Testing and acceptance for project ownership: Design and implementation

Hands-on course of 3 days - 21h
Ref.: REA - Price 2024: €2 180 (excl. taxes)

EDUCATIONAL OBJECTIVES
At the end of the training, the trainee will be able to:

Learn aspects of the profession: Psychology, positioning, limits

Understand the interdependencies of accepting a software project

Gain an understanding of functional and non-functional testing techniques

Develop test cases and test scenarios

Document and manage tests

Carry out the tests, up to the acceptance report

HANDS-ON WORK
Scenarios involving a case study that leads to the design, implementation and execution of application's acceptance tests in a tool-equipped environment.

THE PROGRAMME

last updated: 07/2021

1) Introduction to acceptance testing
- What is software testing and acceptance?
- The difficulties of testing and noncompliance.
- Tester psychology.
- Test players: Acceptance manager, tester, automation engineer, ergonomist, etc.
- Functional testing.
- The concept of "V&V", verification and validation.
- Types of tests.
- The notion of test coverage.
Exercise : Intuitive testing research based on an example.

2) The project context
- Understanding roles within the project (project owner, lead contractor, acceptance team, users).
- The project lifecycle: Traditional and Agile modes.
- The various processes.
- The deliverables (technical and functional specifications, acceptance plan, testing, etc.).
- Requirements (typology, quality criteria).
Case study : Reviewing the specifications being studied and their business requirements.

3) The approach to acceptance testing
- The acceptance process and its interactions.
- The steps of the acceptance approach (synoptics).
- Acceptance in the project schedule.
- Versioning.
- Non-regression tests.
Case study : Reviewing the acceptance plan being studied.

4) Managing tests and non-compliances
- The principles and practices of test management.
- Test reference spaces, functions. Market tools.
- Test architecture.
- Principles of non-compliance management.
- Anomaly managers, principles and market solutions.
- Test configuration management (principles and practices).

Hands-on work : Receiving and configuring a test repository. Receiving and configuring an anomaly management tool.

5) Test design
- Design principles.
- The course of design.
- Traceability of requirements and tests.
- Techniques of partitioning or defining "equivalence classes".
- "Limit testing" techniques.
- Decision tables or defining input-output combinations.
- Cause-effect graphing. Finite state graphing.
- Non-functional testing techniques (ergonomics, usability, performance, etc.).
- Covering test objectives.

Hands-on work : Designing tests for the application being tested with the test repository.

6) Implementing tests
- Principles and conditions of implementation.
- Providing functional specifications.
- Testing platforms
- Manual testing.
- Test campaigns, databases.
- Test data generators.
- Automated testing: Principles, benefits, and drawbacks.
- Functional test robots (overview).
- Peer review of tests.

Hands-on work : Implementing tests for the tested application. Presenting the implementation of an automatic test using a functional test robot. Cross-review of tests.

7) Executing tests
- Preparing the campaign.
- Delivery of the version being tested and its documentation.
- Conducting the tests.
- Recording the results and anomalies.
- Corrective/upgrading versions (Change Control Board CCB).
- Consolidating the results.
- Stopping and acceptance criteria (Go/No go).
- The acceptance report.

Hands-on work : Carrying out tests for the application being tested. Recording the results and final evaluation.

DATES

REMOTE CLASS
2025 : 26 Mar, 18 Jun, 24 Sep, 01
Dec