

Manual de Programador

ADMINISTRACIÓN DE FINCAS



Cristoffer Alcalde

Índice

1. Introducción
2. Análisis
 - 2.1. Requisitos
 - 2.2. Ciclo de Vida
3. Diseño
 - 3.1. Diagramas de casos de uso
 - 3.2. Modelo Entidad-Relación
 - 3.3. Modelo Relacional
 - 3.4. Estructura de las tablas de BD
 - 3.5. Diagrama de Clases
4. Diseño Ventanas
 - 4.1. Ventana Fichar
 - 4.2. Ventana Acceso al Sistema
 - 4.3.
5. Codificación
6. Pruebas
 - 6.1. Prueba Escritorio
 - 6.2. Ventanas
 - 6.2.1. Ventana Gestión Tareas
 - 6.2.2. Ventana Gestión Cuadrilla
 - 6.2.2.1. Insertar Cuadrilla
 - 6.2.2.2. Borrar Cuadrilla
 - 6.2.2.3. Buscar Cuadrilla
 - 6.2.3. Ventana Gestion Trabajador
 - 6.2.3.1. Insertar Trabajadores
 - 6.2.3.2. Modificar Trabajadores
 - 6.2.3.3. Borrar Trabajadores
 - 6.2.3.4. Buscar Trabajadores
 - 6.2.4. Ventana Gestión Fruta
 - 6.2.4.1. Insertar Fruta

6.2.4.2. Borrar Fruta

6.2.4.3. Buscar Fruta

7. Pruebas Realizadas

8. Android

8.1. Diagrama de Clases Android

8.2. Ventanas

8.2.1. Acceso al Sistema

8.2.2. Usuario Perito

8.2.2.1. Insertar Informe

8.2.2.2. Modificar Informe

8.2.2.3. Borrar Informe

8.2.3. Usuario Manijero

8.2.3.1. Insertar tarea

8.2.3.2. Modificar tarea

8.2.3.3. Borrar Tarea

8.2.3.4. Lista de Personas Presentes

9. Código Servidor

1. Introducción:

El objetivo de la aplicación es permitir al encargado de la finca (administrador) consultar las tareas que están llevando,

Gestionar a los trabajadores, las cuadrillas, las frutas y consultar los informes de los peritos.

Y permitir a los manijeros gestionar sus respectivas cuadrillas, controlando la asistencia de sus trabajadores y gestionando la tareas, y a los peritos gestionar sus informes y enviarlos para que el encargado pueda revisarlo.

2. Análisis:

La empresa de EDÉN SL nos ha solicitado que diseñemos y creemos dos aplicaciones una de escritorio y otra móvil que nos permita guardar la información en nuestra BD y poder consultarla.

Aplicación de escritorio:

Esta aplicación será manejada por el Encargado de la finca y sus funciones van a ser:

- Dar de alta, borrar, modificar y consultar los trabajadores.
- Asignar a los trabajadores a las cuadrillas
- Dar de alta, borrar las cuadrillas.
- Dar de alta, borrar las frutas.
- Consultar las tareas que están llevando a cabo las cuadrillas.
- Consultar los informes de los peritos
- Fichar a los trabajadores a la entrada y la salida del trabajo.

Aplicación de móvil:

Esta aplicación será manejada por los manijeros y los peritos de la finca y sus funciones van a ser:

Usuario Manijero

- Consultar a los trabajadores de su cuadrilla que están presentes en el día.
- Consultar un listado de todos los trabajadores de su cuadrilla.
- Consultar de forma detallada los días, las horas y el sueldo de cada uno de sus trabajadores.
- Consultar el registro de de tareas que ha llevado a cabo la cuadrilla. (Cada manijero solo puede ver sus propios tareas y no lo de los otros)
- Insertar una nueva tarea, modificarla, borrarla y consultarla.

Usuario Perito

- Consultar el registro de los informes que ha creado y subido el perito. (Cada perito solo puede ver sus propios informes y no lo de los otros)
- Insertar un nuevo informe, modificarlo, borrarlo y consultarlo.

2.1. Requisitos de la aplicación:

Aplicación de Escritorio:

Requisitos funcionales:

1°Trabajadores:

Insertar, borrar, modificar, consultar los datos de los trabajadores.

2° Cuadrillas:

Insertar, borrar, modificar, consultar las cuadrillas.

3° Frutas:

Insertar, borrar, consultar las variedades de frutas.

4° Informes de los peritos:

Consultar y revisar los informes mandados por los peritos.

5° Fichar:

Fichar a los trabajadores a la entrada y la salida.

6° Tareas:

Consultar las tareas que se están llevando a cabo o supervisar el registro de las tareas.

Requisitos no funcionales:

1°Confidencialidad:

Confidencialidad con los datos de los trabajadores.

2° Controlar el acceso:

Esta funcion se lleva a cabo gracias a que el encargado de la finca es el único usuario autorizado para manejar la aplicacion.

3° Asignación:

Controlar las asignaciones de los tractoristas a los manijeros, y de los trabajadores a sus manijeros.

Aplicación Móvil:

La aplicación móvil al ser manejada por dos usuarios los requisitos se van a dividir en dos partes Manijero y Perito:

1º Usuario peritos:

Requisitos funcionales:

→ CRUD INFORMES:

Los peritos podrán crear informes, modificarlos, borrarlos, consultarlos y subirlos a la BD de esta forma los peritos podrán enviar de forma más rápida los informes para que el encargado pueda consultarlos.

Requisitos no funcionales:

→ Rapidez;

Los peritos tendrán una mayor agilidad y velocidad a la hora de crear y enviar los informes de la finca a la BD

→ -> Interfaz amigable

→ -> Gestión de excepciones y rapidez en los requisitos

2º Usuario manijeros:

→ Controlar la cantidad de personas por cuadrilla:

Los manijeros podrán controlar las personas de su cuadrilla y en caso de necesitar a más personas podrá pedirselo a los otros manijeros antes de que vayan cada uno a sus sitio o por el contrario si una cuadrilla tiene muchos trabajadores podrá repartirlos a las cuadrillas que no tengan para de esta manera tener a las cuadrilla con el mismo número de trabajadores.

→ Crear o modificar la tarea de la cuadrilla:

El maniiero podrá crear, modificar y borrar las tareas que están llevando a cabo la cuadrilla de modo que ésta esté lo más coordinada posible.

→ Obtener Información detallada:

Los trabajadores muchas veces no llevan la cuenta de los días ni las horas que lleva echada y tampoco de sueldo que llevan ganado, pues de esta manera el maniiero va a tener un opción de consultar los datos de los trabajadores para revisar de manera más detallada y quitar las dudas a los trabajadores que le pregunten y mostrárselo de forma más detallada.

Requisitos No funcionales

- > Los maniieros podrán actualizar las tareas que está llevando a cabo su cuadrilla.
- >Interfaz amigable
- >Gestión de excepciones y rapidez en los requisitos

Los Requisitos recomendados para poder manejar nuestra aplicación son:

- JDBC 8.0.31 para manejar la aplicación escritorio y vamos a usar php para la aplicación de móvil
- Apache NetBeans IDE 14 o superior.
- JDK 1.8 o superior.
- Xampp Control Panel v3.3 o superior para gestionar nuestro servidor Apache y servidor Mysql, donde podamos subir nuestros PHP y podamos crear nuestra BD y gestionarla.
- Sublime Text 3.2 o superior para visualizar los archivos php que se encuentran en (C:\xampp\htdocs\aff)

- Android Studio 12.0 o superior para poder visualizar el código de la aplicación
- Sistema operativo Windows 10, 8GB de RAM y un espacio en disco de 75 GB

2.2. Ciclo de Vida

He elegido el ciclo de cascada con realimentación:

He elegido este ciclo ya que de esta manera me permite volver atrás en cualquier momento para corregir, modificar o depurar la aplicación.

3. Diseño:

Vamos a usar el mysql para guardar los datos en nuestra BD y donde accederemos mediante la aplicación de escritorio usando el JDBC mencionado anteriormente con lenguaje java y para la aplicación de móvil usaremos php.

3.1. Diagramas de casos de uso

Diagrama De Caso de uso Aplicacion de Escritorio:

DIAGRAMA DE CASO DE USO DE ESCRITORIO

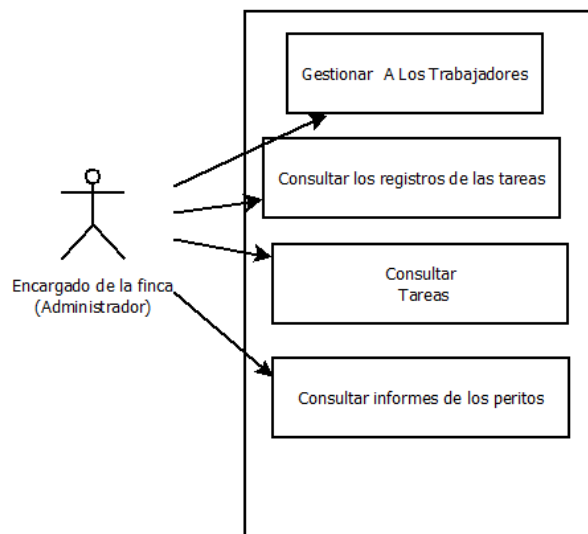
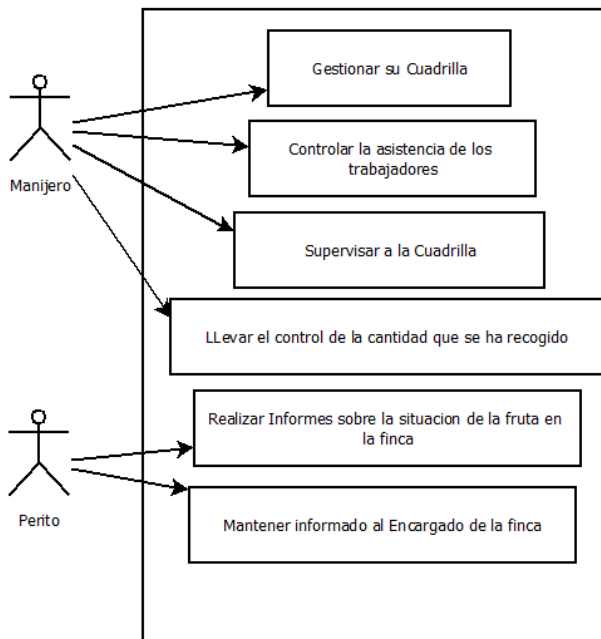


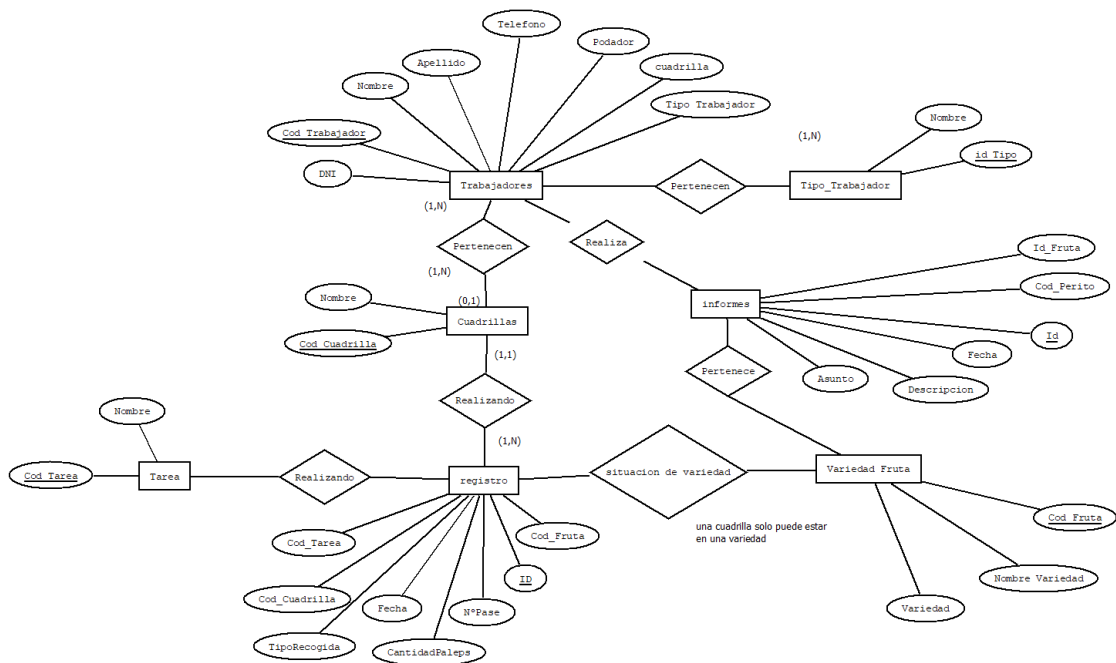
Diagrama de casos de uso de Aplicación Móvil:

DIAGRAMA DE CASO DE USO DE APLICACION MOVIL



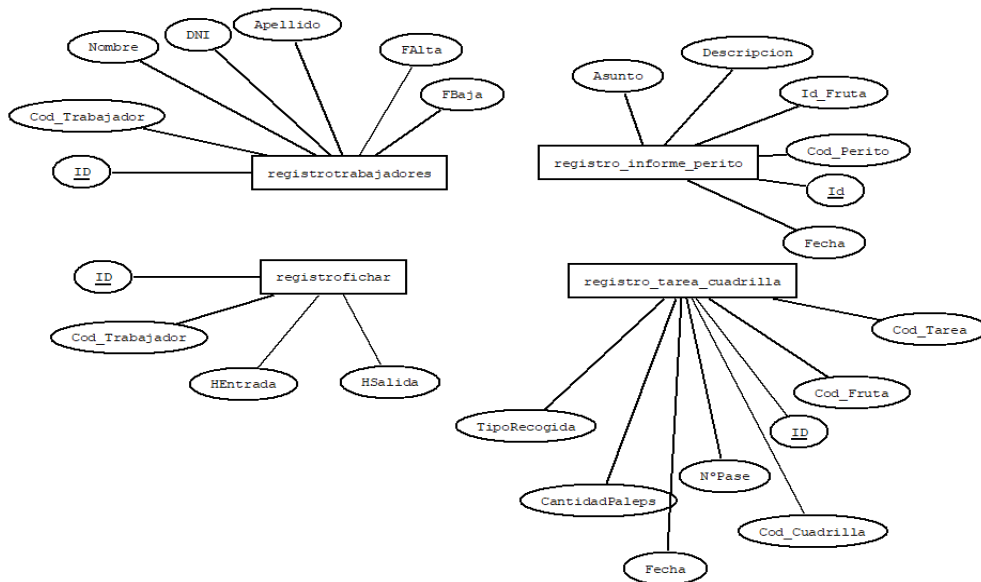
3.2. Modelo Entidad-Relación: E/R Relacionadas

DIAGRAMA ENTIDAD-RELACION

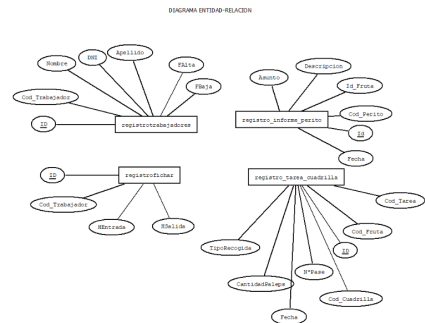
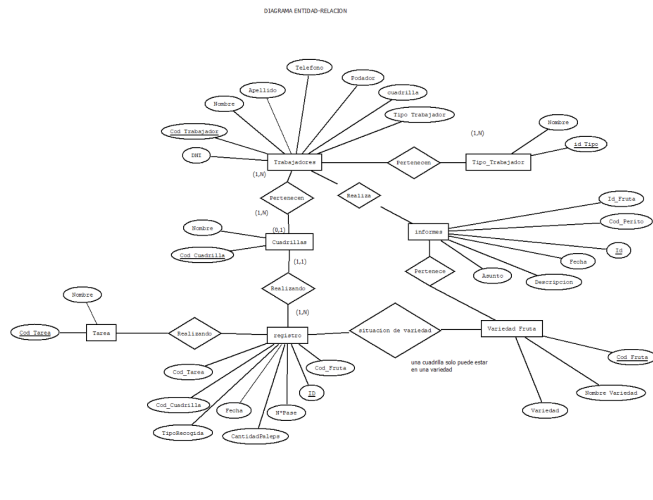


E/R Registro

DIAGRAMA ENTIDAD-RELACION



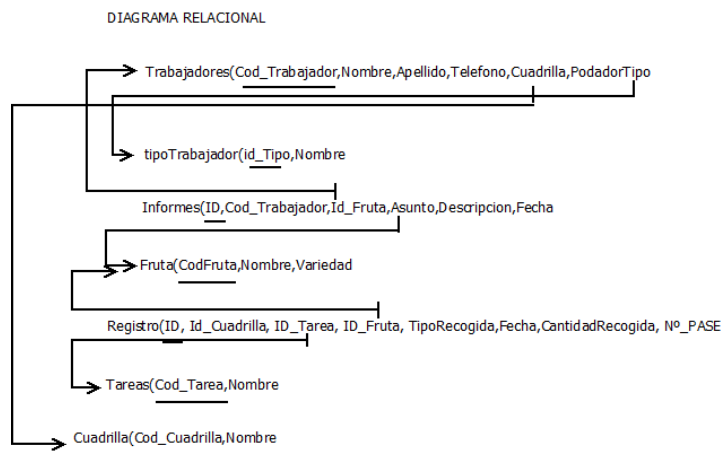
E/R Completo:



Estructura relacional

3.3. Modelo Relacional:

Modelo Relacional



RegistroFichar(ID, Cod_Trabajador, H/ENTRADA, H/SALIDA)

Registro_tarea_cuadrilla(ID, Id_Cuadrilla, ID_Tarea, ID_Fruta, TipoRecogida, Fecha, CantidadRecogida, Nº_PASE)

RegistroTrabajadores(ID, Cod_Trabajador, DNI, F/Alta, F/Baja, Nombre, Apellidos)

RegistroInformesPeritos(ID, Cod_Trabajador, Id_Fruta, Asunto, Descripcion, Fecha)

3.4. Estructuras de las tablas:

Tabla Cuadrilla:


	#	Nombre	Tipo	Cotejamiento
<input type="checkbox"/>	1	Cod_Cuadrilla 	int(11)	
<input type="checkbox"/>	2	nombre	varchar(20)	utf8mb4_general_ci

Tabla Fruta:


	#	Nombre	Tipo	Cotejamiento
<input type="checkbox"/>	1	Cod_Fruta 	int(11)	
<input type="checkbox"/>	2	Variedad	varchar(30)	utf8mb4_general_ci
<input type="checkbox"/>	3	nombre	varchar(20)	utf8mb4_general_ci

Tabla informes




	#	Nombre	Tipo	Cotejamiento	A
<input type="checkbox"/>	1	Id 	int(11)		
<input type="checkbox"/>	2	Cod_Perito 	int(11)		
<input type="checkbox"/>	3	Id_Fruta 	int(11)		
<input type="checkbox"/>	4	Asunto	varchar(50)	utf8mb4_general_ci	
<input type="checkbox"/>	5	Descripcion	varchar(500)	utf8mb4_general_ci	
<input type="checkbox"/>	6	Fecha	date		

Tabla Tipo trabajador;

	#	Nombre	Tipo	Cotejamiento	A
<input type="checkbox"/>	1	id_Tipo 	int(11)		
<input type="checkbox"/>	2	nombre	varchar(20)	utf8mb4_general_ci	

Tabla Registro:

#	Nombre	Tipo	Cotejamiento	Atributo
<input type="checkbox"/> 1	ID	int(11)		
<input type="checkbox"/> 2	ID_Cuadrilla	int(11)		
<input type="checkbox"/> 3	ID_Tarea	int(11)		
<input type="checkbox"/> 4	ID_Fruta	int(11)		
<input type="checkbox"/> 5	TipoRecogida	varchar(20)	utf8mb4_general_ci	
<input type="checkbox"/> 6	Fecha	date		
<input type="checkbox"/> 7	CantidadPaleps	int(11)		
<input type="checkbox"/> 8	N°_Pase	int(11)		

Tabla registro Fichar:

#	Nombre	Tipo	Cotejamiento
<input type="checkbox"/> 1	ID	int(11)	
<input type="checkbox"/> 2	Cod_Trabajador	int(11)	
<input type="checkbox"/> 3	HEntada	timestamp	
<input type="checkbox"/> 4	HSalida	timestamp	

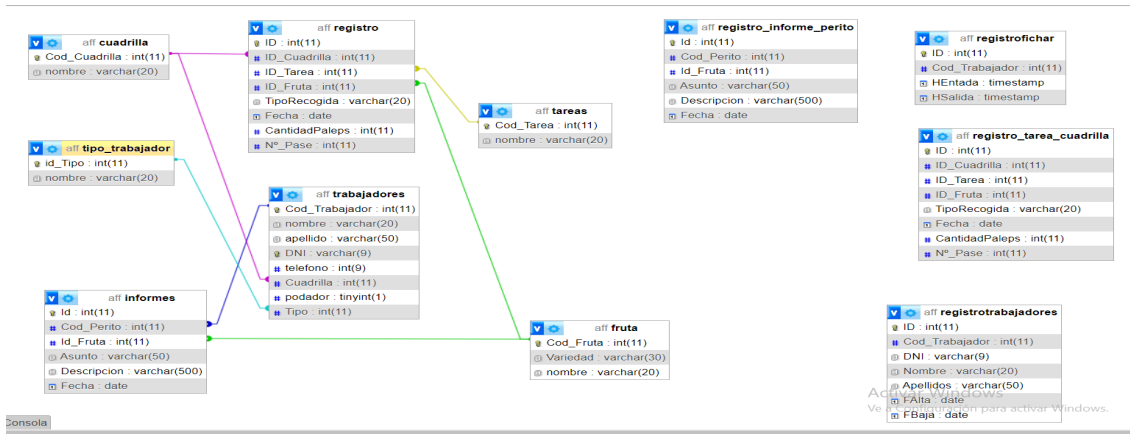
Tabla de todas las tareas disponibles a realizar

#	Nombre	Tipo	Cotejamiento
<input type="checkbox"/> 1	Cod_Tarea	int(11)	
<input type="checkbox"/> 2	nombre	varchar(20)	utf8mb4_general_ci

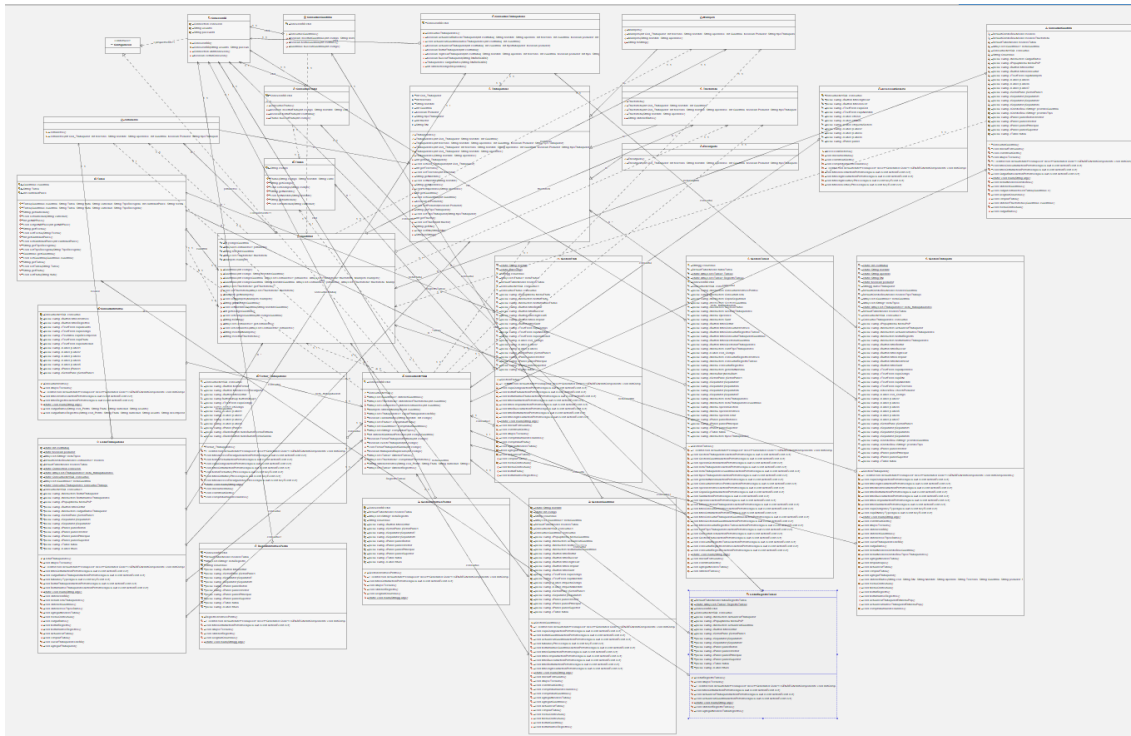
Tabla de los trabajadores:

#	Nombre	Tipo	Cotejamiento	Atributo
<input type="checkbox"/> 1	Cod_Trabajador	int(11)		
<input type="checkbox"/> 2	nombre	varchar(20)	utf8mb4_general_ci	
<input type="checkbox"/> 3	apellido	varchar(50)	utf8mb4_general_ci	
<input type="checkbox"/> 4	DNI	varchar(9)	utf8mb4_general_ci	
<input type="checkbox"/> 5	telefono	int(9)		
<input type="checkbox"/> 6	Cuadrilla	int(11)		
<input type="checkbox"/> 7	podador	tinyint(1)		
<input type="checkbox"/> 8	Tipo	int(11)		

Diseñador de la Base de Datos:



3.5. Diagrama de Clases de la aplicación de escritorio



4. Diseños Ventanas

4.1. Fichar Trabajadores:



EL EDÉN S.L

FICHAR

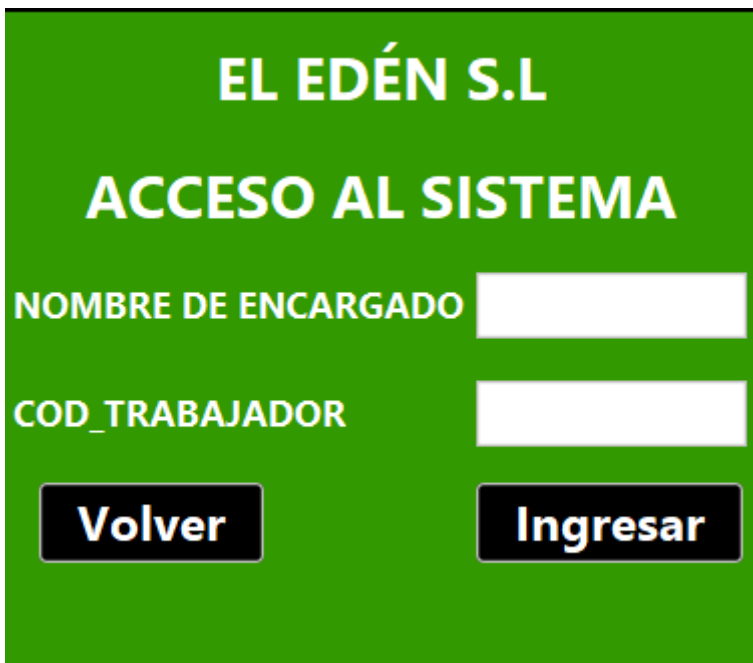
☐ H/ENTRADA ☐ H/SALIDA

Codigo

SISTEMA **Fichar**

CERRAR

4.2. Acceso Al Sistema



EL EDÉN S.L

ACCESO AL SISTEMA

NOMBRE DE ENCARGADO

COD_TRABAJADOR

Volver **Ingresar**

5. Codificación:

El lenguaje de programación que vamos a usar para la aplicación de escritorio es java, ya que es un lenguaje de programación orientada a objetos.













Para manejar la conexión de nuestra aplicación a nuestra BD en mysql vamos a usar un JDBC que es el siguiente:

Mysql-connector-java-8.2-bin.jar o superior
(IMAGEN DE LAS LIBRERÍAS)



Para manejar la conexión de nuestra aplicación móvil a nuestra BD en mysql vamos usar archivos PHP:

-Usaremos lenguaje PHP para gestionar nuestra aplicación android mediante Webservice.

Librerías usadas en la aplicación de escritorio:

- ▼  lib
 - >  commons-beanutils-1.9.4.jar
 - >  commons-collections4-4.4.jar
 - >  commons-digester-2.1.jar
 - >  commons-logging-1.2.jar
 - >  gson-2.10.1.jar
 - >  hamcrest-core-1.3.jar
 - >  itext-pdfa-5.5.8.jar
 - >  itextpdf-5.5.8.jar
 - >  jasperreports-6.20.0.jar
 - >  junit-4.13.2.jar
 - >  mysql-connector-j-8.0.31.jar

PHP usados para la aplicación de android:

 config	15/()
 controlador	26/()
 GestionarTareas	25/()
 GestionInformesPerito	24/()
 ObtenerDatosGenerales	25/()
 ObtenerDatosManijero	26/()
 ObtenerDatosPerito	26/()

6. Pruebas

6.1. Prueba Escritorio:

Ventana Fichar:

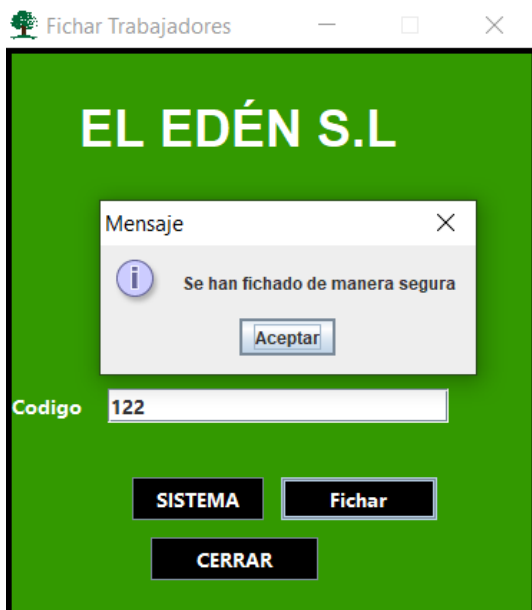
Para poder fichar a un trabajador a la hora de entrada o a la salida se necesita el código de trabajo que se le ha proporcionado y seleccionar la hora a la que ficha si es de entrada o salida.

The screenshot shows a green application window titled 'EL EDÉN S.L.' with the main heading 'FICHAR'. At the top, there are two radio buttons: 'H/ENTRADA' (selected) and 'H/SALIDA'. Below these is a text input field labeled 'Codigo'. At the bottom, there are three buttons: 'SISTEMA', 'Fichar', and 'CERRAR'.

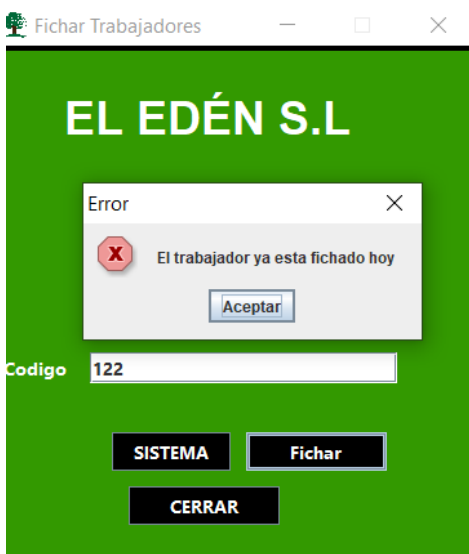
Hay que tener en cuenta que al introducir el código para fichar al trabajador la aplicación solo acepta valores numéricos. Si intentamos fichar con un código no registrado nos saldrá el siguiente error:

The screenshot shows a window titled 'Fichar Trabajadores' with a green background. It displays the same 'FICHAR' interface as the first screenshot. An error dialog box is overlaid on the screen with the title 'Error' and the message 'El código introducido no está registrado'. The dialog box has a red 'X' icon and an 'Aceptar' button. In the background, the 'Codigo' field contains the value '555555'.

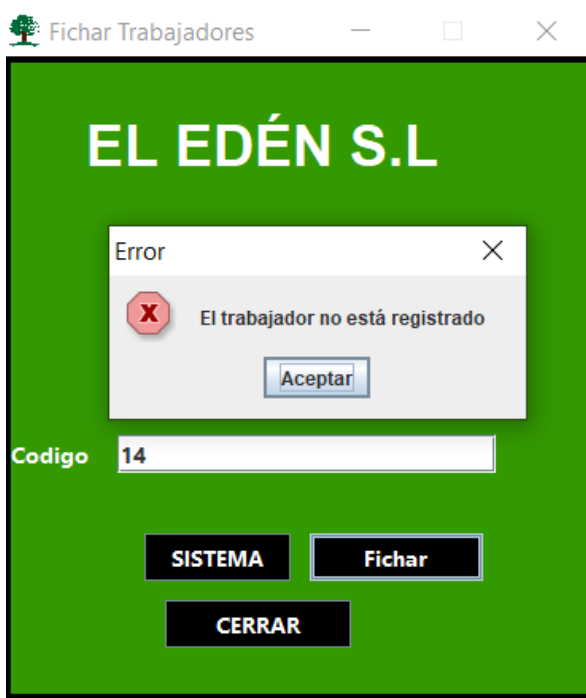
y si el codigo es correcto entonces



y si intentamos volver a fichar al trabajador a la entrada otra vez:



y si intentamos fichar a la hora de salida sin haber fichado antes en la hora de entrada:

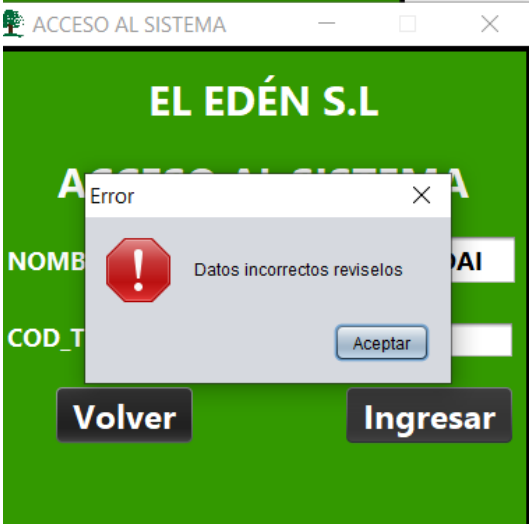


Ventana Acceso Al Sistema:

Para poder acceder al sistema en cualquier momento se necesita:
El nombre de un encargado y su respectivo código de otra
manera no tendrás acceso al sistema.

The image shows a window titled 'EL EDÉN S.L.' with a green background. Below the title, it says 'ACCESO AL SISTEMA' in white. There are two text input fields: the first is labeled 'NOMBRE DE ENCARGADO' and the second is labeled 'COD_TRABAJADOR'. At the bottom, there are two buttons: 'Volver' and 'Ingresar'.

Hay que tener en cuenta que al introducir el nombre solo acepta letras y para el código solo acepta valores numéricos.
Si introducimos valores incorrectos



y si introducimos datos válidos se nos abrirá la siguiente ventana:

TAREAS ASIGNADAS							
Cuadrilla	Tarea	Fruta	Variedad	Fecha	Tipo Recogida	Pase Realizado	Cantidad Recogi...
2	Entresacar	Melocoton	Baby God	2023-05-26	Cubo	1	0
2	INDICA LA TAREA		INDICA LA FRUTA	2023-05-26	INDICA EL TIPO ...	1	0

6.2. Ventanas

6.2.1. Gestión de Tareas

Desde esta ventana podemos acceder al resto de las ventanas de la gestión, consultas de la aplicación

Cuadrilla	Tarea	Fruta	Variedad	Fecha	Tipo Recogida	Pase Realizado	Cantidad Reco...
2	Entresacar	Melocoton	Baby God	2023-05-26	Cubo	1	0
2	INDICA LA TAR...		INDICA LA FRU...	2023-05-26	INDICA EL TIP...	1	0

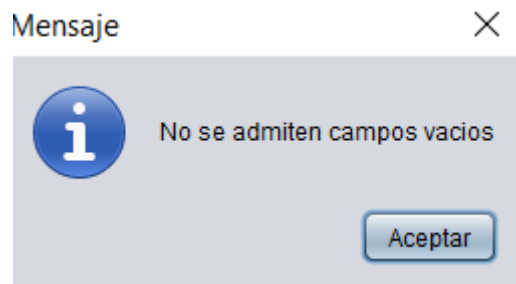
6.2.2. Gestión de Cuadrillas:

Desde esta ventana podemos crear nuevas cuadrilla, borrarlos y consultarlas

CodCuadrilla	Nombre	Cantidad de Personas
0	IOS	3
1	Venadores	0
2	Veteranos	9
6	ve	1
77	IOS	0

6.2.2.1. Insertar Cuadrilla

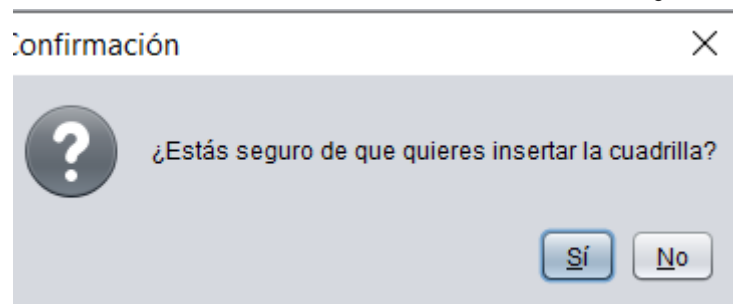
Si intentamos insertar una cuadrilla sin ningún valor válido no saldrá este mensaje:



Si intentamos ingresar esta cuadrilla y pulsando inserta nos saldrá el siguiente mensaje

Codigo-> 77		Definicion-> IOS	
INSERTAR (ALT+I)		BORRAR (ALT+B)	
BUSCAR (ALT+S)		LIMPIAR (ALT+C)	
SALIR (ALT+X)			
CodCuadrilla	Nombre	Cantidad de Personas	
0	IOS	3	
1	Venadores	0	
2	Veteranos	9	
6	ve	1	
77	IOS	0	

En esta ventana si pulsamos en si se nos guardará cuadrilla con los datos que hemos introducido con 0 trabajadores



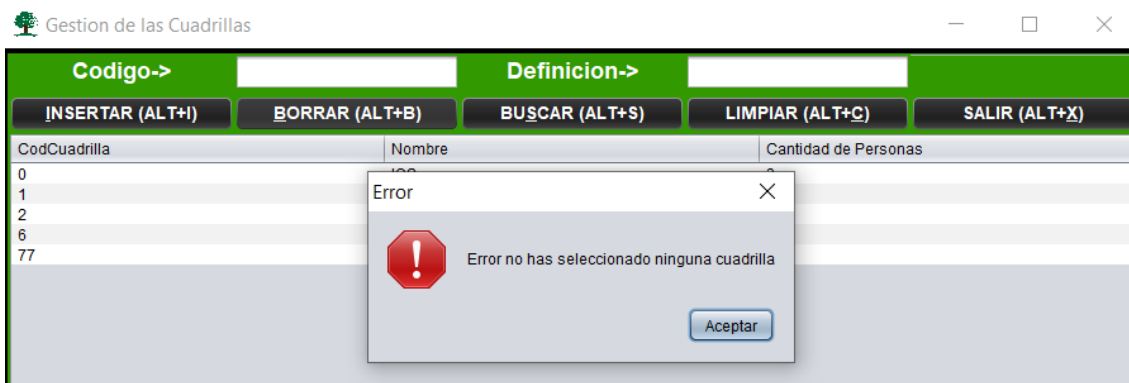
Y como comprobamos ya hemos insertado la cuadrilla y ya está disponible para asignarle trabajadores.

