**Project Proposal: Online Daily Prize Draw System with Fraud Detection**

**Project Domain / Category**

Data Science / Machine Learning

**Abstract / Introduction:**

We propose the development of an online platform that facilitates a daily prize draw for a committee of 10,000 participants, each contributing 100 PKR per day. The system will announce a daily winner who will receive a cash prize of 10,00,000 PKR. The platform will handle user management, prize distribution, and ensure transparency in the process. In addition, we will integrate a fraud detection module that will be trained by users on publicly available datasets, such as Credit Card Fraud Detection. The goal is to create an engaging, fair, and secure system for participants while also contributing to participant welfare.

**Functional Requirements:**

1. **User Management:**

Users will be able to create accounts with personal details, including name, address, email, and phone number.

Admin functionalities will include creating, editing, and deleting user accounts.

1. **Deposit and Participation:**

Users will deposit 100 PKR per day, and the system will maintain their contribution records.

The system will automatically enter them into the daily prize draw.

1. **Daily Winner Selection:**

The system will randomly select a winner each day from the pool of participants.

The winner will receive 80% of the daily collected amount, i.e., 10,00,000 PKR.

1. **Prize Distribution:**

The system will calculate and deduct a 5% service charge from the daily prize amount.

The remaining 15% will be saved for the participant's welfare fund.

1. **Monthly Returns:**

In financial terms, a monthly return refers to the profit or gain an investor earns on an investment within a month. It represents the percentage increase or decrease in the value of an investment over the course of a single month. The formula to calculate the monthly return is:

Monthly Return=(End Value−Start ValueStart Value)×100%Monthly Return=(Start ValueEnd Value−Start Value​)×100%

Where:

* End Value is the value of the investment at the end of the month.
* Start Value is the initial value of the investment at the beginning of the month.

Monthly returns are a measure of investment performance over a short period. It helps investors assess the profitability of their investments within a specific month, allowing them to track and evaluate their portfolio's performance over time.

In the context of the provided scenario with a system providing a monthly return of 1,000 PKR along with the 10 PKR/day deposit for a specified duration, it implies that the system guarantees a fixed monthly return of 1,000 PKR to the winner in addition to the daily deposit of 10 PKR made by the user for a certain period. This could be a form of incentive or bonus to the winner based on their participation in the system's daily deposit and prize draw mechanism.

The system will provide the winner with a monthly return of 1,000 PKR along with the 10 PKR/day deposit for a specified duration.

1. **Transparency and Audit:**

The system will ensure transparency in winner selection and fund management.

Audit logs and reports will be available for scrutiny.

1. **Fraud Detection Module:**

Users will have the opportunity to train the fraud detection module by providing it with publicly available datasets, such as bank transaction data.

The module will use machine learning to detect potential fraud within the prize draw system.

Users' input and training data will continuously improve the module's accuracy.

**7.1) Fraud Detection Module with "Credit Card Fraud Detection" Dataset:**

For our fraud detection module, we will utilize publicly available datasets to train the system effectively. Specifically, we will employ the "Credit Card Fraud Detection" dataset, which can be accessed on Kaggle using the following link:

**Dataset Kaggle**: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

**7.2) Deep Overview of the Module:**

Our fraud detection module plays a critical role in maintaining the integrity and security of the daily prize draw system. Users' contributions and participation generate valuable data that our machine learning model processes to identify potential fraudulent activities. This module operates as follows:

1. **Data Acquisition:**

We will fetch and preprocess data from the "Credit Card Fraud Detection" dataset, which contains both legitimate and fraudulent credit card transactions. The dataset includes features related to transaction amounts, timestamps, and various anonymized attributes.

1. **Model Training:**

Machine learning algorithms, such as anomaly detection models or supervised classifiers, will be used to train the fraud detection model on this dataset. The model will learn to distinguish between legitimate and fraudulent transactions.

