So testen Sie Ihre WordPress-Site auf Funktionalität, Geschwindigkeit und Sicherheit

Warum sind WordPress-Tests wichtig?

Es gibt viele Vorteile, wenn du deine WordPress-Website regelmäßig testest. Wie bereits erwähnt, kannst du mit dem Design und den Elementen der Benutzeroberfläche (UI) experimentieren, ohne dass dies Auswirkungen auf deine Live-Site hat.

So kannst du deine aktuelle Website beibehalten und den Geschäftsbetrieb aufrechterhalten, während du neue Ideen ausprobierst. Wenn in der Testumgebung etwas schief geht, musst du dir keine Sorgen über die Auswirkungen machen, die ein Ausfall auf deinen Webverkehr und deine Einnahmen haben könnte.

Andererseits kannst du deine WordPress-Website auch testen, um Probleme zu erkennen, die Besucher/innen haben könnten, wenn sie versuchen, deine Seiten aufzurufen. Zum Beispiel kann es sein, dass deine Seite in einem bestimmten Browser langsam läuft oder dass dein Menü auf mobilen Geräten nicht richtig angezeigt wird.

Außerdem kann eine Testumgebung eine gute Möglichkeit sein, um Sicherheitslücken zu vermeiden. Vielleicht möchtest du neue Plugins und Themes ausprobieren, bevor du sie auf deiner Website installierst. In der Zwischenzeit kannst du Updates auf deiner Testseite durchführen, um sicherzustellen, dass sie sicher sind.

Während viele Anfänger/innen davon profitieren können, mit WordPress in einem sicheren, privaten Umfeld zu experimentieren, ist das Testen auch für fortgeschrittene Entwickler/innen sehr wichtig. Mit den richtigen Tools können Entwickler/innen eine permanente Testumgebung einrichten, um die Funktionalität ihrer Produkte zu testen, bevor sie sie der Öffentlichkeit zugänglich machen.

Was sind die gängigsten Arten von Tests?

Da du nun weißt, warum es wichtig ist, WordPress sicher zu testen, werfen wir einen Blick auf einige der gängigsten Methoden.

- **Funktionstests.** So kannst du dir ein genaues Bild davon machen, wie sich die Nutzer/innen auf deiner Seite bewegen. Du kannst zum Beispiel überprüfen, ob Formulare, Buttons und Checkout-Seiten richtig funktionieren.
- **Leistungs- und Geschwindigkeitstests.** Wenn du sicherstellst, dass deine Website schnelle Ladezeiten hat, kannst du die Benutzerfreundlichkeit verbessern, die Suchmaschinenoptimierung (SEO) unterstützen und deine Core Web Vitals verbessern.
- **Sicherheitstests.** Dazu gehört die Analyse der Sicherheitsmechanismen auf deiner Website, wie SSL-Zertifikate, HTTPS, Web Application Firewalls und mehr. Sie hilft dir, sensible Daten zu schützen, böswillige Angriffe zu verhindern und WordPress-Schwachstellen zu erkennen.

Unabhängig davon, welche Art von Website du betreibst, solltest du dir angewöhnen, regelmäßig Funktions-, Leistungs- und Sicherheitstests durchzuführen.

Best Practices für WordPress-Tests

Es ist wichtig, den Wert der Tests deiner Website in verschiedenen Umgebungen zu erkennen. Wenn du den Unterschied zwischen den verschiedenen Umgebungen kennst, ist es einfacher, die richtige Option für deine Bedürfnisse zu wählen.

Eine lokale Umgebung wird auf deinem eigenen Computer gehostet. Daher hat nichts, was du dort tust, Auswirkungen auf deine Live-Site. Für den allgemeinen Gebrauch bietet sie eine gute Möglichkeit, neue Funktionen und Features zu testen. Für Entwicklerinnen und Entwickler ist eine lokale Umgebung der ideale Ort, um Bugs und Fehler in deinem Code zu finden.

Eine Staging-Umgebung hingegen bietet eine Kopie der Daten deiner Website auf einem Server (und nicht auf einem lokalen Rechner). Sie ist der ideale Ort, um größere Versions-Updates, Konfigurationsänderungen und Datenbankmigrationen durchzuführen. Wenn du Websites für Kunden entwirfst, eignet sich eine Staging-Site außerdem gut als Demo-Site, um den Kunden zu zeigen, wie die Website aussehen wird.

Wie du Testumgebungen einrichtest

Jetzt, wo du die verschiedenen Arten von Testumgebungen besser kennst, schauen wir uns an, wie du sie einrichtest!

So richtest du eine Testumgebung mit einer Staging-Site ein

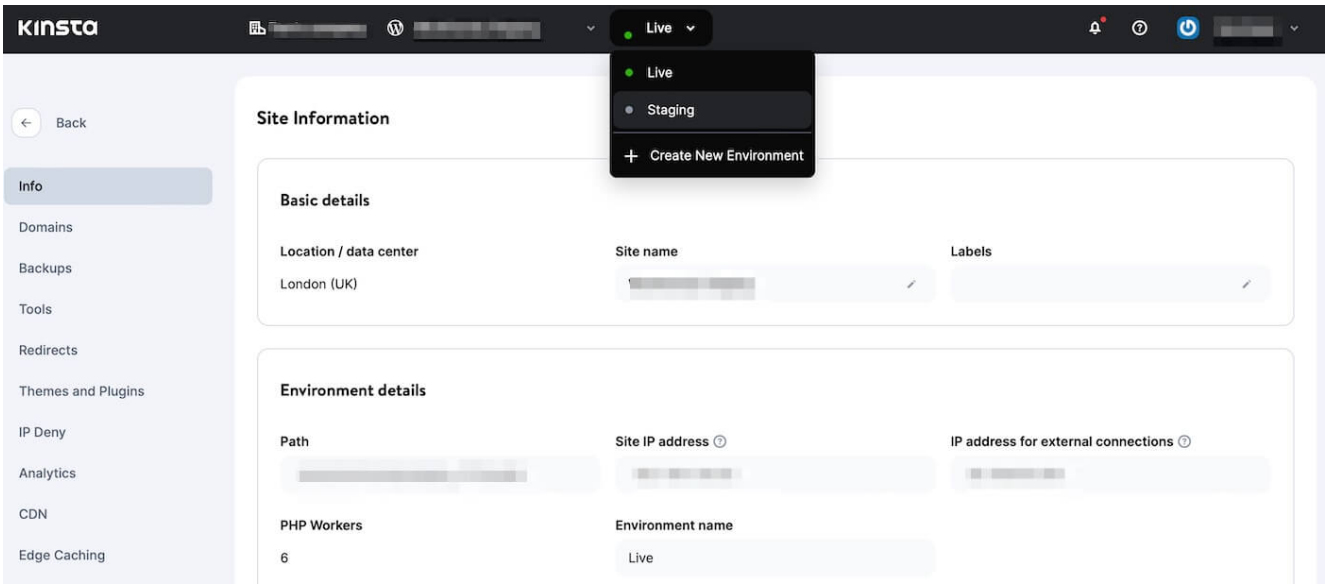
Wie bereits erwähnt, ist eine Staging-Site im Grunde eine vollständige Kopie deiner Live-Website. Normalerweise wird sie auf demselben Server gehostet wie deine Live-Website. Der einzige Unterschied besteht darin, dass Besucher/innen nicht auf sie zugreifen können.

Das Beste am Staging ist, dass es einem realen Aufbau folgt.

So kannst du genau nachvollziehen, wie sich die Kunden auf deinen Seiten bewegen.

Der einfachste Weg, eine Staging-Site einzurichten, führt über deinen Webhoster. Nicht alle Webhoster bieten Staging-Umgebungen mit ihren Hosting-Diensten an. Aber bei Kinsta ist es super einfach, die integrierte WordPress-Staging-Umgebung zu erstellen und zu konfigurieren.

Du kannst auf deine Staging-Site zugreifen, indem du dich in dein MyKinsta-Dashboard einloggst. Wähle einfach deine Website aus der Liste aus. Oben auf dem Bildschirm kannst du dann über das Dropdown-Menü von **Live** zu **Staging** wechseln:



Eine Staging-Site mit Kinsta einrichten

Denke daran, dass es bis zu fünfzehn Minuten dauern kann, bis deine Staging-Site zum ersten Mal erstellt wird. Danach wird sie als Subdomain deiner Hauptdomain existieren (beide nutzen denselben Server).

Sobald du bereit bist, die Änderungen auf deine Live-Website zu übertragen, kannst du einfach die Schaltfläche **Push-Umgebung** in deinem Dashboard verwenden.

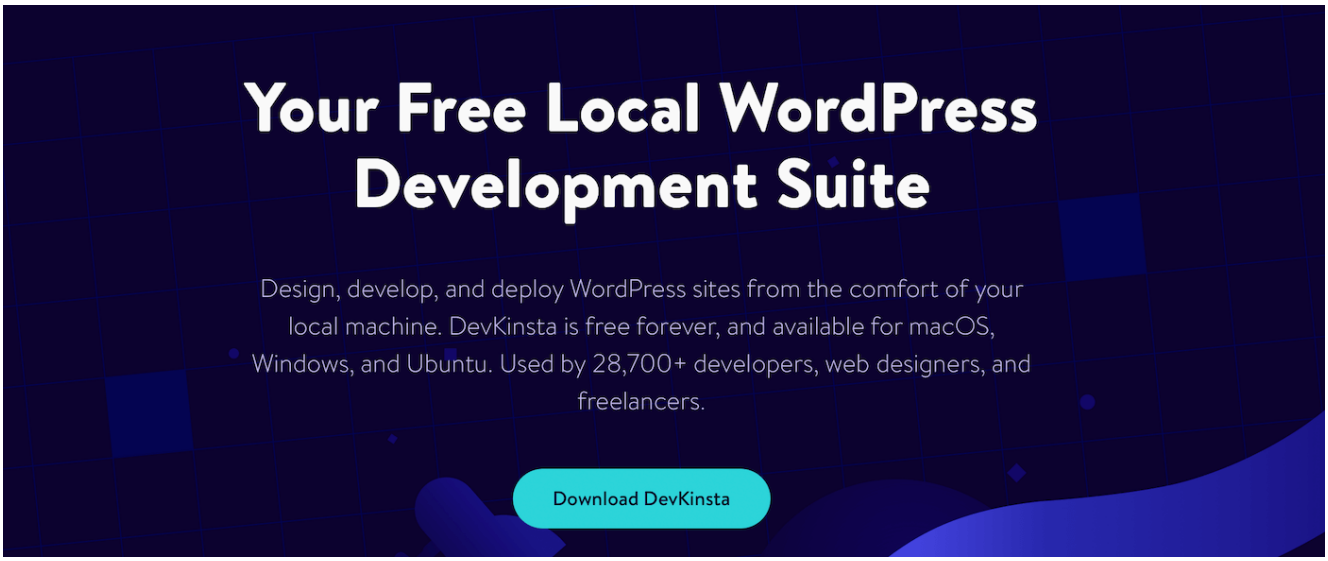
Wie du eine lokale Testumgebung einrichtest

Eine lokale Umgebung funktioniert ähnlich wie eine Staging-Site, allerdings musst du die Umgebung nicht extern hosten.

Stattdessen befindet sich deine lokale Umgebung auf einem lokalen Rechner (meistens auf deinem Computer).

Um eine WordPress-Testumgebung lokal zu installieren, musst du dir einen AMP-Stack für deinen Computer besorgen. Diese Software (Apache, MySQL und PHP) wird verwendet, um deine echte WordPress-Website zu imitieren.

Einige der beliebtesten Möglichkeiten, WordPress lokal zu installieren, sind WAMP und XAMPP. Der einfachste Weg ist jedoch die Verwendung von DevKinsta:



Verwende DevKinsta, um eine lokale Umgebung zu erstellen

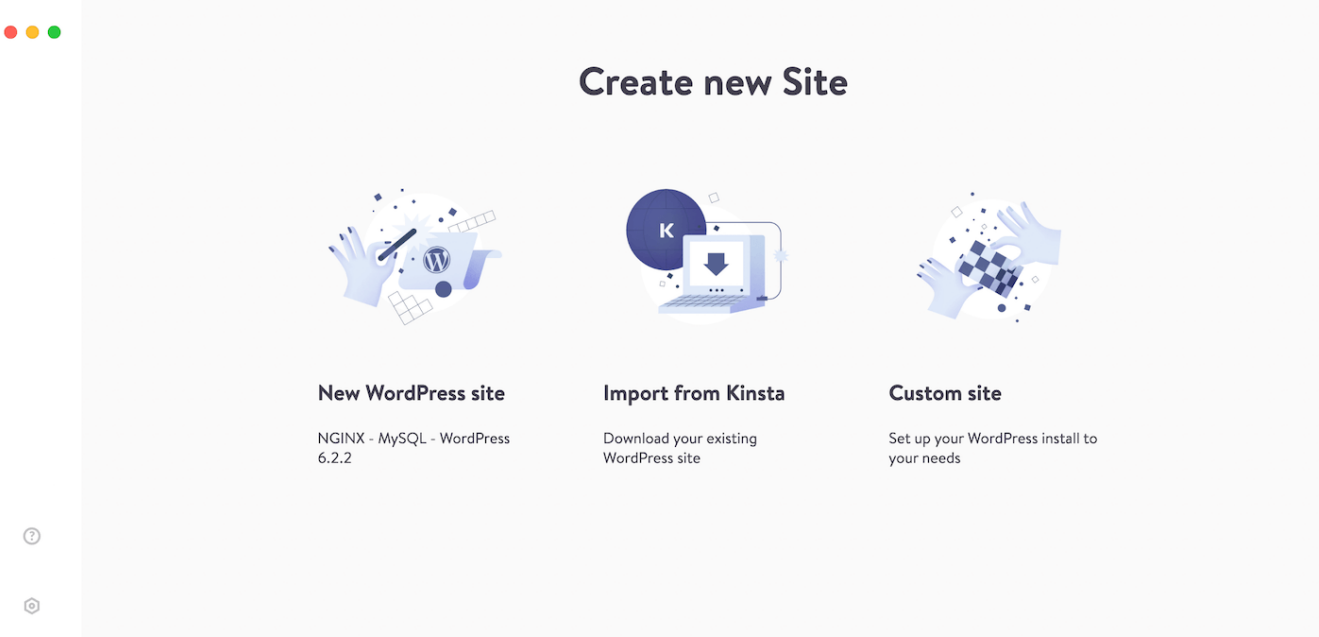
DevKinsta ist ein kostenloses lokales Entwicklungstool für WordPress Single oder Multisite. Mit DevKinsta erhältst du Zugang zu einer Vielzahl von Datenbank- und E-Mail-Verwaltungstools. Außerdem lässt es sich nahtlos in MyKinsta integrieren (du musst allerdings kein Kinsta-Kunde sein, um DevKinsta zu nutzen).

