Document Project Crawler Db

1. Technology

* Ngôn ngữ: Java
* Lib-crawl: Jsoup.
* Db: MongoDb, Redis.
* Các Lib hỗ trợ: Jedis, Gson, Mongo-java-driver, joda-time.
* Other: proxy, user-agent.

1. Vai trò của của các technology được sử dụng.

* Jsoup có 2 nhiệm vụ:

+ Fetch html từ page.(Riêng với trang product-information sử dụng package của java là HttpUrlConnection để lấy html vì sử dụng Jsoup ko lấy được toàn bộ html của page này)

+ Đưa html về dom để lấy thông tin.

* MongoDb: Lưu trữ data về product-informations, product-reviews, categories(Thể loại của product được dùng để làm keyword cho tìm kiếm và phân loại sản phẩm)
* Redis: Dùng để làm queue, lưu trữ các thông tin tạm, ngoài ra còn dùng để lưu trữ form(Form sẽ quyết định lấy vị trí thông tin cần lấy)
* Jedis: Kết nối và thao tác với redis.
* Gson: Chuyển đổi object và json.
* Mongo-java-drive: Kết nối và thao tác với MongoDb.
* Joda-time: Chuyển đổi giữa các object xác định thời gian.
* Proxy: Fake id, xử lý trong trường hợp bị captcha.
* User-Agent: Fake user-agent cho crawler, xử lý trong trường hợp bị captcha.

1. Mục tiêu của project.

* Crawl thông tin về sản phẩm.
* Crawl thông tin review của sản phẩm.
* Sử dụng được cho nhiều website khác nhau.
* Hoạt động trong thời gian lien tục.

1. Cấu hình cho project.

* Cấu hình MongoDb.
* Cấu hình Redis.

1. Vai trò của từng class và package trong project.

* Package: com.zohar.crawler.dto

+ Chứa các class định nghĩa các object.

* Package: com.zohar.crawler.help

+ Chứa class exception tự định nghĩa(Exeception sẽ bị ném khi truyền thông tin thẻ vào mà ko lấy được thông tin tại vị trí đó).

* Package: com.zohar.crawler.main

+ Chứa 4 process thực thi:

+ ProcessProductLink: Process này làm 2 việc:

++ Lấy các link của product lần lượt từ page này sang page khác cho đến hết.

++ Đẩy các link product lên danh sách linkProduct trên redis.

+ ProcessProductInfo: Nhiệm vụ

++ Lấy từng link trong danh sách productLink trên redis.(Link bị lấy sẽ bị xóa khỏi danh sách)

++ Lấy thông tin Product trên page.

+++ Nếu không lấy được sẽ log (url, lý do error) ra file LogErrors.txt

+++ Nếu lấy được sẽ lưu thông tin về product vào mongodb. Sau đó đẩy link review của product và id của product lên danh sách productReviewDto

+ ProcessProductReview: Nhiệm vụ

++ Lấy Link review và id product từ danh sách productReviewDto.(Thông tin bị lấy ra khỏi danh sẽ bị xóa)

++ Lấy thông tin review của Product trên Page.

+++ Nếu lấy được sẽ lưu vào mongoDb

+++ Nếu ko lấy được sẽ add form mới -> Nếu form mới hợp lệ sẽ đẩy form mới lên redis và lưu thông tin vào db

* Nếu ko sẽ ném ngoại lệ yêu cầu nhập lại
* Hiện tại phần này đang để nhập console lên tách ra 1 process riêng giống như ProcessAddFormProductInfo

+ ProcessAddFormProductInfo: Xử lý việc thêm các form mới cho page ProductInfo

++ Thêm form mới cho những url product bị sai form

* Nếu form mới thêm đúng sẽ được add vào danh sách formProduct trên redis, thông tin product sẽ lưu vào db, link review và id của product sẽ được đẩy vào productReviewDto trên redis
* Nếu sai sẽ ném ngoại lệ và yêu cầu nhập lại
* Package : com.zohar.crawler.repos

+ Chứa các class thao tác trực tiếp với db

* Package: com.zohar.crawler.service.

+ Chứa các class xử lý service. Các class này là cầu nối giữa các process main và db. Ngoài ra còn có các class xử lý logic.

1. Chú ý.

* Nên tách add form trong processProductLink và processProductReview giống như với processProductInfo.
* Hiện tại có nhiều xử lý chỉ giành riêng cho amazon như xử lý date, url, … cần lưu ý khi dùng cho các website khác.
* Nên bổ sung thêm nhiều proxy để đảm bảo project chạy liên tục ko bị gián đoạn.
* Chưa deploy nên server để chạy liên tục trong thời gian dài nên có thể sẽ có bug phát sinh cần xử lý.
* Đối với amazon chưa cần dùng cookie.