**Travel Expense Reimbursement – Memo**

***Conceptual Modeling***

Our modeling is about the university’s rules and regulations on providing the reimbursement for the different travels being done by individuals. Below mentioned is the summary of how our EERD diagram will look like:

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. Travel 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 one.

These can be travelers, supervisors, budget agencies and university financial services, like who will be the people going to travel who will arrange 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 approve.

After this, it comes to what type of reimbursement can be an individual file, is 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.

Challenges:

* The biggest challenge we faced while trying to find the elements for the Entity relationship diagram were about figuring out the different relationships between the entities and their attributes. In order to tackle this problem, we took the team's recommendation for the optimal set of conditions was taken into consideration.
* As the diagram grew, it became more challenging to navigate and put it into a manner that the audience could understand, which presented another challenge. We were able to overcome this problem by having one team member create the diagram, another assists him with each component, and a third monitor the entire process.

***Relational Database Design and Queries***

Part 1: EER Model Revisions:

* Removed the [Sales tax] attribute from the [Lodging] entity:

The [Sales tax] attribute was initially included in the [Lodging] entity to represent the sales tax applied to lodging expenses. However, we found the sales tax is not an important attribute here. Therefore, we removed the [Sales tax] attribute from the [Lodging] entity.

* Changed the primary key of the reimbursement table to [Reimbursement Code] instead of [Trip code]:

The initial primary key of the reimbursement table was [Trip code]. However, we decided to change it to [Reimbursement Code] because the reimbursement code is a unique identifier for each reimbursement request. This change ensures that each row in the reimbursement table is uniquely identified by the reimbursement code.

* Created a fact table called Travel Expenses to calculate travel expenses using SQL: We created a [Travel Expenses] fact table to calculate the total amount of travel expenses incurred during a trip. This table will be populated using SQL queries that aggregate data from the [Lodging], [Transportation], [Insurance]and [Meals] entities. By creating a separate fact table for travel expenses, we can easily calculate and analyze the total travel expenses for each trip and employee.

* Removed the external and internal departments and created a new [Department] entity to represent Financial Services and Reimbursement agency:

After consulting with our professor, we decided that combining people and departments into one entity could cause confusion. Therefore, we removed the external and internal departments and created a new department entity to represent Financial Services and Reimbursement agency. We also added some attributes to this new entity. By doing this, we have simplified the structure of our EER model and made it easier to understand.

Part 2: Rationale and justification for your database schema and constraints (you may include a relational schema diagram)

“People” is an entity that has attributes like Emp\_ID, Name, Address, Sex, Dept\_Id. The Emp\_Id is the primary key for the for the entity as each person would have a unique id for identification. “Trip” is an entity that has attribute slike Vendor\_Id, Trip\_Id, Purpose of trip, Area\_Code,

Reimbursement\_ rate. The primary key for the Trip entity would be Trip\_Id since every trip made will have a unique identification to find the details about it. Also, we added Vendor\_Id as a foreign key to identify the which vendor had arranged the trip.

“Travel Arranger” entity has attributes like Vendor\_Id, Trip\_Id, Emp\_Id, vendor \_name, Traveler\_name. This entity has primary key as Vendor\_Id to identify different vendors and it has foreign key Trip\_Id and Emp\_Id to identify which person has arranged what type of trip with the travel arranger.

“Supervisor” entity has entities like Approval status, Trip\_Id, Supervisor Name, Supervisor Id. Here Sup\_Id is a primary key to identify supervisor uniquely also we have added foreign key Trip\_Id to know about the supervisor assigned to a certain trip.

“Department” entity has primary key as Dept\_Id which identifies every department uniquely. Also, it has other attributes like Emp\_Id, Dept\_Name, Dept\_Type. We have Ep\_Id as a foreign key to show which employee is part of which department.

“Reimbursement Agency” entity has attributes like Agency\_Id, Dept\_Id, Agency\_Type, Reimb\_Approval\_Status. It has primary key as Agency\_id to identify agencies uniquely and foreign key as Dep\_Id to find out which dept is handling approvals from certain agency. “Reimbursement Type” entity keeps record of different type of reimbursements requested by people. Hence, containing the Trip\_Id and Emp\_Id as foreign keys which help in identifying the person who has issued for reimbursement request for which trip.

A diagram of a company

Description automatically generated

Part 3: Brief explanation of the SQL queries

• Business Question: What are the total costs for individual expenses and overall cost spent by each Traveler?

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.

*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:*



• Business Question: How many trips have been done by each vendor for employees and which department do these employees belong to?

