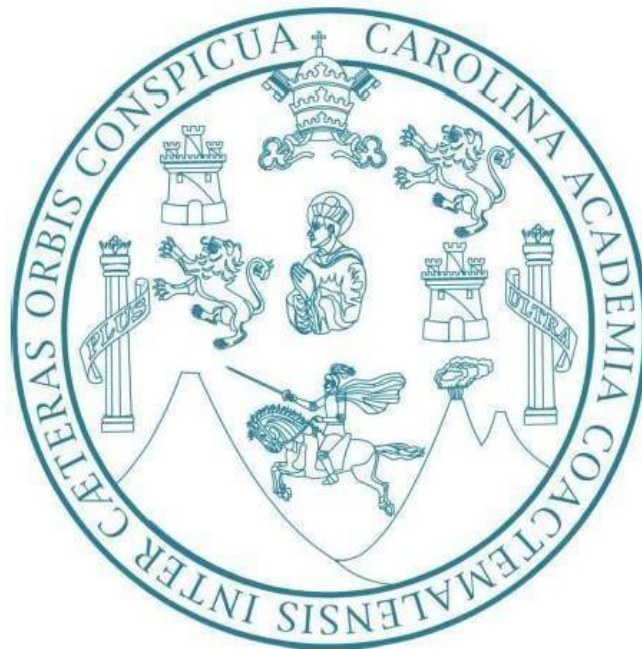


Facultad de Ingeniería  
Introducción a la Programación y  
Computación 1Sección: E  
Cat. Ing. Nefthalí De Jesús Calderón Méndez  
Tutor académico: Rodrigo Antonio Porón De León



# Manual Técnico

JUEGO EN CONSOLA

# Universidad de San Carlos de Guatemala

## INDICE

### Contenido

INDICE .....	2
Lenguaje utilizado.....	3
Lógica utilizada .....	3
Diagramas de flujo (clases importantes):.....	8
UML .....	12

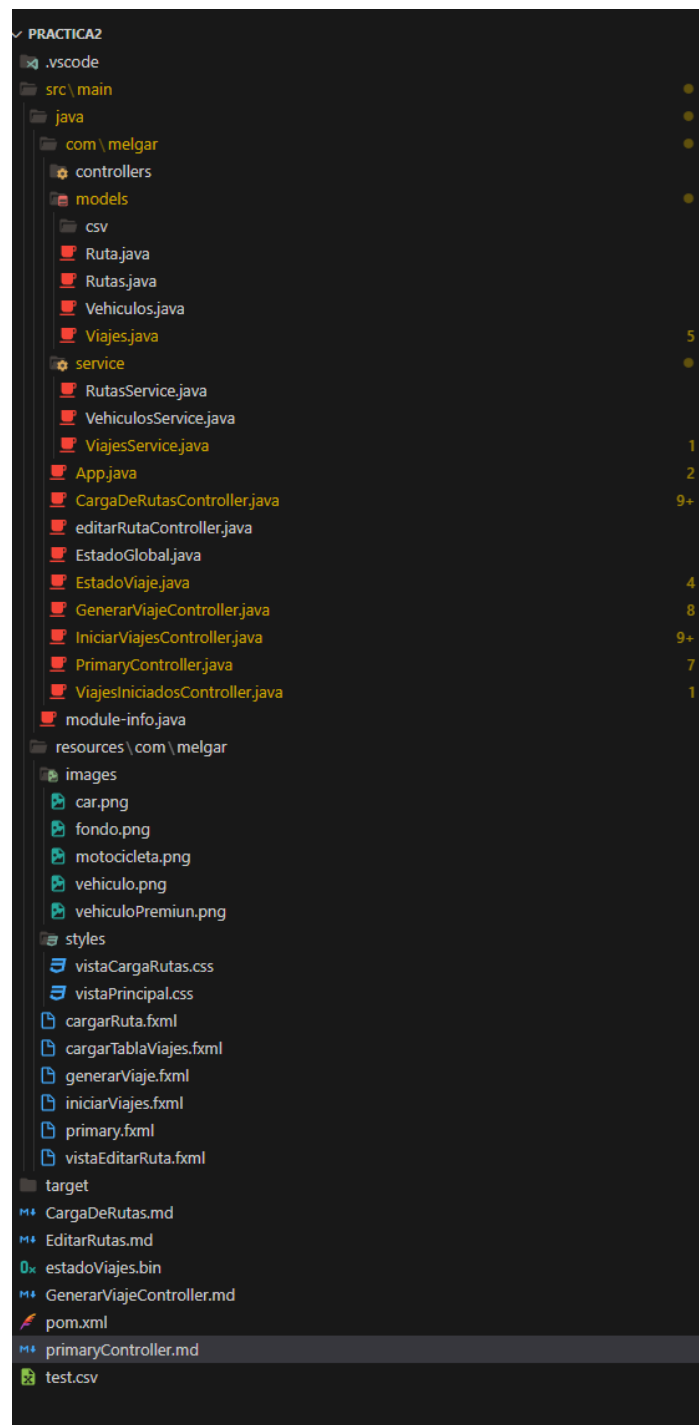
# Universidad de San Carlos de Guatemala

