# **Incident Management Report System - Software Requirements Specification**

## ****Description****

The Incident Management Software System is a software platform giving RideXpress employees the ability to add and edit the Car Inventory available and also the ability to create new incident reports for each car available in the inventory. The Incident Management Software System gives the internal employees the ability to keep track of the whole inventory in an easy platform.

## ****Purpose****

The purpose of this document is to provide a guide for developers and testers who are responsible for the development of this CRM platform. It provides the information necessary to design, develop, and test the software.

## ****Scope****

This document serves as a description of the required improvements to this CRM project. It contains a workflow description, functional and nonfunctional requirements, and appropriate diagrams to fully describe the system.

## ****System Overview****

This enterprise project consists of three major layers and also a Models layer that is used between the three major layers.

The **Presentation Layer** is the Web Portal, which is what the user sees and interacts with. In this case this would be the ASP.NET Web Forms application.

The **business logic layer** coordinates the application, processes the interaction by the user, and makes calculations and decisions based on the user’s input. The business layer processes and sends data between the Data Access Layer and the Presentation Layer. The business logic layer is represented by a class library and consists of classes that perform the mentioned calculations and transfer of data between layers.

The **data access layer** is responsible for sending data from the database to the business logic layer for calculations and vice-versa. The Data Access Layer is represented by a class library that has methods using ADO.NET Framework to connect to the SQL Server Database.

The **Model Layer** is used between all of the three major layers to transfer information between the three layers. The Model Layer consists of classes that represent the databases with properties that are columns in your database.

This architecture has been chosen to keep separate modules in order for scalability. The Presentation Layer should only interact with the Business Layer, and the Data Layer should only interact with the database and the Business Layer. The Business Layer is responsible for any calculations and processing of data between the Data Layer and the Presentation Layer.

## ****Use Cases****

1. **All Users (Personal Trainers)**
   1. **Car Inventory Management**
      1. Add a Car to the Inventory
      2. Update a Car in the Inventory
      3. Delete a Car from the Inventory
   2. **Incident Report Management**
      1. Create a new Incident Report for a Car
      2. Update an existing Incident Report for a Car
      3. Delete an Incident Report for a Car

## ****Use Case Diagrams****

https://preview-v2-app.revature.com/core/resources/ckfileupload?f_name=308a53c2423fb9714db9fc1e5d35f327.png

${{USE\_CASE\_DIAGRAM}}$

## ****Business Requirements****

**BR1: Ability to Add a Car to the Inventory:** After the User provides the required fields, the page should return to the Car Inventory Listing to view the added Car

**BR2: Ability to Edit an Already Existing Car in the Inventory:**After the User edits the required fields, the page should return to the Car Listing to view changes

**BR3: Ability to Delete an Existing Trainee:**The User should be able to confirm if he/she truly wants to delete the Trainee

**BR4: Ability to Add an Incident Report:** After the User provides the required fields, the page should return to the Incident Report listing to view the added Incident Report; Each Incident Report should contain the Date of the Incident, Name, Description, and the Car that the Report is tied to.

**BR5: Ability to Edit an Incident Report:** After the User provides the required fields, the page should return to the Incident Report listing to view the edited Incident Report.

**BR6: Ability to Delete an Incident Report:** The User should be able to confirm if he/she truly wants to delete the Incident Report

## ****Functional Requirements****

**FR1: All Fields in the Add Car form are required**

**FR2: All Fields in the Edit Car form are required**

**FR3: All Fields in the Add Incident Report form are required**

**FR4: All Fields in the Edit Incident Report form are required**