CSC423 Project Case Study: SuperMaids Cleaning Company

Team 9:
Zubaer Chowdhury
Oscar Arana
Justin Prince

Part 2:

For Part 2 of the project, we derived relations from the conceptual model, validated through normalization, checked against user transactions, defined integrity constraints and generated the E-R diagram at the logical level.

Develop a logical data model based on the following requirements:
a. Derive relations from the conceptual model.
Ans a: {PK} = Primary Key, (CK) = Candidate Key, {FK} = Foreign Key

Entities:

Employee (**staffNo**, staffpallFName, staffLName, staffAddress, staffSalary, **staffTelNo**)
Primary Key: staffNo
Candidate Key: staffTelNo

Client (**clientNo**, clientFName, clientLName, clientAddress, **clientTelNo**)
Primary Key: clientNo
Candidate Key: clientTelNo

Request (**requestID**, startDate, startTime, duration, comments, **clientNo**)
Primary Key: requestID
Foreign Key: clientNo references Client(clientNo)

Equipment (**equipmentID**, description, usage, cost)
Primary Key: equipmentID

Relationships

RequestEquipmentAssignment (**requestID**, **equipmentID**, quantity)
Foreign Key:
requestID references Request(requestID)
equipmentID references Equipment(equipmentID)

EmployeeRequestAssignment (**staffNo**, **requestID**, assignmentDate)
Foreign Key:
staffNo references Employee(staffNo)
requestID references Request(requestID)

Client to Request (One-to-Many):
In the Request table, clientNo will be a FK referencing to the Client Table

Request to Equipment (Many-to-Many):
A separate table (RequestEquipmentAssignment) with FKs referencing both Request and Equipment tables

Employee to Request (Many-to-Many):
Another separate table (EmployeeRequestAssignment) with FKs referencing both Employee and Request tables

b. Validate the logical model using normalization to 3NF.
Ans b:
<u>First Normal Form (1NF)</u>:
All tables seem to have atomic values so no repeating groups or arrays

<u>Second Normal Form (2NF)</u>:
All non-key attributes are functionally dependent on the entire primary key in their respective tables
No partial dependencies are apparent

<u>Third Normal Form (3NF)</u>:
Check for transitive dependencies

<u>Checking for Transitive Dependencies</u>:

- Employee Table:
  No apparent transitive dependencies; all attributes depend directly on the primary key

- Client Table:
  Similar to the Employee table, all attributes depend directly on the primary key

- Request Table:
  startDate, startTime, duration & comments are attributes directly related to the

request and do not cause transitive dependencies

- Equipment Table:
  The attributes (description, usage, cost) are directly related to the equipment and do not cause transitive dependencies.

- RequestEquipmentAssignment Table:
  Contains only foreign keys and quantity so no transitive dependencies exist.

- EmployeeRequestAssignment Table:
  Only contains foreign keys and an assignment date so no transitive dependencies exist.

Since no transitive dependencies exist in all the tables, the logical model is valid up to 3NF

c. Validate the logical model against user transactions.
Ans c: User transactions:
Creating a new employee record:
The Employee table structure allows for the addition of a new record without issues

Adding a new client:
Similar to the creating employee scenario, the Client table allows for the addition of a new record.

Registering a cleaning service request for a client:
A new entry in the Request table linked to an existing client is feasible as it complies with the relationship constraints

Assigning employees to fulfill a cleaning service request:
The EmployeeRequestAssignment table allows the assignment of multiple employees to a service request, adhering to the many-to-many relationship between Employee and Request.

Updating employee details (e.g. salary change):
The Employee table can be updated without affecting other tables, preserving data integrity.

Updating assignments for employees:

The EmployeeRequestAssignment allows for the requestID to be modified so employees can be switched for specific service requests.

Updating client information:
Client information is allowed as all foreign keys are set to cascade on update.

Updating equipment/Equipment Maintenance:
The Equipment table allows modification on the description, usage, and cost while ensuring associated service requests remain accurate

Updating requests:
The request table allows for modification as the requestID can be modified to change the service required. The results will be reflected in EmployeeRequestAssignment as it will cascade on update

Deleting a client who no longer uses the service:
Deletion from the Client table is allowed as the Request table will cascade in the event of the client being deleted. Since the request table is the only dependency, it works fine.

