



**PROYECTO FINAL**

**ASIGNATURA: PROGRAMACIÓN**

**PROFESOR: PACO ALDARIAS RAYA**

**CENTRO EDUCATIVO: CEED <http://www.ceedcv.es>**

**CURSO: 2013/14**

**ALUMNO/S:**

**José Marín Malagón**

**Javier Garcia Garcia**

**TITULO:**

**Aplicación que gestiona las entradas y salidas trabajadores en una empresa, muestra estadísticas sobre los trabajadores (días-horas trabajadas, proyectos en los que esta integrado, sueldo...).**

**FECHA:**

**30/03/14**

**CVS:**

**<https://bitbucket.org/jmarinmalagon/prg1314finalgrupo5>**

## ÍNDICE:

### Índice de contenido

1INTRODUCCIÓN .....	3
2DISEÑO DE LA APLICACIÓN .....	3
2.1DIAGRAMA DE CLASES .....	3
2.2DIAGRAMA DE PAQUETES .....	3
2.3DESCRIPCIÓN DEL MVC .....	3
3DISEÑO DE LA BASE DE DATOS .....	4
3.1ANÁLISIS DE REQUISITOS. ....	4
3.2DISEÑO CONCEPTUAL ENTIDAD-RELACIÓN .....	4
3.3DISEÑO LÓGICO RELACIONAL. ....	5
3.4DISEÑO FÍSICO.....	5
3.4.1GRÁFICO. ....	5
3.4.2TABLAS EN MYSQL. ....	6
4CAPTURAS DE PANTALLA.....	8
4.1Login.....	8
5SOFTWARE UTILIZADO PARA EL DESARROLLO. ....	9
6Estructura paquetes en netbeans .....	9
7DESPLIEGUE .....	10
7.1Mysql .....	10
8CONTROL DE VERSIONES .....	10
9PROBLEMAS ENCONTRADOS .....	11
10MEJORAS. ....	11
11BIBLIOGRAFIA.....	11

## 1 INTRODUCCIÓN

La aplicación gestiona las entradas y salidas de los trabajadores en un centro de trabajo, y muestra estadísticas sobre los trabajadores (días-horas trabajadas, proyectos en los que esta integrado, sueldo...). La aplicación esta diseñada utilizando el MVC. Permite guardar la información en ficheros, base de datos relacionales y en base de datos orientada a objetos.

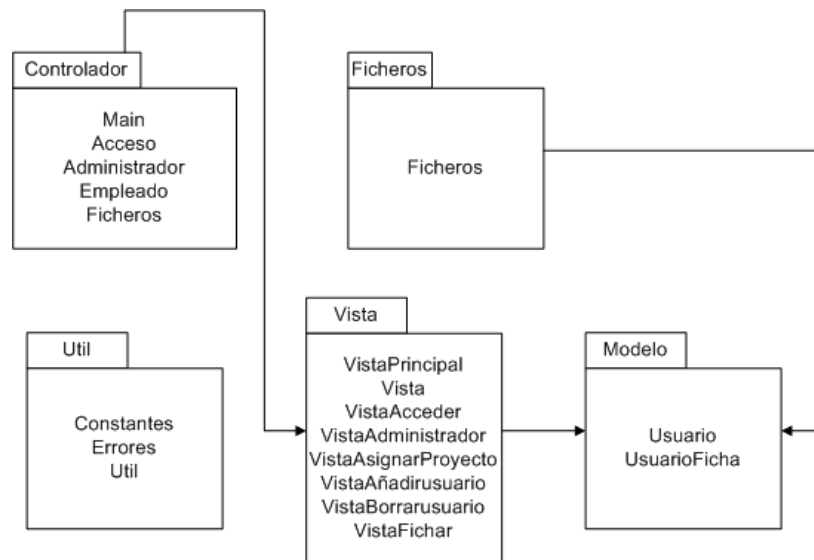
## 2 DISEÑO DE LA APLICACIÓN

### 2.1 DIAGRAMA DE CLASES

El diagrama clases es:

