IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022

DOI: 10.17148/IJARCCE.2022.116126

Product Information Monitoring and Product Price Tracking Engine for E-Commerce

Parashiva Murthy B M[1], Aayushi Bardia[2], Arya S Gangadkar[3], Pareekshith Jain M P[4], Sarungbam Dinraj[5]

Assistant Professor, Computer Science & Engineering, JSS Science & Technology University, Mysuru, India[1]

Student, Computer Science & Engineering, JSS Science & Technology University, Mysuru, India[2-5]

Abstract: Online purchasing is gradually displacing traditional shopping methods in every manner. As a result of pandemic, it has gained traction and has become the new normal. From shoes to food, everything is now available on E-cart. People like to purchase online since there are so many possibilities in each area. Users can notice the price differences between websites and, as a result, the majority people will choose the service with the lowest price. So, in order to stay ahead of the competition, business minds are constantly devising new strategies in order to maximize the profit from each sale. They come up with enticing offers in order to attract more clients. In the E-commerce industry, dynamic pricing is currently the most popular method. Consumer may find it challenging to keep up with the pricing changes that occur every 10 minutes on average. They give the illusion that the goods is on sale on that particular by doing so. We offer a basic price tracker application in this paper that tracks the price of a product and notifies the customer when the price reaches the desired level. It will get the customer pricing from the browser and act in accordance with the customer's wish.

Keywords: MIMIC 2, EHR, Clinical notes, NLP, Bidirectional LSTM, BERT, Ensemble learning.

I. INTRODUCTION

The purpose of this document is to build a convenient and easy-to-use application displays the details of the product and their prices for the users and provides information regarding changes in price over time which helps the users to take decision, before buying the products online. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate . This document is intended for both the stakeholders and the developers of the system.

This software product will be an application system will be designed to provide information regarding changes in price over time which helps the users to take decision before buying the products online. This system is designed to allow an administrator to manage and edit details of the product if required. The software will facilitate notification to the subscriber via E- Mail. The system will contain a relational database containing a list of products, details, prices, user details and followers. Moreover, we hope to provide a comfortable user experience.

With the rapid advancements in the Electronic Commerce sector, modern consumers are becoming increasingly aware of the prospects of the various pricing strategies, which have made their purchase decisions more sophisticated. Recent literature depicts an increased tendency among online shoppers to wait for best offers on prices prior making their purchase decisions. This project aims at tracking down the prices and check whether the displayed offers upon the price are actual discounts or just a marketing stunt to increase their sales.

II. METHODOLOGY

These are the application that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product unlike the non- functional requirements. The functional requirements are:

There are two actors in this Application. They are:

Applicationadmin(owner)

Add product: admin of the application will add product details which he want to sell and the products will be displayed on application which customers can view and buy theproduct.

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022
DOI: 10.17148/IJARCCE.2022.116126

Set offer on the product: Admin of the application will set offers on the products which are eligible. so that customers will be attracted to buy the product .the offered products can be the most selling products or least selling products.
Set discount on the products: After setting offers on the product, admin will decide to set discounts on the products if eligible.
Display on the shopping application: All the products will be displayed on one application, so that it is easy to view and buy for customers.
End user (or) customer:
View products: Endusers can view products in one application so that it is easy to view, compare the products and also can buy in one application
View offers and Discount: The offered and discount products can be viewed by the customers and can buy the products if the price is attractively priced.
Check whether offered discount is genuine or not: The customers can check the product discount is the real discounted price or its simply for marketing.
Buying product: if the customers are satisfied with the product and discounted price customer may buy the product.

III. ARCHITECTURE

We have deployed the website using 3-tier architecture.This separates applications into three logical and physical computing tiers. The chief benefit of three-tier architecture is that because each tier runs on its own infrastructure, each tier can be developed simultaneously by a separate development team and can be updated or scaled as needed without impacting the other tiers.

