

QA Technical Task

Surge Global

Naduni De Silva

nadunidesilva111@gmail.com

+9470 1857220

Contents

Question 1	3
Question 2	6
Comprehensive test strategy document	6
Question 3	18
Positive test.....	25
Negative test.....	40
Edge test.....	47
Question 4	55

Question 1

1. How is "same category" defined? Is it founded on the main category or the secondary category or both of ebay?
2. In case a wallet appears in several categories, which category should the system take to retrieve related products?
3. What is the definition of a best seller product? (sales volume, rating, number of reviews, conversion rate etc.)
4. Does it have a minimum requirement to become a best seller
5. What is the calculation of best seller status?
6. Is time period a factor in best seller status?
7. Are stopped or out-of-stock items to be included on best sellers?
8. What is meant by the same price range?
9. Is it grounded on precise product price or mean category price?
10. What acceptable variation?
11. And should the main product be on sale, is original price to be used in comparison or discounted price?
12. What are the price ranges of high-varying categories?
13. Is it possible that the system should not include the items that have abnormal prices (extremely cheap/expensive outliers)?
14. The requirement is to include up to six products the requirement is to display fewer than 6?
15. Inventory low in category?
16. Lack of sufficient best sellers?
17. None of the best sellers according to price range?
18. How many related products should be shown? (1, 3 or ought the section not to exist at all)?
19. Is there any priority when there are less than 6 best sellers?
20. How would you rank the 6 products? (best seller rank, price, rating, quantity of reviews, newest)
21. Should products be randomly selected by each user session or similar between the users?
22. Does it have any personalization in user browsing history or purchase history?

23. What product information would be important on the related products section? Image, name, cost, rating, the number of reviews, seller details, badge of prime/fast shipping.
24. Should product titles be dropped off?
25. Are there to be promotional badges or the signs of new, hot, or best seller on cards?
26. What about multiple images of products? Some pictures are accompanied by text, including carousels and other styles.< human >(primary image alone, or carousel)
27. Is it better not to include the main product in the list of related products?
28. Are related products supposed to have products of the same seller?
29. What is the issue of private seller and business seller?
30. Is this to be excluded in the case of items which have seller return policy conflict?
31. Filtered out be counterfeit or disputed items?
32. Should it open in a new tab or the same tab when a related product is clicked by a user?
33. Do related products have the option of adding products in the wishlist/cart directly in related products section?
34. Should there be an indication of a related product contained within the cart of the user?
35. Is it desirable to have an add to cart button on the card of every product?
36. What is the frequency of the associated products list? (real-time, hourly, daily)
37. Do the related products change every time that a user visits, or are they the same throughout a session?
38. Should the out of stock related products be shown?
39. Are there any differences in the way items with the labels of low stock should look?
40. Are products related to geographical area (in the same region, country)?
41. In the event that conversion of currency impacts price range matching?
42. Would the location of shipment have an impact on the availability of the products in the correlative products?
43. What will be shown when there are no best sellers within that category and price range?
44. In case the section is completely removed, a message is shown or only products which do not meet the required criteria are shown?
45. What would happen to the api of the concerned products?

46. What browsers should be supported? Outdated chrome and firefox browsers may experience protection issues when used with this platform.< human >older versions of chrome and firefox browsers can have protection problems with this platform.
47. What needs to be tested as far as mobile devices and os versions are concerned? (ios, android)
48. Does it have any accessibility needs? (screen reader support, wcag compliance)
49. Does this mean that related products should be operated the same way on desktop, tablet, and mobile?
50. What should related products do when the user is or is not logged-in?

Question 2

Comprehensive test strategy document

Table of contents

1. Executive summary
2. Feature overview
3. Test strategy framework
4. Test scope & objectives
5. Test categories & approach
6. Test scenarios & cases
7. Risk assessment & mitigation
8. Test execution plan & timeline
9. Test deliverables
10. Success metrics & kpis

1. Executive summary

Purpose

This paper defines the overall test approach to the ebay relevant products functionality that serves as the showcase of the most intensive purchases in the same categorical and price range as the product itself does.

This test strategy aims at achieving the following objectives:

- Refine a formalized, strict system to verify the related products feature in all functional, non-functional, as well as all edge-case scenarios.
- Check the reliability of product selection and visual presentation as well as interaction mechanisms.
- Early identification of the risks and critical blockers.
- Provide benchmarks on the quality that is to be continually assessed.
- Support a sure release with test coverage that is recorded

Scope

- Featured display of six top selling products at the related products section.
- Reasoning on matching category and filtering range of prices. Normal user interactions, like adding to cart, adding to wishlist and navigating to other products.
- User interface/user experience design and flexibility in a range of devices.
- Performance, caching and api combination. Disaster management plans and mitigation strategies

Success criteria

- 100% of test cases are passed with zero critical defects.
- Page loading time is less than the target (this should be set). The related products section is also seen in all the tested browsers and devices. Analytics events will record every interaction of the users.
- The accessibility criteria are met based on the compliance with wcag 2.1 aa.

2. Feature overview

Feature description

The related products option features as many as six highest selling items on the product description page. Such products have to be of the same category on the main item and they have to have similar prices. This capability improves exploration by users and increases cross-sell possibilities.

Key requirements

- Have 6 products (or less) on display (or less when the inventory is low).
- It should be products that are of the same category as the main product.
- Products are to be within the required price range.
- Products will require meeting the stipulated criteria as the best sellers.
- The title of the section must be similar sponsored items and must have a hyperlink of see all.
- The section allows the user to add products to the cart or wishlist.

Critical questions requiring clarification

Category	Clarification needed
Price range	Define acceptable variance: $\pm 10\%$, $\pm 20\%$, or fixed amount?
Best seller criteria	Define metric: sales volume, rating, time period?
Fallback logic	Show fewer products, show non-sellers, or hide section?
Refresh rate	Real-time, hourly, daily, or cached?
Category definition	Primary only, or include secondary categories?

3. Test strategy framework

Testing approach

The testing approach will be layered and comprehensive, combining multiple testing levels to ensure quality at all stages of development and deployment.

1. Unit testing

- Test individual functions: category matching, price range calculation, best seller ranking
- Validate data transformation and filtering logic
- Mock external dependencies (api, database queries)

2. Integration testing

- Test api calls for product data retrieval
- Validate database query performance and correctness
- Test integration with inventory, wishlist, and cart systems

3. Functional testing

- Manual and automated ui testing

- Verify product display, sorting, and filtering
- Test all user interactions (click, scroll, add to cart, wishlist)

4. Non-functional testing

- **Performance:** page load time, api response time, rendering speed
- **Accessibility:** wcag 2.1 aa compliance, keyboard navigation, screen reader support
- **Compatibility:** multiple browsers, devices, screen sizes
- **Usability:** user feedback, task completion, comprehension

5. Edge case & error handling testing

- Api failures, timeouts, malformed responses
- Insufficient inventory, no best sellers, price mismatches
- Data race conditions, concurrent requests, cache inconsistencies

6. Regression testing

- Ensure changes don't break existing product page functionality, cart, wishlist, or checkout flows

4. Test scope & objectives

In scope

- Related products section ui and interactions on product detail page
- Product selection algorithm (category, price, best seller logic)
- Add to cart/wishlist from related products section
- Page performance with section loaded
- Responsive design (mobile, tablet, desktop)
- Accessibility compliance
- Analytics event tracking

Out of scope

- Changes to product catalog management system

- Seller account management features
- Payment processing or checkout flows
- Global site search or category navigation improvements
- Mobile app testing (unless feature is deployed to app)

Primary objectives

1. Validate that related products are correctly selected based on category and price criteria
2. Verify ui displays exactly as designed across all supported devices
3. Confirm user interactions work correctly (cart, wishlist, navigation)
4. Ensure performance is acceptable with section loaded
5. Validate accessibility requirements are met
6. Identify and document any gaps in specifications or issues for correction

5. Test categories & approach

5.1 functional testing

Product selection & filtering

- Verify 6 best-selling products display for main product in wallet category
- Confirm all displayed products are in same category as main product
- Validate price range logic: all products within acceptable variance
- Confirm products are ranked as best sellers by defined metric
- Test with main product at various price points (low, mid, high)
- Verify main product is excluded from related products list

