



Règles Codex – v1 (issues Audit R2)

But : règles prescriptives minimales pour exécuter les tickets Arka sans improvisation. **À lier dans chaque ticket** via :

Règles Codex :

`arka-meta/codex_rules/codex_rules_current.md`

1) Branché ou rien

- Aucune livraison **Storybook-only**. Une US UI est **DONE** si la **page** et la **route** sont présentes et **branchées** aux API prévues.
- Parcours fumée obligatoire : `/` → `/login` → `POST /api/auth/login` → `/console`.

2) Contrats stricts

- Respect exact des codes/réponses listés dans les tickets/specs.
Ex. Webhook : pas de `204/500` si non prévus ; `ratelimit` = `202` ou `403` selon spec.

3) Logs JSON partout

- Sur routes clés (login/projects/agents/health/webhook) : `{ts, level, msg, route, status, ...}`.
- Évidence : extraits de logs inclus dans `audit_synthese.md`.

4) RBAC appliqué & visible

- Middleware côté back **et** états UI (viewer RO, operator limité, owner total).
- Évidence : tests table-driven (route × rôle) + capture UI.

5) Sécurité basique

- **Aucun secret par défaut.** Démarrage interdit si secret manquant (fail-fast).
- HMAC/GitHub : `403` si invalide ; allowlist obligatoire.

6) Performance & A11y

- Budgets : LCP $\leq 2.5s$ (`/`, `/login`), TTI $\leq 2s$ (`/console`), CLS < 0.1 , niveau **AA**.
- Évidence : rapport Lighthouse/CI joint.

7) Tests & Schémas

- Unitaires + E2E (login→console).
- Contrats JSON via schémas (AJV ou équivalent).
- Rapport de tests exporté.

8) Arbo & Routage

- Présence des pages : `/`, `/login`, `/console`, sous-routes (`/console/chat|documents|prompt-builder|observabilite`).
- Zéro doublon incohérent (`types.d.ts`, imports). Nettoyage requis.

9) Clarification obligatoire

- Si un point est flou ou manquant dans un ticket : **ne pas boucher les trous**.
- Poser 1 question courte et bloquer jusqu'à réponse.

10) Livrables normalisés

- Respect des noms/chemins demandés dans tickets et cadrage (reports, patches, synthèse).

- **Toujours** livrer `audit_synthese.md` + mettre à jour `codex_rules_current.md`.
-

✓ Évidence minimale à fournir par ticket

- Capture/parcours (si UI), extrait logs, liens fichiers : source:ligne.
 - Tests PASS pertinents (unit/E2E/contrats).
 - Mention du rôle testé (viewer/operator/owner quand applicable).
-

11) Compte-rendu de livraison Codex (obligatoire)

- ⚠ **Obligation Codex** : produire en sortie un fichier **CR-Codex.yaml** conforme au template `arka-meta/backlog/templates/TPL-CR-Codex.yaml`.
- **Sans ce CR rempli → la livraison est considérée invalide.**
- **Emplacement** : `arka-meta/reports/codex/CR-<TCK-XXXX-YY>.yaml`

Template attendu (extrait) :

- Codex-Delivery-Report:
- ticket_id: <TCK-XXXX-YY>
- lot: <M3>
- version: <R1-M3 v1.1>
- repo: <arka-console | arka-backend>
- branch: <feature/TCK-XXXX>
- commit: <sha>
-
- delivered_components:
- - "path/to/file1"
- - "path/to/file2"
-
- tests_local:
- passed: true|false
- coverage:
- lines: <int> # %
- branches: <int> # %
-

- perf_local:
- chat_ttft_ms_p95: <int>
- voice_rtt_ms_p95: <int>
- api_latency_ms_p95: <int>
-
- security_local:
- hmac: true|false
- idempotence: true|false
- logs_trace_id: true|false
-
- evidence:
- - "screenshot_*.png"
- - "logs_*.json"
- - "lighthouse_*.pdf"
-
- remarks:
- - "Limites connues ou points à surveiller"
-
- next_step: "QA vérifie preuves + relaye AGP"
- ts: YYYY-MM-DDThh:mmZ

Versionnement des règles

- Fichier courant : `codex_rules_current.md` (pointe vers la dernière vX).
- Nouveau cycle d'audit ⇒ `codex_rules_v{n+1}.md` + changelog en tête.