To execute this query, we joined the table “People” giving out details on the Employees with the Vendor table which consists of information on the different trips completed by each traveler using the different vendors. The total count of the trips was then calculated using the function Count and aggregated them using Group By function. Lastly, using the Employee id, the employees were matched to their respective departments and the names were listed out.

*Query:*

Select t.Vendor, p.Name, d.Name, COUNT(t.Employee\_ID) as Total\_Trip

FROM Travel\_Arranger AS t JOIN People AS p

on t.Employee\_ID = p.Employee\_ID JOIN Department AS d on t.Employee\_ID = d.Employee\_ID

group by 1,2,3; *Output:*

A screenshot of a computer

Description automatically generated

• Business Question: Which Supervisor has given the maximum approvals and to which employee?

Each trip being taken by a traveler needs to be approved by their supervisor. Below query gives out the details of the total number of trips of employees which were approved by the supervisor. Three tables have been joined here which are Supervisor, Employee details and Trip details table to get the expected output. We have used a Where filter to only count the trips which have been approved.

*Query:*

Select p.Name as Supervisor\_Name, t.Name as Employee,Count(Approval\_Status) as Total\_Approval

FROM Supervisor as s

JOIN Traveller as t ON s.Trip\_ID = t.Trip\_ID

JOIN PEOPLE as p ON p.Employee\_ID = s.Supervisor\_ID

Where Approval\_Status = "Yes" Group by 1,2; *Output:*

A screenshot of a phone

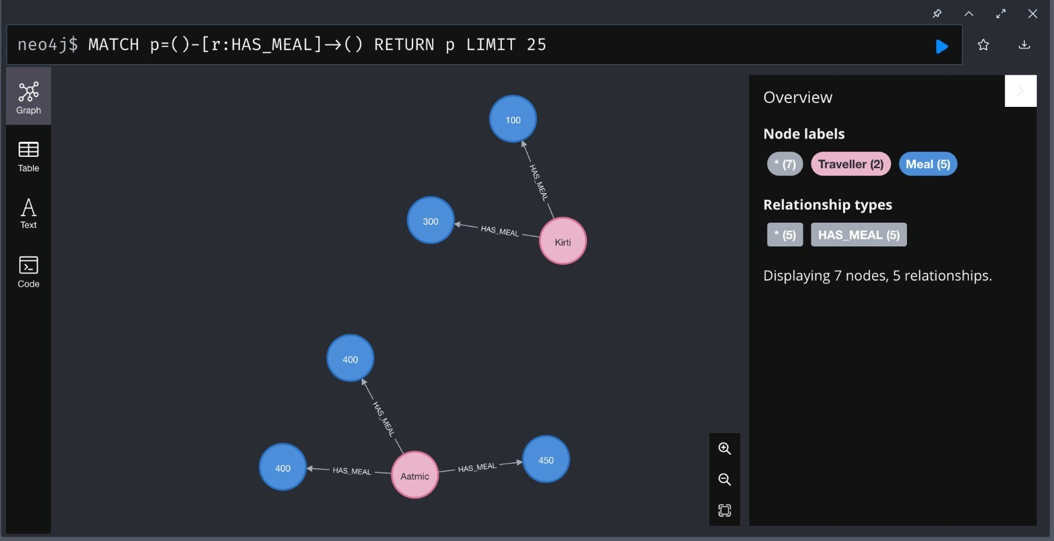
Description automatically generated

***Neo4j Queries***

The database created in Phase 2 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:  
  
Question 1: 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 out 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* : The above output lists the total cost spent by each employee for their respective meals I.e. Breakfast/Lunch/Dinner.

