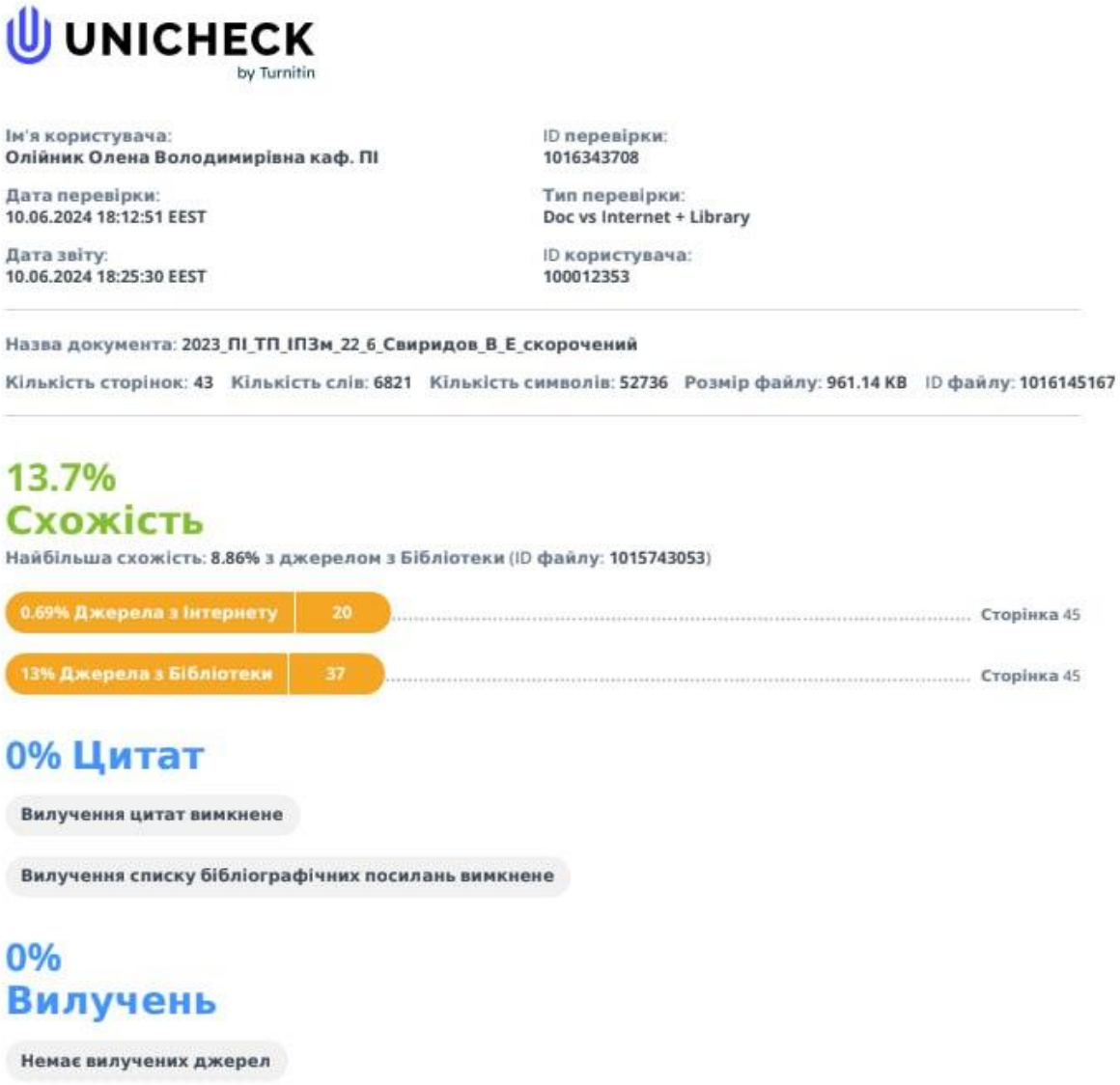
ДОДАТОК А

Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ



UNICHECK
by Turnitin

Ім'я користувача:
Олійник Олена Володимирівна каф. ПІ

ID перевірки:
1016343708

Дата перевірки:
10.06.2024 18:12:51 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
10.06.2024 18:25:30 EEST

