# Test Levels and Test Types

**# Source**: **Chatgpt (GPT-5-Standard) at [8/14/2025]**

Alright — let’s go deep into **Test Levels** and **Test Types** because these are fundamental in understanding how testing is structured in software development.

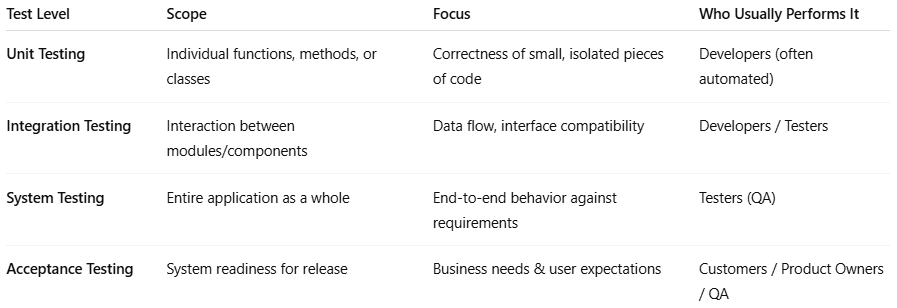
## **1. Test Levels — *"When and where we test in the development cycle"***

**# Source**: **Chatgpt (GPT-5-Standard) at [8/14/2025]**

Test levels describe the **scope** and **stage** of testing. Each level focuses on a different part of the system.

### **Main Test Levels**

**# Source**: **Chatgpt (GPT-5-Standard) at [8/14/2025]**



|  |  |  |  |
| --- | --- | --- | --- |
| **Test Level** | **Scope** | **Focus** | **Who Usually Performs It** |
| **Unit Testing** | Individual functions, methods, or classes | Correctness of small, isolated pieces of code | Developers (often automated) |
| **Integration Testing** | Interaction between modules/components | Data flow, interface compatibility | Developers / Testers |
| **System Testing** | Entire application as a whole | End-to-end behavior against requirements | Testers (QA) |
| **Acceptance Testing** | System readiness for release | Business needs & user expectations | Customers / Product Owners / QA |

### **Details of Each Level**

**# Source**: **Chatgpt (GPT-5-Standard) at [8/14/2025]**

1. **Unit Testing**
   * Smallest testable parts of the application.
   * Example: Testing a calculateTotal() function.
   * Tools: JUnit, NUnit, pytest.
2. **Integration Testing**
   * Checks whether different parts/modules work together.
   * Can be:  
     + *Big Bang* (all modules together)
     + *Incremental* (step-by-step integration)
   * Example: Verifying if the payment module correctly interacts with the order module.
3. **System Testing**
   * Tests the **whole system** in an environment close to production.
   * Includes functional and non-functional checks.
   * Example: Testing an entire e-commerce site from browsing products to checkout.
4. **Acceptance Testing**
   * Confirms the system meets business requirements.
   * Often written in **business language** (e.g., Gherkin for BDD/ATDD).
   * Example: "Given I add a product to the cart, When I check out, Then I should see a confirmation email."

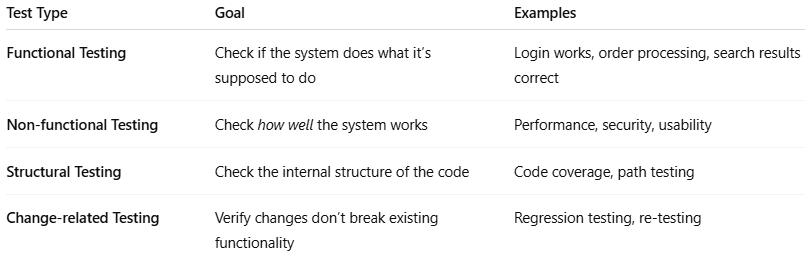
## **2. Test Types — *"What kind of quality aspect we are testing"***

**# Source**: **Chatgpt (GPT-5-Standard) at [8/14/2025]**

Test types describe the **purpose** or **goal** of a test, regardless of when it happens.

### **Main Test Types**

**# Source**: **Chatgpt (GPT-5-Standard) at [8/14/2025]**



|  |  |  |
| --- | --- | --- |
| **Test Type** | **Goal** | **Examples** |
| **Functional Testing** | Check if the system does what it’s supposed to do | Login works, order processing, search results correct |
| **Non-functional Testing** | Check *how well* the system works | Performance, security, usability |
| **Structural Testing** | Check the internal structure of the code | Code coverage, path testing |
| **Change-related Testing** | Verify changes don’t break existing functionality | Regression testing, re-testing |

### **Details of Each Type**

**# Source**: **Chatgpt (GPT-5-Standard) at [8/14/2025]**

1. **Functional Testing**
   * Validates features against functional requirements.
   * Includes:  
     + Smoke Testing
     + Sanity Testing
     + Integration Testing
     + System Testing
     + User Acceptance Testing
2. **Non-functional Testing**
   * Measures quality attributes like performance, usability, security.
   * Examples:  
     + Load Testing (how it handles many users)
     + Stress Testing (limits of the system)
     + Security Testing (vulnerability scanning)
     + Usability Testing (ease of use for end users)
3. **Structural Testing**
   * White-box testing that looks inside the code.
   * Example: Ensuring all branches of a decision are tested.
4. **Change-related Testing**
   * **Regression Testing** → Check that new changes didn’t break old features.
   * **Re-testing** → Check that a previously failed test now passes after a fix.