### 2.2 DIAGRAMA DE PAQUETES

El diagrama de paquetes es:



### 2.3 DESCRIPCIÓN DEL MVC

La siguiente aplicación cumple el diseño MVC (Modelo – Vista – Controlador). Para ello se crean los paquetes siguientes:

- **Modelo:** Contiene todas las clases de objetos que se desean guardar información. En este caso coincide con las tablas del diseño Entidad Relación.
- **Vista:** Contienen las clases que interactúan con el usuario. Se ha extraído todo el código posible de las vistas para ponerlo en controlador.
- **Controlador:** Estás las clases que

### 3 DISEÑO DE LA BASE DE DATOS

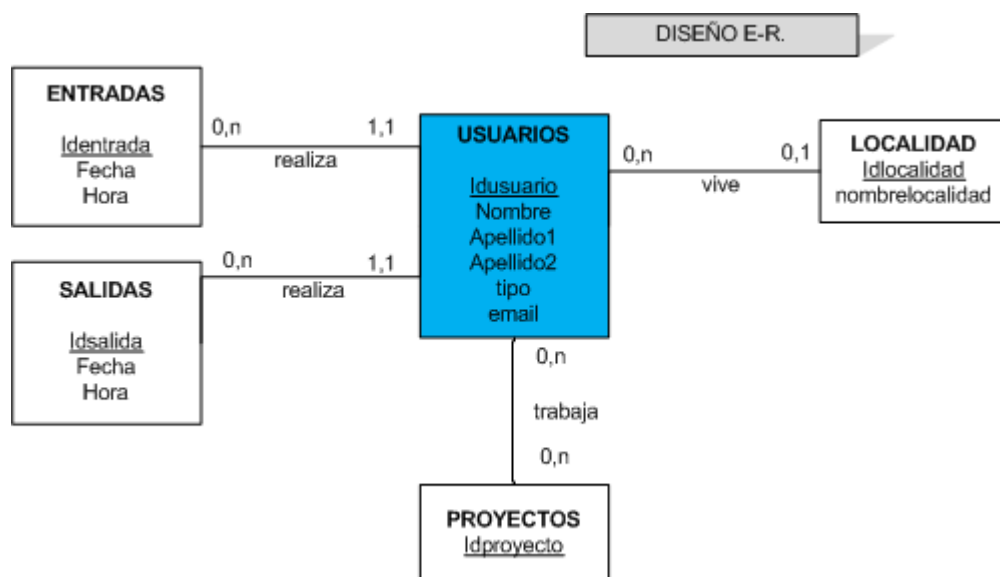
#### 3.1 ANÁLISIS DE REQUISITOS.

Se desea realizar la gestión de entradas y salidas de los trabajadores de una empresa. Para ello se grabarán los datos de los trabajadores, proyectos en los que están integrados y se registrarán las entradas y salidas de los usuarios. Para más tarde, mostrar las estadísticas de cada uno (días-horas trabajadas, proyectos en los que está integrado, sueldo...).

Restricciones de cardinalidad de las relaciones:

- Un usuario puede realizar 0 o muchas salidas.
- Una salida corresponde a un usuario y sólo uno.
- Un usuario puede realizar 0 o muchas entradas.
- Una entrada corresponde a un usuario y sólo uno.
- Un usuario puede vivir en 0 o 1 localidad.
- En una localidad pueden vivir 0 o muchos usuarios.
- En un proyecto pueden trabajar 0 o muchos usuarios.
- Un usuario puede trabajar en 0 o muchos proyectos.

#### 3.2 DISEÑO CONCEPTUAL ENTIDAD-RELACIÓN



Aclaraciones:

Todas las tablas tienen como clave primaria el atributo id, este es un número asignado por el administrador en *idusuario*, *idlocalidad* e *idproyecto*. Mientras que en *identrada* y *idsalida* es número autoincremento simplificando la elección de claves primarias.

