

UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

**ESCUELA DE INGENIERÍA DE
TELECOMUNICACIÓN Y ELECTRÓNICA**



PROYECTO FIN DE CARRERA

**OPTIMIZACIÓN SOFTWARE PARA PROCESAMIENTO
ESTADÍSTICO DE LOS TELEGRAMAS DE
COMUNICACIÓN GENERADOS POR LAS INSTALACIONES
KNX PARA SU PUBLICACIÓN COMO SOFTWARE LIBRE Y
APLICACIÓN MÓVIL**

Titulación: *Ingeniero Técnico de Telecomunicación
Especialidad en Sistemas de
Telecomunicación*

Tutores: *Dr. Domingo Benítez Díaz
Dr. Carlos M. Travieso González*

Autor: *Sergio Martín Medina*

Fecha: *Julio 2015*

UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

**ESCUELA DE INGENIERÍA DE
TELECOMUNICACIÓN Y ELECTRÓNICA**



PROYECTO FIN DE CARRERA

**OPTIMIZACIÓN SOFTWARE PARA PROCESAMIENTO
ESTADÍSTICO DE LOS TELEGRAMAS DE
COMUNICACIÓN GENERADOS POR LAS INSTALACIONES
KNX PARA SU PUBLICACIÓN COMO SOFTWARE LIBRE Y
APLICACIÓN MÓVIL**

Presidente:

Secretario:

Vocal:

Tutor/a:

Tutor/a:

Fdo.: Domingo Benítez Díaz

Fdo.: Carlos M. Travieso González

Autor:

Fdo.: Sergio Martín Medina

NOTA : _____

Titulación: *Ingeniero Técnico de Telecomunicación Especialidad en
Sistemas de Telecomunicación*

Tutores: *Dr. Domingo Benítez Díaz
Dr. Carlos Travieso González*

Autor: *Sergio Martín Medina*

Fecha: *Julio 2015*

A mi familia.
Y a ti, por estar siempre ahí.

AGRADECIMIENTOS

Tras muchos años de esfuerzos y batallas, sólo queda agradecer a todos aquellos que han estado ahí en todo momento, desde el principio hasta el final, sin importarles los altibajos por los que esta etapa de mi vida ha pasado.

A mis padres y mi abuelo, que me han inculcado una educación envidiable. Que siempre han mostrado su apoyo y nunca me han permitido doblegarme a los problemas y las hostilidades.

A mi hermana, que siempre me ha guiado por el camino más correcto, basándose en su amplia experiencia, y que ha sido siempre un ejemplo a seguir para mi persona.

A Tere, porque me ha dado el aliento necesario para seguir el camino y me ha ayudado a superar los obstáculos siempre de la mano.

A mis compañeros, con los cuales he vivido miles de experiencias inolvidables y cuya amistad se que perdurará en el tiempo.

A Domingo Benítez y Carlos Travieso, tutores de este proyecto, con los que he podido contar en todo momento, aprovechándome de sus infinitos conocimientos, de su cercanía y de su cordialidad.

A todos, GRACIAS.

Índice de contenidos

Índice de contenidos.....	I
Índice de figuras.....	V
Índice de tablas.....	IX
Acrónimos.....	XI
Capítulo 1: Introducción y antecedentes	1
1.1 Introducción.....	1
1.2 Antecedentes.....	1
1.3 Objetivos del PFC	2
1.4 Peticiónario.....	4
1.5 Estructura del documento	4
1.5.1 Memoria	4
1.5.2 Pliego de condiciones	5
1.5.3 Presupuesto.....	5
1.5.4 CD-ROM	6
Capítulo 2: Análisis previo. El software SKoA	7
2.1 Introducción.....	7
2.1.1 Aplicaciones SCADA	8
2.1.2 Aplicación SKoA 0.2	10
2.2 Análisis Previo del Código Fuente de SKoA 0.2.....	12
2.2.1 Organización del código	12
2.2.2 Redundancia en las validaciones	13
2.2.3 Dependencias innecesarias	13
2.2.4 Implementación errónea de la internalización.....	13

2.2.5	Proyectos no acoplados	14
2.2.6	Iconos e imágenes duplicadas en la estructura	14
2.2.7	Resumen del capítulo	14
Capítulo 3: Optimización del software		17
3.1	Introducción.....	17
3.2	Tareas de optimización.....	17
3.2.1	Librería "tuwien.auto.*"	17
3.2.2	Optimización de las librerías.....	21
3.2.3	Reorganización de clases	23
3.2.4	Código comentado.....	24
3.2.5	Errores de comparación de ristas.....	25
3.2.6	Optimización de búsqueda dentro de los nodos en el archivo XML.....	26
3.2.7	Reestructuración de paquetes.....	27
3.2.8	Implementación correcta de la internacionalización	31
3.2.9	Integración de proyectos independientes	34
3.2.10	Iconos modificados	35
3.2.11	Entorno gráfico mejorado.....	43
3.3	Resumen del capítulo	45
Capítulo 4: Publicación de SKoA 1.0 como software libre		47
4.1	Introducción.....	47
4.2	Consideraciones iniciales	47
4.3	Ventajas e inconvenientes del software libre	48
4.4	Tipos de licencia de software libre	49
4.5	Decisión final	51
4.6	Licencia Pública General de GNU en su versión 3 (GPLv3)	51
4.6.1	Fundamentos de la GPLv3	52

4.6.2	Neutralización de las leyes que prohíben el software libre	53
4.6.3	Protección del derecho de modificación	54
4.6.4	Protección fuerte contra amenazas de patentes.....	54
4.6.5	Compatibilidad con otras licencias libres	55
3.2.6	Una licencia de carácter global	56
4.7	Términos y Condiciones	57
4.8	Texto de Licencia de GPLv3	73
4.9	Resumen del capítulo	76
Capítulo 5: Aplicación móvil Proyecto PILEV.....		77
5.1	Introducción.....	77
5.2	Antecedentes.....	77
5.3	Proyecto PILEV	80
5.4	Diseño de la aplicación móvil.....	83
5.4.1	Tipos de aplicaciones	84
5.4.2	PhoneGap	87
5.4.3	Desarrollo y diseño de la aplicación	88
5.4.4	Vista final de la aplicación.....	90
5.4.5	Resumen del capítulo	90
Capítulo 6: Conclusiones.....		91
6.1	Introducción.....	91
6.2	Conclusiones	91
Bibliografía.....		95
Anexo I: Manual de Usuario SKoA 1.0.....		102
Anexo II: Código fuente PILVE APP.....		164
Pliego de condiciones		181
PL.1	Introducción.....	181

PL.2 Condiciones hardware	181
PL.3 Condiciones software	181
<i>Presupuesto</i>	185
P.1 Trabajo tarifado por tiempo empleado	185
P.2 Amortización del inmovilizado material.....	186
P.2.1 Amortización del material hardware.....	187
P.2.2 Amortización del material software.....	187
P.3 Redacción del proyecto.....	188
P.4 Presupuesto final.....	189

Índice de figuras

Figura 2.1.- Esquema de un sistema SCADA típico.....	9
Figura 2.2.- Módulo de control y monitorización	10
Figura 2.3.-Módulo de configuración SKoA 0.2.....	11
Figura 2.4.- Estructura de paquetes y librerías del código fuente.....	12
Figura 3.1.- Posición de la librería “twien.auto” antes de su optimización.....	20
Figura 3.2.- Optimización final de la librería “twien”.....	21
Figura 3.3.- Librerías duplicadas en la versión antigua.....	22
Figura 3.4.- Clases sin CamelCase.	24
Figura 3.5.- Clase modificadas con UpperCamelCase.	24
Figura 3.6.- Comparativa nomenclatura orientativa	24
Figura 3.7.- Diagrama MVC.....	27
Figura 3.8.- Típica colaboración entre componentes de un MVC.....	28
Figura 3.9.- Estructura de los paquetes tras la optimización.	30
Figura 3.10.- Proyectos con paquetes duplicados para distintos idiomas.	32
Figura 3.11.- Paquetes skoa.view.resources.	33
Figura 3.12.- Archivos About.properties y About_en.properties.....	34
Figura 3.13.- Menú versión antigua SKoA.....	35
Figura 3.14.- Actuadores versión antigua SKoA.	36
Figura 3.15.- Interfaz principal antigua.....	44
Figura 3.16.- Interfaz Principal Nueva.....	44
Figura 4.1. Gráfico que ilustra las compatibilidades de GPLv3 con otras licencias.	55
Figura 5.1.- Esquema de las instalaciones utilizadas en el proyecto PIVE.....	78
Figura 5.2.- Esquema de las instalaciones utilizadas en el proyecto PIVE.....	79
Figura 5.3.- Archivo .txt generado por el servidor del Proyecto PIVE.....	81

<i>Figura 5.4.- Datos Totales de los hoteles en el día de HOY.</i>	81
<i>Figura 5.5.- Datos Hotel Baobab en el día de HOY.</i>	82
<i>Figura 5.6.- Datos Hotel Baobab en el día de AYER.</i>	82
<i>Figura 5.7.- Esquema de desarrollo de PhoneGap</i>	87
<i>Figura 5.8.- Vista final de la aplicación</i>	90
<i>Figura A.1.- Interfaz de selección de idioma</i>	112
<i>Figura A.2.- Interfaz principal.</i>	113
<i>Figura A.3.- Panel superior de opciones</i>	113
<i>Figura A.4.- Secciones panel superior</i>	114
<i>Figura A.5.- Interfaz zona central.</i>	115
<i>Figura A.6.- Ejemplo de proyecto con plantas y estancias</i>	115
<i>Figura A.7.- Métodos para conectar Bus EIB.</i>	117
<i>Figura A.8.- Cuadro Conexión BUS EIB</i>	117
<i>Figura A.9.- Métodos para conectar Bus EIB.</i>	118
<i>Figura A.10.- Pestaña nuevo proyecto</i>	119
<i>Figura A.11.- Ventana nuevo proyecto</i>	120
<i>Figura A.12.- Módulo de configuración.</i>	121
<i>Figura A.13.- Apartado añadir “Nueva Planta” del Módulo de Configuración</i>	122
<i>Figura A.14.- Módulo “Añadir Planta”</i>	122
<i>Figura A.15.- Ejemplo de plantas creadas.</i>	123
<i>Figura A.16.- Apartado “Eliminar Planta” del Módulo de Configuración.</i>	123
<i>Figura A.17.- Desplegable con las plantas creadas en el proyecto</i>	124
<i>Figura A.18.- Mensaje de advertencia mostrado al eliminar una planta.</i>	124
<i>Figura A.19.- Apartado “Añadir Nueva Estancia” del Módulo de Configuración</i>	125
<i>Figura A.20.- Método para añadir una nueva estancia en una planta existente.</i>	125
<i>Figura A.21.- Visualización de la nueva estancia añadida</i>	126
<i>Figura A.22.- Apartado “Eliminar Estancia” del Módulo de Configuración.</i>	126

<i>Figura A.23.- Método para eliminar una estancia existente.....</i>	<i>127</i>
<i>Figura A.24.- Mensaje de advertencia mostrado al eliminar una estancia.....</i>	<i>127</i>
<i>Figura A.25.- Apartado “Añadir Actuadores” del Módulo de Configuración</i>	<i>128</i>
<i>Figura A.26.- Interfaz del Módulo de Configuración de Actuadores/Sensores.....</i>	<i>129</i>
<i>Figura A.27.- Ventana de configuración de parámetros de actuadores/sensores</i>	<i>130</i>
<i>Figura A.28.- Actuadores/Sensores colocados sobre el plano de la estancia.....</i>	<i>131</i>
<i>Figura A.29.- Botones “Guardar Cambios” y “Borrar todo”</i>	<i>132</i>
<i>Figura A.30.- Apartado “Eliminar Actuadores” del Módulo de Configuración.....</i>	<i>132</i>
<i>Figura A.31.- Módulo “Eliminar Actuadores” del Módulo de Configuración.....</i>	<i>133</i>
<i>Figura A.32.- Apartado “Modificar Plano” del Módulo de Configuración.....</i>	<i>133</i>
<i>Figura A.33.- Módulo “Modificar Plano” del Módulo de Configuración</i>	<i>134</i>
<i>Figura A.34.- Apartado “Modificar Nombre de la Vivienda/Hotel” del Módulo de Configuración</i>	<i>134</i>
<i>Figura A.35.- Módulo “Modificar Nombre de la Vivienda/Hotel” del Módulo de Configuración.....</i>	<i>135</i>
<i>Figura A.36.- Apartado “Modificar Email” del Módulo de Configuración</i>	<i>135</i>
<i>Figura A.37.- Módulo “Modificar Email” del Módulo de Configuración.....</i>	<i>136</i>
<i>Figura A.38.- Menú Contadores</i>	<i>136</i>
<i>Figura A.39.- Método de guardado de fichero .dat.....</i>	<i>137</i>
<i>Figura A.40.- Opción “Guardar Proyecto”.....</i>	<i>137</i>
<i>Figura A.41.- Opción “Abrir Proyecto”</i>	<i>138</i>
<i>Figura A.42.- Seleccionar un proyecto existente</i>	<i>139</i>
<i>Figura A.43.- Proyecto abierto en plano de plantas.....</i>	<i>139</i>
<i>Figura A.44.- Menú “Opciones”.....</i>	<i>140</i>
<i>Figura A.45.- Opción “Cerrar Proyecto”</i>	<i>140</i>
<i>Figura A.46.- Interfaz “Procesamiento Estadístico”</i>	<i>141</i>
<i>Figura A.47.- Método de apertura de “Procesamiento Estadístico”</i>	<i>142</i>
<i>Figura A.48.- Nuevo proyecto de “Procesamiento Estadístico”</i>	<i>143</i>
<i>Figura A.49.- Cuadro de diálogo número de gráficas.....</i>	<i>143</i>

<i>Figura A.50.- Vista correspondiente a una gráfica.....</i>	<i>144</i>
<i>Figura A.51.- Vista correspondiente a cuatro gráficas.....</i>	<i>144</i>
<i>Figura A.52.- Opciones de tipo de analisis</i>	<i>145</i>
<i>Figura A.53.- Campos de tipo de análisis</i>	<i>146</i>
<i>Figura A.54.- Opción " Intervalo"</i>	<i>147</i>
<i>Figura A.55.- Opción "Completo"</i>	<i>148</i>
<i>Figura A.56.- Añadir datos</i>	<i>149</i>
<i>Figura A.57.- Generar gráficas</i>	<i>149</i>
<i>Figura A.58.- Salir.....</i>	<i>150</i>
<i>Figura A.59.- Actualizar base de datos.....</i>	<i>150</i>
<i>Figura A.60.- Menú contadores interfaz principal.....</i>	<i>152</i>
<i>Figura A.61.- Ventana administrar contadores.....</i>	<i>153</i>
<i>Figura A.62.- Logs de Contadores</i>	<i>155</i>
<i>Figura A.63.- Vista ventana "Escribir"</i>	<i>156</i>
<i>Figura A.64.- Módulo de control y monitorización.....</i>	<i>157</i>
<i>Figura A.65.- Menú "Ayuda"</i>	<i>165</i>
<i>Figura A.66.- Vista "Configuración Actual"</i>	<i>165</i>
<i>Figura A.67.- Opción "Acerca de"</i>	<i>166</i>
<i>Figura A.68.- Vista "Acerca de"</i>	<i>167</i>

Índice de tablas

<i>Tabla 3.1.- Iconos Luces Regulables antiguos en formato JPG</i>	<i>36</i>
<i>Tabla 3.2.- Iconos Luces Regulables nuevos en formato PNG.....</i>	<i>37</i>
<i>Tabla 3.3.- Iconos Luces Conmutables antiguos en formato JPG</i>	<i>37</i>
<i>Tabla 3.4.- Iconos Luces Conmutables nuevos en formato PNG</i>	<i>37</i>
<i>Tabla 3.5.- Iconos Persianas antiguos en formato JPG</i>	<i>38</i>
<i>Tabla 3.6.- Iconos Persianas nuevos en formato PNG.....</i>	<i>38</i>
<i>Tabla 3.7.- Iconos Electroválvula antiguos en formato JPG</i>	<i>38</i>
<i>Tabla 3.8.- Iconos Electroválvula nuevos en formato PNG</i>	<i>39</i>
<i>Tabla 3.9.- Iconos Puerta antiguos en formato JPG.....</i>	<i>39</i>
<i>Tabla 3.10.- Iconos Puerta nuevos en formato PNG</i>	<i>39</i>
<i>Tabla 3.11.- Iconos Detector de Movimiento antiguos en formato JPG.....</i>	<i>40</i>
<i>Tabla 3.12.- Iconos Detector de Movimiento nuevos en formato PNG.....</i>	<i>40</i>
<i>Tabla 3.13.- Iconos Sensor Temp. antiguos en formato JPG</i>	<i>40</i>
<i>Tabla 3.14.- Iconos Sensor Temp. nuevos en formato PNG</i>	<i>41</i>
<i>Tabla 3.15.- Icono Sensor Combinado nuevo en formato PNG</i>	<i>41</i>
<i>Tabla 3.16.- Iconos Sensor In. antiguos en formato JPG</i>	<i>41</i>
<i>Tabla 3.17.- Iconos Sensor In. nuevos en formato PNG</i>	<i>42</i>
<i>Tabla 3.18.- Icono Contadores nuevo en formato PNG.....</i>	<i>42</i>
<i>Tabla 3.19.- Iconos de Menú</i>	<i>43</i>
<i>Tabla 4.1.- Comparativa de licencias de software libre</i>	<i>50</i>
<i>Tabla 5.1.- Tabla de OS más utilizados entre 2012 y 2015</i>	<i>82</i>
<i>Tabla 5.2.- Ventajas e inconvenientes Aplicación Nativa.....</i>	<i>84</i>
<i>Tabla 5.3.- Ventajas e inconvenientes Web Apps</i>	<i>84</i>
<i>Tabla 5.4.- Ventajas e inconvenientes Aplicaciones Híbridas.....</i>	<i>85</i>

<i>Tabla P.1.- Coeficientes reductores para trabajo tarifado.....</i>	<i>100</i>
<i>Tabla P.2.- Precios y coste de amortización del hardware</i>	<i>101</i>
<i>Tabla P.3.- Precios y costes de amortización de software.....</i>	<i>102</i>
<i>Tabla P.4.- Gastos derivados de la redacción del proyecto.....</i>	<i>102</i>
<i>Tabla P.5.- Presupuesto final.....</i>	<i>103</i>
<i>Tabla PL.1.- Recursos hardware utilizados.....</i>	<i>104</i>
<i>Tabla PL.2.- Recursos software utilizados</i>	<i>104</i>

Acrónimos

.jar	Java Archive
APDU	Application Protocol Data Unit
API	Application Programming Interface
ASL	Apache Software License
BSD	Berkeley Software Distribution
CD-ROM	Compact Disc – Read Only Memory
cEMI	Common External Message Interface
CLI	Command Line
COITT	Colegio Oficial de Ingenieros Técnicos de Telecomunicación
DMCA	Digital Millennium Copyright Act
DPT	Data Point Type
DRM	Digital Rights Management
EIB	European Installation Bus
EPL	Eclipse Public License
FTP	File Transfer Protocol
GPL	General Public License
IP	Internet Protocol
IUSIANI	Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en la Ingeniería
JPG	Joint Photographic Group

KNX	Konnex
LAN	Local Area Network
MPL	Mozilla Public License
MVC	Modelo Vista Controlador
OSSL	Open Secure Sockets Layer
PC	Personal Computer
PDF	Portable Document Format
PFC	Proyecto Fin de Carrera
PNG	Portable Network Graphics
SCADA	Supervisory Control And Data Acquisition
SKoA	Scada Konnex Autoconfigurable
ULPGC	Universidad de Las Palmas de Gran Canaria
XML	Extensible Markup Language

MEMORIA

Capítulo 1:

Introducción y antecedentes

1.1 Introducción

La domótica es la ciencia que estudia la aplicación de la informática y las comunicaciones al hogar, con el fin de conseguir unas viviendas o instalaciones inteligentes.

Desde hace décadas, los avances de la ciencia domótica han ido a la par con la necesidad del ser humano de mejorar su vida y de los que están a su alrededor. Por esta razón día a día se buscan los métodos necesarios para hacer que la domótica sea una realidad en cada vivienda y cada instalación, mejorando la eficiencia energética, la gestión, el confort y la seguridad de los usuarios que disfruten de este tipo de infraestructuras. [1]

En eso se basa este proyecto. A través de él se logrará llevar a cabo la optimización del software SKoA, aplicación destinada al control, monitorización y visualización en término real, sin intervención humana, de los dispositivos domóticos Konnex instalados en cualquier edificio, para su publicación como software libre en su versión 1.0

De esta forma se logrará poner al alcance de los usuarios finales una aplicación que les permita gestionar y controlar infraestructuras Konnex de forma universal para cualquier tipo de vivienda o instalación.

1.2 Antecedentes

La evolución ha sido la que ha marcado el ritmo de la vida en nuestro planeta. Desde el descubrimiento del fuego, pasando por las velas, las antorchas o las lámparas de aceite, hasta llegar a uno de los grandes hallazgos de la historia de la humanidad: la electricidad.

La electricidad ha cambiado de forma radical nuestro sentido del confort y nuestro alrededor. Desde su uso en la industria hasta su empleo dentro de todas y cada una de las viviendas, haciendo que nuestro nivel de vida, y por tanto nuestra esperanza de vida, hayan aumentado de forma imprevisible varios siglos atrás.

La electricidad dió paso a la electrónica, que nos ha permitido elevar el nivel de confort en nuestras casas y ha dado paso a la entrada de los electrodomésticos: lavadora, frigorífico, lavavajillas, horno, placas vitrocerámicas, etc, máquinas capaces de realizar tareas cotidianas de forma casi autónoma.

Estas máquinas no existirían sin el desarrollo y la evolución de la electrónica, permitiendo realizar programaciones (rutinas), que regulan cada proceso.

En base a dicha evolución electrónica nace otra revolución: la domótica. Esta ciencia se encarga de la integración y regulación de ambos sistemas (eléctricos y electrónicos), de tal manera que la infraestructura es capaz de “sentir” (detectar la presencia de personas, la temperatura, el nivel de luz,...) y reaccionar por sí sola, a estos estímulos (regulando el clima, la iluminación, conectando la alarma,...), al mismo tiempo que es capaz de comunicarse e interactuar con nosotros (telecontrol) por multitud de medios (pantalla táctil, PC, móvil,...), llegando a elevadas cotas de confort, seguridad y, sobre todo, ahorro energético.

La historia de la domótica comprende una serie de etapas: desde los primeros protocolos orientados al "control remoto" hasta los grandes protocolos capaces de realizar "funciones lógicas complejas", para satisfacer las más exigentes programaciones. [2]

1.3 Objetivos del PFC

El presente proyecto tiene como objetivo principal la optimización del software SKoA, creado por varios alumnos en Proyectos de Carrera anteriores a éste, para su posterior publicación como software libre. Además, se complementará con la creación de una aplicación móvil que muestre los datos de consumo energético de los hoteles de una de las webs del Plan PILVE desarrollado por el IUSIANI (Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería).

Para cumplir con este propósito se han establecido una serie de hitos o bloques con los que se ha pretendido estructurar el buen desarrollo de este Proyecto Fin de Carrera.

- **Bloque 1:** Estudio y análisis detallado del código fuente del software SKoA, en su última versión 0.2, desarrollado por varios alumnos de la ULPGC en proyectos anteriores. Adaptación al diseño y la dinámica del mismo software.
- **Bloque 2:** Familiarización con el lenguaje JAVA.
A partir de la implementación de diversos ejemplos se pretende familiarizarse con este lenguaje de programación así como obtener un manejo fluido del software a utilizar.
- **Bloque 3:** Limpieza del código y reorganización de los archivos. También se procederá a eliminar o modificar partes del código fuente y a mejorar la experiencia visual con el fin de mejorar la eficiencia del software. Todo esto estará enmarcado en las tareas de optimización del software.
- **Bloque 4:** Publicación del software bajo licencia libre para su uso en la comunidad.
- **Bloque 5:** Creación de la aplicación móvil incluida en este proyecto que mostrará de forma interactiva los datos energéticos producidos por las placas fotovoltaicas de los hoteles del Grupo Lopesan inmersos en el Proyecto PILEV de la ULPGC. Se realizará el código de la misma a través del framework multiplataforma Phonegap.

1.4 Peticionario

La realización de este Proyecto Fin de Carrera se lleva a cabo a petición de la Escuela de Ingeniería de Telecomunicación y Electrónica de la Universidad de Las Palmas de Gran Canaria como requisito para la obtención del título académico de Ingeniero Técnico de Telecomunicación en Sistemas de Telecomunicación por parte del alumno encargado de su realización, una vez ha superado éste la totalidad de los créditos de los que consta dicha titulación.

1.5 Estructura del documento

El presente documento está estructurado en cuatro grandes bloques: memoria, presupuesto, pliego de condiciones y anexos. Además, se adjunta un CD-ROM en el que irá incluida toda la documentación en formato digital resultante de la elaboración de este proyecto.

A continuación se realizará una breve descripción de los contenidos de cada uno de sus puntos a fin de poder ofrecer una idea general de los aspectos tratados en este Proyecto Fin de Carrera.

1.5.1 Memoria

La memoria conforma la parte descriptiva de la totalidad del proyecto. Está dividida en cinco capítulos, el primero de los cuales lo constituye esta introducción, y un capítulo bibliográfico.

Capítulo 1: en él se realiza una presentación inicial en la que se exponen los antecedentes generales de la temática a tratar y por los que se pretende dar una visión global de los objetivos y estructura de este Proyecto Fin de Carrera.

Capítulo 2: en el segundo capítulo se detallan las funcionalidades del software SKoA y se analiza el código fuente de la aplicación con el fin de buscar las principales bazas para la optimización del mismo, de forma que logremos hacerla sencilla de comprender y eficiente.

Capítulo 3: en este capítulo se especifica paso a paso las mejoras realizadas para la optimización de la aplicación SKoA, desde las modificaciones que mejorarán su eficiencia hasta las que desarrollarán un entorno gráfico más moderno y visual.

Capítulo 4: en el cuarto capítulo de esta memoria se ponen exponen las posibles licencias libres bajo las que publicar SKoA, haciéndose un análisis completo de cada opción y decidiendo cuál será la mejor elección respecto a nuestras circunstancias

Capítulo 5: en él se esclarecen los procedimientos llevados a cabo para la creación e implementación de una aplicación móvil que mostrará los datos energéticos recogidos en los hoteles donde se desarrolla el Proyecto PILVE desarrollado por el IUSIANI.

Capítulo 6: en este capítulo se presenta un resumen de las principales conclusiones obtenidas a partir de la elaboración de este Proyecto Fin de Carrera.

Bibliografía: finalmente se enumeran en este capítulo las referencias bibliográficas consultadas.

Anexos: en este bloque se incluyen aquellas cuestiones que deban ser desarrolladas con mayor detalle y que sirvan para ofrecer un mejor entendimiento del proyecto.

1.5.2 Pliego de condiciones

Se describen en este apartado las condiciones de trabajo en las que se elaboró este Proyecto Fin de Carrera.