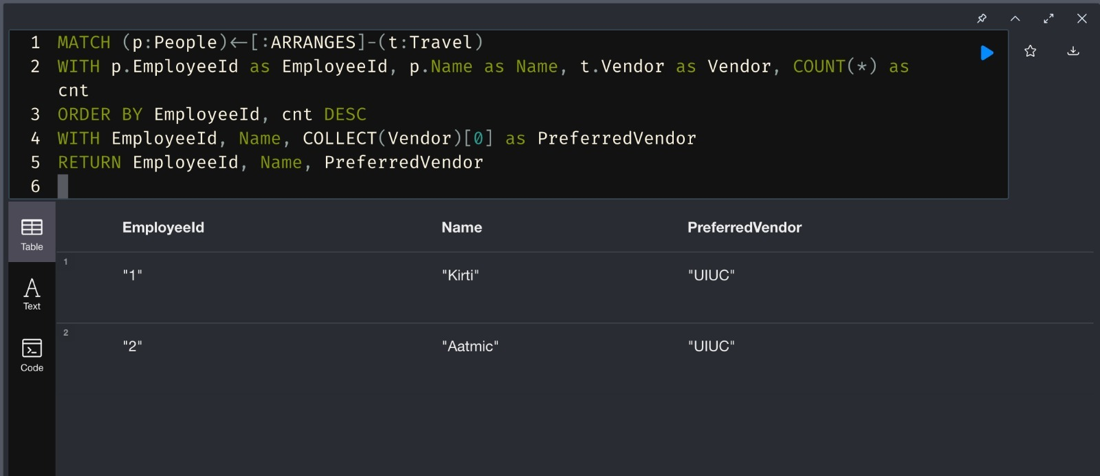
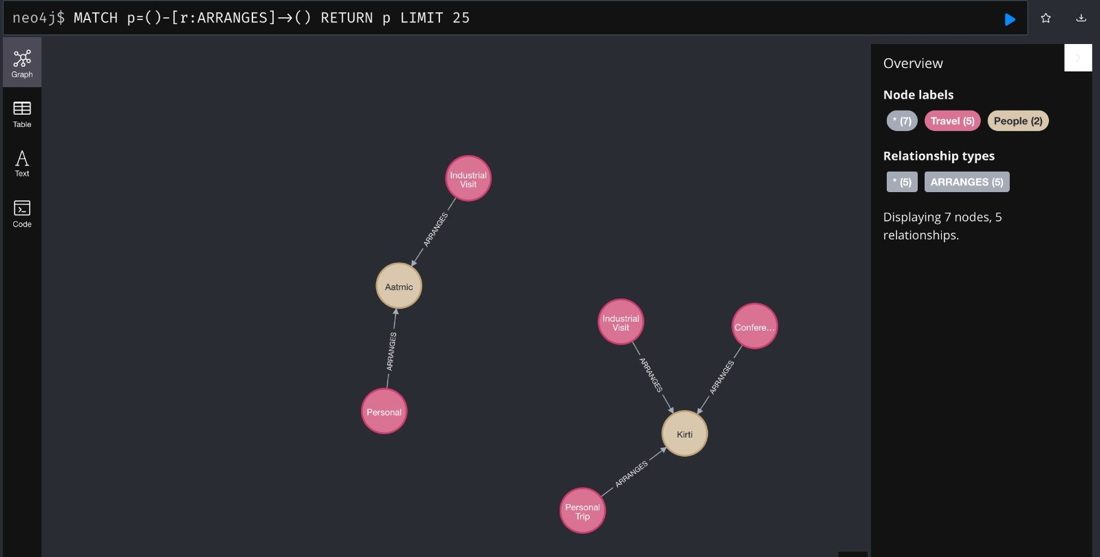
page1image49956688

Question 2: What was the purpose of each Traveler to make a trip?   
  
The two tables used to answer these queries are:  
1. Traveler\_Arranger: Contains the travel details of the employees which have taken at least 1 trip   
2. Employee: The table lists out the details of all the employees

*Query*:  
//Loading the Employee Detail File  
LOAD CSV with headers FROM 'file:///People.csv' as row  
MERGE(:People {EmployeeId: row.Employee\_ID, Name: row.Name, Sex:row.Sex, Address:row.Address}); //Loading the Travel Details file  
LOAD CSV with headers FROM 'file:///Travel\_arranger.csv' as row  
MERGE(t:Travel {EmployeeId: row.Employee\_ID, TripID: row.Trip\_ID, Vendor:row.Vendor, Purpose:row.Purpose}); //Displaying the Purpose of Trips for Each Employee  
MATCH (p:People), (t:Travel)  
WHERE p.EmployeeId = t.EmployeeId  
MERGE (t)-[:ARRANGES{Vendor: t.Vendor}]->(p);

Question 3: Based on the data, what will be the preferred vendor for the travelers if they make the next trip?

We are making use of Neo4j Inferencing here to collect the total count of each traveler for each Vendors. This information can be utilized to guess the preferred Vendor for the next trip of the travelers.



Challenges:

1. One of the difficulties arose when creating nodes since we were unable to display the properties on the graph mode, resulting in the output being returned in the form of a table.
2. Another issue with importing external libraries like apoc was that the Neo4j database will stop functioning if the configuration file was changed.
3. Due to the insufficient amount of data, we were forced to switch to our favored vendor while trying to infer each traveler's chosen meal.