

# **Escola Universitària Politécnica de Mataró**

Centre adscrit a:



**UNIVERSITAT POLITÈCNICA  
DE CATALUNYA**

**Enginyeria Tècnica de Telecomunicacions: Especialitat Telemàtica**

**DESARROLLO DE UN SOFTWARE DE GESTIÓN DE MANTENIMIENTO  
ASISTIDO POR ORDENADOR (GMAO) PARA PYMES**

**Anexos**

**JUAN PABLO MARCO CARDONA  
PONENTE: EDUARD DE BRU DE SALA CASTELLS**

**TARDOR 2012**



**TecnoCampus  
Mataró-Maresme**



## Indice.

Anexo I. Modelo físico de la base de datos.....	1
Anexo II. Manual de usuario del programa GMAO v1.0.....	23
Anexo II.1 - Menú principal.....	23
Anexo II.1 - Menú incidencias.....	24
Anexo II.1.2 – Formulario de Correctivo.....	25
Anexo II.1.3 – Formulario Gestión del Conocimiento.....	26
Anexo II.2 - Menú Equipos.....	28
Anexo II.2.1 – Formulario Equipos.....	29
Anexo II.2.2 – Formulario Recambios.....	30
Anexo II.2.3 – Formulario Líneas.....	31
Anexo II.2.4 – Formulario Categorías Recambios.....	32
Anexo II.3 - Menú Trabajos.....	32
Anexo II.3.1 – Formulario Operaciones.....	33
Anexo II.3.2 – Formulario Gamas.....	34
Anexo II.3.3 – Formulario Workorders.....	38
Anexo II.3.4 – Formulario Lanzar planificación.....	43
Anexo II.3.5 – Formulario Listados.....	43
Anexo II.3.6 – Menú Predictivos.....	44
Anexo II.3.6.1 – Formulario Predictivo equipos.....	44
Anexo II.3.6.1 – Formulario Lecturas.....	46
Anexo II.4 - Menú Recursos Humanos.....	47
Anexo II.4.1 – Formulario Proveedores.....	48
Anexo II.4.2 – Formulario Tipo de suministros.....	49
Anexo II.4.3 – Formulario Operarios.....	49
Anexo II.4.4 – Formulario Categorías profesionales.....	51
Anexo II.4.5 – Formulario Turnos.....	51
Anexo II.5 – Formulario Listados.....	51
Anexo III. Contenido del CD-ROM.....	53



## **Anexo I. Modelo físico de la base de datos.**

A continuación se muestra el modelo físico de la base de datos del programa GMAO.

Para realizar este modelo se utiliza la herramienta “MS Access to MySQL” que permite generar en un formato del SGDB MySQL la estructura de tablas del GMAO. Así como las relaciones entre tablas a través de las claves foráneas.

```
DROP TABLE IF EXISTS `categoria_profesional`;
```

```
CREATE TABLE `categoria_profesional` (  
  `id_categoria_profesional` INTEGER NOT NULL AUTO_INCREMENT,  
  `categoria_profesional` VARCHAR(255),  
  PRIMARY KEY (`id_categoria_profesional`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'categorias'
```

```
#
```

```
DROP TABLE IF EXISTS `categorias`;
```

```
CREATE TABLE `categorias` (  
  `id_categoria` INTEGER NOT NULL AUTO_INCREMENT,  
  `nombre_categoria` VARCHAR(50),  
  INDEX (`nombre_categoria`),  
  PRIMARY KEY (`id_categoria`)
```

```
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'conocimientos'
```

```
#
```

```
DROP TABLE IF EXISTS `conocimientos`;
```

```
CREATE TABLE `conocimientos` (
```

```
  `id_conocimiento` INTEGER NOT NULL AUTO_INCREMENT,
```

```
  `id_equipo` INTEGER,
```

```
  `elemento` VARCHAR(255),
```

```
  `fecha_registro` DATETIME,
```

```
  `problema` VARCHAR(255),
```

```
  `sintoma` VARCHAR(255),
```

```
  `solucion` VARCHAR(255),
```

```
  `palabra_clave_sintoma` VARCHAR(255),
```

```
  `palabra_clave_solucion` VARCHAR(255),
```

```
  `notas` VARCHAR(255),
```

```
  `path_documentacion` VARCHAR(255),
```

```
  `path_imagen` VARCHAR(255),
```

```
  INDEX (`id_equipo`),
```

```
  PRIMARY KEY (`id_conocimiento`)
```

```
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#  
# Table structure for table 'equipos'  
#  
  
DROP TABLE IF EXISTS `equipos`;  
  
CREATE TABLE `equipos` (  
  `id_equipo` INTEGER NOT NULL AUTO_INCREMENT,  
  `codigo_equipo` VARCHAR(250) NOT NULL,  
  `nombre_equipo` VARCHAR(250) NOT NULL,  
  `tipo_equipo` VARCHAR(250),  
  `estado` VARCHAR(50) DEFAULT 'EN FUNCIONAMIENTO',  
  `fecha_cambio_estado` DATETIME,  
  `id_fabricante` INTEGER DEFAULT 0,  
  `n_serie` VARCHAR(250),  
  `fecha_adquisicion` DATETIME,  
  `observaciones` LONGTEXT,  
  `horas_por_dia` INTEGER NOT NULL DEFAULT 24,  
  `dias_por_semana` INTEGER NOT NULL DEFAULT 5,  
  `imagen` VARCHAR(250),  
  `id_linea` INTEGER DEFAULT 1,  
  INDEX (`id_linea`),  
  UNIQUE (`codigo_equipo`),  
  PRIMARY KEY (`id_equipo`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;  
  
SET autocommit=1;  
  
#
```

```
# Table structure for table 'especialidades'
```

```
#
```

```
DROP TABLE IF EXISTS `especialidades`;
```

```
CREATE TABLE `especialidades` (  
  `id_especialidad` INTEGER NOT NULL AUTO_INCREMENT,  
  `nombre_especialidad` VARCHAR(50),  
  INDEX (`id_especialidad`),  
  PRIMARY KEY (`id_especialidad`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'gamas'
```

```
#
```

```
DROP TABLE IF EXISTS `gamas`;
```

```
CREATE TABLE `gamas` (  
  `id_gama` INTEGER NOT NULL AUTO_INCREMENT,  
  `nombre_gama` VARCHAR(250) NOT NULL,  
  `denominacion_gama` VARCHAR(250) NOT NULL,  
  `fecha_modificacion` DATETIME,  
  `clase_trabajo` VARCHAR(250) NOT NULL,  
  `observaciones` LONGTEXT,  
  PRIMARY KEY (`id_gama`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```



```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'gamas_asociadas_equipos'
```

```
#
```

```
DROP TABLE IF EXISTS `gamas_asociadas_equipos`;
```

```
CREATE TABLE `gamas_asociadas_equipos` (
```

```
  `id_gama_asociada` INTEGER NOT NULL AUTO_INCREMENT,
```

```
  `id_equipo` INTEGER NOT NULL DEFAULT 0,
```

```
  `id_gama` INTEGER NOT NULL DEFAULT 0,
```

```
  `fecha_asociacion` DATETIME,
```

```
  `fecha_ultima_realizacion` DATETIME,
```

```
  `horas_ciclos` INTEGER,
```

```
  `ciclos_vs_horas_inicio` INTEGER,
```

```
  `tiempo_estimado` INTEGER,
```

```
  `horas_operarios` INTEGER DEFAULT 0,
```

```
  `documentacion_anexa` VARCHAR(100),
```

```
  INDEX (`id_gama`),
```

```
  INDEX (`id_equipo`),
```

```
  PRIMARY KEY (`id_gama_asociada`)
```

```
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

#

# Table structure for table 'gastos\_workorder'

#

DROP TABLE IF EXISTS `gastos\_workorder`;

```
CREATE TABLE `gastos_workorder` (  
  `id_workorder_gasto` INTEGER NOT NULL AUTO_INCREMENT,  
  `gasto` DECIMAL(19,4) NOT NULL DEFAULT 0,  
  `concepto` VARCHAR(50),  
  `id_workorder` INTEGER NOT NULL,  
  `fecha` DATETIME,  
  INDEX (`id_workorder`),  
  PRIMARY KEY (`id_workorder_gasto`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

SET autocommit=1;

#

# Table structure for table 'gestor\_conocimiento'

#

DROP TABLE IF EXISTS `gestor\_conocimiento`;

```
CREATE TABLE `gestor_conocimiento` (  
  `id_mejora` INTEGER NOT NULL AUTO_INCREMENT,  
  `fecha` DATETIME,
```

```
`id_tipo` INTEGER,  
`id_equipo` INTEGER DEFAULT 0,  
`id_elemento` INTEGER DEFAULT 0,  
`causa` VARCHAR(50),  
`palabra_clave` VARCHAR(255),  
`solucion` LONGTEXT,  
INDEX (`id_equipo`),  
INDEX (`id_mejora`),  
PRIMARY KEY (`id_mejora`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'lineas'
```

```
#
```

```
DROP TABLE IF EXISTS `lineas`;
```

```
CREATE TABLE `lineas` (  
  `id_linea` INTEGER NOT NULL AUTO_INCREMENT,  
  `nombre_linea` VARCHAR(255),  
  `observaciones` VARCHAR(255),  
  PRIMARY KEY (`id_linea`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

#

# Table structure for table 'mano\_obra\_workorder'

#

DROP TABLE IF EXISTS `mano\_obra\_workorder`;

```
CREATE TABLE `mano_obra_workorder` (  
  `id_mano_obra_workorder` INTEGER NOT NULL AUTO_INCREMENT,  
  `id_operario` INTEGER NOT NULL,  
  `id_workorder` INTEGER NOT NULL DEFAULT 0,  
  `hora_inicio` DATETIME NOT NULL,  
  `hora_fin` DATETIME NOT NULL,  
  `total_horas` DATETIME,  
  `coste_hora_imputacion` DECIMAL(19,4),  
  INDEX (`id_operario`),  
  PRIMARY KEY (`id_mano_obra_workorder`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

SET autocommit=1;

#

# Table structure for table 'normas'

#

DROP TABLE IF EXISTS `normas`;

```
CREATE TABLE `normas` (  
  `id_norma` INTEGER NOT NULL AUTO_INCREMENT,
```

```
`norma` VARCHAR(250) NOT NULL,  
`elemento` VARCHAR(250),  
`especialidad` INTEGER,  
`descripcion` LONGTEXT,  
`clase_mtto` VARCHAR(255) NOT NULL,  
`fecha_modificacion` DATETIME,  
PRIMARY KEY (`id_norma`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'normas_en_gamas'
```

```
#
```

```
DROP TABLE IF EXISTS `normas_en_gamas`;
```

```
CREATE TABLE `normas_en_gamas` (  
  `id_norma_en_gama` INTEGER NOT NULL AUTO_INCREMENT,  
  `id_gama` INTEGER NOT NULL,  
  `id_norma` INTEGER,  
  INDEX (`id_gama`),  
  INDEX (`id_norma`),  
  PRIMARY KEY (`id_norma_en_gama`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

