OpenAct Implementation Plan (v1.0)

- Branch: `feature/openact-design-spec-v1`
- Rule: Implement one task at a time. For each task: add tests → make it pass → mark done → proceed.
- Status tags: 未实现, 进行中, 已完成

0. Conventions

- Config format: YAML (default)
- Expression engine: jsonada with `{% %}` syntax (per spec)
- Merge order: provider → action → sidecar (deep merge; arrays replace)

1) 配置与加载 (Configuration & Loading)

- ☐ Decide config file locations and naming — 未实现

    - Deliverable: finalize paths
        - `config/provider-auth-defaults.yaml`
        - `config/provider-defaults.yaml`
        - `config/sidecar-overrides.yaml` (optional)
        - `actions/<provider>/<action>.yaml`
    - Test: N/A (docs reviewed)
    - Done when: paths fixed in README and used by loader

- ☐ Implement YAML loader utility — 未实现

    - Deliverable: `utils/yaml_loader.rs`
    - Test: load valid/invalid YAML, error includes filename and line
    - Done when: returns `serde_json::Value` for valid files; precise errors for invalid ones

- ☐ Load provider-auth-defaults.yaml into registry — 未实现

    - Deliverable: `config/provider_auth_defaults.rs`
    - Test: load sample; query `api.github.com` returns injection template
    - Done when: cached by hostname (no hot-reload initially)

- ☐ Load provider-defaults.yaml into registry — 未实现

    - Test: `api.github.com` has retry/timeout defaults
    - Done when: accessible via provider key

- ☐ Load sidecar-overrides.yaml into registry (optional) — 未实现

    - Test: override by `operationId` merges on top
    - Done when: merges applied at runtime

---

2) 配置模型与合并 (Models & Merge)

- ☐ Define AuthConfig and related structs per spec — 未实现

  - Files: `manifest/src/action/auth.rs`
  - Includes: `connection_trn, scheme, injection{type=jsonada,mapping}, expiry, refresh, failure`, retry, pagination, output
  - Test: serde (round-trip), defaulting rules
  - Done when: compiles and unit tests pass

- ☐ Parse x-auth from Action YAML — 未实现

  - Files: `manifest/src/action/parser.rs`
  - Test: minimal action with only `connection_trn`
  - Done when: parsed into `AuthConfig`

- ☐ Parse x-retry, x-pagination, x-timeout-ms, x-ok-path, x-error-path, x-output-pick — 未实现

  - Test: each field parsed; defaults applied
  - Done when: validated and surfaced in action model

- ☐ Implement deep-merge provider → action → sidecar — 未实现

  - Files: `manifest/src/config/merger.rs`
  - Test: object fields deep-merge; arrays replace; precedence respected
  - Done when: deterministic merged snapshot printed for debug

---

3) 表达式与注入 (Expressions & Injection)

- ☐ Add jsonada engine dependency / integration — 未实现

  - Files: `Cargo.toml` (manifest crate)
  - Test: trivial evaluation `{% 'x' %}` → "x"

- ☐ Implement ExpressionEngine for `{% %}` — 未实现

  - Files: `manifest/src/action/expression_engine.rs`
  - Test: mapping string → headers map; invalid expr → readable error

- ☐ Build expression context ($access_token, $expires_at, $ctx) — 未实现

  - Files: `manifest/src/utils/expression_context.rs`
  - Test: all expected keys present; timestamps as ISO8601

---

4) AuthFlow 集成 (AuthFlow Integration)

- ☐ Create AuthFlowIntegration to fetch by connection TRN — 未实现

  - Files: `manifest/src/action/authflow_integration.rs`

- Test: given `connection_trn`, returns `{access_token, token_type, expires_at, provider}`

- ☐ Support expiry/refresh strategies (proactive_or_401, proactive, on_401) — 未实现

  - Files: `manifest/src/action/auth.rs`
  - Test: simulate expired token; refresh invoked by strategy

---

5) Action Runner 执行 (Execution)

- ☐ Wire ActionRunner auth injection using mapping — 未实现

  - Files: `manifest/src/action/runner.rs`
  - Test: merged config → fetch auth → evaluate mapping → set headers

- ☐ Inject headers and query from mapping result — 未实现

  - Test: headers present on HTTP request; query params applied if provided

- ☐ Apply x-timeout-ms to HTTP client — 未实现

  - Test: short timeout triggers timeout error

- ☐ Implement x-retry with backoff and Retry-After — 未实现

  - Test: 500/503 retried; 429 respects Retry-After; jitter applied

- ☐ Implement pagination: cursor, pageToken, link — 未实现

  - Test: aggregates items until stop_when; respects cursor_param

- ☐ Evaluate x-ok-path and map success — 未实现

  - Test: ok-path true → success; null → 2xx treated as success

- ☐ Evaluate x-error-path and map errors — 未实现

  - Test: provider error extracted to standardized error

- ☐ Apply x-output-pick projection — 未实现

  - Test: projection applied to final payload

---

6) 测试与样例 (Tests & Samples)

- ☐ Unit tests: parser and config merger — 未实现

  - Test: cover edge cases (missing fields, overrides)

- ☐ Unit tests: expression engine and injector — 未实现

  - Test: mapping variations, invalid syntax, context values

- ☐ E2E golden test: GitHub Get User — 未实现

    - Test: action YAML + provider defaults → headers injected → response recorded

- ☐ Add sample Action YAML and providers YAML — 未实现

    - Files: `samples/actions`, `config/*.yaml`

- ☐ Add CLI lint tool for Action YAML — 未实现

    - Test: validates single operation, x-* schema, expression compile

- ☐ Add logging and traces for injection/retries — 未实现

    - Test: structured logs; correlation IDs; redacted secrets

- ☐ Document config layout and authoring guide — 未实现

    - Files: `docs/authoring.md`

里程碑建议 (Milestones)

- M1: 打通最小链路（加载→合并→取TRN→注入→请求）
- M2: 超时/重试/错误/输出裁剪
- M3: 分页与 E2E 测试
- M4: 工具化（Lint/Logs/Samples/Docs）