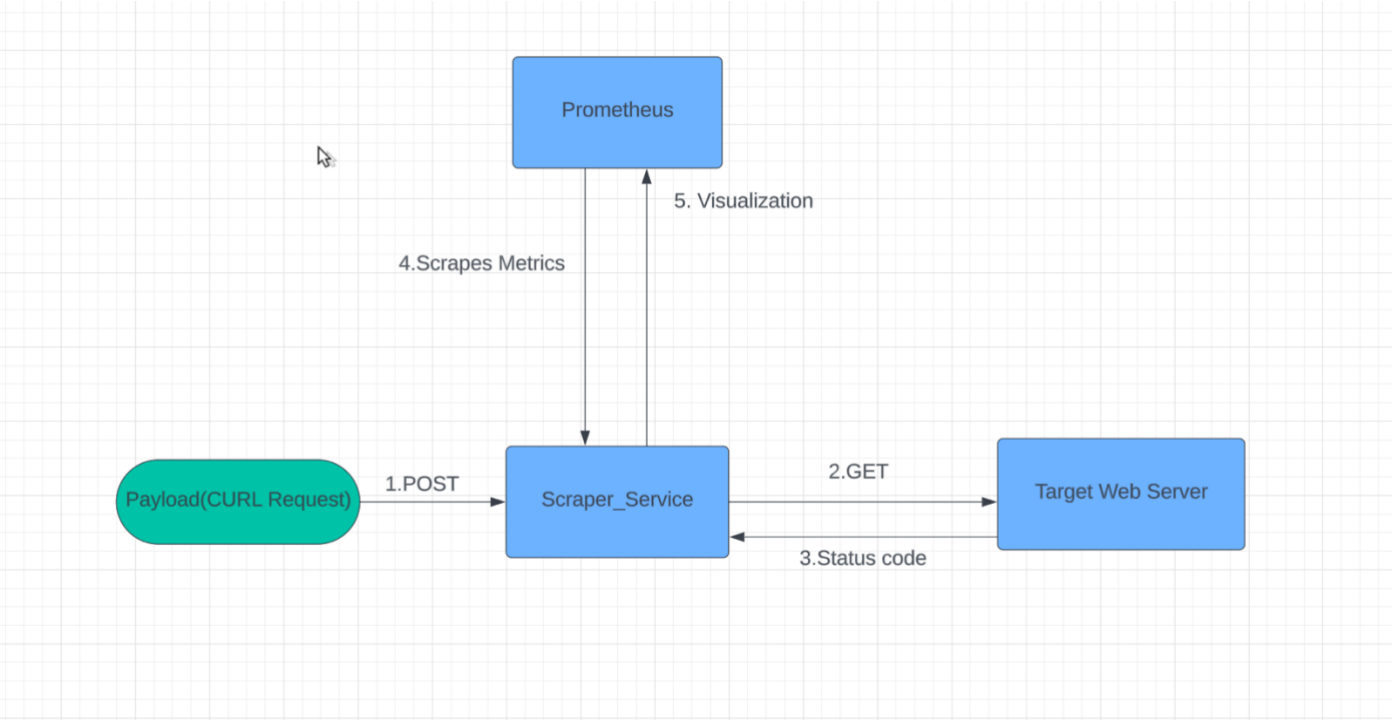
# Scraper Service Workflow




Steps:

1. **Send POST request with URL to Scraper Service on port 8080.**
   a. Client sends a JSON POST request with the URL to be scraped.

b. Example command: curl -X POST -H "Content-Type: application/json" d '{"url": "http://google.com"}' http://localhost:8080

2. **Scraper Service receives and parses the request.**
   a. Scraper Service (Flask-based) listens on port 8080 for POST requests.
   b. Parses JSON payload to extract the target URL.

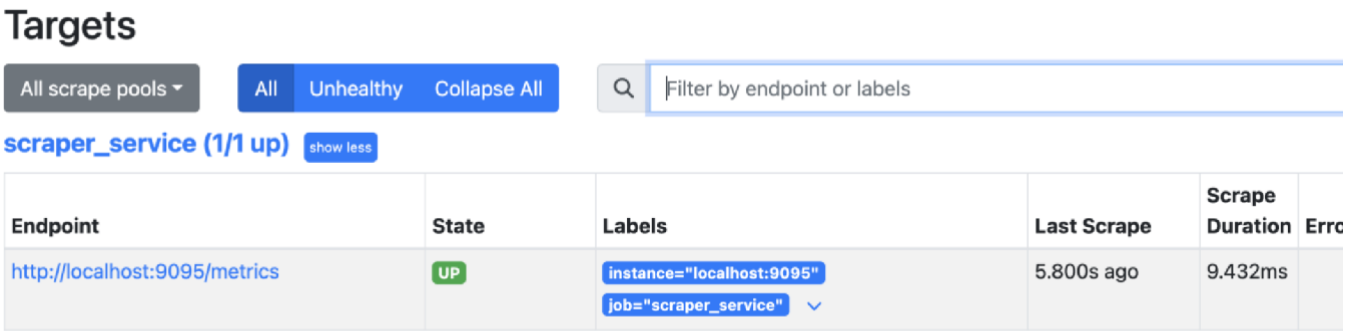
3. **Scraper Service performs HTTP GET request to target URL.**
   a. Uses the **requests** library to make an HTTP GET request to the provided URL.

4. **Target web server responds with HTTP status code.**
   a. The target web server returns an HTTP status code (e.g., 200, 404).

5. **Scraper Service increments Prometheus counter with URL and status code.**
   a. Increments the Prometheus counter (**http_get**) with labels for the URL and status code.
   b. Uses the **prometheus_client** library for metric handling.

6. **Scraper Service exposes metrics at /metrics endpoint on port 9095.**
   a. Metrics are accessible at the **/metrics** endpoint on port 9095.
7. **Prometheus scrapes metrics from Scraper Service.**
   a. Prometheus is configured to scrape the **/metrics** endpoint at regular intervals.
   b. Configuration is done in **prometheus.yml**.

8. **Metrics are available for viewing in Prometheus UI.**
   a. Access the Prometheus UI at **http://localhost:9090** to view and analyze




metrics.
   b. Validate the web service status by going to **status->targets**


9. **Analyze the metrics exposed by Scraper Service by accessing the Graph tab and entering queries.**
    a. Example query: `http_get_total` to retrieve the total count of HTTP GET requests.

**Tools Involved**

- **Scraper Service**: Flask-based service for URL scraping and exposing Prometheus metrics.
- **Prometheus**: Monitoring and alerting toolkit to scrape and store metrics.
- **Prometheus Client**: Python library (**prometheus_client**) for defining and exposing metrics.