Deleting an employee (Fired/Quit):
Deletion from the Employee table is allowed as EmployeeRequestAssignment cascades when an employee is deleted.

Deleting equipment from the system:
Deleting from the Equipment table is allowed as RequestEquipmentAssignment will accommodate for this loss by setting the equipment in the assignment to NULL.

Deleting a request from the system:
Deleting from the Request table is allowed as RequestEquipmentAssignment will delete all related records when a request is deleted.

Since the logical model allows for these basic user transactions, the logical model is valid against user transactions.

d. Define integrity constraints:

i. Primary key constraints.
Ans i:
Employee: staffNo
Client: clientNo
Request: requestID

Equipment: equipmentID

ii. Referential integrity/Foreign key constraints.
Ans ii:

Entities:

Employee (**staffNo**, staffFName, staffLName, staffAddress, staffSalary, **staffTelNo**)
Primary Key: staffNo
Candidate Key: staffTelNo

Client (**clientNo**, clientFName, clientLName, clientAddress, **clientTelNo**)
Primary Key: clientNo
Candidate Key: clientTelNo

Request (**requestID**, startDate, startTime, duration, comments, **clientNo**)
Primary Key: requestID
Foreign Key: clientNo references Client(clientNo) ON UPDATE CASCADE ON DELETE CASCADE

Equipment (**equipmentID**, description, usage, cost)
Primary Key: equipmentID

Relationships:

To handle Many-to-Many relationship between Request and Equipment
RequestEquipmentAssignment (**requestID**, **equipmentID**, quantity)
Foreign Key:
requestID references Request(requestID) ON UPDATE CASCADE ON DELETE CASCADE
equipmentID references Equipment(equipmentID) ON UPDATE CASCADE ON DELETE SET NULL

To handle Many-to-Many relationship between Employee and Request:
EmployeeRequestAssignment (**staffNo**, **requestID**, assignmentDate)
Foreign Key:
staffNo references Employee(staffNo) ON UPDATE CASCADE ON DELETE SET NULL

requestID references Request(requestID) ON UPDATE CASCADE ON DELETE CASCADE

iii. Alternate key constraints (if any).
Ans iii: Employee: staffTelNo
Assumption: each phone number is unique

Client: clientTelNo
Assumption: each phone number is unique

iv. Required data.
Ans iv: All Primary Key attributes of each entity must not be null (staffNo, clientNo, requestID, equipmentID)
All attributes, except Foreign Keys, must not be null.

v. Attribute domain constraints.
Ans v:
Telephone numbers (staffTelNo & clientTelNo) must match the standard phone number format of the country. Cleaning service is only provided within the same country.
Assumption: Country is USA so must follow US phone format)

Salaries (staffSalary) must be a non-negative integer.
Assumption: Salaries are always integer numbers

Dates (startDate) must be in the correct date format and cannot be earlier than current date.
Assumption: Country is USA so must follow US date format (mm/dd/yyyy)

