Credit Fraud Detection Using the Hidden Markov Model

Team Members:

Sergio Gabriel Jiawei Kun ; s_kun@mail.fhsu.edu

Lee Joseph Judkins ; ljjudkins@mail.fhsu.edu

Project Github: https://github.com/kunsergio117/CreditFraudDetectionHMM.git

Instructor: Prof. Hieu Vu

Revision History:

Part 2	23/09/2024

Table of Contents

Section 3: Use Cases

Use Cases:

A: Stakeholders

1. **Users:** The users would be the credit card holders who benefit from the protection against fraud.
2. **Fraud Analysts:** The individuals who would be responsible for reviewing flagged transactions and investigating potential fraud.
3. **Financial Institutions:** Banks, credit card companies, and other online retailers who would be responsible for managing and securing transactions.
4. **System Administrators:** Those people who are responsible for maintaining and updating system infrastructure.
5. **Developers:** The technical team responsible for development, deployment, and maintenance of the HMM-based fraud detection system.
6. **Regulatory Bodies:** Government or financial regulatory authorities ensuring compliance with fraud prevention standards.

B: Actors and Goals

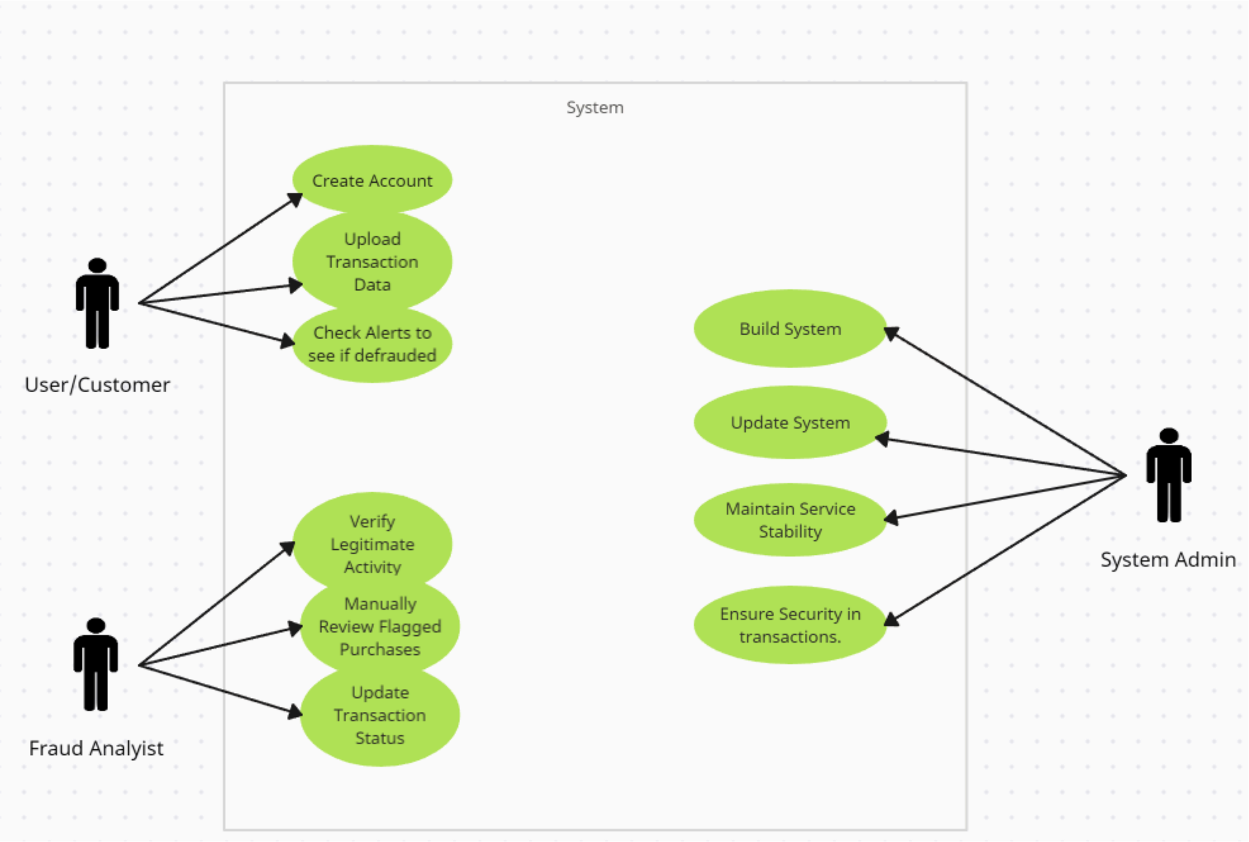
1. **Initiating Actors:**
    a. **Credit Card Holder:** Their goal is to make secure transactions without being blocked or flagged incorrectly.
    b. **Fraud Analyst:** Their goal is to quickly and efficiently review flagged transactions. They will then confirm if it is fraud, and take action.
    c. **System Administrator:** Their goal is to maintain the system, ensuring continuous functionality, and ensure regular maintenance.
2. **Participating Actors:**
    a. **Transaction System:** This processes transactions and interacts with the HMM for fraud detection.
    b. **HMM Model:** Identifies suspicious transactions based on behavioral patterns.