1. **Continuous Learning:**

The fraud detection module will continuously evolve and improve as more data is fed into it. User input and ongoing training data will be integrated to enhance the model's accuracy in identifying potential fraud.

1. **Real-time Monitoring:**

The model will monitor daily prize draw transactions in real-time, evaluating each transaction against the learned patterns of fraudulent behavior. Any anomalies or deviations from the learned behavior will be flagged for further investigation.

1. **Alerting and Reporting:**

When potential fraudulent activity is detected, the system will generate alerts for administrators or moderators to investigate further. Users will be encouraged to report any suspicious activity as well.

**Conclusion:**

The proposed online daily prize draw system with an integrated user-trained fraud detection module aims to create an exciting, transparent, and secure platform for participants while contributing to their welfare. We believe that this project will provide added value and security, and we look forward to your consideration and collaboration.

**Tools:**

**Python**: For machine learning, web scraping, and back-end development.

**JavaScript**: For front-end interactivity and real-time updates.

**Frameworks and Libraries:**

**Django**: A Python web framework for building the back end of the web application.

**HTML**: For structuring web content.

**CSS**: For styling and layout.

**Bootstrap**: A CSS framework for responsive web design.

**Machine Learning:**

**scikit-learn**: A Python library for machine learning and data analysis.

**Supervisor**

Name: Muhammad Kaleemullah

Email ID: m.kaleem@vu.edu.pk

Skype ID: kaleembhatti561

Certainly! To design a database schema for the proposed online platform, here's an outline that aligns with the functional requirements and fraud detection module:

### Tables:

1. \*\*Users\*\*

- Columns:

- UserID (Primary Key)

- Name

- Address

- Email

- Phone Number

- Account\_Created\_Date

- Other relevant user details

2. \*\*Deposits\*\*

- Columns:

- DepositID (Primary Key)

- UserID (Foreign Key referencing Users.UserID)

- Deposit\_Date

- Amount (100 PKR)

- Status (Successful/Failed/Pending)

3. \*\*Daily\_Prize\_Draw\*\*

- Columns:

- DrawID (Primary Key)

- Date

- Winner\_UserID (Foreign Key referencing Users.UserID)

- Prize\_Amount (80% of the daily collected amount)

- Service\_Charge (5% of the daily prize amount)

- Welfare\_Fund\_Amount (15% of the daily prize amount)

4. \*\*Monthly\_Returns\*\*

- Columns:

- ReturnID (Primary Key)

- Winner\_UserID (Foreign Key referencing Users.UserID)

- Return\_Date

- Return\_Amount (1,000 PKR)

5. \*\*Fraud\_Detection\_Training\_Data\*\*

- Columns:

- TrainingDataID (Primary Key)

- UserID (Foreign Key referencing Users.UserID)

- Training\_Data (Fraud training data provided by users)

- Date\_Contributed

- Source (e.g., Credit Card Fraud Detection dataset)

6. \*\*Audit\_Logs\*\*

- Columns:

- LogID (Primary Key)

- Action\_Performed

- Performed\_By (Admin/User)

- Timestamp

- Description

### Relationships:

- Each user can have multiple deposits (one-to-many relationship between Users and Deposits).

- Each user can be a winner multiple times (one-to-many relationship between Users and Daily\_Prize\_Draw).

- Each winner can receive multiple monthly returns (one-to-many relationship between Users and Monthly\_Returns).

- Each user can contribute multiple sets of fraud training data (one-to-many relationship between Users and Fraud\_Detection\_Training\_Data).

- Audit logs record actions performed by users/admins for transparency.

### Functionality:

1. \*\*User Management:\*\* The 'Users' table stores participant details, including personal information and account creation date.

2. \*\*Deposit and Participation:\*\* The 'Deposits' table tracks daily deposits made by users for participation in the prize draw.

