

Patrick Nicholson
Alexander Sherrow
Ashwin Kannan

Data Types:

User

Attribute	Data Type	Nullable
username	String	Not null
password	String	Not null
firstName	String	Not null
lastName	String	Not null

Customer

Attribute	Data Type	Nullable
city	String	Not null
state	String	Not null
postalCode	String	Not null
phoneNumber	String	Not null
email	String	null

Business

Attribute	Data Type	Nullable
contactTitle	String	Not null
businessName	String	Not null
businessTaxID	String	Not null

Individual

Attribute	Data Type	Nullable
driverLicenseNumber	String	Not null

Employees

Attribute	Data Type	Nullable
-----------	-----------	----------

Owner

Attribute	Data Type	Nullable
privileges	List<String>	Not null

Salespeople

Attribute	Data Type	Nullable
privileges	List<String>	Not null

Inventory Clerks

Attribute	Data Type	Nullable
privileges	List<String>	Not null

Managers

Attribute	Data Type	Nullable
privileges	List<String>	Not null

Bought

Attribute	Data Type	Nullable
purchaseDate	Date	Not null

Sold

Attribute	Data Type	Nullable
sellDate	Date	Not null
salesPrice	Float	Not null

Vehicle Condition

Attribute	Data Type	Nullable
condition	String	Not null

Vehicle

Attribute	Data Type	Nullable
modelName	String	Not Null
modelYear	Int	Not Null
mileage	Int	Not Null
description	String	Not null
VIN	Int	Not null

Car Type

Attribute	Data Type	Nullable
chasisType	String	Not null

Manufacturer

Attribute	Data Type	Nullable
name	String	Not null

Fuel Type

Attribute	Data Type	Nullable
typeOfFuel	String	Not null

Car Color

Attribute	Data Type	Nullable
color	String	Not null

Parts History

Attribute	Data Type	Nullable
-----------	-----------	----------

Part Order

Attribute	Data Type	Nullable
purchaseOrderNumber	int	Not Null
phoneNumber	String	Not Null
vendorName	String	Not null

city	String	Not Null
Street	String	Not null
Postal Code	String	Not Null
State	String	Not null

Part

Attribute	Data Type	Nullable
partQuantity	int	Not Null
partNumber	int	Not Null
costPerPart	float	Not Null
description	String	Not Null

Status

Attribute	Data Type	Nullable
currentStatus	String	Not Null

Search Input and results

Task Decomp

Lock Types: Lookup Vehicle Table/Parts table.

Number of Locks: Single

Enabling Conditions: Returns additional information if user is owner or manager

Enable VIN search if user is salespeople or manager or owner

Frequency: High

Consistency (ACID): Vehicle Table/Parts table must complete all operations first

Subtasks: Mother Task is not needed. No decomposition needed.

Abstract Code

- Vehicle Search
 - User enters the following fields.
 - Vehicle Type(this.state.carType) (Drop Down Box)
 - Manufacturer(this.state.manufacturer) (Drop Down Box)
 - Model year(this.state.modelYear) (Input Box)

- Fuel Type(this.state.fuelType) (Drop Down Box)
 - Color (this.state.color) (Drop Down Box)
 - Keyword(this.state.keyword) (Input Box)
 - if user == Salespeople or Manager or Owner
 - VIN(this.state.VIN) (Input Box)
- Upon: Search Button Click:
 - if 1 field is entered:
 - if fields are valid data types in accordance to the above tables.
 - if returned results.length > 0:
 - Return query from vehicles table with the fields as where clauses using foreign keys when needed.
 - Display following information sorted by VIN in ascending order.
 - IF user not manager or owner and partsPending == false:
 - VIN
(this.state.searchResults[index].VIN)
 - Vehicle Type
(this.state.searchResults[index].vehicleType)
 - Model Year
(this.state.searchResults[index].modelYear)
 - Manufacturer
(this.state.searchResults[index].manufacturer)
 - Model
(this.state.searchResults[index].model)
 - Fuel Type
(this.state.searchResults[index].fuelType)
 - Colors
(this.state.searchResults[index].)
 - Mileage
(this.state.searchResults[index].mileage)
 - Sales Price
(this.state.searchResults[index].salesPrice)
 - IF user.privilege == manager or owner:
 - Return results + filterBy(sold, unsold, all) and vehicles with parts pending
 - Upon:

- Click of car row -> navigate to the car detail page of clicked car
- else returned results.length == 0:
 - Display error message:
 - “Sorry, it looks like we don’t have that in stock!”
 - Else fields are not valid data types in accordance to the above tables
 - Return error message telling users to correct invalid fields
 - Else no field is entered
 - Return error message instructing user to fill in at least 1 field

Business Logic Constraints

Users:

Username must be unique and assigned from admin.

Password is assigned by DBA.

Owner:

Has privileges that managers, salespeople and inventory clerks have.

Vehicles:

Vehicles Type and Manufacturer are updatable by DBA.

Vehicle year cannot exceed the current year + 1

The vehicle year has to be entered with the century digits

The fuel type has to be Gas, Diesel, Natural Gas, Hybrid, Plugin Hybrid, Battery, or Fuel Cell

The vehicle condition has to be Excellent, Very Good, Good, or Fair

Car color may be multiple colors.

A car can only be sold to Buzzcars once.

A car can only be sold by Buzzcars once

Should a buyer purchase several vehicles, each is handled as a separate transaction.

Vehicle search must return vehicles that only match all criteria.

If no vehicle matches the criteria, display: “Sorry, it looks like we don’t have that in stock!”

A newly added car should display \$0 for total for parts.

Parts:

Parts status can not be reverted to its prior status.

Parts order number is generated by adding the order number to the VIN

Parts order status has to be Ordered, Received, or Installed

A vehicle cannot be returned for any public searches if it has pending parts.

Parts can not be added to a car that has been sold.

Once a part is installed, its status can no longer be updated.

Buyers/Sellers

Sales Price is 125% the original purchase price combined with 110% of any parts cost associated with the vehicle.

A car can only be sold once.

The sales price of a car can not be changed

Reports

Seller History:

Any seller who has sold vehicles and shows an average of five or more parts on this report, or where the average cost of parts is \$500 or more, should have their resulting row highlighted with a red background

Average Time in Inventory:

When counting days spent in the inventory, the first and last day should be counted as a day, and if a vehicle was added and sold on the same day, it will be counted as one day.

If a vehicle type has no sales history, the report should display "N/A" for that vehicle type.

Price per condition report:

If a vehicle type or condition has never been purchased, the report should display "\$0" for that result.

Monthly Sales:

If a year or month does not have sales data, it can be excluded from this report.

Public Search Page

Task Decomp

Lock Types: Lookup Vehicle table.

Number of Locks: Single

Enabling Conditions: Reports is enabled by login of a privileged user

Frequency: High

Consistency (ACID): Vehicle/Part table must complete all operations first.

Subtasks: Search Input and results. Seller History Report. Average Time in Inventory Report. Price per condition report. Parts Statistics report. Monthly sales report.

Abstract Code

- Show “Total Available Cars” text box, “Search Criteria” input fields, “Username Login” input field, “Password Login” input field, “Login” button
- Upon:
 - Click Login button
 - If a User record is found but user.password != ‘\$Password’:
 - Display error message
 - Else:
 - Store login information as session variable ‘\$UserID’.
 - Check User privilege type user.privilege
 - If user.privilege == clerk
 - Reveal an add vehicle button
 - Reveal input field in search criteria for previously sold vehicles
 - Reveal number of vehicles available with parts pending
 - Reveal a VIN input field to the search criteria input fields
 - If user.privilege == sales
 - Reveal a VIN input field to the search criteria input fields
 - If user.privilege == manager
 - Reveal a VIN input field to the search criteria input fields
 - Reveal number of vehicles available with parts pending
 - Reveal number of vehicles available with parts pending
 - Reveal option drop down to filter queries by sold, unsold, or all

