**Vehicle Fare System**

**( VFS )**

**Members:**

**Jayson Mancol**

**Program:**

**BS Info Tech 1A**

**(2023-2024)**

Table of Contents

* **Introduction**

**Purpose of the project:**

**Objective of the Project:**

**Scope of the Project:**

* **Project Overview**

**Description of the project:**

**Key features:**

* **User Manual**

**Getting Started:**

**User Guide:**

* **Code Documentation**

**Overview of Code base:**

**Comments and Documentation:**

**Coding Conventions and Style Guide:**

* **Conclusion**

**Summary of Project Outcomes:**

**Acknowledgments:**

**Purpose of the project:**

The purpose of Vehicle Fare System is to prevent overpricing and error by giving change and to avoid time-consuming by giving a ticket and ensuring that only drivers with a franchise are allowed to drive. Also, it makes it easy for the passenger to give their fare to the driver. In terms of an emergency, it is easy for them to track in as it includes the name of the passenger and the driver, along with the plate number and the date when the passenger boarded.

**Objective of the Project:**

* Prevent overpricing through transparent and regulated fare calculations.
* Ensuring passengers are charged fairly for their rides.
* Allow only authorized drivers with valid franchises to provide transportation services.
* Enhance passenger safety with emergency tracking feature, including passenger name, driver details, vehicle plate number, and boarding date for swift assistance during emergencies.

**Scope of the Project:**

EXIT PROGRAM which refers to close the program.

LIST OF THE FARE AND ROUTE so that the passenger knows where the route of the vehicle is and how much the fare is.

LIST OF THE PASSENGER the passengers can also know who they are traveling with.

INFORMATION OF THE DRIVER the passengers know who is the driver of the vehicle their ride.

NEW PASSENGER which is for new passengers who want to ride.

SEARCHING PASSENGERto check the passenger who rides in the vehicle.

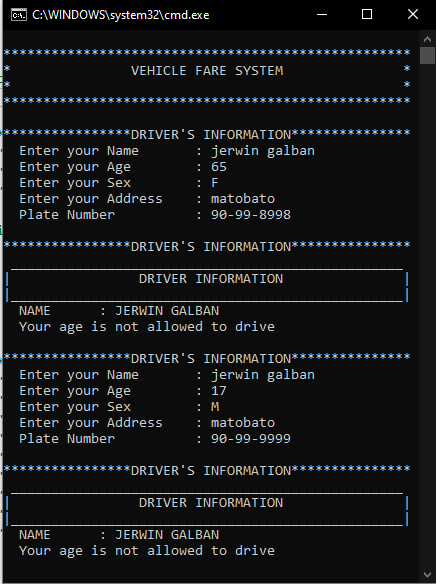
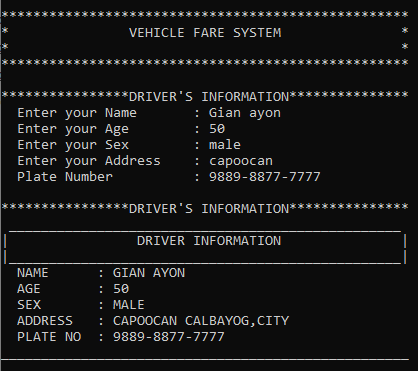
**Description of the project:**

The Vehicle fare system is developed in Java language, it aims to prevent overpricing by implementing a standard fare calculation mechanism, ensuring passengers are charged fairly for their rides. Also, the system ensures that only licensed drivers with a franchise are allowed to operate, enhancing safety and reliability for passengers.