Ui display & elements

- Section title displays as "similar sponsored items"
- "see all" link displays and is clickable
- Product cards display with image, title, price, rating, reviews count
- Product titles truncate appropriately (no overflow)
- Wishlist heart icon visible and functional on each product card
- Section properly positioned on page below main product details

User interactions

- Click product card opens product detail page in same tab
- Add to wishlist from card updates wishlist immediately
- Add to cart button available and functional
- Cart confirmation message displays after add
- Hover states and visual feedback work correctly

5.2 edge cases & error handling

No related products scenarios

- Category with only 1-3 best sellers: section shows only available products
- No best sellers matching price range: section hidden or shows fallback message
- Niche category with no products in price range: verify fallback behavior
- Product price constantly changing: verify price range recalculation

Api & performance issues

- Api timeout: section does not load, no impact on main product display
- Api returns malformed response: graceful error handling, no crash
- Api returns empty results: appropriate fallback or message
- Slow api response: verify section loads asynchronously (lazy load)

5.3 performance testing

Load time

- Product page load without section slowdown: target <2 seconds (tbd)
- Related products section async load: <3 seconds additional (tbd)
- Api response time acceptable: <500ms (tbd)

Caching

- Repeated page loads: section loads from cache
- Cache invalidation: products update when inventory changes

- Cross-session consistency: same related products show across user sessions

5.4 responsive design & compatibility

Desktop

- 6 products display in 3x2 grid or 2x3 grid layout
- No horizontal scrolling or overflow
- Spacing and alignment consistent with design

Tablet (ipad, large android)

- Products display as 2x3 grid or carousel
- Touch targets appropriately sized for touch input
- Section remains readable without horizontal scrolling

Mobile (iphone, android phone)

- Products display as single column or horizontal scrollable carousel
- Product cards appropriately sized for mobile viewing
- Add to cart/wishlist buttons accessible and easily tappable
- No layout shift or reflow issues during scroll

Browser compatibility

- Chrome (latest 2 versions)
- Firefox (latest 2 versions)
- Safari (latest 2 versions, macos and ios)
- Edge (latest version)

5.5 accessibility testing

Wcag 2.1 level aa compliance

- Color contrast: text meets minimum 4.5:1 ratio
- Product images have alt text describing content
- Section marked with proper semantic html (headings, landmarks)

- All interactive elements have visible focus indicators

Keyboard navigation

- Tab order logical: section navigable with keyboard only
- Add to cart/wishlist buttons accessible via keyboard
- Product links navigable via enter key
- No keyboard traps or focus loss

Screen reader support

- Section announced properly (role, label) by nvda, jaws, voiceover
- Product information readable: image alt text, price, rating
- Button labels descriptive: "add to cart", not "click here"

6. Test scenarios & cases

Scenario 1: standard related products display

- **S1-1:** display 6 best-seller products → 6 products display in section; all marked as best sellers
- **S1-2:** all products in same category → all 6 products in "wallets" category
- **S1-3:** products within price range → all products within ±20% of main product price (tbd)

Scenario 2: price range validation

- **S2-1:** main product \$50 wallet → all related products within price range (e.g., \$40-\$60)
- **S2-2:** main product \$200 luxury wallet → all related products in premium range (e.g., \$180-\$220)
- **S2-3:** main product price changes → related products update to match new price range
- **S2-4:** product on sale → verify price range uses current (sale) price

Scenario 3: edge cases

- **S3-1:** only 3 best sellers available → section displays 3 products, not 6

- **S3-2:** no best sellers in price range → section hidden or shows "no similar products" message
- **S3-3:** main product is only item in category → section does not appear or shows appropriate fallback
- **S3-4:** related products api timeout → section does not load; main product page still functional

Scenario 4: user interactions

- **S4-1:** click related product → opens product detail page for that product
- **S4-2:** add related product to cart → product added, confirmation message shown
- **S4-3:** add related product to wishlist → wishlist updated, heart icon changes
- **S4-4:** click "see all" link → navigate to filtered product listing page
- **S4-5:** add multiple related products to cart → cart accumulates quantities correctly

Scenario 5: responsive design

- **S5-1:** desktop (1920x1080) → 6 products visible in 3x2 or 2x3 grid
- **S5-2:** tablet (768x1024) → 6 products visible as 2x3 grid or carousel
- **S5-3:** mobile (375x667) → products display as single-column or horizontal carousel
- **S5-4:** orientation change (portrait to landscape) → layout adapts smoothly

Scenario 6: accessibility

- **S6-1:** keyboard navigation → user tabs through all product cards and buttons
- **S6-2:** screen reader (nvda) → section announced, product info readable
- **S6-3:** color contrast → all text meets 4.5:1 wcag aa minimum
- **S6-4:** focus indicators → clear visual focus on all interactive elements

7. Risk assessment & mitigation

Critical risks

Risk	Impact	Probability	Mitigation

Unclear product selection criteria (price range, best seller definition)	High: incorrect products displayed	High: questions remain unanswered	Get stakeholder sign-off on criteria before development
Related products api performance impact	High: slow page load	Medium: depends on query optimization	Load section async; set performance budget; performance test early
Inventory changes affecting product availability	Medium: stale data shown	High: real-time inventory fluctuates	Define cache refresh strategy; validate on-page
Cross-browser/device incompatibilities	High: poor user experience	Medium: html/css variations	Start cross-browser testing early; use css prefixes; automate testing

Medium risks

- **Fallback behavior undefined** → test what happens when no products found or api fails
- **Analytics tracking missing** → implement tracking before beta launch; test events fire correctly
- **Accessibility overlooked** → ensure wcag aa compliance from start; audit early and often
- **Documentation incomplete** → create test plans, specs, and runbooks before development completes

8. Test execution plan & timeline

Phase 1: requirements validation (week 1)

- Clarify all open questions with product and development teams
- Document agreed criteria: price range formula, best seller metric, refresh rate
- Create detailed test scenarios based on confirmed specs

Phase 2: development & functional testing (weeks 2-3)

- Execute unit test cases as code is written
- Run functional tests on staging environment
- Log defects and track resolution
- Perform edge case and error handling tests

Phase 3: non-functional testing (week 4)

- Execute performance and load tests
- Complete responsive design and browser compatibility tests
- Run accessibility audits (automated + manual)
- Verify analytics events fire correctly

Phase 4: uat & launch prep (week 5)

- Regression testing on main product page functionality
- Final smoke testing on production-like environment
- Create launch checklist and rollback plan
- Document test results and sign-off for release

9. Test deliverables

Documents

- Test strategy document (this document)
- Detailed test case specifications with expected results
- Test execution report with pass/fail results
- Defect log with severity levels and resolution status
- Launch checklist and release sign-off form

Automation artifacts

- Automated test scripts (functional, api, performance)
- Accessibility audit automation

- Performance testing scripts (load time, api response)

10. Success metrics & kpis

Quality metrics

- **Test coverage:** 100% of identified scenarios covered
- **Defect escape rate:** zero critical/high severity escapes to production
- **Pass rate:** 98% or higher of test cases pass (excluding known issues)
- **Test execution rate:** 100% of planned tests executed before launch

Performance metrics

- **Page load time:** product page load <2 seconds (tbd based on requirements)
- **Related products load time:** <3 seconds additional (tbd)
- **Api response time:** <500ms 95th percentile (tbd)

User experience metrics

- **Cross-browser/device compatibility:** 100% pass on supported platforms
- **Accessibility:** 100% wcag 2.1 aa compliance
- **User interaction success:** add to cart/wishlist functions 100% reliably
- **Analytics:** 100% of events tracked and firing correctly

Question 3

Assumptions & clarifications

Product selection

Category definition

- **Assumption:** related products are filtered by primary category only (e.g., "wallets"), not secondary categories or tags
- **Assumption:** category matching is exact (e.g., wallets must match exactly; women's wallets is a sub-category and may have separate logic)

Validation needed: confirm if primary category only or include subcategories

Price range logic

- **Assumption:** price range is calculated as $\pm 20\%$ of the main product price
Example: \$50 product → \$40-\$60 range
- **Assumption:** if main product is on sale, the current (discounted) price is used for range calculation
- **Assumption:** price comparison uses the main product price at page load time

Validation needed: confirm price variance percentage ($\pm 10\%$, $\pm 15\%$, or $\pm 20\%$) and if sale prices are included

