

Test Brain Introduction



1.

SOFTWARE TESTING

Current State

- » In 2010 average deployment was quarterly.
- » Now average deployment is monthly.
- » By 2020 average deployment will be weekly.

30% of software development cost is spent on testing,
increasing to 40% by 2020

<50%

The percentage of tests that are automated

11%

The percentage of UI Tests that are automated

3%

The percentage of test automation failures that are caused by defects

19%

Of teams that succeed with test automation on their first attempt

34%

Of teams have tried to implement automation multiple times and failed

56%

Of teams can't use automation for a significant part of their testing

44%

Of teams that practise test management

18%

Of applications meet user expectations

4%

Of apps are used after a month

44%

Of teams don't have enough time for core testing

21%

Of testers are happy with the distribution of their automated tests

8%

Of testers have an understanding of the automation coverage
(including unit tests)

40%

The amount of time testers spend triaging automated tests

2.

SOFTWARE TESTING

Challenges

1. Application changes frequently
2. Coping with timeframes
3. Managing test cases
4. Don't have the right tools
5. Delivery process doesn't support automation

1. Test Stability
2. Object Identification and Management
3. Maintenance
4. Validating Test Coverage

(97%) of test bugs found by automation are false alarms

- » Semantic Bugs (25%)
- » Flaky Tests (21%)
- » Environmental Bugs (18%)
- » Inappropriate Handling of Resources (14%)
- » Obsolete Tests (14%)
- » False positives (3%)

What needs to be improved in automation

- » Maintenance
- » Run tests faster
- » Smarter more intelligent tests
- » Less false negatives
- » Better feedback and reporting
- » Improved traceability and coverage

3.

TEST AUTOMATION

Common Issues

CHALLENGES

15

Automated tests especially UI tests take a long period of time to run. Even a large suite of unit tests can take an inordinate amount of time to execute, if not parallelized.



There is a lack of tooling allowing the comparison of automated test results across different executions.



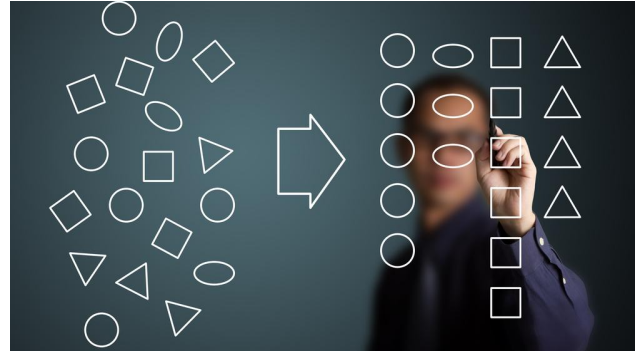
Frequently the manual and automated tests are treated as separate entities, with the coverage of the automated tests not influencing the manual test execution.



Tests are not executed in a prioritized order. We do not take into account the likelihood of the test finding a defect or the importance of the area under test.



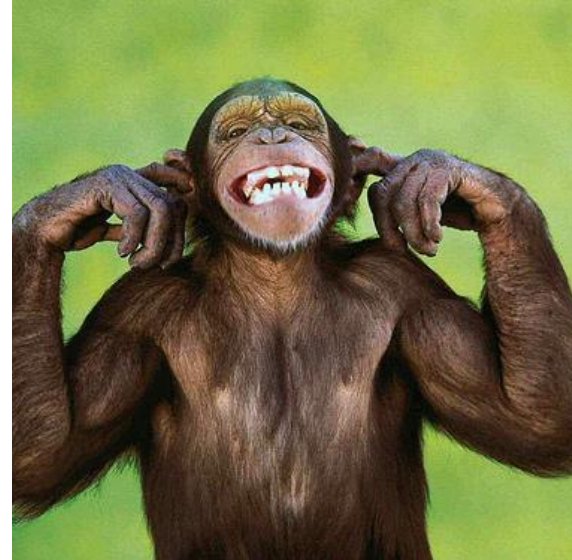
After each automation run tests need to be triaged.
With real results separated from flaky and tests moved into different runs depending on their status.



Automation failures are frequent and cause emails to be sent to the entire team rather than just the responsible parties.



Due to automated tests frequently failing, teams often lose faith in the automated tests leading to “boy who cried wolf” syndrome and automation providing no value.



4.

TEST BRAIN

Functionality

FUNCTIONALITY

Carefully constructed dashboard displaying information on: manual tests, automated tests, trends, defects, changes in test runs and allowing comparison of test results.

BENEFIT

Allows for better feedback and reporting for both managers and team members. The automated coverage can influence the manual testing coverage.

FUNCTIONALITY

Using a specialised algorithm and AI we can predict the areas which are most likely to have defects. We can also prioritize the order in which tests should be executed.

BENEFIT

Testers find defects earlier in the testing cycle, which reduces the cost of resolving the defect. Additionally, test suite execution time can be decreased.

FUNCTIONALITY

When tests fail during your test run, a defect is automatically created. Defects will also automatically be classified and prioritized. Additionally, defects will be closed when the issue is resolved.

BENEFIT

This reduces the time testers spend both creating defects after the automation run and also closing these defects.