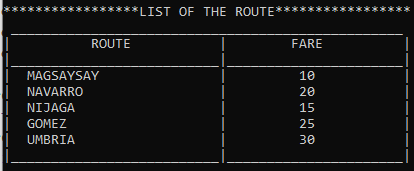
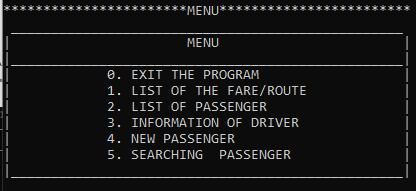
**Key features:**

Exit Program (0): Allows users to exit the program or return to the main menu.  
List Fare and Route (1): Provides passengers with information about the fare for different routes.  
List of Passengers (2): Allows passengers to view a list of other passengers who are sharing the ride. Provides transparency and awareness of who else is traveling in the vehicle.  
Driver Information (3): Provides passengers with information about the driver of the vehicle.  
Displays details such as the driver's name,address and plate number of the vehicle  
New Passenger (4): Enables new passengers to request a ride and join the vehicle.  
Initiates the process for registering a new passenger to issue a ticket.  
Searching Passenger (5): Allows users to search for specific passengers who are already riding in the vehicle.

**User Manual :**

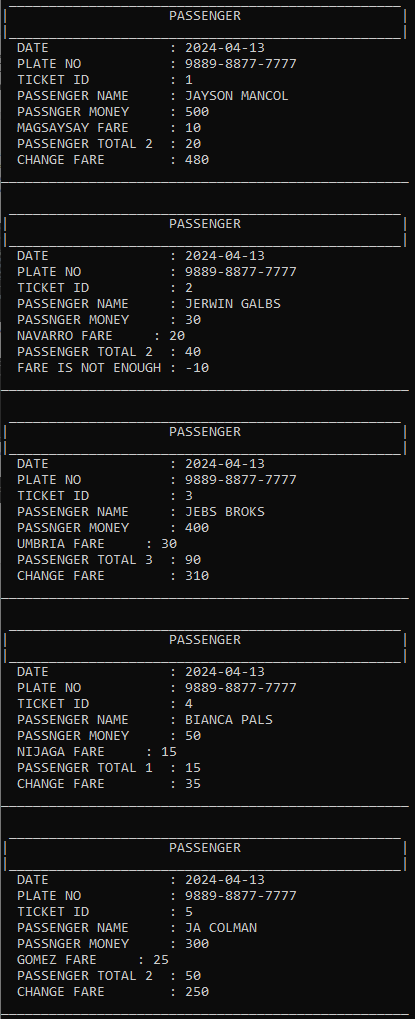
This is the driver's information. Before the system runs, the driver will fill in his information so that the passengers know who he is, and also they can be sure that they are safe on the road, and one of them is that they will know the plate number of the vehicle they drive to avoid scammers and those without a franchise.

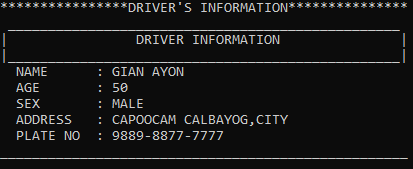
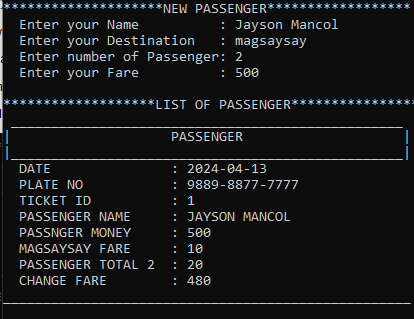
This is the example where the driver is over or under the age that does not meet the requirement of our System, we did this to avoid accidents where if our System does not accept the age of 18 to 60yrs old, it will not continue to travel.

This is the menu; there are 6 choices for the passenger. Exit Program (0): Allows users to exit the program or return to the main menu.List Fare and Route (1): Provides passengers with information about the fare for different routes.List of Passengers (2): Allows passengers to view a list of other passengers who are sharing the ride. Provides transparency and awareness of who else is traveling in the vehicle. Driver Information (3): Provides passengers with information about the driver of the vehicle. Displays details such as the driver's name,address and plate number of the vehicle. New Passenger (4): Enables new passengers to request a ride and join the vehicle.Initiates the process for registering a new passenger to issue a ticket. Searching Passenger (5): Allows users to search for specific passengers who are already riding in the vehicle.