Um loszulegen, musst du die neueste Version von DevKinsta herunterladen. Auf dem Mac fügst du DevKinsta zu **Programme** hinzu und öffnest die DevKinsta-App mit einem Doppelklick.

Der Installationsprozess unterscheidet sich leicht von Betriebssystem zu Betriebssystem, aber du kannst bei Bedarf die vollständige Installationsanleitung für DevKinsta einsehen. Anschließend kannst du Docker Desktop installieren,

um Container für das lokale WordPress zu erstellen.

Sobald du DevKinsta und Docker erfolgreich installiert hast, kannst du deine lokale Website erstellen. Du kannst entweder eine neue WordPress-Site erstellen, eine bestehende Site von Kinsta importieren oder eine eigene Site erstellen:



Eine lokale Website mit DevKinsta erstellen

Wähle einfach deine bevorzugte Option. Wenn du eine Website von Kinsta importierst, musst du die richtige Website für den Import auswählen und deine Anmeldedaten eingeben. Dann wirst du zum Bildschirm mit **den Website-Informationen** weitergeleitet, der wie ein Dashboard für deine lokale Umgebung funktioniert.

Du kannst die Kinsta-API auch nutzen, um eine neue WordPress-Seite/Installation zu erstellen, ohne auf DevKinsta zuzugreifen.

So testest du die Funktionalität deiner WordPress-Website (5 Funktionen)

Sehen wir uns nun fünf Möglichkeiten an, wie du die

Funktionalität deiner WordPress-Website testen kannst. Das Beste an den Funktionstests ist, dass du sie direkt in deiner lokalen Umgebung oder mit DevKinsta durchführen kannst (im Gegensatz zu anderen Testarten, bei denen deine Website live sein muss).

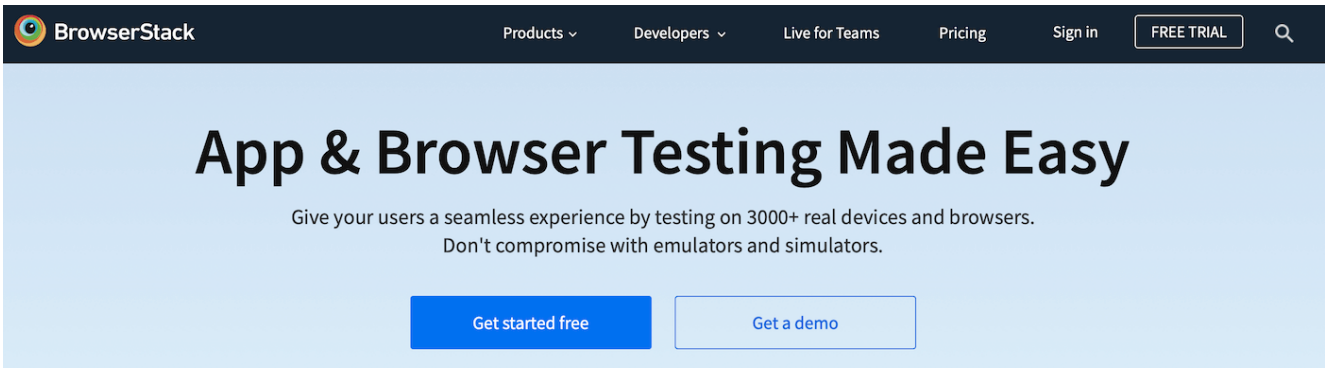
Cross-Browser-Unterstützung

Es ist wichtig, deine WordPress-Website in verschiedenen Browsern zu testen, um zu sehen, wie deine Website für alle Besucher aussieht. Das liegt daran, dass verschiedene Browser unterschiedlichen Code verwenden. Daher behandelt und zeigt jeder Browser Elemente auf seine eigene Weise an.

Ein Nutzer, der deine Website mit Chrome aufruft, sieht sie vielleicht anders als ein Nutzer, der deine Website mit Firefox besucht. Und obwohl 3,2 Milliarden Internetnutzer/innen im Jahr 2021 Chrome als Hauptbrowser bevorzugen, nutzen viele weiterhin Firefox, Edge, Opera und Safari.

Vielleicht möchtest du herausfinden, welche Browser bei deinen Besuchern beliebt sind, um deine Seite speziell für diese Browser zu optimieren. Wenn du Google Analytics verwendest, kannst du diese Informationen in deinen Besucherberichten finden.

Dann kannst du deine Website mit einem Tool wie BrowserStack auf Browserunterstützung testen:



Browserübergreifende Tests mit BrowserStack durchführen

Mit BrowserStack kannst du deine Website in 3000 verschiedenen Browsern testen, darunter die neuesten Versionen von Edge, Safari, Firefox und Chrome. Du kannst auch eine kostenlose Testversion nutzen, bevor du dich für einen kostenpflichtigen Plan entscheidest.

Unit-Tests

Beim Unit Testing wird die kleinste Einheit einer Anwendung isoliert getestet. Das kann eine Funktion, eine Eigenschaft oder eine Methode sein. Diese Einheiten werden dann auf ihre Funktionstüchtigkeit untersucht, um sicherzustellen, dass sich die Anwendung wie erwartet verhält.

Du kannst Unit-Tests automatisch mit einem Drittanbieter-Tool wie Travis CI durchführen. Es ist jedoch schneller, die Tests lokal während der Entwicklung durchzuführen, als Änderungen vorzunehmen und darauf zu warten, dass Travis CI sie ausführt.

Du könntest zum Beispiel ein Theme oder ein Plugin einem Unit-Test unterziehen. Hierfür musst du Git, SVN, PHP und Apache installieren. Außerdem musst du dein Plugin fertig haben.

Um loszulegen, öffne DevKinsta, um deine lokale Entwicklungsumgebung zu starten. Installiere dann PHPUnit. Nun musst du die Testdateien für das Plugin mit folgendem Befehl erstellen:

bash
```
wp scaffold plugin-tests my-plugin
```

Jetzt kannst du die Testumgebung lokal initialisieren, indem du das Installationsskript ausführst:

bash
```
bash bin/install-wp-tests.sh wordpress_test root '' localhost latest
```

Dieses Skript installiert eine Kopie von WordPress auf /tmp directory und in den WordPress Unit Testing Tools.

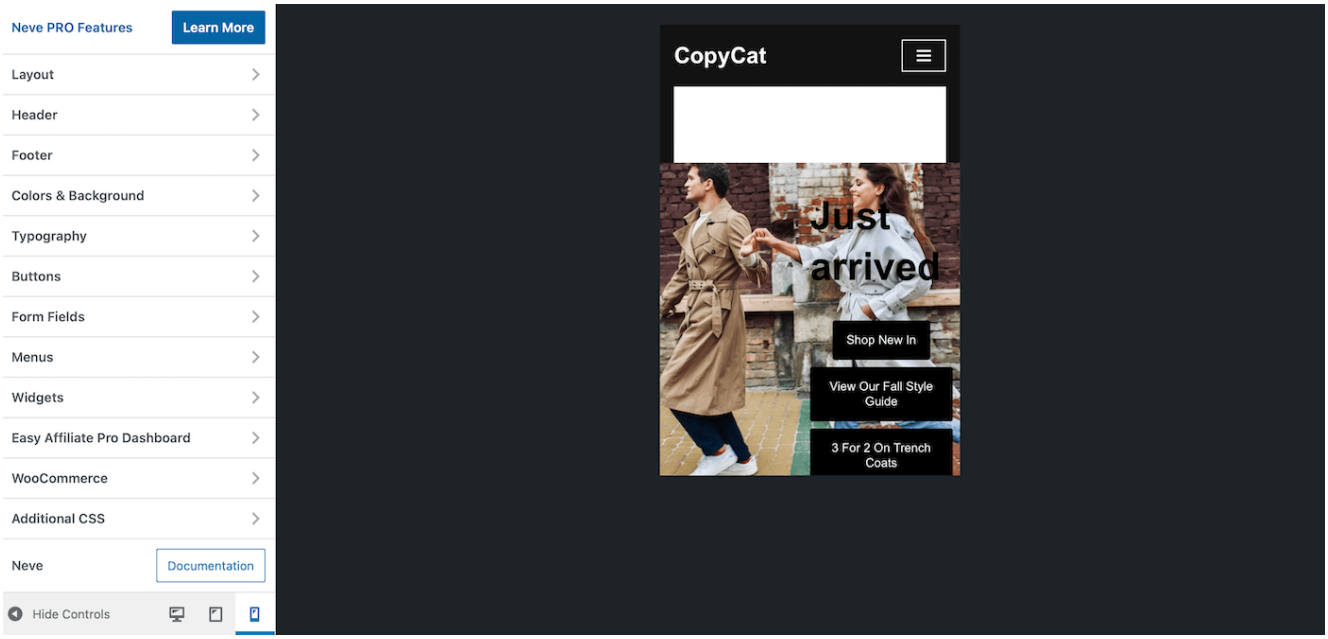
Im nächsten Schritt führst du die Plugin-Tests mit Hilfe von phpunit aus. Eine ausführliche Anleitung findest du in diesem Leitfaden zu Unit-Tests.

Responsivität für Mobilgeräte/Desktop

Da über 60 Prozent der Menschen mit einem mobilen Gerät online gehen, ist es wichtiger denn je, dass deine WordPress-Website responsive ist. Auf diese Weise kannst du sicherstellen, dass deine Seiten auf allen Bildschirmgrößen, einschließlich Desktop, Tablet und Handy, reibungslos angezeigt werden.

Am einfachsten kannst du die Reaktionsfähigkeit deiner Website testen, indem du die URL deiner Website auf deinem mobilen Gerät eingibst. Wenn du jedoch die Darstellung deiner Website von deinem Desktop aus testen möchtest, kannst du den WordPress Customizer verwenden.

Gehe einfach zu **Darstellung** > **Anpassen:**

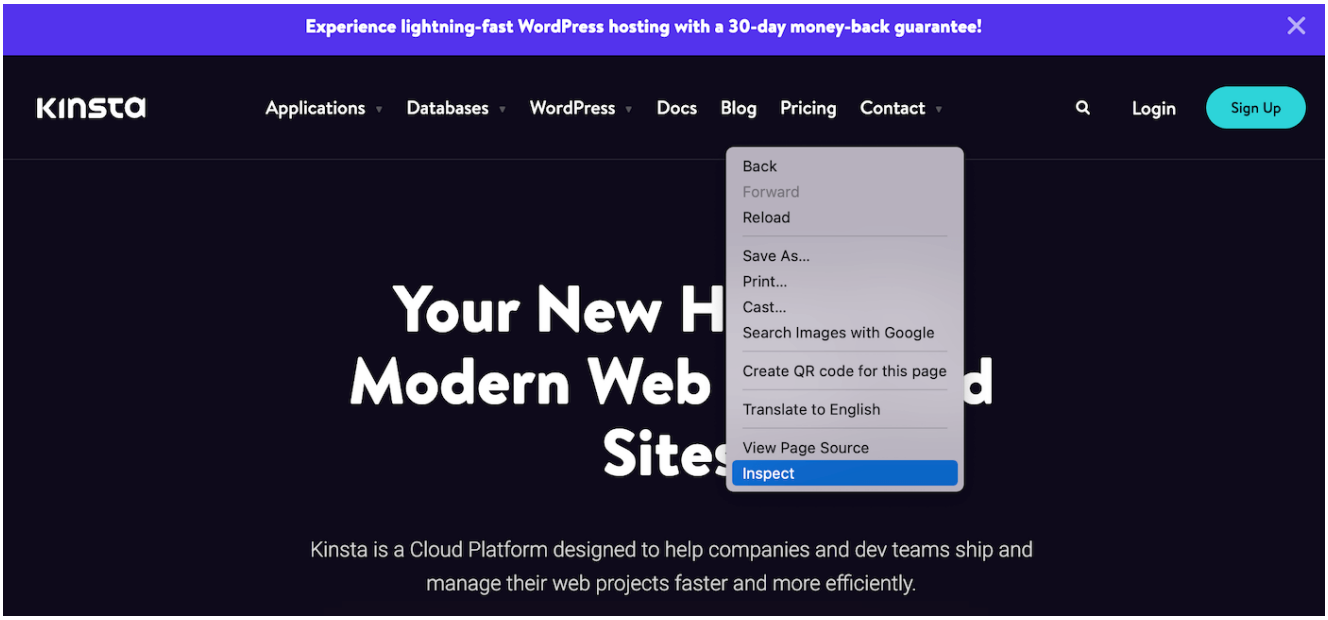


Teste die Reaktionsfähigkeit von WordPress mit dem WordPress Customizer

Je nach Theme siehst du unterschiedliche Panels. Unten auf deiner Seite kannst du auf das Symbol für Mobilgeräte oder Tablets klicken, um eine Vorschau deiner Website in der angegebenen Bildschirmgröße anzuzeigen.

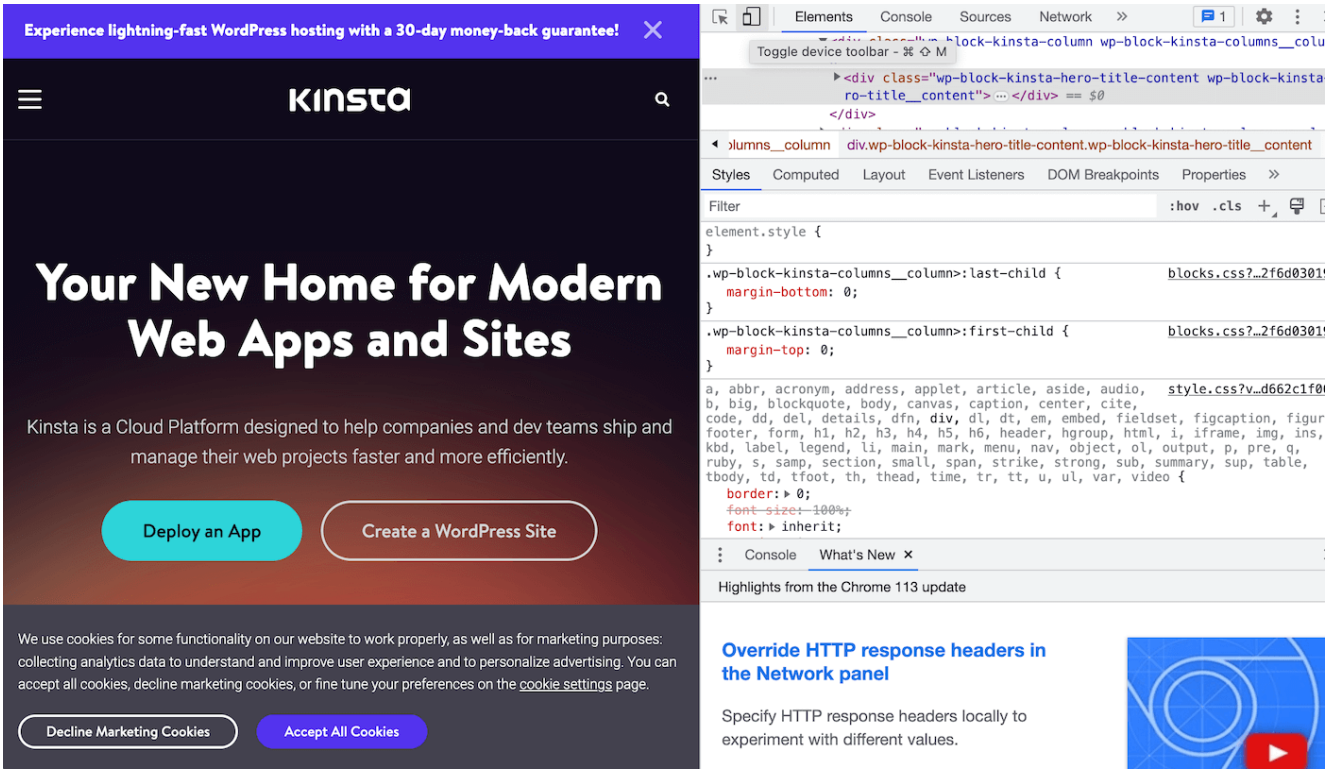
Außerdem kannst du auf die Entwicklertools von Google Chrome zugreifen, um zu sehen, wie deine WordPress-Website auf mobilen Geräten aussieht. Dazu musst du nur eine Seite deiner Website in Google Chrome öffnen.