1.5.3 Presupuesto

En este bloque se cuantifican los costes de desarrollo del Proyecto Fin de Carrera que nos ocupa según los últimos baremos publicados por el Colegio Oficial de Ingenieros Técnicos de Telecomunicación (COITT).

1.5.4 CD-ROM

Vendrá incluida en este CD-ROM la copia digitalizada en formato PDF de la presente memoria además de los distintos archivos generados en la elaboración de este Proyecto Fin de Carrera (código fuente).

Capítulo 2:

Análisis Previo. El software SKoA

2.1 Introducción

En la actualidad, son muchas las viviendas “inteligentes”, es decir, las viviendas domóticas, automatizadas. Hoy en día, tanto compradores como vendedores son conscientes de las ventajas que ofrece la gestión de todos los elementos del entorno doméstico, incluyendo la seguridad y el confort. Los sistemas pueden ser controlados desde el interior o el exterior mediante un ordenador, un teléfono móvil o la voz.

El incremento del número de personas conscientes de las ventajas que aporta una vivienda automatizada ha sido generado en los últimos años por el auge de las nuevas tecnologías y la introducción de Internet en los hogares. Este hecho ha llevado a desarrollar productos capaces de satisfacer las necesidades de las viviendas y de sus usuarios añadiendo nuevas funciones, sin olvidar que dichos productos deben tener entre sus características: facilidad de uso, integración de funciones e interactividad entre ellos mismos y con el usuario, haciendo un uso óptimo de la energía.

El mercado domótico actual tiene una gran diversidad de productos debido a que se trata de un mercado reticular, es decir, varían las tecnologías, los actores y las informaciones haciendo difícil medirlo y valorarlo para saber qué espera la demanda y por tanto qué oferta se debe presentar. Las empresas dedicadas a este sector ofrecen sus propios productos y en la mayoría de los casos, no son genéricos sino personalizados para cada instalación domótica.

En el caso de las instalaciones domóticas Konnex, cada instalación puede ser distinta a las demás respecto al número y tipo de componentes, y en la estructura del edificio donde se encuentra. Por ello, hasta el momento se desarrollaba una aplicación SCADA diferente para cada instalación domótica.

La aplicación SKoA pretende solucionar este problema aportando una aplicación genérica que se personalice para cada instalación. SKoA es una innovación en el mercado, y en este proyecto vamos a realizar una optimización y renovación gráfica, también innovadora, así como proceder a su publicación como software libre para el uso y disfrute de todos los usuarios que deseen disfrutar de sus funcionalidades y ventajas.

El software SKoA es una aplicación de tipo SCADA que se ha ido desarrollando de forma gradual a través de diversos proyectos adjuntos a la ULPGC, llevados a cabo por alumnos de la propia universidad.

El objetivo de este proyecto es optimizar y limpiar el código de este software, haciéndolo lo más eficiente y comprensible posible, para su publicación como software libre.

2.1.1 Aplicaciones SCADA

SCADA está compuesto por las siglas “*Supervisory Control And Data Acquisition*”, que traducido del inglés significa “Control de Supervisión y Adquisición de Datos”. Se trata de una aplicación software diseñada especialmente para funcionar sobre ordenadores en el control de producción, proporcionando comunicación con los dispositivos de campo (controladores autónomos, autómatas programables, etc.) y controlando el proceso de forma automática desde la pantalla del ordenador [3].

Además, provee a diversos usuarios, tanto del mismo nivel como de otros supervisores dentro de la empresa, de toda la información que se genera en el proceso productivo.

El software necesario y, en su caso, el hardware adicional se denomina generalmente *sistema SCADA*. En este tipo de sistemas usualmente existe un ordenador que efectúa tareas de supervisión y gestión de alarmas, así como tratamiento y almacenamiento de datos y control de procesos (de calidad, de producción, ..., etc).

La comunicación se realiza mediante buses especiales o redes LAN. Las tareas y la comunicación se ejecutan normalmente en tiempo real, y están diseñados para dar al operador de planta la posibilidad de supervisar y controlar dichos procesos.

Todo sistema SCADA debe poder ofrecer:

- Posibilidad de crear paneles de alarma, que exigen la presencia del operador para reconocer una parada o situación de alarma, con registro de incidencias.
- Generación de históricos de señal de planta, que puedan ser volcados para procesarlos.
- Ejecución de programas, que modifican la ley de control, o incluso anulan o modifican las tareas asociadas al autómatas, bajo ciertas condiciones.
- Posibilidad de programación numérica, que permita realizar cálculos aritméticos de elevada resolución sobre el procesador del ordenador.

Así, se pueden desarrollar aplicaciones para ordenadores, con captura de datos, análisis de señales, presentaciones en pantalla, envío de resultados a disco e impresora, etc.

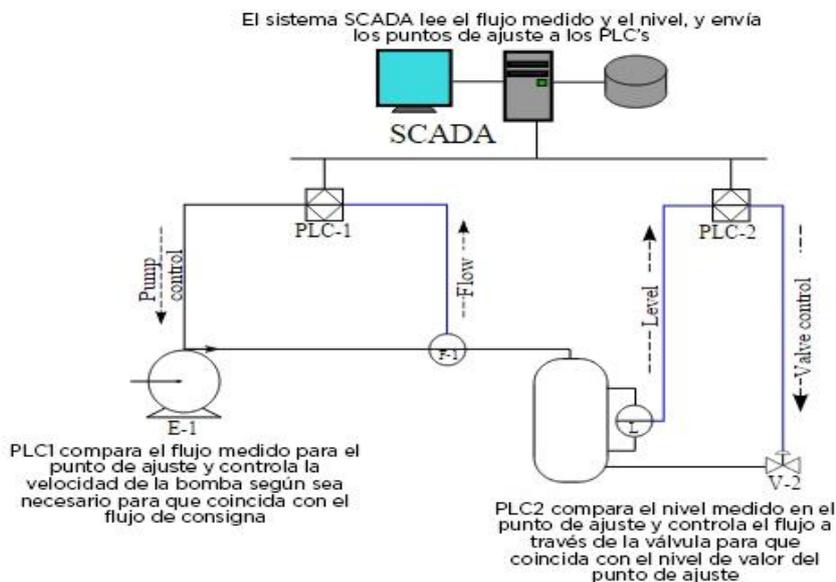


Figura 2.1 Esquema de un sistema SCADA típico

2.1.2 Aplicación SKoA 0.2

SKoA, acrónimo de “*Scada-Konnex-Autoconfigurable*”, es una aplicación informática de tipo SCADA que dispone de una interfaz gráfica destinada al control, monitorización y visualización en término real, sin intervención humana, de los dispositivos domóticos Konnex [4] instalados en cualquier edificio. Es decir, controla y monitoriza una instalación domótica Konnex de forma centralizada a través de un ordenador [5].

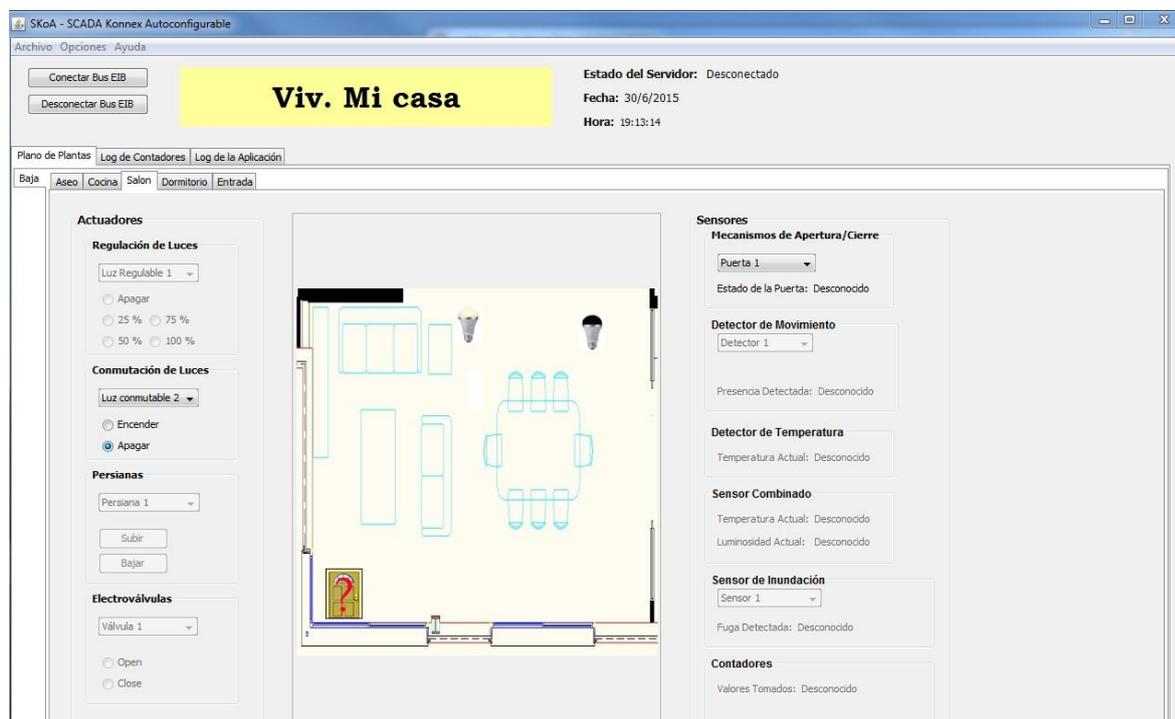


Figura 2.2 Módulo de control y monitorización SKoA 0.2

Esta aplicación se puede personalizar para cualquier instalación Konnex a través de un fichero de configuración, que una vez generado, permite autoconfigurar la aplicación automáticamente.

Los archivos de configuración representan la estructura arquitectónica del edificio y todos los dispositivos instalados en él. Para facilitar la labor de generación del fichero de autoconfiguración, SKoA dispone de un módulo software que permite generar o modificar este fichero de forma gráfica, sencilla e intuitiva, permitiendo al usuario configurar las

plantas y las estancias del edificio, con sus correspondientes planos, además de las direcciones de grupo KNX a monitorizar y controlar.

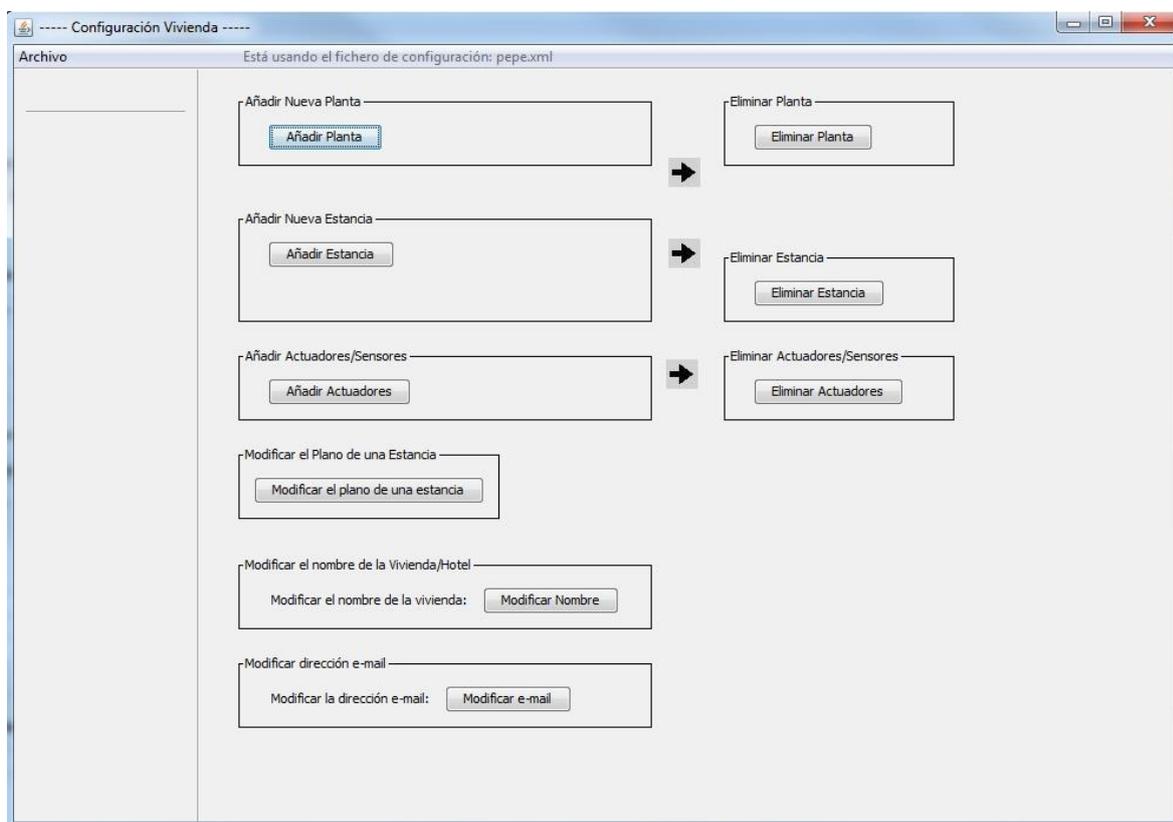


Figura 2.3 Módulo de Configuración SKoA 0.2

Adicionalmente, SKoA permite enviar mensajes de alarma o de monitorización de forma automática mediante correos electrónicos a la dirección de correo electrónico especificada en el fichero de autoconfiguración.

La última característica relevante es la posibilidad de generar y visualizar diagramas gráficos donde se muestra la evolución temporal de los datos, enviados por el bus KNX, asociados a una determinada dirección de grupo. Así, se facilita el análisis temporal del funcionamiento de la instalación domótica.

Al iniciar este proyecto se trabaja exclusivamente sobre la última versión existente, que es SKoA 0.2

2.2 Análisis previo del código fuente de SKoA 0.2

En primer lugar, y para asegurar una correcta optimización del software, nos centramos en el análisis a fondo del código fuente realizado durante los anteriores proyectos.

El objetivo de esta actividad será comprender las funciones y acciones que el código realiza para llevar a cabo el correcto funcionamiento de la aplicación software, así como comprender y manejar todos los componentes que conforman el código, como llamadas, bibliotecas, etc.

El hecho de que el proyecto haya sido realizado durante varios proyectos llevados a cabo por personas diferentes puede darnos algunas pistas sobre posibles códigos enmarañados o incluso con partes de código muerto.

2.2.1 Organización del código

Al proceder al análisis a fondo del código de la aplicación SKoA en su última versión 0.2, nos encontramos ante una estructura bastante desorganizada y sin patrones que nos permitan llevar a cabo una lectura ordenada y detallada de su funcionamiento interno.

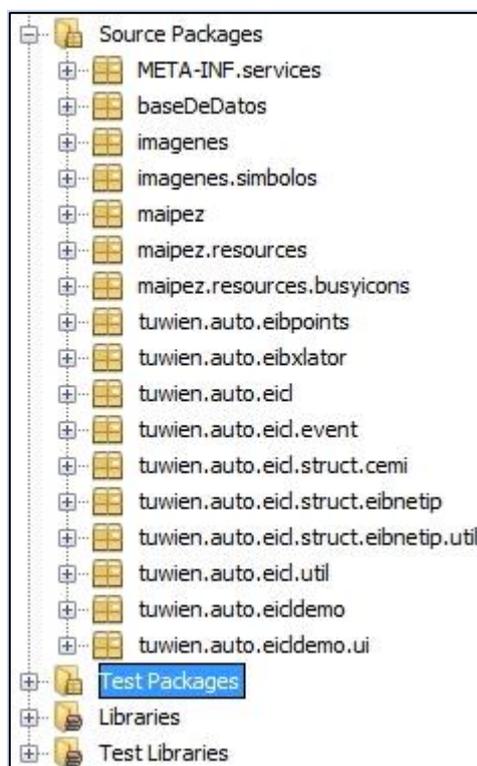


Figura 2.4 Estructura de paquetes y librerías del código fuente

La falta de un patrón concreto que nos ayude a establecer un sistema de trabajo para proceder a la modificación del código hace que esta tarea sea extremadamente compleja y correosa.

2.2.2 Redundancia en las validaciones

En gran parte del código, específicamente en las clases que se encargaban de recorrer el árbol de XML y encontrar un nodo o hacer una operación en base a un parámetro dentro del árbol XML, nos encontramos que dichas validaciones se hacen de una manera errónea, ya que muchas de ellas podrían ser ejecutadas en una sola sentencia y muchas otras comparan cadena de caracteres o ristas de una manera errónea ya que validan la referencia al objeto y no el valor como tal.

2.2.3 Dependencias innecesarias

También nos percatamos que algunas dependencias estaban repetidas con versiones anteriores y otras que no se utilizaban.

Además, el software se apoya principalmente en la librería “**tuwien.auto.***”, una librería de terceros que realiza la comunicación con los dispositivos de domótica, dicha librería está contenida en su totalidad dentro del código fuente de SKoA así como en la generación de gráficas.

2.2.4 Implementación errónea de la internacilización

En la aplicación SKoA existen dos versiones: una en el idioma español y otra en el idioma inglés. Ambas versiones se encuentran en la estructura de forma independiente y tan sólo difieren en las traducciones pertinentes para cada idioma.

Concretamente la principal diferencia la encontramos en los archivos de configuración de las clases encargadas de crear la vista de cada interfaz.

Esta estructuración externalizada hace que nuestro software pierda eficiencia y además podría confundir al usuario final a la hora de proceder a la instalación y uso de esta aplicación

2.2.5 Proyectos no acoplados

Como decíamos anteriormente, este software ha sido diseñado y modificado en diversos proyectos anteriores, en los que sea realizados mejoras o se le han añadido nuevas aplicaciones y funcionalidades al conjunto.

Aparte del proyecto principal SKoA, existe un proyecto anexo llamado “gráficas” que es un componente o una extensión del sistema como tal, pero que es invocado de manera externa como una aplicación diferente.

Además, éste presenta el mismo problema de expuesto en el apartado anterior respecto al idioma de ejecución, dado que dicha aplicación anexa también cuenta con dos versiones diferentes para los distintos idiomas en los que permite su uso: Español e Inglés.

2.2.6 Iconos e imágenes duplicadas en la estructura

Si recorremos la estructuración del código, encontramos que tanto los iconos usados en el software como las imágenes empleadas para diferentes funcionalidades, se encuentran repetidos en varias carpetas.

De esta formando se está produciendo una sobrecarga de datos innecesaria, que podría estar afectando a la velocidad de ejecución de nuestra aplicación, haciéndolo menos eficiente y más lento.

Además, hemos visto que las imágenes e iconos están algo desfasadas y además presentan un diseño incorrecto que hace que la visualización por parte del usuario final sea bastante desagradable y poco atractiva.

El uso de archivos de imagen .jpg, con su respectiva carga de fondo blanco detrás de la imagen principal, hace que el icono u objeto en si pierda contraste con el entorno que lo

envuelve, haciendo que la vista final sea chocante.

2.2.7 Resumen del capítulo

En este capítulo hemos realizado un profundo análisis del código fuente de la última versión del software SKoA con el fin de hallar los principales factores en los que podríamos centrarnos a la hora de realizar la optimización del software.

Tras este estudio esperamos haber recopilado todas las claves que nos ayuden a mejorar el código y su estructura para llevar a cabo uno de los objetivos de este proyecto: lograr un software optimizado y con buena impresión visual para el usuario final.

Capítulo 3:

Optimización del software

3.1 Introducción

En este apartado pasaremos a redactar todas las acciones realizadas para la optimización del software SKoA, de manera que se convierta en una aplicación más eficiente, más atractiva y lista para su publicación como software libre.

3.2 Tareas de optimización

Cuando se quiere desarrollar cualquier código, programa y/o software se han de cumplir al menos tres premisas casi axiomáticas: la primera de ellas es que dicho programa debe realizar la función o funciones para las que ha sido diseñado; la segunda es que debe realizarla correctamente y sin errores; y la tercera es que debe estar implementado de la forma más eficiente posible. Para cumplir con estas condiciones, y reducir el coste de recursos necesario, se ha sometido a este código a un cuidadoso proceso de optimización.

3.2.1 Librería “twien.auto.*”

Una de las librerías en las que más se apoya nuestro código fuente es “twien.auto.*” y por ellos centramos nuestra atención en ella, dado que al tener tanta presencia e importancia en el funcionamiento de la aplicación, podremos encontrar grandes puntos de optimización trabajando en ella.

3.2.1.1 ¿Qué es “twien.auto.*”

“Twien.auto.*” es el nombre dado al paquete en el que se define la librería Calimero con la que se trabaja en SKoA.

Calimero es una colección de APIs Java que, en conjunto, forman una base para su

posterior conexión en aplicaciones de alto nivel EIB / KNX (incluyendo acceso remoto y control). Existe una aplicación cliente, pero destinada sólo para fines de demostración.

La librería Calimero incluye:

Eibclient es una librería cliente EIBnet/IP que soporta conexiones de *tunelling*, así como búsqueda y descripción. El objetivo de este API es permitir el establecimiento de las conexiones EIBnet / IP con el mínimo esfuerzo. El intercambio de mensajes EIBnet / IP completo, incluyendo el corazón latidos se oculta desde el programador cliente, que sólo se enfrenta a una API para enviar y recibir mensajes CEMI. Además, eibclient es capaz de analizar los mensajes CEMI para extraer el los datos pertinentes, así como montar ellos. La implementación actual se limita a TP1 marcos estándar.

Eibxlator es una colección de codificadores / decodificadores para unidades de datos de protocolo de capa de aplicaciones (APDU) relevante para el intercambio y la interpretación de los valores del grupo. En esta versión sólo un subconjunto de KNX DPT de se implementan, pero el paquete fue diseñado para hacer que la adición de más fáciles.

Eibpoints ofrece la capacidad de mantener una lista de los puntos de datos en el sistema KNX / EIB y sus datos relevantes, incluyendo dirección de grupo, nombre descriptivo, y el tipo de DPT. La lista de puntos de datos abstracción ofrece facilidades de búsqueda por nombre y dirección. La configuración completa puede ser exportada e importada como un archivo XML o una transmisión [5].

Estas API se pueden utilizar en gran medida de forma independiente.

▪ Ejemplo de código

```
import java.net.*;
import tuwien.auto.*;
import tuwien.auto.eibxlator.*;
import tuwien.auto.eicl.*;
import tuwien.auto.eicl.util.*;
import tuwien.auto.eicl.struct.cemi.*;
import tuwien.auto.eicl.struct.eibnetip.util.*;

public class MainClass {

    public static void main(String[] args) {

        try {
            CEMI_Connection tunnel = new CEMI_Connection(
                new InetSocketAddress("tunnelserver.somewhere.net",
                    EIBNETIP_Constants.EIBNETIP_PORT_NUMBER),
```

```

        new TunnellingConnectionType());

        PointPDUXlator dimVal = PDUXlatorList.getPointPDUXlator(
            PDUXlatorList.TYPE_8BIT_UNSIGNED[0],
            PointPDUXlator_8BitUnsigned.DPT_SCALING[0]);

        dimVal.setServiceType(PointPDUXlator.A_GROUPVALUE_WRITE);
        dimVal.setASDUfromString("75");

        CEMI_L_DATA message = new CEMI_L_DATA(
            CEMI_L_DATA.MC_L_DATAREQ,
            new EIB_Address(),
            new EIB_Address("0/0/1"),
            dimVal.getAPDUByteArray());

        tunnel.sendFrame(message, CEMI_Connection.WAIT_FOR_CONFIRM);
    }
    catch (EICLException ex) { } // connection error
}
}

```

Este ejemplo de código completo muestra la facilidad con la que un regulador, puede modificar su valor utilizando las API Calimero.

▪ Problemas Conocidos

Calimero tiene que presentar el servidor EIBnet/IP con la dirección IP de su host para establecer la conexión. Dado que no parece ser una manera portátil de acceso a la información de enrutamiento, esta dirección sólo puede ser detectada automáticamente de forma correcta si el host tiene una única interfaz Ethernet con una única dirección IP.

En otros casos, la dirección IP correcta tiene que ser dada de forma manual. De lo contrario, el servidor EIBnet / IP puede ser inaccesible.

3.2.1.2 Funcionalidad en SKoA

La librería “twien.auto.*” se encontraba embebida dentro del código de la aplicación, y esto podía tender a cierto grado de confusión haciendo su comprensión más compleja.

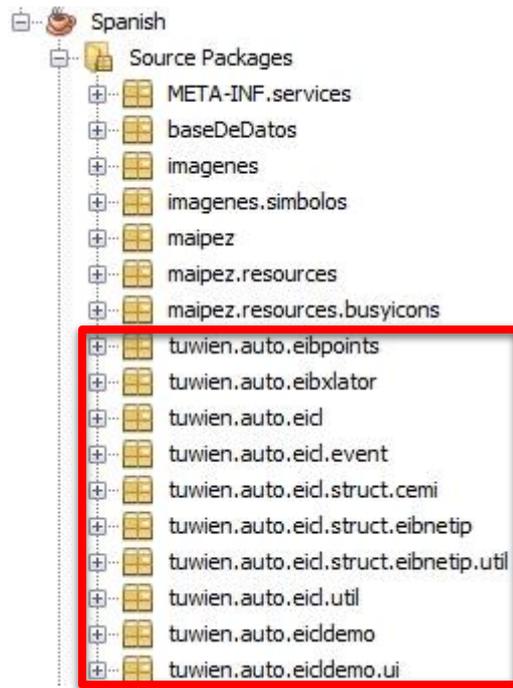


Figura 3.1 Posición de la librería “twien.auto” antes de su optimización

Con el fin de optimizar la aplicación y facilitar una estructuración ordenada y limpia, hemos procedido a separar los componentes de la librería “twien.auto.*” integrándola en un único archivo ejecutable .jar

Esto nos permitirá trabajar y acceder al código de manera más eficaz y sencilla para su utilización en futuras tareas de desarrollo o mantenimiento.

En la siguiente figura podemos apreciar cómo ha quedado integrada la librería “twien” junto al resto de librerías incluídas en la aplicación

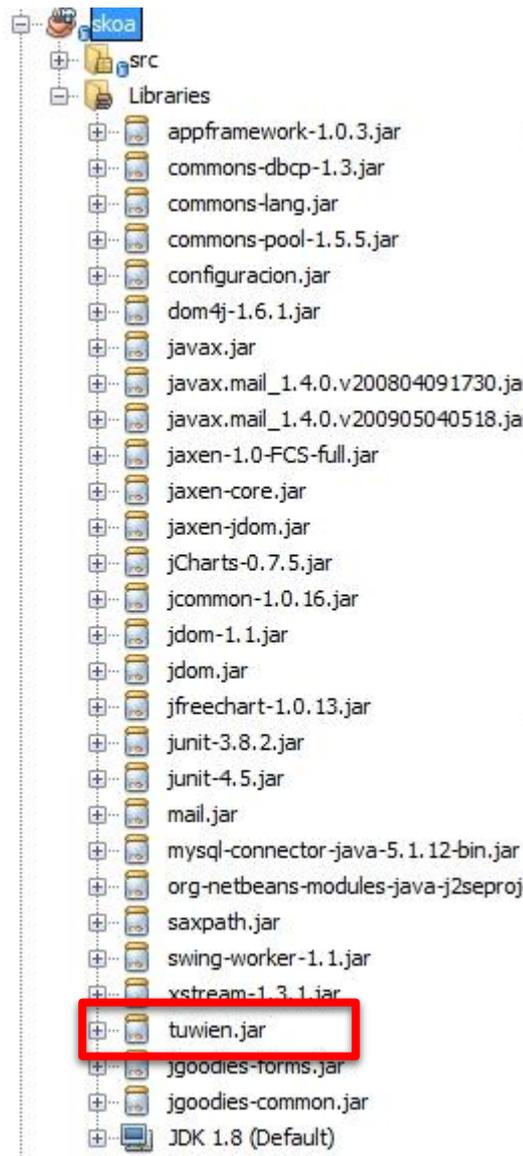


Figura 3.2 Optimización final de la librería “tuwien”

3.2.2 Optimización de las librerías

Nuestra aplicación cuenta con una extensa cantidad de librerías embebidas para realizar las innumerables opciones que se nos permite llevar a cabo en la funcionalidad del software.

