



# GENERAL INTRODUCTION

- This project, titled **Fraud Detection System**, was developed as part of the course **ICT2140: Introduction to Software Engineering**. It was undertaken by **Group 9**, consisting of dedicated team members with clearly defined roles to ensure effective collaboration and successful project delivery. The group leader, **Njume Brian Epile**, served as the Scrum Master, coordinating tasks and ensuring smooth workflow. Other members, including **Aboubakar Garba Mamoudou**, **Mohammed Haibai Moustapha Aboubakar**, **Moustapha Sidi Mohamed**, and **Ebode Kesha Farella**, each contributed to specific development tasks, ranging from data collection and preprocessing to model training, evaluation, and reporting. Together, the team designed and implemented a machine learning-based system capable of detecting fraudulent financial transactions, demonstrating the application of both software engineering principles and advanced data science techniques.



# INTRODUCTION

- The rapid expansion of digital financial services has increased both the convenience of transactions and the risk of fraud. Fraudulent activities such as identity theft, credit card misuse, and unauthorized transfers now occur at unprecedented levels. Traditional detection methods, which rely heavily on rigid rule-based systems, are often unable to keep pace with evolving fraud techniques. This project emphasizes the use of **machine learning**, which offers adaptive, data-driven, and real-time solutions. By analyzing large-scale transaction datasets, machine learning models can detect unusual patterns and improve the efficiency and accuracy of fraud detection systems.




# Aim and Objectives

- The main aim of this project is to **design and implement a machine learning-based fraud detection system** that can analyze financial transactions and distinguish between legitimate and fraudulent activities. To achieve this aim, the project set specific objectives: conducting **exploratory data analysis (EDA)** to uncover patterns and anomalies in transaction data, preprocessing and transforming raw data into a machine-readable format, developing and evaluating machine learning models for classification, and ensuring that the system is scalable and secure. Ultimately, the project seeks to provide a reliable framework that minimizes financial risks and enhances trust in digital payment systems.



# Problem Statement

- The growing reliance on digital transactions has resulted in increased opportunities for fraudsters to exploit financial systems. Traditional rule-based detection methods, while useful, are slow to adapt to new fraud patterns and fail to manage the vast and complex datasets involved in modern financial operations. This leads to challenges such as high false negatives, which allow fraudulent activities to go undetected, and high false positives, which disrupt legitimate transactions. There is therefore a pressing need for advanced solutions that can automatically analyze data in real time, adapt to new fraud schemes, and provide accurate detection with minimal human intervention. This project addresses that gap by leveraging machine learning techniques to build a more efficient fraud detection system.
- 



# Literature Review



- Fraud detection research highlights the importance of machine learning algorithms in addressing fraud-related challenges. Models such as **Logistic Regression, Decision Trees, Random Forests, and Gradient Boosting** have been widely used to identify patterns within large financial datasets. Recent studies also emphasize the use of **deep learning and neural networks** for uncovering complex fraud patterns. One recurring challenge is the issue of **imbalanced datasets**, since fraudulent cases are typically far fewer than legitimate ones. Regarding software engineering methodologies, various models have been proposed: the **Waterfall model**, which is structured but inflexible; the **Agile model**, which adapts well to change but may lack formal validation; and the **V-Model**, which provides parallel testing and development. For this project, the **V-Model methodology** was chosen, as it ensures systematic validation and verification at every stage, which is crucial for fraud detection systems that demand high reliability.





# Research Methodology



- ▶ The research methodology is rooted in a **quantitative and experimental design**, ensuring both technical robustness and practical relevance. Data was collected from two main sources: **synthetic datasets**, which simulate realistic transactions, and **publicly available datasets** such as the Kaggle Credit Card Fraud dataset. The methodology followed several phases, starting with **Exploratory Data Analysis (EDA)** to understand transaction behavior, followed by **feature engineering** to create derived attributes such as transaction frequency and account balance shifts. In model development, both **supervised learning methods** (e.g., Logistic Regression, Random Forests) and **unsupervised anomaly detection techniques** (e.g., Isolation Forest, Autoencoders) were applied. Model performance was evaluated using metrics such as **precision, recall, F1-score, and ROC-AUC**, with an emphasis on minimizing false negatives, since missed fraud cases pose the greatest risks.



# System Requirements



- The system requirements were divided into **functional** and **non-functional categories**. Functional requirements included the ability to input and preprocess transaction data, train and evaluate machine learning models, detect fraud in real time, and generate reports for decision-makers. Non-functional requirements focused on ensuring high performance, accuracy, scalability, and usability. In addition, the system was designed with strong **security measures**, such as encryption and access control, to protect sensitive financial data. These requirements ensure that the fraud detection system is not only effective but also user-friendly, reliable, and compliant with data protection standards.



# System Design



- The system follows a **layered architecture** to promote modularity and scalability. At the base is the **Data Layer**, which handles input sources, preprocessing, and data storage. The **Processing Layer** includes the machine learning core, consisting of model training, evaluation, and a repository for storing trained models. The **Application Layer** implements fraud detection, supported by a continuous learning module that retrain models with new data. The **Presentation Layer** provides dashboards, visualization tools, and reporting modules for administrators. Finally, the **Security Layer** ensures data protection, access control, and audit logging. This layered design supports flexibility, reliability, and continuous improvement.





# Team Organization and Workflow

- The project team adopted the **V-Model methodology**, ensuring parallel development and testing activities. Team roles were clearly defined, beginning with **requirement analysis**, where the problem was scoped and essential system features identified. Data scientists managed **data collection and preprocessing**, while analysts conducted **Exploratory Data Analysis (EDA)** to uncover fraud patterns. Model developers trained and tested machine learning algorithms, while the entire team collaborated in validation, documentation, and reporting. The workflow followed seven main steps: **data collection, preprocessing, analysis, train-test split, model training, evaluation, and deployment**. This structured organization allowed the team to work systematically and efficiently.



# Proposed Algorithm

- The proposed algorithm for fraud detection follows a structured sequence. First, the **transaction dataset** is loaded, containing attributes such as transaction type, amount, and balances. Next, **data preprocessing** is carried out, including handling missing values, encoding categorical features, and balancing the dataset with techniques like SMOTE. Then, **feature engineering** creates additional variables such as transaction frequency and balance differences. The dataset is then split into training and testing subsets, after which machine learning models—such as **Logistic Regression, Random Forest, and XGBoost**—are trained. Performance is evaluated using **precision, recall, F1-score, and ROC-AUC**, and the best model is selected for deployment. In operation, each new transaction is classified as either legitimate or fraudulent.



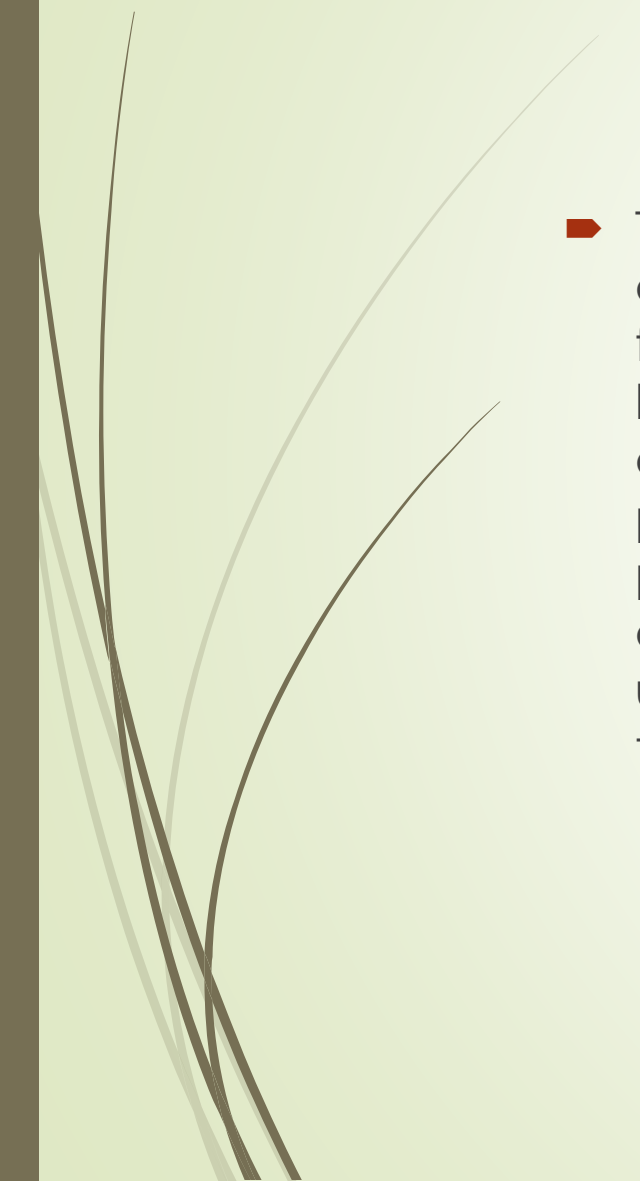
# Technologies Used



- The system was implemented primarily in **Python**, which offers a rich ecosystem for data science and machine learning. Key libraries included **Pandas and NumPy** for data preprocessing, **Scikit-learn and TensorFlow** for model training and evaluation, and **Matplotlib and Seaborn** for visualization. For deployment, the team used **Streamlit**, which provides a user-friendly interface for real-time fraud prediction. Version control was managed through **Git and GitHub**, ensuring collaboration and reproducibility.



# Results and Discussion

- The project successfully delivered a fraud detection system capable of analyzing financial transactions and classifying them as either legitimate or fraudulent. The system achieved strong performance on test datasets, with high levels of precision and recall. It was deployed through a Streamlit dashboard, which allows users to input transaction details and receive predictions in real time. However, challenges were encountered, particularly with **imbalanced datasets**, which made training more difficult, and with integration issues when linking machine learning models to the user interface. Despite these challenges, the system demonstrated both technical feasibility and practical relevance.
- 



# Conclusion and Future Work

- In conclusion, this project demonstrated how **machine learning can be effectively applied to financial fraud detection**. The developed system provides a reliable framework for analyzing transactions, detecting anomalies, and generating fraud alerts in real time. While the system is functional and effective, further improvements are possible. Future work may include incorporating **deep learning models** for more complex fraud patterns, integrating **real-time data streaming** for large-scale deployment, and adopting **Explainable AI (XAI)** techniques to enhance transparency and trust. These enhancements would make the system even more robust and suitable for use in real-world financial institutions.