Best seller criteria

- **Assumption:** best sellers determined by sales volume in the last 30 days (monthly rolling window)
- **Assumption:** minimum threshold for best seller status: 10+ sales in the last 30 days
- **Assumption:** best seller ranking based on sales volume (highest to lowest)
- **Assumption:** out-of-stock items excluded from best seller list

Validation needed: confirm metric (volume vs. Rating), time window, and minimum threshold

Display & count behaviour

- **Assumption:** default display is 6 products per page (carousel or grid)
- **Assumption:** if fewer than 6 best sellers available in category/price range, show all available (e.g., 3, 4, or 5 products)
- **Assumption:** if zero best sellers match criteria, the "related products" section is hidden entirely
- **Assumption:** the main product is explicitly excluded from related products list

Validation needed: confirm fallback behavior when <6 products available

Data refresh & caching

- **Assumption:** related products list is cached for 1 hour per product id
- **Assumption:** cache is invalidated when main product inventory drops to zero or goes out of stock
- **Assumption:** best seller rankings recalculated daily at midnight utc

Validation needed: confirm cache duration and refresh strategy

Api & performance

- **Assumption:** related products section loads asynchronously (does not block main product display)
- **Assumption:** api call timeout set to 3 seconds; if no response by timeout, section shows fallback message
- **Assumption:** main product page load time target: <2 seconds (excluding related products)

Validation needed: confirm api timeout, performance targets, and async loading strategy

User interactions

- **Assumption:** clicking a related product opens its detail page in the same tab (no new tab)
- **Assumption:** add to cart/wishlist from related product section works identically to main product section
- **Assumption:** wishlist heart icon shows immediately updated state (filled/unfilled) after interaction

Validation needed: confirm interaction behavior and ui feedback

Display properties

- **Assumption:** section title: "similar sponsored items"
- **Assumption:** each product card displays: image, title, price, average rating, review count, wishlist icon
- **Assumption:** product titles truncated at 50 characters with ellipsis if longer
- **Assumption:** section placed immediately below main product details and above customer reviews

Validation needed: confirm display properties and section placement

Preconditions & setup requirements

All preconditions must be in place before any test case execution begins.

Environment setup

Requirement	Details	Status
Test environment access	Staging environment with related products feature enabled and deployed	Ready

Database access	Read access to staging product database to verify product selection logic	Ready
Browser tools	Chrome, firefox, safari, edge latest 2 versions with devtools	Ready
Accessibility tools	Nvda (windows), jaws (optional), voiceover (macos/ios) configured	Ready
Performance tools	lighthouse, browser performance tab accessible	Ready
Mobile devices	Iphone 12/13/14+ (ios 15+), samsung galaxy s21+ (android 12+)	Ready

Account & user setup

Requirement	Details
Tester accounts	3 pre-created ebay test accounts with known credentials
Empty wishlist	Each test account should have empty wishlist and cart at start
Fresh session	Clear all cookies and cache before starting each test scenario
Guest user	Ability to test feature as anonymous/guest user

Api & backend readiness

Requirement	Details
Related products api	Endpoint deployed and responding correctly (test with postman)

Mock error scenarios	Ability to simulate: timeout (3s+), 500 errors, malformed json, empty responses
Database snapshots	Staging db backup available for data state restoration between runs
Cache control	Ability to clear/purge related products cache for specific products

Test data requirements & data preparation

Master test products

Primary category: wallets

Product id	Product name	Category	Price	Stock	Purpose
Prod-001	Brown leather wallet	Wallets	\$49.99	50+	Primary test product
Prod-002	Black leather bifold	Wallets	\$39.99	30+	Within price range (\$40-\$60)
Prod-003	Rfid blocking wallet	Wallets	\$59.99	25+	Within price range (\$40-\$60)
Prod-004	Luxury designer wallet	Wallets	\$199.99	5+	High price range test
Prod-005	Budget canvas wallet	Wallets	\$9.99	100+	Low price range test
Prod-006	Blue denim shirt	Clothing	\$29.99	75+	Different category test
Prod-007	Usb-c cable	Electronics	\$12.99	200+	Different category test

Related products dataset for prod-001

Main product: prdo-001 @ \$49.99 (brown leather wallet)

Expected price: \$40-\$60 ($\pm 20\%$)

Product id	Product name	Price	Sales (30d)	Rating	Status
Rp-001	Black leather bifold	\$39.99	25 sales	4.5/5	Active
Rp-002	Rfid blocking wallet	\$59.99	32 sales	4.7/5	Active
Rp-003	Red leather wallet	\$44.99	18 sales	4.3/5	Active
Rp-004	Navy canvas wallet	\$42.99	15 sales	4.1/5	Active
Rp-005	Brown suede wallet	\$55.99	22 sales	4.6/5	Active
Rp-006	Tan leather wallet	\$52.99	20 sales	4.4/5	Active

Edge case test data

Scenario a: insufficient best sellers

- Test product: prod-edge-a @ \$75.00
- Only 3 products available in price range (\$60-\$90)
- Expected behavior: section displays 3 products, not 6

Scenario b: no related products

- Test product: prod-edge-b @ \$1,500.00
- No best sellers exist in price range (\$1,200-\$1,800)
- Expected behavior: section is hidden entirely

Scenario c: out of stock

- Test product: prod-edge-c @ \$45.00

- 4 related products in price range; 2 are out of stock
- Expected behavior: section displays 4 products (in-stock only)

Scenario d: on sale product

- Test product: prod-edge-d @ \$50.00 (normally \$80, on sale 37.5% off)
- Price range based on: current (sale) price = \$40-\$60
- Expected behavior: related products filtered by sale price, not original price

Scenario e: zero inventory

- Test product: prod-edge-e @ \$35.00 (stock = 0)
- Main product is out of stock
- Expected behavior: related products section hidden or displayed with disclaimer

Test case categories

Category	Focus area	# of tcs	Priority
Functional positive test	Category matching, price range, best seller criteria	12	P0-p1
Edge cases - api errors	Timeouts, 500 errors, malformed responses	3	P1
Negative tests	Invalid inputs, boundary conditions, edge values	5	P2
Regression	Main product page, cart, checkout not broken	5	P1

Total		25	
--------------	--	-----------	--

Possitive test

Tc-001: verify related products section displays on wallet product page

Category: positive | priority: p0 (critical)

Description: verify that the "related products" section is visible and accessible when navigating to a wallet product page.

Assumptions:

- Ebay wallet product page loads successfully without security blocks
- Related products section is always displayed for wallet products (or clearly shows when unavailable)
- Browser javascript is enabled (required for dynamic content)
- Ebay server is responding with 200 ok status

Preconditions:

1. Test environment has internet connectivity
2. Ebay website is accessible (not blocked by firewall/cdn)
3. Playwright browsers are installed (chromium, firefox, webkit)
4. Java 8+ and maven are installed
5. No vpn/proxy that might trigger ebay security blocks

Test data required:

- Valid ebay wallet product url: <https://www.ebay.com/itm/314710838801>
- Expected section identifier: div[class*='vi_vr'], text=/[rf]ecomended/

Test steps:

- Create new playwright browser instance

- Navigate to wallet product url
- Wait for page load state (networkidle)
- Identify related products container using flexible selectors
- Verify container is visible in dom
- Verify section has visible text content

Expected result:

- Related products section is visible
- Container has valid html structure
- Section contains at least one product card

Actual result: [to be filled during execution]

Tc-002: verify related products count does not exceed 6

Category: positive | priority: p0 (critical)

Description: verify that the related products section displays a maximum of 6 products as per ebay's design specification.

Assumptions:

- Ebay limits related products to 6 items maximum
- Product cards are consistently structured
- Hidden/invisible cards are not counted
- Ebay may display fewer than 6 for niche products

Preconditions:

1. Tc-001 must pass (related products section visible)
2. Product page fully loaded
3. Javascript execution complete
4. No dynamic loading/pagination involved

Test data required:

- Product card selector: div[class*='vi_vr_relitem'], div.s-item
- Expected max count: 6
- Test product: <https://www.ebay.com/itm/314710838801>

Test steps:

1. Navigate to wallet product url
2. Wait for related products section to load
3. Count all visible product card elements
4. Assert count is ≥ 1 (at least one product)
5. Assert count is ≤ 6 (max 6 products)
6. Verify all counted items are visible (not hidden/display:none)

Expected result:

- Product count is between 1 and 6
- All visible products are in viewport
- Count remains consistent on page refresh

Tc-003: verify each related product card has required elements (image, title, price)

Category: positive | priority: p0 (critical)

Description: verify that every related product card contains all required ui elements: product image, title/name, and price.

