# Software Engineering Project Report

**Report :**   
This report shall be submitted in three steps:  
      1.   Part 1 (Section 1 Customer Problem Statement  and   Section 2 System Requirements)  
      2.   Part 2 (Section 3 Functional Requirements Specification  and  Section 4 User Interface Specification )  
      3.   Entire Report

**Software Engineering Project Report**

Prepared by

GAJENDRA GOSWAMI

DIVYAM SINGH THAKUR

DEVASHISH BHAND

**CAR IN YOUR MIND**

**Problem domain**

SECTION 1

This is the developing world where everyone wishes to feel his life with comfort and luxuries.

Most of the people have confusions for the car.

Which car should they buy?

Which will be their pocket friendly?

Which car will provide more space?

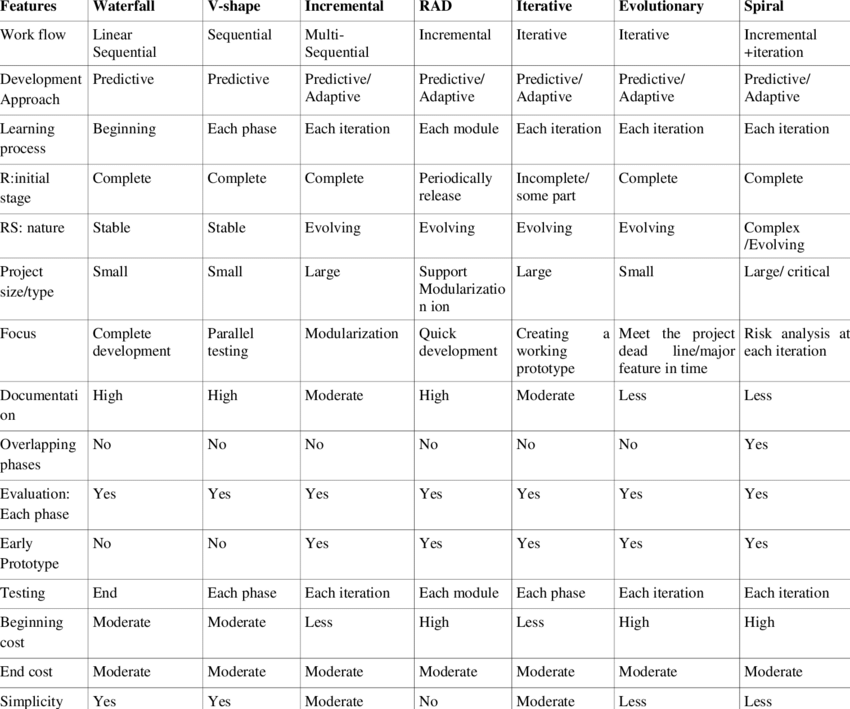
Which car will be having greater mileage?

And all about their specifications and technologies…

# Solution domain

To solve this issue as an aspiring engineer on the field of computer science and engineering we are trying to make a software regarding car problems by which a customer can get full details by just entering their area of interests.

This will include all important features like car specifications, car design, car maintenance, car insurance, car allowability, car budget, car mileage, car backup, about the airbags, space, and all-important features which the customer is seeking for!

