

Pourquoi CI/CD ? (10')

- Réduire le coût caché des releases manuelles.
 - Capitaliser sur les conventions : même pipeline pour tous.
 - Fournir des feedbacks rapides, mesurables (minutes, pas heures).
-

Objectifs du workshop

1. Lint & format : code homogène, pre-commit efficace.
 2. Stratégie de tests : pyramide utile, coverage qualité.
 3. Pipeline build/test : rapide, fiable, lisible.
 4. Quality gates : critères PR, DOR/DoD réalistes.
-

Lint & format (20')

- ESLint / Stylelint / Prettier / Ng lint...
- Pre-commit (Husky, Lefthook), cache local.
- Règles blocking vs warning (ex : `no-floating-promises`).

Tableau : `package json scripts` (lint:base, lint:fix, format:write).

Exercice 01 · Linting & conventions

Objectif : définir règles + hooks + auto-format.

- Scripts npm (lint, lint:fix, format, format:check).
 - Husky pre-commit (lint-staged) ou alternatif.
 - Convention naming, import order, commit lint.
-

Stratégie de tests (30')

Pyramide : unit > integration > e2e. Couverture "utile" :

- Prioriser domaines critiques (domestique 80%+).
 - Éviter vanity metrics (100% meaningless).
 - Gérer flaky tests (retry, quarantine).
-

Exercice 02 · Testing strategy

Livable : matrice tests (unit/integration/e2e), owners, cadence.

- Choisir coverage cible (ex : 70% global, 90% sur modules critique).
 - Définir critères ajout test (PR template).
 - Traquer flaky tests (tag `@flaky` , monitoring).
-

Pipeline build/test (20')

- Caching npm/pnpm, bundlers, Cypress.
 - Matrix OS/node (si besoin). Skip inutile.
 - Artifacts : coverage, storybook, report lint.
-

Exercice 03 · Pipeline minimal

Tâches :

- Décrire jobs `lint` `test` `build` .
 - Ajouter cache (npm + build).
 - Générer artifacts (coverage, storybook).
 - Calculer temps total cible (< 5 min).
-

Quality gates (10')

- PR checks bloquants : lint, test, build, review.
 - Checklist revue (lisibilité, tests, perf, accessibilité).
 - Documentation vivante (README, ADR, runbook).
-

Exercice 04 · Quality gates

Livrable : tableau DOR/DoD, critères PR.

- Exemple : commits signés, tests mis à jour, docs.
 - Rôles : reviewer tech, QA, product.
 - Mesures : % PR bloquées, temps cycle.
-

Synthèse & plan d'action

Livrables :

- Pipeline type + scripts.
- Guide revue & docs template.
- Checklists CI & Qualité prêtes.

Next step : pilot sur repo critique, mesure temps pipeline.