

INTERCITy BUSES



oNUR UYAN

MERT ÖZDEMİR

BEŞİRE NUR ÇELEBİ

Scenario

Our company runs buses inter cities. The routes should be known before the trip. for this reason, we assign buses and drivers to the roads in advance. Each bus has a maintenance story. On each trip we assign a stewardess. On each road we have predefined stops. You want to keep track of schedules and real time departure and arrival of the buses.

İntercity Bus Travel

InterCitys bus travel is the management system designed for Bus travel companies. It is designed to manage bus-related information such as tickets and ticket prices, stations, routes and record keeping of customers, hosts and drivers.

The system is designed to record all the transactions and trips made by passengers and company itself of daily sales, recognize all registered drivers, buses and stewards, stations of each trip, available trips price and date, company could see the arranged departure date, time and delayed time of each trip.

Database Design

The task of designing a database involves designing conceptual schema, logical schema and design of access. The need of the company plays a central role in the design process. The design of the database environment should meet the needs of the enterprise.

We proceeded through the following design phases to build the database system for our hypothetical client:

The initial phase of database design is to characterize fully the data needs of our prospective client, we were able to come up with a specification of user requirements. A textual description of use requirements is briefly outlined below:

* The company keeps track of trips, routes, buses, seats, stewards, drivers, passengers, tickets, time and stations.
* Passengers are identified by an SSN. First and last name, and phone number must be recorded
* Drivers are identified by the Driver id. For each driver the first and last name and license number must be recorded.
* Each host identified by Host id. First and last name and phone number must be recorded.
* For each bus, the bus type, license plate number, last maintenance date and make of the bus should be recorded.
* Each bus has seats depending on the type of bus. Every seat should have a number, for making assignments to a specific passenger.
* Every trip needs a crew, so the crew needs to keep records of the bus, driver and host that assigned to a trip.
* Each route is identified by Route id and should keep records of the Route name and Cycle date of the route.
* Every trip has a ticket, predefined price and seat. Each ticket buyer must be known by the company. Also, the company wants to keep records of which tickets is sold, and which one isn’t.
* The company arranges trips, keeps records of crew, which route it will take, predefined date and time.
* The company should keep a record of the stations, the name of them and arrival date of the bus.
* The company wants to track the time when a bus arrives to a station and know the if there is any delay occur.

A diagram of a passenger transportation system

Description automatically generatedERD

Problems We Encounter

* When we are creating the station entity, we realize that we need an entirely different entity for time because there could be some delays occurring. So, we create an entity for time to calculate the delayed time
* When we create a ticket entity, we add a seat attribute to it but then we realize that we don’t know which bus has how many seats in it. So, we add a seat entity then create an entity between bus and seat. By doing that we can find which seat is sold to WHO and how many seats a bus has.
* In the beginning we only had route entity. We realized that was insufficient because we were not able to see the total number of trips. So, we create an entity called trip. Then we made a relationship between the trip to the station.
* We tried to include the host, driver and bus under the entity named Trip. Then, since this would create a many-to-many relationship, we added an entity called Crew. After that, we centralized everything in Crew to avoid the many-to-many relationship.
* We added an entity name is transaction because of a problem. We encountered a problem when we wanted to see who is the buyer of the ticket and who is the owner. Now we can see the buyer in “Transaction” and owner in “Ticket”.

QUERIES

TABLES