

Magento Vs. Shopware

Die ultimative Vergleichstabelle

	Magento 2	Shopware 6
Technischer Hintergrund	<ul style="list-style-type: none">→ PHP→ MySQL→ Framework: Magento-Framework mit Symfony- und Laminas-Komponenten→ Frontend: Layout konfiguriert über XML, PHP/HTML für Templates, LESS für CSS, RequireJS und Knockout.js für dynamische Inhalte→ Erweiterbar durch Dependency Injection, Events, Plugins, Austausch von beliebigen Business-Logik- und Design-Dateien→ Keine Quellcodebestandteile oder verschlüsselte Module→ Elasticsearch für die Suche (verpflichtend)→ Varnish und / oder Redis für Caches (optional)→ Redis für Sessions (optional)→ RabbitMQ für asynchrone Operationen (optional)	<ul style="list-style-type: none">→ PHP→ MySQL→ Framework: Symfony→ Frontend: Twig, Bootstrap (Shop-Frontend), Vue.js, twig.js (Admin-Panel)→ Erweiterbar durch Dependency Injections, Events etc.→ API-First zur Integration von externen Systemen und Frontend-Applikationen→ Keine Quellcodebestandteile oder verschlüsselte Module→ Elasticsearch für die Suche (optional)→ Varnish und / oder Redis für Caches (optional)→ Redis für Sessions (optional)→ Symfony Messenger und Enqueue für asynchrone Operationen (optional)

Performance	<ul style="list-style-type: none"> → Varnish Cache (optional) mit ESI-Tags für nicht-cachebare Inhalte → Alternativ: HTTP-Cache über Redis → Komprimierung von Bildern, CSS, JavaScript → Mit Fachwissen für hohen Traffic skalierbar 	<ul style="list-style-type: none"> → Varnish Cache (optional) mit ESI-Tags für nicht-cachebare Inhalte → Alternativ: HTTP-Cache Symfony → Komprimierung von Bildern, CSS, JavaScript → Mit Fachwissen für hohen Traffic skalierbar
Support	<ul style="list-style-type: none"> → Support und Community vor allem englischsprachig → Sehr umfangreiche Dokumentation und Unterlagen für Anwender:innen und Entwickler:innen → Sehr große Community, international → Viele Events und Meetups, weltweit → Offizieller Hersteller-Support für Händler:innen mit lizenzkostenpflichtigen Editionen, Support-Level unterschiedlich 	<ul style="list-style-type: none"> → Support und Community vor allem deutschsprachig → Umfangreiche Dokumentation und Unterlagen für Anwender:innen und Entwickler:innen → Große Community, vor allem deutschsprachig, teils europäisch → Einige Events und Meetups, vor allem in Deutschland / DACH-Raum → Offizieller Hersteller-Support für Händler:innen mit lizenzkostenpflichtigen Editionen, Support-Level unterschiedlich
Internationalität	<ul style="list-style-type: none"> → Kommt aus USA → Für Internationalisierung vorbereitet → Weltweite Internationalisierung gut möglich: viele Übersetzungen, internationale Zahlungs- und Versandarten etc. vorhanden, Magento-Shops in (fast) allen Ländern der Welt implementiert → Gute Agenturen haben ein Setup für den DACH-Raum, sodass wenig bis keine Startnachteile gegenüber Shopware bestehen 	<ul style="list-style-type: none"> → Kommt aus Deutschland → Für Internationalisierung vorbereitet → Besonders gute Vorbereitung für Standard-Shops im DACH-Raum "out-of-the-box" → Internationalisierung für europäischen Raum liegt bei Shopware aktuell im Fokus → Weltweite Internationalisierung für US-Märkte, Asien, Afrika, etc. erfordert höhere Aufwände und Eigenleistungen

