# **Lộ trình chuyên nghiệp để phát triển dự án**

## **Phase 0 — Căn chỉnh mục tiêu & phạm vi (1–2 ngày)**

**Mục tiêu:** chốt mục tiêu, phạm vi, tiêu chí hoàn thành (DoD), rủi ro pháp lý.

* **Tài liệu cần có:**
  + PRODUCT\_GOALS.md (mục tiêu, người dùng, KPI/OKR).
  + SCOPE.md (phạm vi: discovery, crawling, extraction, scoring, storage; ngoài phạm vi).
  + LEGAL\_COMPLIANCE.md (robots.txt, ToS, bản quyền, chỉ lưu metadata với paywall).
  + DELIVERABLES.md (đúng bộ RA phải nộp: config, code, DB/JSONL, báo cáo, log).
* **Checklist nhanh:**
  + Xác định domain ban đầu (reinforced concrete) + khả năng đổi domain qua config.yaml.
  + Định nghĩa **Definition of Done (DoD)** cho MVP (đếm số item discovered/fetched/scored/kept).
  + Rủi ro & giảm thiểu (IP bị chặn, paywall, lỗi PDF, chất lượng text).

## **Phase 1 — Quy trình làm việc & quản trị dự án (1 ngày)**

**Mục tiêu:** tạo nền nếp làm việc như dự án thực tế.

* **Git & quy ước:**
  + Nhánh: main (ổn định), dev (tích hợp), feature/\*.
  + **Conventional Commits**: feat:, fix:, docs:, refactor:, test:, …
  + PR template + Issue template + Labels.
* **CI/Lint/Test cơ bản:**
  + Lint: ruff/flake8; Format: black; Type checking: mypy.
  + Unit test khởi đầu: pytest.
* **Files khởi tạo:**
  + README.md, CONTRIBUTING.md, CODE\_OF\_CONDUCT.md, SECURITY.md
  + .editorconfig, .gitignore, .pre-commit-config.yaml
  + .github/ISSUE\_TEMPLATE.md, .github/PULL\_REQUEST\_TEMPLATE.md
* **Checklist nhanh:**
  + Tạo repo Git (public/private tùy giáo sư).
  + Bật **pre-commit** (tự động format/lint trước khi commit).
  + Thiết lập CI (GitHub Actions) chạy lint + test.

## **Phase 2 — Thiết kế kiến trúc & chuẩn dữ liệu (1–2 ngày)**

**Mục tiêu:** chốt cấu trúc module, schema dữ liệu, logging, cấu hình.

* **Tài liệu:**
  + ARCHITECTURE.md (sơ đồ pipeline: discovery → fetch → parse → score → store → export).
  + DATA\_SCHEMA.md (schema SQLite + JSONL cho export).
  + CONFIG\_SPEC.md (cấu trúc config.yaml: keywords, sources, files, scoring, storage, runtime).
  + OBSERVABILITY.md (logging, metrics, counters).
* **Quyết định kỹ thuật:**
  + SQLite + JSONL export.
  + Scoring v1: keyword-based (embedding hook để sau).
  + Tôn trọng robots.txt, retry/backoff, user-agent rõ ràng.
* **Checklist nhanh:**
  + Khoá các interface module (hàm public, input/output).
  + Thống nhất **Standard JSON schema** cho downstream.

## **Phase 3 — Thiết lập môi trường & skeleton code (1 ngày)**

**Mục tiêu:** dựng khung dự án chạy được lệnh rỗng (CLI).

* **Hành động:**
  + Tạo pyproject.toml/requirements.txt.
  + Dựng package uwss/ với thư mục con (utils, discovery, fetching, parsing, scoring, storage, provenance).
  + Tạo uwss/cli.py với lệnh: discover, fetch, score, export.
* **Checklist nhanh:**
  + uwss import OK, CLI in ra trợ giúp OK.
  + Thư mục data/raw, data/text, data/out tự tạo khi chạy.

