|  |
| --- |
|  |

**Vehicle Reservation System**

**Technical Design Document**

**V 1.0**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | |  | **Prepared By / Last Updated By** | **Reviewed By** | **Approved By** | | **Name** | Md Azhar  Shouvik Dey | Pragya Das  Diksha Jaiswal | Koyel Kujan Kundu | | **Role** | Team Member | Team Member | Scrum Master | | **Signature** |  |  |  | | **Date** |  |  |  | |
|  |

Table of Contents

[1.0 Introduction 3](#_Toc534892426)

[1.1 Purpose of this document 3](#_Toc534892427)

[1.2 Project Overview 3](#_Toc534892428)

1.2.1 Objectives 3

[1.3 Intended Audience 3](#_Toc534892429)

[1.4 Definition & Acronyms 3](#_Toc534892430)

[2.0 Conventions and Standards Followed 4](#_Toc534892431)

[3.0 Assumptions, Dependencies and Risks 4](#_Toc534892432)

[3.1 Assumptions: 4](#_Toc534892433)

[3.2 Dependencies: 4](#_Toc534892434)

[4.0 Schematic Diagram 4](#_Toc534892435)

[5.0 Use Case Realization 5](#_Toc534892436)

[5.1.1 User Registration 5](#_Toc534892437)

[a. Functional Requirement 6](#_Toc534892438)

[b. UI Validations 6](#_Toc534892439)

[c. UI Controls 7](#_Toc534892440)

[d. UI Methods Definition 8](#_Toc534892441)

[e. UI Design 9](#_Toc534892442)

[6.0 Database Design 10](#_Toc534892443)

[6.1 Data Model 10](#_Toc534892444)

[6.2 Tables Structure 10](#_Toc534892446)

[7.0 Change Log 13](#_Toc534892447)

# Introduction

## Purpose of this document

This document is aimed at:

 Providing the necessary inputs to the detailed requirements gathering phase and further on for the SDLC processes.

 This document also serves to establish the traceability between the Business Objectives and the requirements identified in the proposed solution and how they satisfy the stated objectives.

 Provide expectation traceability in terms of the requirements and the user expectation

 Serves as a formal template for documenting the Business Requirements which also includes statutory and regulatory requirements.

The purpose of this document is to systematically capture the requirements of the project and the system to be developed. The document also captures the Functional requirements and serves as an input for the scope of project.

## Project Overview

### Objectives

The objective of this document is to have the overall requirement that would let the fleet management company be able to record & assimilate live inventory data from the branch stores and trigger the need to supply necessary vehicles & vehicle spares to the branch store that is running out of stock.

The application would include the following modules:

* User Registration and login
* Update Vehicle details
* Search Vehicle details

## Intended Audience

*The intended audience for this document are*

* Interens / Project Team.
* Mentors and SME’s.
* Delivery assurance / excellence group.

## Definition & Acronyms

|  |  |
| --- | --- |
| **Acronyms** | **Definitions** |
| VRS | Vehicle Reservation System |

# Conventions and Standards Followed

## Methodology Followed:

In this project we are going to follow **Agile** methodology

AGILE SOFTWARE DEVELOPMENT is an approach to [software development](https://en.wikipedia.org/wiki/Software_development) under which

requirements and solutions evolve through the collaborative effort of [self-organizing](https://en.wikipedia.org/wiki/Self-organization#Human_society) andcross functionalteams and their [customer(s)](https://en.wikipedia.org/wiki/Customer)[/end user(s).](https://en.wikipedia.org/wiki/End_user) it advocates adaptive planning, evolutionary development, [empirical knowledge,](https://en.wikiquote.org/wiki/Empirical_knowledge) and [continual improvement,](https://en.wikipedia.org/wiki/Continual_improvement_process) and it encourages rapid and flexible response to change.

The term agile (sometimes written agile) was popularized, in this context, by the [manifesto for agile software development.](https://en.wikipedia.org/wiki/Agile_software_development#The_Agile_Manifesto) the values and principles espoused in this manifesto were derived from and underpin a broad range of [software development frameworks,](https://en.wikipedia.org/wiki/Software_development_process) including [scrum](https://en.wikipedia.org/wiki/Scrum_(software_development)) and KANBAN [.](https://en.wikipedia.org/wiki/Kanban_(development))

There is significant anecdotal evidence that adopting agile practices and values improves the agility of software professionals, teams and organizations; however, some empirical studies have found no scientific evidence.

AGILESOFTWARE DEVELOPMENT PRINCIPLES

The manifesto for agile software development is based on twelve principles:

1. Customer satisfaction by early and continuous delivery of valuable software.
2. Welcome changing requirements, even in late development.
3. Deliver working software frequently (weeks rather than months)
4. Close, daily cooperation between business people and developers
5. Projects are built around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)
7. Working software is the primary measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. Best architectures, requirements, and designs emerge from self-organizing teams
12. Regularly, the team reflects on how to become more effective, and adjusts accordingly

**FEATURES OF AGILE PRINCIPLES:**

ITERATIVE, INCREMENTAL AND EVOLUTIONARY

Most agile development methods break product development work into small increments that minimize the amount of up-front planning and design. iterations, or sprints, are short time frames [(time boxes)](https://en.wikipedia.org/wiki/Timeboxing) that typically last from one to four weeks. each iteration involves a [cross-functional team](https://en.wikipedia.org/wiki/Cross-functional_team) working in all functions: [planning,](https://en.wikipedia.org/wiki/Project_planning) [analysis,](https://en.wikipedia.org/wiki/Requirements_analysis) [design,](https://en.wikipedia.org/wiki/Software_design) [coding,](https://en.wikipedia.org/wiki/Computer_programming) [unit testing,](https://en.wikipedia.org/wiki/Unit_testing) and [acceptance testing.](https://en.wikipedia.org/wiki/Acceptance_testing) at the end of the iteration a working product is demonstrated to stakeholders. this minimizes overall risk and allows the product to adapt to changes quickly. an iteration might not add enough functionality to warrant a market release, but the goal is to have an available release (with minimal [bugs)](https://en.wikipedia.org/wiki/Software_bug) at the end of each iteration. multiple iterations might be required to release a product or new features. working software is the primary measure of progress.

EFFICIENT AND FACE-TO-FACE COMMUNICATION

The principle of [co-location](https://en.wikipedia.org/wiki/Colocation_(business)) is that co-workers on the same team should be situated together to better establish the identity as a team and to improve communication. this enables [face-to-face interaction,](https://en.wikipedia.org/wiki/Face-to-face_interaction) ideally in front of a whiteboard, that reduces the cycle time typically taken when questions and answers are mediated through phone, persistent chat, wiki, or email.

no matter which development method is followed, every team should include a [customer representative](https://en.wikipedia.org/wiki/Customer_representative) ("product owner" in [scrum)](https://en.wikipedia.org/wiki/Scrum_(software_development)). this person is agreed by stakeholders to act on their behalf and makes a personal commitment to being available for developers to answer questions throughout the iteration. at the end of each iteration, stakeholders and the customer representative review progress and re-evaluate priorities with a view to optimizing the [return on investment](https://en.wikipedia.org/wiki/Rate_of_return) (ROI) and ensuring alignment with customer needs and company goals.