# Table structure for table 'operarios'

#

DROP TABLE IF EXISTS `operarios`;

CREATE TABLE `operarios` (

`id\_operario` INTEGER NOT NULL AUTO\_INCREMENT,

`cod\_operario` VARCHAR(6) NOT NULL,

`nombre\_operario` VARCHAR(50) NOT NULL,

`id\_especialidad` INTEGER DEFAULT 0,

`id\_categoria\_profesional` INTEGER,

`telefono` VARCHAR(15),

`direccion` VARCHAR(254),

`cod\_postal` VARCHAR(5),

`curriculum` VARCHAR(255),

`id\_empresa` INTEGER NOT NULL,

`turno` INTEGER DEFAULT 1,

`coste\_hora` DECIMAL(19,4) DEFAULT 0,

`coste\_hora\_extra` DECIMAL(19,4) DEFAULT 0,

`coste\_hora\_festiva` DECIMAL(19,4) DEFAULT 0,

UNIQUE (`cod\_operario`),

INDEX (`id\_especialidad`),

PRIMARY KEY (`id\_operario`)

) ENGINE=innodb DEFAULT CHARSET=utf8;

SET autocommit=1;

#

# Table structure for table 'predictivo\_equipos'

#

```
DROP TABLE IF EXISTS `predictivo_equipos`;
```

```
CREATE TABLE `predictivo_equipos` (  
  `id_predictivo_equipos` INTEGER NOT NULL AUTO_INCREMENT,  
  `id_gama` INTEGER,  
  `id_equipo` INTEGER,  
  `variable` VARCHAR(255),  
  `valor_objetivo` INTEGER,  
  `unidades` VARCHAR(255),  
  `valor_superior` INTEGER,  
  `valor_inferior` INTEGER,  
  INDEX (`id_equipo`),  
  INDEX (`id_gama`),  
  PRIMARY KEY (`id_predictivo_equipos`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

#

```
# Table structure for table 'predictivo_lecturas'
```

#

```
DROP TABLE IF EXISTS `predictivo_lecturas`;
```

```
CREATE TABLE `predictivo_lecturas` (  
  `id_predictivo_lecturas` INTEGER NOT NULL AUTO_INCREMENT,  
  `id_gama` INTEGER,
```

```
`id_equipo` INTEGER,  
`variable` VARCHAR(255),  
`valor_lectura` INTEGER,  
`unidades` VARCHAR(255),  
`fecha_lectura` DATETIME,  
`resultado` VARCHAR(255),  
INDEX (`id_equipo`),  
INDEX (`id_gama`),  
PRIMARY KEY (`id_predictivo_lecturas`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'proveedores'
```

```
#
```

```
DROP TABLE IF EXISTS `proveedores`;
```

```
CREATE TABLE `proveedores` (  
  `id_proveedor` INTEGER NOT NULL AUTO_INCREMENT,  
  `id_tipo_suministro` VARCHAR(255),  
  `nombre_proveedor` VARCHAR(50),  
  `numero_cliente` VARCHAR(10),  
  `persona_de_contacto` VARCHAR(50),  
  `direccion` VARCHAR(255),  
  `ciudad` VARCHAR(50),  
  `codpostal` VARCHAR(10),  
  `edo_o_prov` VARCHAR(20) DEFAULT 'Barcelona',
```



```
`pais` VARCHAR(50) DEFAULT 'Españ±a',
`telefono` VARCHAR(30),
`fax` VARCHAR(30),
`email` VARCHAR(50),
`notas` LONGTEXT,
`iso_evaluacion` VARCHAR(50),
INDEX (`codpostal`),
INDEX (`email`),
INDEX (`persona_de_contacto`),
INDEX (`nombre_proveedor`),
INDEX (`fax`),
INDEX (`telefono`),
PRIMARY KEY (`id_proveedor`)
) ENGINE=innodb DEFAULT CHARSET=utf8;

SET autocommit=1;

#
# Table structure for table 'recambios'
#

DROP TABLE IF EXISTS `recambios`;

CREATE TABLE `recambios` (
  `id_recambio` INTEGER NOT NULL AUTO_INCREMENT,
  `nombre_recambio` VARCHAR(50),
  `referencia` VARCHAR(50),
  `descripcion` VARCHAR(255),
  `id_categoria` INTEGER,
```

```
`id_fabricante` INTEGER,  
`id_proveedor` INTEGER,  
`id_proveedor_alternativo` INTEGER DEFAULT 0,  
`unidades_en_existencia` INTEGER,  
`unidades_en_reserva` INTEGER,  
`unidades_pedidas` INTEGER,  
`unidad_de_medida` VARCHAR(255),  
`precio_unitario` DECIMAL(19,4),  
`nivel_nuevo_pedido` INTEGER,  
`tiempo_aprovisionamiento` VARCHAR(30),  
`observaciones` LONGTEXT,  
INDEX (`id_categoria`),  
INDEX (`id_fabricante`),  
INDEX (`id_proveedor`),  
INDEX (`id_proveedor_alternativo`),  
INDEX (`nombre_recambio`),  
PRIMARY KEY (`id_recambio`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'recambios_en_gamas'
```

```
#
```

```
DROP TABLE IF EXISTS `recambios_en_gamas`;
```

```
CREATE TABLE `recambios_en_gamas` (
```

```
`id_recambio_en_gama` INTEGER NOT NULL AUTO_INCREMENT,  
`id_recambio` INTEGER NOT NULL,  
`id_gama` INTEGER NOT NULL,  
`unidades` INTEGER,  
INDEX (`id_gama`),  
INDEX (`id_recambio`),  
PRIMARY KEY (`id_recambio_en_gama`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'recambios_equipos'
```

```
#
```

```
DROP TABLE IF EXISTS `recambios_equipos`;
```

```
CREATE TABLE `recambios_equipos` (  
  `id_recambio_equipo` INTEGER NOT NULL AUTO_INCREMENT,  
  `id_equipo` INTEGER,  
  `situacion` VARCHAR(50),  
  `id_recambio` INTEGER DEFAULT 0,  
  `cantidad` INTEGER DEFAULT 0,  
  `criticidad` VARCHAR(50),  
  INDEX (`id_equipo`),  
  INDEX (`id_recambio`),  
  PRIMARY KEY (`id_recambio_equipo`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'tipo_suministros'
```

```
#
```

```
DROP TABLE IF EXISTS `tipo_suministros`;
```

```
CREATE TABLE `tipo_suministros` (  
  `id_tipo_suministro` INTEGER NOT NULL AUTO_INCREMENT,  
  `nombre_tipo_suministro` VARCHAR(50),  
  INDEX (`nombre_tipo_suministro`),  
  PRIMARY KEY (`id_tipo_suministro`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'tipos_incidentes'
```

```
#
```

```
DROP TABLE IF EXISTS `tipos_incidentes`;
```

```
CREATE TABLE `tipos_incidentes` (  
  `id_tipo_incidencia` INTEGER NOT NULL AUTO_INCREMENT,  
  `nombre_tipo_incidencia` VARCHAR(50) NOT NULL,  
  PRIMARY KEY (`id_tipo_incidencia`)
```

```
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
# Table structure for table 'turnos'
```

```
#
```

```
DROP TABLE IF EXISTS `turnos`;
```

```
CREATE TABLE `turnos` (
```

```
  `id_turno` INTEGER NOT NULL AUTO_INCREMENT,
```

```
  `nombre_turno` VARCHAR(25) NOT NULL,
```

```
  INDEX (`id_turno`),
```

```
  PRIMARY KEY (`id_turno`)
```

```
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
#
```

```
# Table structure for table 'workorders'
```

```
#
```

```
DROP TABLE IF EXISTS `workorders`;
```

```
CREATE TABLE `workorders` (
```

```
  `id_workorder` INTEGER NOT NULL AUTO_INCREMENT,
```

```
  `codigo_workorder` INTEGER,
```

```
  `descripcion_workorder` VARCHAR(250),
```

```
  `fecha_creacion` DATETIME,
```

```
`fecha_programacion` DATETIME NOT NULL,  
`id_turno` INTEGER,  
`solicitante` VARCHAR(250) NOT NULL,  
`responsable_ejecucion` INTEGER,  
`id_equipo` INTEGER NOT NULL,  
`id_gama` INTEGER NOT NULL,  
`id_tipo_incidencia` INTEGER,  
`clase_mtto` VARCHAR(255) DEFAULT 'PREVENTIVO',  
`descripcion` VARCHAR(255),  
`estado_workorder` VARCHAR(255) DEFAULT 'PENDIENTE',  
`fecha_cierre` DATETIME,  
`tiempo_paro` VARCHAR(100),  
`coste_mano_obra` DECIMAL(19,4) DEFAULT 0,  
`coste_otros` DECIMAL(19,4) DEFAULT 0,  
`coste_recambios` DECIMAL(19,4) DEFAULT 0,  
`observaciones` LONGTEXT,  
INDEX (`id_gama`),  
INDEX (`id_turno`),  
PRIMARY KEY (`id_workorder`),  
INDEX (`id_equipo`)  
) ENGINE=innodb DEFAULT CHARSET=utf8;
```

