**Documentation for the Multi-System JavaFX Application**

**1. Overview**

This JavaFX application integrates two systems:

* **Accident Reporting**: Allows users to log details of an accident.
* **Vehicle Transfer Management**: Facilitates the process of transferring a vehicle from one owner to another.

The application includes a main menu for navigation between the two systems, each implemented as separate views. The application ensures a responsive and user-friendly interface, proper error handling, and modular code for maintainability.

**2. Components**

**a. Main Menu**

* **FXML File:** MainView.fxml
* **Controller:** MainController
* **Description:** Acts as the central navigation menu. Contains buttons to navigate to the Accident Reporting and Vehicle Transfer Management systems.

**b. Accident Reporting System**

* **FXML File:** AccidentView.fxml
* **Controller:** AccidentController
* **Model:** Accident
* **Description:** Handles input and management of accident details, generates a report, and allows users to review or close the application.

**c. Vehicle Transfer Management System**

* **FXML File:** TransferView.fxml
* **Controller:** TransferController
* **Model:** Owner, Vehicle
* **Description:** Manages the process of transferring ownership of a vehicle by recording relevant details.

**3. Implementation Detail**

* **Main Menu**

1. **FXML File:** MainView.fxmlUses a VBox to align two buttons vertically, each leading to a specific system. The layout is simple and responsive.
2. **Controller Class:** MainController

Handles navigation between views. Buttons load the respective FXML files and replace the current scene.

* **Accident Reporting System**

1. **FXML File: AccidentView.fxml**

* The form is divided into three panes:
  1. Vehicle Information Input (using a GridPane).
  2. Accident Details Input (hidden initially, using another GridPane).
  3. Accident Report Display (hidden initially, using a VBox).

1. **Controller Class: AccidentController**

* Handles the retrieval of vehicle data, input validation, accident report generation, and pane visibility toggling.
* **Vehicle Transfer System**

1. **FXML File: TransferView.fxml**

The UI for the Vehicle Transfer System uses a GridPane layout to organize input fields and buttons in a structured format.

* **Structure**:
  1. Input fields for vehicle and owner details.
  2. Input fields for new owner details.
  3. A button to confirm the transfer.
  4. An area of the details entered and the payment field
  5. Two buttons “pay” and “cancel”

1. **Controller Class: TransferController**

This class handles the business logic for the vehicle transfer system.

* **Responsibilities**:
  1. Validate user input.
  2. Simulate transferring ownership of the vehicle.
  3. Display confirmation or error messages

**4. Features**

* Modular FXML files ensure clear separation of views and functionality.
* Responsive layouts (VBox and GridPane) for better user experience.
* Simple navigation through the main menu.
* Proper error handling with user feedback.

**5. Improvements**

* Integrate a database for real-world VIN validation.
* Implement logging for error tracking.