### How Test Levels and Test Types Work Together

**# Source**: **Chatgpt (GPT-5-Standard) at [8/14/2025]**

* **Test levels** tell you **when and where** you test in the lifecycle.
* **Test types** tell you **what** you’re testing for.

Example:

At the **integration test level**, you might do **functional testing** (does the payment work with order processing?) and **non-functional testing** (is it fast enough?).

## 3-Testing Quadrants

### 📌 Agile Testing Quadrants (Testing Quadrants)

**# Source**: **Chatgot (GPT-5-Standard) at [8/18/2025]**

The **Agile Testing Quadrants** is a model created by **Brian Marick** and later refined by **Lisa Crispin & Janet Gregory**.  
 It helps teams understand **different types of tests** needed to deliver quality software.

The quadrants are shown as a **2D matrix** with two axes:

* **Horizontal Axis (Purpose):**
  + **Support the team** (left) vs. **Critique the product** (right).
* **Vertical Axis (Type):**
  + **Business-facing** (top) vs. **Technology-facing** (bottom).

### 🔹 The Four Quadrants

#### Q1: Technology-facing tests that support the team

**# Source**: **Chatgot (GPT-5-Standard) at [8/18/2025]**

* **Goal:** Verify code at the technical level.
* **Examples:** Unit Tests, Component Tests, Automated API tests.
* **Who:** Mostly developers (with tester collaboration).
* **Tools:** JUnit, pytest, NUnit, TestNG.

#### Q2: Business-facing tests that support the team

**# Source**: **Chatgot (GPT-5-Standard) at [8/18/2025]**

* **Goal:** Confirm the system behaves as the **business expects**.
* **Examples:** Functional Tests, Story Tests, Example-based Tests, BDD scenarios.
* **Who:** Testers + Product Owners + Developers.
* **Tools:** Cucumber, FitNesse, SpecFlow.

#### Q3: Business-facing tests that critique the product

**# Source**: **Chatgot (GPT-5-Standard) at [8/18/2025]**

* **Goal:** Explore the product to discover missing functionality or issues.
* **Examples:** Exploratory Testing, Usability Testing, User Acceptance Testing (UAT), Alpha/Beta testing.
* **Who:** Testers, End-users, Product Owners.
* **Approach:** Often manual, though some automation may assist.

#### Q4: Technology-facing tests that critique the product

**# Source**: **Chatgot (GPT-5-Standard) at [8/18/2025]**

* **Goal:** Evaluate the system’s **non-functional qualities**.
* **Examples:** Performance Testing, Security Testing, Load/Stress Testing, Scalability, Maintainability.
* **Who:** Testers + Specialists (e.g., performance engineers, security experts).
* **Tools:** JMeter, Gatling, OWASP ZAP.

### Is Related to Test Levels and Test Types

**# Source**: **Chatgot (GPT-5-Standard) at [8/24/2025]**

Yes, **Testing Quadrants** are related to the topic of **Test Levels** and **Test Types**, but they are **not the same thing**. Here’s how they relate:

#### ✅ Relationship with Test Levels

**# Source**: **Chatgot (GPT-5-Standard) at [8/24/2025]**

* **Test Levels** (Unit, Integration, System, Acceptance) define **WHEN** testing happens in the development lifecycle.
* Quadrants (Q1–Q4) focus on **WHAT types of tests** and **WHY** they are performed.
* Example:  
  + **Q1 (Unit Tests)** often aligns with the **Unit Testing level**.
  + **Q2 (Functional Tests)** aligns with **System or Acceptance Testing**.

