merriweather:wght@300;400;700;900&family=Merriweather+Sans:wght@300;400;700;800&display=swap"
rel="stylesheet"
/>
<link
href="https://fonts.googleapis.com/css2?family=Oswald:wght@200;300;400;500;600;700&display=swap" rel="stylesheet" />
<link
href="https://fonts.googleapis.com/css2?family=Source+Sans+Pro:wght@200;300;400;600;700;900&family=Source+Serif+Pro:wght@400;600;700&display=swap"
rel="stylesheet"
/>
<link
href="https://fonts.googleapis.com/css2?family=Raleway:wght@100;200;300;400;500;600;700;800;900&display=swap"
rel="stylesheet"
/>
<link
href="https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;500;600;700;800;900&display=swap"
rel="stylesheet"
```

44165850.1551

```
 />

<style>
  * {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
  }

  body {
    font-family: "Roboto", "Open Sans", -apple-system, BlinkMacSystemFont,
"Segoe UI", Oxygen, Ubuntu, Cantarell,
    sans-serif;
    background: #0f1419;
    color: #e0e0e0;
    line-height: 1.6;
  }

  header {
    background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
    padding: 60px 40px;
    text-align: center;
    color: white;
    position: relative;
    overflow: hidden;
  }

  header::before {
    content: "";
    position: absolute;
    top: -50%;
    right: -10%;
    width: 500px;
    height: 500px;
    background: radial-gradient(circle, rgba(255, 255, 255, 0.1) 0%, transparent
70%);
    border-radius: 50%;
  }

  header h1 {
    font-family: "Playfair Display", "Merriweather", serif;
    font-size: 3rem;
    font-weight: 700;
    margin-bottom: 10px;
    position: relative;
    z-index: 1;
  }

  .strategy-badge {
```

44165850.1551

```
display: inline-block;
background: rgba(255, 255, 255, 0.2);
padding: 8px 16px;
border-radius: 20px;
font-size: 0.9rem;
backdrop-filter: blur(10px);
position: relative;
z-index: 1;
}

nav {
  background: #1a1f26;
  padding: 20px 40px;
  border-bottom: 1px solid #333;
  position: sticky;
  top: 0;
  z-index: 100;
}

nav a {
  font-family: "Montserrat", "Raleway", sans-serif;
  color: #667eea;
  text-decoration: none;
  margin-right: 30px;
  font-weight: 600;
  transition: color 0.3s;
}

nav a:hover {
  color: #764ba2;
}

.container {
  max-width: 900px;
  margin: 0 auto;
  padding: 60px 40px;
}

article {
  background: #1a1f26;
  padding: 40px;
  border-radius: 12px;
  margin-bottom: 40px;
  border: 1px solid #333;
}

h2 {
  font-family: "Oswald", "Raleway", sans-serif;
  font-size: 2rem;
```

44165850.1551

```
    font-weight: 600;
    color: #667eea;
    margin-bottom: 30px;
    padding-bottom: 15px;
    border-bottom: 2px solid #764ba2;
}

h3 {
    font-family: "Poppins", "Source Sans Pro", sans-serif;
    font-size: 1.5rem;
    font-weight: 600;
    color: #764ba2;
    margin-top: 30px;
    margin-bottom: 15px;
}

p {
    font-family: "Lato", "Open Sans", sans-serif;
    margin-bottom: 15px;
    color: #b0b0b0;
    text-align: justify;
}

.metric-box {
    background: linear-gradient(135deg, rgba(102, 126, 234, 0.1) 0%, rgba(118,
75, 162, 0.1) 100%);
    border-left: 4px solid #667eea;
    padding: 20px;
    margin: 25px 0;
    border-radius: 8px;
    font-family: "Source Sans Pro", "Roboto", sans-serif;
}

.metric-box strong {
    font-family: "Roboto Slab", "Merriweather", serif;
    font-weight: 700;
    color: #667eea;
}

code {
    background: #0f1419;
    padding: 2px 6px;
    border-radius: 4px;
    font-family: "JetBrains Mono", monospace;
    color: #67f26b;
    font-size: 0.9em;
}

pre {
```

44165850.1551

```
background: #0f1419;
border: 1px solid #333;
border-radius: 8px;
padding: 20px;
margin: 20px 0;
overflow-x: auto;
color: #e0e0e0;
}

pre code {
background: none;
padding: 0;
color: #67f26b;
}

.hero-image {
width: 100%;
height: 400px;
border-radius: 12px;
margin: 30px 0;
object-fit: cover;
background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
}