In agile software development, an **information radiator** is a (normally large) physical display located prominently near the development team, where passers-by can see it. it presents an up-to date summary of the product development status. a [build light indicator](https://en.wikipedia.org/wiki/Build_light_indicator) may also be used to inform a team about the current status of their product development.

VERY SHORT FEEDBACK LOOP AND ADAPTATION CYCLE

A common characteristic in agile software development is the [daily stand-up](https://en.wikipedia.org/wiki/Stand-up_meeting) (also known as the daily scrum). in a brief session, team members report to each other what they did the previous day toward their team's iteration goal, what they intend to do today toward the goal, and any roadblocks or impediments they can see to the goal.

QUALITY FOCUS

Specific tools and techniques, such as [continuous integration,](https://en.wikipedia.org/wiki/Continuous_integration) automated [unit testing,](https://en.wikipedia.org/wiki/Unit_testing) [pair programming,](https://en.wikipedia.org/wiki/Pair_programming) [test-driven development,](https://en.wikipedia.org/wiki/Test-driven_development) [design patterns,](https://en.wikipedia.org/wiki/Software_design_pattern) [behavior-driven development,](https://en.wikipedia.org/wiki/Behavior-driven_development) [domain](https://en.wikipedia.org/wiki/Domain-driven_design) https://en.wikipedia.org/wiki/Domain-driven\_design [driven design,](https://en.wikipedia.org/wiki/Domain-driven_design) [code refactoring](https://en.wikipedia.org/wiki/Code_refactoring) and other techniques are often used to improve quality and enhance product development agility. this is predicated on designing and building quality in from the beginning and being able to demonstrate software for customers at any point, or at least at the end of every iteration.

## Technology Used:

### HTML 5:

**HTML 5** (formerly and commonly spelled **HTML5**) is a [software solution stack](https://en.wikipedia.org/wiki/Solution_stack) that defines the properties and behaviors of [web page](https://en.wikipedia.org/wiki/Web_page) [content](https://en.wikipedia.org/wiki/Web_content) by implementing a [markup](https://en.wikipedia.org/wiki/Markup_language) based [pattern](https://en.wikipedia.org/wiki/Software_design_pattern) to it. HTML 5 includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the markup available for documents, and introduces markup and [application programming interfaces](https://en.wikipedia.org/wiki/Application_programming_interface) (APIs) for complex [web applications](https://en.wikipedia.org/wiki/Web_application).[[11]](https://en.wikipedia.org/wiki/HTML5#cite_note-HTML5diffHTML4-12) For the same reasons, HTML 5 is also [a candidate for cross-platform mobile applications](https://en.wikipedia.org/wiki/HTML5_in_mobile_devices), because it includes features designed with low-powered devices in mind. Many new [syntactic](https://en.wikipedia.org/wiki/Syntax_(programming_languages)) features are included. To natively include and handle [multimedia](https://en.wikipedia.org/wiki/Multimedia) and [graphical](https://en.wikipedia.org/wiki/2D_computer_graphics) content.

**FEATURES**

HTML 5 introduces [elements](https://en.wikipedia.org/wiki/HTML_element) and attributes that reflect typical usage on modern [websites](https://en.wikipedia.org/wiki/Website). Some of them are semantic replacements for common uses of generic block (<div>) and inline (<span>) elements, for example <nav> (website navigation block), <footer> (usually referring to bottom of web page or to last lines of HTML code), or <audio> and <video> instead of <object>. Some deprecated elements from [HTML 4.01](https://en.wikipedia.org/wiki/HTML_4.01) have been dropped, including purely presentational elements such as <font> and <center>, whose effects have long been superseded by the more capable [Cascading Style Sheets](https://en.wikipedia.org/wiki/Cascading_Style_Sheets). There is also a renewed emphasis on the importance of [DOM scripting](https://en.wikipedia.org/wiki/DOM_scripting) in Web behavior.

The HTML 5 syntax is no longer based on [SGML](https://en.wikipedia.org/wiki/Standard_Generalized_Markup_Language)despite the similarity of its markup. It has, however, been designed to be backward-compatible with common parsing of older versions of HTML. It comes with a new introductory line that looks like an SGML [document type declaration](https://en.wikipedia.org/wiki/Document_type_declaration), <!DOCTYPE html>, which triggers the standards-compliant [rendering mode](https://en.wikipedia.org/wiki/Quirks_mode).

### JAVA SCRIPT:

**JavaScript**, often abbreviated as **JS**, is a [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), [interpreted](https://en.wikipedia.org/wiki/Interpreted_language) [programming language](https://en.wikipedia.org/wiki/Programming_language) that conforms to the [ECMAScript](https://en.wikipedia.org/wiki/ECMAScript) specification. JavaScript has [curly-bracket syntax](https://en.wikipedia.org/wiki/List_of_programming_languages_by_type#Curly-bracket_languages), [dynamic typing](https://en.wikipedia.org/wiki/Dynamic_programming_language), [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming) [object-orientation](https://en.wikipedia.org/wiki/Object-oriented_programming), and [first-class functions](https://en.wikipedia.org/wiki/First-class_function).

Alongside [HTML](https://en.wikipedia.org/wiki/HTML) and [CSS](https://en.wikipedia.org/wiki/CSS), JavaScript is one of the core technologies of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web). JavaScript enables interactive [web pages](https://en.wikipedia.org/wiki/Web_page) and is an essential part of [web applications](https://en.wikipedia.org/wiki/Web_application). The vast majority of [websites](https://en.wikipedia.org/wiki/Website) use it, and major [web browsers](https://en.wikipedia.org/wiki/Web_browser) have a dedicated [JavaScript engine](https://en.wikipedia.org/wiki/JavaScript_engine) to execute it.

As a multi-paradigm language, JavaScript supports [event-driven](https://en.wikipedia.org/wiki/Event-driven_programming), [functional](https://en.wikipedia.org/wiki/Functional_programming), and [imperative](https://en.wikipedia.org/wiki/Imperative_programming) (including [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) and [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming)) [programming styles](https://en.wikipedia.org/wiki/Programming_paradigm). It has [APIs](https://en.wikipedia.org/wiki/Application_programming_interface) for working with text, [arrays](https://en.wikipedia.org/wiki/Array_data_type), dates, [regular expressions](https://en.wikipedia.org/wiki/Regular_expression), and the [DOM](https://en.wikipedia.org/wiki/Document_Object_Model), but the language itself does not include any [I/O](https://en.wikipedia.org/wiki/Input/output), such as [networking](https://en.wikipedia.org/wiki/Computer_network), [storage](https://en.wikipedia.org/wiki/Data_storage), or [graphics](https://en.wikipedia.org/wiki/Computer_graphics) facilities. It relies upon the host environment in which it is embedded to provide these features.