ID користувача:
100012353

Назва документа: 2023_ПІ_ТП_ІПЗм_22_6_Свиридов_В_Е_скорочений

Кількість сторінок: 43 Кількість слів: 6821 Кількість символів: 52736 Розмір файлу: 961.14 KB ID файлу: 1016145167

13.7%
Схожість

Найбільша схожість: 8.86% з джерелом з Бібліотеки (ID файлу: 1015743053)

0.69% Джерела з Інтернету 20 ..... Сторінка 45

13% Джерела з Бібліотеки 37 ..... Сторінка 45

0% Цитат

Вилучення цитат вимкнене

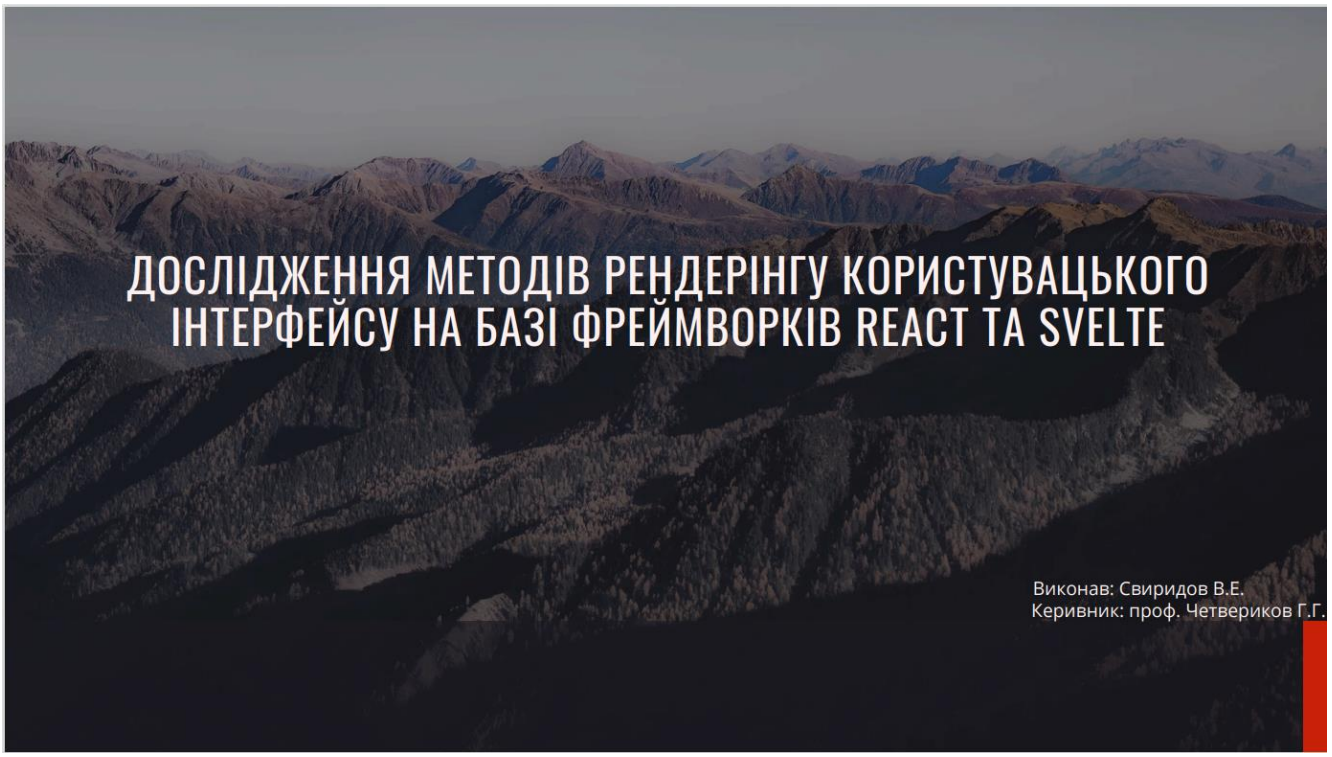
Вилучення списку бібліографічних посилань вимкнене