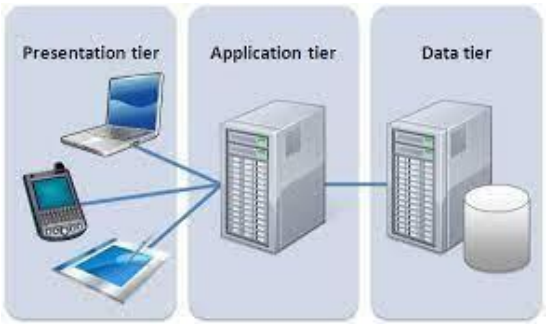

Fig. 1 Three Tier Architecture

Presentation tier:
The presentation tier is the user interface and communication layer of the application, where the end user interacts with the application. Its main purpose is to display information to and collect information from the user. This top-level tier can run on a web browser. Web presentation tier is developed using HTML, CSS and Bootstrap.
Application tier:
The application tier, also known as the logic tier or middle tier, is the heart of the application. In this tier, information collected in the presentation tier is processed-some times against other information in the data tier –u sing business logic, a specific set of business rules. The application tier can also add, delete or modify data in the data tier. We have used .NET for this tier implementation.
Data tier:
The data tier, sometimes called database tier, data access tier or back-end, is where the information processed by the application is stored and managed. This can be relational database management system such as PostgreSQL, MySQL, MariaDB, Oracle, DB2, Informix or Microsoft SQL Server, or in aNoSQL Database server such as Cassandra, CouchDB or MongoDB. We have developed this tier using MySQL

IV. SYSTEM IMPLEMENTATION

1. Creating web pages for E-commerce application.
2. Creating local database for storing data.
3. Creating ODBC connection.
4. Creating application which manages the crawled data in the background.
5. Primary unit and integration testing at local machine level.
6. Hosting the website.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022

DOI: 10.17148/IJARCCE.2022.116126

7. Final unit testing and integration testing in the cloud host level.

Creating web pages for E-commerce application: In the client's side we need to create two main pages

1. Page for browsing items.
2. Cart for the user.

In the server's side we need to create, Page for item management of the website.

