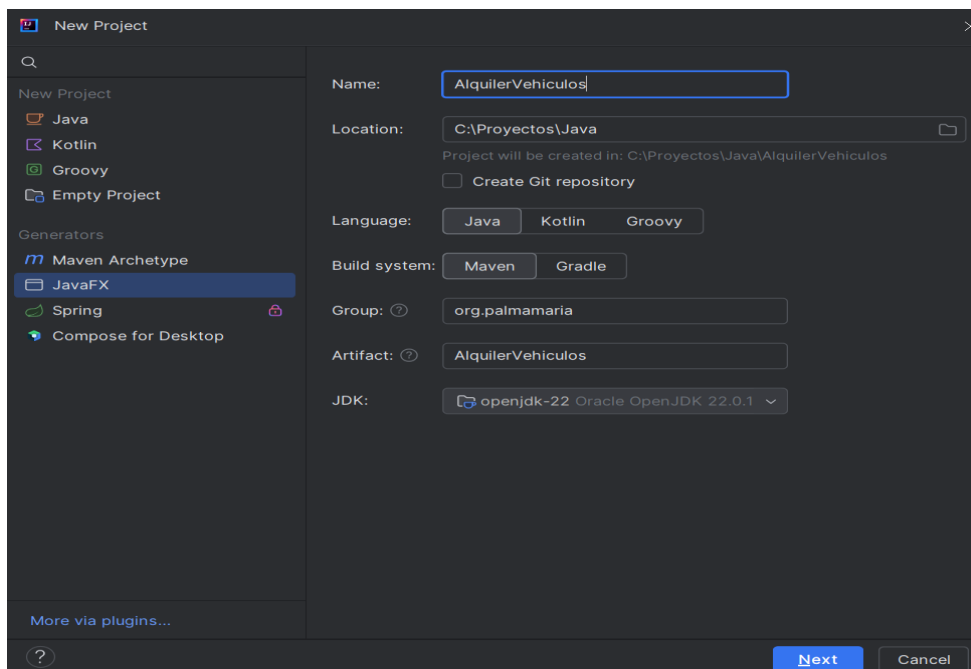


TAREA 5: DESARROLLO DE INTERFACES EN XML

Para comenzar a crear nuestro proyecto de interfaz en JavaFx, nos metemos en Java y le damos a new project. Indicaremos el nombre que le vamos a poner a nuestro proyecto y le damos a la opción de JavaFx.



A continuación creamos dentro del main las siguientes clases con las que vamos a interactuar con sus respectivas clases fxml.

