

# QEdge Best Practice

## Contents

1. Best Practices for Independent evaluation of the vendors .....	3
2. Best Practices for Process and Metrics .....	3
3. Best Practice for Communication management.....	3
4. Best Practice for readiness of test environment .....	3
5. Best Practice for Multi-location testing .....	4
6. Best Practice for Regression testing .....	4
7. Best Practice for Test Management plan.....	4
8. Best Practice for defect lifecycle management .....	4
9. Best Practices for Test Plan design .....	5
10. Best Practices for Test Script design .....	5
11. Best Practices for Test Data .....	5
12. Best Practices in Metrics .....	5
13. Best Practices in Framework design .....	5
14. Best Practices in selecting Automation Tool.....	6
15. Best Practice in Test Scripts design.....	6
16. Best Practices in Test Plan design .....	6
17. Value addition of Automation .....	6
18. Best Practices – Performance Testing.....	7

## 1. Best Practices for Independent evaluation of the vendors

- 3<sup>rd</sup> party unbiased evaluation of product & applications.
- Best evaluation processes like
  - Performance criteria- concurrent users/ through put / response time.
  - Usability criteria – Follows best human Computer interface techniques
  - Feature / documentation.
- Product certifying procedure.
- Research & Development in developing techniques for latest technology.
- By identifying and developing common functionality across modules.
- Study Mechanism for understanding inter-module Interactions and solutions for the global and integration issues.
- Legal compliance associated with product globally.

## 2. Best Practices for Process and Metrics

- Standard Practice for Knowledge transfer
- Transparent Testing efforts estimations
- Tool evaluations criteria's
- Test Plans - For web testing, Multi platform, Multi Location, UAT, Multi lingual
- Customized Test Execution models for Regression, Performance, Functional, Stress, Load, Security testing –
  - Developing Automation framework
  - Preparing Test Strategy
  - Test Design
  - Evaluation by defect analysis & log reviews
  - Six Sigma methodology & CMM practice, Metrics collection, Process analysis & improvement

## 3. Best Practice for Communication management

- Setup Program Management - Project Office.
- Onsite Offshore Communication – time zones consideration.
- Establish Common testing language & terminology for test deliverables and artifacts.
- Establish SLAs

## 4. Best Practice for readiness of test environment

- Proper Tool evaluation before Test Environment
- Multi Platform simulation & Testing Procedures
- Establish Business Analyst & DBA & tester team for Test Data
- Setting up Configuration Status Reporting tools for team
- Automated environment generator

- Practice captures the environments/ break down issues, tools, and techniques.

## 5. Best Practice for Multi-location testing

- Onsite- Offshore execution Model
- Onsite Program/ coordination management Office
- Offshore Team & environment setup Procedures
- Smooth Offshore transition by customized knowledge acquisition process
- Domain training & expertise to test team members

## 6. Best Practice for Regression testing

- Evaluation & use of Right Automation Tool
- Regression Planning for Functional & Performance Test
- Tight Integration with Unified Process for test release/ build management
- Frequent builds mechanism from changes that are being promoted into the change control system
- Use case scenario implementation helps in high ROI
- Minimizing regression test cases

## 7. Best Practice for Test Management plan

- Matured knowledge transfer & Acquisition process
- Capturing Process data helps in continuous improvement
- Fast paced detailed planned training (Domain/ technology/ tools) to associates on multi platform.
- Identifying leaders to undertake initiatives for estimation, training, Knowledge Repository, domain and tools.
- Implementing V model and involving test team in early phase of the requirement gathering
- Implement iterative testing model
- Establish Access & security policy for Test environment
- Early implementation of Review/ code walk /inspections
- Develop Beta testing program by identifying target resources
- Mechanism for newer releases of software availability to developers and testers sooner.

