Database Systems Project Part 2: Logical Design
By: Julian Cantillo, Spencer Terwilliger, Andrew Shatsky

2a: List of Relations

Client Relation

clientNo (primary key)	firstName	LastName	address	PhoneNumber

Requirement Relation

RequirmentID (primary key)	ClientNo (foreign key)	StartDate	StartTime	Duration	Comments

Equipment Relation

EquipmentID (primary key)	Description	usage	Cost

Employee Relation

StaffNo (primary key)	FirstName	LastName	Address	Salary	Phone Number

Equipment_Requirment Relation

EquipmentID (foreign key)(Primary key 1/2)	RequirmentID (foreign key) (primary key 2/2)	Quantity

Employee_Requirement Relation

StaffNo(foreign key) (Primary Key 1/2)	RequirmentID (foreign key) (Primary Key 2/2)

Database Systems Project Part 2: Logical Design
By: Julian Cantillo, Spencer Terwilliger, Andrew Shatsky

2b: Normalization

Functional Dependencies

1. Client
    a. clientNumber → firstName, lastName, address, phoneNumber
2. Requirement
    a. requirementID → clientNo, startDate, startTime, duration, comments
3. Equipment
    a. equipmentID → description, usage, cost
4. Employee
    a. staffNo → firstName, lastName, address, salary
5. Equipment_Requirement
    a. {requirementID,equipmentID} → quantity
    b. requirementID → equipmentID (Partial Dependency)
6. Employee_Requirement
    a. requirementID → staffNo (Partial Dependency)

1NF

Since each row in each relation only contains a single value, the relations are in first normal form.

2NF

The Client, Requirement, Equipment, and Employee tables have no partial dependencies. For Equipment_Requirement and Employee_Requirement, The partial dependencies are acceptable since they make up the composite primary key. Since the table is also already in 1NF and has no problematic partial dependencies, it is also in Second Normal Form

3NF

Since there are no Transitive dependencies present, and the relations are in 2NF, the relations are also in Third Normal Form

Database Systems Project Part 2: Logical Design
By: Julian Cantillo, Spencer Terwilliger, Andrew Shatsky

2c: User Transaction Validation

List of User Transactions and Solutions

1. Add/View Client/s
    a. This works since only the Client Relation would need to be accessed to add all of the information required for a new client or view all clients
2. Add/View Cleaning Requirement/s
    a. This requires access to just the Requirement Relation
3. Add Equipment
    a. This requires access to just the Equipment Relation
4. Add Employee
    a. This requires access to just the Employee Relation
5. Assign Equipment to Requirement / View equipment allocation status
    a. Joining the Equipment Relation to the Requirement Relation through the Equipment_Requirement relation would allow these transactions to happen.
    b. Equipment (eq) and Equipment_Requirement (eqr) would be joined by eq.staffNo = eqr.staffNo.
    c. Requirement (r) and Equipment_Requirement (eqr) would be joined by eqr.requirementID = r.requirementID
6. Assign Employee to Requirement / View status of all employees
    a. Joining the Employee Relation to the Requirement Relation through the Employee_Requirement relation would allow these transactions to happen.
    b. Employee (em) and Employee_Requirement (emr) would be joined by em.staffNo = emr.staffNo.
    c. Requirement (r) and Employee_Requirement (emr) would be joined by emr.requirementID = r.requirementID
7. Delete Cleaning Requirement
    a. Deleting a cleaning requirement would need the Requirement, Employee_Requirement, and Equipment_Requirement relations to be joined.
    b. Requirement (r) and Employee_Requirement (emr) would be joined by emr.requirementID = r.requirementID
    c. Requirement (r) and Equipment_Requirement (eqr) would be joined by eqr.requirementID = r.requirementID
8. Delete Client
    a. Since deleting a client could lead to the deletion of one or more cleaning requirements, it requires the steps outlined for the deletion of a cleaning requirement (#7).
    b. Deleting a client would also need the Client and Requirement relations to be joined
    c. Client (c) and Requirement (r) would be joined by c.clientNo = r.clientNo

Database Systems Project Part 2: Logical Design
By: Julian Cantillo, Spencer Terwilliger, Andrew Shatsky

2d: Define Integrity Constraints

I. Primary key constraints:

- Client Relation
  - ClientNumber
- Requirments Relation
  - RequirmentsID
- Equipment Relation
  - EquipmentID
- Employee Relation
  - StaffNo

II. Foreign key constraints:

- Equipment_Requirment Relation
  - EquipmentID
- Equipment_Requirment Relation
  - RequirmentID
- Requirments Relation
  - clientNo
- Employee_Requirement Relation
  - staffNo
- Employee_Requirement Relation
  - RequirmentID

III. Alternate key constraints

- No alternate key constraints

IV. Required data

- Client Relation:
  - First_name
  - last_name
  - Address
  - telephone_number
- Equipment Relation:
  - Description
  - usage
  - Cost
- Employee Relation:
  - First_name
  - last_name
  - address
  - salary
  - telephone_number
- Requirement Relation:
  - start_date
  - start_time
  - duration

V. Attribute domain constraints

- Client: client number must be a unique identifier
- Client: first name, last name must be strings
- Employee: staff number must be a unique identifier
- Employee: first name, last name must be strings
- Equipment: description must be a string
- Equipment: equipment identifier must be a unique identifier
- Requirement: start date, start time, duration must be dates and times
- Requirement: comments must be strings
- Requirement: requirement identifier must be a unique identifier

VI. General constraints

- End time needs to be later then start time
- End date needs to be later then start date
- Duration must be positive
- A client is only allowed to have one active requirement at a time.
- An employee can only work on one requirement at a time.
- Equipment can only be used on one requirement at a time.

Database Systems Project Part 2: Logical Design
By: Julian Cantillo, Spencer Terwilliger, Andrew Shatsky

2e: Logical Level Entity-Relationship Diagram