Test Automation: A Project Management Perspective

PNSQC 2013

Amith Pulla
Intel Corp.
Twitter: @pamith

Introduction

- QA Manager at Intel
- Involved in software testing strategies and processes
- Works on sales and marketing projects/applications
- Worked with XP, Scrum and Scaled Agile Framework
- Got PMP, CSTE in 2006 and CSM in 2012
Test Automation

- Better quality, Faster delivery, Lower risk
- Not all software development projects use test automation (Agile included)
- Value of test automation not visible to project PMs
- Most Quality experts agree that some level of test automation is essential
- Lack of funding and priority are some of the biggest obstacles

Challenges in Adoption

- Getting the project manager/sponsor to invest in test automation
- Costs related to resources with automation skills, automation tools, test machines, servers and training
- Ongoing costs of maintaining the automation framework and scripts until application EOL
- A senior program manager once said, “Test automation is great thing to have, but not a great thing to invest in”
- Measuring value or ROI (Return on Investment) is difficult
- Showing the value of test automation in the terms that project managers can relate to
**Project Management Constraints**

- **Scope:** Product Scope
- **Cost:** Resources + dollars
- **Schedule:** Amount of time to complete
- **Quality:** Product defects, Stability
- **Risk:** Project failure, Not meeting user expectations

Direct correlation between project variables and investing in test automation is missing

Usually reluctant to invest in test automation

Translating investments and ongoing costs into positive impact on project attributes

Individual metric for projects’ attributes to shows tangible and measurable value

---

**Project Management Constraints**

- **Scope:** No impact
- **Cost:** Investing in test automation can have an initial spike in project costs, ROI should move to positive territory at some point
- **Schedule:** Can clearly show positive impact, duration for regression test cycles will shrink
- **Quality:** Allows multiple regression cycles for same project schedule, team to improve overall quality and increase the users’ confidence
- **Risk:** More test coverage means better understanding of application behaviors and defects
Test Automation and Quality

- **Regression Testing**
  - Ensures that previously working functionality is not altered or impacted
  - All or a subset of existing test cases that need to be run depending on the application or platform
  - Running them manually can take days or weeks to complete

- **Test-Fix-Test Cycles**
  - Better testing or test coverage is not enough
  - Needs series of Test-Fix-Test cycles until defect counts are reduced to below thresholds
  - Run several regression cycles to ensure the defect numbers are below acceptable levels

Metrics and Measures

- **Test Automation Coverage**
  Number of test cases automated Vs. total number of automatable test cases

\[ TAC = \frac{\text{No. of Automated Test Cases}}{\text{No. of Test Cases that can be automated}} \times 100 \]
**Time Gain per Regressing Cycle**

- Time it takes to run the regression cycle manually Vs. running it using automation tool
- The time gain/effort reduced per cycle means cost saving
- Need fewer testers as manual execution declines
- Initial cost of automation is high
- There is certain cost saving per regression cycle, it will add up to the initial cost of automation eventually
- ROI on test automation efforts moves into positive territory

---

**Defects Found per Regression Cycle**

- Tells the team about the quality of the application
- Shows effectiveness of the test automation
- Defects found per regression cycle are a good indicator of quality
- As defects counts trend downwards over a series of regression cycles, it builds the confidence of the stakeholders
Cost Savings (ROI) and Challenges

- ROI on test automation can be calculated as
  - \( \text{ROI} = \frac{\text{cost of manual testing} - \text{cost of automation}}{\text{cost of automation}} \)
- Not many quality experts agree that this is the right approach
- Does not consider ongoing costs for automation, mainly related to maintenance
- Changes throughout product life impact test automation scripts
- The real return on investment for automation is based on different types of automation done by the team and the value it adds to the overall testing effort (Michael Kelly, 2004)
- Tangible and Intangible benefits
  - Benefits related to hours saved and time gained
  - Intangible benefits can include faster feedback from users, finding defects sooner in the development cycle

Test Automation with Continuous Integration (CI)

- Continuous integration (CI) is a software development practice of merging all developer workspaces into a common codebase once or several times a day
- Originally intended for automated unit tests written through the test-driven development, running automated functional test cases has become a standard now
- Most agile teams today have some level of continuous integration (CI) built in
- Continuous integration is not just source control integration and automated builds, but complete process including
  - Integrating code to baseline
  - Automated compilation and build
  - Running unit tests and automated regression tests (if available)
  - Most importantly if the tests fail, all the activities that follow to make the build clean and stable again
Cost Benefit Analysis on Tools

- Initial investment, licensing and ongoing costs for the automation tool
- Flexibility and scalability of the automation tool
- Initial framework setup necessary for the automation tool
- Skillset and training required to use the tool
- Compatibility with application under test

Automation Tools

- Open Source vs. Vendor Tools
  - **Open Source:**
    - Upfront work to create a framework
    - More adoption, better body of knowledge
  - **Vendor Tools**
    - Licensing costs
    - User friendly and designed to hit the ground running
    - Vendors provide support and training if needed
Test Automation in Agile

- Becoming a foundational practice in agile development
- Test automation as an engineering practice
- Show value in terms of Velocity and Quality
- Helps deliver better quality and business value faster
- Needs to be used daily within the CI model
  - Value is only realized when it’s run every day as part of CI, it allows the teams to find defects faster when there is enough time to fix them
  - Helps meet velocity, release cadence and quality expectations

Continuous Deployment with Test Automation

- Takes CI and automation to new level of agility and delivery
- Transform software delivery
  - Deliver features and fixes as they become ready without having to wait for set release date
- Automated Deployments/Infrastructure Automation: Key aspect of DevOps (development and operations team working together)
- Agile teams can push working features, patches to production (to users) quickly and efficiently
- Industry Examples
  - Facebook, Etsy and LinkedIn
Conclusion

- Project managers or stakeholders sometimes may not see the value of test automation and benefits it offers
- Understand impediments to investing in and prioritizing test automation efforts
- Influence the decision to invest in automation by showing the value in terms of positive impact to key project management attributes
- Show the value of test automation in terms of positive effect on project’s schedule, cost and quality (aspects that project managers and stakeholders most care about)

References

- Michael Kelly. 2004. The ROI of Test Automation
- Rayn Tate. 2013. The Software Revolution Behind LinkedIn’s Gushing Profits
- Ross Snyder. 2013. Continuous Deployment at Etsy: A Tale of Two Approaches