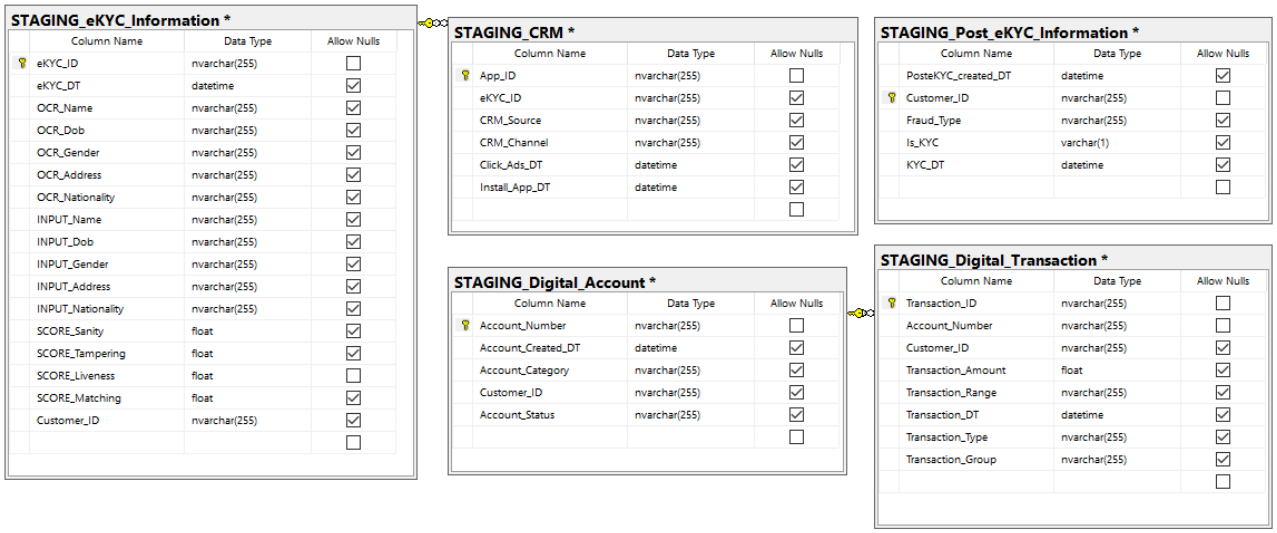
Case 2 - Metadata - Security and Quality

1. Position and role:

Position	Role
Business Analyst	Provide requirements for building a data warehouse system
Data Architect	Design and build a data warehouse system.
Data Engineer	Preprocess and integrate data into the data warehouse.
Data Govenance Analyst	Manage data quality

2. Create Schema for STAGING zone:



STAGING_eKYC_Information *

Column Name	Data Type	Allow Nulls
eKYC_ID	nvarchar(255)	☐
eKYC_DT	datetime	☑
OCR_Name	nvarchar(255)	☑
OCR_Dob	nvarchar(255)	☑
OCR_Gender	nvarchar(255)	☑
OCR_Address	nvarchar(255)	☑
OCR_Nationality	nvarchar(255)	☑
INPUT_Name	nvarchar(255)	☑
INPUT_Dob	nvarchar(255)	☑
INPUT_Gender	nvarchar(255)	☑
INPUT_Address	nvarchar(255)	☑
INPUT_Nationality	nvarchar(255)	☑
SCORE_Sanity	float	☑
SCORE_Tampering	float	☑
SCORE_Liveness	float	☐
SCORE_Matching	float	☑
Customer_ID	nvarchar(255)	☑
		☐

STAGING_CRM *

Column Name	Data Type	Allow Nulls
App_ID	nvarchar(255)	☐
eKYC_ID	nvarchar(255)	☑
CRM_Source	nvarchar(255)	☑
CRM_Channel	nvarchar(255)	☑
Click_Ads_DT	datetime	☑
Install_App_DT	datetime	☑
		☐

STAGING_Post_eKYC_Information *

Column Name	Data Type	Allow Nulls
PosteKYC_created_DT	datetime	☑
Customer_ID	nvarchar(255)	☐
Fraud_Type	nvarchar(255)	☑
Is_KYC	varchar(1)	☑
KYC_DT	datetime	☑
		☐

