MULTI■SHOP AGGREGATOR WEB APPLICATION

Final Year Project Report

Name: Alishba Mehmood

Registration No: 2022-AG-7950

Degree: BS (Information Technology)

Section: B

TABLE OF CONTENTS

1. Introduction

This project aims to simplify online shopping by aggregating product data from multiple e■commerce platforms such as Daraz, Temu, Shein, and Alibaba. Users can search once and get images, prices, and purchase links from different platforms instantly.

2. Problem Statement

Shoppers spend significant time visiting separate websites to compare prices and availability. No unified system currently provides cross■platform product comparison in Pakistan.

3. Objectives

• Centralize product search from multiple platforms.

• Fetch and deliver product information including names, prices, images, and links.

• Reduce user effort and browsing time.

• Provide a clean and efficient user interface.

4. Project Scope

The scope covers frontend development, backend API creation, Selenium-based scraping, and data presentation. Additional analytics such as price trends may be added later.

5. Technologies Used

Frontend: HTML, CSS, JavaScript

Backend: Python Flask

Libraries: Flask 3.0.0, flask■cors 4.0.0, selenium 4.15.2, webdriver■manager 4.0.1

Tools: VS Code, Chrome WebDriver

6. System Architecture

• User enters search keyword in frontend.

• Flask backend processes the request.

• Selenium scrapes each shopping website.

• Results are formatted in JSON.

• Frontend displays cards containing images, names, prices, and purchase links.

7. Methodology

The project follows a modular development methodology:

1. Frontend UI creation using HTML, CSS, JavaScript.

2. Developing Flask APIs to handle search requests.

3. Performing Selenium scraping and standardizing the collected data.

4. Displaying results in responsive cards.

8. Features

• Unified product search

• Real-time scraping

• Product cards with images, prices, and links

• Cross-platform comparison

9. Challenges

• Dynamic web content requiring waits

• Different HTML structures among websites

• Captcha barriers

• Selenium performance limitations

10. Results

The working prototype demonstrates successful cross-platform product aggregation, improving user convenience significantly.

11. Conclusion

The Multi-Shop Aggregator is an effective solution to reduce e■commerce browsing time. It provides unified searching and can be expanded into a commercial tool.

12. Future Enhancements

• Price trend graphing

• AI recommendation engine

• Mobile app version

• Browser extension

13. SYSTEM DIAGRAMS

Flowchart:

User Search Input → Flask API → Selenium Scraper → Data Process → JSON Output → Frontend Display

DFD (Level 0):

[User] → (Search Query) → [Aggregator System] → (Scraped Data Displayed) → [User]

DFD (Level 1):
User → Search Module → Scraper Engine → Website Data → Result Formatter → User Interface

ERD Structure:

ENTITY: Product
• product_id
• title
• price
• image_url
• product_link
• source_platform

ENTITY: User_Search
• search_id
• keyword
• timestamp

Relationship: One search → many products

14. References

Flask Documentation

Selenium Documentation

Webdriver■Manager Library

Daraz, Temu, Shein, Alibaba Websites