En la versión anterior nos encontrábamos ante una cantidad considerable de librerías duplicadas por pertenecer a diferentes versiones de las mismas. Esto hace que se ocupe un espacio innecesario en nuestra aplicación, dado que una de ellas no se está ejecutando ni utilizando en ninguna parte de nuestro código fuente.



Figura 3.3 Librerías duplicadas en la versión antigua

Se procede a la eliminación de aquellas librerías que se encontraran duplicadas con el fin de mejorar la eficiencia de la aplicación. Finalmente las librerías quedan así:

3.2.3 Reorganización de clases

Tras el análisis realizado al código, se detectó que varias de las clases utilizadas en el mismo no cumplían en reiteradas ocasiones los protocolos generales establecidos por los desarrolladores Java.

Se ha decidido cambiar el nombre de las clases por nomenclaturas que cumplan los convenios establecidos por los desarrolladores de Java (CamelCase). Además de que esta nomenclatura de una idea más concisa de la función que realiza dicha clase.

3.2.3.1 CamelCase

CamelCase es un estilo de escritura que se aplica a frases o palabras compuestas. El nombre se debe a que las mayúsculas a lo largo de una palabra en CamelCase se asemejan a las jorobas de un camello [7].

El nombre CamelCase se podría traducir como Mayúsculas/Minúsculas Camello. El término *case* se traduce como "caja tipográfica", que a su vez implica si una letra es mayúscula o minúscula y tiene su origen en la disposición de los tipos móviles en casilleros o cajas.

Existen dos tipos de CamelCase:

- **UpperCamelCase**, cuando la primera letra de cada una de las palabras es mayúscula.

Ejemplo: *EjemploDeUpperCamelCase*

- **lowerCamelCase**, igual que la anterior con la excepción de que la primera letra es minúscula.

Ejemplo: *ejemploDeLowerCamelCase*

Para respetar esta escritura, se han realizado modificaciones en varias clases del código que no respetaban este protocolo, como por ejemplo en una de las clases llamada *estancia_nueva* se realizó una modificación a *EstanciaNueva*.

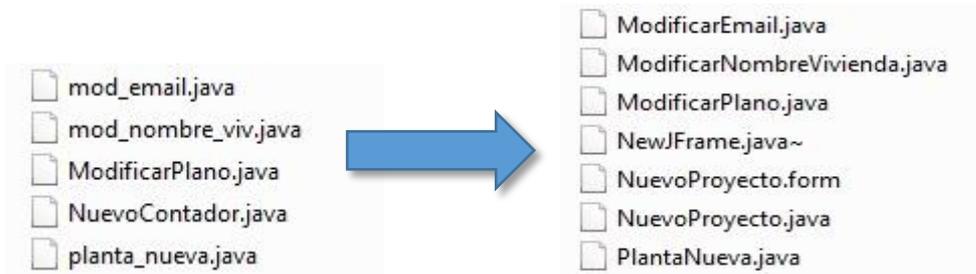


Figura 3.4 Clases sin CamelCase

Figura 3.5 Clases modificadas con UpperCamelCase

Para una mejor visualización de las clases, hemos procedido renombrarlas todas utilizando el tipo **UpperCamelCase**, generando una sensación de homogeneidad hacia el desarrollador.

3.2.3.2 Nombres orientativos

Con el fin de que el desarrollador pueda guiarse a través de la nomenclatura de las clases y los métodos, hemos procedido a la modificación de muchos de ellos con el objetivo de que dicho nombre describa de una forma más directa y concisa su función dentro del código.

Podemos ver algunos ejemplos de las clases renombradas en este proyecto:



Figura 3.6 Comparativa de nomenclatura orientativa

3.2.4 Código comentado

Gran parte del código fuente de la aplicación no contaba con referencia alguna respecto a las funciones que realizaba con el fin de guiar a un nuevo usuario que pretenda llevar a cabo algún tipo de modificación.

Por ello, hemos realizado comentarios en gran parte del código, para lograr que el usuario sea guiado y sepa para qué se utiliza cada parte del mismo.

En el siguiente fragmento de código podemos apreciar como se ha realizado el comentado de cada uno de las funciones:

```
/**
 * Metodo que busca dentro del archivo xml la ruta de la imagen
 *
 * @param planta
 * @param estancia
 * @return
 */
public static String SeleccionarImagen(String planta, String estancia) {
    String dir_imagen = "";
    try {
        // crea una instancia de la clase File del archivo de configuracion
        // xml
        File aFile = new File(NuevoProyecto.archivo);

        // instancia de SAXReader para leer y recorrer el arbol del xml
        SAXReader xmlReader = new SAXReader();

        // xmlReader.setEncoding("UTF-8");
        xmlReader.setEncoding("iso-8859-1");

        // Comenzamos a leer el árbol
        Document doc = xmlReader.read(aFile);
        Element node = (Element) doc.selectSingleNode("//vivienda");
        Element node2 = null;
    }
}
```

3.2.5 Errores de comparación de ristras

En gran parte del código se realizan comparaciones entre ristra de una manera errónea ya que se compara en muchas ocasiones por el objeto y no por el valor del objeto. Es decir, que están comparando si el objeto está vacío o si el objeto es la referencia a otro objeto, pero no su valor.

En la versión anterior se estaba comparando de esta manera:

```
if(node.getName() == "planta"){
    if(node.valueOf( "@alias" ).equals(planta)){
        node2 = node;
    }
}
```

La manera adecuada para comparar si el valor de la ristra es igual a otro valor, se debe comparar con el método *equals* (método contenido dentro de la clase *String*).

```
if (node.getName().equals( "planta")
    && node.valueOf("@alias").equals(planta)) {
    node2 = node;
}
```

3.2.6 Optimización de búsqueda dentro de los nodos en el archivo XML

Durante el recorrido del árbol contenido en el archivo XML de un proyecto nuevo o uno ya existente, nos percatamos de que habían muchas comparaciones repetitivas, es decir, que los bloques *if* contenían la misma instrucción tan sólo cambiando el valor con el que se comparaba.

```
if(node3.getName() == "actuadores"){
    if(nombreIcono == "regulacion"){
        Attribute atributo = node3.attribute("regulacion");
        atributo.setText("si");
    }
    if(nombreIcono == "conmutacion"){
        Attribute atributo = node3.attribute("conmutacion");
        atributo.setText("si");
    }
    if( nombreIcono == "persianas"){
        Attribute atributo = node3.attribute("persianas");
        atributo.setText("si");
    }
    if( nombreIcono == "electrovalvula" ){
        Attribute atributo = node3.attribute("electrovalvula");
        atributo.setText("si");
    }
}
```

Para optimizar este bloque de código poco eficiente, lo que hicimos fue contener las variables dentro de un array y de esta manera simplemente recorrer y verificar si era el valor que estábamos buscando.

```
if (node3.getName().equals("actuadores")) {
    String[] fieldsToInclude = {"regulacion",
        "conmutacion", "persianas",
        "electrovalvula"};
};
for (String field : fieldsToInclude) {
    if (nombreIcono.equals(field)) {
        Attribute atributo = node3.attribute(field);
        atributo.setText("si");
    }
}
} // end if actuadores y sensores
```

3.2.7 Restructuración de paquetes

Debido a que la estructuración del código mostraba una clara desorganización de los paquetes, nos vimos obligados a realizar una concienzuda reordenación según sus funcionalidades.

Para ello seguimos el patrón de arquitectura de software MVC (Modelo-Vista-Controlador)

3.2.7.1 Modelo-Vista-Controlador

El modelo–vista–controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones.

Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario.^{1 2} Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento [8].

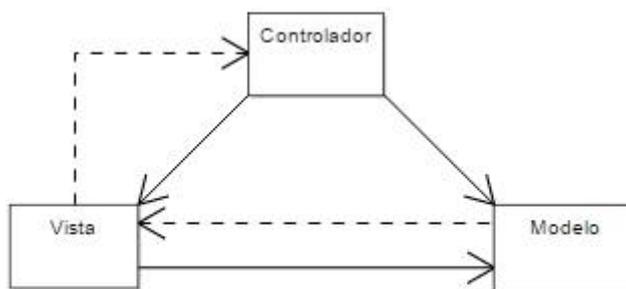


Figura 3.7 Diagrama MVC

3.2.7.2 Descripción del Patrón

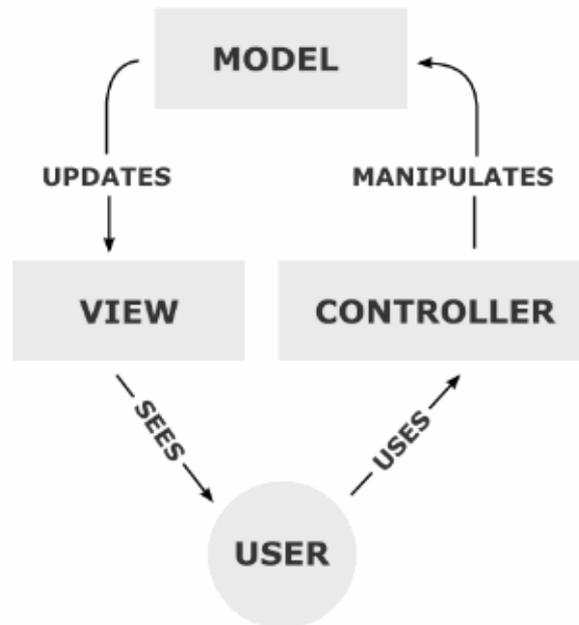


Figura 3.8 Típica colaboración entre componentes de un MVC

De manera genérica, los componentes de MVC se podrían definir como sigue:

- **Modelo:** Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación (lógica de negocio). Envía a la 'vista' aquella parte de la información que en cada momento se le solicita para que sea mostrada (típicamente a un usuario). Las peticiones de acceso o manipulación de información llegan al 'modelo' a través del 'controlador'.
- **Controlador:** Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información (por ejemplo, editar un documento o un registro en una base de datos). También puede enviar comandos a su 'vista' asociada si se solicita un cambio en la forma en que se presenta el 'modelo' (por ejemplo, desplazamiento o scroll por un documento o por los diferentes registros de una base de datos), por tanto se podría decir que el 'controlador' hace de intermediario entre la 'vista' y el 'modelo' (véase Middleware).

- **Vista:** Presenta el 'modelo' (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida.

Interacción de los componentes

Aunque se pueden encontrar diferentes implementaciones de MVC, el flujo de control que se sigue generalmente es el siguiente:

1. El usuario interactúa con la interfaz de usuario de alguna forma (por ejemplo, el usuario pulsa un botón, enlace, etc.)
2. El controlador recibe (por parte de los objetos de la interfaz-vista) la notificación de la acción solicitada por el usuario. El controlador gestiona el evento que llega, frecuentemente a través de un gestor de eventos (handler) o callback.
3. El controlador accede al modelo, actualizándolo, posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario (por ejemplo, el controlador actualiza el carro de la compra del usuario). Los controladores complejos están a menudo estructurados usando un patrón de comando que encapsula las acciones y simplifica su extensión.
4. El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se reflejan los cambios en el modelo (por ejemplo, produce un listado del contenido del carro de la compra). El modelo no debe tener conocimiento directo sobre la vista. Sin embargo, se podría utilizar el patrón Observador para proveer cierta indirección entre el modelo y la vista, permitiendo al modelo notificar a los interesados de cualquier cambio. Un objeto vista puede registrarse con el modelo y esperar a los cambios, pero aun así el modelo en sí mismo sigue sin saber nada de la vista. Este uso del patrón Observador no es posible en las aplicaciones Web puesto que las clases de la vista están desconectadas del modelo y del controlador.

En general el controlador no pasa objetos de dominio (el modelo) a la vista aunque puede dar la orden a la vista para que se actualice. Nota: En algunas implementaciones la vista no tiene acceso directo al modelo, dejando que el controlador envíe los datos del modelo a la vista. Por ejemplo en el MVC usado por Apple en su framework Cocoa. Suele citarse como Modelo-Interface-Control, una variación del MVC más puro

5. La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente....

Nuestro propósito fue agrupar el mayor número de clases dentro de un paquete siguiendo la metodología MVC, ya que una correcta organización del código provee al desarrollador una vista más simple del proyecto así como una visión más ordenada del cómputo global.



Figura 3.9 Estructura de los paquetes tras la optimización

A continuación detallamos el contenido de cada uno de los paquetes:

- **skoa.db:** es el paquete que contiene todas las clases con relación a base de datos, una base de datos mysql.
- **skoa.helpers:** Este paquete contiene clases de ayuda, como el tratado del archivo XML, la carga de imágenes, etc.
- **skoa.models:** Contiene los diferente objetos genéricos que nos sirven de modelo dentro de nuestro proyecto
- **skoa.views:** Este paquete contiene todas las interfaces del sistema

- **sko.resources**: Este paquete contiene todos los archivos de configuración necesarios para las vistas. Estos archivos proveen la funcionalidad de ser un sistema multi-idioma
- **sko.resources.icon**: Contiene todos los iconos utilizados por la aplicación

3.2.8 Implementación correcta de la internacionalización

En un primer momento, cuando se ha tomado el proyecto, nos hemos percatado que para que la aplicación de soporte a varios idiomas, en este caso inglés y español, se implementó dos aplicaciones prácticamente iguales, que diferían tan sólo en las etiquetas mostradas en la vista, así como botones y diferentes componentes que aparecen en las mismas.

Por consiguiente, esto ocasionaba que si se pretendía hacer un cambio en la vista, se tendría que hacer dos veces. Este problema también se daba en el proyecto anexo llamado *Gráficas*.

Esto quiere decir que se había implementado una forma errónea para que la aplicación se mostrara en varios idiomas.

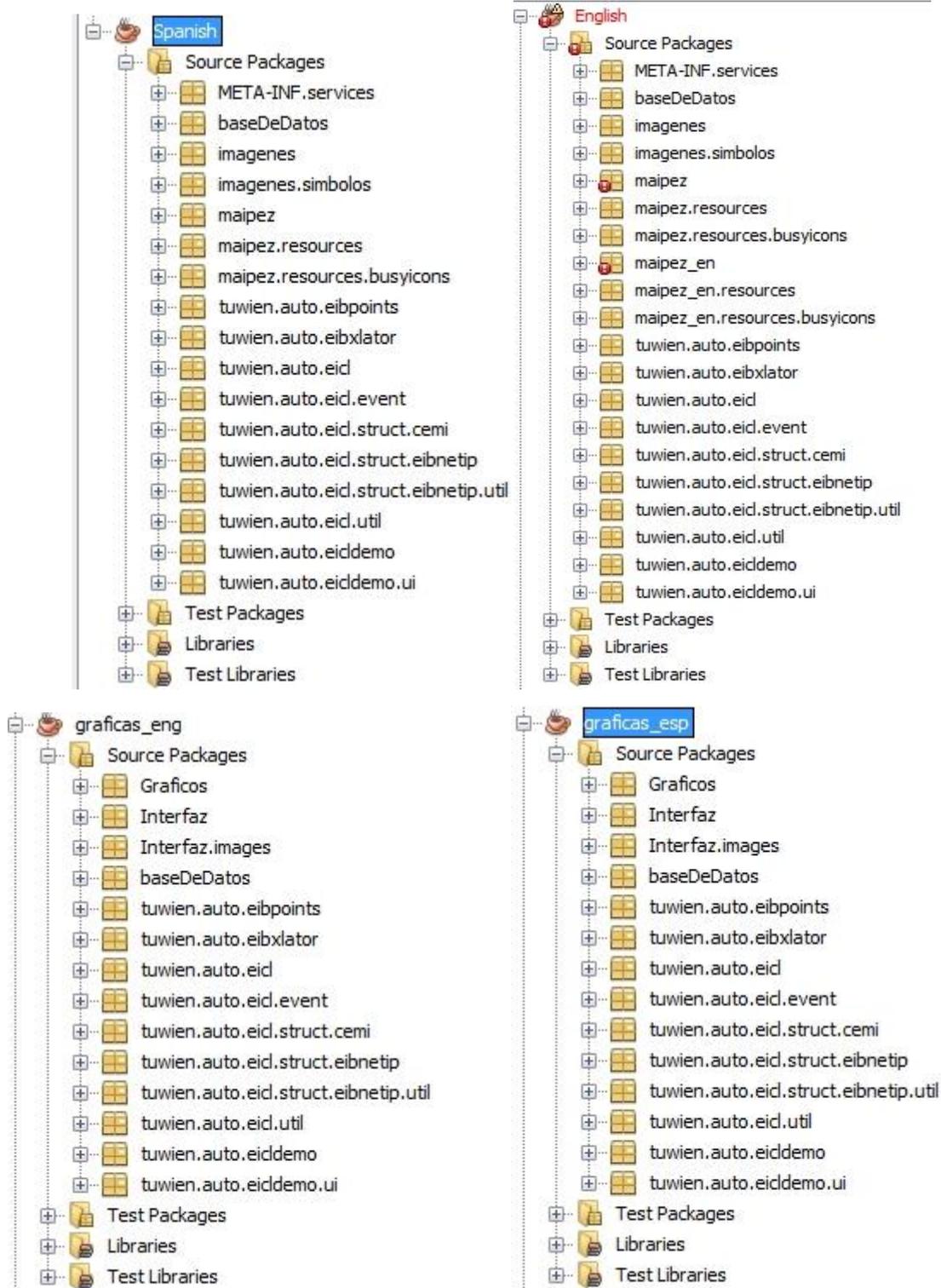


Figura 3.10 Proyectos con paquetes duplicados para distintos idiomas

Para este problema, la solución que hemos hayado ha sido utilizar los archivos de

configuración (**.properties*), dicho de otra forma, es un archivo que se carga mediante la clase *ResourceMap*, la cual provee de una manera simple cargar diferentes configuraciones para los distintos idiomas, concretamente estos archivos se encuentran en el paquete *Skoa.views.resources* y en el cual encontraremos para cada clase que se encargue de una vista dos archivos de configuración, uno en español y otro en inglés.

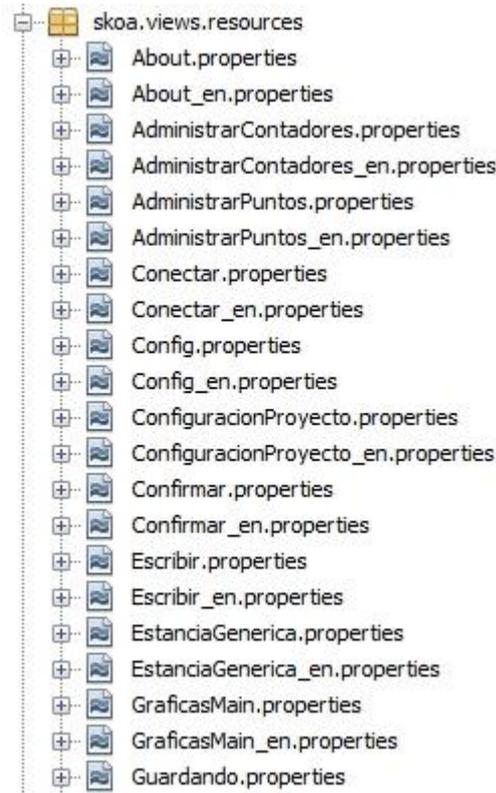


Figura 3.11 Paquete *skoa.views.resources*

Estos archivos básicamente contienen pares de claves y valores:

<pre> title = Acerca de: Software SKoA closeAboutBox.Action.text = &Close vendorLabel.text=Autores: #NOI18N imageLabel.icon=icons/acercade.png closeButton.text=Cerrar jLabel1.text=Inmaculada Rodríguez Nuez jLabel2.text=Domingo Benitez Diaz jLabel3.text=Carlos González Muñoz jLabel4.text=Estefania Tramunt Rubio versionLabel.text=Versión: jLabel6.text=2.0 jLabel5.text=David Monné Chavez </pre>	<pre> title = About us: SKoA Software closeAboutBox.Action.text = &Close versionLabel.text=Version: vendorLabel.text=Authors: #NOI18N imageLabel.icon=icons/acercade.png closeButton.text=Close jLabel1.text=Inmaculada Rodríguez Nuez jLabel2.text=Domingo Benitez Diaz jLabel3.text=Carlos González Muñoz jLabel4.text=2.0 jLabel5.text=David Monné Chávez jLabel6.text=Estefania Tramunt Rubio </pre>
--	--

Figura 3.12 Archivos About.properties y About_en.properties

3.2.9 Integración de proyectos independientes

En el conjunto de la versión anterior de SKoA podemos encontrar diferentes aplicaciones creadas en proyectos anteriores, que se han adherido a la aplicación sin llegar a ser integradas en la misma.

Esto quiere decir que dentro del proyecto SKoA habían subproyectos que eran llamados de la aplicación principal pero que los mismos no habían sido integrados dentro de la propia aplicación.

Ejemplo de ello es el proyecto *Gráficas* (español e inglés), SKoA en Inglés y el proyecto bajo el paquete “*twien.auto.**”.

Se puede apreciar que debido a esto había mucho código repetido, mal utilizado e implementado de un manera errónea. Como se ha explicado anteriormente se han ido subsanando estos problemas mediante la internacionalización, esto solventó el problema que había de código duplicado en los proyectos Gráficas Español, Gráficas Inglés, SKoA Español, SKoA Inglés...y una vez implementados se han unido en un solo proyecto.

La librería *twien.auto* se ha extraído como una dependencia más en este proyecto.

Con esto hemos conseguido que la redundancia de código se haya simplificado en un 100% haciendo que el código del proyecto sea más manejable, más sencillo de analizar y entender para futuras tareas de optimización y mantenimiento.

3.2.10 Iconos modificados

Los iconos utilizados en las anteriores versiones del software no cumplen con una correcta visualización por parte del usuario final.

En muchos casos se trata de imágenes o iconos bastante desfasados a los tiempos actuales y en muchas ocasiones no están correctamente diseñados para su colocación en ciertas áreas de la aplicación, en gran medida por la utilización de formatos incorrectos para el correcto contraste con el entorno gráfico.

En la siguiente imagen de la *Figura 3.7* observamos un menú de SKoA en su antigua versión, en el que podemos apreciar que el incorrecto uso del formato *.jpg* en las imágenes de los iconos provoca que se cargue un fondo blanco que choca visualmente con el entorno gráfico. Lo mismo ocurre en la *Figura 3.8* donde se muestra un plano con actuadores en los que también se han utilizado imágenes en formato *.jpg*, creando un molesto fondo blanco que no nos permiten visualizar de forma correcta el plano sobre el que estamos trabajando.

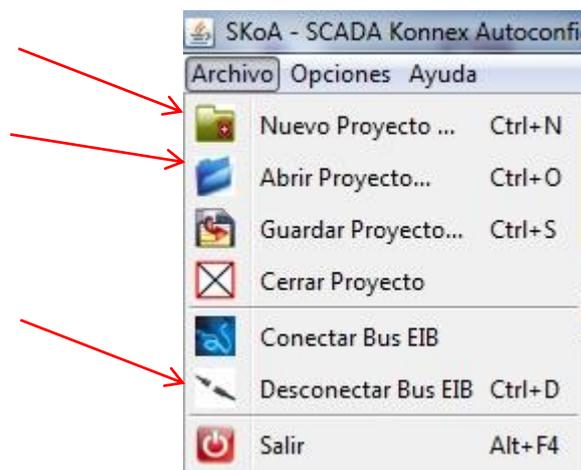


Figura 3.13 Menú versión antigua SKoA

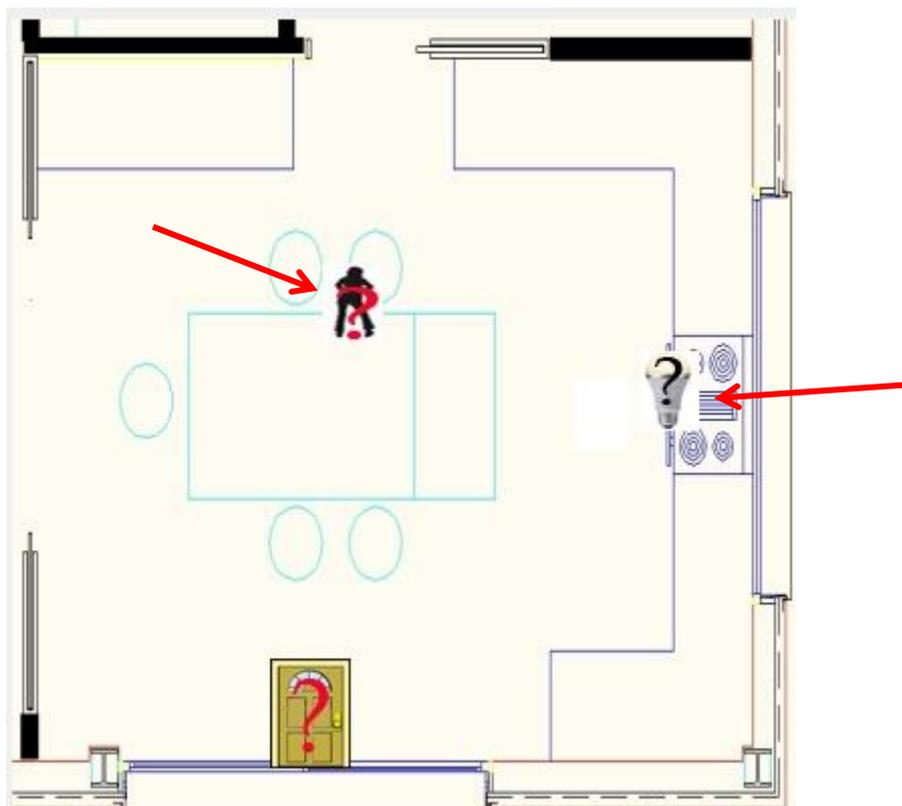


Figura 3.14 Actuadores versión antigua SKoA

Para dar solución a este problema, hemos modificado todas las imágenes utilizadas como iconos o actuadores en el software, utilizando en formato *.png* entrelazado, que permite el muestreo de la imagen sin que se cargue ningún fondo, permitiendo una interfaz más limpia y visualmente agradable.

En las siguientes figuras podemos apreciar con mayor detenimiento los antiguos iconos y los que hemos añadido en este proyecto.