- Reveal drop down with links to Seller History, Average Time in Inventory, Price Per Condition, Parts Statistics, and Monthly Sales reports
 - If user.privilege == owner
 - Reveal input field in search criteria for previously sold vehicles
 - Reveal an add vehicle button
 - Reveal a VIN input field to the search criteria input fields
 - Reveal number of vehicles available with parts pending
 - Reveal option drop down to filter queries by sold, unsold, or all
 - Reveal drop down with links to Seller History, Average Time in Inventory, Price Per Condition, Parts Statistics, and Monthly Sales reports
- Enter Search Criteria
 - Try query with entered criteria
 - If criteria does not match data in database
 - Display error
 - Else:
 - Display results VIN, Vehicle Type, Model Year, Manufacturer, Model, Fuel Type, Color(s), Mileage, Sales Price, and link to detail page for each matching search criteria
- Click Detail Page button- Jump to the Detail Page Task
- Click Add Vehicle button- Jump to the Add Vehicle Form Task
- Click Seller History Report button- Jump to the Seller History Report Task
- Click Average Time In Inventory Report button- Jump to the Average Time In Inventory Report Task
- Click Price Per Condition button- Jump to the Price Per Condition Report Task
- Click Parts Statistics Report button- Jump to the Parts Statistics Report Task
- Click Monthly Sales Report button- Jump to the Monthly Sales Report Task

Add Vehicle Form

Task Decomp

Lock Types: Insert vehicle table.

Number of Locks: Single.

Enabling Conditions: Enabled by user type of clerk or owner

Frequency: High

Consistency (ACID): Customer table must complete all operations first

Subtasks: Add/Search Customer Page

Abstract Code

Upon:

- Click: Add Vehicle -
 - Add/Search Customer
 - Upon
 - Click: Search/Add Customer - Navigate to Add/Search Customer
 - Vehicle Details
 - model(this.state.newCar.model)
 - Model Year (this.state.newCar.modelYear)
 - Mileage (this.state.newCar.mileage)
 - Description (this.state.newCar.description)
 - VIN (this.state.newCar.VIN)
 - Car Type (this.state.newCar.chasis)
 - Manufacturer ((this.state.newCar.manufactuer)
 - Fuel Type (this.state.newCar.fuelType)
 - Color (this.state.newCar.color)
 - Vehicle Condition (this.state.newCar.condition)
 - Purchase Price (this.state.newCar.purchasePrice)
 - Purchase Date (this.state.newCar.purchaseDate)
 - Upon:
 - Click: Submit Vehicle
 - If fields are valid and submission inserted into Database:
 - Navigate to detail page for submitted vehicle
 - else:
 - Highlight errors in fields

Add/Search Customer Page

Task Decomp

Lock Types: Read customer table.

Number of Locks: Single.

Enabling Conditions: Enabled by user type of sales or owner

Frequency: High

Consistency (ACID): Critical. Customer table must complete all operations first

Subtasks: Add Customer Form.

Abstract Code

Upon:

- Click: Add Vehicle -
 - Add/Search Customer
 - Upon
 - Click: Search Customer - Return customer results
 - Click: Add Customer - Opens add customer form

Add Customer Form

Task Decomp

Lock Types: Insert Customer Table

Number of Locks: Single.

Enabling Conditions: Enabled by user type of sales or owner

Frequency: High

Consistency (ACID): None.

Subtasks: Mother Task is not needed. No decomposition needed.

Abstract Code

Upon:

- Click: Add Customer-
 - Enter the following fields
 - Street
 - City
 - State
 - Postal code
 - Phone number
 - email(optional)
 - IF customer == individual
 - First name
 - Last name
 - Drivers license no.
 - IF customer == business
 - Business tax id
 - Business name
 - Primary contact
 - title

Detail Form

Task Decomp

Lock Types: Lookup Vehicle, lookup parts

Number of Locks: Double

Enabling Conditions: Trigger by link and user login type

Frequency: high

Consistency (ACID): critical, order is critical

Subtasks: Parts order form. Sales order form.

