

# Phase 1 Report

## 2021Fall CS6400 Team011

### Table of Contents

<b>1. Data Type .....</b>	<b>3</b>
1.1 User.....	3
1.2 Vehicle .....	3
1.3 Car .....	3
1.4 Convertible.....	3
1.5 Truck .....	3
1.6 Van.....	3
1.7 SUV.....	3
1.8 Manufacturer .....	4
1.9 Sales_Transaction .....	4
1.10 Part .....	4
1.11 Repair_Details.....	4
1.12 Customer.....	4
1.13 Individual.....	4
1.14 Business.....	4
<b>2. Constraints .....</b>	<b>5</b>
<b>3. Task Decomposition &amp; Abstract Code.....</b>	<b>5</b>
3.1 Task: Search Vehicle:.....	5
Task Decomposition .....	5
Abstract Code .....	6
3.2 Task: log In .....	7
Task Decomposition .....	7
Abstract Code .....	8
3.3 Task: Lookup Customer.....	8
Task Decomposition .....	8
Abstract Code .....	8
3.4 Task: Add Customer .....	9
Task Decomposition .....	9
Abstract Code .....	10
3.5 Task: View Vehicle Detail .....	10
Task Decomposition .....	10
Abstract Code .....	12

<b>3.6 Task: Add Vehicle.....</b>	<b>12</b>
Task Decomposition .....	12
Abstract Code .....	13
<b>3.7 Task: Create/Confirm Sale.....</b>	<b>14</b>
Task Decomposition .....	14
Abstract Code .....	15
<b>3.8 Task: Start New Repair Order .....</b>	<b>15</b>
Task Decomposition .....	15
Abstract Code .....	16
<b>3.9 Task: Update&amp;Complete Repair.....</b>	<b>17</b>
Task Decomposition .....	17
Abstract Code .....	17
<b>3.10 Task: Report Form.....</b>	<b>18</b>
Task Decomposition .....	18
Abstract Code .....	19

# 1. Data Type

## 1.1 User

Attribute	Data Type	Nullable
User_Name	String	Not Null
Password	String	Not Null
First_Name	String	Not Null
Last_Name	String	Not Null

## 1.2 Vehicle

Attribute	Data Type	Nullable
List_Price	float	Not Null
Invoice_Price	float	Not Null
Color	List<String>	Not Null
Model_Name	String	Not Null
Model_Year	int	Not Null
VIN	String	Not Null
Description	String	Null
Add_Date	Date	Not Null
isSold	Boolean	Not Null

## 1.3 Car

Attribute	Data Type	Nullable
Number_of_Doors	int	Not Null

## 1.4 Convertible

Attribute	Data Type	Nullable
Roof_Type	String	Not Null
Back_Seat_Count	int	Not Null

## 1.5 Truck

Attribute	Data Type	Nullable
Cargo_Capacity	float	Not Null
Cargo_Cover_Type	String	Null
Number_Of_Rear_Axles	int	Not Null

## 1.6 Van

Attribute	Data Type	Nullable
Has_Driver_Side_Back_Door	Boolean	Not Null

## 1.7 SUV

Attribute	Data Type	Nullable
Drivetrain_Type	String	Not Null
Number_Of_Cupholder	int	Not Null

## 1.8 Manufacturer

Attribute	Data Type	Nullable
Name	String	Not Null

## 1.9 Sales Transaction

Attribute	Data Type	Nullable
Transaction Date	Date	Not Null
Sold Price	float	Not Null

## 1.10 Part

Attribute	Data Type	Nullable
Quantity	int	Not Null
Vendor	String	Not Null
Price	float	Not Null
Part Number	String	Not Null

## 1.11 Repair Details

Attribute	Data Type	Nullable
Start Date	Date	Not Null
Completed Date	Date	Null
Odometer Value	int	Not Null
Labor Charges	float	Null
Total Cost	float	Null

## 1.12 Customer

Attribute	Data Type	Nullable
Address	String	Not Null
Email	String	Null
Phone Number	String	Not Null

## 1.13 Individual

Attribute	Data Type	Nullable
First Name	String	Not Null
Last Name	String	Not Null
Driver's Liscence	String	Not Null

## 1.14 Business

Attribute	Data Type	Nullable
Tax Identification ID	String	Not Null
Contact Title	String	Not Null
Contact Name	String	Not Null
Business Name	String	Not Null

## 2. Constraints

1. The vehicle's year  $\leq$  current year plus one
2. The vehicle's year must include century digits.(eg. 1999,2001 but not 99, 01)
3. The color of the vehicle needs to be chosen from the list below:

