MIS 6060 | Enhancing Credit Risk Assessment |

Vishaka Sharma, Vyshali Poola, Marko Khotynetskyi



# Final Project Report

## ***1. Organization description, problem statement, and approach used***

Our Credit Union offers credit cards and financial services, focusing on transaction processing, account management, and credit risk evaluation to support financial stability. As digital banking grows, the organization faces challenges like fraud detection and risk assessment. To stay ahead, it needs smart, data-driven solutions that can prevent fraud, accurately evaluate credit risk, and support better decision-making.

### ***1.1 Problem Statement***

USA National Bank, a leading credit card issuer, has been facing growing challenges in managing credit risk and fraud due to outdated analytical systems. Our data was scattered across multiple platforms, making it difficult to get a complete picture of each customer. As a result, some high-risk applicants were approved without catching key red flags, such as unusual transactions or inconsistent credit scores. This led to poor risk decisions and significant financial losses, highlighting the urgent need for a more innovative and unified data solution.

### ***1.2 Approach Used***

1. **Data Generation (Python):** We used Python tools to generate synthetic data that realistically simulates credit card transactions, customer profiles, and fraud patterns. This allowed us to build a robust dataset that effectively captures real-world risk scenarios and supports practical analysis.
2. **ETL Pipeline (SSIS):** We used SQL Server Integration Services (SSIS) to extract, clean, transform, and load the synthetic data into SQL Server. Multiple ETL packages were built to handle data from various sources, including transactions, customer profiles, merchants, chargebacks, and fraud rules.
3. **Data Warehousing (SSAS Cubes):** After loading the data, we designed a star schema data warehouse using SQL Server Analysis Services (SSAS). OLAP cubes were then created to enable fast, multidimensional analysis of customer behavior, fraud patterns, and transaction categories.
4. **Machine Learning Modeling:** Leveraging the processed data, we trained machine learning models like Logistic Regression, Random Forest, and XGBoost. These models analyzed key indicators such as credit utilization, transaction frequency, and online activity to flag high-risk customers and detect fraudulent behavior.
5. **BI Dashboards (Power BI):** As the final step, we built interactive dashboards using Power BI. These visualizations highlighted fraud trends, high-risk customer segments, and credit risk scores, enabling better decision-making and delivering actionable insights to stakeholders.

## ***2. Types and Sources of Data***

* **Customer Data**: Credit score, income level, location.
* **CreditCards Data**: Card Type, Limit, Expiry.
* **Transaction Data**: Amount, time, fraud status.
* **Merchant Data**: Category, location.
* **Chargeback Data**: Dispute Reason, resolution status.
* **Branch Data**: City, state, zip.
* **Fraud Rules**: Rule descriptions and flags.

## ***3. Managerial Decision Problems & DW/BI Role***

In today’s banking environment, preventing fraud and identifying high-risk customers are crucial for maintaining financial integrity and customer trust. Our project focuses on addressing these challenges through the strategic implementation of Data Warehousing and Business Intelligence (DW/BI) systems designed to enhance fraud detection and credit risk assessment.

### ***3.1 Key Questions***

1. Who are the high-risk customers?

We aimed to segment customers based on behavioral indicators, such as credit utilization, transaction patterns, and history of fraud involvement, to identify those most likely to pose a risk.

1. What are the most fraud-prone transaction types?

By analyzing different transaction types (e.g., online purchases, ATM withdrawals, and wire transfers), we identified patterns and pinpointed the types most vulnerable to fraud, enabling focused surveillance and intervention.

1. How can credit score segmentation help detect fraud?

Customers with lower credit scores were more frequently involved in fraudulent activity. Segmenting based on credit scores enhanced the accuracy of our predictive models and contributed to more reliable fraud scoring.

1. Can we detect fraud faster to avoid losses?

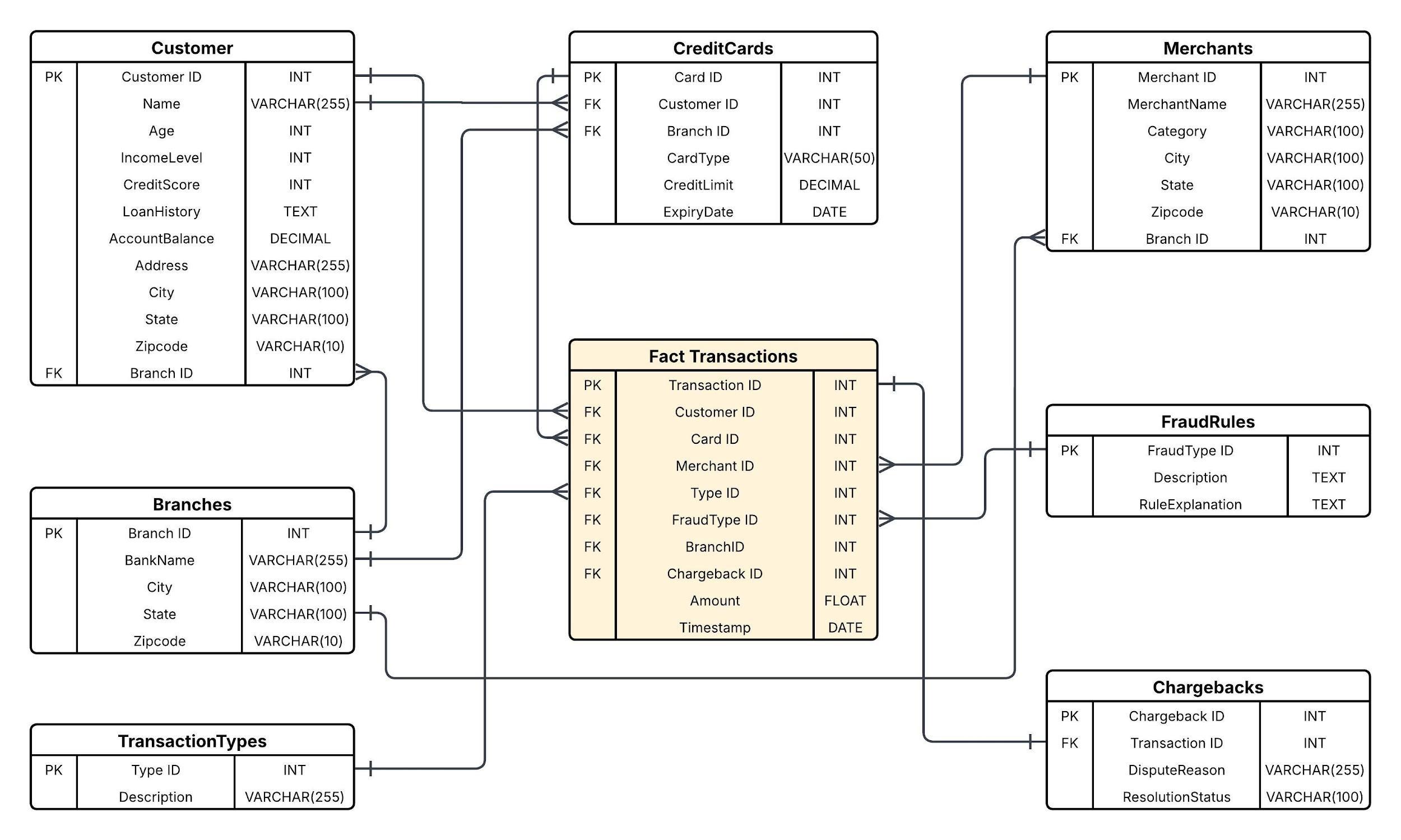
Early detection is critical. This project focused on speeding up detection timelines by utilizing real-time data pipelines and dashboards for actionable insights.

