

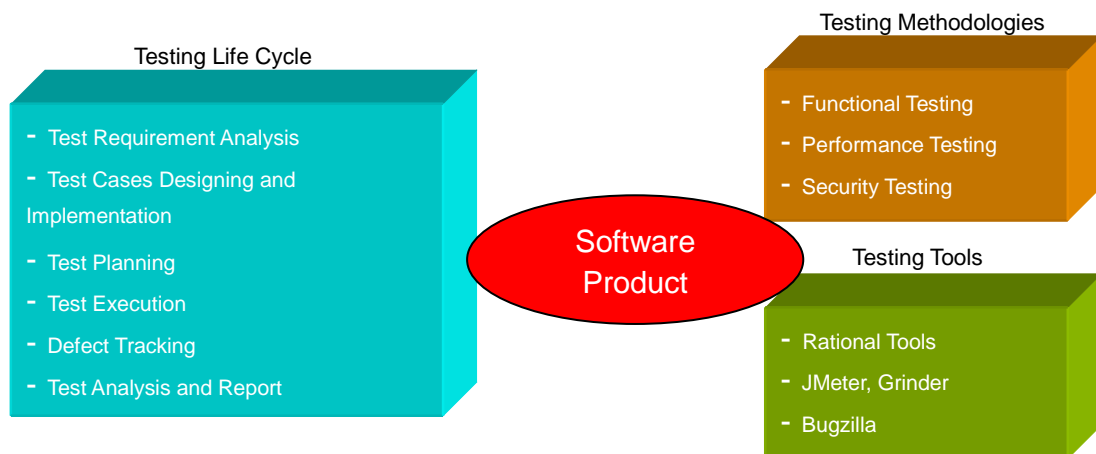
Software Quality Assurance

Overview

Hengtian aims to provide the software products to customers with high quality. A separated software quality assurance department is established in Hengtian to ensure that SQA plan and software processes have been effectively established and executed in applicable project to ensure the highest quality of the delivered software product to the customers. The SQA activities mainly include: test-oriented quality assurance, process-oriented quality assurance and metric-oriented quality assurance.

Test-Oriented Quality Assurance

Software testing is regarded as an important way to assure the quality of the software product, which is a process with efficient methodologies, tools and metrics. Specific testing group is established in the project to execute the formal software test life cycle. Different test methodologies and tools will be adopted according to the project requirements.



● Test Life Cycle

In order to ensure the quality of the software product produced by DEV, test resources are usually involved in the early stage of the SDLC. The activities in the different stages of the whole software test life cycle (STLC) include:

- 1) Test Requirement Analysis
- 2) Test Cases Designing and Implementation
- 3) Test Planning
- 4) Test Execution
- 5) Defect Tracking
- 6) Test Analysis and Reporting

● Test Methodologies

The test methodologies will be chosen according to the analysis of the test strategies for the different stages of the specific project.

- 1) White-Box Testing/Code Review/Unit Testing
- 2) Functional Testing/Regression Testing
- 3) Performance Testing/Stress Testing/Load Testing
- 4) Security Testing

● **Test Tools**

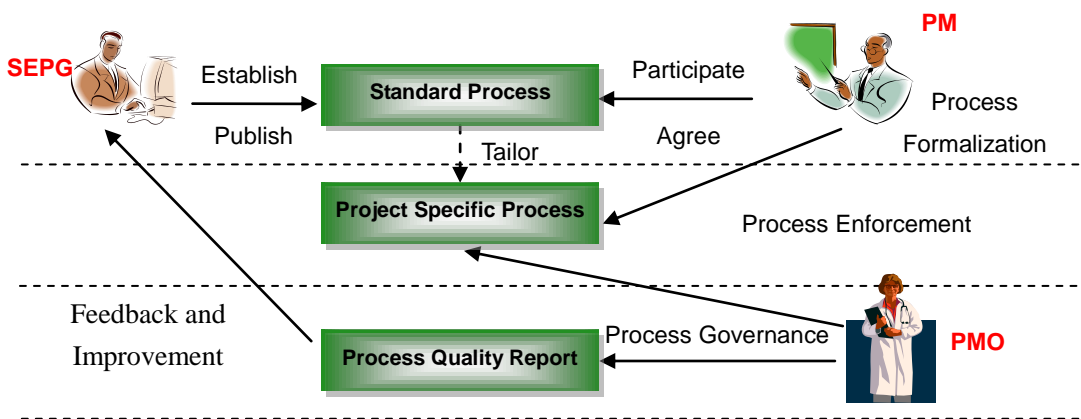
The application of the test tools will usually increase the efficiency of the test work and reduce the related test efforts. The selection of test tools will also be decided according to the requirement of the specific project. Following shows the common test tools that will be adopted in Hengtian’s projects.