Colors

Aluminum	Beige	Black	Blue	Brown	Bronze	Claret
Copper	Cream	Gold	Gray	Green	Maroon	Metallic
Navy	Orange	Pink	Purple	Red	Rose	Rust
Silver	Tan	Turquoise	White	Yellow		

4. Sold\_price(vehicle), price(part), labor\_charge and total cost (repair) need to include dollar and cents(2 decimal places, eg. \$10.00,\$109.67)
5. The list price of a vehicle is 125% of its invoice price.
6. The repair completed  $\geq$  repair start date
7. A vehicle can only have one repair on the same date.
8. The repair Completed\_Date is null until the repair is finished
9. The new repair can only be started if the existing repair is completed for the same vehicle
10. The user can only update labor charges, add parts or complete the repair if the Completed\_Date is null.
11. Total repair cost = labor charge + parts cost
12. If user type is salespeople, The sold price must  $>$  95% of invoice price
13. If user type is Owner (Rolando), sold price could  $\leq$  95% of invoice price
14. If user type is service writer, the labor charges must  $\geq$  previous value when updating the unfinished repair form,
15. If user type is Owner (Rolando), the labor charges could  $<$  previous value when updating the unfinished repair form,

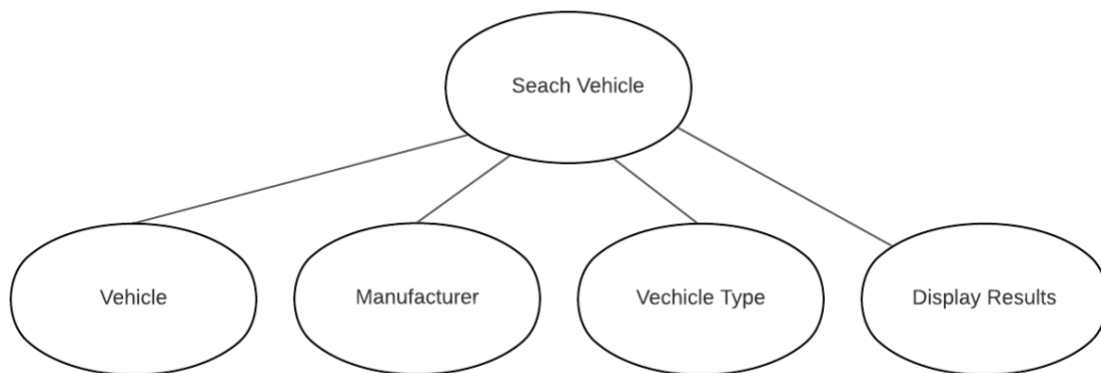
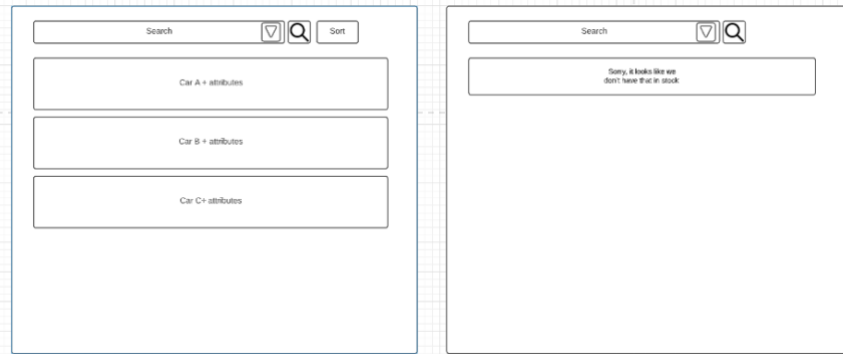
## 3. Task Decomposition & Abstract Code

### 3.1 Task: Search Vehicle:

#### Task Decomposition

The image shows three wireframe diagrams illustrating the task decomposition for searching a vehicle. Each diagram includes a search bar, a 'Log In' button, and a 'Vehicle Available' status indicator.

- Diagram 1 (Left):** Shows the initial search interface. The search bar is empty, and the 'Vehicle Available' status is 'XXX'. A 'Log In' button is at the bottom right.
- Diagram 2 (Middle):** Shows the search interface with filters applied. The search bar contains 'Search'. Below it, there are two columns of filters: 'Color' and 'Manufacturer' on the left, and 'Type' and 'Year' on the right. A 'Log In' button is at the bottom right.
- Diagram 3 (Right):** Shows the search interface with results. The search bar contains 'Search'. Below it, there are two columns of filters: 'Color' and 'Manufacturer' on the left, and 'Type' and 'Year' on the right. A 'Log In' button is at the bottom right.



