



INFORMATICS  
INSTITUTE OF  
TECHNOLOGY

# **Informatics Institute of Technology**

## **Department of Computing**

BSc (Hons) Artificial Intelligence and Data Science

**Module: CM1601 Programming**

**Fundamentals**

**Module Coordinator: Mr. Iresh Bandara**

**Coursework Report**

Student Name : Damitha Udara Weerasinghe

RGU ID number : 2236765

IIT ID number : 20210669

## **1. Executive Summary**

A functional GUI was created using Java, JavaFX, and Scene Builder to manage the World Rally Cross Championship. The paper includes the corresponding Java code along with a brief explanation of each function. Flowcharts are also provided to illustrate the process of adding and deleting a driver's details. The paper contains J-Units, test cases, and test plans, as well as an assessment of the program's robustness and maintainability. Assumptions made during the program's development are included in the conclusion and reference list.

## 2. Contents Page

|   |    |
|---|----|
| 1. Executive Summary.....   | 2  |
| 2. Contents Page.....   | 3  |
| 3. Flow Charts .....  | 5  |
| Add driver details flowchart. ....  | 5  |
| Delete driver details flowchart. ....   | 6  |
| 4. Introduction to functions with code .....  | 7  |
| JavaFx.....   | 7  |
| Menu-View.fxml .....  | 7  |
| AddDriver-View.fxml .....   | 12 |
| DeleteDriver-View.fxml .....  | 16 |
| UpdateDriver-View.fxml.....   | 18 |
| Standing-View.fxml .....  | 23 |
| RandomRace-View.fxml .....  | 26 |
| RacTable-View.fxml .....  | 30 |
| Java Code .....   | 33 |
| Main. ....  | 33 |
| The code is for a JavaFX application that loads and presents a Graphical User Interface (GUI) from an FXML file. The GUI pertains to a program named "World Rally Cross Championship." The application spawns a window with the GUI, which the user cannot resize, and it sets up an event listener to handle requests for closing the window. Whenever the user attempts to close the window, the application prompts them with a confirmation message to ensure that they indeed want to exit the program. The method named exit() manages this confirmation message and proceeds to close the window if the user confirms their intent to exit. Finally, the main() method launches the JavaFX application. .... | 33 |
| Scene Controller. ....  | 35 |
| responsible for bridging the user interface and the application logic. It initializes the application, links UI components to data models, and sets up event handlers. It also creates helper methods to implement the logic for various application actions and behaviors.....   | 35 |
| Driver.....   | 40 |
| This code defines a model representing a racing driver, with properties such as name, age, team, car, and current points. Upon instantiation, the constructor initializes all these properties. Getter and setter methods are also provided for accessing and modifying each property. Moreover, the code contains an equals() method, which takes two driver objects as input and checks if they share the same name. This method can come in handy when dealing with collections that may contain duplicate drivers. ....   | 40 |
| Race. ....  | 42 |
| has two attributes, "date" and "location", which represent the date and location of the race. The class has a   |    |

constructor that takes in values for these attributes and initializes them. It also has getter methods for each attribute that allow for accessing the values of these attributes from other parts of the code. This class can be used to create and store race objects in a collection or database. ....42

**Add driver details.** .....43

**Delete driver details.** .....45

**Update driver details.**.....46

**View standing table.**.....48

Simulate a random race. ....49

**View race table.** .....51

**Save data and Load data** .....52

**5. J-Units, Test plan and test Cases** .....55

1. **Test Cases** .....55

**6. Robustness and the maintainability** .....57

**7. Conclusion and Assumptions** .....58

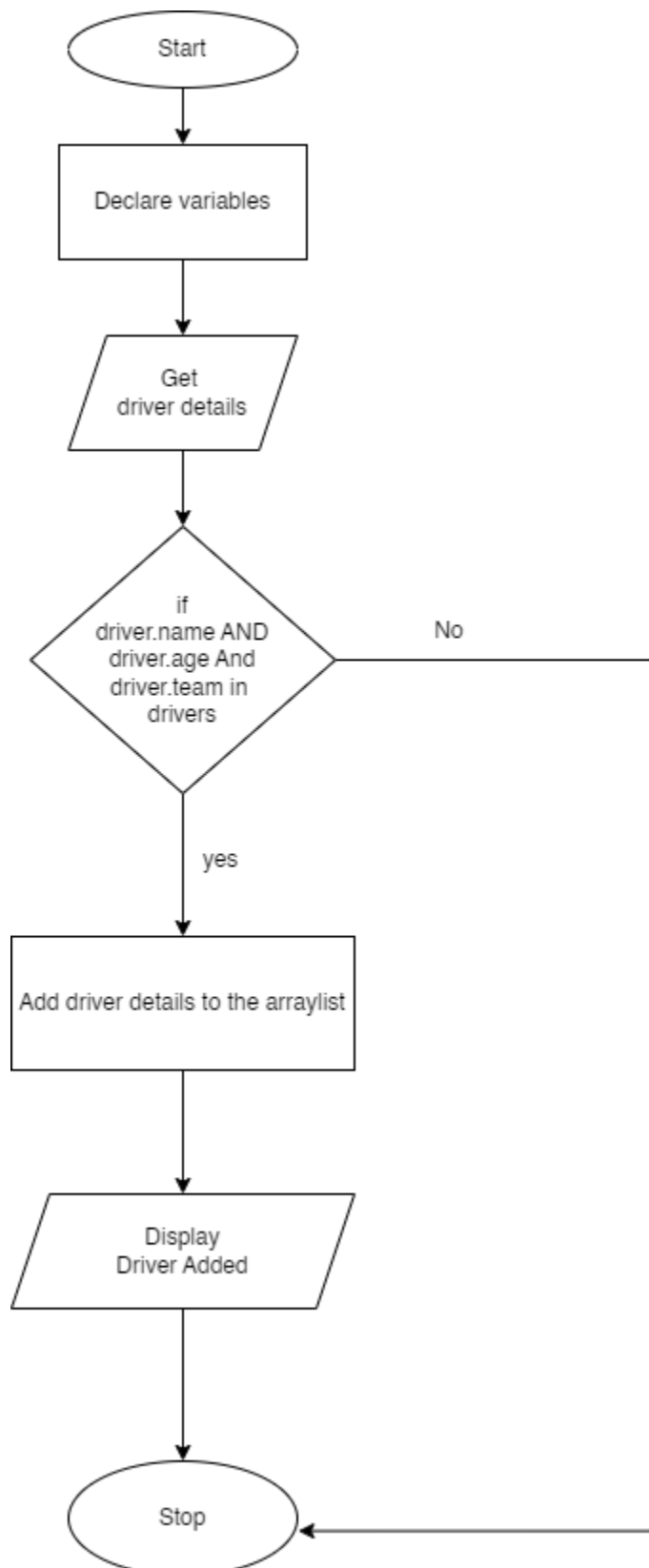
1. **Conclusion**.....58

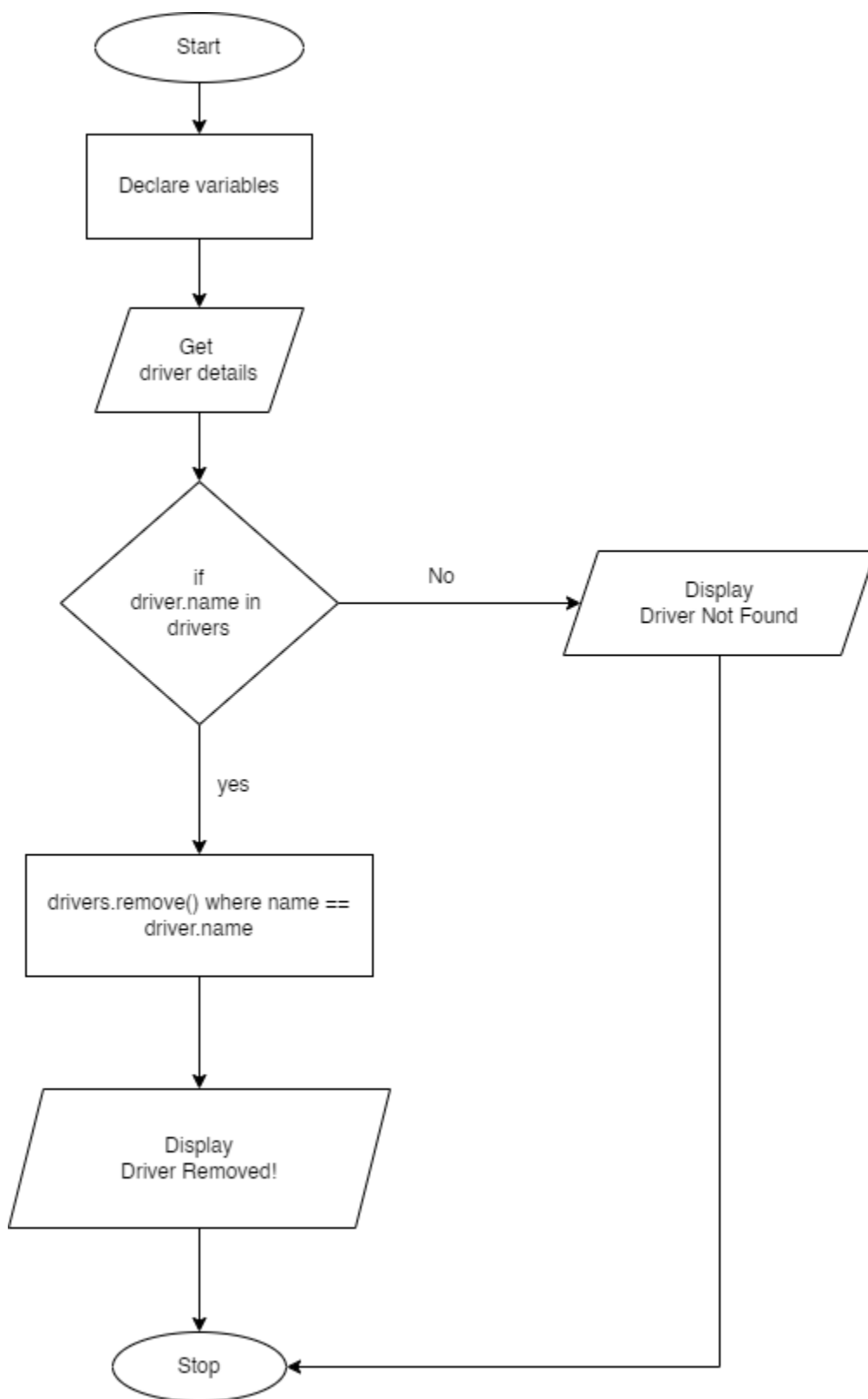
2. **Assumptions** .....58

**8. Reference list**.....59

### 3. Flow Charts

Add driver details flowchart.



