

Release, Migration, and Incident Quality

Lesson 10 · Make the safest justified decision

Today's outcome

- Read release evidence critically
- Understand artifact/migration/environment alignment
- Make ship, hold, rollback recommendations
- Communicate incidents without invented root causes

A release is a chain of proof

Source/commit → immutable build → migration → candidate environment → required QA → promotion/canary → post-release smoke

The evidence must name the same candidate all the way through.

Required means fail closed

State	Release meaning
Passed	evidence supports this check
Failed	blocker until resolved/controlled
Skipped/cancelled/missing	not a pass
Advisory	useful signal, not sole gate

Migration risk

Test:

- clean replay from zero
- schema/application compatibility
- tenant/RLS behavior
- data backfill limits
- rollback/recovery plan
- environment drift

Demo: a misleading green page

The public page renders.

The authenticated API call fails.

What is the correct release conclusion?

Incident language

Facts: what we observed

Impact: who/what is affected

Unknowns: what we have not proven

Containment: what is safe now

Next evidence: how we learn more

Lab: Ship/No-Ship Release Room

Three evidence rounds. Update your recommendation each time.

Deliver: release recommendation and incident update.

Break-glass is not “ignore the test”

It requires a named approver, reason, incident record, and mandatory follow-up. It does not change failed evidence into passing evidence.

Exit ticket

What exact evidence must exist before you promote a candidate?