Initially only implemented [client-side](https://en.wikipedia.org/wiki/Client-side) in web browsers, JavaScript engines are now embedded in many other types of host software, including [server-side](https://en.wikipedia.org/wiki/Server-side) in web servers and databases, and in non-web programs such as word processors and [PDF](https://en.wikipedia.org/wiki/Portable_Document_Format) software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.

Although there are similarities between JavaScript and [Java](https://en.wikipedia.org/wiki/Java_(programming_language)), including language name, [syntax](https://en.wikipedia.org/wiki/Syntax_(programming_languages)), and respective [standard libraries](https://en.wikipedia.org/wiki/Standard_library), the two languages are distinct and differ greatly in design. JavaScript was influenced by programming languages such as [Self](https://en.wikipedia.org/wiki/Self_(programming_language)) and [Scheme](https://en.wikipedia.org/wiki/Scheme_(programming_language)).

### AJAX:

AJAX stands for **A**synchronous **Ja**vaScript and **X**ML. AJAX is a new technique for creating better, faster, and more interactive web applications with the help of XML, HTML, CSS, and Java Script.

* Ajax uses XHTML for content, CSS for presentation, along with Document Object Model and JavaScript for dynamic content display.
* Conventional web applications transmit information to and from the sever using synchronous requests. It means you fill out a form, hit submit, and get directed to a new page with new information from the server.
* With AJAX, when you hit submit, JavaScript will make a request to the server, interpret the results, and update the current screen. In the purest sense, the user would never know that anything was even transmitted to the server.
* XML is commonly used as the format for receiving server data, although any format, including plain text, can be used.
* AJAX is a web browser technology independent of web server software.
* A user can continue to use the application while the client program requests information from the server in the background.
* Intuitive and natural user interaction. Clicking is not required, mouse movement is a sufficient event trigger.

### JAVA SERVER PAGE:

**JavaServer Pages (JSP)** is a technology for developing Webpages that supports dynamic content. This helps developers insert java code in HTML pages by making use of special JSP tags, most of which start with <% and end with %>.

A JavaServer Pages component is a type of Java servlet that is designed to fulfill the role of a user interface for a Java web application. Web developers write JSPs as text files that combine HTML or XHTML code, XML elements, and embedded JSP actions and commands.

Using JSP, you can collect input from users through Webpage forms, present records from a database or another source, and create Webpages dynamically.

JSP tags can be used for a variety of purposes, such as retrieving information from a database or registering user preferences, accessing JavaBeans components, passing control between pages, and sharing information between requests, pages etc.

### ORACLE:

An **Oracle** **database** is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information. A database server is the key to solving the problems of information management. In general, a [**server**](https://docs.oracle.com/cd/B19306_01/server.102/b14220/glossary.htm#i432724) reliably manages a large amount of data in a multiuser environment so that many users can concurrently access the same data. All this is accomplished while delivering high performance. A database server also prevents unauthorized access and provides efficient solutions for failure recovery.  
Oracle Database is the first database designed for enterprise grid computing, the most flexible and cost effective way to manage information and applications. Enterprise grid computing creates large pools of industry-standard, modular storage and servers. With this architecture, each new system can be rapidly provisioned from the pool of components. There is no need for peak workloads, because capacity can be easily added or reallocated from the resource pools as needed.  
The database has **logical structures** and **physical structures**. Because the physical and logical structures are separate, the physical storage of data can be managed without affecting the access to logical storage structures.

### CSS:

**Cascading Style Sheets**, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.  
CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand.

### BOOTSTRAP:

**Bootstrap** is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source) [CSS framework](https://en.wikipedia.org/wiki/CSS_framework) directed at responsive, mobile-first [front-end web development](https://en.wikipedia.org/wiki/Front-end_web_development). It contains [CSS](https://en.wikipedia.org/wiki/CSS)- and (optionally) [JavaScript](https://en.wikipedia.org/wiki/JavaScript)-based design templates for [typography](https://en.wikipedia.org/wiki/Web_design#Typography), [forms](https://en.wikipedia.org/wiki/Form_(HTML)), [buttons](https://en.wikipedia.org/wiki/Button_(computing)#HTML), [navigation](https://en.wikipedia.org/wiki/Web_navigation#Local_website_navigation) and other interface components.

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to [web apps](https://en.wikipedia.org/wiki/Web_Apps)). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all [HTML elements](https://en.wikipedia.org/wiki/HTML_element). The end result is a uniform appearance for prose, tables and form elements across [web browsers](https://en.wikipedia.org/wiki/Web_browser). In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.

Bootstrap also comes with several JavaScript components in the form of [jQuery](https://en.wikipedia.org/wiki/JQuery) plugins. They provide additional user interface elements such as [dialog boxes](https://en.wikipedia.org/wiki/Dialog_box), [tooltips](https://en.wikipedia.org/wiki/Tooltip), and carousels. Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields.

### JAVA SPRING:

The **Spring Framework** is an [application framework](https://en.wikipedia.org/wiki/Application_framework) and [inversion of control](https://en.wikipedia.org/wiki/Inversion_of_control) [container](https://en.wikipedia.org/wiki/Servlet_container) for the [Java platform](https://en.wikipedia.org/wiki/Java_platform). The framework's core features can be used by any Java application, but there are extensions for building web applications on top of the [Java EE](https://en.wikipedia.org/wiki/Java_EE) (Enterprise Edition) platform. Although the framework does not impose any specific [programming model](https://en.wikipedia.org/wiki/Programming_model), it has become popular in the Java community as an addition to, or even replacement for the [Enterprise JavaBeans](https://en.wikipedia.org/wiki/Enterprise_JavaBeans) (EJB) model. The Spring Framework is [open source](https://en.wikipedia.org/wiki/Open-source_software).

**Modules**

The Spring Framework includes several modules that provide a range of services:

* Spring Core Container: this is the base module of Spring and provides spring containers (BeanFactory and ApplicationContext).[[11]](https://en.wikipedia.org/wiki/Spring_Framework#cite_note-11)
* [Aspect-oriented programming](https://en.wikipedia.org/wiki/Aspect-oriented_programming): enables implementing [cross-cutting concerns](https://en.wikipedia.org/wiki/Cross-cutting_concern).
* [Authentication](https://en.wikipedia.org/wiki/Authentication) and [authorization](https://en.wikipedia.org/wiki/Authorization): configurable security processes that support a range of standards, protocols, tools and practices via the [Spring Security](https://en.wikipedia.org/wiki/Spring_Security) sub-project (formerly Acegi Security System for Spring).
* [Convention over configuration](https://en.wikipedia.org/wiki/Convention_over_configuration): a rapid application development solution for Spring-based enterprise applications is offered in the [Spring Roo](https://en.wikipedia.org/wiki/Spring_Roo) module
* [Data access](https://en.wikipedia.org/wiki/Data_access): working with [relational database management systems](https://en.wikipedia.org/wiki/RDBMS) on the Java platform using [Java Database Connectivity](https://en.wikipedia.org/wiki/Java_Database_Connectivity) (JDBC) and [object-relational mapping](https://en.wikipedia.org/wiki/Object-relational_mapping) tools and with [NoSQL](https://en.wikipedia.org/wiki/NoSQL) databases
* [Inversion of control](https://en.wikipedia.org/wiki/Inversion_of_control) container: configuration of application components and lifecycle management of Java objects, done mainly via [dependency injection](https://en.wikipedia.org/wiki/Dependency_injection)
* Messaging: configurative registration of message listener objects for transparent message-consumption from [message queues](https://en.wikipedia.org/wiki/Message_queue) via [Java Message Service](https://en.wikipedia.org/wiki/Java_Message_Service) (JMS), improvement of message sending over standard JMS APIs
* [Model–view–controller](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller): an [HTTP](https://en.wikipedia.org/wiki/HTTP)- and [servlet](https://en.wikipedia.org/wiki/Java_Servlet_API)-based framework providing hooks for extension and customization for web applications and [RESTful](https://en.wikipedia.org/wiki/REST) (representational state transfer) Web services.
* Remote access framework: configurative [remote procedure call](https://en.wikipedia.org/wiki/Remote_procedure_call) (RPC)-style [marshalling](https://en.wikipedia.org/wiki/Marshalling_(computer_science)) of Java objects over networks supporting [Java remote method invocation](https://en.wikipedia.org/wiki/Java_remote_method_invocation) (RMI), [CORBA](https://en.wikipedia.org/wiki/CORBA) (Common Object Request Broker Architecture) and [HTTP](https://en.wikipedia.org/wiki/HTTP)-based protocols including [Web services](https://en.wikipedia.org/wiki/Web_services) ([SOAP (Simple Object Access Protocol)](https://en.wikipedia.org/wiki/SOAP_(protocol)))
* [Transaction management](https://en.wikipedia.org/wiki/Transaction_processing): unifies several transaction management APIs and coordinates transactions for Java objects
* Remote management: configurative exposure and management of Java objects for local or remote configuration via [Java Management Extensions](https://en.wikipedia.org/wiki/Java_Management_Extensions) (JMX)
* [Testing](https://en.wikipedia.org/wiki/Software_testing): support classes for writing unit tests and integration tests.

### HIBERNATE FRAMEWORK:

**Hibernate ORM** (Hibernate in short) is an [object-relational mapping](https://en.wikipedia.org/wiki/Object-relational_mapping) tool for the [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) programming language. It provides a [framework](https://en.wikipedia.org/wiki/Software_framework) for mapping an [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) domain model to a [relational database](https://en.wikipedia.org/wiki/Relational_database). Hibernate handles [object-relational impedance mismatch](https://en.wikipedia.org/wiki/Object-relational_impedance_mismatch) problems by replacing direct, [persistent](https://en.wikipedia.org/wiki/Persistence_(computer_science)) database accesses with high-level object handling functions.

Hibernate is [free software](https://en.wikipedia.org/wiki/Free_software) that is distributed under the [GNU Lesser General Public License](https://en.wikipedia.org/wiki/GNU_Lesser_General_Public_License) 2.1.

Hibernate's primary feature is mapping from Java classes to [database tables](https://en.wikipedia.org/wiki/Table_(database)), and mapping from Java data types to [SQL](https://en.wikipedia.org/wiki/SQL) data types. Hibernate also provides data query and retrieval facilities. It generates SQL calls and relieves the developer from the manual handling and object conversion of the result set.

**Mapping**

The mapping of Java classes to database tables is implemented by the configuration of an [XML](https://en.wikipedia.org/wiki/XML) file or by using [Java Annotations](https://en.wikipedia.org/wiki/Java_annotation). When using an XML file, Hibernate can [generate](https://en.wikipedia.org/wiki/Program_synthesis) skeleton [source code](https://en.wikipedia.org/wiki/Source_code) for the persistence classes. This is auxiliary when annotations are used. Hibernate can use the XML file or the Java annotations to maintain the [database schema](https://en.wikipedia.org/wiki/Database_schema).

There are provided facilities to arrange [one-to-many](https://en.wikipedia.org/wiki/One-to-many_(data_model)) and [many-to-many](https://en.wikipedia.org/wiki/Many-to-many_(data_model)) relationships between classes. In addition to managing associations between objects, Hibernate can also manage [reflexive](https://en.wikipedia.org/wiki/Reflexive_relation) associations wherein an object has a one-to-many relationship with other instances of the [class](https://en.wikipedia.org/wiki/Class_(computer_programming)) [type](https://en.wikipedia.org/wiki/Data_type).

Hibernate supports the mapping of custom value types. This makes the following scenarios possible:

* Overriding the default SQL type when mapping a column to a property.
* Mapping Java [Enums](https://en.wikipedia.org/wiki/Enumerated_type" \o "Enumerated type) to columns as though they were regular properties.
* Mapping a single property to multiple columns.

**Definition:** Objects in an [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) application follow [OOP](https://en.wikipedia.org/wiki/Object-oriented_programming) principles, while objects in the back-end follow [database normalization](https://en.wikipedia.org/wiki/Database_normalization) principles, resulting in different representation requirements. This problem is called "[object-relational impedance mismatch](https://en.wikipedia.org/wiki/Object-relational_impedance_mismatch)". Mapping is a way of resolving the object-relational impedance mismatch problem.

Mapping informs the ORM tool of what Java class object to store in which database table.

**Hibernate Query Language(HQL)**

Hibernate provides an [SQL](https://en.wikipedia.org/wiki/SQL) inspired language called [Hibernate Query Language](https://en.wikipedia.org/wiki/Hibernate_Query_Language) (HQL) for writing SQL-like queries against Hibernate's data objects. *Criteria Queries* are provided as an [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming)alternative to HQL. Criteria Query is used to modify the objects and provide the restriction for the objects. HQL (Hibernate Query Language) is the object-oriented version of SQL. It generates database independent queries so that there is no need to write database-specific queries. Without this capability, changing the database would require individual SQL queries to be changed as well, leading to maintenance issues.

**Persistence**

Hibernate provides transparent persistence for [Plain Old Java Objects](https://en.wikipedia.org/wiki/Plain_Old_Java_Object) (POJOs). The only strict requirement for a persistent class is a [no-argument constructor](https://en.wikipedia.org/wiki/Nullary_constructor), though not necessarily public. Proper behavior in some applications also requires special attention to the equals() and hashCode() methods in the object classes.[[1]](https://en.wikipedia.org/wiki/Hibernate_(framework)#cite_note-1) Hibernate recommends providing an identifier attribute, and this is planned to be a mandatory requirement in a future release.[[2]](https://en.wikipedia.org/wiki/Hibernate_(framework)#cite_note-2)

Collections of data objects are typically stored in Java collection classes, such as implementations of the Set and List interfaces. [Java generics](https://en.wikipedia.org/wiki/Generics_in_Java), introduced in Java 5, are also supported. Hibernate can be configured to [lazy load](https://en.wikipedia.org/wiki/Lazy_load) associated collections. Lazy loading is the default as of Hibernate 3.

