SMARTSHOP: A Price comparison system with Email Notifications and Graphs

Prof. Shalini Nigam[1], Kshitija Gholap[2], Sayli Jadhav[3] , Manish Singh

[1]*Head of Department, Dept. of I.T., G.S.M.C.O.E, Savtribai Phule Pune University, Pune, Maharashtra, India*
[2]B. E Student, Dept. of I.T., G.S.M.C.O.E, Savtribai Phule Pune University, Pune, Maharashtra, India
[3]*B. E Student, Dept. of I.T., G.S.M.C.O.E, Savtribai Phule Pune University, Pune, Maharashtra, India*

Abstract - E-Commerce nowadays plays a vital role in our daily lives. It is redefining commercial activities around the world. Over the years, E-Commerce has evolved in profound ways. So, Online Shopping with a price comparison system offers trends and deals in different shopping sites like Amazon, Flipkart and is beneficial for consumers. This paper presents a price comparison system for comparing prices, offers, deals and shows trends in products of different categories from Amazon and Flipkart and graphs to analyze. In this proposed system crawler is implemented for retrieval of data and react is used for easy handling of users.

***Key Words*:** System scrapper; crawler; data indexing and retrieval; Price Comparison System(PCS)

1. INTRODUCTION

Due to the changes in lifestyles or development customers now take more interest in online shopping rather than going to shop. Online shopping site has many offers and deals for the user but going to each and every site is time-consuming and frustrating too ,so to have a price comparison system which can shows trends and deals at one place helps a lot. Creating a system that is capable of bringing information from one or more E-Commerce Sites to a common place for user's ease. The system is capable of extracting information from various real-time sources of the products, aggregate it, analyze it, and deliver the obtained results in the form of graphs and notifications to the users.

This system uses the following technologies:

1. Web crawler: The system deals with prices from different sites. The first thing is to gather data from different e-commerce websites. It is impossible to manually collect the data from websites. Hence, we created a web crawler that will navigate to these e-commerce websites and will fetch required data and send it to the database.

2. Mysql-Database: MySQL is a widely used relational database management system (RDBMS). It is free and open-source as well as it is ideal for both small and large applications. This system deals with a large amount of data so MySQL is helpful and easy to perform operations on stored data. Here the data gathered from the crawler get stored in the MySQL database from a CSV format.

3) React-js frontend: React is an open-source front-end JavaScript library used for developing a user-friendly system. In this price comparison system graphs are created by a chart-js.

2. Architecture of the system

The system is mainly divided into three modules. Here the working of each module is independent but is dependent on each other's data which is generated at each stage.

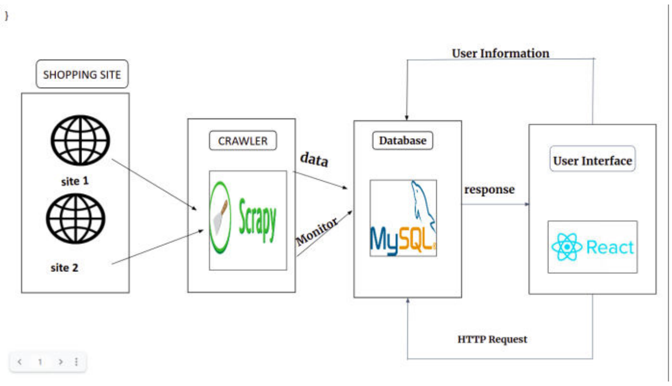


Fig. 1 - System Architecture

Fig.1 describes system architecture and detailed working procedures. The front-end provides a graphical user interface (GUI) in the form of a website where users can interact with the comparison system whereas the backend consists of scrapping techniques to extract product information like price, offers, site, brand, etc. from different e-commerce websites.

The extracted information of e-commerce products is stored in the form of CSV and is imported into the MySQL database. When the Client requests for desired product from the main website the data is fetched from an API which is created from the database by firing a query to the local database and then the Product Information is displayed on the main web page.

Another Feature is provided that users can see the comparison through the live charts like trends and offers which is easier to understand for the users. Users may also analyze and see the products with more details and specifications about them. A user can be transferred to the site where the product is shown to buy which they are searching for.