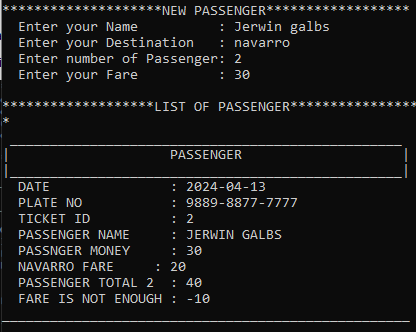
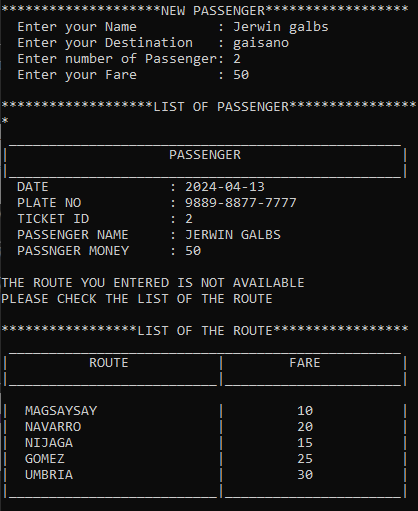
This is the “1. LIST OF THE FARE AND ROUTE” so that the passenger knows where the route of the vehicle is and how much the fare is.

This is the example where when you select button 2, which is empty this is what will print.

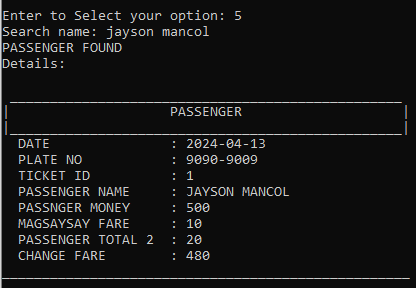
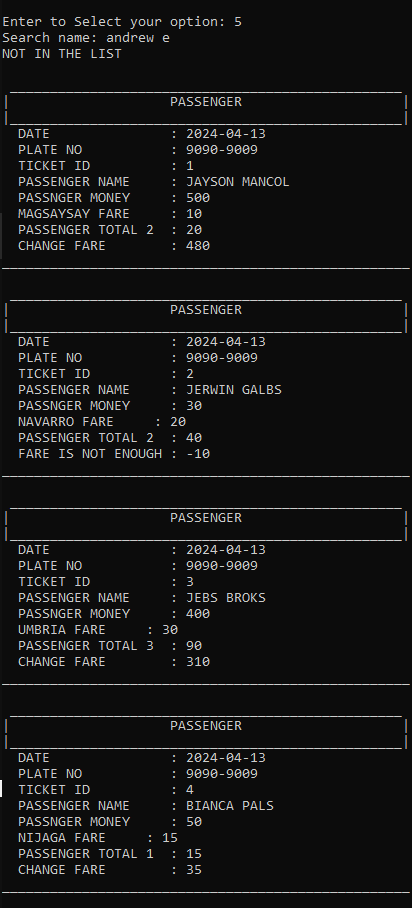
This is the LIST OF THE ROUTE here you can see the names of the passengers and their different Ticket id and routes

 This is “3. INFORMATION OF DRIVER “ where easily know who is the driver of the vehicle and to easily know what is the plate number of the vehicle.

This is the “4. NEW PASSENGER " The new passenger will enter their name, destination, and passenger number this is if the passenger is accompanied by others and the passenger's fare. When it is done it will go to the list of passengers, here their specific details, the DATE when the passenger boarded will be seen here, it is directed to the real date, PLATE NUMBER for what is the plate number of the vehicle or the one put by the driver, TICKET ID is to easily count how many passengers there are, PASSENGER NAME, so that it will be easy to identify who the passenger is, PASSENGER MONEY, how much the passenger paid, route fare where it is written how much the fare of your route is, PASSENGER TOTAL it is written here how many are with you and it will automatically will be deducted from your fare, and CHANGE FARE, for change so that the passenger knows how much he will spend.