**Delete driver details flowchart.**

## 4. Introduction to functions with code

### JavaFx

#### Menu-View.fxml

A main menu graphical user interface created for the WRC system.

```
<?xml version="1.0" encoding="UTF-8"?>

<?import java.lang.String?>
<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.BorderPane?>
<?import javafx.scene.layout.HBox?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<BorderPane fx:id="mainWindow" minWidth="-Infinity" prefHeight="730.0"
prefWidth="1280.0" styleClass="body" stylesheets="@styles/body.css"
xmlns="http://javafx.com/javafx/19" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.cm1601_cw_partb.SceneController">
    <top>
        <Pane prefHeight="97.0" prefWidth="1280.0" style="-fx-background-color:
#121E2A;" BorderPane.alignment="CENTER">
            <children>
                <Label alignment="CENTER" graphicTextGap="30.0" prefHeight="98.0"
prefWidth="1280.0" text="World Rally Cross Championship" textFill="#cbcaed">
                    <font>
                        <Font name="HelveticaRounded-Bold" size="59.0" />
                    </font>
                    <graphic>
                        <ImageView fitHeight="73.0" fitWidth="111.0">
                            <image>
                                <Image url="@imgs/logo.png" />
                            </image>
                        </ImageView>
                    </graphic>
                </Label>
            </children>
        </Pane>
    </top>
    <left>
        <AnchorPane prefHeight="674.0" prefWidth="915.0" BorderPane.alignment="CENTER">
            <children>
                <VBox alignment="CENTER" layoutX="-1.0" prefHeight="634.0"
prefWidth="908.0" style="-fx-background-color: rgba(0, 0, 0, 0.8);">
                    <children>
                        <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToAddDriver" prefHeight="50.0"
prefWidth="800.0" styleClass="menu-button" stylesheets="@styles/button.css" text="Add
```

```

Driver">
    <VBox.margin>
        <Insets top="15.0" />
    </VBox.margin>
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <graphic>
        <ImageView fitHeight="30.0" fitWidth="30.0">
            <image>
                <Image url="@imgs/icons/addDriver.png" />
            </image>
        </ImageView>
    </graphic>
</Button>
<Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToUpdateDriver" prefHeight="50.0"
prefWidth="800.0" styleClass="menu-button" stylesheets="@styles/button.css"
text="Update Driver">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <VBox.margin>
        <Insets top="25.0" />
    </VBox.margin>
    <graphic>
        <ImageView fitHeight="30.0" fitWidth="30.0">
            <image>
                <Image url="@imgs/icons/updateDriver.png" />
            </image>
        </ImageView>
    </graphic>
</Button>
<Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToDeleteDriver" prefHeight="50.0"
prefWidth="800.0" styleClass="menu-button" stylesheets="@styles/button.css"
text="Delete Driver">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <VBox.margin>
        <Insets top="25.0" />
    </VBox.margin>
    <graphic>
        <ImageView fitHeight="30.0" fitWidth="30.0">
            <image>
                <Image url="@imgs/icons/delteDriver.png" />
            </image>
        </ImageView>
    </graphic>
</Button>
<Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToStandings" prefHeight="50.0"
prefWidth="800.0" styleClass="menu-button" stylesheets="@styles/button.css"
text="View Standings">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />

```



```

        </font>
        <VBox.margin>
            <Insets top="25.0" />
        </VBox.margin>
        <graphic>
            <ImageView fitHeight="35.0" fitWidth="35.0">
                <image>
                    <Image url="@imgs/icons/standings.png" />
                </image>
            </ImageView>
        </graphic>
    </Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToRandomRace" prefHeight="50.0"
prefWidth="800.0" styleClass="menu-button" stylesheets="@styles/button.css"
text="Simulate Random Race">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
        <VBox.margin>
            <Insets top="25.0" />
        </VBox.margin>
        <graphic>
            <ImageView fitHeight="35.0" fitWidth="35.0">
                <image>
                    <Image url="@imgs/icons/randomRace.png" />
                </image>
            </ImageView>
        </graphic>
    </Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToRaceTable" prefHeight="50.0"
prefWidth="800.0" styleClass="menu-button" stylesheets="@styles/button.css"
text="Race Table">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
        <VBox.margin>
            <Insets top="25.0" />
        </VBox.margin>
        <graphic>
            <ImageView fitHeight="35.0" fitWidth="35.0">
                <image>
                    <Image url="@imgs/icons/raceTable.png" />
                </image>
            </ImageView>
        </graphic>
    </Button>
    <HBox prefHeight="53.0" prefWidth="920.0">
        <VBox.margin>
            <Insets top="25.0" />
        </VBox.margin>
        <children>
            <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#saveData" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Save">
                <font>

```

```

        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <HBox.margin>
        <Insets left="59.0" />
    </HBox.margin>
    <styleClass>
        <String fx:value="menu-button" />
        <String fx:value="save-button" />
    </styleClass>
    <graphic>
        <ImageView fitHeight="30.0" fitWidth="30.0">
            <image>
                <Image url="@imgs/icons/save.png" />
            </image>
        </ImageView>
    </graphic>
</Button>
<Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#loadData" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Load">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <HBox.margin>
        <Insets left="101.0" />
    </HBox.margin>
    <styleClass>
        <String fx:value="menu-button" />
        <String fx:value="save-button" />
    </styleClass>
    <graphic>
        <ImageView fitHeight="44.0" fitWidth="44.0">
            <image>
                <Image url="@imgs/icons/load.png" />
            </image>
        </ImageView>
    </graphic>
</Button>
</children>
</HBox>
<Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#exit" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Exit">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <styleClass>
        <String fx:value="menu-button" />
        <String fx:value="exit-button" />
    </styleClass>
    <VBox.margin>
        <Insets top="16.0" />
    </VBox.margin>
    <graphic>
        <ImageView fitHeight="35.0" fitWidth="45.0">
            <image>
                <Image url="@imgs/icons/exit.png" />
            </image>
        </ImageView>
    </graphic>

```

```

        </image>
    </ImageView>
</graphic>
</Button>
</children>
</VBox>
</children>
</AnchorPane>
</left>
<right>
    <AnchorPane prefHeight="634.0" prefWidth="621.0" style="-fx-background-color:
    rgba(0, 0, 0, 0.9);" BorderPane.alignment="CENTER">
        <children>
            <VBox alignment="CENTER" layoutX="7.0" prefHeight="634.0"
            prefWidth="366.0">
                <children>
                    <Label alignment="CENTER" prefHeight="385.0" prefWidth="357.0"
                    text="Menu" textFill="WHITE">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="75.0" />
                        </font>
                        <padding>
                            <Insets top="100.0" />
                        </padding>
                    </Label>
                    <Label text="Devoloped by: Damitha Weerasinghe(2236765)"
                    textFill="WHITE">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="15.0" />
                        </font>
                        <VBox.margin>
                            <Insets top="204.0" />
                        </VBox.margin>
                    </Label>
                </children>
            </VBox>
        </children>
    </AnchorPane>
</right>
</BorderPane>

```

### AddDriver-View.fxml

```
<?xml version="1.0" encoding="UTF-8"?>

<?import java.lang.String?>
<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.BorderPane?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<BorderPane fx:id="mainWindow" minWidth="-Infinity" prefHeight="730.0"
prefWidth="1280.0" styleClass="body" stylesheets="@styles/body.css"
xmlns="http://javafx.com/javafx/19" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.cm1601_cw_partb.SceneController">
    <top>
        <Pane prefHeight="97.0" prefWidth="1280.0" style="-fx-background-color:
#121E2A;" BorderPane.alignment="CENTER">
            <children>
                <Label alignment="CENTER" graphicTextGap="30.0" prefHeight="98.0"
prefWidth="1280.0" text="World Rally Cross Championship" textFill="#cbcaed">
                    <font>
                        <Font name="HelveticaRounded-Bold" size="59.0" />
                    </font>
                    <graphic>
                        <ImageView fitHeight="73.0" fitWidth="111.0">
                            <image>
                                <Image url="@imgs/logo.png" />
                            </image>
                        </ImageView>
                    </graphic>
                </Label>
            </children>
        </Pane>
    </top>
    <left>
        <AnchorPane prefHeight="674.0" prefWidth="915.0" BorderPane.alignment="CENTER">
            <children>
                <VBox alignment="CENTER" prefHeight="634.0" prefWidth="908.0" style="-fx-
background-color: rgba(0, 0, 0, 0.8);">
                    <children>
                        <TextField fx:id="driverNameField" alignment="CENTER" maxHeight="-
Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="580.0" promptText="Name"
styleClass="text-field" stylesheets="@styles/text.css">
                            <font>
                                <Font name="HelveticaRounded-Bold" size="15.0" />
                            </font>
                            <VBox.margin>
                                <Insets bottom="25.0" />
                            </VBox.margin>
                        </TextField>
                    </children>
                </VBox>
            </children>
        </AnchorPane>
    </left>
</BorderPane>
```

```

        </VBox.margin>
    </TextField>
    <TextField fx:id="driverAgeField" alignment="CENTER" maxHeight="-Infinity"
maxWidth="-Infinity" prefHeight="44.0" prefWidth="580.0" promptText="Age"
styleClass="text-field" stylesheets="@styles/text.css">
        <font>
            <Font name="HelveticaRounded-Bold" size="15.0" />
        </font>
    </VBox.margin>
    <Insets bottom="25.0" />
</VBox.margin>
</TextField>
    <TextField fx:id="driverTeamField" alignment="CENTER" maxHeight="-Infinity"
maxWidth="-Infinity" prefHeight="44.0" prefWidth="580.0" promptText="Team"
styleClass="text-field" stylesheets="@styles/text.css">
        <font>
            <Font name="HelveticaRounded-Bold" size="15.0" />
        </font>
    </VBox.margin>
    <Insets bottom="25.0" />
</VBox.margin>
</TextField>
    <TextField fx:id="driverCarField" alignment="CENTER" maxHeight="-Infinity"
maxWidth="-Infinity" prefHeight="44.0" prefWidth="580.0" promptText="Car"
styleClass="text-field" stylesheets="@styles/text.css">
        <font>
            <Font name="HelveticaRounded-Bold" size="15.0" />
        </font>
    </VBox.margin>
    <Insets bottom="25.0" />
</VBox.margin>
</TextField>
    <TextField fx:id="driverPointsField" alignment="CENTER"
maxHeight="-Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="580.0"
promptText="Current Points" styleClass="text-field" stylesheets="@styles/text.css">
        <font>
            <Font name="HelveticaRounded-Bold" size="15.0" />
        </font>
    </VBox.margin>
    <Insets bottom="50.0" />
</VBox.margin>
</TextField>
    <Button fx:id="addDriverBtn" alignment="CENTER"
graphicTextGap="15.0" mnemonicParsing="false" onAction="#addDriver" prefHeight="50.0"
prefWidth="350.0" styleClass="menu-button" stylesheets="@styles/button.css"
text="Add">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
    </VBox.margin>
    <Insets bottom="30.0" />
</VBox.margin>
</Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToMenu" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Back">
        <font>

