Usability and Compatibility Testing PDF

Usability testing checks how easily real users complete tasks, using task scenarios, observation and heuristics. Compatibility testing checks that the product works across browsers, operating systems, devices and screen sizes. Build a matrix from real usage data, test the most important combinations first and record findings with screenshots.

Key takeaways

- Usability testing watches real users attempt realistic tasks.
- Nielsen-style heuristics offer a quick expert review method.
- A compatibility matrix lists priority browsers, devices and versions.
- Use analytics to decide which combinations matter most.
- Responsive layout, fonts and input methods cause many cross-device issues.

What is usability and compatibility testing?

Usability testing evaluates how easy, efficient and satisfying a product is to use, and compatibility testing verifies that it behaves correctly across different browsers, devices, operating systems and configurations.

What is usability testing?

Usability testing asks whether people can reach their goals without confusion. A product can pass every functional test and still frustrate users with unclear labels, hidden actions or long forms. Usability testing finds these problems by watching how people behave, not by checking requirements alone.

How do I run a simple usability session?

1. Define tasks, such as "find a blue shirt in size M and add it to the cart".
2. Recruit a handful of participants who resemble real users.
3. Ask them to think aloud while you observe without helping.
4. Note where they hesitate, make mistakes or give up.
5. Ask short follow-up questions.
6. Group findings by severity and share with the team. Even five sessions often reveal repeated problems.

What can I check without users?

A heuristic review uses established principles: visibility of system status, match with the real world, user control and freedom, consistency, error prevention, recognition rather than recall, flexibility, minimal design, helpful error messages and clear help. Walk through key flows and list violations. It is quick but is no replacement for watching users.

What measures indicate usability?

Measure	Question
Task success	Did the user finish the task?
Time on task	How long did it take?
Errors	How many wrong turns occurred?
Satisfaction	How did the user rate the experience?
Learnability	Can a new user manage without training?
Compare results before and after changes.	

What is compatibility testing?

Users arrive with many browsers, devices and settings. Compatibility testing checks that layout, features and performance remain acceptable across them. It covers browser compatibility, operating system compatibility, device and screen size, network speed, and backward compatibility with older versions of the product or data formats.

How do I build a compatibility matrix?

Use analytics to list the browsers, operating systems and devices your visitors use. Mark the top combinations as priority one, test them fully, and give the rest smoke tests. Include the latest and previous major browser versions, a small phone, a large phone, a tablet and a desktop. Review the matrix every quarter, since usage changes.

Which issues appear often?

Layout that breaks at certain widths, fonts that differ, date pickers that behave differently, touch versus mouse events, missing support for a newer browser feature, differences in file upload or download behaviour, and slow performance on older devices. Test orientation changes and zoom as well.

How can tools help?

Browser developer tools emulate devices and throttle the network. Cloud device farms offer many real devices. Automation tools such as Playwright can run the same test on several browsers.

Emulators are quick but real devices reveal touch, performance and hardware differences, so use both wisely.

How do I report findings?

Include the browser or device, version, operating system, steps, screenshot and expected look. State whether the issue affects a priority combination. For usability findings, describe what the user tried and what happened, with a quote when useful, and suggest options without prescribing a design.

What should I avoid?

Avoid testing only on your own laptop, treating a resized window as a real phone, asking leading questions in usability sessions and arguing with participants. Observe, listen and learn.

How do I test responsive design effectively?

Resize the browser through common breakpoints, test portrait and landscape on a phone, check touch targets that are large enough, verify that menus collapse and expand correctly and confirm that text remains readable without horizontal scrolling. Test with slow network throttling to see how images and fonts load. Responsive bugs often appear between breakpoints, so also test odd widths rather than only standard device sizes.

How can I combine these tests with automation?

Automate the repeatable part: run the same smoke flow on several browsers and viewport sizes with a tool such as Playwright, and capture screenshots for review. Keep usability observation manual, since automation cannot judge confusion or satisfaction. Together, the automated matrix catches layout and functional differences quickly, while human sessions reveal whether people can actually succeed with the product.

What is a simple checklist to start?

Pick three user tasks, test them on two browsers and two screen sizes, record screenshots and note anything confusing. Share the findings in a short table with severity and a suggested next step so the team can act quickly.

How often should the matrix be reviewed?

Review it every quarter using analytics, and after any major browser or device release that your users adopt quickly.

Frequently asked questions

What is usability testing?

It evaluates how easily real users can complete tasks with a product, usually by observing people use it while thinking aloud.

What is compatibility testing?

It checks that the product works correctly across browsers, operating systems, devices and screen sizes that users actually use.

How many users do I need for a usability test?

Even a handful of participants often reveals repeated problems, though more users help for different groups and larger products.

How do I choose which browsers to test?

Use analytics from your own product to find the most-used combinations, test those fully and run lighter checks on the rest.

Are emulators enough for device testing?

They are quick and useful, but real devices show touch, performance and hardware differences, so use both when risk is high.

Sources and further reading

- ISTQB glossary and resources
- ISTQB glossary
- MDN web docs

About this PDF

These PDFs are study material for learning and revision. They are not a certificate, a degree or a promise of a job, and technology changes, so check official documentation for the latest details. Spot an error? Write to info@uncodemy.com and the trainer team will review it.

More free resources and courses: uncodemy.com | info@uncodemy.com | +91 98183 66550