### 3.2 ***Key Achievements & Solutions Implemented***

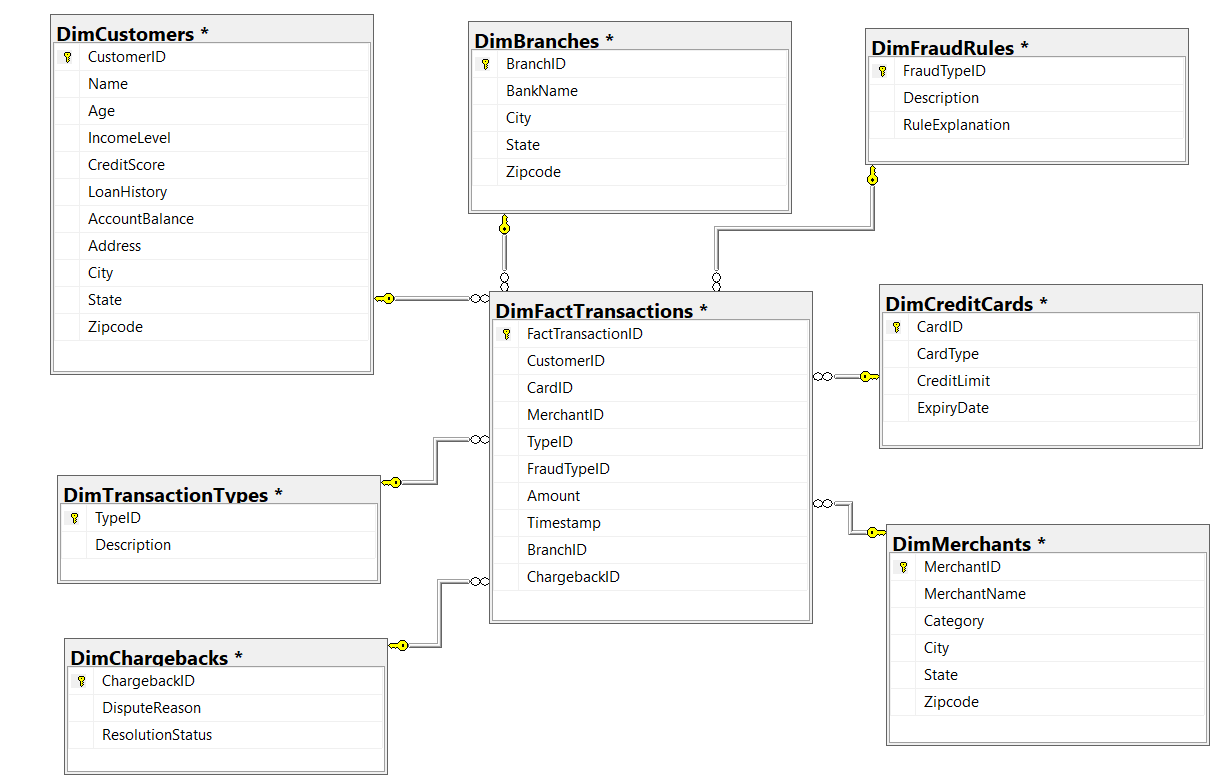
* Integrating Scattered Data into a Central Warehouse
  + Initially, data was dispersed across multiple Excel sheets and isolated systems. Through the development of ETL pipelines and a centralized SQL Server data warehouse, we successfully consolidated all transactional, customer, and merchant data into one unified platform for comprehensive analysis.
* Enabling Cube-Based Queries for Real-Time Risk Analysis
  + With the implementation of SSAS cubes, we enabled powerful multidimensional analysis. This allowed our stakeholders to explore fraud data across various dimensions such as customer, time, region, or merchant, uncovering hidden patterns and risks through slice-and-dice capabilities.
* Visualizing Fraud Patterns and Customer Segments
  + Power BI dashboards transformed complex data into intuitive visuals, highlighting high-risk customers, branches, and transaction types. This enabled our decision-makers to quickly identify threats and intervene proactively.
* Supporting Fraud Scoring and Early Flagging
  + By combining machine learning models with historical fraud records, we created a system that automatically flags high-risk patterns. This empowers teams to take preventive actions early, reducing the chance of financial loss and improving overall risk management.

## ***4. Data Model and Data Warehouse Model***

*Data Model:*



*Data Warehouse Model:*



## 

## ***5. Extraction, Transformation, Load***

We built an ETL package using SSIS to transfer our Python-generated synthetic data into SQL Server, where our data warehouse is hosted. This step laid the foundation for a clean, well-structured platform that supports the rest of our analytics and reporting solutions.