Abstract Code

- Query unprivileged data and display VIN, vehicle type, model year, model name, manufacturer, fuel type, color(s), mileage, sales price, and description
- Check session id:
 - If session id privilege == clerk
 - Query and Display purchase price and total cost of all parts
 - Query and Display list of each part ordered for the car with fields for part number, description, vendor, purchase order, cost, and status
 - For each time the status drop down is updated
 - If status == ordered
 - Drop down lists received/installed as options
 - If installed or received is selected, update vehicle part status in database accordingly
 - If status == received
 - Drop down lists installed as options
 - If installed is selected, update vehicle part status in database
 - If status == installed
 - Drop down turns into immutable text field
 - Reveal link to parts order form
 - If session id privilege == sales
 - Reveal link to sales order form
 - If session id privilege == manager
 - Query and Display seller contact information, and inventory clerk first/last name, original price, purchase date, total parts cost, and list of each part ordered
 - Query and Display if the car has been sold
 - If the car has been sold
 - Query and Display buyer contact information, sales date, and salesperson first/last name
 - If session id privilege == owner
 - Query and Display seller contact information, and inventory clerk first/last name, original price, purchase date, total parts cost, and list of each part ordered
 - Query and Display if the car has been sold
 - If the car has been sold

- Query and Display buyer contact information, sales date, and salesperson first/last name
 - Query and Display purchase price and total cost of all parts
 - Query and Display list of each part ordered for the car with fields for part number, description, vendor, purchase order, cost, and status
 - For each time the status drop down is updated
 - If status == ordered
 - Drop down lists received/installed as options
 - If installed or received is selected, update vehicle part status in database accordingly
 - If status == received
 - Drop down lists installed as options
 - If installed is selected, update vehicle part status in database
 - If status == installed
 - Drop down turns into immutable text field
 - Reveal link to parts order form
 - Reveal link to sales order form
 - Upon
 - Click Parts Order Form button- Jump to the Parts Order Form Task
 - Click Sales Order Form button- Jump to the Sales Order Form Task

Sell Vehicle Form

Task Decomp

Lock Types:. Write vehicle table.

Number of Locks: Single

Enabling Conditions: form is enabled by user type of salesperson or owner

Frequency: Medium

Consistency (ACID): Critical. Customer table must complete all operations first.

Subtasks: Add/Search Customer Page

Abstract Code

Upon:

- Click: Add/Search Customer
 - Upon
 - Click: Search/Add Customer - Opens Add/Search Customer Page
- Click: Confirm Sale
 - Salespeople will enter the date
 - If Sales Date confirmed

- This will add the sale to the database

Seller History Report

Task Decomp

Lock Types: Lookup employees/vehicles/parts tables.

Number of Locks: Several different schema constructs are needed

Enabling Conditions: Reports is enabled by user type of manager or owner

Frequency: Low

Consistency (ACID): Critical. All related tables must complete all operations first

Subtasks: Mother Task is not needed. No decomposition needed.

Abstract Code

Display the following fields:

- Order By: desc(Cars Sold (this.state.sellerHistory.carSolds)) followed by ascen(Average Purchase price of cars sold (this.state.sellerHistory.avgPurchasePrice))
 - If customerName is not null:
 - displayCustomerName
 - Else:
 - displayBusinessName
 - Cars Sold (this.state.sellerHistory.carSolds)
 - Average Purchase price of cars sold (this.state.sellerHistory.avgPurchasePrice)
 - Average number of parts order per vehicle (this.state.sellerHistory.avgPartsPerVehicle)
 - Average cost of parts per vehicle (this.state.sellerHistory.avgCostPartsPerVehcile)
- If Average number of parts order per vehicle (this.state.sellerHistory.avgPartsPerVehicle) >= 5 or Average cost of parts per vehicle (this.state.sellerHistory.avgCostPartsPerVehcile) >= 500
 - Highlight rows of seller

Average Time In Inventory Report

Task Decomp

Lock Types: Lookup: Vehicles table

Number of Locks: Single

Enabling Conditions: Reports is enabled by user type of manager or owner

Frequency: Low

Consistency (ACID): All related tables must complete all operations first

Subtasks: Mother Task is not needed. No decomposition needed.