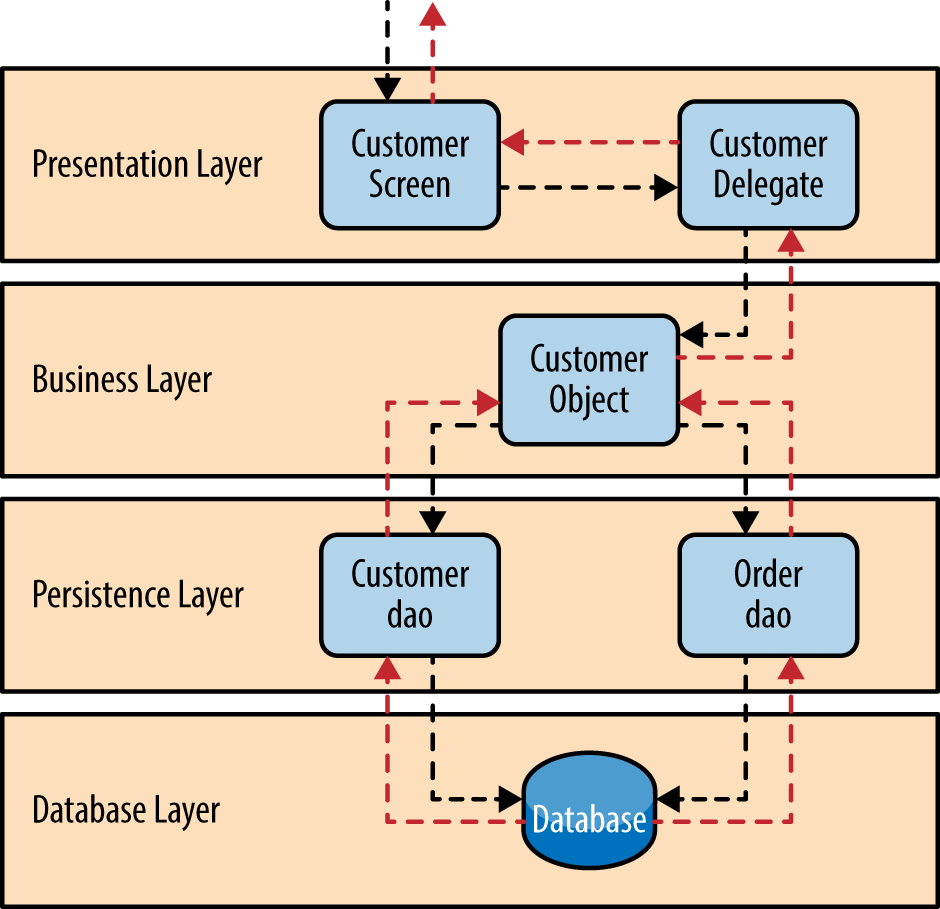


**The presentation layer:**It contains all categories related to the presentation layer.(car display…original documents)

**The business layer:** It contains business logic.(for profit motto)

**The persistence layer:**It’s used for handling functions like object-relational mapping(for user and dealer friendly relations)

**The database layer:**This is where all the data is stored.(data is being stored in whatever positions )



**SECTION 2**

**Agile model**

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like −

* Planning
* Requirements Analysis
* Design
* Coding
* Unit Testing and
* Acceptance Testing.

in our model we will be using agile software development model because this is a current and every time repeated topic because lots of cars get launched back to back so to adjust them in our software and according to user needs we have to use this model. since to add some of the car we cant use other models and write other engineering requirements again..this models helps in consuming time as well also previously stored data remains back to hold the data accurately.

**Agile manifesto and its 12 principles**

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

2. Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.

3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

4. Business people and developers must work together daily throughout the project.

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

7. Working software is the primary measure of progress.

8.Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

9. Continuous attention to technical excellence and good design enhances agility.

10. Simplicity–the art of maximizing the amount of work not done–is essential

11. The best architectures, requirements, and designs emerge from self-organizing teams.

12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

**SECTION 3  
What is a Functional Requirement?**

In software engineering, a functional requirement defines a system or its component. It describes the functions a software must perform. A function is nothing but inputs, its behavior, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform.

Functional software requirements help you to capture the intended behaviour of the system. This behaviour may be expressed as functions, services or tasks or which system is required to perform.

Functional requirement in our project

basically, our project is to determine which car best fits into your budget so the requirements will be

example:

Input from user: best car under 5 lacs.

output: name of the cars

performance

maintenance

mileage

other specifications related to the car

## What is Non-Functional Requirement?

A non-functional requirement defines the quality attribute of a software system. They represent a set of standards used to judge the specific operation of a system. Example, how fast does the website load?

A non-functional requirement is essential to ensure the usability and effectiveness of the entire software system. Failing to meet non-functional requirements can result in systems that fail to satisfy user needs.

Non-functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlog

Non functional requirement in our project

as soon as the user inputs the data and budget according to his needs the list will be appear in ten or less than ten seconds and also it will display the car.