

Open in app ↗

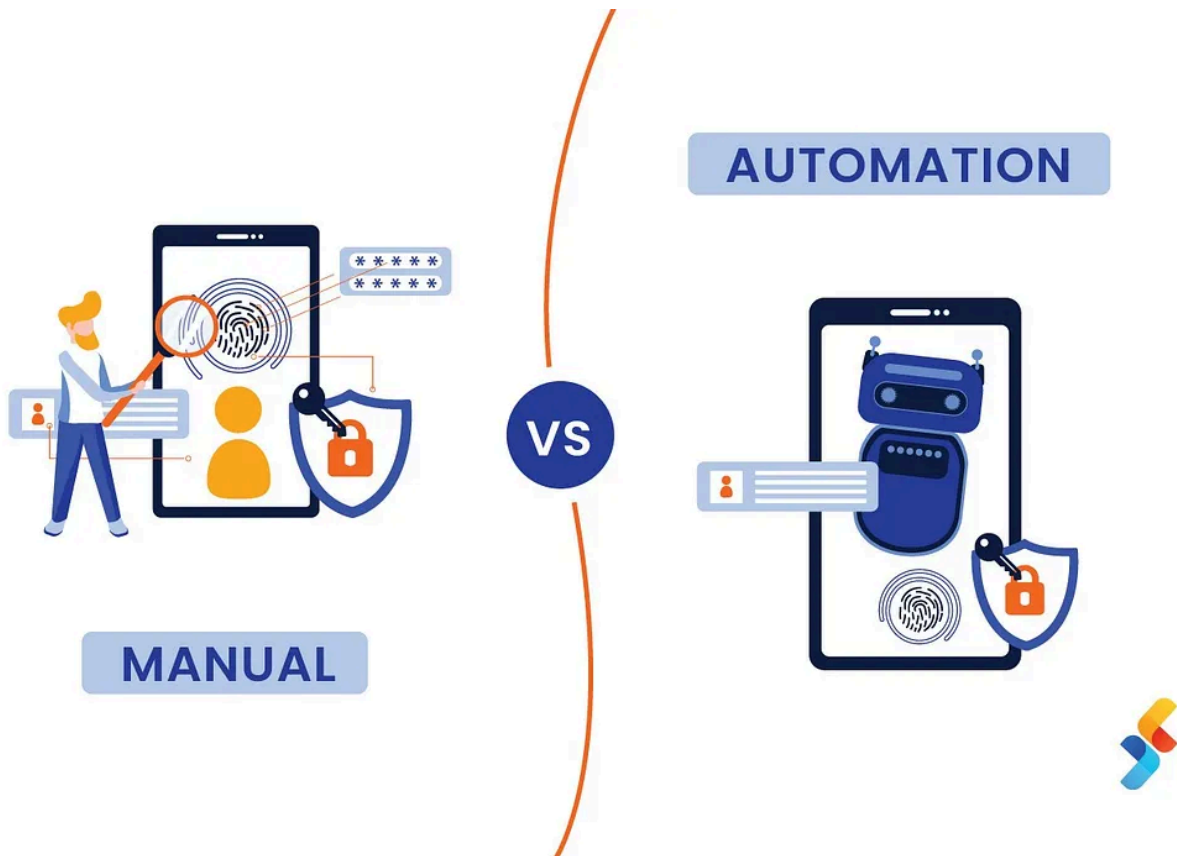
≡ Medium

🔍 Search

✍ Write



Rashmi Ruwanthika Athukorala · 4 min read · Just now



## Manual vs. Automated Testing: Which Should You Choose for Your Project?

Choosing the right testing approach is crucial for delivering high quality software. Should you rely on people performing tests step-by-step (manual

testing), or use tools that execute pre-written scripts (**automated testing**)?

Both methods have their strengths and weaknesses. The best choice often depends on your **project size, complexity, budget, and timeline**. Let's break down each method to help you make an informed decision.