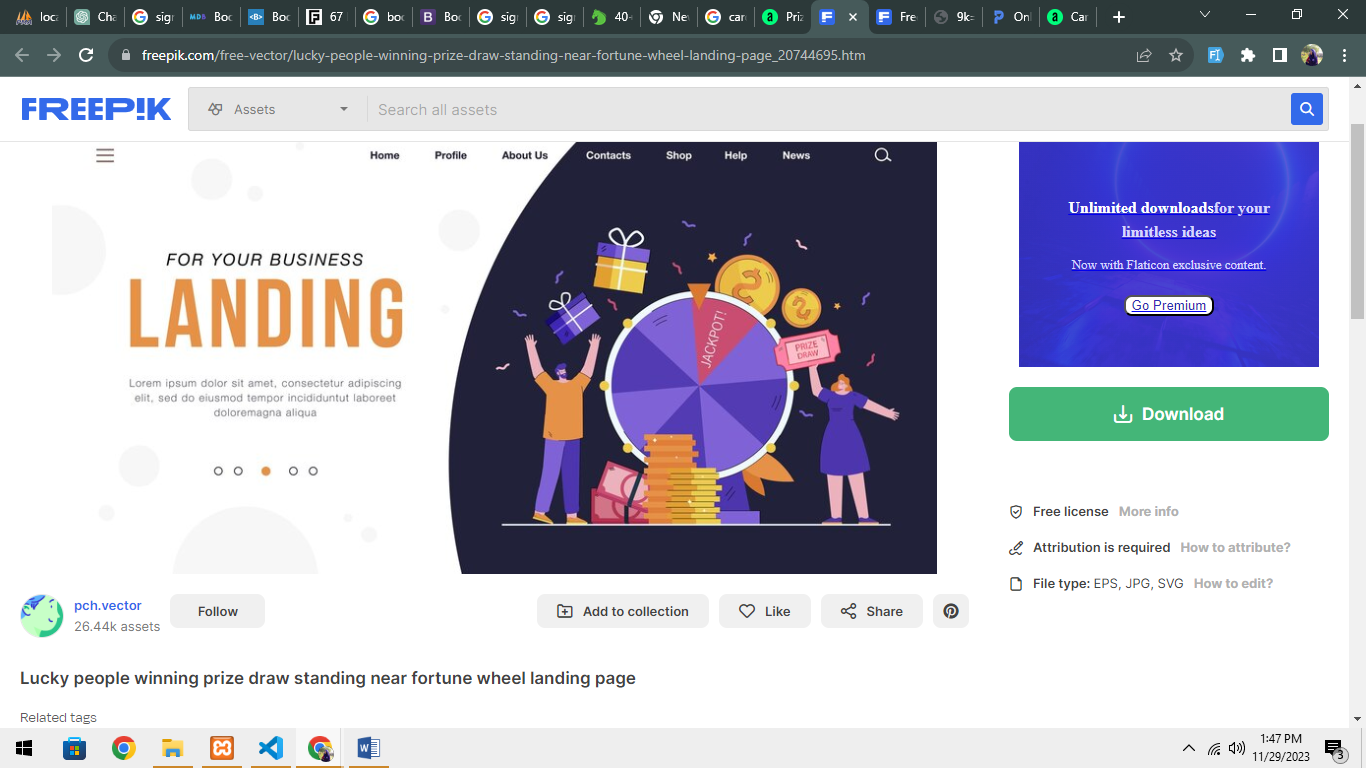
3. \*\*Daily Winner Selection:\*\* The 'Daily\_Prize\_Draw' table records daily winners and manages the distribution of prize amounts, service charges, and welfare fund allocation.