STAGING_Digital_Account *

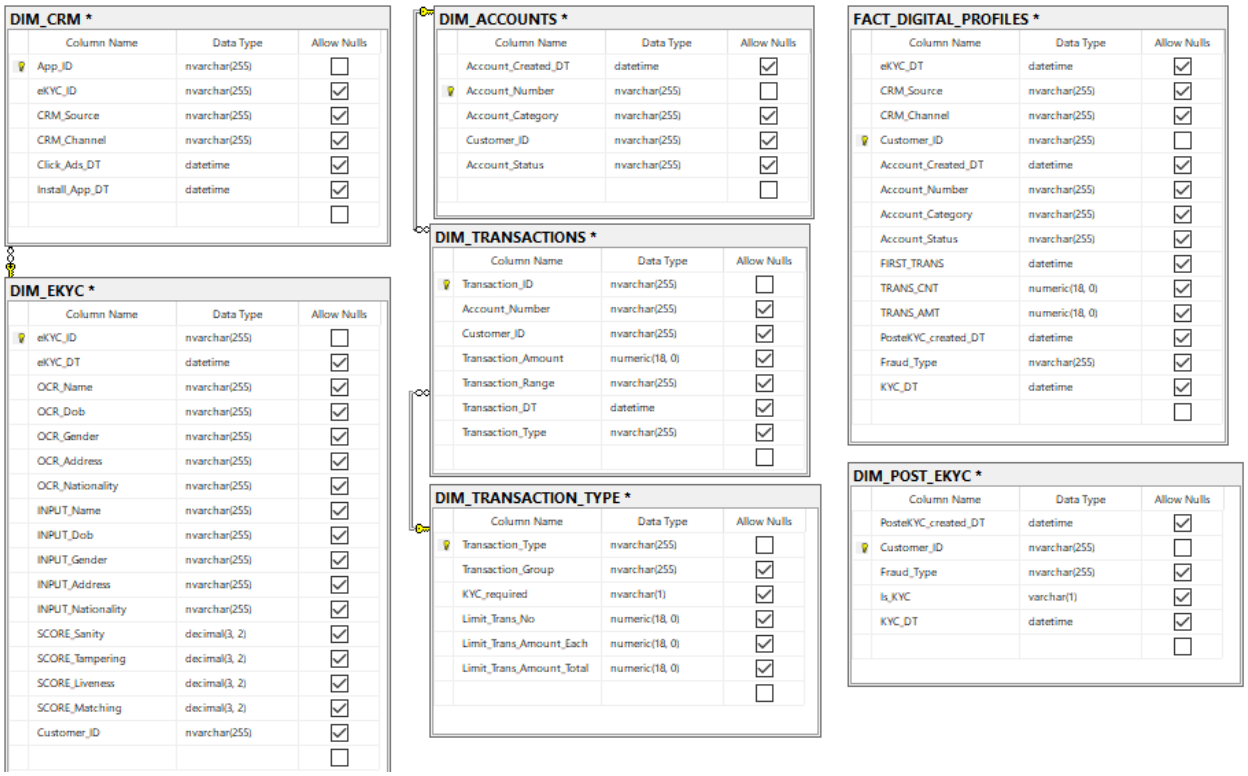
Column Name	Data Type	Allow Nulls
Account_Number	nvarchar(255)	☐
Account_Created_DT	datetime	☑
Account_Category	nvarchar(255)	☑
Customer_ID	nvarchar(255)	☑
Account_Status	nvarchar(255)	☑
		☐

STAGING_Digital_Transaction *

Column Name	Data Type	Allow Nulls
Transaction_ID	nvarchar(255)	☐
Account_Number	nvarchar(255)	☐
Customer_ID	nvarchar(255)	☑
Transaction_Amount	float	☑
Transaction_Range	nvarchar(255)	☑
Transaction_DT	datetime	☑
Transaction_Type	nvarchar(255)	☑
Transaction_Group	nvarchar(255)	☑
		☐

Figure 2-1: STAGING Schema.