➤ Luces regulables

Estado	Desconocido	Apagado	Al 25 %	Al 50%	Al 75%	Al 100%
Icono						

Tabla 3.1 Iconos Luces Regulables antiguos en *.jpg*

Estado	Desconocido	Apagado	Al 25 %	Al 50%	Al 75%	Al 100%
Icono		 Negro	 Azul	 Amarillo	 Naranja	 Rojo

Tabla 3.2 Iconos Luces Regulables nuevos en .png

➤ **Luces conmutables**

Estado	Desconocido	Apagado	Encendido
Icono			

Tabla 3.3 Iconos Luces Conmutables antiguos en .jpg

Estado	Desconocido	Apagado	Encendido
Icono			

Tabla 3.4 Iconos Luces Conmutables nuevos en .png

➤ **Persianas**

Estado	Desconocido	Cerrada	Subida
Icono			

Tabla 3.5 Iconos Persianas antiguos en .jpg

Estado	Desconocido	Cerrada	Subida
Icono			

Tabla 3.6 Iconos Persianas nuevos en .png

➤ **Electroválvulas**

Estado	Desconocido	Cerrada	Abierta
Icono			

Tabla 3.7 Iconos Electroválvula antiguos en .jpg

Estado	Desconocido	Cerrada	Abierta
Icono			

Tabla 3.8 Iconos Electroválvula nuevos en .png

➤ Puertas

Estado	Desconocido	Cerrada	Abierta
Icono			

Tabla 3.9 Iconos Puerta antiguos en .jpg

Estado	Desconocido	Cerrada	Abierta
Icono			

Tabla 3.10 Iconos Puerta nuevos en .png

➤ **Detector de movimiento**

Estado	Desconocido	Presencia no detectada	Presencia detectada
Icono			

Tabla 3.11 Iconos Detector Movimiento antiguos en .jpg

Estado	Desconocido	Presencia no detectada	Presencia detectada
Icono			

Tabla 3.12 Iconos Detector Movimiento nuevos en .png

➤ **Sensor de temperatura**

Estado	Presente en la estancia
Icono	

Tabla 3.13 Icono Sensor Temp. antiguo en .jpg

Estado	Presente en la estancia
Icono	

Tabla 3.14 Icono Sensor Temp. nuevo en .png

➤ **Sensor combinado**

Estado	Presente en la estancia
Icono	

Tabla 3.15 Icono Sensor Comb. nuevo en .png

➤ **Sensor de inundación**

Estado	Presente en la estancia	Fuga detectada	Fuga no detectada
Icono			

Tabla 3.16 Icono Sensor In. antiguo en .jpg

Estado	Presente en la estancia	Fuga detectada	Fuga no detectada
Icono			

Tabla 3.17 Icono Sensor In. nuevo en .png

➤ **Contadores**

Estado	Presente en la estancia
Icono	

Tabla 3.18 Icono Contadores nuevo en .png

➤ **Iconos de menú**

Antiguo	Icono	Nuevo
	Abrir fichero contadores	
	Abrir proyecto	
	Acerca de	

	Administrar contadores	
	Ver configuración actual	
	Conectar Bus EIB	
	Editar archivo de configuración	
	Cerrar Proyecto	
	Desconectar Bus EIB	
	Escribir	
	Gráficas	
	Guardar Proyecto	
	Nuevo Proyecto	
	Salir	

Tabla 3.19 Iconos de Menú

3.2.11 Entorno gráfico mejorado

Siguiendo las pautas de mejora gráfica y visual, se ha procedido a un nuevo modelo de diseño de interfaz que hace que nuestro entorno sea algo más moderno, y además se muestra igual para cualquier sistema operativo en el que se ejecute nuestra aplicación.

En las siguientes figuras (**Figura 3.9** y **Figura 3.10**) podemos apreciar los cambios en la interfaz al comparar la vista antigua con la actual.



Figura 3.15 Interfaz principal antigua



Figura 3.16 Interfaz principal nueva

Normalmente las ventanas Java tienen su propio aspecto, su propio estilo de botones y demás. Java cuenta con 2 librerías para diseñar interfaces gráficas (ventanas, botones, etiquetas, etc.), estas 2 librerías son: **AWT** y **Swing** [9].

Las ventanas hechas con **AWT** conservan el tema del Sistema Operativo, sin embargo, las ventanas hechas con **Swing** son muy fáciles de cambiar su aspecto (look and

feel o skin) para que tengan aspecto java, aspecto windows, aspecto Linux, etc. Basta con tener la librería adecuada y unas líneas de código.

En cuanto a la apariencia de la vista, nos ha parecido oportuno cambiar el estilo en el que se muestran las diferentes interfaces de la aplicación utilizando un aspecto de Java llamada *Look and Feel*, el cual provee una serie de métodos para modificar el aspecto de los componentes visuales de una manera rápida y sencilla.

En este caso se ha implementado un estilo de *Look and Feel* llamado *Nimbus*:

```
try {
    for (LookAndFeelInfo info : UIManager.getInstalledLookAndFeels()) {
        if ("Nimbus".equals(info.getName())) {
            UIManager.setLookAndFeel(info.getClassName());
            break;
        }
    }
} catch (Exception e) {
}
```

3.3 Resumen del Capítulo

En este capítulo hemos tratado las modificaciones que se han realizado sobre el código fuente de la aplicación SKoA para mejorar su eficiencia y su entorno gráfico y visual.

Además, esta optimización permitirá a futuros desarrolladores modificar el código de forma más sencilla y rápida, dado que se ha ordenado la estructuración y se han realizado comentarios que guiarán al usuario y le ayudarán a conocer de antemano las funciones y responsabilidades de cada apartado.

Capítulo 4:

Publicación de SKoA 1.0 como Software Libre

4.1 Introducción

Software libre es la denominación del *software* que respeta la libertad de todos los usuarios que adquirieron el producto y, por tanto, una vez obtenido el mismo, puede ser usado, copiado, estudiado, modificado, y redistribuido libremente de varias formas. Según su principal impulsora, la organización Free Software Foundation, el software libre se refiere a la seguridad de los usuarios para ejecutar, copiar, distribuir y estudiar el software, e incluso modificarlo y distribuirlo modificado.

Un programa informático es software libre si otorga a los usuarios todas estas libertades de manera adecuada. De lo contrario no es libre. Existen diversos esquemas de distribución que no son libres, y si bien podemos distinguirlos en base a cuánto les falta para llegar a ser libres, su uso bien puede ser considerado contrario a la ética en todos los casos por igual [10].

4.2 Consideraciones iniciales

En el movimiento del software libre se sostiene que los usuarios de ordenadores deben tener la libertad para cambiar y redistribuir el software que usan. El calificativo “*free*” en la expresión “*free software*” se refiere a la libertad: los usuarios tienen la libertad para ejecutar, modificar y redistribuir el software.

El software libre contribuye al conocimiento humano, al contrario del software privativo. Por tanto, las universidades deben impulsar el software libre por el bien del progreso del saber humano, como así también alentar a los científicos y estudiantes a publicar sus trabajos.

La aplicación sobre la que se trata en este proyecto ya trabaja con software libre, por lo que es adecuado y recomendable seguir en la misma línea en la que se dio inicio en proyectos anteriores.

Un programa es software libre si los usuarios tienen las cuatro libertades esenciales:

- La libertad de ejecutar el programa como se desea, con cualquier propósito (Libertad 0).
- La libertad de estudiar cómo funciona el programa, y modificarlo para adaptarlo a las propias necesidades (Libertad 1). El acceso al código fuente es una condición necesaria para ello.
- La libertad de redistribuir copias para ayudar a otros usuarios (Libertad 2).
- La libertad de distribuir copias de sus versiones modificadas a terceros (Libertad 3). Esto le permite ofrecer a toda la comunidad la oportunidad de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello [11].

4.3 Ventajas e inconvenientes del software libre

Las principales ventajas del software libre son:

- Bajo costo de adquisición
- Innovación tecnológica
- Independencia del proveedor
- Escrutinio público
- Adaptación del software
- Idiomas
- Fomento de la industria

Las principales desventajas que podemos encontrar son:

- Falta de control de calidad y garantías formales
- Conocimientos técnicos necesarios
- Desconfianza y resistencia al cambio

- El desarrollo del software requiere importantes inversiones
- Dificultad en el intercambio de archivos
- Menor compatibilidad con el hardware

4.4 Tipos de Licencia de Software Libre

En este apartado se compararemos varias licencias de software libre para diagnosticar cuál sería la mejor opción según nuestras circunstancias actuales.

Licencia	Características
<p data-bbox="204 766 635 797">Berkeley Software Distribution</p> 	<ul style="list-style-type: none"> ▪ Solamente exige que los derechos de autor y las resignaciones de garantía sean incluidas en todas las distribuciones del software ▪ Sin copyleft
<p data-bbox="341 1227 497 1258">Apache 2.0</p> 	<ul style="list-style-type: none"> ▪ Permite usar el software para cualquier propósito, distribuirlo, modificarlo y distribuir las modificaciones. ▪ Sin copyleft: No requiere que las versiones modificadas tengan que ser distribuidas como software libre. ▪ Señalar los cambios en los ficheros que se hayan modificado. ▪ Compatible con GPLv3, no compatible con las versiones anteriores de la licencia GPL. ▪ Provisiones de protección respecto a patentes
<p data-bbox="220 1675 619 1738">Python Software Foundation License</p> 	<ul style="list-style-type: none"> ▪ Sin copyleft: No requiere que las versiones modificadas tengan que ser distribuidas como software libre. ▪ Compatible con GPL

<p>General Public License v.3</p> 	<ul style="list-style-type: none"> ▪ Permiso ilimitado para ejecutar el programa ▪ Permite redistribuir el código objeto, siempre acompañado de las fuentes correspondientes ▪ No obliga a divulgar modificaciones hechas en el ámbito privado. ▪ El software carece de garantía y la licencia tiene una cláusula de limitación de responsabilidad. ▪ Se han de mantener intactos todos los textos relativos a los derechos de copyright sobre el código y la licencia. ▪ Incluye cláusulas de protección respecto a demandas por infracción de patentes
<p>Eclipse Public License</p> 	<ul style="list-style-type: none"> ▪ No requiere ningún seguimiento en los cambios ▪ Sólo exige la publicación del código fuente cuando las modificaciones se consideran un trabajo derivado y no una extensión o un módulo separado ▪ Los trabajos derivados deben ser publicados siempre bajo la licencia EPL
<p>Open Secure Sockets Layer</p> 	<ul style="list-style-type: none"> ▪ Garantía contra patentes ▪ Sin copyleft: No requiere que las versiones modificadas tengan que ser distribuidas como software libre
<p>Mozilla Public License</p> 	<ul style="list-style-type: none"> ▪ Exige la publicación del código fuente y que cualquier derivado o extensión sea publicado bajo la misma licencia MPL ▪ Permite que módulos libres puedan convivir con módulos privativos ▪ No requiere que los cambios realizados sean reflejados en el código fuente ▪ Los derechos de autor deben mantenerse tanto

	en el código fuente como en los binarios
--	--

Tabla 4.1 Comparativa de licencias de software libre [12]

4.5 Decisión final

Guiados por las características anteriores, debemos tomar una decisión sobre qué licencia será la que mejor defina las necesidades de nuestro proyecto.

SKoA debe ser un software al que cualquier usuario tenga acceso y permiso para realizar modificaciones a su libre elección, incluso para uso particular si así lo desea. Además debemos asegurar que no abandone la línea del software libre para que todas aquellas mejoras que se realicen sobre la aplicación puedan ser disfrutadas por toda la comunidad.

Además debemos de tener en cuenta que la librería Calimero, en la que se basa prácticamente SKoA, cuenta con una licencia libre GPL, por lo tanto debemos seleccionar una licencia que sea compatible con la misma.

La licencia GPLv3 cumple con todos los requisitos que necesitamos para la publicación de nuestro software. Si le añadimos que dicha licencia cuenta con cláusulas de protección respecto a demandas por infracción de patentes o que todas las versiones que surjan deben seguir siendo de código abierto, las ventajas siguen en aumento.

4.6 Licencia Pública General de GNU en su versión 3 (GPLv3)

En esta sección nos internaremos en los términos y condiciones de la licencia GPLv3 de GNU y realizaremos una guía rápida de sus principales características, así como de sus fundamentos básicos.

4.6.1 Fundamentos de la GPLv3

Nadie debe ser restringido por el software que utilizan. Hay cuatro libertades que cada usuario debe tener:

- la libertad de usar el software para cualquier propósito
- la libertad de cambiar el software para satisfacer sus necesidades
- la libertad para compartir el software con sus amigos y vecinos
- la libertad para compartir los cambios que realice

Como se indicó anteriormente en el apartado 4.2 *Consideraciones iniciales* de este capítulo, cuando un programa ofrece a los usuarios todas estas libertades, lo llamamos software libre.

Los desarrolladores que escriben software pueden crear licencias bajo los términos de la GNU GPL. Cuando lo hacen, será software libre y permanecerá siendo software libre, sin importar quién cambia o distribuye el programa. A esto le llamamos *copyleft*: el software tiene copyright, pero en lugar de utilizar esos derechos para restringir a los usuarios como el hace software privativo, los utilizan para asegurar la libertad de cada usuario.

En la última versión de la licencia GPL de GNU (GPLv3) se protege al copyleft de ser delimitado por los desarrollos legales o tecnológicos. La versión más reciente protege a los usuarios de tres amenazas recientes:

- **Tivoización:** Algunas empresas han creado varios tipos diferentes de dispositivos que ejecutan el software GPL, y luego el hardware es manipulado para que pueda cambiar el software que se está ejecutando, pero en esta versión no puede. Si un dispositivo puede ejecutar software de forma arbitraria, quiere decir que es un ordenador de propósito general, y su dueño debe controlar lo que hace. Cuando un dispositivo impide hacer eso, se le denomina *tivoización*.
- **Leyes que prohíben el software libre:** Legislaciones como la Digital Millennium Copyright Act y la Directiva de Derechos de Autor de la Unión Europea hacen que sea un delito escribir o compartir software que puede romper el DRM (Gestión

Digital de Restricciones). Estas leyes no deben interferir con los derechos que la GPL concede.

- **Ofertas de patentes discriminatorias:** Algunas empresas han declarado que no van a demandar a los usuarios de software libre por infracción de patentes, como ocurre siempre que se obtiene el software de un proveedor que financiado por empresas concretas. En última instancia, estas empresas están tratando de recaudar regalías por el uso de software libre, lo que interfiere de foma directa en la libertad de los usuarios. Ninguna empresa debe tener la capacidad de hacer esto.

Esta tercera versión también tiene una serie de mejoras para que la licencia sea más fácil de usar y entender por toda la comunidad. Pero aún con todos estos cambios, la GPLv3 no es una nueva licencia radical; sino que es una evolución de la versión anterior. Aunque una gran cantidad de texto ha cambiado, gran parte de él simplemente aclara lo definido en GPLv2. Con esto en mente, vamos a revisar los principales cambios en la GPLv3, y traremos las mejoras de la licencia para usuarios y desarrolladores [13].

4.6.2 Neutralización de las leyes que prohíben el software libre

Probablemente somos muchos los que estamos familiarizados con las Restricciones de Gestión Digital (DRM) en DVD y otros medios de comunicación, al igual que también estaremos familiarizados con las leyes que hacen que sea ilegal escribir sus propias herramientas para eludir esas restricciones, como la Digital Millennium Copyright Act y la Directiva de la Unión Europea Derecho de Autor. Nadie debería dejar de escribir cualquier código que desee y la licencia GPLv3 se encarga de proteger este derecho.

Siempre es posible usar código GPL para escribir software que implementa DRM. Sin embargo, si alguien hace eso con código protegido por la GPLv3, en el apartado 3 (véase 4.7 *Términos y Condiciones*) se indica que el sistema no contará como una medida de "protección" tecnológicamente eficaz. Esto significa que si se quebranta el DRM, el usuario será libre para distribuir su propio software, y no se verá amenazado por la DMCA o leyes similares.

Como de costumbre, la GPL de GNU no restringe lo que se hace en el software; sólo impide que se restrinja a otros.

4.6.3 Protección del derecho de modificación

La tivoización es un peligroso intento de restringir la libertad de los usuarios: el derecho de modificar su software no tendría sentido si ninguno de sus equipos le permiten hacerlo. GPLv3 detiene la tivoización al exigir al distribuidor ofrecer toda la información o los datos necesarios para instalar software modificado en cualquier dispositivo.

Esto puede ser tan simple como un conjunto de instrucciones, o se pueden incluir datos especiales, tales como claves criptográficas o información sobre cómo eludir un control de integridad en el hardware. Dependerá de cómo ha sido diseñado el hardware, pero no importa qué tipo de información necesita el usuario: éste debería ser capaz de conseguirla sin mayores obstáculos.

Este requisito tiene un alcance limitado. Todavía se les permite a los distribuidores utilizar claves criptográficas para cualquier propósito, y sólo estarán obligados a revelar una clave si es necesaria para modificar el software GPL en el dispositivo que dado.

El Proyecto GNU en sí utiliza GnuPG para probar la integridad de todo el software en su sitio FTP, y medidas beneficiosas para los usuarios.

GPLv3 no bloquea el uso de la criptografía; sino que directamente no quiere que se haga. Sólo se impide que se retiren derechos de licencia, ya sea a través de la ley de patentes, tecnología, o cualquier otro medio.

4.6.4 Protección fuerte contra amenazas de patentes

Las licencias de software libre han desarrollado nuevas estrategias para abordar el tema de las patentes. GPLv3 también refleja estos cambios. Cada vez que un usuario licencia un software por la GPLv3, que han escrito o modificado, deben proporcionar a cada destinatario con licencia alguna patente para ejercer los derechos que la GPL le

otorga. Además de eso, si otro usuario con licencia trata de usar patentes para detener a otro usuario de ejercer esos derechos, se dará por terminada su licencia.

Esto significa que usuarios y desarrolladores serán capaces de trabajar con software cubierto por la GPLv3 sin preocuparse de que un contribuyente trate de demandarlos por infracción de patente en un futuro. Con estos cambios, la GPLv3 ofrece a sus usuarios más defensas contra la agresión de patentes que cualquier otra licencia de software libre.

4.6.5 Compatibilidad con otras licencias libres

Además compatibilizar con otras licencias GPL, GPLv3 también es compatible con algunas otras licencias.

La licencia *Apache 2.0* es un buen ejemplo. Gran parte del software libre está disponible bajo esta licencia, debido a las grandes comunidades con las que cuenta. Esperamos que este cambio en la GPLv3 fomente una mayor cooperación e intercambio dentro de la comunidad del software libre. La tabla a continuación ayuda a ilustrar algunas relaciones de compatibilidad comunes entre diferentes licencias de software libre:

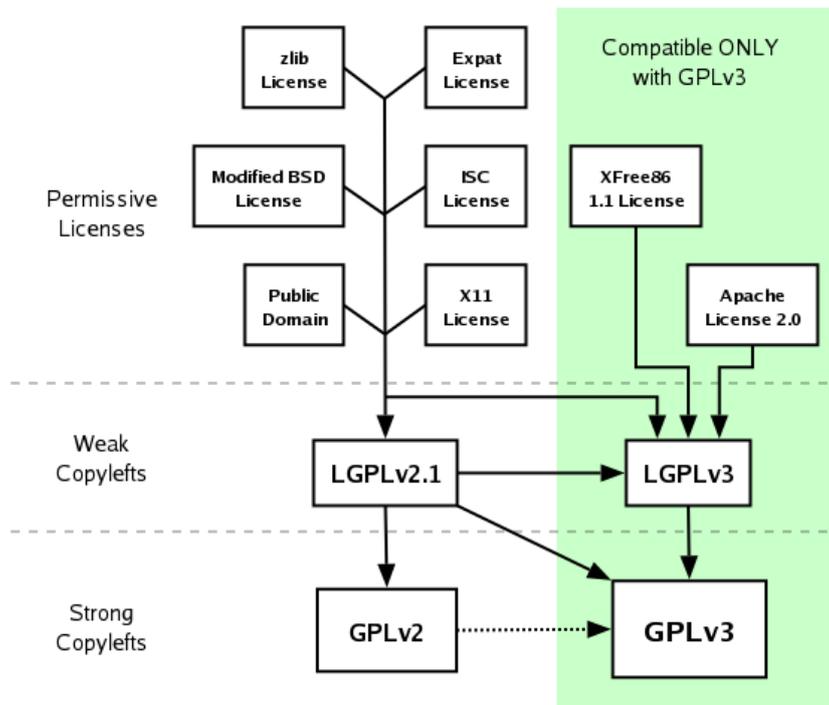


Figura 4.1 Gráfico que ilustra las compatibilidades de GPLv3 con otras licencias

En el gráfico de la figura anterior, observamos que las flechas que señalan de una licencia a otra indican que la primera licencia es compatible con la segundo.

Esto es cierto incluso si se siguen varias flechas para obtener de una licencia a la otra; así, por ejemplo, la licencia ISC es compatible con la GPLv3. GPLv2 es compatible con la GPLv3 si el programa le permite elegir "cualquier versión posterior" de la GPL, que es el caso para la mayoría del software liberado bajo esta licencia.

Este diagrama no es exhaustivo, pero claramente ilustra que la GPLv3 es compatible con casi toda la GPLv2, y algunas más.

4.6.6 Una licencia de carácter global

En versiones anteriores de GPL se trata largo y tendido el término "distribución".

La licencia nunca indica qué es la distribución, ya que el término fue tomado de la ley de copyright de Estados Unidos. Se esperaba que los jueces acudieran a esta ley para la definición. Sin embargo, después nos enteramos de que las leyes de derechos de autor en otros países utilizan la misma palabra, pero le dan diferentes significados. Debido a esto, un juez en un país podría analizar, por ejemplo GPLv2, de forma diferente que un juez en los Estados Unidos.

GPLv3, sin embargo, utiliza un nuevo término, "transportar" (*"convey"*), y ofrece una definición de ese término. "Transportar" tiene el mismo significado que se le pretendía otorgar al término "distribuir", pero ahora que esto se explica directamente en la licencia, será más sencillo para los usuarios de todo el mundo entender esta definición. Hay otros cambios menores a lo largo de la licencia que también ayudarán a asegurar que se aplique de forma coherente en todo el mundo.

4.7 Términos y Condiciones de GPLv3

LICENCIA PÚBLICA GENERAL GNU

Versión 3, 29 de junio 2007

Copyright © 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Se permite la copia y distribución de copias literales de este documento de licencia, pero cambiándolo no está permitido.

Preámbulo

La Licencia Pública General de GNU es una licencia copyleft libre para software y otro tipo de trabajos.

Las licencias para la mayoría del software y otros trabajos prácticos están diseñadas para quitarle a usted la libertad de compartir y modificar los trabajos. Por el contrario, la Licencia Pública General de GNU pretende garantizarle la libertad de compartir y modificar todas las versiones de un programa - para asegurarse de que sigue siendo software libre para todos sus usuarios. Nosotros, la Fundación para el Software Libre, usamos la Licencia Pública General GNU para la mayoría de nuestro software; se aplica también a cualquier otro trabajo publicado esta forma por sus autores. Usted puede aplicarla a sus propios programas.

Cuando hablamos de software libre, nos referimos a la libertad, no de precio. Nuestras Licencias Públicas Generales están diseñadas para asegurarnos de que usted tiene la libertad de distribuir copias de software libre (y cobrar por ellos si lo desea), que reciba el código fuente o que pueda conseguirlo si lo quiere, de que pueda modificar el software o utilizar fragmentos de él en nuevos programas libres, y que usted sabe que puede hacer estas cosas.

Para proteger sus derechos, necesitamos evitar que otros le nieguen estos derechos o pedirle que renuncie a los derechos. Por lo tanto, usted tiene ciertas responsabilidades si distribuye copias del software, o si lo modifica: responsabilidades que persiguen respetar la libertad de los demás.

Por ejemplo, si distribuye copias de un programa, ya sea gratuitamente o por una tarifa, debe transmitir a los destinatarios los mismos derechos que usted recibió. Debe asegurarse de que ellos también reciben, o pueden conseguir el código fuente. Y debe mostrarles estas condiciones de forma que conozcan sus derechos.

Desarrolladores que utilizan la GNU GPL protegen sus derechos con dos pasos: afirman los derechos de autor del software, y le ofrecemos esta licencia que le da permiso legal para copiar, distribuir y / o modificarlo.

Para la protección de los desarrolladores y autores, la GPL explica claramente que no hay ninguna garantía para este software libre. Por el bien de ambos, usuarios y autores, la GPL establece que las versiones modificadas serán marcados como cambiado, por lo que sus problemas no sea atribuido por error a los autores de versiones anteriores.

Algunos dispositivos están diseñados para negar al usuario para instalar o ejecutar versiones modificadas del software dentro de ellos, a pesar de que el fabricante pueda hacerlo. Esto es fundamentalmente incompatible con el objetivo de proteger la libertad de los usuarios para modificar el software. El patrón sistemático de tal abuso se produce en el área de productos para particulares de usar, que es precisamente donde es más inaceptable. Por lo tanto, hemos diseñado esta versión de la GPL para prohibir la práctica de dichos productos. Si apareciesen problemas similares en otros ámbitos, estaremos preparados para extender estas prestaciones a las de las futuras versiones de la GPL, según sea necesario para proteger la libertad de los usuarios.

Por último, todo programa está constantemente amenazado por las patentes de software. Los Estados no deben permitir que las patentes restrinjan el desarrollo y uso de software en ordenadores de uso general, pero en los que sí, deseamos evitar el peligro especial que las patentes aplicadas a un programa libre puedan hacerlo efectivamente propietario. Para evitar esto, la GPL establece que las patentes no se pueden utilizar para hacer que el programa no libre.

Los términos y condiciones para la copia, distribución y modificación son los siguientes:

TÉRMINOS Y CONDICIONES

0. Definiciones.

"Esta Licencia" se refiere a la versión 3 de la Licencia Pública General de GNU.

"Derechos de autor" también significa las leyes del derecho de autor como el que se aplican a otros tipos de obras, como las máscaras de semiconductores.

"El Programa" se refiere a cualquier obra sujeta al derecho licenciado bajo esta licencia. Cada concesionario se dirige como "usted". "Licenciarios" y "destinatarios" pueden ser personas u organizaciones.

Para "modificar" un trabajo significa copiar o adaptar todo o parte de la obra de un modo que requiera permiso de copyright, que no sea la realización de una copia exacta. El trabajo resultante se denomina "versión modificada" de la obra anterior o trabajo "basado en" el trabajo anterior.

Un "trabajo amparado" puede ser tanto el Programa no modificado como un trabajo basado en el Programa.

Para "propagar" un trabajo significa hacer cualquier cosa con él que, sin permiso, que le haga directa o indirectamente responsable de infringir las leyes de copyright aplicables, excepto ejecutarlo en una computadora o la modificación de una copia privada. Propagación incluye la copia, distribución (con o sin modificaciones), poner a disposición del público, y en algunos países de otras actividades también.

"Distribuir" un trabajo se entiende cualquier tipo de difusión que permite a otras partes para hacer o recibir copias. La mera interacción con un usuario a través de una red informática, sin transferencia de una copia, no está transmitiendo.

