Implementing an Information System in Traveling Reimbursement

Contents

[1. Abstract: 2](#_Toc133920185)

[2. Introduction: 2](#_Toc133920186)

[3. Methodology: 3](#_Toc133920187)

[*3.1* *Entity-Relationship Diagramming (EERD)* 3](#_Toc133920188)

[3.1.1 Summary: 3](#_Toc133920189)

[3.1.2 Explanation: 3](#_Toc133920190)

[3.1.3 Advantages of Entity-Relationship Diagramming (EERD): 4](#_Toc133920191)

[3.1.4 Disadvantages of Entity-Relationship Diagramming (EERD): 4](#_Toc133920192)

[*3.2* SQL Query Analysis*:* 4](#_Toc133920193)

[3.2.1 Summary: 4](#_Toc133920194)

[3.2.2 Explanation: 5](#_Toc133920195)

[3.2.3 Example of SQL Query: 5](#_Toc133920196)

[3.2.4 Advantage of SQL Query: 5](#_Toc133920197)

[3.2.5 Disadvantage of SQL Query: 6](#_Toc133920198)

[3.3 *Graph Database (Neo4j):* 6](#_Toc133920199)

[3.3.1 Summary: 6](#_Toc133920200)

[3.3.2 Example of Neo4j: 6](#_Toc133920201)

[3.3.3 Advantage of Neo4j: 7](#_Toc133920202)

[3.3.4 Disadvantage of Neo4j: 7](#_Toc133920203)

[4. Conclusion: 7](#_Toc133920204)

# Abstract:

This report presents a comparative analysis of various methodologies and tools utilized in the design, development, and management of an information system for a university's travel reimbursement process. The study shows the entire process from entity-relationship diagramming to SQL query analysis and graph database implementation.

# Introduction:

This report explains the process of creating an Entity Relationship Diagram (EERD) to model the reimbursement procedure for travel expenses at a university. The model encompasses various entities including travelers, travel arrangers, supervisors, budget officers, and university financial services, with each having specific roles in the process. Travel expenses are influenced by factors such as the travel agency involved, trip location, and the means of transportation used.

Our group faced challenges in determining the relationships between the entities and their attributes, and in organizing the diagram for easy comprehension. Solutions were sought through team discussions and division of tasks. Revisions were made to the EERD model based on feedback, which included removing certain attributes, changing primary keys, and creating a new fact table for calculating travel expenses.

The report also outlined the database schema and constraints and discussed the SQL queries used for data analysis. The schema consisted of entities such as "People," "Trip," "Travel Arranger," "Supervisor," "Department," "Reimbursement Agency," and "Reimbursement Type," each with unique identifiers and related attributes. SQL queries were used to answer business questions about total costs and overall expenditure by each traveler.

In the final phase, the database was converted into CSV files, which were then uploaded into a Neo4j database. Cypher queries were used to answer business questions such as the total amount spent by each employee on meals, the purpose of each trip, and the preferred vendor for future trips. The team overcame challenges associated with creating nodes and importing external libraries and made inferences based on available data.

Throughout the project, group members contributed to various tasks such as understanding the case study, creating the EER diagram, writing SQL queries, importing the database into Neo4j, and writing the final report.

# Methodology:

### Entity-Relationship Diagramming (EERD)

#### Summary:

In the first phrase, we implemented the foundational step involving the creation of an EERD model, a visual representation of data relationships and attributes. Our illustration shows how individuals who use the University's Travel Procurement Card Program apply for travel reimbursements, as well as the regulations and procedures for travel-related costs incurred on university business.

#### Explanation:

Traveling expenses, the travel agency that is arranging the trip, the topography of the location, and cost comparisons for various means of transportation all have an influence on the reimbursement procedure and prices.

The people involved in the reimbursement process will be our generalized entitles and their subcategories like Travelers, Travel arrangers, supervisors, budget officers and university financial services will be the specialized ones. These can be travelers, supervisors, budget agencies and university financial services, like who will be the people going to travel who will make all the arrangements for them and decide the budget for their visit.

After this, we specify which agency covers which part of the travel, like what is their mode, did they travel via flights or by road and according to that agency, attributes are defined. If they travel internationally, different agencies will take over, and last, based on their status, they must submit their receipts. According to that, university financial services can be approved.

After this, it comes to what type of reimbursement can be an individual file, if an individual is going for a sponsored project or any conference, if any family members are travelling along with them, and where they are staying.

If we go to the diagram at a more granular level, then we can add the meals, their expenses, baggage or extra baggage, and housekeeping tips, which might not get into the reimbursement section.

Next, we should focus on the destinations people are travelling to because according to that, CONUS and OCONUS will take those activities under their radar, and going deeper into that, did they take any kind of Mediclaim like health insurance or baggage insurance or were there any sabbatical leaves were involved as well, these all will be coming under the attributes of a trip entity.

Timeline

Description automatically generated

#### Advantages of Entity-Relationship Diagramming (EERD):

EERD provides a clear understanding of the system's entities and relationships, facilitating an easier transition to database design.

Besides, it also enables the identification of key entities, attributes, and relationships, helping to avoid data redundancy and improve data integrity.

#### Disadvantages of Entity-Relationship Diagramming (EERD):

However, there are disadvantages of EERD too. For example, the complexity of our diagram increases with the growth of the diagram, making it challenging to navigate and comprehend.

Not only that, determining the optimal relationships between entities and attributes can be challenging and time-consuming.