Codigo-> 77		Definicion-> IOS
INSERTAR (ALT+I)		BORRAR (ALT+B)
		BU ^S CAR (ALT+S)
		LIMPIAR (ALT+C)
		SALIR (ALT+X)
CodCuadrilla	Nombre	Cantidad de Personas
0	IOS	3
1	Venadores	0
2	Veteranos	9
6	ve	1
77	IOS	0

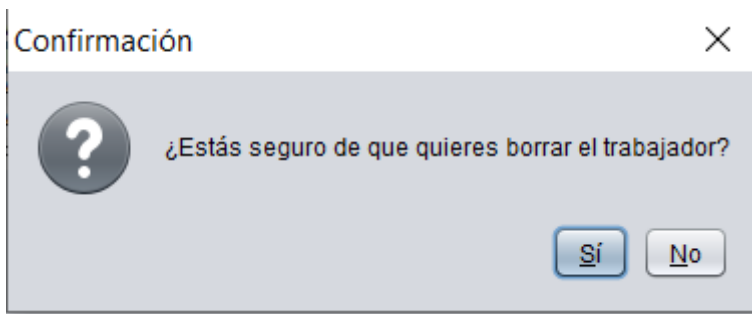
6.2.2.2. Borrar Cuadrilla

Para borrar una cuadrilla debemos seleccionarla o bien mediante la tabla seleccionandolo o buscando mediante su código antes de borrar una cuadrilla debemos asegurarse de que no tenga trabajadores :

Si no hemos seleccionado ni buscado ninguna cuadrilla y procedemos a intentar borrarla nos saldrá el siguiente mensaje:



Una vez seleccionado el registro que queremos borrar y procedemos a borrarlo.

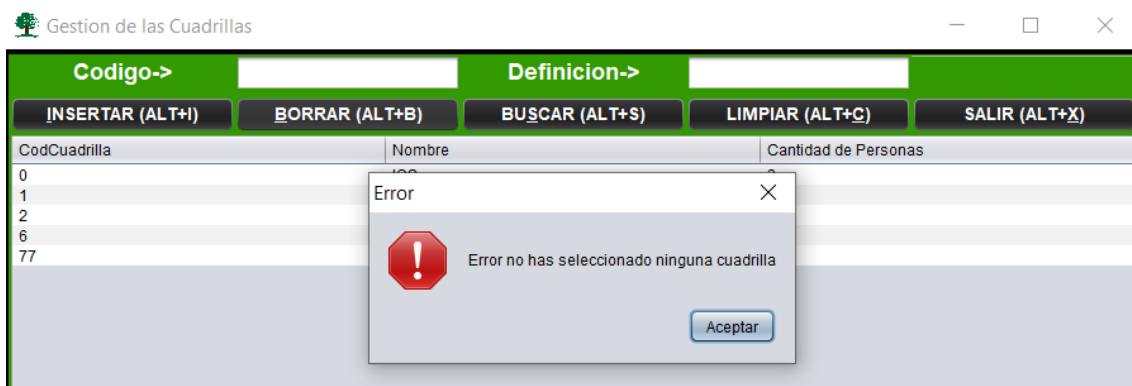


lo confirmamos y entonces ya hemos borrado al trabajador.

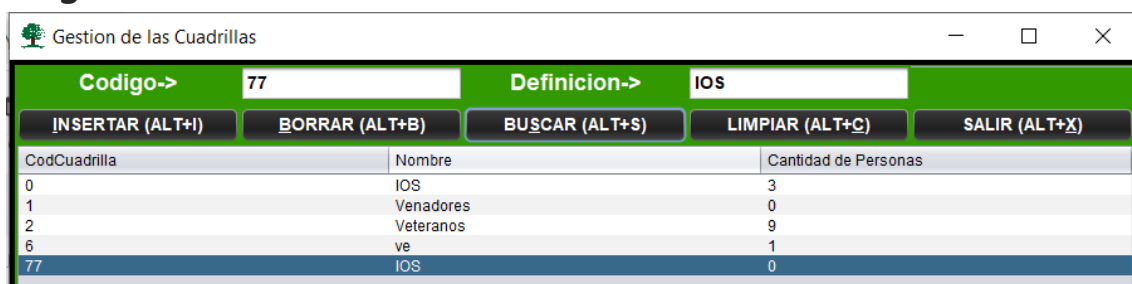
6.2.2.3. Buscar Cuadrilla

Para buscar una cuadrilla debemos buscarlo mediante su código.

Si no hemos puesto ningún código se nos mostrar el siguiente mensaje:



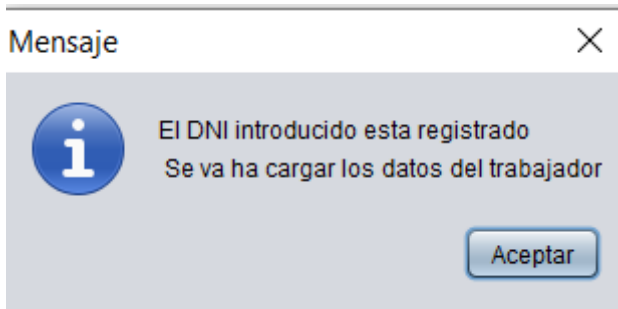
pero si hemos puesto un código correcto y lo buscamos se nos cargará los datos:



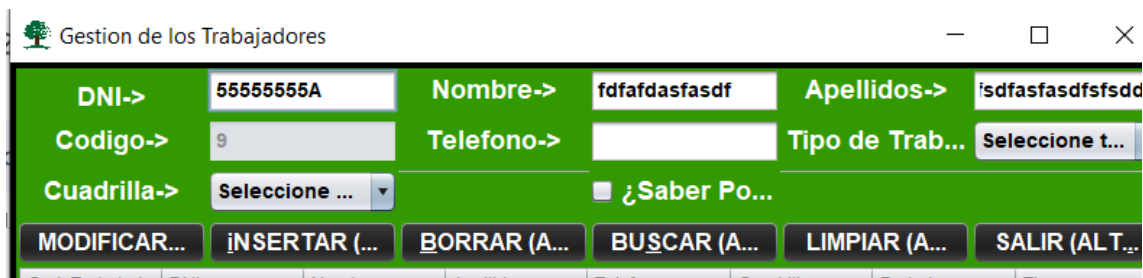
6.3. Gestión de Trabajadores:

6.3.1. Insertar Trabajadores:

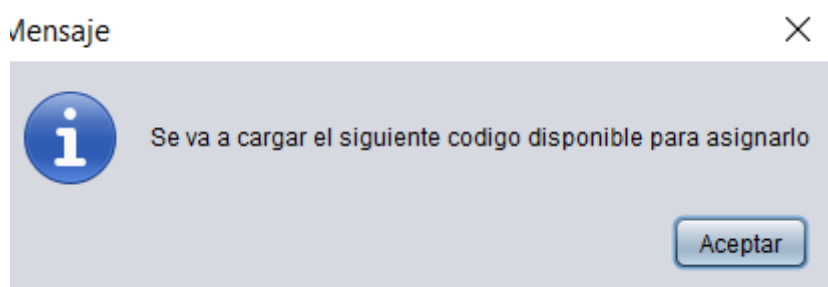
Para insertar un trabajador insertamos su DNI, si hemos tenido dado de alta en algún momento al trabajador nos mostrar el siguiente mensaje indicándonos que se van a asignar los datos del trabajador:



y se nos carga de esta manera el codigo siempre tendrá el mismo código.



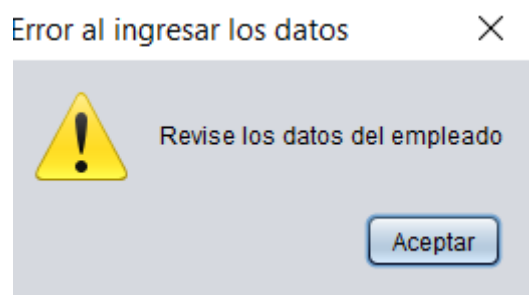
Si el trabajador nunca ha estado dado de alta en lugar del mensaje anterior se nos mostrar el siguiente mensaje:



De este modo siempre se nos asigna el código disponible:

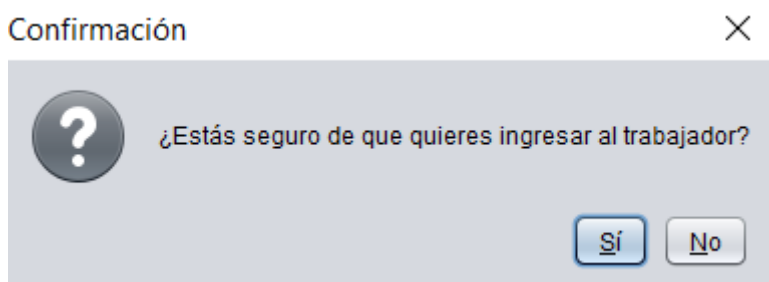
DNI->	<input type="text" value="00000000Z"/>	Nom
Codigo->	<input type="text" value="21"/>	Telef
Cuadrilla->	<input type="text" value="Seleccione ..."/>	

Al insertar un trabajador siempre se debe rellenar los datos del trabajador de forma valida si hay algún error en los datos introducidos nos saldrá el siguiente mensaje:

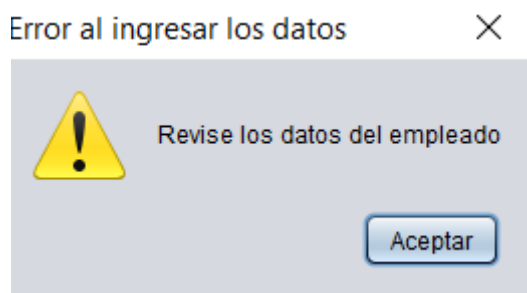


y deberemos revisar los datos que hemos introducido

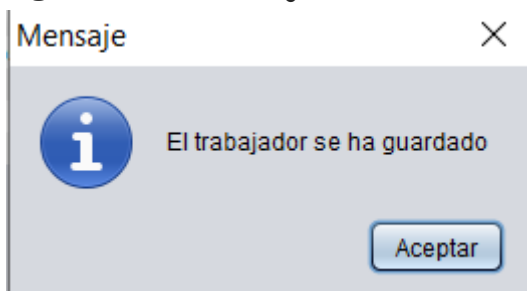
Ahora después de revisar los datos y corregir los datos erróneos pulsamos en insertar y nos mostrará el siguiente mensaje



si pulsamos sí y hay algún error en los datos nos saldrá el siguiente mensaje:



pero si no hay ningún error en los datos nos saldrá el siguiente mensaje



y si volvemos a comprobar la tabla de gestiona de trabajadores comprobamos que nos aparecer el nuevo trabajador que hemos insertado nosotros ;

DNI-> 00000000Z
Nombre-> fdfafdasfasdf
Apellidos-> fsdfasfasdfsdf

Codigo-> 21
Telefono-> 222222222
Tipo de Trab... Tractorista

Cuadrilla-> 2--Veteranos
¿Saber Po...

MODIFICAR...
INSERTAR (...
BORRAR (A...
BUSCAR (A...
LIMPIAR (A...
SALIR (ALT...