Un usuario interactivo interfaz muestra "Avisos Legales Apropriados" en la medida en que incluye una herramienta de gran utilidad y destacadas que muestra un anuncio de copyright, y le dice al usuario que no hay ninguna garantía para el trabajo (excepto para la medida en que se proporcionan garantías), que los licenciarios pueden transmitir el trabajo bajo esta Licencia, y cómo ver una copia de esta licencia. Si la interfaz presenta una lista de comandos u opciones de usuario, como un menú, un elemento destacado en la lista

cumple con este criterio.

1. Código Fuente.

El "código fuente" de un trabajo se entiende la forma preferida de la obra para realizar modificaciones a la misma. "Código objeto" significa cualquier forma no fuente de un trabajo.

Una "Interfaz Estándar" se refiere a una interfaz que sea es un estándar oficial definido por un organismo de normalización reconocido, o bien, en el caso de las interfaces especificadas para un lenguaje de programación particular, uno que se utiliza ampliamente entre los desarrolladores que trabajan en ese idioma.

Las "Bibliotecas de Sistema" de un trabajo ejecutable incluyen nada, aparte de la obra en su conjunto, que (a) se incluye en la forma normal de envasado de un componente principal, pero que no es parte de ese componente principal, y (b) sólo sirve para permitir el uso de la obra con ese componente principal, o para implementar un Interfaz Estándar para el cual una aplicación está disponible al público en forma de código fuente. Un "componente principal", en este contexto, significa un importante componente esencial (núcleo, sistema de ventanas, etc.) del sistema operativo específico (si existe) en la que se ejecuta el trabajo ejecutable, o un compilador utilizado para producir la obra, o un intérprete del código objeto utilizado para ejecutarlo.

La "Fuente Correspondiente" de un trabajo en forma de código objeto significa todo el código fuente necesario para generar, instalar, y (para un trabajo ejecutable) ejecutar el código objeto y modificar el trabajo, incluyendo secuencias de comandos para controlar esas actividades. Sin embargo, no incluye el Sistema de Bibliotecas de la obra, o herramientas de propósito general o programas gratuitos disponibles en general que se utilizan sin modificaciones en la realización de esas actividades, pero que no forman parte de la obra. Por ejemplo, la Fuente Correspondiente incluye los archivos de definición de interfaz asociados con archivos de origen para el trabajo, y el código fuente de las bibliotecas compartidas y subprogramas enlazados dinámicamente que el trabajo se ha diseñado específicamente para exigir, como por la comunicación de datos intrínseca o el control de flujo entre esos subprogramas y otras partes de la obra.

La Fuente Correspondiente no necesita incluir nada que los usuarios pueden

regenerar automáticamente de otras partes de la Fuente Correspondiente.

La Fuente Correspondiente de un trabajo en código fuente es ese mismo trabajo.

2. Permisos básicos.

Se otorgan todos los derechos concedidos en virtud de esta licencia por el término de los derechos de autor en el Programa, y son irrevocables siempre que se cumplan las condiciones establecidas. Esta Licencia afirma explícitamente su permiso ilimitado para ejecutar el Programa sin modificaciones. El resultado de la ejecución de un trabajo amparado está cubierto por esta licencia sólo si la salida, por su contenido, constituya un trabajo amparado. Esta Licencia reconoce sus derechos de uso razonable u otro equivalente, según lo dispuesto por la ley de derechos de autor.

Usted puede hacer, ejecutar y difundir trabajos amparados que no distribuya, sin condiciones, siempre y cuando su licencia queda de otra manera en vigor. Usted podrá distribuir trabajos amparados a terceros con el único propósito de que ellos hagan modificaciones exclusivamente para usted, o le proporcionará instalaciones para el funcionamiento de dichas obras, siempre y cuando cumpla con los términos de esta licencia en la transmisión de todos los materiales para los que no controlas los derechos de autor. Aquellos haciendo así o correr los trabajos cubiertos para usted debe hacerlo exclusivamente en su nombre, bajo su dirección y control, en los términos que les prohíban realizar copias de su material con derechos de autor fuera de su relación con usted.

Transmitir en cualquier otra circunstancia se permite únicamente bajo las condiciones indicadas a continuación. Sublicencia no está permitido; la sección 10 lo hace innecesario.

3. Protección de Derechos Legales de los Usuarios frente a Leyes contra la elusión.

Ningún trabajo amparado debe considerarse parte de una medida tecnológica efectiva bajo cualquier ley aplicable que cumpla las obligaciones que le impone el artículo 11 del tratado de copyright WIPO adoptado el 20 de diciembre de 1996, o leyes similares que prohíben o restringen la burla de tales medidas.

Cuando distribuya un trabajo amparado, renuncia a cualquier poder legal para

prohibir la elusión de medidas tecnológicas en la medida tales burlas se realicen en ejercicio de derechos bajo esta Licencia con respecto al trabajo amparado, y usted renuncia cualquier intención de limitar el funcionamiento o modificación de la trabajar como un medio de hacer cumplir, en contra de los usuarios de la obra, sus derechos legales o de terceros para prohibir la elusión de medidas tecnológicas.

4. Distribución de copias literales.

Usted podrá distribuir copias literales del código fuente del Programa, según lo has recibido, en cualquier medio, siempre que la adecuada y bien visible publique en cada copia un anuncio de copyright; mantener intactos todos los avisos que indican que esta Licencia y cualquier término no permisivas añadido de acuerdo con la sección 7 se aplican al código; mantener intactos todos los avisos de la ausencia de cualquier garantía; y dar a todos los destinatarios una copia de esta Licencia junto con el Programa.

Puede cobrar cualquier precio o ningún precio por cada copia que distribuya, y podrá ofrecer soporte o garantía de protección para un honorario.

5. Distribución de Versiones Modificadas de Código.

Usted podrá distribuir un trabajo basado en el Programa, o las modificaciones para producirlo a partir del Programa, en forma de código fuente bajo los términos del artículo 4, supuesto que además cumpla las siguientes condiciones:

a) El trabajo debe incluir avisos destacados indicando que usted lo ha modificado y dando una fecha pertinente.

b) El trabajo debe incluir avisos destacados indicando que es liberado bajo esta Licencia y cualquier condición añadida en virtud del artículo 7. Este requisito modifica el requisito de la sección 4 de "mantener intactos todos los avisos".

c) Debe licenciar toda la obra, en su conjunto, bajo esta licencia a cualquier persona que esté en posesión de una copia. Por tanto, esta Licencia se aplicará junto con cualquier sección 7 términos adicionales aplicables, a la totalidad de la obra, y todas sus partes, con independencia de la forma en que se empaquetan. Esta licencia no da permiso para licenciar la obra de cualquier otra manera, pero no se anula dicho permiso si usted ha

recibido por separado.

d) Si el trabajo tiene interfaces de usuario interactivos, cada uno debe mostrar Avisos Legales Apropriados; Sin embargo, si el Programa tiene interfaces interactivos que no muestran Avisos Legales Apropriados, su trabajo no necesita que lo hagan.

Una recopilación de un trabajo amparado con otros trabajos separados e independientes, que no son por sus extensiones naturales del trabajo amparado, y que no se combinan con él como para formar un programa más amplio, en o sobre un volumen de almacenamiento o distribución medio, se llama un "agregado" si la recopilación y su copyright resultante no se utilizan para limitar el acceso o los derechos legales de los usuarios de la compilación más allá de lo que permiten las obras individuales. La inclusión de un trabajo amparado en un paquete no hace esta licencia se aplique a las otras partes del conjunto.

6. Distribución No Fuente.

Usted podrá distribuir un trabajo amparado en forma de código objeto bajo los términos de las secciones 4 y 5, siempre que también distribuya las Fuentes Correspondientes máquina, de acuerdo a los términos de esta Licencia, en una de las siguientes maneras:

a) Distribuir el código objeto en, o embebido en, un producto físico (incluyendo medios de distribución físicos), acompañado de las Fuentes Correspondientes en un medio físico duradero utilizado habitualmente para el intercambio de software.

b) Distribuir el código objeto en, o embebido en, un producto físico (incluyendo medios de distribución físicos), acompañado de una oferta por escrito, válida durante al menos tres años y válida durante el tiempo que usted ofrece piezas de repuesto o de atención al cliente para que modelo del producto, para dar cualquier persona que posea el código objeto, ya sea una copia de la Fuente Correspondiente para todo el software en el producto que está cubierta por esta Licencia, en un medio físico duradero habitual para el intercambio de software, por un precio no más que su costo razonable de realizar físicamente las fuentes, o acceso para copiar las fuentes correspondientes desde un servidor de red sin coste.

c) Distribuir copias individuales del código objeto junto con una copia de la oferta por escrito para proporcionar la Fuente Correspondiente. Esta alternativa se permite sólo ocasionalmente y no comercial y sólo si usted recibió el código objeto con tal oferta, de acuerdo con el inciso 6b.

d) Distribuir el código objeto ofreciendo acceso desde un lugar determinado (gratuitamente o por un cargo), y ofrecer acceso equivalente a las Fuentes Correspondientes de la misma manera por el mismo lugar sin cargo adicional. Usted no tiene que requiere receptores para copiar las Fuentes Correspondientes junto con el código objeto. Si el lugar para copiar el código objeto es un servidor de red, la Fuente Correspondiente puede estar en un servidor diferente (gestionado por usted o un tercero) que soporta instalaciones de copia equivalentes, siempre que mantenga instrucciones claras junto al código objeto diciendo dónde encontrar la Fuente Correspondiente. Independientemente de qué servidor aloja las Fuentes Correspondientes, usted seguirá obligado a asegurar que está disponible durante todo el tiempo que sea necesario para satisfacer estos requisitos.

e) Distribuir el código objeto mediante la transmisión punto a punto, siempre que informe a otros usuarios dónde se ofrecen el código objeto y las Fuentes Correspondientes de la obra al público en general sin costo alguno en la subcláusula 6d.

Una parte separable del código objeto, cuyo código fuente está excluido de las Fuentes Correspondientes como Biblioteca de Sistema, no necesita ser incluido en la transmisión de la obra de código objeto.

Un "Producto de Usuario" es un "producto de consumo", lo que significa que cualquier propiedad personal tangible que normalmente se utiliza para propósitos familiares, personales o domésticos, o cualquier cosa diseñada o vendida para su incorporación en una vivienda. Para determinar si un producto es un producto de consumo, los casos dudosos se resolverán en favor de la cobertura. Para un producto particular, recibida por un usuario en particular, "que se utiliza normalmente" se refiere a un uso típico o común de ese tipo de producto, independientemente de la situación del usuario particular o de la forma en que el usuario particular utiliza realmente o de espera o se espera que utilizar, el producto. Un producto es un producto de consumo independientemente de si el producto tiene usos sustanciales comerciales, industriales o no

de consumo, a menos que tales usos representen la única forma posible de utilizar el producto.

"Información de instalación" para un Producto de Usuario significa cualquiera de los métodos, procedimientos, llaves de autorización u otra información necesarios para instalar y ejecutar versiones modificadas de un trabajo amparado en ese Producto de Usuario a partir de una versión modificada de su Fuente Correspondiente. La información debe ser suficiente para asegurar que el funcionamiento continuo del código objeto modificado en ningún caso o intromisión por el solo hecho de haber sido modificado.

Si usted transmite una obra de código objeto bajo esta sección en, o con, o específicamente para su uso en, un producto de usuario, y el transporte se produce como parte de una transacción en la que el derecho de posesión y uso del Producto de Usuario se transfiere a la destinatario a perpetuidad o por un período determinado (independientemente de cómo se caracteriza la transacción), la Fuente Correspondiente transmitida en virtud de esta sección debe ir acompañada de la información de instalación. Pero este requisito no se aplica si ni usted ni ningún tercero conserva la posibilidad de instalar el código objeto modificado en el Producto de usuario (por ejemplo, el trabajo ha sido instalado en la ROM).

El requisito de proporcionar información de instalación no incluye la obligación de continuar proporcionando servicio de soporte, garantía, o actualizaciones para un trabajo que ha sido modificado o instalado por el destinatario, o para el Producto de Usuario en el que ha sido modificado o instalado. Acceso a una red puede ser denegado cuando la propia modificación afecte materialmente y de forma adversa el funcionamiento de la red o viole las reglas y protocolos para la comunicación a través de la red.

Correspondiente Fuente transmitió, y Información de instalación proporcionado, de acuerdo con esta sección debe estar en un formato que se documenta públicamente (y con una implementación disponible para el público en código fuente), y debe requerir ninguna clave o contraseña especial para el desembalaje, la lectura o copia.

7. Condiciones adicionales.

"Permisos Adicionales" son condicionantes que amplían los términos de esta Licencia permitiendo excepciones a una o más de sus condiciones. Los permisos

adicionales que son aplicables a todo el programa se tratarán como si estuviesen incluidos en esta Licencia, en la medida en que son válidas bajo la ley aplicable. Si los permisos adicionales se aplican sólo a una parte del Programa, esa parte se puede utilizar por separado bajo los permisos, pero el Programa completo seguiría estando afectado por esta Licencia con independencia de los permisos adicionales.

Cuando distribuya una copia de un trabajo amparado, es posible que a su opción eliminar cualquier permiso adicional de esa copia, o de cualquier parte de ella. (Permisos adicionales pueden ser escritos para requerir su propia eliminación en ciertos casos cuando se modifica el trabajo.) Usted puede establecer permisos adicionales en material añadido por usted a un trabajo amparado, para el que tiene o puede otorgar un permiso apropiado de derechos de autor.

No obstante cualquier otra disposición de esta Licencia, para el material que añada a un trabajo amparado, usted puede (si está autorizado por los propietarios del copyright de ese material) añadir condiciones a esta Licencia con los términos:

- a) Ausencia de garantía o limitación de responsabilidad diferente de los términos de los artículos 15 y 16 de esta Licencia; o
- b) Obligación de mantener determinados avisos legales razonables o atribuciones de autoría en el material o en los Avisos Legales Correspondientes mostrados por los trabajos que lo contengan; o
- c) Prohibir la tergiversación del origen de ese material, o requerir que las versiones modificadas de este tipo de material se marcarán de forma razonable como diferentes de la versión original; o
- d) Limitar el uso con fines publicitarios de los nombres de los autores o de los materiales; o
- e) Negarse a ofrecer derechos bajo la ley de marcas para el uso de algunos nombres comerciales, marcas registradas, o marcas de servicio; o
- f) Exigir la indemnización de los licenciantes y autores de ese material por cualquier persona que distribuya el material (o versiones modificadas del mismo) con obligaciones

contractuales de responsabilidad para el receptor, para cualquier responsabilidad que estas obligaciones contractuales impongan directamente sobre los licenciantes y autores.

Todos los demás términos adicionales no-permisivas son consideradas "otras restricciones" en el sentido del artículo 10. Si el Programa, tal como usted lo recibió, o cualquier parte de ella, contiene un aviso indicando que se rige por esta Licencia junto con un término que es una restricción adicional, usted puede quitar ese término. Si un documento de licencia contiene una restricción adicional pero permite la renovación de licencias o de transporte bajo esta Licencia, puede añadir a un material de trabajo cubierto regido por los términos de ese documento de licencia, siempre que la restricción no se mantenga dicha renovación de licencias o de transporte.

Si agrega términos de un trabajo amparado de acuerdo con esta sección, debe colocar, en los archivos de origen pertinentes, una declaración de los términos adicionales que se aplican a esos archivos, o un aviso que indica dónde encontrar los términos aplicables.

Los términos adicionales, permisivas o no permisivas, pueden expresarse en la forma de una licencia por escrito separado, o establecen como excepciones; los requisitos anteriores se aplican en ambos sentidos.

8. Terminación.

Usted no puede distribuir o modificar un trabajo amparado salvo lo dispuesto expresamente en esta Licencia. Cualquier intento de lo contrario se propague o modificarlo es nulo, y terminará automáticamente sus derechos bajo esta Licencia (incluyendo cualquier patente conseguida según el párrafo tercero del apartado 11).

Sin embargo, si usted deja toda violación de esta Licencia, entonces su licencia desde el poseedor del copyright correspondiente será restituida (a) provisionalmente, a menos que y hasta que el titular del derecho de autor termina de forma explícita y, finalmente, su licencia, y (b) de forma permanente, si el titular del derecho de autor no para notificarle de la violación por parte de algunos medios razonables antes de 60 días después de la cesación.

Por otra parte, su licencia de un titular de derechos de autor en particular se

restableció de forma permanente si el titular del copyright le notifica de la violación por parte de algunos medios razonables, esta es la primera vez que ha recibido la notificación de la violación de esta Licencia (para cualquier trabajo) de ese poseedor de los derechos de autor, y curar la violación antes de 30 días desde la recepción de la notificación.

La terminación de sus derechos bajo esta sección no termina las licencias de terceros que hayan recibido copias o derechos de usted bajo esta Licencia. Si se han terminado sus derechos y no reintegrados de forma permanente, usted no califica para recibir nuevas licencias para el mismo material en virtud del artículo 10.

9. Aceptación no obligatoria por tenencia de copias.

Usted no está obligado a aceptar esta licencia con el fin de recibir o ejecutar una copia del Programa. Propagación auxiliar de un trabajo amparado surgida simplemente como consecuencia de la utilización de la transmisión de punto a punto de recibir una copia tampoco requiere aceptación. Sin embargo, nada más que esta Licencia le otorga permiso para distribuir o modificar cualquier trabajo amparado. Estas acciones infringen los derechos de autor si no acepta esta Licencia. Por lo tanto, al modificar o distribuir un trabajo amparado, usted indica su aceptación de esta Licencia para poder hacerlo.

10. Herencia automática de licencia para destinatarios.

Cada vez que distribuya un trabajo amparado, el destinatario recibe automáticamente una licencia del los licenciadores originales, para ejecutar, modificar y propagar ese trabajo, sujeto a esta licencia. Usted no es responsable de asegurar el cumplimiento por parte de terceros con esta licencia.

Una "transacción de entidad" es un control de transacciones de transferencia de una organización, o sustancialmente todos los activos de una, o subdivide una organización, o la fusión de las organizaciones. Si la propagación de un trabajo amparado surge de una transacción de entidad, cada parte en esa transacción que reciba una copia de la obra también recibe cualesquiera licencias para el trabajo predecesor de la parte interesada tenía o podía dar en el párrafo anterior, además de un derecho de posesión de la Fuente Correspondiente de la obra de su predecesor en el interés, si el predecesor tiene o puede conseguir con un esfuerzo razonable.

Usted no puede imponer ninguna restricción más sobre el ejercicio de los derechos otorgados o concedidos en virtud de esta licencia. Por ejemplo, usted no puede imponer una cuota de licencia, derechos u otros cargos por el ejercicio de los derechos otorgados bajo esta licencia, y usted no puede iniciar un litigio (incluyendo una reconvencción o contrademanda en un juicio), alegando que cualquier reivindicación de patente se ha infringido al hacer, usar, vender, ofrecer en venta o importar el Programa o cualquier parte del mismo.

11. Patentes.

A "colaborador" es un titular de los derechos que se autoriza el uso bajo esta Licencia del Programa o un trabajo en el que se basa el Programa. El trabajo con esta licencia se denomina "versión en colaboración" con el colaborador.

"Las reivindicaciones de patentes esenciales" de un contribuyente son todos los reclamos de patentes de propiedad o controladas por el contribuyente, ya sea ya o adquiridas con posterioridad, que serían infringidos por alguna manera, permitidos por esta Licencia, de hacer, usar o vender la versión en colaboración, pero no incluyen reclamos que se infringieron sólo como consecuencia de una mayor modificación de la versión en colaboración. Para efectos de esta definición, "control" incluye el derecho de conceder sublicencias de patente de una manera consistente con los requerimientos de esta Licencia.

Cada colaborador le concede una licencia no exclusiva, mundial, libre de regalías licencia de patente en virtud de las reivindicaciones de patentes esenciales del contribuyente, para hacer, usar, vender, ofrecer en venta, importación y de otra manera ejecutar, modificar y propagar el contenido de la versión en colaboración.

En los siguientes tres párrafos, una "licencia de patente" es cualquier acuerdo o compromiso expreso, sin embargo denominados, no para hacer cumplir una patente (como un permiso expreso de practicar una patente o de pacto de no demandar por violación de patentes). "Conceder" estas licencias de patente a un tercero significa llegar a tal acuerdo o compromiso que no imponga una patente al tercero.

Si usted distribuya un trabajo amparado, conociendo que en una licencia de patente, y la Fuente Correspondiente de la obra no está disponible para cualquier persona para copiar, de forma gratuita y en los términos de esta Licencia, a través de un servidor de red

a disposición del público o de otro tipo de fácil acceso significa, entonces debe o bien (1) causar la Fuente Correspondiente a ser tan disponible, o (2) ordenar que privarse de los beneficios de la licencia de patente para este trabajo en particular, o (3) organizar, de manera coherente con el requisitos de esta Licencia, para extender la licencia de patente a los beneficiarios aguas abajo. "A sabiendas que confía" significa que usted tiene conocimiento efectivo de que, pero para la licencia de patente, el transporte de la obra amparada en un país, o el uso de su receptor de la obra amparada en un país, infringiría una o más patentes existentes en ese país que tiene razones para creer son válidos.

Si, en virtud de o en conexión con una sola transacción o arreglo, usted transporta o distribuye con fines de transporte de un trabajo amparado, y otorga una licencia de patente a algunas de las partes que recibe el trabajo cubierto que les autorice a utilizar, propagar, modificar o transmitir una copia específica del trabajo amparado, entonces la licencia de patente que usted otorgue se extiende automáticamente a todos los destinatarios de la obra y las obras cubiertas basados en él.

Una licencia de patente es "discriminatoria" si no incluye dentro del ámbito de su cobertura, prohíbe el ejercicio de, o está condicionada a la falta de ejercicio de uno o más de los derechos concedidos específicamente en virtud de esta licencia. Usted no puede transmitir un trabajo amparado si usted es parte de un acuerdo con un tercero que está en el negocio de distribución de software, bajo las cuales usted hace el pago a la tercera parte sobre la base de la extensión de su actividad de transporte de la obra, y en virtud del cual las subvenciones tercero del partido, a ninguna de las partes que recibirían el trabajo cubierto de usted, una licencia de patente discriminatoria (a) en relación con las copias de la obra cubierta transmitida por usted (o copias hechas de esas copias), o (b) principalmente para y en relación con los productos o compilaciones que contienen el trabajo amparado específicos, a menos que usted entró en ese arreglo, o que se le concedió licencia de patente, antes del 28 de marzo de 2007.

Nada en esta licencia se interpretará en el sentido de excluir o limitar cualquier licencia implícita o de otras defensas a la infracción que de otra manera puede estar disponible para usted en virtud de la ley de patentes aplicable.

12. No condicionamiento de la libertad de terceros.

Si se imponen condiciones (ya sea por orden judicial, acuerdo u otros) que contradigan las condiciones de esta Licencia, ello no le exime de cumplir las condiciones de esta Licencia. Si no puede transmitir un trabajo amparado con el fin de satisfacer simultáneamente sus obligaciones bajo esta licencia y cualquier otra obligación pertinente entonces, como consecuencia, no puede transmitir en absoluto. Por ejemplo, si usted está de acuerdo con los términos que obliguen a cobrar un canon por transmitir más lejos de aquellos a los que transmitir el Programa, la única forma de satisfacer ambos condicionantes y esta Licencia sería abstenerse completamente de distribuir el Programa.

13. Uso conjunto con la Licencia GNU Affero General Public License.

No obstante cualquier otra disposición de esta Licencia, usted tiene permiso para enlazar o combinar cualquier trabajo amparado con otro trabajo amparado por la versión 3 de la Licencia Pública General Affero GNU en un solo trabajo combinado, y transmitir el trabajo resultante. Los términos de esta Licencia seguirán aplicándose a la parte que es el trabajo amparado, pero los requisitos especiales de la Licencia Pública GNU Affero General, el artículo 13, en relación con la interacción a través de una red se aplicarán a la combinación como tal.

14. Versiones Revisadas de esta Licencia.

La Free Software Foundation puede publicar versiones revisadas y / o nuevas de la Licencia Pública General GNU de vez en cuando. Tales versiones nuevas serán similares en espíritu a la presente versión, pero pueden diferir en detalles para considerar nuevos problemas o situaciones.

Cada versión recibe un número de versión que la distingue. Si el Programa especifica que cierta versión numerada de la Licencia Pública General de GNU "o cualquier versión posterior" se aplica a él, usted tiene la opción de seguir los términos y condiciones, bien de esa versión numerada o de cualquier versión posterior publicada por la Free Software Fundación. Si el Programa no especifica un número de versión de la Licencia Pública General de GNU, usted puede elegir cualquier versión publicada por la Free Software Foundation.

Si el Programa especifica que un apoderado puede decidir que se pueden utilizar las futuras versiones de la Licencia Pública General de GNU, la declaración pública de ese

proxy de la aceptación de una versión permanentemente le autoriza a elegir esa versión para el Programa.

Las versiones posteriores de la licencia le puede dar permisos adicionales o diferentes. Sin embargo, no hay obligaciones adicionales se imponen a cualquier autor o copyright titular como consecuencia de que usted adopte una versión posterior.

15. Renuncia de garantía.

NO HAY GARANTÍA PARA EL PROGRAMA, EN LA MEDIDA PERMITIDA POR LA LEY APLICABLE. Excepto cuando se indique de otra forma por escrito, los tenedores del copyright y / u otras partes proporcionan el programa "TAL CUAL" SIN GARANTÍA DE NINGÚN TIPO, EXPRESA O IMPLÍCITAS, INCLUYENDO, PERO NO LIMITADO A, LAS GARANTÍAS DE COMERCIALIZACIÓN Y APTITUD PARA UN PROPÓSITO PARTICULAR . EL RIESGO EN CUANTO A LA CALIDAD Y EL DESEMPEÑO DEL PROGRAMA ES CON USTED. SI EL PROGRAMA TIENE UN ERROR, USTED ASUME EL COSTE DE CUALQUIER SERVICIO, REPARACIÓN O CORRECCIÓN.

16. Limitación de responsabilidad.

EN NINGÚN CASO, salvo que lo requiera la legislación aplicable o haya sido acordado por escrito, ningún tenedor del copyright ni ninguna otra parte que MODIFICA Y / O TRANSMITE EL PROGRAMA COMO PERMITIDO ANTERIORMENTE, SERÁ RESPONSABLE ANTE USTED POR DAÑOS, INCLUYENDO CUALQUIER GENERAL, ESPECIAL, ACCIDENTAL O CONSECUENTE DAÑOS DERIVADOS DEL USO O LA IMPOSIBILIDAD DE USO DEL PROGRAMA (INCLUYENDO PERO NO LIMITADO A LA PÉRDIDA DE DATOS, DATOS CONSIDERADOS PÉRDIDAS inexacta o sufridas por usted o por terceras partes oa un fallo del Programa al funcionar con cualquier otro programa), INCLUSO SI dicho tenedor u otra parte ha sido advertido de la posibilidad de tales daños.

FIN DE LOS TÉRMINOS Y CONDICIONES

4.8 Texto de licencia GPLv3

1. Copyright (C) 2015 Sergio Martín, Domingo Benítez y Carlos Travieso

Este programa es software libre. Puede redistribuirlo y/o modificarlo bajo los términos de la Licencia Pública General de GNU tal como está publicada por la Free Software Foundation, bien de la versión 3 de dicha Licencia o bien (según su elección) de cualquier versión posterior.

