|  |  |
| --- | --- |
|  |  |

**eProject Document**

Vehicle Showroom Management System

|  |  |
| --- | --- |
| **Group 2** | |
| **Members** | Tran Quoc Dat - Student1183197  Vo Nhu Nhat - Student1183211  Phung Nguyen Anh Tu – Student1183163 |
| **Instructor** | Lai Duc Chung |
| **Batch** | C1808J1 |
| **Semester** | III |

**🙠 Date of issue 2020 🙢**

**Table of Content**

1. **Problem definition**
   1. ***Introduction***
   2. ***Existing Scenario***
   3. ***Proposed scenario***
2. **Customer Requirement Specification**
   1. ***Customer Requirement***
   2. ***Hardware – Software Requirements***
3. **Architecture and Design of the System**
4. **Flow Chart**
5. **Data Flow Diagram**
6. **Entity Relationship Diagram**
   1. ***Entity relationship diagram***
   2. ***Entity - Attributes***
7. **Database Design/Structure**
   1. ***Database relationship diagram***
   2. ***Table design***
8. **Some typical interface**
9. **Task Sheet**
10. **Check List of Validation**
11. **Submission Checklist**
12. **Problem Definition**
    1. **Introduction**

* This software is to be prepared for a car show room, to maintain details of the vehicle, quotation printing, invoice printing etc. Our client is a car dealer cars.
* The software will store the information of all the employees, sales data and will also generate the monthly reports with the help of which owner can analyze the sales and take necessary steps in order to enhance the business further.
* The software hereafter would be called as “Vehicle Showroom Management System”.
  1. **Existing Scenario**
* There are many executives are working in the show room. Lot of attrition keeps on happening every now and then. And new people are recruited. For them time is required to understand all vehicles details as well as keep track of all the processes taking place in the showroom. To over ride the gap the owner thought of this software, where all the information is pre loaded, and at any moment, any car information is available at just a click.
  1. **Proposed Scenario**
* Vehicle showroom management system describes the complete process of selling a vehicle to a customer from the dealer’s showroom. Before selling, the vehicle belongs to the fixed asset of the dealer’s showroom. So the main point of this scenario is posting the vehicle from a fixed asset to a current asset. While executing the process, the dealer can manually maintain the vehicle business transaction type, which means, the vehicle can be set as a new or used vehicle to sell to the customer. This scenario shows the process of new vehicle sales.
* Vehicle showroom management system is ideal for dealers or resellers of any size. The features include:

+ Easy stock inventory updates

+ Full user management

+ Photograph uploads

+ Marketing reports

+ Easy installation

+ etc…

1. **Customer Requirement Specification**
   1. **Customer Requirement**
      1. **Users of the system**

* Admin
* Employee
* Customer
  + 1. **System functions**

|  |  |
| --- | --- |
| **User** | **Functions** |
| **Admin** | * Login * Change profile * User management * Brand management * Customer management * Purchase order management * Model car management * Car management   - Order management  - View showroom |
| **Employee** | * Login * Change profile * Customer management * Purchase order management * Car management   - Order management   * View showroom |
| **Customer** | * View showroom |

* 1. **Admin functions**

|  |  |
| --- | --- |
| **Login** | |
| Input | Login to system, user must be enter username and password |
| Process | Access information in database |
| Output | Result of login |
| **Change Profile** | |
| Input | Information details of user such as Name, Address, Phone, Email … |
| Process | Maintain user’s information details |
| Output | Result of Change profile |
| **User management** | |
| Input | Information details of user such as ID, Name, Birthday, Email, Address, Phone, … |
| Process | Access information in database to modify, add new user to database, delete from database. |
| Output | Display or maintain Information detail of User. |
| **Brand management** | |
| Input | Information details of Brand such as Name, Image |
| Process | Access information in database to modify, add new brand to database, delete from database. |
| Output | Display or maintain Information detail of Brand. |
| **Customer management** | |
| Input | Information details of Customer such as Name, Address, Phone, Email, … |
| Process | Access information in database to modify, add new customer to database, delete from database. |
| Output | Display or maintain Information detail of Customer. |
| **Purchase order management** | |
| Input | Information details of Purchase order such as Quantity, Order date, Status, … |
| Process | Access information in database to modify, add new purchase order to database, delete from database |
| Output | Display or maintain Information detail of Purchase order. |
| **Model car management** | |
| Input | Information details of Model car such as Name, Brand, Price, … |
| Process | Access information in database to modify, add new Model car to database, delete from database. |
| Output | Display or maintain Information detail of Model car. |
| **Car management** | |
| Input | Information details of Car such as Model number, Name, Price, Color, … |
| Process | Access information in database to modify, add new car to database, delete from database. |
| Output | Display or maintain Information detail of car. |
| **Order management** | |
| Input | Information details of Order such as Model number car, Total money, Status, … |
| Process | Access information in database to modify, add new Order to database, change status, delete from database |
| Output | Display or maintain Information detail of Order. |