## Lenguaje utilizado

El juego en consola esta desarrollado en Java, para su ejecución nada mas necesitamos el uso del JDK del kit 21.0, siendo esta la versión mas nueva de nuestro compilador de Java.

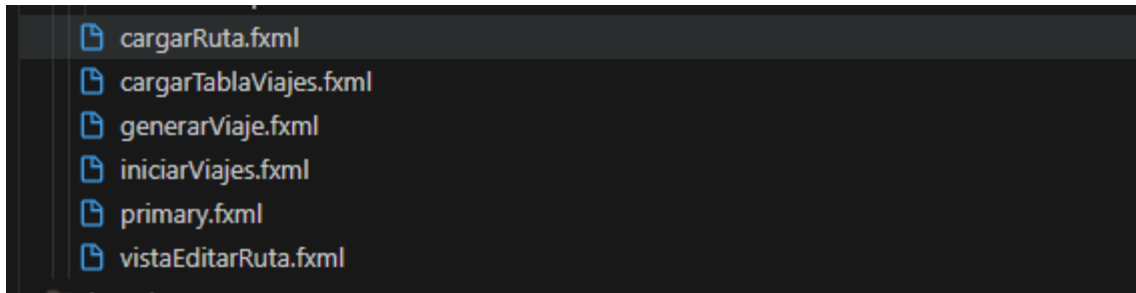
## Lógica utilizada

El programa este hecho con una lógica de las siguientes clases que este hecho a base de un proyecto de javaFX:



# Universidad de San Carlos de Guatemala

Donde tenemos un empaquetado que es de com.melgar adentro de estos paquetes podemos encontrar los servicios, recursos (adentro de los recursos encontramos las carpetas de imágenes, archivos css y los fxml) que son parte fundamental del proyecto que es donde vemos toda la vista de lo que estamos ejecutando en nuestros controladores.



Debemos de tomar en cuenta que es un proyecto basado en javaFx, entonces los elementos debemos inyectarlos a nuestro FXML de la siguiente forma:

```
@FXML
private void regresarMenu() throws IOException {
    App.setRoot(fxml:"primary");
}
```