Este programa se distribuye con la esperanza de que sea útil, pero SIN NINGUNA GARANTÍA, incluso sin la garantía MERCANTIL implícita o sin garantizar la CONVENIENCIA PARA UN PROPÓSITO PARTICULAR. Véase la Licencia Pública General de GNU para más detalles.

Usted debería haber recibido una copia de la Licencia Pública General junto con este programa. Si no ha sido así, consulte <<http://www.gnu.org/licenses>>.

Permiso adicional bajo la GPL de GNU versión 3, sección 7

Si usted modifica este programa, o cualquier obra incluida en él, enlazándolo o combinándolo con [nombre de la biblioteca] (o una versión modificada de esta biblioteca), conteniendo así partes cubiertas por los términos de la licencia [nombre de la licencia de la biblioteca], los licenciadores de este programa le conceden permiso adicional para distribuir la obra resultante. {La Fuente Correspondiente de una forma no fuente de tal combinación incluirá el código fuente de las partes de [nombre de la biblioteca] utilizadas así como de la obra cubierta.}

2. Este programa es software libre. Puede redistribuirlo y/o modificarlo bajo los términos de la Licencia Pública General de GNU tal como está publicada por la Free Software Foundation, bien de la versión 2 de dicha Licencia o bien (según su elección) de cualquier versión posterior.

Este programa se distribuye con la esperanza de que sea útil, pero SIN NINGUNA GARANTÍA, incluso sin la garantía MERCANTIL implícita o sin garantizar la CONVENIENCIA PARA UN PROPÓSITO PARTICULAR. Véase la Licencia Pública General de GNU para más detalles.

Usted debería haber recibido una copia de la Licencia Pública General junto con este programa. Si no ha sido así, consulte <<http://www.gnu.org/licenses>>.

Enlazar *SKoA 1.0* estática o dinámicamente con otros módulos es hacer una obra combinada basada en *SKoA 1.0*. De este modo, los términos y condiciones de la Licencia Pública General de GNU cubren la combinación en conjunto.

Además, como excepción especial, los titulares del copyright de *SKoA 1.0* le conceden permiso para combinar *SKoA 1.0* con programas de software libre o bibliotecas publicadas bajo la LGPL de GNU y con código incluido en la edición estándar de [nombre de la biblioteca] bajo la licencia [nombre de la licencia de la biblioteca] (o versiones modificadas de dicho código, siempre que la licencia no haya cambiado). Puede copiar y distribuir el sistema así construido, según los términos de la GPL de GNU para *SKoA 1.0* y las licencias del resto del código implicado{, siempre que incluya el código fuente de ese otro código cuando y de la manera en que la GPL de GNU exija la distribución del código fuente}.

Tenga en cuenta que quien haga versiones modificadas de *SKoA 1.0* no está obligado a garantizar esta excepción especial en sus versiones modificadas; queda a su discreción hacerlo o no. La Licencia Pública General de GNU concede permiso para publicar una versión modificada sin esta excepción; esta excepción hace también posible publicar una versión modificada que contenga dicha excepción.

3. No puede imponer al receptor ninguna restricción ulterior sobre el ejercicio de los derechos aquí concedidos.

4. Al enlazar ABC estática o dinámicamente con otros módulos se está haciendo una obra combinada basada en ABC. Así, pues, los términos y condiciones de la Licencia Pública General de GNU cubren la combinación en conjunto.

Como excepción especial, los titulares del copyright de ABC le conceden permiso para combinar el programa ABC con programas de software libre o bibliotecas publicadas bajo la LGPL de GNU y con módulos independientes que se comuniquen con ABC únicamente a través de la interfaz ABCDEF. Puede copiar y distribuir el sistema así construido, según los términos de la GPL de GNU para ABC y de las licencias del resto del código implicado, siempre que incluya el código fuente de ese otro código, en el momento y de la manera que la GPL de GNU exija la distribución del código fuente, y siempre que usted no modifique la interfaz de ABCDEF.

Nótese que quien haga modificaciones de ABC no está obligado a mantener esta excepción

especial en sus versiones modificadas; queda a discreción de ellos hacerlo o no. La Licencia Pública General de GNU concede permiso para publicar una versión modificada sin esta excepción; esta excepción hace también posible publicar una versión modificada que contenga dicha excepción. Si usted modifica la interfaz de ABCDEF, esta excepción no se aplica a su versión modificada de ABC, y debe eliminar esta excepción cuando distribuya su versión modificada.

Esta excepción constituye un permiso adicional bajo la sección 7 de la Licencia Pública General de GNU, versión 3 («GPLv3»).

5. Como excepción especial, si crea un documento que utiliza este tipo de letra, e inserta en el documento este tipo de letra o partes inalteradas del mismo, este tipo de letra por sí mismo no hace que el documento resultante esté cubierto por la Licencia Pública General de GNU. No obstante, esta excepción no invalida ninguna otra razón por la cual el documento podría estar cubierto por la Licencia Pública General de GNU. Si modifica este tipo de letra puede hacer extensiva esta excepción a su versión del tipo de letra, pero no está obligado a hacerlo. Si no desea hacerlo, borre esta nota de excepción de su versión.

6. Como excepción especial a la GPL, cualquier archivo HTML que simplemente haga llamadas a funciones de este código y que a tal fin lo incluya por referencia, a efectos de las leyes de copyright habrá de considerarse como una obra separada. Además, los titulares del copyright de este código le conceden permiso para combinar este programa con bibliotecas libres publicadas bajo la LGPL de GNU. Puede copiar y distribuir el sistema así construido, según los términos de la GPL de GNU para este código y según los de la LGPL para las bibliotecas. Si modifica este código puede hacer extensiva esta excepción a su versión del código, pero no está obligado a hacerlo. Si no desea hacerlo, borre esta nota de excepción de su versión.

7. Esta traducción de la GPL es informal y no ha sido aprobada oficialmente por la Free Software Foundation como válida. Para estar completamente seguro de lo que está permitido, consulte la GPL original (en inglés): <http://www.gnu.org/licenses/>

4.9 Resumen del capítulo

En este capítulo hemos realizado un análisis exhaustivo para lograr seleccionar la mejor licencia con la que lanzar SKoA como software libre.

Finalmente nos decantamos por la licencia GPLv3 de GNU, dado que las características que nos ofrece son más atractivas que el resto de las licencias analizadas y además permite que nuestro software continúe siendo libre aún siendo modificado por otros usuarios.

Aparte de todo esto, cumple en compatibilidad con la librería Calimero, sumamente importante en este proyecto.

Con este procedimiento logramos cumplir nuestro objetivo de publicar el software bajo una licencia libre, y tanto la aplicación como el código fuente estarán a disposición de cualquier usuario en la red.

Capítulo 5:

Aplicación Móvil Proyecto PILEV

5.1 Introducción

En este proyecto se ha incluido la creación de una aplicación para dispositivos móviles que muestre las informaciones energéticas de varios hoteles incluidos en el **Proyecto PILEV** (Punto de Información Lopesan de las Energías Verdes) llevado a cabo por la Universidad de Las Palmas de Gran Canaria a través del IUSIANI (Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en la Ingeniería) en colaboración con la cadena de hoteles LOPESAN.

Esta aplicación será creada para su funcionamiento para sistemas operativos Android, así como para cualquier dispositivo móvil existente, de forma que podamos acceder a este punto de información en cualquier momento del día y obtener los datos energéticos en directo.

5.2 Antecedentes

Para entender qué es el Proyecto PILEV en primer lugar tenemos que llevar a cabo un análisis a su antecesor: el Proyecto PIVE (**P**unto de **I**nformación de energías **V**erdes).

El Proyecto PIVE se ha desarrollado para resolver un problema de accesibilidad a la información relacionada con la cantidad de Energías Renovables que un edificio hotelero es capaz de producir [14].

El principal objetivo de este proyecto consiste en diseñar, desarrollar y construir un producto informático innovador en el Sector Turístico que denominado Punto de Información Turística de la Producción de Energías Verdes (de ahora en adelante “PIVE”).

En la siguiente figura se puede observar un esquema de las instalaciones que se que

se utilizan en este proyecto.



Figura 5.1 Esquema de las instalaciones utilizadas en el proyecto PIVE

El segundo objetivo consiste en que varios puntos PIVE se conecten a través de líneas ADSL de Telefónica con un computador servidor de la Universidad de Las Palmas de Gran Canaria que centralizará la información generada en los puntos PIVE.

En la siguiente figura se puede observar un diagrama que representa a la red informática de sistemas PIVE empleada en este proyecto.

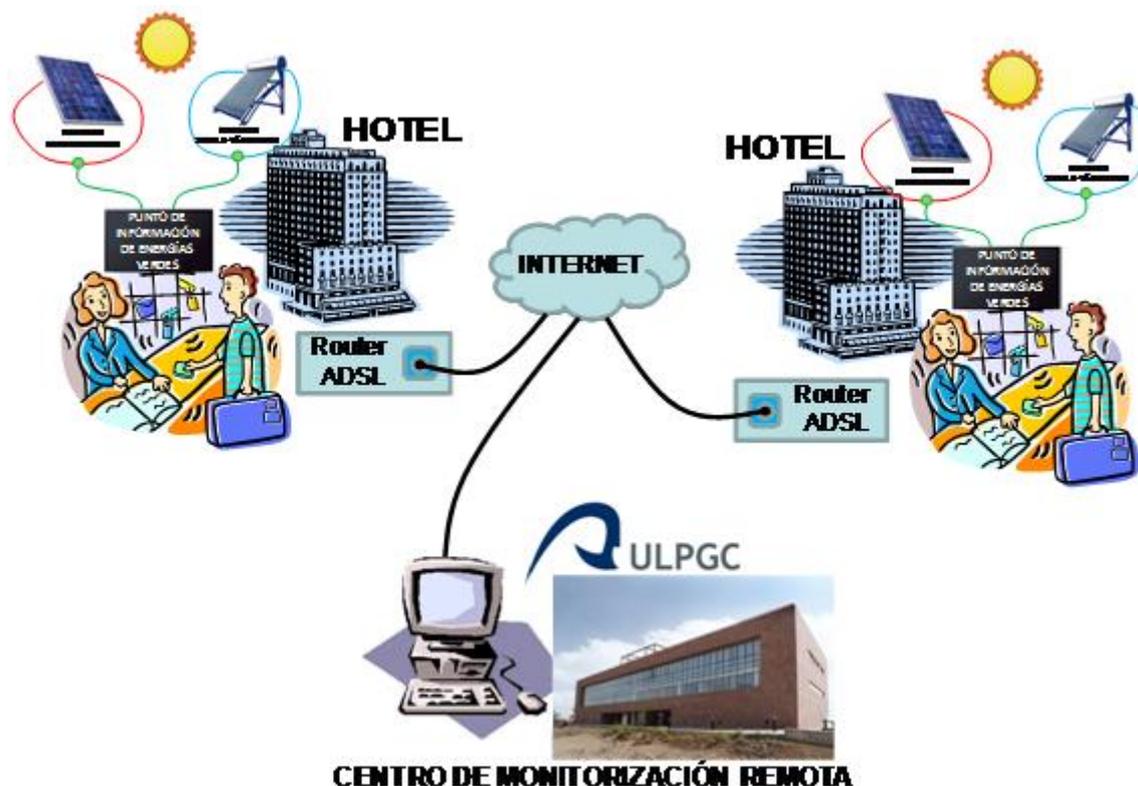


Figura 5.2 Esquema de las instalaciones utilizadas en el proyecto PIVE

El tercer objetivo del proyecto PIVE consiste en hacer accesible al gran público los beneficios medioambientales de las energías alternativas.

El principal beneficio obtenido de este proyecto PIVE consiste en aportar valor añadido al producto turístico a través de la implantación de infraestructuras basadas en las Tecnologías de la Información y la Comunicaciones.

De esta forma, se ha logrado una nueva estrategia empresarial innovadora que ha mejorado la productividad del Sector Turístico.

En este proyecto participan un equipo de personas que tienen un marcado carácter multidisciplinar. Las disciplinas en las que cada una de estas personas es especialista son claramente distintas: Empresa, Turismo, Ingeniería Industrial, Telemática, Informática.

Las entidades participantes son tres: la Universidad de Las Palmas de Gran Canaria (ULPGC) a través del Instituto Universitario SIANI, y dos empresas turísticas, Scanaria Invest S.A. y Sicilia & Roig, S.L.

5.3 Proyecto PILEV

El proyecto PIVE ha creado un centro de divulgación por Internet en colaboración con la empresa hotelera LOPESAN, denominado **PILEV** (Punto de Información Lopesan de las Energías Verdes).

En él se muestra la cantidad de energía solar que cada uno de sus hoteles genera en cada minuto, así como sus parámetros equivalentes que indican el grado de protección medioambiental aplicada.

Los datos obtenidos provienen de 5 hoteles:

- Lopesan Baobab Resort (Costa Meloneras, Gran Canaria)
- Hotel IFA Faro (Maspalomas, Gran Canaria)
- Hotel IFA Buenaventura (Playa del Inglés, Gran Canaria)
- Hotel IFA Continental (San Bartolomé de Tirajana, Gran Canaria)
- Hotel IFA Interclub Atlantic (San Agustín, Gran Canaria)

Los datos son recogidos por las infraestructuras del proyecto PIVE y volcados en un archivo .txt en uno de los servidores utilizados para esta actividad.

Estos datos se actualizan continuamente mostrando los últimos datos cada pocos minutos. Esta es la estructura que presenta el archivo .txt al que se puede acceder a través de la dirección: <http://pilve.iusiani.ulpgc.es/PILEV1.1-war/ficherosTXT/datosLopesan.txt>

```

PILEV 1.1 - Datos totales de HOY del GRUPO LOPESAN
Ultima actualizacion: 2015-07-19 20:30
Energia: 8363 kWh
CO2      : 5018 kilos
Arboles: 544 arboles
PILEV 1.1 - Datos Solar-Termica totales de HOY del hotel BAOBAB
Ultima actualizacion: 2015-07-19 20:30
Energia: 4833 kWh
CO2      : 2900 kilos
Arboles: 314 arboles
PILEV 1.1 - Datos Solar-Termica totales de HOY del hotel FARO
Ultima actualizacion: 2015-07-19 20:30
Energia: 390 kWh
CO2      : 234 kilos
Arboles: 25 arboles
PILEV 1.1 - Datos Solar-Termica totales de HOY del hotel CONTINENTAL
Ultima actualizacion: 2015-07-19 20:30
Energia: 2800 kWh

```

```

C02      : 1680 kilos
Arboles: 182 arboles
PILEV 1.1 - Datos Solar-Termica totales de HOY del hotel INTERCLUB1
Ultima actualizacion: 2015-07-19 20:30
Energia: 340 kWh
C02      : 204 kilos
Arboles: 22 arboles
PILEV 1.1 - Datos Fotovoltaica totales de HOY del hotel BUENAVENTURA
Ultima actualizacion: 2015-07-19 20:30
Energia: 0 kWh
C02      : 0 kilos
Arboles: 0 arboles
PILEV 1.1 - Datos Solar-Termica totales de AYER del hotel BAOBAB
Ultima actualizacion: 2015-07-19 20:30
Energia: 4600 kWh
C02      : 2760 kilos
Arboles: 299 arboles
PILEV 1.1 - Datos Solar-Termica totales de AYER del hotel FARO
Ultima actualizacion: 2015-07-19 20:30
Energia: 390 kWh
C02      : 234 kilos
Arboles: 25 arboles
PILEV 1.1 - Datos Solar-Termica totales de AYER del hotel CONTINENTAL
Ultima actualizacion: 2015-07-19 20:30
Energia: 0 kWh
C02      : 0 kilos
Arboles: 0 arboles
PILEV 1.1 - Datos Solar-Termica totales de AYER del hotel INTERCLUB1
Ultima actualizacion: 2015-07-19 20:30
Energia: 330 kWh
C02      : 198 kilos
Arboles: 21 arboles
PILEV 1.1 - Datos Fotovoltaica totales de AYER del hotel BUENAVENTURA
Ultima actualizacion: 2015-07-19 20:30
Energia: 1192 kWh
C02      : 715 kilos
Arboles: 77 arboles
    
```

Figura 5.3 Archivo .txt generado por el servidor del Proyecto PIVE

Como podemos observar en la figura anterior, se nos muestran los datos recogidos por el servidor en un momento concreto del día.

El primer apartado nos muestra los datos totales de los 5 hoteles de los que se está recogiendo la información

```

PILEV 1.1 - Datos totales de HOY del GRUPO LOPESAN
Ultima actualizacion: 2015-07-19 20:30
Energia: 8363 kWh
C02      : 5018 kilos
Arboles: 544 arboles
    
```

Figura 5.4 Datos Totales de los hoteles en el día de HOY

La primera línea nos indica el nombre genérico PILEV 1.1 que se refiere al punto de información del que se está extrayendo la información, seguido del título que nos indica qué datos se nos está mostrando. En este caso se trata de los *Datos Totales de HOY del GRUPO LOPESAN*, es decir, los datos de los 5 hoteles juntos.

La segunda línea nos muestra cuándo se realizó la última actualización de datos en formato “AA-MM-DD HH:MM”.

La tres líneas siguientes nos muestran los valores obtenidos para los diferentes parámetros que recoge el proyecto PIVE: Energía, CO2 y Árboles.

- **Energía:** Cantidad de energía solar generada en kW/h
- **CO2:** Cantidad de CO2 que se ha evitado emitir a la atmósfera en kilogramos
- **Árboles:** Cantidad equivalente en árboles al CO2 ahorrado

El resto de líneas corresponde a los datos de los 5 hoteles examinados de forma individual, tanto del día actual como del anterior. Cada hotel está identificado en la primera línea por su nombre y junto a cada nombre se especifica si la información corresponde al día de hoy o al día de ayer.

Ejemplo:

- HOTEL BAOBAB HOY

```

PILEV 1.1 - Datos Solar-Termica totales de HOY del hotel BAOBAB
Ultima actualizacion: 2015-07-19 20:30
Energia: 4833 kWh
CO2      : 2900 kilos
Arboles: 314 arboles
    
```

Figura 5.5 Datos Hotel Baobab en el día de HOY

- HOTEL BAOBAB AYER

```

PILEV 1.1 - Datos Solar-Termica totales de AYER del hotel BAOBAB
Ultima actualizacion: 2015-07-19 20:30
Energia: 4600 kWh
CO2      : 2760 kilos
Arboles: 299 arboles
    
```

Figura 5.6 Datos Hotel Baobab en el día de AYER

5.4 Diseño de la aplicación móvil

El problema que se nos plantea es diseñar un aplicación móvil multiplataforma, la cual muestre los datos contenidos en el fichero .txt creado por el servidor del proyecto PIVE y al que podemos acceder desde la siguiente dirección:

<http://pilve.iusiani.ulpgc.es/PILEV1.1-war/ficherosTXT/datosLopesan.txt>

Dicho recurso es un archivo de texto plano, el cual contiene los datos de consumo de las placas fotovoltaicas de los hoteles pertenecientes al Grupo LOPESAN y se actualiza de forma constante.

Dicho esto y entrando en materia nos planteamos cómo desarrollar dicha aplicación móvil.

Antes de decidir en qué plataforma queremos que la aplicación esté disponible, lo más recomendable es hacer un estudio de las plataformas más destacadas para tener una idea de cuales serían las más efectivas en nuestras circunstancias.

En ocasiones, la presencia en todas las plataformas existentes es desaconsejable y poco eficiente, por lo que basaremos nuestra elección en aquellas plataformas que se adapten a nuestros objetivos. Más adelante, y en función de los resultados, podemos extender nuestra app y recursos a otras plataformas.

Actualmente, las plataformas más potentes son **Android** e **iOS**, según los datos recogidos en el informe Q3 del mercado global, y por detrás de éstas tenemos a Windows Phone y Blackberry.

Period	Android	iOS	Windows Phone	BlackBerry OS	Others
Q1 2015	78.0%	18.3%	2.7%	0.3%	0.7%
Q1 2014	81.2%	15.2%	2.5%	0.5%	0.7%
Q1 2013	75.5%	16.9%	3.2%	2.9%	1.5%
Q1 2012	59.2%	22.9%	2.0%	6.3%	9.5%

Tabla 5.1 Tabla de OS más utilizados entre 2012 y 2015

Considerando todos estos datos, queda mucho más claro para qué plataformas debemos desarrollar en una primera parte. Android e iOS son las que mayor presencia tienen en el mundo de las “apps” por lo que es lógico empezar desarrollando nuestra aplicación para una o ambas plataformas, posteriormente, y dependiendo de los resultados, extenderla a las demás.

5.4.1 Tipos de Aplicaciones

Si hay un mercado que no para de crecer, ese es el de las aplicaciones móviles. El auge de los teléfonos inteligentes y la necesidad del usuario para hacer su vida más cómoda, así como las innumerables ofertas de entretenimiento e información, han hecho que este tipo de aplicaciones se multipliquen de forma colosal. Sin embargo, son muy pocos los que conocen qué tipos de aplicaciones móviles existen y cuál es la mejor para ellos.

En este apartado analizaremos los tipos de aplicaciones que tenemos a nuestra disposición y seleccionaremos aquel que sea idóneo para nuestro proyecto de aplicación [15].

5.4.1.1 Aplicaciones nativas

Una aplicación nativa es la que se desarrolla de forma específica para un determinado sistema operativo, llamado *Software Development Kit* o SDK. Cada una de las plataformas, Android, iOS o Windows Phone, tienen un sistema diferente, por lo que si queremos que nuestra app esté disponible en todas las plataformas se deberán de crear varias apps con el lenguaje del sistema operativo seleccionado.

Ventajas	Inconvenientes
<ul style="list-style-type: none"> ▪ Acceso completo al dispositivo ▪ Mejor experiencia del usuario ▪ Visibilidad en App Store 	<ul style="list-style-type: none"> ▪ Diferentes habilidades, idiomas y herramientas para cada plataforma de destino.

<ul style="list-style-type: none"> ▪ Envío de notificaciones o “avisos” a los usuarios ▪ La actualización de la app es constante 	<ul style="list-style-type: none"> ▪ Tienden a ser más caras de desarrollar ▪ El código del cliente no es reutilizable entre diferentes plataformas
--	---

Tabla 5.2 Ventajas e Inconvenientes Aplicación Nativa

5.4.1.2 Web App

Una aplicación web o *webapp* es la desarrollada con lenguajes muy conocidos por los programadores, como es el HTML, Javascript y CSS. La principal ventaja con respecto a la nativa es la posibilidad de programar de forma independiente del sistema operativo en el que se usará la aplicación. De esta manera se pueden ejecutar en diferentes dispositivos sin tener que crear varias aplicaciones.

Ventajas	Inconvenientes
<ul style="list-style-type: none"> ▪ El mismo código base es reutilizable en múltiples plataformas. ▪ Proceso de desarrollo más sencillo. ▪ No se necesita ninguna aprobación externa para publicarse. ▪ El usuario siempre dispones de la última versión. ▪ Pueden reutilizarse sitios “responsive” ya diseñados. 	<ul style="list-style-type: none"> ▪ Requiere de conexión a internet ▪ Acceso muy limitado a los elementos y características del hardware del dispositivo. ▪ La experiencia del usuario y el tiempo de respuesta es menor que una app nativa ▪ Requiere de mayor esfuerzo en promoción y visibilidad.

Tabla 5.3 Ventajas e Inconvenientes Web Apps

5.4.1.2 Aplicaciones híbridas

Una aplicación híbrida es una combinación de las dos anteriores, se podría decir que recoge lo mejor de cada una de ellas. Las apps híbridas se desarrollan con lenguajes propios de las webabpp, es decir, HTML, Javascript y CSS por lo que permite su uso en diferentes plataformas, pero también dan la posibilidad de acceder a gran parte de las características del hardware del dispositivo. La principal ventaja es que a pesar de estar desarrollada con HTML, Java o CSS, es posible agrupar los códigos y distribuirla en app store.

Ventajas	Inconvenientes
<ul style="list-style-type: none"> ▪ Es posible distribuirla en las tiendas de iOS y Android. ▪ Instalación nativa pero construida con JavaScript, HTML y CSS. ▪ El mismo código base para múltiples plataformas. ▪ Acceso a parte del hardware del dispositivo. 	<ul style="list-style-type: none"> ▪ Experiencia de usuario más propia de la aplicación web que de la app nativa ▪ Diseño visual no siempre relacionado

Tabla 5.4 Ventajas e Inconvenientes Aplicaciones Híbridas

Una vez que hemos analizado los tipos de aplicaciones existentes debemos tomar una decisión para elegir con cuál de ellos vamos a desarrollar nuestra aplicación.

Desde el punto de vista de desarrollador, la opción más atractiva para nuestro proyecto, dado que necesitamos que sea compatible con Android, va a ser el desarrollo de una aplicación híbrida. Este tipo de aplicaciones nos dan la posibilidad de ejecutar nuestra aplicación en diversas plataformas y además usaremos lenguajes más comunes a los que estamos más habituados.

5.4.2 PhoneGap

Para el desarrollo de nuestra aplicación híbrida nos hemos decantado por uno de los frameworks más conocidos: PhoneGap

Principalmente, PhoneGap permite a los programadores desarrollar aplicaciones para dispositivos móviles utilizando herramientas genéricas tales como JavaScript, HTML5 y CSS3. Las aplicaciones resultantes son híbridas, ya que el renderizado se realiza mediante vistas web y no con interfaces gráficas específicas de cada sistema, pero no se tratan tampoco de aplicaciones web.

El esquema en el que se basa el framework PhoneGap (Apache Cordova) es el siguiente:

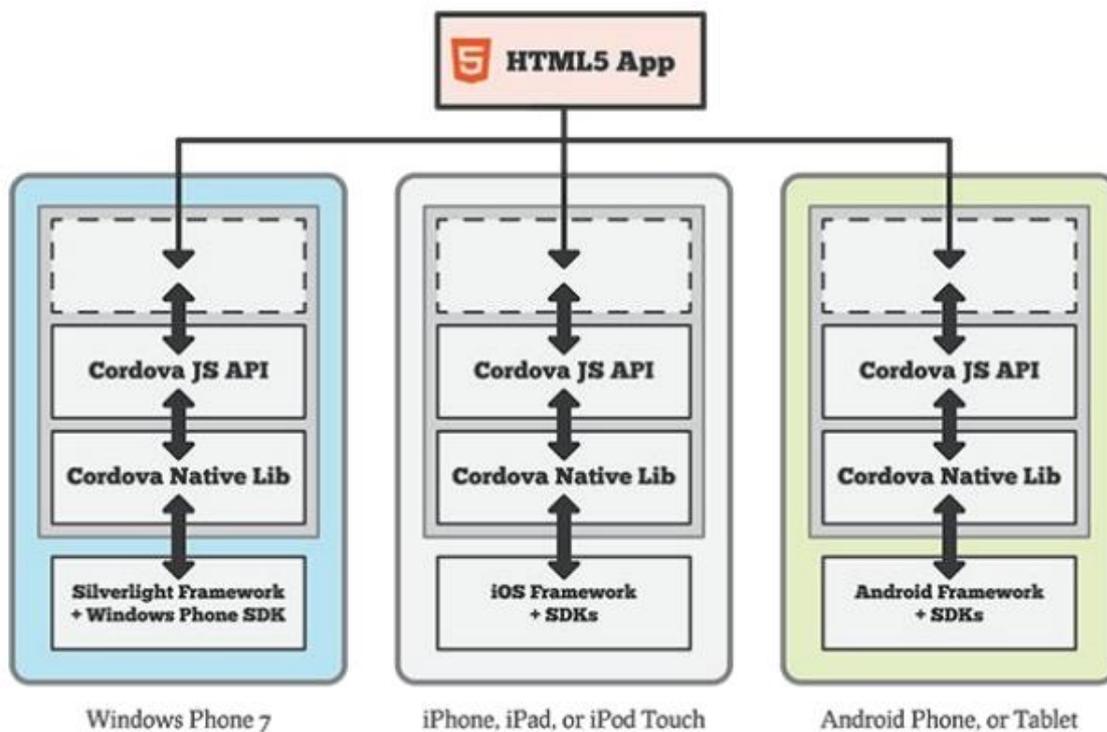


Figura 5.7 Esquema de desarrollo de PhoneGap

Este framework nos ofrece aplicaciones de desarrollar y con una curva de aprendizaje relativamente baja.