Cod_Trabajad...	DNI	Nombre	Apellidos	Telefono	Cuadrilla	Podador	Tipo
2	fasfas	Prueba	fafdsaf	0	2	No	Manijero
4	12345678B	Cristo	alcaldebAUTL...	147963258	2	Si	Perito
5	12345678A	Cristo	alcalde	123456789	2	No	Manijero
7	55555555G	Mary	Muñoz	154765824	6	No	Manijero
11	33333333f	Bartolome	Alcalde	444444444	2	No	Manijero
13	98998465F	JosuE	Gonzalez Lopez	555555555	2	No	Manijero
14	33333333a	Mario	Escribnado	85258258	0	No	Tractorista
15	33434242F	OJFOISDAFOI...	FSDOFJASOF	444444444	0	No	Manijero
18	65432123B	Benitez Benan...	Lopez	765434567	2	Si	Tractorista
19	67892091D	Mario	Escribnado	333333333	0	No	Jornalero
20	88888888F	Juan Carlos	Mendoza Guit...	777777777	2	No	Manijero
21	00000000Z	fdfafdasfasdf	fsdfasfasdfs...	222222222	2	No	Tractorista
97	12345673G	Alcalde	Mata	123456789	2	No	Manijero
122	12345678W	Alcalde	Mata	123456789	2	No	Encargado
6700	12348567f	PABLITO	Feval	159753202	2	Si	Encargado

6.3.2. Modificar Trabajadores

Para modificar a un trabajador debemos seleccionarlo o bien mediante la tabla seleccionandolo o buscando mediante su DNI:

Gestion de los Trabajadores

DNI-> 00000000Z Nombre-> fdfafdasfasdf Apellidos-> fsdfasfasdfsdf

Codigo-> 21 Telefono-> 222222222 Tipo de Trab... Tractorista

Cuadrilla-> 2--Veteranos ¿Saber Po...

MODIFICAR... INSERTAR (...) BORRAR (A...) BUSCAR (A...) LIMPIAR (A...) SALIR (ALT..)

Cod_Trabajad...	DNI	Nombre	Apellidos	Telefono	Cuadrilla	Podador	Tipo
2	fasfas	Prueba	fafdsaf	0	2	No	Manijero
4	12345678B	Cristo	alcaldebAUTI...	147963258	2	Si	Perito
5	12345678A	Cristo	alcalde	123456789	2	No	Manijero
7	55555555G	Mary	Muñoz	154765824	6	No	Manijero
11	33333333f	Bartolome	Alcalde	444444444	2	No	Manijero
13	98998465F	JosuE	Gonzalez Lopez	555555555	2	No	Manijero
14	33333333a	Mario	Escribnado	85258258	0	No	Tractorista
15	33434242F	OJFOISDAFOI...	FSDOFJASOF	444444444	0	No	Manijero
18	65432123B	Benitez Benan...	Lopez	765434567	2	Si	Tractorista
19	67892091D	Mario	Escribnado	333333333	0	No	Jornalero
20	88888888F	Juan Carlos	Mendoza Guit...	777777777	2	No	Manijero
21	00000000Z	fdfafdasfasdf	fsdfasfasdfs...	222222222	2	No	Tractorista
97	12345673G	Alcalde	Mata	123456789	2	No	Manijero
122	12345678W	Alcalde	Mata	123456789	2	No	Encargado
6700	12348567f	PABLITO	Feval	159753202	2	Si	Encargado

Si no hemos seleccionado ni buscado ningún trabajador y procedemos a modificar nos saldrá el siguiente mensaje:

Error al ingresar los datos

Revise los datos del empleado

Aceptar

Una vez seleccionado procedemos a modificar los datos que queramos:

Gestion de los Trabajadores

DNI-> 00000000Z Nombre-> Esto es una Apellidos-> Modificacion
 Codigo-> 21 Telefono-> 222222222 Tipo de Trab... Tractorista
 Cuadrilla-> 2--Veteranos ¿Saber Po...

MODIFICAR... INSERTAR (... BORRAR (A... BUSCAR (A... LIMPIAR (A... SALIR (ALT...

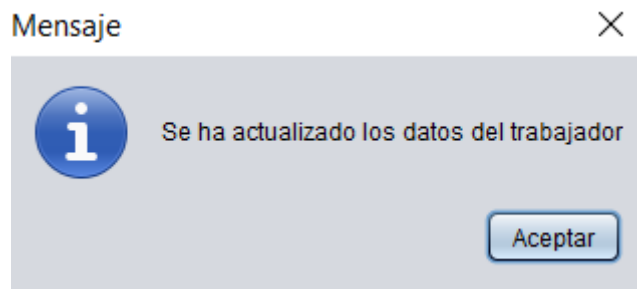
Cod_Trabajad...	DNI	Nombre	Apellidos	Telefono	Cuadrilla	Podador	Tipo
2	fasfas	Prueba	fafdsaf	0	2	No	Manijero
4	12345678B	Cristo	alcaldebAUTL...	147963258	2	Si	Perito
5	12345678A	Cristo	alcalde	123456789	2	No	Manijero
7	55555555G	Mary	Muñoz	154765824	6	No	Manijero
11	33333333f	Bartolome	Alcalde	444444444	2	No	Manijero
13	98998465F	JosuE	Gonzalez Lopez	555555555	2	No	Manijero
14	33333333a	Mario	Escribnado	85258258	0	No	Tractorista
15	33434242F	OJFOISDAFOI...	FSDOFJASOF	444444444	0	No	Manijero
18	65432123B	Benitez Benan...	Lopez	765434567	2	Si	Tractorista
19	67892091D	Mario	Escribnado	333333333	0	No	Jornalero
20	88888888F	Juan Carlos	Mendoza Guit...	777777777	2	No	Manijero
21	00000000Z	Esto es una	Modificacion	222222222	2	No	Tractorista
97	12345673G	Alcalde	Mata	123456789	2	No	Manijero
122	12345678W	Alcalde	Mata	123456789	2	No	Encargado
6700	12348567f	PABLITO	Feval	159753202	2	Si	Encargado

y pulsamos en modificar nos saldra los siguiente mensajes:

Confirmación

¿Estás seguro de que quieres modificar los datos del trabajador?

pulsamos en si y entonces



y si volvemos a ver la tabla comprobamos que se nos ha modificado el registro que hemos seleccionado:

Gestion de los Trabajadores

DNI-> 00000000Z Nombre-> Esto es una Apellidos-> Modificacion

Codigo-> 21 Telefono-> 222222222 Tipo de Trab... Tractorista

Cuadrilla-> 2--Veteranos ☐ ¿Saber Po...

MODIFICAR... INSERTAR (... BORRAR (A... BUSCAR (A... LIMPIAR (A... SALIR (ALT...

Cod_Trabajad...	DNI	Nombre	Apellidos	Telefono	Cuadrilla	Podador	Tipo
2	fasfas	Prueba	fafdsaf	0	2	No	Manijero
4	12345678B	Cristo	alcaldebAUTI...	147963258	2	Si	Perito
5	12345678A	Cristo	alcalde	123456789	2	No	Manijero
7	55555555G	Mary	Muñoz	154765824	6	No	Manijero
11	33333333f	Bartolome	Alcalde	444444444	2	No	Manijero
13	98998465F	JosuE	Gonzalez Lopez	555555555	2	No	Manijero
14	33333333a	Mario	Escribnado	85258258	0	No	Tractorista
15	33434242F	OJFOISDAFOI...	FSDOFJASOF	444444444	0	No	Manijero
18	65432123B	Benitez Benan...	Lopez	765434567	2	Si	Tractorista
19	67892091D	Mario	Escribnado	333333333	0	No	Jornalero
20	88888888F	Juan Carlos	Mendoza Guit...	777777777	2	No	Manijero
21	00000000Z	Esto es una	Modificacion	222222222	2	No	Tractorista
97	12345673G	Alcalde	Mata	123456789	2	No	Manijero
122	12345678W	Alcalde	Mata	123456789	2	No	Encargado
6700	12348567f	PABLITO	Feval	159753202	2	Si	Encargado

6.3.3. Borrar Trabajadores

Para borrar a un trabajador debemos seleccionarlo o bien mediante la tabla seleccionandolo o buscando mediante su DNI:

Gestion de los Trabajadores

DNI-> 00000000Z Nombre-> fdfafdasfasdf Apellidos-> 'sdfasfasdfsfsddf

Codigo-> 21 Telefono-> 222222222 Tipo de Trab... Tractorista

Cuadrilla-> 2--Veteranos ¿Saber Po...

MODIFICAR... INSERTAR (...) BORRAR (A...) BUSCAR (A...) LIMPIAR (A...) SALIR (ALT..)

Cod_Trabajad...	DNI	Nombre	Apellidos	Telefono	Cuadrilla	Podador	Tipo
2	fasfas	Prueba	fafdsaf	0	2	No	Manijero
4	12345678B	Cristo	alcaldebAUTI...	147963258	2	Si	Perito
5	12345678A	Cristo	alcalde	123456789	2	No	Manijero
7	55555555G	Mary	Muñoz	154765824	6	No	Manijero
11	33333333f	Bartolome	Alcalde	444444444	2	No	Manijero
13	98998465F	JosuE	Gonzalez Lopez	555555555	2	No	Manijero
14	33333333a	Mario	Escribnado	85258258	0	No	Tractorista
15	33434242F	OJFOISDAFOI...	FSDOFJASOF	444444444	0	No	Manijero
18	65432123B	Benitez Benan...	Lopez	765434567	2	Si	Tractorista
19	67892091D	Mario	Escribnado	333333333	0	No	Jornalero
20	88888888F	Juan Carlos	Mendoza Guit...	777777777	2	No	Manijero
21	00000000Z	fdfafdasfasdf	fsdfasfasdfs...	222222222	2	No	Tractorista
97	12345673G	Alcalde	Mata	123456789	2	No	Manijero
122	12345678W	Alcalde	Mata	123456789	2	No	Encargado
6700	12348567f	PABLITO	Feval	159753202	2	Si	Encargado

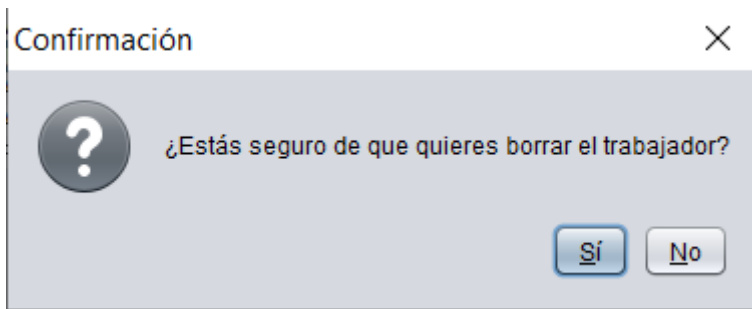
Si no hemos seleccionado ni buscado ningún trabajador y procedemos a modificar nos saldrá el siguiente mensaje:

Error al ingresar los datos

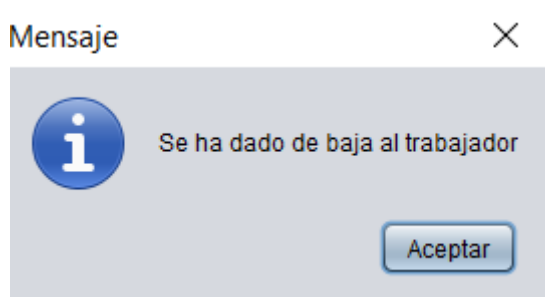
Revise los datos del empleado

Aceptar

Seleccionamos el registro que queremos borrar y pulsamos en borrar.



lo confirmamos

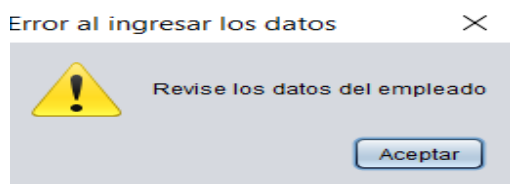


y entonces ya hemos borrado al trabajador.

6.3.4. Buscar Trabajadores

Para buscar a un trabajador debemos buscarlo mediante su DNI.

Si no hemos puesto ningún DNI o el DNI buscado esta mal escrito nos saldrá el siguiente mensaje:



pero si hemos puesto un DNI correcto y lo buscamos se nos cargará los datos:

Gestion de los Trabajadores

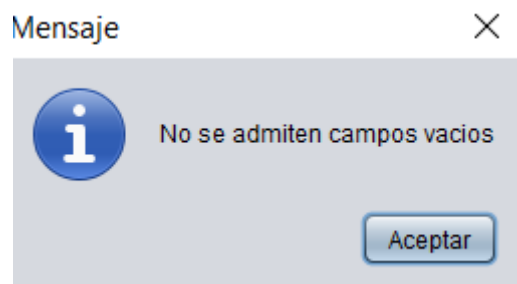
DNI-> 00000000Z Nombre-> Esto es una Apellidos-> Modificacion
 Codigo-> 21 Telefono-> 222222222 Tipo de Trab... Tractorista
 Cuadrilla-> 2--Veteranos ¿Saber Po...
 MODIFICAR... INSERTAR (... BORRAR (A... BUSCAR (A... LIMPIAR (A... SALIR (ALT...