Dann klickst du mit der rechten Maustaste auf die Seite und wählst **Inspizieren**:

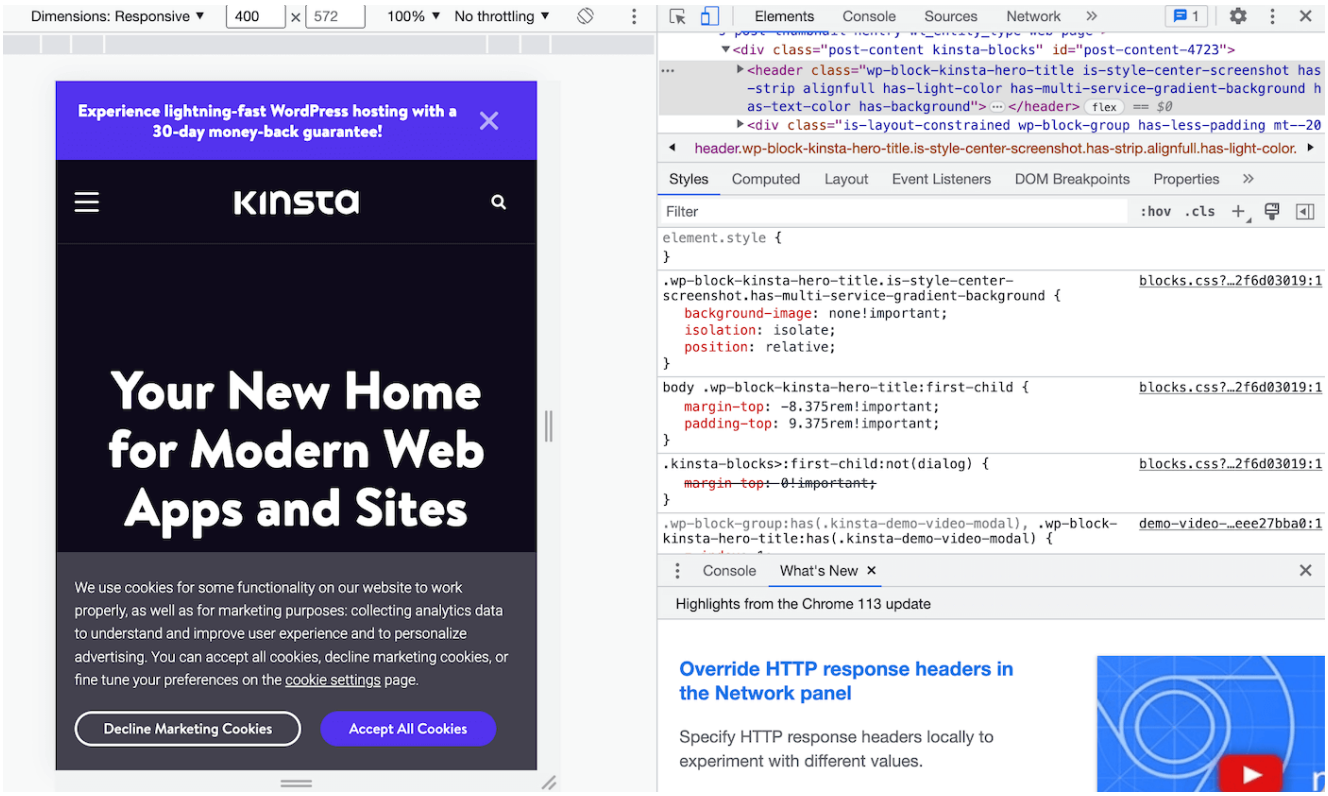


Teste die Reaktionsfähigkeit von WordPress mit Google Chrome Inspect

Jetzt suchst du die **Symbolleiste Gerät umschalten** oben im Popup (links neben dem Reiter **Elemente**):



Klicke auf die Symbolleiste Gerät umschalten in Chrome Inspect Klicke darauf und dein Bildschirm wird sofort angepasst:



Betrachte deine Website in der mobilen Ansicht mit Google Chrome Inspect

Wie du siehst, kannst du jetzt testen, wie deine Website in **Responsive** Dimensionen angezeigt wird. Wenn du auf das Dropdown-Menü **Dimensionen** klickst, kannst du deine Seite auf

weiteren Geräten testen, z. B. auf verschiedenen iPhone- und Samsung Galaxy-Modellen.

Testen der Benutzeroberfläche (UI)

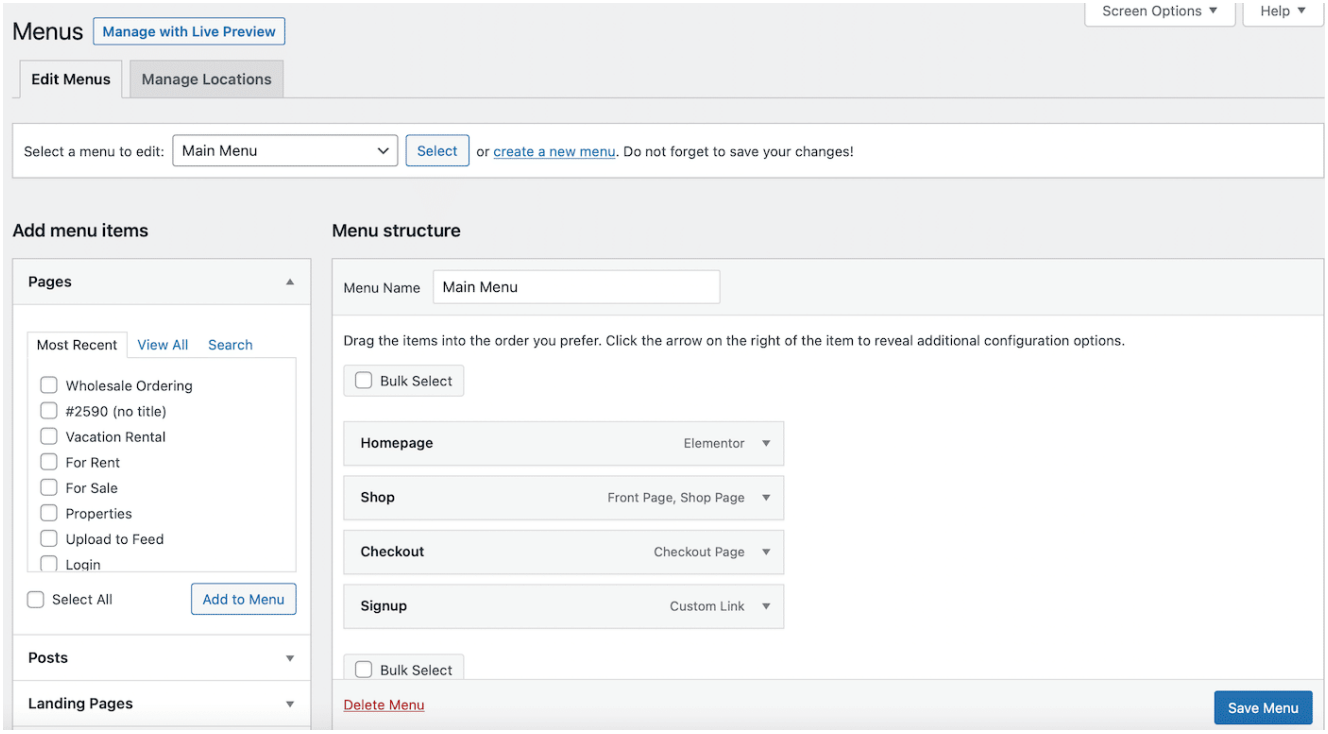
Wenn wir von der Benutzeroberfläche (User Interface, UI) deiner Website sprechen, meinen wir damit alle Komponenten deiner Website, mit denen Besucher interagieren können. Die meisten Websites enthalten zum Beispiel Links, Schaltflächen, Menüs usw. Irgendwann müssen die Nutzer mit diesen Elementen interagieren.

Deshalb ist es besonders wichtig, deine Benutzeroberfläche zu testen. Wenn etwas nicht richtig funktioniert, können Besucher/innen frustriert sein und deine Seite verlassen.

Du kannst eine lokale Umgebung einrichten, um deine UI-Elemente zu testen. Du könntest zum Beispiel ein neues Navigationsmenü entwickeln und es ausprobieren.

In diesem Fall kannst du in deinem DevKinsta-Dashboard deinen lokalen Verwaltungsbereich öffnen. Dann navigierst du auf der lokalen Seite zu **Erscheinungsbild** > **Menüs** . Jetzt klickst du auf **Neues Menü erstellen**.

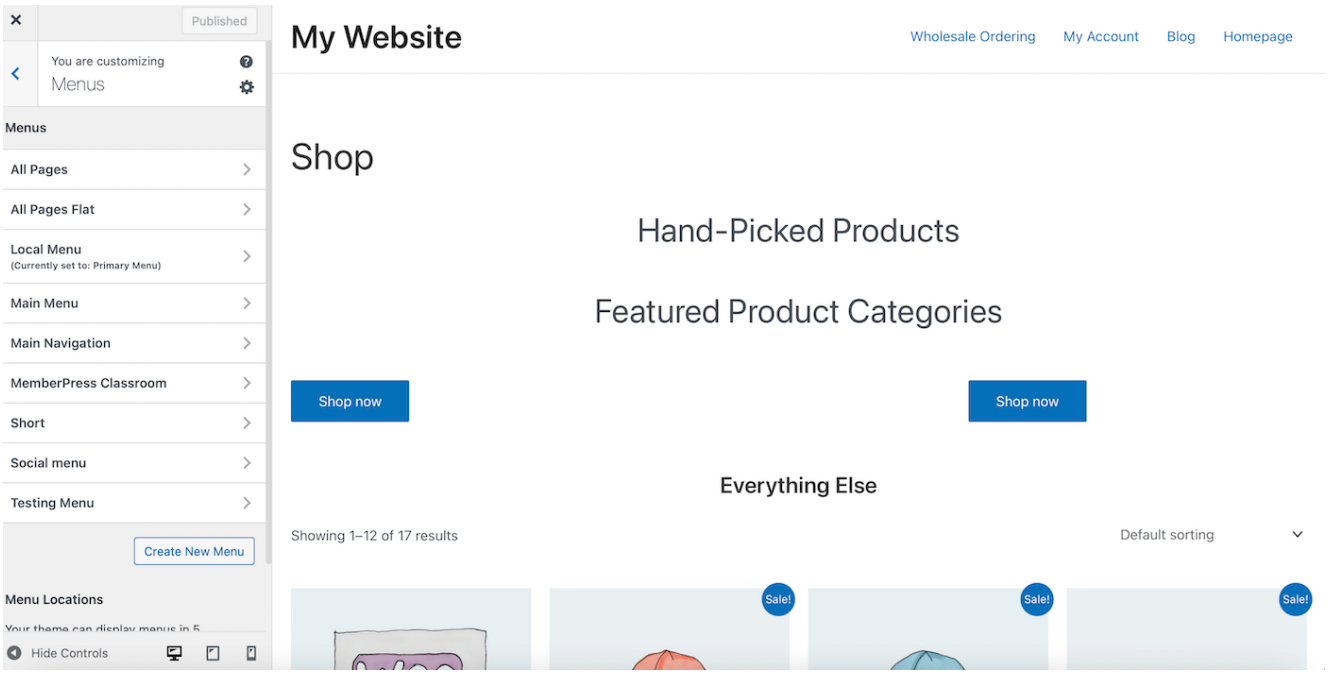
Gib deinem Menü einen Namen und klicke auf **Menü speichern**. Füge dann auf der linken Seite deines Bildschirms Menüpunkte hinzu und wähle **Zu Menü hinzufügen**:



Lokales WordPress für UI-Tests

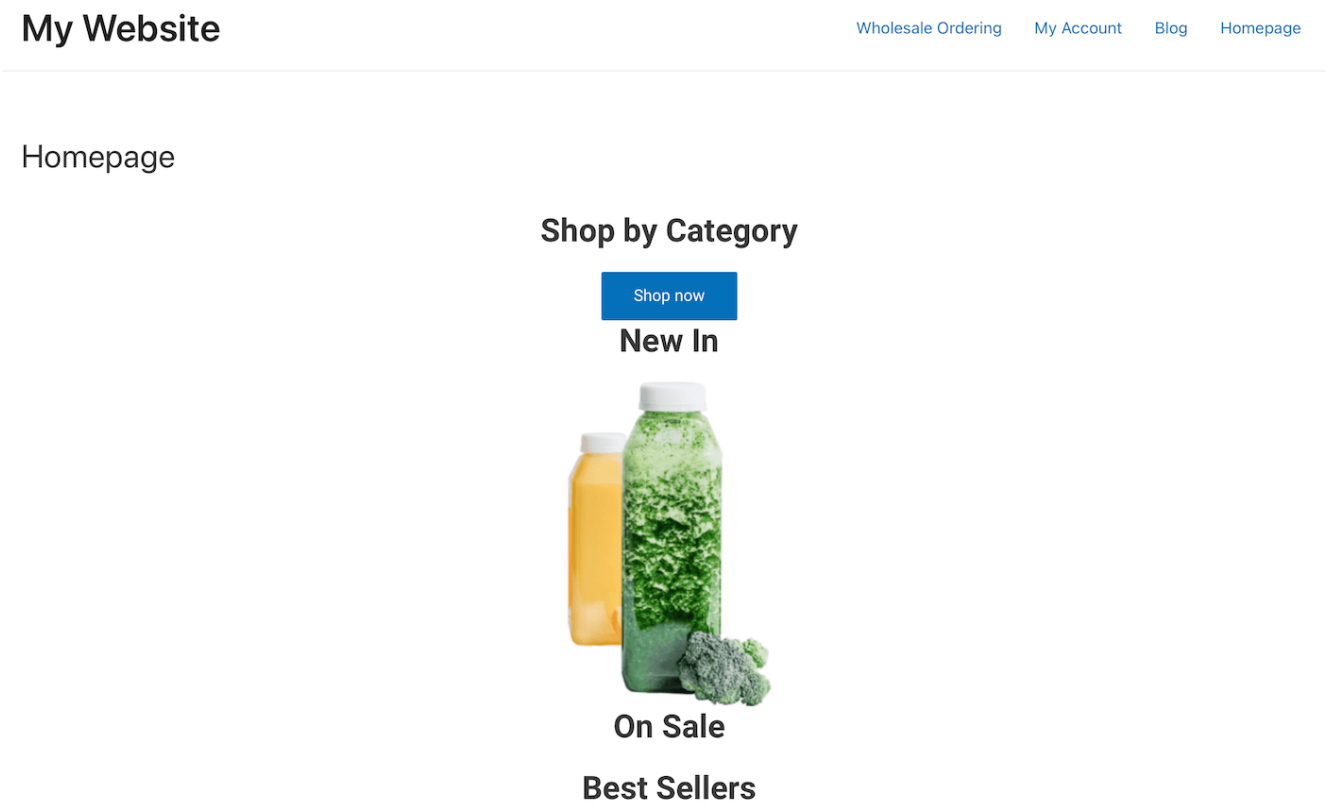
Aktiviere unter **Menüeinstellungen** das Kästchen **Primäres Menü**. Dann speicherst du deine Änderungen.