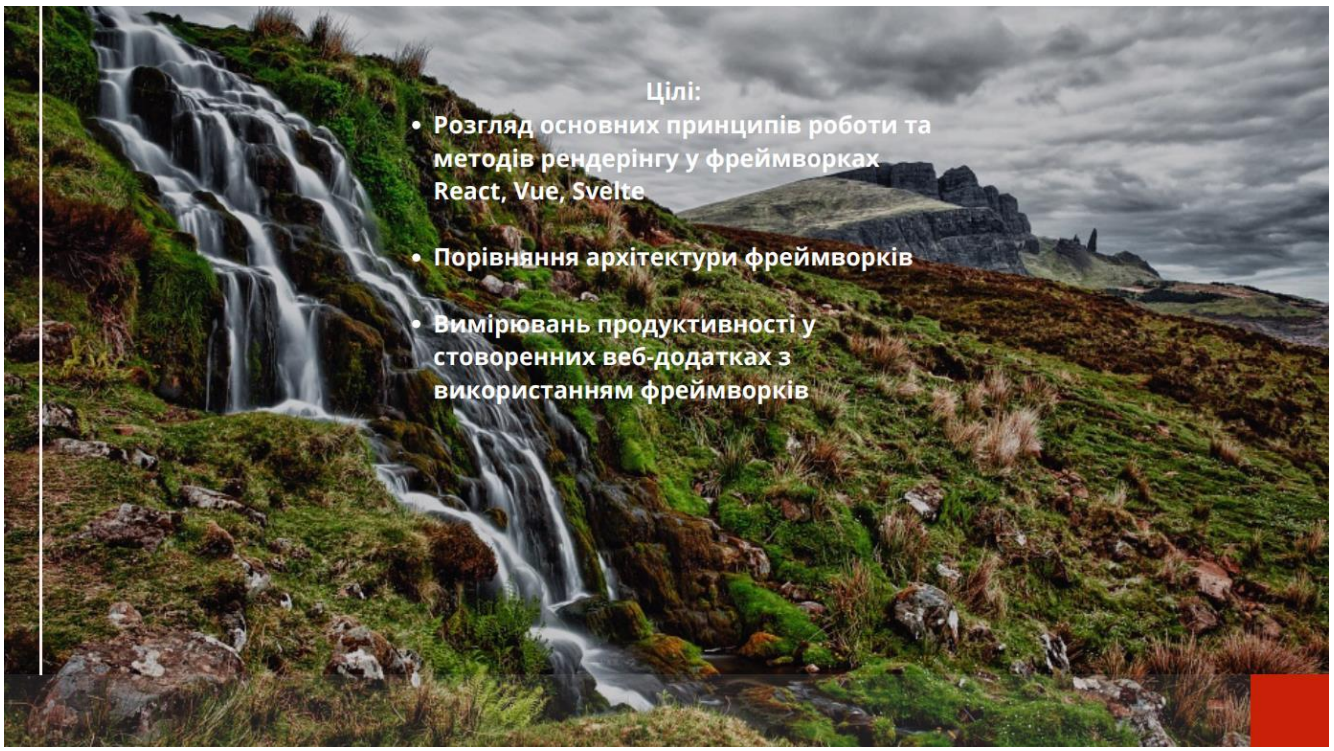
0%
Вилучень

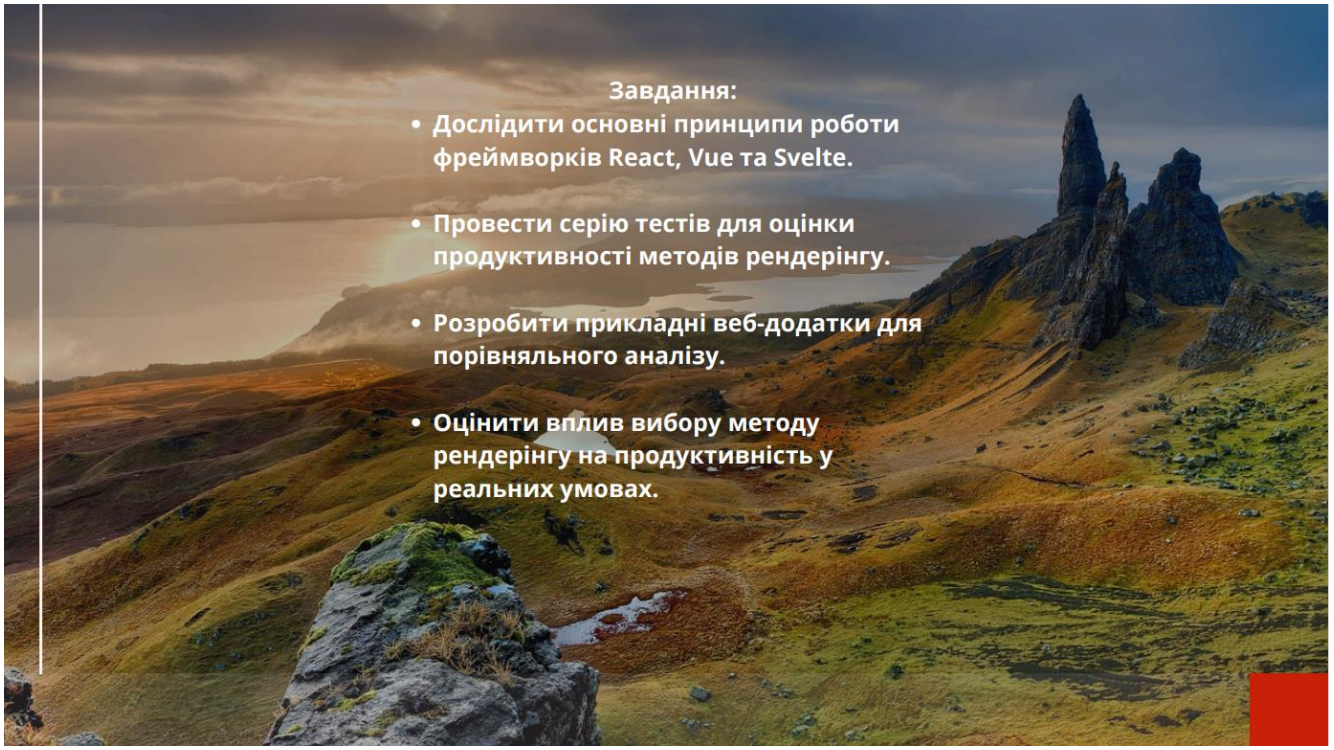
Немає вилучених джерел

ДОДАТОК Б

Слайди презентації




Цілі:
• Розгляд основних принципів роботи та методів рендерінгу у фреймворках React, Vue, Svelte
• Порівняння архітектури фреймворків
• Вимірювань продуктивності у створенних веб-додатках з використанням фреймворків


Завдання:
• Дослідити основні принципи роботи фреймворків React, Vue та Svelte.
• Провести серію тестів для оцінки продуктивності методів рендерінгу.
• Розробити прикладні веб-додатки для порівняльного аналізу.
• Оцінити вплив вибору методу рендерінгу на продуктивність у реальних умовах.

