MIDNIGHT WEBS

БЕСПЛАТНЫЙ МАТЕРИАЛ · ЧЕК-ЛИСТ ПО СКОРОСТИ

Чек-лист скорости сайта и Core Web Vitals

Google оценивает качество страницы по трём показателям: как быстро появляется основной контент, как быстро страница реагирует и насколько она «прыгает». Чек-лист поможет понять эти цифры, найти причины медлительности и не дать им ухудшиться.

Обновлено в ноябре 2026 | 12 мин чтения | проверок: 39

ДЛЯ КОГО	ЧТО ОХВАТЫВАЕТ	ЦЕЛИ
Все, кто отвечает за сайт, который кажется медленным или не проходит PageSpeed Insights	6 областей: от измерений до мониторинга	LCP ≤ 2,5 с · INP ≤ 200 мс · CLS ≤ 0,1 на 75-м процентиле мобильных визитов

С ЧЕГО НАЧАТЬ

Сначала исправьте самое большое изображение на самой посещаемой странице. На большинстве сайтов малого бизнеса одна слишком тяжёлая главная картинка — главная причина медленного Largest Contentful Paint.

ОТМЕТЬТЕ КВАДРАТ, КОГДА ГОТОВО · ОБЯЗАТЕЛЬНО = МАКСИМАЛЬНЫЙ ЭФФЕКТ

01 Измерьте страницы, которые важны

Тестируйте шаблоны, которые приносят визиты и выручку. Идеальный балл на странице, которую никто не открывает, ничего не меняет.

- [ ] ОБЯЗАТЕЛЬНО **Запомните три цели**

  Largest Contentful Paint (LCP) ≤ 2,5 с, Interaction to Next Paint (INP) ≤ 200 мс, Cumulative Layout Shift (CLS) ≤ 0,1. Google оценивает их на 75-м процентиле: три четверти реальных визитов должны укладываться в каждую цель. INP заменил First Input Delay в марте 2024 года.

- [ ] ОБЯЗАТЕЛЬНО **Начинайте с полевых данных**
  В PageSpeed Insights сначала читайте блок с опытом реальных пользователей. Он основан на Chrome UX Report, охватывает последние 28 дней и именно его использует Google.

- [ ] **Используйте лабораторные данные, чтобы искать причины**
  Lighthouse (в Chrome DevTools или PageSpeed Insights) указывает на конкретные файлы и элементы. Лабораторные оценки колеблются между запусками, поэтому тестируйте три раза и берите медиану.

- [ ] **Тестируйте каждый шаблон, а не только главную**
  Главная, категория, товар или услуга, статья и оформление заказа. Страницы одного шаблона обычно делят одни и те же проблемы.

- [ ] **Зафиксируйте исходный уровень**
  Дата, URL, устройство и три показателя в таблице. Без «до» «после» — лишь мнение.

- [ ] **Разделяйте мобильные и десктоп**
  Мобильный тест строже, а Google индексирует прежде всего мобильную версию.

КАК ПРОВЕРИТЬ

Откройте отчёт Core Web Vitals в Search Console. Он делит адреса на «Хорошо», «Требует улучшения» и «Плохо», так что видно, какой шаблон исправлять.

02 Исправьте изображения и Largest Contentful Paint

На большинстве сайтов элемент LCP — изображение, а изображения — самые тяжёлые файлы страницы.

- [ ] ОБЯЗАТЕЛЬНО **Найдите свой элемент LCP**
  В Chrome DevTools, на панели Performance, или в диагностике PageSpeed Insights. Исправляйте сначала именно его; ускорение других изображений не поможет.

- [ ] ОБЯЗАТЕЛЬНО **Не применяйте lazy-loading к изображению LCP**
  Добавьте к нему `fetchpriority="high"` и уберите `loading="lazy"`. Лениво загружать всё ниже видимой области правильно; для главного изображения это замедляет загрузку.

- [ ] **Используйте современные форматы**
  AVIF или WebP обычно значительно легче JPEG при похожем качестве. Оставляйте запасной JPEG, только если нужно поддерживать очень старые браузеры.

- [ ] **Отдавайте подходящий размер для каждого экрана**
  Используйте `srcset` и sizes, чтобы телефон загружал картинку шириной 600 px, а не оригинал в 3000 px.