.content-section {
margin-bottom: 40px;
}

ul,
ol {
margin-left: 20px;
margin-bottom: 15px;
}

li {
font-family: "Merriweather Sans", "Open Sans", sans-serif;
margin-bottom: 8px;
color: #b0b0b0;
}

footer {
font-family: "Roboto Condensed", "Oswald", sans-serif;
text-align: center;
color: #666;
padding: 40px;
border-top: 1px solid #333;
}

.loading-animation {
```

44165850.1551

```

display: inline-block;
width: 20px;
height: 20px;
border: 3px solid #667eea;
border-top: 3px solid #764ba2;
border-radius: 50%;
animation: spin 1s linear infinite;
margin-right: 10px;
}

@keyframes spin {
  to {
    transform: rotate(360deg);
  }
}
</style>
</head>
<body>
  <header>
    <h1>Preloading Demo</h1>
    <span class="strategy-badge">Link Rel Preload</span>
  </header>

  <nav>
    <a href="#overview">Overview</a>
    <a href="#architecture">Architecture</a>
    <a href="#optimization">Optimization</a>
    <a href="#performance">Performance</a>
  </nav>

  <div class="container">
    <div class="metric-box">
      <strong>Strategy:</strong> Critical resources (fonts, hero images, scripts)
are preloaded in <code><link rel="preload" /></code><br />
      <strong>Effect:</strong> Faster initial render, reduced LCP, improved FCP
    </div>

    <article id="overview">
      <h2>System Architecture Overview</h2>

      <div class="content-section">
        <h3>Introduction</h3>
        <p>
          Modern web applications require sophisticated loading strategies to
ensure optimal performance across

```

44165850.1551

diverse network conditions and device capabilities. This documentation explores advanced preloading

techniques that significantly improve Time to First Byte (TTFB), First Contentful Paint (FCP), and Largest Contentful Paint (LCP) metrics.

```
</p>
```

```
<p>
```

Preloading is a declarative fetch mechanism that tells the browser to begin fetching a resource

immediately while still parsing the HTML. This differs from lazy loading where resources are fetched

on-demand, and prefetching where resources are loaded during idle time.

```
</p>
```

```
</div>
```

```
<div class="content-section">
```

```
<h3>Core Principles</h3>
```

```
<p>
```

The preload strategy prioritizes resources based on their criticality to the initial render. Key resources

include:

```
</p>
```

```
<ul>
```

```
<li><strong>Web Fonts:</strong> Preload font files to prevent FOUT (Flash of Unstyled Text)</li>
```

```
<li><strong>Hero Images:</strong> Preload above-the-fold images to improve LCP</li>
```

```
<li><strong>Critical CSS:</strong> Inline critical styles and preload secondary stylesheets</li>
```

```
<li><strong>Essential JavaScript:</strong> Preload render-blocking scripts to reduce TTI</li>
```

```
</ul>
```

```
</div>
```

```
<div class="content-section">
```

```
<h3>Implementation Guide</h3>
```

```
<pre><code>&lt;head&gt;
```

```
&lt;!- Preload fonts --&gt;
```

```
&lt;link rel="preload" as="font" href="/fonts/main.woff2" crossorigin&gt;
```

```
&lt;!- Preload hero images --&gt;
```

```
&lt;link rel="preload" as="image" href="/images/hero.webp"&gt;
```

```
&lt;!- Preload critical scripts --&gt;
```

```
&lt;link rel="preload" as="script" href="/js/core.js"&gt;
```

```
&lt;/head&gt;</code></pre>
```

```
</div>
```

```
</article>
```

44165850.1551

```

<article id="architecture">
  <h2>Technical Architecture</h2>

  <div class="content-section">
    <h3>Resource Loading Phases</h3>
    <p>
      Web applications are composed of multiple resource types that load in
      specific phases. Understanding these
      phases enables strategic placement of preload directives.
    </p>

    <div class="metric-box">
      <strong>Phase 1 - Head Parsing (0-200ms):</strong> Browser parses HTML
      head and discovers critical
      resources via preload directives<br />
      <strong>Phase 2 - Resource Fetching (200-1000ms):</strong> Preloaded
      resources are fetched in parallel<br />
      <strong>Phase 3 - DOM Construction (1000-2000ms):</strong> Critical
      resources become available for
      rendering
    </div>

    <h3>Resource Priority Matrix</h3>
    <p>Different resources require different loading strategies based on
    impact on user-perceived performance:</p>

