Conceptual Model Proyect 1 - Carpooling App

Course: Bases de Datos I - Tecnológico de Costa Rica

Members:

Natalia María Acuña Alfaro

Carlos Mario Castillo Mena

Luis Felipe Benavides Solórzano

Notes for independant entities:

- ***Parameter:*** *created to fulfill the need for a parameter that tracks daily trips per institution.*
- ***AuditLog:*** *It is going to be an experiment approved by our professor. We will associate AuditLog with each and every class.*
- ***LogBook:*** *As asked by the professor, we will model the logbook or also known as the "audit table", which will register any changes that are made to the entities.*

Notes on DailyReport:
Reports are directly linked to the trips they summarize through a dedicated relationship rather than relying on loosely structured data fields. Each report-trip connection records additional details including the date and time of the trip and its sequential position within the report. Furthermore, institutions maintain their own separate daily reports with dedicated audit logs, enabling monitoring across the entire organizational structure.

Notes on rankings and reports:

- *The top 5 drivers with the most trips can be obtained by filtering trips where status = 'completed'.*
- *The top 5 most visited stops by date range can be calculated using the departureDateTime attribute from Trip and the has relationship with Stop, counting how often each stop appears.*
- *The top 5 most active users are determined by the number of trips where a user participates, either as a driver with relationship or as a passenger with relationship join).*
- *The full list of trips, including information about drivers, passengers, and routes with relationship stop, is already covered through existing relationships.*
- *The average cost per trip can be calculated from the cost attribute in the has relationship between Trip and Stop.*
- *The number of new users can be obtained using the registrationDate attribute in User.*
- *Since all this information can be queried using the current entities and relationships, they are considered functional requirements that are already supported by the model, without the need for new entities.*

Domain of name in AccessStatus:
1. *AccessGranted*
2. *AccessPending*
3. *AccessRemoved*

Domain of status in UserStatus:
1. *Verified*
2. *NotVerified*

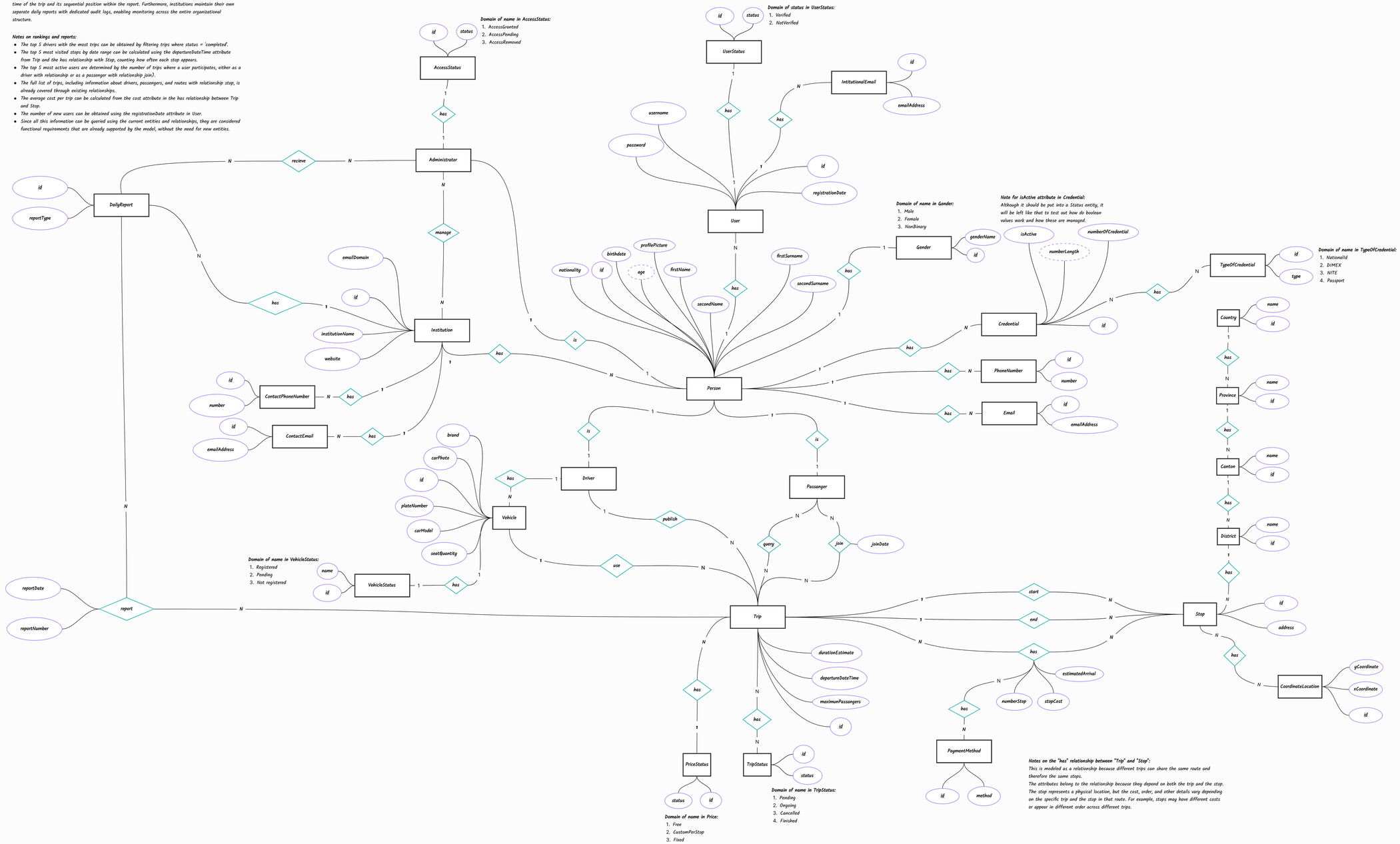
Domain of name in Gender:
1. *Male*
2. *Female*
3. *NonBinary*

Note for isActive attribute in Credential:
Although it should be put into a Status entity, it will be left like that to test out how do boolean values work and how these are managed.

Domain of name in TypeOfCredential:
1. *NationalId*
2. *DIMEX*
3. *NITE*
4. *Passport*

Domain of name in VehicleStatus:
1. *Registered*
2. *Pending*
3. *Not registered*

Domain of name in TripStatus:
1. *Pending*
2. *Ongoing*
3. *Cancelled*
4. *Finished*

Domain of name in Price:
1. *Free*
2. *CustomPerStop*
3. *Fixed*

Notes on the "has" relationship between "Trip" and "Stop":
This is modeled as a relationship because different trips can share the same route and therefore the same stops.
The attributes belong to the relationship because they depend on both the trip and the stop. The stop represents a physical location, but the cost, order, and other details vary depending on the specific trip and the stop in that route. For example, stops may have different costs or appear in different order across different trips.