ShopBot Backend Documentation

Overview

The backend of ShopBot is built using Django and integrates with Google Dialogflow to process chatbot queries. This documentation outlines the setup, API endpoints, and authentication mechanisms.

Project Structure

```
ShopBot/
|-- chatbot_backend/  # Main Django backend
|   |-- settings.py  # Configuration settings
|   |-- urls.py  # API routing
|   |-- views.py  # Logic for handling requests
|-- agent.json  # Service account credentials for Dialogflow
|-- requirements.txt  # Required dependencies
```

Installation & Setup

1. Clone the Repository

```
git clone <>
cd ShopBot
```

2. Create and Activate a Virtual Environment

```
python3 -m venv venv
source venv/bin/activate  # On Linux/macOS

### 3. Install Dependencies

```bash
pip install -r requirements.txt
```

4. Set Up Google Dialogflow Authentication

- Download the `agent.json` file from Google Cloud Console.
- Move it to the project directory (`ShopBot/`).
- Configure Django to use this file by adding the following in `settings.py` :

```
import os
from pathlib import Path

BASE_DIR = Path(__file__).resolve().parent.parent
SERVICE_ACCOUNT_FILE = os.path.join(BASE_DIR, "agent.json")
os.environ["GOOGLE_APPLICATION_CREDENTIALS"] = SERVICE_ACCOUNT_FILE
```

API Endpoints

1. Chatbot Query Endpoint

URL: `/api/chatbot/`
Method: `POST`

Request:

```
{
  "query": "What is the price of a Shirt?"
}
```
```

Response:

```
{
  "response": "The price of a shirt is $20."
}
```

2. Product List Endpoint

URL: `/api/products/`
Method: `GET`

Response:

```
[
  { "name": "Shirt", "price": 20 },
  { "name": "Jeans", "price": 40 }
]
```

Troubleshooting

1. "403 IAM permission denied" Error

- Ensure the correct Google Cloud IAM roles are assigned to the service account.
- Check that `agent.json` is properly referenced in `settings.py`.

2. "File not found: agent.json"

- Verify that `agent.json` is in the project root.
- Ensure the environment variable `GOOGLE_APPLICATION_CREDENTIALS` is set.

Running the Server

Start the Django development server:

```
python manage.py runserver
```

Your backend should now be accessible at `http://127.0.0.1:8000/`.