1. **Lock Types:** Read Only on Vehicle, Vehicle Type and Manufacturer table.
2. **Number of Locks:** Three.
3. **Enabling Conditions:** None except searching by VIN option only available to privilege users.
4. **Frequency:** Frequency is the same for all search criteria.
5. **Consistency(ACID):** Consistency is not essential.
6. **Subtasks:** Mother task is needed. Order is unnecessary.

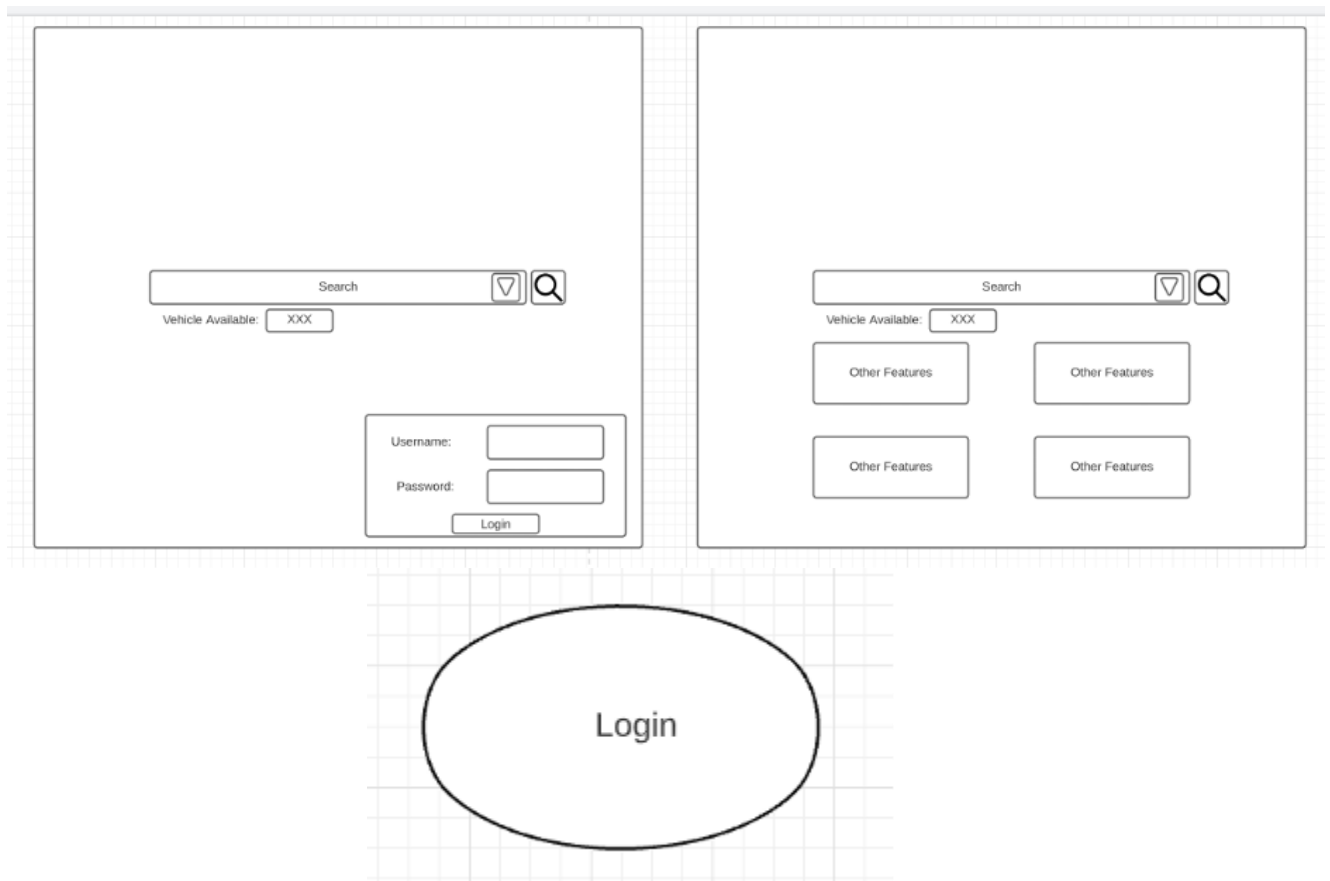
### Abstract Code

- Display number of Unsold **Vehicle**.
- If Click **Search Dropdown** – Display dropdown selections to Search by **Vehicle Type**, **Manufacturer**, Color or Year;
- if **User** is logged in: Display option to search by VIN.
  - Click **Vehicle Type** – Display dropdown selections to **Vehicle Types**.
    - Click a **Vehicle Type** – **Search Vehicle** by **Vehicle Type** and **Display Results**.
  - Click **Manufacturer** – Display dropdown selections to **Manufacturer**.
    - Click a **Manufacturer** – **Search Vehicle** by **Manufacturer** and **Display Results**.
  - Click **Color** – Display dropdown selections to Colors.
    - Click a **Color** – **Search Vehicle** by Color and **Display Results**.
  - Click **Year** – Display dropdown selections to Years.

