

Grooming for a “Cleaner World”

The use of software in almost all walks of life is increasingly visible. In this context, any defect in software has serious economic impact. At STAG, our vision is to strive for a *Cleaner World* by helping companies build excellent software. We firmly believe that this can be achieved not by practicing ad hoc testing techniques but only by an application of a formal body of test knowledge - science and engineering of testing. We take pride in expanding the testing community at large and contribute to a *world free of defects*.

IT corporate houses who recruit large scale fresh graduates as part of their business plan increasingly find many of the questions unanswered - **See Box**. In order to effectively fulfill its staffing needs, the executives of the organization need to reflect on how soon the new recruits will start contributing value to the organization, before it engages its precious internal resources to train these fresh recruits.

- ❖ Do we have internal trainers with desired level of competency and communication skills? (Ability to articulate concepts/ideas. Smooth flow of presentation of topics. Possess adequate knowledge of subject matter)
- ❖ Do we have internal bandwidth to undertake training on regular frequency for the fiscal year?
- ❖ Is it justifiable to move a high value internal resource to conduct training?
- ❖ Is there adequate focus in training when internal resource is engaged?
- ❖ What is the right length and depth of topics to discuss in the program?
- ❖ Will the participants be provided with adequate examples for better understanding?
- ❖ Will the participants be provided with adequate lab work?
- ❖ Will the class be time effectively utilized?
- ❖ Will the participants’ queries be clarified adequately?
- ❖ Can effective class participation be ensured during the training?
- ❖ Will the training fulfill the stated objective of the organization?
- ❖ Can the course material of adequate quality be ensured?

- ✓ Groomed around 5000 test engineers
- ✓ Over 100 IT companies have employed STAG trained test engineers

At STAG we have rich experience and expertise in software test engineering and our infrastructure is designed as testing workforce development platform - *Knowledge ‘factory’* - that is tested and proven. In short we build competency in software test engineering for IT companies.

STAG Finishing School from Knowledge Services division is an industry-oriented program. The curriculum lays strong foundation in test engineering techniques. It is specifically designed to build competency in software test engineering to aspiring test engineers for entry level positions in IT companies and transform them into high productive individuals from day one - make them project-ready before assigning them to real world testing.

Objective of the program

STAG Finishing School aims to impart test engineering techniques to aspiring test engineers. The program is tailor-made in building competency for *test execution* in the test life cycle. As a test practitioner they can create efficient test plan, design effective test cases, measure quality, and build robust automation and setup pragmatic process. The program enables the aspiring test engineers be conversant with testing terminologies, software development lifecycle, process models and aware of popular tools available in the market. In short, transform the raw talent into high caliber skilled individuals.

People and Diamonds have real market value once polished

Benefits to IT organizations are

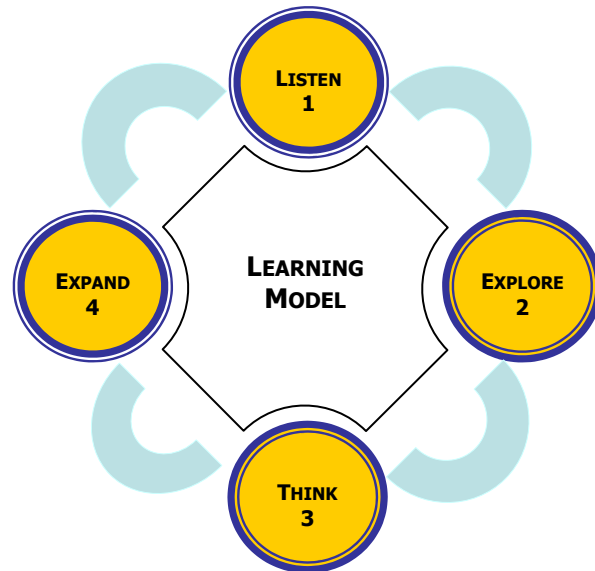
- ✓ Training from specialist who are practitioners of testing - subject matter experts - with rich experience
- ✓ The fresh recruits are transformed into project-ready testing professionals
- ✓ Faster throughput in terms of fresh recruits contributing to business goals
- ✓ Better utilization of internal resources to derive optimal results
- ✓ Reducing total cost of ownership thereby savings for companies

STAG Finishing School

Methodology

The spirit of imparting knowledge is very much on the lines of building a healthy body. We have distilled our experience and knowledge and put together a cost effective knowledge transfer mechanism.