Related objects can be configured to [*cascade*](https://en.wikipedia.org/wiki/Propagation_constraint) operations from one object to the other. For example, a parent Album class object can be configured to cascade its save and delete operations to its child Track class objects.

**Integration**

Hibernate can be used both in standalone [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) applications and in [Java EE](https://en.wikipedia.org/wiki/Java_EE) applications using [servlets](https://en.wikipedia.org/wiki/Java_Servlet), [EJB](https://en.wikipedia.org/wiki/Enterprise_JavaBeans) session beans, and [JBI](https://en.wikipedia.org/wiki/Java_Business_Integration) service components. It can also be included as a feature in other programming languages. For example, [Adobe](https://en.wikipedia.org/wiki/Adobe_Systems) integrated Hibernate into version 9 of [ColdFusion](https://en.wikipedia.org/wiki/ColdFusion) (which runs on J2EE app servers) with an abstraction layer of new functions and syntax added into [CFML](https://en.wikipedia.org/wiki/CFML).

**Entities and Components**

In Hibernate [jargon](https://en.wikipedia.org/wiki/Jargon), an *entity* is a stand-alone object in Hibernate's [persistent](https://en.wikipedia.org/wiki/Persistence_(computer_science)) mechanism which can be manipulated independently of other objects. In contrast, a *component* is subordinate to an entity and can be manipulated only with respect to that entity. For example, an Album object may represent an entity; but the Tracks object associated with the Album objects would represent a *component* of the Album entity, if it is assumed that Tracks can only be saved or retrieved from the database through the Album object. Unlike J2EE, Hibernate can switch databases.

**Application Programming Interface**

The Hibernate API is provided in the [Java package](https://en.wikipedia.org/wiki/Java_package) org.hibernate.

org.hibernate.SessionFactory interface

org.hibernate.Session interface

The org.hibernate.Session interface[[10]](https://en.wikipedia.org/wiki/Hibernate_(framework)" \l "cite_note-10) represents a Hibernate session, i.e., the main point of the manipulation performed on the database entities. The latter activities include (among the other things) managing the persistence state ([transient](https://en.wikipedia.org/wiki/Transient_(computer_programming)), [persisted](https://en.wikipedia.org/wiki/Java_Persistence_API), detachedof the objects, fetching the persisted ones from the database and the management of the transaction demarcation

A session is intended to last as long as the logical transaction on the database. Due to the latter feature, Session implementations are not expected to be thread safe nor to be used by multiple clients.

### SERVER:

We have Used Tomcat Apache 8.5 server.

**Apache Tomcat**, often referred to as **Tomcat Server**, is an open-source [Java Servlet Container](https://en.wikipedia.org/wiki/Servlet_container) developed by the [Apache Software Foundation](https://en.wikipedia.org/wiki/Apache_Software_Foundation) (ASF). Tomcat implements several [Java EE](https://en.wikipedia.org/wiki/Java_Platform,_Enterprise_Edition) specifications including [Java Servlet](https://en.wikipedia.org/wiki/Java_Servlet), [JavaServer Pages](https://en.wikipedia.org/wiki/JavaServer_Pages" \o "JavaServer Pages) (JSP), [Java EL](https://en.wikipedia.org/wiki/Unified_Expression_Language), and [WebSocket](https://en.wikipedia.org/wiki/WebSocket" \o "WebSocket), and provides a "pure [Java](https://en.wikipedia.org/wiki/Java_(programming_language))" [HTTP](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol) [web server](https://en.wikipedia.org/wiki/Web_server) environment in which [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) code can run.

Tomcat is developed and maintained by an open community of developers under the auspices of the [Apache Software Foundation](https://en.wikipedia.org/wiki/Apache_Software_Foundation), released under the [Apache License](https://en.wikipedia.org/wiki/Apache_License) 2.0 license, and is [open-source software](https://en.wikipedia.org/wiki/Open-source_software).

**Components**

### Catalina

Catalina is Tomcat's [servlet container](https://en.wikipedia.org/wiki/Web_container). Catalina implements [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems)'s specifications for [servlet](https://en.wikipedia.org/wiki/Java_servlet) and JavaServer Pages (JSP). In Tomcat, a Realm element represents a "database" of usernames, passwords, and roles (similar to Unix groups) assigned to those users. Different implementations of Realm allow Catalina to be integrated into environments where such authentication information is already being created and maintained, and then use that information to implement Container Managed Security as described in the Servlet Specification.[[2]](https://en.wikipedia.org/wiki/Apache_Tomcat#cite_note-2)

### Coyote

Coyote is a Connector component for Tomcat that supports the HTTP 1.1 protocol as a web server. This allows Catalina, nominally a Java Servlet or JSP container, to also act as a plain web server that serves local files as HTTP documents.[[3]](https://en.wikipedia.org/wiki/Apache_Tomcat#cite_note-3) Coyote listens for incoming connections to the server on a specific [TCP](https://en.wikipedia.org/wiki/Transmission_Control_Protocol) port and forwards the request to the Tomcat Engine to process the request and send back a response to the requesting client. Another Coyote Connector, Coyote JK, listens similarly but instead forwards its requests to another web server, such as Apache, using the [JK protocol](https://en.wikipedia.org/wiki/Apache_JServ_Protocol).[[4]](https://en.wikipedia.org/wiki/Apache_Tomcat#cite_note-4) This usually offers better performance.[*[citation needed](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed" \o "Wikipedia:Citation needed)*]

Jasper

Jasper is Tomcat's JSP Engine. Jasper [parses](https://en.wikipedia.org/wiki/Parsing) [JSP files](https://en.wikipedia.org/wiki/JavaServer_Pages) to compile them into Java code as servlets (that can be handled by Catalina). At runtime, Jasper detects changes to JSP files and recompiles them.

As of version 5, Tomcat uses Jasper 2, which is an implementation of the Sun Microsystems's JSP 2.0 specification. From Jasper to Jasper 2, important features were added:

* JSP Tag library pooling - Each tag markup in JSP file is handled by a tag handler class. Tag handler class objects can be pooled and reused in the whole JSP servlet.
* Background JSP compilation - While recompiling modified JSP Java code, the older version is still available for server requests. The older JSP servlet is deleted once the new JSP servlet has finished being recompiled.
* Recompile JSP when included page changes - Pages can be inserted and included into a JSP at runtime. The JSP will not only be recompiled with JSP file changes but also with included page changes.
* JDT Java compiler - Jasper 2 can use the Eclipse JDT (Java Development Tools) Java compiler instead of [Ant](https://en.wikipedia.org/wiki/Apache_Ant) and [javac](https://en.wikipedia.org/wiki/Javac" \o "Javac).