Assumptions:

- All products have images (no missing images)
- All products have english titles (no special characters breaking layout)
- All products have valid prices (no free products)
- Price format is consistent (currency symbol + amount)
- Images are optimized and load quickly

Preconditions:

1. Related products section visible (tc-001 passed)
2. Page fully loaded
3. All lazy-loaded images are rendered
4. No ad blockers interfering with content

Test data required:

- Image selector: img
- Title selector: a[href*='ebay.com'], h3, .vi_vr_relitemtitle
- Price selector: .vi_vr_cvipprice, .positive, span[class*='price']
- Test product: <https://www.ebay.com/itm/314710838801>

Test steps:

- Navigate to wallet product url
- Get all related product cards (max 6)
- For each card (index 0 to count-1):
 - Verify img element exists and is visible
 - Verify title/heading element exists and has text content
 - Verify price element exists and contains numeric value
- Log results for each product

Expected result:

- All 6 (or fewer) products have image element
- All products have non-empty title text
- All products have price with currency symbol and numeric value

Tc-004: verify main product is not included in related products list

Category: positive | priority: p1 (high)

Description: verify that the main wallet product being viewed does not appear in its own related products recommendations.

Assumptions:

- Main product title is unique or identifiable in dom
- Ebay filters out main product from recommendations
- Product matching is case-insensitive or exact match
- Main product id is available in url or page source

Preconditions:

1. Related products visible and loaded (tc-001, tc-002 passed)
2. Main product title extractable from h1 or meta tags
3. At least 1 related product displayed

Test data required:

- Main product url: <https://www.ebay.com/itm/314710838801>
- Main product id: 314710838801
- Main product title selector: h1.it-title, h1[class*='title']
- Related product card selector: div[class*='vi_vr_relitem']

Test steps:

- Navigate to wallet product url
- Extract main product title from h1 element
- Get main product id from url
- Retrieve all related product titles (1-6)
- For each related product:
 - Extract product title
 - Assert title ≠ main product title
 - Verify product url/id ≠ main product id
- Log all product titles found

Expected result:

- Main product title not found in related products list
- Main product id not in any related product url
- Related products are different items from main product

Tc-005: verify related products are in same category (wallets)

Category: positive | priority: p1 (high)

Description: verify that all related products are from the same category as the main product (wallets/leather goods).

Assumptions:

- Product categories are tagged in page metadata or structure
- All related products are wallets or similar leather goods
- Category information accessible via html attributes or page text
- No cross-category recommendations

Preconditions:

1. Product page fully loaded
2. Related products visible (tc-001 passed)
3. Category metadata available in page source

Test data required:

- Main product category: "wallets" or "leather goods"
- Category indicators in html: breadcrumb, category attributes, page text
- Test product: <https://www.ebay.com/itm/314710838801>
- Category validation keywords: wallet, leather, billfold, purse

Test steps:

1. Navigate to wallet product page
2. Extract main product category from breadcrumb or page structure
3. Verify main category contains "wallet" or "leather"
4. Get all related product titles and descriptions
5. For each related product:
 - Extract product title/description
 - Verify title contains wallet-related keywords

6. Count matches vs. Non-matches

Expected result:

- Main product is in wallets/leather goods category
- All related products have wallet-related keywords in title
- No cross-category items (shoes, bags, etc.) In recommendations

Tc-006: verify price range logic (related products ±20% of main product)

Category: positive | priority: p1 (high)

Description: verify that related products are priced within ±20% of the main product price to ensure relevance and similar price point.

Assumptions:

- All products have valid numeric prices
- Price currency is consistent (usd)
- Price extraction correctly handles currency symbols
- At least 70% of related products fall within range (some may be outliers for relevance)
- Price comparisons use extracted numeric values only

Preconditions:

1. Main product price is available
2. Related products loaded (tc-001, tc-002 passed)
3. Price elements render correctly
4. No regional pricing variations

Test data required:

- Main product price selector: .vi-vr-cvipprice, span[id*='price']
- Related product price selector: .vi_vr_cvipprice, .positive
- Price range percentage: ±20% (0.80 - 1.20 multiplier)
- Acceptable match rate: ≥70%

Test steps:

1. Navigate to wallet product url
2. Extract main product price
3. Remove currency symbols and convert to double
4. Calculate lower limit = price \times 0.80
5. Calculate upper limit = price \times 1.20
6. Extract all related product prices
7. For each related product:
 1. Extract and clean price
 2. Check if price \geq lowerlimit and price \leq upperlimit
 3. Count valid prices
8. Calculate match percentage: validcount / totalcount
9. Assert match percentage \geq 0.70 (70%)

Expected result:

- Main product price successfully extracted
- At least 70% of related products within $\pm 20\%$ price range
- Price extraction handles various formats correctly

Tc-007: verify clicking related product navigates to product detail page**Category: positive | priority: p1 (high)****Description:** verify that clicking on a related product card navigates user to that product's detail page with correct url and content.**Assumptions:**

- Product links are functional
- Navigation completes within reasonable timeout (10 seconds)
- New page loads with product detail content
- Product id is accessible in new url
- No popups or modal dialogs block navigation

Preconditions:

1. Related products visible (tc-001 passed)
2. At least 1 related product card available
3. Network connectivity stable
4. Browser allows navigation

Test data required:

- First related product selector: div[class*='vi_vr_relitem'] index 0
- Product link selector within card: a[href*='ebay.com']
- Expected url pattern: /itm/[0-9]+

Test steps:

1. Navigate to main wallet product url
2. Capture current page url
3. Locate first related product card
4. Click on product link/card
5. Wait for page load (networkidle)
6. Verify url changed
7. Verify new url matches product detail pattern
8. Verify product title/image loaded
9. Verify back button returns to main product

Expected result:

- Url changes after click
- New url contains product id (/itm/[0-9]+)
- New page displays related product details
- Navigation history preserved
- Can return to original page with back button

Tc-008: verify related products card layout and spacing

Category: positive | priority: p2 (medium)

Description: verify that related product cards are displayed in proper grid/list layout with consistent spacing and alignment.

Assumptions:

- Cards use css grid or flexbox for layout
- Spacing is consistent between all cards
- Cards maintain aspect ratio for images
- Layout adapts based on viewport
- No overlapping elements

Preconditions:

1. Related products visible (tc-001 passed)
2. Css fully loaded and applied
3. Page rendered without layout shifts
4. Viewport set to standard size (1920x1080)

Test data required:

- Product card selector: div[class*='vi_vr_relitem']
- Expected layout: grid with 3-6 columns
- Expected spacing: 10-20px between cards
- Image aspect ratio: square or 4:3

Test steps:

1. Set viewport to 1920x1080 (desktop)
2. Navigate to product page
3. Wait for render complete
4. Get all product card elements
5. For each card:
 1. Get computed dimensions (width, height)

2. Get position (x, y coordinates)
3. Verify consistent width across cards
4. Verify consistent spacing from neighbors
6. Verify cards form aligned grid
7. Check for overlaps/misalignment

Expected result:

- Cards display in aligned grid pattern
- All cards have consistent width
- Spacing between cards is uniform ($\pm 2\text{px}$ variance acceptable)
- Cards vertically aligned
- No overlapping elements
- Image maintains aspect ratio

Tc-009: verify desktop responsiveness (1920x1080)

Category: positive | **priority:** p1 (high)

Description: verify that related products section is fully functional and properly displayed on desktop viewport (1920x1080).

Assumptions:

- Desktop users have 1920x1080+ viewport
- All elements visible without horizontal scrolling
- Grid displays 6 columns or appropriate layout
- Touch targets not applicable on desktop
- Font sizes readable (14px minimum)

Preconditions:

1. Viewport set to 1920x1080 before navigation
2. Browser window maximized
3. Browser zoom at 100%

4. No browser extensions affecting layout

Test data required:

- Desktop viewport: 1920x1080
- Test product: <https://www.ebay.com/itm/314710838801>
- Expected visible products: 6 or maximum available

Test steps:

1. Set page viewport to 1920x1080
2. Navigate to wallet product url
3. Wait for page load
4. Verify related products section visible
5. Verify all 6 products visible in viewport (no horizontal scroll)
6. Verify text is readable (font size \geq 14px)
7. Verify images load correctly
8. Verify no overlapping elements
9. Verify buttons/links are accessible (24px+ height)

Expected result:

- Related products fully visible without horizontal scroll
- All 6 products display in optimal grid layout
- Text readable without zoom
- Images load quickly and properly
- No layout breaks or misalignment

Tc-010: verify mobile responsiveness (375x667)

Category: positive | priority: p1 (high)

Description: verify that related products section is fully functional and properly displayed on mobile viewport (375x667, typical mobile device).

Assumptions:

- Mobile users have 375-425px viewport width
- Products stack vertically (single or 2-column layout)
- Touch-friendly spacing (44px+ tap targets)
- Images responsive and scale appropriately
- No horizontal scroll required
- Text readable at mobile zoom level

Preconditions:

1. Viewport set to 375x667 before navigation
2. Mobile user agent simulated (if applicable)
3. Touch events enabled (if testable)
4. Browser zoom at 100%

Test data required:

- Mobile viewport: 375x667
- Test product: <https://www.ebay.com/itm/314710838801>
- Expected layout: vertical stack (1 or 2 columns)
- Touch target size: 44x44px minimum

Test steps:

1. Set page viewport to 375x667
2. Navigate to wallet product url
3. Wait for page load
4. Verify related products section visible
5. Verify products stack vertically (no horizontal scroll)
6. Verify touch targets \geq 44x44 pixels
7. Verify text readable (no zoom required)
8. Verify images scale appropriately
9. Verify spacing between products adequate for touch

Expected result:

- Related products display in mobile-optimized layout
- No horizontal scrolling required
- Touch targets easily clickable
- Text readable without zoom
- Images responsive and properly sized
- Vertical scrolling only (responsive to content)

Tc-011: verify page load performance (< 10 seconds)**Category: positive | priority: p2 (medium)**

Description: verify that the product page with related products loads within acceptable performance threshold (< 10 seconds) for good user experience.

Assumptions:

- Performance measured from navigation start to networkidle
- Network condition: standard 4g (typical user)
- Server is responsive and not overloaded
- No external api timeouts
- Cdn functioning properly

Preconditions:

1. Network connectivity stable
2. Ebay server responding normally
3. No throttling applied (unless testing specific network conditions)
4. Page cache cleared

Test data required:

- Test product url: <https://www.ebay.com/itm/314710838801>
- Performance threshold: 10,000ms (10 seconds)

- Acceptable range: 2,000-8,000ms (typical)

Test steps:

1. Clear browser cache
2. Record start timestamp
3. Navigate to wallet product url
4. Wait for page load complete (networkidle)
5. Record end timestamp
6. Calculate total load time
7. Assert load time < 10 seconds
8. Log load time for reporting

Expected result:

- Page loads in < 10 seconds
- Related products section visible in < 5 seconds (optimal)
- No timeout errors
- All resources loaded successfully

Tc-012: verify related products display across different price points

Category: positive | priority: p1 (high)

Description: verify that related products feature works consistently for wallet products across different price ranges: budget (\$10-30), mid-range (\$50-150), and premium (\$200+).

Assumptions:

- Ebay has products in all price tiers
- Related products algorithm works for all tiers

- Price variations don't affect feature availability
- Test products are in good standing (not suspended)

Preconditions:

1. Three wallet products selected with different price points
2. All products are active/available for purchase
3. All products have sufficient data for recommendations

Test data required:

- Budget wallet: [https://www.ebay.com/itm/\[budget-wallet-id\]](https://www.ebay.com/itm/[budget-wallet-id])
Expected price: \$10-30
- Mid-range wallet: [https://www.ebay.com/itm/\[mid-range-wallet-id\]](https://www.ebay.com/itm/[mid-range-wallet-id])
Expected price: \$50-150
- Premium wallet: [https://www.ebay.com/itm/\[premium-wallet-id\]](https://www.ebay.com/itm/[premium-wallet-id])
Expected price: \$200+

Test steps:

1. For each price tier (budget, mid, premium): a. Navigate to wallet product url b. Wait for page load c. Verify related products section visible d. Get product count e. Verify at least 1 related product f. Verify prices relevant to tier g. Log results
2. Compare results across tiers

Expected result:

- Related products display for budget wallets
- Related products display for mid-range wallets
- Related products display for premium wallets
- Product count ≥ 1 for each tier
- Price points consistent within each tier

Negative test

Tc-013: verify graceful error handling - invalid product id

Category: negative | priority: p1 (high)

Description: verify that the application handles invalid/non-existent product ids gracefully without crashes or showing broken ui.

Assumptions:

- Ebay returns error page for invalid product ids
- Error page displays appropriate message
- Related products section not visible on error pages
- Application doesn't crash or show 500 errors
- User can navigate back or to home page

Preconditions:

1. Browser can access ebay
2. Network connectivity available
3. Invalid product id prepared (e.g., 999999999)

Test data required:

- Invalid product url: <https://www.ebay.com/itm/999999999>
- Expected response: 404 error page or redirect
- Error message indicators: "not found", "no longer available", "invalid"

Test steps:

1. Navigate to invalid product id url
2. Wait for page load
3. Verify page loads (no 500 error)
4. Verify related products section not visible
5. Verify error message displayed (if applicable)
6. Verify page structure intact
7. Verify navigation elements available (back button, home link)
8. Attempt to navigate back

9. Verify browser history intact

Expected result:

- Page loads without crashing (200 or 404 status)
- Related products section not visible
- User sees appropriate error message or "product not found"
- Page layout not broken
- Can navigate away successfully

Tc-014: verify empty state - product with no related items

Category: negative | priority: p2 (medium)

Description: verify that when a product has no related products available, the application displays appropriate empty state or handling without confusion.

Assumptions:

- Some products legitimately have no related items
- Empty state message is user-friendly
- Application doesn't crash with empty list
- Main product still displays properly
- Section either hidden or shows "no related products"

Preconditions:

1. Identify product with no related items (or use stub)
2. Product is valid and loads successfully
3. Related products genuinely unavailable (not loading error)

Test data required:

- Test product (no related items): rare/niche wallet product
- Expected behavior: section hidden or "no recommendations" message

Test steps:

1. Navigate to product with no related items
2. Wait for page full load
3. Check if related products section visible
4. If visible, verify:
 1. No product cards displayed
 2. Message shown: "no related products" or similar
 3. Section properly styled (not broken)
5. If hidden, verify:
 1. Section completely hidden
 2. Space properly allocated
 3. No layout shift
6. Verify main product displays correctly
7. Verify page usable despite empty recommendations

Expected result:

- Related products section appropriately hidden or shows empty message
- Main product displays fully functional
- No broken ui elements
- User not confused about feature
- Page remains usable

Tc-015: verify out-of-stock products handled correctly

Category: negative | priority: p2 (medium)

Description: verify that related products feature works even when products are out of stock, and stock status is clearly indicated.

Assumptions:

- Ebay allows viewing out-of-stock products
- Out-of-stock indicator visible (label, button state, price)
- Feature doesn't break for out-of-stock items

- Related products still displayed (or appropriately handled)
- Add to cart disabled for out-of-stock

Preconditions:

1. Product is valid but out of stock
2. Product detail page loads successfully
3. Stock status information available on page

Test data required:

- Out-of-stock product url: identified or created
- Out-of-stock indicators: "out of stock", disabled button, "sold out"
- Expected: feature still works, stock status clear

Test steps:

1. Navigate to out-of-stock product
2. Wait for page load
3. Verify stock status clearly displayed
4. Verify related products section visible (or appropriately handled)
5. If related products shown:
 1. Verify product count ≥ 1
 2. Verify prices display
 3. Verify can view related product details
6. Verify "add to cart" button disabled
7. Verify product details display despite out-of-stock

Expected result:

- Out-of-stock status clearly visible
- Related products still displayed
- Can view related products
- Add to cart disabled
- Page fully functional

- User not confused about status

Tc-016: verify no duplicate products in related list

Category: negative | priority: p1 (high)

Description: verify that no product appears more than once in the related products list (no duplicates shown to user).

Assumptions:

- Each related product is unique
- Product ids uniquely identify products
- Urls or product links don't duplicate
- Duplicate detection uses product id or exact url

Preconditions:

1. Related products visible (tc-001 passed)
2. At least 2 related products displayed
3. Can extract product ids or urls

Test data required:

- Related product selector: div[class*='vi_vr_relitem']
- Product link selector: a[href*='ebay.com/itm']
- Product id extraction pattern: /itm/([0-9]+)

Test steps:

1. Navigate to wallet product
2. Get all related products (up to 6)
3. For each product:
 1. Extract product url
 2. Extract product id from url
 3. Add to list

4. Check for duplicates:
 1. Create set of ids
 2. Compare set size to list size
5. Assert no duplicates (set size == list size)

Expected result:

- All 6 related products are unique
- No product id appears twice
- No product url appears twice
- Set size equals list size

Tc-017: verify product diversity - no seller bias

Category: negative | priority: p2 (medium)

Description: verify that related products come from diverse sellers and don't show disproportionate representation from single seller (no algorithm bias).

Assumptions:

- Related products from multiple sellers
- Single seller doesn't monopolize recommendations
- Seller information visible or extractable
- Diversity indicates algorithm is working correctly
- Ebay prevents seller bias in recommendations

Preconditions:

1. Related products visible (tc-001 passed)
2. At least 3 related products displayed
3. Seller information extractable from product links or page

Test data required:

- Related product card selector: div[class*='vi_vr_relitem']

- Seller name pattern in url or page

Test steps:

1. Navigate to wallet product
2. Get main product seller name (if visible)
3. Get all related product seller info (if available)
4. For each related product:
 1. Try to extract seller name/id
 2. Count products per seller
5. Verify:
 1. Related products from different sellers (if seller info available)
 2. Main product seller doesn't dominate list
 3. No more than 2 products from same seller (if 6 total)

Expected result:

- Related products from multiple sellers
- No single seller has > 50% of recommendations
- Diversity in product sources
- Main product seller not over-represented

Edge test

Tc-018: verify browser back button navigation

Category: edge case | priority: p2 (medium)

Description: verify that after clicking a related product and viewing its details, the back button returns to original product with related products still visible.

Assumptions:

- Browser back button maintains page state
- Related products reload correctly
- Scroll position may or may not be preserved

- No data loss or broken state on return

Preconditions:

1. Related products visible (tc-001 passed)
2. At least 1 related product available
3. Browser history tracking enabled

Test steps:

1. Navigate to main wallet product
2. Capture url
3. Click first related product
4. Wait for new page load
5. Verify on new product page
6. Click browser back button
7. Wait for page load
8. Verify back to original wallet product
9. Verify related products section still visible
10. Verify product count same as before

Expected result:

- Back button returns to original product page
- Related products section visible
- Product list intact
- Page fully functional
- No data loss

Tc-019: verify data consistency on page refresh

Category: edge case | priority: p2 (medium)

Description: verify that related products list remains consistent when page is refreshed (same products appear in same order).

Assumptions:

- Related products are deterministic (same set returned each time)
- Server caches recommendations properly
- No randomization in display order
- Refresh doesn't change recommendations

Preconditions:

1. Related products loaded (tc-001 passed)
2. At least 2 related products visible
3. Product ids extractable

Test steps:

1. Navigate to wallet product
2. Extract all related product ids (positions 0-5)
3. Refresh page
4. Wait for page load
5. Extract related product ids again
6. Compare lists:
 1. Same number of products
 2. Same product ids in same positions
7. Repeat 2-3 times

Expected result:

- Related products consistent across refreshes
- Same products appear in same order
- No randomization
- Stable recommendations

Tc-020: verify cross-browser compatibility

Category: edge case | priority: p1 (high)

Description: verify that related products feature works consistently across different browsers: chromium, firefox, and webkit (safari).

Assumptions:

- All browsers render html/css correctly
- Javascript works in all browsers
- Feature is not browser-specific
- Css transforms/animations work correctly
- No browser-specific bugs

Preconditions:

1. All three browsers installed (playwright)
2. Browser compatibility mode not required
3. Test product accessible from all browsers

Test steps:

1. For each browser (chromium, firefox, webkit):
 - A. Create browser instance with that engine
 - B. Navigate to wallet product url
 - C. Wait for page load
 - D. Verify related products section visible
 - E. Get product count
 - F. Verify at least 1 product
 - G. Click first related product
 - H. Verify navigation works
 - I. Go back
 - J. Verify back on original page
 - K. Close browser
2. Compare results across browsers

Expected result:

- Related products visible in chromium
- Related products visible in firefox
- Related products visible in webkit
- All interactions work in each browser
- Consistent product count across browsers
- Navigation works in all browsers

Regression test

Tc-reg-001: main product page functionality not broken

Field	Value
Test id	Tc-reg-001
Title	Verify main product page functionality remains intact
Priority	P0 (critical)
Test steps	1. Navigate to prod-001 2. Verify main product elements: • product title displays • images display and zoom • price shows correctly • stock status shown • main "add to cart" works • main "add to wishlist" works
Expected result	All main product features work No regression from related products feature All original functionality intact

Tc-reg-002: shopping cart not affected

Field	Value
Test id	Tc-reg-002
Title	Verify shopping cart functionality not broken
Priority	P0 (critical)
Test steps	<ol style="list-style-type: none"> 1. Add main product to cart 2. Add related product to cart 3. Navigate to shopping cart 4. Verify both items in cart 5. Update quantities 6. Remove items 7. Verify cart totals correct 8. Proceed to checkout
Expected result	<p>Both products in cart</p> <p>Quantities editable</p> <p>Cart totals correct</p> <p>Checkout works</p> <p>No cart errors</p>

Tc-reg-003: wishlist not affected

Field	Value
Test id	Tc-reg-003
Title	Verify wishlist functionality not broken
Priority	P0 (critical)

Test steps	<ol style="list-style-type: none"> 1. Add main product to wishlist 2. Add related product to wishlist 3. Navigate to wishlist page 4. Verify both items present 5. Remove item from wishlist 6. Add items to cart from wishlist 7. Verify wishlist updates
Expected result	<p>Main and related products in wishlist</p> <p>Wishlist page loads</p> <p>Items removable</p> <p>Add to cart works</p> <p>No wishlist errors</p>

Tc-reg-004: customer reviews not affected

Field	Value
Test id	Tc-reg-004
Title	Verify customer reviews section not affected
Priority	P1 (high)
Test steps	<ol style="list-style-type: none"> 1. Navigate to prod-001 2. Scroll past related products 3. Verify reviews section present 4. Verify reviews load 5. Verify review count correct 6. Verify overall rating displays

Expected result	<ul style="list-style-type: none"> ✓ reviews section displays ✓ reviews load correctly ✓ review count matches product ✓ rating accurate ✓ review filtering/sorting works
-----------------	---

Tc-reg-005: page load performance not degraded

Field	Value
Test id	Tc-reg-005
Title	Verify page load performance not degraded
Priority	P1 (high)
Tool	Lighthouse,
Baseline	Performance score: >85
Test steps	<p>Run lighthouse audit on product page</p> <p>Compare to baseline</p> <p>Check fcp, lcp metrics</p> <p>Verify no degradation</p> <p>5. Performance score should remain >85</p>
Expected result	<p>Performance score: ≥ 85</p> <p>Fcp: <1.5 sec</p> <p>Lcp: <2.5 sec</p> <p>No regression from baseline</p> <p>Async loading working</p>

Question 4

Bug report

Ebay related products feature

Project: ebay related products feature

test suite execution: phase 2 (week 3)

report date: january 18, 2026

reported by: qa team

total issues: 5 (1 critical, 2 high, 2 medium)

Issue 01: related products section not loading on mobile devices

Bug report summary

Field	Value
Bug id	Bug-001
Title	Related products section fails to load on mobile devices
Severity	Critical
Priority	P0
Status	Open
Component	Related products api / mobile frontend
Environment	Staging
Devices affected	Iphone 13 (ios 15.7), samsung galaxy s21+ (android 12)
Reported date	January 18, 2026
Assigned to	Dev lead (mobile team)

Description

The related products section does not load on mobile devices (ios and android). When navigating to a product detail page on mobile, the section either shows as blank with no content, or fails to load entirely after waiting more than 3 seconds. The section title "similar sponsored items" may be visible, but product cards do not render.

Steps to reproduce

3. Open iphone 13 (375x667 viewport)
4. Navigate to product prod-001 (brown leather wallet)
5. Scroll down to related products section

6. **Expected:** 6 product cards display in single-column or carousel layout
7. **Actual:** section shows empty space or loading indicator that never resolves
8. Wait 5+ seconds
9. **Actual:** no product cards appear; section remains blank

Also reproducible on: samsung galaxy s21+ (384x809 viewport)

Expected behavior

- Related products section should load and display 6 product cards
- On mobile (375px width), cards should display in single-column or horizontal carousel layout
- Section should load asynchronously without blocking main product display
- Complete within 3 seconds as per design specification

Actual behavior

- Section displays "similar sponsored items" header but no product cards
- Blank white space where product cards should appear
- Loading spinner may show but never completes
- After 5+ seconds, section remains empty
- No error messages shown to user
- No javascript errors visible (or errors occur silently)

Screenshots/video evidence

Device: iphone 13, safari

url: staging.ebay.com/itm/prod-001

description:

- Main product information loads correctly
- Related products title visible
- No product cards appear below title
- Blank section with no content

Browser console errors:

(to be verified - check if api response fails on mobile)

Root cause analysis (hypothesis)

1. Api endpoint may be returning mobile-specific response format that frontend doesn't handle
2. Css media query for mobile layout may be hiding product cards due to sizing/display issues
3. Javascript may be blocking on loading product images (not lazy-loaded on mobile)
4. Api call may timeout more frequently on mobile (slower networks)
5. Responsive design css may have conflicting rules causing cards not to render

Impact

Severity: critical

- Related products feature completely broken on mobile
- Affects ~60% of ebay traffic (mobile users)
- Users cannot see related products or perform cross-sell actions
- Feature non-functional for majority of user base
- Launch blocker - must be fixed before production release

Acceptance criteria for fix

Related products section loads within 3 seconds on mobile

all 6 product cards visible on iphone 13 (375x667)

all 6 product cards visible on samsung s21+ (384x809)

single-column layout or horizontal carousel on mobile

no product cards cut off or hidden

images lazy-loaded and scale appropriately

touch targets >44x44px

no javascript errors in console

passes tc-ui-009, tc-resp-003, tc-resp-006

Additional notes

- **Network conditions:** tested on wifi and 4g/lte - issue occurs on both
- **Browser versions:** tested on safari (ios 15.7) and chrome (mobile)
- **Regression risk:** low - only affects mobile responsive feature, not core logic

Recommended fix actions

1. **Debug on mobile device:** use devtools remote debugging on real iphone/android
2. **Check api response:** verify api returns correct data structure on mobile requests
3. **Check css media queries:** review mobile css rules for related products section
4. **Check image loading:** ensure images are lazy-loaded, not blocking render
5. **Test async loading:** verify api call doesn't timeout on slower mobile networks
6. **Verify viewport meta:** ensure viewport meta tag correctly configured for mobile

Related issues

- Potentially related to: bug-002 (price range filtering not working)
- May impact: mobile user conversion rates, cross-sell metrics

Issue 02: price range filtering showing out-of-range products

Bug report summary

Field	Value
Bug id	Bug-002
Title	Products outside price range appearing in related products
Severity	High
Priority	P1
Status	Open
Component	Related products api / product filtering logic

Environment	Staging
Date discovered	January 18, 2026
Assigned to	Backend development lead

Description

When viewing product prod-001 (brown leather wallet @ \$49.99), the related products section displays products outside the expected price range ($\pm 20\% = \$40-\60). Specifically, product rp-004 is shown at \$42.99 when it should be within range, but some test runs show products as low as \$35.99 or as high as \$65.99 appearing in the related products list, which violates the price range filtering requirement.

Steps to reproduce

1. Navigate to prod-001 (brown leather wallet, \$49.99)
2. Scroll to related products section
3. Examine price of each product displayed
4. Compare against expected range: \$40.00-\$60.00 ($\pm 20\%$)
5. **Expected:** all 6 products within \$40.00-\$60.00
6. **Actual:** found product priced at \$35.99 (below range) and \$65.99 (above range)

Test runs showing issue:

- Run 1: rp-007 @ \$35.99 displayed (fail)
- Run 2: rp-008 @ \$65.99 displayed (fail)
- Run 3: all products within range (pass) - intermittent!

Expected behavior

- All related products must be within $\pm 20\%$ of main product price (\$49.99)
- Price range: \$40.00 - \$60.00
(calculated as $\$49.99 \times 0.80 = \39.99 and $\$49.99 \times 1.20 = \59.99)

- Products priced at \$35.99, \$65.99, or outside range should not appear
- Filtering should be consistent across all product views

Actual behavior

- Some products appear outside the expected price range
- The issue is **intermittent** - not consistent on every page load
- Sometimes all 6 products are within range
- Sometimes out-of-range products appear
- Suggests caching or data freshness issue
- Price filtering logic may have off-by-one or rounding error

Screenshots/video evidence

Product details:

- Main: prod-001 @ \$49.99
- Expected range: \$40.00-\$60.00
- Actual: rp-007 @ \$35.99 , rp-008 @ \$65.99

Impact

Severity: high

- Core filtering requirement violated
- Related products not truly "related" by price
- Violates test requirement tc-fs-003
- May show budget items with luxury items (confusing to users)
- Cross-sell effectiveness reduced (wrong price points)
- Must be fixed before launch - functional requirement

Affected test cases

- **Tc-fs-003:** verify price range logic ($\pm 20\%$) - **failed**
- **Tc-ec-007:** price boundary test - exactly at range limit - **failed**
- **Tc-ec-008:** price boundary test - just outside range - **failed**

Acceptance criteria for fix

All related products within $\pm 20\%$ of main product price
No products below lower range limit
no products above upper range limit
consistent across all page loads (not intermittent)
rounding handled correctly for edge cases
cache invalidation working correctly
works for low (\$9.99), mid (\$49.99), high (\$199.99) price points
passes tc-fs-003, tc-ec-007, tc-ec-008

Testing strategy for fix verification

Test	Purpose	Data
Boundary test	Verify exact boundary products	Test products at \$40.00, \$60.00
Decimal test	Handle .99 and .50 prices	\$49.99 main, \$39.99-\$59.99 range
Low price test	Handle budget products	\$9.99 main, \$7.99-\$11.99 range
High price test	Handle luxury products	\$199.99 main, \$159.99-\$239.99 range
Consistency test	No intermittent failures	Load same product 10 times

Additional notes

- **Frequency:** issue reproduced 2 out of 5 test runs (~40% failure rate)
- **Consistent on:** desktop and mobile both show the issue
- **Data issue?** May indicate test data not set up correctly with correct prices
- **Database check needed:** verify rp-007 and rp-008 actual prices in staging db

Related issues

- May be related to: bug-005 (api returning inconsistent data)
- Could impact: user trust in recommendations, conversion rates

Issue 03: wishlist heart icon not updating after interaction

Bug report summary

Field	Value
Bug id	Bug-003
Title	Wishlist heart icon state not updating immediately
Severity	High
Priority	P1
Status	Open
Component	Related products ui / frontend state management
Environment	Staging
Date discovered	January 18, 2026
Assigned to	Frontend development lead

Description

When clicking the wishlist heart icon on a related product card, the heart icon does not immediately update its state (filled/unfilled) to provide visual feedback to the user. The product is successfully added to the wishlist (verified on wishlist page), but the heart icon on the product card continues to show the old state (unfilled) even after being added. User must refresh page to see updated state.

Steps to reproduce

1. Login to staging account (test user: test@example.com)
2. Navigate to prod-001 (brown leather wallet)
3. Scroll to related products section
4. Locate rp-001 (black leather bifold) with unfilled heart icon
5. Click wishlist heart icon
6. **Expected:** heart icon immediately fills with red color
7. **Actual:** heart icon remains unfilled; no visual change
8. Wait 2-3 seconds
9. **Actual:** heart icon still appears unfilled on this page
10. Navigate to wishlist page (separate page load)
11. **Actual:** rp-001 appears in wishlist ✓ (product was added successfully)
12. Navigate back to prod-001
13. **Actual:** heart icon now shows as filled

Expected behavior

- Clicking heart icon should immediately change its visual state
- Unfilled heart (outline) → filled heart (solid red) when added to wishlist
- No page refresh required
- No loading delay visible to user
- Feedback is instant (within 100ms)
- State change persists until user removes from wishlist

Actual behavior

- Click heart icon
- No immediate visual feedback
- Heart icon remains in previous state (unfilled)
- Product is successfully added to wishlist (backend working)
- User unaware if click was registered (confused about action)
- Icon only updates after page refresh
- Suggests frontend state not synchronized with backend action

Evidence

Test results:

- Product added to wishlist: (verified on wishlist page)
- Heart icon updated immediately: (did not change)
- Heart icon updated after refresh: (correct state after reload)

Impact

Severity:

high

- Confusing user experience - no feedback for action
- User unsure if click was registered
- May click multiple times thinking first click didn't work
- Can result in duplicate wishlist additions
- Violates tc-ix-004 acceptance criteria
- Affects user confidence in feature

Affected test cases

- **Tc-ix-004:** add related product to wishlist - **failed**
- **Tc-ui-005:** verify wishlist heart icon state - **failed**
- **Tc-ix-006:** remove related product from wishlist - **likely fails**

Acceptance criteria for fix

Heart icon changes immediately when clicked
unfilled → filled when added to wishlist
filled → unfilled when removed from wishlist
no page refresh required for state change
change visible within 100ms
state persists after page reload
no duplicate items added
no javascript errors in console
works on all browsers and devices

Testing strategy for fix verification

Test	Purpose
Click and observe	Verify heart fills immediately
Refresh page	Verify state persists (in db)
Remove and re-add	Verify toggle works both directions
Network throttle	Verify works on slow networks
Multiple clicks	Verify no duplicate additions

Additional notes

- **Related products:** issue affects wishlist for related products only
- **Main product:** main product wishlist likely works (needs verification)
- **User impact:** medium - feature works but ux is poor
- **Workaround:** users can navigate to wishlist to confirm addition

Issue 04: add to cart confirmation message not displaying

Bug report summary

Field	Value
Bug id	Bug-004
Title	Add to cart confirmation message not showing for related products
Severity	Medium
Priority	P2
Status	Open
Component	Related products ui / toast notifications
Environment	Staging
Date discovered	January 18, 2026
Assigned to	Frontend development lead

Description

When adding a related product to cart by clicking "add to cart" button on a product card, no confirmation message/toast notification is displayed to the user. The product is successfully added to cart (verified by checking cart), but there is no visual feedback confirming the action. User must navigate to cart page to verify product was added.

Steps to reproduce

1. Navigate to prod-001 (brown leather wallet)
2. Scroll to related products section
3. Click "add to cart" button on rp-001 (black leather bifold)
4. **Expected:** toast/notification message appears: "added black leather bifold to cart"

5. **Actual:** no message appears; page looks unchanged
6. Product may be added to cart (backend works)
7. Navigate to shopping cart page
8. **Actual:** rp-001 is in cart with quantity 1

Comparison test:

- Add main product (prod-001) to cart: confirmation message shows
- Add related product (rp-001) to cart: confirmation message does not show

Expected behavior

- Clicking "add to cart" should show success confirmation
- Toast notification appears with message: "added [product name] to cart"
- Toast appears for 3-5 seconds then disappears
- Toast color: green (success)
- Alternative: brief confirmation message, "+ added to cart" indicator, or cart count increases
- Visual feedback should be clear and obvious to user

Actual behavior

- Clicking "add to cart" shows no feedback
- Page appears unchanged
- No message, toast, or notification displayed
- Cart count may increase silently (hard to notice)
- User unsure if click registered or succeeded
- Must navigate to cart page to verify addition

Impact

Severity: medium

- Feature works correctly (product added)

- Ux is poor - lacks user feedback
- Users uncertain about action success
- May lead to double-clicking "add to cart"
- Affects user satisfaction but not functionality
- Should fix before launch for better ux

Acceptance criteria for fix

Confirmation message displays when adding related product to cart

message text: "added [product name] to cart"

message visible for 3-5 seconds

toast styled consistently with main product (same color/design)

toast appears in same location as main product toast

can close toast manually (x button)

cart count updates (if visible)

works on all devices and browsers

Testing strategy for fix verification

Test	Purpose
Add to cart	Verify toast shows
Multiple adds	Verify multiple toasts show
Different products	Verify product name correct
Slow network	Verify works on 3g/4g
Mobile	Verify toast visible on small screens

Additional notes

- **Related issue:** bug-003 has similar pattern (missing ui feedback)
- **Main product works:** confirms toast capability exists, just not used for related

- **Low severity:** feature works, just lacks polish
- **Easy fix:** likely one-line fix (add toast trigger)

Issue 05: api returning inconsistent data for related products

Bug report summary

Field	Value
Bug id	Bug-005
Title	Api returns different related products on repeated requests
Severity	Medium
Priority	P2
Status	Open
Component	Related products api / backend data layer
Environment	Staging
Date discovered	January 18, 2026
Assigned to	Backend development lead

Description

When requesting related products for the same product (prod-001) multiple times, the api returns different sets of products or different orderings. The 6 products may change between requests, or the ranking order may differ, suggesting either:

1. Query results not deterministic (randomization or timestamp-based)
2. Cache inconsistency (serving stale data from different cache nodes)
3. Multiple backend instances with different data
4. Race condition in data fetching

This makes it difficult to predict which products will display and breaks test automation.

Steps to reproduce

1. Open browser devtools → network tab
2. Navigate to prod-001 (brown leather wallet)
3. Wait for related products api response: /api/related-products/prod-001
4. **Response 1:** returns [rp-001, rp-002, rp-003, rp-004, rp-005, rp-006]
5. Reload page (ctrl+r)
6. **Response 2:** returns [rp-002, rp-001, rp-006, rp-003, rp-004, rp-005] (different order!)
7. Wait 1 minute, then reload
8. **Response 3:** returns [rp-001, rp-003, rp-005, rp-002, rp-006, rp-004] (different set?)

Test automation consequence:

- Test tc-fs-001 expects specific products in specific order
- Fails on 3rd run because product set is different
- Cannot rely on fixed product list for testing

Expected behavior

- Api should return same set of 6 products for prod-001
- Products should be in consistent order (by sales ranking)
- If ranking changes, should be because actual sales data changed
- Multiple requests within same cache window should return identical data
- Results deterministic and predictable

Actual behavior

- Api returns different products on different requests
- Product order varies between requests
- Sometimes returns only 5 products instead of 6
- Inconsistency suggests caching or data source issue
- Makes feature unpredictable and difficult to test

Impact

Severity: medium

- Feature works but is unpredictable
- Test automation becomes flaky (intermittent failures)
- Affects test reliability and developer confidence
- Makes debugging difficult (inconsistent reproduction)
- Not critical to user experience (works, just unpredictable)
- Should fix for test stability

Acceptance criteria for fix

Api returns consistent set of 6 products for prod-001
 product order consistent across requests (sorted by ranking)
 multiple requests within cache window return identical data
 changes only when actual sales data/rankings change
 no flaky tests due to data inconsistency
 deterministic results (same input = same output)
 query includes explicit order by clause

Testing strategy for fix verification

Test	Purpose
10 sequential requests	Verify identical response
Concurrent requests	Verify no race conditions

After cache expiry	Verify refresh deterministic
Different products	Verify not affecting each other
Check logs	Verify cache hits/misses correct

Additional notes

- **Test environment issue:** may only occur in staging with shared data
- **Production risk:** could be same issue in production
- **Workaround:** pin test data to specific products in test setup
- **Data quality:** verify test data doesn't have duplicate sales records

Summary of all issues

By severity

Severity	Count	Issues
Critical	1	Bug-001: mobile not loading
High	2	Bug-002: price range filtering, bug-003: wishlist not updating
Medium	2	Bug-004: no cart confirmation, bug-005: inconsistent api data

Report prepared by: qa team

date: january 18, 2026

status: ready for dev team review

End of bug reports