3. Create schema for RECONCILIATION zone:



DIM_CRM *

Column Name	Data Type	Allow Nulls
App_ID	nvarchar(255)	☐
eKYC_ID	nvarchar(255)	☑
CRM_Source	nvarchar(255)	☑
CRM_Channel	nvarchar(255)	☑
Click_Ads_DT	datetime	☑
Install_App_DT	datetime	☑
		☐

DIM_ACCOUNTS *

Column Name	Data Type	Allow Nulls
Account_Created_DT	datetime	☑
Account_Number	nvarchar(255)	☐
Account_Category	nvarchar(255)	☑
Customer_ID	nvarchar(255)	☑
Account_Status	nvarchar(255)	☑
		☐

FACT_DIGITAL_PROFILES *

Column Name	Data Type	Allow Nulls
eKYC_DT	datetime	☑
CRM_Source	nvarchar(255)	☑
CRM_Channel	nvarchar(255)	☑
Customer_ID	nvarchar(255)	☐
Account_Created_DT	datetime	☑
Account_Number	nvarchar(255)	☑
Account_Category	nvarchar(255)	☑
Account_Status	nvarchar(255)	☑
FIRST_TRANS	datetime	☑
TRANS_CNT	numeric(18, 0)	☑
TRANS_AMT	numeric(18, 0)	☑
PosteKYC_created_DT	datetime	☑
Fraud_Type	nvarchar(255)	☑
KYC_DT	datetime	☑
		☐

DIM_TRANSACTIONS *

Column Name	Data Type	Allow Nulls
Transaction_ID	nvarchar(255)	☐
Account_Number	nvarchar(255)	☑
Customer_ID	nvarchar(255)	☑
Transaction_Amount	numeric(18, 0)	☑
Transaction_Range	nvarchar(255)	☑
Transaction_DT	datetime	☑
Transaction_Type	nvarchar(255)	☑
		☐

DIM_EKYC *

Column Name	Data Type	Allow Nulls
eKYC_ID	nvarchar(255)	☐
eKYC_DT	datetime	☑
OCR_Name	nvarchar(255)	☑
OCR_Dob	nvarchar(255)	☑
OCR_Gender	nvarchar(255)	☑
OCR_Address	nvarchar(255)	☑
OCR_Nationality	nvarchar(255)	☑
INPUT_Name	nvarchar(255)	☑
INPUT_Dob	nvarchar(255)	☑
INPUT_Gender	nvarchar(255)	☑
INPUT_Address	nvarchar(255)	☑
INPUT_Nationality	nvarchar(255)	☑
SCORE_Sanity	decimal(3, 2)	☑
SCORE_Tampering	decimal(3, 2)	☑
SCORE_Liveness	decimal(3, 2)	☑
SCORE_Matching	decimal(3, 2)	☑
Customer_ID	nvarchar(255)	☑
		☐

DIM_TRANSACTION_TYPE *

Column Name	Data Type	Allow Nulls
Transaction_Type	nvarchar(255)	☐
Transaction_Group	nvarchar(255)	☑
KYC_required	nvarchar(1)	☑
Limit_Trans_No	numeric(18, 0)	☑
Limit_Trans_Amount_Each	numeric(18, 0)	☑
Limit_Trans_Amount_Total	numeric(18, 0)	☑
		☐

DIM_POST_EKYC *

Column Name	Data Type	Allow Nulls
PosteKYC_created_DT	datetime	☑
Customer_ID	nvarchar(255)	☐
Fraud_Type	nvarchar(255)	☑
Is_KYC	varchar(1)	☑
KYC_DT	datetime	☑
		☐

Figure 3-1: RECILINATION Schema.

4. Integate data by using ETL Tools:

4.1. Transform and filter data:

Please refer to the Excel file for a better understanding of the BA's requirements regarding data mapping between tables.