C: Use Cases

A. **Casual Description**
    1. **Upload Transaction Data**
        i. Description: The user uploads a CSV file containing historical transaction data.
        ii. Requirements: Addressed in REQ1, REQ5
    2. **Monitor Transactions for Fraud**
        i. Description: HMM based system analyzes incoming transactions in real-time, identifying anomalies for fraudulent transactions.
        ii. Requirements: Addressed in REQ2, REQ3, REQ4,
    3. **Manual Transaction Review**
        i. Description: A fraud analyst manually reviews flagged transactions to determine their legitimacy.
        ii. Requirements: Addressed in REQ6, REQ8, REQ13

4. **Generate Fraud Reports**
   i. Description: The system generates detailed reports of fraudulent activity, including detection accuracy and false positives.
   ii. Requirements: Addressed REQ5
5. **Real Time Alerts**
   i. Description: The system sends alerts to users and fraud analysts when suspicious transactions are detected.
   ii. Requirements: Addressed REQ3, REQ11, REQ12

B. Use Case Diagram


System
Create Account
Upload Transaction Data
Check Alerts to see if defrauded
User/Customer
Verify Legitimate Activity
Manually Review Flagged Purchases
Update Transaction Status
Fraud Analyist
Build System
Update System
Maintain Service Stability
Ensure Security in transactions.
System Admin

C. Traceability Matrix

Requirement ID	Use Case Name	Priority 1-5
REQ 1	**Upload Transaction Data**	**5**
REQ 2	**Monitor Transactions for Fraud**	**5**
REQ 3	**Monitor Transactions for Fraud/ Real Time Alerts**	**4**
REQ 4	**Monitor Transactions for Fraud**	**5**
REQ 5	**Upload Transaction Data/ Generate Fraud Reports**	**3**
REQ 6	**Manual Transaction Review**	**4**
REQ 8	**Manual Transaction Review**	**2**
REQ 11	**Real Time Alerts**	**4**
REQ 12	**Real Time Alerts**	**2**
REQ 13	**Manual Transaction Review**	**4**

D: Fully Dressed Descriptions

1. **Use Case: Upload Transaction Data**
    - **Primary Actor:** Credit Cardholder, Fraud Analyst
    - **Goal:** Upload CSV file containing transaction data for analysis.
    - **Preconditions:** Users must be authenticated and have a valid CSV file with transaction data.
    - **Main Flow of Events:**
        1. The user logs in and navigates to the upload page.
        2. The user selects the CSV file containing transaction data.
        3. The system validates the file format and structure.
        4. If the file is valid, it is uploaded, and the data is processed.
        5. If invalid, the system provides an error message indicating the problem.
    - **Postconditions:** Transaction data is stored in the system for analysis.
    - **Exceptions:** If the file is incorrectly formatted, the system rejects the upload and provides instructions for correction.

2. Use Case: Real-Time Alerts

- **Primary Actor:** Credit Cardholder, Fraud Analyst
- **Goal:** Notify users of suspicious activity when a transaction is flagged as potentially fraudulent.