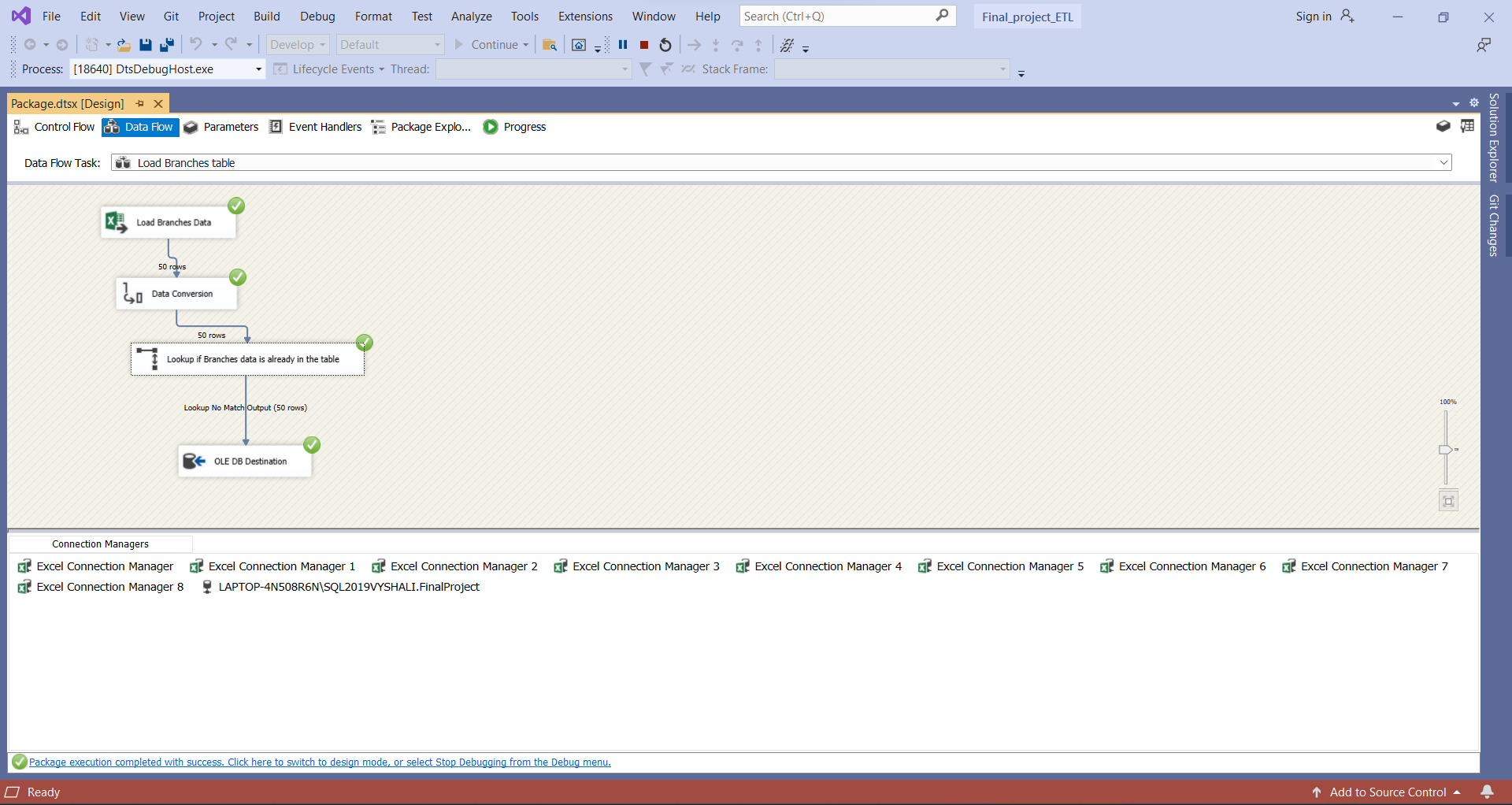
### ***5.1 Data Flow Task Example: Branches Table***

For example, the ETL process for the Branches table followed these key steps:

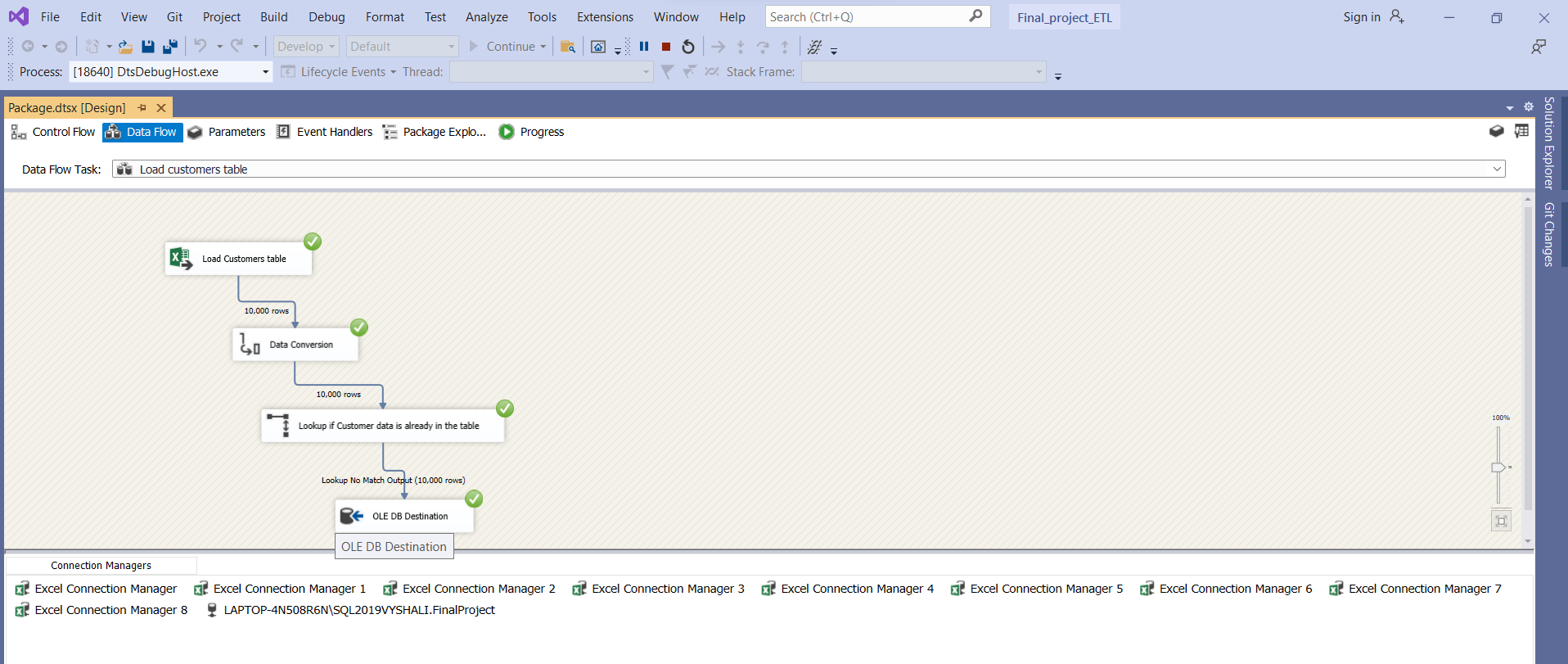
1. Data Source: An Excel file containing branch details was set up as the source.
2. Type Conversion: Data types were adjusted to align with the SQL Server schema.
3. Lookup Checks: Logic was added to identify and handle duplicates or invalid records before loading.
4. OLE DB Destination: The cleaned data was then loaded into the DimBranches table in SQL Server.

This approach served as a template and was similarly applied to other dimension tables like Customers, Merchants, Credit Cards, and Chargebacks, with the necessary transformations tailored to each dataset.