Source-Target_Mapping.xlsx

4.2. ETL:

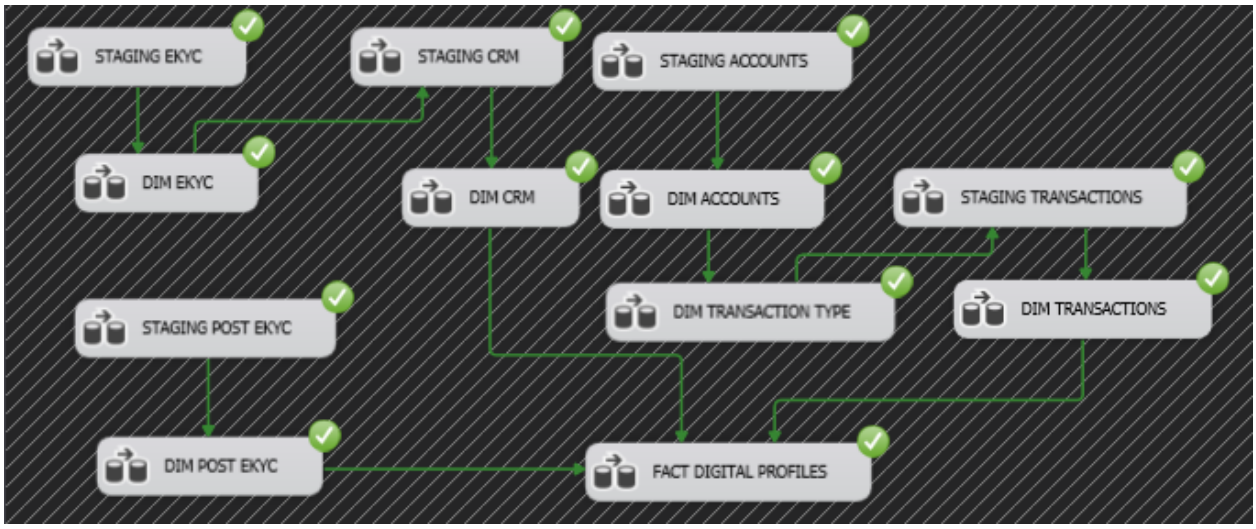


Figure 4-1: Result

In addition to the one-to-one mapping tables, there are other tables that need to be filtered according to requirements:

- [dbo].[STAGING_eKYC_Information]:

```
SELECT [eKYC_ID]
      ,[eKYC_DT]
      ,JSON_VALUE([OCR_INFO],'$.name') [OCR_Name]
         ,JSON_VALUE([OCR_INFO],'$.dob') [OCR_Dob]
         ,JSON_VALUE([OCR_INFO],'$.gender') [OCR_Gender]
         ,JSON_VALUE([OCR_INFO],'$.nationality') [OCR_Nationality]
         ,JSON_VALUE([OCR_INFO],'$.address') [OCR_Address]
         ,JSON_VALUE([INPUT_INFO],'$.name') [INPUT_Name]
         ,JSON_VALUE([INPUT_INFO],'$.dob') [INPUT_Dob]
         ,JSON_VALUE([INPUT_INFO],'$.gender') [INPUT_Gender]
         ,JSON_VALUE([INPUT_INFO],'$.address') [INPUT_Address]
         ,JSON_VALUE([INPUT_INFO],'$.nationality') [INPUT_Nationality]
      ,CAST([SANITY_SCORE] AS decimal(3,2)) [SCORE_Sanity]
         ,CAST([SANITY_SCORE] AS decimal(3,2)) [SCORE_Tampering]
         ,CAST([SANITY_SCORE] AS decimal(3,2)) [SCORE_Liveness]
         ,CAST([SANITY_SCORE] AS decimal(3,2)) [SCORE_Matching]
      ,[CUSTOMER_ID]
   FROM [ONBOARDING].[dbo].[ONBOARDING_Data]
```

- [dbo].[STAGING_Digital_Account]:

```
SELECT  [CREATED_DT]
      ,[Transaction_Account]
      ,[Account_Category]
      ,[CUSTOMER_ID]
      ,[ACCOUNT_STATUS]
FROM [CORE_T24].[dbo].[T24_ACCOUNT]
```

```
   WHERE Account_Category in ('1001','1002')
```

