Basic overview of the application

Amazon Product Scraper web application is built using the Django framework and uses the Selenium library for web scraping. It also uses Celery to schedule and run the scraping process at regular intervals (Once in a day in this case). The core files of the application include the **models.py**, **views.py**, **serializers.py**, **urls.py**, **product_scraper.py**, **tasks.py**, **settings.py**, and **requirements.txt**.

- **amazon_product_scraper** - This is the main Django project Folder.

- **scraper_app** - This is the main Django app of the project. It contains all the models, views, serializers, and URLs for the application.
  Note: We don't have any view or urls for this app

  product_scraper.py - This is the core file of the application which contains the implementation of the **ScrapProduct** class. It is responsible for scraping data from Amazon search results using the Selenium library. This file is used to initialize the web driver, accept cookies and start the scraper, and also to get data from the webpage.

- **models.py** - This file contains the database schema for the two models used in the application, **Keyword** and **ScrapedData**.

We created separate app for API endpoints.

- **views.py** - This file contains the implementation of the view classes that are responsible for serving the API endpoints of the application. It includes the implementation of the **ScrapedDataList** view, which returns a list of all scraped data and the **TriggerScraper** view, which initiates the web scraping process when hit the endpoint.

- **serializers.py** - This file contains the serializer classes used to convert the **ScrapedData** model instances to JSON format for the API responses.

- **urls.py** - This file contains the URL configuration for the **scraper_app** app.

- **tasks.py** - This file contains the Celery tasks for running the scraping process at regular intervals. The **run_scraper** function is called as a Celery task, which is used to start the scraping process.

- **settings.py** - This file contains the configuration settings for the Django project.

- **requirements.txt** - This file contains a list of all the required packages and their versions to run the application.