When to Automate Your Testing (and When Not To)

Joe Fernandes (Oracle)
Alex Di Fonzo (Synchronoss Technologies)
Three Myths of Automated Testing

1. Automated testing always results in improved software quality

2. Every application development project or testing team can use automated testing tools

3. Automated testing is an all or nothing proposition
Three **Realities** of Automated Testing

1. Automated testing requires a higher initial investment but can yield a higher ROI

2. Skills and training are required to be successful with any automated testing tool

3. Every company doing automated testing still does some amount of manual testing
Testing Fact

- Industry surveys indicate that 75% of all functional testing is still done manually
Question #1:

- Why do most companies still rely so heavily on Manual Testing?
Why Manual Testing?

• **Time**: Testing teams may not have time to investigate alternatives to manual testing, learn how to use tools, and/or build & maintain scripts

• **Application Complexity**: Certain applications may be too complex and not suitable for automated testing

• **Skillset**: Some testers (business analysts, etc.) may lack skills needed to be successful with test automation tools

• **Cost**: Organizations may not own automated testing tools and may lack the budget to invest in tools

• **Job Security**: Testers / QA organizations may be comfortable and experienced with manual testing and feel threatened by automation

• **Awareness**: Organizations may lack the awareness that viable automated testing alternatives exist
Question #2:

- When is Manual Testing a better alternative than Automated Testing?
When is Manual Testing a Good Idea?

- **Subjective Validation**: For application functions that must be validated subjectively by humans such as usability or look-and-feel, manual testing may be the only option.

- **New/Changing Functionality**: For new application functions that are still being developed and evolving / changing frequently, creating automated scripts may be a waste of time.

- **Strategic Development**: For strategic application functions that you want testers to pay specific attention to, hands-on manual testing may be a better alternative.

- **Complex Functionality**: For application functions that are extremely complex, test automation may be a major challenge (time & cost investment outweighs the benefit).
Question #3:

- When is Automated Testing a good alternative to Manual Testing?
When is Automated Testing a Good Idea?

- **Regression Testing**: For re-testing preexisting application functions that are being carried forward to new versions (usually the majority, unless app is brand new)

- **Smoke Testing**: For getting a quick high-level assessment on the quality of a build and making go / no-go decision on deeper testing

- **Static & Repetitive Tests**: For automating testing tasks that are repetitive and relatively unchanging from one test cycle to the next

- **Data Driven Testing**: For testing application functions where the same functions needs to be validated with lots of different inputs & large data sets (i.e. login, search)

- **Load & Performance Testing**: No viable manual alternative exists
Oracle Application Quality Management: A Lifecycle Approach to Quality

- **Design Test Plans Based on Application Requirements**
- **Load Testing for Web Applications**
- **Develop Manual Test Cases and Automated Test Scripts**
- **Functional Testing for Web Applications**
- **Test Manager for Web Applications**
- **Develop Manual Test Cases and Automated Test Scripts**
- **Execute Load Tests and Tune Application Performance**
- **Execute Functional Tests to Validate Application Requirements**
Test Manager for Web Applications

Test Process Management

- Manage test process from centralized Web-based console
- Define test requirements
- Develop manual & automated test cases
- Document and track defects
- Create reports
Functional Testing for Web Applications

Functional & Regression Testing

- Automate transactions for Web applications & Web Services
- Execute rigorous functional test cases
- Create automated regression test suites
- Identify and report on functional application failures
- Reuse functional test scripts for load testing and 24x7 monitoring
Load Testing for Web Applications

Load and Performance Testing and Tuning

- Create realistic load test scenarios that simulate end-user behavior
- Scale to thousands of concurrent users
- Perform functional content validation under load
- Monitor server-side performance & correlate to end-user response times
- Isolate and resolve performance bottlenecks
Synchronoss Technologies
Alex’s Bio

- Quality Assurance Manager for Synchronoss Technologies, Inc. a 15 year veteran of the software testing industry, extensive experience with sales force automation and transaction management systems. In both my current and most recent prior positions, I started the Quality Assurance Departments and built them from the ground up.
Agenda

• Synchronoss Overview
  • Company Overview
  • Our QA processes & cycles
  • When we evaluate automation vs. manual testing
• Automation: What works and what doesn’t
• Automation with e-Tester
• Automation ROI (cost versus benefits)
• Summary
Synchronoss Overview

• Synchronoss Technologies is the premier provider of on-demand transaction management software to Tier One communications service providers. The software platforms automate, synchronize and simplify electronic service creation and management of advanced wireline, wireless and IP services across existing networks.