* 1. **Employee functions**

|  |  |
| --- | --- |
| **Login** | |
| Input | Login to system, user must be enter username and password |
| Process | Access information in database |
| Output | Result of login |
| **Change Profile** | |
| Input | Information details of user such as Name, Address, Phone, Email … |
| Process | Maintain user’s information details |
| Output | Result of Change profile |
| **Customer management** | |
| Input | Information details of Customer such as Name, Address, Phone, Email, … |
| Process | Access information in database to modify, add new customer to database, delete from database. |
| Output | Display or maintain Information detail of Customer. |
| **Purchase order management** | |
| Input | Information details of Purchase order such as Quantity, Order date, Status, … |
| Process | Access information in database to modify, add new purchase order to database, delete from database |
| Output | Display or maintain Information detail of Purchase order. |
| **Car management** | |
| Input | Information details of Car such as Model number, Name, Price, Color, … |
| Process | Access information in database to modify, add new car to database, delete from database. |
| Output | Display or maintain Information detail of car. |
| **Order management** | |
| Input | Information details of Order such as Model number car, Total money, Status, … |
| Process | Access information in database to modify, add new Order to database, change status, delete from database |
| Output | Display or maintain Information detail of Order. |

* 1. **Hardware – Software requirement**

### Minimum Hardware Requirements

* CPU: Intel Core i3 2.00 GHz.
* RAM: 4GB.
* Hard Disk: 50 GB.
* OS: Window 7.

### Recommended Hardware Requirements

* CPU: Intel Core i5 2.70 GHz or higher.
* RAM: 8 GB or higher.
* Hard Disk: 100 GB or higher.
* OS: Window 10.

### Software Requirements

* Environment: Microsoft .NET Framework 4.0 or later
* Database Management System: Microsoft SQL Server 2014 or later
* IDE: Microsoft Visual Studio 2017 or later
* Web Server: Internet Information Services (IIS 10.0) or later

1. **Architecture and design of the system**

The Application will be made of a Web-based distributed three-tier architecture.