Three new components were added with the release of Tomcat 7:

### Cluster

This component has been added to manage large applications. It is used for [load balancing](https://en.wikipedia.org/wiki/Load_balancing_(computing)) that can be achieved through many techniques. Clustering support currently requires the JDK version 1.5 or higher.

Web application

It has also added user- as well as system-based web applications enhancement to add support for deployment across the variety of environments. It also tries to manage sessions as well as applications across the network.

Tomcat is building additional components. A number of additional components may be used with Apache Tomcat. These components may be built by users should they need them or they can be downloaded from one of the mirrors.

**Features:**

Tomcat 7.x implements the Servlet 3.0 and JSP 2.2 specifications.[[7]](https://en.wikipedia.org/wiki/Apache_Tomcat#cite_note-7) It requires Java version 1.6, although previous versions have run on Java 1.1 through 1.5. Versions 5 through 6 saw improvements in [garbage collection](https://en.wikipedia.org/wiki/Garbage_collection_(computer_science)), JSP parsing, performance and scalability. Native wrappers, known as "Tomcat Native", are available for [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) and Unix for platform integration.

Tomcat 8.x implements the Servlet 3.1 and JSP 2.3 Specifications.[[8]](https://en.wikipedia.org/wiki/Apache_Tomcat#cite_note-8) Apache Tomcat 8.5.x is intended to replace 8.0.x and includes new features pulled forward from Tomcat 9.0.x. The minimum Java version and implemented specification versions remain unchanged.

### IDE:

We have Used Eclipse Oxygen IDE.

**Eclipse** is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) used in [computer programming](https://en.wikipedia.org/wiki/Computer_programming), and is the most widely used Java IDE.[[6]](https://en.wikipedia.org/wiki/Eclipse_(software)#cite_note-6) It contains a base [workspace](https://en.wikipedia.org/wiki/Workspace) and an extensible [plug-in](https://en.wikipedia.org/wiki/Plug-in_(computing)) system for customizing the environment. Eclipse is written mostly in [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) and its primary use is for developing Java applications, but it may also be used to develop applications in other [programming languages](https://en.wikipedia.org/wiki/Programming_language) via plug-ins, including [Ada](https://en.wikipedia.org/wiki/Ada_(programming_language)), [ABAP](https://en.wikipedia.org/wiki/ABAP), [C](https://en.wikipedia.org/wiki/C_(programming_language)), [C++](https://en.wikipedia.org/wiki/C%2B%2B), [C#](https://en.wikipedia.org/wiki/C_Sharp_(programming_language)), [Clojure](https://en.wikipedia.org/wiki/Clojure), [COBOL](https://en.wikipedia.org/wiki/COBOL), [D](https://en.wikipedia.org/wiki/D_(programming_language)), [Erlang](https://en.wikipedia.org/wiki/Erlang_(programming_language)), [Fortran](https://en.wikipedia.org/wiki/Fortran), [Groovy](https://en.wikipedia.org/wiki/Groovy_(programming_language)), [Haskell](https://en.wikipedia.org/wiki/Haskell_(programming_language)), [JavaScript](https://en.wikipedia.org/wiki/JavaScript), [Julia](https://en.wikipedia.org/wiki/Julia_(programming_language)),[Lasso](https://en.wikipedia.org/wiki/Lasso_(programming_language)), [Lua](https://en.wikipedia.org/wiki/Lua_(programming_language)), [NATURAL](https://en.wikipedia.org/wiki/Software_AG), [Perl](https://en.wikipedia.org/wiki/Perl), [PHP](https://en.wikipedia.org/wiki/PHP), [Prolog](https://en.wikipedia.org/wiki/Prolog), [Python](https://en.wikipedia.org/wiki/Python_(programming_language)), [R](https://en.wikipedia.org/wiki/R_(programming_language)), [Ruby](https://en.wikipedia.org/wiki/Ruby_(programming_language)) (including [Ruby on Rails](https://en.wikipedia.org/wiki/Ruby_on_Rails)framework), [Rust](https://en.wikipedia.org/wiki/Rust_(programming_language)), [Scala](https://en.wikipedia.org/wiki/Scala_(programming_language)), and [Scheme](https://en.wikipedia.org/wiki/Scheme_(programming_language)). It can also be used to develop documents with [LaTeX](https://en.wikipedia.org/wiki/LaTeX" \o "LaTeX) (via a TeXlipse plug-in) and packages for the software [Mathematica](https://en.wikipedia.org/wiki/Mathematica). Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++, and Eclipse PDT for PHP, among others.

The initial [codebase](https://en.wikipedia.org/wiki/Codebase) originated from [IBM VisualAge](https://en.wikipedia.org/wiki/IBM_VisualAge) The Eclipse [software development kit](https://en.wikipedia.org/wiki/Software_development_kit) (SDK), which includes the Java development tools, is meant for Java developers. Users can extend its abilities by installing plug-ins written for the Eclipse Platform, such as development toolkits for other programming languages, and can write and contribute their own plug-in modules. Since the introduction of the [OSGi](https://en.wikipedia.org/wiki/OSGi" \o "OSGi) implementation ([Equinox](https://en.wikipedia.org/wiki/Equinox_(OSGi))) in version 3 of Eclipse, plug-ins can be plugged-stopped dynamically and are termed (OSGI) bundles

Server Platform

Eclipse supports development for [Tomcat](https://en.wikipedia.org/wiki/Apache_Tomcat), [GlassFish](https://en.wikipedia.org/wiki/GlassFish" \o "GlassFish) and many other servers and is often capable of installing the required server (for development) directly from the IDE. It supports remote debugging, allowing a user to watch variables and step through the code of an application that is running on the attached server.

Web Tools Platform

The Eclipse Web Tools Platform (WTP) project is an extension of the Eclipse platform with tools for developing Web and Java EE applications. It includes source and graphical editors for a variety of languages, wizards and built-in applications to simplify development, and tools and APIs to support deploying, running, and testing apps.

# Assumptions, Dependencies and Risks

## Assumptions:

* Once the application is built the Vehicle Reservation System will manually load the user list into the system.
* There will be no concurrent users for the application. The Vehicle profile will be maintained by the Admin.
* Any changes to the design mentioned in the BRD would be taken as a Change Request.

## Dependencies:

* Availability of database design characteristics & access shall be defined.
* Resource availability throughout each of the sprint

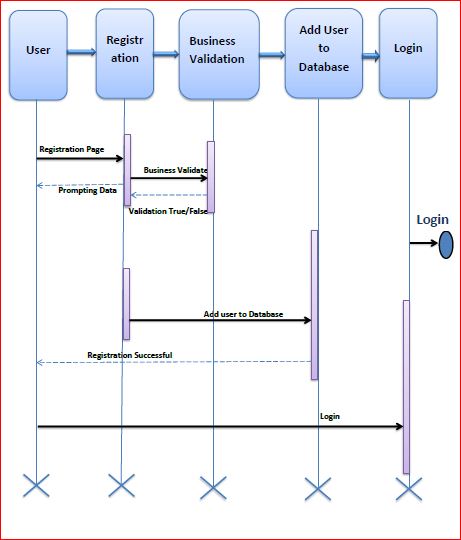
# Schematic Diagram



# Use Case Realization

## Use Case DiagramC:\Users\761231\Downloads\use_case_diagram(VRS).jpg

### User Registration



### Functional Requirement

. Login Page checks the authentication of the user/admin and directs the user to the respective page. Admin will provided with Add Vehicle, Details, and Search Vehicles.