```
SET autocommit=1;
```

```
ALTER TABLE `recambios` ADD CONSTRAINT `{923861FF-A392-4140-85ED-  
2BE422F4EB1B}` FOREIGN KEY (`id_categoria` ) REFERENCES  
`categorias`(`id_categoria` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `operarios` ADD CONSTRAINT `categoria_profesionaloperarios`  
FOREIGN KEY (`id_categoria_profesional` ) REFERENCES  
`categoria_profesional`(`id_categoria_profesional` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `gamas_asociadas_equipos` ADD CONSTRAINT  
`equiposgamas_asociadas_equipos` FOREIGN KEY (`id_equipo` ) REFERENCES  
`equipos`(`id_equipo` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `gestor_conocimiento` ADD CONSTRAINT  
`equiposgestor_conocimiento` FOREIGN KEY (`id_equipo` ) REFERENCES  
`equipos`(`id_equipo` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `recambios_equipos` ADD CONSTRAINT `equiposrecambios_maquina`  
FOREIGN KEY (`id_equipo` ) REFERENCES `equipos`(`id_equipo` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `workorders` ADD CONSTRAINT `equiposworkorders` FOREIGN  
KEY (`id_equipo` ) REFERENCES `equipos`(`id_equipo` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `operarios` ADD CONSTRAINT `especialidadesoperarios` FOREIGN  
KEY (`id_especialidad` ) REFERENCES `especialidades`(`id_especialidad` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `gamas_asociadas_equipos` ADD CONSTRAINT  
`gamasgamas_asociadas_equipos` FOREIGN KEY (`id_gama` ) REFERENCES  
`gamas`(`id_gama` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `normas_en_gamas` ADD CONSTRAINT `gamasnormas_en_gamas`  
FOREIGN KEY (`id_gama` ) REFERENCES `gamas`(`id_gama` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `predictivo_equipos` ADD CONSTRAINT `gamaspredictivo_equipos`  
FOREIGN KEY (`id_gama` ) REFERENCES `gamas`(`id_gama` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `predictivo_lecturas` ADD CONSTRAINT `gamaspredictivo_lecturas`  
FOREIGN KEY (`id_gama`) REFERENCES `gamas`(`id_gama`);
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `recambios_en_gamas` ADD CONSTRAINT  
`gamasrecambios_en_gamas` FOREIGN KEY (`id_gama`) REFERENCES  
`gamas`(`id_gama`);
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `workorders` ADD CONSTRAINT `gamasworkorders` FOREIGN KEY  
(`id_gama`) REFERENCES `gamas`(`id_gama`);
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `equipos` ADD CONSTRAINT `lineasequipos` FOREIGN KEY  
(`id_linea`) REFERENCES `lineas`(`id_linea`);
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `normas_en_gamas` ADD CONSTRAINT `normasnormas_en_gamas`  
FOREIGN KEY (`id_norma`) REFERENCES `normas`(`id_norma`);
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `manoobra_workorder` ADD CONSTRAINT  
`operariosmanoobra_workorder` FOREIGN KEY (`id_operario`) REFERENCES  
`operarios`(`id_operario`);
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `workorders` ADD CONSTRAINT `operariosworkorders` FOREIGN  
KEY (`responsable_ejecucion`) REFERENCES `operarios`(`id_operario`);
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `equipos` ADD CONSTRAINT `Proveedoresequipos` FOREIGN KEY  
(`id_fabricante`) REFERENCES `Proveedores`(`id_proveedor`);
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `recambios` ADD CONSTRAINT `Proveedoresrecambios` FOREIGN  
KEY (`id_fabricante`) REFERENCES `Proveedores`(`id_proveedor`);
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `recambios` ADD CONSTRAINT `Proveedoresrecambios1` FOREIGN  
KEY (`id_fabricante`) REFERENCES `Proveedores`(`id_proveedor`);
```



ON UPDATE CASCADE ON DELETE NOT ACTION

```
ALTER TABLE `Proveedores` ADD CONSTRAINT `recambiosProveedores` FOREIGN
KEY (`id_proveedor` , `id_proveedor` ) REFERENCES `recambios`(`IdProveedor`
, `id_proveedor_alternativo` );
```

ON UPDATE CASCADE ON DELETE NOT ACTION

```
ALTER TABLE `recambios_en_gamas` ADD CONSTRAINT
`recambiosrecambios_en_gamas` FOREIGN KEY (`id_recambio` ) REFERENCES
`recambios`(`id_recambio` );
```

ON UPDATE CASCADE ON DELETE NOT ACTION

```
ALTER TABLE `recambios_equipos` ADD CONSTRAINT
`recambiosrecambios_maquina` FOREIGN KEY (`id_recambio` ) REFERENCES
`recambios`(`id_recambio` );
```

ON UPDATE CASCADE ON DELETE NOT ACTION

```
ALTER TABLE `proveedores` ADD CONSTRAINT `tipo_suministrosProveedores`
FOREIGN KEY (`id_tipo_suministro` ) REFERENCES
`tipo_suministros`(`id_tipo_suministro` );
```

ON UPDATE CASCADE ON DELETE NOT ACTION

```
ALTER TABLE `gestor_conocimiento` ADD CONSTRAINT
`tipos_incidenciasgestor_conocimiento` FOREIGN KEY (`id_tipo` ) REFERENCES
`tipos_incidencias`(`id_tipo_incidencia` );
```

ON UPDATE CASCADE ON DELETE NOT ACTION

```
ALTER TABLE `normas` ADD CONSTRAINT `tipos_incidenciasnormas` FOREIGN
KEY (`especialidad` ) REFERENCES `tipos_incidencias`(`id_tipo_incidencia` );
```

ON UPDATE CASCADE ON DELETE NOT ACTION

```
ALTER TABLE `workorders` ADD CONSTRAINT `tipos_incidenciasworkorders`
FOREIGN KEY (`id_tipo_incidencia` ) REFERENCES
`tipos_incidencias`(`id_tipo_incidencia` );
```

ON UPDATE CASCADE ON DELETE NOT ACTION

```
ALTER TABLE `workorders` ADD CONSTRAINT `turnosworkorders` FOREIGN KEY
(`id_turno` ) REFERENCES `turnos`(`id_turno` );
```

ON UPDATE CASCADE ON DELETE NOT ACTION

```
ALTER TABLE `gastos_workorder` ADD CONSTRAINT `workordersgastos_workorder`  
FOREIGN KEY (`id_workorder` ) REFERENCES `workorders`(`id_workorder` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

```
ALTER TABLE `manoobra_workorder` ADD CONSTRAINT  
`workordersmanoobra_workorder` FOREIGN KEY (`id_workorder` ) REFERENCES  
`workorders`(`id_workorder` );
```

```
ON UPDATE CASCADE ON DELETE NOT ACTION
```

## Anexo II. Manual de usuario del programa GMAO v1.0.

### Anexo II.1 - Menú principal.

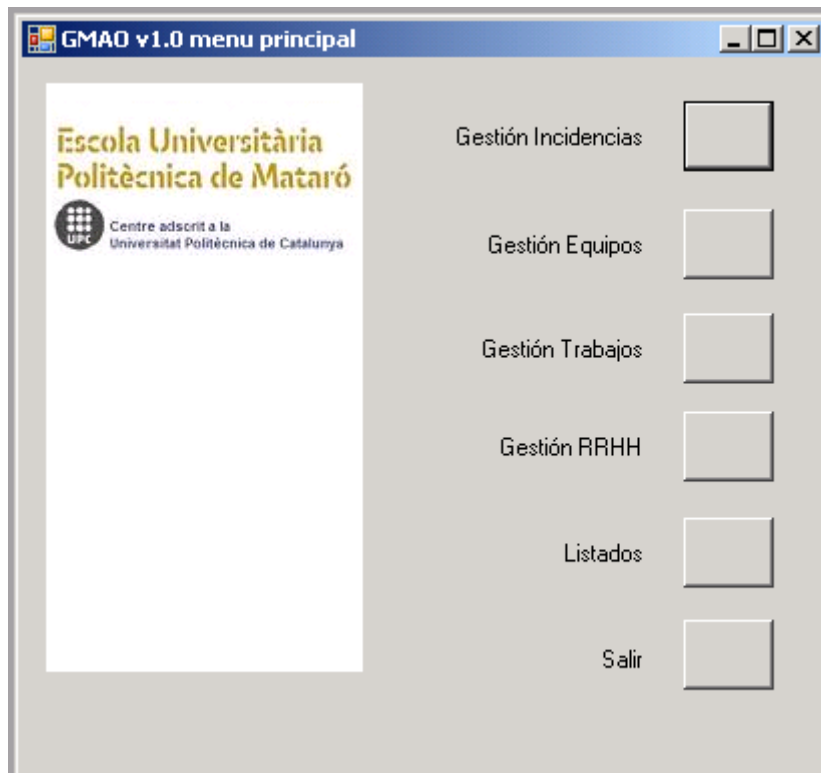


Fig. II.1. Vista menú principal.

El menú principal constituye el punto de acceso a la aplicación. En él el usuario puede elegir el área en la que va a trabajar.

- Gestión incidencias : para todo lo relacionado con el mantenimiento correctivo.
- Gestión equipos: altas , bajas de equipos y lo relativo a ellos.
- Gestión trabajos: en este apartado se confirmarán las tareas y se planificará su ejecución.
- Gestión RRHH: para acceder a lo relativo al personal y proveedores.
- Listados: permite seleccionar el listado elegido para su lanzamiento.
- Salir: finaliza la ejecución de la aplicación

## Anexo II.1 - Menú incidencias.

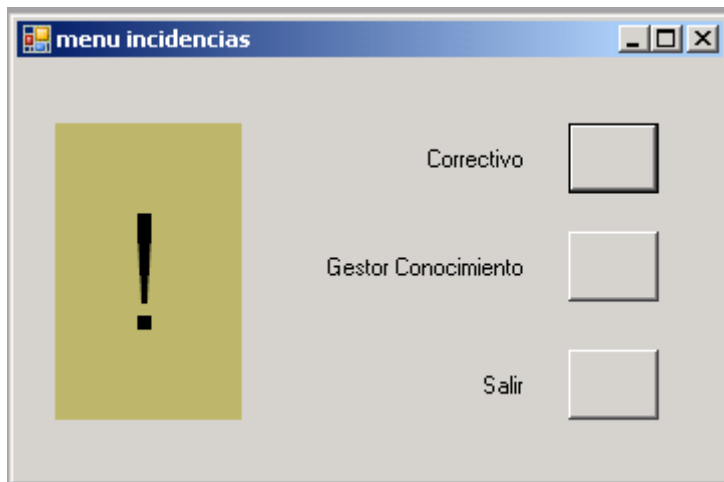


Fig. II.2. Vista menú Gestión de incidencias.

En este menú, el usuario dispone de tres opciones:

- **Correctivo:** Da acceso al formulario para generar intervenciones de mantenimiento correctivo.
- **Gestor conocimiento:** Da acceso al formulario de gestión de conocimiento, donde el usuario puede recopilar información útil sobre la resolución de incidencias, para ser reutilizada en el futuro.
- **Salir :** Cierra el menú y regresa al menu principal.

## Anexo II.1.2 – Formulario de Correctivo.

Este formulario se utiliza para generar directamente una orden de trabajo como reacción ante una incidencia.

The screenshot shows a software window titled "Form\_correctivo". At the top, there are two input fields: "gama/intervención correctivo" and "denominación correctivo". To the right, there is a dropdown menu labeled "Label9" with the value "CORRECTIVO" and a date field labeled "fecha" with the value "05/01/2013". Below these are several smaller input fields: "id\_equipo", "nombre equipo", "estado", "número de serie", and "fecha de adquisición". A button labeled "Equipo" is positioned between "nombre equipo" and "estado". Underneath, there is a section titled "norma/accion correctiva" which contains three sub-fields: "norma", "elemento", and "especialidad" (a dropdown menu). Below this is a large text area labeled "acción correctiva". At the bottom of the window, there are two buttons: "Quitar" and "Añadir acción", and a large grey rectangular area labeled "Label12". A "Generar W.O." button is located at the bottom right corner.

Fig. II.3. Vista Formulario Correctivo.