```

```

        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <styleClass>
        <String fx:value="menu-button" />
        <String fx:value="save-button" />
    </styleClass>
    <graphic>
        <ImageView fitHeight="30.0" fitWidth="30.0">
            <image>
                <Image url="@imgs/icons/back.png" />
            </image>
        </ImageView>
    </graphic>
</Button>
<Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#exit" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Exit">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <styleClass>
        <String fx:value="menu-button" />
        <String fx:value="exit-button" />
    </styleClass>
    <VBox.margin>
        <Insets top="16.0" />
    </VBox.margin>
    <graphic>
        <ImageView fitHeight="35.0" fitWidth="45.0">
            <image>
                <Image url="@imgs/icons/exit.png" />
            </image>
        </ImageView>
    </graphic>
</Button>
</children>
</VBox>
</children>
</AnchorPane>
</left>
<right>
    <AnchorPane prefHeight="634.0" prefWidth="621.0" style="-fx-background-color:
rgba(0, 0, 0, 0.9);" BorderPane.alignment="CENTER">
        <children>
            <VBox alignment="CENTER" layoutX="7.0" prefHeight="634.0"
prefWidth="366.0">
                <children>
                    <Label alignment="CENTER" prefHeight="323.0" prefWidth="357.0"
text="Add Driver" textAlignment="CENTER" textFill="WHITE" wrapText="true">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="75.0" />
                        </font>
                        <padding>
                            <Insets top="100.0" />
                        </padding>
                    </Label>
                    <Label fx:id="errorLabel" alignment="CENTER" prefHeight="67.0"

```

```

prefWidth="366.0" textAlignment="CENTER" textFill="#ff3a3a" wrapText="true">
    <font>
        <Font name="HelveticaRounded-Bold" size="12.0" />
    </font>
</Label>
<Label text="Devoloped by: Damitha Weerasinghe(2236765)"
textFill="WHITE">
    <font>
        <Font name="HelveticaRounded-Bold" size="15.0" />
    </font>
    <VBox.margin>
        <Insets top="204.0" />
    </VBox.margin>
</Label>
</children>
</VBox>
</children>
</AnchorPane>
</right>
</BorderPane>

```

## DeleteDriver-View.fxml

```

<?xml version="1.0" encoding="UTF-8"?>

<?import java.lang.String?>
<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.BorderPane?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<BorderPane fx:id="mainWindow" minWidth="-Infinity" prefHeight="730.0"
prefWidth="1280.0" styleClass="body" stylesheets="@styles/body.css"
xmlns="http://javafx.com/javafx/19" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.cm1601_cw_partb.SceneController">
    <top>
        <Pane prefHeight="97.0" prefWidth="1280.0" style="-fx-background-color:
#121E2A;" BorderPane.alignment="CENTER">
            <children>
                <Label alignment="CENTER" graphicTextGap="30.0" prefHeight="98.0"
prefWidth="1280.0" text="World Rally Cross Championship" textFill="#cbcaed">
                    <font>
                        <Font name="HelveticaRounded-Bold" size="59.0" />
                    </font>
                    <graphic>
                        <ImageView fitHeight="73.0" fitWidth="111.0">
                            <image>
                                <Image url="@imgs/logo.png" />
                            </image>
                        </ImageView>
                    </graphic>
                </Label>
            </children>
        </Pane>
    </top>
    <left>
        <AnchorPane prefHeight="674.0" prefWidth="915.0" BorderPane.alignment="CENTER">
            <children>
                <VBox alignment="CENTER" prefHeight="634.0" prefWidth="908.0" style="-fx-
background-color: rgba(0, 0, 0, 0.8);">
                    <children>
                        <TextField fx:id="driverNameField" alignment="CENTER" maxHeight="-
Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="580.0" promptText="Driver
Name To Be Deleted" styleClass="text-field" stylesheets="@styles/text.css">
                            <font>
                                <Font name="HelveticaRounded-Bold" size="15.0" />
                            </font>
                            <VBox.margin>
                                <Insets bottom="25.0" />
                            </VBox.margin>
                        </TextField>
                    </children>
                </VBox>
            </children>
        </AnchorPane>
    </left>

```



```

        </VBox.margin>
    </TextField>
    <Button fx:id="addDriverBtn" alignment="CENTER"
graphicTextGap="15.0" mnemonicParsing="false" onAction="#deleteDriver"
prefHeight="50.0" prefWidth="350.0" styleClass="menu-button"
stylesheets="@styles/button.css" text="Delete">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
    </VBox.margin>
    <Insets bottom="30.0" />
</VBox.margin>
</Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToMenu" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Back">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
        <styleClass>
            <String fx:value="menu-button" />
            <String fx:value="save-button" />
        </styleClass>
        <graphic>
            <ImageView fitHeight="30.0" fitWidth="30.0">
                <image>
                    <Image url="@imgs/icons/back.png" />
                </image>
            </ImageView>
        </graphic>
    </Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#exit" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Exit">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
        <styleClass>
            <String fx:value="menu-button" />
            <String fx:value="exit-button" />
        </styleClass>
        <VBox.margin>
            <Insets top="16.0" />
        </VBox.margin>
        <graphic>
            <ImageView fitHeight="35.0" fitWidth="45.0">
                <image>
                    <Image url="@imgs/icons/exit.png" />
                </image>
            </ImageView>
        </graphic>
    </Button>
</children>
</VBox>
</children>
</AnchorPane>
</left>

```

```

    <right>
        <AnchorPane prefHeight="634.0" prefWidth="621.0" style="-fx-background-color:
        rgba(0, 0, 0, 0.9);" BorderPane.alignment="CENTER">
            <children>
                <VBox alignment="CENTER" layoutX="7.0" prefHeight="634.0"
                prefWidth="366.0">
                    <children>
                        <Label alignment="CENTER" contentDisplay="CENTER"
                        prefHeight="323.0" prefWidth="357.0" text="Delete Driver" textAlignment="CENTER"
                        textFill="WHITE" wrapText="true">
                            <font>
                                <Font name="HelveticaRounded-Bold" size="75.0" />
                            </font>
                            <padding>
                                <Insets top="100.0" />
                            </padding>
                        </Label>
                        <Label fx:id="errorLabel" alignment="CENTER" prefHeight="67.0"
                        prefWidth="366.0" textAlignment="CENTER" textFill="#ff3a3a" wrapText="true">
                            <font>
                                <Font name="HelveticaRounded-Bold" size="12.0" />
                            </font>
                        </Label>
                        <Label text="Devoloped by: Damitha Weerasinghe(2236765)"
                        textFill="WHITE">
                            <font>
                                <Font name="HelveticaRounded-Bold" size="15.0" />
                            </font>
                            <VBox.margin>
                                <Insets top="204.0" />
                            </VBox.margin>
                        </Label>
                    </children>
                </VBox>
            </children>
        </AnchorPane>
    </right>
</BorderPane>

```

### UpdateDriver-View.fxml

```

<?xml version="1.0" encoding="UTF-8"?>

<?import java.lang.String?>
<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.BorderPane?>
<?import javafx.scene.layout.HBox?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

```

```

<BorderPane fx:id="mainWindow" minWidth="-Infinity" prefHeight="730.0"
prefWidth="1280.0" styleClass="body" stylesheets="@styles/body.css"
xmlns="http://javafx.com/javafx/19" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.cm1601_cw_partb.SceneController">
    <top>
        <Pane prefHeight="97.0" prefWidth="1280.0" style="-fx-background-color:
#121E2A;" BorderPane.alignment="CENTER">
            <children>
                <Label alignment="CENTER" graphicTextGap="30.0" prefHeight="98.0"
prefWidth="1280.0" text="World Rally Cross Championship" textFill="#cbcaed">
                    <font>
                        <Font name="HelveticaRounded-Bold" size="59.0" />
                    </font>
                    <graphic>
                        <ImageView fitHeight="73.0" fitWidth="111.0">
                            <image>
                                <Image url="@imgs/logo.png" />
                            </image>
                        </ImageView>
                    </graphic>
                </Label>
            </children>
        </Pane>
    </top>
    <left>
        <AnchorPane prefHeight="674.0" prefWidth="915.0" BorderPane.alignment="CENTER">
            <children>
                <VBox alignment="CENTER" prefHeight="634.0" prefWidth="908.0" style="-fx-
background-color: rgba(0, 0, 0, 0.8);">
                    <children>
                        <TextField fx:id="updateDriverField" alignment="CENTER"
maxHeight="-Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="815.0"
promptText="Name" styleClass="text-field" stylesheets="@styles/text.css" text="Name
of the driver to be updated">
                            <font>
                                <Font name="HelveticaRounded-Bold" size="15.0" />
                            </font>
                            <VBox.margin>
                                <Insets bottom="5.0" />
                            </VBox.margin>
                        </TextField>
                        <HBox alignment="CENTER" prefHeight="60.0" prefWidth="908.0">
                            <children>
                                <TextField fx:id="driverNameField" alignment="CENTER"
maxHeight="-Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="814.0"
promptText="Updated Name" styleClass="text-field" stylesheets="@styles/text.css">
                                    <font>
                                        <Font name="HelveticaRounded-Bold" size="15.0" />
                                    </font>
                                </TextField>
                            </children>
                        </HBox>
                        <HBox alignment="CENTER" prefHeight="60.0" prefWidth="200.0">
                            <children>
                                <TextField fx:id="driverAgeField" alignment="CENTER"
maxHeight="-Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="812.0"