Порівняння фреймворків

Рендерінг	**Використовує віртуальний DOM**	**Компілює в нативний JavaScript.**	**Використовує реактивну систему**
Архітектура компонентів	Концепція компонентів (функціональні та класові).	Проста та декларативна архітектура.	Однофайлові компоненти.
Спрямованість та класи залежностей	Частково спрямований (JSX).	Спрямований (декларативний) за замовчуванням.	Частково спрямований.
Зворотна сумісність	Зазвичай дотримується, але можливі проблеми під час оновлень.	Обмежена зворотна сумісність через філософію компіляції.	Зазвичай дотримується, робить оновлення менш болісним.
Документація та підтримка	Широка документація та активна спільнота.	Докладна документація, менша спільнота.	Добре структурована документація та активна спільнота.

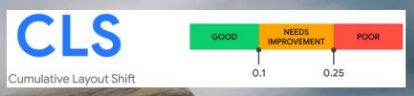
Переваги фреймворків

переваги	• **Віртуальний DOM забезпечує високу продуктивність.** • **Велика спільнота та кількість бібліотек.** • **Односторонній потік даних спрощує управління станом.**	• **Компіляція на етапі збірки зменшує розмір та навантаження на браузер.** • **Проста та декларативна архітектура.** • **Відсутність віртуального DOM підвищує продуктивність.**	• **Реактивна система забезпечує високу продуктивність та простоту у використанні.** • **Однофайлові компоненти спрощують розробку.** • **Добре структурована документація та активна спільнота.**
недоліки	• **Велика кількість додаткових бібліотек може ускладнювати проект.** • **Велика кількість додаткових бібліотек може ускладнювати проект.**	• **Відносно молода технологія з меншою спільнотою.** • **Обмежена кількість готових рішень та бібліотек.**	• **Можливі проблеми з масштабованістю у великих проектах.** • **Часткова спрямованість на конкретні завдання може обмежувати гнучкість.**

Аналіз характеристик продуктивності та стратегій рендерингу

Час, необхідний для завантаження та відображення найбільшого контенту на сторінці

Час затримки між першим взаємодією користувача з елементом на сторінці та відгуком браузера на цю дію



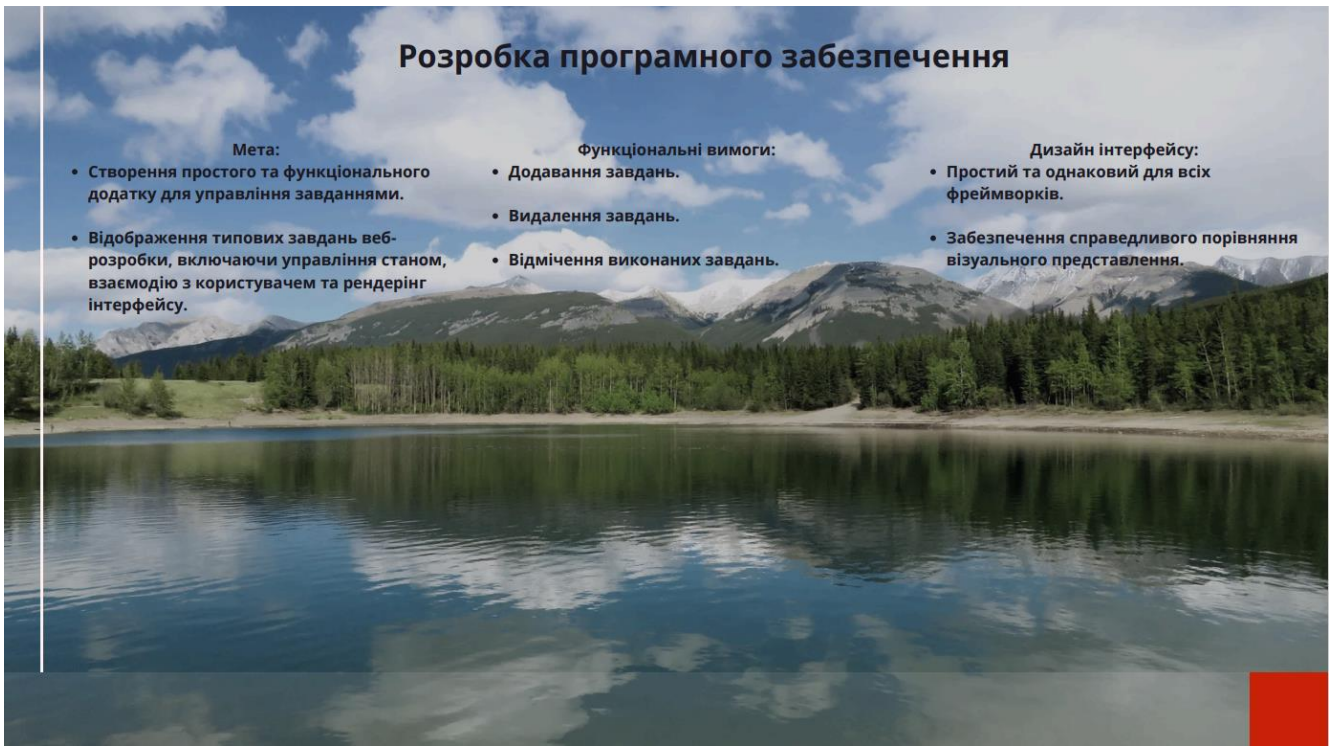
Стабільність візуального вмісту на сторінці під час завантаження та взаємодії з користувачем