Los campos que debe rellenar el usuario son ( de izquierda a derecha y de arriba a abajo ):

- gama/intervención correctivo: nombre o acción que se va a realizar.
- denominación correctivo: El nombre que se le dará a la intervención.
- Equipo: Pulsando este botón el usuario accede al formulario de selección de equipos que le permite indicar sobre cuál de los equipos del parque se va a intervenir.
- Operacion/acción correctiva: El usuario indica la operación a realizar y la añade ( pulsando el botón añadir ) a la lista de operaciones que conformarán la orden de trabajo del correctivo a realizar, que es visible en la rejilla inferior.

Los datos a introducir para cada operación son:

- el nombre de la misma ( norma)
- el elemento del equipo sobre el que se actuará ( elemento )
- el tipo de intervención ( seleccionable de una lista )
- la descripción de la operación.

Repetir tantas veces como operaciones sean necesarias. En caso de querer suprimir alguna operación, seleccionar la operacion en la rejilla inferior y pulsar el botón quitar.

El botón generar w.o. (workorder) guardará los datos en las diferentes tablas correspondientes y generará la orden de trabajo para ser impresa.

### Anexo II.1.3 – Formulario Gestión del Conocimiento.

Da acceso al formulario de gestión de conocimiento, donde el usuario puede recopilar información útil sobre la resolución de incidencias, para ser reutilizada en el futuro como ayuda en la diagnosis de averías que vuelvan a producirse.

seleccione un equipo

id_equipo	nombre_equipo	estado	nserie	fecha_adquisicion
177	BAÑO CATAFORÉSICO	EF	4332354	01/06/2009
178	EXTRUSORA 100TM	EF	35435435	01/04/2011
179	SECADOR DE GRANZA Nº1	EF	98786552	01/06/2009
180	INYECTORA 211	EF	879887687	01/04/2010
189	PRENSA HIDRAULICA 3 EJES	EF	BB-98978	01/06/2009
▶ 190	AUTOMATA DE CONTROL	EF	666352	12/09/2011
193	ENROLLADORA TERMOESTATICA	EX	TX-2012-10-...	
*				

id\_equipo: 190    nombre equipo: AUTOMATA DE CONTROL    estado: EF    número de serie: 666352    fecha de adquisición: 12/09/2011

elemento:     fecha registro: 05/01/2013

problema:     palabra clave problema:

síntomas:     palabra clave sintoma:

solucion:

Notas:

adjuntar documentacion:  Adjuntar PDF

adjuntar imagen:  Adjuntar imagen Ver Guardar

En este formulario el usuario debe seleccionar un equipo de la lista visible en la rejilla superior y rellenar diferentes informaciones referentes a la incidencia de la que queremos guardar información.

- elemento: Se debe informar del elemento del equipo de que se trata.
- problema: Se debe indicar cuál es la incidencia que ocurre.
- Palabra clave problema: servira de referencia para las futuras búsquedas.
- Síntomas: Se debe informar si se aprecia algún síntoma del problema que ayude a su identificación.
- Palabra clave síntoma: servira de referencia para las futuras búsquedas.
- Solución: indicar la solución del problema
- Notas: permite describir con más detalle la solución al problema u otras informaciones que se consideren relevantes del mismo.
- Adjuntar PDF: En caso que de requerir de un informe más detallado, se puede adjuntar este en forma de fichero PDF. Para ello se abre un dialogo de selección del fichero.
- Adjuntar imagen: Si se desea documentar con una fotografía se puede adjuntar también. Para ello se abre un dialogo de selección del fichero.
- El botón guardar permitirá grabar los datos en la tabla correspondiente.

## Anexo II.2 - Menú Equipos.



Fig. II.3. Vista menú Equipos

En este menú, el usuario dispone de cinco opciones:

- Gestión Equipos: Para realizar las altas y bajas en el parque de equipos.
- Recambios: Para realizar las altas y bajas de los diferentes tipos de recambios.
- Líneas: En él se definen las líneas o agrupaciones funcionales a las que pertenecerán los equipos.
- Categorías: altas y bajas de los diferentes categorías de suministros a que pertenecerán los recambios.
- Volver: Cierra el menú y regresa al menu principal.



## Anexo II.2.1 – Formulario Equipos.

En este formulario el usuario puede dar de alta, modificar o dar de baja los diferentes quipos que componen el parque de la instalación. Representa la ficha del equipo.

The screenshot shows a web-based form titled "Equipos" with a navigation bar at the top indicating "1 de 15" items. The form contains several input fields and dropdown menus. The "código equipo" field contains "9999". The "nombre" field contains "ENSAMBLADO". The "tipo" dropdown is set to "U.S.". The "estado" dropdown is set to "EN FUNCIONAMIENTO". The "fecha cambio estado" dropdown is set to "22/12/2012". The "fabricante" dropdown is empty. The "nº de serie" field contains "11555". The "fecha adquisición" dropdown is set to "07/05/2009". There are two empty input fields for "horario trabajo equipo", "horas x día", and "días x semana". The "imagen" field contains a file path: "tttings\All Users\Documentos\Mis imágenes\Imágenes de muestra\Nenúfares.jpg". There are two buttons: "Adjuntar imagen" and "Ver imagen". The "observaciones" field contains the text "povarasgsdfgbsdfg".

Fig. II.4. Vista Formulario Equipos

Los campos que debe rellenar el usuario son ( de izquierda a derecha y de arriba a abajo ):

- Código equipo: código o referencia interna que le quiera asignar el usuario.
- Nombre: nombre del equipo.
- Tipo: información sobre el tipo de equipo que se trate según criterio del usuario.
- Estado: indica si el equipo esta en funcionamiento, en disposición de ser utilizado o ha sido dado de baja ( obsoleto ).

- Fecha cambio de estado.
- Fabricante: el nombre del fabricante del equipo, que debe estar previamente dado de alta como proveedor.
- N° de serie del equipo.
- Fecha de adquisición del equipo.
- Horario de trabajo del equipo: da información acerca del número de horas reales detrabajo efectivo del equipo y se utilizará para los cálculos de planificación .
- Adjuntar imagen: Si se desea documentar con una fotografía se puede adjuntar también. Para ello se abre un dialogo de selección del fichero.
- Observaciones: todas aquellas que se quieran añadir a la ficha del equipo.

## Anexo II.2.2 – Formulario Recambios.

En este formulario el usuario puede dar de alta, modificar o dar de baja los recambios que se utilizarán para el mantenimiento de los equipos.

The screenshot shows a software window titled "Gestión de Recambios". At the top, there are navigation icons and a status bar showing "11 de 23" items. The main form contains the following fields:

- recambio:** Text input with "ELECTROVALVULA".
- referencia:** Text input with "0355-3004-302-00".
- descripción:** Text input with "ELECTROVALVULA 24 VDC 12 W".
- Proveedores:** A section with three dropdown menus:
  - fabricante:** "CASIO"
  - proveedor:** "CASIO"
  - proveedor alternativo:** "TERCOMIENDO"
- Logistics and Pricing:** A group of input fields:
  - unidades en stock:** "0"
  - unidades pedidas:** "2"
  - precio unitario:** "125" €
  - nivel nuevo pedido:** "1"
  - plazo aprovisionamiento:** "5" días
- Observaciones:** A text area containing "Pedir la versión metálica."

Fig. II.5. Vista Formulario Recambios

Los campos que debe rellenar el usuario son ( de izquierda a derecha y de arriba a abajo ):

- Nombre del recambio: nombre con el que se lo conoce comunmente.
- Categoría: tipo de suministro al que pertenece el recambio.
- Referencia: la del fabricante ( o interna si procede).
- Descripción: características funcionales del recambio.
- Proveedores: nombre del fabricante del recambio, del suministrador principal y de uno alternativo, si lo hubiera.
- Unidades: cantidad que hay en el stock, las que están pedidas.
- Precio unitario del recambio.
- Nivel nuevo pedido
- Plazo aprovisionamiento: indica el tiempo en que un recambio tarda en ser servido.
- Observaciones: cualquier otra información que el usuario considere añadir a la ficha del recambio.

### Anexo II.2.3 – Formulario Líneas.

En este formulario el usuario puede dar de alta, modificar o dar de baja las diferentes agrupaciones de equipos ( líneas ) que sirven para vincular funcionalmente el equipo a su correspondiente línea. A nivel de GMAO sirve para listar información.

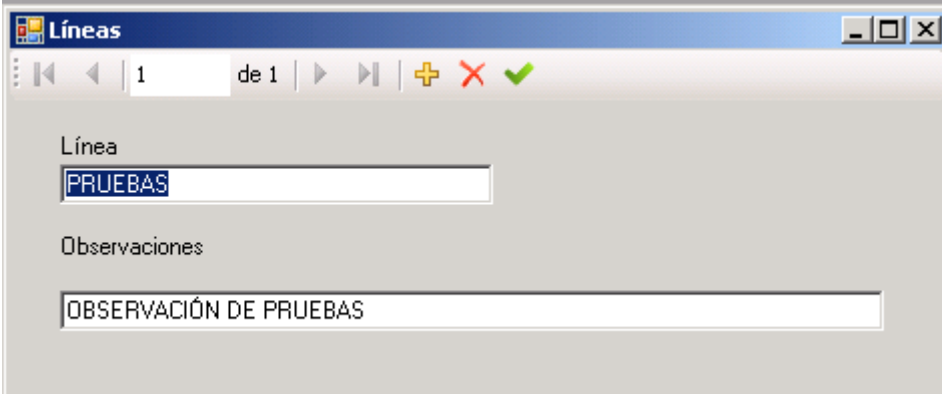


Fig. II.6. Vista Formulario Líneas

La línea es un conjunto de equipos destinados a realizar una tarea, que actúan como una unidad productiva. Los equipos se asignan a una línea, para poder listar la información referente al desempeño de la línea en su conjunto.

## Anexo II.2.4 – Formulario Categorías Recambios.

Permite añadir categorías de tipo de suministros.