Du kannst auch auf **Verwalten mit Live-Vorschau** oben auf dem Bildschirm klicken, um zu sehen, wie sich dein Menü macht:



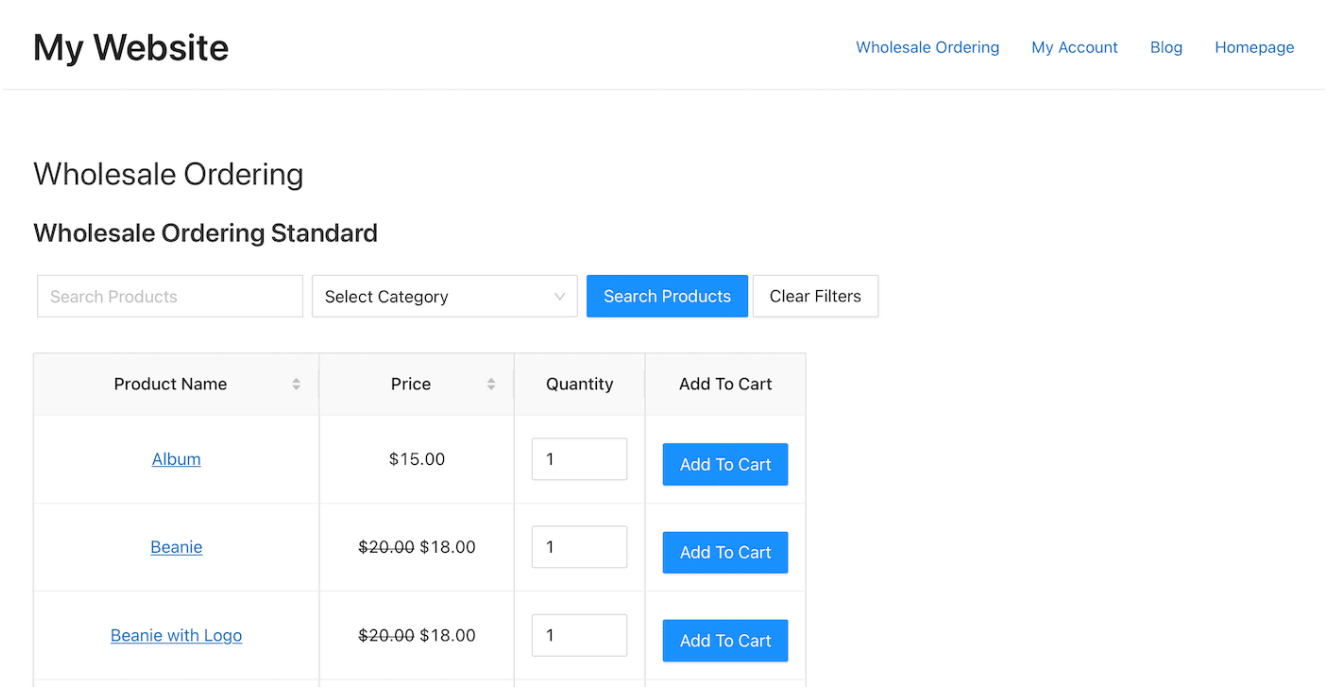
Dein lokales Menü mit Live-Vorschau anzeigen

Als Nächstes öffnest du deine lokale Website in einem neuen Browser, um dein neues Menü auf dem Frontend zu sehen:



Teste deine Benutzeroberfläche in deiner lokalen Umgebung

Du kannst auch die Navigationslinks testen, indem du auf jeden der Menüpunkte klickst. Wenn wir zum Beispiel auf den Link **Großhandelsbestellung** klicken, werden wir zu der entsprechenden Seite weitergeleitet, die wir unserem Menü hinzugefügt haben:



Menülinktests in lokaler Umgebung

Auf diese Weise kannst du neue Designelemente testen und

sicherstellen, dass deine Benutzeroberfläche richtig funktioniert.

Visuelle Tests

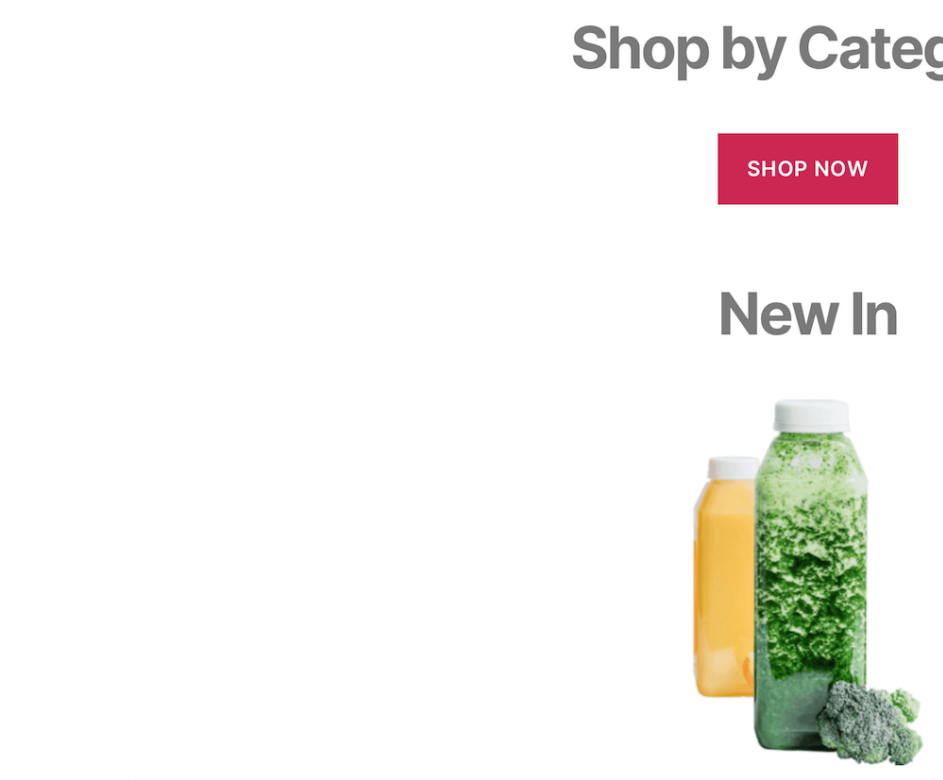
Visuelle Regressionstests (VRT) stellen sicher, dass alle deine Designelemente und Layouts so aussehen, wie sie sollen. Aus diesem Grund wird VRT oft nach Änderungen an der Website durchgeführt, z. B. wenn du das Theme wechselst oder ein Plugin aktualisierst.

Auf diese Weise kannst du sicherstellen, dass die Änderungen deine visuellen Elemente nicht beeinträchtigen. So kann es zum Beispiel sein, dass dein Inhalt falsch ausgerichtet ist oder Schaltflächen verschwunden sind.

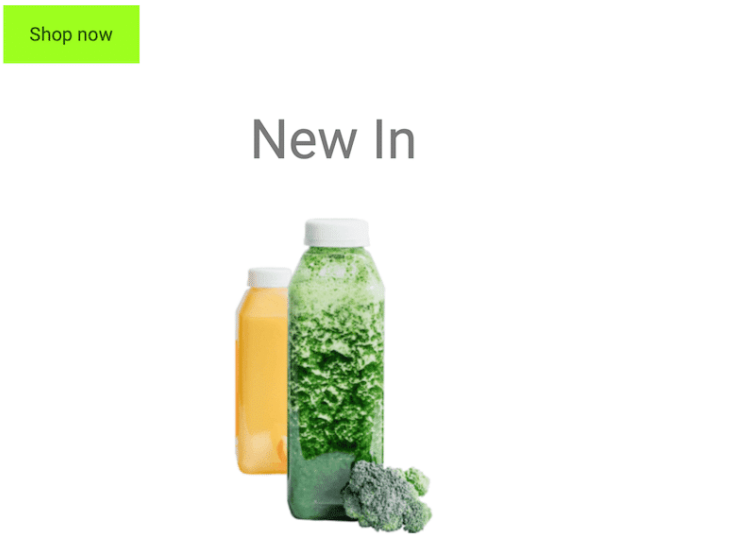
Wie bei den UI-Tests würdest du solche Probleme oft gar nicht bemerken, wenn du deine Website nicht am Frontend besuchst. Es gibt automatisierte VRT-Tools, die deine Website kontinuierlich auf visuelle Anomalien prüfen.

Oder du kannst deine Seiten einfach manuell vergleichen, bevor und nachdem du deine Änderungen vorgenommen hast. Angenommen, du willst das Theme wechseln. Am sichersten ist es, dies in einer lokalen Umgebung wie DevKinsta zu tun, damit du visuelle Tests durchführen kannst, bevor du die Änderung auf deiner Live-Website anwendest.

Im Moment haben wir das Twenty Twenty-Theme auf unserer lokalen Website aktiviert. Wie du siehst, sind auf der Startseite alle Schaltflächen, Texte und Bilder mittig angeordnet:



Visuelle Tests in DevKinsta durchführen
Wenn wir jedoch zum Twenty Twenty-Three-Theme wechseln, kannst du sehen, dass die Schaltfläche **„Jetzt einkaufen“** falsch ausgerichtet ist:

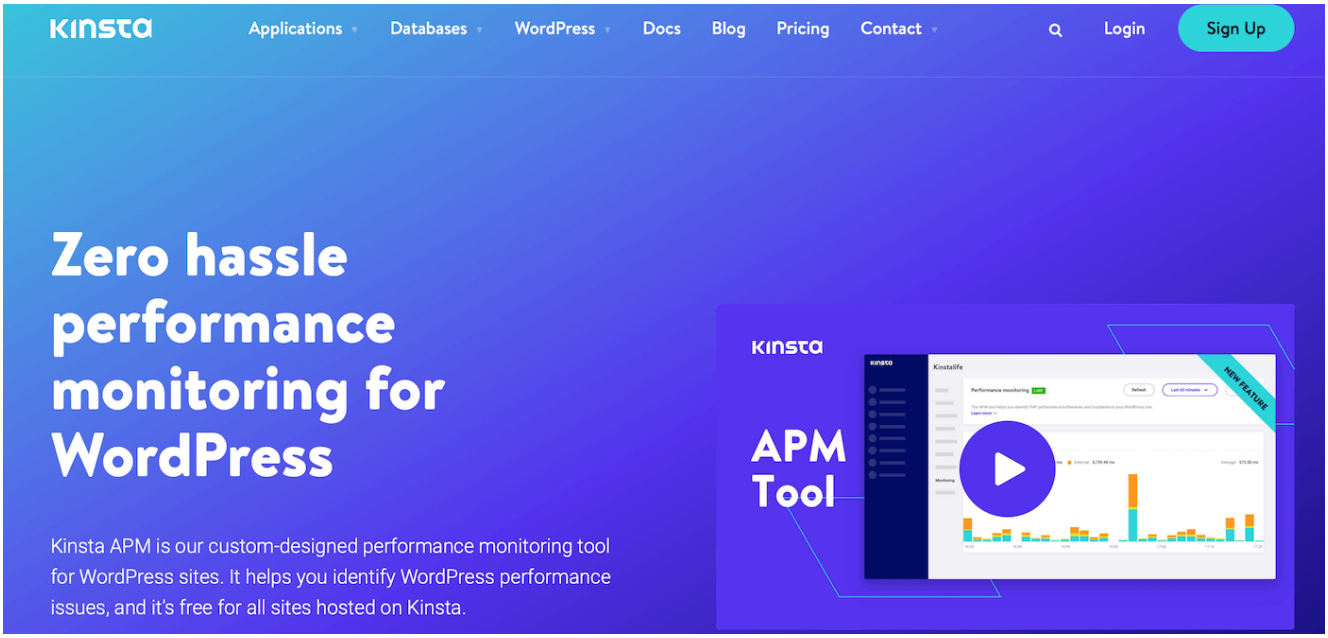


Erkenne visuelle Fehler mit visuellen WordPress-Tests
Wenn du eine lokale Umgebung für deinen Test einrichtest, kannst du visuelle Anomalien wie diese aufspüren.

Wie du die Geschwindigkeit deiner WordPress-Website testest (6 Überlegungen)

Eine weitere wichtige Methode, um deine WordPress-Website zu testen, ist die Überprüfung der aktuellen Geschwindigkeit deiner Website. In diesem Abschnitt gehen wir auf sechs Punkte ein, mit denen du die Leistung deiner Website testen kannst.

Vor diesem Hintergrund kann es hilfreich sein, mit Kinsta APM zu beginnen. Mit unserem Application Performance Monitoring Tool ist es ganz einfach, WordPress-Leistungsprobleme zu erkennen:



Kinsta APM-Tool

Du erhältst zum Beispiel Einblick in alle PHP-Prozesse, MySQL-Datenbankabfragen und externen HTTP-Aufrufe. Dadurch bist du in der Lage, lange API-Aufrufe, langsame Datenbankabfragen und nicht optimierten Plugin- und Theme-Code besser zu erkennen.

