

# Proyector-Pangel

Proyector-Pangel es una aplicación de simulación de la temporada 2007 de Fórmula 1, desarrollada en **Java** con **Swing** y persistencia **SQL en MySQL**.

Permite gestionar y simular equipos y pilotos de F1.

## Tabla de contenidos



Características

Funcionalidades principales de la aplicación



Estructura

Organización de archivos y carpetas



Base de datos

Diseño y operaciones SQL



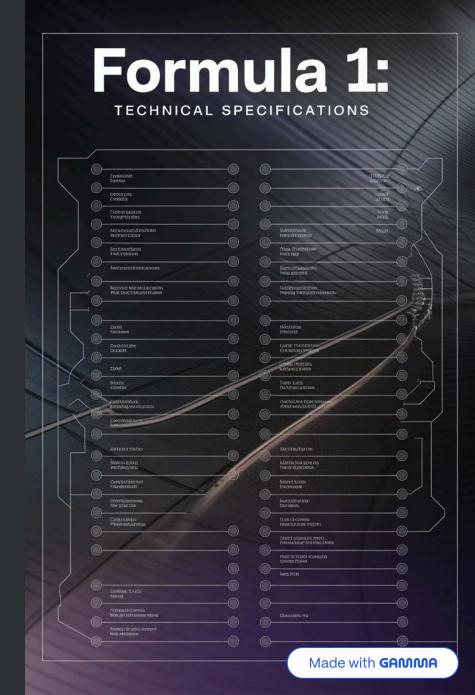
Programación

Interfaz Grafica, Eventos, POO



Entornos De Desarrollo

Junit, Depuración, Refactorización





### Características

### Simulación completa

Recreación visual de la temporada 2007 de Fórmula 1.

#### Gestión avanzada

Edición de equipos y pilotos con búsqueda y filtrado.

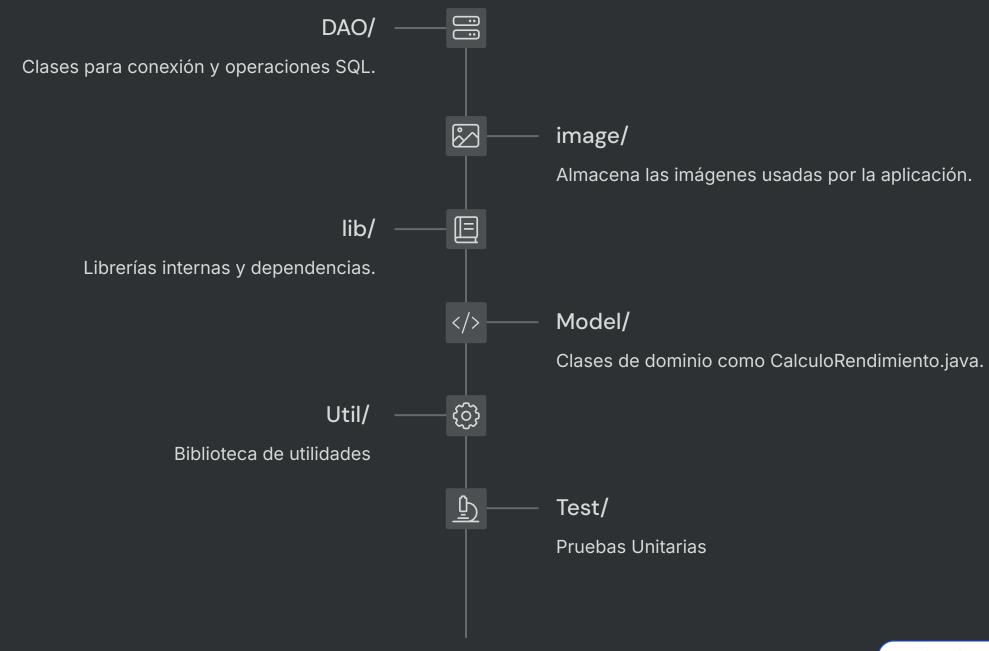
#### Persistencia de datos

Almacenamiento con MySQL mediante phpMyAdmin/XAMPP.