```

```

promptText="Updated Age" styleClass="text-field" stylesheets="@styles/text.css">
    <font>
        <Font name="HelveticaRounded-Bold" size="15.0" />
    </font>
</TextField>
</children>
</HBox>
<HBox alignment="CENTER" prefHeight="60.0" prefWidth="200.0">
    <children>
        <TextField fx:id="driverTeamField" alignment="CENTER"
maxHeight="-Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="812.0"
promptText="Updated Team" styleClass="text-field" stylesheets="@styles/text.css">
            <font>
                <Font name="HelveticaRounded-Bold" size="15.0" />
            </font>
        </TextField>
    </children>
</HBox>
<HBox alignment="CENTER" prefHeight="60.0" prefWidth="200.0">
    <children>
        <TextField fx:id="driverCarField" alignment="CENTER"
maxHeight="-Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="807.0"
promptText="Updated Car" styleClass="text-field" stylesheets="@styles/text.css">
            <font>
                <Font name="HelveticaRounded-Bold" size="15.0" />
            </font>
        </TextField>
    </children>
</HBox>
<HBox alignment="CENTER" prefHeight="60.0" prefWidth="200.0">
    <children>
        <TextField fx:id="driverPointsField" alignment="CENTER"
maxHeight="-Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="813.0"
promptText="Updated Points" styleClass="text-field" stylesheets="@styles/text.css">
            <font>
                <Font name="HelveticaRounded-Bold" size="15.0" />
            </font>
        </TextField>
    </children>
</HBox>
<Button fx:id="addDriverBtn" alignment="CENTER"
graphicTextGap="15.0" mnemonicParsing="false" onAction="#updateDriver"
prefHeight="50.0" prefWidth="350.0" styleClass="menu-button"
stylesheets="@styles/button.css" text="Update">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <VBox.margin>
        <Insets bottom="30.0" top="40.0" />
    </VBox.margin>
</Button>
<Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToMenu" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Back">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>

```

```

        <styleClass>
            <String fx:value="menu-button" />
            <String fx:value="save-button" />
        </styleClass>
        <graphic>
            <ImageView fitHeight="30.0" fitWidth="30.0">
                <image>
                    <Image url="@imgs/icons/back.png" />
                </image>
            </ImageView>
        </graphic>
    </Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#exit" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Exit">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
        <styleClass>
            <String fx:value="menu-button" />
            <String fx:value="exit-button" />
        </styleClass>
        <VBox.margin>
            <Insets top="16.0" />
        </VBox.margin>
        <graphic>
            <ImageView fitHeight="35.0" fitWidth="45.0">
                <image>
                    <Image url="@imgs/icons/exit.png" />
                </image>
            </ImageView>
        </graphic>
    </Button>
</children>
</VBox>
</children>
</AnchorPane>
</left>
<right>
    <AnchorPane prefHeight="634.0" prefWidth="621.0" style="-fx-background-color:
rgba(0, 0, 0, 0.9);" BorderPane.alignment="CENTER">
        <children>
            <VBox alignment="CENTER" layoutX="7.0" prefHeight="634.0"
prefWidth="366.0">
                <children>
                    <Label alignment="CENTER" prefHeight="323.0" prefWidth="357.0"
text="Update Driver" textAlignment="CENTER" textFill="WHITE" wrapText="true">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="75.0" />
                        </font>
                        <padding>
                            <Insets top="100.0" />
                        </padding>
                    </Label>
                    <Label fx:id="errorLabel" alignment="CENTER" prefHeight="67.0"
prefWidth="366.0" textAlignment="CENTER" textFill="#ff3a3a" wrapText="true">
                        <font>

```

```

        <Font name="HelveticaRounded-Bold" size="12.0" />
    </font>
</Label>
<Label text="Devoloped by: Damitha Weerasinghe(2236765)"
textFill="WHITE">
    <font>
        <Font name="HelveticaRounded-Bold" size="15.0" />
    </font>
    <VBox.margin>
        <Insets top="204.0" />
    </VBox.margin>
</Label>
</children>
</VBox>
</children>
</AnchorPane>
</right>
</BorderPane>

```

### Standing-View.fxml

```
<?xml version="1.0" encoding="UTF-8"?>

<?import java.lang.String?>
<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.BorderPane?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<BorderPane fx:id="mainWindow" minWidth="-Infinity" prefHeight="730.0"
prefWidth="1280.0" styleClass="body" stylesheets="@styles/body.css"
xmlns="http://javafx.com/javafx/19" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.cm1601_cw_partb.SceneController">
    <top>
        <Pane prefHeight="97.0" prefWidth="1280.0" style="-fx-background-color:
#121E2A;" BorderPane.alignment="CENTER">
            <children>
                <Label alignment="CENTER" graphicTextGap="30.0" prefHeight="98.0"
prefWidth="1280.0" text="World Rally Cross Championship" textFill="#cbcaed">
                    <font>
                        <Font name="HelveticaRounded-Bold" size="59.0" />
                    </font>
                    <graphic>
                        <ImageView fitHeight="73.0" fitWidth="111.0">
                            <image>
                                <Image url="@imgs/logo.png" />
                            </image>
                        </ImageView>
                    </graphic>
                </Label>
            </children>
        </Pane>
    </top>
    <left>
        <AnchorPane prefHeight="674.0" prefWidth="915.0" BorderPane.alignment="CENTER">
            <children>
                <VBox alignment="CENTER" prefHeight="634.0" prefWidth="908.0" style="-fx-
background-color: rgba(0, 0, 0, 0.8);">
                    <children>
                        <TableView fx:id="standingsTable" prefHeight="395.0"
prefWidth="908.0">
                            <columns>
                                <TableColumn fx:id="standingsName" prefWidth="314.0"
sortType="DESCENDING" sortable="false" text="Name" />
                                <TableColumn fx:id="standingsAge" prefWidth="56.0"
sortable="false" text="Age" />
                            </columns>
                        </TableView>
                    </children>
                </VBox>
            </children>
        </AnchorPane>
    </left>
</BorderPane>
```

```

        <TableColumn fx:id="standingsTeam" prefWidth="234.0"
sortable="false" text="Team" />
        <TableColumn fx:id="standingsCar" prefWidth="214.0"
sortable="false" text="Car" />
        <TableColumn fx:id="standingsPoints" prefWidth="107.0"
sortType="DESCENDING" sortable="false" text="Points" />
    </columns>
</TableView>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#standings" prefHeight="50.0" prefWidth="350.0"
styleClass="menu-button" stylesheets="@styles/button.css" text="Genarate">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <VBox.margin>
        <Insets top="35.0" />
    </VBox.margin>
</Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToMenu" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Back">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <styleClass>
        <String fx:value="menu-button" />
        <String fx:value="save-button" />
    </styleClass>
    <graphic>
        <ImageView fitHeight="30.0" fitWidth="30.0">
            <image>
                <Image url="@imgs/icons/back.png" />
            </image>
        </ImageView>
    </graphic>
    <VBox.margin>
        <Insets top="15.0" />
    </VBox.margin>
</Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#exit" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Exit">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <styleClass>
        <String fx:value="menu-button" />
        <String fx:value="exit-button" />
    </styleClass>
    <VBox.margin>
        <Insets top="16.0" />
    </VBox.margin>
    <graphic>
        <ImageView fitHeight="35.0" fitWidth="45.0">
            <image>
                <Image url="@imgs/icons/exit.png" />
            </image>
        </ImageView>
    </graphic>

```



```

        </ImageView>
    </graphic>
</Button>
</children>
</VBox>
</children>
</AnchorPane>
</left>
<right>
    <AnchorPane prefHeight="634.0" prefWidth="621.0" style="-fx-background-color:
    rgba(0, 0, 0, 0.9);" BorderPane.alignment="CENTER">
        <children>
            <VBox alignment="CENTER" layoutX="7.0" prefHeight="630.0"
prefWidth="366.0">
                <children>
                    <Label alignment="CENTER" contentDisplay="CENTER"
prefHeight="320.0" prefWidth="357.0" text="View Standings" textAlignment="CENTER"
textFill="WHITE" wrapText="true">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="72.0" />
                        </font>
                        <padding>
                            <Insets top="100.0" />
                        </padding>
                    </Label>
                    <Label fx:id="errorLabel" alignment="CENTER" prefHeight="67.0"
prefWidth="366.0" textAlignment="CENTER" textFill="#ff3a3a" wrapText="true">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="12.0" />
                        </font>
                    </Label>
                    <Label text="Devoloped by: Damitha Weerasinghe(2236765)"
textFill="WHITE">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="15.0" />
                        </font>
                        <VBox.margin>
                            <Insets top="204.0" />
                        </VBox.margin>
                    </Label>
                </children>
            </VBox>
        </children>
    </AnchorPane>
</right>
</BorderPane>