## **Phase 4 — MVP theo luồng dữ liệu tối thiểu (2–4 ngày)**

**Mục tiêu:** chạy được từ discovery → export với dữ liệu thật (nhỏ).

* **Bước nhỏ:**
  + **Discovery (OpenAlex)**: lấy metadata theo keywords → lưu SQLite.
  + **Fetch**: tải URL/DOI hợp lệ (nếu PDF/HTML/…); log HTTP code.
  + **Parse/Extract**: PDF→text, HTML→text, DOCX, CSV; lưu data/text/\*.txt.
  + **Score**: tính điểm theo keyword & boost; đặt kept theo threshold.
  + **Export**: data/out/uwss\_export.jsonl.
* **Checklist nhanh:**
  + Ít nhất 50–100 records discover được.
  + Tải thành công ≥30% (tùy domain), còn lại status=failed có lý do.
  + JSONL có đủ trường: id, title, doi, year, score, kept, file paths.

## **Phase 5 — Chất lượng: kiểm thử, logging, chức năng an toàn (2–3 ngày)**

**Mục tiêu:** tăng độ tin cậy & minh bạch.

* **Tests:**
  + Unit test cho: tokenizer/score, parser (HTML/PDF), mime detect, config loader.
  + Integration test nhỏ: chạy discover→fetch→score trên 3–5 URL mẫu.
* **Logging & số liệu:**
  + Counters: discovered/fetched/scored/kept/failed theo từng source.
  + Thêm **provenance** (thời gian, công cụ, lỗi).
* **Kim chỉ nam tuân thủ:**
  + Chặn tải nếu robots.txt không cho.
  + Không vượt rate limit; delay mặc định (configurable).
* **Checklist nhanh:**
  + pytest pass; CI xanh.
  + Log đủ để tái hiện lỗi.

## **Phase 6 — Tài liệu sử dụng & báo cáo RA (1–2 ngày)**

**Mục tiêu:** bàn giao “dự án dễ chạy”.

* **Tài liệu:**
  + RUNBOOK.md (cách cài, chạy từng lệnh, debug, thông số).
  + RA\_REPORT\_TEMPLATE.md (mẫu báo cáo: số lượng, phân bố filetype, %OA, lỗi thường gặp).
  + CHANGELOG.md (mốc thay đổi).
* **Checklist nhanh:**
  + Copy file JSONL mẫu và screenshot log.
  + Báo cáo ngắn 1–2 trang kèm thống kê.

## **Phase 7 — Cải tiến (Optional nhưng nên có)**

**Gợi ý nâng cấp:**

* **Embeddings** (sentence-transformers) + blend điểm cosine.
* **Unpaywall API** để tìm OA PDF hợp pháp qua DOI.
* **Duplicate detection** (DOI/title hash + fuzzy).
* **Selenium** chỉ khi cần render JS (module riêng).
* **Scheduler** (cron/apscheduler) để chạy định kỳ.

# **Chuẩn “chuyên nghiệp” kèm checklist tổng hợp**

**Chuẩn kỹ thuật:**

* PEP8 + black + ruff + mypy (nếu có type hints).
* pre-commit hook chạy trước commit.
* CI chạy lint + test trên pull request.
* .env / secrets không đẩy lên git (dùng GitHub Secrets nếu cần API keys).

**Chuẩn tài liệu:**

* README ngắn gọn (giới thiệu, cách chạy, ví dụ).
* Architecture & Schema rõ ràng.
* Legal & Ethics ghi riêng.
* Runbook cho RA.

**Chuẩn pháp lý/đạo đức:**

* Tôn trọng robots.txt, ToS.
* Paywall → chỉ metadata.
* Ghi nguồn & thời gian crawl (provenance).

**Chuẩn quan sát/đo lường:**

* Metrics: discovered/fetched/scored/kept/failed theo ngày/lần chạy.
* Báo cáo: filetype distribution, top domains, top keywords.

## **Bắt đầu ngay (việc làm hôm nay)**

1. Tạo repo git + khởi tạo cấu trúc thư mục chuẩn (như tôi đã đề xuất).
2. Thêm các file “quy ước” tối thiểu:  
   * README.md, LICENSE (nếu cần), .gitignore, .editorconfig
   * .pre-commit-config.yaml (black, ruff)
   * requirements.txt
3. Thêm tài liệu:  
   * DELIVERABLES.md (dán mục tiêu RA)
   * LEGAL\_COMPLIANCE.md (tóm tắt robots, paywall)
4. Tạo config.yaml mẫu (domain\_keywords + sources)
5. Đẩy khung mã MVP (CLI rỗng) và bật CI.