AI/ML Intern Test: Transaction Risk Analysis Project

Project Overview

Build a system that receives transaction data via webhook, uses an LLM to analyse risk patterns, and notifies administrators about suspicious transactions.

Learning Objectives

- Implement an API endpoint to receive and validate JSON data
- Craft effective LLM prompts for specialised tasks
- Apply AI to a real-world financial scenario
- You can Python, Node or a tool like N8N to achieve this task.

Technical Requirements

1. Webhook Service

- Create an endpoint that accepts POST requests with JSON transaction data
- Implement basic authentication for the webhook
- Validate incoming transaction data structure
- Return appropriate HTTP status codes and responses

2. LLM Integration

- Use the provided prompt template to query an LLM (OpenAI, Claude, etc.)
- Ensure the prompt effectively instructs the LLM to analyse transaction risk
- Parse and validate the LLM's response
- Implement error handling for API failures or malformed responses

3. Admin Notification System

- Create an API to notify administrators about high-risk transactions
- Include relevant transaction details and risk analysis in notifications

4. Testing and Documentation

- Create unit tests for critical components
- Document your API endpoints
- Provide examples of successful and failed transactions
- Include instructions for setting up and running the project

Data Structures

Transaction Webhook JSON

json

Unset

```json
{
  "transaction_id": "tx_12345abcde",
  "timestamp": "2025-05-07T14:30:45Z",
  "amount": 129.99,
  "currency": "USD",
  "customer": {
    "id": "cust_98765zyxwv",
    "country": "US",
    "ip_address": "192.168.1.1"
  },
  "payment_method": {
    "type": "credit_card",
    "last_four": "4242",
    "country_of_issue": "CA"
  },
  "merchant": {
    "id": "merch_abcde12345",
    "name": "Example Store",
    "category": "electronics"
  }
}
```

Admin Notification JSON

json

Unset

```
{
  "alert_type": "high_risk_transaction",
  "transaction_id": "tx_12345abcde",
  "risk_score": 0.85,
  "risk_factors": [
    "Customer country (US) differs from card country (CA)",
    "Transaction amount significantly higher than customer
average",
    "Multiple transactions within short timeframe"
  ],
  "transaction_details": {
    // The original transaction JSON
  },
  "llm_analysis": "This transaction shows multiple risk
indicators including cross-border payment method, unusual amount
for this customer, and velocity pattern concerns."
}
```

Test Cases

Your implementation should handle the following scenarios:

1. **Normal Transaction**: A domestic transaction with matching customer and payment method countries

2. **Cross-Border Transaction**: Different customers and payment methods in different countries
3. **High-Value Transaction**: An unusually large transaction amount
4. **High-Risk Country**: A transaction involving a country on the risk list
    a. `const HIGH_RISK_COUNTRIES = ['RU', 'IR', 'KP', 'VE', 'MM'];`
5. **Missing Fields**: A transaction with incomplete data
6. **Invalid Authentication**: A request with missing or invalid authentication

LLM Prompt

Please fine-tune the prompt to use fewer tokens but provide the best output.

````
# Transaction Risk Analysis Prompt

## System Instructions

You are a specialised financial risk analyst. Your task is to evaluate
transaction data and determine a risk score from 0.0 (no risk) to 1.0
(extremely high risk) based on patterns and indicators of potential fraud.
You must also provide clear reasoning for your risk assessment.

## Response Format

Respond in JSON format with the following structure:
\`\`\`json
{
  "risk_score": 0.0-1.0,
  "risk_factors": ["factor1", "factor2"...],
  "reasoning": "A brief explanation of your analysis",
  "recommended_action": "allow|review|block"
}
\`\`\`

## Risk Factors to Consider

1. **Geographic Anomalies**:
   - Transactions where the customer country differs from the payment
method country
````

```
   - Transactions from high-risk countries (consider jurisdiction with weak AML controls)
   - IP address location inconsistent with the customer's country

2. **Transaction Patterns**:
   - Unusual transaction amount for the merchant category
   - Transactions outside normal business hours for the merchant's location
   - Multiple transactions in short succession

3. **Payment Method Indicators**:
   - Payment method type and associated risks
   - New payment methods have recently been added to accounts

4. **Merchant Factors**:
   - Merchant category and typical fraud rates
   - Merchant's history and reputation

## Additional Guidelines

- Assign higher risk scores to combinations of multiple risk factors
- Consider the transaction amount - higher amounts generally warrant more scrutiny
- Account for normal cross-border shopping patterns while flagging unusual combinations
- Provide actionable reasoning that explains why the transaction received its risk score
- Recommend "allow" for scores 0.0-0.3, "review" for scores 0.3-0.7, and "block" for scores 0.7-1.0

## Transaction Data
{{TRANSACTION_JSON}}
  `;
}
```