Después de inyectarlos a nuestro FXML podemos usarlo de la siguiente manera:

```
<VBox xmlns="http://javafx.com/javafx/16" xmlns:fx="http://javafx.com/fxml/1" fx:controller="com.melgar.GenerarViajeController" stylesheets="@styles/vistaCargaRutas.css">
  <children>
    <AnchorPane>
      <children>
        <Button text="Regresar al Menú" layoutX="135" layoutY="10" onAction="#regresarMenu" />
        <Label text="Seleccionar punto inicial" layoutX="135" layoutY="80" />
        <ComboBox fx:id="puntoInicialComboBox" layoutX="135" layoutY="80" prefWidth="200" />
        <Label text="Seleccionar punto final" layoutX="135" layoutY="120" />
        <ComboBox fx:id="puntoFinalComboBox" layoutX="135" layoutY="120" prefWidth="200" />
        <Label text="Seleccionar tipo de transporte" layoutX="135" layoutY="160" />
        <ComboBox fx:id="tipoTransporteComboBox" layoutX="135" layoutY="160" prefWidth="200" />
      </children>
    </AnchorPane>
    <VBox prefHeight="50"/>
    <HBox alignment="CENTER">
      <Button text="Generar Viaje" prefWidth="200" onAction="#handleAgregarViaje"/>
    </HBox>
  </children>
</VBox>
```

# Universidad de San Carlos de Guatemala

También existe métodos de esta forma que lo hace es ejecutar de manera automática al iniciar la vista:

```
@FXML
public void initialize() {
    File file = new File(pathname:"viajes.bin");
    if (file.exists()) {
        try (ObjectInputStream ois = new ObjectInputStream(new FileInputStream(file))) {
            List<Viajes> viajesExistentes = (List<Viajes>) ois.readObject();
            contadorViajes = viajesExistentes.size();
            for (Viajes viaje : viajesExistentes) {
                vehiculosUsados.add(viaje.getNombreTransporte());
            }
        } catch (IOException | ClassNotFoundException e) {
            e.printStackTrace();
        }
    }
    cargarRutas();
    puntoInicialComboBox.setItems(FXCollections.observableArrayList(ubicaciones));
    puntoFinalComboBox.setItems(FXCollections.observableArrayList(ubicaciones));
    viajesService = ViajesService.getInstance();
    vehiculosService = new VehiculosService();
    List<Vehiculos> vehiculos = vehiculosService.getVehiculos();
    tipoTransporteComboBox.setItems(FXCollections.observableArrayList(vehiculos));
}
```

Después existen otros métodos que no se inyectan directamente a la vista, pero los podemos usar en los métodos que sí inyectamos a nuestra vista:

```
161 public void leerRutasDesdeCSV(String rutaArchivo, List<String> ubicaciones) {
162     try (BufferedReader br = new BufferedReader(new FileReader(rutaArchivo))) {
163         String linea;
164         br.readLine();
165
166         while ((linea = br.readLine()) != null) {
167             String[] partes = linea.split(regex:",");
168             if (partes.length >= 2) {
169                 String inicio = partes[0];
170                 String fin = partes[1];
171
172                 if (!ubicaciones.contains(inicio)) {
173                     ubicaciones.add(inicio);
174                 }
175
176                 if (!ubicaciones.contains(fin)) {
177                     ubicaciones.add(fin);
178                 }
179             }
180         }
181     } catch (IOException e) {
182         e.printStackTrace();
183     }
184 }
185 }
```

# Universidad de San Carlos de Guatemala

Es de suma importancia que nosotros estamos trabajando con lectura y escritura en archivos bin para hacer serializable nuestra aplicación, entonces al generar viajes y guardar el estado de los viajes tenemos la generación de estos archivos.

En iniciar viajes tenemos

```
estadoViaje.setCapacidad(capacidad.get());
});

tt.setOnFinished(e -> {
    estadoViaje.setAnimacionY(tt.getNode().getTranslateY());
    estadoViaje.setAnimacionDuracion(tt.getDuration());

    // Agrega estadoViaje a la lista
    estadosViaje.add(estadoViaje);
    try (ObjectOutputStream oos = new ObjectOutputStream(
        new FileOutputStream(name:"estadoViajes.bin"))) {
        oos.writeObject(estadosViaje);
    } catch (IOException ex) {
        ex.printStackTrace();
    }
});
});
```