- [ ] **Всегда задавайте width и height**
  Чтобы браузер резервировал место до загрузки изображения, что предотвращает сдвиги макета.

- [ ] **Стремитесь к лёгкому главному изображению**
  В качестве рабочей цели держите изображение LCP ниже примерно 200 КБ на мобильных; сжимайте, пока не увидите разницу, затем сделайте шаг назад.

- [ ] **Используйте CDN или сервис изображений**
  Для автоматического изменения размера, конвертации и кэширования, чтобы редактор случайно не загрузил фото на 6 МБ.

КАК ПРОВЕРИТЬ

Запустите PageSpeed Insights снова и убедитесь, что элемент LCP и его время улучшились. Каскад в WebPageTest показывает, когда изображение действительно начинает загружаться.

03 Сократите вес JavaScript и сторонних скриптов

JavaScript — главная причина, по которой страницы кажутся неотзывчивыми, а именно отзывчивость и измеряет INP.

- [ ] ОБЯЗАТЕЛЬНО **Составьте список всех сторонних скриптов и их владельцев**
  Аналитика, чат, реклама, пиксели, A/B-инструменты, баннеры согласия, тепловые карты. Уберите всё, на что никто не смотрел последний квартал.

- [ ] ОБЯЗАТЕЛЬНО **Загружайте второстепенные скрипты позже**
  Используйте `defer` или `async` и загружайте виджеты чата и видео только при взаимодействии посетителя. Статичная картинка-превью («facade») вместо iframe YouTube экономит сотни килобайт.

- [ ] **Разбивайте длинные задачи**
  Любая задача дольше 50 мс блокирует реакцию страницы. Найдите их на панели Performance и разделите работу или вынесите её из основного потока.

- [ ] **Удалите неиспользуемый код**
  Проверьте покрытие в DevTools. Многие сайты поставляют целую библиотеку ради одной маленькой функции.

- [ ] **Проведите аудит менеджера тегов**
  Один контейнер может загружать десятки тегов на каждой странице. Ограничьте теги теми страницами, где они нужны.

- [ ] **Тестируйте с ограничением процессора**
  Используйте замедление 4x в DevTools, чтобы приблизиться к телефону среднего класса; ваш ноутбук скрывает проблему.

КАК ПРОВЕРИТЬ

Нажмите на меню, фильтр или «В корзину» сразу после загрузки страницы. Если заметна задержка, у INP, вероятно, проблемы; подтвердите это в полевых данных.

04 Шрифты, CSS и стабильность макета

Текст должен появляться сразу и оставаться на месте. Поздние шрифты, поздний CSS и поздние баннеры — обычные причины «прыгающего» контента.

- [ ] ОБЯЗАТЕЛЬНО **Резервируйте место для всего, что загружается позже**
  Реклама, встроенный контент, баннеры cookie, панели подписки и изображения. Используйте фиксированные размеры или свойство CSS `aspect-ratio`, чтобы ничто не сдвигало контент вниз.
- [ ] **Размещайте шрифты у себя в формате WOFF2**
  Это убирает лишнее соединение с другим доменом и даёт контроль над кэшированием.
- [ ] **Ограничьте количество файлов и начертаний шрифтов**
  Двух семейств и трёх начертаний вполне достаточно. Делайте подмножество только для нужных языков.
- [ ] **Используйте `font-display: swap` и предзагружайте ключевой файл**
  Предзагружайте только один-два шрифта первого экрана с атрибутом `crossorigin`.
- [ ] **Уменьшите сдвиг при замене шрифта**
  Подгоните метрики запасного шрифта через `size-adjust` и связанные дескрипторы, чтобы замена почти не двигала текст.
- [ ] **Держите критический CSS небольшим**
  Встройте то, что нужно для первого экрана, а остальное загружайте, не блокируя отрисовку. Удалите CSS фреймворков, которыми больше не пользуетесь.
- [ ] **Предпочитайте встроенный SVG иконочным шрифтам**
  Несколько SVG легче и отображаются, не дожидаясь шрифта.

