# LLM Guardrails: A Field Guide for Professional Testers

*Companion to the Guardrails Matter game — TestingTitbits.com*

*Created by Rahul Parwal*

**Table of Contents**



## What This Is

This guide extracts the professional testing knowledge embedded in the Guardrails Matter game. Play the game for pattern recognition under pressure. Use this guide before and after engagements to build the mental model behind the mechanics.

# The Core Model: Guardrails Are Not Model Properties

***The single most important concept in LLM testing:***

> *The LLM itself has no knowledge of your product's rules. Guardrails are engineered constraints layered around the model.*
>
> *If those layers are absent or misconfigured, the model behaves as it was designed to: helpfully, without your constraints. That is not a bug in the model. It is a gap in the system you are testing.*

# Part 1 — The Five Guardrail Categories

Every guardrail failure belongs to one of five system layers. Misclassifying a failure in a defect report wastes time, sends it to the wrong team, and obscures patterns in your defect log.

| Layer | Description |
|---|---|
| Input | Intercepts and sanitises user prompts before the model processes them |
| Output | Validates model responses before they reach the user |
| Behavioural | System prompt constraints governing identity, scope, and conduct |
| Operational | Rate limits, token caps, session controls |
| Regulatory | Compliance obligations such as GDPR, EU AI Act, HIPAA |

## Why Classification Matters in Practice

When you write vague defects, they go nowhere. When you classify precisely, they reach the right owner with a fixable description.

# Part 2 — Threshold Calibration

Guardrails are not binary switches. They are calibrated thresholds.

## Failure Modes

| Condition | Result |
|---|---|
| Fires too early | false positive |
| Fires too late | false negative |
| Never fires | miss |

## Testing Calibration

| Step | Action | Boundary |
|---|---|---|
| 1 | Validate exact trigger point | N |
| 2 | Validate non-trigger before threshold | N−1 |
| 3 | Validate still triggers after threshold | N+2 |

# Part 3 — The Five Attack Patterns

| Pattern | Name |
|---|---|
| A | Direct Instruction Override |
| B | Persona Replacement |
| C | Fictional / Narrative Wrapper |
| D | Multi-Turn Gradual Escalation |
| E | Language Shift / Encoding |

*Key principle: You cannot test what you cannot name.*

# Part 4 — System Prompt as Test Specification

A system prompt is a functional specification. Every clause must be testable.

## A Testable Clause Includes

- Trigger condition
- Expected behaviour
- Optional exact response

If you cannot derive a test case, the clause is weak.

# Part 5 — Guardrail Implementation Verification

A guardrail not enforced in code is not a guardrail.

## Common Failure Patterns

- Save-before-check in rate limiting
- Incomplete regex in PII masking
- Shared input/output blocklists

Testers should focus on logic, not syntax.

# Putting It Together

## Before Testing

- Review system prompt
- Review implementation
- Prepare adversarial cases

## During Testing

- Validate all guardrail layers

- Test all attack patterns
- Check calibration boundaries

### Defect Reporting

- Identify layer
- Identify attack pattern
- Identify spec gap or implementation gap

## Glossary

| Term | Definition |
|---|---|
| Input guardrail | Filters user input |
| Output guardrail | Filters model output |
| Behavioural guardrail | Governs AI behaviour |
| Operational guardrail | System-level controls |
| Regulatory guardrail | Compliance requirements |