```

**RandomRace-View.fxml**

```

<?xml version="1.0" encoding="UTF-8"?>

<?import java.lang.String?>
<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.BorderPane?>
<?import javafx.scene.layout.HBox?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<BorderPane fx:id="mainWindow" minWidth="-Infinity" prefHeight="730.0"
prefWidth="1280.0" styleClass="body" stylesheets="@styles/body.css"
xmlns="http://javafx.com/javafx/19" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.cm1601_cw_partb.SceneController">
    <top>
        <Pane prefHeight="97.0" prefWidth="1280.0" style="-fx-background-color:
#121E2A;" BorderPane.alignment="CENTER">
            <children>
                <Label alignment="CENTER" graphicTextGap="30.0" prefHeight="98.0"
prefWidth="1280.0" text="World Rally Cross Championship" textFill="#cbcaed">
                    <font>
                        <Font name="HelveticaRounded-Bold" size="59.0" />
                    </font>
                    <graphic>
                        <ImageView fitHeight="73.0" fitWidth="111.0">
                            <image>
                                <Image url="@imgs/logo.png" />
                            </image>
                        </ImageView>
                    </graphic>
                </Label>
            </children>
        </Pane>
    </top>
    <left>
        <AnchorPane prefHeight="674.0" prefWidth="915.0" BorderPane.alignment="CENTER">
            <children>
                <VBox alignment="CENTER" prefHeight="634.0" prefWidth="908.0" style="-fx-
background-color: rgba(0, 0, 0, 0.8);">
                    <children>
                        <TableView fx:id="standingsTable" prefHeight="395.0"
prefWidth="908.0">
                            <columns>
                                <TableColumn fx:id="standingsName" prefWidth="314.0"
sortType="DESCENDING" sortable="false" text="Name" />

```

```

        <TableColumn fx:id="standingsAge" prefWidth="56.0"
sortable="false" text="Age" />
        <TableColumn fx:id="standingsTeam" prefWidth="234.0"
sortable="false" text="Team" />
        <TableColumn fx:id="standingsCar" prefWidth="214.0"
sortable="false" text="Car" />
        <TableColumn fx:id="standingsPoints" prefWidth="107.0"
sortType="DESCENDING" sortable="false" text="Points" />
    </columns>
</TableView>
<HBox alignment="CENTER" prefHeight="56.0" prefWidth="908.0">
    <children>
        <TextField fx:id="randomRaceDate" alignment="CENTER"
maxHeight="-Infinity" maxWidth="-Infinity" prefHeight="44.0" prefWidth="331.0"
promptText="Date" styleClass="text-field" stylesheets="@styles/text.css"
text="DD/MM/YYYY">
            <font>
                <Font name="HelveticaRounded-Bold" size="15.0" />
            </font>
        </TextField>
        <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#randomRace" prefHeight="50.0" prefWidth="350.0"
styleClass="menu-button" stylesheets="@styles/button.css" text="Simulate">
            <font>
                <Font name="HelveticaRounded-Bold" size="30.0" />
            </font>
            <HBox.margin>
                <Insets left="100.0" />
            </HBox.margin>
        </Button>
    </children>
    <VBox.margin>
        <Insets top="15.0" />
    </VBox.margin>
</HBox>
<Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToMenu" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Back">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <styleClass>
        <String fx:value="menu-button" />
        <String fx:value="save-button" />
    </styleClass>
    <graphic>
        <ImageView fitHeight="30.0" fitWidth="30.0">
            <image>
                <Image url="@imgs/icons/back.png" />
            </image>
        </ImageView>
    </graphic>
    <VBox.margin>
        <Insets top="15.0" />
    </VBox.margin>
</Button>
<Button alignment="CENTER" graphicTextGap="15.0"

```

```

mnemonicParsing="false" onAction="#exit" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Exit">
    <font>
        <Font name="HelveticaRounded-Bold" size="30.0" />
    </font>
    <styleClass>
        <String fx:value="menu-button" />
        <String fx:value="exit-button" />
    </styleClass>
    <VBox.margin>
        <Insets top="16.0" />
    </VBox.margin>
    <graphic>
        <ImageView fitHeight="35.0" fitWidth="45.0">
            <image>
                <Image url="@imgs/icons/exit.png" />
            </image>
        </ImageView>
    </graphic>
</Button>
</children>
</VBox>
</children>
</AnchorPane>
</left>
<right>
    <AnchorPane prefHeight="634.0" prefWidth="621.0" style="-fx-background-color:
rgba(0, 0, 0, 0.9);" BorderPane.alignment="CENTER">
        <children>
            <VBox alignment="CENTER" layoutX="7.0" prefHeight="630.0"
prefWidth="366.0">
                <children>
                    <Label alignment="CENTER" contentDisplay="CENTER"
prefHeight="320.0" prefWidth="357.0" text="Simulate Random Race"
textAlignment="CENTER" textFill="WHITE" wrapText="true">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="72.0" />
                        </font>
                        <padding>
                            <Insets top="50.0" />
                        </padding>
                    </Label>
                    <Label fx:id="errorLabel" alignment="CENTER" prefHeight="169.0"
prefWidth="366.0" textAlignment="CENTER" textFill="#ff3a3a" wrapText="true">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="12.0" />
                        </font>
                    </Label>
                    <Label text="Devoloped by: Damitha Weerasinghe(2236765)"
textFill="WHITE">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="15.0" />
                        </font>
                        <VBox.margin>
                            <Insets top="100.0" />
                        </VBox.margin>
                    </Label>

```

```
        </children>
      </VBox>
    </children>
  </AnchorPane>
</right>
</BorderPane>
```

### RacTable-View.fxml

```
<?xml version="1.0" encoding="UTF-8"?>

<?import java.lang.String?>
<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.BorderPane?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<BorderPane fx:id="mainWindow" minWidth="-Infinity" prefHeight="730.0"
prefWidth="1280.0" styleClass="body" stylesheets="@styles/body.css"
xmlns="http://javafx.com/javafx/19" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.cm1601_cw_partb.SceneController">
    <top>
        <Pane prefHeight="97.0" prefWidth="1280.0" style="-fx-background-color:
#121E2A;" BorderPane.alignment="CENTER">
            <children>
                <Label alignment="CENTER" graphicTextGap="30.0" prefHeight="98.0"
prefWidth="1280.0" text="World Rally Cross Championship" textFill="#cbcaed">
                    <font>
                        <Font name="HelveticaRounded-Bold" size="59.0" />
                    </font>
                    <graphic>
                        <ImageView fitHeight="73.0" fitWidth="111.0">
                            <image>
                                <Image url="@imgs/logo.png" />
                            </image>
                        </ImageView>
                    </graphic>
                </Label>
            </children>
        </Pane>
    </top>
    <left>
        <AnchorPane prefHeight="674.0" prefWidth="915.0" BorderPane.alignment="CENTER">
            <children>
                <VBox alignment="CENTER" layoutX="-5.0" prefHeight="634.0"
prefWidth="912.0" style="-fx-background-color: rgba(0, 0, 0, 0.8);">
                    <children>
                        <TableView fx:id="raceTableView" prefHeight="368.0"
prefWidth="508.0">
                            <columns>
                                <TableColumn fx:id="raceTableDate" prefWidth="242.0"
sortType="DESCENDING" sortable="false" text="Date" />
                                <TableColumn fx:id="raceTableLocation" prefWidth="386.0"
sortType="DESCENDING" sortable="false" text="Location" />
                            </columns>
                        </TableView>
                    </children>
                </VBox>
            </children>
        </AnchorPane>
    </left>
</BorderPane>
```

```

        </columns>
        <VBox.margin>
            <Insets left="200.0" right="200.0" />
        </VBox.margin>
        <columnResizePolicy>
            <TableView fx:constant="CONSTRAINED_RESIZE_POLICY" />
        </columnResizePolicy>
    </TableView>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#raceTable" prefHeight="50.0" prefWidth="350.0"
styleClass="menu-button" stylesheets="@styles/button.css" text="Genarate">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
        <VBox.margin>
            <Insets top="35.0" />
        </VBox.margin>
    </Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#switchToMenu" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Back">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
        <styleClass>
            <String fx:value="menu-button" />
            <String fx:value="save-button" />
        </styleClass>
        <graphic>
            <ImageView fitHeight="30.0" fitWidth="30.0">
                <image>
                    <Image url="@imgs/icons/back.png" />
                </image>
            </ImageView>
        </graphic>
        <VBox.margin>
            <Insets top="15.0" />
        </VBox.margin>
    </Button>
    <Button alignment="CENTER" graphicTextGap="15.0"
mnemonicParsing="false" onAction="#exit" prefHeight="50.0" prefWidth="350.0"
stylesheets="@styles/button.css" text="Exit">
        <font>
            <Font name="HelveticaRounded-Bold" size="30.0" />
        </font>
        <styleClass>
            <String fx:value="menu-button" />
            <String fx:value="exit-button" />
        </styleClass>
        <VBox.margin>
            <Insets top="16.0" />
        </VBox.margin>
        <graphic>
            <ImageView fitHeight="35.0" fitWidth="45.0">
                <image>
                    <Image url="@imgs/icons/exit.png" />
                </image>
            </ImageView>
        </graphic>
    </Button>

```

```

        </ImageView>
    </graphic>
    </Button>
</children>
</VBox>
</children>
</AnchorPane>
</left>
<right>
    <AnchorPane prefHeight="634.0" prefWidth="621.0" style="-fx-background-color:
    rgba(0, 0, 0, 0.9);" BorderPane.alignment="CENTER">
        <children>
            <VBox alignment="CENTER" layoutX="7.0" prefHeight="630.0"
prefWidth="366.0">
                <children>
                    <Label alignment="CENTER" contentDisplay="CENTER"
prefHeight="320.0" prefWidth="357.0" text="Race Table" textAlignment="CENTER"
textFill="WHITE" wrapText="true">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="72.0" />
                        </font>
                        <padding>
                            <Insets top="100.0" />
                        </padding>
                    </Label>
                    <Label fx:id="errorLabel" alignment="CENTER" prefHeight="67.0"
prefWidth="366.0" textAlignment="CENTER" textFill="#ff3a3a" wrapText="true">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="12.0" />
                        </font>
                    </Label>
                    <Label text="Devoloped by: Damitha Weerasinghe(2236765)"
textFill="WHITE">
                        <font>
                            <Font name="HelveticaRounded-Bold" size="15.0" />
                        </font>
                        <VBox.margin>
                            <Insets top="204.0" />
                        </VBox.margin>
                    </Label>
                </children>
            </VBox>
        </children>
    </AnchorPane>
</right>
</BorderPane>