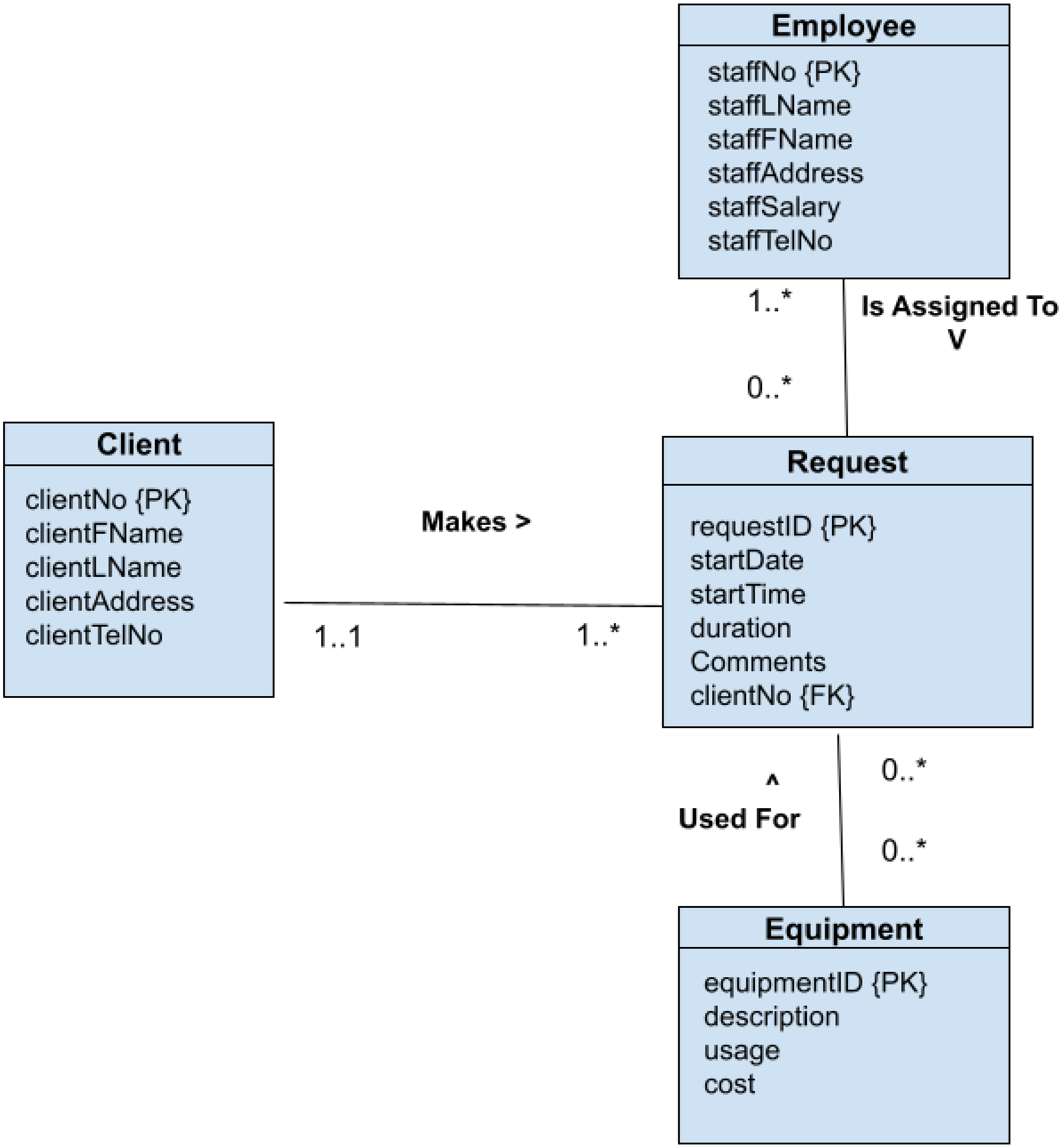
Request's start time (startTime) must be in hh:mm:ss tt (24-hour with leading zeros) format with Eastern Standard Time (EST) implied.
Assumption: Country is the USA so we must follow the US time convention.
The company is situated in Miami and records all time in the local EST timezone.

Equipment's cost (cost) must be a non-negative integer.
Assumption: Nobody sponsored or paid the company to use the equipment so There's an equipment cost to the company.

vi. General constraints (if any).
Ans vi: A client must have at least one request

e. Generate the E-R diagram for the logical level (contains FKs as attributes).
Ans e:

Logical Data Model E-R Diagram


Employee
staffNo {PK}
staffLName
staffFName
staffAddress
staffSalary
staffTelNo
1..*
Is Assigned To
V
0..*
Client
clientNo {PK}
clientFName
clientLName
clientAddress
clientTelNo
Makes >
1..1
1..*
Request
requestID {PK}
startDate
startTime
duration
Comments
clientNo {FK}
^
Used For
0..*
0..*
Equipment
equipmentID {PK}
description
usage
cost