Y en la generación de viajes podemos crear nuestro archivo de la siguiente forma:

```
vehiculosSeleccionados.add(vehiculosSeleccionados.getNombre());
contadorViajes++;
System.out.println("Contador de viajes: " + contadorViajes);

Alert alert = new Alert(AlertType.INFORMATION);
alert.setTitle("Información");
alert.setHeaderText(null);
alert.setContentText("El viaje se agregó con éxito.");
alert.showAndWait();

File file = new File(pathname:"viajes.bin");
if (!file.exists()) {
    try {
        file.createNewFile();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

try (ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream(file))) {
    oos.writeObject(viajesService.getViajes());
} catch (IOException e) {
    e.printStackTrace();
}

} else {
    Alert alert = new Alert(AlertType.ERROR);
    alert.setTitle("ERROR");
    alert.setHeaderText(null);
    alert.setContentText("POR FAVOR LLENE TODOS LOS DATOS");
    alert.showAndWait();
}
```

# Universidad de San Carlos de Guatemala

Cabe aclarar que también estamos trabajando con un archivo CSV que este se lee de esta forma utilizando un FileChooser para navegar por los archivos de mi computadora.

```
@FXML
private void handleCargarRutas(ActionEvent event) {
    FileChooser fileChooser = new FileChooser();
    File file = fileChooser.showOpenDialog(null);
    if (file != null) {
        List<Rutas> rutas = new ArrayList<Rutas>();
        tableView.getItems().clear();

        try {
            CSVReader reader = new CSVReaderBuilder(new java.io.FileReader(file)).withSkipLines(skipLines:1).build();
            String[] nextLine;

            while ((nextLine = reader.readNext()) != null) {
                Rutas ruta = new Rutas(nextLine[0], nextLine[1], Float.parseFloat(nextLine[2]));

                rutas.add(ruta);

                RutasService.addRoute(ruta);

                tableView.getItems().add(ruta);
            }

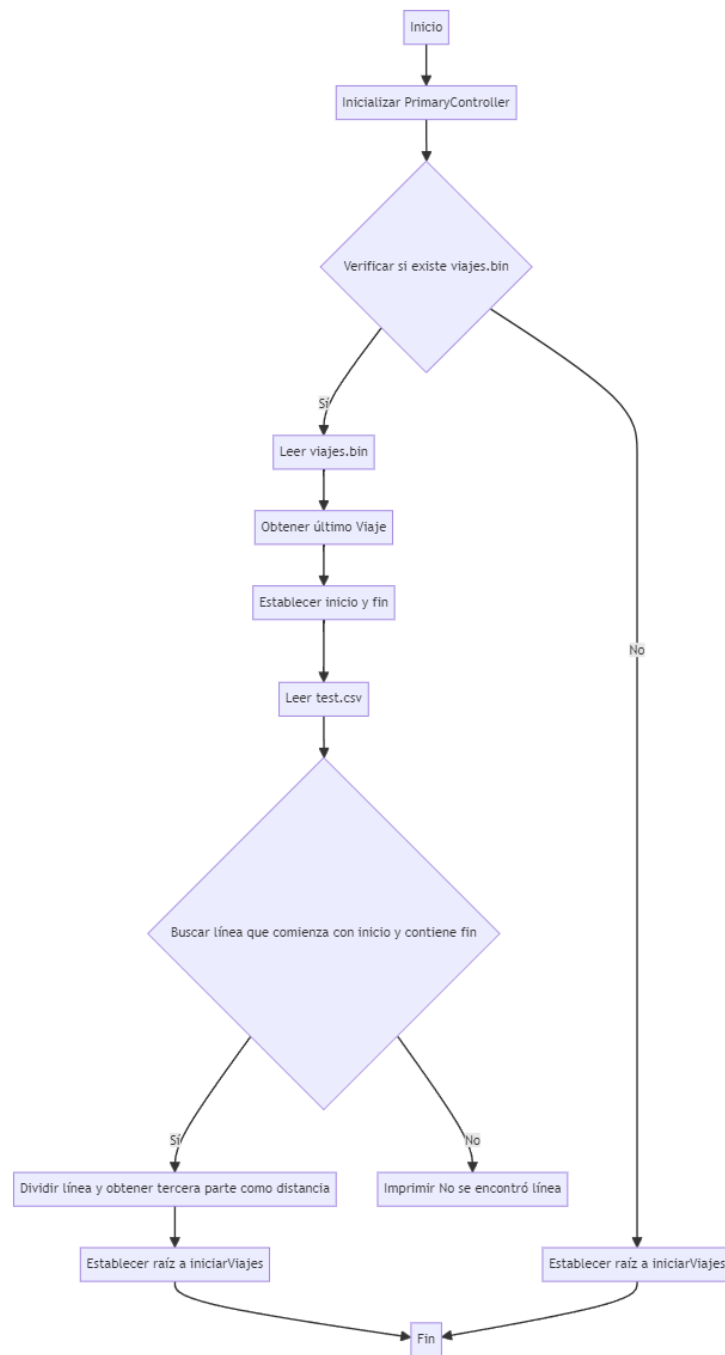
            EstadoGlobal estadoGlobal = EstadoGlobal.getInstance();
            estadoGlobal.setRutasCargadas(rutasCargadas:true);
        } catch (Exception e) {
        }
    }
}
```