- **Preconditions:** The system must be monitoring incoming transactions in real-time.
- **Main Flow of Events:**
  1. A transaction occurs and is processed by the system.
  2. The HMM flags the transaction as anomalous.
  3. The system sends an alert via email or SMS to the user and fraud analyst.
  4. The alert contains transaction details such as the amount, merchant, date, and reason for flagging.
  5. The user or analyst reviews the alert and takes appropriate action (e.g., freeze the account, investigate further).
- **Postconditions:** Users and analysts are informed of potentially fraudulent activity and can take action.
- **Exceptions:** If the system fails to send an alert (e.g., due to network failure), the transaction remains flagged for manual review.

3. Use Case: Generate Fraud Reports

- **Primary Actor:** Fraud Analyst
- **Goal:** Create downloadable reports that summarize system performance and fraudulent activity.
- **Preconditions:** The system must have accumulated transaction and fraud detection data.
- **Main Flow of Events:**
  1. The analyst navigates to the report generation page.
  2. The analyst selects parameters for the report (e.g., date range, transaction type).
  3. The system retrieves relevant transaction data.
  4. The system generates a report summarizing fraud detection statistics (e.g., false positives, true positives, accuracy).
  5. The analyst downloads the report in PDF or CSV format.
- **Postconditions:** A report is generated and available for download.
- **Exceptions:** If no data is available for the selected parameters, the system informs the analyst and allows them to adjust the criteria.

Use Case: Manual Transaction Review

- **Primary Actor:** Fraud Analyst
- **Goal:** Review and validate transactions flagged as suspicious.
- **Preconditions:** The system must have flagged transactions based on the HMM model.
- **Main Flow of Events:**
  1. The analyst accesses the list of flagged transactions.
  2. The analyst filters or sorts transactions based on criteria such as date, amount, or merchant.
  3. The analyst selects a transaction to review.
  4. The system displays transaction details and metrics that led to the flagging.
  5. The analyst makes a decision to either mark the transaction as legitimate or confirm it as fraudulent.
  6. The system updates the transaction’s status accordingly.

- **Postconditions:** The flagged transaction is resolved, and the decision is logged in the system.
- **Exceptions:** If the analyst is unable to make a decision, the transaction remains flagged for further review.

D: System Sequence Diagrams

Section 4: User Interface Specification

Preliminary Design

Project Management

Both team members will be actively engaged in the development process, utilizing the GitHub repository to track progress and contributions clearly.

Responsibilities will be delegated based on each member's strengths and expertise, with a shared accountability structure ensuring that all aspects of the project are covered. This is because our group consists of only a pair and thus realistically we will both need input and validation from one another in all development areas.

Below is our projected milestones, with week 1 representing the week beginning in 23 Sept 2024.

Week	**Task Description**	**Responsible Team Members**
Week 1	Project kick-off meeting, define scope and objectives	Sergio Kun, Joseph Judkins
Week 2	Research HMM algorithms and relevant literature	Sergio Kun
	Familiarization with data sources and datasets	Joseph Judkins
Week 3	Develop initial design and architecture of the application	Sergio Kun, Joseph Judkins
Week 4	Implement CSV transaction data upload feature	Sergio Kun
	Initial development of the user authentication process	Joseph Judkins

Week	Task Description	Responsible Team Members
Week 5	Develop the manual checking function for transaction validity	Sergio Kun
Week 6	Implement the alerting function for suspicious transactions	Joseph Judkins
Week 7	Integrate the simulated fraudulent transaction feature	Sergio Kun
Week 8	Testing functionality and debugging	Sergio Kun, Joseph Judkins
Week 9	Develop and integrate reporting tools for transaction summaries	Joseph Judkins
	User interface refinement and user experience enhancements	Joseph Judkins
Week 10	Conduct user testing and gather feedback	Sergio Kun, Joseph Judkins
Week 11	Finalize features based on user feedback	Sergio Kun, Joseph Judkins
Week 12	Prepare project presentation and documentation	Sergio Kun, Joseph Judkins
Week 13	Project review and adjustments based on tutor feedback	Sergio Kun, Joseph Judkins
Week 14	Final project deployment and presentation to the class	Sergio Kun, Joseph Judkins

References