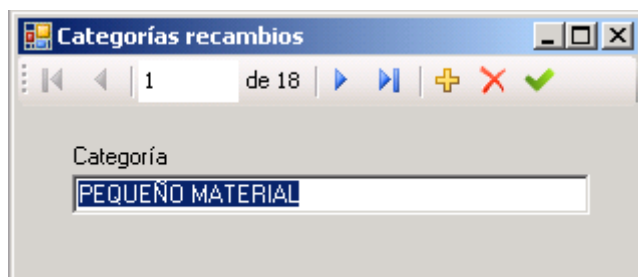


Fig. II.8. Vista Formulario Categorías

## Anexo II.3 - Menú Trabajos.

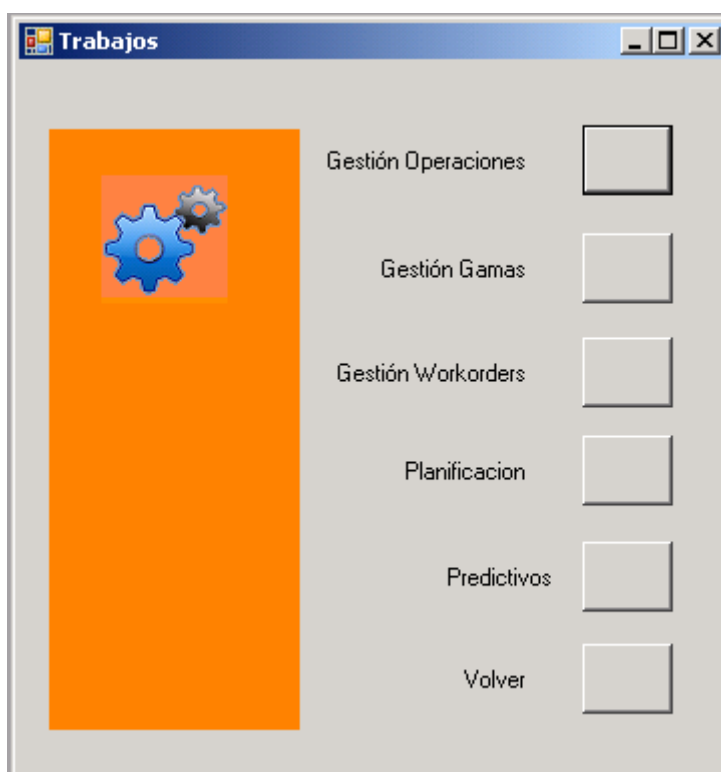


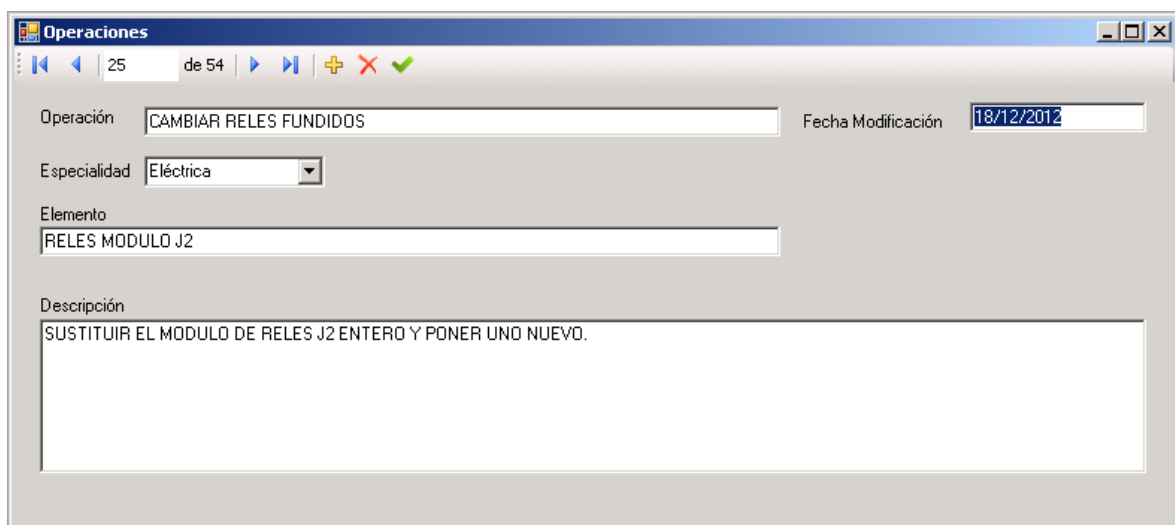
Fig. II.8. Vista Menú Trabajos

En este menú, el usuario dispone de seis opciones:

- **Gestión Operaciones:** Para realizar introducir nuevas operaciones que poder asignar en las intervenciones de mantenimiento ( Gamas ).
- **Gestion Gamas:** Permite la creación de nuevas fichas de intervención.
- **Gestión Workorders:** Permite trabajar con las órdenes de trabajo, bien sea para confirmar su programación, lanzarlas, o cerrarls una vez realizadas.
- **Planificación:** Lanza la planificación de las tareas periódicas de mantenimiento .
- **Predictivos:** da acceso al menu de mantenimiento predictivo.
- **Volver:** Cierra el menú y regresa al menu principal.

### Anexo II.3.1 – Formulario Operaciones.

Este formulario permite al usuario crear nuevas operaciones para posteriormente ser elegidas , al crear nuevas gamas. Se informan los datos básicos así como la descripción de la operación a realizar.



The screenshot shows a software window titled "Operaciones". At the top, there is a navigation bar with icons for back, forward, and search, along with a status bar showing "25 de 54" and three icons: a plus sign, a red X, and a green checkmark. The main form area contains the following fields:

- Operación:** CAMBIAR RELES FUNDIDOS
- Fecha Modificación:** 18/12/2012
- Especialidad:** Eléctrica (dropdown menu)
- Elemento:** RELES MODULO J2
- Descripción:** SUSTITUIR EL MODULO DE RELES J2 ENTERO Y PONER UNO NUEVO.

Fig. II.9. Vista Formulario Operaciones

Las operaciones son genéricas, pueden por tanto ser utilizadas en más de una gama, por ejemplo en aquellos casos en que exista más de una unidad del mismo equipo, o la operación designada en la ficha, sea aplicable a varios equipos. Es en la ficha de gamas donde las operaciones quedarán vinculadas a un único equipo.

Los campos que debe rellenar el usuario son ( de izquierda a derecha y de arriba a abajo ):

- Operación: nombre de la operación.
- Fecha modificación: fecha en que se modifica la ficha de operación. Se actualiza automáticamente al grabar los cambios.
- Especialidad: indica el tipo de perfil requerido para realizarla (eléctrica, mecánica, etc.)
- Elemento: sobre el que se realizará la operación.
- Descripción de la operación.

## Anexo II.3.2 – Formulario Gamas.

Este formulario permite al usuario crear nuevas gamas o intervenciones. La gama la constituye un conjunto de operaciones a realizar sobre un equipo de forma periódica, lo que constituye una acción de mantenimiento planificado.

The screenshot shows a software window titled "Gamas". At the top, there is a navigation bar with icons for back, forward, and search, along with the text "27 de 27". Below this, the form contains several input fields and buttons:

- Gama:** A text box containing "LECTURA MOTOR PRENSA".
- Clase Trabajo:** A text box containing "PREDICTIVO".
- Fecha Modificación:** A text box containing "01/01/2013".
- Denominación:** A text box containing "PREDICTIVO PRENSA1".
- Equipos asociados:** A button located to the right of the "Denominación" field.
- Recambios:** A button located below the "Equipos asociados" button.
- Observaciones:** A large empty text area for notes.
- Operaciones asociadas en gama actual:** A large empty greyed-out area for associated operations.
- Buttons at the bottom:** "Añadir operaciones(s) a la gama actual" on the left and "Quitar operacione(s) seleccionada(s)" on the right.

Fig. II.10. Vista Formulario Gamas

Los campos que debe rellenar el usuario son ( de izquierda a derecha y de arriba a abajo ):

- Gama: nombre de la gama.
- Clase de trabajo: informado directamente por el programa en función de si la gama corresponde a un mantenimiento correctivo, preventivo o predictivo .
- Fecha modificación: fecha en que se modifica la fecha de operación. Se actualiza automáticamente al grabar los cambios.
- Denominación: permite dar un nombre descriptivo a la gama.
- Observaciones que se quieran añadir a la ficha de la Gama.

En la rejilla inferior, mediante el botón de añadir operaciones, el usuario puede realizar la asignación de operaciones a la gama

Mediante el botón “equipos” el usuario puede ver los equipos que están asignados a la gama, en el formulario de equipos asignados.

id_equipo	codigo_equipo	nombre_equipo	estado	n_serie	horas_ciclos	ciclos_vs_horas_inicio	tiempo_estimado	horas_operarios	fecha_ultima_realizacion
10	9399	ENSAMBLADO	EN FUNCIONAMIENTO	11555	555	555	55	5	01/10/2012
131	A324	AGITADOR DE MEZCLA ISO+POLYOL	EF	555555	500	100	10	8	16/12/2012

Fig. II.11. Vista Formulario Equipos asociados

Si se desea añadir equipos a la gama se puede hacer mediante el botón “Añadir equipos” que da acceso al formulario de asignación de equipos.

id_equipo	nombre_equipo	estado	n_serie	fecha_adquisicion
10	ENSAMBLADO	EN FUNCIONAMIENTO	11555	07/05/2009
61	SOLDADURA TAPA COVER	EF	JBC123123	15/06/2008
84	INYECTORA DE 1200TM	OBSOLETO	55-88-156987	07/05/2009
119	ILUMINACION OFICINAS	EF	333-333-00	01/06/2009
124	PRENSA DE FORRADO 150Tm	EF	A542355	01/06/2009
126	CORTE AUTOMÁTICO DE CHAPAS	IF	JK-00-1929839	01/06/2009
131	AGITADOR DE MEZCLA ISO+POLYOL	EF	555555	01/06/2009
134	CAMBIADOR TAPAS VERTICAL	FS	6545321	07/05/2009
177	BAÑO CATAFORÉSICO	EF	4332354	01/06/2009
178	EXTRUSORA 100TM	EF	35435435	01/04/2011

Fig. II.12. Vista Formulario Añadir equipos

En este formulario, el usuario debe seleccionar un equipo de los listados en la rejilla superior y completar la información en referente a la gama (periodicidad, etc.) en los campos de texto situados en la parte inferior del formulario.

- Horas\_ciclo: indica cada cuánto se realizará la gama, en número de horas.



- Ciclo inicio: en los casos que se de de alta un equipo indica cuantas horas faltan para el primer lanzamiento de la gama. A partir de entonces se hara cada número de horas indicada en el campo horas ciclo.
- Tiempo estimado que tarda en realizarse la gama. Es el tiempo completo y debería considerar el tiempo necesario para dejar el equipo en condiciones de ser intervenido ( paro total, enfriado, etc. )
- Horas operarios: Es el tiempo de trabajo real que estarán el/los operario(s) trabajando .
- Fecha de asociación del equipo a la gama.
- Adjuntar documentación anexa. Se puede adjuntar un fichero Pdf para cada equipo asociado conteniendo información más detallada si hace falta.

Pulsando el botón “añadir equipo” se completará la operación de asociar el equipo a la gama.

Mediante el botón “recambios” el usuario puede ver en el formulario recambios en gamas, los recambios que a priori están previstos consumirse durante la realización de la gama.