Das Beste daran ist, dass Kinsta APM in allen Kinsta-Tarifen kostenlos ist und du direkt von deinem MyKinsta-Dashboard aus auf das Tool zugreifen kannst. Insgesamt ist es eine einfach zu bedienende Lösung, die dir hilft, die Leistung und die

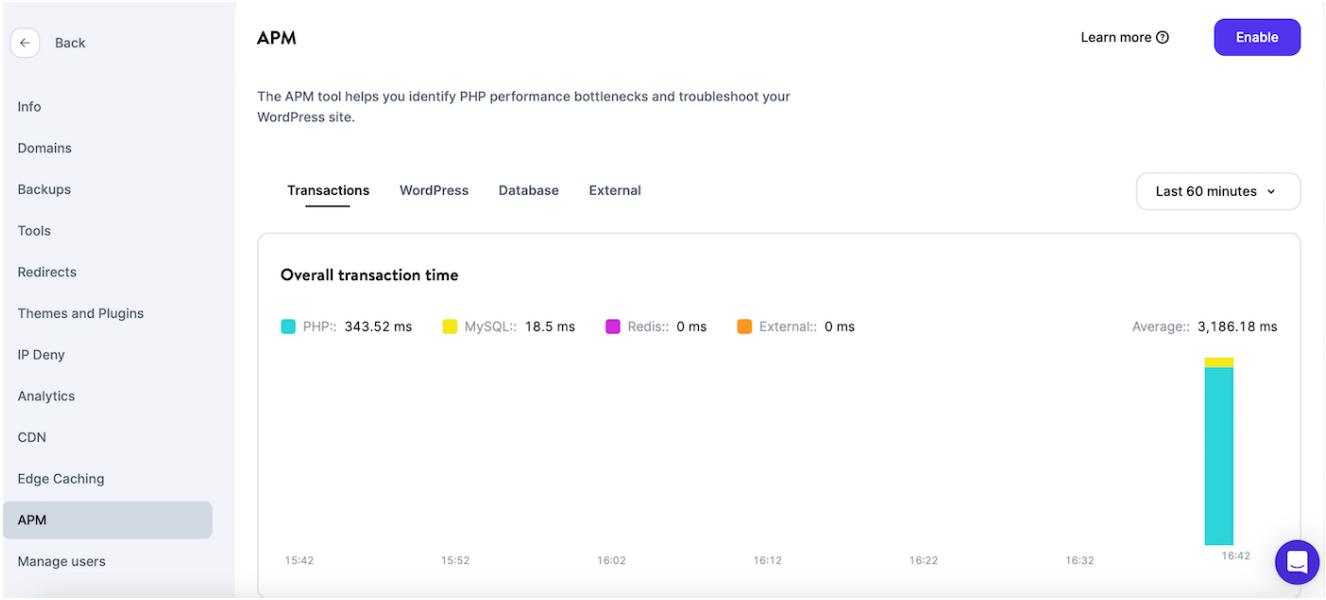
Ladezeiten deiner Website zu verbessern.

Langsame Abfragen oder Skripte

Um sicherzustellen, dass deine Website auf höchstem Niveau funktioniert, kannst du WordPress auf langsame Abfragen und Skripte testen. Langsame Abfragen und Skripte wirken sich auf die Gesamtgeschwindigkeit deiner Seite aus und machen deine Website weniger effizient.

Der einfachste Weg, langsame Abfragen und Skripte zu erkennen, ist die Aktivierung von Kinsta APM. Wenn du ein Kinsta-Kunde bist, kannst du das Tool kostenlos nutzen. Du musst es jedoch über dein MyKinsta-Dashboard aktivieren.

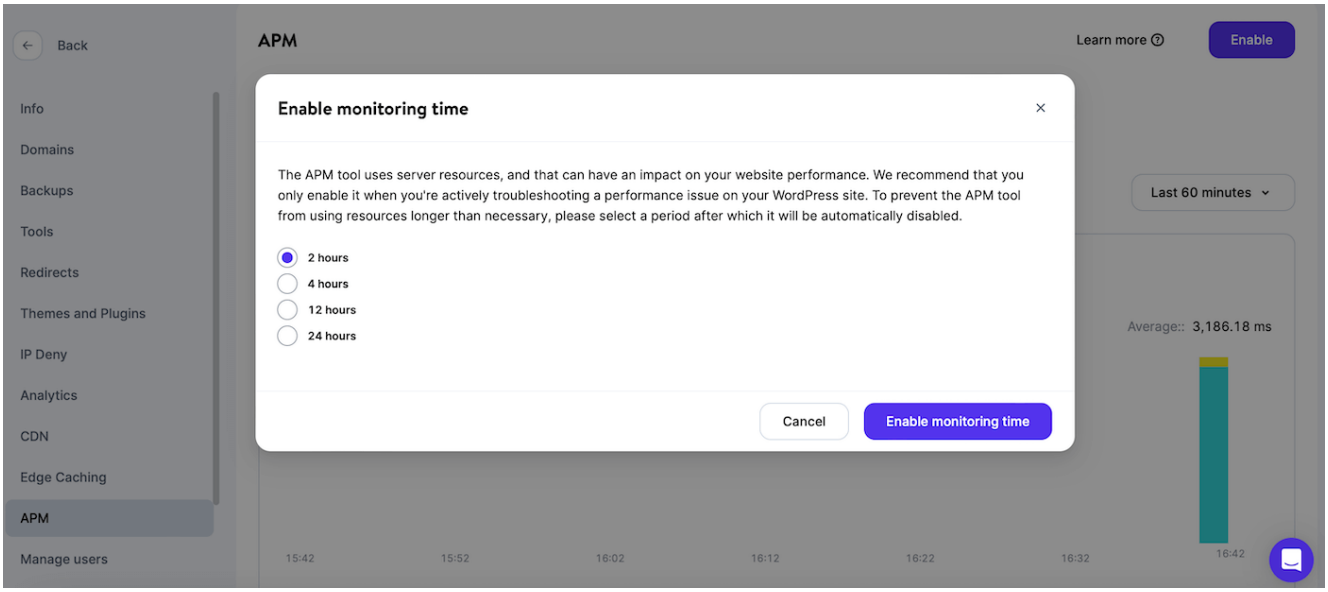
Logge dich dazu einfach in dein Konto ein und wähle die Website aus, auf der du das APM-Tool nutzen möchtest. Navigiere nun zum Reiter **APM** und klicke auf **Aktivieren**:



Aktiviere das Kinsta-APM-Tool über dein MyKinsta-Dashboard

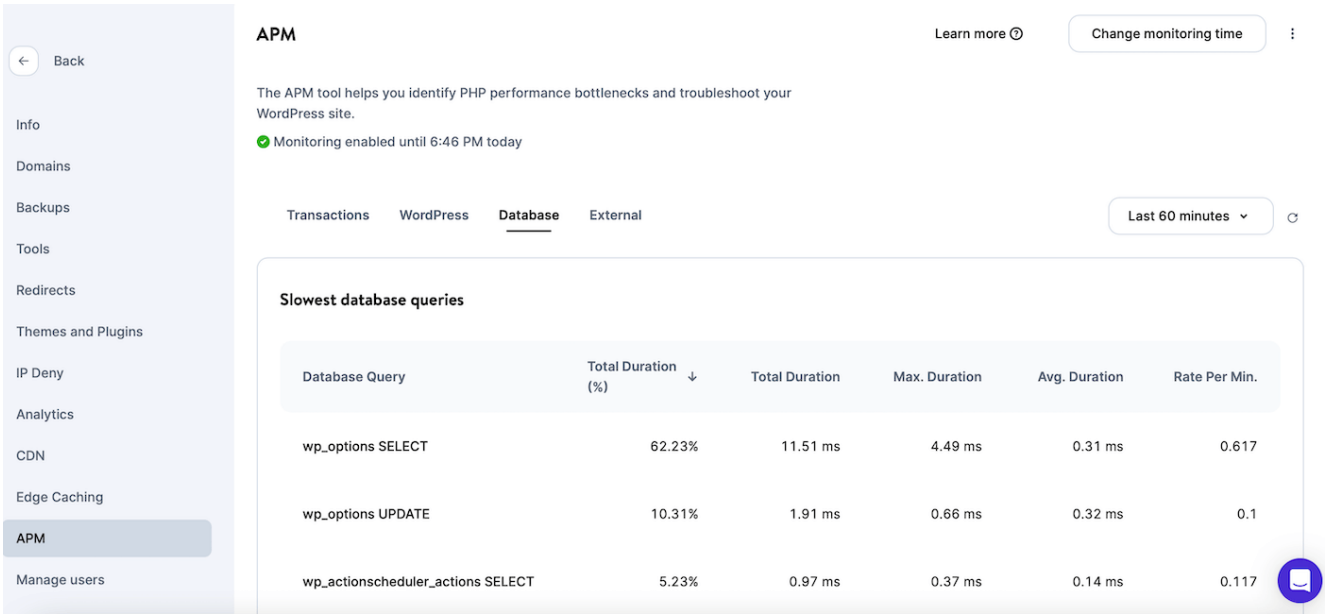
Dann musst du die Dauer auswählen, für die du das Tool nutzen willst. Da das APM-Tool Serverressourcen verbraucht, kann es sich auf die Leistung deiner Website auswirken. Daher ist es am besten, das Tool nur für den Zeitraum zu aktivieren, in dem du aktiv an der Behebung eines Leistungsproblems arbeitest.

Triff deine Wahl und klicke auf **Überwachungszeit einschalten**:



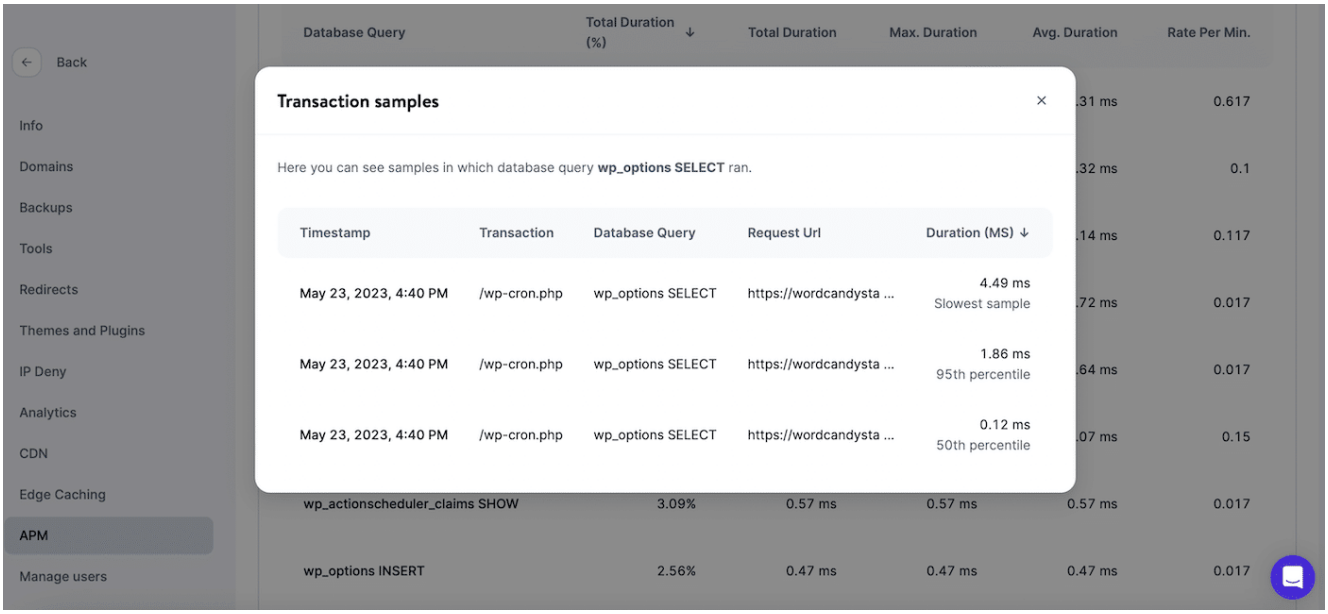
Aktiviere die Überwachungszeit für Kinsta APM

Es kann ein paar Minuten dauern, bis das Tool Daten über deine Website gesammelt hat. Wechsle danach auf die Registerkarte **Datenbank** und suche den Abschnitt **Langsamste Datenbankabfragen**:



Langsamste Datenbankabfragen anzeigen

Hier findest du die zehn langsamsten Datenbankabfragen auf deiner Website. Wenn du auf eine Abfrage klickst, kannst du dir auch die Transaktionsmuster ansehen:

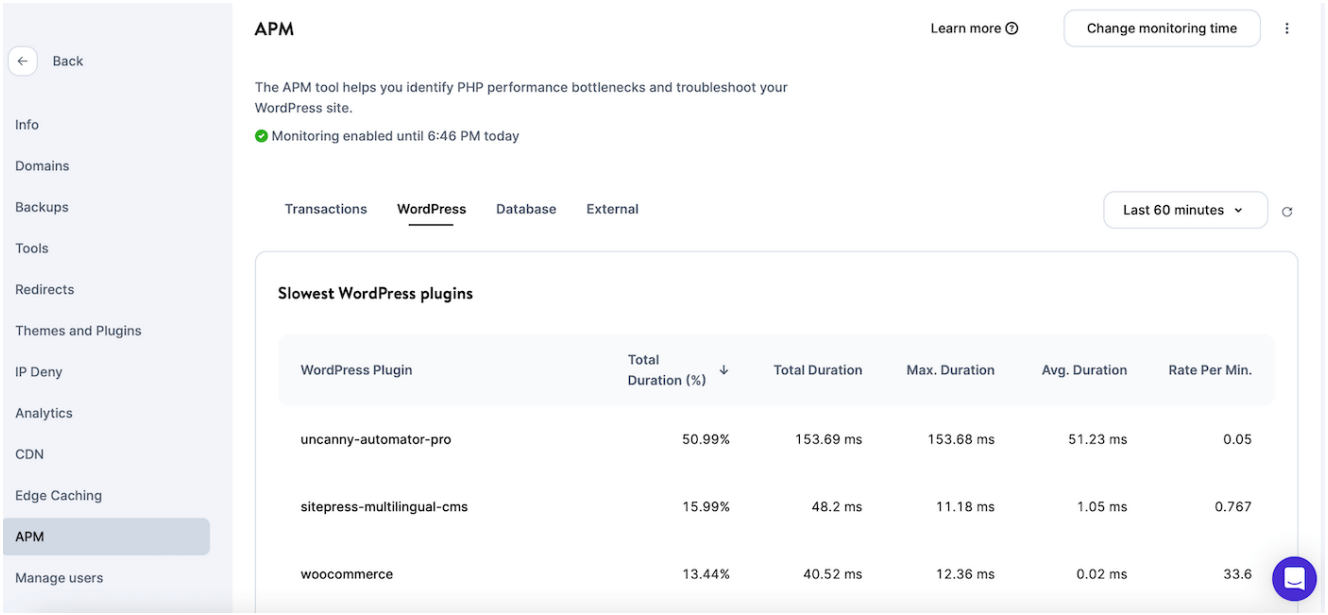


Transaktionsbeispiele anzeigen
Auf diese Weise kannst du mehr Informationen über die Probe, die Zeitleiste, die Spandetails und den Stacktrace herausfinden.