Cod_Trabajad...	DNI	Nombre	Apellidos	Telefono	Cuadrilla	Podador	Tipo
2	fasfas	Prueba	fafdsaf	0	2	No	Manijero
4	12345678B	Cristo	alcaldebAUTI...	147963258	2	Si	Perito
5	12345678A	Cristo	alcalde	123456789	2	No	Manijero
7	55555555G	Mary	Muñoz	154765824	6	No	Manijero
11	33333333f	Bartolome	Alcalde	444444444	2	No	Manijero
13	98998465F	JosuE	Gonzalez Lopez	555555555	2	No	Manijero
14	33333333a	Mario	Escribnado	85258258	0	No	Tractorista
15	33434242F	OJFOISDAFOI...	FSDOFJASOF	444444444	0	No	Manijero
18	65432123B	Benitez Benan...	Lopez	765434567	2	Si	Tractorista
19	67892091D	Mario	Escribnado	333333333	0	No	Jornalero
20	88888888F	Juan Carlos	Mendoza Guit...	777777777	2	No	Manijero
21	00000000Z	Esto es una	Modificacion	222222222	2	No	Tractorista
97	12345673G	Alcalde	Mata	123456789	2	No	Manijero
122	12345678W	Alcalde	Mata	123456789	2	No	Encargado
6700	12348567f	PABLITO	Feval	159753202	2	Si	Encargado

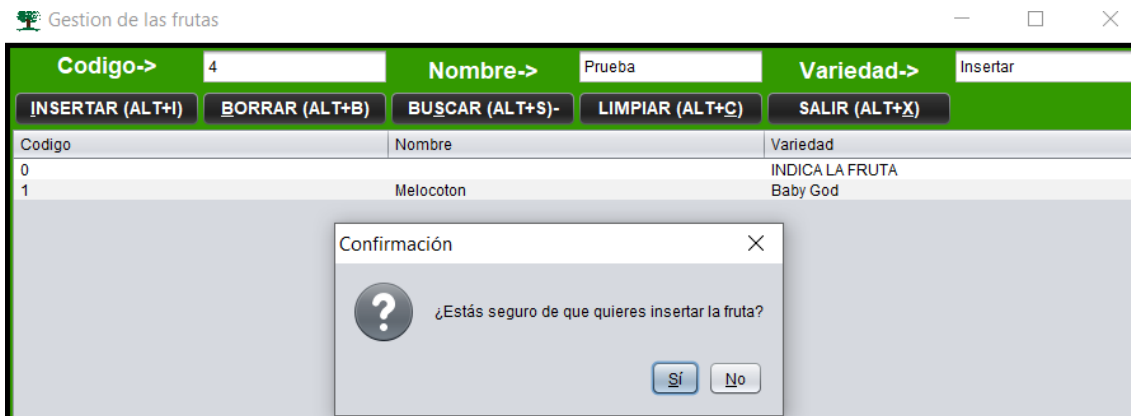
6.4. Gestión de Frutas

6.4.1. Insertar Fruta

Si intentamos insertar una fruta sin ningún valor válido no saldrá este mensaje:



Si intentamos ingresar esta cuadrilla y pulsando inserta nos saldrá el siguiente mensaje



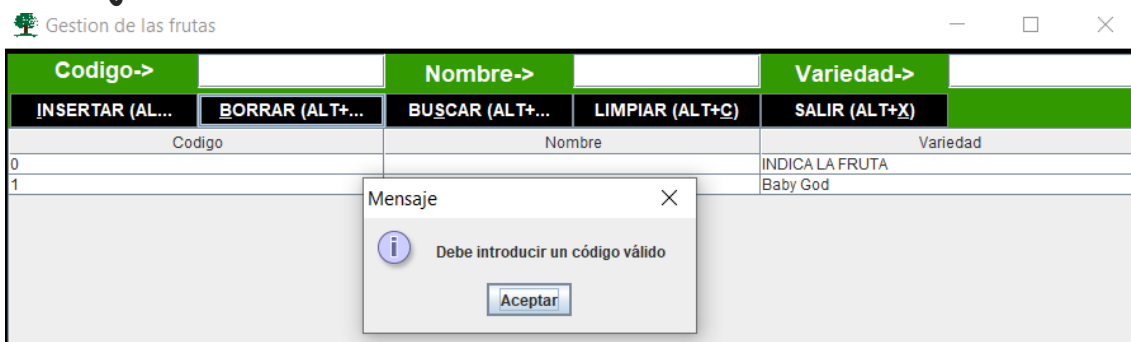
Si pulsamos en si y comprobamos la ventana comprobamos que se nos ha insertado la fruta



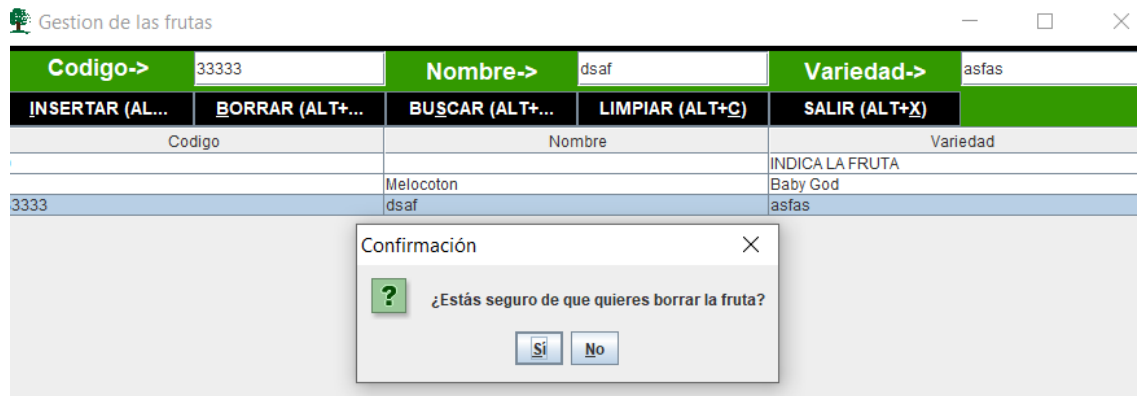
6.4.2. Borrar Fruta

Para borrar una fruta debemos seleccionarla o bien mediante la tabla seleccionandolo o buscando mediante su código.W:

Si no hemos seleccionado ni buscado ninguna cuadrilla y procedemos a intentar borrarla nos saldrá el siguiente mensaje:



Una vez seleccionado el registro que queremos borrar y procedemos a borrarlo.

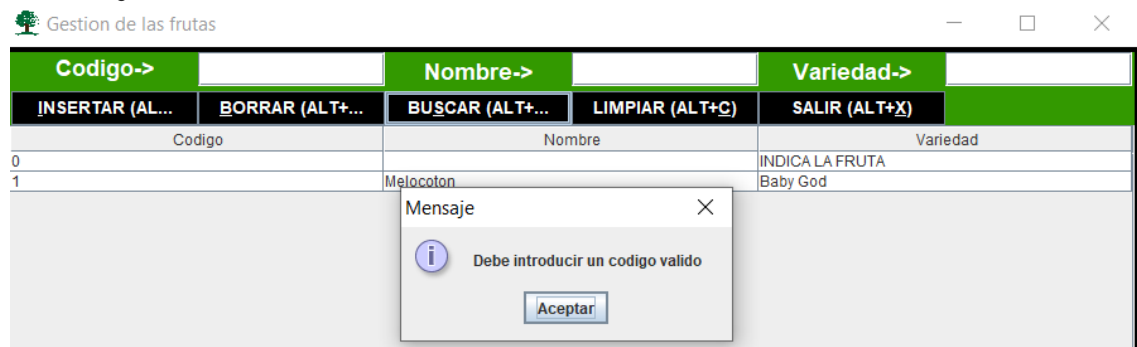


lo confirmamos y entonces ya hemos borrado la fruta.

6.4.3. Buscar Fruta

Para buscar una cuadrilla debemos buscarlo mediante su código.

Si no hemos puesto ningún código se nos mostrar el siguiente mensaje:

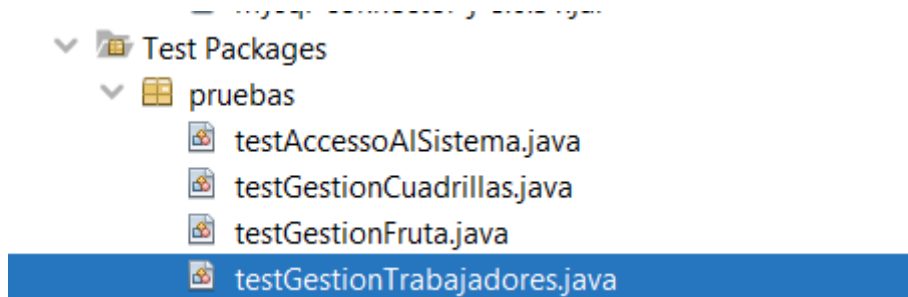


pero si hemos puesto un código correcto y lo buscamos se nos cargará los datos:



7. Pruebas:

Estas pruebas se han ido realizando durante la creación de la aplicación para supervisar que los metodos cumplan con su funcion:



- La prueba de acceso al sistema: Comprobaremos si el usuario que estaba intentando acceder al sistema era válido o no.
- Gestión Cuadrilla: Hemos probado los métodos de insertar, borrar y buscar de forma que funcionara de la manera mas optima.
- Gestión Fruta: Hemos probado los métodos de insertar, borrar y buscar de forma que funcionara de la manera mas optima.
- Gestión Trabajadores: Hemos probado los métodos de insertar, modificar, borrar y buscar de forma que funcionara de la manera mas optima.

8. Aplicación de Android:

8.1. Diagrama de clases

8.2. Ventanas

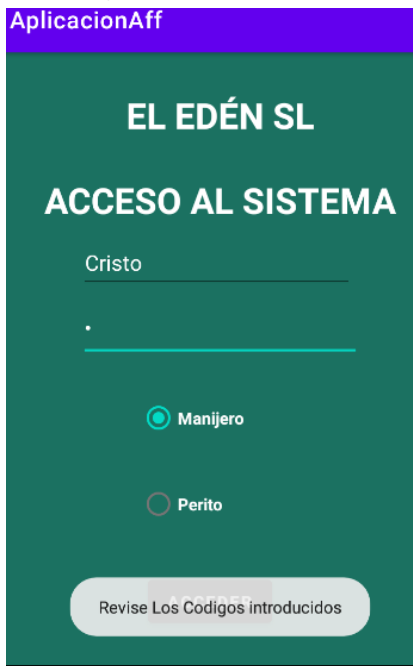
8.2.1. Acceso al Sistema

Desde esta ventana solo los trabajadores autorizados pueden acceder a la aplicación en este ejemplo solo pueden los manijeros y los peritos.



Para acceder debemos poner el nombre, el código del trabajador y seleccionar que tipo de trabajador es si es manijero o perito,

Si introducimos algún valor o no equivocamos de tipo de trabajador pues entonces nos mostrar este mensaje



AplicacionAff

EL EDÉN SL

ACCESO AL SISTEMA

Cristo

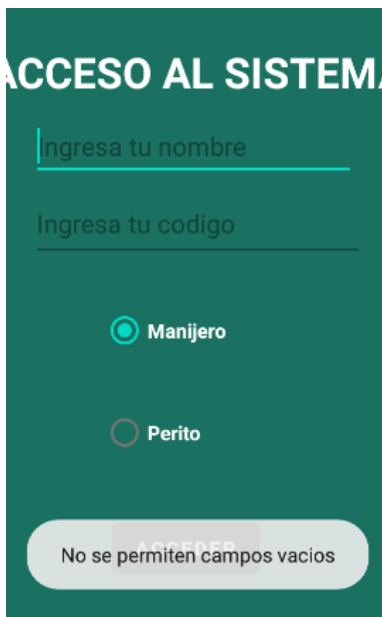
.

☒ Manijero

☐ Perito

Revise Los Codigos introducidos

O si no introducimos ningún valor nos muestra este mensaje



ACCESO AL SISTEMA

Ingresa tu nombre

Ingresa tu codigo

☒ Manijero

☐ Perito

No se permiten campos vacios

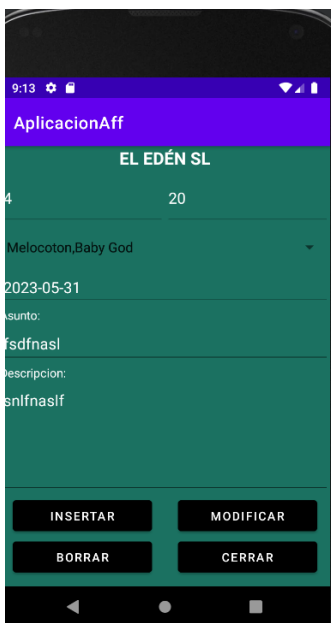
En esta ventana la opcion de codigo solo acepta números, y el nombre solo acepta letras

8.2.2. Usuario Perito:

En esta ventana los perito pueden consultar los informes que han subido, cada perito solo puede consultar sus propios informes.



si pulsamos en algún informe que hemos creado se nos mostrará de forma más detallada en el crud de los informes:



En esta ventana podemos modificar el informe seleccionado, borrarlo

Y si pulsando en gestionar informes se nos abrirá la siguiente ventana en blanco para poder insertar un nuevo informe

The screenshot shows a mobile application interface with a purple header bar labeled 'AplicacionAff'. Below the header, there is a green section titled 'EL EDÉN SL' with a subtext '(No hace falta rellenar)'. This is followed by a section labeled 'INDICA LA FRUTA' with a dropdown arrow and the text '(No hace falta rellenar)'. Below this, there are two input fields: 'Asunto:' and 'Descripcion:'. The 'Descripcion:' field has a placeholder text 'Cuerpo del informe'. At the bottom, there are four buttons: 'INSERTAR', 'MODIFICAR', 'BORRAR', and 'CERRAR'.