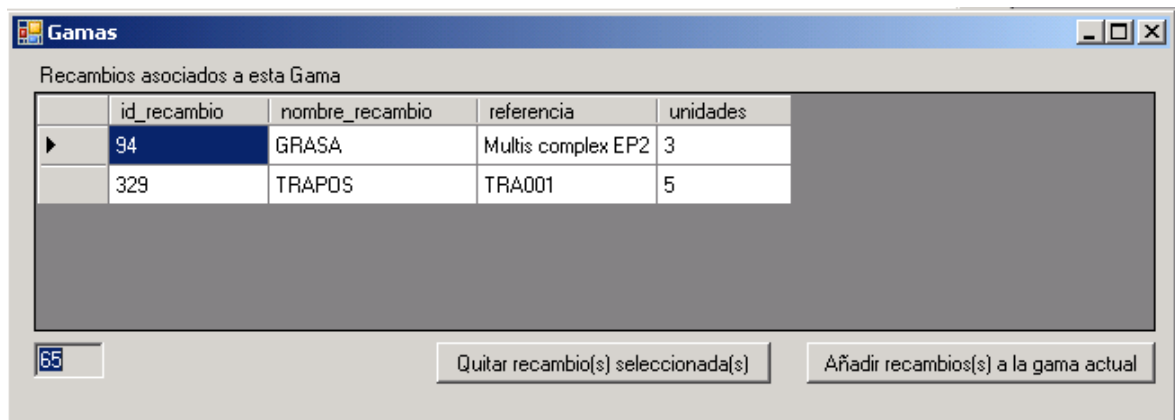


Fig. II.13. Vista Formulario Recambios en Gama

El usuario accede al formulario de asignación de recambios, pulsando el botón “Añadir recambio(s) a la gama actual” para indicar el tipo y número de recambios que consumirá la gama. Para ello seleccionará el recambio de la lista e informará del número de unidades a emplear.

The screenshot shows a window titled 'Form\_Añade\_Recambios'. It contains a table with two columns: 'nombre\_recambio' and 'referencia'. The first row is selected, showing 'ACEITES' and 'total carter ep-220'. Below the table, there is a button labeled 'Añadir recambio seleccionado' and a text field labeled 'unidades'.

nombre_recambio	referencia
ACEITES	total carter ep-220
ACEITE	hidráulico ISO H...
ACEITE	Total SAE 30
ACEITE	Rubia H SAE 30
ACEITE	Total Azolla ZS ...
ACEITE	AIL-5
ACEITE	Repsol aceite ter...
GRASA	BH 72-422
GRASA	Multis complex EP2
GRASA	Specis EC
ELECTROVALV...	0355-3004-302-00

Fig. II.14. Vista Formulario Añadir Recambios en Gama

### Anexo II.3.3 – Formulario Workorders.

Este formulario permite al usuario revisar las workorders creadas bien sea por una acción correctiva desde el formulario de correctivo o a través del lanzamiento planificado de gamas.

The screenshot shows a window titled 'Listar Workorders'. It contains a table with the following columns: 'id\_workorder', 'codigo\_workorder', 'descripcion\_workorder', 'Fecha creación', 'Fecha programación', 'id\_turno', and 'Solicitante'. The table lists 18 work orders. Below the table, there is a filter section with a dropdown menu for 'Filtrar Workorders por estado' and a button labeled 'Ver/modificar workorder seleccionada'.

id_workorder	codigo_workorder	descripcion_workorder	Fecha creación	Fecha programación	id_turno	Solicitante
2	12345	RELLENAR ACEITE TELLUS	11/06/2009	11/06/2009	10	ENC2
34	101	REPARACIONES	15/06/2009	11/06/2009	4	ENC11
84	213	ELIMINAR FUGAS DEL GRUPO HIDRAULICO	07/09/2009	11/06/2014	3	Responsab
85	256	PREVENT. MECÁNICO: ESTADO DE CIRCUITO NEUMÁTICO	02/09/2009	11/06/2009	6	Responsab
164	500032	PREVENT. MECÁNICO: ESTADO DE CIRCUITO NEUMÁTICO	01/05/2009	11/06/2009	10	pepe
165	500033	PREVENT. MECÁNICO: ESTADO DE CIRCUITO NEUMÁTICO	01/05/2009	11/06/2009	4	Responsab
166	500034	PREVENT. HIDRÁULICO: GRUPO HIDRÁULICO	01/05/2009	11/06/2009	1	Responsab
168	12567	LECTURA DE CICLOS	02/11/2009	02/11/2009	1	ENC1
174	115	SUSTITUCION DE LUMINARIAS DEFECTUOSAS	01/01/2010	12/10/2015	1	Produccion
178	128	REPARAR CARTELAS REFUERZO COLUMNA	10/11/2009	10/11/2009	2	Responsab
186	257	SE DESCONECTA LA MAQUINA DE REPENTE	17/03/2010	11/06/2009	2	ENC1
187	258	SE DESAJUSTA DETECTOR SE23	17/03/2010 7:32	17/03/2010	2	Produccion
188	259	ALARMA DE ALTA PRESION	17/03/2010	20/08/2010	2	Produccion
250	50002	REPARAR PULSADOR BOMBA	20/08/2010	20/08/2010	3	Responsab
253	50003	AJUSTAR PRESOSTATO MAX. A 100BAR	23/08/2010	04/05/2010	3	Produccion
257	50004	REVISAR CONSUMOS MOTOR PRPAL.	04/05/2010	04/05/2010	3	Operario MI

Fig. II.15. Vista Listar Workorders

El usuario puede acotar las workorders según el estado en que se encuentren

- **Pendientes:** Son las workorders generadas por el sistema. Este estado significa que están a la espera de que el usuario revise la orden de trabajo, la modifique, si es el caso ( asigne un afecha de ejecución concreta, asigne un responsable de su ejecución, etc. ). Una vez hecho esto es el propio usuario quien da la conformidad a la ejecución de la workorder pasando su estado a “lanzada”. Si no hay suficientes unidades en existencia de alguno de los recambios necesarios, el usuario es notificado.
- **Lanzadas:** Este estado significa que la workorder se ha impreso y entregado al responsable de su ejecución , quedando pendiente la finalización de la misma. Una vez realizados los trabajos correspondientes, el usuario podrá cerrar la orden. Al realizar este cambio de estado, quedan reservadas las unidades de los recambios que se van a emplear. Si la orden vuelve a estado pendiente quedan liberados esos recambios.
- **Cerradas:** cierra la orden de trabajo. El usuario completará con la información definitiva acerca de el personal que ha intervenido, los recursos empleados, etc.
- **Stand-by:** La orden se pone en este estado de forma automática, cuando se intenta cerrar la orden sin que haya suficientes recambios en el stock para haberla llevado a cabo y el usuario es notificado. También puede poner una orden en este estado el propio usuario, si decide aplazar su ejecución, por el motivo que fuese.

Para validar o modificar una orden de trabajo, el usuario entrara en el formulario de modificar workorders seleccionando una wo en la rejilla de datos y pulsando el botón “ ver/modificar wo seleccionada”.

Modifica Workorder

Workorder: 187 Estado LANZADA

Descripción  
SE VUELVE A AJUSTAR

Solicitante  
PRODUCCION PRODUCCION

Responsable de ejecución W.O.  
Operario generico Turno Tarde

Creación 17/03/2010 Programación 13/01/2013 Cierre 13/01/2013

Imputacion de gastos

Mano Obra €

Otros €

Recambios €

Recambios gama

Observaciones

Cierre W.O. Confirmar cambios

Fig. II.16. Vista Formulario Modificar Workorder

En este formulario el usuario puede definir varios datos de la workorder:

- Descripción de la tarea a realizar
- Solicitante. El usuario puede indicar quién está en el origen de la tarea
- Responsable de la ejecución. Es el nombre del operario al que se le adjudicará la responsabilidad de que la tarea sea ejecutada.
- Turno. En el cuál el equipo estará disponible para la tarea.
- Fecha de creación de la wo, bien sea por planificación ( preventivo y predictivo ) o directamente por a través del formulario de correctivo.
- Fecha de programación. Fecha en la que se debe ejecutar la tarea.

- Fecha de cierre. Fecha en la que finalmente se ha realizado la intervención y se procede a cerrar la orden.
- Gastos. Mediante el correspondiente botón podemos imputar los gastos correspondientes a los concepto de :
  - Mano de Obra.

id_mano_obra_wor	mano_obra_workor	id_workorder	hora_inicio	hora_fin	total_horas	coste_hora_imputa	operarios.id_operar	cod_operario
53	10	187	30/12/1899 15:00	30/12/1899 15:30	30/12/1899 0:30	50	10	101
61	10	187	12/01/2013 18:03	12/01/2013 19:03	30/12/1899 1:00	50	10	101
*								

Fig. II.17. Vista Formulario Coste de mano de obra

En este formulario se van añadiendo las diferentes imputaciones de los trabajos realizados por los distintos operarios que han intervenido en la realización de la workorder.

- Operario: Persona de la que se realiza la imputación.
- Hora Inicio / Hora fin. Fecha y hora del inicio y conclusión del trabajo realizado por esa persona.
- Horas: al seleccionar el tipo de hora se imputa con el coste correspondiente, según sea normal, extra o festiva.

#### Otros Gastos

De igual manera podemos imputar los gastos de otra naturaleza ( alquiler de herramientas, personal externo, etc. ) que se hayan podido generar durante la realización de este trabajo.

Concepto: SACO DE CEMENTO 187

Gasto: 63 Fecha: jueves, 20 de diciembre de 2012 20/12/2012

	id_workorder_gasto	gasto	concepto	id_workorder	fecha
▶	10	150	ALQUILER DE F...	187	20/12/2012
	11	63	SACO DE CEME...	187	20/12/2012
*			SACO DE CEMENTO		

Borrar gasto seleccionado Añadir gasto introducido

Fig. II.7. Vista Formulario Costes Otros

### Recambios

Además el programa informa de forma automática el campo donde en el cuál se almacena el coste de los recambios asociados a la ejecución de la orden de trabajo.

Por último, regresando al formulario de modificar la workorder, podemos añadir las observaciones que consideremos pertinentes.

### Anexo II.3.4 – Formulario Lanzar planificación.

En formulario tiene como objeto desencadenar el cálculo de todas las gamas planificadas entre las fechas deseadas por el usuario.