User Interface with HTML

Web - Tier

And ASPX pages

Middle Tier

C# (Code - behind) file containing

business logic

Database - Tier

SQL Server 2014

**Web – based distributed 3 - Tier Architecture of the Project**

1. **Flow chart**
   1. **Login**

Start

Start

Display confirm exit

Input data

Check Data

Display Error

**false**

Logout

**true**

Stop

Successful login

b. System Logout

Stop

1. System Login
   1. **Add / Update Data**

Start

Start

Load data

Input Data

**false**

Display error

Input Data

Display error

Check Data

**true**

Check Data

Save database

**false**

**true**

Save database

Display messenger

Display messenger

Stop

1. Add data

Stop

b. Update data

* 1. **Search data**

Start

Input keyword

Display error

Check data

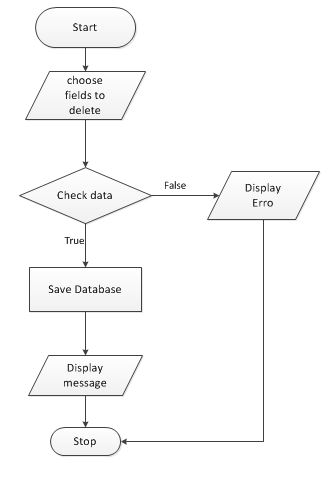
**false**

**true**

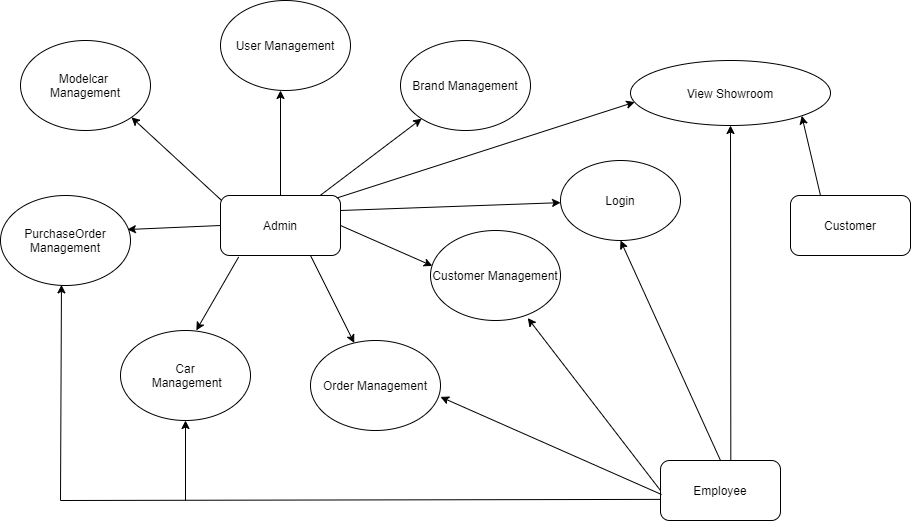
Display result

Stop

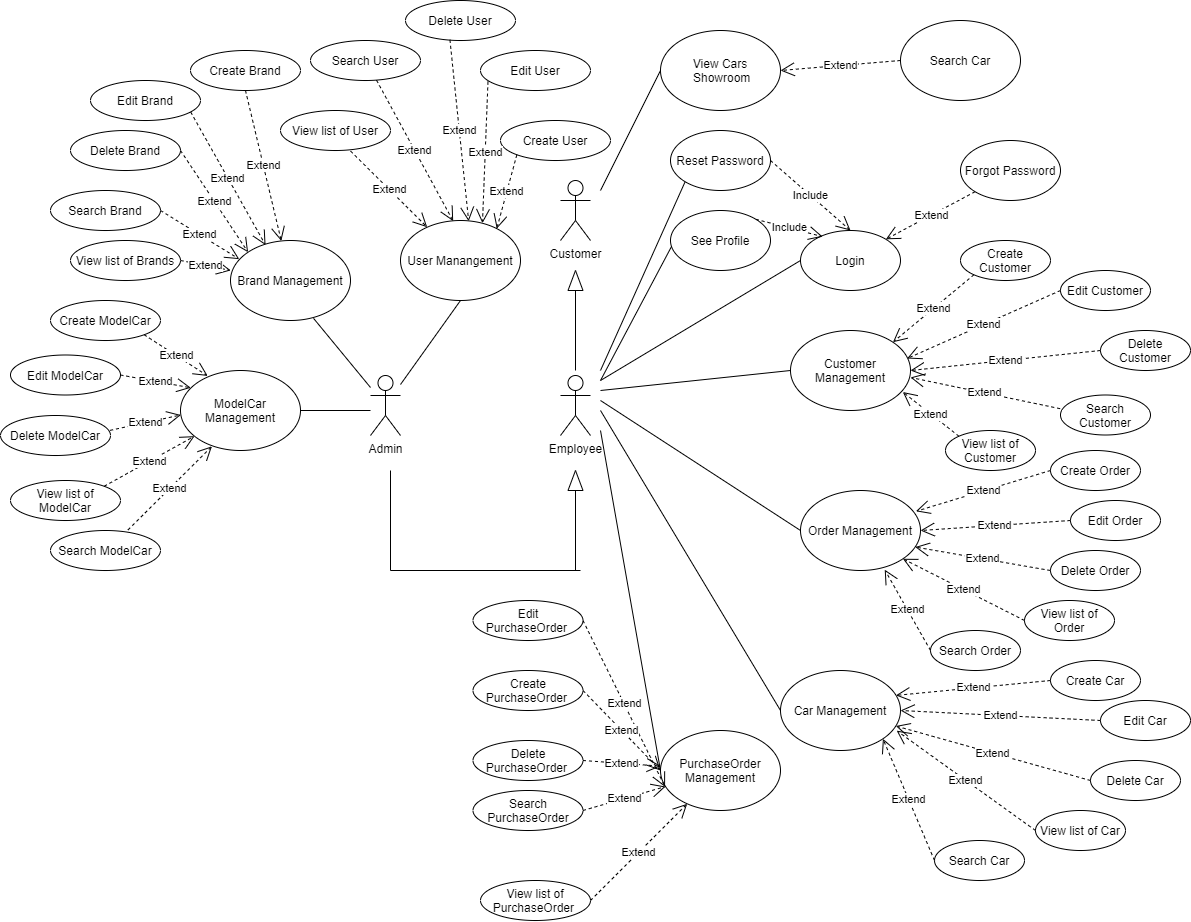
* 1. **Delete Data**

****

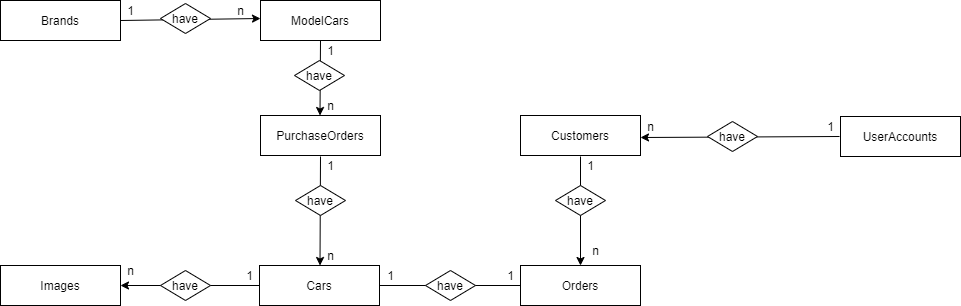
1. **Data flow diagram**
   1. **General Diagram**

****

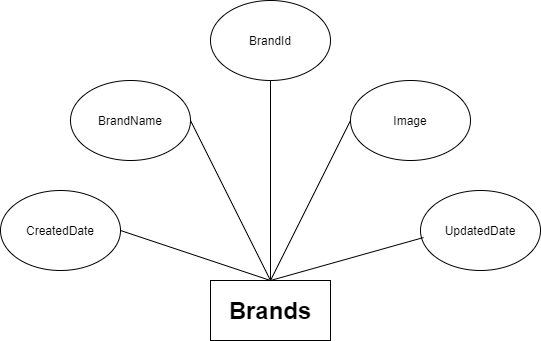
* 1. **Use case**

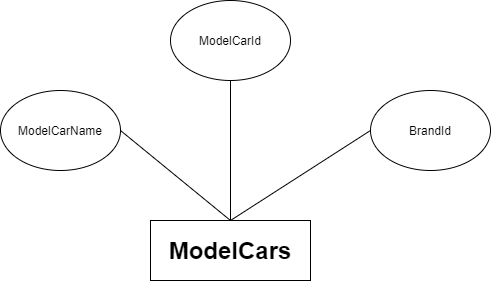
****

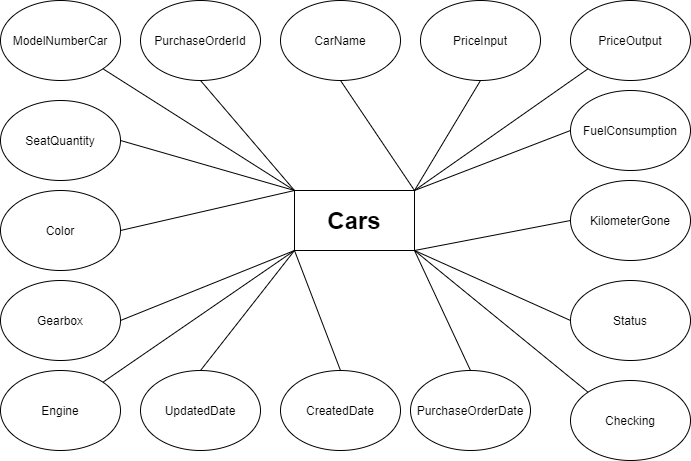
1. **Entity relationship diagram**
   1. **Entity relationship diagram**