- Click a **Year** – **Search Vehicle** by Year and **Display Results**.
- Click **VIN** – Select VIN as the Search Criteria.
- **User** enters search input fields.
- Click **Search Vehicle**
  - If VIN is Search Criteria: **Search Vehicle** by VIN using input and Display Results.
  - Else: **Search Vehicle** by List price or keyword using input and Display Results.
    - If **Vehicle** not found: display “Sorry, it looks like we don’t have that in stock!”
    - Else: for each matching **Vehicle** display VIN, Type, Year, Manufacturer, Model, Color and List price.

### 3.2 Task: log In

#### Task Decomposition



1. **Lock Types:** Read only on **User** table.
2. **Number of Locks:** One.
3. **Enabling Conditions:** None.
4. **Frequency:** Frequency is around 200 times a day.
5. **Consistency(ACID):** Consistency is not essential.

6. **Subtasks:** Mother task is not needed. No decomposition needed.

### Abstract Code

- Upon: Click **Login** - Display **Login Form**.
- User enters *username, password* input fields.
- If data validation is successful for username and password input fields:
- Upon:
  - Click **Enter**:
    - If **User** record is not found: Go back to **Login Form**, Display error message.
    - Else: go to **Search Vehicle** and display functions according to **User** permission.
  - Else: display **Login Form**, with error message.

### 3.3 Task: Lookup Customer

#### Task Decomposition

Customer Name:

Look up Customer

1. **Lock Types:** Read Only on **Customer**.
2. **Number of Locks:** One, Customer.
3. **Enabling Conditions:** All privileged users.
4. **Frequency:** Frequency is consistent. Only one task.
5. **Consistency(ACID):** Consistency is not essential.
6. **Subtasks:** No mother task needed. No decomposition needed

### Abstract Code



- User enters Customer input fields.
  - If data validation is successful for input fields:
  - Upon:
    - Click *Search*:
      - if Customer is found: display Customer name.
      - else: *Add Customer*.
    - Else: Display error message.

### 3.4 Task: Add Customer

#### Task Decomposition

The diagram illustrates a user interface for adding a customer. It features a top navigation bar with a close button, radio buttons for 'Individual' and 'Business' selection, and a 'Text' label. Below this bar is a 'Select' button. The central area is a large form labeled 'Individual info form/Business info form'. At the bottom, there is a 'Confirm' button.



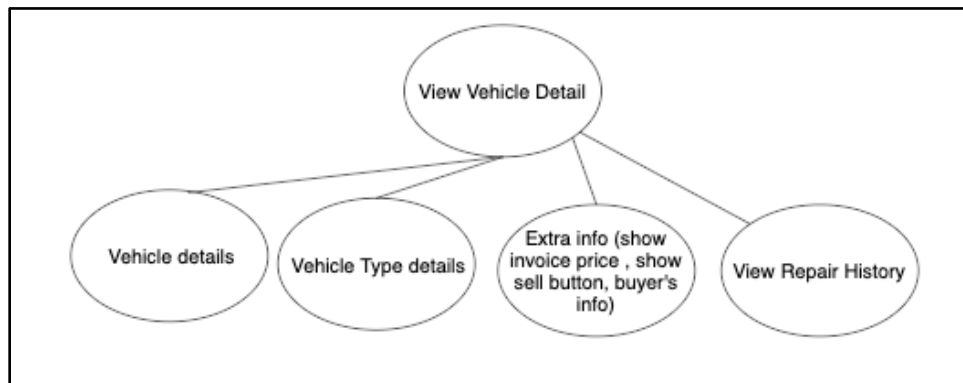
1. **Lock Types:** Write only on **Customer** table.
2. **Number of Locks:** One.
3. **Enabling Conditions:** All privileged users.
4. **Frequency:** Low.
5. **Consistency(ACID):** Consistency is essential.
6. **Subtasks:** No mother task needed. No decomposition needed.