```



## Java Code

### Main.

The code is for a JavaFX application that loads and presents a Graphical User Interface (GUI) from an FXML file. The GUI pertains to a program named "World Rally Cross Championship." The application spawns a window with the GUI, which the user cannot resize, and it sets up an event listener to handle requests for closing the window. Whenever the user attempts to close the window, the application prompts them with a confirmation message to ensure that they indeed want to exit the program. The method named `exit()` manages this confirmation message and proceeds to close the window if the user confirms their intent to exit. Finally, the `main()` method launches the JavaFX application.

```
package com.example.cm1601_cw_partb;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.scene.control.Alert;
import javafx.scene.control.ButtonType;
import javafx.stage.Stage;
import java.io.*;

public class Main extends Application {
    @Override
    public void start(Stage stage) throws IOException {
        FXMLLoader fxmlLoader = new FXMLLoader(Main.class.getResource("Menu-View.fxml"));
        Scene scene = new Scene(fxmlLoader.load());
        stage.setResizable(false);
        stage.setTitle("World Rally Cross Championship");
        stage.setScene(scene);
        stage.show();

        // display and get confirmation when exiting using the 'x' icon in the window
        stage.setOnCloseRequest(event -> {
            event.consume();
            exit(stage);
        });
    }

    public static void main(String[] args) {
        launch();
    }

    public void exit(Stage stage){
        // confirm when exiting the program
        Alert alert = new Alert(Alert.AlertType.CONFIRMATION);
        alert.setTitle("Exit");
        alert.setHeaderText("You're about to exit");
        alert.setContentText("Do you want to exit the program?");
    }
}
```

```
        if (alert.showAndWait().get() == ButtonType.OK) {  
            stage.close();  
        }  
    }  
}
```

### Scene Controller.

responsible for bridging the user interface and the application logic. It initializes the application, links UI components to data models, and sets up event handlers. It also creates helper methods to implement the logic for various application actions and behaviors.

```
package com.example.cm1601_cw_partb;

import javafx.beans.property.SimpleIntegerProperty;
import javafx.beans.property.SimpleSetProperty;
import javafx.beans.property.SimpleStringProperty;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.scene.Node;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.layout.BorderPane;
import javafx.stage.Stage;

import java.io.FileWriter;
import java.io.IOException;
import java.util.ArrayList;
import java.util.Formatter;
import java.util.Objects;

public class SceneController {
    private Stage stage;
    private Scene scene;
    private Parent parent;
    @FXML
    private BorderPane mainWindow;
    @FXML
    private TextField driverNameField, driverAgeField, driverTeamField,
driverCarField, driverPointsField, updateDriverField, randomRaceDate;
    @FXML
    private Label errorLabel, standingsLabel;
    @FXML
    private TableColumn<Driver, Integer> standingsAge;
    @FXML
    private TableColumn<Driver, String> standingsCar;
    @FXML
    private TableColumn<Driver, String> standingsName;
    @FXML
    private TableColumn<Driver, Integer> standingsPoints;
    @FXML
    private TableView<Driver> standingsTable;
    @FXML
    private TableColumn<Driver, String> standingsTeam;
    @FXML
    private TableView<Race> raceTableView;
    @FXML
    private TableColumn<Race, String> raceTableDate;
```

```

@FXML
private TableColumn<Race, String> raceTableLocation;

private static ArrayList<Driver> drivers = new ArrayList<Driver>();
private static ArrayList<Driver> randomPosDrivers = new ArrayList<Driver>();
public static ArrayList<Race> races = new ArrayList<>();
private static Driver updatingDriver;

public void switchToMenu(ActionEvent event) throws IOException {
    Parent root = FXMLLoader.load(getClass().getResource("Menu-View.fxml"));
    stage = (Stage) ((Node) event.getSource()).getScene().getWindow();
    scene = new Scene(root);
    stage.setScene(scene);
    stage.setResizable(false);
    stage.show();
}

public void switchToAddDriver(ActionEvent event) throws IOException{
    Parent root =
FXMLLoader.load(Objects.requireNonNull(getClass().getResource("AddDriver-
View.fxml")));
    stage = (Stage) ((Node) event.getSource()).getScene().getWindow();
    scene = new Scene(root);
    stage.setScene(scene);
    stage.setResizable(false);
    stage.show();
}

public void switchToDeleteDriver(ActionEvent event) throws IOException{
    Parent root = FXMLLoader.load(getClass().getResource("DeleteDriver-
View.fxml"));
    stage = (Stage) ((Node) event.getSource()).getScene().getWindow();
    scene = new Scene(root);
    stage.setScene(scene);
    stage.setResizable(false);
    stage.show();
}

public void switchToUpdateDriver(ActionEvent event) throws IOException{
    Parent root = FXMLLoader.load(getClass().getResource("UpdateDriver-
View2.fxml"));
    stage = (Stage) ((Node) event.getSource()).getScene().getWindow();
    scene = new Scene(root);
    stage.setScene(scene);
    stage.setResizable(false);
    stage.show();
}

public void switchToStandings(ActionEvent event) throws IOException{
    Parent root = FXMLLoader.load(getClass().getResource("Standings-View.fxml"));
    stage = (Stage) ((Node) event.getSource()).getScene().getWindow();
    scene = new Scene(root);
    stage.setScene(scene);
    stage.setResizable(false);
    stage.show();
}

```

```

    public void switchToRandomRace(ActionEvent event) throws IOException{
        Parent root = FXMLLoader.load(getClass().getResource("RandomRace-
View.fxml"));
        stage = (Stage) ((Node)event.getSource()).getScene().getWindow();
        scene = new Scene(root);
        stage.setScene(scene);
        stage.setResizable(false);
        stage.show();
    }

    public void switchToRaceTable(ActionEvent event) throws IOException{
        Parent root = FXMLLoader.load(getClass().getResource("RaceTable-View.fxml"));
        stage = (Stage) ((Node)event.getSource()).getScene().getWindow();
        scene = new Scene(root);
        stage.setScene(scene);
        stage.setResizable(false);
        stage.show();
    }

    public void exit(ActionEvent event) throws IOException{
        Alert alert = new Alert(Alert.AlertType.CONFIRMATION);
        alert.setTitle("Exit");
        alert.setHeaderText("You're about to exit");
        alert.setContentText("Do you want to exit the program?");

        if (alert.showAndWait().get() == ButtonType.OK){
            stage = (Stage) mainWindow.getScene().getWindow();
            stage.close();
        }
    }

    public void addDriver(ActionEvent event) throws IOException {
        AddDriver addD = new AddDriver();
        addD.addDriver(driverNameField, driverAgeField, driverTeamField,
driverCarField, driverPointsField, errorLabel, drivers);
    }

    public void deleteDriver(ActionEvent event){
        DeleteDriver deleteD = new DeleteDriver();
        deleteD.deleteDriver(driverNameField, drivers, errorLabel);
    }

    public void updateDriver(ActionEvent event) throws IOException {
        UpdateDriver updatedD = new UpdateDriver();
        updatedD.updateDriverP1(updateDriverField, drivers, updatingDriver,
errorLabel, driverNameField, driverAgeField, driverTeamField, driverCarField,
driverPointsField);
    }

    public void standings(ActionEvent event){
        Standings standings = new Standings();
        standings.standings(drivers);

        standingsName.setCellValueFactory(new PropertyValueFactory<>("name"));
    }

```

```

        standingsAge.setCellValueFactory(new PropertyValueFactory<>("age"));
        standingsTeam.setCellValueFactory(new PropertyValueFactory<>("team"));
        standingsCar.setCellValueFactory(new PropertyValueFactory<>("car"));
        standingsPoints.setCellValueFactory(new
PropertyValueFactory<>("currentPoints"));

        for (Driver driver: drivers){
            standingsTable.getItems().add(driver);
        }
    }

    public void randomRace(ActionEvent event){
        RandomRace randomR = new RandomRace();
        randomPosDrivers = (ArrayList<Driver>) drivers.clone();
        randomR.randomRace(randomRaceDate, errorLabel, randomPosDrivers,
standingsTable, standingsName, standingsAge,standingsTeam, standingsCar,
standingsPoints);
    }

    public void raceTable(ActionEvent event){
        raceTable raceT = new raceTable();
        raceT.raceStat(races, raceTableView, raceTableDate, raceTableLocation);

        raceTableDate.setCellValueFactory(new PropertyValueFactory<>("date"));
        raceTableLocation.setCellValueFactory(new
PropertyValueFactory<>("location"));

        for (Race race: races){
            raceTableView.getItems().add(race);
        }
    }

    public void saveData(ActionEvent event){
        SaveData ssd = new SaveData();
        ssd.saveDataDriver(drivers, errorLabel);
        ssd.saveDataRaces(races, errorLabel);

        Alert alert = new Alert(Alert.AlertType.WARNING);
        alert.setTitle("");
        alert.setHeaderText("Data Saved");
        alert.setContentText("Data has been Saved");
        alert.show();
    }

    public void loadData(ActionEvent event){
        SaveData ssd = new SaveData();
        ssd.loadDataDriver(drivers, errorLabel);
        ssd.loadDataRaces(races, errorLabel);

        Alert alert = new Alert(Alert.AlertType.WARNING);
        alert.setTitle("");
        alert.setHeaderText("Data loaded");
        alert.setContentText("Data has been loaded");
        alert.show();
    }
}

```



### Driver.

This code defines a model representing a racing driver, with properties such as name, age, team, car, and current points. Upon instantiation, the constructor initializes all these properties. Getter and setter methods are also provided for accessing and modifying each property. Moreover, the code contains an equals() method, which takes two driver objects as input and checks if they share the same name. This method can come in handy when dealing with collections that may contain duplicate drivers.

```
package com.example.cml601_cw_partb;

public class Driver
{
    private String name;
    private int age;
    private String team;
    private String car;
    private int currentPoints;

    public Driver(String name, int age, String team, String car, int currentPoints) {
        this.name = name;
        this.age = age;
        this.team = team;
        this.car = car;
        this.currentPoints = currentPoints;
    }
    // Getter methods for the fields
    public String getName() {
        return name;
    }

    public int getAge() {
        return age;
    }

    public String getTeam() {
        return team;
    }

    public String getCar() {
        return car;
    }

    public int getCurrentPoints() {
        return currentPoints;
    }

    // Setter methods for the fields
    public void setName(String name) {
        this.name = name;
    }

    public void setAge(int age) {
        this.age = age;
    }
}
```



```
public void setTeam(String team) {
    this.team = team;
}

public void setCar(String car) {
    this.car = car;
}

public void setCurrentPoints(int currentPoints) {
    this.currentPoints = currentPoints;
}

public boolean compare(Driver driver1, Driver driver2){
    return driver1.name.equals(driver2.name) && driver1.age == driver2.age &&
driver1.team.equals(driver2.team);
}
}
```