Branches Table

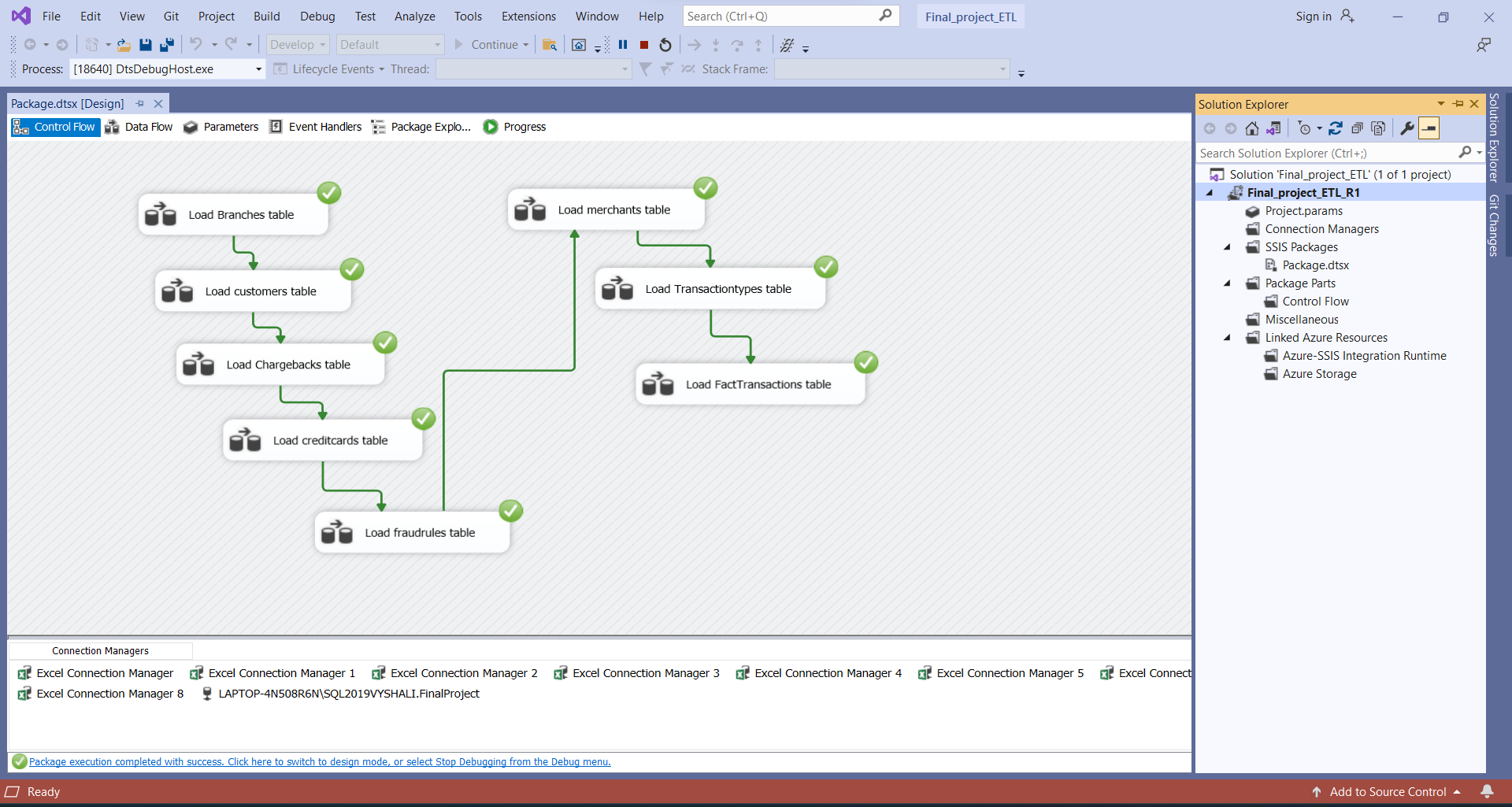


Customers Table



Similarly, this structured ETL approach was applied to the remaining tables, as detailed in the attached ETL file.

### 5.2 ***Control Flow Task for all the tables***



### 

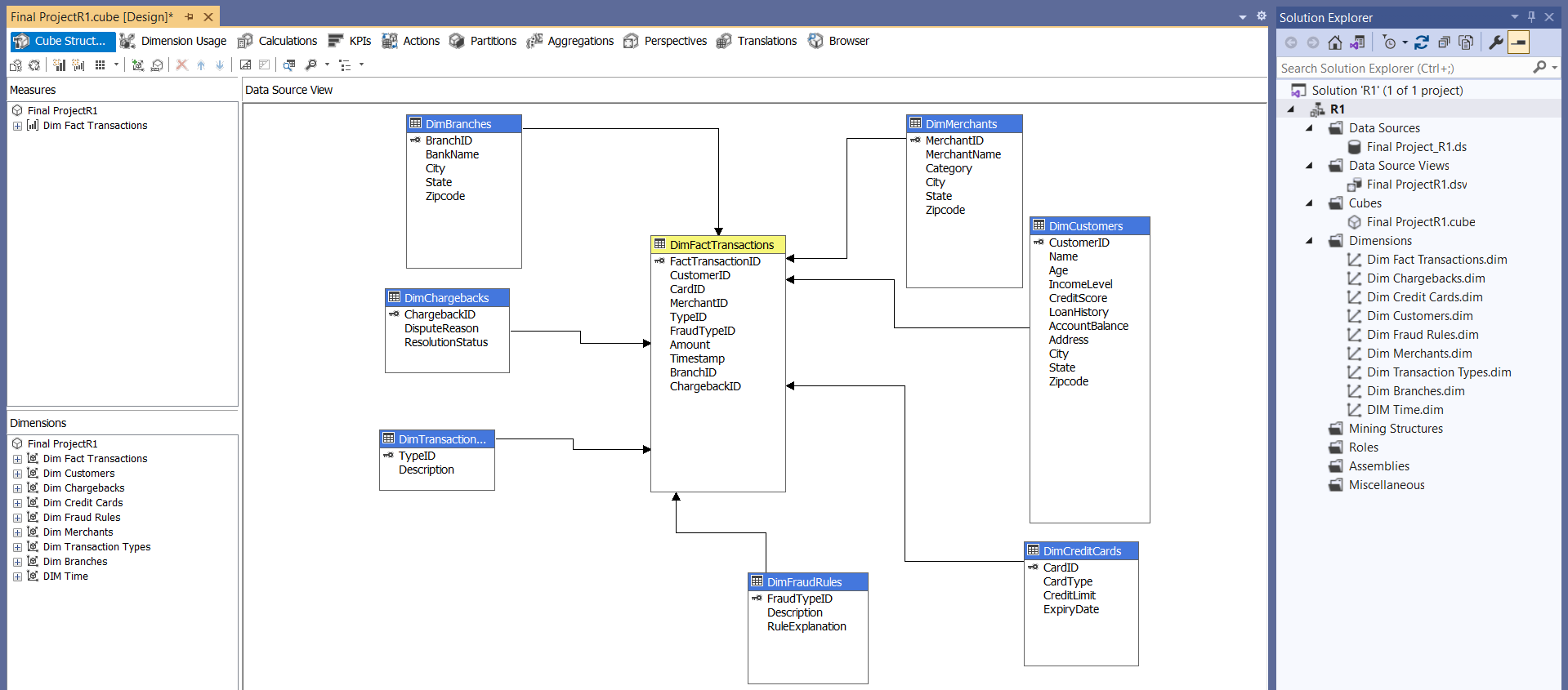
### ***5.3 Challenges Faced During ETL***