### UI Validations

* User Id and Password fields are mandatory.
* Invalid User Id and / or password should not allow the user to get into the system.

### UI Controls

UI for Admin Login Page

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | UI Component | Type | Name | Details |
| 1 | Admin Id | Text Field | username | To enter Admin Id |
| 2 | Password | Text Field | password | To enter password |

UI for Admin Registration Page

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | UI Component | Type | Name | Details |
| 1 | Admin Id | Text Field | userId | To enter Admin Id |
| 2 | First Name | Text Field | firstName | To enter first name |
| 3 | Last Name | Text Field | lastName | To enter last name |
| 4 | Age | Text Field | age | To enter age |
| 5 | Gender | Radio | gender | To enter gender |
| 6 | Contact Number | Number | contact | To enter contact |
| 7 | Email id | Email | emailId | To enter Email id |
| 8 | Password | Password | password | To enter password |
| 9 | Branch | Text Field | branch | To enter Branch Name |

UI for Create Vehicle Details Page

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | UI Component | Type | Name | Details |
| 1 | Vehicle No | Text Field | vehicleNo | To enter Vehicle No |
| 2 | Branch | Text Field | branch | To enter branch name |
| 3 | Vehicle Type | Text Field | vehicleType | To enter Vehicle Type |
| 4 | Insurance Expiry Date | Date | insuranceExpiryDate | To enter Insurance Expiry Date |
| 5 | Last Service Date | Date | lastServiceDate | To enter Last Service Date |
| 6 | Service Due Date | Date | serviceDueDate | To enter Service Due Date |

UI for Search Vehicle Details Page

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | UI Component | Type | Name | Details |
| 1 | Branch | Text Field | branch | To enter branch name |
| 2 | Vehicle Type | Text Field | vehicleType | To enter Vehicle Type |
| 3 | Last Service Month | Date | lastServiceDate | To enter Last Service Date |
| 4 | Service Due Month | Date | serviceDueDate | To enter Service Due Date |

UI for Update Vehicle Details Page

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | UI Component | Type | Name | Details |
| 1 | Vehicle No | Text Field(readOnly) | vehicleNo | To enter Vehicle No |
| 2 | Branch | Text Field | branch | To enter branch name |
| 3 | Vehicle Type | Text Field(readOnly) | vehicleType | To enter Vehicle Type |
| 4 | Insurance Expiry Date | Date | insuranceExpiryDate | To enter Insurance Expiry Date |
| 5 | Last Service Date | Date | lastServiceDate | To enter Last Service Date |
| 6 | Service Due Date | Date | serviceDueDate | To enter Service Due Date |

UI for Update Admin Details Page

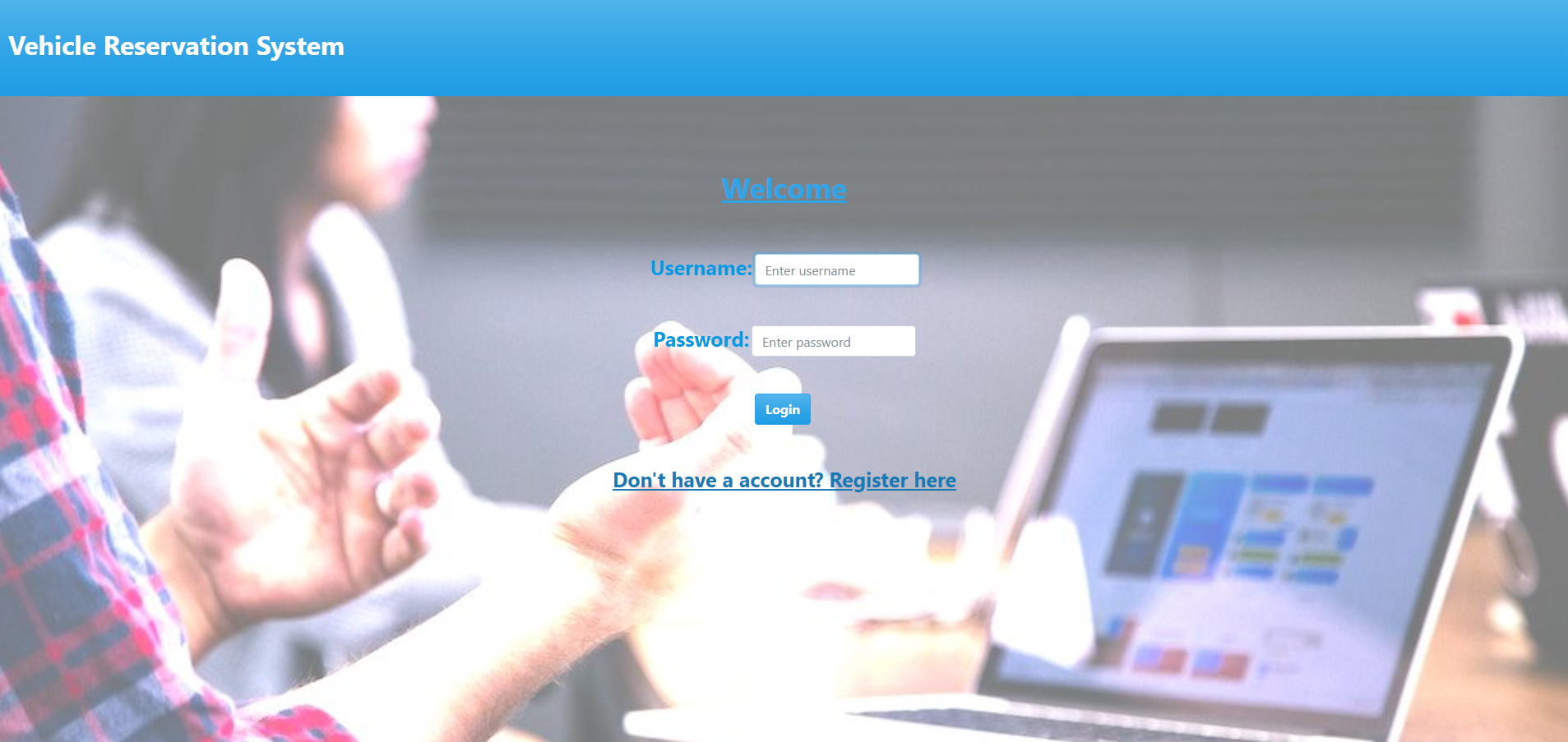
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | UI Component | Type | Name | Details |
| 1 | Admin Id | Text Field(readOnly) | userId | To enter Admin Id |
| 2 | First Name | Text Field(readOnly) | firstName | To enter first name |
| 3 | Last Name | Text Field(readOnly) | lastName | To enter last name |
| 4 | Age | Text Field | age | To enter age |
| 5 | Gender | Radio(readOnly) | gender | To enter gender |
| 6 | Contact Number | Number | contact | To enter contact |
| 7 | Email id | Email | emailId | To enter Email id |
| 8 | Password | Password | password | To enter password |
| 9 | Branch | Text Field | branch | To enter Branch Name |

