

# Checklist For Choosing The Best Ecommerce Website Development Company

A practical buyer scorecard to compare ecommerce development partners by strategy, UX, performance, SEO, security, integrations, delivery maturity, ownership, and post-launch growth support.

## How to use it

Score each vendor from 0 to 5 for every quality. Ask for proof artifacts, not just promises. Use the red flags section before signing an SOW.

Score	Meaning	What It Usually Looks Like
0	No evidence	Vendor makes claims but provides no sample process, artifact, or acceptance criteria.
1-2	Weak proof	Basic explanation, but limited specificity, unclear ownership, or no ecommerce-specific examples.
3	Adequate proof	Clear process, usable examples, and enough detail for moderate-scope ecommerce work.
4-5	Strong proof	Repeatable artifacts, specific acceptance criteria, strong technical rationale, and post-launch ownership.

**Total score guide:** 40-50 = strong candidate; 30-39 = workable but validate weak areas; below 30 = high risk unless scope is very simple and timeboxed.

## Vendor Comparison Scorecard

Use this summary after completing the 10 quality checks. Score each vendor from 0 to 5, then compare totals and risk notes.

Quality	Vendor A	Vendor B	Vendor C	Notes
1. Strategy-First Discovery				
2. Platform And Architecture Expertise				
3. Conversion-Focused UX And Merchandising				
4. Performance Engineering				
5. SEO And Structured Data Readiness				
6. Secure-By-Design Development				
7. Integration Depth				
8. Transparent Delivery Process				
9. Ownership Mindset				
10. Post-Launch Growth Support				
<b>Total Score</b>				

### Quick Interpretation

<b>40-50</b>	Strong candidate for complex builds and long-term ecommerce iteration.
<b>30-39</b>	Likely workable for moderate scope, but validate weak areas through paid discovery.
<b>Below 30</b>	High risk unless the project is simple, limited, and timeboxed.

# 10-Point Ecommerce Development Partner Checklist

For each quality, check whether the vendor can provide proof, answer the key questions, and commit to measurable acceptance criteria.

<b>1. Strategy-First Discovery</b>		<b>Score: 0 1 2 3 4 5</b>
<b>Outcome:</b> The partner links business goals, user journeys, risks, analytics needs, and backlog priorities before design or code starts.		
<b>Proof To Request</b>	<b>Questions To Ask</b>	
<input type="checkbox"/> Discovery workshop agenda and output deck <input type="checkbox"/> Success metrics and KPI framework <input type="checkbox"/> User journeys, event taxonomy, and prioritized backlog	1. What are your top discovery deliverables in the first two weeks? 2. How do you prioritize backlog items: revenue impact, effort, risk, or dependencies? 3. What metrics will confirm the launch improved performance or conversion?	
<b>Notes:</b> _____	<b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject	
<b>2. Platform And Architecture Expertise</b>		<b>Score: 0 1 2 3 4 5</b>
<b>Outcome:</b> The partner can justify platform-first, custom, headless, or composable choices based on cost, speed, scalability, and integration needs.		
<b>Proof To Request</b>	<b>Questions To Ask</b>	
<input type="checkbox"/> Architecture diagrams and integration map <input type="checkbox"/> Build-versus-buy reasoning <input type="checkbox"/> Clear explanation of where business logic and data ownership will live	1. What breaks first when traffic doubles: frontend, backend, or integrations? 2. Where will business logic live: storefront, backend, or a service layer? 3. How do you avoid vendor lock-in while keeping delivery realistic?	
<b>Notes:</b> _____	<b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject	
<b>3. Conversion-Focused UX And Merchandising</b>		<b>Score: 0 1 2 3 4 5</b>
<b>Outcome:</b> The partner understands category browsing, filters, search, PDP clarity, trust signals, cart persistence, checkout friction, and merchandising rules.		
<b>Proof To Request</b>	<b>Questions To Ask</b>	
<input type="checkbox"/> UX research method and wireframe examples <input type="checkbox"/> Design system or component samples <input type="checkbox"/> CRO workflow and merchandising-heavy template examples	1. What are your top UX risks for our catalog and customer journey? 2. How do you design faceted navigation without hurting SEO? 3. Which checkout optimizations have you seen reliably lift conversion?	
<b>Notes:</b> _____	<b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject	
<b>Quick Notes: Strategy, Architecture, And Conversion Fit</b>		
<b>Strongest proof:</b> _____		
<b>Main risk / gap:</b> _____		
<b>Follow-up question:</b> _____		

# 10-Point Ecommerce Development Partner Checklist

4. Performance Engineering		Score: 0 1 2 3 4 5
<b>Outcome:</b> Performance is treated as an acceptance criterion across PDP, PLP, cart, checkout, scripts, images, caching, and peak-traffic behavior.		
<b>Proof To Request</b>	<b>Questions To Ask</b>	
<input type="checkbox"/> Core Web Vitals targets by template type <input type="checkbox"/> Lighthouse reports and performance budget <input type="checkbox"/> CDN, caching, image optimization, and load-testing plan	1. What are your CWV acceptance criteria for PDP, PLP, and checkout? 2. How do you prevent third-party scripts from degrading performance? 3. What is your plan for flash sales: queueing, rate limiting, or autoscaling?	
<b>Notes:</b> _____	<b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject	
5. SEO And Structured Data Readiness		Score: 0 1 2 3 4 5
<b>Outcome:</b> Technical SEO is built into development through redirect mapping, schema planning, canonical rules, sitemap handling, and indexation safeguards.		
<b>Proof To Request</b>	<b>Questions To Ask</b>	
<input type="checkbox"/> Technical SEO checklist and schema plan <input type="checkbox"/> Redirect map and migration QA process <input type="checkbox"/> Canonical, faceted URL, sitemap, and staging indexation rules	1. How do you validate redirects and prevent soft 404s at scale? 2. What is your process to protect rankings during a replatform? 3. How do you handle out-of-stock pages and discontinued products?	
<b>Notes:</b> _____	<b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject	
6. Secure-By-Design Development		Score: 0 1 2 3 4 5
<b>Outcome:</b> Security is considered a revenue feature, covering secure SDLC, dependency scanning, access control, secrets management, and PCI scope awareness.		
<b>Proof To Request</b>	<b>Questions To Ask</b>	
<input type="checkbox"/> OWASP-aligned practices and secure code review <input type="checkbox"/> Dependency scanning, secrets management, and environment controls <input type="checkbox"/> Pen testing approach and remediation workflow	1. How do you manage secrets, keys, and environment configs? 2. What is your approach to role-based access in admin tools? 3. How do you handle PII: encryption, retention, and audit logs?	
<b>Notes:</b> _____	<b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject	
<b>Quick Notes: Performance, SEO, And Security Readiness</b>		
<b>Strongest proof:</b> _____		
<b>Main risk / gap:</b> _____		
<b>Follow-up question:</b> _____		

# 10-Point Ecommerce Development Partner Checklist

7. Integration Depth	Score: 0 1 2 3 4 5
<b>Outcome:</b> The partner plans for the real complexity outside the storefront: payments, ERP, CRM, PIM, OMS, analytics, inventory, shipping, and support systems.	
<b>Proof To Request</b>	<b>Questions To Ask</b>
<input type="checkbox"/> Integration case examples and data contracts <input type="checkbox"/> Retry, reconciliation, and error-handling strategy <input type="checkbox"/> Monitoring plan for sync failures, latency, and duplication risks	1. What happens when ERP is down? Do we degrade gracefully? 2. How do you reconcile inventory across channels? 3. How do you prevent double-charging or order duplication?
<b>Notes:</b> _____	<b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject
8. Transparent Delivery Process	Score: 0 1 2 3 4 5
<b>Outcome:</b> The team makes Agile rituals, QA, staging, release management, acceptance criteria, UAT, feature flags, and rollback planning visible before kickoff.	
<b>Proof To Request</b>	<b>Questions To Ask</b>
<input type="checkbox"/> Sprint cadence and sample status reports <input type="checkbox"/> QA strategy, device/browser matrix, and Definition of Done <input type="checkbox"/> Release playbook, UAT workflow, feature flags, and rollback plan	1. Who writes acceptance criteria: PM, QA, or engineers? 2. How do you handle scope changes without surprises? 3. What does UAT look like, and who is responsible for sign-off?
<b>Notes:</b> _____	<b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject
9. Ownership Mindset	Score: 0 1 2 3 4 5
<b>Outcome:</b> The partner builds like the system will need to be maintained, handed over, extended, and understood by another engineer later.	
<b>Proof To Request</b>	<b>Questions To Ask</b>
<input type="checkbox"/> Repo standards, branching strategy, and code review rules <input type="checkbox"/> Architecture notes, runbooks, and onboarding plan <input type="checkbox"/> Full access and ownership model for code, environments, and documentation	1. How quickly can a new engineer onboard this codebase? 2. What is your approach to reducing long-term complexity? 3. Do we get full access and ownership of repos and infrastructure?
<b>Notes:</b> _____	<b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject
<b>Quick Notes: Integrations, Delivery, And Ownership</b>	
<b>Strongest proof:</b> _____	
<b>Main risk / gap:</b> _____	
<b>Follow-up question:</b> _____	

