AI Chat Bot for E-commerce Website

By

SNEHA SADANAND KUDCHADKAR

P.R. No.:202111979

Internship report Submitted in partial fulfilment of the requirements for the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER ENGINEERING

OF GOA UNIVERSITY

Under the guidance of

Mr. Teslin Jacob
(Assistant Professor)

August 2024

COMPUTER ENGINEERING

GOA COLLEGE OF ENGINEERING

(Government of Goa)

Farmagudi, Ponda, Goa - 403401

INDEX

DECLARATION

I hereby declare that the internship work entitled "AI Chat bot for E-commerce website" is an authentic record of my own work carried out at Persistent Systems Ltd. as partial requirements for the award of degree of B.E. of Computer Engineering - Goa University, under the guidance of Ms. Apurva Tiloji and Mr. Teslin Jacob, during July2024 – August 2024

_____

Name of Student: Sneha Kudchadkar

Student ID: 211105086

Date:

Certified that the above statement made by the student is correct to the best of our knowledge and belief.

_____

Name: Mr. Teslin Jacob

Designation: Assistant Prof. GEC
Faculty Coordinator

_____

Name: Ms. Apurva Tiloji

Designation: Senior Engineering Lead
Industry Coordinator

Acknowledgement

I would like to express my deep gratitude and appreciation to all those who have contributed to the successful completion of my internship and this report. This journey would not have been possible without the support, guidance, and encouragement of several individuals and organizations.

First and foremost, I would like to thank Ms. Apurva Tiloji and Mr. Teslin Jacob for their mentorship and continuous support throughout my internship. Their insights, expertise, and patience have been invaluable in shaping my learning experience.

I am also grateful to the entire Persistent Systems Ltd. team for providing me with the opportunity to be part of their workforce. The exposure to real-world projects and the collaborative environment was truly enriching.

I extend my sincere thanks to my fellow interns for their collaboration, which made my work environment positive and enjoyable.

Special thanks go to my friends and family for their unwavering support, encouragement, and understanding during this period.

Finally, I want to express my gratitude to the professionals and resources in the industry that I accessed for research and inspiration during this internship.

This internship has been a significant milestone in my academic and professional journey, and the above-mentioned individuals and organizations played a pivotal role in making it a valuable and enriching experience.

Sneha S. Kudchadkar

Goa College of Engineering

Brief Overview

Over a two-month internship at Persistent Systems Ltd., active involvement in various projects and responsibilities was maintained to align with the company's objectives.

In the following sections, an exploration of the industry in which Persistent Systems operates, a review of the specific tasks undertaken, discussions about challenges faced, suggestions for improvement, and insights regarding future work prospects will be presented. Supplementary materials, such as a daily activity diary and a list of references, have also been included.

This internship has provided a valuable learning opportunity, allowing the application of theoretical knowledge gained during academic studies to real-world situations. The intention is to share the key experiences and insights acquired during the tenure at Persistent Systems Limited.

Industrial overview:

About Persistent System ltd.

Persistent Systems Ltd is a software development company that develops and maintains software systems and solutions. The company's service portfolio comprises digital strategy and design, software product engineering, CX transformation, cloud and infrastructure, intelligent automation, enterprise information technology security, data and analytic, enterprise applications and integration services. It is an Indian multinational technology services company. It was listed on the Bombay Stock Exchange and the National Stock Exchange. Persistent Systems is a trusted Digital Engineering and Enterprise Modernization partner, combining deep technical expertise and industry experience to help their clients anticipate what's next and act.

Persistent Systems' offerings, proven solutions and innovative partner ecosystem create unique competitive advantage for their clients by giving them the power to see beyond and rise above. The company operates in the US, France, India, South Africa, Singapore, Japan and Australia.

Industry: - Information Technology

Founded: - 1990 in Pune

Founder: - Anand Deshpande

Headquarters: - Pune, Maharashtra, India

Revenue: - Rs 8.350 crore [1.1B dollars]

Website: - www.persistent.com

Its corporate values involve being Ingenious, Responsible, and Persistent & Confident. Persistent has crossed $ 1 Billion in Annual Revenue. It is a good company to explore your growth, have proper work life balance, learning and immense hands-on opportunities

Persistent systems have a huge pool of services offered by the company with the expertise in the following industries: -

Banking and Financial - As traditional institutions and fintech alike strive for growth and to improve margins, reducing complexity and optimizing operations is imperative to deliver differentiated experiences. We help our clients to achieve this through the implementation of a digital mosaic of disruptive technologies empowering them to deliver hyper-personalized financial services that are valued by their customers.

Insurance - Our Digital Insurance solutions assist firms in fulfilling rising consumer demands and streamlining operational procedures in a highly competitive industry environment.

Healthcare - Improving patient outcomes while reducing costs can only be achieved by combining technology innovation with process transformation. We drive transformational programs with industry leading software and solution partners that span the areas of patient engagement, care delivery, population health management, operations, as well as data, analytics, and business intelligence.

Life sciences – Persistent's Life Sciences division helps analytical instrument, medical devices, and pharmaceutical companies to better utilize technology to bring new products and solutions to market. It builds the software

that delivers the connected digital lab of the future and enables pharmaceutical companies to execute decentralized trials efficiently. Their extensive industry experience includes supporting 3 of the 5 largest contract research organizations (CROs) and 6 of top 10 device manufacturers.

Industrial - Persistent integrates the industrial value chain. They deliver business value by tightly integrating legacy design and engineering, manufacturing and operational systems with industry leading processes, software and analytic capabilities. The result is an integrated technology stack that ensures a seamless flow of consistent, correct and complete data from product design and engineering to manufacturing and beyond.

Software and Hi-tech - Technology products and business models don't stay current for long due to the rapid pace of innovation, accelerated development cycles, and disruptive business models. This has increased both the importance and complexity of managing innovation, engineering, and management. With services across the entire product development spectrum, Persistent is the ideal engineering partner to co-innovate around new product development and as a transformation catalyst for existing products and businesses.

Telecom and Media - Persistent helps communications and media companies struggling to redefine the customer experience, capitalize on digital convergence and launch new business models. We deliver solutions that drive agility and responsiveness by helping organizations develop new products faster, manage costs in business operations and derive insights ffrom performance data. Our services span software engineering, network management and the distributed edge. We bring a rich partner ecosystem and the right skills that enable industry transformation while managing the risks that come from rapid change.

About Persistent Systems Ltd. Verna, South Goa

Persistent Systems Ltd. is in Verna, South Goa. There are at least 19 Software companies in Verna, out of which this Software company has an overall rank of 1. Address of the Software company is L-44, Software Technology Park, Verna Industrial Estate, Verna, Goa 403722. It comprises of two buildings namely Bhaskar and Charak. Bhasker was established first later a semiconductor company got merged and that building was named as charak.

Established in 1998, Persistent Systems Ltd. has gained immense expertise in offering. IT Infrastructure Services, Product Engineering Development Services, Industrial Design & Engineering Services to the clients.

Persistent Systems' Role in Ecommerce:

Persistent Systems has a proven track record of helping ecommerce companies:

- **Digital Commerce Enablement:** They assist businesses in building robust online platforms, integrating with various channels, and creating compelling digital storefronts.
- **Data and Analytics:** Persistent helps extract valuable insights from vast amounts of ecommerce data, enabling businesses to make data-driven decisions and optimize their operations.
- **Machine Learning and AI:** Their expertise in these areas helps ecommerce companies personalize customer experiences, improve recommendations, and optimize marketing campaigns.
- **Cloud Transformation:** Persistent assists businesses in migrating their ecommerce operations to the cloud, enhancing scalability, flexibility, and cost-efficiency.
- **Supply Chain Optimization:** They help streamline supply chain processes, reducing costs and improving delivery times

Review of work done:

Project Objectives: To develop and deploy an AI-powered chat bot capable of efficiently and accurately assisting users in product search and discovery within an ecommerce platform, thereby enhancing customer satisfaction, increasing product discoverability, and driving sales conversion.

Project Steps: AI Chat bot for Ecommerce Search

Project Initiation and Planning

- Define project scope, goals, and objectives.
- Identify target audience and customer personas.
- Conduct a competitive analysis of existing chat bot solutions.

- Assemble project team with necessary skills (data scientists, NLP experts, developers, UI/UX designers).

Data Collection and Preparation

- Gather relevant product data, customer data, and historical search data.
- Cleanse and preprocess data to ensure accuracy and consistency.
- Create training and testing datasets for the AI model.

Natural Language Processing (NLP) Model Development

- Select appropriate NLP techniques (e.g., tokenization, stemming, and lemmatization).
- Develop a language model to understand user queries and intents.
- Create an entity recognition model to identify product names, attributes, and other relevant information.

Chat bot Development

- Design chat bot conversation flow and user interface.
- Integrate NLP models into the chat bot to enable natural language understanding.
- Develop chat bot responses and actions based on user intents.
- Implement knowledge base for product information and FAQs.

Integration with Ecommerce Platform

- Integrate the chat bot into the ecommerce website or mobile app.
- Ensure seamless interaction between the chat bot and product catalog.
- Implement real-time updates for product availability and pricing.

Testing and Refinement

- Conduct thorough testing to identify and fix errors.
- Gather user feedback to improve chat bot performance.
- Iterate on chat bot responses and logic based on user interactions.
- Monitor chat bot performance metrics (e.g., accuracy, response time, customer satisfaction).

Deployment and Monitoring

- Deploy the chat bot to the live environment.
- Set up monitoring and analytics to track chat bot performance.
- Continuously collect user feedback and data for model improvement.

Ongoing Optimization

- Analyze chat bot performance data to identify areas for improvement.
- Retrain NLP models with new data to enhance accuracy.
- Expand chat bot capabilities based on user needs and business goals.

Project Implementation:

The aim of the project was to create an AI-powered chat bot that significantly improves the product search experience on an ecommerce platform.

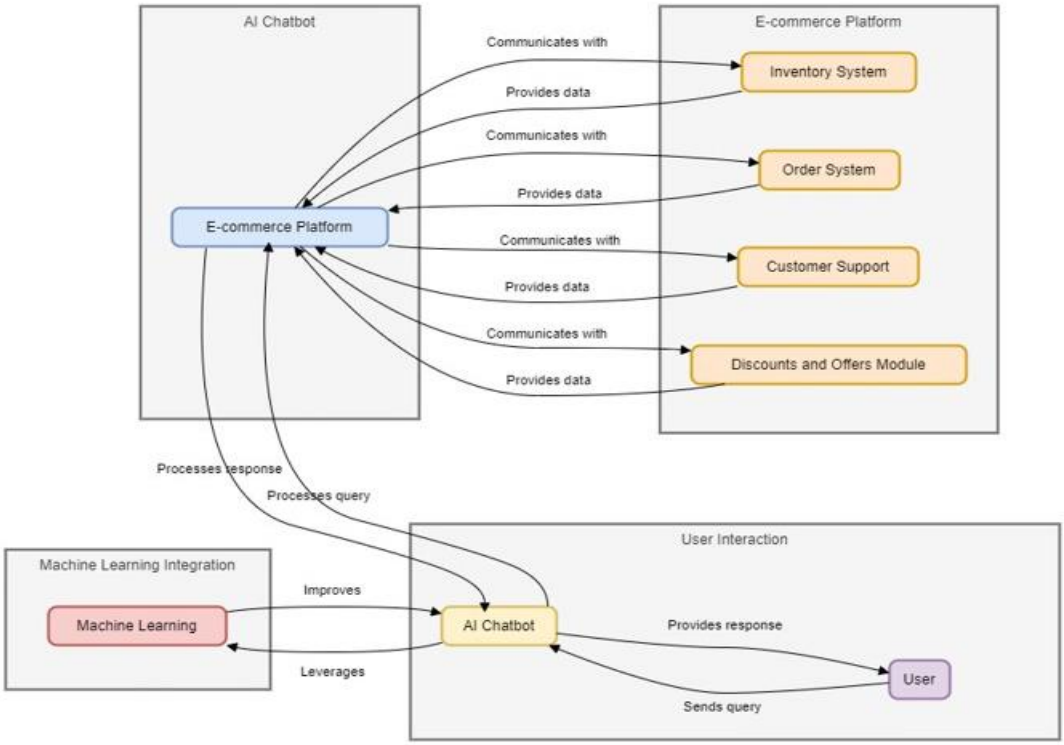
By developing this chat bot, we aim to:

- Enhance product discoverability for customers.
- Increase customer satisfaction through efficient and personalized assistance.
- Drive sales by facilitating the purchasing process and recommending relevant products.
- Optimize the ecommerce platform by gathering valuable user data and insights.

Ultimately, the chat bot should become an asset to the business by contributing to overall growth and profitability.

While many ecommerce platforms have successfully deployed chatbots to enhance customer service, this project takes a different approach by focusing specifically on product search**.** Instead of providing general support, our AI chat bot will be designed to assist users in efficiently finding desired products within the ecommerce platform. This specialized focus aims to improve product discoverability, reduce search friction, and ultimately drive sales by offering a more intuitive and personalized shopping experience

Architecture design of the project:



Explanation of architecture diagram:

The diagram illustrates the architecture of an AI-powered chat bot integrated into an e-commerce platform. Here's a breakdown of its components and interactions:

User Interaction:

- The user initiates a conversation with the chat bot by sending a query or question.

AI Chat bot:

- The chat bot acts as an interface between the user and the e-commerce platform.
- It employs machine learning techniques to understand and respond to user queries effectively.
- It uses natural language processing to interpret complex or nuanced language.

E-commerce Platform:

- The chat bot communicates with various subsystems within the e-commerce platform, including:
    - **Inventory System:** To check product availability and stock levels.
    - **Order System:** To provide updates on order status and tracking.
    - **Customer Support:** To handle customer inquiries, support requests, and resolve issues.
    - **Discounts and Offers Module:** To inform users about active promotions and offers.

Interaction Flow:

- When the user sends a query, the chat bot processes it and performs a product search.
- It communicates with the e-commerce platform's internal systems to obtain necessary information.
- The chat bot then provides a response to the user, incorporating the information it gathered.

Machine Learning Integration:

- The chat bot leverages machine learning to continuously improve its responses and recommendations.
- It adapts to user preferences over time, providing more personalized and relevant information.

Dependencies

1. Pandas (pandas==1.3.5)

- **Purpose**: Pandas is a powerful library for data manipulation and analysis. It provides data structures like Data Frames and Series, which are essential for handling and analysing structured data (like CSV files, Excel sheets, and SQL databases).
- **Usage in Your Project**: You use Pandas to load, manipulate, and analyse product data stored in a CSV file. The data is then used to match user queries to product descriptions.

2. Flask (Flask==2.0.3)

- **Purpose**: Flask is a lightweight web application framework for Python. It's designed to be simple, flexible, and easy to use, making it popular for building web applications and APIs.
- **Usage in Your Project**: Flask serves as the backend framework for your chat bot. It handles HTTP requests from the frontend (PHP part of your project), processes them (using the product data and model), and sends responses back to the frontend.

3. Transformers (transformers==4.18.0)

- **Purpose**: The transformers library, developed by Hugging Face, provides pre-trained models for various natural language processing (NLP) tasks. These models can perform tasks such as text classification, translation, summarization, and more.
- **Usage in Your Project**: You use the transformers library to load a pre-trained BERT tokenizer, which is used to tokenize user queries into tokens (words or sub-words) that the model can process for understanding and matching products.

4. Torch (torch==1.10.2)

- **Purpose**: PyTorch (imported as torch) is a popular deep learning framework. It provides tensor computations (like NumPy) with strong GPU acceleration and a dynamic computational graph that makes building neural networks more flexible and easier.
- **Usage in Your Project**: While not explicitly used in the code you provided, torch would be required if you're using deep learning models for tasks like NLP (Natural Language Processing), especially if you're leveraging the models from the transformer's library. Py Torch is the underlying framework that many of these models are built upon.

5. FuzzyWuzzy (fuzzywuzzy==0.18.0)

- **Purpose**: FuzzyWuzzy is a library for fuzzy string matching. It allows you to find approximate matches between strings by computing a similarity score between them. This is useful when you want to match user input with existing strings that may not exactly match but are close enough.
- **Usage in Your Project**: You use FuzzyWuzzy to match the tokenized user query against product descriptions in your data. This helps in finding

the most relevant products based on the user's input, even if the input isn’t an exact match.

6. Werkzeug (werkzeug==2.0.3)

- **Purpose**: Werkzeug is a WSGI (Web Server Gateway Interface) utility library for Python. It's used by Flask to handle the underlying request and response objects. It also provides utilities for routing, debugging, and error handling.
- **Usage in Your Project**: Werkzeug is used under the hood by Flask to manage HTTP requests and responses, as well as various other web-related tasks. You typically don’t interact with Werkzeug directly, but it is critical for Flask’s operation.

7. Scikit-learn (scikit-learn==0.24.2)

- **Purpose**: Scikit-learn is a widely used machine learning library in Python. It provides simple and efficient tools for data mining, data analysis, and machine learning. It includes algorithms for classification, regression, clustering, and dimensionality reduction, among others.
- **Usage in Your Project**: While scikit-learn is often used for building machine learning models, in your project, it may be used for tasks like tokenizing and transforming data, possibly for pre-processing before using it with models from the transformer's library or for matching with FuzzyWuzzy.

Project Structure:

```
project/
│
├── app.py                  # Main application script
├── data/
│   └── products.csv        # CSV file containing product data
├── templates/
│   └── index.html          # HTML template for the web interface
├── static/
│   └── css/                # CSS files (optional)
├── requirements.txt        # Python dependencies
└── README.md               # Project documentation
```

1. Project Root (project/)

- The root directory of your Flask application. This is where your main application script and other key files are located. This directory contains everything needed to run your project.

2. app.py

- **Purpose**: This is the main application script for your Flask project.
- **Role**: It contains the core logic of your application, including routes that handle incoming requests, interactions with data, and rendering templates. When you start your Flask server, this script is executed, and it runs the application.
- **Example Content**: Defines routes (URLs) and connects them to functions that process requests and return responses. For example, it might have routes for displaying the home page, handling user queries, and returning JSON responses for chat bot interactions.

3. data/

- **Purpose**: This directory is used to store data files that your application needs to access and use.
- **Role**: In your case, it contains products.csv, which holds the product information that your application will search, and display based on user queries.

- **Example Content**: The products.csv file might contain columns such as product names, descriptions, prices, and availability, which your app reads and uses to respond to user queries.

4. templates/

- **Purpose**: This directory contains HTML templates for your web application's front-end.
- **Role**: Flask uses Jinja2 as its template engine, which allows you to create dynamic HTML pages. The index.html file here is a template that serves as the main interface for your application, allowing users to interact with the chat bot.
- **Example Content**: The index.html might include the structure of your webpage, placeholders for dynamic content (e.g., chat bot responses), and integration with CSS and JavaScript for styling and functionality.

5. static/

- **Purpose**: This directory is where you store static files like CSS, JavaScript, and images.
- **Role**: Static files are served directly to the browser without any server-side processing. The CSS / subdirectory would contain any stylesheets you use to style your HTML templates.
- **Example Content**: The CSS/ directory might include files like styles.css, which could define the look and feel of your application, including layouts, colours, fonts, etc.

6. requirements.txt

- **Purpose**: This file lists all the Python libraries that your project depends on, along with their specific versions.
- **Role**: It allows others (or automated systems) to install all necessary dependencies for your project in one step using pip. This ensures that your project has a consistent environment across different setups.
- **Example Content**: It includes lines like Flask==2.0.3, which tells pip to install Flask version 2.0.3.

7. README.md

- **Purpose**: A markdown file that provides documentation for your project.

- **Role**: The README typically explains what the project is, how to set it up, and how to use it. It might also include information about the project structure, dependencies, and any other important details.

Example Content: It could contain sections like "Introduction," "Installation," "Usage," and "Contributing," guiding new users or developers on how to work with your project.

Usage

This application is designed to help users search for products based on specific queries related to product features. For example, a user might enter a query like "high storage phone with good camera," and the application will return the most relevant product based on the available descriptions.

API Endpoints

/ [GET]

- **Description**: Serves the homepage of the application.
- **Response**: Renders the index.html template.

/query [POST]

- **Description**: Processes a user query and returns the best matching product.
- **Request**: Expects a JSON payload with a query field. The query is a string that describes the features or characteristics the user is looking for in a product.
- **Response**: Returns JSON data containing details of the best-matching product, including:
    - Product name: The name of the product.
    - Price: The price of the product.
    - Availability: Whether the product is in stock.
- Description: A brief description of the product.

Example Request:

```json
{
  "query": "high storage phone with good camera"
}
```

Example Response:

```json
{
  "product_name": "SuperPhone 3000",
  "price": 999,
  "availability": "In Stock",
  "description": "A phone with 128GB storage and a 48MP camera."
}
```

KEY COMPONENTS

Loading Data

```python
data = pd.read_csv('data/products.csv')
```

-> **Purpose**: Load the product data from a CSV file into a pandas DataFrame.

Tokenizing Queries

```python
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
tokens = tokenizer.tokenize(query)
tokenized_query = " ".join(tokens)
```

Purpose: Tokenizes the user query using a pre-trained BERT tokenizer**.**

Product Matching

```python
best_match = process.extractOne(tokenized_query, descriptions)
```

Purpose: Finds the best matching product description using fuzzy string matching.

Flask Routes

- **index ()**

```python
@app.route('/')
def index():
    return render_template('index.html')
```

Purpose: This route handles GET requests to the root URL (/). It renders and returns the index.html template, which serves as the homepage of the application.

- **query product ()**

```python
@app.route('/query', methods=['POST'])
def query_product():
    query = request.json['query']
    product = find_product(query)
    ...
    return jsonify(product_data)
```

Purpose: This route handles POST requests to the /query endpoint. It processes the user's query, finds the most relevant product using the find product function, and returns the product details in a JSON format.

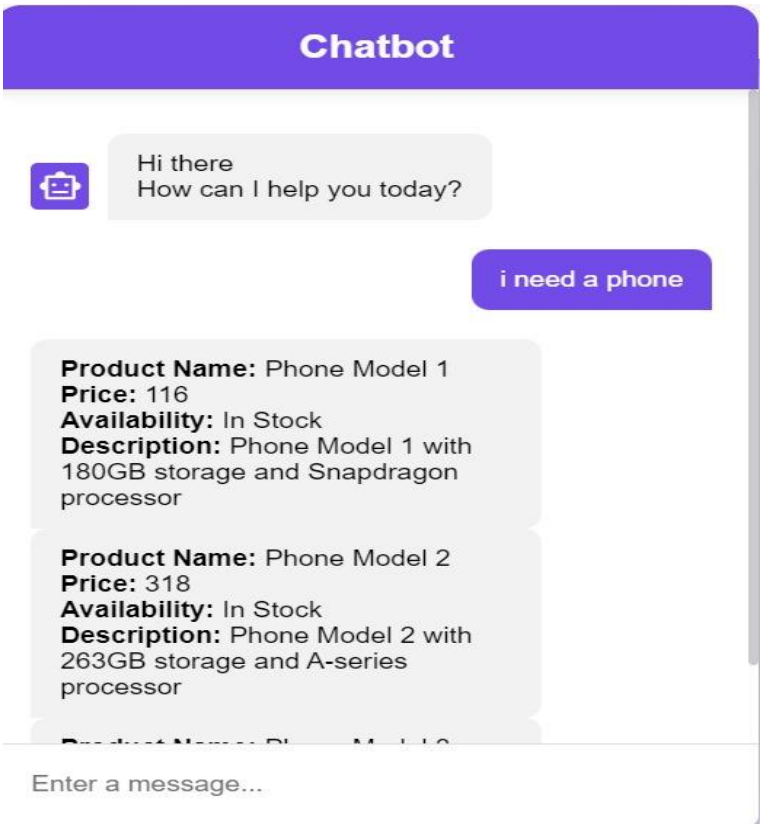
Chatbot Interface

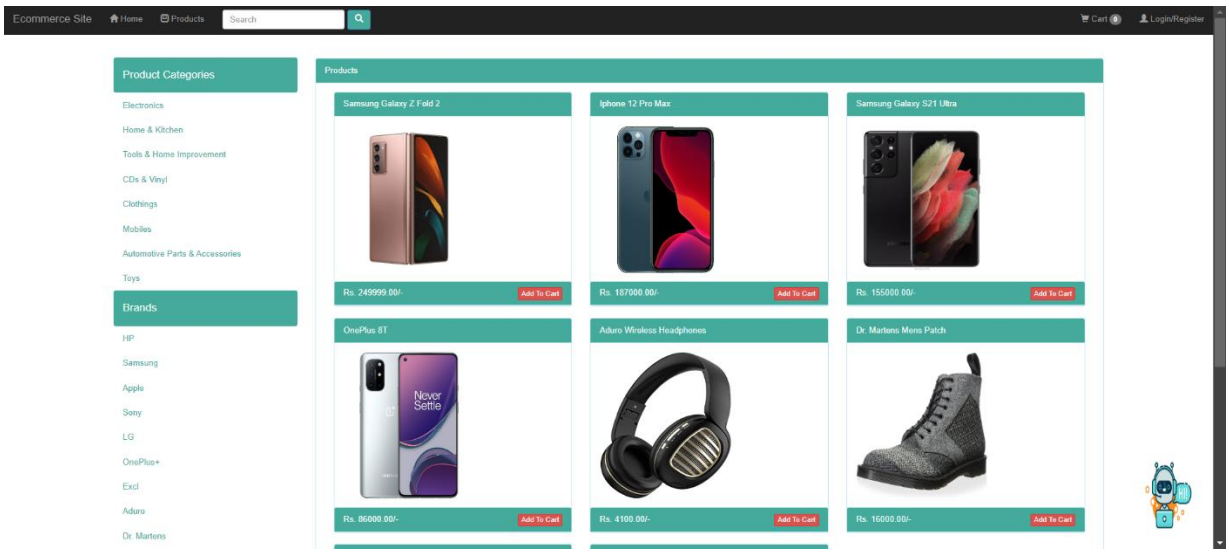
Running the Application

To run the Flask-based product search application, follow these steps:

1. **Navigate to Your Project Directory**:
   cd E:\xampp\htdocs\ecommerce\project
2. **Start the Flask Development Server:**
   python app.py
3. **Access the Application in Your Web Browser:**
   http://127.0.0.1:5000

Output photos:





Difficulties faced during internship:

1. Insufficient prerequisite knowledge about the software development and various tools and technology involved.

2. The extensive and numerous codebases made it challenging to locate and

rectify errors

3. Ensuring the availability of high-quality and sufficient product data, customer data, and search history is crucial for training the AI model effectively.

4. Accurately interpreting diverse and complex user queries can be challenging, especially when dealing with ambiguous or misspelled terms.

Conclusion And Future scope for project:

Conclusion:

- The development of an AI-powered chat bot for ecommerce search has demonstrated the potential to significantly enhance the shopping experience.
- By effectively addressing user queries, providing personalized recommendations, and streamlining the search process, the chat bot has proven to be an asset for ecommerce platforms.
- The integration of natural language processing techniques has enabled the chat bot to understand and respond to user inputs with accuracy and efficiency.
- While challenges related to data quality, NLU, and system integration were encountered, the project successfully overcame these obstacles to deliver a functional and user-friendly chat bot.

Future scope for project:

The AI chat bot for ecommerce search offers several avenues for future exploration and improvement:

- **Enhanced Natural Language Understanding:** Further development of NLP capabilities can lead to more nuanced and accurate interpretation of user queries.
- **Visual Search Integration:** Incorporating image recognition technology to allow users to search for products based on visual cues.
- **Sentiment Analysis:** Analyzing customer sentiment to identify areas for improvement in product descriptions or customer service.
- **Multilingual Support:** Expanding the chat bot's language capabilities to cater to a wider audience.
- **Voice-Enabled Search:** Integrating voice recognition technology to provide hands-free search functionality.
- **Predictive Search:** Anticipating user needs and suggesting products proactively.

- **Personalization:** Leveraging advanced machine learning techniques to create highly personalized shopping experiences.

References for project:

- **Articles available based on idea:**

https://blog.hubspot.com/website/python-ai-chat-bot

https://www.upgrad.com/blog/how-to-make-chatbot-in-python/

https://www.analyticsvidhya.com/blog/2021/12/creating-chatbot-building-using-python/

file:///C:/Users/user/Downloads/Web_based_E-commerce_system_integrated_with_Chatbo.pdf

GitHub libraries:

https://github.com/balajisrinivas/ChatBot-using-Keras-TkinterGUI/blob/main/train_chatbot.py

https://github.com/Coding-with-Adam/Dash-by-Plotly/blob/master/AI/EmbedChain-Chatbot/app.py

YouTube references:

https://youtu.be/a37BL0stIuM?feature=shared

https://youtu.be/t933Gh5fNrc?feature=shared