#### Interfaz moderna

Diseño visual atractivo implementado con Java Swing.

# Estructura del proyecto

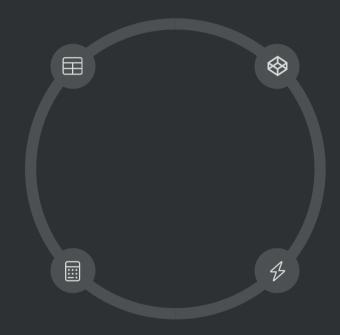


### Diseño del Esquema

Esquema de relación de tablas incluido en carpeta BBDD.

### Función SQL

Calcula el promedio de habilidad de los pilotos.

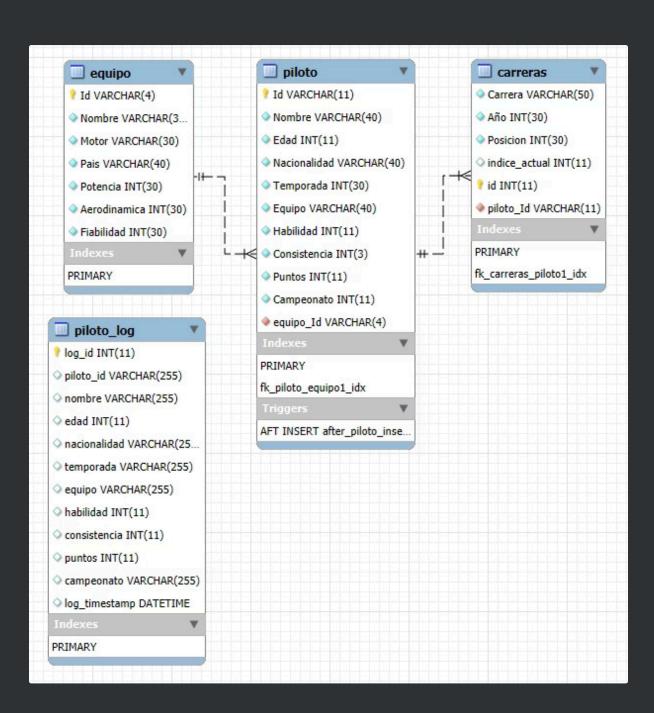


### **Operaciones SQL**

Incluye SELECT, INSERT, UPDATE y DELETE.

### Trigger

Implementado para registrar logs de pilotos añadidos.



#### Relaciones entre tablas

Se han establecido relaciones entre la tabla equipo, piloto y carreras mediante el uso de PK y FK

Se ha añadido la tabla piloto\_log pero no participa en la relación

```
BEGIN

DECLARE promedio DECIMAL(5,2);

SELECT AVG(Habilidad) INTO promedio

FROM piloto;

RETURN promedio;

END
```

#### Función

Calcula el promedio de habilidad de los pilotos.

```
// TRIGGER
String createTriggerSQL =
    "CREATE OR REPLACE TRIGGER after_piloto_insert " +
    "AFTER INSERT ON piloto " +
    "FOR EACH ROW " +
    "BEGIN " +
    " INSERT INTO piloto_log (" +
    " piloto_id, nombre, edad, nacionalidad, temporada, " +
    " equipo, habilidad, consistencia, puntos, campeonato" +
    " ) " +
    " VALUES (" +
    " NEW.Id, NEW.Nombre, NEW.Edad, NEW.Nacionalidad, NEW.Temporada, " +
    " NEW.Equipo, NEW.Habilidad, NEW.Consistencia, NEW.Puntos, NEW.Campeonato" +
    " );" +
    "END;";
```