## What is Manual Testing?

Manual testing is when a **quality assurance (QA) tester** sits in front of the application and acts as an end-user. They click, tap, and interact with the software to ensure it works correctly according to the requirements.

### *Pros of Manual Testing:*

- 1. Human Intuition & Exploratory Testing:** Humans can notice subtle issues like visual misalignments, awkward workflows, or confusing instructions. Exploratory testing allows testers to think creatively and find bugs that scripted tests may miss.
- 2. Cost-Effective for Small Projects:** For apps that change frequently or are small, setting up an automation framework can be too expensive and time-consuming. Manual testing is simpler to start.
- 3. User Experience (UX) Evaluation:** Manual testing helps evaluate the feel and usability of the app from a real user's perspective.

### *Cons of Manual Testing:*

- 1. Time-Consuming:** Repeating the same tests (e.g., full regression testing) can take a lot of human hours.

2. **Prone to Human Error:** Testers can get tired, distracted, or miss steps, which may lead to overlooked bugs.
3. **Not Suitable for Performance Testing:** Manually simulating hundreds or thousands of users at the same time is impossible.

## What is Automated Testing?

Automated testing uses **software tools** to run pre-designed test scripts. These scripts check specific functions, report results automatically, and compare them against expected outcomes.

### *Pros of Automated Testing:*

1. **Speed and Efficiency:** Automated tests run much faster than humans, giving quick feedback.
2. **Consistency & Reliability:** Scripts perform the same steps every time, eliminating human error.
3. **Excellent for Regression Testing:** Automation is perfect for repeatedly checking that new code hasn't broken existing features.
4. **Scalability:** You can run thousands of tests simultaneously across multiple browsers, devices, and operating systems.

### *Cons of Automated Testing:*

1. **High Initial Setup Cost:** Developing and maintaining an automation framework requires specialized skills and upfront time.
2. **Requires Regular Maintenance:** If the application UI changes, scripts need updates, which is ongoing work.

3. **Limited for Usability Testing:** Automation can check if a button works, but it cannot judge whether the app feels intuitive or easy to use.

## **Factors to Consider When Choosing**

When deciding between manual and automated testing, consider:

- **Project size & duration:** Small, short-lived projects often benefit more from manual testing.
- **Budget & resources:** Automation requires initial investment but saves money in the long run for large projects.
- **Frequency of testing:** Tests that run repeatedly are perfect for automation.
- **Complexity & stability:** Stable features that rarely change are good candidates for automation.
- **Focus on user experience:** Anything requiring human judgment should be manually tested.

## **Manual, Automated, or Both?**

The best approach is often a **hybrid strategy** that combines both methods. Here's a simple guideline:

# Access to all of it.

Become a Medium member  
for unlimited reading.



Upgrade now

## Scenario Best Choice Why?

New features & usability Manual Humans are best for spotting UX issues and performing ad-hoc tests. Repetitive, stable tests Automated Perfect for regression testing and nightly builds. Performance/load testing Automated Required to simulate hundreds or thousands of users. Small, short-lived projects Manual Automation setup may not justify cost/time. Cross-browser/device checks Automated Scripts run the same test across multiple devices quickly.

In most professional environments, **automation handles repetitive and large scale testing**, while **manual testers focus on usability, exploratory testing, and complex scenarios**.

## ✂ Popular Tools for Both Testing Types

- **Manual Testing:** JIRA (for bug tracking), TestRail, Zephyr, Bugzilla
- **Automated Testing:** Selenium, Cypress, Playwright, Appium, Postman (for APIs)
- **Performance Testing:** JMeter, LoadRunner, Locust

