



# Manual Testing – Part 2: Testing Artifacts

From Test Plan to Test Execution—The Documents  
That Define QA



# What are Testing Artifacts?

Artifacts are documents created during the Software Testing Life Cycle (STLC) to plan, design, execute, and track test activities.



## Common artifacts:

- Test Case
- Test Scenario
- Test Plan
- Test Strategy
- Traceability Matrix (RTM)
- Test Data
- Test Environment setup

# Test Case – Core Artifact

**Definition:** A Test Case is a detailed document that outlines the steps, inputs, and expected result to verify a particular functionality.

## Test Case Format:

Field	Example
Test Case ID	TC\_001\_Login\_Valid
Module	Login
Test Scenario	Login with valid credentials
Pre-conditions	User should be registered
Steps	1. Open login page 2. Enter valid email and password 3. Click login
Expected Result	Dashboard should load
Actual Result	Dashboard loaded
Status	Pass

# Test Scenario vs Test Case

Point	Test Scenario	Test Case
Meaning	High-level functionality to test	Step-by-step validation
Detail level	Broad idea	Specific instructions
Example	“Test Login Feature”	“Login with valid email and password”
Use case	When time is short or in agile	For complete, accurate testing

# **Test Plan – Project Testing Blueprint**

**What is a Test Plan?** It's a document that defines what will be tested, how, who, and when. Written by the Test Lead or QA Manager.

## **Example Content:**

- Scope of testing
- Testing types (Functional, Regression, UI)
- Entry/Exit criteria
- Roles & responsibilities
- Deliverables
- Tools (Selenium, TestNG)
- Risk & mitigation

**Live Example:** On our automation project, we created a test plan for the login & signup module — defining the timeline, resources, and scenarios.



# **Test Strategy – Org-Level Guide**

**What is a Test Strategy?** It's a high-level approach used across all projects. Defines how testing will be done at an organization level.

## **Includes:**

- Types of testing used (Manual, Automation, Performance)
- Test levels (Unit, Integration, System)
- Tool stack (Selenium, JMeter, Jenkins)
- Team roles and communication protocols

 Created once per organization or large project.

# RTM – Requirement Traceability Matrix

**Definition:** A document that maps requirements to test cases to ensure full test coverage.

## **Format:**

Requirement ID	Description	Test Case ID
REQ-01	User Login	TC\_001, TC\_002
REQ-02	User Signup	TC\_003, TC\_004

## **Why it matters:**

- Shows test coverage
- Helps during impact analysis
- Detects missed test cases



# Test Data & Test Environment

 **Test Data:** → Sample inputs used to test the application (emails, passwords, card numbers, etc.)

 **Example:** email: testuser@mail.com  
password: Test@123

 **Test Environment:** → The setup where testing is done. Includes:

- OS, browser
- Test server
- Application build
- DB config

**Live Example:** I tested our signup feature in the UAT environment with Chrome 120.0



# Interview Questions + Answers

## 1. What is a Test Case and what are its components?

A test case is a document that contains steps, input, expected result to verify a feature.

**Key components:** Test Case ID, Steps, Expected/Actual Result, Status, Remarks.

## 2. What is the difference between a Test Plan and a Test Strategy?

Test Plan	Test Strategy
Project-level document	Organization-level document
Written by Test Lead	Written by QA Manager
Defines what to test, who tests, schedule	Defines tools, levels, and approach
Changes from project to project	Static, rarely changes

## 3. Why is Traceability Matrix important?

It ensures that all requirements are covered by test cases. It helps track which tests are affected if a requirement changes and shows testing completeness.

## 4. What is a real-world example of using test data and environment?

In our signup test, I used multiple emails and phone numbers in UAT to test different edge cases across browsers.

# Thank You!

We hope this deep dive into Testing Artifacts helps you master QA documentation.

Questions?