### Trigger

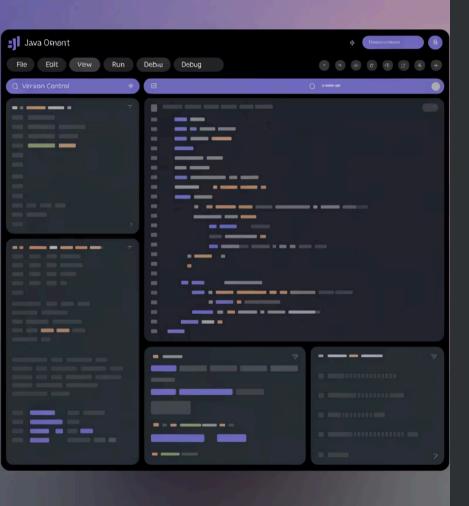
Implementado en AnadirPilotos.java (línea 239).

Función: Crea una tabla en la base de datos para registrar logs de pilotos añadidos.

```
// Inserta un nuexo equipo
String sentencia = "INSERT INTO equipo (Id, Nombre, Motor, Pais, Potencia, Ad
    "'" + textId.getText() + "', " +
    "'" + textNombre.getText() + "', " +
    "'" + textMotor.getText() + "', " +
    "'" + textPais.getText() + "', " +
    textPotencia.getText() + ", " +
    textField_Aerodinamica.getText() + ", " +
    textField_Fiabilidad.getText() + ")";
```

### **Operaciones SQL**

Incluye SELECT, INSERT, UPDATE y DELETE





#### Pruebas unitarias

Verifican el comportamiento de unidades específicas de código con JUnit.



### Depuración

Reestructuración del código para mejorar legibilidad y eficiencia.



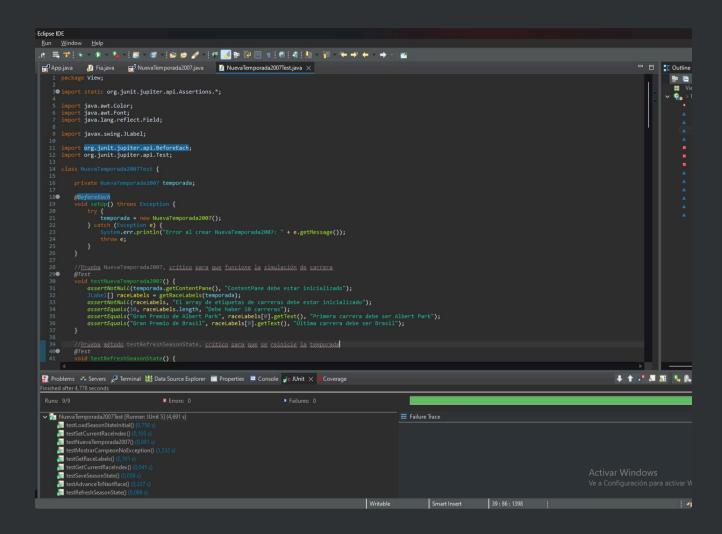
#### Control de Versiones

Progreso compartido mediante commits frecuentes.



#### Documentación

JavaDoc generado para clases y métodos públicos.



### Pruebas unitarias con Junit

Verifican el comportamiento de unidades específicas de código, como métodos o clases, asegurando que funcionen correctamente.

#### Refactorización

```
JButton btnNewButton = new JButton("Gestionar Pilotos");
btnNewButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        VerPilotos temp=new VerPilotos();
        temp.setVisible(true);
    });
btnNewButton.setFont(new Font("Tahoma", Font.PLAIN, 20));
btnNewButton.setBounds(113, 180, 272, 54);
contentPane.add(btnNewButton);
```

Pre-refactorización

```
JButton btnGestionarPilotos = new JButton("Gestionar Pilotos");
btnGestionarPilotos.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        VerPilotos temp=new VerPilotos();
        temp.setVisible(true);
    }
});
btnGestionarPilotos.setFont(new Font("Tahoma", Font.PLAIN, 20));
btnGestionarPilotos.setBounds(113, 180, 272, 54);
contentPane.add(btnGestionarPilotos);
```