Langsame Plugins

Schlecht programmierte Plugins können nicht nur die Sicherheit deiner WordPress-Website beeinträchtigen, sondern auch die Leistung. Deshalb ist es wichtig, dieses Problem so schnell wie möglich zu erkennen.

Auch hier kannst du das Kinsta APM-Tool verwenden, um langsame Plugins zu identifizieren. Sobald du das Tool in deinem MyKinsta-Dashboard aktiviert hast, navigiere zum Reiter **APM** . Wechsle dann zu **WordPress**:



Testen auf langsame Plugins
Der erste Bereich, den du siehst, ist **Langsamste WordPress-Plugins**. Die langsamsten aufgezeichneten Plugins werden oben im Abschnitt aufgelistet.

Um mehr Informationen über die Leistungsprobleme zu erhalten, klicke auf eines der aufgelisteten Plugins. Dadurch werden die Transaktionsbeispiele geladen, die das Plugin ausgeführt hat. Du kannst dir zum Beispiel den Zeitstempel, die Zeitleiste der Transaktionsverfolgung, die Details der Spanne, die Zeitleiste der Verfolgung und vieles mehr ansehen.

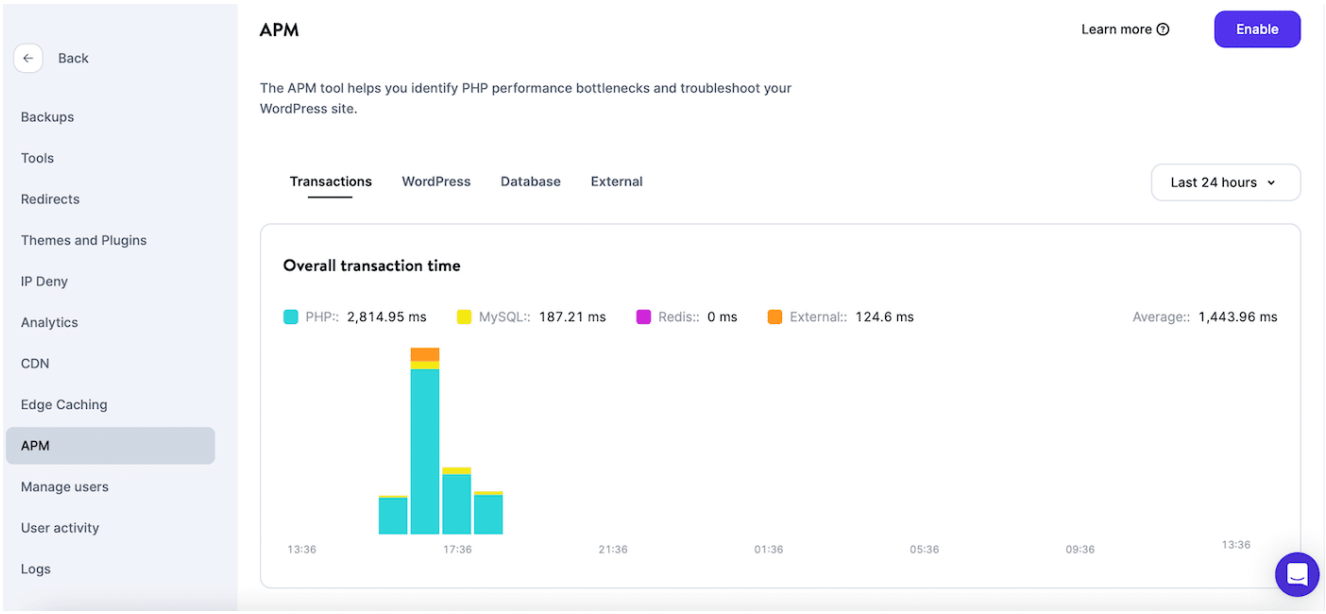
Langsame Seiten

Es ist auch wichtig, WordPress auf langsame Seiten zu testen, da dies zu einer schlechten UX führen kann. Außerdem ist die Seitengeschwindigkeit ein Rankingfaktor für Suchmaschinen wie Google.

Du kannst ein kostenloses Website-Geschwindigkeitstest-Tool wie Pingdom oder PageSpeed Insights verwenden, um eine schnelle Bewertung der Seitengeschwindigkeit zu erhalten. Mit dem APM-Tool von Kinsta kannst du jedoch einen genaueren Einblick in die Geschwindigkeit deiner Seite gewinnen.

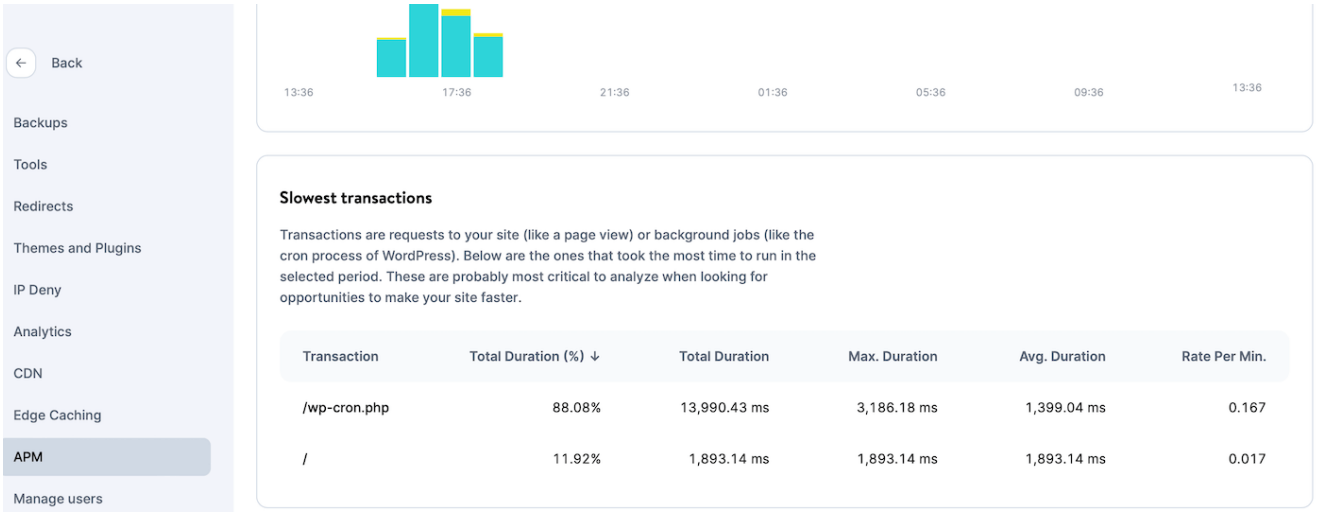
Sobald du Kinsta APM aktiviert hast, dauert es ein paar

Sekunden, bis die Leistungskennzahlen deiner Website geladen sind. Gehe danach auf den Reiter **Transaktionen** :



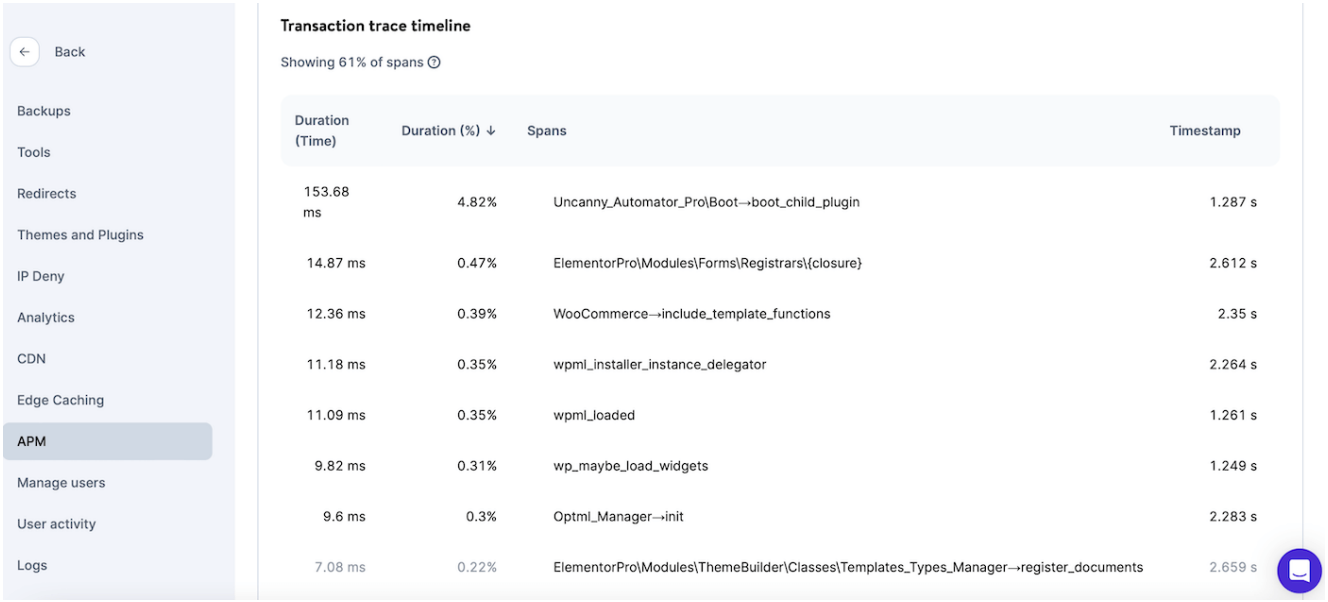
Teste langsame WordPress-Seiten mit Kinsta APM

Hier siehst du einige Daten über die gesamte Transaktionszeit deiner Website. Du kannst aber auch nach unten zu **Langsamste Transaktionen** scrollen, um die PHP-Prozesse zu sehen, die die meiste Transaktionszeit benötigen:



Langsamste Transaktionen anzeigen

Wenn du eine Transaktion auswählst, kannst du die URL herausfinden, die sie erzeugt. Klicke dann auf die URL, um die **Zeitleiste der Transaktionsverfolgung** anzuzeigen:



Zeitleiste für die langsamsten Transaktionen

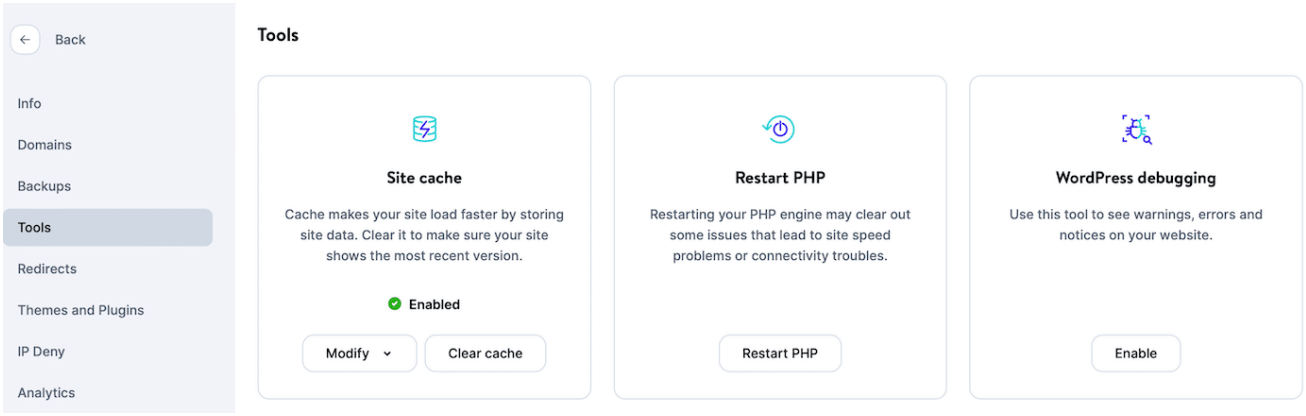
Auf diese Weise kannst du die Zeitspanne finden, die am meisten Zeit in Anspruch nimmt. Wenn diese Zeitspannen als kritisch für deine Leistung eingestuft werden, werden sie in der Regel orange oder rot hervorgehoben.

Caching

Caching ist eine einfache Methode, um deine Ladezeiten zu verbessern. Dabei werden Kopien deiner Website auf dem Server gespeichert. Wenn ein Nutzer deine Seite aufruft, kann dein Server die im Cache gespeicherte Version anzeigen, so dass die Daten viel schneller übertragen werden können.

Bei Kinsta erhältst du Zugang zum Server-Level-Caching, das automatisch auf allen Live-Websites aktiviert ist. Wenn du jedoch eine Staging-Umgebung verwendest, musst du den Cache manuell aktivieren.

Klicke in deinem MyKinsta-Dashboard auf **WordPress-Sites** und wähle deine Website aus. Dann navigierst du zu **Tools** und klickst unter **Site Cache** auf **Enable**:



Aktiviere den Cache auf Serverebene in MyKinsta

Der einfachste Weg, dein Caching zu testen, ist, deine Website mit einem Web-Speed-Test-Tool wie Pingdom zu testen. Es ist jedoch wichtig, dass du den Test mehr als einmal durchführst. Denn wenn du ihn nur einmal durchführst, kann es sein, dass der Inhalt noch nicht auf dem Server des Hosts oder im CDN zwischengespeichert ist.

Gib deine URL in das **URL-Feld** bei Pingdom ein und wähle einen Ort aus. Suche nun unter **Response Headers** nach **x-kinsta-cache**. Wenn hier **MISS** steht, wird deine Website nicht aus dem Cache geladen.

Um das zu beheben, musst du deine Website noch ein paar Mal durch den Pingdom-Test laufen lassen. Dies sollte dazu führen, dass die **x-kinsta-cache** und **x-cache** Header einen **HIT** registrieren. Jetzt überprüfst du die Ergebnisse und schaust auf den großen gelben Balken, der die Wartezeit oder Time to First Byte (TTFB) anzeigt.

Diese Zahl ist in der Regel hoch, wenn eine Seite nicht aus dem Cache gekommen ist. Auch hier empfiehlt es sich, den Test einmal mit deaktiviertem und dann noch einmal mit aktiviertem Cache durchzuführen, um den Unterschied deutlich zu sehen.