• Headquartered in Bridgewater NJ, with offices in Bethlehem PA, Herndon VA and Bellevue WA
Our QA processes & cycles

• Our Applications:
  • Used both external and internal
  • Short description, transaction management
  • Functionality changes often

• Our Testing Processes:
  • Between Agile and Extreme
  • Test cycles depend on project, client and application
    • Short 6 week SDLC (Requirements, Development, Test)
    • Every other month, with 3 weeks for testing
When do we evaluate automation vs. manual testing


- Evaluating functions for automation is the responsibility of the entire project team and needs to happen during the entire SDLC.
  - Includes automated unit tests, nightly builds and scripts with or without Functional Testing for Web Applications (formerly eTester) that can validate build files, DB, configurations and GUI
When do we evaluate automation vs. manual testing

- **Reviewing Requirements**
  - Can this function be automated?
  - What, if anything, does development need to do?
  - When writing test cases
  - Can this function be automated?
    - If yes, ensure test case is written to ease scripting (step by step)

- **While Testing**
  - Ensure the test case is clear and precise
  - Are the results predictable?
  - Does the test have to be run many times to get the desired result?
When do we evaluate automation vs. manual testing

• What to think about
  • Positives
    • Will productivity increase?
    • Will test coverage increase?
    • Will test accuracy increase?
    • Is this a large data input?
    • Is this GUI intensive?
  • Negatives
    • Requires human intervention
    • Requires 3rd party system
    • Test has unpredictable results
    • How often will this function change?
Automation: What works and what doesn’t

• Lessons learned
  • Automate functions that are 80% stable and unchanging
  • Work with development on unique names for controls and fields
  • Don’t overlook bulk data loads for the support of regression testing
  • Remember to include script maintenance in estimates
  • Make scripts as generic as possible
  • Use control file for URL, User IDs, and Passwords

• Upper management will always feel that regression should be 100% automated. This perception must be managed while setting proper expectations of what can be accomplished.
Automation with Functional Testing for Web Applications (formerly e-Tester)

• Key needs
  • All controls and fields have unique names
  • Test harness
  • Stable environment controlled only by QA
  • Established application
  • Don’t forget about data loads – we increased productivity of manual regression testing by 28% by automation the loading of data used for regression testing.
  • Script maintenance must be included in your estimates for testing going forward. Functions will change and scripts will need to be updated.
Automation with Functional Testing for Web Applications (formerly e-Tester)

- Script management
  - Each project has a dedicated eTester desktop
  - Those that work automation, work with the project team to ensure they are working on areas of the application that are stable, somewhat unchanging and will improve productivity.
  - Come up with, and stick to a naming convention for scripts
  - Smoke testing can be done quickly and more reliably
  - Can have scripts run overnight, review results in the morning and advise development of issues faster
Automation ROI

- Items to think about when calculating your ROI
  - Tool Investment
  - Learning Curve
    - For Tool
    - For Application
  - Employee Job Satisfaction
  - You will Get
    - Overnight testing
    - Testing reports emailed to you
    - More test coverage in same or less time
    - More repeatable tests
    - Faster test coverage
  - You will not get
    - All this immediately
  - Expectations and implementation must be well managed
Summary

• Do
  • Use this as a guideline and change it to fit your process
  • Work to control expectations of automation
  • Use the QAZone (now in OTN) for Tips, Trick and Information

• Don’t
  • Try and automate without support from development
  • Over estimate what can be automated
  • Let someone else set the expectations of automation
Thank you!
For More Information

search.oracle.com

or

oracle.com