Our approach to teaching is experiential; we have devised a unique **Learning Model** that is cogent, result-oriented and fosters evolution. The different components are as follows -



Listen - Ensure correct understanding of concepts

Explore - Make learning application-centric

Think - Build confidence, transform people

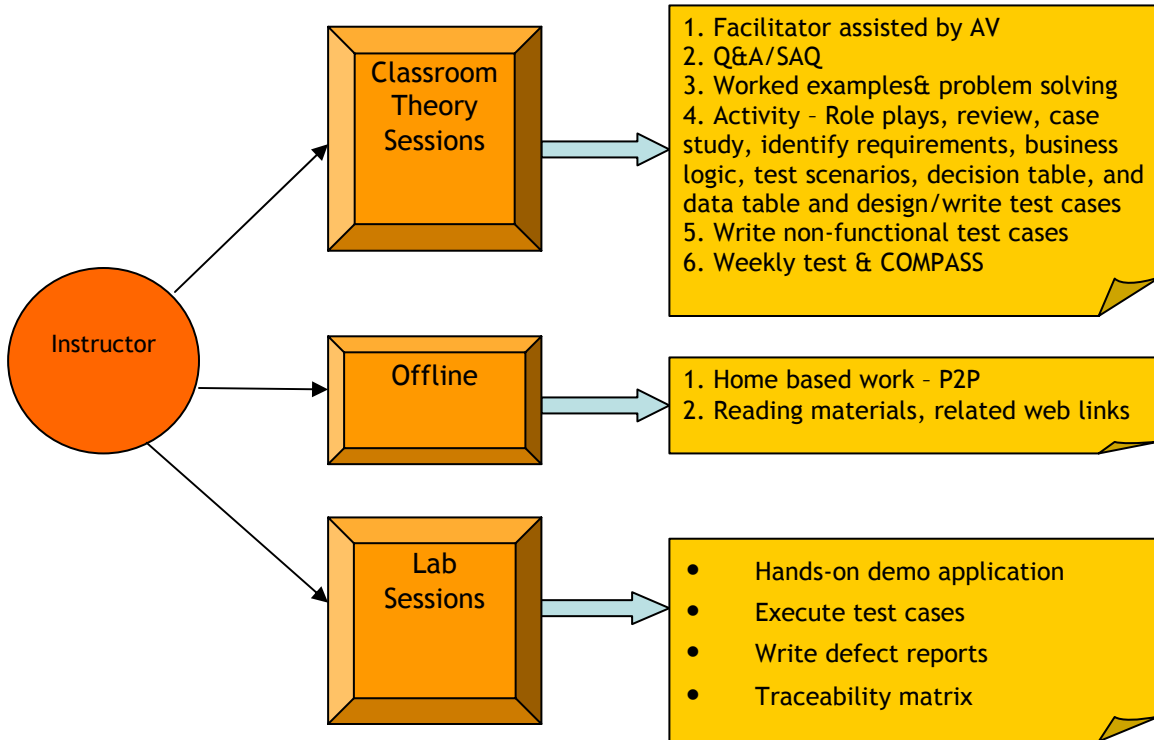
Expand - Ensure continuous learning

Highlights of STAG Finishing School

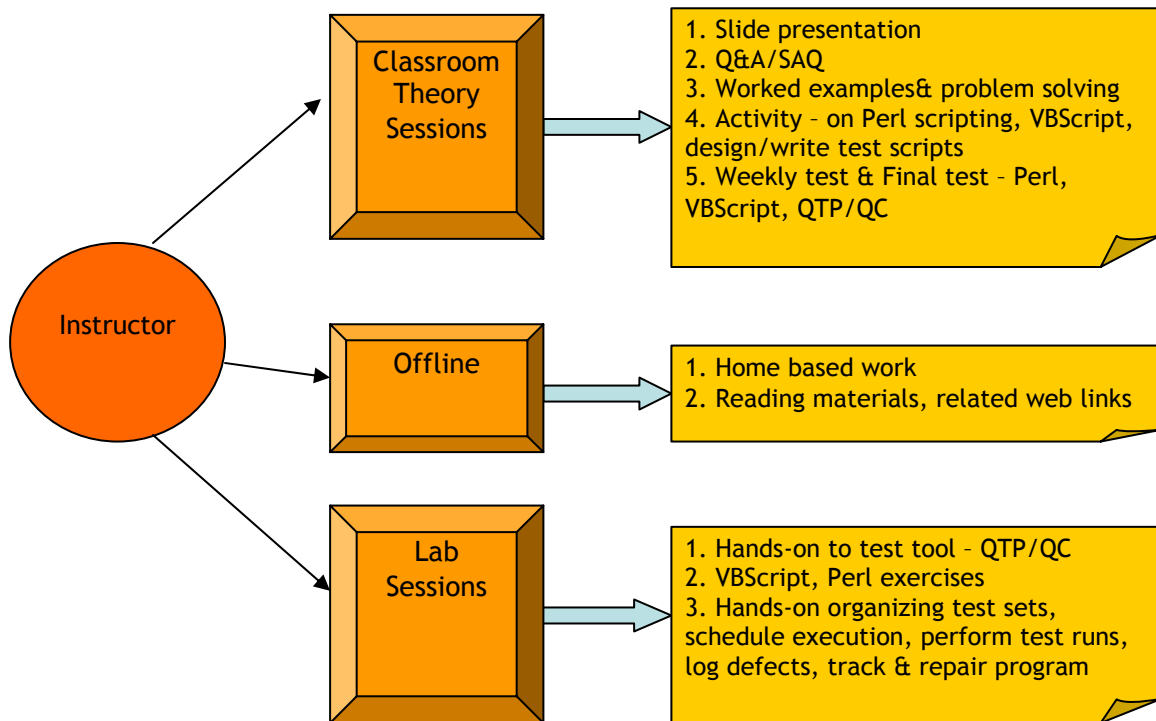
Component	Highlights of STAG Finishing School
Content	Concise, effective for easy understanding created by test professionals with over twenty-five year’s industry experience.
Delivery Theory (Testing concepts) (Test automation)	- Modular. Lecture sessions facilitated by instructor and supported by audio-video with well-modulated voice-over of a test professional with over twenty-five years industry experience, interspersed with related examples. The emphasis is to ensure proper assimilation of concepts rather than just reading out text from the slides. Instructor based with slide presentation by experienced test automation professional. The emphasis is on relating the importance of test automation, and how to use test tool efficiently.
Delivery Lab sessions	- The lab is designed to be an environment to do a set of things, explore concepts and ask questions. This ensures that the theoretical concepts are understood correctly and facilitate real-world application.
Transforming people	An innovative aspect of this program is ‘P2P - Points to Ponder’. P2P are questions to think on, and ‘pick’ your brain, pleasantly. Self-assessment questions (SAQ) at the end of each module help the participants think and clarify. Learning extends beyond classroom. To enable participants expand their knowledge horizon they are provided access to related reading materials - books, web links and worked examples.
Role of Instructor	Facilitator to ensure effective and correct transfer of knowledge. Play vital role hand-holding participants, answering their queries and assist them during the hands on lab session. Guide in case study discussions, problem solving with Q&A session
Duration	4 weeks - 2 weeks Testing Concepts module& 2 weeks Test Automation module - 5 days/week -8hrs/day with 15 minutes break in morning and afternoon session and 1 hour lunch break
Environment	Formal classroom.
Quality of training	Consistent, independent of instructor, tailor-made to the proposed role of participant - entry-level test engineer
Elicit feedback	Yes - central to this program

Both theory and lab sessions are well-balanced and constitute 50:50 for both Testing Concepts and Test Automation module.