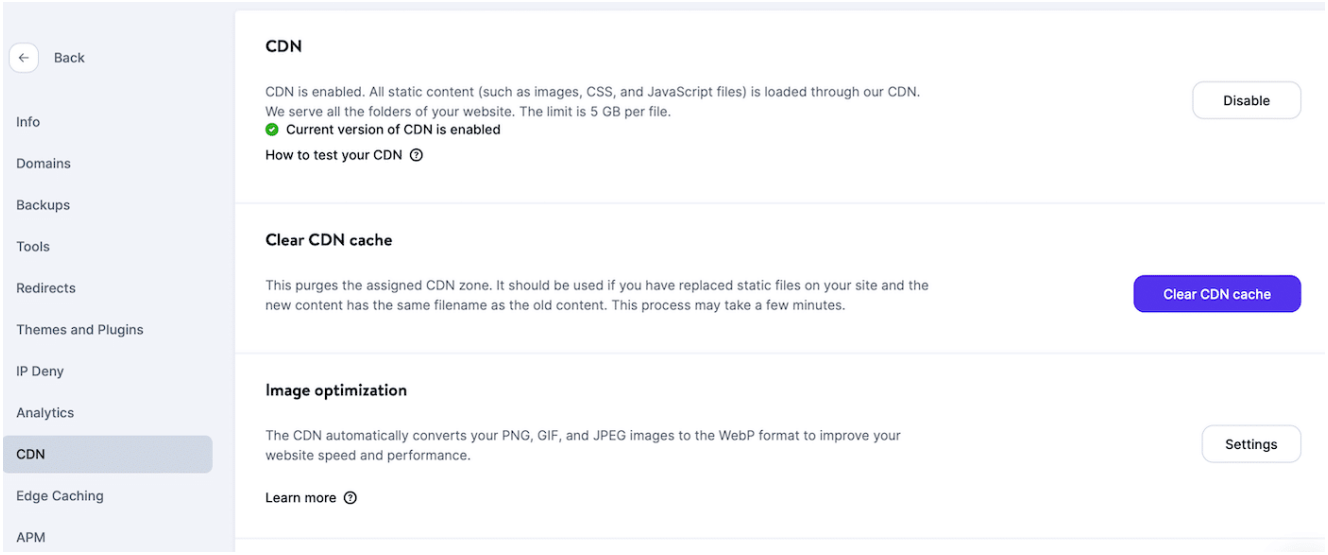
Content Delivery Network (CDN)

Ein Content Delivery Network (CDN) ermöglicht es dir, deine Ladezeiten zu verbessern, indem es deine Webseiten über einen Server ausliefert, der physisch näher bei deinen Besuchern

steht. Mit allen Kinsta-Tarifen erhältst du Zugang zu einem von Cloudflare betriebenen CDN.

Bei neuen Websites ist das CDN standardmäßig aktiviert. Du kannst aber überprüfen, ob dein CDN aktiviert ist, indem du dich in dein MyKinsta-Dashboard einloggst.

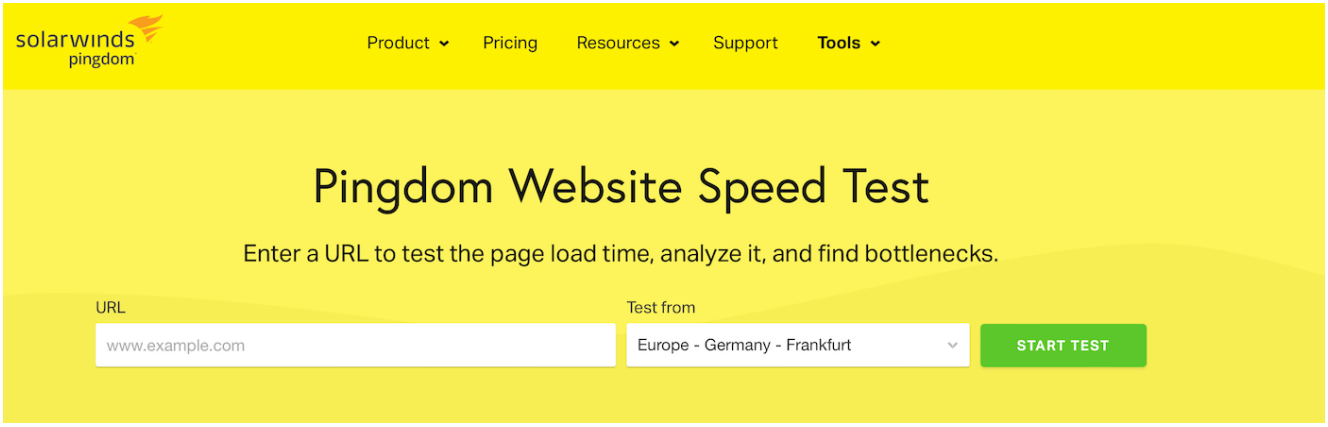
Gehe zu **WordPress Sites** und wähle den Namen deiner Website aus. Klicke auf den Reiter **CDN** und dann auf **Aktivieren**. Wenn du **Deaktivieren** siehst, weißt du, dass das CDN aktiv ist:



Aktiviere das Kinsta CDN

Um dein CDN zu testen, ist es am einfachsten, ein Tool zum Testen der Website-Geschwindigkeit zu verwenden. Aber zuerst ist es eine gute Idee, die HTTP-Header eines statischen Assets zu überprüfen, um sicherzustellen, dass es vom Kinsta CDN geladen wird.

Du kannst dies mit dem Inspect Tool deines Browsers oder mit unserem kostenlosen HTTP-Status- und Redirect-Checker überprüfen. Jetzt musst du ein Tool zum Testen der Website-Geschwindigkeit auswählen, z. B. Pingdom:



Pingdom

Du kannst den ersten Test durchführen, nachdem du das CDN abgeschaltet hast. Dann kannst du deine Website mit aktiviertem CDN erneut testen, um den Unterschied zu sehen. Außerdem solltest du dein CDN von verschiedenen Standorten aus testen.

Wenn dein Test abgeschlossen ist, solltest du dir die Anfragen ansehen, die vom Kinsta CDN *(xxxxkinstacdn.com)* geladen werden. Ausführliche Informationen zu diesem Thema findest du in unserem Beitrag über die Durchführung eines CDN-Tests.

Lasttests

Entgegen der landläufigen Meinung gibt es einen wichtigen Unterschied zwischen Geschwindigkeitstests und Lasttests. Bei Geschwindigkeitstests wird im Wesentlichen die Ladezeit einer Seite gemessen, einschließlich der MySQL- und PHP-Antwortzeiten.

Andererseits bieten Lasttests eine feinere Granularität als Geschwindigkeitstests. Er kann zum Beispiel dazu verwendet werden, die Ladezeiten in bestimmten Situationen zu messen, z. B. wenn deine Website von einem hohen Verkehrsaufkommen betroffen ist.

Das Einrichten eines Lasttests ist ziemlich komplex. Deshalb kann es eine gute Idee sein, einen Entwickler um Hilfe zu bitten. Wenn du einen Lasttest für deine Kinsta-Website durchführen möchtest, wende dich an einen Mitarbeiter unseres

Support-Teams.

Wie du die Sicherheit deiner WordPress-Website testest

Wenn du WordPress testest, musst du sicherstellen, dass die gesamte Software auf deiner Website sicher ist. Das betrifft nicht nur die WordPress-Kernsoftware, die die Plattform nutzt, sondern auch die Sicherheit von Themes und Plugins.

Das Testen von Themes und Plugins kann sogar noch wichtiger sein, da sie nicht immer aus einer seriösen Quelle stammen. Wenn du Themes und Plugins von Drittanbieter-Websites installierst, gibt es keine Möglichkeit zu überprüfen, ob die Software alle erforderlichen Sicherheitsprüfungen durchlaufen hat.

Das heißt, das Plugin oder Theme könnte schlecht programmiert sein oder sogar bösartige Skripte oder Fehler enthalten, die deine Website beschädigen können. Außerdem ist es wichtig, dass du die Software auf deiner Website immer auf dem neuesten Stand hältst, denn veraltete Software kann als Hintertür für böswillige Akteure genutzt werden, um sich Zugang zu verschaffen.

Kernsicherheit

Obwohl WordPress eine sichere Plattform ist, ist sie nicht immun gegen Cyberangriffe. Deshalb ist es wichtig, dass du die Sicherheit deiner Kernsoftware regelmäßig überprüfst.

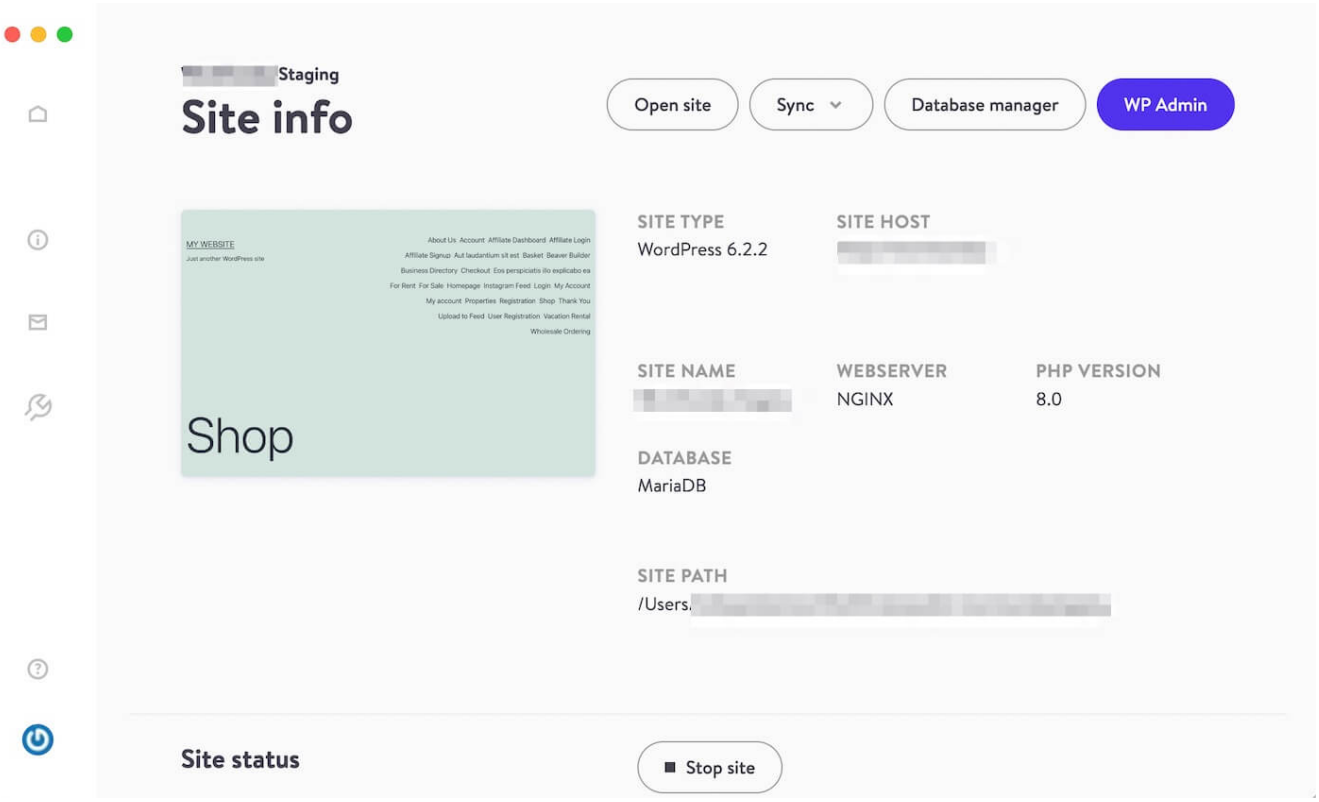
Eine der besten Möglichkeiten, deine Kernsoftware zu schützen, ist die Entscheidung für einen guten Webhoster. Bei Kinsta bekommst du zum Beispiel Zugang zu DDoS-Schutz, Firewalls und Malware-Scans. Außerdem haben wir ein spezielles Malware-Entfernungsteam vor Ort. Selbst wenn deine Website infiziert wird, können wir sie wieder in ihren ursprünglichen Zustand

versetzen.

Wenn ein neues WordPress-Update veröffentlicht wird, kannst du es auf jeden Fall zuerst auf seine Sicherheit testen, indem du es auf einer Staging-Seite oder in einer lokalen Umgebung ausführst.

Bei Kinsta ist das ganz einfach. Du musst nur zu **WordPress Sites** navigieren und deine Website aus der Liste auswählen. Stelle dann sicher, dass deine Website auf **Staging** eingestellt ist, wenn du das Update ausführst.

Wenn du sicher bist, dass die neue WordPress-Version sicher ist, kehrst du zu diesem Bildschirm zurück und klickst auf **Push environment** > **Push to LIVE** , um die Änderung zu übernehmen:



Änderungen von der Staging-Website live schalten

Triff deine Wahl (zwischen Dateien oder Datenbank) und bestätige deine Entscheidung mit einem Klick auf **Push to live**.

Theme-Sicherheit

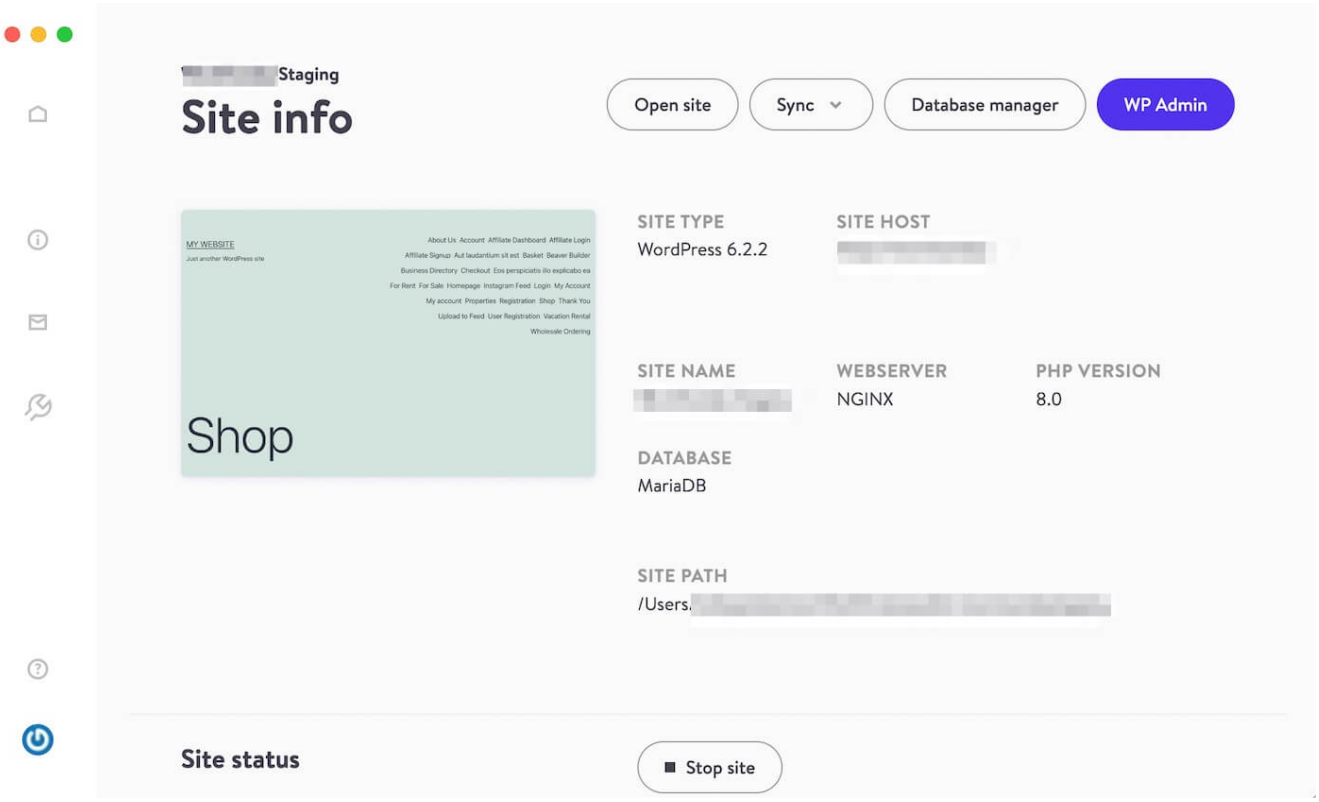
Wenn du ein neues Theme findest, das du installieren möchtest, aktivierst du es am besten in einer lokalen Entwicklungsumgebung oder auf deiner Staging-Site. Das Gleiche gilt, wenn ein bestehendes Theme auf deiner Seite ein Update veröffentlicht.