### **3.3 DISEÑO LÓGICO RELACIONAL.**

Las relaciones son:

1. USUARIOS ( idusuario,nombre,apellido1,apellido2,email, tipo, idlocalidad)

CP:idusuario

CA:idlocalidad → LOCALIDAD

2. LOCALIDAD(idlocalidad)

CP:localidad

3. ENTRADAS (identrada, idusuario, fecha, hora)

CP: idpedido

VNN: idusuario

CA: idusuario → USUARIOS

4. SALIDAS (idsalida, idusuario, fecha, hora)

CP: idpedido

VNN: idusuario

CA: idusuario → USUARIOS

5. PROYECTOS (idproyecto,tipo, precio\_hora)

CP: idpedido

6. PROYECTO-USUARIOS (idusuario, idproyecto)

CP: idusuario, idproyecto

CA: idusuario → USUARIOS

CA: idproyecto → PROYECTOS

NOTA:

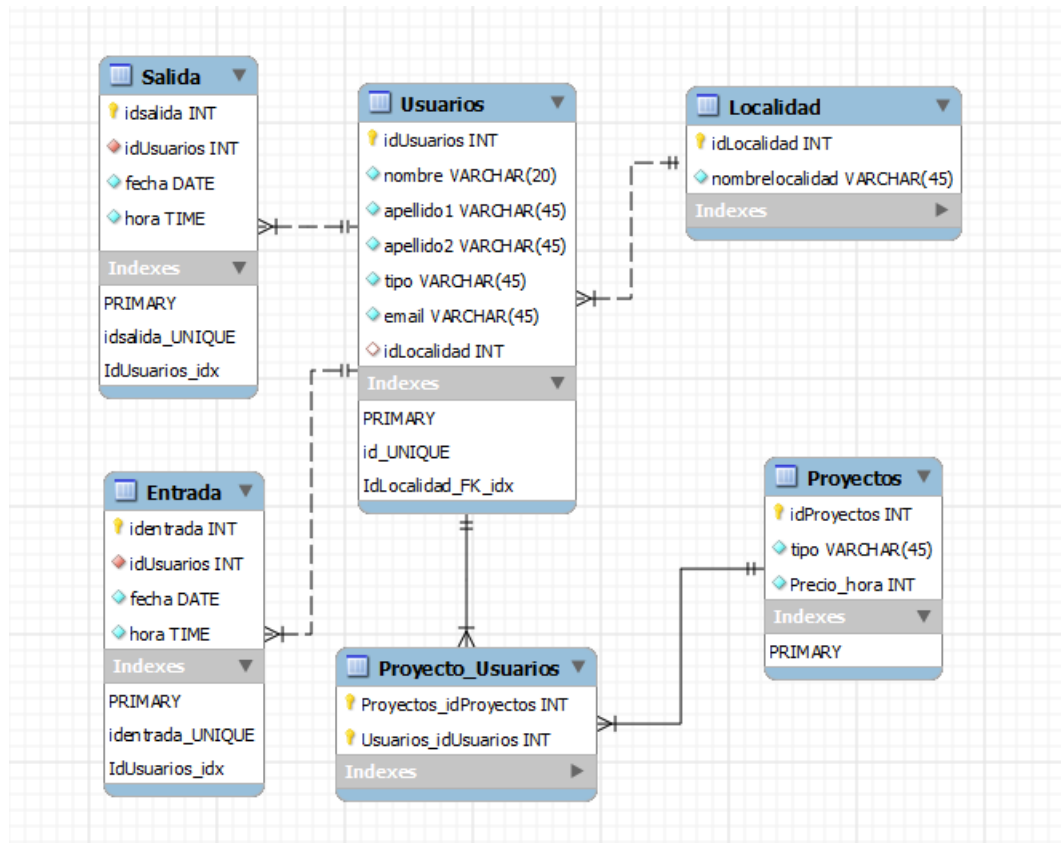
- CP: Indica Clave Primaria
- CA: Indica Clave Ajena

Todas las relaciones están en 3ra Forma Normal.