While building the ETL pipeline, we faced several practical challenges:

1. Data Type Mismatches: Some Excel columns didn’t match the expected data types in SQL Server particularly ZIP codes and date fields which required careful handling during transformation.
2. Foreign Key Integrity: To maintain proper relationships in the Fact table, we had to load the dimension tables in the correct sequence and ensure all dependencies were respected.
3. Missing Keys in Fact Table: Initially, the BranchID and ChargebackID fields were missing from the FactTransactions table. We used Excel's VLOOKUP to match and add these values from the respective dimension tables. This step was essential for maintaining referential integrity and enabling accurate KPI calculations, such as fraud and chargeback trends by branch.
4. Deployment Issues: Setting up SSIS with the correct file paths and permissions proved tricky, especially across different environments.

By addressing each of these issues step by step, we ensured that the data pipeline is reliable and ready to support accurate analysis and reporting.

## ***6. SSAS Cube***



Our SSAS multidimensional cube was built around the FactTransactions table and related dimension tables, making it easy for users to slice, dice, and aggregate data for credit risk and fraud analysis. Thanks to this structure, querying in Power BI is fast, smooth, and highly interactive.

### ***6.1 Measures and KPIs***

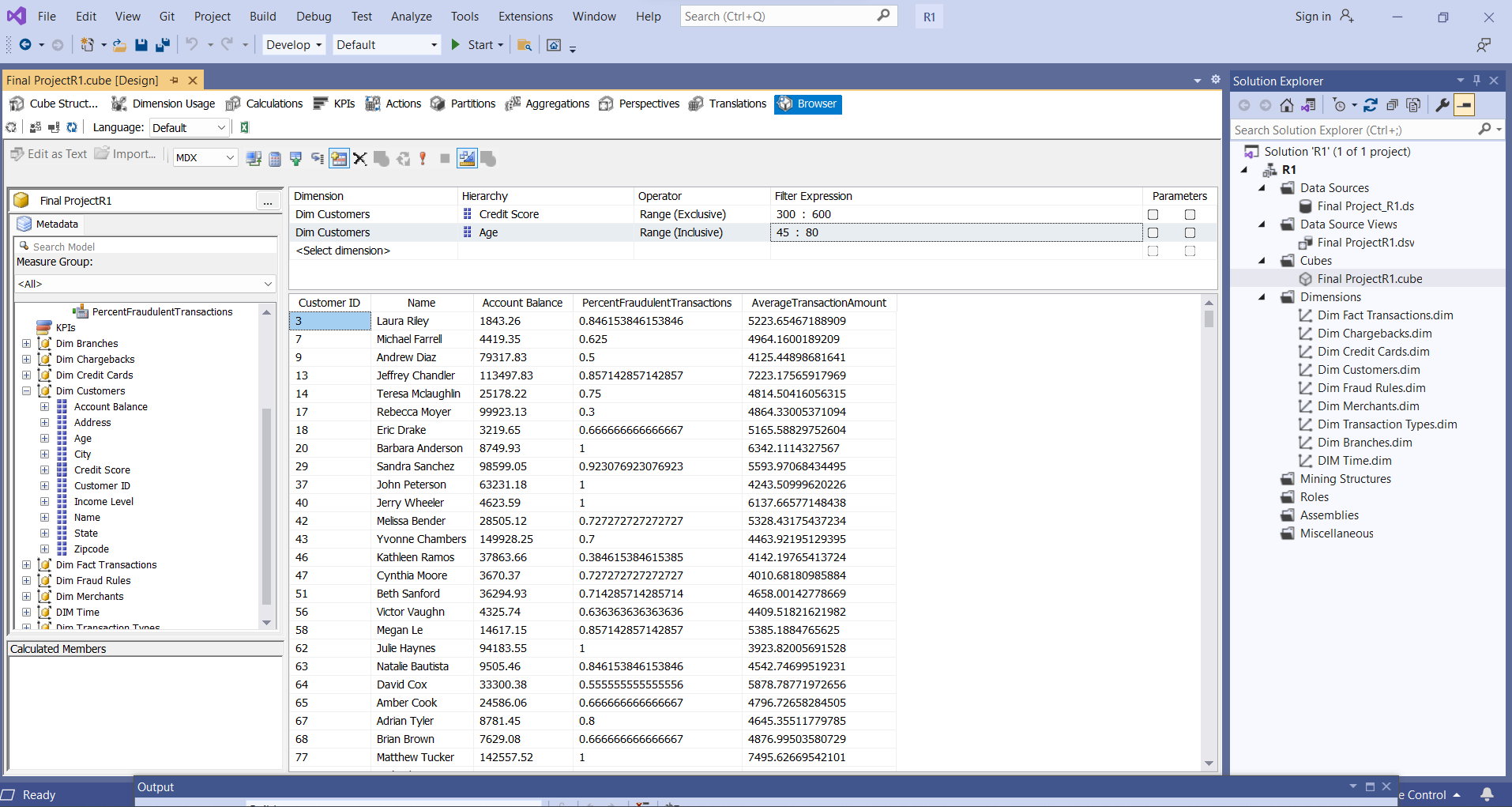
To support strategic decision-making and fraud analysis, we developed key measures from the Fact Transactions table and transformed them into actionable KPIs. These KPIs became crucial for tracking operational trends, visualizing fraud patterns, and answering key business questions, delivered effectively through Power BI dashboards.

We created a set of key measures from the FactTransactions table to help turn raw data into meaningful insights:

1. Average Transaction Amount:  
   This helps reveal general spending behavior across the customer base. Higher values might suggest potential high-value customers or suspiciously large transactions that could be signs of fraud.
2. Average Fraud Amount:  
   Focuses on the average value of fraudulent transactions. When filtered by month or year, it gives a clearer picture of the financial impact of fraud over time.
3. Chargeback Percentage:  
   Shows how many sales resulted in refunds or disputes. A high percentage can point to unhappy customers or weak points in the system.
4. Fraud Count:  
   Simply counts the number of fraud cases. It’s a useful way to track trends across different customer groups or time periods.
5. Percent of Fraudulent Transactions:  
   Calculates how much of the total transaction volume is made up of fraud. It’s ideal for comparing risk across different branches, merchants, or periods.
6. Total Transactions Count:  
   Tracks the total number of transactions and is used to give context to other metrics, like fraud rates by branch or region.

