**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

# Weekly Report

# Foundation Internship

## *Project Title: Smart Shop AI Assistant*


Instructor: Kim Ngoc Bach

Student Name : Le Tran Quoc Bao

Student ID : B22DCVT050

Class : E22CQCN05-B


*Hà Nội - 2025*

**IINTERNSHIP BASE REPORT – WEEK 10**
**WEEK 10 INTERNSHIP REPORT**

## 1. Overview of This Week's Work

During the tenth week of the internship, I focused on enhancing the chatbot by adding image-based product recognition. The system was upgraded to handle three main user interaction scenarios: text-only messages, image-only messages, and messages containing both text and images. Image processing was implemented using image embeddings and similarity comparison between the uploaded image and stored product images, allowing the system to identify the most similar product.

## 2. Work Completed

### 2.1. Upgraded Chatbot with Image Recognition Capability

- Image Processing Pipeline: When a user sends an image, the system automatically generates an embedding using a suitable image model (e.g., CLIP or equivalent).

- Similarity Comparison: Used cosine similarity to compare the user-provided image embedding with embeddings of stored product images.

- Product Identification: The product image with the shortest distance (most similar) is selected, and relevant product information is returned to the user.

### 2.2. Handling Three Main Scenarios

- Text-only Messages: The chatbot functions normally by detecting intents and responding based on the text content.

- Image-only Messages: The system analyzes the image, finds the closest matching product, and replies with its name, price, and details if a suitable match is found.

- Messages with Both Text and Image: The system processes both components independently and combines the results to deliver the most accurate and helpful response.

### 2.3. Testing and Optimization

- Tested various cases with real product images and low-quality images to ensure stable performance.
- Tuned the embedding model and similarity threshold to reduce false matches.
- Added exception handling for unclear or unrelated images to ensure smooth operation.

## 3. Technologies & Tools Used

- Programming Language: Python
- Image Embedding Model: CLIP or equivalent
- Similarity Calculation: Cosine similarity
- Frontend/Backend Integration: JavaScript (image upload interface), FastAPI (API handling), MySQL (product database)
- API Support: Gemini API or Grok

## 4. Outcomes

- The chatbot can now understand and process both image and text input, making interactions more flexible and intelligent.
- Image-based product identification works effectively, allowing users to find products without knowing their exact names.

- Handling mixed messages (text + image) brings the chatbot closer to real-world usage scenarios, significantly improving the user experience.