### Abstract Code

- Upon:
- Click **Select**:
  - If **Individual** is selected: Display **Individual Info Form**.
  - Else If **Business** is selected: Display **Business Info Form**.
  - Else: Display an error message.
- **User** Enters input fields of displayed Info Form.
  - If data validation is successful for input fields:
  - Upon:
    - Click **Confirm: Add Customer**.
  - Else: Display an error message.

### 3.5 Task: View Vehicle Detail

#### Task Decomposition



Vehicle Details	
VIN	1243543523
Manufacturer	Toyota
Model Year	2010
Model Type	1243543523
Vehicle Type	Convertible
Roof Type	open
Back Seat Count	2
Vehicle Color	Red White
Invoice Price	10,000
List Price	12,500
Description	This is a good car. Many people like this car. It is classic
Add By	Inventory Clerk A
Add Date	2021-09-22
Back to Search	Sell this Vehicle
buyer's contact info: address, email, phone Number first name, lastName (or Contact title, contact name, business name)	
sold price sales date salesperson's name: first last	
Repair customer nameA(first last), service writer, repair start date, end date, labor charge, parts cost, total cost customer nameB(first last), service writer, repair start date, end date, labor charge, parts cost, total cost customer nameC(company name), service writer, repair start date, end date, labor charge, parts cost, total cost	

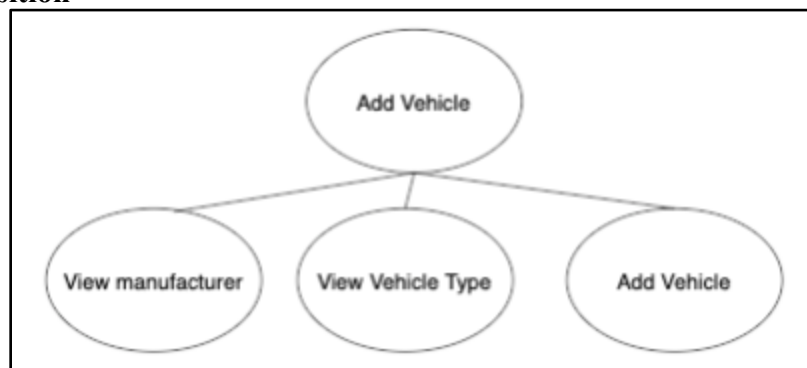
1. **Lock Types:** Read only on [User](#), [Vehicle](#), [Vehicle Type](#), [Repair\\_Details](#)
2. **Number of Locks:** Four
3. **Enabling Conditions:** All users are able to see common info. Only inventory clerk, manager, owner can view invoice price, only sales people and owner can see Sell the vehicle button, only manager and owner can see buyer's contact info, sales info(if the vehicle is sold) and repair history.
4. **Frequency:** high on view vehicle details, type details, low on enter info and view repair history as only a certain group of people has the right to view them.
5. **Consistency(ACID):** Consistency is not essential.
6. **Subtasks:** Mother task is needed. Order is not critical.

## Abstract Code

- Find the **Vehicle** using the VIN
- Display VIN, Manufacturer, Model Year, Model Type, Vehicle Type, Vehicle Color, List Price, Description
- Find the **Vehicle Type** by Vehicle Type Name, Display corresponding attributes(like roof type and back seat count for convertible)
- If **User** belongs to **inventory writer** or **manager** or **owner**:
  - Display Invoice Price
  - If **User** belongs to manager or owner:
    - If the **Vehicle.isSold** is True:
      - Display **Customer** address, email, phone number, and (first name, last name) or (contact title, contact name, business name)
      - Display sold price, sales date, salesperson's name(first last)
    - If the **Repair\_Details list** is not empty:
      - Display list of **Repair\_Details** (customer name, service writer, repair start date, repair end date, labor charge, parts cost, total cost)
- If **User** belongs to **Sales\_writer**:
  - Display ***Sell this Vehicle*** button
- If ***Back to Search***, go to **Access Form**
- If ***Sell this Vehicle***, go to **Sales Order Form**