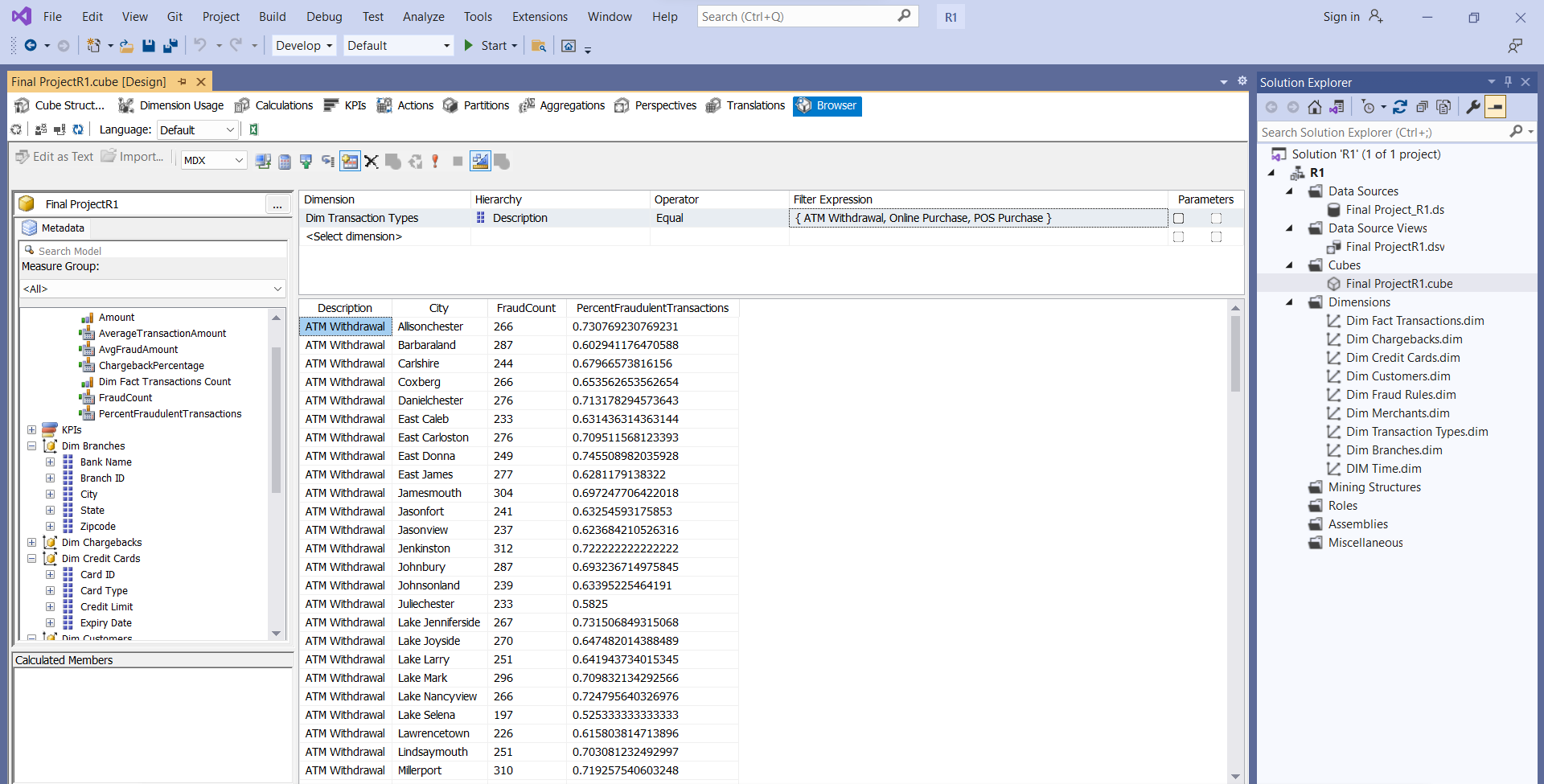
By turning complex transaction data into these clear, focused KPIs, we made it easy for managers to slice and explore the data in Power BI. The dashboards helped highlight risk areas, spot inefficiencies, and stay ahead of emerging fraud trends.

#### *KPI 1. Identify High-Risk Customers*



* The query results reveal that customers aged 45 to 80 with credit scores between 300 and 600 are considered high-risk. Their fraud exposure is evaluated using a custom KPI, PercentFraudulentTransactions, which helps business teams spot patterns and proactively flag accounts that show signs of potential fraud before it escalates.

#### *KPI 2. Analyze Fraud Trends by Transaction Type*



## 

* This KPI highlights which transaction types, like ATM withdrawals or online purchases, are more frequently targeted by fraud. These insights are crucial for shaping focused fraud detection and prevention strategies.

## ***7. Machine Learning Modeling for Fraud Risk Prediction***

To support our mission of enhancing credit risk assessment, we applied supervised machine learning models to predict the likelihood of fraudulent transactions based on historical customer and transaction data. Traditional rule-based methods often fall short when dealing with the complexity and volume of fraud, especially in environments like USA National Bank, where transaction volumes are high.

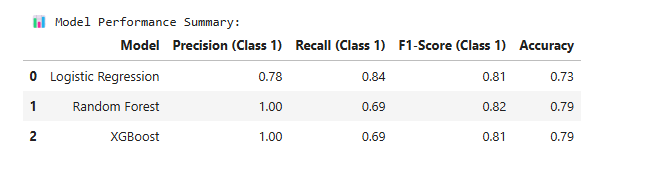
Before training the models, we carried out essential data preprocessing steps, such as handling missing values, encoding categorical variables, and selecting features based on correlation and business relevance. Our final feature set included critical indicators like credit score, income level, chargeback history, credit utilization, and transaction type all of which have a strong impact on fraud prediction.

### ***7.1 Models Implemented***

To build a robust fraud detection system, we applied and compared the following supervised classification models:

1. Logistic Regression – as a baseline statistical model that is mainly used to understand linear correlations between fraud risk and key features.
2. Random Forest Classifier – ensemble learning method that establishes several decision trees to achieve higher accuracy and decreased overfitting.
3. XGBoost Classifier – a fast gradient boosting model that is regarded as a choice of excellence not only for its speed but also for its predictive performance and is particularly effective for imbalanced datasets.

The models were examined on the consistent metrics precision, recall, F1-score, and support to evaluate their effectiveness in the fraud detection task.



### 7.2 ***Model Performance Summary***

1. Logistic Regression

Logistic Regression delivered the highest recall (0.84), making it the most effective at identifying fraud cases in our synthetic credit card dataset. However, its lower precision (0.78) and accuracy (0.73) led to a higher number of false positives. This model proved useful in scenarios where flagging all potential fraud cases was a priority, even if it meant more alerts for the investigation team.

1. Random Forest

The Random Forest model achieved perfect precision (1.00) in our testing, meaning all fraud predictions were accurate. It also maintained a balanced F1-score (0.82) and accuracy (0.79). This model stood out as the most dependable overall, offering a solid balance between catching fraud and minimizing false alarms, ideal for operational use in fraud screening.