Y de esa forma es como leemos nuestro archivo CSV que tiene esta forma:

```
test.csv > data
1 inicio, fin, distancia
2 Cuidad de Guatemala, La Mesilla, 25
3 Flores, El Estor, 20
4 El tejar, Zaragoza, 60
5 Villa Nueva, Mixco, 15
6 Moyuta, San Pedro la Laguna, 300
```

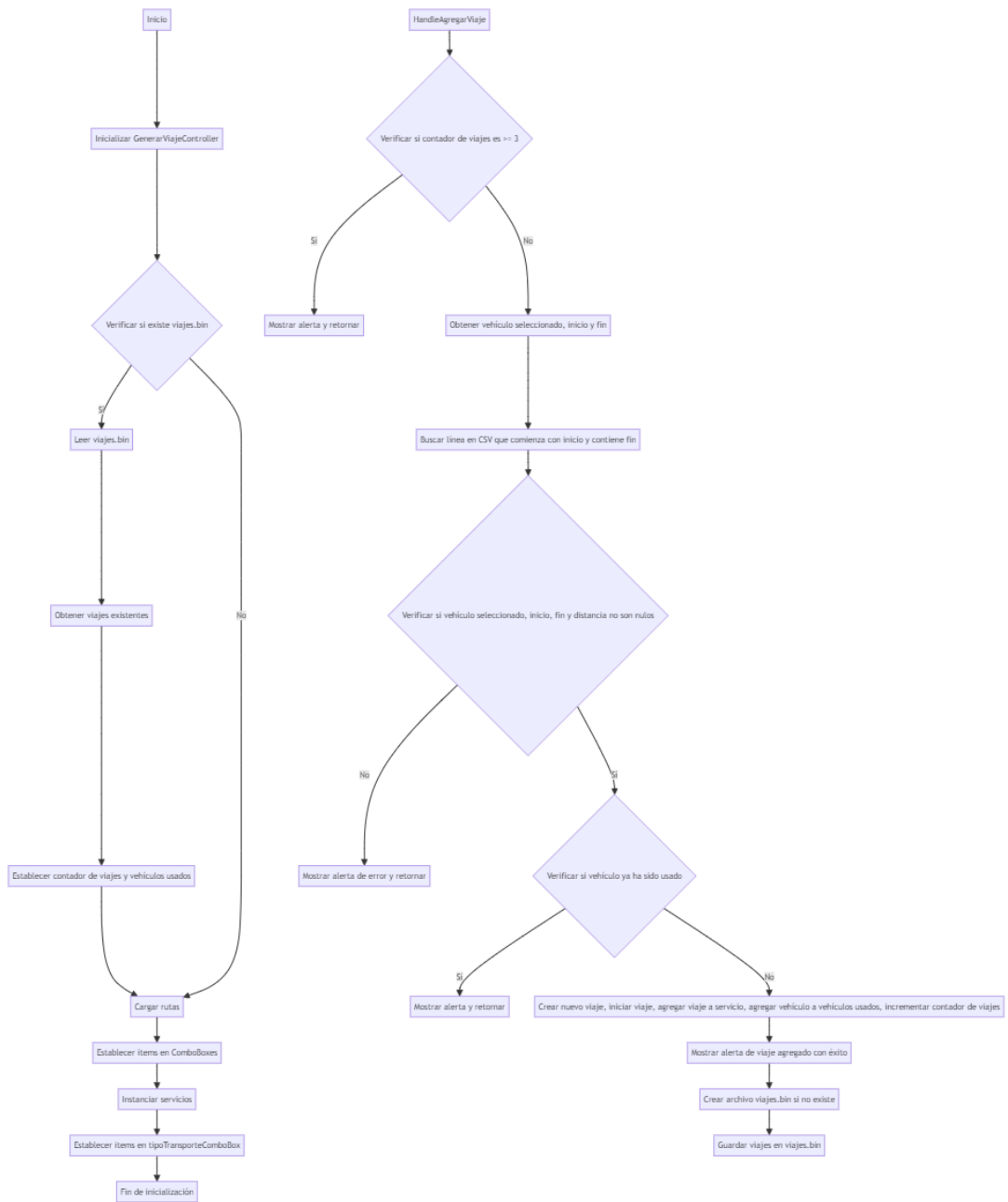
# Universidad de San Carlos de Guatemala