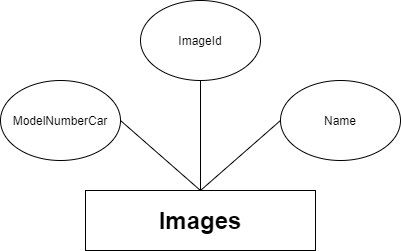
****

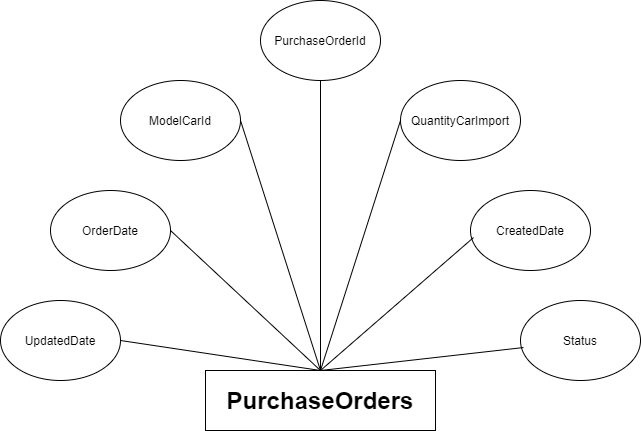
* 1. **Entity – Attributes**

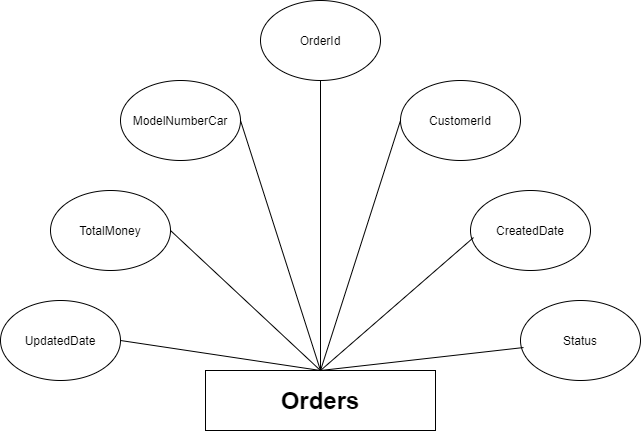
****

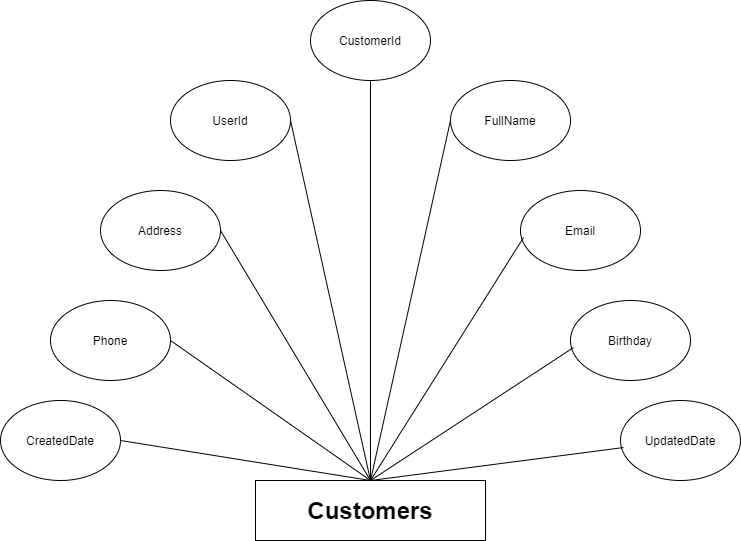
****

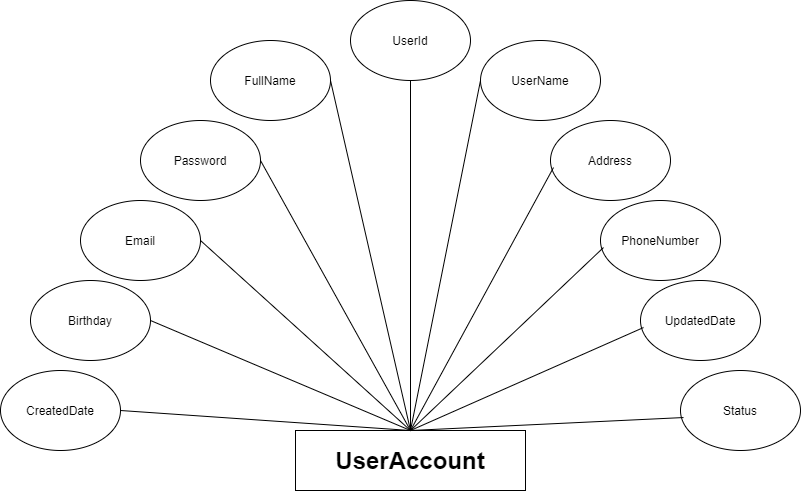
****

****

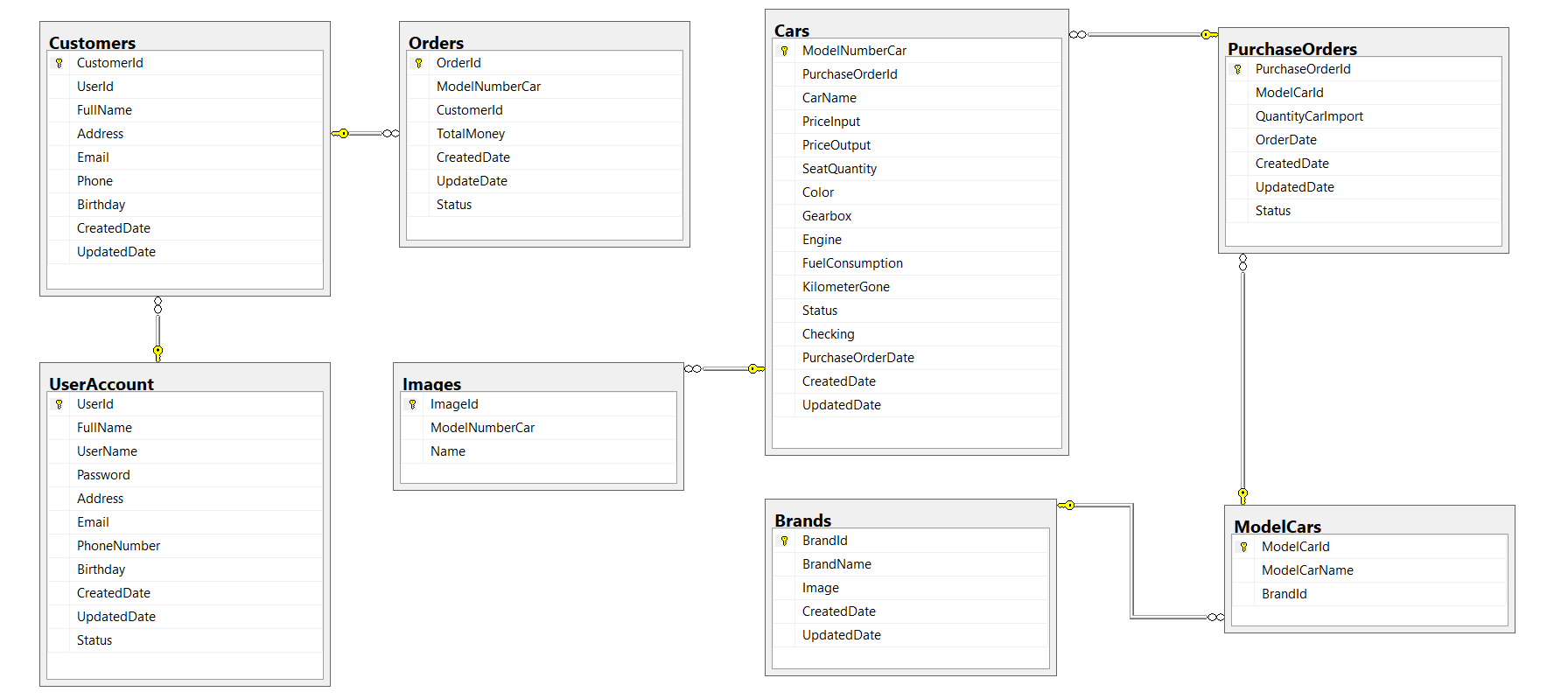
****





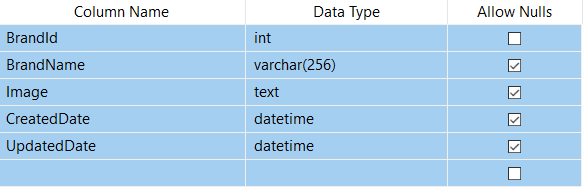
****

1. **Database Design/Structure**
   1. **Database relationship diagram**

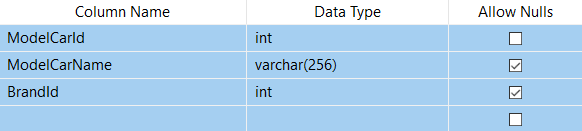
****

* 1. **Table Design**

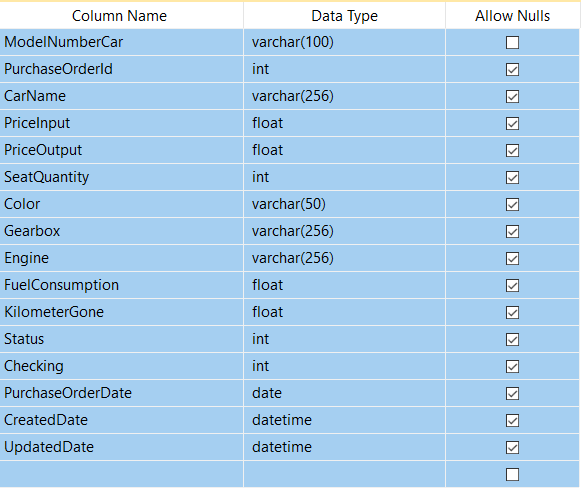
