ALEX TORRES

DATA ENGINEER

+1 206-579-1244

almatob@hotmail.com

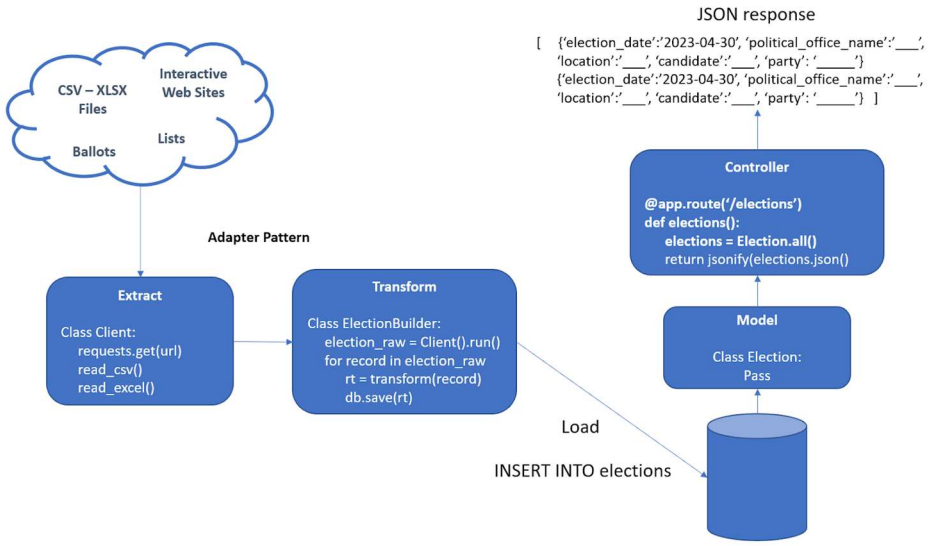
Seattle, WA - USA

linkedin.com/in/alextorreswa

May 1, 2023

Dear Test Evaluation Team,

To contribute to Ballotpedia to expand local elections coverage, I would use the adapter pattern with python code to extract data from the different sources. I would do so by making a main Client class that extracts the data from various formats. Starting with the raw data, it would carry out the corresponding transformations to be able to host it in the staging area (relational database, data lake or data warehouse according to needs). Then, to deliver the already processed data to the stakeholders, I would make a web application using Flask or Django to deliver it in Json format or to generate csv files for further analysis.



Expanding a little more regarding the extraction of data from different formats:

- excel or csv files
  - For not very large volumes, we can use the pandas functions of read_csv() or read_excel().
  - However, for high volumes of data, a good alternative would be Pyspark.
- Web pages with a lot of Javascript
  - We could use a browser automation tool like Selenium. That way we could control the interaction with the website to generate the desired content, then develop Python code with a web scraping framework like Beautiful Soup to deal with each of the tables and extract the data points according to schema.
- To extract the data from the ballots,
  - We can use the OpenCV library to process the images and we can use Regex and OCR for text recognition. I would perform different tests using Pytest framework to find the most optimal.
- For the extraction of data from the lists
  - I would use the Regex library with OCR in python. I've had good experience with Regex for unstructured data extraction in my previous job as a Data Scientist Intern in an AI company.

Expanding a little more regarding transformation:

- Pandas has a good set of functions to interact with dataframes (concat(), merge(), join(), groupby(), pivot_table())

Taking into account that working with images is required, a data lake is probably required to store unstructured data and also store csv, excel files, for which amazon AWS S3 could be used.

Considering that the data can be generated in different periods of time, we could use Apache Airflow for orchestrating the ETL pipeline according to the needs of each place, format, time, etc.