Порівняння фреймворків

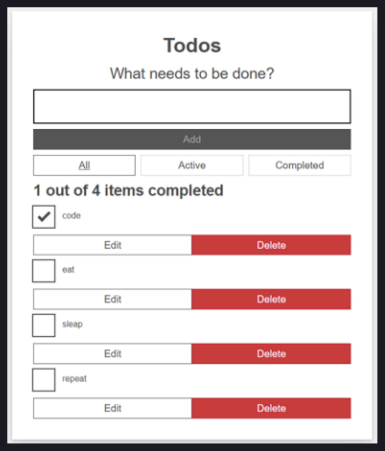
LCP	**Оптимізації, як код-спліттінг та Lazy Loading, покращують LCP.**	**Компіляція на етапі збірки забезпечує швидше завантаження.**	**Однофайлові компоненти сприяють швидкому завантаженню.**
FID	Оптимізація коду та Web Workers покращують FID.	Спрощена структура коду забезпечує низький FID.	Реактивність забезпечує швидку реакцію на дії користувача.
CLS	Використання правильних CSS правил покращує CLS.	Оптимізація на етапі компіляції допомагає уникнути зсувів макету.	Ефективне управління станом зберігає високу стабільність макету.

Порівняння стратегій рендерингу

	Опис	Переваги	Недоліки
SSR	• Сервер генерує повний HTML для кожного запиту. • Клієнт отримує готовий HTML, що дозволяє швидше відобразити контент.	• Поліпшення SEO, оскільки пошукові системи бачать повний контент. • Швидке перше завантаження (First Paint).	Однофайлові компоненти сприяють швидкому завантаженню.
CSR	• Клієнт отримує мінімальний HTML, основний контент генерується JS. • Використовується для високої інтерактивності.	• Зменшення навантаження на сервер. • Висока інтерактивність та можливість динамічних оновлень.	Реактивність забезпечує швидку реакцію на дії користувача.
SSG	• Генерує статичні HTML файли під час збірки, які зберігаються на сервері та віддаються клієнту. • Використовується для контенту, який рідко змінюється.	• Дуже швидке завантаження і відображення контенту. • Низьке навантаження на сервер. • Добре підходить для SEO.	Ефективне управління станом зберігає високу стабільність макету.