Para el correcto funcionamiento del framework PhoneGap necesitamos tener

instaladas las siguientes herramientas:

- Node js
- SDK Android
- PhoneGap

Estas herramientas son las encargadas de generar el apk que se ejecutará en los terminales.

Para el desarrollo utilizamos un editor de texto, bastante popular en la comunidad de desarrolladores llamado Sublime-text 3

Para realizar las pruebas se usará el emulador de PhoneGap llamado Ripple y como sistema de versionado de código utilizamos Git y Github.

5.4.3 Desarrollo y diseño de la aplicación

Una vez que hemos decidido la tecnología con la que construiremos la aplicación, buscaremos un framework de JavaScript Mobile que nos ayude tanto con el control como con la visualización de componentes.

Para ello se han utilizado dos framework bastante comunes hoy en día: *Angular.js* y *Onsen*. El primero nos ayudará con el control de los eventos y el segundo nos permitirá mostrar una interfaz acorde con los tiempos que corren: amigable, minimalista y bastante eficiente.

Dicho esto, definimos nuestro archivo *Index.html* dentro de la carpeta *www* en el proyecto *PhoneGap*. Este archivo será el punto de inicio de nuestra aplicación.

Index.html no es más que un archivo de HTML5 en el cual declaramos componentes descritos en la documentación de *Onsen*. Además de esto también creamos un archivo dentro de la carpeta *js* llamado *App.js*, que se encargará de controlar y cargar toda la información de la aplicación, así como también se encarga de realizar el *scrapping*.

Scraping es una técnica web para extraer datos de una página web o de un recurso en Internet por medio de una solicitud *http* con el fin de visualizar los datos de una manera diferente o utilizarlos para un propósito diferente.

Básicamente este archivo realiza una solicitud *http* al siguiente enlace http://pilve.iusiani.ulpgc.es/PILEV1.1_war/ficherosTXT/datosLopesan.txt, filtra los datos y los carga en un objeto de un modelo *json*, para posteriormente pasárselo a la vista y que esta lo muestre.

En el siguiente trozo del código del archivo de control *App.js* podemos observar la toma de datos desde el recurso en internet.

```
/*
  Funcion que realiza el request
*/
function get() {
  var deferred = $q.defer();
  $http.get('http://pilve.iusiani.ulpgc.es/PILEV1.1
war/ficherosTXT/datosLopesan.txt')
  .success(function(data, status, headers, config) {

    var items = [];
    var scrapingItems = scraping(data);
    deferred.resolve(scrapingItems);
  })
  .error(deferred.reject);
  return deferred.promise;
}
```

5.4.4 Vista final de la aplicación

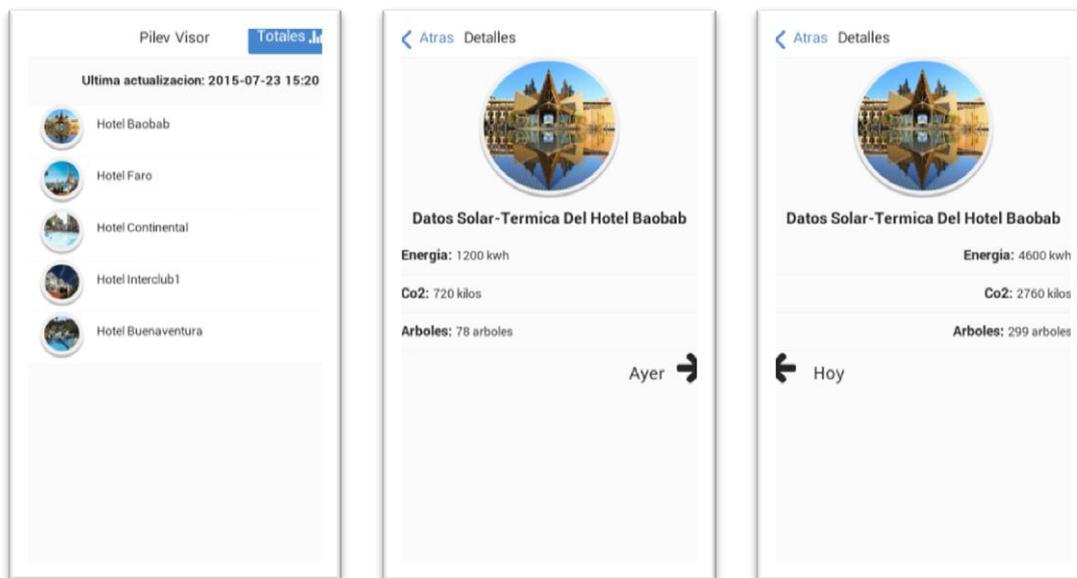


Figura 5.8 Vista final de la aplicación

5.4.5 Resumen del capítulo

A lo largo de este capítulo hemos narrado el diseño y creación de una aplicación móvil que nos mostrará los datos obtenidos de los puntos energéticos del proyecto PILEV en diferentes hoteles del Grupo Lopesan.

Se ha creado una aplicación dinámica y sencilla en la que el usuario podrá acceder a los datos actualizados que nos vuelca el servidor en la URL de la que hemos obtenido el acceso a la información energética.

Capítulo 6:

Conclusiones

6.1 Introducción

A través de este capítulo se pretende ofrecer una visión general de las cuestiones tratadas en la elaboración de este PFC en todas sus fases de desarrollo así como las conclusiones finales que de él se han obtenido.

6.2 Conclusiones

Durante la duración de este proyecto se han ido cumpliendo cronológicamente los objetivos establecido en un principio. Este proceso se puede dividir en dos grandes apartados: Optimización de la aplicación SKoA para su posterior publicación como software libre y el diseño de una aplicación móvil que muestre los datos obtenidos de los puntos de información energéticos del proyecto PILEV.

Durante el proceso de optimización del software hemos ido mejorando la eficiencia y la funcionalidad de la aplicación. Teniendo en cuenta que este software fue creado y diseñado en diferentes etapas y Proyectos Fin de Carrera distintos de varios alumnos de la ULPGC, se puede hacer una idea de lo embrollado y caótico que podía llegar a ser el código.

Desde partes de códigos añadidos y sin comentar hasta proyectos complementos funcionando de forma independiente al software, sin haber sido implementados.

Todo esto a generado un código en muchos casos incomprensible para desarrolladores que lo ven por vez primera y que tendrían que pasar muchas horas “descifrando” las funcionalidades del código fuente.

Por todo ello, hemos llevado a cabo un exhaustivo trabajo de limpieza y depuración del código, además de realizar un comentado en todas sus partes que pueda servir de guía

al desarrollador a la hora de tratar de entender el código, ahorrándole tiempo y algún que otro quebradero de cabeza.

Además de la limpieza del código, se ha procedido a la reestructuración de los paquetes del código, cuya organización previa no seguía ningún tipo de patrón o lógica y lograba llevar a la confusión hasta al desarrollador más experimentados.

Clases con nombres no orientativos o paquetes totalmente fuera de lugar nos han obligado a realizar una organización más adecuada para un software que se pretende publicar en código abierto para la comunidad.

Por todo ello, hemos basado esta reestructuración en el patrón Modelo-Vista-Controlador, un patrón que favorece la reutilización del código y la separación de conceptos, facilitando así las tareas de desarrollo de aplicaciones así como el posterior mantenimiento.

Tras haber optimizado el código desde su interior no podía faltar una optimización gráfica del software.

En la versión anterior nos topábamos con iconos e imágenes totalmente desfasados a los tiempos actuales, así como un entorno gráfico por defecto. Además, se utilizaban formatos de imágenes incorrectos, que cargaban molestos fondos blancos que en ocasiones afectaban directamente a la correcta visualización y uso de la aplicación.

En base a lo anterior, se ha realizado una renovación de iconos e imágenes mucho más actuales y que llaman la atención del usuario final por su carácter realista y moderno, y también se han usado formatos de imagen que eliminan los molestos fondos que impedían una visualización correcta.

En cuanto al entorno gráfico general de la aplicación, se ha generado un nuevo diseño, algo más atractivo y visual que el utilizado por defecto, y que además será el mismo para cualquier Sistema Operativo en el que se ejecute nuestro software.

Todos estos cambios y optimizaciones se han llevado a cabo por un objetivo: publicar nuestra aplicación como software libre, para que toda la comunidad pueda hacer uso de ella, además de hacerla crecer aún más.

Para ello hemos realizado un profundo análisis sobre las posibilidades de las muy abundantes licencias de código abierto que nos oferta el mercado.

Debemos cumplir una serie de requisitos previos para la elección de nuestra licencia y son la compatibilidad con las librerías libres licenciadas que utilizamos en este proyecto, la seguridad de que nuestro software se mantendrá libre a pesar de las modificaciones y la protección de patentes y grandes multinacionales que pretendan bloquear el libre uso de nuestra aplicación.

Esto nos ha llevado a seleccionar la licencia que más se adaptaba a nuestros requisitos: la GPLv3 de GNU.

Sus condiciones y términos generales encajaban a la perfección con nuestros propósitos y por ello ha obtenido la mejor nota en nuestros análisis respecto a otras licencias existentes.

Con este paso podemos decir que hemos cumplido nuestro objetivo de lanzar a la comunidad una aplicación libre de la que se pueda hacer uso y disfrute sin mayores problemas.

Tras finalizar con nuestra aplicación software nos centramos en la aplicación móvil también creada en este proyecto.

Había una necesidad de mostrar en una aplicación móvil los datos de un servidor del Proyecto PIVE que nos generaba los parámetros energéticos recogidos en las instalaciones fotovoltaicas de varios hoteles del Grupo Lopesan.

Estos datos eran mostrados en formato .txt y debíamos diseñar una app amena y visualmente atractiva que nos ayudara a presentarlos al usuario final.

Para ello nos decantamos por una aplicación híbrida como mejor opción sobre otras posibles, debido a su mayor sencillez y su capacidad multiplataforma, que nos permite ejecutar la aplicación no sólo en Android sino también en otros sistemas operativos.

El resultado ha sido una app muy dinámica, con una gráfica muy potente y bastante sencilla de usar por parte del usuario final.

Finalmente, de la realización de este Proyecto Fin de Carrera se pueden obtener estos tres puntos importantes:

- La optimización de SKoA, cuyo código estaba desorganizado y era bastante complicado de entender, ha sido válida a efectos de eficiencia y velocidad de ejecución
- El entorno gráfico de la aplicación es ahora más atractiva para el usuario final.
- El diseño de la aplicación móvil nos permitirá acceder a los datos energéticos de los hoteles incluidos en el Proyecto PILEV en cualquier dispositivo móvil que se precie.

Bibliografía

- [1] Wikipedia: The Free Encyclopedia. “Domótica” 2004 [Consulta: 14 Febrero 2012]
Disponible en: <https://es.wikipedia.org/wiki/Dom%C3%B3tica>
- [2] HISTORIA DE LA DOMÓTICA: PASADO, PRESENTE Y FUTURO_Posted
in domoTECA - Protocolos de comunicación y Sistemas domóticos. Disponible en:
<http://www.domoprac.com/protocolos-de-comunicacion-y-sistemas-domoticos/historia-de-la-domotica-pasado-presente-y-futuro.html>
- [3] Wikipedia: The Free Encyclopedia. “SCADA”. Noviembre 2012 [Consulta: de 20 de febrero de 2014]. Disponible en: <https://es.wikipedia.org/wiki/SCADA>
- [4] Konnex Association “Manual para la gestión técnica de viviendas y edificios”
- [5] TRAMUNT RUBIO, Estefanía “Software para el procesamiento estadístico de los telegramas de comunicación generados por las instalaciones domóticas Konnex”
Universidad de Las Palmas de Gran Canaria. Mayo 2011
- [6] SourceForge. “Calimero EIBnet/IP client” [Consulta: 2 de Noviembre de 2014]
Disponible en: <http://calimero.sourceforge.net/calimero10.html>
- [7] Wikipedia: The Free Encyclopedia. “CamelCase” Enero 2005 [Consulta: 20 de Marzo 2014] Disponible en: <https://es.wikipedia.org/wiki/CamelCase>
- [8] Wikipedia: The Free Encyclopedia. “Modelo-Vista-Controlador” Mayo 2004
[Consulta: 14 de Enero 2014] Disponible en:
<https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>
- [9] Oracle. “Nimbus Look and Feel” Disponible en:
<https://docs.oracle.com/javase/tutorial/uiswing/lookandfeel/nimbus.html>
- [10] Wikipedia: The Free Encyclopedia. “Software Libre” Septiembre 2001
[Consulta: 24 de Junio 2014] Disponible en:
https://es.wikipedia.org/wiki/Software_libre
- [11] STALLMAN, Richard M. “Free Software Free Society: Selected Essays”
2nd Edition Paperback, Diciembre 2009
- [12] SlideShare RUIZ, Miriam “Licencias de Software Libre” 2011 Disponible
en: <http://es.slideshare.net/inniyah/licencias-de-software-libre>
- [13] SMITH, Brett “A quick guide to GPLv3” Disponible en:

<http://www.gnu.org/licenses/quick-guide-gplv3.html>

- [14] Cátedra Telefónica de la ULPGC “*Proyecto PIVE*” Julio 2010. Disponible en: <http://catedratelefonica.ulpgc.es/proyectos/pive>
- [15] Geospatial. “*Tipo de Aplicaciones Móviles*” Disponible en: <http://geospatialtrainings.com/recursos-gratuitos/tipos-de-aplicaciones-moviles/>
- [16] PRESSMAN, Roger S. “*Ingeniería del software: un enfoque práctico*”, Editorial McGraw Hill, 2010

ANEXOS

Anexo I:

Manual de Usuario

Software SKoA 1.0

Manual de Usuario

Índice

Índice	103
1. Introducción	104
1.1 Cómo usar este manual	104
1.2 ¿Qué es SKoA?	105
1.3 Guía rápida de funciones principales del software.....	2
2. Bus EIB	4
2.1 Conexión del bus EIB	4
2.2 Desconexión del bus EIB	4
3. Implementación de Proyectos	7
3.1 Crear un nuevo proyecto	7
3.2 Abrir un proyecto existente	7
3.3 Editar un proyecto existente.....	7
3.4 Cerrar proyecto	7
4. Procesos gráficos y de control	8
4.1 Procesamiento estadístico.....	7
4.2 Menú contadores	7
4.3 Control y monitorización de los dispositivos domóticos.....	7
4.4 Menú Ayuda	7
4.5 Iconos utilizados.....	7
5. Glosario	49

1. Introducción

En este Manual de Usuario estableceremos una guía de manera sencilla para la correcta ejecución y uso de la aplicación SKoA (SCADA **K**onnex **A**utoconfigurable).

El usuario podrá acudir a este manual con el fin de resolver dudas sobre la utilización de la aplicación, que serán resueltas a través de sencillas instrucciones y explicaciones que le facilitarán desenvolverse en el entorno desarrollado en la misma.

1.1 ¿Cómo usar este manual?

Este manual está estructurado de la siguiente manera:

- ❖ En primer lugar realizaremos una descripción general de cada una de las ventanas con las que el usuario podrá interactuar (*Guía rápida de funciones principales del software*)
- ❖ Seguidamente se describirán con detalle la funcionalidad asociada a cada interfaz de la aplicación.
- ❖ Finalmente veremos una explicación de la simbología usada y el menú de versión de la aplicación y los autores de la misma.

1.2 ¿Qué es SKoA?

SKoA (**Scada-Konnex-Autoconfigurable**) es una aplicación informática de tipo SCADA que se puede ejecutar tanto en entornos Windows como Linux, y que dispone de una interfaz gráfica destinada al control, monitorización y visualización en tiempo real sin intervención humana de los dispositivos domóticos Konnex instalados en cualquier edificio.

Esta aplicación se puede personalizar para cualquier instalación Konnex a través de un fichero de configuración que una vez generado permite autoconfigurar la aplicación de forma automática. Los archivos de configuración representan la estructura arquitectónica del edificio y todos los dispositivos domóticos instalados en el mismo.

Para facilitar la labor de generación del fichero de autoconfiguración, SKoA incluye un módulo software que permite generar o modificar este fichero de forma gráfica, sencilla e intuitiva.

Adicionalmente, SKoA permite enviar mensajes de alarma o monitorización de forma automática a través de correos electrónicos.

Una última característica relevante de esta aplicación es la posibilidad de generar diagramas gráficos donde se muestran la evolución temporal de los datos enviados por el bus KNX y que estén asociados a una determinada dirección de grupo. De esta forma, se facilita el análisis temporal del funcionamiento de la instalación domótica.

1.3 Guía rápida de funciones principales del software

- **Interfaz de Selección de Idioma:** Es la primera ventana con la que se encontrará un usuario al ejecutar la aplicación. El usuario deberá seleccionar el idioma en el que desea visualizar la aplicación, pudiéndose escoger entre los idiomas español e inglés.



Figura A.1 Interfaz de Selección de Idioma

- **Interfaz Principal:** Es la ventana con la que se encontrará el usuario tras seleccionar un idioma. Se trata del panel donde se tendrá acceso a la mayoría de funcionalidades de la aplicación y dónde se establecerán las principales acciones a realizar por el usuario. Al no haber un proyecto previamente cargado, el título por defecto que nos mostrará como nombre del proyecto será el nombre del propio software “Software SkoA”, como se puede apreciar en la **Figura P.2:**



Figura A.2 Interfaz Principal

En la parte superior izquierda podemos observar en el menú principal las siguientes opciones:

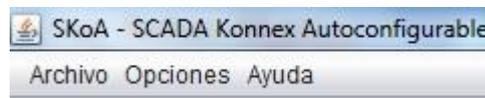


Figura A.3 Panel superior de opciones

Las acciones que encontraremos para cada opción son:

➤ **Archivo:**

- Nuevo proyecto
- Abrir proyecto
- Guardar proyecto
- Cerrar proyecto
- Conectar al bus EIB
- Desconectar del bus EIB
- Salir

- **Opciones:**
 - Editar archivo configuración
 - Contadores
 - Gráficas

- **Ayuda:**
 - Ver Configuración Actual
 - Acerca de...

Debajo del menú principal tenemos 3 secciones:



Figura A.4 Secciones panel superior

- En la parte superior izquierda encontramos los botones que nos permitirán interactuar con el bus EIB conectado a la instalación domótica. Desde estos botones podemos “Conectar” o “Desconectar” el Bus EIB.

- En la zona central tenemos una imagen que es variable; cuando se carga un proyecto, adoptará el nombre del hotel o vivienda que haya indicado el usuario.

- En la parte superior derecha se puede observar la fecha y hora actual y el estado del servidor que indica si se encuentra conectado o no.

En la zona central de la interfaz encontramos las tres pestañas principales de esta ventana:

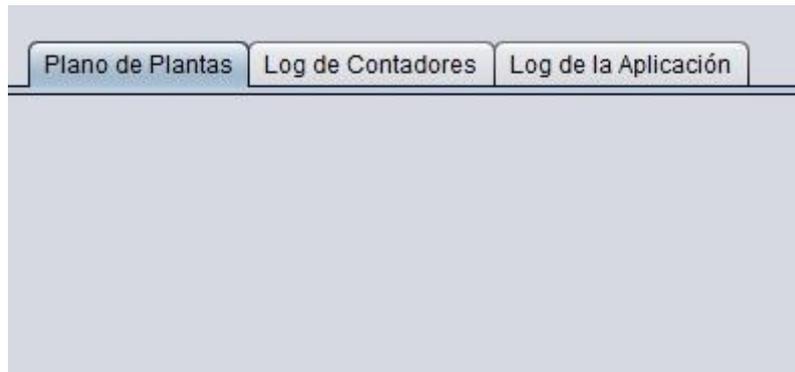


Figura A.5 Interfaz zona central

- **Planos de planta:** Mostrará visualmente un edificio en concreto, mediante pestañas verticales y horizontales. En las pestañas verticales se mostrarán las diferentes plantas del edificio y en las horizontales, las estancias que hay dentro de cada planta. El contenido de esta pestaña es variable ya que cuando se cargue un proyecto mostrará en la columna izquierda las diferentes plantas configuradas en el archivo de configuración XML correspondiente, y en la fila superior aparecerán las estancias configuradas para cada planta.

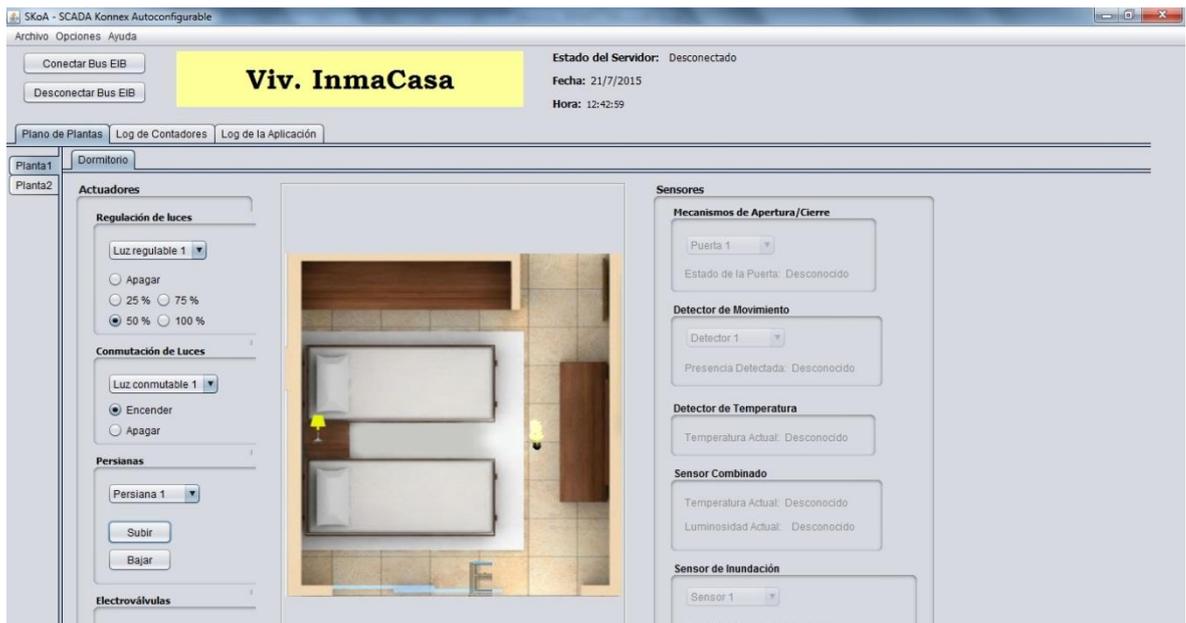


Figura A.6 Ejemplo de proyecto con plantas y estancias

Según se seleccione una planta y una estancia determinada, en la zona central se mostrará el plano asociado a dicha estancia seleccionada con los diferentes dispositivos domóticos configurados para dicha estancia. Más adelante veremos con mayor detalle la funcionalidad de esta pestaña.

- **Log de contadores:** Mostrará la actividad reflejada en los contadores, es decir, las medidas de los valores que éstos proporcionan.

- **Log de la aplicación:** Mostrará la actividad de toda la aplicación, es decir, cuando se realiza una conexión/desconexión, una petición de lectura de un contador, alguna anomalía en el programa, etc.

2. Bus EIB

2.1 Conexión Bus EIB

Para realizar la conexión con el bus EIB, el usuario puede hacerlo mediante el botón que aparece en la interfaz principal o bien mediante la opción del menú principal “Conectar al bus EIB”.

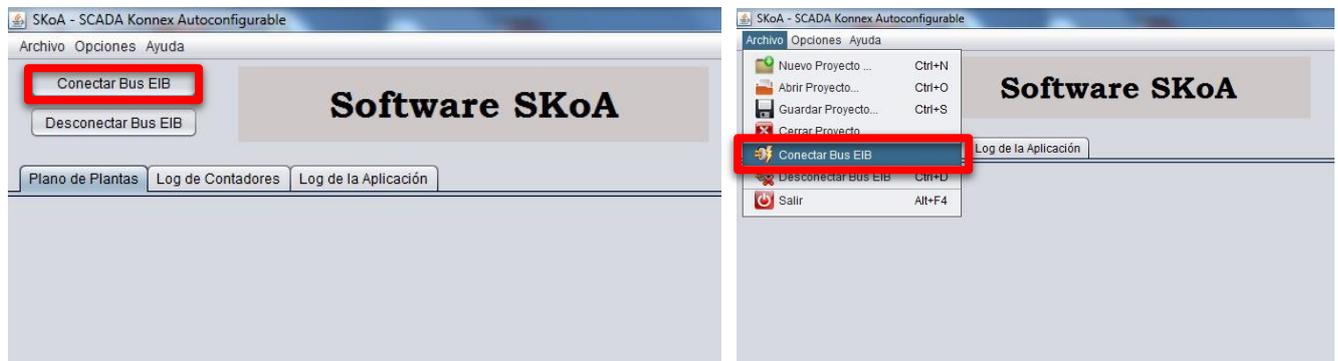


Figura A.7 Métodos para Conectar Bus EIB

En ambos casos a continuación, aparecerá una ventana en la que el usuario ha de indicar la dirección IP KNX a la que se conectará, el puerto, la dirección IP del PC cliente y un nombre de fichero, en el que se almacenará la actividad de la aplicación (mismo funcionamiento que la pestaña “Log de la aplicación” de la interfaz principal).

Figura A.8 Cuadro Conexión Bus EIB

Por defecto, todos estos parámetros ya aparecen rellenos, dependiendo de la información que haya en el fichero MCA.cnf (se verá más adelante). A pesar de esto, el usuario puede modificar cualquiera de los campos de diálogo si así lo desea.

La conexión quedará registrada en el log de la aplicación, en la pestaña de la interfaz principal y en el fichero de log almacenado en el disco duro.

2.2 Desconexión Bus EIB

Para realizar la desconexión del bus, el usuario puede realizarlo a través del botón que aparece en la interfaz principal o bien mediante la opción del menú Archivo “Desconectar del bus EIB”.