3. Implementation

Implementation of SmartShop is as below:

i) Crawler Creation for Data Collection:

For the creation of a crawler, the scrapy library of python is used were using various functions like request, response to extract link of each product their name, price, savings, etc. from various sites. Used methods like extract and extract all for the extraction of information in Xpath. It is a language for selecting nodes in XML documents, which can also be used with HTML and CSS. It defines selectors to associate those styles with specific HTML elements all the data get store in form of CSV.

ii) Data Cleaning and Database Creation:

After getting data in the CSV format from the crawler we cleaned it with the help of the Python Pandas library.

We used read_csv() to read the CSV file from the location it is saved and applied astype() to convert the data type to the appropriate type for database creation.

We removed the 'nan ' values by replace() to be replaced with 0 or removed.

We added more columns for storing further information like site name, which type of product is it, URL link for that particular product. We added an Image column for our front-end use for the image showing for each product.

We dropped the unwanted columns that the crawler may fetch while scraping the data by the drop(). By using the Lambda split() function we split the data we got to extract the brand name.

and Assign() for creating a column for site name and product name.

After cleaning and inserting the required data now we connect the Python Jupyter to MySQL database with MySQL.connector and connect() function where we pass the host, user, password, and the database name in MySQL database to work with. And create a variable for moving into MySQL database with a cursor(). Now to insert or create a table we used to execute() with SQL query in it. And with the data rame variable having the cleaned data is pushed by errors() and with Insert command. For the whole project, we only created two tables named users and Shop_product.

iii) API creation and connection setup:

As for data to be shown on the front-end, it has to be taken from the database and there should be real time connection of front-end to the database for live data so, with the libraries called Mysql, Express, body-parser, and Cors we created the pool between MySQL database and front-end in React-Js. Wherewith the help of getting () we get the data from the database by SQL query and post() to put the data taken from front-end to the MySQL database to users table. We made API data listen to the specified port number mentioned in the listen(). So the whole data is displayed in JSON format in that URL and is ready to use to build the UI for users.

iv) Front-end Creation :

For this we worked with react library of JavaScript. We divided the UI creation into two parts client and server.

Where server part connects the MySQL database and UI by creating Api URL for further data fetching and work as middle-ware in between UI and MySQL database.

And client part is the UI part with react for users interaction.

We used react-router-dom for routing to different components with links extension with their URL.

There are 4 subparts in the client module :

1. Home: In this part we have the email getting for email sending to users when there are offers. For this we used UseState for storing the data taken by the user to variable and that variable is assigned to post() to put it in the user's table with function is called by onClick() when the user submits the info. In that function, we used Axios for assigning state variables to database columns. We have links to other subparts/components of the UI in this subpart.
2. Search: As for user convenience, if a particular user wants to search the product to compare or see the offer in that particular product directly without seeing the offers or the products one by one. For this, we used useState and UseEffect for storing the data inside the UI for sync data access without buffer. We used await with fetch() with Api URL passed to it for data fetching and is stored in variables created by useState().

Here we have to work with data without altering the data in the mysql database so we use Api and useEffect() to filter the products from database that user has searched.

Here we used html tags and css with bootstrap for interactive UI creation.

3. Products: This subpart/component has the Subparts of products component that is different category of products like Mobile, laptop, camera etc. Each category has different component here , we filtered the products by product category name that we have in database. For this we used filter() with condition whether the particular product is of that category or not and then creating the product cards with map() where we took product_id as key to uniquely select the data without repeating. Here also we fetched the Api with fetch() and filter it when gets to particular component with product category.

After this we have provided the Brand select and price order features to users to select with their own interests. For this filters to apply we created two functions handleChangeBrand and handleChangeSort for price from descending to ascending order. Here we applied the compare logic for price between two products price. And for brand we compared it with database product name to user entered data with filter().

We also gave a feature where user can see how many products did got on filtering the database of their interest.

For image display we have to create folder of images in public section of the react so that the images can be fetched anywhere in the other components. Here we have saved the image with particular product_id ,so when the particular product is called the image is also fetched from this folder.

4. Graphs: The idea or objective for show the graph to the user is to show the Trends in Brand and the offers in that particular Brand from all sites where we are comparing and got the products details.

