Zachary Florez

Mihir Yadav

Gerry Guardiola

Rubin Yang

Final Design Document

i:

For our diagram we wanted to be able to simplify our design by reducing each department table down to one single department with a key determining the service. This is because our queries did not require a specific table for each department for getting services for calculating transaction costs and it still follows 3rd normal form's definition. In terms of general design, the customer table holds the start and end dates of their license/ids to keep track of expiration dates. They also hold foreign keys for the department number for the type of service they received and a transaction id to connect it to the weak entity set of the transaction table. They also have a unique id and information like their full names.

For transactions it simply uses the department id to determine a cost for each transaction. Then it holds a time for the date a transaction takes place. The next connection is that between the customer, appointment, and employee tables. A customer books an appointment and an employee is assigned to that appointment. The appointment is a weak entity set that connects the customer to the employee by containing their unique id's. It also holds the status of the appointment to determine if a service was fulfilled or not. The employee table is like a simpler customer table with a unique id for each employee and information for their full name. The only other attribute it contains is the department id which tells us the department the employee works for. The final table is the aforementioned department table which only holds an id for the department and then a name for that department.

Restraints present for our schema is that there is a limit on the number of licenses and state ids being limited to one per customer at a time and vehicles can only have one registration per vehicle at a time as well.

ii:

customer

cID	fName	lName	DID	TID	startDate	endDate

cID (Primary key): numeric(5) 5-digit number unique to each customer.

fName (attribute): varchar(25) first name.

lName (attribute): varchar(25) last name.

DID (foreign key): int Department id used to track type of service received.

TID (foreign key): int Transaction id for the specific service.

startDate (attribute): datetime Date license was started.

endDate (attribute): datetime Date license expires.

transaction

tID	DID	time	amount

tID (primary key): numeric(7) 7-digit unique integer for each transaction.

DID (foreign key): int 1-digit integer for each of the 4 departments.

time (attribute): datetime for the day service was received.

amount (attribute): numeric(7) integer cost of transaction depending on service.

appointment

aID	CID	EID	Status

aID (primary key): int integer representing the appointment made by a customer.

CID (foreign key): int 5-digit integer of customer who made appointment.

EID (foreign key): int 5-digit integer of Employee assigned to the appointment.

status (attribute): int 1-digit integer represents if service was successful (0 for false 1 for true).

employee

eID	fName	lName	DID

eID (primary key): numeric(5) 5-digit unique integer for employee.

fName (attribute): varchar(25) for the first name of employee.

lName (attribute): varchar(25) for the last name of employee.

DID (foreign key): int(1) 1-digit integer representing the department the employee works for.

department

dID	dName

dID (primary key): numeric(1) 1-digit integer representing each of the 4 departments (1: Permit, 2: License, 3: Registration, 4: State ID).

dName (attribute): varchar(25) name of the department.

iii:

Transaction = {tID->time, DID->amount}

Primary key determines the time the transaction takes place, the foreign key DID is the department id which determines the cost of a specific transaction. No transitive dependency is present, so table follows 3NF.

Customer = {cID->fName lName, startDate DID->endDate}

Customer id determines the first name and last name. While the issue date (startDate) and the department id determines the expiration date since the type of services expire at different times. Again, no transitive dependencies are present.

Appointment = {aID->status}

Appointment id determines if the service was completed. Only dependency present so no transitive dependencies.

Employee = {eID-> fName lName}

Employee id determines the employee name. No transitive dependencies in relation since dependents are not determinates anything.

Department = {dID->dName}

Department id determines the department name. No transitive dependencies are present because only one determinant and dependent in relation.

As seen, this adheres to 3rd Normal Form since no Transitive dependencies are present in any of the relations.

iv:

For our fourth query we ask for a single digit integer input to represent the department id (1-4) and it returns the full names of those who have received a permit/license/registration/state id from that department. This is useful to determine the types and number of licenses a customer has received. This can be useful to gather information whether someone needs to check which customer has a valid registration for a vehicle or whichever license is needed. Can also be used for further data gathering/tracking which can be useful when a large number of services are performed during a large time frame.