This is the example when the passenger enters a route that is not available in the list, the route list will automatically be displayed.

This is the example when the passenger is entering an insufficient fare, the system will put a note below that the passenger's fare is insufficient

This is the “5. SEARCHING PASSENGER” is used to search for the passenger if they are on the rides; the passenger's data will appear on the screen if they are found.

This is the example when the passenger is searching for a name that is not on the list, the system will show the name of the passenger.

**Code Documentation :**

import java.util.Scanner;//it's a class of java.util package that allows you to read

                         //input from different sources in Java programs.

import java.time.LocalDate;//its a package of java.time that represents a date without

                           //time information (year, month, and day).

public class VechicleFareSystem {

    //I used static method to define constants that are shared among all

    // instances of a class or accessed without creating an instance.

    // This is the 2D array I use this to store the data of the passenger and Driver.

    static String Route[][] = new String[50][5];

    static String Driver[][] = new String[3][6];

    static int LimitRoute = 1;// I use this to count the list of the passenger.

    static LocalDate DT = LocalDate.now();//This is the object of the LocalDate class

    static Scanner input = new Scanner(System.in);//This is the object of the Scanner class

    public static void main(String[] args) {

        // This is the main method of my program.

        System.out.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

        System.out.println("\*               VEHICLE FARE SYSTEM               \*");

        System.out.println("\*                                                 \*");

        System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

        DriverInfo();//This is the DriverInfo method used to print the information of the driver

        System.out.println("Let's Gooo!!!");//When the program is end this message will print.

    }

    // This is the menu botton to choose the user.

    public static void Menu() {

        System.out.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*MENU\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

        System.out.println(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

        System.out.println("|                      MENU                       |");

        System.out.println("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|");

        System.out.println("|            0. EXIT THE PROGRAM                  |");

        System.out.println("|            1. LIST OF THE FARE/ROUTE            |");

        System.out.println("|            2. LIST OF PASSENGER                 |");

        System.out.println("|            3. INFORMATION OF DRIVER             |");

        System.out.println("|            4. NEW PASSENGER                     |");

        System.out.println("|            5. SEARCHING  PASSENGER              |");

        System.out.println("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|");

        System.out.print("\nEnter to Select your option: ");

        char select = input.next().charAt(0);

        // This is the condition to print what user choose.

        switch (select) {

            case '0':// if the user select "0" the program will end.

                return;

            case '1':// if the user select "1" the program will go to ListFare Method.

                ListFare();

                break;

            case '2':// if the user select "1" the program will go to ListPassenger Method.

                ListPassenger();

                break;

            case '3':// if the user select "1" the program will go to ListFare Method.

                DriverInformation();

                break;

            case'4':// if the user select "1" the program will go to NewPassenger Method.

                NewPassenger();

                break;

            case '5':

                Search();//if the user select "5" the program will go to Search Method.

                break;

            default:

                System.out.println("INVALID INPUT");

                Menu();// if the user selects none of the choices the program will return to the Menu Method.

                break;

         }

    }

    // This is the ListPassenger Method to print the list of the passenger who want

    // to rides.

    public static void ListPassenger() {

        if (LimitRoute == 1) {

            System.out.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*LIST OF PASSENGER\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

            System.out.println("                      NO BOOKING :( ");

        } else {

            // This is for loop used to print the passenger information store in the array.

            for (int i = 1; i < LimitRoute; i++) {

                Passenger(i);

            }

        }

        Menu();

    }

    // This is the NewPassenger Method to add who wants to ride,

    // the user input there name to know who is the passenger

    // in vehicle just incase in the emergency, and also user

    // input there destination to know diin siya mahawas and

    // how many people na kaupod san user and also the fare of the user.

    public static void NewPassenger() {

        try{//I use exception to maintain my error, even its error the program will not stop running.