КАК ПРОВЕРИТЬ

Загрузите страницу при ограниченном соединении и следите за движением. В Chrome DevTools опция Layout Shift Regions подсвечивает именно то, что сдвинулось.

05 Хостинг, кэширование и ответ сервера

Быстрый фронтенд не спасёт медленный сервер. Time to First Byte (TTFB) задерживает всё, что идёт дальше.

- [ ] ОБЯЗАТЕЛЬНО **Измеряйте TTFB оттуда, где находятся ваши клиенты**
  Хороший TTFB — около 0,8 с и меньше, плохой — больше 1,8 с. Сервер на другом континенте добавляет задержку ещё до выполнения любого кода.
- [ ] ОБЯЗАТЕЛЬНО **Кэшируйте статические файлы надолго**
  CSS, JS, шрифты и изображения с отпечатком в имени могут использовать `Cache-Control: public, max-age=31536000, immutable`. Повторные визиты тогда почти ничего не загружают.

- [ ] **Включите сжатие Brotli или Gzip**
  Обычно оно уменьшает текстовые файлы (HTML, CSS, JS, SVG) больше чем вдвое.

- [ ] **Поставьте CDN перед сайтом**
  Он отдаёт файлы из точки рядом с посетителем и сглаживает всплески трафика.

- [ ] **Кэшируйте HTML осторожно**
  Безопасно для публичных страниц, если можно сбрасывать кэш при каждом обновлении; никогда для корзины, аккаунтов и оформления заказа.

- [ ] **Уберите цепочки редиректов**
  Каждый редирект добавляет полный круг запроса. Ссылайтесь сразу на конечный адрес, включая https:// и завершающий слэш.

- [ ] **Проверьте медленные запросы и тяжёлые плагины**
  Если TTFB высокий только на динамических страницах, причина скорее в базе данных или плагине, чем в сети.

КАК ПРОВЕРИТЬ

На вкладке Network в DevTools нажмите на основной документ и посмотрите «Waiting for server response». Повторите с телефона по мобильному интернету.

06 Тестируйте на реальных устройствах и следите после запуска

Производительность «плывёт». Новые изображения, скрипты и функции сводят на нет хорошую работу, если за цифры никто не отвечает.

- [ ] ОБЯЗАТЕЛЬНО **Тестируйте на телефоне среднего класса и при медленном соединении**
  Chrome DevTools предлагает Slow 4G и ограничение CPU, но реальный Android среднего класса честнее.

- [ ] ОБЯЗАТЕЛЬНО **Задайте бюджет производительности**
  Например, изображение LCP меньше 200 КБ, JavaScript меньше 300 КБ в сжатом виде, ограниченное число сторонних тегов. Согласуйте его с теми, кто добавляет функции.

- [ ] **Постоянно измеряйте реальных пользователей**
  Библиотека web-vitals отправляет LCP, INP и CLS из реальных визитов в вашу аналитику, чтобы не ждать 28-дневного отчёта.

- [ ] **Добавьте проверки в процесс релиза**
  Lighthouse CI или регулярный тест PageSpeed предупредит, когда изменение замедляет ключевую страницу.

- [ ] **Раз в месяц разбирайте самые медленные страницы**
  Исправьте главную причину, запишите результат, повторите.

- [ ] **Тестируйте заново после изменений шаблонов, тем или плагинов**
  Одно обновление плагина может добавить 200 КБ скриптов.

КАК ПРОВЕРИТЬ

Сравните отчёт Core Web Vitals в Search Console за этот месяц с прошлым. Число адресов в категории «Плохо» должно снижаться, а не расти.

Источники и дополнительное чтение

web.dev: Web Vitals (пороговые значения)
https://web.dev/articles/vitals

web.dev: Interaction to Next Paint
https://web.dev/articles/inp

web.dev: Оптимизация Largest Contentful Paint
https://web.dev/articles/optimize-lcp

web.dev: Time to First Byte
https://web.dev/articles/ttfb

Нужна помощь?

Midnight Webs проектирует, защищает и поддерживает сайты для малого бизнеса. Расскажите, что нужно, и мы подготовим план работ.

midnightwebs.com/ru/estimate

Читать онлайн: midnightwebs.com/ru/resources/website-speed-core-web-vitals-checklist