

«Самостоятельная работа 2 Внутренняя оптимизация»

Выполнил: Подстрелов Вячеслав

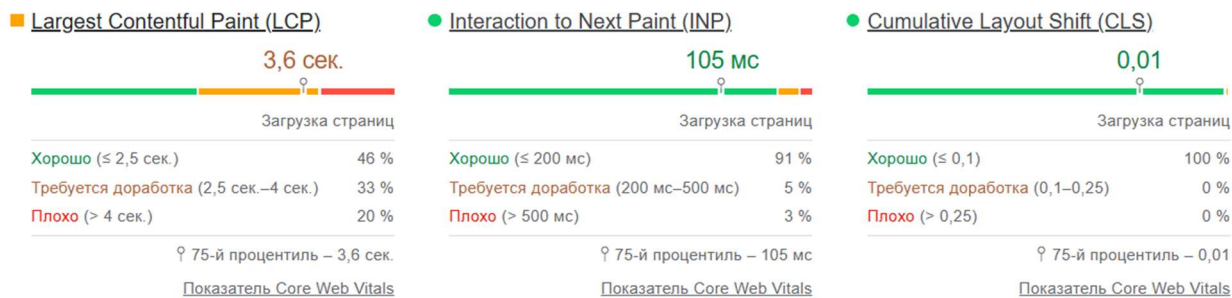
Группа: ПР-23.102

Аналитика с помощью выбранного ПО

1.1 Фактическая производительность сайта

Для аналитики был выбран сайт “War Thander”. Производительность сайта была анализирована относительно мобильной версии.

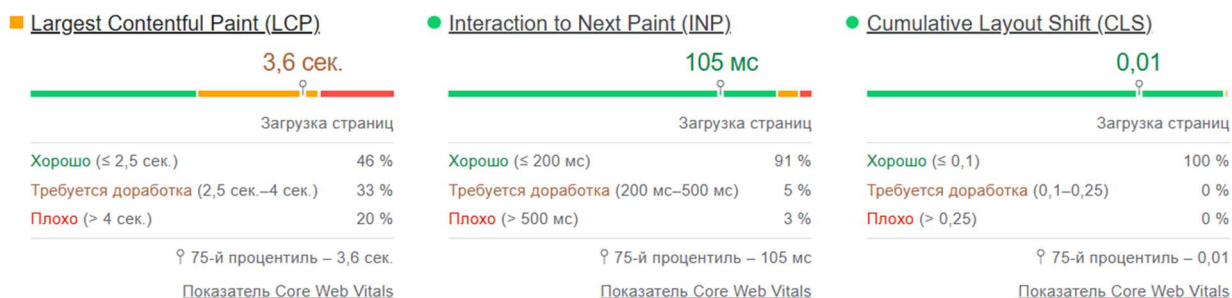
Для начала были измерены основные веб-показатели - Метрики, со временем они будут меняться. Текущий набор фокусируется на трех аспектах пользовательского опыта — загрузке, интерактивности и визуальной стабильности — и включает в себя следующие показатели.



Самая большая краска по содержанию (LCP): измеряет производительность загрузки. Чтобы обеспечить удобство работы пользователя, LCP должен происходить в течение 2,5 секунд с момента первой загрузки страницы.

Interaction to Next Paint (INP): измеряет интерактивность. Чтобы обеспечить хорошее взаимодействие с пользователем, страницы должны иметь INP 200 миллисекунд или меньше.

Совокупный сдвиг макета (CLS): измеряет визуальную стабильность. Чтобы обеспечить хорошее взаимодействие с пользователем, страницы должны поддерживать CLS 0,1. или меньше.



Веб-сайт демонстрирует смешанные результаты. Он показывает превосходную стабильность верстки (CLS), но имеет серьезные проблемы со скоростью загрузки основного контента (LCP) и нормальную отзывчивость на действия пользователя (INP).

1.2 Выявление проблем с производительностью



Производительность

Производительность равна 65:

Оценка производительности представляет собой средневзвешенное значение показателей.

Наблюдается сильная разница в показателях производительности. Страница показывает как отличные, так и не очень хорошие результаты.

Сильные стороны:

Идеальный показатель по CLS (0.01): Это лучший возможный результат. Страница демонстрирует исключительную визуальную стабильность. Элементы не прыгают при загрузке, что обеспечивает комфортный пользовательский опыт.

Критические проблемы:

Крайне низкая скорость загрузки основного контента: Показатели LCP (Largest Contentful Paint) в 9727 мс (9.7 секунд) и FCP (First Contentful Paint) в 2567 мс (2.5 секунды) являются недопустимо высокими. Пользователю приходится очень долго ждать, прежде чем он увидит полезную информацию на странице.

Проблемы с интерактивностью: Высокий показатель TBT (Total Blocking Time) в 428 мс указывает на то, что страница долгое время не реагирует на действия пользователя (клики, ввод с клавиатуры) после загрузки контента.

1.3 Оценка специальных возможностей



Специальные возможности

Этот показатель показывает, насколько трудно будет использовать веб-сайт людям с ограниченными возможностями.

Автоматическая проверка выделяет несколько проблем:

Цвета фона и переднего плана недостаточно контрастны. Многие пользователи не видят текст с низкой контрастностью, или им сложно его воспринимать.

Некоторые ссылки на странице невозможно идентифицировать, так как они визуально отличаются только цветом.