### 3.4 DISEÑO FÍSICO.

El diseño físico se va a realizar con el SGBD Mysql.

#### 3.4.1 GRÁFICO.



#### 3.4.2 TABLAS EN MYSQL.

El diseño físico se va a realizar sobre la base de datos relacional MYSQL.

Se encuentra en el fichero tienda.sql. Se deberá importar una vez creada la base de datos.

```
-- Table `Localidad`
```

```
CREATE TABLE `Localidad` (  
  `idLocalidad` INT NOT NULL,  
  `nombrelocalidad` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`idLocalidad`)  
) ENGINE=InnoDB;
```

Proyecto final programación. prg1314finalgrupo5. CEED.

```
CREATE UNIQUE INDEX `idLocalidad_UNIQUE` ON `Localidad` (`idLocalidad` ASC);
```

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

```
INSERT INTO `Localidad` VALUES (46240, 'Carlet');
INSERT INTO `Localidad` VALUES (46470, 'Catarroja');
INSERT INTO `Localidad` VALUES (46023, 'Valencia');
INSERT INTO `Localidad` VALUES (46900, 'Torrent');
INSERT INTO `Localidad` VALUES (46970, 'Alaquas');
```

```
-- Table `Usuarios`
```

```
CREATE TABLE IF NOT EXISTS `Usuarios` (
  `idUsuarios` INT NOT NULL,
  `nombre` VARCHAR(20) NOT NULL,
  `apellido1` VARCHAR(45) NOT NULL,
  `apellido2` VARCHAR(45) NOT NULL,
  `tipo` VARCHAR(45) NOT NULL,
  `email` VARCHAR(45) NOT NULL,
  `idLocalidad` INT NOT NULL,
  PRIMARY KEY (`idUsuarios`),
  CONSTRAINT `idLocalidad_FK` FOREIGN KEY (`idLocalidad`)
    REFERENCES `Localidad` (`idLocalidad`)
) ENGINE=InnoDB;
```

```
CREATE UNIQUE INDEX `id_UNIQUE` ON `Usuarios` (`idUsuarios` ASC);
CREATE INDEX `idLocalidad_FK_idx` ON `Usuarios` (`idLocalidad` ASC);
```

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

```
INSERT INTO `Usuarios` VALUES
(12345,'Jose','Marin','Malagon','admin','jmarin@gmail.com',46023);
INSERT INTO `Usuarios` VALUES (23456,
'Luisa','Fernandez','Vera','user1','luferve@gmail.com',46023);
INSERT INTO `Usuarios` VALUES (34567,
'Julio','Marin','Campos','user1','jumarca@gmail.com',46470);
```

```
INSERT INTO `Usuarios` VALUES (45678,  
'Javier','Garcia','Garcia','admin','jagarga@gmail.com',46900);  
INSERT INTO `Usuarios` VALUES (56789,  
'Emilio','Pascual','Martinez','user1','empamar@gmail.com',46023);  
INSERT INTO `Usuarios` VALUES (67890,  
'Amparo','Marzal','Pascual','user2','amarpas@gmail.com',46900);  
INSERT INTO `Usuarios` VALUES (78901,  
'Sergio','Moreno','Blanco','user2','semobla@gmail.com',46023);
```

```
-- -----  
-- Table `Entrada`  
-- -----
```

```
CREATE TABLE IF NOT EXISTS `Entrada` (  
  `identrada` INT NOT NULL AUTO_INCREMENT,  
  `idUsuarios` INT NOT NULL,  
  `fecha` DATE NOT NULL,  
  `hora` TIME NOT NULL,  
  PRIMARY KEY (`identrada`),  
  CONSTRAINT `IdUsuarios`  
    FOREIGN KEY (`idUsuarios`)  
    REFERENCES `Usuarios` (`idUsuarios`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;  
  
CREATE UNIQUE INDEX `identrada_UNIQUE` ON `Entrada` (`identrada` ASC);  
CREATE INDEX `IdUsuarios_idx` ON `Entrada` (`idUsuarios` ASC);
```