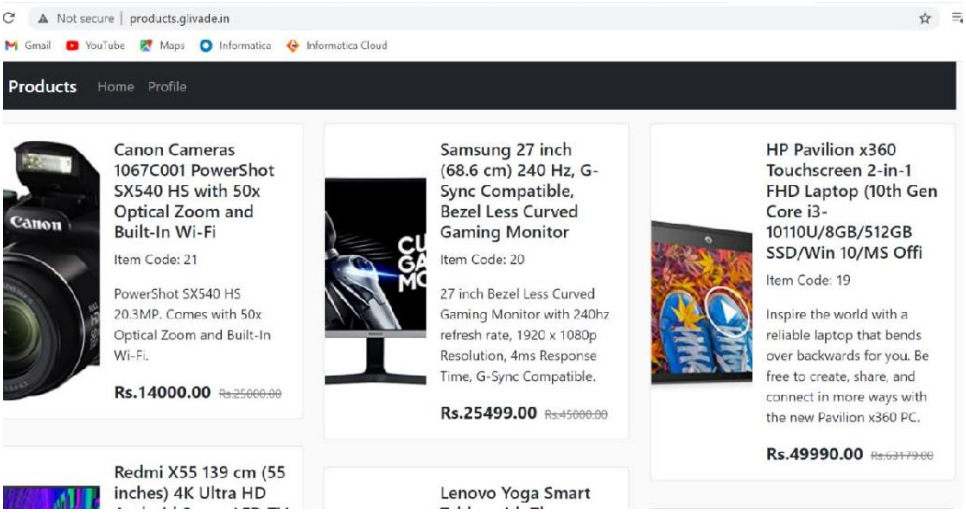

Fig. 2 Browsing Page

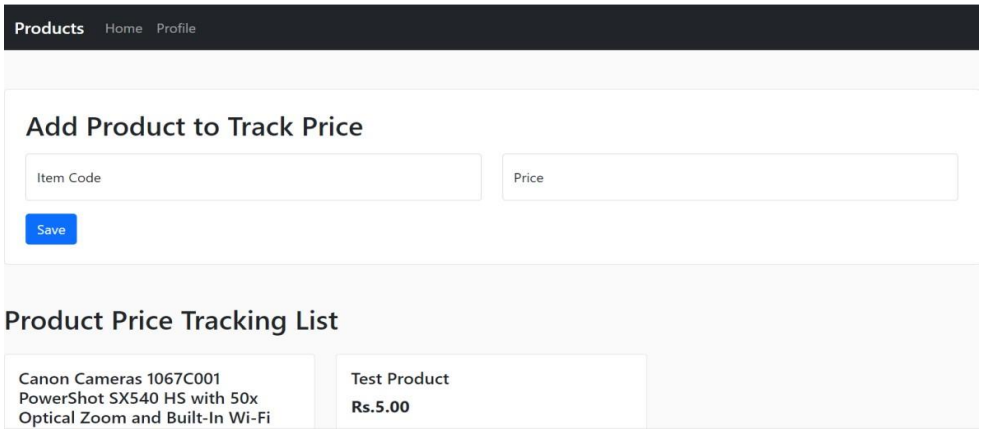

Fig. 3 Cart for the use

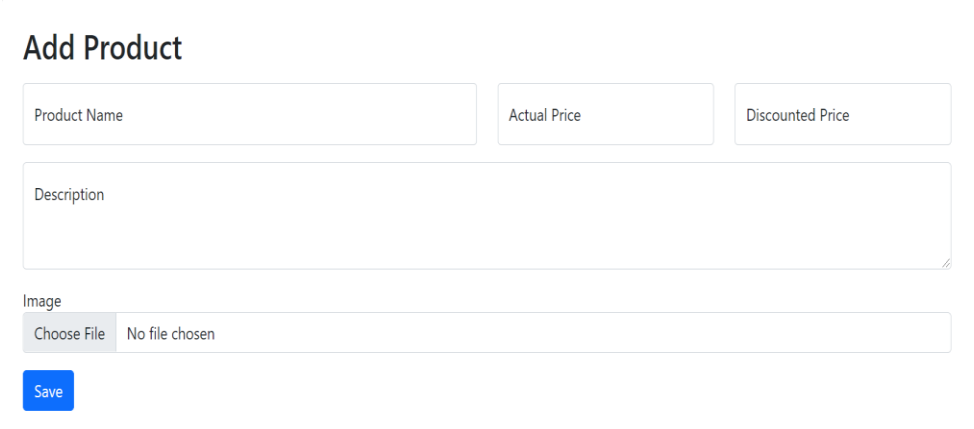

Fig. 4 Administration page

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022

DOI: 10.17148/IJARCCE.2022.116126

Creating local database for storing data

1. Database is created using MySQL.
2. At the local machine level database is constructed using Microsoft SQL server.

At the cloud we have to create database which can be accessed at any remote location using Heidi SQL.

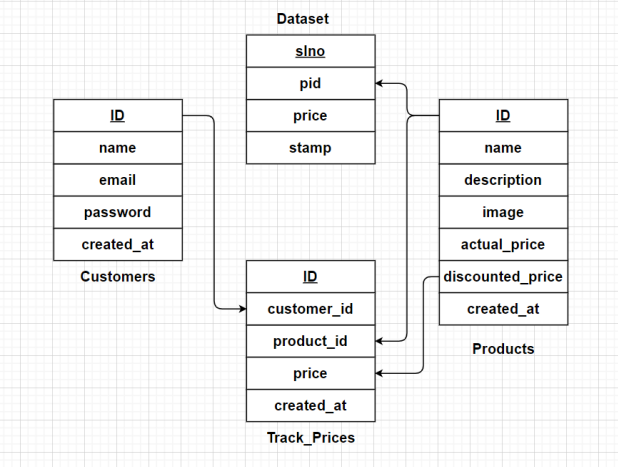

Fig. 4 Database

Creating ODBC connection

1. We have to set up an ODBC connection in order to get and store data at database.
2. Here we have used Connection string method to setup the connection.
3. Get the connection string by using ODBC data sources.
4. We have to create a new data source here and thereby get connection key.
5. Get the necessary field information from the DSN file which will be saved in the previous step.
6. Put the correct values in to the connection string.
7. Place the connection string inside XML configuration file with suitable source and destination.