- [dbo].[DIM_TRANSACTION_TYPE]:

```
SELECT  [Transaction_Type]
      ,[Transaction_Group]
FROM [CORE_T24].[dbo].[T24_TRANSACTION]
GROUP BY  [Transaction_Type]
         ,[Transaction_Group]
```

- [dbo].[STAGING_Digital_Account]

```
SELECT [Transaction_ID]
      ,[Transaction_Account]
      ,[CUSTOMER_ID]
      --,[Channel]
         ,[Transaction_Amount]
         , IIF([Transaction_Amount] < 1000000, 'LOW'
                     , IIF([Transaction_Amount] < 10000000, 'MEDIUM LOW'
                           , IIF([Transaction_Amount] < 100000000, 'MEDIUM HIGH',
'HIGH')))   [Transaction_Range]
         ,[Transaction_DT]
      ,[Transaction_Type]
      ,[Transaction_Group]
  FROM [CORE_T24].[dbo].[T24_TRANSACTION]
     WHERE [Channel] = 'APP'
```

- FACT

```
SELECT E.[eKYC_DT], E.[Customer_ID]
                    , C.[CRM_Source], c.[CRM_Channel]
                    , A.[Account_Created_DT], A.[Account_Number],
A.[Account_Category], A.[Account_Status]
                    , T.FIRST_TRANS, T.TRANS_CNT, T.TRANS_AMT
                    , P.[PosteKYC_created_DT]
                    , P.[Fraud_Type]
                    , P.[KYC_DT]
                    --,
       FROM [DWH].[dbo].[DIM_EKYC] E
       LEFT JOIN [DWH].[dbo].[DIM_CRM] c on E.[eKYC_ID] = C.[eKYC_ID]
       LEFT JOIN [DWH].[dbo].[DIM_ACCOUNTS] A on E.[Customer_ID] = A.[Customer_ID]
       LEFT JOIN (SELECT [Customer_ID],[Account_Number]
                    , MIN([Transaction_DT]) FIRST_TRANS
                    , COUNT(distinct [Transaction_ID]) TRANS_CNT
                    , SUM([Transaction_Amount]) TRANS_AMT
                          FROM [DWH].[dbo].[DIM_TRANSACTIONS]
                          GROUP BY  [Customer_ID],[Account_Number]
                                 ) T ON E.[Customer_ID] = T.[Customer_ID]
                                 AND A.[Account_Number] = T.[Account_Number]
       LEFT JOIN [DWH].[dbo].[DIM_POST_EKYC] P ON   E.[Customer_ID] =
P.[Customer_ID]
       WHERE E.[Customer_ID] IS NOT NULL
```

4.3. Some issues encountered during the data ETL process:

– Data type conflicts:

- During the data loading process from CORE_T24 into the table [STAGING Digital Transaction], a data type conflict arises because the input data (CORE_T24) in the Transaction_Range column is of type String, while the Transaction_Range column in the STAGING Digital Transaction table in the database is of type nvarchar. Therefore, it's necessary to convert the input data to Unicode string [DT_WSTR].

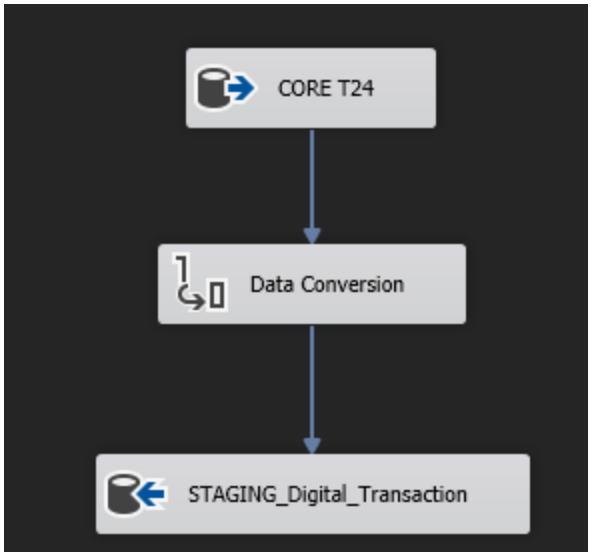


Figure 4-2:STAGING_Digital_Transaction Data Flow

- During the data loading process from an Excel file into the table [STAGING_Post_eKYC_Information], the input data type for the IS_KYC column is String. However, in the database, its data type is varchar, which means ASCII. Therefore, it's necessary to change the code page for the Data Conversion from 1258 to 1252.

