HPE UFT120 – Unified Functional Testing

Overview

This course provides a comprehensive understanding of using Unified Functional Testing (UFT) 12.5x as an automated functional testing tool for different environments. You will use UFT's graphical point and click interface to record and playback tests, add synchronization points and verification steps, and create multiple action tests. You will build upon fundamental topics by using debug tools to troubleshoot tests and use additional checkpoints and product options to broaden the scope of business processes that can be automated. Once tests are created, you will discover and correct common record and playback problems. All topics are supported by hands-on exercises based on real-life examples.

Included is an introduction to HP UFT for API testing, which contains an extensible framework for the construction and execution of functional tests of headless systems (systems that do not have a user interface). The course concludes with an introduction to LeanFT, a powerful, lightweight, developer-oriented testing tool.

Learning Method

You will receive expert instruction from a UFT specialist who will present the course using slide presentation and facilitated discussion. At the end of each chapter there will be review questions followed by a hands on exercise to ensure understanding of each lesson.

Each student will receive a copy of the HPE Using Unified Functional Testing manual. This manual is used throughout the course and proves a useful reference tool upon completion of the course.

Course Objectives
Upon successful completion of this course, you should be able to:
- Use UFT to automate GUI and API tests
- Add additional UFT features make the test robust and provide better coverage
- Use LeanFT to create tests

Duration: 5 days

Who will benefit from this course?

This course is intended for:
- New users of UFT who need to automate manual testing and verification in a short amount of time.
- Quality assurance engineers who assume technical lead roles in the use of UFT application.

Prerequisites for this course

Working knowledge of:
- Windows
- Web sites and browsers

Any experience with:
- Programming languages
- Scripting languages

Course Content

Course Overview

- Identify the contents and objectives of the course
- Define the class schedules and class logistics
- Identify the related courses
- Discuss the lab environment details

Software Overview

- Describe the advantages of UFT as a testing tool
- Preview the UFT Start page and Help menus
- Recognize the sample applications used in the labs
- Identify resources for getting assistance

Preparing to Record

- Identify functional testing principles, and the benefits of automated testing
- Navigate the typical GUI testing workflow
- Document the steps of a business process
- Prioritize business processes using effective criteria
- Gather sufficient test data
- Prepare the test environment for automated testing

Creating a Basic Test

- Create a basic test from a manual test case
- Run a test and check for errors
- Save a test and view test results

Working with Objects

- Identify objects and their properties
- Define a UFT for GUI Testing object
- Discuss basics of the Object Repository

Utilizing a Shared Object Repository

- Identify the types of object repositories
- Usage of Object Repository Manager
- Use visual relation identifiers

Adding Synchronization

- Define & identify the uses of synchronization
- Add synchronization steps for a specified object

Verifying with Standard Checkpoints

- Define standard checkpoints
- Add standard checkpoints to a test
- Use regular expressions to add flexibility to a standard checkpoint

Using Parameters

- Use input and output parameters
- Parameterise a checkpoint
- Evaluate test result with iterative tests

Building Multiple, Reusable Actions

- Identify actions in GUI testing
- Identify action types
- Identify action and test iterations
- Identify calls to existing actions and copies of actions
- Share values using the global data table
- Call actions with parameters
- Store action return values
- Create multiple actions from a single action
- Create a new action
- Call a reusable action from another test
- Use local and global data sheets
- Resolve missing actionsCreate a reusable action

Adding Steps without Recording

- List the types of steps that can be added to a test without using the record feature
- Use conditional statements in a test
- Add steps using the Step Generator

Creating Tests on a Web Application

- Record and run a test on a web application
- Insert a text checkpoint in a test for a web application

Testing web services with UFT API

- Define Service Oriented Architecture (SOA)
- Identify components of SOA
- Define the terminology and principles surrounding component testing
- Navigate the UFT UI for API testing
- Create a basic Service Test (API Test) in UFT

Using UFT API

- Import a service
- Define test step inputs and outputs
- Identify data drive steps
- Work with controls flows
- Use the Results Viewer

Enhancing UFT API Testing

- Use checkpoints and reporting
- Configure database validation with UFT
- Configure security with UFT

Introduction to LeanFT

- Build a LeanFT web based test using Visual C#
- Build a LeanFT Windows based test using Visual C#
- Run the tests with NUnit

Using the Object Identification Center
- Use the Object Identification Center (OIC) to add steps
- Use the OIC to capture values from the AUT
- Output data to the Visual Studio log

Iterating with LeanFT
- Create an array
- Set test iterations based on the TestCaseSource area of the LeanFT script
- Create a loop for a section of a test
- Launch HTML reports

LeanFT Checkpoints
- Build checkpoints by inserting Assert methods for the framework being used
- Report results to the test results
- Control test pass/fail status
- Use the Assert within a try-catch statement

Appendix A: Using Database Checkpoints
- Identify the purpose of a database checkpoint
- Create a Structured Query Language (SQL) statement using Microsoft Query
- Create a database checkpoint
- Parameterize a database query

Appendix B: Significant Checkpoints for GUI Testing
- Create table checkpoints
- Create test and text area checkpoints
- Create file content checkpoints
- Create XML checkpoints

Appendix C: Object Identification Techniques
- Configure object identification
- Describe the mandatory and assistive properties
- Use ordinal identifiers
- Use smart identifiers
- Describe when to use Smart Identification
- Use the Smart Identification process
- Describe how UFT for GUI Testing uses Smart Identification – Use Case Scenario

Appendix D: Using Recovery Scenarios
- Identify exceptions in a test
- Create a recovery scenario
- Associate a recovery scenario with a test
- Set an optional step in a test

Related Courses
- ALM120 – Application Lifecycle Management 12.0 Essentials