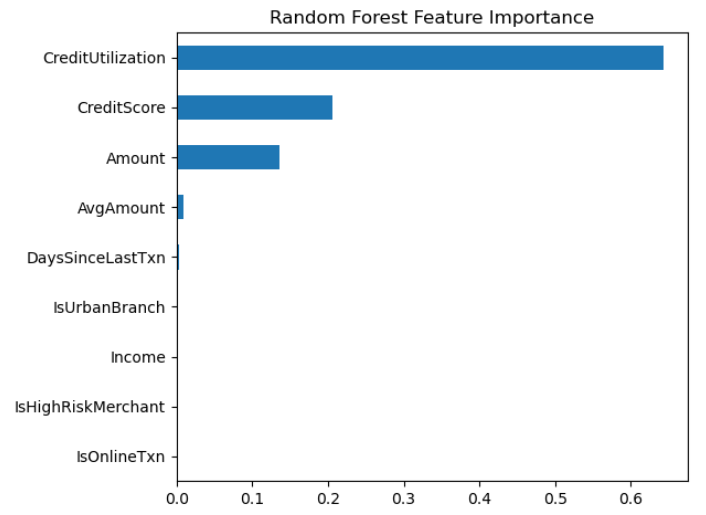
1. XGBoost

XGBoost also achieved perfect precision (1.00) with an F1-score of 0.81 and accuracy of 0.79. However, its recall (0.69) was lower than the other models, meaning it missed some fraud cases in our test runs. Despite that, its fast performance and scalability make it a strong candidate for real-time fraud detection, especially as transaction volumes grow across digital platforms at our Credit Union.

### ***7.3 Model Interpretation and Insights***

Through the assessment of numerous classification models, we pinpointed the best-performing characteristics that have a direct impact on fraud detection. The analysis was performed through the utilization of both feature importance scores and correlation analysis to authenticate the engineered variables and find out their behavior in connection with fraud.

Feature Importance (Random Forest)



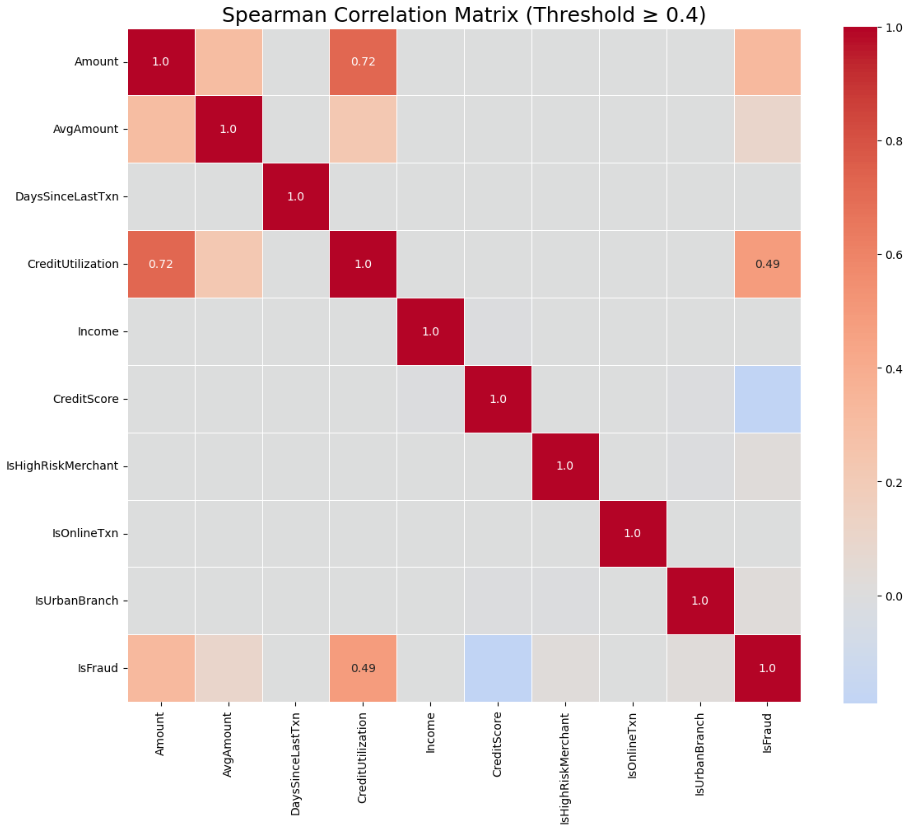
The Random Forest model highlighted the following as the most critical predictors:

1. Credit Utilization was the most decisive one, which was seen to have a significant influence in discriminating between the two classes.
2. Credit Score and Transaction Amount followed, both contributing significantly to the model’s predictive performance.

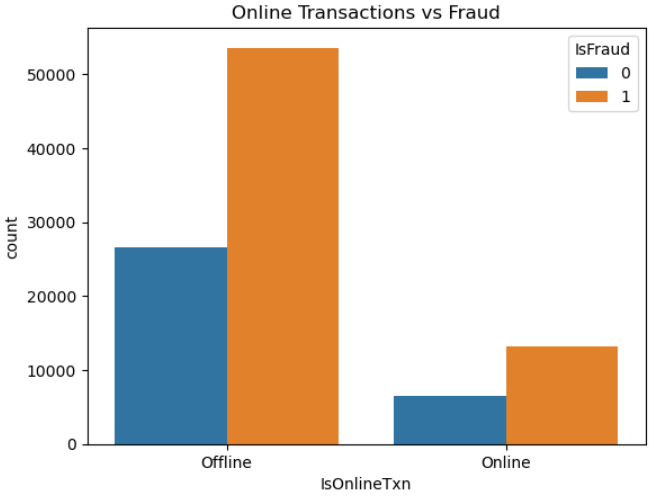
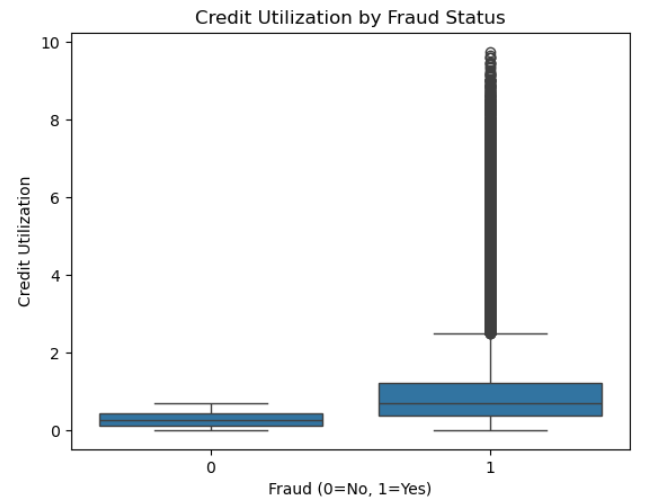
Spearman Correlation Insights

Spearman correlation matrix is used to further validate these findings:

1. Credit Utilization had a robust positive correlation (0.49) with fraud.
2. Amount and AvgAmount also showed moderate correlations with fraud.
3. Income and Credit Score had very low or even negative correlation values, which indicated the possibility of non-linear relationships or interactions.