    <table style="width: 100%; border-collapse: collapse; margin: 20px 0">
      <tr style="background: #0f1419; border: 1px solid #333">
        <th style="padding: 12px; text-align: left; color: #667eea">Resource
Type</th>
        <th style="padding: 12px; text-align: left; color:
#667eea">Priority</th>
        <th style="padding: 12px; text-align: left; color:
#667eea">Strategy</th>
      </tr>
      <tr style="border: 1px solid #333">
        <td style="padding: 12px">Web Fonts</td>
        <td style="padding: 12px">Critical</td>
        <td style="padding: 12px">Preload with crossorigin</td>
      </tr>
      <tr style="border: 1px solid #333">
        <td style="padding: 12px">Hero Images</td>
        <td style="padding: 12px">Critical</td>
        <td style="padding: 12px">Preload in head</td>
      </tr>
      <tr style="border: 1px solid #333">

```

44165850.1551

```

    <td style="padding: 12px">CSS Stylesheets</td>
    <td style="padding: 12px">High</td>
    <td style="padding: 12px">Inline critical, preload rest</td>
  </tr>
  <tr style="border: 1px solid #333">
    <td style="padding: 12px">JavaScript</td>
    <td style="padding: 12px">Medium</td>
    <td style="padding: 12px">Preload, defer execution</td>
  </tr>
</table>
</div>

<div class="content-section">
  <h3>Browser Preload Behavior</h3>
  <p>
    When a browser encounters a preload directive, it immediately begins
    fetching the resource with
    appropriate priority. For fonts, the crossorigin attribute must be
    present even if the font is
    same-origin, as this ensures the browser uses CORS mode for fetching.
  </p>
  <pre><code>&lt;!- Correct: Font preload with crossorigin --&gt;
  &lt;link rel="preload" as="font" href="/fonts/custom.woff2"
crossorigin&gt;

  &lt;!- Image preload (supports WebP) --&gt;
  &lt;link rel="preload" as="image" href="/img/hero.webp"
type="image/webp"&gt;

  &lt;!- Fallback for non-WebP browsers --&gt;
  &lt;link rel="preload" as="image" href="/img/hero.jpg"&gt;</code></pre>
</div>
</article>

<article id="optimization">
  <h2>Optimization Techniques</h2>

  <div class="content-section">
    <h3>Advanced Preloading Strategies</h3>

    <div class="metric-box">
      <strong>Font Display Property:</strong> Combine preload with <code>font-
display: swap</code> to prevent
      layout shift and FOUT
    </div>
  </div>

```

44165850.1551

```

    <pre><code>@font-face {
font-family: 'CustomFont';
src: url('/fonts/custom.woff2') format('woff2');
font-display: swap;
font-weight: 400;
}

/* Preload in head */
<link rel="preload" as="font" href="/fonts/custom.woff2"
crossorigin"></code></pre>

    <h3>Image Optimization with Preload</h3>
    <p>
        Combine preload with responsive images and modern formats for optimal
        performance. Use the picture element
        with preload to handle multiple image formats.
    </p>

    <pre><code><link rel="preload" as="image" href="/img/hero.webp"
type="image/webp">

<picture>
    <source srcset="/img/hero.webp" type="image/webp">
    
</picture></code></pre>
    </div>

    <div class="content-section">
    <h3>Script Preloading</h3>
    <p>
        Preload scripts that are critical for interactivity, but use
        <code>defer</code> or
        <code>async</code> attributes to avoid blocking HTML parsing.
    </p>

    <pre><code><!-- Preload critical app bundle -->
<link rel="preload" as="script" href="/js/app.js">

<!-- Defer execution until page load -->
<script src="/js/app.js" defer"></script></code></pre>
    </div>
</article>

<article id="performance">
    <h2>Performance Metrics & Monitoring</h2>

```

44165850.1551

```

<div class="content-section">
  <h3>Measuring Preload Effectiveness</h3>

  <div class="metric-box">
    <strong>Key Metrics to Monitor:</strong><br />
    • Time to First Byte (TTFB): Should be < 600ms<br />
    • First Contentful Paint (FCP): Should be < 1.8s<br />
    • Largest Contentful Paint (LCP): Should be < 2.5s<br />
    • Cumulative Layout Shift (CLS): Should be < 0.1
  </div>

  <h3>Performance Budget</h3>
  <p>Establish a performance budget for your application. With preloading,
you should see:</p>
  <ul>
    <li>15-25% reduction in LCP when hero images are preloaded</li>
    <li>10-20% reduction in FCP when fonts are preloaded</li>
    <li>30% improvement in Time to Interactive (TTI) when critical scripts
are preloaded</li>
  </ul>
</div>

<div class="content-section">
  <h3>Monitoring Tools</h3>
  <p>
    Use tools like Lighthouse, WebPageTest, and the Performance API to
    monitor preload effectiveness. The
    PerformanceResourceTiming API provides detailed information about
    resource fetch timings.
  </p>