#### ✅ Relationship with Test Types

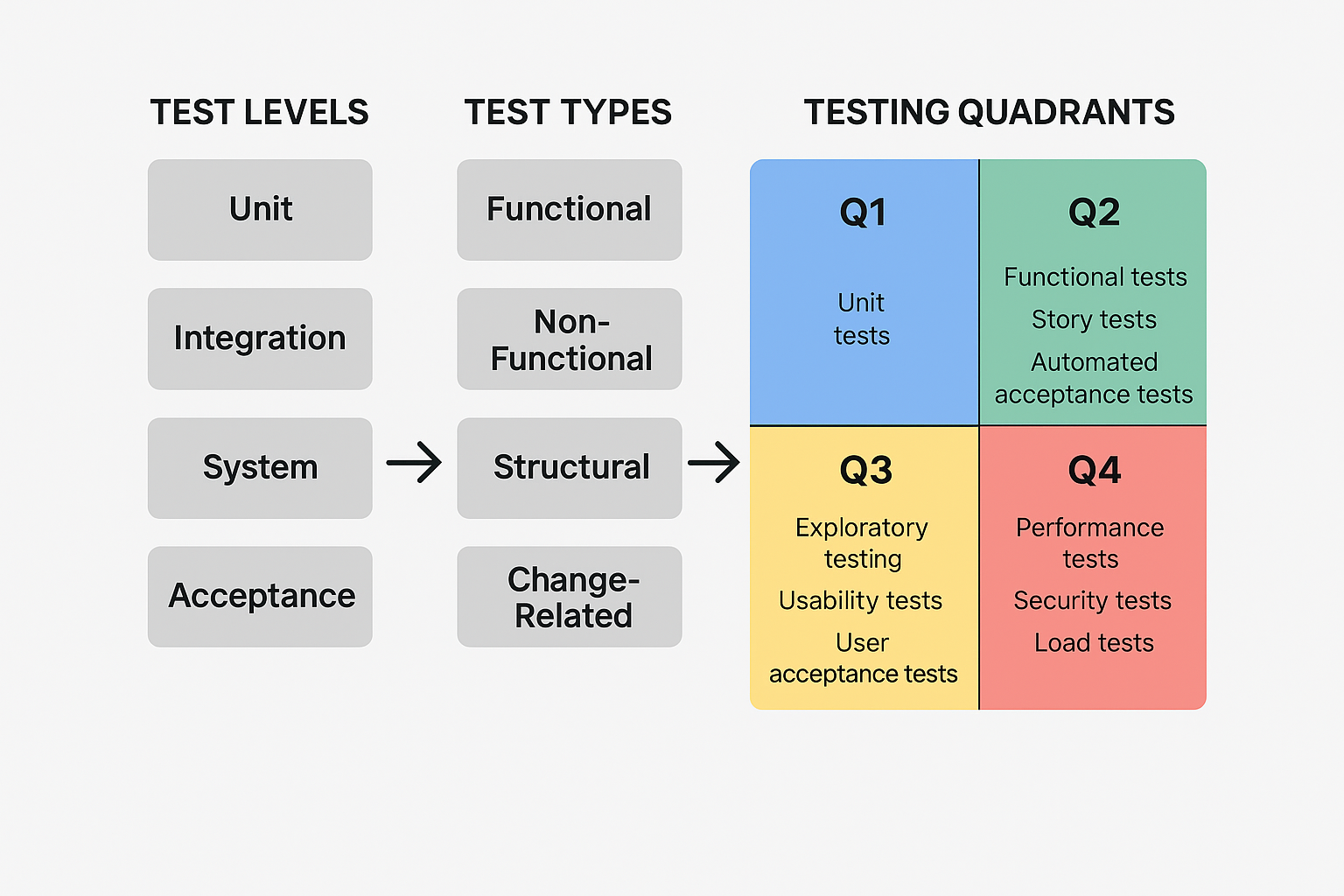
**# Source**: **Chatgot (GPT-5-Standard) at [8/24/2025]**

* **Test Types** (Functional, Non-Functional, Structural, Change-Related) define the **nature of the tests**.
* Quadrants include both **functional** and **non-functional** test types:  
  + **Q1 & Q2** → Mostly **functional testing** (technology-facing & business-facing).
  + **Q3** → Includes **usability testing, exploratory testing**.
  + **Q4** → Includes **performance testing, security testing** (non-functional).

#### visual diagram showing Test Levels, Test Types, and Testing Quadrants

**# Source**: **Chatgot (GPT-5-Standard) at [8/24/2025]**

**visual diagram showing Test Levels, Test Types, and Testing Quadrants to clarify their relationship**



#### ✅ Summary

**# Source**: **Chatgot (GPT-5-Standard) at [8/24/2025]**

* **Test Levels = Where in the lifecycle**
* **Test Types = What aspect is tested**
* **Testing Quadrants = A planning tool to balance business & technology focus across test types**

### 📌 Visual Summary

**# Source**: **Chatgot (GPT-5-Standard) at [8/18/2025]**



Business-Facing

+---------------------------------+

| Q2: Functional, BDD, Examples | Support the Team

| Q3: Exploratory, UAT, Usability | Critique the Product

+---------------------------------+

| Q1: Unit, Component, API | Support the Team

| Q4: Performance, Security, NFR | Critique the Product

+---------------------------------+

Technology-Facing

✅ **In short:**

* **Q1 & Q2** → Help the **team build the right product** (preventing defects).
* **Q3 & Q4** → Help the **team evaluate the product** (finding defects, risks).
* Together, the quadrants ensure **both functional & non-functional quality**.