Delivery flow - Testing Concepts



Delivery flow - Test Automation



Curriculum

The curriculum lays strong foundation in test engineering techniques - test process, test types, test metrics, test terminologies, test levels. Further, it also includes awareness to test automation and tool.

Module - Testing Concepts -1 - Theory	Duration - 4 days
Introduction to testing-1	
Verification and Validation, Test Objectives, What is testing? Defect types, Defect economics	
Introduction to testing-2	
Introduction to Levels, Types and Techniques of testing	
Methods to uncover defects and the role of testing in uncovering defects What types of defects to discover at each stage in SDLC	
Test lifecycle	
Understand the various phases that constitute testing and its relationship to SDLC, Roles people play in testing, Deliverables in each phase of the lifecycle	
Verification-Reviews, Walkthroughs, Inspections	
Understand the various types of verification, Understand the process of Software inspection, Inspection related metrics	
Black box test design Techniques	
Understand how to use specification to design minimal yet effective test cases. Test scenarios/cases, Use High level and Low level Techniques to design Test Scenarios and Test cases respectively	
White box technique	
Understand how to use code structure to design/Improve test cases What is Code structure, Control flow and Data flow aspects, Cyclomatic complexity, Coverage based testing, How to measure coverage and use them effectively	
Unit, Integration testing	
Understand clearly what a Unit/Interface is, what type of defects to focus on at each level, Designing unit test cases and deciding on Integration test strategy	
System testing	
Understand what type of defects to discover, Understand the various types of tests that constitute system tests, What is system? What is quality?	
LSPS tests	
Understand the focus of Load, Stress, Performance and Scalability test. Understand operational profile technique for Designing LSPS test scenarios	
Test strategy and test planning	
Understand how to de develop effective test strategy and plan, Test criteria, Test cycle, Estimating test effort, Understand various reports generated and their objectives Ex. Defect reporting, Weekly reporting, Test Cycle Reporting etc.,	
Test Reporting and Management-2	
Understand key aspects of Test Management Defect Management, Risk Management, Lab Management, People Management Analysis and Metrics	



Module - Testing Concepts -2 - Theory	Duration -1 day
Web Testing introduction	
Understand architecture of an web application	
What issues to uncover and therefore the types of tests to focus on	
Testing Client Server Applications	
Understand what it takes to test an Client Server Application	
Testing Database Applications	
Understand what it takes to test an Database Application	
Quality concepts	
Appreciate basic quality concepts and quality frameworks	
What is quality? QC vs. QA, Quality Costs, ISO 9000, Malcolm Baldrige Award	
Testing packaged applications	
Data Conversion Testing, Data Migration Testing	

Testing Concepts - Lab	Duration -5 days
Requirements Traceability Matrix	
Test Cases	
Test execution	
Test Results	
Defect Reports	

Module - Test Automation -1 - Theory	Duration - 1 day
Test Automation -1	
Why and What of automation?	
Benefits and Limitations of automation	
Aspects to watch out for in automation	
Skill set required for a good automation engineer	
Test tools Landscape	
How do Functional test tools and Load Test tools work	
Framework for Test Automation	
Scripting Languages	
What are Scripting languages	
How are they useful to a test team?	
What does a scripting language allow us to do?	
A very short introduction to Perl	
Test Automation -2	
Why Test Automation?	
Test Candidates for Automation	
Test Automation process	
Automation feasibility analysis	
Planning and Design	
Test Environment setup	
Automation script generation	
Automation test script execution	
Keyword driven automation framework	
Defect analysis and fixing	
Documentation	
Costs involved in automation	
Tool cost and Manpower cost	