**Race.**

has two attributes, "date" and "location", which represent the date and location of the race. The class has a constructor that takes in values for these attributes and initializes them. It also has getter methods for each attribute that allow for accessing the values of these attributes from other parts of the code. This class can be used to create and store race objects in a collection or database.

```
package com.example.cm1601_cw_partb;

public class Race {
    private String date;
    private String location;

    public Race(String date, String location){
        this.date = date;
        this.location = location;
    }

    public String getDate() {
        return date;
    }

    public String getLocation() {
        return location;
    }
}
```

**Add driver details.**

```

package com.example.cm1601_cw_partb;

import javafx.fxml.FXML;
import javafx.scene.control.*;
import javafx.scene.paint.Paint;
import java.util.ArrayList;

public class AddDriver {

    public void addDriver(TextField name, TextField age, TextField team, TextField
car, TextField points, Label error, ArrayList<Driver> drivers) {

        String driverName, driverTeam, driverCar;
        int driverAge, driverPoints;

        error.setText("");
        error.setTextFill(Paint.valueOf("#FF3A3A"));

        driverName = name.getText();
        driverName = driverName.replaceAll("\\s+", "_").toLowerCase();
        if (isIntAge(age, error)){
            driverAge = Integer.parseInt(age.getText());
        }else {
            return;
        }
        driverTeam = team.getText();
        driverTeam = driverTeam.replaceAll("\\s+", "_").toLowerCase();
        driverCar = car.getText();
        driverCar = driverCar.replaceAll("\\s+", "_").toLowerCase();
        if (isIntPoints(points, error)){
            driverPoints = Integer.parseInt(points.getText());
        }else {
            return;
        }

        if (name.getText().equals("") || age.getText().equals("") ||
team.getText().equals("") || car.getText().equals("") ||
points.getText().equals("")){
            error.setText("Please Fill All The Fields");
        }else {
            Driver newDriver = new Driver(driverName, driverAge, driverTeam,
driverCar, driverPoints);
            Driver driver;
            for (int i = 0; i < drivers.size(); i++){
                driver = drivers.get(i);
                if (driver.compare(newDriver, driver)){
                    error.setText(driverName + " already exists");
                    return;
                }
            }
            drivers.add(newDriver);
        }
    }
}

```

```
        error.setText(driverName + " Added!");
    }
}

private boolean isIntAge(TextField input, Label label){
    try {
        int driverAge = Integer.parseInt(input.getText());
        return true;
    } catch (NumberFormatException e){
        label.setText("Age should be a number. Please check your details");
        return false;
    }
}

private boolean isIntPoints(TextField input, Label label){
    try {
        int driverAge = Integer.parseInt(input.getText());
        return true;
    } catch (NumberFormatException e){
        label.setText("Current Points should be a number. Please check your
details");
        return false;
    }
}
}
```

**Delete driver details.**

```

package com.example.cm1601_cw_partb;

import javafx.scene.control.Alert;
import javafx.scene.control.ButtonType;
import javafx.scene.control.Label;
import javafx.scene.control.TextField;

import java.util.ArrayList;

public class DeleteDriver {
    public void deleteDriver(TextField name, ArrayList<Driver> drivers, Label error){
        Driver deleteDriver;
        String driverDeleteName = name.getText();
        error.setText("");
        driverDeleteName = driverDeleteName.replaceAll("\\s+", "_").toLowerCase();
        for (int i = 0; i < drivers.size(); i++){
            if (drivers.get(i).getName().equals(driverDeleteName)){
                deleteDriver = drivers.get(i);
                Alert alert = new Alert(Alert.AlertType.CONFIRMATION);
                alert.setTitle("Are you sure?");
                alert.setHeaderText("You're about to delete " + driverDeleteName);
                alert.setContentText("Driver name: " + deleteDriver.getName() + "\n" +
                    "Age: " + deleteDriver.getAge() + "\n" +
                    "Team: " + deleteDriver.getTeam());

                if (alert.showAndWait().get() == ButtonType.OK){
                    drivers.remove(i);
                    error.setText(driverDeleteName + " has been removed.");
                }
            }
            else {
                error.setText(driverDeleteName + " doesn't exist. Please Try again");
            }
        }
    }
}

```

**Update driver details.**

```

package com.example.cm1601_cw_partb;

import javafx.scene.control.Label;
import javafx.scene.control.TextField;

import java.util.ArrayList;

public class UpdateDriver {
    public void updateDriverPl(TextField updateName, ArrayList<Driver> drivers,
        Driver updatingDriver, Label error, TextField name, TextField age, TextField team,
        TextField car, TextField points){
        String driverName, driverTeam, driverCar;
        int driverAge, driverPoints;

        error.setText("");
        String driverUpdateName = updateName.getText();
        driverUpdateName = driverUpdateName.replaceAll("\\s+", "_").toLowerCase();

        for (int i = 0; i < drivers.size(); i++){
            if (drivers.get(i).getName().equals(driverUpdateName)){
                updatingDriver = drivers.get(i);
                drivers.remove(i);
                error.setText(updatingDriver.getName() + " Found");
            }
        }

        driverName = name.getText();
        driverName = driverName.replaceAll("\\s+", "_").toLowerCase();
        if (isIntAge(age, error)){
            driverAge = Integer.parseInt(age.getText());
        }else {
            return;
        }
        driverTeam = team.getText();
        driverTeam = driverTeam.replaceAll("\\s+", "_").toLowerCase();
        driverCar = car.getText();
        driverCar = driverCar.replaceAll("\\s+", "_").toLowerCase();
        if (isIntPoints(points, error)){
            driverPoints = Integer.parseInt(points.getText());
        }else {
            return;
        }

        if (name.getText().equals("") || age.getText().equals("") ||
            team.getText().equals("") || car.getText().equals("") ||
            points.getText().equals("")){
            error.setText("Please Fill All The Fields");
        }else {
            Driver updatedDriver = new Driver(driverName, driverAge, driverTeam,
            driverCar, driverPoints);
            drivers.add(updatedDriver);
            error.setText(driverName + " updated!");
        }
    }
}

```

```
}

private boolean isIntAge(TextField input, Label label){
    try {
        int driverAge = Integer.parseInt(input.getText());
        return true;
    } catch (NumberFormatException e){
        label.setText("Age should be a number. Please check your details");
        return false;
    }
}

private boolean isIntPoints(TextField input, Label label){
    try {
        int driverAge = Integer.parseInt(input.getText());
        return true;
    } catch (NumberFormatException e){
        label.setText("Current Points should be a number. Please check your
details");
        return false;
    }
}
}
```

**View standing table.**

```
package com.example.cm1601_cw_partb;

import java.util.ArrayList;

public class Standings {
    public void standings(ArrayList<Driver> drivers){
        int length = drivers.size();
        Driver temp;

        for(int i = 0; i < length; i++){
            for (int j =1; j < (length - i); j++){
                if (drivers.get(j - 1).getCurrentPoints() <
drivers.get(j).getCurrentPoints()){
                    temp = drivers.get(j - 1);
                    drivers.set((j - 1), drivers.get(j));
                    drivers.set(j, temp);
                }
            }
        }
    }
}
```



Simulate a random race.

```
package com.example.cm1601_cw_partb;

import javafx.scene.control.Label;
import javafx.scene.control.TableColumn;
import javafx.scene.control.TableView;
import javafx.scene.control.TextField;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.text.Text;

import java.util.ArrayList;
import java.util.Collections;
import java.util.Random;

import static com.example.cm1601_cw_partb.SceneController.races;

public class RandomRace {
    String location, date;
    private String[] locations = {"Nyirád", "Höljes", "Montalegre", "Barcelona",
    "Rīga", "Norway"};

    public void randomRace(TextField date, Label error, ArrayList<Driver> drivers,
    TableView<Driver> standingsTable, TableColumn<Driver, String> standingsName,
    TableColumn<Driver, Integer> standingsAge, TableColumn<Driver, String> standingsTeam,
    TableColumn<Driver, String> standingsCar, TableColumn<Driver, Integer>
    standingsPoints) {
        int first, second, third, dd, mm, yy, max1;
        String dateS = date.getText();
        String[] dateSplit = dateS.split("/");

        dd = Integer.parseInt(dateSplit[0]);
        mm = Integer.parseInt(dateSplit[1]);
        yy = Integer.parseInt(dateSplit[2]);

        if (mm == 1 || mm == 3 || mm == 5 || mm == 7 || mm == 8 || mm == 10 || mm ==
12) {
            max1 = 31;
        }
        else if (mm == 4 || mm == 6 || mm == 9 || mm == 11) {
            max1 = 30;
        }
        else if (yy % 4 == 0 && yy % 100 != 0 || yy % 400 == 0) {
            max1 = 29;
        }
        else {
            max1 = 28;
        }

        if (mm < 1 || mm > 12) {
            error.setText("Invalid date");
            return;
        }
        else if (dd < 1 || dd > max1) {
```

```

        error.setText("Invalid Date");
        return;
    }

    Random rand = new Random();
    int locationIndex = rand.nextInt(locations.length);
    location = locations[locationIndex];

    error.setText("Date: " + dateS + "\n" +
        "Location: " + location);

    Collections.shuffle(drivers);

    first = drivers.get(0).getCurrentPoints();
    second = drivers.get(1).getCurrentPoints();
    third = drivers.get(2).getCurrentPoints();

    first = first + 10;
    second = second + 7;
    third = third + 5;

    drivers.get(0).setCurrentPoints(first);
    drivers.get(1).setCurrentPoints(second);
    drivers.get(2).setCurrentPoints(third);

    standingsName.setCellValueFactory(new PropertyValueFactory<>("name"));
    standingsAge.setCellValueFactory(new PropertyValueFactory<>("age"));
    standingsTeam.setCellValueFactory(new PropertyValueFactory<>("team"));
    standingsCar.setCellValueFactory(new PropertyValueFactory<>("car"));
    standingsPoints.setCellValueFactory(new
PropertyValueFactory<>("currentPoints"));

    for (Driver driver: drivers){
        standingsTable.getItems().add(driver);
    }

    String[] driverPos = new String[drivers.size()];

    for (int i = 0; i < drivers.size(); i++){
        driverPos[i] = drivers.get(i).getName();
    }

    Race newRace = new Race(dateS, location);
    races.add(newRace);
}
}