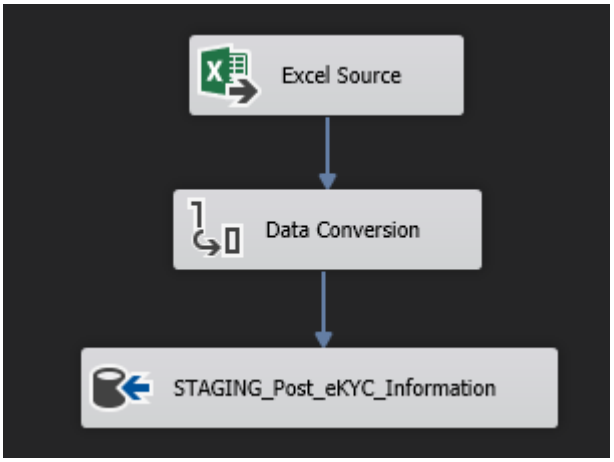


Figure 4-3: STAGING_Post_eKYC_Information Data Flow

5. Data quality:

5.1. Data Accuracy Report:

- Write a query with the following fields:
  - **IS_TIME_ALERT**: Evaluate the timestamps when customers perform actions to determine if they are reasonable (click -> install -> eKYC -> Account Created -> First Transaction).
  - **IS_SCORE_ALERT**: Customers have completed KYC, but their scores do not meet the required criteria.
  - **IS_ONFO_ALERT**: Customers have completed KYC, but the data from OCR does not match the data entered by the customers.
  - **IS_CATEGORY_ALERT**: Customer accounts are not of type 1001 and 1002.
  - **Fraud_Type**: Classification of customers

```
select F.Customer_ID
       , IIF(C.Click_Ads_DT>C.Install_App_DT OR C.Click_Ads_DT > F.eKYC_DT OR
C.Click_Ads_DT > F.Account_Created_DT OR C.Click_Ads_DT > F.FIRST_TRANS
              OR C.Install_App_DT> F.eKYC_DT OR C.Install_App_DT > F.Account_Created_DT
OR C.Install_App_DT > F.FIRST_TRANS
              OR F.eKYC_DT > F.Account_Created_DT OR F.eKYC_DT > F.FIRST_TRANS
              OR F.Account_Created_DT > F.FIRST_TRANS, 1,0) IS_TIME_ALERT
       , IIF(E.[SCORE_Sanity] < 0.85
              OR E.[SCORE_Tampering] < 0.85
              OR E.[SCORE_Liveness] < 0.85
              OR E.[SCORE_Matching] < 0.85 , 1 , 0) IS_SCORE_ALERT
       , IIF( [OCR_Name]<>[INPUT_Name]
              OR [OCR_Dob] <> [INPUT_Dob]
              OR [OCR_Gender] <> [INPUT_Gender]
        OR [OCR_Address] <> [INPUT_Address]
        OR [OCR_Nationality]<>[INPUT_Nationality],1,0) IS_INFO_ALERT
       , IIF(F.Account_Category NOT IN ('1001','1002'),1,0) IS_CATEGORY_ALERT
       , F.Fraud_Type
FROM FACT_DIGITAL_PROFILES F
LEFT JOIN DIM_EKYC E ON E.Customer_ID=F.Customer_ID
LEFT JOIN DIM_CRM C ON C.eKYC_ID=E.eKYC_ID
```

With the data obtained from the query above and using an Excel Pivot Table, we have the following two tables :

Row Labels	Sum of IS_TIME_ALERT	Sum of IS_SCORE_ALERT	Sum of IS_INFO_ALERT	Sum of IS_CATEGORY_ALERT	Count of Customer_I
CHECKED	116	157	0	0	3553
FRAUD	138	271	0	0	6000
RISK	43	75	0	0	1783
NORMAL	551	1504	0	0	35780
Grand Total	848	2007	0	0	47116

Figure 5-1: Accuracy Report.

