

Guide animateur — CI/CD & Qualité (1h30)

Préparation (J-3 à J-1)

- Collecter info pipeline actuel : temps moyen, taux échec, scripts.
- Identifier douleurs : lint absent, flaky tests, docs obsolètes.
- Préparer un exemple pipeline (GitHub/GitLab) propre.
- Vérifier que participants ont accès au repo ou à une sandbox.

Agenda détaillé

Temps	Thème	Livrables	Risques / Parades
00:00-00:10	Contexte, enjeux	Slide ROI, coût non qualité	Discussions vagues → montrer métriques actuelles
00:10-00:30	Lint & conventions	Script lint/format + hooks	Résistance "trop strict" → distinguer blocking/warning
00:30-01:00	Stratégie tests	Matrice tests, coverage utile	Pyramide non comprise → fournir exemples concrets
01:00-01:20	Pipeline build/test	Pipeline minimal (cache, artifacts)	Pipeline trop lent → fixer objectif < 5 min
01:20-01:30	Quality gates	Checklist revue, DoD	Bloque PR → insister sur quick wins

Animation

Introduction (10')

- Visualiser coût non qualité : régression prod, temps patch.
- But : pipeline rapide, fiable, lisible.

Lint & conventions (20')

- Démo scripts npm + Husky (diff blocking/warning).
- Faire compléter exercice 01 (scripts + hooks).

Tests (30')

- Décrire pyramide : unit (fast) / integration (targeted) / e2e (happy path).
- Critique : coverage utile, tests flaky, budget QA.
- Exercice 02 : matrice tests + seuil coverage.

Pipeline (20')

- Walkthrough pipeline type (jobs, cache, matrix, artifacts).
- Mesurer temps actuel vs objectif.
- Exercice 03 : pipeline minimal.

Quality gates (10')

- Revue : checklist actionnable (lisibilité, perf, a11y).
- DOR/DoD : docs à jour, tests, story.

- Exercice 04 : quality gates.

Anti-patterns (à pointer)

- Monolithes jobs (lint/test/build dans un seul job).
- Coverage 100% imposé (vanity).
- Hooks pre-commit trop lents.
- Docs obsolètes (README jamais mis à jour).

Quick wins suggérés

- Prettier + lint-staged → homogénéité.
- Tests snapshot + coverage minimal par module.
- Pipeline cache npm + build, artifacts coverage.
- Template PR + checklist revue.

Post-session

- Partager pipeline type, checklist CI, guide revue.
- Identifier “champion CI/CD” pour suivi.
- Planifier rétrospective pipeline après 2 sprints (mesures).