TEAM MEMBERS:
dmmehta2, rsshives, ysonar, kpatil5

DBMS (Fall 2022) Project 1 – Team 15 Database Design Document (Discussions)

While designing the database for this project, multiple things were considered including constraints, normal-form, functional dependencies, redundancy etc. to ensure data consistency and accuracy.

Changes made to original ER diagram

Based on the application flow and re-evaluation of the requirements following changes were made:

1. Included new attributes, like start_date, role in EMPLOYEE, and min_wage, max_wage in SERVICE_CENTER
2. Updated the ISA of EMPLOYEE from ISA: Manager, Receptionist and Mechanic to ISA: Salried, HourlyPaid. This will make the database more extensible as adding new employee roles won't result in addition of new tables, instead the current database structure will work as expected.
3. Provides link between SERVICE_CENTER and SERVICE to set price, is updated to link between SERVICE_CENTER and aggregation of SERVICE and CAR, since the price varies for each serviced car.
4. Furthermore, the diagram was completed to include other components like CUSTOMER, CUSTOMER_CAR, SERVICE_EVENT.

Extended Functional Dependencies

1. **SERVICE_CENTER_ID** -> ADDRESS, TELEPHONE_NO, OPERATIONAL_STATUS, WEEKEND_WORKING, MIN_WAGE, MAX_WAGE
2. **CAR_ID** -> MANUFACTURER, YEAR
3. **CUSTOMER_ID, SERVICE_CENTER_ID** -> FIRST_NAME, LAST_NAME, EMAIL, ADDRESS, USERNAME, PASSWORD, STATUS
4. **EMPLOYEE_ID, SERVICE_CENTER_ID** -> NAME, ADDRESS, START_DATE, ROLE, EMAIL, PHONE, USERNAME, PASSWORD
5. **SERVICE_ID** -> ID, MECHANIC_ID, SERVICE_CENTER_ID, VIN, INVOICE_ID, START_TIME, END_TIME
6. **INVOICE_ID** -> INVOICE_STATUS, TOTAL_COST
7. **SERVICE_CENTER_ID, ID, CAR_ID** -> PRICE

TEAM MEMBERS:
dmmehta2, rsshives, ysonar, kpatil5

Constraints

(Note: This is not an extensive list)

UI Constraints

Add Service Center	Ensure phone number is 10 digits
Add Leave	Start Date is before End Date
Add Employee	Compensation is validated to be between min_wage and max_wage
Create Schedule	Date input, Sunday is disabled, Saturday is disabled for those service center which are closed on Saturday
Add Manager	Email is validated

Database key constraints

CUSTOMER Foreign Key (SERVICE_CENTER_ID) ON DELETE CASCADE	This foreign key constraint ensures that the service center is valid and exists in the system. Delete cascade deletes all customers when service center is deleted.
SERVICE_EVENT Foreign Key (VIN)	This foreign key ensures that the VIN exists in the CUSTOMER_CAR table. This doesn't have delete on cascade, because we don't want someone to delete a car when a service is already scheduled for that car.
LEAVE (START_DATE) NOT NULL	NOT NULL constraint ensures that the field START_DATE is not left blank when pushing data to the database table LEAVE.
CUSTOMER_CAR Foreign Key (CUSTOMER_ID, SERVICE_CENTER_ID) ON DELETE CASCADE	This ensures that the car that belongs to a valid existing customer in the database and when that customer is deleted, the car is also deleted.

TEAM MEMBERS:
dmmehta2, rsshives, ysonar, kpatil5

Database check constraints

CHECK (WEEKEND_WORKING IN (0, 1)	This ensures that the WEEKEND_WORKING column in SERVICE_CENTER has no value other than 0 (No) or 1 (Yes)
CHECK (EMAIL LIKE '%@%.%')	This check ensures the email id entered is of a valid email format and thus ensuring data integrity.
CHECK (ROLE IN (1, 2, 3)	This check ensures that ROLE in EMPLOYEE table is either 1(Manager), 2(Receptionist), 3(Mechanic)
CHECK (START_DATE < END_DATE)	This check ensures that the START_DATE is always before END_DATE in LEAVE table

Database Trigger constraints

VERIFY_WAGE_IS_IN_LIMIT	This trigger runs every time the hourly_wage of mechanic is updated to ensure its within min_wage and max_wage defined for that SERVICE_CENTER. If it's not, it is reverted to old wage.
UPDATE_SERVICE_CENTER_STATUS	This trigger runs every time a new mechanic is added to a service center. If the number of mechanics is more than or equal to 3, the service center status is automatically updated to operational.
CHECK_CUSTOMER_STATUS	This trigger runs when a new car is added for a customer. This changes the customer status to Active.
AUTO_APPROVE_LEAVE	This trigger runs when a new leave request is submitted by a mechanic. Depending on 2 conditions, firstly the mechanic doesn't have any services scheduled during leave period and secondly there are at least 3 mechanics during that leave period to keep the service center operational. Depending on the output of the above checks, the leave status is updated to accepted or rejected.
ON_SWAP_SLOT_APPROVE	This trigger runs when a swap slot request is approved. This updates the mechanic ids on the invoices and services exchanged by the mechanics.