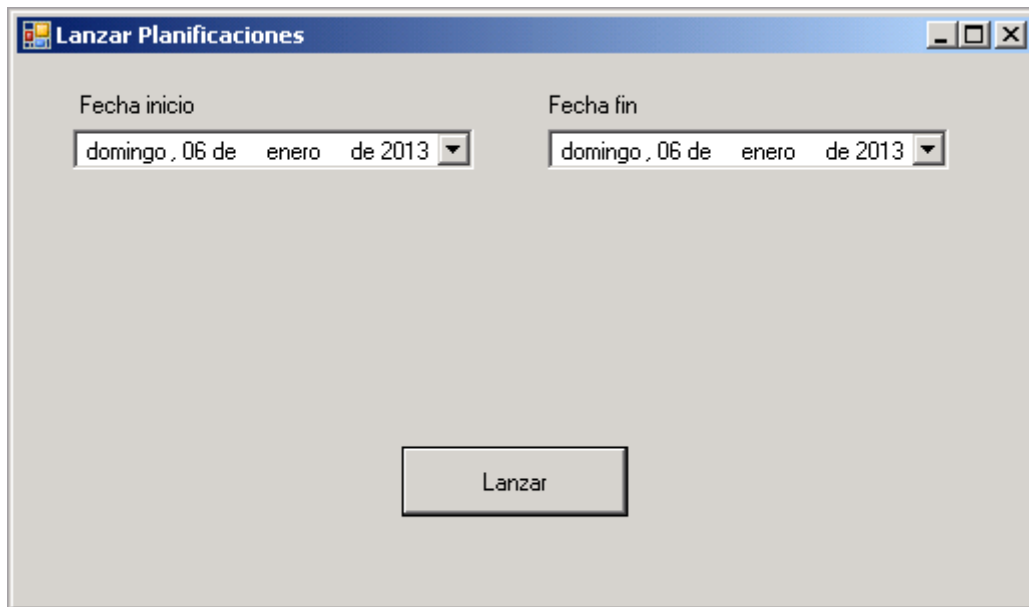
The image shows a software window titled "Lanzar Planificaciones". Inside the window, there are two date selection fields. The first field is labeled "Fecha inicio" and the second is labeled "Fecha fin". Both fields display the date "domingo, 06 de enero de 2013". Below these two fields, centered, is a button labeled "Lanzar".

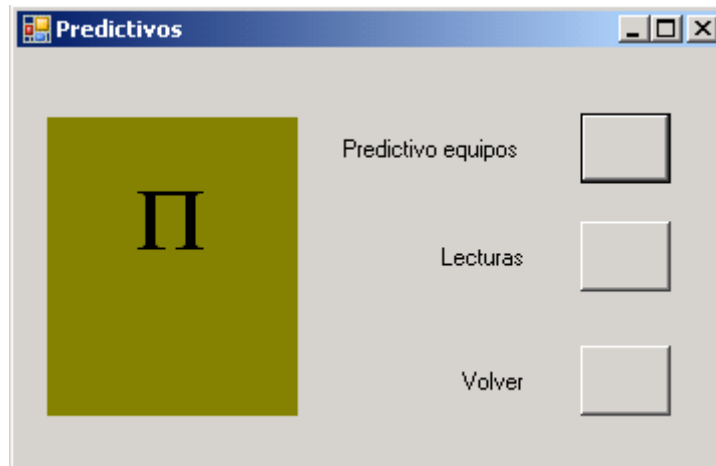
Fig. II.8. Vista Formulario Lanzar planificación.

El usuario debe seleccionar entre que dos fechas quiere desencadenar la preparación de las órdenes de trabajo ( wo) correspondientes a aquellas gamas vencidas y el programa generará de forma automática las que por fecha deban realizarse.

### Anexo II.3.5 – Formulario Listados.

Se trata de un acceso al mismo formulario de listado que desde el menú principal, colocado en este menú para comodidad del usuario para que después de lanzar workoders, pueda mandar a imprimirlas.

## Anexo II.3.6 – Menú Predictivos.



En este menú, el usuario dispone de tres opciones:

- Predictivo equipos: Para realizar las definiciones y modificaciones de las lecturas de mantenimiento predictivo en un equipos, que se lanzarán periódicamente como una gama más.
- Lecturas. Donde quedará registrado el valor obtenido en la lectura .
- Volver: Cierra el menú y regresa al menu principal.

### Anexo II.3.6.1 – Formulario Predictivo equipos.

En este formulario el usuario puede definir una nueva tarea de lectura de una variable que se quiera controlar periódicamente en un equipo. Una vez rellenada la ficha/formulario se generará automáticamente una operación y una gama de lectura con la periodicidad indicada que se lanzará cuándo corresponda según la planificación.



Los campos que debe rellenar el usuario son ( de izquierda a derecha y de arriba a abajo ):

- Id\_equipo/nombre equipo/estado/número de serie/fecha de adquisición: pulsando el botón “Equipo” el usuario accede a la pantalla de selección de equipos y elige aquel sobre el que quiere crear una ficha de lectura.
- Variable a medir: nombre del parámetro o variable a medir en el equipo.
- Valor objetivo: el valor que debería tener la variable a medir si el equipo funciona correctamente.
- Unidades: en que se expresa la variable. Dada la variedad de unidades que pueden emplearse este campo lo introduce el propio usuario.
- Límite superior /inferior : valores márgen a partir de los cuáles la desviación en el valor de la variable medida indica una anomalía.
- Gama/lecturapredictivo: nombre que tendrá la gama correspondiente a la lectura de esta variable sobre este equipo.
- Denominación del predictivo: nombre descriptivo.
- Operacion: idéntico nombre que la gama.
- Elemento: punto del equipo en el que se realizará la medida.

- Especialidad.
- Acción lectura: descripción en detalle de la lectura.

Finalmente pulsando el botón “guardar” se guardarán los cambios y se generarán la operación y gama correspondiente.

Una vez creada un ficha de predictivo, está no podrá ser modificada posteriormente, sino que en caso de querer variar los valores de la lectura, habrá que crear una nueva, pudiéndose eliminar la existente , mediante el borrado de su gama correspondiente ( en el formulario de gamas ) y quedando los valores de la lecturas sin borrar como registro histórico.

### Anexo II.3.6.1 – Formulario Lecturas.

En este formulario se introducen las lecturas correspondientes al conjunto equipo/variable deseado.

The screenshot shows a software window titled "Lecturas Predictivo". At the top, there are five input fields: "id\_equipo", "nombre equipo", "estado", "número de serie", and "fecha de adquisición". A button labeled "Equipo" is located between the "nombre equipo" and "estado" fields. Below these fields is a large, empty rectangular area. At the bottom of the window, there are several more fields: "Variable a medir", "Valor objetivo", "Unidades", "Lím. superior", and "Lím. inferior". To the right of these fields is a button labeled "Ver". Below these fields, there are "Valores de lectura" (with a dropdown menu showing "06/01/2013") and "Valor lectura" fields. At the bottom right corner, there is a button labeled "Guardar".

Los campos que debe rellenar el usuario son ( de izquierda a derecha y de arriba a abajo ):

- Id\_equipo/nombre equipo/estado/número de serie/fecha de adquisición: pulsando el botón “Equipo” el susario accede a la pantalla de selección de equipos y elige aquel sobre el que quiere registrar una lectura.

El usuario verá todas las fichas de lectura de variables disponibles para ese equipo seleccionado en la rejilla de datos de debajo. Seleccionando una fila se indica cuál es la medida que se quiere registrar, y sus datos principales pasan al conjunto de “textbox” inferior.

- Variable a medir: nombre del parámetro o variable a medir en el equipo.
- Valor objetivo: el valor que debería tener la variable a medir si el equipo funciona correctamente.
- Unidades: en que se expresa la variable. Dada la variedad de unidades que pueden emplearse este campo lo introduce el propio usuario.
- Límite superior /inferior : valores márgen a partir de los cuáles la desviación en el valor de la variable medida indica una anomalía.
- Valor lectura : Valor físico registrado en la lectura. Es el valor que introduce el usuario propiamente

#### Anexo II.4 - Menú Recursos Humanos.

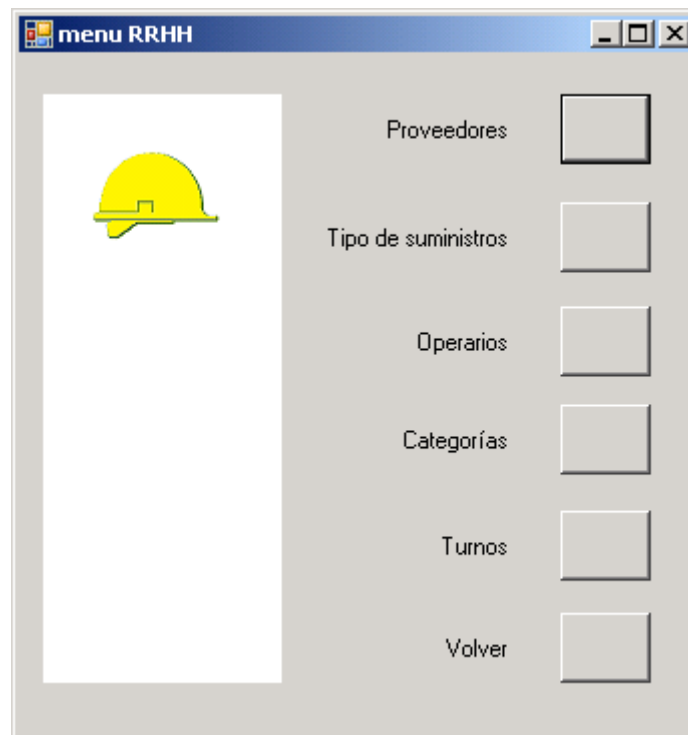


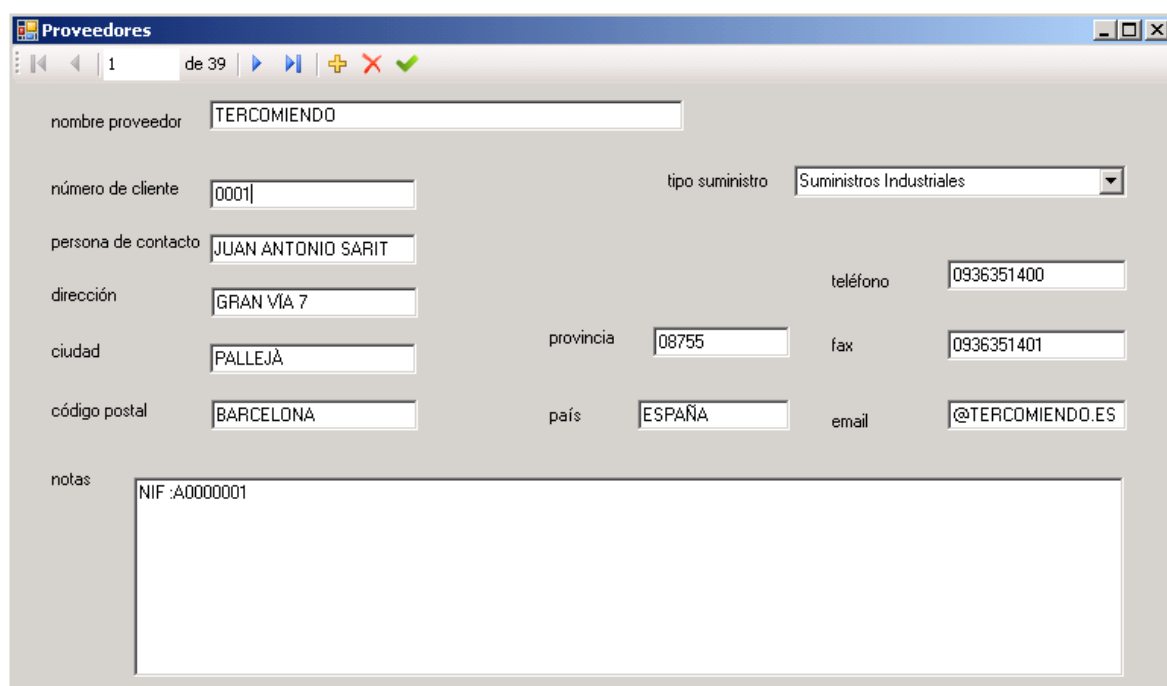
Fig. II.9. Vista Menú Recursos Humanos.