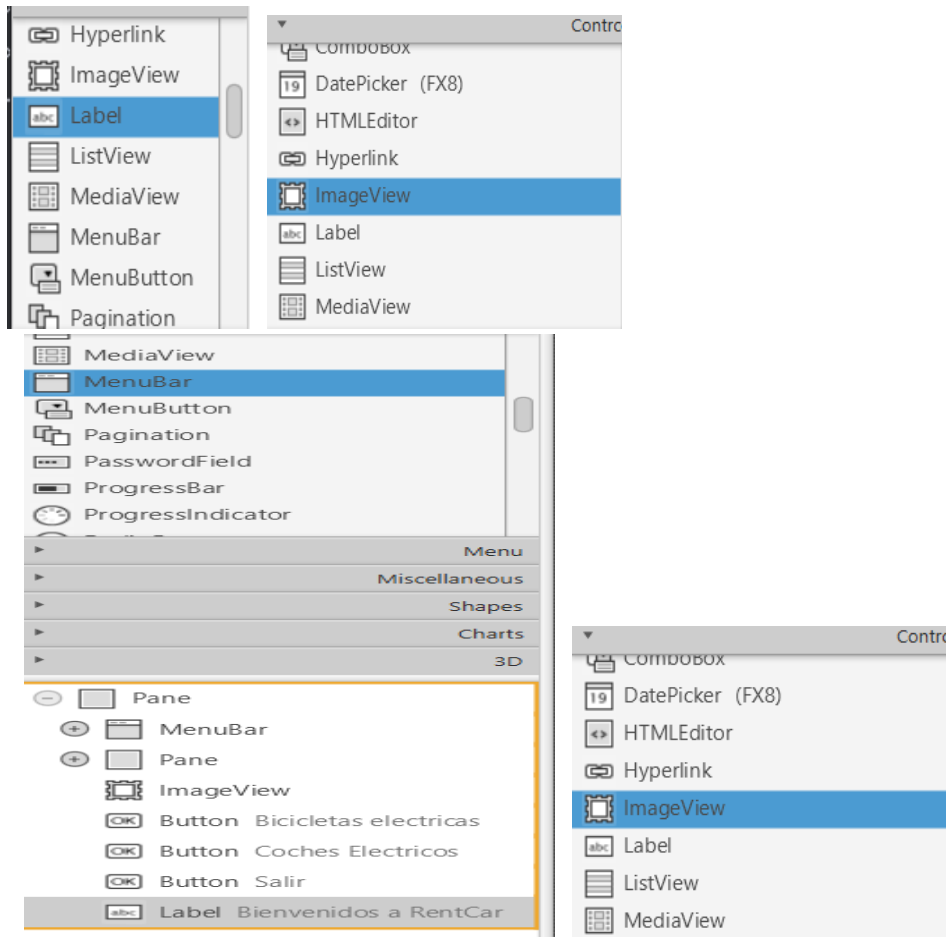


Primero vamos a comenzar con hacer la interfaz principal en MainMenu.fxml.

Un menú principal con las opciones: "Coche eléctrico", "Bicicleta eléctrica", y un botón para salir de la aplicación y un título de nuestra aplicación.

Para el título he utilizado un label.

Para las opciones del menú he utilizado un MenuBar con sus respectivos hijos en la opción MenuItem, un botón para salir de la aplicación con Button y un ImageView para ponerle la imagen que he utilizado.



La opción de Bicicleta eléctrica te lleva a una pequeña ventana que te dice que no disponemos de bicicletas en este momento:

Este trozo de código lo implementamos en la clase MainController que es la clase fxml de nuestra clase MainController

```
@FXML
private void mostrarMensajeError() {
    // Mostrar mensaje de error para bicicletas
    Alert alert = new Alert(Alert.AlertType.ERROR);
    alert.setTitle("Error");
    alert.setHeaderText(null);
    alert.setContentText("NO DISPONEMOS DE BICICLETAS EN ESTE MOMENTO");
    alert.showAndWait();
}
```

La opción de coches eléctricos te llevará a la segunda interfaz que vamos a crear. Este interfaz se conecta con la otra mediante `abrirFormulario`, vinculado en el `CarRentalForm.fxml`

```
private void abrirFormulario() {
    try {
        // Cargar el archivo FXML del formulario
        FXMLLoader loader = new FXMLLoader(getClass().getResource("name: \"CarRentalForm.fxml\""));
        Scene scene = new Scene(loader.load());

        // Crear una nueva ventana para el formulario
        Stage stage = new Stage();
        stage.setScene(scene);
        stage.setTitle("Formulario de Alquiler");
        stage.show();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

Y el botón salir hará que nos salgamos de la aplicación:

```
@FXML
private void salirAplicacion() {
    // Salir de la aplicación
    System.exit( status: 0);
}
```

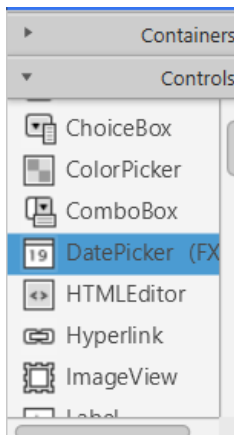
En la segunda interfaz que vamos a realizar la hacemos igual que la primera. En las opciones de coches eléctricos te llevará a la segunda interfaz en la que está el formulario con campos para que el usuario ingrese sus datos (Nombre, Apellidos, Teléfono, DNI). Esto lo he hecho con labels.

Para los campos de nombre y apellidos le he puesto un `ToolTip` a cada uno que lo he implementado en la parte de texto en `CarRentalForm.fxml`