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

```
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (12345,'2014-01-20','10:11:12');  
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (23456, '2014-01-20','09:00:12');  
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (34567, '2014-01-20','09:01:12');  
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (45678, '2014-01-20','09:00:00');  
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (56789, '2014-01-20','09:01:00');  
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (67890, '2014-01-20','09:00:00');  
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (78901, '2014-01-20','09:01:34');  
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (12345, '2014-01-20','15:00:00');
```



```
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (23456, '2014-01-20','15:01:34');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (34567, '2014-01-20','15:00:00');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (45678, '2014-01-20','15:01:34');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (56789, '2014-01-20','15:00:00');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (67890, '2014-01-20','15:01:34');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (78901, '2014-01-20','15:00:00');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (12345, '2014-01-21','10:11:12');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (23456, '2014-01-21','09:00:12');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (34567, '2014-01-21','09:01:12');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (45678, '2014-01-21','09:00:00');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (56789, '2014-01-21','09:01:00');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (67890, '2014-01-21','09:00:00');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (78901, '2014-01-21','09:01:34');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (12345, '2014-01-21','15:00:00');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (23456, '2014-01-21','15:01:34');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (34567, '2014-01-21','15:00:00');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (45678, '2014-01-21','15:01:34');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (56789, '2014-01-21','15:00:00');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (67890, '2014-01-21','15:01:34');
INSERT INTO `Entrada` (`idUsuarios`,`fecha`,`hora`) VALUES (78901, '2014-01-12','15:00:00');
```

```
-- -----
-- Table `Salida`
-- -----
```

```
CREATE TABLE IF NOT EXISTS `Salida` (
  `idsalida` INT NOT NULL AUTO_INCREMENT,
  `idUsuarios` INT NOT NULL,
  `fecha` DATE NOT NULL,
  `hora` TIME NOT NULL,
  PRIMARY KEY (`idsalida`),
  CONSTRAINT `IdUsuarios_FK_Salida`
  FOREIGN KEY (`idUsuarios`)
  REFERENCES `Usuarios` (`idUsuarios`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
CREATE UNIQUE INDEX `idsalida_UNIQUE` ON `Salida` (`idsalida` ASC);  
CREATE INDEX `IdUsuarios_idx` ON `Salida` (`idUsuarios` ASC);
```

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

```
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (12345, '2014-01-20','13:11:12');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (23456, '2014-01-20','13:00:12');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (34567, '2014-01-20','13:01:12');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (45678, '2014-01-20','13:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (56789, '2014-01-20','13:01:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (67890, '2014-01-20','13:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (78901, '2014-01-20','13:01:34');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (12345, '2014-01-20','17:01:34');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (23456, '2014-01-20','17:01:34');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (34567, '2014-01-20','17:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (45678, '2014-01-20','17:01:34');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (56789, '2014-01-20','17:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (67890, '2014-01-20','17:01:34');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (78901, '2014-01-20','18:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (12345, '2014-01-21','13:11:12');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (23456, '2014-01-21','13:00:12');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (34567, '2014-01-21','13:01:12');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (45678, '2014-01-21','13:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (56789, '2014-01-21','13:01:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (67890, '2014-01-21','13:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (78901, '2014-01-21','13:01:34');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (12345, '2014-01-21','18:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (23456, '2014-01-21','17:01:34');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (34567, '2014-01-21','17:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (45678, '2014-01-21','17:01:34');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (56789, '2014-01-21','17:00:00');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (67890, '2014-01-21','17:01:34');  
INSERT INTO `Salida` (`idUsuarios`,`fecha`,`hora`) VALUES (78901, '2014-01-21','18:00:00');
```