Row Labels	Sum of IS_TIME_ALERT	Sum of IS_SCORE_ALERT	Sum of IS_INFO_ALERT	Sum of IS_CATEGORY_ALERT	Count of Customer_I
CHECKED	3%	4%	0	0	3553
FRAUD	2%	5%	0	0	6000
RISK	2%	4%	0	0	1783
NORMAL	2%	4%	0	0	35780

Figure 5-2: Accuracy Report (percentage).

- Evaluate:
  - Based on the report, we can observe that there are no violations in the IS_INFO_ALERT and IS_CATEGORY_ALERT fields, indicating consistency between OCR data and customer-entered data. Additionally, all customers have accounts belonging to categories 1001 and 1002.
  - However, for the IS_SCORE_ALERT and IS_TIME_ALERT fields, the number of customers violating these criteria is quite high, especially among customers labeled as NORMAL in the IS_SCORE_ALERT field, reaching up to 1504 customers. Therefore, it is necessary to review and verify this customer segment to prevent them from causing negative impacts on the bank. Additionally, when calculating the percentage of customers relative to the total, the IS_SCORE_ALERT field also accounts for a relatively high proportion (4-5%). Therefore, it is essential to check whether the OCR system is functioning properly or if the criteria for scores are not appropriate.

5.2. Data Consistency Report:

- Create a report table with the following columns:
  - **eKYC_MONTH**: month of eKYC execution
  - **FRAUD_NOT_CLOSED**: customers flagged as Fraud but whose accounts are not closed
  - **RISK_NOT_SUSPENDED**: customers flagged as Risk but whose activities are not suspended or warned
  - **CHECK_NOT_ACTIVE**: normal customers whose accounts are not allowed to operate
  - **RISK_TRANS**: transactions performed by Risk-labeled customers that should have been restricted
  - **FRAUD_TRANS**: transactions performed by Fraud-labeled customers

```
SELECT CONVERT(VARCHAR(6),EKYC_DT,112) eKYC_MONTH
              , F.Customer_ID
              , IIF(Fraud_Type = 'FRAUD' AND Account_Status <> 'CLOSED',1,0)
FRAUD_NOT_CLOSED
              , IIF(Fraud_Type = 'RISK' AND Account_Status <> 'SUSPENDED',1,0)
RISK_NOT_SUSPENDED
              , IIF(Fraud_Type = 'CHECKED' AND Account_Status <> 'ACTIVE',1,0)
CHECK_NOT_ACTIVE
```

```
                , IIF(FRAUD_TRANS.Account_Number IS NOT NULL,1,0) FRAUD_TRANS
                , FRAUD_TRANS_ID
                , FRAUD_TRANS_AMOUNT
                , FRAUD_TRANS_GROUP
                , FRAUD_TRANS_RANGE
                , IIF(RISK_TRANS.Account_Number IS NOT NULL,1,0) RISK_TRANS
                , RISK_TRANS_ID
                , RISK_TRANS_AMOUNT
                , RISK_TRANS_GROUP
                , RISK_TRANS_RANGE
        FROM FACT_DIGITAL_PROFILES F
        LEFT JOIN (
                SELECT F.Customer_ID
                        , T.Transaction_ID FRAUD_TRANS_ID
                        , T.Account_Number
                        , T.Transaction_Amount FRAUD_TRANS_AMOUNT
                        , TT.Transaction_Group FRAUD_TRANS_GROUP
                        , T.Transaction_Range FRAUD_TRANS_RANGE
                FROM DIM_TRANSACTIONS T LEFT JOIN FACT_DIGITAL_PROFILES F ON
F.Account_Number=T.Account_Number
                LEFT JOIN DIM_TRANSACTION_TYPE TT ON TT.Transaction_Type=T.Transaction_Type
                WHERE T.Transaction_DT>=F.PosteKYC_created_DT
                        AND F.Fraud_Type='FRAUD'
        ) FRAUD_TRANS ON F.Account_Number=FRAUD_TRANS.Account_Number
        LEFT JOIN (
                SELECT F.Customer_ID
                        , T.Transaction_ID RISK_TRANS_ID
                        , T.Account_Number
                        , T.Transaction_Type RISK_TRANS_AMOUNT
                        , TT.Transaction_Group RISK_TRANS_GROUP
                        , T.Transaction_Range RISK_TRANS_RANGE
                FROM DIM_TRANSACTIONS T LEFT JOIN FACT_DIGITAL_PROFILES F ON
F.Account_Number=T.Account_Number
                LEFT JOIN DIM_TRANSACTION_TYPE TT ON TT.Transaction_Type=T.Transaction_Type
                WHERE T.Transaction_DT>=F.PosteKYC_created_DT
                        AND TT.Transaction_Group='DEPOSIT'
                        AND F.Fraud_Type='RISK'
        ) RISK_TRANS ON F.Account_Number=RISK_TRANS.Account_Number
```