Abstract Code

Display the following fields:

- Order by: Vehicle Type
 - Vehicle Type(this.state.averageTimeReport.vehicleType)
 - Amount of time in Inventory(this.state.averageTimeReport.time)
 - If boughtDate(this.state.averageTimeReport.bought) == soldDate(this.state.averageTimeReport.sold):
 - Display 1 Day
 - Else if boughtDate is not null and soldDate is not null:
 - Display diff(soldDate - boughtDate)
 - Else:
 - Display N/A

Price Per Condition Report

Task Decomp

Lock Types: Lookup Vehicles table.

Number of Locks: Single

Enabling Conditions: Reports is enabled by user type of manager or owner

Frequency: Low

Consistency (ACID):Critical. All related tables must complete all operations first

Subtasks: Mother Task is not needed. No decomposition needed.

Abstract Code

Display the following fields:

- Order by: Vehicle Type and Condition
 - Vehicle Type(this.state.pricePerConditionReport.vehicleType)
 - Condition (this.state.pricePerConditionReport.condition)
 - Average of all cars of condition(this.state.pricePerConditionReport.averageAll)
 - If vehicle Type or Condition sales == 0:
 - Return \$0
 - else
 - Return Average of all Cars of Condition

Parts Statistics Report

Task Decomp

Lock Types: Lookup Parts History Table

Number of Locks: Single.

Enabling Conditions: Reports is enabled by user type of manager or owner

Frequency: Low

Consistency (ACID): Critical. All related tables must be updated first.

Subtasks: Mother Task is not needed. No decomposition needed.

Abstract Code

Display the following fields:

- Vendor Name (this.state.partsStats.vendorName)
- Number of parts supplied by Vendor (this.state.partsStats.numberOfParts)
- Total Amount spent on parts (this.state.partsStats.totalAmountSpent)

Monthly Sales Report

Task Decomp

Lock Types: Lookup Vehicle/Employees/Sold/Bought

Number of Locks: Several different schema constructs are needed

Enabling Conditions: Reports is enabled by user type of manager or owner

Frequency: low

Consistency (ACID):Critical. All related tables must complete all operations first

Subtasks: Mother Task is not needed. No decomposition needed.

Abstract Code

Summary pages

Displays the following fields order by year and month descending:

IF year or Month has sale data:

- Total number of vehicles sold (this.state.monthlySales.totalVehicles)
- Total Sales Income (this.state.monthlySales.totalSalesIncome)
- Total net income (this.state.monthlySales.totalNetIncome)

Display List of Salespeople sorted by total vehicles sold per salesperson:

- Fields include salesperson first/last name, number of vehicles sold, and total dollar value of vehicles sold

Parts Order Form

Task Decomp

Lock Types: Read Vehicle Table. Read Parts Table. Write Parts Order Table.

Number of Locks: Several different schema constructs are needed

Enabling Conditions: Reports is enabled by user type of manager or owner

Frequency: High

Consistency (ACID): Critical. All related tables must complete all operations first

Subtasks: Mother Task is not needed. No decomposition needed.

Abstract Code

- Display input fields for part number, description, vendor, purchase order, cost, and status
- Display a + button and submit button
- Upon
 - Click + button
 - Generate a new set of parts order input fields
 - Click submit button
 - Query for the amount of parts orders associated with ar VIN
 - Generate parts order number sequentially with the VIN and the amount of parts orders + 1
 - Store data for each input cell in the database