### UI Methods Definition

|  |  |  |
| --- | --- | --- |
| **Method Name** | **Parameter** | **Details** |
| Sign Button\_Click | object sender, EventArgs e | Allows the user to login into the System and redirects to Home Page. |
| Registration Submit Button\_Click | object sender, EventArgs e | Allows the user to insert the User personal details. |
| Page\_Load | None | None |
| Create Button\_Click | object sender, EventArgs e | Allows the admin to create vehicle details. |
| Search Button\_Click | object sender, EventArgs e | Allows the user to Search particular Vehicle. |
| List Button\_Click | object sender, EventArgs e | Redirects to the Vehicle List Page. |
| Edit Button\_Click | object sender, EventArgs e | Allows the admin to update Vehicle details. |
| Delete Button\_Click | object sender, EventArgs e | Allows the admin to delete selected vehicle details. |
| Send-email Button\_Click | object sender, EventArgs e | Sends notification as email to corresponding admin. |
| Log-Out Button\_Click | object sender, EventArgs e | Ends the Session and redirects to the Login Page. |

### UI Design

### e.1 Login Page



### e.2 rdtd

e.3 dfesrser

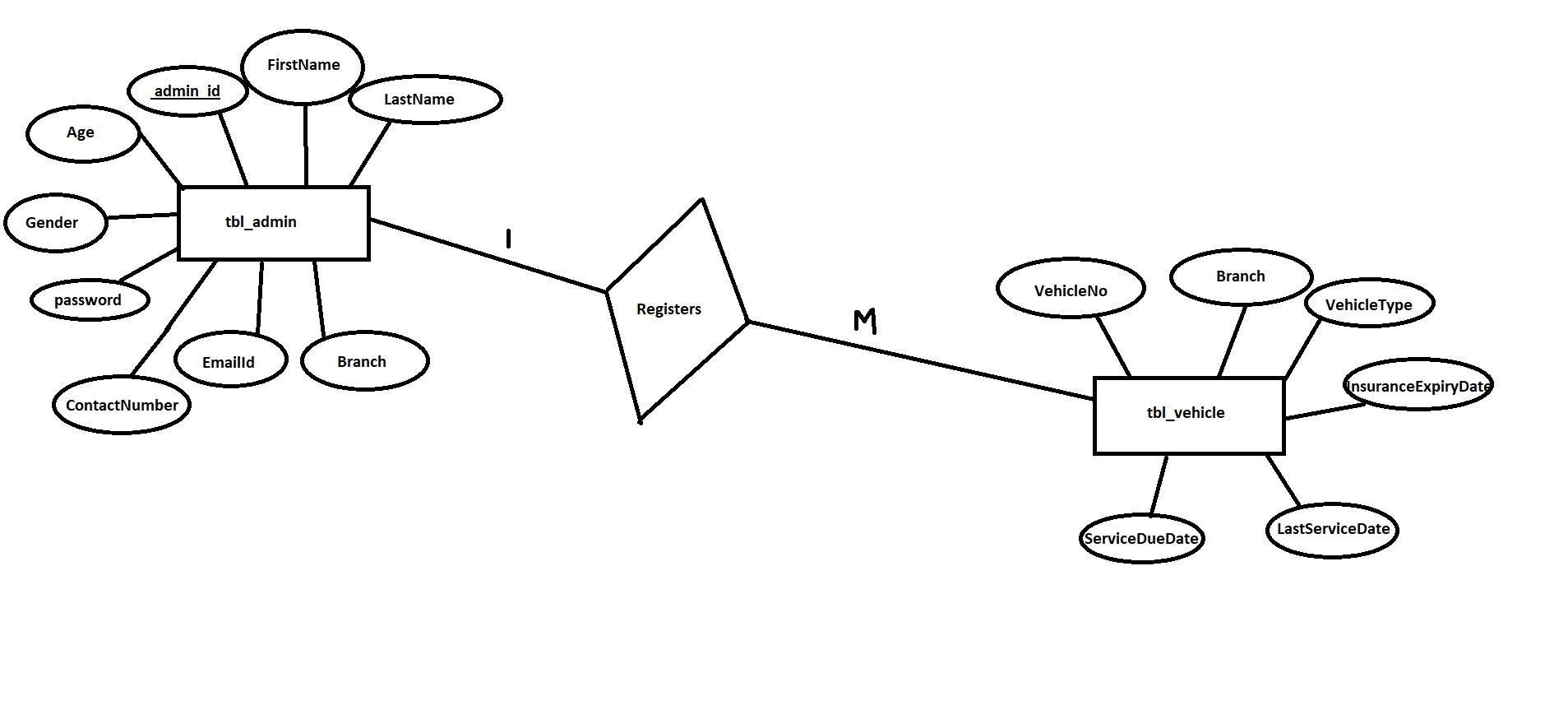
e.4

e.5

# Database Design

## Data Model

### The following is a schematic view of the database design



## Tables Structure

**User Information Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Data Type** | **Possible Values** |
| Employee Id | Text(6) | Alphanumeric |  |
| First Name | Text(50) | Alphabetic |  |
| Last Name | Text(50) | Alphabetic |  |
| Age | Numeric(2) | Numeric |  |
| Gender | Drop Down | NA | Male,Female |
| Contact Number | Text(10) | Numeric |  |
| Email ID | Text(50) | Alphanumeric |  |
| Password | Text(15) | Alphanumeric |  |
| Branch | Text(5) | Alphanumeric |  |

**Vehicle Information Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Data Type** | **Possible Values** |
| Vehicle No | Text(10) | Alphanumeric |  |
| Branch | Text(5) | Alphabetic |  |
| Vehicle Type | Text(15) | Alphabetic |  |
| Insurance Expiry Date | Text(10) | MM-DD-CCYY |  |
| Last Serviced Date | Text(10) | MM-DD-CCYY |  |
| Service Due Date | Text(10) | MM-DD-CCYY |  |

# Area of Improvement

# Change Log

*Please note that this table needs to be maintained even if a Configuration Management tool is used.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version Number | Changes made | | | |
| V1.0 | *<First version>* | | | |
| V1.1 | *<If the change details are not explicitly documented in the table below, reference should be provided here>* | | | |
| Page no | Changed by | Effective date | Changes effected |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| V1.2 | *<If the change details are not explicitly documented in the table below, reference should be provided here>* | | | |
| Page no | Changed by | Effective date | Changes effected |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |