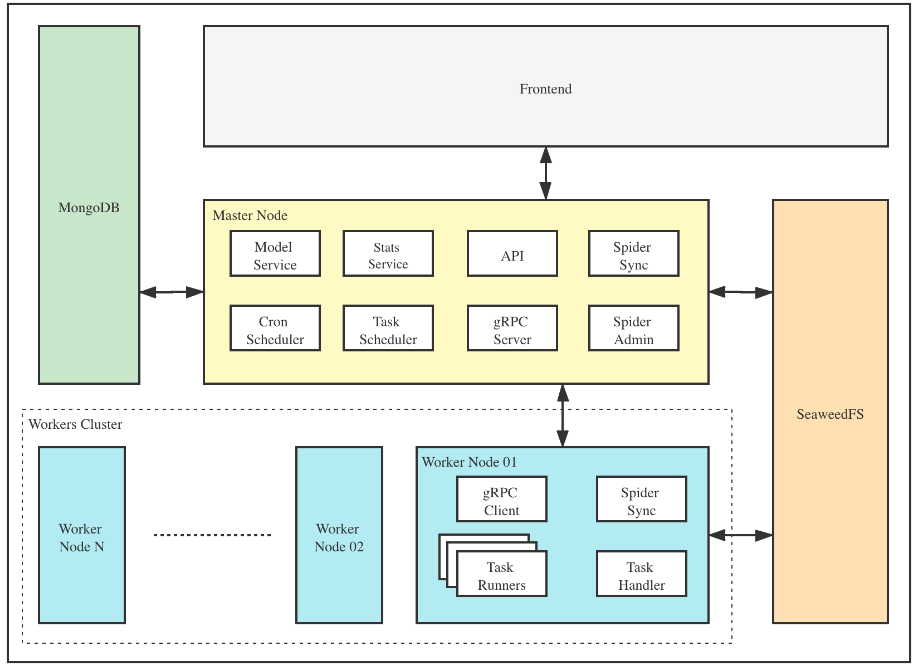
1. **Mô hình crawlab**



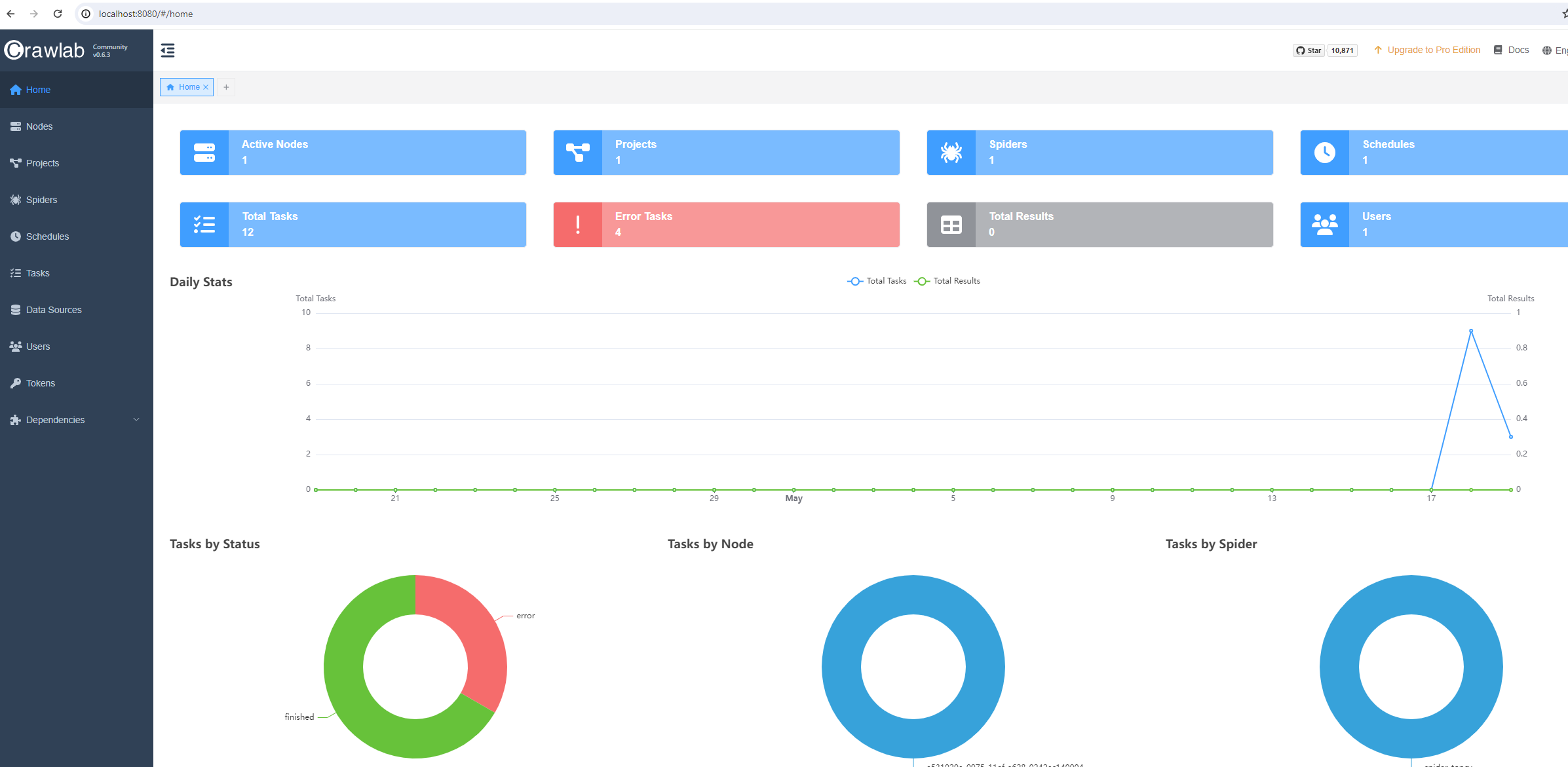
Crawlab là một hệ thống giúp chạy các spider, và quản lí, điều phối, theo dõi, monitor các spider.

Crawl là một hệ thống phân tán, Crawlab bao gồm nhiều module, có nhiều thành phần Master Node, Worker Nodes, Database, File System và Frontend. Mỗi mô-đun (ngoài Master Node hiện tại) đều có thể mở rộng để toàn bộ hệ thống phân tán có thể được mở rộng quy mô linh hoạt nhằm hỗ trợ các yêu cầu hiệu suất cao.

Reference document:

<https://docs.crawlab.cn/en/principle/architecture/>

1.1 Các thành phần trong crawlab



**Node**: quản lí các node, on/off node, config ip/host cho từng node ….

**Project**: đặt tên project, 1 project có nhiều spider

**Spider**: tạo spider, có thể cấu hình độ ưu tiên spider, cấu hình node chạy spider

**Scheduler**: đặt lịch để chạy spider, về qui tắc set scheduler có thể xem ở

[https://crontab.guru/#5\_0\_\*\_8\_\*](https://crontab.guru/%235_0_*_8_*)

**Task**: giúp quản lí các lần chạy spider, theo dõi trạng thái, monitor, log, record db của spider.

**Datasource**: Quản lí config các loại db.

**User**: cấp acc, phân quyền vào crawlab

**Dependency**: cài cắm các plugin, library của crawlab.

1.2 về cấu trúc và các setup project:

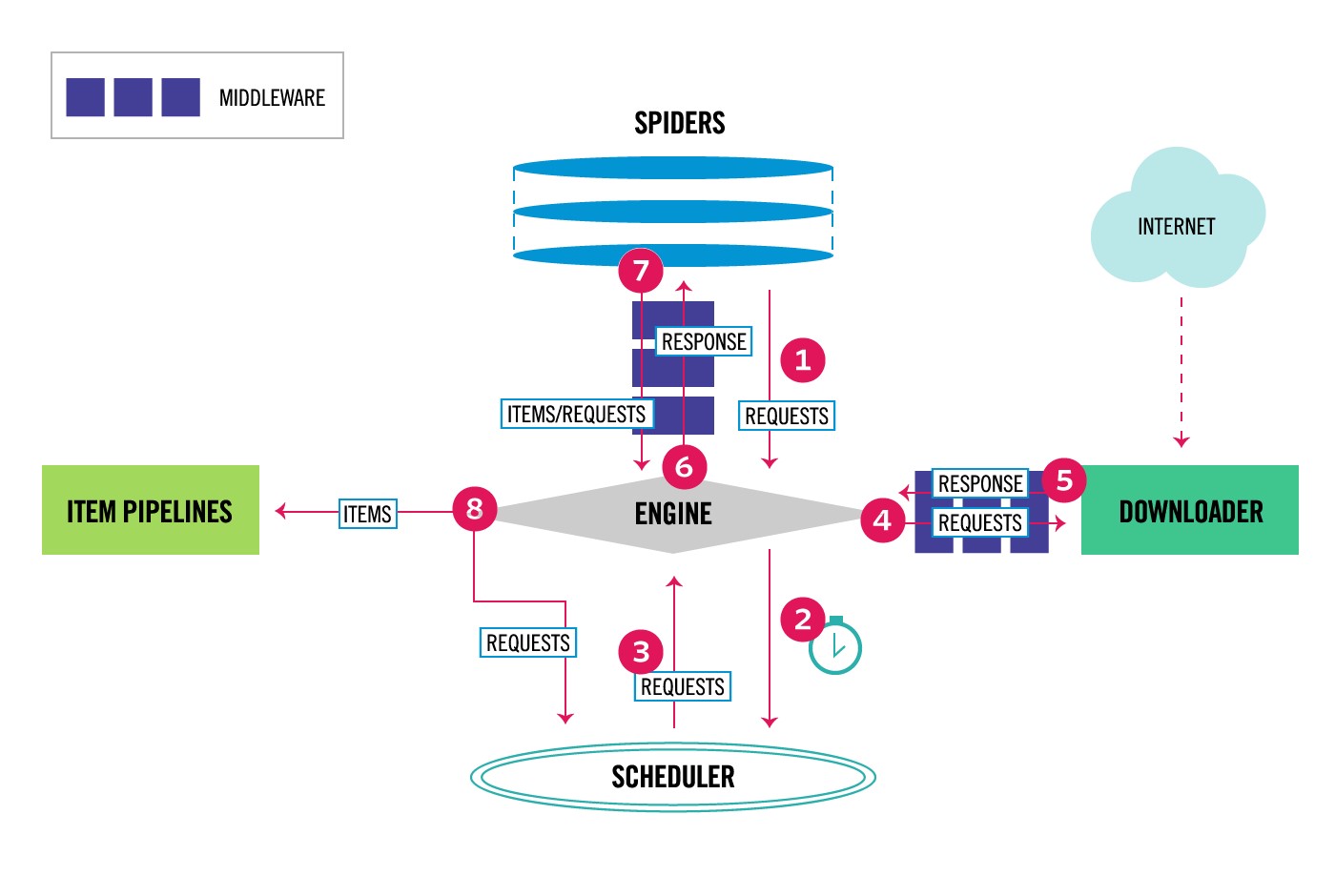
* host bằng cách dụng docker hoặc k8s

<https://docs.crawlab.cn/en/guide/installation/>

* Có thể xem tham khảo docker-compose file trong zip.

1. **Mô hình scrapy framework**

* Scrapy là 1 framework sử dụng để tạo các con spider nhiệm vụ crawl.



1. Engine là tp chịu trách nhiệm điều khiển luồng giữa các Component trong hệ thống và triggering events xảy ra, step 1 engine nhận request đầu tiên từ spider.

2. engine gửi request đó đến scheduler

3. scheduler get các sublink và trả về cho engine.

4. engine tiếp nhận gửi đến Downloader Middlewares để download các sublink này.

5. sau khi download sẽ gửi response ngược lại cho engine.

6. engine gửi lại spider.

7, 8 Spider xử lí response và build model output gửi tới itempipeline, itempipeline xử lí và lưu data vào db.

9.Quy trình lặp lại (từ bước 3) xử lí sublink tiếp theo cho đến khi không còn yêu cầu nào từ Bộ lập lịch nữa.

Document reference:

<https://docs.scrapy.org/en/latest/topics/architecture.html>

2.1 Về các setup môi trường development:

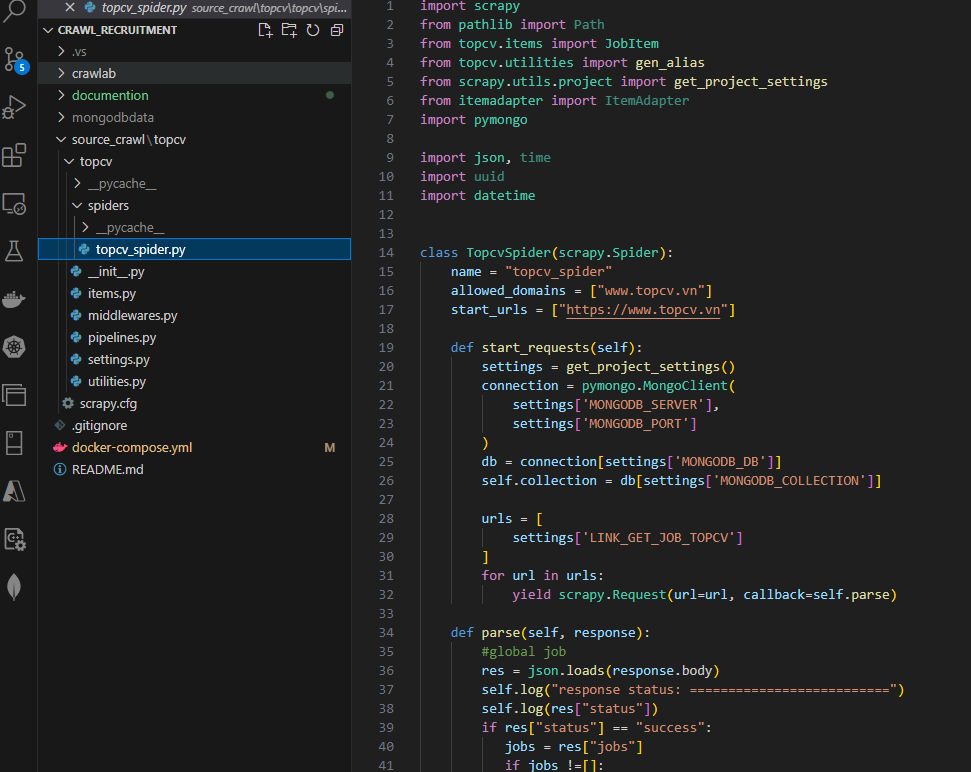
Mọi người có thể xem đầy đủ qua link này:

<https://docs.scrapy.org/en/latest/intro/install.html>

1. cài python: https://www.python.org/downloads/
2. sau đó cài scrapy framework

pip install Scrapy

2.2 Các thành phần và cấu trúc dự :



* topcv\_spider là spider để xử lí về nghiệp vụ
* item.py là model output
* middleware.py can thiệp giúp xử lí request, response, vd sau khi request trả ra httpcode 200, 400 xử lí ra sau ….
* Pipeline : xử lí lưu data vào db, check duplicate record db ….
* Setings.py chứa các config môi trường vd config kết nối mongo, lưu config cấu hình proxy …
* Utilities gồm các hàm chung, các func tiện ích vd convert chuỗi, get id từ url, build model …

2.3 cách tạo, run spider:

1. tạo project

scrapy startproject [ten\_du an]

vd: scrapy startproject crawl\_job

2. tạo spider:

scrapy genspider [ten\_spider] [linkcraw]

vd: scrapy genspider topcv topcv.vn

3. cách run spider:

vào thư mục chứa spider:

scrapy runspider topcv\_spider.py

cách dubug:

-Dùng hàm print hoặc self.log()