10. Post-Launch Growth Support	Score: 0 1 2 3 4 5
<p><b>Outcome:</b> The partner supports stabilization, monitoring, incident response, A/B testing, roadmap planning, and iterative optimization after launch.</p>	
<b>Proof To Request</b>	<b>Questions To Ask</b>
<p><input type="checkbox"/> Monitoring stack for logs, metrics, traces, and alerts</p> <p><input type="checkbox"/> SLA options and incident response process</p> <p><input type="checkbox"/> A/B testing workflow, feature flags, and monthly roadmap plan</p>	<p>1. What do you monitor on day one: checkout errors, latency, or payment declines?</p> <p>2. How do you decide what to test next: data, heuristics, or stakeholder requests?</p> <p>3. What is your stabilization plan for the first 30 days post-launch?</p>
<p><b>Notes:</b> _____</p>	<p><b>Decision:</b> <input type="checkbox"/> Pass <input type="checkbox"/> Validate <input type="checkbox"/> Reject</p>

## Red Flags Before You Sign

Use this section as a final risk check before approving an ecommerce development proposal or SOW.

Red Flag	What To Check
<b>Vague estimates with no assumptions</b>	<p><input type="checkbox"/> No template count, integration count, QA plan, analytics line item, SEO migration plan, or change-control process.</p> <p><b>Ask:</b> What must be true for this timeline to hold? What is excluded that we will likely need later?</p>
<b>No performance or security acceptance criteria</b>	<p><input type="checkbox"/> No Core Web Vitals targets, load-testing plan, secure SDLC specifics, or penetration-testing approach.</p> <p><b>Ask:</b> What measurable gates must be passed before release?</p>
<b>No analytics or event tracking plan</b>	<p><input type="checkbox"/> No event taxonomy, staging validation, consent handling, deduplication plan, or attribution safeguard.</p> <p><b>Ask:</b> Which events will be validated before launch? How will marketing attribution be protected?</p>
<b>Proprietary lock-in without clear rationale</b>	<p><input type="checkbox"/> Limited repo access, unclear hosting ownership, black-box tooling, custom CMS without export paths, or missing documentation.</p> <p><b>Ask:</b> How would another team take over in 90 days if needed?</p>

## First Two Weeks: Deliverables To Request

Deliverable	What It Should Include	Received?
<b>Architecture Decision Record</b>	Platform, hosting, deployment approach, major tradeoffs, and rejected alternatives.	[ ]
<b>Measurement Plan</b>	Event taxonomy, funnel tracking, dashboard outline, ownership, and validation workflow.	[ ]
<b>SEO Migration Plan</b>	Redirects, canonical rules, sitemap handling, staging controls, and indexation safeguards.	[ ]
<b>Performance Budget</b>	CWV targets, image strategy, script rules, cache/CDN approach, and testing process.	[ ]
<b>Integration Map</b>	Systems, owners, data contracts, failure modes, retry strategy, and monitoring needs.	[ ]
<b>Release Plan</b>	Environments, feature flags, rollback strategy, launch checklist, and UAT sign-off flow.	[ ]

## Decision Path

### 1. Shortlist 3-5 vendors

Choose based on scope fit, integration depth, compliance needs, and evidence quality.

### 2. Validate with artifacts

Ask for architecture diagrams, QA approach, SEO migration plan, performance targets, and documentation samples.

### 3. Kick off with paid discovery or Phase 1

The first phase should produce a backlog, risk register, timeline, acceptance criteria, and launch plan.

## Vendor Notes And Follow-Up

Use this page to capture final questions, risks, and follow-up items after reviewing proposals.

Topic	Notes	Owner / Date
Scope assumptions		
Integration risks		
SEO migration concerns		
Security/compliance review		
Analytics/event tracking		
Launch risks		
Post-launch support		
Final decision		

### Need help validating your ecommerce development plan?

BrainX can help you review scope, architecture, performance, integrations, and launch readiness before the build starts.