Introduction to functionality Test Automation Tools	
Major Functional test automation tools	
Automation scripting languages	
Environment support(OS, Technology, Browsers, DB)	
Basic principles of test automation tools	
What is a Test Script?	
Test object verification	
How does the test automation tool verify the test object?	
What are synchronization points? What are verification points?	
Test automation tool classification:	
Object Based automation tools	
Bitmap based automation tools	
Approaches for Test Automation	
Record & playback	
Scripting	
Framework (Definition of Framework)	
Introduction to Test Automation framework	
Data driven framework	
Keyword driven automation framework	
Hybrid test automation framework	
Best practices for developing automated test scripts	
Version control of the scripts	
Documentation	
Test scripts naming standards	
Preferred scripting standards and checks	
Integration of Test automation tools with Test management tools	
Various Test management tools	
Test management tool architecture	
Additional features of Test management tools	
Use of Test Management tool in different phases of Testing	
Requirements management	
Test Planning	
Test execution	
Defect reporting	
Metrics and Reports	
Myths and realities of automation	
People concerns	

Module - Test Automation -2 - QTP version 9.0 - Theory	Duration - 2 days
Introduction to QTP	
QTP architecture and its Salient Features	
Environment Supported	
Creating and Enhancing a test suite	
Integration of QTP with Test management tools	

Module - Test Automation -2 - QC version 9.0 - Theory	Duration - 1 day
Introduction to QC	
Architecture	
Test Management Process	
Organizing and managing the testing process using different modules	
Requirements	
Test Plan	
Test Lab	
Defects	
Other important features	
Reports & Graphs	



Import excel and word docs	
Document Generator	
To create and setup a Quality Centre project using the Site Administrator	
Customize the project according to the project attributes	

Module - Test Automation -3 - Scripting - Theory	Duration -1 day
Introduction to VB Scripting	0.5 Day
Introduction to Perl Scripting	0.5 Day

Test Automation and tool - Lab	Duration - 5 days
Add Test artifacts in QC	
Prepare the test scripts for the selected TCs	
Logging the defects in QC	
Preparation of RTM, graphs and reports	

Note:

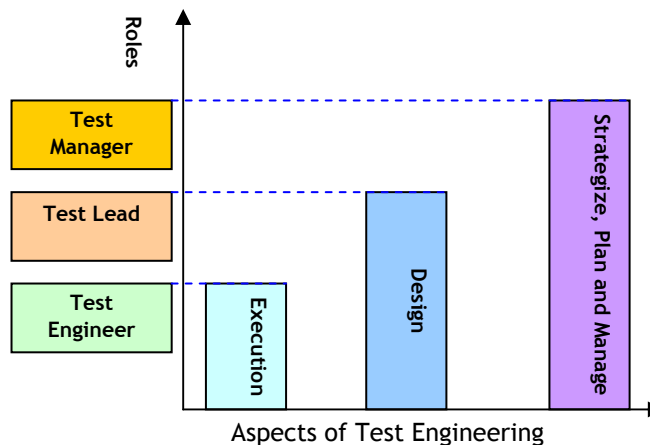
- a. There shall be some variation in covering certain topics in duration. However, total no of training days would be 20 (which include Testing Concepts & Hands on Lab and Test Automation & Hands on Lab).
- b. The above course contents are customizable to the organization needs. In the case of Testing Concepts module, the topics may be rearranged as desired. In the case of Test Automation - Tool, though we have listed the topics under QTP 9.0 and QC 9.0, we can provide tool training for Borland SilkTest and IBM RFT as desired by organization.

Assessment

We believe specific competencies are needed in the area of Test Engineering to perform a role in the test life cycle. Hence, we have a unique assessment model to test skills of QA team - Test Competency Assessment - **COMPASS**.

The spirit of COMPASS is to break down the competency list to more fine-grained key competency areas in Test Engineering and assess for the same.

The specific key competencies to be possessed by Test personnel vary according to the roles within the hierarchy of the Test Division/Organization. Typical roles would be Test Engineer, Test Lead, and Test Manager. The competency required to perform each of the stated role is different.



The participants are administered

COMPASS as a written exam for a duration spanning 60 to 120 minutes at the end of Testing Concepts module. The questions are aligned to the competency aspect - fundamentals of testing, knowledge of test process, ability to write test cases, perform test execution, ability to write defect report - with respect to the defined role - *Test Engineer*. The answers are evaluated on the basis of strong understanding of the test concepts, methods, techniques and its application. The assessment is aimed to determine areas of strength and areas of improvement in the participant with respect to that competency aspect for the defined role. Therefore it is not merely a Pass or Fail.

