

Software Testing (Automation) - Syllabus

MODULE 1

Software development life cycle
SDLC development models
Water fall model
Evolutionary development model
Agile model
Test driven development
Scrum model

MODULE 2

Types of automation
Unit test, integration, system testing
GUL testing, API testing, load testing
Smoke /sanity testing importance
Regression/ Functional testing

MODULE 3

Uses cases for testers
Writing good use cases
Elevator, mobile, phone, pen, coffee vending machine
List of technologies
Black box test technology
Boundary value analysis
Equivalence class partition
Error guessing
White box test technology
Statement coverage
Condition coverage
Path coverage
Branch coverage
Types of test cases
Positive and negative cases
UI test cases
Usability test cases
Field validation
Functional test cases

MODULE 4

Test plan document
Title
Revision history
Objective of document
Scope of document
Objective of testing
Metric collection

Software Testing (Automation) - Syllabus

Project description
Critical functionality
Test data requirement
Features not to be used
Test environment
Training requirements
Effort estimation
Resource requirement
Scheduling
Test strategy
Input/ entry criteria
Exit criteria
Test suspension and resumption criteria
Test completion criteria
Acceptance criteria
Bug classification
Test deliverables
Standards to be followed
Risk analysis

MODULE 5

Bugs
Bug classifications
Bug template
Bug tracking
Bug tracking tools
Bug life cycle
Statues for bug life cycle

MODULE 6

JAVA
OO concepts
Encapsulation
Inheritance
Abstract classes, interface, final
Polymorphism
Overriding, overloading, this, super, constructor
General
Collection
Lists, sets, revise basic algorithms if time permits
Exception handling
JDBC

MODULE 7

Software Testing (Automation) - Syllabus

Log4J
ANT
SVN
HUDSON
JIRA, Bugzilla
JUNIT, Test NG
Other languages
SQL
PERL
UNIX

MODULE 8

Mercury quick test pro
Introduction
Recording
Object repository
Standard checkpoints
Database checkpoints need to look
Parameterization
Data drove testing
Output values
Actions
Descriptive programming

MODULE 9

Load testing (load runner)
Fundamentals of load runner
Planning an effective load test
Load runner installation
Virtual user generator scripting
Recording and playback
Action and transactions
Parameters, checkpoints correlation
Advanced correlation
Enhance V user output log
Error handling
Introduction to scenarios
Using run-time setting
Scenarios execution
Scheduling scenarios
Performance monitors
Result analysis
Building effective load test scripts
Load runner hand on exercises

Software Testing (Automation) - Syllabus

MODULE 10

Test management tools
Adding test requirements
Create tests
Executing test case manually
Analyze project progress
Run tests and analyze the results
Report and trace defects
Document generator
Executing test scripts remotely and more
The test case with requirements
Descriptive programming

MODULE 11

Manual testing real project
Take a real project and do the following in different phases of the QA lifecycle
QA basic
Requirement
Test plan
Sizing
Test case
Bug lifecycle
Log- with log4j
Build
Boundary value analysis and equivalence partitioning
End and end testing
Status reporting
UAT
Production check out

MODULE 12

OR
SOR
Basic of web syntax
Descriptive programming
Functions
Functional library
Excel integration
Option explicit
Loop
Original identifier
Get TO property
Get RO property
Showing manual test script
Error handling using script

Software Testing (Automation) - Syllabus

Recovery scenario

For more details about this course, Click on this link: [Software Testing \(Automation\) Training](#)