  <pre><code>// Monitor preload effectiveness
const resources = performance.getEntriesByType('resource');
resources.forEach(resource => {
  console.log(` ${resource.name}: ${resource.duration}ms`);
  console.log(` Fetch start: ${resource.fetchStart}ms`);
  console.log(` Request start: ${resource.requestStart}ms`);
});</code></pre>
</div>
</article>
</div>

<footer>
  <p>© 2025 URL Audit Demo - Preloading Strategy</p>
</footer>

<script>
  document.addEventListener("DOMContentLoaded", () => {
    const heavyData = Array(10000)

```

44165850.1551

```
.fill(0)
.map((_, i) => ({
  id: i,
  value: Math.random(),
  timestamp: Date.now(),
}));

console.log("Page fully loaded with preloaded resources");
});
</script>
</body>
</html>
```

44165850.1551

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЗАТВЕРДЖУЮ  
Перший проректор  
Українського  
державного  
університету науки і  
технологій  
Анатолій РАДКЕВИЧ

ВПЛИВ СТРАТЕГІЙ ПОПЕРЕДНЬОГО ЗАВАНТАЖЕННЯ  
РЕСУРСІВ НА ШВИДКІСТЬ ВІДОБРАЖЕННЯ ВЕБ-СТОРИНОК

Тези до наукової конференції

ЛИСТ ЗАТВЕРДЖЕННЯ  
44165850.1551-01 91-01 ЛЗ

Завідувач кафедри КІТ  
Вадим ГОРЯЧКІН  
Керівник розробки Тетяна  
ГРИШЕЧКІНА  
Виконавець  
Андрій ПРИЙМИЧ  
Нормоконтролер  
Світлана ВОЛКОВА

44165850.1551

ЗАТВЕРДЖЕНО

44165850.1551-01 91-01

ВПЛИВ СТРАТЕГІЙ ПОПЕРЕДНЬОГО ЗАВАНТАЖЕННЯ  
РЕСУРСІВ НА ШВИДКІСТЬ ВІДОБРАЖЕННЯ ВЕБ-  
СТОРИНОК

Тези до наукової конференції

44165850.1551-01 91-01

Листів 3

44165850.1551



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
НАУКИ І ТЕХНОЛОГІЙ

ABSTRACTS  
OF THE XVIII INTERNATIONAL CONFERENCE  
«MODERN INFORMATION AND COMMUNICATION  
TECHNOLOGIES ON A TRANSPORT, IN INDUSTRY AND  
EDUCATION»  
12-13, December, 2024

СУЧАСНІ ІНФОРМАЦІЙНІ ТА  
КОМУНІКАЦІЙНІ  
ТЕХНОЛОГІЇ НА ТРАНСПОРТІ,  
В ПРОМИСЛОВОСТІ І ОСВІТІ

ПРИСВЯЧЕНО ПАМ'ЯТІ ПРОФЕСОРА ВЛАДИСЛАВА СКАЛОЗУБА

ТЕЗИ

XVIII МІЖНАРОДНОЇ  
НАУКОВО-  
ПРАКТИЧНОЇ  
КОНФЕРЕНЦІЇ  
12-13 ГРУДНЯ 2024

ДНІПРО  
2024

44165850.1551

Комплекс програм для чисельного моделювання теплових режимів в опалювальних культивацийних спорудах .....	114
Біляєва В. В., Український державний університет науки і технологій, Україна, Бразалук Ю. В., Дніпровський національний університет ім. О.Гончара, Україна, Шербіна С. А., Український державний університет науки і технологій, Україна	
Комп'ютерні моделі для дослідження процесів масопереносу .....	115
Козачина В. А., Український державний університет науки і технологій, Україна, Калашников А. В., Кириченко М. В., Філоненко Г. К., Український державний університет науки і технологій, Україна	
Математичне моделювання розльоту уламків при вибуху дрону .....	116
Біляєв М. М., Берлов О. В., Український державний університет науки і технологій, Україна Калашников І. В., Харківське відділення філії «Проектно-вишукавальний Інститут залізничного транспорту» АТ «Українська залізниця», Україна Кіриченко П. С., Криворізький національний університет, Україна Козачина В. А., Український державний університет науки і технологій, Україна	
Чисельне моделювання в прикладних задачах аеродинаміки та тепломасопереносу, що виникають на етапі передстартової підготовки ракети- носія .....	117
Біляєв М. М., Берлов О. В., Біляєва В. В., Козачина В. А., Український державний університет науки і технологій, Україна Семененко П. В., ДП «КБ «ПІВДЕННЕ» ім. М.К. Янгеля»	
Математичне моделювання гідродинамічної взаємодії тіл в каналі .....	118
Бразалук Ю. В., Дніпровський національний університет імені Олеса Гончара, Україна Біляєва В. В., Український державний університет науки і технологій, Україна	
Конструювання алгоритмів сортування .....	119
Макаров О. В., Шинкаренко В. І., Український державний університет науки і технологій, Україна	
Обробка зображень номерів вантажних вагонів .....	120
Пашук В.В., Український державний університет науки і технологій, Україна	
Аналіз методів розпізнавання сміттєзвалищ на аерокосмічних зображеннях з використанням машинного навчання .....	121