En este menú, el usuario dispone de seis opciones:

- Proveedores: Para realizar altas, bajas y modificaciones de los proveedores de equipos, recambios y suministros diversos.
- Tipo de suministros: Permite añadir nuevos tipos de suministros.
- Operarios: Permite dar altas, bajas y modificaciones de los datos del personal de mantenimiento.
- Categorías profesionales de los operarios.
- Turnos: los diversos turnos de trabajo que puedan existir.
- Volver: Cierra el menú y regresa al menu principal.

### Anexo II.4.1 – Formulario Proveedores.

En este formulario el usuario podrá realizar las altas, bajas y modificaciones de los datos de los proveedores ( fabricantes o proveedores ) de los equipos, recambios y suministros diversos.



The screenshot shows a window titled "Proveedores" with a toolbar at the top containing navigation and action icons. The form fields are as follows:

nombre proveedor	TERCOMIENDO	tipo suministro	Suministros Industriales
número de cliente	0001	teléfono	0936351400
persona de contacto	JUAN ANTONIO SARIT	provincia	08755
dirección	GRAN VÍA 7	fax	0936351401
ciudad	PALLEJÀ	país	ESPAÑA
código postal	BARCELONA	email	@TERCOMIENDO.ES
notas	NIF :A0000001		

Fig. II.10 Vista Formulario proveedores.

Los campos que debe rellenar el usuario son ( de izquierda a derecha y de arriba a abajo ):

- Nombre proveedor.
- Número de cliente: es el número que asignado por el proveedor al usuario.
- Tipo de suministro: indica que tipo de suministro o servicio que provee.
- Persona de contacto.
- Dirección.
- Teléfono.
- Ciudad
- Provincia
- Fax ( número de .)
- Código postal
- País
- Email.
- Notas: cualquier dato extra que se quiera incorporar a la ficha del proveedor.

#### Anexo II.4.2 – Formulario Tipo de suministros.

En el se informa al sistema de los diferentes tipos de servicios o suministros que pueden ser provistos según se vayan incorporando nuevos proveedores.

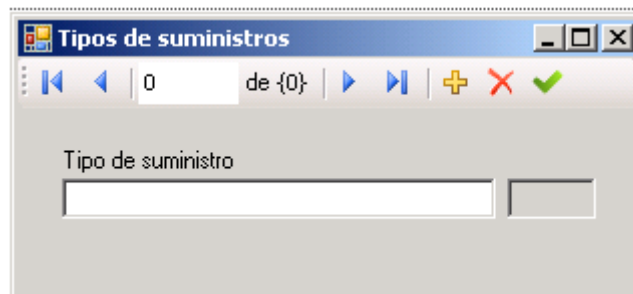


Fig. II.11. Vista Formulario tipo de suministros.

#### Anexo II.4.3 – Formulario Operarios.

Permite dar altas, bajas y modificaciones de los datos del personal de mantenimiento.

The image shows a software window titled "Operarios" with a standard Windows-style title bar. Below the title bar is a navigation bar with a back arrow, a page indicator "1 de 4", and icons for adding, deleting, and saving records. The main area contains a form with the following fields:

cód. operario	101	nombre	MENDIL CUTILLAS, JULIAN L.
especialidad	Mecánico	empresa	QUIMICON S.A.
categoría	Oficial de 2º	turno	30
teléfono contacto	96-1267893	coste/hora	
código postal	49034	dirección	C/PALITO 45, 1ªA

currículum

1986: FP 2º GRADO, 1989:FORD, 1992: NANTA

Fig. II.12. Vista Formulario Operarios.

Los campos que debe rellenar el usuario son ( de izquierda a derecha y de arriba a abajo ):

- Código de operario: Se trata de un código asignado libremente por el usuario para permitir su identificación. Por ejemplo, si se trata de una subcontrata podría ser el código de empleado.
- Nombre del Operario
- Especialidad. Conocimientos profesionales/formación del operario ( eléctrico, mecánico )
- Empresa.
- Categoría profesional del operario
- Turno de trabajo
- Coste/ hora del operario
- Teléfono de contacto
- Dirección y código postal
- Currículum. Breve descripción de su experiencia profesional.

### Anexo II.4.4 – Formulario Categorías profesionales.

En el se informa al sistema de los diferente tipos de categorías profesionales que existen.

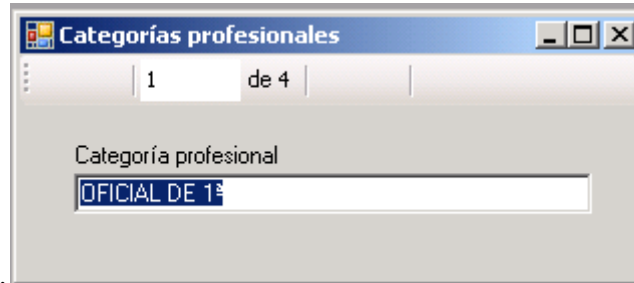
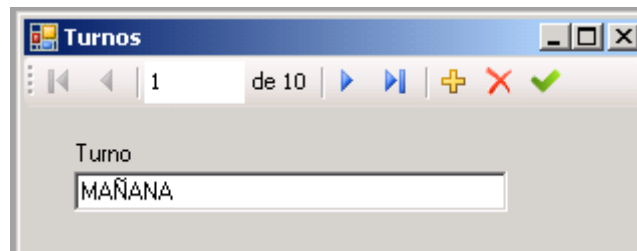


Fig. II.13. Vista Formulario Categorías profesionales.

### Anexo II.4.5 – Formulario Turnos.

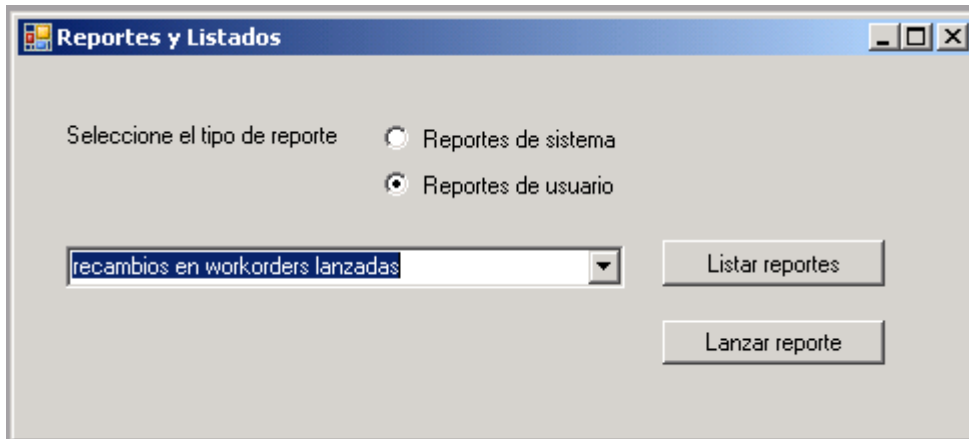
En el se informa al sistema de los diferentes turnos con que se trabaja.



### Anexo II.5 – Formulario Listados.

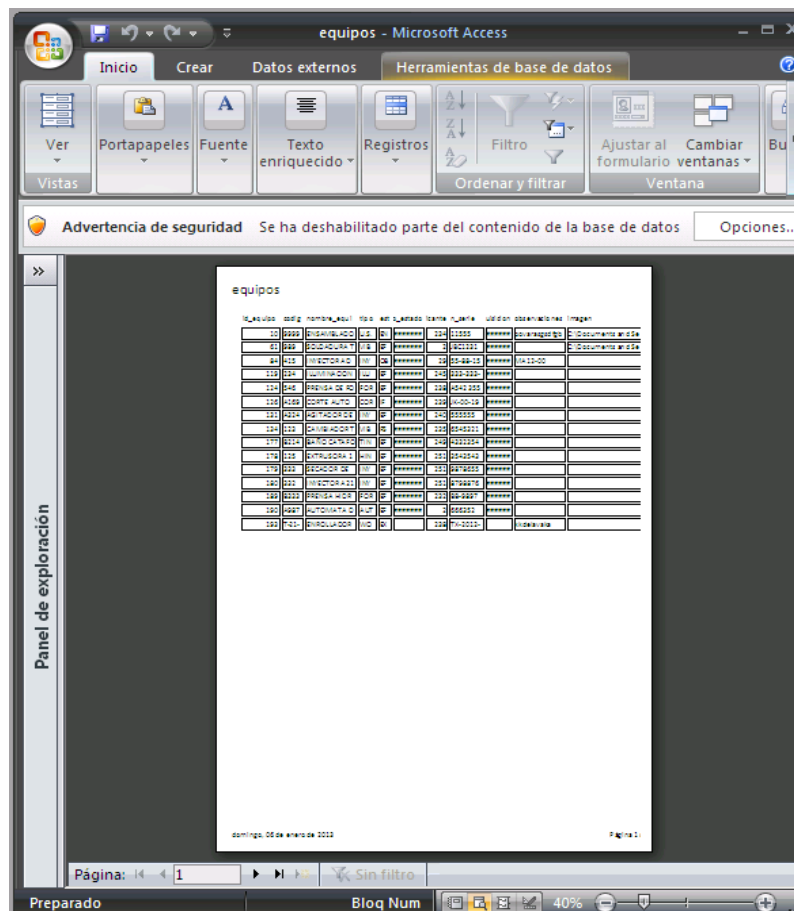
En este formulario el usuario puede seleccionar y lanzar cualquiera de los reportes que están previamente creados en la BD Access.

El usuario puede lanzar los reportes que están predefinidos con la aplicación ( y que se identifican con el prefijo GMAO-nombre del reporte ), así como también los propios que el usuario cree según su conveniencia y que serán listados por la aplicación.



El usuario selecciona mediante el “radiobutton” el tipo de reporte que quiere lanzar y pulsa el botón “listar”, para que se listen todos los reportes de ese tipo en el combo box.

El usuario selecciona el reporte deseado y pulsa el botón lanzar. El reporte se abrirá entonces en modo vista preliminar, dentro de la propia aplicación access, para que el usuario pueda visualizarlo y/o imprimirlo.





## **Anexo III. Contenido del CD-ROM.**

- Documentación del proyecto (memoria, código fuente y estudio económico).
- Base de datos del programa.
- Archivos fuentes y ejecutables del programa GMAO v1.0
- Notas de instalación.