Frontend	<ul style="list-style-type: none">→ Magentos Standard-Tool-Set und Referenz-Design “Luma” ist technisch nicht auf dem neuesten Stand, in der Anpassung aufwändig, eine Top-Performance ist nur mit viel Aufwand zu erreichen→ Mit “Hyvä” gibt es aber ein alternatives Tool-Set und Referenz-Design, das technisch sehr modern ist, eine Top-Performance bringt und auch schneller implementierbar ist→ Daneben gibt es mit “Magento PWA Studio” eine offizielle Headless-Implementierung bzw. PWA sowie mehrere alternative Headless-Implementierungen (darunter Vue Storefront)→ Achtung bei den Buzzwords Headless / PWA: man muss pro Projekt entscheiden, ob dieser Ansatz Sinn ergibt	<ul style="list-style-type: none">→ Shopwares Standard-Tool-Set mit den Twig und Bootstrap ist etabliert, auf gute Weise “nicht aufregend” und performt gut→ Die Geschwindigkeit des Webshops und Umsetzungszeit ist mit dem Standard-Tool-Set in der Regel besser als mit Magentos Standard “Luma”, aber mit Magentos Alternative “Hyvä” dreht sich das Bild. Man muss sich mit Shopware anstrengen, um ähnliche Werte zu erreichen→ Mit “Shopware PWA” kommt die offizielle Headless-Implementierung bzw. PWA von Shopwares Partner Vue Storefront→ Achtung bei den Buzzwords Headless / PWA: man muss pro Projekt entscheiden, ob dieser Ansatz Sinn ergibt
-----------------	--	---

Backend	<p>Umfangreiche Shop-Administration, unter anderem:</p> <ul style="list-style-type: none"> → Verwaltung von Produkten und Kategorien → Verwaltung von Kunden und Kundengruppen → Verwaltung von Bestellungen, Rechnungen, Sendungen, Gutschriften (Reklamationen in Adobe Commerce) → Produkt-Bewertungen → Rabatte & Aktionen → Newsletter-Empfänger-Verwaltung → CMS-Inhalte über Page Builder / Drag & Drop → Medien-Dateien-Verwaltung → Daten-Analyse und -Reportings → Multiple Lagerstände → Daten-Importe und -Exporte 	<p>Umfangreiche Shop-Administration, unter anderem:</p> <ul style="list-style-type: none"> → Verwaltung von Produkten und Kategorien → Verwaltung von Kunden und Kundengruppen → Verwaltung von Bestellungen → Produkt-Bewertungen → Rabatte & Aktionen → Newsletter-Empfänger-Verwaltung → CMS-Inhalte über Page Builder / Drag & Drop → Medien-Dateien-Verwaltung → Verwaltung von Verkaufskanälen wie Social Shopping, Vergleichsportale
Features	<ul style="list-style-type: none"> → Standard-E-Commerce-Funktionen schon zu Zeiten von Magento 1 etabliert und nach Magento 2 übernommen → Adobe sieht Kernfunktionen eher als "Feature-complete" → Neue Funktionen werden zunehmend vom Magento-Core entkoppelt und als losgelöste Extensions oder SaaS-Lösungen angeboten → Neue Features zielen vor allem auf zukunftsweisende und Enterprise-fähige Lösungen ab, zum Beispiel KI-gestützte Suchergebnisse und Produkt-Empfehlungen → Ausnahme: Page Builder, mit dem Magento Open Source im CMS-Bereich mit Shopwares Erlebniswelten gleich zog 	<ul style="list-style-type: none"> → Teile der Shopware 5 Standard-E-Commerce-Funktionen wurden erst nach und nach in Shopware 6 integriert → "Erlebniswelten" zur Erstellung von Landingpages → Empowerment von Shopbetreiber:innen durch Features wie "Rule Builder" und "Flow Builder" → Regelmäßige Veröffentlichung neuer, größerer Features