        System.out.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*NEW PASSENGER\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

        System.out.print("  Enter your Name          : ");

        Route[LimitRoute][0] = input.next();

        Route[LimitRoute][1] = input.next();

        System.out.print("  Enter your Destination   : ");

        Route[LimitRoute][2] = input.next();

        System.out.print("  Enter number of Passenger: ");

        Route[LimitRoute][3] = input.next();

        System.out.print("  Enter your Fare          : ");

        Route[LimitRoute][4] = input.next();

        System.out.print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*LIST OF PASSENGER\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

        Passenger(LimitRoute);

        LimitRoute++;

         } catch (Exception e) {

                    System.out.println("\n\t\tINVALID KEYWORD");//This is the message that will

                                                               //display if the user input is error.

        }

        Menu();

    }

  //This is the Search Method use to search the passenger if they are on the rides.

  public static void Search() {

      System.out.print("Search name: ");

      String name = input.next();

      String names = input.next();

      boolean found = false;

    // This is for loop use to find what the user is looking for that is stored in the array.

      for (int i = 1; i < LimitRoute; i++) {

          if (name.equalsIgnoreCase(Route[i][0])&&names.equalsIgnoreCase(Route[i][1])) {

              System.out.println("PASSENGER FOUND\nDetails:");

              Passenger(i);

              found = true;

              break; // Stop searching once name is found.

            }

       }

       if (!found) {

          System.out.println("NOT IN THE LIST");

          ListPassenger();

      }

    Menu(); // Go back to the main menu after searching.

  }

    // This is the Passenger Method to see the information of the passenger,

    // it also indicate the date when they book, and also the plate number

    // and ticket ID to easly identify the vehicle. And the fare of the passenger

    // is automatically calculate depend to the passenger number and

    // the system automatically give the change fare of the passenger.

    public static void Passenger(int LimitRoute) {

        System.out.println();

        String Destination = Route[LimitRoute][2];

        System.out.println(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

        System.out.println("|                    PASSENGER                    |");

        System.out.println("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|");

        System.out.println("  DATE               : " + DT  );

        System.out.println("  PLATE NO           : " + Driver[0][5]);

        System.out.println("  TICKET ID          : " + LimitRoute);

        System.out.println("  PASSENGER NAME     : " + Route[LimitRoute][0].toUpperCase()+" "+Route[LimitRoute][1].toUpperCase());

        System.out.println("  PASSNGER MONEY     : " + Route[LimitRoute][4]);

        int fare = 0;

        if (Destination.equalsIgnoreCase("MAGSAYSAY")) {

            fare = 10;

        } else if (Destination.equalsIgnoreCase("NAVARRO")) {

            fare = 20;

        } else if (Destination.equalsIgnoreCase("NIJAGA")) {

            fare = 15;

        } else if (Destination.equalsIgnoreCase("GOMEZ")) {

            fare = 25;

        } else if (Destination.equalsIgnoreCase("UMBRIA")) {

            fare = 30;

        } else {

            System.out.println("\nTHE ROUTE YOU ENTERED IS NOT AVAILABLE\nPLEASE CHECK THE LIST OF THE ROUTE");

            ListFare();

        }

        System.out.println("  "+Route[LimitRoute][2].toUpperCase() + " FARE     : " + fare);

        int passtotal = Integer.parseInt(Route[LimitRoute][3]);

        fare = fare \* passtotal;

        System.out.println("  PASSENGER TOTAL "+passtotal+"  : " + fare);

        int Fare = Integer.parseInt(Route[LimitRoute][4]);

        int change = 0;

        if (Destination.equalsIgnoreCase("MAGSAYSAY")) {

            change = Fare - fare;

        } else if (Destination.equalsIgnoreCase("NAVARRO")) {

            change = Fare - fare;

        } else if (Destination.equalsIgnoreCase("NIJAGA")) {

            change = Fare - fare;

        } else if (Destination.equalsIgnoreCase("GOMEZ")) {

            change = Fare - fare;

        } else if (Destination.equalsIgnoreCase("UMBRIA")) {

            change = Fare - fare;