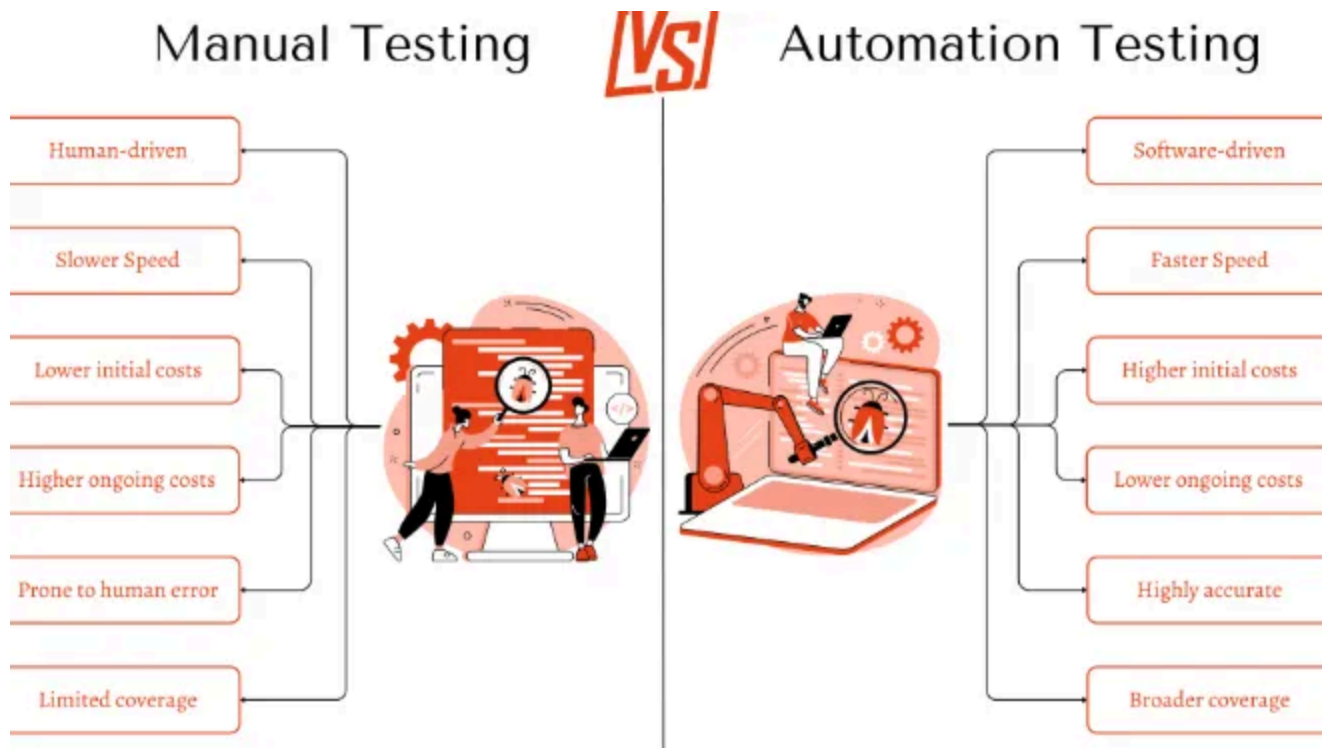
## **Tips for Using Both Effectively**

1. **Automate where it saves time:** Focus on regression tests, repetitive tasks, or cross-browser/device checks.
2. **Manually test new features first:** Humans are best at discovering tricky bugs that automation might miss.
3. **Keep scripts maintainable:** Regularly review and update automated tests to match app changes.
4. **Track and report bugs clearly:** Whether manual or automated, documentation is key.
5. **Balance cost and time:** Use manual testing for small, frequent changes; automate large, stable, or repeated tests.

## **Conclusion**

There's no one size fits all answer. A **hybrid approach** gives the best results: the **speed and reliability of automation** combined with the **intuition and flexibility of manual testing** ensures high-quality software delivered efficiently.

By understanding the strengths and weaknesses of each method, your team can make informed decisions and **deliver a superior product faster**.



**Written by Rashmi Ruwanthika Athukorala**

[Edit profile](#)

0 followers · 1 following

BSc Hons Computer Science and Technology | Undergraduate at Uva Wellassa University, Sri Lanka | QA Enthusiast | Software Tester

No responses yet



Rashmi Ruwanthika Athukorala