-----

-- Table `Proyectos`

```
-----  
CREATE TABLE IF NOT EXISTS `Proyectos` (  
  `idProyectos` INT NOT NULL,  
  `tipo` VARCHAR(45) NOT NULL,  
  `Precio_hora` INT NOT NULL,  
  PRIMARY KEY (`idProyectos`))  
ENGINE = InnoDB;
```

-- Volcado de datos para la tabla `Proyectos`

```
INSERT INTO `Proyectos` VALUES (54321, 'Diseño web',13);  
INSERT INTO `Proyectos` VALUES (43210, 'Aplicaciones multimedia',19);  
INSERT INTO `Proyectos` VALUES (98765, 'Bases de datos',11);
```

-- Table `Proyecto\_Usuarios`

```
-----  
CREATE TABLE IF NOT EXISTS `Proyecto_Usuarios` (  
  `Proyectos_idProyectos` INT NOT NULL,  
  `Usuarios_idUsuarios` INT NOT NULL,  
  PRIMARY KEY (`Proyectos_idProyectos`, `Usuarios_idUsuarios`),  
  CONSTRAINT `fk_Proyectos_has_Usuarios_Proyectos1`  
    FOREIGN KEY (`Proyectos_idProyectos`)  
    REFERENCES `Proyectos` (`idProyectos`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Proyectos_has_Usuarios_Usuarios1`  
    FOREIGN KEY (`Usuarios_idUsuarios`)  
    REFERENCES `Usuarios` (`idUsuarios`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
CREATE INDEX `fk_Proyectos_has_Usuarios_Usuarios1_idx` ON `Proyecto_Usuarios`  
(`Usuarios_idUsuarios` ASC);  
CREATE INDEX `fk_Proyectos_has_Usuarios_Proyectos1_idx` ON `Proyecto_Usuarios`
```

Proyecto final programación. prg1314finalgrupo5. CEED.

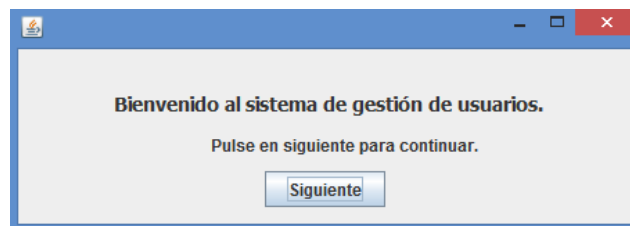
```
(`Proyectos_idProyectos` ASC);
```

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

```
INSERT INTO `Proyecto_Usuarios` VALUES (54321, 12345);  
INSERT INTO `Proyecto_Usuarios` VALUES (54321, 23456);  
INSERT INTO `Proyecto_Usuarios` VALUES (54321, 34567);  
INSERT INTO `Proyecto_Usuarios` VALUES (43210, 45678);  
INSERT INTO `Proyecto_Usuarios` VALUES (43210, 56789);  
INSERT INTO `Proyecto_Usuarios` VALUES (98765, 67890);
```

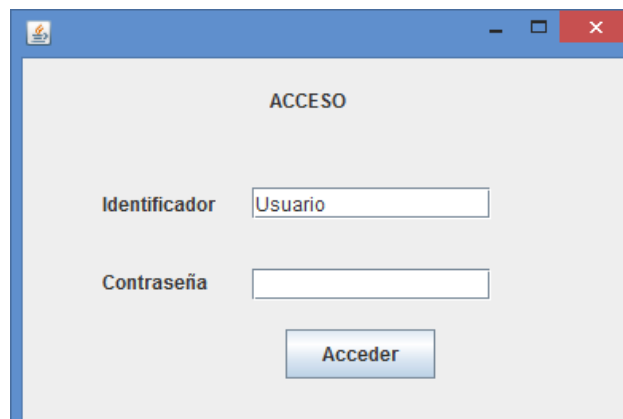
## 4 CAPTURAS DE PANTALLA

### 4.1 Pantalla Bienvenida.



### 4.2 Acceso.

Poniendo usuario: 1 y contraseña: 123. Podemos acceder a la aplicación como administrador.