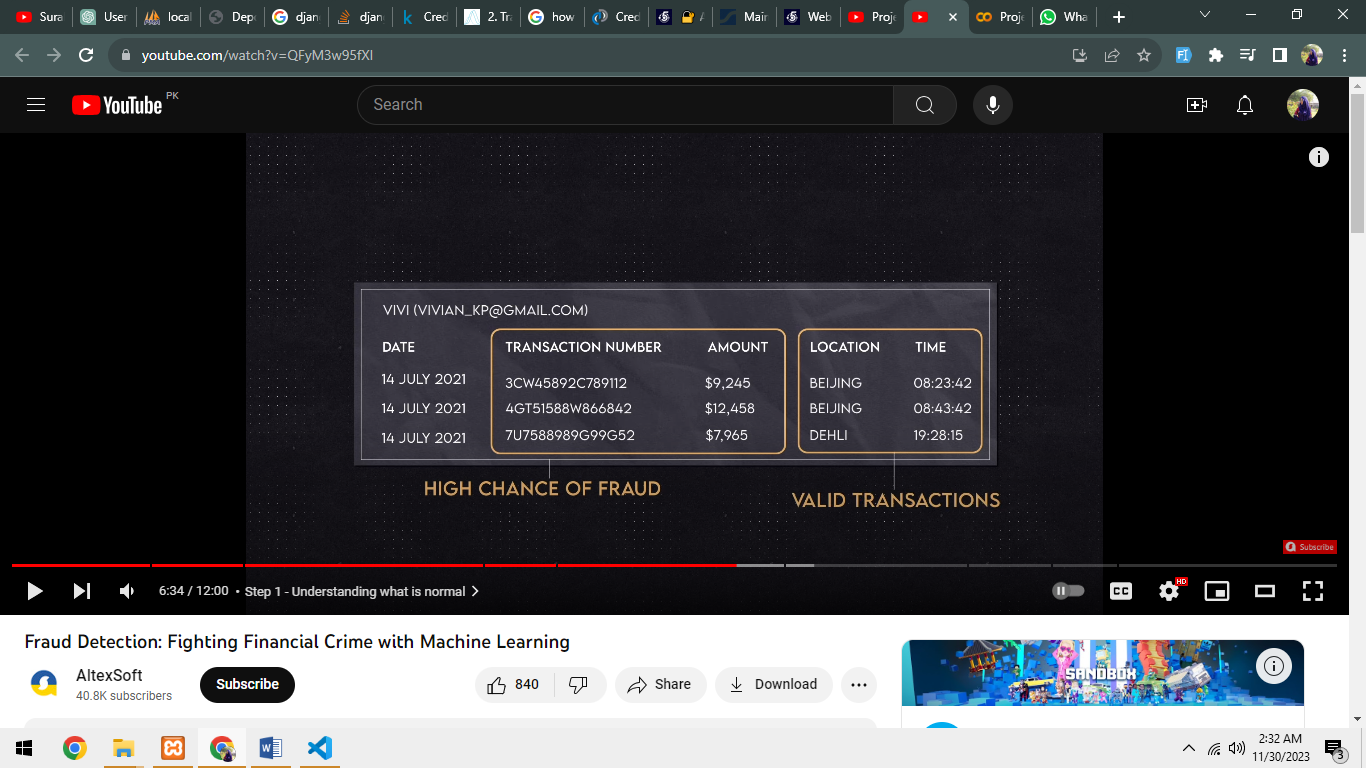
4. \*\*Monthly Returns:\*\* The 'Monthly\_Returns' table maintains records of monthly returns given to the winners.

5. \*\*Fraud Detection Training:\*\* The 'Fraud\_Detection\_Training\_Data' table collects data contributed by users for training the fraud detection module.

6. \*\*Audit Logs:\*\* The 'Audit\_Logs' table keeps a log of system actions for transparency and auditing purposes.

This schema provides the structure to manage users, their deposits, prize distribution, monthly returns, fraud detection training data, and audit logs, aligning with the outlined functional requirements and the fraud detection module's integration with publicly available datasets. It can be further refined or expanded based on specific software implementation requirements and additional functionalities.