## 3.6 Task: Add Vehicle

### Task Decomposition



**Add Vehicle**

VIN

Manufacturer

Model Year

Model Type

Vehicle Type

Roof Type

Back Seat Count

Vehicle Color

Invoice Price

Description 

Fill the text

Add Date May 10, 2021

Back to Search

Add Vehicle

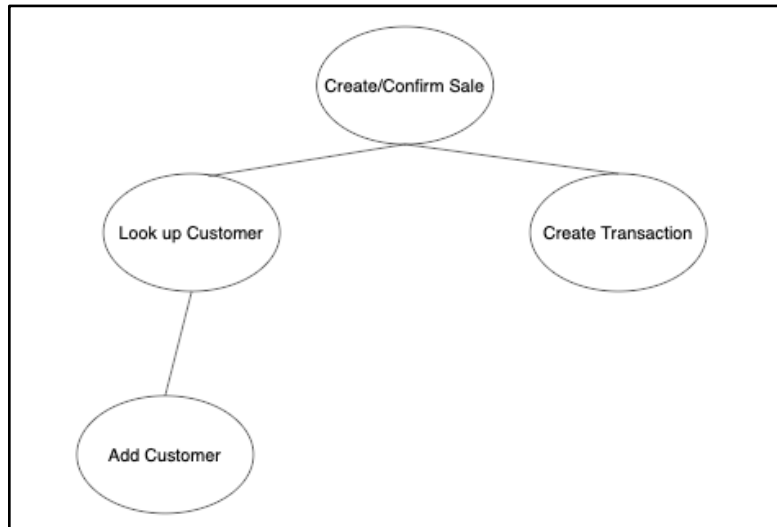
1. **Lock Types:** Write only on **Vehicle**, Read-only on **Manufacturer** and **Vehicle Type**.
2. **Number of Locks:** Three.
3. **Enabling Conditions:** All privileged user can add **Vehicle**.
4. **Frequency:** Same frequency for all three sub tasks.
5. **Consistency(ACID):** Consistency is essential for adding vehicle.
6. **Subtasks:** Mother task is needed. Sequence is important: populate drop list first, add vehicle later.

### Abstract Code

- Populate **Manufacturer** and **Vehicle Type** dropdowns
- While no buttons are pushed, do nothing.
- When a button is pushed, then do the following:
  - If **Back to Search**, go to **Access** Form
  - If **Add Vehicle**:
    - {If there is blank fields, populate error message,
    - If there is entering that is not valid, populate error message,
    - Else **Add Vehicle**, Then go to **Vehicle Detail Form**.}

### 3.7 Task: Create/Confirm Sale

#### Task Decomposition



Sales Order	
Vehicle	VIN: 1234346456
Customer	<div>Look up Customer</div> <div><b>Individual</b> Driver Licences Number: 12324234 Customer Name:</div>
Sold Price	_____
Sold Date	2021-Sep-12(Today)
<div>Cancel &amp; Back to Vehicle</div>	<div>Confirm Order</div>

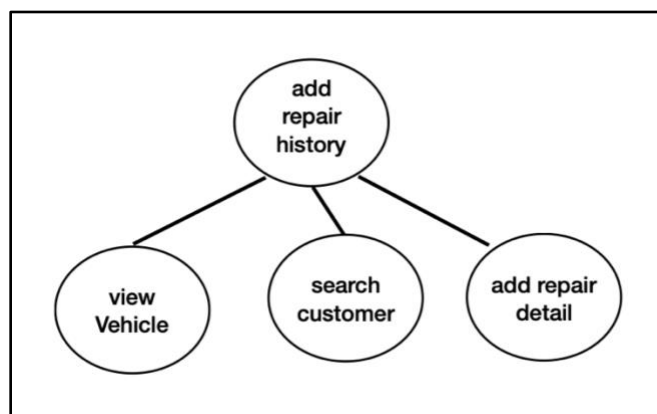
- **Lock Types:** Read only on **Vehicle**, **Sales\_writer**, Read only or Write only on **Customer** depend if need to add customer
- 1. **Number of Locks:** Three
- 2. **Enabling Conditions:** All privileged user can add Vehicle
- 3. **Frequency:** Same frequency for all three sub tasks
- 4. **Consistency(ACID):** Consistency is essential for adding vehicle
- 5. **Subtasks:** Mother task is needed. Sequence is important: only add customer when no customer is found, only create transaction after put in a customer

### Abstract Code

- Display VIN for current **Vehicle**, display current date as Sold date
- While no buttons are pushed, do nothing.
- When a button is pushed, then do the following:
  - if **Look up Customer**:
    - Go to **Look up Customer** Form
    - if no customer found:
      - populate message 'no customer found',
      - Go to **Add Customer** Form
    - Display **Customer** info;
  - if **Cancel & Back to Vehicle**, go back to **Vehicle details** for current **Vehicle**
  - If **Confirm Order**:
    - If there is blank fields(no customer, no price), populate error message,
    - If there is entering is not valid(sold price less than 95% invoice price & not owner) populate error message,
    - Else insert transaction information and go back to **Privileged Access Common Form**

