|  |  |
| --- | --- |
|  | **Car Information Maintenance Application Development** |
|  |  |
|  | ELF / HTD Training Programs  Java Full Stack - Use Cases |

|  |  |
| --- | --- |
|  | Software Solutions |

 

Table of Contents

[**1. INTRODUCTION** 2](#_Toc27629261)

[**2. PROBLEM STATEMENT** 2](#_Toc27629262)

[**3 OBJECTIVE** 2](#_Toc27629263)

[**4 ABSTRACT OF THE PROJECT** 2](#_Toc27629264)

[**5. IMPLEMENTATION** 2](#_Toc27629265)

[**6. DATABASE STRUCTURE** 3](#_Toc27629266)

# **1. INTRODUCTION**

This document outlines the Project. The project is to develop a Car Information maintenance application. This document contains the work flow of the system and gives guidelines on how to build the functionality gradually in each of the course modules.

# **2.** **PROBLEM STATEMENT**

If people want to buy a car then they will do some R&D on the cars, which car is good on what price like that. So people has to travel a lot to get the information about the car. They have to go to so many show rooms, get the car details and compare each car to decide which is better and then buy the car. There is so many manual work to do to get the information

# **3 OBJECTIVE**

To remove all the manual work of the people who wants to get the car information. This application will provide all the information about the car and people can see from the home.

# **4 ABSTRACT OF THE PROJECT**

The car information maintenance application provides all the necessary information about the car. This will give the information about car showroom price, on-road price, petrol or diesel or electric, ac or non-ac, power steering, ABS, mileage, etc.

# **5. IMPLEMENTATION**

Following is a list of functionalities of the system. Wherever, the description of functionality is not adequate, you can make appropriate assumptions and proceed.

There are two types of user i.e. admin and visitor

1. Admin can
   1. Login into the application with his/her credentials
   2. Insert car info into the database
   3. Modify car info from the database
   4. Search car info from the database
   5. Delete car info from the database
2. Visitor (no login for visitor user) can
   1. Search for a car using
      1. Car name
      2. Car company name
      3. Fuel type
   2. Get all the information about the cars after the search

**Note:**

1. There must be only one search form and using the same search form user have to search for name, company name, fuel type
2. The search must use like (%) operator while executing the query
3. On road price for the car should be calculated by following information
   1. If the car showroom price is less than 5 lakhs – 13% of the cost of the car
   2. If the car showroom price is between 5 – 10 lakhs – 14% of the cost of the car
   3. If the car showroom price is between 10 – 20 lakhs – 17% of the cost of the car
   4. If the car showroom price is more than 20 lakhs – 18% of the cost of the car
   5. If the car fuel type is electric – 4% of the cost of the car

# **6. DATABASE STRUCTURE**

Car\_Details

|  |  |
| --- | --- |
| Id | Int (primary key) (auto generate) |
| Name | Varchar(255) |
| Company | Varchar(255) |
| Fuel\_Type | Varchar(255) |
| Power Steering | Boolean |
| Break\_System | Varchar(255) |
| Showroom\_Price | Double |
| Onroad\_Price | Double |
| Image\_URL | Varchar(255) |
| Mileage | Double |
| Seating\_Capacity | Int |
| Engine\_Capacity | Int |
| Gear\_Type | Varchar(255) |

Note:

Each candidate:

1. **Will get 1 or 2 user stories (balancing the complexity levels)**
2. **Must develop the UI Layer, Integration Layer, Business Logic and Datalayer Logic**
3. **Must develop the JUNIT Test Script, RESTAPI TestScripts**