```
<Label layoutX="66.0" layoutY="86.0" prefHeight="17.0" prefWidth="61.0" text="Nombre:" />
<TextField fx:id="nombreField" layoutX="140.0" layoutY="82.0" prefHeight="25.0" prefWidth="200.0">
    <tooltip>
        <Tooltip text="Introduzca su nombre" />
    </tooltip>
</TextField>
<Label layoutX="66.0" layoutY="132.0" prefHeight="17.0" prefWidth="61.0" text="Apellidos: " />
<TextField fx:id="apellidoField" layoutX="140.0" layoutY="128.0" prefHeight="25.0" prefWidth="200.0">
    <tooltip>
        <Tooltip text="Introduzca sus apellidos" />
    </tooltip>
</TextField>
```

Datos de alquiler de coche eléctrico debe incluir:

- Fechas de inicio y fin del alquiler.

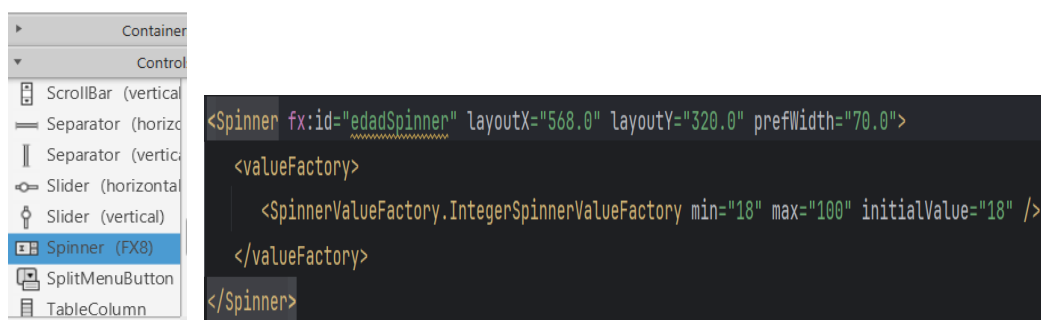


- Tipo de vehículo (5 opciones).



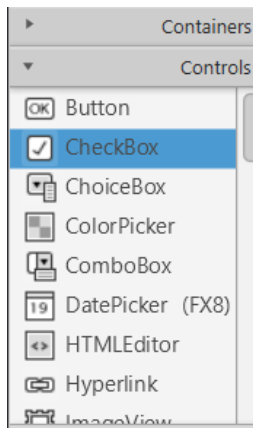
En la clase CarRentalForm.fxml implementamos ComboBox para meterle los diferentes tipos de vehículos que hay.

- Edad del conductor.

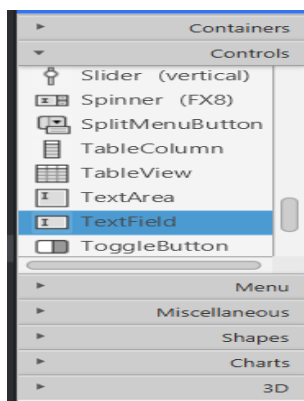


En la clase CarRentalForm.fxml implementamos para la edad Spinner metiéndole un mínimo de 18 y máximo un 100.

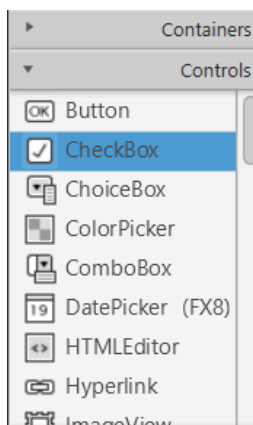
- Necesidad de cable de carga.



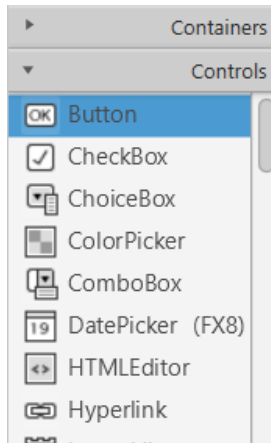
- Número de kilómetros.



- Servicios extras como cadenas, cancelación gratuita, seguro, silla bebé, etc. Para todas estas opciones le he puesto un checkBox a cada una.



Botones Cancelar y Reservar, donde el primero regresa al menú principal y el segundo muestra un mensaje de éxito. Esto lo hacemos desde CarRentalController.



```
@FXML
private void completarReserva() {
    // Mostrar mensaje de confirmación
    Alert alert = new Alert(Alert.AlertType.INFORMATION);
    alert.setTitle("Reserva Completada");
    alert.setHeaderText(null);
    alert.setContentText("¡Reserva realizada con éxito!");
    alert.showAndWait();
}

@FXML
private void irAlMenuPrincipal() {
    // Cerrar el formulario y volver al menú principal
    Stage stage = (Stage) nombreField.getScene().getWindow();
    stage.close();
}
```