        } else {

            System.out.println("PLEASE CHECK THE LIST OF THE FARE");

        }

        if (Fare < fare) {

            System.out.println("  FARE IS NOT ENOUGH : " + change);

        } else {

            System.out.println("  CHANGE FARE        : " + change);

        }

    System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

    }

   //This is the DriverInformation method to fill up the information of the driver.

    public static void DriverInformation() {

        System.out.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*DRIVER'S INFORMATION\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

        System.out.println(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

        System.out.println("|                DRIVER INFORMATION               |");

        System.out.println("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|");

        System.out.println("  NAME      : "+ Driver[0][0].toUpperCase() +" "+Driver[0][1].toUpperCase());

        if(Integer.parseInt(Driver[0][2])>=65||Integer.parseInt(Driver[0][2])<=17){

        System.out.println("  YOUR AGE IS NOT ALLOWED TO DRIVE");

        DriverInfo();

        }else{

        System.out.println("  AGE       : "+ Integer.parseInt(Driver[0][2]));

        }

        if(Driver[0][3].equalsIgnoreCase("Male")||

         Driver[0][3].equalsIgnoreCase("Famale")||

              Driver[0][3].equalsIgnoreCase("M")||

              Driver[0][3].equalsIgnoreCase("F")){

        System.out.println("  SEX       : "+ Driver[0][3].toUpperCase());

        }else{

        System.out.println("\n\t\tINVALID INPUT ");

        DriverInfo();

        }

        System.out.println("  ADDRESS   : "+ Driver[0][4].toUpperCase()+ " CALBAYOG,CITY" );

        System.out.println("  PLATE NO  : "+ Driver[0][5]  );

        System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

        Menu();

    }

   //This is the ListFare method to display the available route and fare of the vehicle.

    public static void ListFare() {

        System.out.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*LIST OF THE ROUTE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

        System.out.println(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

        System.out.println("|          ROUTE           |        FARE          |");

        System.out.println("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_| ");

        System.out.println("|  MAGSAYSAY               |         10           |");

        System.out.println("|  NAVARRO                 |         20           |");

        System.out.println("|  NIJAGA                  |         15           |");

        System.out.println("|  GOMEZ                   |         25           |");

        System.out.println("|  UMBRIA                  |         30           |");

        System.out.println("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|");

        Menu();

    }

    //This is the DriverInfo method this method is use to print the information of the driver.

    public static void DriverInfo() {

        try{//I use exception to maintain my error, even its error the program will not stop running.

        System.out.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*DRIVER'S INFORMATION\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

        System.out.print("  Enter your Name       : ");

        Driver[0][0] = input.next();

        Driver[0][1] = input.next();

        System.out.print("  Enter your Age        : ");

        Driver[0][2] = input.next();

        System.out.print("  Enter your Sex        : ");

        Driver[0][3] = input.next();

        System.out.print("  Enter your Address    : ");

        Driver[0][4] = input.next();

        System.out.print("  Plate Number          : ");

        Driver[0][5] = input.next();

        DriverInformation();

        }catch(Exception e){

               System.out.println("\n\t\tINVALID KEYWORD");//This is the message that will

                                                          //display if the user input is error.

        DriverInfo();

       }

    }

 }

**Summary of Project Outcomes:**

Development and deployment of a functional vehicle fare system with all planned features and functionalities.

To prevent overpricing and error by giving change and to avoid time-consuming by giving a ticket and ensuring that only drivers with a franchise are allowed to drive.

Implementation of a user-friendly interface for passengers and operators to access fare information, make payments, and manage rides efficiently.

Integration with relevant databases and systems for driver verification, route, and emergency response.

**Acknowledgments:**

I am writing to express my deep gratitude for the unwavering support and guidance given to develop the Vehicle Fare System. First of all, thanks to our instructor who tirelessly taught us and guided us to make this System, To my members, thank you for your help and the ideas given to the development of the Vehicle Fare System, and also the support you gave me was a big help so that I could do it well.