For the data obtained from the query above and using an Excel Pivot Table, we have the following report:

eKYC_MONTH	FRAUD_NOT_CLOSED	RISK_NOT_SUSPENDED	CHECK_NOT_ACTIVE	FRAUD_TRANS	RISK_TRANS
202201	0	0	0	0	0
202202	0	0	0	1	0
202203	0	0	0	3	0
202204	0	0	0	8	1
202205	0	0	0	17	0
202206	0	0	0	29	1
202207	0	0	0	43	2
202208	0	0	0	61	7
202209	0	0	0	128	5
202210	0	0	0	186	16
202211	0	0	0	309	20
202212	0	0	0	549	33
Grand Total	0	0	0	1334	85

Figure 5-3: Data Consistency Report.

eKYC_MONTH	DEPOSIT				PAYMENT						TRANSFER							
	MEDIUM HIGH		HIGH		MEDIUM LOW		MEDIUM HIGH		HIGH		LOW		MEDIUM LOW		MEDIUM HIGH		HIGH	
	CNT	Total_AMT	CNT	Total_AMT	CNT	Total_AMT	CNT	Total_AMT	CNT	Total_AMT	CNT	Total_AMT	CNT	Total_AMT	CNT	Total_AMT	CNT	Total_AMT
202202															1	61.776.328		
202203									2	815.158.406					1	75.861.090		
202204	1	79.877.054	2	882.304.902					4	1.016.052.012					1	83.392.978		
202205			2	781.683.503					9	2.588.012.330					2	135.887.112	4	544.450.362
202206	1	68.705.325	5	2.049.743.148	1	6.352.865	1	51.174.150	7	2.236.631.720	1	286.012	2	5.250.112	7	312.659.833	4	535.991.359
202207			9	3.641.863.800			6	271.192.674	20	6.706.762.574					3	106.261.696	5	807.861.491
202208	1	24.425.022	7	4.587.574.226	1	4.574.979	3	167.995.998	14	4.285.078.231			1	8.496.865	14	607.623.916	20	3.037.604.672
202209	4	197.301.506	32	15.301.070.084	1	6.634.659	7	322.158.384	42	13.539.664.874			6	23.892.958	15	739.330.351	21	3.217.045.089
202210	4	185.898.646	28	14.013.686.302	1	1.547.964	8	538.629.814	82	24.746.129.500			2	13.998.147	24	1.460.453.600	37	5.795.739.942
202211	4	245.603.539	51	30.638.784.816	3	22.711.394	19	855.421.289	113	32.540.314.548			3	13.102.249	58	3.167.823.602	58	8.764.453.123
202212	10	519.229.480	114	68.330.383.881	3	16.183.861	45	2.480.221.259	183	57.623.433.939			11	70.804.703	97	5.550.954.519	86	12.922.210.885
Grand Total	25	1.321.040.572	250	140.227.094.662	10	58.005.722	89	4.686.793.568	476	146.097.238.134	1	286.012	25	135.545.034	223	12.302.025.025	235	35.625.356.923

Figure 5-4: Detail report on transactions by customers labeled as FRAUD.