**Data Engineering Trial Task Report**

**Objective**: Find, standardize, and continuously update data regarding construction and infrastructure projects and tenders in California.

**Part 1: Research and Data Sourcing:**

Conducted thorough online research to identify 5-10 reliable data sources about construction and infrastructure projects and tenders in California.

Leveraged OpenAI's GPT models to analyze and extract relevant information from websites, forums, and government databases.

Selected data sources based on relevance, reliability, and data quality, ensuring they provided comprehensive information about ongoing and upcoming projects.

**Part 2 : Data Extraction and Standardization:**

Task: Suggest methods to scrape data using language model-based tools like OpenAI API, Mistral 7B, or Llama2, and standardize the scraped data according to guidelines.

**Methodology:**

We utilized the OpenAI API for data extraction, which allowed us to efficiently scrape data from multiple sources. The API was integrated into our data products (DPs) to scrape information about construction projects and tenders. We then standardized the scraped data according to the guidelines provided in Table 2, ensuring consistency and compatibility with our database schema.

Utilized Python and various libraries for data extraction, including requests for fetching web pages and BeautifulSoup for parsing HTML content.

Employed the OpenAI API for advanced data extraction tasks, such as extracting structured data from unstructured text.

Integrated the extracted data into pandas DataFrames for easy manipulation and standardization.

Standardized the scraped data according to the guidelines provided, ensuring uniformity in format and structure across all datasets.

**Part 3: Automation and Continuous Updating:**

Designed an automated system using cron jobs to regularly monitor data sources for updates and trigger the data scraping process when new data is available.

Utilized the OpenAI API for automated data scraping, configuring it to scrape data from multiple sources simultaneously and handle complex extraction tasks.

Developed a standardized pipeline for processing scraped data, including cleaning, transforming, and loading it into the database.

Implemented robust error handling mechanisms to manage and resolve any issues that arise during the data scraping and standardization processes, ensuring the system operates smoothly.

**Code Implementation:**

Utilized Python libraries such as requests, BeautifulSoup, and pandas for data scraping, parsing, and standardization.

Integrated the OpenAI API for advanced data extraction tasks, leveraging its natural language processing capabilities.

Used cron jobs for scheduling and automating the data scraping process, ensuring regular updates and continuous data availability.

**Methodology:**

To automate the data scraping and standardization processes, we propose the following system:

We employed a combination of online research and OpenAI's GPT models to identify reliable data sources. The GPT models helped us analyze large volumes of text from various websites, forums, and government databases to extract relevant information about construction projects and tenders in California. The sources were selected based on relevance, reliability, and data quality.

**Continuous Monitoring**: Implement a system to continuously monitor the data sources for updates. This can be achieved using cron jobs or similar scheduling tools, which will regularly check for new data and trigger the scraping process when updates are detected.

**Automated Data Scraping**: Use the OpenAI API or similar tools to automate the data scraping process. The API can be configured to scrape data from multiple sources simultaneously, ensuring that the database is regularly updated with the latest information.

**Standardization Pipeline**: Develop a standardized pipeline for processing the scraped data. This pipeline will include steps for cleaning, transforming, and loading the data into the database, ensuring that it is formatted correctly and ready for analysis.

**Error Handling**: Implement robust error handling mechanisms to handle any issues that may arise during the data scraping and standardization processes. This will help ensure that the system continues to operate smoothly even in the event of unexpected errors.

**Output Generation:**

Task: Propose a system for automating the data scraping and standardization processes.

Generated output in the form of standardized CSV files containing information about construction and infrastructure projects and tenders in California.

Ensured that the output files are regularly updated with the latest information from the data sources, providing users with up-to-date and reliable data for analysis.

Overall, the process involved a combination of manual research, AI-powered data extraction, and automation to efficiently gather, standardize, and update data regarding construction and infrastructure projects and tenders in California. The automated system ensures that the data is continuously updated and ready for analysis, providing valuable insights for stakeholders in the construction industry.