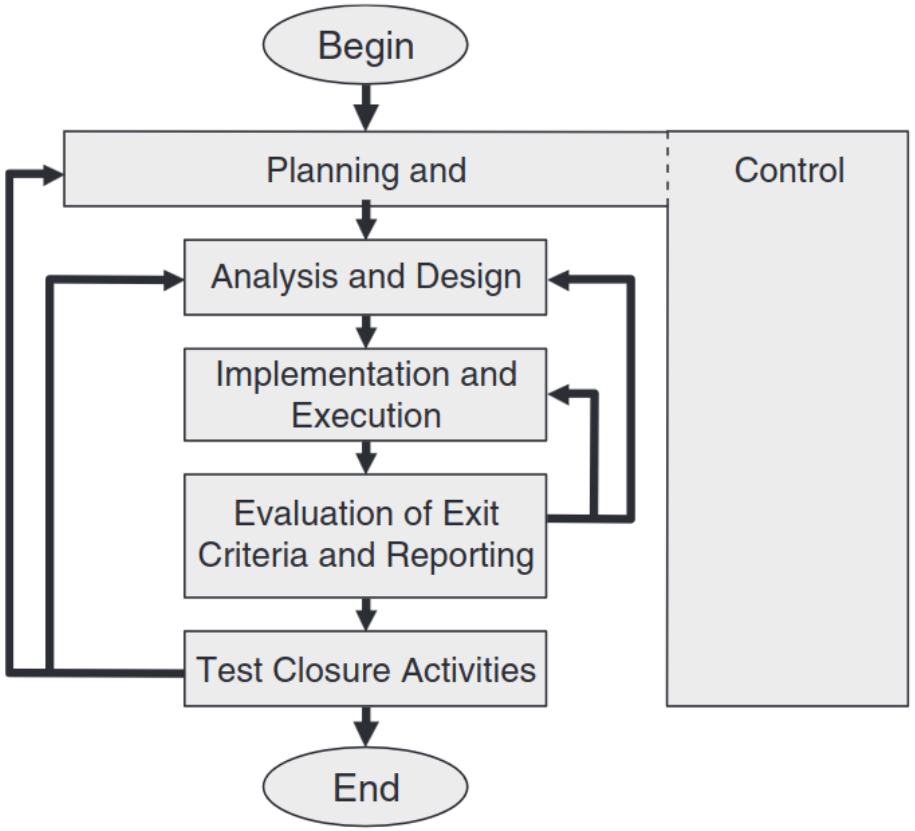
Fundamentals of the test process

Test Process Fundamentals



Test Planning and Control

- Planning of the test process starts at the beginning of project.
- The mission and objectives of testing must be defined.
- resources necessary for the test process.
- Which employees are needed for the execution of which tasks.
- The main task of planning is to determine the test strategy or Approach.

Test Planning and Control

- Test control is the monitoring of the test activities.
- includes reporting the status of deviations from the plan and taking any actions.

Test Strategies

- Test objectives:

It defines the specific goals and outcomes you want to accomplish during the testing phase of a project.

Example: Verify that the login functionality works correctly for both registered and new users.

- Scope:

Defines the scope of testing, outlining what will be tested and what will not be tested.

- Testing types:

Functional testing, non functional testing.

Test Strategies

- Test environment:

Outlines the hardware, software, and infrastructure required for testing.

Example: browser, devices and network.

- Test data:

Specifies the test data that will be used, how it will be generated.

- Test execution schedule:

Example of Test Strategy

- The VSR-System consists of the following subsystems:

1. DreamCar allows the individual configuration of a car and its extra equipment
2. ContractBase manages all customer information and contract data.

- JustInTime implements the ability to place online orders (within the first expansion stage by the dealer).

4. EasyFinance calculates an optimal method of financing for the customer.

Test Intensity

- The intensity of testing depends very much on the test techniques that are used.
- Test coverage serves as a test exit criterion.
- Risk of failure when defining the test intensity.

Test Analysis and Design

1. Review the test basis:

- First task is reviewing the test basis i.e., the specification of what should be tested.
- The specification should be concrete and clear enough to develop test cases.
- The basis for the creation of a test can be the specification or architecture documents, the results of risk analysis, or other documents produced during the software development process

Test Analysis and Design

2. Check testability:

- This process includes checking the ease with which interfaces can be addressed (interface openness).
- the ease with which the test object can be separated into smaller, more easily testable units.
- The results of this analysis are also used to state and prioritize the test conditions.

Test Analysis and Design

3. Consider the risk:

- The test strategy determined in the test plan defines which test techniques shall be used.
- The test strategy is dependent on requirements for reliability and safety
- If there is a high risk of failure for the software, very thorough testing should be planned.

Test Analysis and Design

3. Tracebility is important:

- Ensure traceability between the specifications to be tested and the tests themselves.
- must be clear which test cases test which requirements

Logical and Concrete test cases

- Specification of the test cases takes place in two steps
- Logical test cases have to be defined first.
- After that, the logical test cases can be translated into concrete test cases
- The test basis guides the selection of logical test cases

Logical and Concrete test cases

- Test cases can be determined:

1) Black box test case design technique:

- Test cases can be determined from the test object's specification.
- Example: Testing the login functionality of a web application.

