Technical Documentation

Developed by: Muhammd faleh almutiri

Summer 2022

Table of Contents

1.0 General information.....3

1.1 Purpse.....3

2.0 Summary of Functionality and Requirements.....3

2.1 Functional Requirements.....3

2.2 Functional Summary.....4

Project name

Client Targeting Automation

Project sponsor	Project Start date	Project End date
Hudhud ai	6/19/2022	8/18/2022

Purpose

The Purpose of this project is to automate the info gathering from websites such as (Maroof.sa, Twitter, google play store and business own websites) with the aim to be able to assess if the client has a need for the services provided by the company.

Deliverables

Planed	Actual	Comments
Maroof.sa Scraper via scrapy	Deliverad	
Twitter scraper via twint	Deliverad using twitter api	Twint had issues
Instagram scraper	NaN	Any scraping attempt resulted in a ban from meta.
Website scraper	Deliverad	
Googleplay scraper	Deliverad	

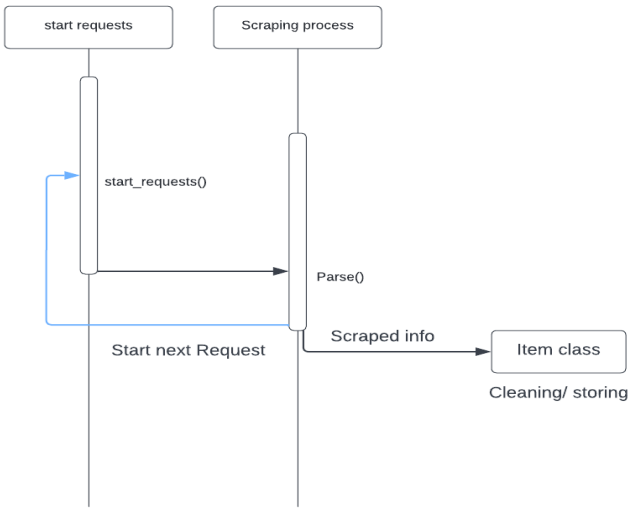
Functional Requirements

- Scrapy ver 2.6.2
- Tweepy ver 4.10
- Pandas ver 1.4
- Configparser ver 5.2

Functional Summary

- CheckWebsites Spider

The Scraper takes a list of website urls taken from Maroof.csv file (scraped by MaroofMainInfo.csv, **start_requests()** start the Requesting process with a callback method of **parse()**, for every url the scraper will look for either salla or zid urls and will return ecom variable and the website urls,

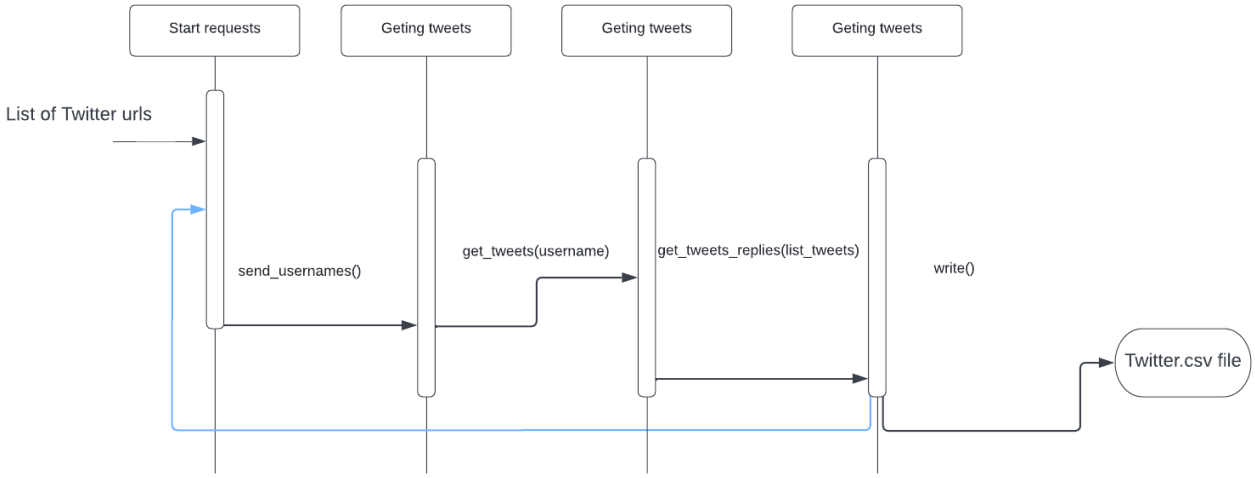


Those items are sent to Website Item class in items.py for filtration, finally the info is stored in a csv file in files directory (any values that need to be added also need to be added to websiteitem class)

- GooglePlay spider

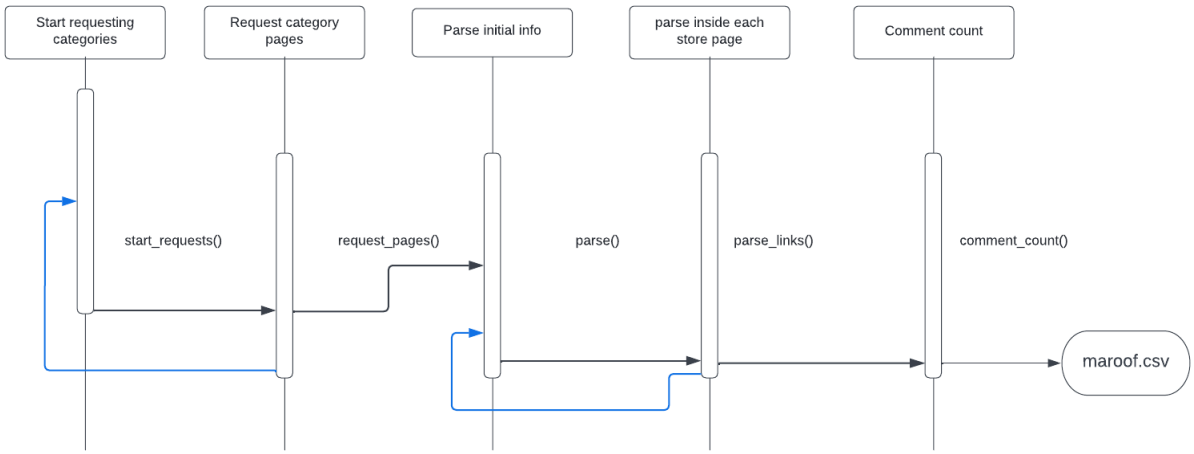
The Scraper takes a list of Google play urls taken from Maroof.csv file (scraped by MaroofMainInfo.csv, **start_requests()** start the Requesting process with a callback method of **parse()**, items scraped are (downloads count, reviews, score, last patch date)

- Twitter Scraper



The Twitter class takes a Twitter urls list from Maroof.csv, the keys and tokens are read from config.ini, **send_names(),** loops over the urls and shorten them into usernames, send each to **get_tweets()** which will use twitter api v1 to get the tweets that match the date condition and get followers count, **get_tweets_replies()** takes a list of tweets ids that match the date condition and get the public metrics.

- MaroofMainInfo spider



start_requests() will start sending category links to **request_pages()**, that then determines how many pages there is and start requesting them, **parse()** will get the initial info such as store urls, activity, Crnumber, rating and number of ratters, than request each store url and call **parse_links()** to parse store info such as social urls, each sent item between spider function is sent through meta objects, **comment_count()** will get comment count that matches the date by using Beautiful soup .

- run.py

It calls each function and connect the csv files into one final file