# Test Design Template

Test design for feature.

Interface 1: Name of UI Element, Page or API Endpoint here.

This data below is a sample, replace with valid values for your needs.

* Black Box Testing
  + Idea: You test the software from the outside – like a user would.
  + You don’t look at the code, you just give inputs and check outputs.
  + You don’t care *how* it’s implemented inside; you only care *what result* you get.
  + you need clear and detailed requirements (or at least acceptance criteria) to do proper black box testing.
    - Why requirements are important for black box testing.
      * In black box testing, you don’t see the code.
      * You rely on what the system is supposed to do (requirements, user stories, ACs, design docs).
      * Without requirements, you can only do exploratory testing (based on your intuition and product knowledge), but you can’t confirm if the output is correct or just what you happen to expect.
  + Example:
    - Imagine you’re testing a login form.
      * You enter username + password and click Login.
      * If valid → dashboard opens.
      * If invalid → error message.
    - You don’t check how the password is stored, hashed, or validated internally.

Black box = behavior-focused testing (functional, system, regression, UAT, etc.).

Black box Testing Techniques

1. Equivalence class testing
2. Boundary value testing
3. Decision table testing
4. State transition testing

|  |  |  |
| --- | --- | --- |
| Input Parameter | Valid equivalence classes | Invalid equivalence classes |
| clientName (String) | Any text of valid URI characters (Validate with the Developer that there’s no length constraints) | None |
| Priority (String) | Low, Medium, High | Anything else |

Criteria & Constraints

|  |  |
| --- | --- |
| Criteria 1 | Input must be valid URL input as defined by RFC 3986: Uniform Resource Identifier (URI): Generic Syntax |
| Constraint 1 | In the case where the base URL contains invalid characters you will get an error because you cannot reach the web server, not the API we’ve built.  When the query parameters have invalid characters, then the web server will return an error code (part of the standard HTTP Status Codes). https://status.js.org/ |
|  |  |
| Criteria 2 |  |
| Constraint 2 |  |
|  |  |
| Criteria 3 |  |
| Constraint 3 |  |

* White Box Testing
  + Idea: You test the software from the inside – you know the code, structure, and logic.
  + You design tests by looking at how the code works internally.
  + You check execution paths, conditions, loops, and data flow.
  + Example:
    - For the same login function:
      * You write unit tests to check if password hashing is done correctly.
      * You test what happens if the database returns null.
      * You test different code branches (like if/else logic).

White box = *implementation-focused testing* (unit testing, code coverage, path testing, etc.).

White box Testing Techniques

1. Statement coverage
2. Decision coverage
3. Branch coverage
4. Condition coverage