Литвинов Є. В., Гнатушенко Вік.В., Гнатушенко В.В., Український державний університет науки і технологій, Україна	
Моніторинг поверхневих вод на основі геопросторових даних .....	122
Кавац Ю.В., Фененко Т.М., Горський М.Д., Український державний університет науки і технологій, Україна	
Супутниковий моніторинг зміни лісових насаджень .....	123
Кавац Ю.В., Фененко Т.М., Денісов Д.Г., Український державний університет науки і технологій, Україна	
Індексція кодової бази як ключ до ефективного рефакторингу з використанням штучного інтелекту .....	124
Сирота О. А., Горячкін В. М., Український державний університет науки і технологій, Україна	

### **Вплив стратегій попереднього завантаження ресурсів на швидкість відображення веб-сторінок**

Приймич А.М., Український державний університет науки та технологій, Україна

Швидкість завантаження веб-сторінок відіграє ключову роль у наданні користувачам найкращого досвіду, тому важливо, щоб вона була максимально високою як з людської, так і з технічної точки зору. Кожна односекундна затримка може викликати зниження конверсії на 20%, за даними Google (2021). Ніхто не любить чекати, з точки зору SEO та загально-го користувацького досвіду, швидкість сайту залишається критично важливим фактором, адже відвідувачі очікують миттєвого реагування. Це підкреслює необхідність застосування розумних методів оптимізації, таких як попередня вибірка, попереднє завантаження та ліниве завантаження.

Попередня вибірка є ефективним підходом до попереднього завантаження ресурсів, які можуть знадобитися згодом. Вона істотно зменшує затримки під час переходу між сторінками, що прискорює взаємодію користувача з веб-ресурсом. За результатами досліджень, попередня вибірка здатна скоротити час завантаження сторінок на 20-30% (Google Web Fundamentals, n.d.).

Попереднє завантаження медіа це техніка, спрямована на завантаження ключових медіа ресурсів ще до повного відображення сторінки. Цей спосіб дозволяє швидше рендерити основні елементи інтерфейсу, створюючи позитивне перше враження у відвідувачів. Використання попереднього завантаження значно скорочує час завантаження, необхідний для повного відображення сторінки (Mozilla Developer Network, 2023).

Ліниве завантаження є ще однією ефективною стратегією оптимізації, яка полягає у відкладеному завантаженні таких медіафайлів, як зображення або відео, які не потрібні для первинного перегляду сторінки. Цей підхід знижує початкове навантаження на сервер і пришвидшує роботу веб-сторінки. Упровадження лінивого завантаження може скоротити час завантаження на 30-50% (Smashing Magazine, 2023). Крім того, це дозволяє уникнути завантаження зайвих даних і забезпечує більш ефективне використання ресурсів.

Швидкість завантаження веб-сайту також є актуальним викликом для бізнесу. Згідно з дослідженням Akamai, 53% користувачів закривають сторінку, якщо завантаження триває понад 3 секунди (Akamai, 2020). Це не лише призводить до втрати потенційних клієнтів і зниження продажів, але й може нашкодити репутації бренду через високий рівень незадоволення відвідувачів і збільшення показника відмов (HubSpot, 2021). Наприклад, збій у роботі Amazon у 2018, через який сайт завантажувався надзвичайно довго, призвів до багатомільярдних збитків компанії та вплинув на ціну її акцій. Компанія HubSpot також наголошує на тому, що продуктивність веб-сайту має вирішальний вплив на залучення й утримання клієнтів, що безпосередньо позначається на доходах бізнесу (HubSpot, 2021). Застосування технологій оптимізації швидкості завантаження є відмінним способом прискорити роботу сайту. Це допомагає покращити взаємодію користувачів із веб-ресурсом, суттєво скорочує час завантаження сторінок і сприяє збільшенню рівня конверсії. У підсумку ці заходи підвищують ефективність бізнесу. Таким чином, оптимізація швидкості є невід'ємною складовою успішної веб-розробки та бізнес-стратегії.