TECHNICAL DOCUMENT

Scrapy

MỤC LỤC

Contents

[I ARCHITECTURE OVERVIEW 3](#_Toc49195871)

[1 Architecture 3](#_Toc49195872)

[2 Components 4](#_Toc49195873)

[II CONCEPTS 5](#_Toc49195874)

[1 Spider 5](#_Toc49195875)

[2 Item và ItemLoader 5](#_Toc49195876)

[3 Item pipelines 5](#_Toc49195877)

[4 Settings 6](#_Toc49195878)

# ARCHITECTURE OVERVIEW

## Architecture

Mô hình dòng dữ liệu mô tả lại quá trình thực hiện cào dữ liệu của scrapy. Bao gồm các thành phần bên trong và các dòng dữ liệu được thực hiện.



Bước 1: Engine nhận các request cho việc crawl từ các spider.

Bước 2: Engine gửi các request tới Scheduler để lên lịch các request và yêu cầu Scheduler gửi request tiếp theo để crawl.

Bước 3: Scheduler gửi request tiếp theo tới engine.

Bước 4: Engine gửi các request tới Dowloader, các request đi qua các Dowloader Middleware.

Bước 5: Sau khi trang hoàn tất tải xuống, Dowloader sẽ tạo các response và gửi về cho engine, thông qua các Dowloader Middleware.

Bước 6: Engine nhận các response và gửi tới spider thông qua các Spider Middleware.

Bước 7: Spider xử lý các response để tạo ra dữ liệu cần cào về, sau đó gửi dữ liệu và các request mới tới engine, thông qua Dowloader Middleware.

Bước 8: Engine nhận dữ liệu được cào tới item pipeline, đồng thời gửi các request tới Scheduler và yêu cầu Scheduler gửi request tiếp theo để crawl.

Bước 9: Tiếp tục thực hiện từ bước 1 cho tới khi không còn request từ Shecdular

## Components

**Spider**: Là các class spider dùng để phân tích và trích xuất các thành phần mong muốn.

**Engine**: Quản lý các luồng dữ liệu và sự kiện giữa các thành phần trong hệ thống. Kích hoạt các sự kiện khi có hành động xảy ra.

**Scheduler**: quản lý các request do engine gửi tới. Để thực hiện các request theo trình tự.

**Downloader**: Tải dữ liệu từ trang wedsite và trả về response.

**Item Pipelines**: Xử lý các dữ liệu sau khi được cào về. Thường làm các công việc như là làm sạch dữ liệu, kiểm tra, lưu trữ (database).

**Dowloader Middleware**: là thành phần nằm giữa Engine và Dowloader, xử lý các request khi đi từ Engine tới Dowloader, và các response khi đi từ Dowloader về lại Engine.

Nên sử dụng Dowloader Middleware khi thực hiện các việc sau:

Xử lý các request trước khi được gửi tới Dowloader

Thay đổi các response trước khi gửi tới Engine (Spider)

Gửi tiếp request tới Dowloader thay vì nhận response được trả về

Hủy một số request

**Spider Middleware**: là thành phần nằm giữa Spider và Engine có thể xử lý các input (response) và output (item/request).

Nên sử dụng Spider Middleware khi thực hiện các việc sau:

* Xử lý các output như thay đơi, thêm xóa các item, request
* Thực hiện các exception

# CONCEPTS

## Spider

Spider là class cho phép tùy chỉnh để cào dữ liệu từ những site nhất định. Spider là nơi để phân tích cấu trúc các site và thu thập dữ liệu theo một cấu trúc mong muốn.

Chu kỳ cào dữ liệu của spider:

1. Spider khởi tạo các request ban đầu để thu nhập dữ liệu từ các URL, và gọi hàm callback để xử lý các response nhận từ các request.
2. Trong hàm callback, nó nhận các response và phân tích cú pháp để trả về các item theo cấu trúc, và các request tiếp theo. Những requets này có thể gọi tới một hàm callback khác (hoặc gọi chính nó) sau đó được nhận các response bằng Scrapy và xử ly nó bằng hàm callback cụ thể.
3. Trong những hàm callback cụ thể, nó phân tích nội dung page thông qua các cơ chế (như Selectors, BeautifulSoup, lxml, ...) để tạo ra những item chưa các dữ liệu cần thiết.
4. Cuối cùng, những item được tạo ra bỏi spider sẽ được lưu trữ vào database (trong các Item Pipeline) hoặc ghi vào file thông qua Feed exports.

**CrawlSpider** : là class kế thừa từ Spider, nó cũng cấp một cơ chế thuận tiện cho lấy thông tin link theo các bộ rules được xây dựng sẵn (hoặc tự điều chỉnh).

## Item và ItemLoader

**Item** là object cho việc lưu dữ liệu theo cấu trúc nhất định. Sử dụng các item rất là hữu ích cho việc cào dữ liệu, vì mục tiêu của cào dữ liệu là tạo những item có cấu trúc từ nguồn dữ liệu không cấu trúc.

**ItemLoader:** cung cấp một cơ chế thuận tiện cho việc lưu trữ item, bằng cách tự động hóa một số tác vụ phổ biến như phân tích cú pháp dữ liệu được trích xuất thô trước khi gán nó.

## Item pipelines

Sau khi spider cào dữ liệu, dữ liệu sẽ được gửi tới các item pipeline. Item Pipeline xử lý các item thông qua nhiều phần và được thực thi tuần tự.

Mỗi Item Pipeline là một class thực hiện các phương thức đơn giản, nó thực hiện một số hành động lên item và có thể quyết định xem item đó được đi tiếp hoặc là drop.

Item Pipeline thường được dùng cho những việc sau:

* Làm sạch dữ liệu HTML
* Kiểm thực dữ liệu được cào về (có chứa dữ liệu hay không)
* Kiểm tra trùng lặp dữ liệu (drop item)
* Lưu trữ item vào database

## Settings

Scrapy Settings cho phép hiệu chỉnh lại hành vi của tất cả các thành phần của Scrapy, bao gồm cả core, extensions, pipelines và bản thân spider.

Là cơ sở hạ tầng cung cấp những biến toàn cục dạng key value để configuration cho các đoạn code.

Bên cạnh đó cũng có biến custom\_settings cho phép việc custom riêng cho từng spider