Webscraping Report

Gold Nwobu

November 2025

1 Introduction

2 README: STORI Annual Report Downloader

2.1 Project Overview

This mini project implements a small data pipeline that downloads annual financial report PDFs from the Belgian FSMA STORI system. The script communicates directly with the STORI backend API, filters for "Annual financial report" documents in English or Dutch, and saves them locally using a consistent naming convention:

`IssuerName_LEI_AnnualReport_PublicationDate.pdf`

The pipeline is written in Python and is split into two main modules:

- `main.py` – orchestrates the pipeline logic (loading issuers, calling the API, filtering results and saving files).
- `api_client.py` – handles all HTTP requests to the STORI API.

2.2 Prerequisites

To run the project on any system, the following are required:

- Python 3.8 or later (tested with Python 3.10).
- The `requests` Python library.
- Internet access to reach `https://webapi.fsma.be`.
- A local copy of the issuer list file, stored as `issuers.json.txt` in the project directory.

2.3 Folder Structure

After setup, the project folder looks approximately like this:

```
webscraping-pipeline/
 main.py
 api_client.py
 issuers.json.txt
 downloads/        # downloaded PDFs are stored here
 logs/             # log file stori_downloader.log is written here
 venv/ (optional)  # virtual environment, if used
```

2.4 Setup Instructions

1. **Clone or copy the project folder**

   Copy all project files (`main.py`, `api_client.py`, `issuers.json.txt`) into a folder on your machine, e.g. `webscraping-pipeline`.

2. **(Optional) Create and activate a virtual environment**

   ```
   cd webscraping-pipeline
   python -m venv venv
   # Windows:
   venv\Scripts\activate
   # macOS / Linux:
   source venv/bin/activate
   ```

3. **Install Python dependencies**

   The only external dependency is `requests`:

   ```
   python -m pip install requests
   ```

4. **Check the issuer file**

   Make sure `issuers.json.txt` is present in the same folder as `main.py`. This file contains the issuer objects exported from STORI (with fields like `companyId` and `abbreviation`).

2.5 How to Run the Pipeline

Once the dependencies are installed and the issuer file is in place, the pipeline can be started with:

```
python main.py
```

When the script runs, it performs the following steps:

1. Initialises logging and creates a log file at `logs/stori_downloader.log`.
2. Creates an HTTP session with a custom `User-Agent`.
3. Loads and normalises issuers from `issuers.json.txt`.
4. For each issuer:
   (a) Sends a POST request to the STORI `/result` API to search for "Annual financial report" filings from 2011 onwards.
   (b) Filters the returned documents to keep only English or Dutch PDFs.
   (c) Downloads the matching PDFs via the STORI `/download` endpoint.
   (d) Saves each file into the `downloads/` folder using the naming convention `Issuer_LEI_AnnualReport_YYYY-MM-DD.pdf`.
5. The script stops once a global limit on the number of downloads is reached (by default: 5 files).

2.6 Configuration

Some basic configuration can be changed directly in `main.py`:

- **Maximum number of downloads:**
  In the `main()` function:

  ```
  MAX_DOWNLOADS = 5
  ```

  This can be increased if more files should be downloaded.

- **Document type (Annual financial report):**
  The STORI document type ID for "Annual financial report" is stored as:

  ```
  DOCUMENT_TYPE_ANNUAL = "9813c451-9fd4-41ba-ba7d-4e0dda0d3051"
  ```

  If the FSMA ever changes their IDs, this value can be updated.

- **Issuer file name:**
  By default, the script expects:

  ```
  issuers_file = Path("issuers.json.txt")
  ```

  This can be changed if the issuer file is renamed.

2.7 Troubleshooting

- **ModuleNotFoundError:** `requests`
  Make sure the dependency is installed:

  ```
  python -m pip install requests
  ```

- **Issuer file not found**
  Check that `issuers.json.txt` is in the same directory as `main.py`. If it is missing, the script will create a small default issuer list, but only for one hard–coded issuer.

- **HTTP or timeout errors**
  These are usually logged in `logs/stori_downloader.log`. They can happen if the FSMA API is temporarily unavailable or if the network connection drops.

- **No files downloaded**
  In this case, check the log file to see whether:

  1. the issuer IDs in `issuers.json.txt` are correct, and
  2. STORI actually has annual financial reports available for the chosen issuers and date range.