1.4 Оценка поисковой оптимизация



**Поисковая
оптимизация**

Эти проверки позволяют узнать, соответствует ли страница основным рекомендациям к поисковой оптимизации.

Автоматическая проверка выделяет одну проблему:

В документе нет метаописания. Элементы метаописания предоставляет сводку содержимого страницы, которую поисковые системы включают в результаты поиска. Высококачественное уникальное метаописание делает страницу более релевантной и может увеличить поисковый трафик.

Таблица с рекомендациями



Рекомендации

ОБЩИЕ РЕКОМЕНДАЦИИ

▲ Ошибки браузера занесены в журнал консоли	▼
○ Обнаруженные библиотеки JavaScript	▼
▲ Отсутствуют карты исходного кода для собственных больших скриптов JavaScript	▼

НАДЕЖНОСТЬ И БЕЗОПАСНОСТЬ

○ Убедитесь, что политика CSP эффективна против атак XSS	▼
○ Использование строгого механизма HSTS	▼
○ Обеспечьте надлежащую изоляцию источников с помощью COOP	▼
○ Защита от клиджекинга с использованием заголовка XFO или CSP	▼
○ Используйте Trusted Types для предотвращения XSS-атак на основе DOM	▼

Недостаток	Рекомендация	Ожидаемый (предполагаемый эффект от оптимизации)
Низкая производительность загрузки (LCP = 15.0 с, FCP = 11.4 с)	Оптимизировать загрузку критических ресурсов: внедрить lazy loading для изображений, оптимизировать и сжимать изображения (ожидаемая экономия 221 КиБ), использовать современные форматы (WebP), минимизировать и объединять CSS/JS файлы, настроить эффективное кэширование на сервере (ожидаемая экономия 3 102 КиБ).	Улучшение пользовательского опыта и SEO. Сокращение времени ожидания, снижение показателя отказов, потенциальное повышение позиций в поисковой выдаче.
Высокое время до первого байта (TTFB = 3.4 с)	Оптимизировать работу сервера (бэкенд-логика, база данных), использовать кэширование на стороне сервера (CDN), проверить хостинг-провайдера.	Ускорение начала загрузки страницы. Улучшение всех производных метрик (LCP, FCP), зависящих от скорости ответа сервера.
Запросы, блокирующие отрисовку страницы	Вынести некритичные CSS и JS из основного потока, загружать их	Значительное ускорение отображения контента. Улучшение метрик FCP и LCP.

(ожидаемая экономия 9 310 мс)	асинхронно. Критический CSS встраивать в <code><style></code> .	
Наличие длительных задач в основном потоке (обнаружено 4)	Разбить длинные JavaScript-задачи на более мелкие, удалить или отложить неиспользуемый JS (ожидаемая экономия 546 КиБ), избегать длительных синхронных вычислений.	Улучшение отзывчивости интерфейса (INP). Страница будет быстрее реагировать на действия пользователя.
Большой объем неиспользуемого кода (CSS – 1 627 КиБ, JS – 546 КиБ)	Внедрить инструменты для анализа и удаления неиспользуемого кода (Tree Shaking). Разбить код на чанки и загружать по мере необходимости.	Сокращение общего размера ресурсов. Ускорение загрузки и выполнения скриптов, что улучшит TBT, FCP, LCP.
Для изображений не заданы явным образом атрибуты width и height	Для всех тегов <code></code> явно указывать атрибуты <code>width</code> и <code>height</code> для резервирования пространства и предотвращения смещений макета.	Устранение Cumulative Layout Shift (CLS). Повышение визуальной стабильности страницы при загрузке.
Чрезмерная нагрузка на сеть (общий размер страницы 3 585 КиБ)	Провести аудит и оптимизировать все ресурсы: сжать изображения, минифицировать CSS/JS, удалить неиспользуемый код, использовать современные форматы (WebP, Brotli).	Снижение потребления трафика пользователями и ускорение загрузки на медленных соединениях.
Отсутствует политика CSP (Content Security Policy)	Внедрить и настроить заголовок <code>Content-Security-Policy</code> на сервере. Начать с директив <code>default-src 'self'</code> и постепенно ужесточать политику.	Существенное повышение безопасности. Эффективная защита от XSS-атак, предотвращение выполнения вредоносного скрипта.
Отсутствует заголовок Strict-Transport-Security (HSTS)	Добавить заголовок <code>Strict-Transport-Security</code> на сервер с параметром <code>max-age</code> .	Повышение безопасности. Предотвращение downgrade-атак HTTPS на HTTP и принудительное использование шифрованного соединения.
Отсутствует заголовок Cross-Origin-Opener-Policy (COOP)	Добавить заголовок <code>Cross-Origin-Opener-Policy</code> со значением <code>same-origin</code> .	Повышение безопасности. Изоляция контекста страницы от атак, использующих механизм открытия окон.
Отсутствует защита от клидджекинга (X-Frame-Options или CSP frame-ancestors)	Добавить заголовок <code>X-Frame-Options: SAMEORIGIN</code> или директиву <code>frame-ancestors 'self'</code> в политику CSP.	Защита от атак клидджекинга, когда сайт встраивается во вредоносную страницу для обмана пользователей.

Отсутствуют Source Maps для собственных больших скриптов JavaScript	Сгенерировать и опубликовать карты исходного кода (Source Maps) для продакшен- сборки.	Ускорение разработки и отладки. Улучшение качества отчетов об ошибках.
--	--	--