### SQL Query Analysis:

#### Summary:

The next phase involved creating a relational database schema and formulating SQL queries for data analysis.

#### Explanation:

After finishing the EERD diagram, we went into the second phrase. Now we have raised multiple business questions. One of the questions we came up with here is: What are the total costs for individual expenses and overall cost spent

by each Traveler?

To answer this business question, we used SQL query analysis method. Below is the query we have written to answer the above question. This question makes use of 5 tables which are joined together and then the sum of each expense has been aggregated for each Trip ID and Name. We also made use of nested query here to create another column with the total of all the individual expenses.

#### Example of SQL Query:

Query: Select a.\*, sum ( Total\_Meal\_Cost + Total\_Insurance\_Cost +Total\_Lodging\_Cost + Total\_Incidentals\_Cost ) as Overall\_Cost

FROM (SELECT T.Trip\_ID,T.Name, sum (case when M.Total\_cost then M.Total\_cost else 0 end) as Total\_Meal\_Cost, sum (case when I.Total\_cost then I.Total\_cost else 0 end) as Total\_Insurance\_Cost, sum (case when L.Total\_cost then L.Total\_cost else 0 end ) as Total\_Lodging\_Cost, sum (case when IC.Total\_cost then IC.Total\_cost else 0 end ) as Total\_Incidentals\_Cost FROM Traveller as T JOIN Meals M ON T.Trip\_ID=M.Trip\_ID

Left JOIN Insurance as I ON T.Trip\_ID=I.Trip\_ID

Left JOIN Lodging as L ON T.Trip\_id = L.Trip\_ID

Left JOIN Incidentals as IC ON T.Trip\_ID = IC.Trip\_ID Group by 1,2) as a Group by 1;

Output: Graphical user interface, application

Description automatically generated

#### Advantage of SQL Query:

After our study, we agreed on SQL is a powerful tool for data manipulation and retrieval. It allows for complex analysis of large amounts of data. In addition to that, it also offers flexibility in terms of joining tables and aggregating data.

#### Disadvantage of SQL Query:

SQL requires a solid understanding of the database structure to write efficient queries. It is very tedious in a way too. In conclusion, SQL can be difficult to perform calculations across multiple tables.

### Graph Database (Neo4j):

#### Summary:

The database created previously was converted into csv files for each table and then uploaded into the IMPORT folder of Neo4j database. The files were then further loaded to create nodes and answer business questions using Cypher queries.

#### Example of Neo4j:

Business Question: What was the total amount that was utilized by each employee towards their meals?

The two tables used to answer these queries are:

1. Traveler: Contains the details about the employees which have taken at least 1 trip;

2. Meals: The table lists the cost spend by each Traveler on their meals.

Query:

**// Load Traveller nodes**

LOAD CSV WITH HEADERS FROM 'file:///Traveller.csv' AS row

MERGE (: Traveller {TripId: row.Trip\_ID, Name: row.Name});

**// Load Meal nodes**

LOAD CSV WITH HEADERS FROM 'file:///Meals.csv' AS row

MERGE (: Meal {TripId: row.Trip\_ID, MealType: row.Meal\_Type, TotalCost: row.Total\_Cost});

**// Create HAS\_MEAL relationships**

MATCH (t:Traveller), (m:Meal)

WHERE t.TripId = m.TripId

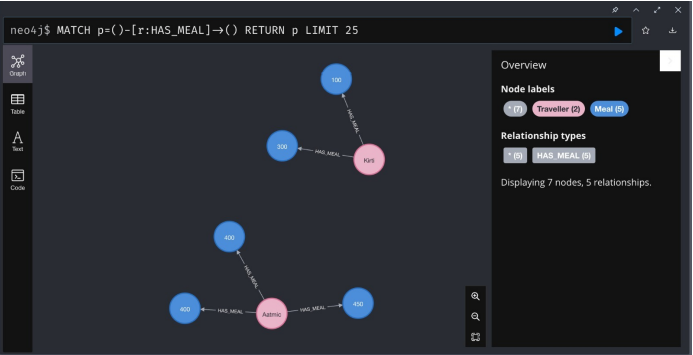
MERGE (t)- [: HAS\_MEAL]->(m);

**// Calculate total cost by trip**

MATCH (t:Traveller)- [: HAS\_MEAL]->(m:Meal)

RETURN m.TripId, SUM (toFloat(m.TotalCost)) AS TotalCostForAllMeals

ORDER BY m.TripId;

Output:

#### Advantage of Neo4j:

Based on previous output of the example, it is clear that graph databases excel in illustrating connections between data points, making them ideal for analyzing relational data.

Generally, Neo4j's Cypher query language is expressive and efficient, allowing for sophisticated querying and data manipulation.

#### Disadvantage of Neo4j:

To prepare the work, the database was converted into CSV files, uploaded into a Neo4j graph database, to be analyzed. So, we assume that importing data and creating nodes can be complex and require additional configurations.

Secondly, graph databases may not perform as well as relational databases when dealing with operations that aren't inherently graph-based.

# Conclusion:

During our journey in the lecture of Information Modeling, we learnt a comprehensive overview of the tools and methodologies employed in the creation and management of the information system.

After our case study, we found out that each tool carries its strengths and weaknesses and is suitable for different scenarios. For an organization to make an informed decision regarding the implementation of an information system, these factors should be carefully considered in the context of the organization's specific needs and capabilities.