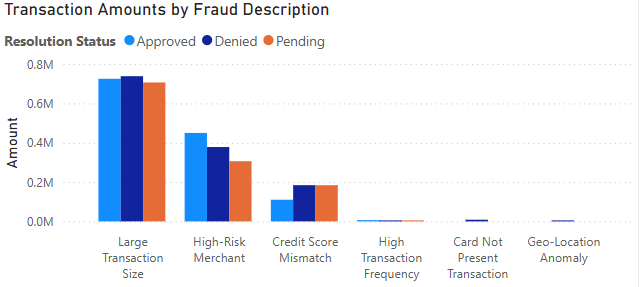
### ***7.4 Final Visual Insights***



To make the model results more meaningful, we visualized a few key features tied to fraud. The boxplot for Credit Utilization clearly showed that fraud cases tend to have much higher usage rates, backing up its role as a strong predictor. In figure 2, we looked at Online Transactions, which showed a noticeably higher number of fraud cases compared to offline ones. While both channels show fraud, the volume is much higher online, highlighting the need for closer monitoring. These visuals not only support what the model found but also help us focus on where to act, like keeping an eye on credit usage and tightening controls around digital transactions.

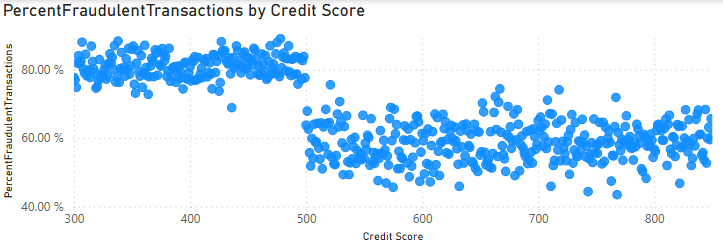
## ***8. PowerBI Dashboards***

### ***KPI 3. Transaction Amounts by Fraud Description***



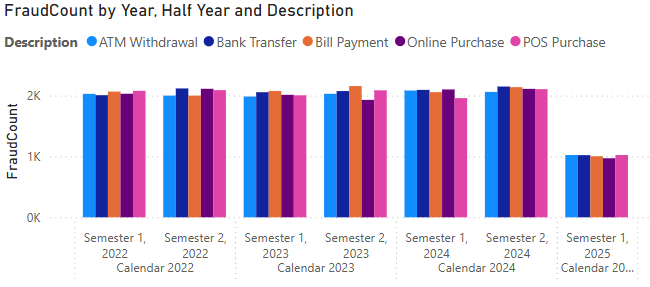
The plot shows that the most fraudulent transaction volume comes from large transactions and high-risk merchants, across all resolution statuses: Approved, Denied, and Pending. These two fraud types account for most losses, highlighting the need for stronger controls in high-value and merchant-related transactions.

### ***KPI 4. Fraud Percentage by Credit Score***



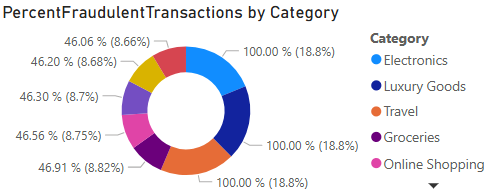
This scatter plot shows that fraud rates are much higher for customers with lower credit scores (300–500). As credit scores increase, the percentage of fraudulent transactions drops significantly, confirming that low credit scores are strongly associated with fraud risk.

### ***KPI 5. Fraud Count Over Time by Transaction Type***



This chart shows that fraud cases stayed steady across all transaction types like ATM withdrawals, bank transfers, and online purchases, from 2022 through 2024. But in early 2025, there’s a clear drop across the board, which could point to better fraud prevention efforts or a shift in how customers are using these services.

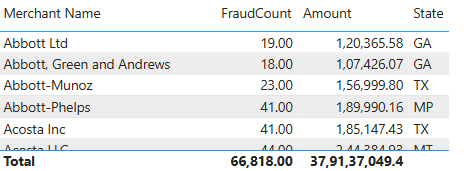
### ***KPI 6. Percent Fraudulent Transactions by Category***



The plot shows that fraud affects all categories, but it's most common in Electronics, Luxury Goods, and Travel, where 100% of the flagged transactions were fraudulent. It highlights the need for extra caution in these high-risk spending areas.

### ***KPI 7. Top Merchants by Fraud Count and Transaction Volume***

The table here identifies merchants who had a range of fraud cases of the most quantity and in terms of the most transaction amount, thus, it assists in locating businesses that are high-risk for scrutiny and policy adjustments.



## ***9. Summary and Lessons Learned***

Data Quality. One of the most important lessons was the understanding of how critical consistent, clean, and meaningful data is. Early in the project, we faced several issues with foreign key mismatches, missing IDs, and Excel formatting, which affected and delayed SSIS ETL package deployment and had downstream impacts on the cube itself. The issues underlined the importance of strong data governance, especially in a high-risk environment like banking

ETL Optimization. During SSIS package development, the team has realized that performance and load balancing are equally important as data transformation. Lookup dependencies, table load order, and error handling required a lot of attention to maintain the referential integrity of the data. We also discovered that optimizing data type conversion and minimizing unnecessary transformations dramatically improved SSIS package reliability.

SSAS Cubes. Designing a multidimensional cube with several measures allowed us to create drill-down analysis and advanced metrics like “Percent Fraudulent Transactions” and “Chargeback Rates”. Even so, the process required extensive trial and error, especially in aligning dimension attributes and measures. We learned that proper measure grouping, variable mapping, and accurate dimension usage are critical to getting the cube to behave as expected.

BI Dashboards. PowerBI dashboard helped us transform previously created cubes and their measures into meaningful insights that would allow us to improve internal decision making, identify fraud hotspots, track risky merchants, and monitor fraud trends by customer or transaction type. For decision-makers, this visual tool is the key to understanding the deeper story behind the data without the need for extensive explanations and reports from the company's data scientists.

## ***10.Instructions to run the entire DW/BI application***

1. Ensure SSAS,SSIS,SQL Server, and PowerBI are installed.
2. Restore the SQL Server Database.
3. Recreate a Data Source View for the Cube.
4. Update the server name and authentication as needed.
5. Deploy and process SSAS Cube.
6. Reconnect PowerBI to the Cube.

Individual Contributions

We acknowledge the collaborative effort involved in completing this project. Below is a summary of each team member’s individual contributions:

1. Vishaka Sharma
2. Vyshali Poola
3. Marko Khotynetskyi