Post-refactorización

```
Botán para visualizar los logs de inserciones de pilotos.
   Abre un diálogo que muestra una tabla con los registros de la tabla piloto_log.
JButton btnVerLogs = new JButton("VER LOGS");
btnVerLogs.setBounds(579, 270, 106, 37);
btnVerLogs.addActionListener(new ActionListener() {
       Maneja el evento de clic en el botón "Ver Logs".
      Llama al método para mostrar los logs en un diálogo.
       Oparam e El evento de acción generado al hacer clic en el botón.
    public void actionPerformed(ActionEvent e) {
        mostrarLogs();
});
contentPane.add(btnVerLogs);
// Configura la tabla de logs y el trigger al iniciar la xentana
setupPilotLogAndTrigger();
```

#### Javadoc

JavaDoc generado para clases y métodos públicos.

# Programación





# Menú principal



# Menú de Gestión



#### X

# VER PILOTOS C Refrescar



Buscar por Nombre Buscar por Equipo

Id	Nombre	Equipo	Nacionalidad	Temporada	Habilidad	Consistencia	
P1	Fernando Alo	Mclaren	España	2007	95	91	4
P2	Lewis Hamilton	Mclaren	Reino Unido	2007	94	93	
P3	Kimi Raikkonen	Ferrari	Finlandia	2007	93	93	Ш
P4	Felipe Massa	Ferrari	Brasil	2007	92	90	Ш
P5	Giancarlo Fisi	Renault	Italia	2007	83	81	
P6	Heikki Kovalai	Renault	Finlandia	2007	85	83	Ш
P7	Nick Heidfeld	BMW Sauber	Polonia	2007	88	88	
P8	Robert Kubica	BMW Sauber	Polonia	2007	87	86	
P9	Nico Rosberg	Williams	Alemania	2007	84	80	Ш
P10	Alexander Wurz	Williams	Reino Unido	2007	82	78	
P11	Rubens Barric	Honda	Brasil	2007	81	81	
P12	Jenson Button	Honda	Reino Unido	2007	83	82	Ш
P13	Sebastian Vett	Red Bull	Alemania	2007	88	85	Н
P14	Mark Webber	Red Bull	Australia	2007	82	78	
P15	Ralf Schumac	Toyota	Alemania	2007	81	75	
P16	Jarno Trulli	Toyota	Italia	2007	82	77	
P17	Anthony David	Super Aguri	Reino Unido	2007	79	74	-

Añadir Piloto

Modificar Piloto

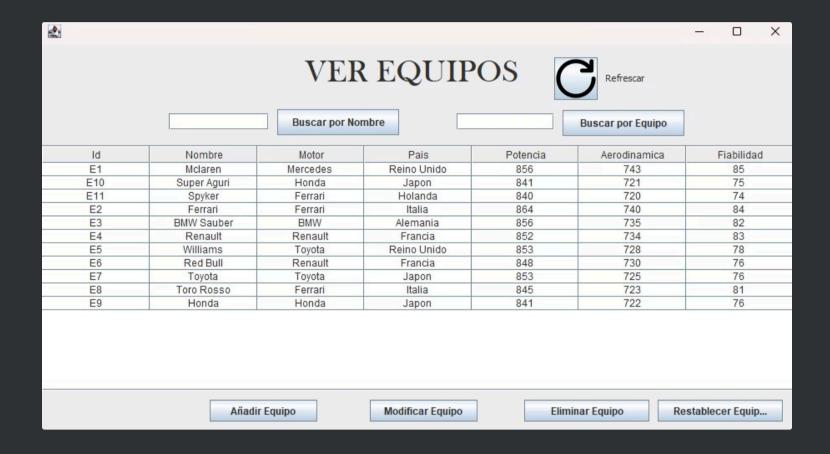
Eliminar Piloto

Restablecer Pilotos

Promedio Habilidad:

Calcular Promedio Habilidad

## Ver Pilotos



# Ver Equipos



## Temporada 2007

Gran Premio de Albert Park	R	Gran Premio de Monza	R
Gran Premio de Barein	R	Gran Premio de Spa	R
Gran Premio de España	R	Gran Premio de Suzuka	R
Gran Premio de Mónaco	R	Gran Premio de China	R
Gran Premio de Silverstone	R	Gran Premio de Brasil	R



Gestionar Temporada



Ver Clasificación



×

Avanzar a Carrera

# Nueva Simulación



# Gestionar Temporada

### CLASIFICACIÓN

#### **PILOTOS**

Puntos	Nombre	Equipo	Puntos	
1	Fernando Alonso	Mclaren	18	4
2	Felipe Massa	Ferrari	15	Г
3	Lewis Hamilton	Mclaren	14	П
4	Kimi Raikkonen	Ferrari	14	1
5 Robert Kubica		BMW Sauber	7	l
6	Heikki Kovalainen	Renault	7	1
7	Nick Heidfeld			۱.
8	Giancarlo Fisichella	Renault	1	ı
9	Sebastian Vettel	Red Bull	1	1
10 Scott Speed		Toro Rosso	0	1
11 Alexander Wurz		Williams	0	1
12 Christijan Albers		Spyker	0	1
13 Ralf Schumacher		Toyota	0	H
14	Anthony Davidson	Super Aguri	0	
15 Rubens Barrichello		Honda	0	
16	Nico Rosberg	Williams	0	
17	Jarno Trulli	Toyota	0	•

#### **EQUIPOS**

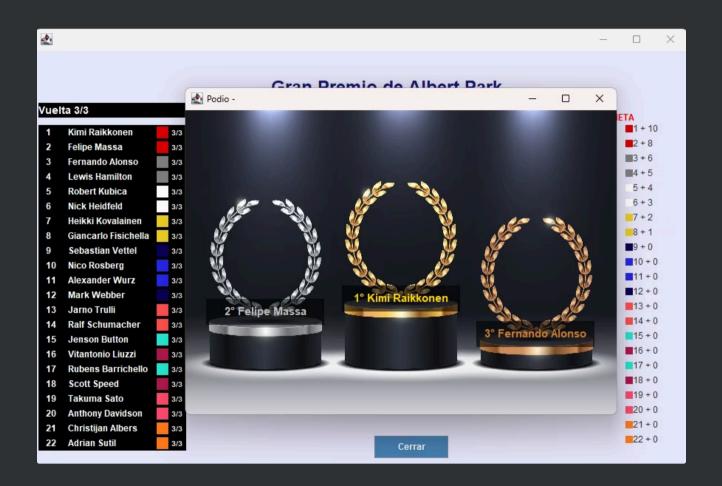
Puntos	Equipo	Puntos Totales	
1	Mclaren	32	
2	Ferrari	29	
3	BMW Sauber	11	
4	Renault	8	
5	Red Bull	1	
6	Toro Rosso	0	
7	Honda		
8	Super Aguri	0	
9	Williams	0	
10 Toyota		0	
11	Spyker	0	

ACTUALIZ.

# Clasificación de Pilotos y Equipos



## **Gran Premio**



# Finalización de Gran Premio

Se guardan los resultados y aparece ventana de Podio

## Notas



### Personalización

Modifica la parrilla y estadísticas antes de iniciar la simulación.



### Base de datos

Revisa los métodos de acceso para deducir la estructura.



### Ampliación

Proyecto ampliable para nuevas temporadas y funcionalidades.



## Creadores



PabloBG

Desarrollador principal



AngelJC

Desarrollador principal

# ¡Disfruta la experiencia F1!