```

**View race table.**

```
package com.example.cm1601_cw_partb;

import javafx.scene.control.TableColumn;
import javafx.scene.control.TableView;
import javafx.scene.control.cell.PropertyValueFactory;

import java.util.ArrayList;

public class raceTable {
    public void raceStat(ArrayList<Race> races, TableView<Race> raceTableView,
        TableColumn<Race, String> racerTableDate, TableColumn<Race, String>
        racerTableLocation){
        for (int i = 0; i < races.size() - 1; i++) {
            for (int j = 0; j < races.size() - i - 1; j++) {
                if (races.get(j).getDate().compareTo(races.get(j+1).getDate()) > 0) {
                    Race temp = races.get(j);
                    races.set(j, races.get(j+1));
                    races.set(j+1, temp);
                }
            }
        }
    }
}
```

### Save data and Load data

```
package com.example.cm1601_cw_partb;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Node;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.Alert;
import javafx.scene.control.ButtonType;
import javafx.scene.control.Label;
import javafx.scene.layout.BorderPane;
import javafx.stage.Stage;

import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileWriter;
import java.io.IOException;
import java.util.ArrayList;
import java.util.Scanner;

public class SaveData {
    private Stage stage;
    private Scene scene;
    @FXML
    private BorderPane mainWindow;

    public void saveDataDriver(ArrayList<Driver> drivers, Label error){
        try {
            FileWriter myWriter = new
FileWriter("src\\main\\resources\\com\\example\\cm1601_cw_partb\\data\\drivers.txt");
            String name, team, car;
            int age, points;
            for (int i = 0; i < drivers.size(); i++){
                name = drivers.get(i).getName();
                age = drivers.get(i).getAge();
                team = drivers.get(i).getTeam();
                car = drivers.get(i).getCar();
                points = drivers.get(i).getCurrentPoints();
                myWriter.write(name + " " + age + " " + team + " " + car + " " +
points + "\n");
            }
            myWriter.close();
        } catch (IOException e) {
            error.setText("An error occurred.");
            e.printStackTrace();
        }
    }

    public void saveDataRaces(ArrayList<Race> races, Label error){
        try {
```

```

        FileWriter myWriter = new
FileWriter("src\\main\\resources\\com\\example\\cm1601_cw_partb\\data\\races.txt");
        String[] names;
        String date, location;

        for (int i = 0; i < races.size(); i++){
            date = races.get(i).getDate();
            location = races.get(i).getLocation();
            myWriter.write(date + " " + location + " ");
            myWriter.write("\n");
        }

        myWriter.close();
    } catch (IOException e) {
        error.setText("An error occurred.");
        e.printStackTrace();
    }
}

public void loadDataDriver(ArrayList<Driver> drivers, Label error){
    try {
        String name, team, car;
        int age, points;
        File myObj = new
File("src\\main\\resources\\com\\example\\cm1601_cw_partb\\data\\drivers.txt");
        Scanner myReader = new Scanner(myObj);
        while (myReader.hasNextLine()) {
            String data = myReader.nextLine();
            String[] driverData = data.split(" ");
            name = driverData[0];
            age = Integer.parseInt(driverData[1]);
            team = driverData[2];
            car = driverData[3];
            points = Integer.parseInt(driverData[4]);
            Driver driver = new Driver(name, age, team, car, points);
            drivers.add(driver);
        }
        myReader.close();
    } catch (FileNotFoundException e) {
        error.setText("An error occurred.");
        e.printStackTrace();
    }
}

public void loadDataRaces(ArrayList<Race> races, Label error){
    try {
        String date, location;
        File myObj = new
File("src\\main\\resources\\com\\example\\cm1601_cw_partb\\data\\races.txt");
        Scanner myReader = new Scanner(myObj);
        while (myReader.hasNextLine()) {
            String data = myReader.nextLine();
            String[] race = data.split(" ");

            date = race[0];
            location = race[1];

```

```
        Race raceTemp = new Race(date, location);
        races.add(raceTemp);

    }
    myReader.close();
} catch (FileNotFoundException e) {
    error.setText("An error occurred.");
    e.printStackTrace();
}

}

public void exit(ActionEvent event) throws IOException {
    Alert alert = new Alert(Alert.AlertType.CONFIRMATION);
    alert.setTitle("Exit");
    alert.setHeaderText("You're about to exit");
    alert.setContentText("Do you want to exit the program?");

    if (alert.showAndWait().get() == ButtonType.OK) {
        stage = (Stage) mainWindow.getScene().getWindow();
        stage.close();
    }
}

}
```

## 5. J-Units, Test plan and test Cases

### 1. Test Cases

| Test No | Test Name                 | Action         | Inputs   | Expected output                                 | Actual Output                                   | Test Result |
|---------|---------------------------|----------------|--|---|---|-------------|
| 01      | Adding a driver           | AddDriver()    | Name: John<br>Age: 21<br>Team: SL<br>Car: Corvette<br>Points: 500  | The Driver added!                               | The Driver added!                               | Pass        |
| 02      | Input Vallidation         | AddDriver()    | Name: John<br>Age: sad<br>Team: SL<br>Car: Corvette<br>Points: 500 | Invalid Input<br>Age must be a number           | Invalid Input<br>Age must be a number           | pass        |
| 03      | Delete a driver           | DeleteDriver() | Name: John   | John deleted                                    | John deleted                                    | pass        |
| 04      | Delete an Unknown Driver  | DeleteDriver() | Name:<br>Unknown   | The Driver Unknown does not exist in our system | The Driver Unknown does not exist in our system | Pass        |
| 05      | Update a Driver's Details | UpdateDriver   | Name: John<br>UpdName: Udara<br>Age: 21<br>Team: Pak               | The Driver Doe has been updated                 | Doe has been updated                            | pass        |

|    |  |                |  |  |  |      |
|----|--|----------------|--|--|--|------|
|    |  |                | Car: Plastic<br>Points: 900  |  |  |      |
| 06 | Update an Unknown driver's Details             | UpdateDriver   | Name: Unknown<br>UpdName: Udara<br>Age: 21<br>Team: Pak<br>Car: Plastic<br>Points: 900 | The driver Unknown is not in our system  | The driver Unknown is not in our system  | pass |
| 07 | Update a driver's details but give wrong data. | UpdateDriver   | Name: John<br>UpdName: Doe<br>Age: rew<br>Team: Pak<br>Car: Plastic<br>Points: 900     | Invalid Input<br>Age must be a number  | Invalid Input<br>Age must be a number  | pass |
| 08 | View cross standings                           | StandingsTable | none   | A table with all the drivers where they are sorted by points in the descending order | A table with all the drivers where they are sorted by points in the descending order | pass |



## **6. Robustness and the maintainability**

Adding comments to code can help programmers understand the purpose of a particular piece of code or provide additional information about its behavior. This is especially beneficial when returning to the code after a long period of time. Furthermore, organizing the code into different classes can aid in the program's functionality and decrease the complexity of individual classes. By dividing the code into smaller, more manageable pieces, it becomes easier to test, debug, and modify. Each class has a specific responsibility or function, which simplifies the comprehension of the program's workings. In conclusion, implementing comments and organizing the code into different classes can significantly enhance the maintainability and robustness of a Java program.

## **7. Conclusion and Assumptions**

### **1. Conclusion**

It is possible to create a Java FX GUI application for managing a World Rally Cross Championship, which must have features for managing driver details, viewing standings, simulating races, and saving and loading data. Designing and architecting the application's components and their interactions, along with its implementation using Java FX, requires careful planning. A detailed report must also be provided, outlining the system's features, implementation, testing, and encountered issues.

In conclusion, developing a comprehensive and intuitive GUI application for World Rally Cross Championship management through Java FX demands rigorous planning, precise execution, and thorough documentation.

### **2. Assumptions**

- The system requires users to provide all driver details again when updating them.
- At the beginning of every program, users must load all data from text files.
- At the end of every program, users must save all data to text files.

## 8. Reference list

Bro Code, 2021, Java Tutorial For Beginners, Available from: [https://www.youtube.com/playlist?list=PLZPZq0r\\_RZOMhCAyywfnYLLrjiVOkdAI1](https://www.youtube.com/playlist?list=PLZPZq0r_RZOMhCAyywfnYLLrjiVOkdAI1) [Accessed Date: 20<sup>th</sup> March 2023]

VeasMKII, 2016, JavaFX Application Menu, Available from: <https://stackoverflow.com/questions/35351097/javafx-application-menu> [Accessed Date: 25<sup>th</sup> March 2023]

BabyGau, 2013, reading object data from text file java, Available From: <https://stackoverflow.com/questions/10758319/reading-object-data-from-text-file-in-java> [Accessed Date: 23<sup>rd</sup> March 2023]

User42155, 2009, How do I generate random integers within a specific range in Java?, Available From: <https://stackoverflow.com/questions/363681/how-do-i-generate-random-integers-within-a-specific-range-in-java/363692#363692> [Accessed Date: 22<sup>nd</sup> March 2023]

W3 Schools, Java OOP, Available From: [https://www.w3schools.com/java/java\\_oop.asp](https://www.w3schools.com/java/java_oop.asp) [Accessed Date: 30<sup>th</sup> March 2023]

GeeksForGeeks, 2022, Exceptions In Java, Available From: <https://www.geeksforgeeks.org/exceptions-in-java/?ref=lbp> [Accessed Date: 23<sup>rd</sup> March 2023]

Java Code Junkie, 2021, Create TableView | JavaFX GUI Tutorial for Beginners, Available From: [https://www.youtube.com/watch?v=vego72w5kPU&ab\\_channel=JavaCodeJunkie](https://www.youtube.com/watch?v=vego72w5kPU&ab_channel=JavaCodeJunkie) [Accessed Date: 9<sup>th</sup> April 2023]