prompt_template.txt

```
You are an AI Financial Assistant helping users with queries about UPI,
interest rates, loans, digital scams, and budgeting.
Answer the question using the context below:

{context}

User question: {question}
```

app.py

Python

```python
from flask import Flask, request, jsonify
from retriever import query_discovery
from translator import translate_to_english, translate_from_english
from dotenv import load_dotenv
import os
from ibm_watsonx_ai.foundation_models import ModelInference
from ibm_watsonx_ai.auth import IAMTokenManager

app = Flask(__name__)
load_dotenv()

# IBM Watsonx Granite Setup
api_key = os.getenv("IBM_API_KEY")
project_id = os.getenv("DISCOVERY_PROJECT_ID")

token_manager = IAMTokenManager(
    api_key=api_key,
    url="https://iam.cloud.ibm.com/identity/token"
)

model = ModelInference(
    model_id="ibm/granite-13b-chat-v1",
    project_id=project_id,
    url="https://us-south.ml.cloud.ibm.com",
    token_manager=token_manager
)

@app.route('/ask', methods=['POST'])
def ask_question():
    data = request.json
    question = data.get("question", "")
    lang = data.get("language", "en")

    if not question:
        return jsonify({"error": "Question is required"}), 400

    # Translate question to English
    translated_q = translate_to_english(question) if lang != 'en' else
question

    # Retrieve relevant context
    context = query_discovery(translated_q)

    # Format prompt
    with open("prompt_template.txt") as f:
        prompt_template = f.read()
```

```
    final_prompt = prompt_template.format(context=context,
question=translated_q)

    # Generate answer
    response = model.generate(prompt=final_prompt,
decoding_method="greedy", max_new_tokens=200)
    answer = response['results'][0]['generated_text'].strip()

    # Translate answer back if needed
    final_answer = translate_from_english(answer, lang) if lang != 'en'
else answer

    return jsonify({
        "question": question,
        "answer": final_answer,
        "language": lang
    })

if __name__ == '__main__':
    app.run(debug=True)
```

requirements.txt

```
flask
langchain
ibm-watson
googletrans==4.0.0-rc1
python-dotenv
ibm-watsonx-ai
```

translator.py

Python

```
from googletrans import Translator

translator = Translator()

def translate_to_english(text):
    return translator.translate(text, dest='en').text

def translate_from_english(text, dest_lang):
    return translator.translate(text, dest=dest_lang).text
```

retriever.py

Python

```
import os
from dotenv import load_dotenv
from ibm_watson import DiscoveryV2
from ibm_cloud_sdk_core.authenticators import IAMAuthenticator

load_dotenv()

def query_discovery(question):
    api_key = os.getenv("IBM_API_KEY")
    url = os.getenv("DISCOVERY_URL")
    project_id = os.getenv("DISCOVERY_PROJECT_ID")

    authenticator = IAMAuthenticator(api_key)
    discovery = DiscoveryV2(
```

```
        version="2023-11-01",
        authenticator=authenticator
    )
    discovery.set_service_url(url)

    response = discovery.query(
        project_id=project_id,
        natural_language_query=question,
        count=3
    ).get_result()

    # Extract relevant context passages
    context = ""
    for result in response["results"]:
        context += result.get("document_passages",
[{}])[0].get("passage_text", "") + "\n"

    return context.strip()
```

README.md

Markdown

```
#💰 AI Agent for Digital Financial Literacy (Problem Statement No.7)
An AI-powered multilingual assistant built using IBM Cloud Lite, IBM
Watson Discovery, and IBM Granite (Watsonx), designed to educate users
on digital financial tools like UPI, interest rates, personal finance,
digital scams, and more.

✅ Features
🔍 RAG (Retrieval-Augmented Generation) using Watson Discovery

💬 Granite LLM (IBM Watsonx) for accurate and contextual responses

🌐 Multilingual Support using Google Translate

🔐 Built with Flask API and .env secure credentials

🔄 Answers questions like:

“How do I send money using UPI?”
“What is a safe interest rate?”
“How to identify a fake banking app?”
📁 Project Structure
ai-digital-financial-agent/
├── app.py                  # Flask backend for the AI chatbot
├── retriever.py            # Retrieves financial context using Watson
Discovery
├── translator.py           # Language translation for multilingual
support
├── prompt_template.txt     # Template for Granite LLM prompt
├── .env                    # Environment variables (keys & IDs)
├── requirements.txt        # All required dependencies
└── README.md               # You're here!
🛠️ Technologies Used
Technology    Role
IBM Watsonx Granite   Large Language Model (LLM)
IBM Watson Discovery  Knowledge retrieval (RAG)
```

Flask API framework
Googletrans Translate user input/output
Python dotenv Secure environment management

🔑 *Setup Instructions*

1. Clone the repository

```
git clone https://github.com/your-username/ai-digital-financial-agent.git
cd ai-digital-financial-agent
```

2. Install dependencies

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

3. Create .env file

```
IBM_API_KEY=your_ibm_cloud_api_key
DISCOVERY_URL=https://api.us-south.discovery.watson.cloud.ibm.com
DISCOVERY_PROJECT_ID=your_discovery_project_id
```

4. Run the Flask app

```
python app.py
```

🔄 *Sample Request (POST /ask)*

```
{
  "question": "यूपीआई से पैसे कैसे भेजें?",
  "language": "hi"
}
```

Sample Response

```
{
  "question": "यूपीआई से पैसे कैसे भेजें?",
  "answer": "आप UPI ऐप जैसे Google Pay, PhonePe या BHIM का उपयोग करके मोबाइल नंबर या UPI ID के माध्यम से पैसे भेज सकते हैं।",
  "language": "hi"
}
```

⚠️ *Notes*

Ensure you have uploaded valid financial documents (like RBI FAQs, NPCI docs) to your Watson Discovery project.
Granite LLM access requires an IBM Cloud project with Watsonx enabled.
This app is intended for educational and awareness purposes only.

🛣️ *Use Cases*

Digital Financial Literacy in rural areas
Banking fraud awareness
Student & youth financial onboarding
Multilingual chatbot on financial apps

👨‍👩‍👧‍👦 *Contributors*

Kalyan Nagu (Developer)
IBM Cloud Lite APIs
Open-source AI community

.env

```
IBM_API_KEY=cpd-apikey-IBMid-69700118IH
DISCOVERY_URL=https://us-south.ml.cloud.ibm.com/ml/v4/deployments/ba025a97-4473-4521-9cfc-e43af56da3be/ai_service?version=2021-05-01
DISCOVERY_PROJECT_ID=ba025a97-4473-4521-9cfc-e43af56da3be
```