- 1) Test Management Tools: Rational Test Manager, TestLink
- 2) Automation Tools: Rational Robot/Functional Tester, QTP
- 3) Performance Test Tools: JMeter, Grinder, LoadRunner
- 4) Defect Tracking Tools: Rational ClearQuest, Bugzilla, JIRA
- 5) Configuration Management Tools: Rational ClearCase, SourceSafe, CVS

Process-Oriented Quality Assurance

Software process plays an important role in the development of software product, which may directly affect the software quality. Besides software testing, process-oriented quality assurance is another important way to assure the quality of the software product, which includes:

- Process Formalization
- Process Enforcement
- Process Governance
- Process Improvement



SEPG will establish and publish standard process with PM’s participation and agreement. When a project is kicked off, PM will select the standard process and generate project specific process by tailoring it. During the execution of the process, PMO will be responsible for governance and provide the process quality report for continuous improvement.

The key points for process-oriented quality assurance are to:

- 1) Assure the quality of the process: process formalization and improvement
- 2) Assure the quality of execution of the process: process enforcement and governance

Metric-Oriented Quality Assurance

In order to achieve the continuous assurance and improvement of software quality, metrics will be defined to collect the quality data from projects to do the quality analysis and evaluation.

● Metrics

The common metrics for Hengtian’s projects include: Defect Density (such as Defect Number/KLOC, Defect Number/Module, and Defect Number/Release), Defect Number/Test Cases Number, Defect Number/Test Effort, Test Efforts/Defect, Test Cases Pass Ratio, Test Coverage, Defect Distribution (per module and per phase), and Schedule Estimation Accuracy, etc.

● Quality Analysis and Evaluation

With the formalized metrics, the quality data for each Hengtian’s project can be collected regularly to do the analysis and evaluation for quality improvement. Following shows an example of monthly bug summary report that will be used to reveal the quality issues in each project for comparison.

Bug Summary Report for June, 2007									
Report Date: 2007-07-02					Reporter: Jingfan Tang				
Dept.	Project	Bug Severity	Fixed Bugs	Pending Bugs	Other Bugs	Total Bugs	Total DEV	Bugs/DEV	
Dept1	Proj1	Critical	0	0	0	0	19	10	1.90
		Normal	10	0	0	10			
		Minor	9	0	0	9			
Dept1	Proj2	Critical	0	1	0	1	22	5	4.40
		Normal	0	13	1	14			
		Minor	0	7	0	7			
Dept1	Proj3	Critical	0	0	0	0	0	4	0.00
		Normal	0	0	0	0			
		Minor	0	0	0	0			
Dept1	Proj4	Critical	0	0	0	0	3	5	0.60
		Normal	3	0	0	3			
		Minor	0	0	0	0			
Dept1	Proj5	Critical	1	4	2	7	15	10	1.50
		Normal	1	4	1	6			
		Minor	0	2	0	2			
Dept1	Proj6	Critical	3	8	0	11	30	8	3.75
		Normal	3	11	0	14			
		Minor	0	5	0	5			
Total			30	55	4	89	42	2.12	
Dept2	Proj7	Critical	0	0	0	0	0	0	N/A
		Normal	0	0	0	0			
		Minor	0	0	0	0			
Dept2	Proj8	Critical	0	0	1	1	44	7	6.29
		Normal	12	0	22	34			
		Minor	4	0	5	9			
Total			16	0	28	44	7	N/A	
Dept3	Proj9	Critical	0	0	0	0	19	4	4.75
		Normal	0	7	0	7			
		Minor	1	11	0	12			



TOP 5 Projects (in Total Bugs)					
Dept.	Project	Fixed Bugs	Pending Bugs	Other Bugs	Total Bugs
Dept1	Proj1	32	32	9	73
Dept2	Proj2	41	12	17	70
Dept3	Proj3	19	40	0	59
Dept1	Proj4	35	9	0	44
Dept4	Proj5	16	28	0	44

TOP 5 Project (in Bugs/DEV)							
Dept.	Project	Fixed Bugs	Pending Bugs	Other Bugs	Total Bugs	Total DEV	Bugs/DEV
Dept1	Proj4	35	9	0	44	3	14.67
Dept1	Proj1	32	32	9	73	7	10.43
Dept1	Proj6	17	10	0	27	4	6.75
Dept1	Proj5	16	28	0	44	7	6.29
Dept5	Proj7	0	9	2	11	2	5.50