Web Scraping Project Plan :

Goal:

Scrape used Hyundai car details from Cars24 (Mumbai) and export the cleaned data into a CSV file.

Website : https://www.cars24.com/buy-used-hyundai-cars-mumbai/?sort=bestmatch&serveWarrantyCount=true&listingSource=Homepage_Filters

Part 1 – Setup and Initialization

Members (2): *Team A*
Responsibilities:

1. Import all necessary libraries:
    - requests, bs4 (BeautifulSoup), pandas etc..
2. Set up project structure in .ipynb(creating cells for codes in jupyter notebook (Code Cell for Importing Libraries, Code Cell for Data Cleaning, Code Cell for Data Extraction) so that the next part-members can code)
3. Create a **clean HTTP request–response framework** using requests.get()
    - Verify status codes (200 OK)
    - Test if the page content responds properly

In short : Importing necessary Libraries , Creating HTTP Request & Response & Project Structure.

Part 2 – Website Declaration and Exception Handling

Members (2): *Team B*
Responsibilities:

1. Declare the **target website URL** (e.g., Hyundai cars page)
2. Build a **robust function** to send requests and handle exceptions such as:
    - Timeout errors
    - Connection errors
    - Invalid responses
3. Use try-except blocks and print descriptive error messages

In short : Use Exception Handling, Declare the website (used-cars, Hyundai brand).

Part 3 – Data Extraction

Members (2): *Team C*
Responsibilities:

1. Identify required **HTML elements or JSON fields** for the following data:
    - Car Name / Model
    - Year of Manufacture
    - Kilometers Driven
    - Fuel Type
    - Transmission
    - Price (₹)
    - City
2. Extract the values using **BeautifulSoup** (if HTML-based) or **JSON parsing** (if API-based)

In Short : Identifying HTML Elements, Data Extraction

Part 4 – DataFrame Creation

Members (2): *Team D*
Responsibilities:

1. Store the extracted details in a structured **list of dictionaries.**
2. Implement logging or print statements to verify correct extraction per record
3. Convert the extracted data into a **Pandas DataFrame**

In Short : Store the extracted details in a structures list of dictionaries and Form a DataFrame from the extracted data.

Part 5 – Data Cleaning, Data Presentation and Export

Members (2): *Team E*
Responsibilities:

1. Perform complete **data cleaning:**

    - Remove duplicates
    - Handle or drop NaN values
    - Convert data types appropriately (e.g., numeric price, integer kilometers)
    - Rename columns properly (e.g., Car_Name, Year, Fuel_Type)
2. Validate data consistency and ensure no corrupted entries remain
3. Export the final DataFrame into a structured **CSV file** (not Excel)
4. Ensure correct encoding (e.g., UTF-8) and proper file naming (e.g., Scraped_Data.csv)
5. (Optional) Visualize simple insights like car count by year or price distribution using Matplotlib to ensure the smooth code workflow.

In short : Data Cleaning, Data Presentation

Coordinator Role (1 person)

Responsibilities:

1. Engage with all parts and help the mates, modify the code at the end (If necessary).
2. Merge all parts together in GitHub main branch everytime.
3. Ensure consistent naming conventions, indentation, and comments.
4. Run the integrated code to verify full workflow from start to end and also during the each part completion.
5. Write a short README.md file describing:
    - Project overview
    - Steps to run the code
    - Team contributions
6. Deliverables:
    - Python Notebook : Modifying the jupyter notebook into a well formatted document .ipynb notebook (at the end, also during the completion of Each Part)
    - Presentation File : Converting the presentation .ipynb file into a PDF before submitting
    - Report : A Brief detailing the process like any challenges faced (from project as well as team side), solutions implemented, and insights gained from the data