Diagramas de flujo (clases importantes):

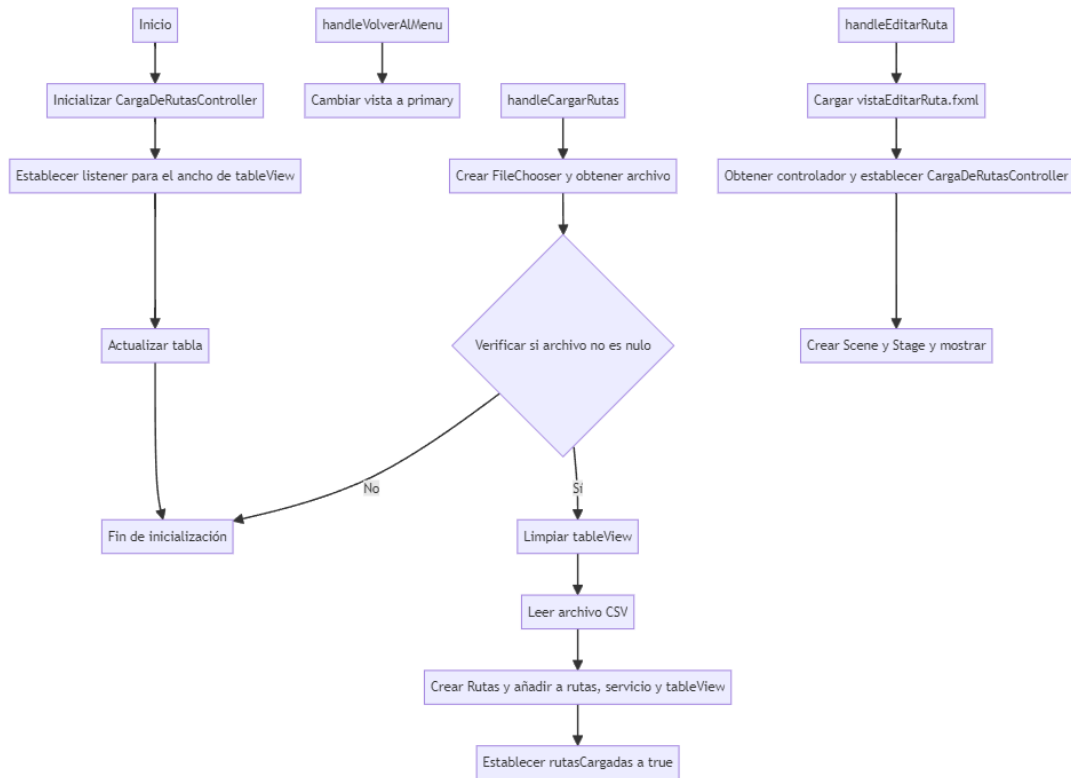




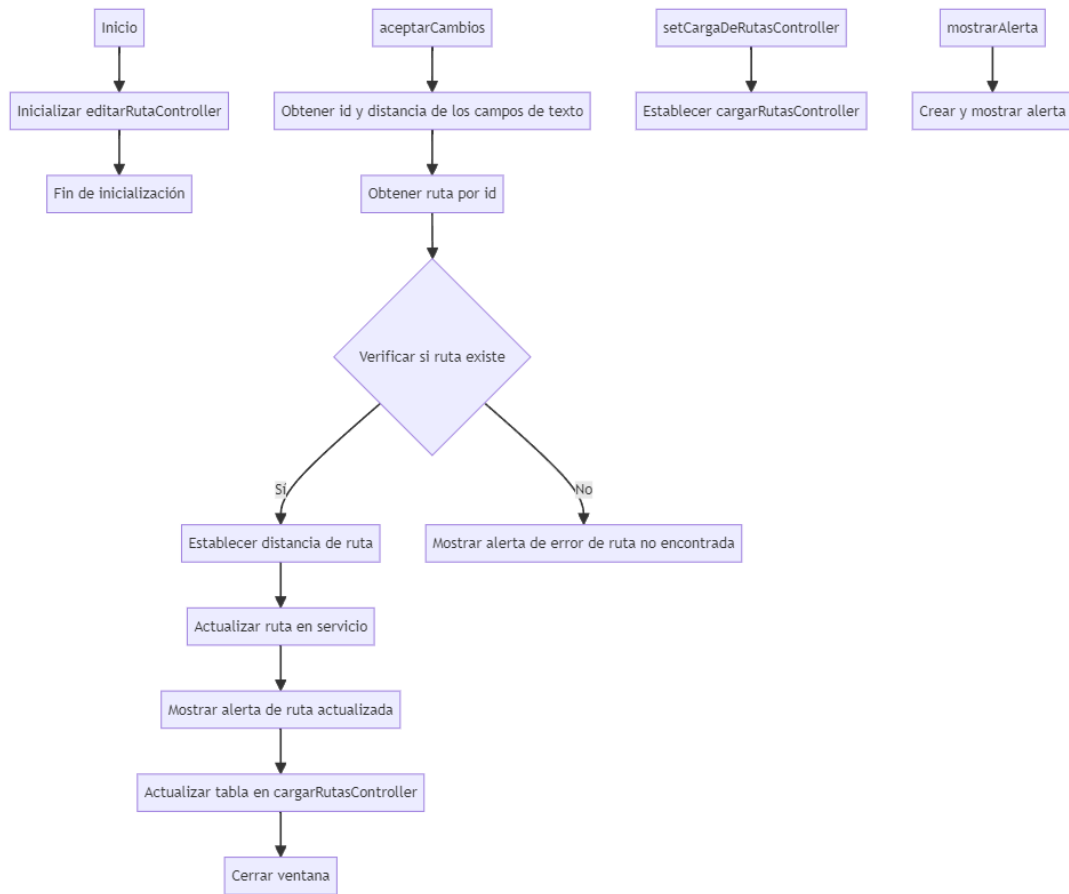
# Universidad de San Carlos de Guatemala



# Universidad de San Carlos de Guatemala



# Universidad de San Carlos de Guatemala



# Universidad de San Carlos de Guatemala

## UML