Die meisten Theme-Updates enthalten Patches für Sicherheitsprobleme. Es kann aber auch passieren, dass du ein schlechtes Update bekommst, das mit einer anderen Software auf deiner Website kollidiert.

Wenn es sich um ein Theme handelt, das du noch nie benutzt hast (und du die Entwickler nicht kennst), ist es viel sicherer, das Theme in einer lokalen Umgebung zu installieren. Das bedeutet, dass selbst wenn das Theme deine Website beschädigt, deine Live-Website davon nicht betroffen ist.

Wenn du Kinsta-Kunde bist, kannst du also eine Testseite einrichten. Wenn deine Website nicht bei Kinsta gehostet wird, kannst du auch kostenlos mit DevKinsta eine lokale Entwicklungsumgebung einrichten.

Wenn du DevKinsta auf deinem Computer geöffnet hast, rufe die Seite **Site Info** auf. Hier klickst du auf **WP Admin**:



Lokale Website von DevKinsta aus starten

Dann installierst und aktivierst du das Theme, wie du es normalerweise in WordPress tun würdest. Normalerweise ist es eine gute Idee, mindestens eine Woche zu warten, bevor du das Theme auf deiner Live-Website installierst (das gilt auch für ein neues Theme-Update).

Wenn du jedoch die Sicherheit eines bestehenden Themes auf deiner Website überprüfen möchtest, ist es am einfachsten, einen Sicherheitsscanner zu verwenden. WPScan ist eine großartige Option, die alle Sicherheitslücken in deinen WordPress-Themes aufspürt.

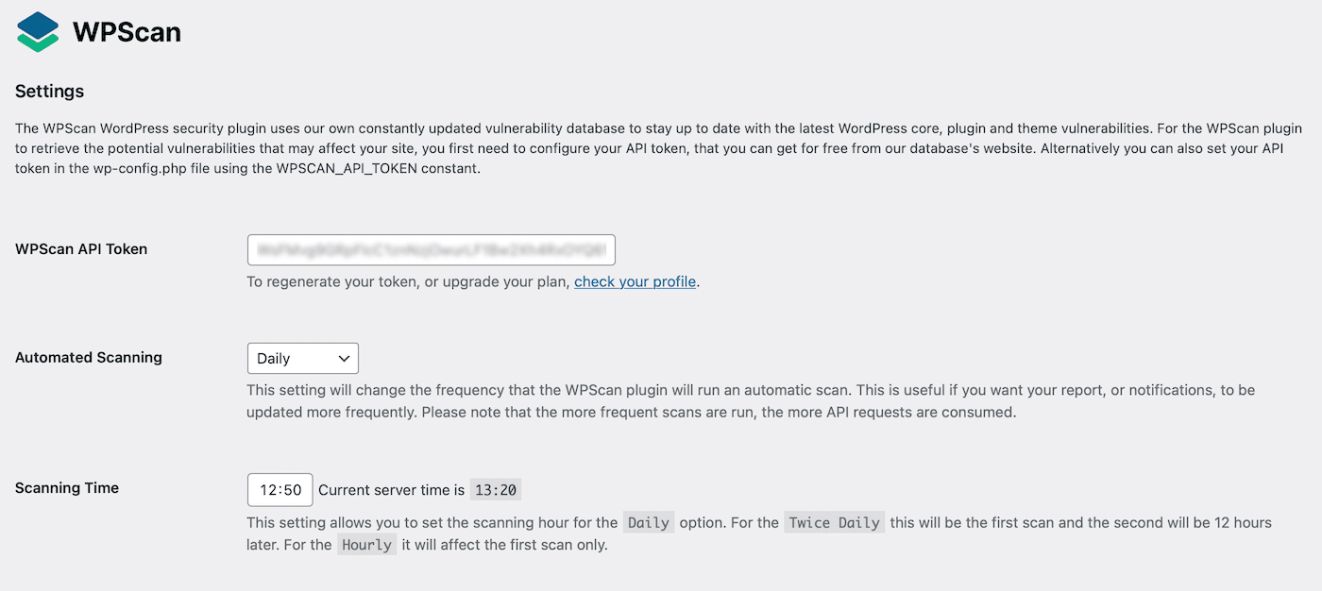
Plugin-Sicherheit

Auch Plugins können eine Gefahr für die Sicherheit deiner Website darstellen. Deshalb ist es eine gute Praxis, die Sicherheit deiner Plugins regelmäßig zu überprüfen.

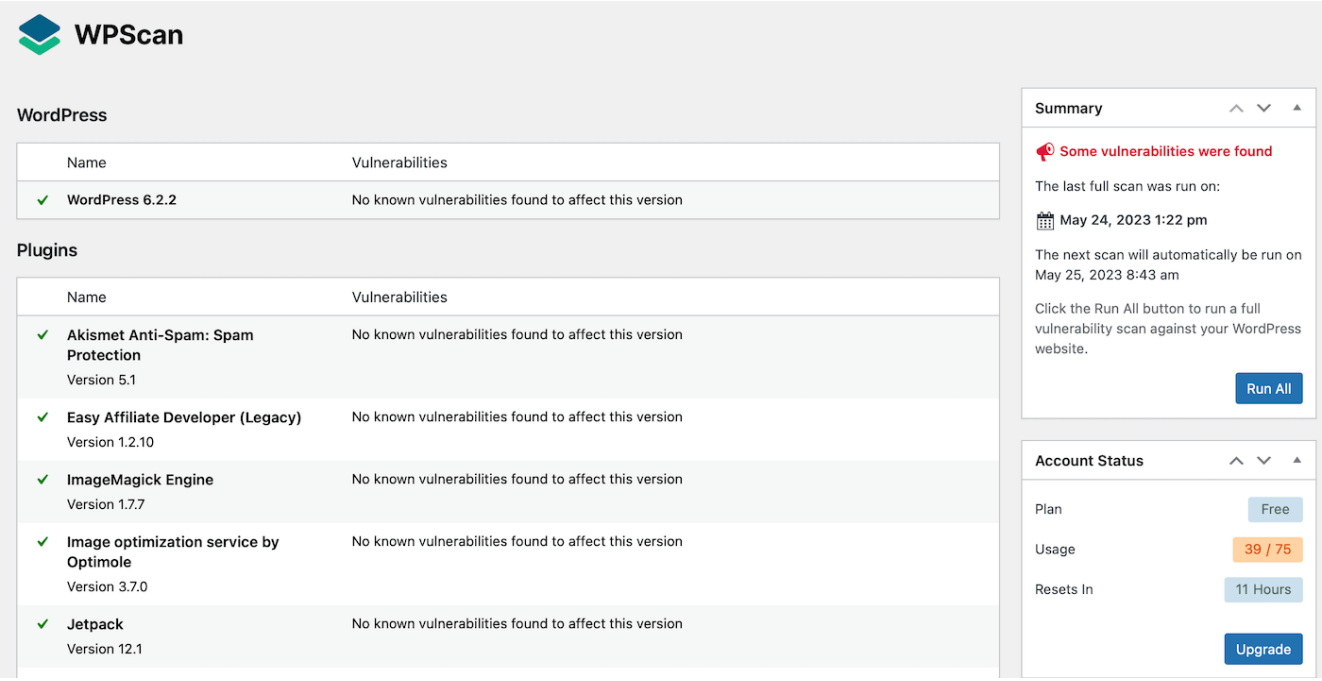
Wie bereits erwähnt, kannst du ein neues Plugin (oder ein Plugin-Update) in einer lokalen Umgebung oder auf einer Staging-Seite installieren. Auf diese Weise bleibt deine Live-Site intakt, falls etwas schief geht.

Wie bei Themes kann es aber auch nützlich sein, einen Schwachstellen-Scanner wie WPScan zu installieren. Die Nutzung dieses Tools ist völlig kostenlos. Alles, was du tun musst, ist, dich für ein Konto zu registrieren. Dann kannst du das API-Token zu deiner WordPress-Seite hinzufügen.

Sobald der Scanner mit deiner Website verknüpft ist, navigierst du zu **WPScan** > **Einstellungen** , wo du automatische tägliche oder stündliche Scans einrichten kannst:



Teste die Plugin-Sicherheit mit WPScan
Oder klicke auf die Registerkarte **Bericht** , um einen manuellen Test durchzuführen. Sobald der Test abgeschlossen ist, scrolle nach unten zum Abschnitt **Plugins** :



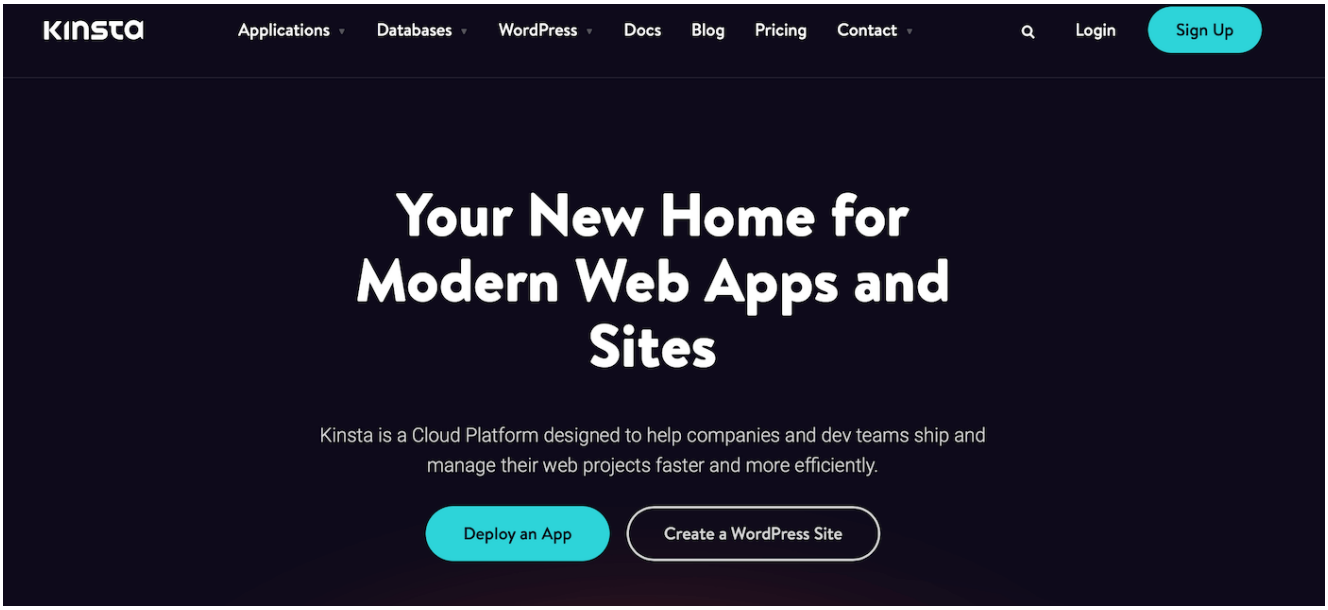
WPScan-Berichte

Hier kannst du eine vollständige Liste aller Plugins auf deiner Website sehen. Wenn deine Plugins sicher sind, siehst du ein Häkchen neben jedem Plugin-Namen. Andernfalls findest du einige Informationen in der Spalte **„Schwachstellen„**.

Geschwindigkeit und Sicherheit sind am besten, wenn du das richtige Hosting wählst

Natürlich kannst du deine Webseiten optimieren und alle notwendigen Sicherheitsmaßnahmen ergreifen, um eine erfolgreiche Website zu betreiben. Der beste Weg, um sicherzustellen, dass deine Website sicher und schnell ist, ist jedoch, einen guten Webhoster zu wählen.

Bei Kinsta legen wir großen Wert auf Geschwindigkeit und Sicherheit:



Kinsta
Alle unsere Angebote werden auf den besten CPUs mit globaler Verfügbarkeit gehostet. Außerdem erhältst du Zugang zu Kinstas Cloudflare-gestütztem CDN mit Servern an über 260+ Standorten.

Für alle, die sich um die Sicherheit im Internet sorgen, bietet Kinsta eine Vielzahl von Funktionen, um deine Website zu sichern. Du kannst tägliche Backups, Malware-Scans, DDoS-Schutz und Firewalls erwarten. Außerdem bieten wir einen sicheren SSH-Zugang und du kannst mit nur einem Klick ein kostenloses SSL-Zertifikat installieren.

Zusammenfassung

Ohne deine WordPress-Website zu testen, kannst du nicht richtig verstehen, wie die Nutzer deine Website erleben. Wer zum Beispiel bestimmte Browser benutzt, hat vielleicht Probleme mit deinem Menü. Mobile Besucher können mit langen Wartezeiten konfrontiert sein. Deshalb ist es wichtig, deine WordPress-Website zu testen.

Am besten testest du deine Website, indem du eine Staging-Site einrichtest oder mit DevKinsta eine lokale Umgebung erstellst. So erhältst du Einblicke in die Funktionalität, Leistung und Sicherheit deiner Website (ohne dein Live-Web-Erlebnis zu stören).

Ein bisschen zusätzliche Sicherheit kann aber nie schaden. Eine der einfachsten Möglichkeiten, um sicherzustellen, dass deine Website jederzeit reibungslos läuft, ist die Entscheidung für einen hochwertigen Webhoster wie Kinsta. Schau dir unsere Tarife an, um loszulegen!

Sparen Sie Zeit und Kosten und maximieren Sie die Leistung Ihrer Seite mit Integrationen auf Unternehmensebene im Wert von über 275$, die in jedem Managed WordPress Plan enthalten sind. Dazu gehören ein leistungsstarkes CDN, DDoS-Schutz, Malware- und Hacking-Abwehr, Edge-Caching und die schnellsten CPU-Maschinen von Google. Legen Sie los – ohne langfristige Verträge, mit Migrationsunterstützung und einer 30-Tage-Geld-zurück-Garantie.

Informieren Sie sich über unsere Pakete oder sprich mit dem Vertrieb, um den für Sie passenden Plan zu finden.