For this firstly we created Api with calculated ratings and savings , with product site name. We created that Api in server module and fetched that Api data to create dynamic graphs with get() with passing the command of 'group by product and company' . The graph gets updated whenever the data in database changes. Here we used chart.js and react-chartjs for representation of the graphs. We fetched the data for graph creation with fetch() and assigned to the data label where the data taken from Api is taken and shown on the graph. And the labels for the X and Y axes is also taken from the Json data.

v) Email Notification: the main aim of sending emails to users is to keep them updated regarding new deals and offer for this the email address from the database and send the email of most saving product by SMTP (It is a set of communication guidelines that allow the software to transmit an electronic mail over the internet is called Simple Mail Transfer Protocol. It is a program used for sending messages to other computer users based on e-mail addresses.) and the php for the backend or selecting the product writing the email.

4.Result:

System sucessfully Compare the product prices ,deals from different ecommerce websites and result is displayed on single web interface. Also system allows user to analyze and compare products using graphs and concept of web crawler is used to extract information of products available on different websites. The system will allow users to redirect to the original website of that specific product selected by the user. The following images show how system works.

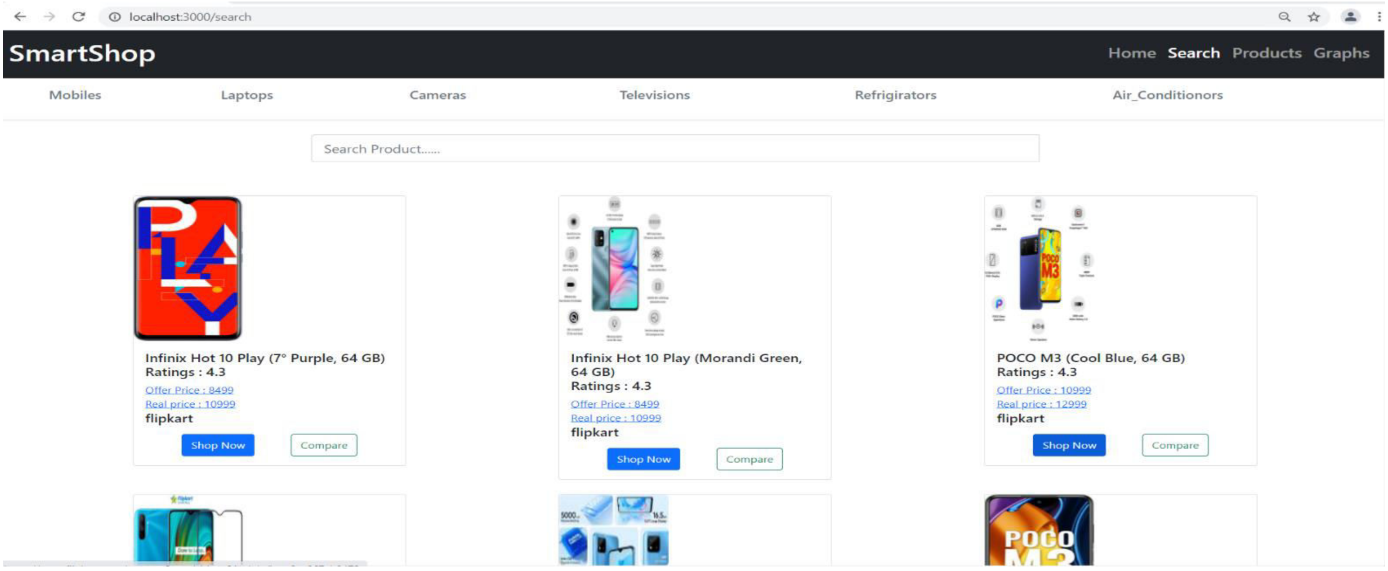
The facility of comparing products with all its specifications that belong to the same category. This will surely save buyers' efforts and valuable time.