FUNCTIONALITY

Tests will be categorized automatically as Flaky. These tests will be re-added to the prioritized test list. All Flaky tests will be added to their own test run in the results, preventing them from affecting the results of the other test runs.

BENEFIT

Test results are more reliable. Real issues are spotted faster and fake issues are easily dealt with. This helps further reduce the time spent triaging tests and increases the consistency of the automation results.

FUNCTIONALITY

See your manual tests and automated tests side by side. Understand what is covered by automation and utilize this during your manual testing cycles.

BENEFIT

This can help reduce the overlap in test coverage and as a consequence of this helps discover defects earlier.

FUNCTIONALITY

Understand which testing activities and which tests are valuable. Discover where defects are escaping and which areas have low test coverage and use this knowledge to determine your investment and resource allocation.

BENEFIT

This can help optimise the utilization of your testing resources, reducing wasted effort and increasing the time spent on valuable activities.

FUNCTIONALITY

Previously testers have “guessed” which tests should be automated. We provide a list of tests prioritized based on the order in which they should be automated. Now you know you are automating the most valuable tests.

BENEFIT

This increases the value of the automated tests being created, which in turn increases the return on investment from your automated testing efforts.

FUNCTIONALITY

As soon as a new defect is found via test automation the owner of the commit/build and the owner of the test are notified.

BENEFIT

Faster feedback from the automated tests and less spam to team members who are not responsible for investigating the defect.

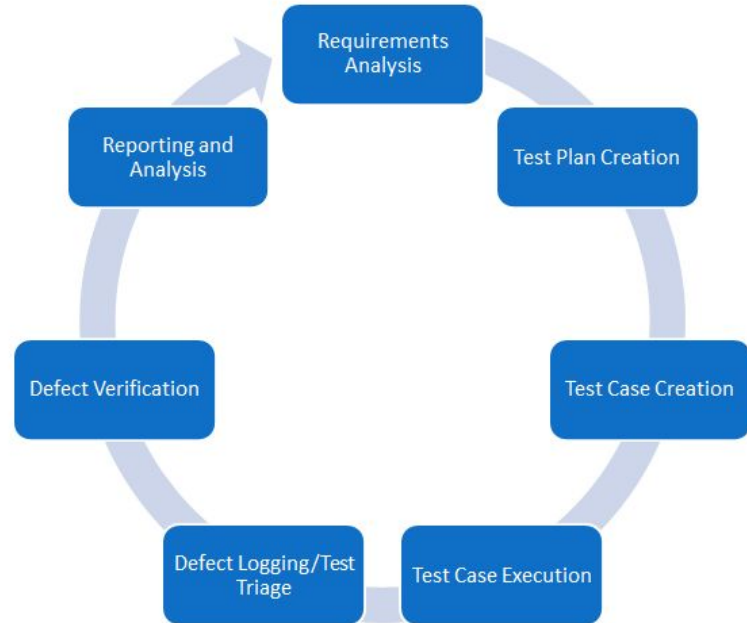
- » Reduced maintenance of automated tests
- » Faster test runs and feedback
- » Smarter more intelligent tests
- » Less false negatives
- » Easier test management
- » Improved traceability and coverage information
- » Increased test stability
- » Better feedback and reporting
- » Allows more targeted testing

5.

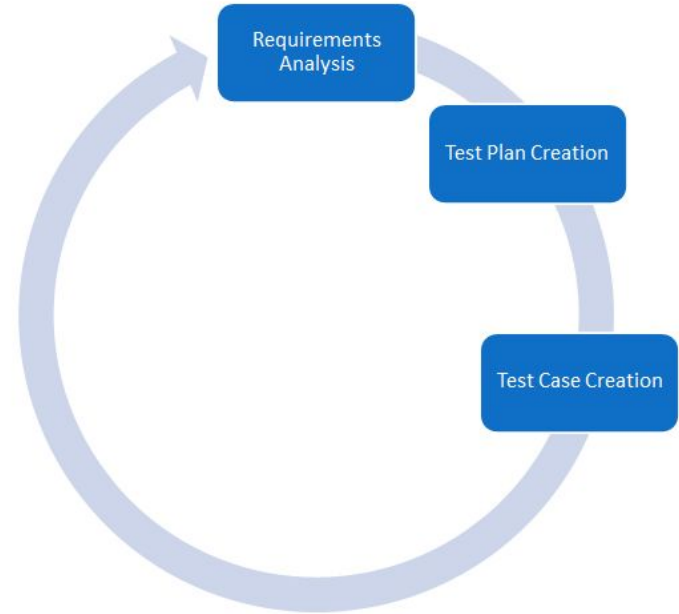
TEST BRAIN

Key Outcomes

Test process prior to implementing the Test Brain



Test automation
process after integrating
with the Test Brain



80%

Reduction in the time testers spend triaging automated test results

1

Defects found one phase earlier

30%

Reduction in manual testing required

0

Number of flaky tests that will break your automated results.

80%

Reduction in emails sent to team members from automated testing

98%

Average pass percentage, compared to 70% previously

~30%

Increase in testing productivity

THANKS!

38

Any questions?

For more information reach out
to us at appsurify.com