### 3.8 Task: Start New Repair Order

#### Task Decomposition



Add repair

Odometer reading: \_\_\_\_\_

Current date(start date): \_\_\_\_\_

Customer:

**Lock Types:** Write customer (write and read) repair detail read: manufacturer transaction

**Number of Locks:** Four

**Enabling Conditions:** Owner and service writer

**Frequency:** Frequency is the same for view vehicle, customer search and add repair detail

**Consistency (ACID):** Consistency is essential.

**Subtasks:** Mother task needed to coordinate subtasks. Order is critical.

### Abstract Code

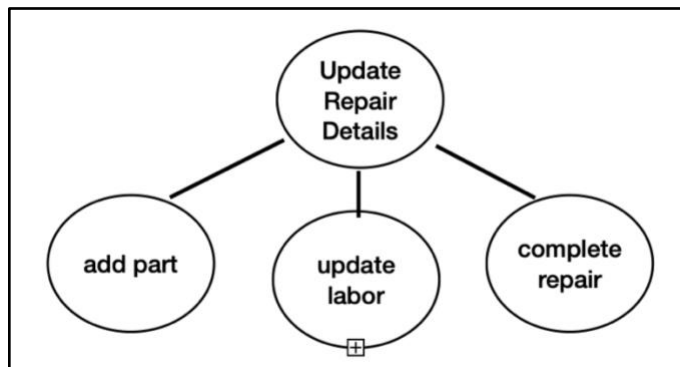
- Search by VIN in [Sales Transaction](#) table.
  - If VIN not found,
    - Display the Error message.
  - If found VIN,
  - If has unfinished repair,
    - Display [Repair Details](#)
    - {Odometer reading, Current date (start date), Customer, Parts, Labor charges, Total charges}
  - If all finished or has no repair history,
    - Add new repair.
    - Add the detail for creating a repair, enter details.
    - Odometer reading, Current date (start date).
    - Search the [Customer](#).
      - If the customer is the original customer
        - choose the customer.
      - If the customer is not the original customer,
        - Add a new customer for repair.
        - View [Repair Details](#).
        - {Odometer reading, Current date (start date), Customer, Parts, Labor charges, Total charges}
  - When a button is pushed, then do the following
    - When the *Add labor charge* button is clicked
      - Enter the labor charge. (Validation: if not owner, labor charge cannot be less than previous value)
      - View [Repair Details](#).



- {Odometer reading, Current date (start date), Customer, Parts, Labor charges, Total charges}
- When the **Add parts button** is clicked,
  - Enter details {Quantity (Validation: quantity is a number), vendor, part number, price}
  - View [Repair Details](#).
    - {Odometer reading, Current date (start date), Customer, Parts, Labor charges, Total charges}

### 3.9 Task: Update&Complete Repair

#### Task Decomposition



Add parts

Quantity: \_\_\_\_\_

Vendor: \_\_\_\_\_

Part number: \_\_\_\_\_

Price: \_\_\_\_\_

**Save**

Add labor charge:

Labor charge: \_\_\_\_\_

**Save**

**Lock Types:** Read and write on repair detail and part

**Number of Locks:** Two

**Enabling Conditions:** Owner and service writer

**Frequency:** Same frequency for add, update and complete

**Consistency (ACID):** Consistency is essential.

**Subtasks:** Mother task is needed to coordinate subtasks. Order is critical.

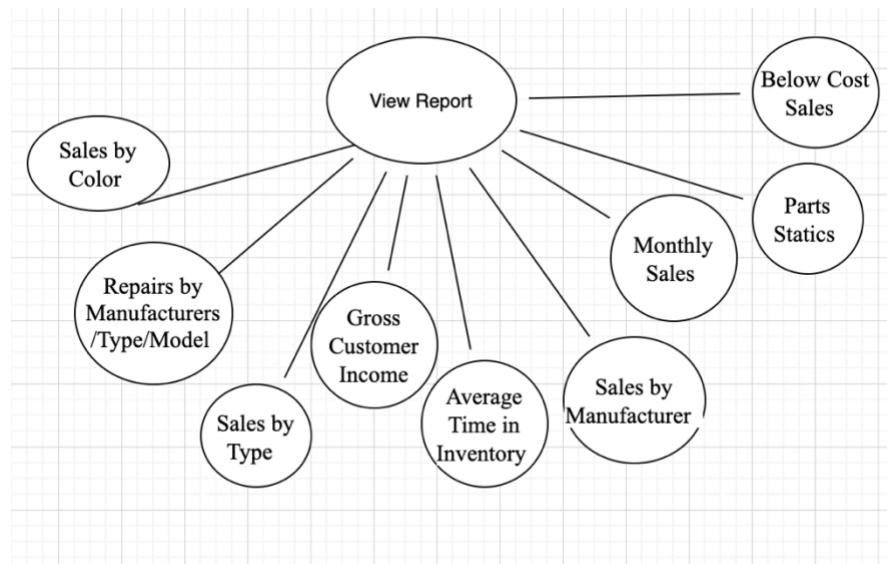
#### Abstract Code

- Display [Repair Details](#).