Creating application which manages the crawled data in the background

1. We have to create a web crawler application which has to run continuously in the background in order to fetch data from the website in the real time.
2. HTML agility pack which is an open source software, is used for crawling data from the website.
3. We have to get the page source from the website and trim the data into just the discounted price.
4. Onceweclickonthestartbuttoncrawlerwillstartcrawlingthedatafromthewebsite.
5. With tracker delay timeout we can set the interval in which crawler fetches the data.

Once the data are fetched it will be displayed in the Web crawled data section.

1. It will track all the price and check if any price is less than the expected price.
2. If the price has fallen below threshold then it will identify the user id and send the email alert to the user who has put that data in his/her cart.
3. We have used System.Net.Mail for the mail alert notifications which contain the classes used to send mail to SMTP servers.
4. If the mail is sent then the pop up window will be displayed on the dashboard.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022

DOI: 10.17148/IJARCCE.2022.116126

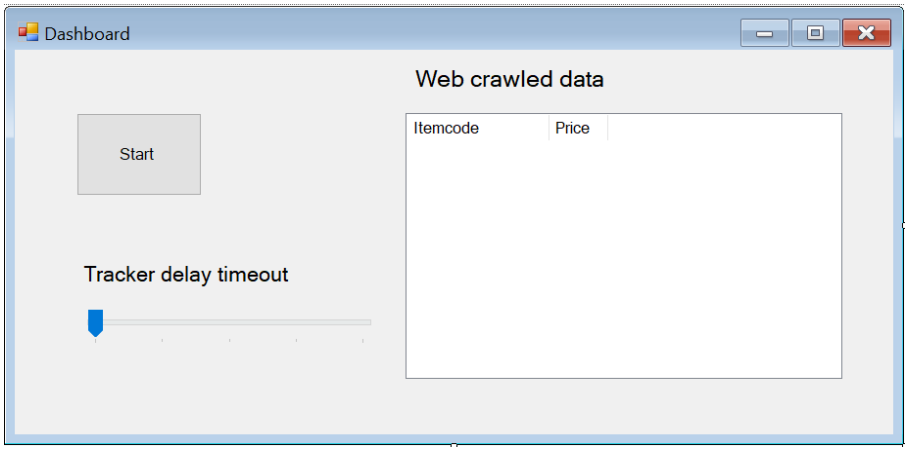

Fig. 5 Dashboard Application

Primary unit and integration testing at local machine level: Initial level unit and integration testing. All the web pages and database have to be tested individually. All the components are connected using ODBC and system is tested as a whole.

Hosting the website: Hosting the website with GoDaDDY. We have to search for the wished URL, if it is available then it will display the price for that URL. After selecting the URL it will display the plans for subscription. We need to create the zip file containing all the codes which developed and tested in local machine. We can upload the zip file to the space we subscribed in the GoDaDDY. We need to create separate database at the GoDaDDY using word press.

Testing: Final unit testing and integration testing will be done in the cloud host level.

V RESULTS

When the current price of the product falls below the customer's threshold price, the user will receive an e-mail alert message and a window will appear indicating that the alert has been delivered.