Conclusion:

Comparison of E-commerce products using web crawling is web based system which will help users to buy products online. This website will help users to analyze prices from different e-commerce shopping websites by graphs so that they get the best deals. The website will also have

ACKNOWLEDGEMENT

We express our sincere and profound thanks to all our teachers Prof.Shalini Nigam (Head of Department), Prof. Shreesudha Kembhavi (Project Coordinator) for their valuable Guidance. We would like to thank our college GSM, Balewadi, for the boost that it has provided. We wish to thank Prof.Shalini Nigam (Information Technology Dept.) for her/his student-like enthusiasm and her guidance from time to time. We heartily thank for all her help and valuable time

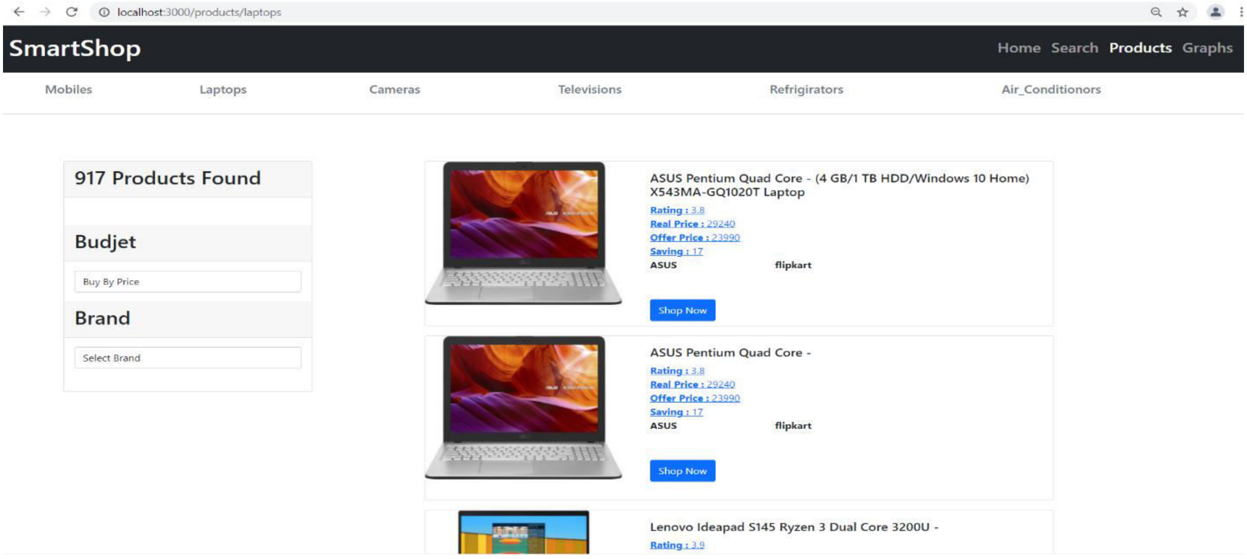


Infinix Hot 10 Play (7° Purple, 64 GB)
Ratings : 4.3
Offer Price : 8499
Real price : 10999
flipkart

Infinix Hot 10 Play (Morandi Green, 64 GB)
Ratings : 4.3
Offer Price : 8499
Real price : 10999
flipkart

POCO M3 (Cool Blue, 64 GB)
Ratings : 4.3
Offer Price : 10999
Real price : 12999
flipkart

Fig.2 Products are displayed on main web page after user queries in search bar



917 Products Found

Budjet

Brand

ASUS Pentium Quad Core - (4 GB/1 TB HDD/Windows 10 Home) X543MA-GQ1020T Laptop
Rating : 3.8
Real Price : 29240
Offer Price : 23990
Saving : 17
ASUS flipkart

ASUS Pentium Quad Core -
Rating : 3.8
Real Price : 29240
Offer Price : 23990
Saving : 17
ASUS flipkart