4.3 Menú administrador.

The screenshot shows a window titled "MENU ADMINISTRADOR". On the left side, the text "Javier" is displayed in a large font, with "30-3-2014" and "21:47:55" below it. At the bottom left is a "Salir" button. On the right side, there are three buttons stacked vertically: "Añadir usuario", "Borrar usuario", and "Asignar proyecto".

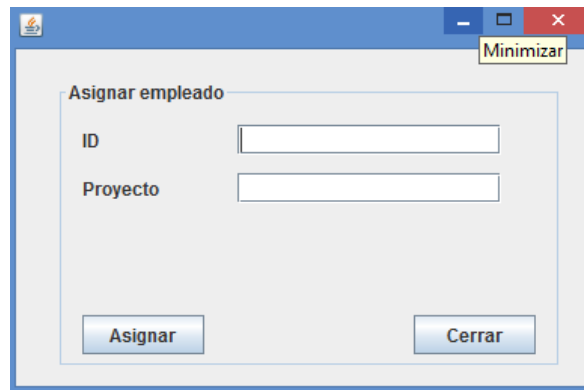
4.4. Añadir empleados.

The screenshot shows a window titled "Añadir empleados". It contains a form with the following fields: "ID", "Nombre", "1ºApellido", "2ºApellido", "Email", and "Contraseña", each with a corresponding text input box. Below these fields is a "Tipo" label followed by a dropdown menu currently showing "Administr...". At the bottom of the form are two buttons: "Añadir" and "Cerrar".

4.5. Borra empleado.

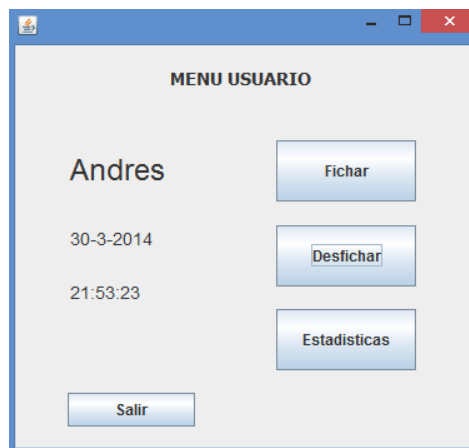
The screenshot shows a window titled "Borrar empleados". It contains a form with a single field labeled "ID" with a text input box. At the bottom of the form are two buttons: "Borrar" and "Cerrar".

#### 4.6 Asignar proyecto.

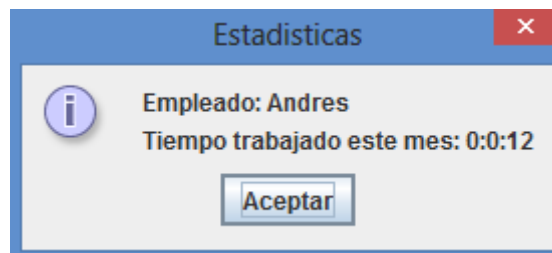


#### 4.7 Menú usuario.

Poniendo usuario: 2 y contraseña: 456. Podemos acceder a la interfaz de usuario.