8.2.2.1. Insertar informe

En la ventana anterior rellenamos los datos con la fruta a la que nos referimos, el asunto y la descripción y pulsamos en insertar nos mostrará el siguiente mensaje:

The screenshot shows a confirmation dialog box with a white background and a black border. It contains the text 'Confirmación' followed by the question '¿Estás seguro de insertar el informe?'. At the bottom, there are two buttons: 'CANCELAR' and 'ACEPTAR'.

Si pulsamos en aceptar y volvemos a nuestros registros comprobamos que se nos ha insertado el informe y se ve reflejado:

,fsdfnasl,31/05/2023
,fsdfnasl,31/05/2023
,pruebaManual,31/05/2023
,Prueba,30/05/2023

8.2.2.2. Modificar Informe

Si seleccionamos un informe y modificamos los datos y pulsamos en modificar

9:13

AplicacionAff

EL EDÉN SL

4 20

Melocoton,Baby God

2023-05-31

asunto:

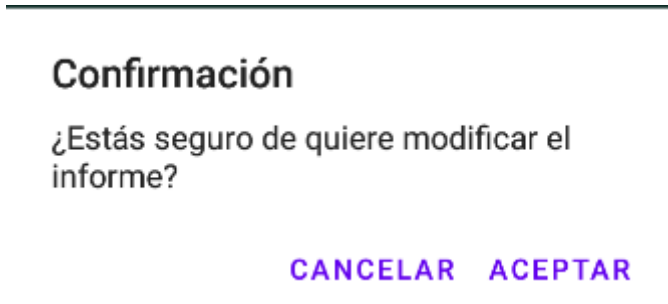
fsdfnasl

descripcion: snlfnaslf

INSERTAR MODIFICAR

BORRAR CERRAR

Nos mostrar el siguiente mensaje si pulsamos en si pues el informe seleccionado ya se ha modificado.

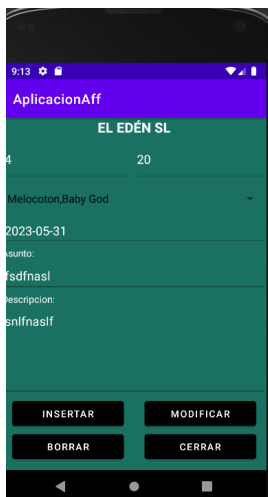


y se ve reflejado en el registro

,fsdfnasl,31/05/2023
,fsdfnasl,31/05/2023
,pruebaManual,31/05/2023
,Prueba,30/05/2023

8.2.2.3. Borrar Informe

Si seleccionamos un informe y pulsamos en borrar



Nos mostrará el siguiente mensaje, si pulsamos en si pues el informe seleccionado ya se ha borrado.

Confirmación

¿Estás seguro de quiere Borrar el informe?

CANCELAR ACEPTAR

y se ve reflejado en el registro

,fsdfnasl,31/05/2023
,fsdfnasl,31/05/2023
,pruebaManual,31/05/2023
,Prueba,30/05/2023

8.3. Usuario Manijero




En esta ventana los manijeros pueden consultar las tareas que ha subido, cada manijero solo puede consultar las respectivas tareas de su cuadrilla.

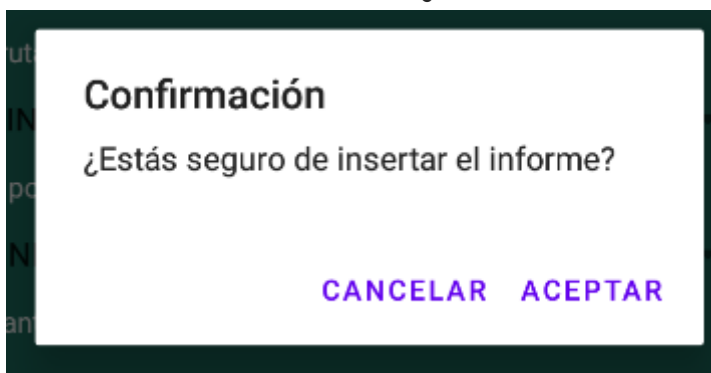
Si pulsamos en algún tarea que hemos creado se nos mostrará de forma más detallada en el crud de las tareas de esta forma en la tarea solo se puede actualizar o borrar la tarea:

8.3.1. Insertar Tarea

En esta ventana para insertar una tarea solo tenemos que seleccionar, la tarea, la fruta, el tipo de recogida si se va a recoger y el número de pase que se va a realizar y pulsamos en insertar.



Nos mostrará el mensaje



y si pulsamos en si ya se nos ha guardado la tareas

8.3.2. Modificar Tarea

Si pulsamos un informe el único valor que se puede modificar de la tarea es la cantidad recogida

10:03

AplicacionAff

EL EDÉN SL
Gestion de Tareas

Cuadrilla: 2 Fecha (No hace falta)

Tarea
INDICA LA TAREA

Fruta:
INDICA LA FRUTA

Tipo Recogida:
INDICA EL TIPO DE RECOGIDA

Cantidad de Paleps:
0

Nº_Pase:
1

INSERTAR MODIFICAR
BORRAR CERRAR

en vez de 0 pondremos 15 y pulsamos en modificar nos saldra el siguiente mensaje

Confirmación

¿Estás seguro de quiere modificar el informe?

CANCELAR ACEPTAR

pulsamos en si y ya se nos habrá actualizado el informe

8.3.3. Borrar Tarea

Si pulsamos un informe



y pulsamos en borrar

nos saldrá el siguiente mensaje

Confirmación

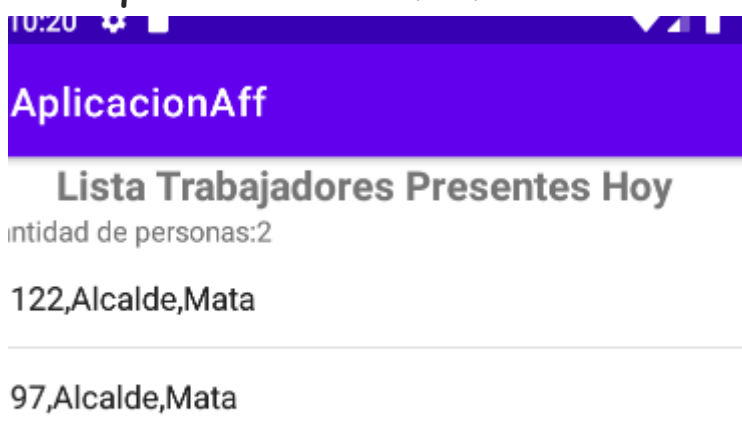
¿Estás seguro de quiere Borrar el informe?

CANCELAR ACEPTAR

pulsamos en si y ya se nos habrá borrado el informe

8.3.4. Lista de Personas Presentes

Desde esta ventana cada vez que un trabajador de una cuadrilla fice a la entrada se le va a ver reflejado al manijero de esta cuadrilla en esta ventana indicando que está presente y además le muestra la cantidad de personas que tiene que están en el mismo día.



8.3.5. Lista de Personas de la cuadrilla

Desde se carga una lista de todos los trabajadores de cada cuadrilla.



The screenshot shows a mobile application interface. At the top, there is a status bar with the time 10:26, a settings icon, a document icon, and signal/battery indicators. Below the status bar is a purple header bar with the text "AplicacionAff". Underneath the header is a title "Lista Trabajadores" in bold. The main content area displays a list of seven entries, each on a separate line with horizontal dividers. The entries are: "2,Prueba,fafdsaf", "5,Cristo,alcalde", "7,Mary,Muñoz", "11,Bartolome,Alcalde", "13,JosuE,Gonzalez Lopez", "15,OJFOISDAFOIAS,FSDOFJASOF", and "97,Alcalde,Mata".

Lista Trabajadores	
2,	Prueba,fafdsaf
5,	Cristo,alcalde
7,	Mary,Muñoz
11,	Bartolome,Alcalde
13,	JosuE,Gonzalez Lopez
15,	OJFOISDAFOIAS,FSDOFJASOF
97,	Alcalde,Mata

Y si pulsamos en algún registro nos muestra de forma detallada la información sobre los días que ha trabajado, las horas y el sueldo ganado

Horas Trabajadas por el trabajador
Trabajador: Alcalde Mata Código: 122
Horas totales trabajadas: 17 Horas. Sueldo total: 119
Euros

Dia-> 29, Mes-> 5, Horas_Trabajaas-> 4

Dia-> 30, Mes-> 5, Horas_Trabajaas-> 10

Dia-> 31, Mes-> 5, Horas_Trabajaas-> 3

VOLVER

9. Código de Servidor

Código para Crear nuestra BD para el correcto funcionamiento de nuestra BD

```
CREATE DATABASE aff;
```

```
USE aff;
```

```
--
```

```
-- Estructura de tabla para la tabla `cuadrilla`
```

```
--
```

```
CREATE TABLE `cuadrilla` (
```

```
  `Cod_Cuadrilla` int(11) NOT NULL,
```

```
  `nombre` varchar(20) NOT NULL
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
```

```
COLLATE=utf8mb4_general_ci;
```

```
--
```

```
-- Volcado de datos para la tabla `cuadrilla`
```

```
--
```

```
INSERT INTO `cuadrilla` (`Cod_Cuadrilla`, `nombre`)
VALUES
(0, 'IOS');
```

```
--
```

```
-----
-----
```

```
--
```

```
-- Estructura de tabla para la tabla `fruta`
```

```
--
```

```
CREATE TABLE `fruta` (
  `Cod_Fruta` int(11) NOT NULL,
  `Variedad` varchar(30) DEFAULT NULL,
  `nombre` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;
```

```
--
```

```
-- Volcado de datos para la tabla `fruta`
```

```
--
```

```
INSERT INTO `fruta` (`Cod_Fruta`, `Variedad`, `nombre`)
VALUES
```

```
(0, 'INDICA LA FRUTA', '');
```

```
--
```

```
-----
-----
```

```
--
```

```
-- Estructura de tabla para la tabla `informes`
```

```
--
```

```
CREATE TABLE `informes` (
```

```
  `Id` int(11) NOT NULL,
```

```
  `Cod_Perito` int(11) NOT NULL,
```

```
  `Id_Fruta` int(11) NOT NULL,
```

```
  `Asunto` varchar(50) DEFAULT NULL,
```

```
  `Descripcion` varchar(500) NOT NULL,
```

```
  `Fecha` date NOT NULL DEFAULT current_timestamp()
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
```

```
COLLATE=utf8mb4_general_ci;
```

```
--
```

```
-- Disparadores `informes`
```



```
--
DELIMITER $$
CREATE TRIGGER `Registrar_Informes` AFTER INSERT ON
`informes` FOR EACH ROW BEGIN
    DECLARE v_nombre VARCHAR(255);
    DECLARE v_variedad VARCHAR(255);

    SELECT nombre, Variedad INTO v_nombre, v_variedad
    FROM fruta
    WHERE Cod_Fruta = new.Id_Fruta;

    INSERT INTO registro_informe_perito (Cod_Perito,
    Id_Fruta, Asunto, Descripcion, nombrefruta, variedad)
    VALUES (new.Cod_Perito, new.Id_Fruta, new.Asunto,
    new.Descripcion, v_nombre, v_variedad);
END
$$
DELIMITER ;
```

```
--
```

```
-----
```

```
-----
```

```
--
```

```
-- Estructura de tabla para la tabla `registro`
--

CREATE TABLE `registro` (
  `ID` int(11) NOT NULL,
  `ID_Cuadrilla` int(11) NOT NULL,
  `ID_Tarea` int(11) NOT NULL,
  `ID_Fruta` int(11) NOT NULL,
  `TipoRecogida` varchar(20) DEFAULT NULL,
  `Fecha` date DEFAULT current_timestamp(),
  `CantidadPaleps` int(11) NOT NULL DEFAULT 0,
  `Nº_Pase` int(11) DEFAULT 1
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;

--

-- Disparadores `registro`
--

DELIMITER $$

CREATE TRIGGER `Actualizar` BEFORE UPDATE ON
`registro` FOR EACH ROW BEGIN

INSERT INTO `registro_tarea_cuadrilla` (`ID_Cuadrilla`,
`ID_Tarea`, `ID_Fruta`, `TipoRecogida`, `CantidadPaleps`,
`Nº_Pase`) VALUES (old.ID_Cuadrilla , new.ID_Tarea ,
```

```

new.ID_Fruta ,new.TipoRecogida,new.CantidadPaleps,
new.Nº_Pase);

END

$$

DELIMITER ;

DELIMITER $$

CREATE TRIGGER `Registrar_Tarea_Cuadrilla` AFTER
INSERT ON `registro` FOR EACH ROW BEGIN

INSERT INTO `registro_tarea_cuadrilla` (`ID_Cuadrilla` ,
`ID_Tarea` , `ID_Fruta` , `TipoRecogida` , `CantidadPaleps` ,
`Nº_Pase`) VALUES (new.ID_Cuadrilla ,new.ID_Tarea ,
new.ID_Fruta ,new.TipoRecogida,new.CantidadPaleps,
new.Nº_Pase);

END

$$

DELIMITER ;

--
-----
-----

--

-- Estructura de tabla para la tabla `registrofichar`
--