Lenovo Ideapad S145 Ryzen 3 Dual Core 3200U -
Rating : 3.9

Fig. 3 By clicking on the specific product button

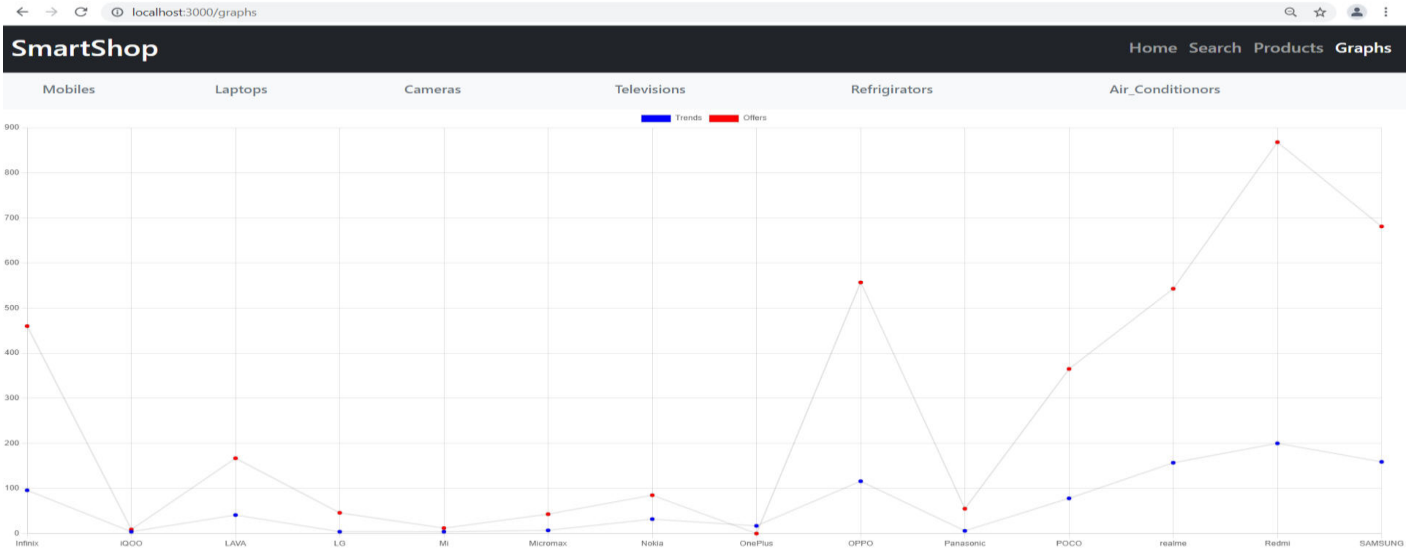


Fig. 4 Graphs for comparison of trends and offers.

REFERENCES

1.Ladislav Beranek and Radim Remes (2019) 'E-commerce network with price comparator sites', 2019 9th International Conference on Advanced Computer Information Technologies (ACIT), (10.1109/ACITT.2019.8779865), pp. [Online]. Available at: https://ieeexplore.ieee.org/ (Accessed: 5-7 June 2019).

2.Vidhi Singrodia ; Anirban Mitra; Subrata Paul (2019) 'A Review on Web Scrapping and its Applications', 2019 International Conference on Computer Communication and Informatics (ICCCI), (), pp. [Online]. Available at: https://ieeexplore.ieee.org/ (Accessed: 23-25 Jan. 2019).

3.K. Jayamalini and M. Ponnavaikko (2017) 'Research on web data mining concepts, techniques and applications', 2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET), (), pp. [Online]. Available at: https://ieeexplore.ieee.org/ (Accessed: 16-18 Feb. 2017).

4.Bo Zhao (2017) 'Web Scraping', Encyclopedia of Big Data , (10.1007/978-3-319-32001-4_483-1), pp. 1-3 [Online]. Available at: https://www.researchgate.net/ (Accessed: 08 may 2017).

5.Jawahire Nakash*, Shaikh Anas, Siddiqi Muzammil Ahmad, Ansari Mohd. Azam, Prof Tabrez Khan (2015) 'Real Time Product Analysis using Data Mining', International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 4(3), pp. [Online]. Available at: http://ijarcet.org (Accessed:).

6.Riya Shah * , Karishma Pathan , Anand Masurkar , Shweta Rewatkar , Prof. (Ms.) P.N.Vengurlekar (2016) 'Comparison of E-commerce Products using *web mining', International Journal of Scientific and Research Publications, 6(5), pp. [Online]. Available at: www.ijsrp.org (Accessed: may 2016).*

BIOGRAPHIES

	Prof.Shalini Nigam *Head of Department, Dept. of I.T., G.S moze college of engineering , Savtribai Phule Pune University, Pune, Maharashtra, India*
	Sayli Jadhav B. E Student, Dept. of I.T., G.S.Moze college of engineering, Savtribai Phule Pune University, Pune, Maharashtra, India
	Kshitija Gholap B. E Student, Dept. of I.T., G.S.Moze college of engineering, Savtribai Phule Pune University, Pune, Maharashtra, India
	Manish Singh B. E Student, Dept. of I.T