Дизайн інтерфейсу



Видалення завдань

Відмічення виконаних завдань

Порівняння роботи веб-сайтів на мобільних пристроях

Продуктивність (Performance)	**82%**	**96%**	**95%**
Доступність (Accessibility)	**87%**	**78%**	**78%**
Найкращі практики (Best Practices)	**100%**	**100%**	**100%**
SEO	**64%**	**82%**	**64%**

Порівняння сайтів на десктопних пристроях у Lighthouse



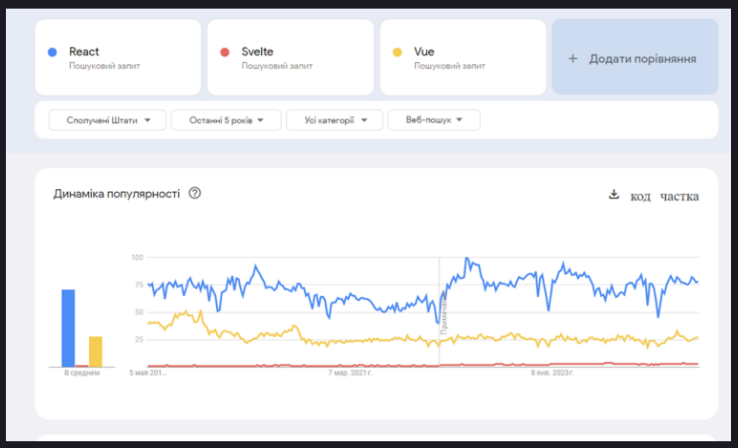
Продуктивність (Performance)	99%	100%	100%
Доступність (Accessibility)	87%	78%	78%
Найкращі практики (Best Practices)	100%	100%	100%
SEO	78%	89%	78%

Порівняння веб-сайтів за допомогою WebPageTest



Time to First Byte (TTFB)	0,731 s	0,731 s	0,743 s
Start Render	1,100 s	2,300 s	1,100 s
First Conten tful Paint (FCP)	1,066 s	2,295 s	1,070 s
Speed Index	1,246 s	2,300 s	1,149 s
Largest Contentful Paint (LCP)	1,896 s	2,295 s	1,451 s
Cumulati ve Layout Shift (CLS)	0,014	0	0,033
Total Blocking Time (TBT)	0,770 s	0,000 s	0,302 s
Page Weight	382k	172k	194k

Перегляд популярності фреймворків у порівняльному аспекті



Аналіз фреймворків на основі даних з Stack Overflow Insights

42.87%	17.64%	6.01%

Висновки

Підходить для динамічних веб-застосунків з великою кількістю інтерактивних елементів. Велика спільнота та екосистема, гнучкість та розширюваність.	Ефективний завдяки компіляції на етапі розробки. Підходить для менших проектів, простота та ефективність.	Спрощений та прогресивний підхід, висока продуктивність. Ідеальний для швидкої розробки та середніх проектів. Рекомендації:

Вибір фреймворку залежить від специфічних вимог проекту.
Враховувати продуктивність, спільноту, зручність у використанні та можливості інтеграції.
Подальші дослідження:

Адаптація сучасних веб-інтерфейсів.
Вивчення впливу технологій штучного інтелекту на автоматизацію розробки.
Оцінка безпеки веб-застосунків та розробка нових методів захисту.

ДОДАТОК Г

Апробація результатів роботи

Міжнародний молодіжний форум "Радіоелектроніка і молодь у XXI столітті". Сторінки 462-465.

ДОДАТОК Д

Експертний висновок результатів перевірки кваліфікаційної роботи

студент (посада) | програмної інженерії (кафедра) | ІПЗм-22-6 (група)

Свиридов Вадим Едуардович

(прізвище, ім'я, по батькові)

Зауваження

Пункт ДСТУ 3008-2015	Зміст пункту	Сторінка кваліфікаційної роботи
1	2	3
	7.1 Загальні положення	
	7.3 Нумерація сторінок звіту	
	7.4 Нумерація розділів, підрозділів, пунктів, підпунктів	
	7.5 Рисунки	
	7.6 Таблиці	
	7.7 Переліки	
	7.8 Примітки	
	7.9 Виноски	
	7.10 Формули та рівняння	
	7.11 Посилання	
	7.13 Список авторів	
	7.14 Скорочення та умовні позначки	
	7.15 Додатки	

зауважень немає

Експерт _____ (підпис) Олена ОЛІЙНИК (прізвище, ініціали)