```

```
CREATE TABLE `registrofichar` (
  `ID` int(11) NOT NULL,
  `Cod_Trabajador` int(11) NOT NULL,
  `HEntada` timestamp NOT NULL DEFAULT
current_timestamp(),
  `HSalida` timestamp NOT NULL DEFAULT
current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;
```

```
--
```

```
-----
```

```
-----
```

```
--
```

```
-- Estructura de tabla para la tabla `registrotrabajadores`
```

```
--
```

```
CREATE TABLE `registrotrabajadores` (
  `ID` int(11) NOT NULL,
  `Cod_Trabajador` int(11) NOT NULL,
  `DNI` varchar(9) NOT NULL,
  `Nombre` varchar(20) NOT NULL,
  `Apellidos` varchar(50) NOT NULL,
```

```

`Falta` date NOT NULL DEFAULT current_timestamp(),
`FBaja` date NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;

--
-----
-----

--
-- Estructura de tabla para la tabla
`registro_informe_perito`
--

```

```

CREATE TABLE `registro_informe_perito` (
  `Id` int(11) NOT NULL,
  `Cod_Perito` int(11) NOT NULL,
  `Id_Fruta` int(11) NOT NULL,
  `Asunto` varchar(50) DEFAULT NULL,
  `Descripcion` varchar(500) NOT NULL,
  `Fecha` date NOT NULL DEFAULT current_timestamp(),
  `nombrefruta` varchar(50) DEFAULT NULL,
  `variedad` varchar(50) DEFAULT NULL

```

```

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;

--

-- Disparadores `registro_informe_perito`

--

DELIMITER $$

CREATE TRIGGER `BorrarRegistros` BEFORE DELETE ON
`registro_informe_perito` FOR EACH ROW BEGIN

    DECLARE fecha_limite DATE;

    -- Calcular la fecha límite (hoy menos un año)

    SET fecha_limite = DATE_SUB(CURDATE(), INTERVAL 1
YEAR);

    -- Verificar si la fecha del registro es anterior a la
fecha límite

    IF OLD.fecha < fecha_limite THEN

        -- Eliminar el registro

        DELETE FROM registro_informe_perito WHERE id =
OLD.id;

    END IF;

END

$$

```

DELIMITER ;

--

--

-- Estructura de tabla para la tabla
`registro_tarea_cuadrilla`

--

```
CREATE TABLE `registro_tarea_cuadrilla` (  
  `ID` int(11) NOT NULL,  
  `ID_Cuadrilla` int(11) NOT NULL,  
  `ID_Tarea` int(11) NOT NULL,  
  `ID_Fruta` int(11) NOT NULL,  
  `TipoRecogida` varchar(20) DEFAULT NULL,  
  `Fecha` date DEFAULT current_timestamp(),  
  `CantidadPaleps` int(11) NOT NULL DEFAULT 0,  
  `Nº_Pase` int(11) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_general_ci;
```

--

```

-- Disparadores `registro_tarea_cuadrilla`
--

DELIMITER $$

CREATE TRIGGER `BorrarTareas` BEFORE DELETE ON
`registro_tarea_cuadrilla` FOR EACH ROW BEGIN

    DECLARE fecha_limite DATE;

    -- Calcular la fecha límite (hoy menos un año)

    SET fecha_limite = DATE_SUB(CURDATE(), INTERVAL 1
YEAR);

    -- Verificar si la fecha del registro es anterior a la
fecha límite

    IF OLD.fecha < fecha_limite THEN

        -- Eliminar el registro

        DELETE FROM registro_tarea_cuadrilla WHERE id =
OLD.id;

    END IF;

END

$$

DELIMITER ;

```



```
--
-----

--

-- Estructura de tabla para la tabla `tareas`

--

CREATE TABLE `tareas` (
  `Cod_Tarea` int(11) NOT NULL,
  `nombre` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;

--

-- Volcado de datos para la tabla `tareas`

--

INSERT INTO `tareas` (`Cod_Tarea`, `nombre`) VALUES
(0, 'INDICA LA TAREA'),
(1, 'Entresacar'),
(2, 'Recogida'),
(3, 'Poda');
```

```
--  
-----  
-----
```

```
--  
-- Estructura de tabla para la tabla `tipo_trabajador`  
--
```

```
CREATE TABLE `tipo_trabajador` (  
  `id_Tipo` int(11) NOT NULL,  
  `nombre` varchar(20) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_general_ci;
```

```
--  
-- Volcado de datos para la tabla `tipo_trabajador`  
--
```

```
INSERT INTO `tipo_trabajador` (`id_Tipo`, `nombre`)  
VALUES  
(1, 'Encargado'),  
(2, 'Manijero'),  
(3, 'Tractorista'),
```

(4, 'Jornalero'),

(5, 'Perito');

--

--

-- Estructura de tabla para la tabla `trabajadores`

--

```
CREATE TABLE `trabajadores` (  
  `Cod_Trabajador` int(11) NOT NULL,  
  `nombre` varchar(20) NOT NULL,  
  `apellido` varchar(50) NOT NULL,  
  `DNI` varchar(9) DEFAULT NULL,  
  `telefono` int(9) NOT NULL,  
  `Cuadrilla` int(11) DEFAULT NULL,  
  `podador` tinyint(1) DEFAULT NULL,  
  `Tipo` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_general_ci;
```

--

-- Volcado de datos para la tabla `trabajadores`

--

```
INSERT INTO `trabajadores` (`Cod_Trabajador`,
`nombre`, `apellido`, `DNI`, `telefono`, `Cuadrilla`,
`podador`, `Tipo`) VALUES
(0, 'ADMIN', 'ADMIN', '1234568A', 123456789, 0, NULL,
1);
```

--

-- Disparadores `trabajadores`

--

DELIMITER \$\$

```
CREATE TRIGGER `Borrar_Trabajador_en_Registro` BEFORE
DELETE ON `trabajadores` FOR EACH ROW BEGIN
```

```
UPDATE `registrotrabajadores` SET `FBaja` = now() WHERE
Cod_Trabajador=old.Cod_Trabajador;
```

```
END
```

\$\$

DELIMITER ;

DELIMITER \$\$

```
CREATE TRIGGER `Insertar_Trabajador_en_Registro`
AFTER INSERT ON `trabajadores` FOR EACH ROW BEGIN
```

```
INSERT INTO registrotrabajadores (`Cod_Trabajador`,
`Nombre`, `Apellidos`, `DNI`) VALUES
(new.Cod_Trabajador, new.nombre, new.apellido,new.dni);

END

$$

DELIMITER ;
```

```
--
```

```
-- Índices para tablas volcadas
```

```
--
```

```
--
```

```
-- Indices de la tabla `cuadrilla`
```

```
--
```

```
ALTER TABLE `cuadrilla`
```

```
ADD PRIMARY KEY (`Cod_Cuadrilla`);
```

```
--
```

```
-- Indices de la tabla `fruta`
```

```
--
```

```
ALTER TABLE `fruta`
```

```
ADD PRIMARY KEY (`Cod_Fruta`);
```

```
--  
-- Indices de la tabla `informes`  
--  
ALTER TABLE `informes`  
  ADD PRIMARY KEY (`Id`),  
  ADD KEY `Cod_Perito` (`Cod_Perito`),  
  ADD KEY `Id_Fruta` (`Id_Fruta`);  
  
--  
-- Indices de la tabla `registro`  
--  
ALTER TABLE `registro`  
  ADD PRIMARY KEY (`ID`),  
  ADD KEY `ID_Cuadrilla` (`ID_Cuadrilla`),  
  ADD KEY `ID_Tarea` (`ID_Tarea`),  
  ADD KEY `ID_Fruta` (`ID_Fruta`);  
  
--  
-- Indices de la tabla `registrofichar`  
--  
ALTER TABLE `registrofichar`  
  ADD PRIMARY KEY (`ID`);
```

```
--  
-- Indices de la tabla `registrotrabajadores`  
--  
ALTER TABLE `registrotrabajadores`  
  ADD PRIMARY KEY (`ID`),  
  ADD KEY `Cod_Trabajador` (`Cod_Trabajador`);
```

```
--  
-- Indices de la tabla `registro_informe_perito`  
--  
ALTER TABLE `registro_informe_perito`  
  ADD PRIMARY KEY (`Id`),  
  ADD KEY `Cod_Perito` (`Cod_Perito`),  
  ADD KEY `Id_Fruta` (`Id_Fruta`);
```

```
--  
-- Indices de la tabla `registro_tarea_cuadrilla`  
--  
ALTER TABLE `registro_tarea_cuadrilla`  
  ADD PRIMARY KEY (`ID`),  
  ADD KEY `ID_Cuadrilla` (`ID_Cuadrilla`),  
  ADD KEY `ID_Tarea` (`ID_Tarea`),  
  ADD KEY `ID_Fruta` (`ID_Fruta`);
```

```
--  
-- Indices de la tabla `tareas`  
--  
ALTER TABLE `tareas`  
  ADD PRIMARY KEY (`Cod_Tarea`);  
  
--  
-- Indices de la tabla `tipo_trabajador`  
--  
ALTER TABLE `tipo_trabajador`  
  ADD PRIMARY KEY (`id_Tipo`);  
  
--  
-- Indices de la tabla `trabajadores`  
--  
ALTER TABLE `trabajadores`  
  ADD PRIMARY KEY (`Cod_Trabajador`),  
  ADD UNIQUE KEY `DNI` (`DNI`),  
  ADD KEY `Cuadrilla` (`Cuadrilla`),  
  ADD KEY `Tipo` (`Tipo`);  
  
--
```



```
-- AUTO_INCREMENT de las tablas volcadas
--

--

-- AUTO_INCREMENT de la tabla `informes`
--

ALTER TABLE `informes`

  MODIFY `Id` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=23;

--

-- AUTO_INCREMENT de la tabla `registro`
--

ALTER TABLE `registro`

  MODIFY `ID` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=20;

--

-- AUTO_INCREMENT de la tabla `registrofichar`
--

ALTER TABLE `registrofichar`

  MODIFY `ID` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=22;
```

```
--  
-- AUTO_INCREMENT de la tabla `registrotrabajadores`  
--  
ALTER TABLE `registrotrabajadores`  
  MODIFY `ID` int(11) NOT NULL AUTO_INCREMENT,  
  AUTO_INCREMENT=50;  
  
--  
-- AUTO_INCREMENT de la tabla `registro_informe_perito`  
--  
ALTER TABLE `registro_informe_perito`  
  MODIFY `Id` int(11) NOT NULL AUTO_INCREMENT,  
  AUTO_INCREMENT=22;  
  
--  
-- AUTO_INCREMENT de la tabla  
`registro_tarea_cuadrilla`  
--  
ALTER TABLE `registro_tarea_cuadrilla`  
  MODIFY `ID` int(11) NOT NULL AUTO_INCREMENT,  
  AUTO_INCREMENT=37;  
  
--
```

```

-- AUTO_INCREMENT de la tabla `tipo_trabajador`
--
ALTER TABLE `tipo_trabajador`
  MODIFY `id_Tipo` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=6;

--
-- Restricciones para tablas volcadas
--

--
-- Filtros para la tabla `informes`
--
ALTER TABLE `informes`
  ADD CONSTRAINT `informes_ibfk_3` FOREIGN KEY
  (`Cod_Perito`) REFERENCES `trabajadores`
  (`Cod_Trabajador`) ON DELETE CASCADE ON UPDATE
  CASCADE,
  ADD CONSTRAINT `informes_ibfk_4` FOREIGN KEY
  (`Id_Fruta`) REFERENCES `fruta` (`Cod_Fruta`) ON
  DELETE CASCADE ON UPDATE CASCADE;

--
-- Filtros para la tabla `registro`

```

--

ALTER TABLE `registro`

ADD CONSTRAINT `registro_ibfk_2` FOREIGN KEY
(`ID_Tarea`) REFERENCES `tareas` (`Cod_Tarea`),

ADD CONSTRAINT `registro_ibfk_4` FOREIGN KEY
(`ID_Cuadrilla`) REFERENCES `cuadrilla` (`Cod_Cuadrilla`) ON
DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `registro_ibfk_5` FOREIGN KEY
(`ID_Fruta`) REFERENCES `fruta` (`Cod_Fruta`) ON
DELETE CASCADE ON UPDATE CASCADE;

--

-- Filtros para la tabla `trabajadores`

--

ALTER TABLE `trabajadores`

ADD CONSTRAINT `trabajadores_ibfk_1` FOREIGN KEY
(`Cuadrilla`) REFERENCES `cuadrilla` (`Cod_Cuadrilla`) ON
DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `trabajadores_ibfk_2` FOREIGN KEY
(`Tipo`) REFERENCES `tipo_trabajador` (`id_Tipo`) ON
DELETE CASCADE ON UPDATE CASCADE;

COMMIT;