- Odometer reading, Current date (start date), Customer, Parts, Labor charges, Total charges}
- When a button is pushed, then do the following:
  - When **Add labor charge** button is click,
  - Enter the labor charge.
    - Validation: if not owner, labor charge cannot be less than previous value)
  - When the **Save** button is click,
  - Display the **Repair Details**.
    - Odometer reading, Current date (start date), Customer, Parts, Labor charges, Total charges}
  - When **Add Part** button is click, enter details
    - {Quantity (Validation: quantity is a number), vendor, part number, price}
    - When the Save button is click,
  - Display the **Repair Details**.
    - {Odometer reading, Current date (start date), Customer, Parts, Labor charges, Total charges}
  - When **Repair Complete** button is click,
  - Display the **Repair Details**.
    - {Odometer reading, Current date (start date), Customer, Parts, Labor charges, Total charges}

### 3.10 Task: Report Form

#### Task Decomposition



Report Name

Date

Report Content

**Lock Types:** Read-only on all tables for Manager and Owner access type.

**Number of Locks:** several different schema constructs are needed.

**Enabling Conditions:** Enabled only by Manager and the Owner user type.

**Frequency:** Low, around daily.

**Consistency (ACID):** is not critical, order is not necessary.

**Subtasks:** Mother task is needed, decomposition needed. Order is not necessary.

### Abstract Code

- **User login** from the **Login Form**
- If the user's type is **Manager** or **Owner**
  - Display **View Report** button on the **Privileged Access Common Form**
- If **User** click view report dropdown list for **Report Request**,
  - Display Dropdown list of selections that include:
    1. Sales by Color
    2. Sales by Type
    3. Sales by Manufacturer
    4. Gross Customer Income
    5. Repairs by Manufacturers/Type/Model
    6. Below Cost Sales
    7. Average Time in Inventory
    8. Parts Statics
    9. Monthly Sales
- **User** click on one of the selections
  - Display the selected report

- Sales by Color Report
  - No. of vehicle sold in different length of days for a color
- Sales by Type Report
  - No. of vehicle sold in different length of days for a type
- Sales by Manufacturer Report
  - No. of vehicle sold in different length of days for a manufacturer
  - If a manufacturer does not have sales, does not display
- Gross Customer Income Report
  - Part 1: display top 15 customers include {customer name, first sale date, recent action date(sale or repair), number of sales, number of repair, gross income} order by gross income, recent action date desc
  - Part 2: if [User](#) click on certain customer for more details, display:
    - {vehicle: sale date, sold price, VIN, year, manufacturer, model, salesperson name } order by sale date desc, VIN ascending
    - {repair: start date, end date, VIN, odometer value, parts cost, labor cost, total cost, service writer}
- Repairs by Manufacturers/Type/Model Report
  - Display repair by Manufacturers/Type/Model
    - {Count of repairs, the sum of all parts costs, the sum of all labor costs, and the sum of total repair costs, including any repairs in progress}
  - Exclude Manufacturers/Vehicle\_Type/Model without repair
- Below Cost Sales Report
  - Display all vehicles sold below Invoice Price
    - {Include transaction detail of that sale, and highlight sale $\leq$ 95%}
- Average Time in Inventory Report
  - No. of days vehicle has stayed in inventory for a Vehicle\_Type
  - If there is no sale for a Vehicle\_Type, display N/A
- Parts Statics Report
  - Display by vendor, including the number of parts supplied by that vendor, and the total dollar amount spent on parts
- Monthly Sales Report
  - Display all Sales\_Transaction by year and month, total number of vehicles sold, total sales income, the total net income, and the sold price/invoice price ratio as a percentage

- If a month/year does not have sales, does not display
  - If price/invoice for a month is  $\geq 125\%$ , row highlighted as a green background.
  - If the ratio is  $\leq 110\%$ , highlighted as yellow background.
  - Order by year and month descending
- Display drilldown report for year and month
  - Display top performing Salespeople
    - {include first and last name, the number of vehicles and total sale in that year and month}
  - Order by total vehicles descending followed by total sales descending.