Important LINKS:

<https://www.kaggle.com/code/guilherme26/diving-into-classification-metrics/notebook>

<https://github.com/GitTeaching/Django_Credit_Card_Fraud>

<https://www.youtube.com/watch?v=JnlM4yLFNuo>

In the scenario of the online daily prize draw system with fraud detection described in the proposal, potential fraudulent activities might occur at several stages despite the implementation of a fraud detection module. Here are some points where fraud could potentially occur:

1. **User Account Creation:** Fraudulent individuals might create multiple fake accounts using different identities to increase their chances of winning. This could exploit the system by having more entries in the daily draw.
2. **Deposit and Participation:** Individuals might find ways to manipulate the system to deposit funds without actually participating, potentially inflating the daily collected amount.
3. **Winner Selection:** Even with a random selection process, there might be attempts to manipulate the winner selection algorithm or find vulnerabilities in the randomization process to unfairly increase someone's chance of winning.
4. **Prize Distribution:** There's a possibility of manipulation in the calculation of service charges or mismanagement of funds allocated for the participant's welfare fund.
5. **Monthly Returns:** False claims or exploitation of the monthly return system by claiming returns for non-existent participants or through fraudulent means.
6. **Transparency and Audit:** Despite claims of transparency, there might be vulnerabilities in the audit logs or reports that could be exploited to cover up fraudulent activities.
7. **Fraud Detection Module:** While the fraud detection module is meant to prevent fraudulent activities, it might not be foolproof. Individuals might attempt to manipulate or deceive the fraud detection algorithms to evade detection.
8. **Data Acquisition and Training:** The data used to train the fraud detection model might not cover all potential fraudulent activities specific to the prize draw system, allowing certain types of fraud to go undetected.

To mitigate these potential fraud scenarios, continuous monitoring, regular audits, security protocols, and improvements in the fraud detection model's accuracy would be necessary. Additionally, robust user authentication, encryption of sensitive data, and strict compliance checks could help enhance the system's security and minimize fraudulent activities. Regular updates and improvements in the fraud detection model based on new patterns of potential fraud are also essential to keep the system resilient against evolving fraudulent tactics.

User

**You**

how this colus hapen "Deposit and Participation: Individuals might find ways to manipulate the system to deposit funds without actually participating, potentially inflating the daily collected amount."

ChatGPT

**ChatGPT**

The scenario mentioned about manipulating the deposit and participation process involves potential vulnerabilities that could allow individuals to deposit funds without actually participating, thus inflating the daily collected amount. Here are a few ways this could happen:

1. **Exploiting Payment System Glitches:** Sometimes, payment systems may have vulnerabilities or glitches that could allow individuals to deposit funds without the proper deduction from their account or without authentic participation. This could lead to an inflated amount in the daily collection, making it seem like more people are participating than actually are.
2. **Automated Scripting or Bots:** Individuals might develop automated scripts or bots that simulate multiple transactions, tricking the system into registering these as legitimate participations. These automated processes could deposit funds without any real human interaction or valid intent to participate.
3. **Identity Theft or Account Manipulation:** Fraudulent individuals might engage in identity theft or manipulate user accounts to deposit funds from other users' accounts without their knowledge or consent. This could be through various means like phishing attacks, stealing login credentials, or exploiting system vulnerabilities.
4. **Manipulating Data Inputs:** There could be potential vulnerabilities in the system's input validation mechanisms. Hackers or individuals with malicious intent might exploit these vulnerabilities to submit false or manipulated data inputs that appear to be valid participations, thereby increasing the daily collected amount artificially.

To prevent these potential issues, implementing robust authentication and authorization mechanisms, regularly monitoring and auditing transactions, employing anti-fraud algorithms, ensuring secure payment gateways, and conducting thorough validation checks on deposited funds and user interactions can help prevent and detect such fraudulent activities. Regularly updating and patching any system vulnerabilities is crucial to maintain the integrity of the deposit and participation process.

123Areesga

Shabnam