## 8. Best Practice for defect lifecycle management

- Formally declaring Product Testing in terms of
  - Coverage

- Defect Density Visibility - Defect Per release/ Per module/ - detailed analysis
- Severity & priority assignment – A Cost Benefit Approach –
- Severity - Low/ Medium/ Major/ Minor
- Priority - Critical/ Urgent/ High/ Low
- Decision in Assignments of this status has major impact on Cost and time to market.
- Validate defect Analysis
- Real Time Defects Analysis
- Defining Project Scope & Out scope
- Formal entry and exit criteria
- Statistical Testing: This practice statistical testing needs to exercise the software.

## 9. Best Practices for Test Plan design

- Business knowledge for Test leads
- Tool Experts with business knowledge for Automation Testing

## 10. Best Practices for Test Script design

- Testing organization need to have good experience in both testing and business knowledge in enterprise applications

## 11. Best Practices for Test Data

- Variety and Volume of Test data improves test coverage
- Effective Error handling and Recovery scenario can be build in building robust frameworks

## 12 Best Practices in Metrics

- Design the metrics as part of Process Improvement
- Customer to be appraised the organization standards in terms of quality
- Testing Efforts to be measured effectively.

## 13 Best Practices in Framework design

- Good control over the automation tool and application
- Coding standards
- General tool library and Application library need to be developed in the QA lab instead doing so as part of project
- Tool Settings

- Application existence checking, login, logout, recovery scenario etc

## 14 Best Practices in selecting Automation Tool

- Select the tool for Automation and Test management based on evaluation points
- Some of the tools will be with exclusive add-ins/plug ins in line with enterprise applications. Some of the objects can not be recognized by the tool even after using these add-ins as Enterprise applications use propriety objects in it's UI design. Testing team need to exercise these exclusively
- Vendor support and Knowledge base support from vendor can not be ignored in selecting the tools

## 15. Best Practice in Test Scripts design

- Create the test scripts by keeping the role and responsibilities of different users' roles
- See that test scripts cover end-to-end architecture and interfaces in terms of Hardware and Software
- Bring modularity and re-usability for automated scripts
- Enterprise applications will use complex back-end database design. The data can be listed with the help of lists and tables. Incorporate the checkpoints to capture the behavior of business transactions in test script design itself.

## 16. Best Practices in Test Plan design

- Difficult to simulate the enterprise application in case testing is in 3rd party QA labs as these applications are connected to variety of systems and applications including legacy systems in the backend with complex database designs. Hence designing of test coverage is crucial in test plan
- Draw the line between Manual and Automation in Test Plan itself
- Define the decision factors in advance in selecting the test cases for Automation
- Evaluate the automation tool for suitability and ease of use and maintenance
- Identify a resource who is an expert in the automation tool and keep this resource involving in supporting the testing team rather than involving the resource in the project

## 17. Value addition of Automation

- Re-usability
  - Effective usage of Resources

- Effective deployment of resources
- Time Management
  - Proper Estimations are possible
  - Scheduling of Regression Test scripts is possible
- Reliability
  - Consistent results
  - Error rate be nullified in Data usage
- Improved Test Coverage
  - A business function can be tested with variety and volume of data
  - Can be integrated with Test Management tools. Test Metrics can be collected easily
  - More business functionality testing can be covered in a shorter time
- Bottom line: Automation improves organization confidence in deploying the application into production. Also it is important to note that all the testing effort can not be automated.

## 18. Best Practices – Performance Testing

- We have created Performance testing SO knowledge portal. Which consists of reusable assets, case studies, templates, process docs etc.
- Early involvement of the Performance test team in the project life cycle leads to better performance of the Application.
- Execution of Performance test on dedicated performance test environment like staging which is sizable or replica of production environment provides accurate results and can extrapolate.
- Developing reusable Performance test scripts independent of date, data and environment. So that scripts can used across all test environments.
- Initial performance testing cycle will focus more on non-code related performance variables, so that we can reduce analysis cycle time.
- Test Data requirements for the performance tests was identified early helped in minimizing test data risk