Erweiterungen	<ul style="list-style-type: none"> → Out-of-the-box lauffähiger Shop → >10.000 Erweiterungen im Marketplace → Qualitätskontrolle im Marketplace → Auch Extensions außerhalb des Marketplace beziehbar 	<ul style="list-style-type: none"> → Out-of-the-box lauffähiger Shop → Zahlreiche Extensions im Shopware Store → Qualitätskontrolle im Shopware Store → Auch Extensions außerhalb des Marketplace beziehbar
Schnittstellen	<ul style="list-style-type: none"> → Fast kompletter Funktionsumfang über Schnittstellen ansprechbar → Fokus: Erweiterbarkeit und Anpassbarkeit des Codes, Sicherheit durch Rechte-Management, Rate Limiting u.ä., Skalierbarkeit durch Einsatz von Caches → 3 Möglichkeiten: "Web APIs" per REST oder SOAP, Headless-Implementierung über GraphQL → Web API: <ul style="list-style-type: none"> ○ REST nach OpenAPI-Spezifikation oder SOAP ○ Für Anbindungen von ERP-, PIM- und ähnlichen Systemen, Automatisierungen, mobile Applikationen, beliebige Software-Programme oder Web-Apps ○ Authentifizierung: OAuth, Token oder Login-Daten von Administrator:innen bzw. Kund:innen → GraphQL: <ul style="list-style-type: none"> ○ Für Implementierung von Headless und Composable Commerce Frontends ○ Wird von Adobe gepusht 	<ul style="list-style-type: none"> → "API first" Ansatz: prinzipiell alles in Shopware 6 über Schnittstelle ansprechbar → Unterscheidung in "Store API" und "Admin API" → Admin API: <ul style="list-style-type: none"> ○ REST nach OpenAPI-Spezifikation ○ Für Anbindung von ERP-, PIM- und ähnlichen Systemen, Automatisierungen ○ Authentifizierung: OAuth ○ Fokus: Leistung bei hoher Last, Datenkonsistenz, Fehlerbehandlung und Transaktionssicherheit → Store API: <ul style="list-style-type: none"> ○ Für Implementierung von Headless Frontends, nativen Apps oder Web-Apps ○ REST nach OpenAPI-Spezifikation ○ Fokus: Geschwindigkeit, Verfügbarkeit

Usability	<ul style="list-style-type: none"> → Eher komplex durch umfangreiche Funktionen → Erfahrung / Einarbeitung nötig 	<ul style="list-style-type: none"> → Übersichtlichkeit → Klare Navigation → Intuitive Bedienbarkeit → Trotzdem Einarbeitung in Vokabular und "Denkweise" nötig
Marketing	<ul style="list-style-type: none"> → Landingpages einfach erstellbar mit Page Builder → Technische SEO → Meta-Tags → Content-Verwaltung durch Medienmanager → Rabattaktionen konfigurieren durch Preisregeln → Erfassung von Kunden- & Newsletter-Daten → Zahlreiche Marketing-Erweiterungen 	<ul style="list-style-type: none"> → Landingpages einfach erstellbar mit Erlebnisswelten → Zentrale Content-Verwaltung → Zentrale Steuerung der Verkaufskanäle → Strukturierte Daten, Rich Snippets, Canonical Tags, Breadcrumbs → Rabatte & Aktionen, Rule Builder → Erfassung von Kunden- & Newsletter-Daten → Zahlreiche Marketing-Erweiterungen
Kosten	<ul style="list-style-type: none"> → Lizenzkosten: es gibt kostenlose und kostenpflichtige Lizenz-Varianten → Kostenpflichtige Lizenzen: abhängig von mehreren Faktoren, Verhandlungssache - keine fixe Preistabelle → Jährliche Kosten für Lizenz, Updates, Support zusammengefasst → Implementierungs-, Wartungs- und Hosting-Kosten durch die Dienstleister in der Regel etwas höher als bei Shopware 6 	<ul style="list-style-type: none"> → Lizenzkosten: es gibt kostenlose und kostenpflichtige Lizenz-Varianten → Kostenpflichtige Lizenzen: abhängig von mehreren Faktoren, Verhandlungssache - keine fixe Preistabelle → Monatliche Kosten für Lizenz, Updates, Support zusammengefasst → Implementierungs-, Wartungs- und Hosting-Kosten durch die Dienstleister in der Regel etwas niedriger als bei Magento 2