**Brands Table**



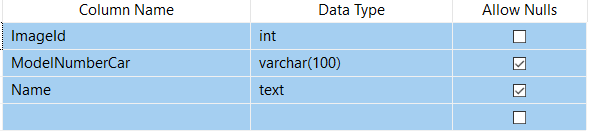
**ModelCars Table**

****

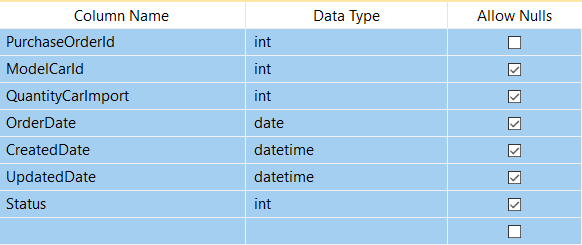
**Cars Table**

****

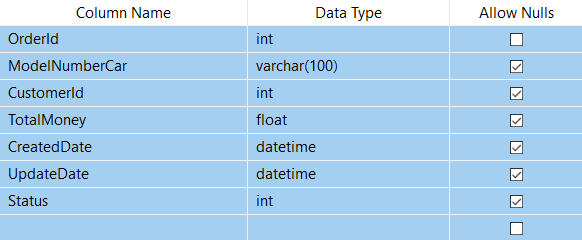
**Image Table**

****

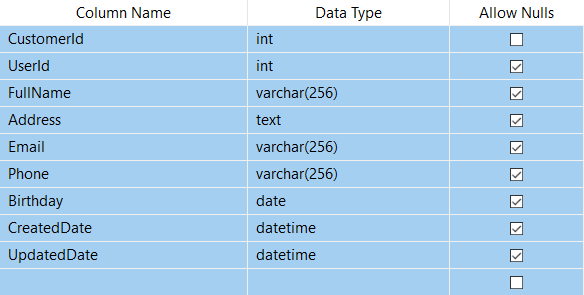
**PurchaseOrder Table**

****

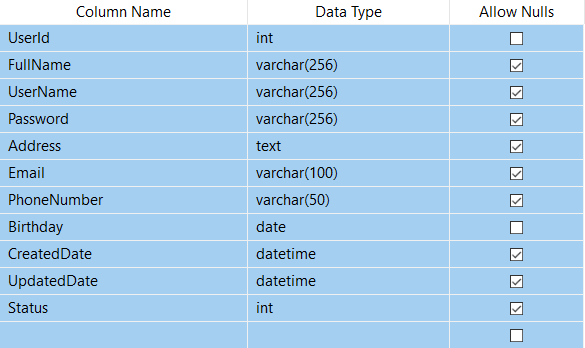
**Order Table**

****

**Customer Table**

****

**UserAccount Table**

****

1. **Some typical interface**
2. **Task Sheet**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No: 1** | | Project Title: Nexus Service Marketing System | **Date of Preparation of Activity Plan** | | | |
| **Sr. No.** | **Task** | | **Actual Start Date** | **Actual Days** | **Team Member Names** | **Status** |
| 1 |  | |  |  |  |  |
| 2 |  | |  |  |
| 3 |  | |  |  |

1. **Check list of validation**

|  |  |
| --- | --- |
| Option | Validate |
|  |  |
|  |  |
|  |  |
|  |  |

1. **Submission checklist**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Number | Particular | Yes | No | Comments |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**THE END**