Finally at the end of Test Automation module, the participants have to take an exam that evaluates their basic competency in test automation, scripting language, and understanding in using test tool.

Performance report

A report on every assessed participant is prepared which highlights the areas of strength and areas of improvement with respect to specific competency aspect for the role. This helps test lead/manager in understanding the competency profile of each new test team member during team composition for a project and their contribution in the success of the project.



SAMPLE PERFORMANCE REPORT

NAME: JOHN ABRAHAM

ROLE: TEST ENGINEER

COMPANY: ABC Co

Employee No 786 Finishing School Program start date June 14, 2007

Finishing School Program end date July 8, 2007

Part 1: Testing Concepts

Evaluation: COMPASS

Date: June 25, 2007

Competency aspect	Improvement opportunities
Definitions, terminologies, objectives	Need to get a clear understanding of levels, types and techniques of testing.
Verification techniques, Inspection	Objectives of review require clarity in understanding
Defect lifecycle	Excellent
Write good defect report	Seem to know about defect severity, but need to know about priority.
Black box techniques	Need to improvise the ability to find inconsistent information.
White box techniques	Need to understand application of code coverage metric.
Levels of testing	Focus on the various issues to be found at each level.
Types of tests	Excellent
Software quality attributes	Focus on non-functional tests.
Test process/life-cycle	Excellent
Test metrics	Need to get a clear understanding of Inspection effectiveness.
Test reporting/Documentation	Focus on reports and documentation of tests.
Test strategy and planning	During estimation, effort required to complete each and every task involved must be considered.
Appreciate the need/limitations of automation	Excellent
Total score (Max score: 60)	49 Percentage 82

Part 2: Test Automation

Evaluation: Written test

Date: July 8, 2007

Total score (Max score: 83)	67 Percentage 81
Overall Grade (Testing Concepts + Test Automation)	A

Program Assessor

Grading

Where client organization requires grading of participants on completion of program, the following grading system will be provided:

A+	90% to 100%
A	80% to 89%
B+	70% to 79%
B	60% to 69%
C+	50%to 59%

Status Reporting and meetings

STAG would provide weekly status reports to client organizations on a weekly basis during the training period. There would be need basis meetings where both parties meet. There shall be lessons learnt meeting at end of every training program.

Deliverables

Include

- ❖ Attendance Reports
- ❖ Evaluation Results (Intermediate & final Test)
- ❖ Hands-on Project deliverables- test cases written by participants - min 30 test cases - defect reports, test scripts - min 10 in number
- ❖ Course material - 600 pages /participant
- ❖ Lab workbook - 100 pages / participant
- ❖ Performance report - includes COMPASS report on testing concepts and marks scored in Test automation by each participant - with overall grading (if required)

Customer Speak* on Instructor	
Ability to articulate concepts / ideas	Good
Flow of presentation of topics	Excellent
Knowledge of the topics taught	Good
Utilization of class time	Good
Ability to clarify doubts	Good
Customer Speak* on Course	
Quality of training material	Good
Quality of lecture	Good
Quality of lab	Good
* Bangalore based Tier-1 IT Services Company	

Target audience

Fresh graduates - BE/B Tech, MCA, M.Sc/B.Sc with specialization in CS, EE, IT

Engagement

The program is conducted onsite at client premises. To conduct this program, the following support is required from client organization - training hall - 50-75 seat capacity- a LCD screen projector with connectivity to PC, 1 PC of standard configuration with CDROM/DVD drive, White board, and marker pens - 2 Black, 1 Green, 1Red, and 1 Blue - and AV systems.

For enquiries contact

In Bangalore

Ph: +91(80) 2535 7161 / 2535 7162

Email: learning@stagsoftware.com

In Chennai

Ph: +91(44) 4214 3469 / 2442 0396