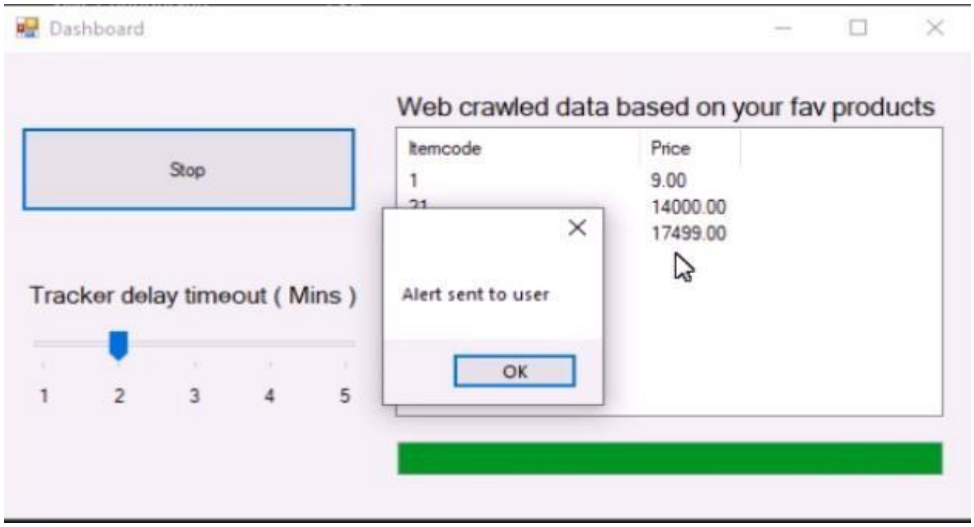

Fig. 6 Background application

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022

DOI: 10.17148/IJARCCE.2022.116126

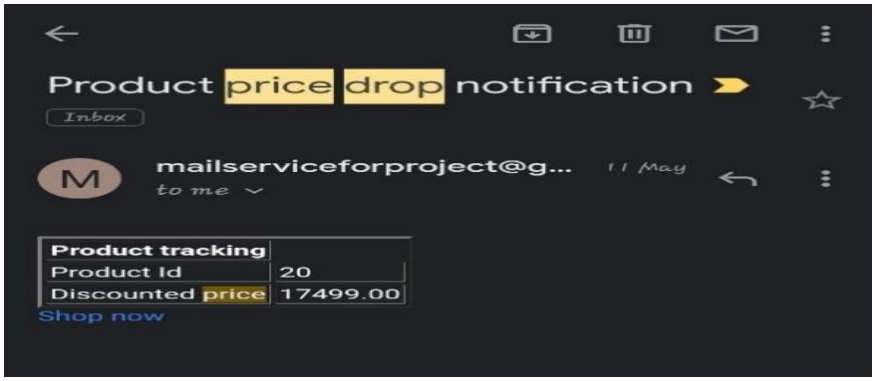

Fig. 7 Mail alert

VI CONCLUSION AND FUTUREENHANCEMENTS

The idea in this article is a real-time program that tracks the website's dynamic price behavior. The customer's pricing will be compared to the current price by crawling the real-time price from the website, and if the condition is met, an alert will be issued to the user telling them of the new price and a link to the product on the website. By hosting the crawler application on the cloud and making it available to all users, the job may be expanded to a larger area. This solution eliminates the need for the crawler to be installed on the local machine. If we use it to track the prices of other websites, it can also be utilized as third-party software.

REFERENCES

[1] Ahmed Z. Gabr, Ahmed A. Helal,2018 “Dynamic pricing; different schemes, related research survey and evaluation”.
[2] Documentation | HTML agility pack https://html-agility-pack.net/documentation.
[3] Documentation for visual studio codehttps://code.visualstudio.com/docs.
[4] Getting Structured Data from the Internet Running Web Crawlers/Scrapers on a Big Data Production Scale Jay M. Patel(2020).
[5] Jianxia Chen,Ri Huang, 2013, “A price comparison system based on IOT”
[6] Microsoft docshttps://docs.microsoft.com/en-us/dotnet/api/system.net.mail.
[7] Nathan R.J., Yeow P.H.P., 2011, Crucial Web Usability Factors of 36 Industries for Students – A Large Empirical Study “Journal of Electronic Commerce Research”.
[8] Powershell and ODBC | Working with the database – Richard Thomas Edwards.
[9] Quanyin Zhu ,2011, “Research on the Price Forecast without Complete Data Based on Web Mining”.
[10] Technical help document – HeidiSQLhttps://www.heidisql.com/help.php.
[11] Web hosting from GoDADDYhttps://www.googleadservices.com.
[12] Wikipedia, the free encyclopediahttps://en.wikipedia.org/wiki/Main_Page