#### 4.8 Estadísticas usuario.



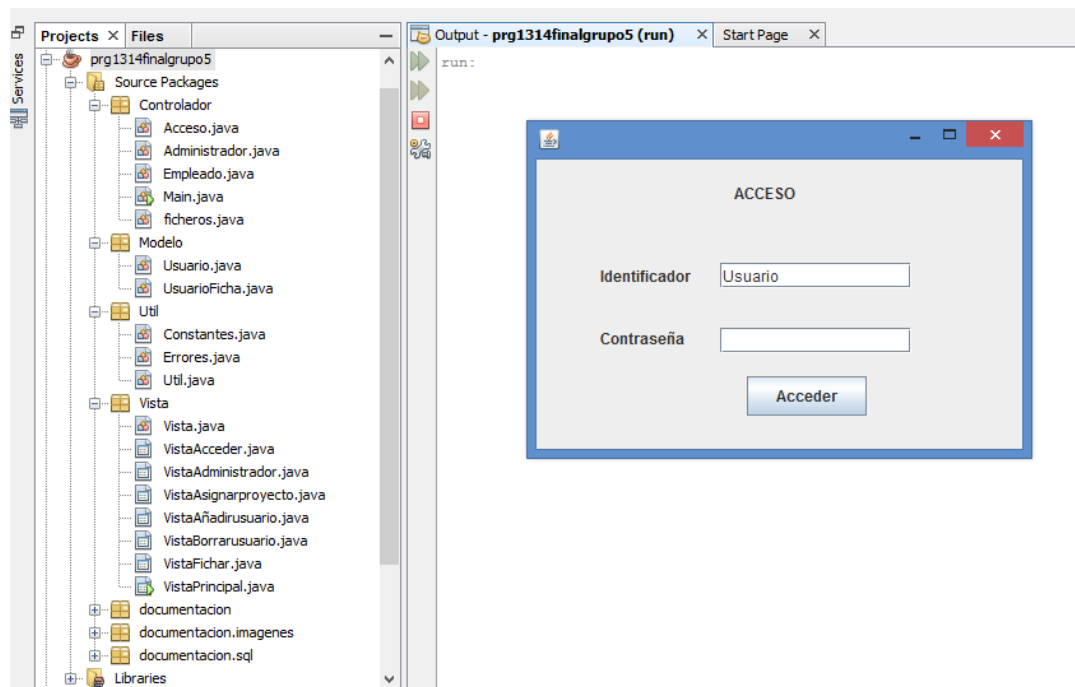
## 5 SOFTWARE UTILIZADO PARA EL DESARROLLO.

Para el desarrollo de la aplicación ha necesitado de :

- Netbeans
- Librerías JDK 1.7
- Librerías de Db40
- Mysql
- Linux

## 6 ESTRUCTURA PAQUETES EN NETBEANS

Los paquetes que se han creado en el proyecto se pueden ver en la siguiente imagen.



## 7 DESPLIEGUE

Requiere tener instalado:

1. Mysql
2. Maquina Virtual Jva o JRE. Para ejecutar el fichero .jar.

### 7.1 Mysql

Crear la base de datos **prg1314finalgrupo5** e importar el fichero **gestrabajo.sql** que se encuentra en la carpeta documentación/sql.

## **8 CONTROL DE VERSIONES**

La aplicación se ha realizado utilizando el control de versiones git y el servidor bitbucket.

Para clonar el proyecto debemos realizar:

git clone <https://bitbucket.org/jmarinmalagon/prg1314finalgrupo5>

## **9 PROBLEMAS ENCONTRADOS**

Ha sido difícil encontrar ejemplo completos de java SE, Netbeans y IGU.

## **10 MEJORAS.**

## **11 BIBLIOGRAFIA.**

### **1. Tutorial Programación pantalla de Acceso(Login) en JAVA Netbeans**

**<http://www.youtube.com/watch?v=ayCeQh8IGgA>**

### **2. Tienda Online con Java EE**

**<https://code.google.com/p/tiendaonlinepijdyb/>**

### **3. Proyecto Universidad**

**<http://www.youtube.com/watch?v=C5PScZ7HN-k>**

### **4. Página web con Iconos.**

**<http://icones.pro/es/>**

### **5. Instancias de una clase:**

**<http://migranitodejava.blogspot.com.es/2011/05/singleton.html>**

### **6. Mini Proyecto de Facturación en Java**

**<http://www.youtube.com/watch?v=NQM5M-C37Qo>**

7.