2) White box test case design technique:

- created by analyzing the source code. Example: statement coverage, branch coverage.

Test Case Structure

Each test contains:

1) Pre-conditions:

- Pre-conditions must be provided before the execution of the test case.
- Specifies specific set of conditions or circumstances that must be met before the test case can be executed.

2) Test data or test input:

- Refers to the specific values, conditions, and parameters that are provided to a software application or system during the execution of a test case.

Test Case Structure

Each test contains:

3) Test oracle:

- A test oracle is a mechanism for predicting the expected results.
- The specification can serve as a test oracle.

Example of test cases

- Using the sales software, the car dealer is able to define discount rules for his salespeople: With a price of less than $15.000, no discount shall be given. For a price of $20.000, 5% is OK. If the price is below $25.000, a 7% discount is possible. For higher prices, 8.5% can be granted.

- From this, the following cases can be derived:

1. Price < 15.000 discount = 0%
2. 15.000 < price < 20.000 discount = 5%
3. 20.000 < price < 25.000 discount = 7%
4. Price < 25.000 discount = 8.5%

Example of test cases

Logical test case	1	2	3	4
input value x (price in dollar)	$x < 15000$	$15000 \leq x \leq 20000$	$20000 < x < 25000$	$x \geq 25000$
predicted result (discount in %)	0	5	7	8.5

Concrete test case	1	2	3	4
input value x (price in dollar)	14500	16500	24750	31800
predicted result (discount in %)	0	825	1732.50	2703

Test Implementation and Execution

Test Implementation:

- logical test cases must be transformed into concrete test case.
- These test cases can then be used without further modifications or additions for executing the test.

Test Execution:

- Decide how the tests will Test case execution be executed.
- The priority of the test cases must be taken into account.
- The test cases should also be grouped into test suite for efficient testing
- Test harness must be programmed, built as part of the test environment

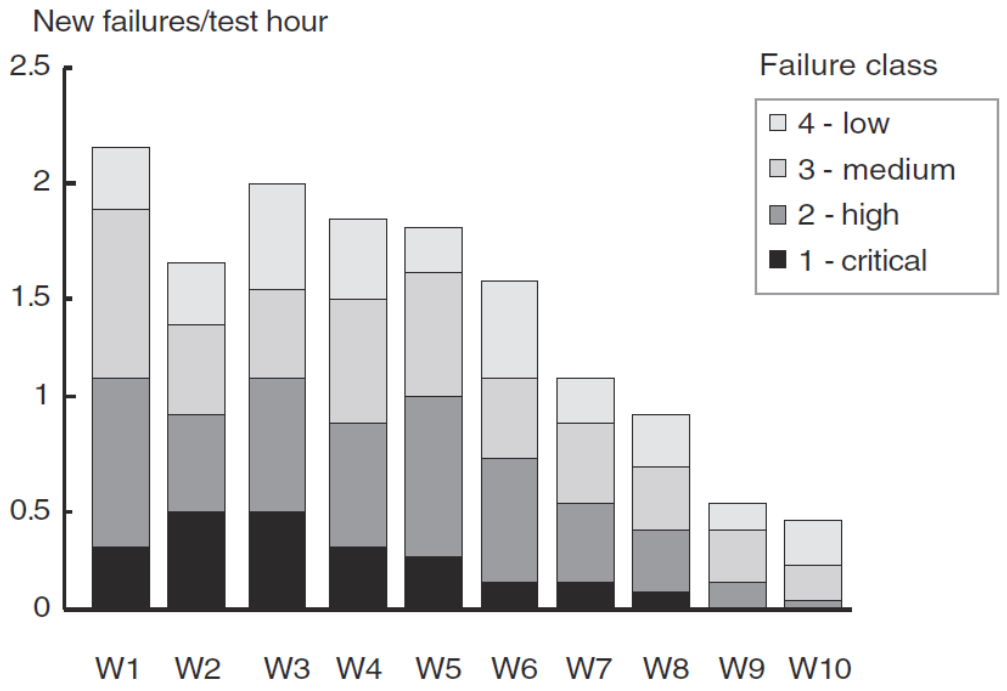
Test Implementation and Execution

Test Execution:

- The parts to be tested are checked for completeness.
- examination of the test object's main functionality a.k.a smoke test
- Test execution must be exactly and completely logged.
- The information related to a test case or test run must be maintained.
- Check if it is failure bound.
- Correction may lead to new faults.

Test Evaluation and Reporting

- the test object is assessed against the set test exit criteria specified during planning.
- adequate exit criterion must be determined for each test technique used.
- Failure rate is a possible criteria to define end of test.



Psychology of Testing

- Can the developer test his own program?
- The main weakness of developer tests is that developers who have to test their own programs will tend to be too optimistic.
- Blindness to one's own mistakes.
- Independent test team: independent testing team is beneficial for test quality
- The tester must report the failures and discrepancies observed to the author and/or to management.

Mutual knowledge of their respective tasks