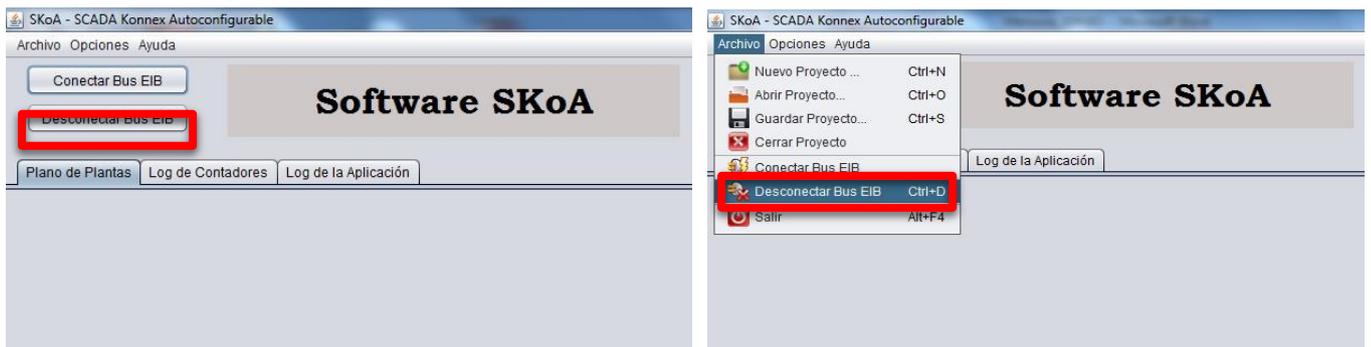


Figura A.9 Métodos para Conectar Bus EIB

La desconexión quedará registrada en el log de la aplicación, en la pestaña de la interfaz principal y en el fichero de log almacenado en el disco duro.

3. Implementación de Proyectos

3.1 Crear un nuevo proyecto

Para crear un proyecto nuevo el usuario tendrá que seleccionar la opción “Nuevo Proyecto” del menú Archivo, del menú principal:

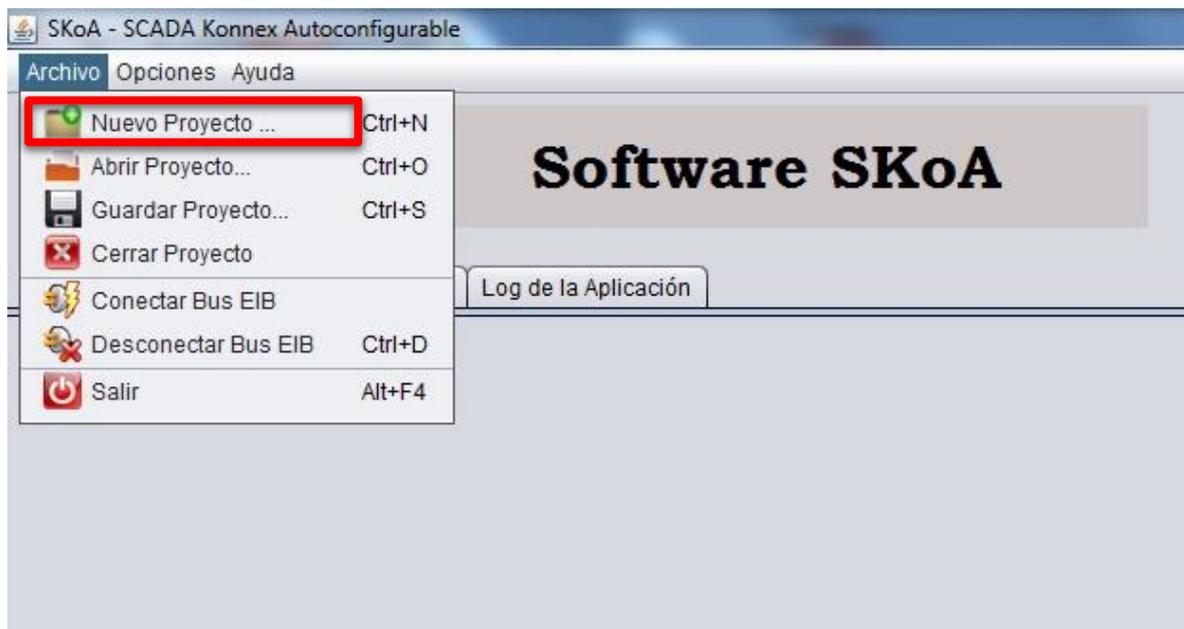


Figura A.10 Pestaña Nuevo Proyecto

A continuación aparecerá la siguiente ventana en la cuál podremos comenzar a crear un nuevo fichero:

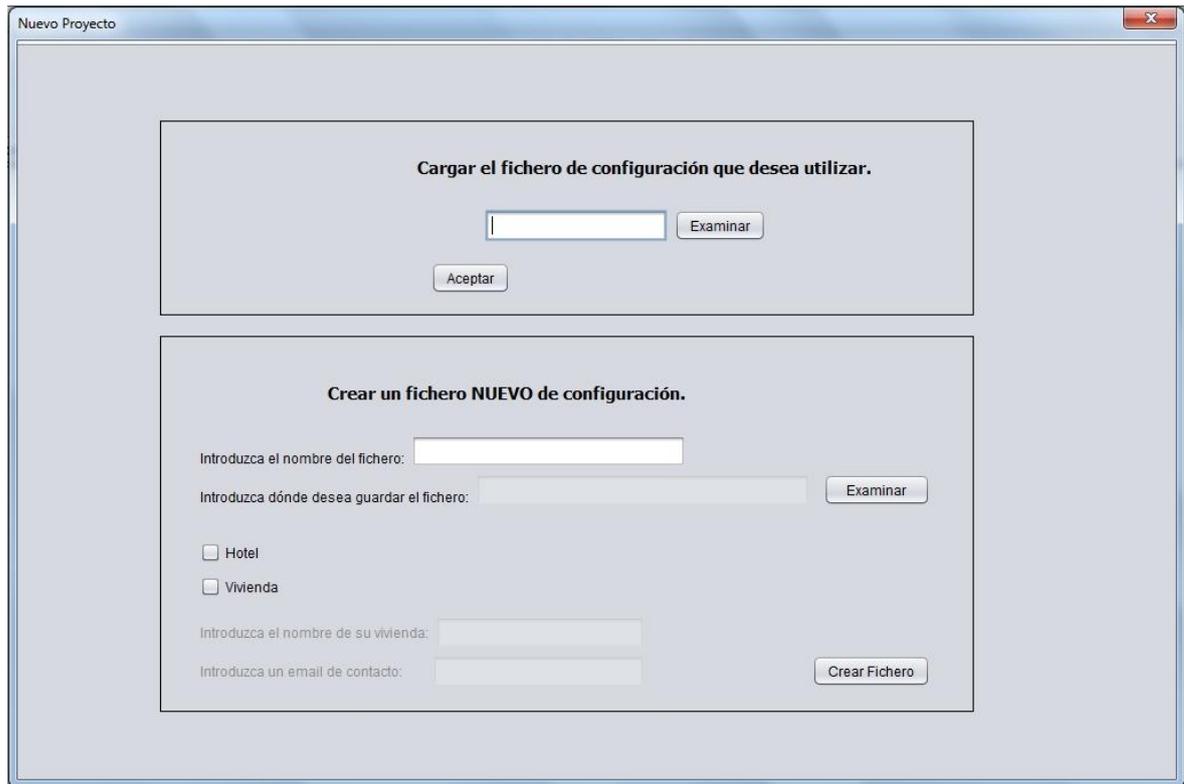


Figura A.11 Ventana Nuevo Proyecto

Para crear un nuevo fichero de configuración el usuario deberá introducir la siguiente información en el formulario:

- **Nombre del fichero:** el nombre del fichero principal de configuración (.XML). Este fichero será el que posteriormente solicite el programa para abrir un proyecto.
- **Directorio del fichero:** el directorio en el que se guardará dicho fichero.
- **Hotel / Vivienda:** el usuario puede seleccionar si se trata de un hotel o una vivienda. Esto tan solo afecta al archivo de configuración .XML
- **Nombre del hotel/vivienda:** el nombre del hotel o vivienda que se desea configurar. Dicho nombre será el que aparezca en la parte superior central de la interfaz, sustituyendo a las palabras “Software SKoA”.
- **Email de contacto:** la dirección de correo electrónica a la que se enviarán

los mails de notificación en caso de producirse alguna alarma en el software.

Además se el usuario puede cargar un fichero ya creado con anterioridad también en este apartado, indicando el fichero correspondiente en el formulario “*Cargar fichero de configuración que desea utilizar*”

3.2 Fichero de Configuración XML

El Fichero de configuración será el lugar donde estarán almacenadas todas las características de un edificio: nombre del edificio, email de la persona responsable, número de plantas, número de estancias por plantas y dispositivos domóticos asociados a cada estancia. Todas las características de este fichero pueden ser modificadas en el módulo de configuración de SKoA. Y a su vez, todas las características configuradas en el fichero podrán ser visualizadas en el módulo de control y monitorización de SKoA.

Una vez rellenados los diferentes campos del formulario para crear un nuevo fichero de configuración, habrá que pulsar en el botón “Crear Fichero”, procediendo a la apertura la ventana del módulo de configuración:

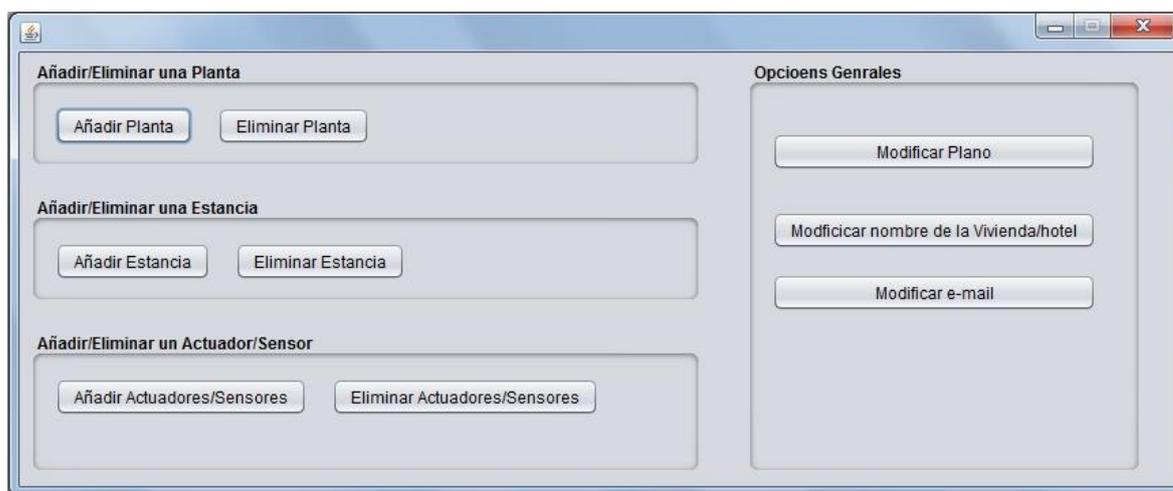


Figura A.12 Módulo de configuración

Para modificar el contenido del fichero de configuración encontraremos las

siguientes opciones:

1.2.1 Añadir planta

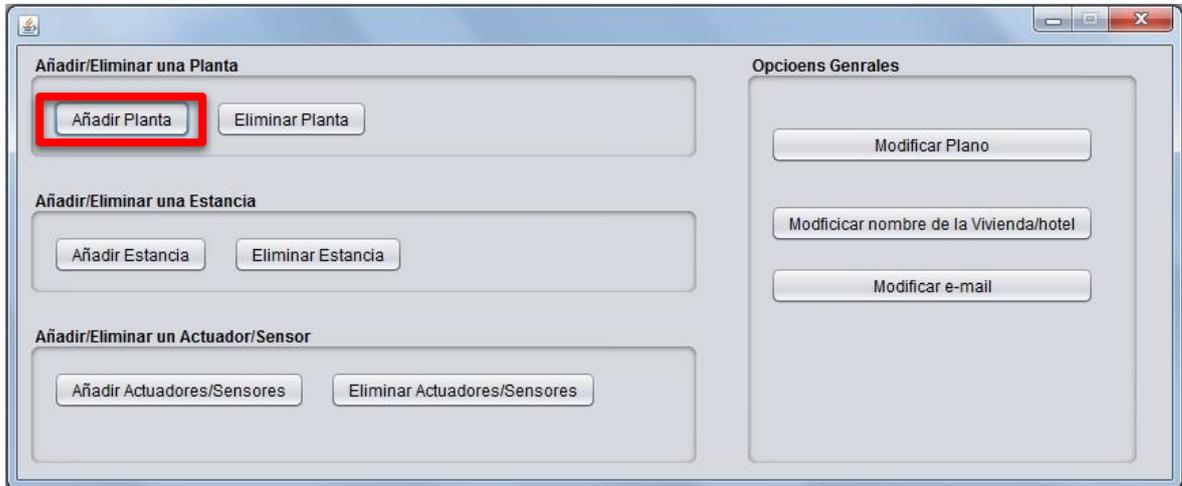


Figura A.13 Apartado “Añadir Nueva Planta” del Módulo de Configuración

Si el usuario desea añadir una planta, una vez seleccionada esta opción en el “menú de configuración”, tan solo debe escribir el nombre de la nueva planta en el campo correspondiente y hacer click en el botón “Añadir Planta”.

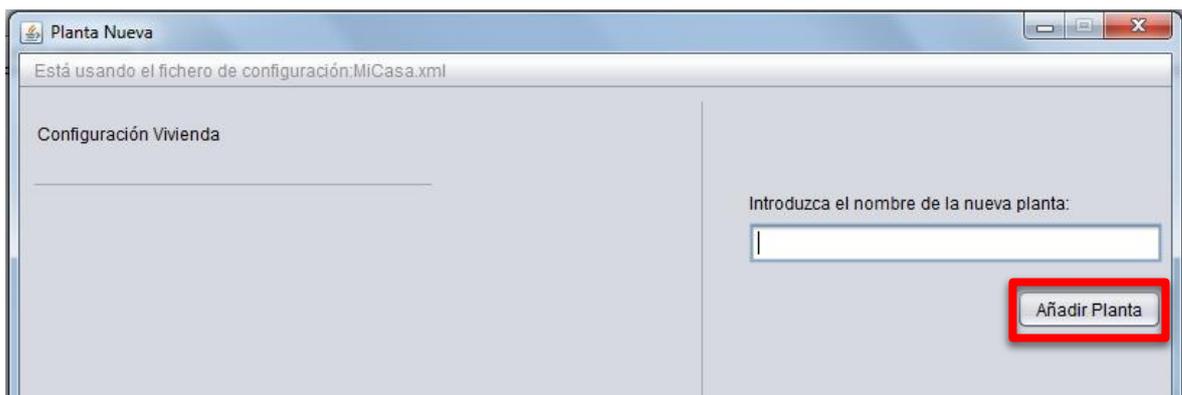


Figura A.14 Módulo “Añadir Planta”

Tras un mensaje de confirmación, la nueva planta se añade al fichero de configuración pudiéndose comprobar en la parte izquierda de la pantalla.

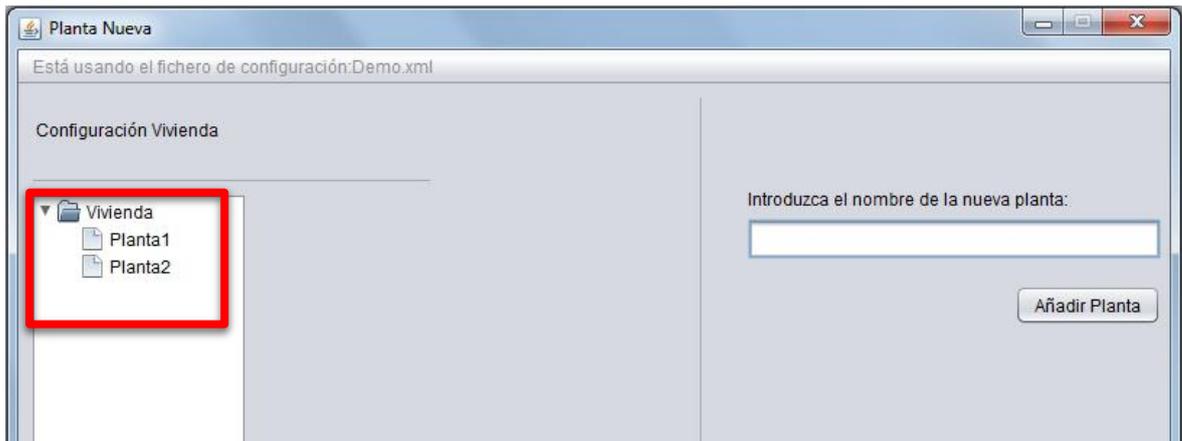


Figura A.15 Ejemplos de Plantas creadas

1.2.2 Eliminar planta

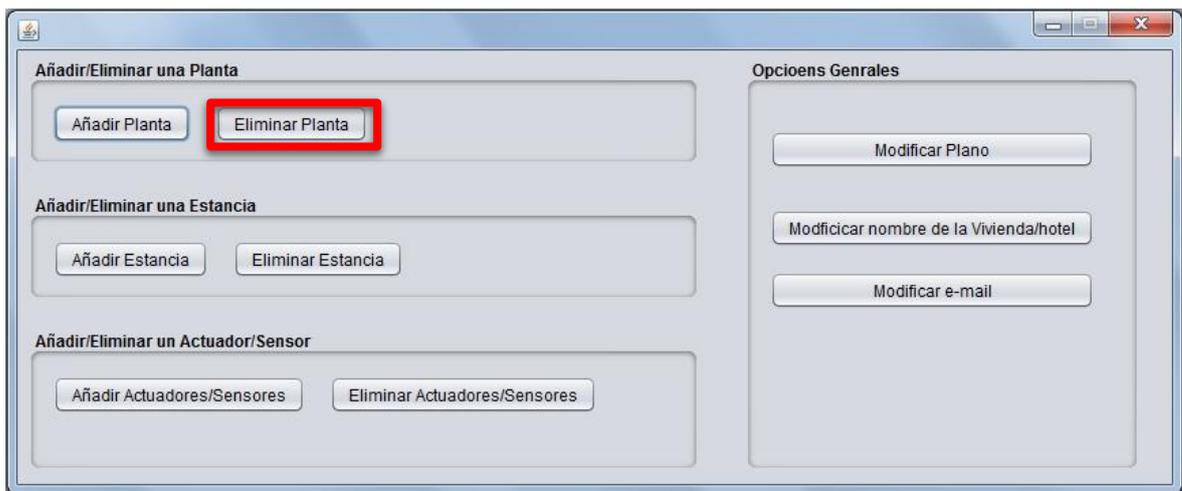


Figura A.16 Apartado "Eliminar Planta" del Módulo de Configuración

Si un usuario desea eliminar una planta, una vez seleccionada esta opción en el "menú de configuración", tan solo debe escoger una planta de entre las disponibles de la lista desplegable y pulsar en el botón "Eliminar Planta".

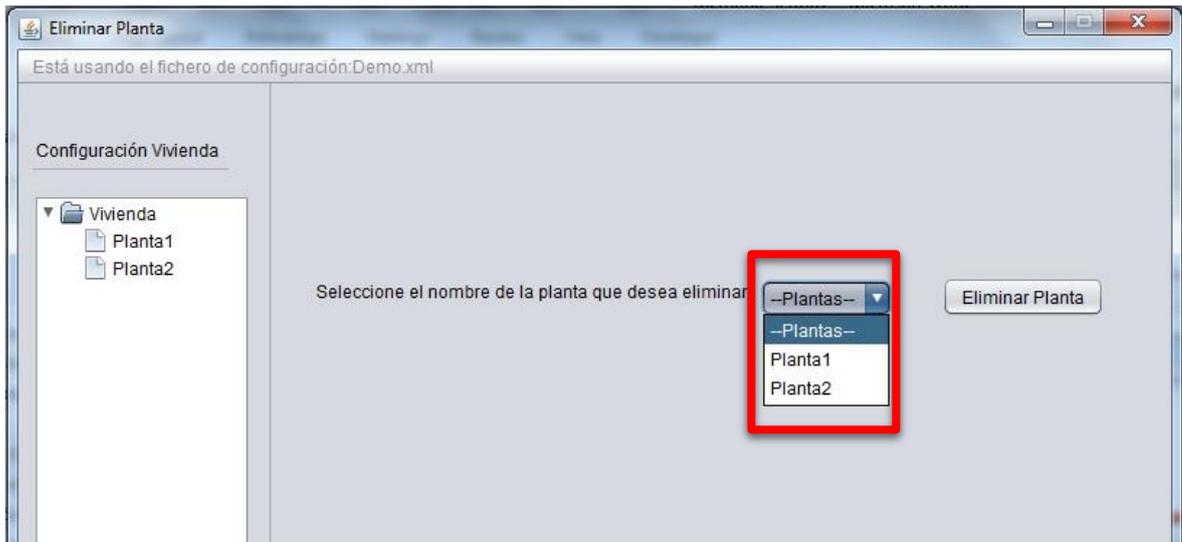


Figura A.17 Desplegable con las Plantas creadas en el proyecto

Al ser una **operación que conlleva cierto riesgo**, ya que eliminar una planta implica eliminar todas las estancias que se encuentran en ella, aparecerá un mensaje de confirmación. Si la respuesta es afirmativa, ésta desaparecerá de la estructura arbórea situada a la izquierda de la pantalla y del fichero de configuración XML.

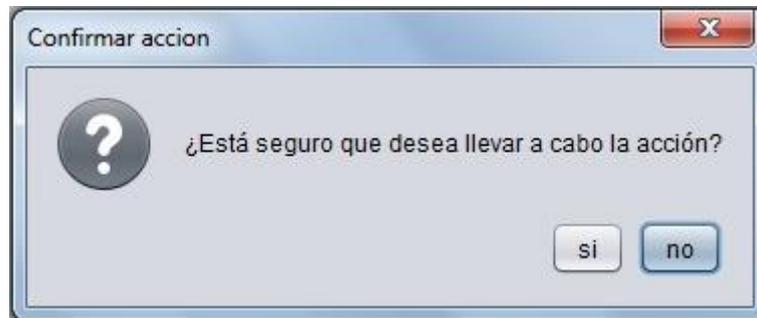


Figura A.18 Mensaje de advertencia mostrado al eliminar una Planta

1.2.3 Añadir estancia

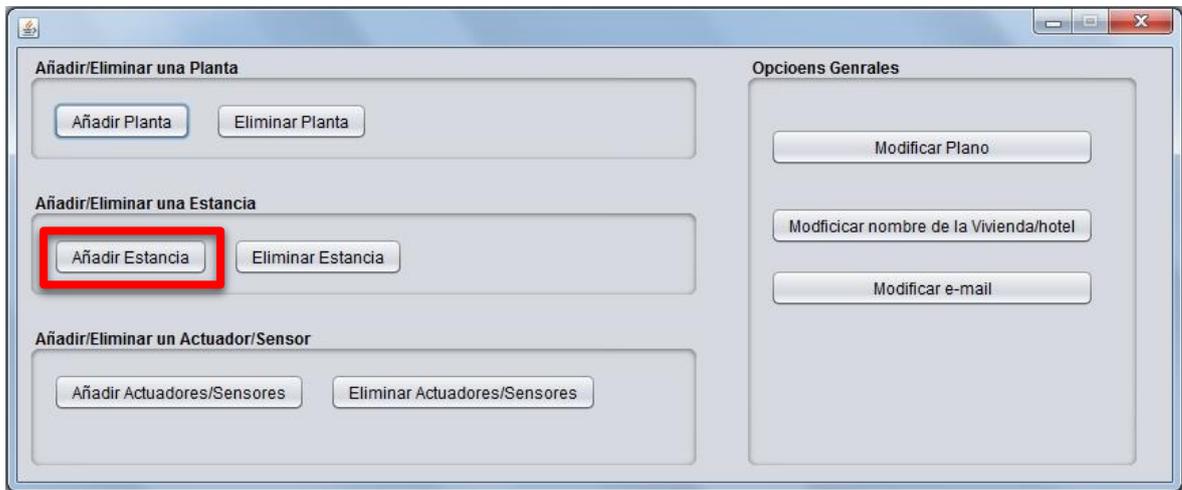


Figura A.19 Apartado “Añadir Nueva Estancia” del Módulo de Configuración

Una vez que se han añadido algunas plantas (como mínimo una) es posible agregar estancias a cada planta. Una vez seleccionada la opción “Añadir Nueva Estancia” en el “menú de configuración”, el usuario debe seleccionar la planta en la que se añadirá la estancia, elegir el nombre que representará a dicha estancia y seleccionar una imagen que servirá para representar el plano de la misma. Después, bastará con pulsar en el botón “Añadir Estancia”.



Figura A.20 Método para añadir una nueva Estancia en una Planta existente

Posteriormente podremos visualizar como la estancia se ha añadido correctamente a

la planta indicada:



Figura A.21 Visualización de la nueva Estancia añadida

1.2.4 Eliminar estancia

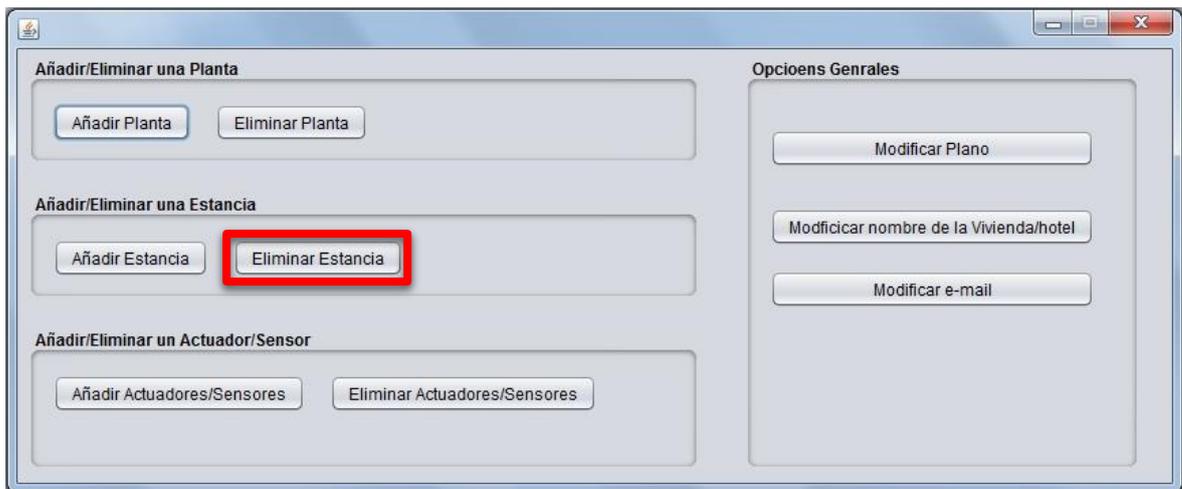


Figura A.22 Apartado “Eliminar Estancia” del Módulo de Configuración

Si el usuario desea eliminar una estancia existente, una vez seleccionada esta opción en el “menú de configuración”, a continuación debe seleccionar la planta a la que pertenece la estancia que se desea eliminar, así como la estancia:

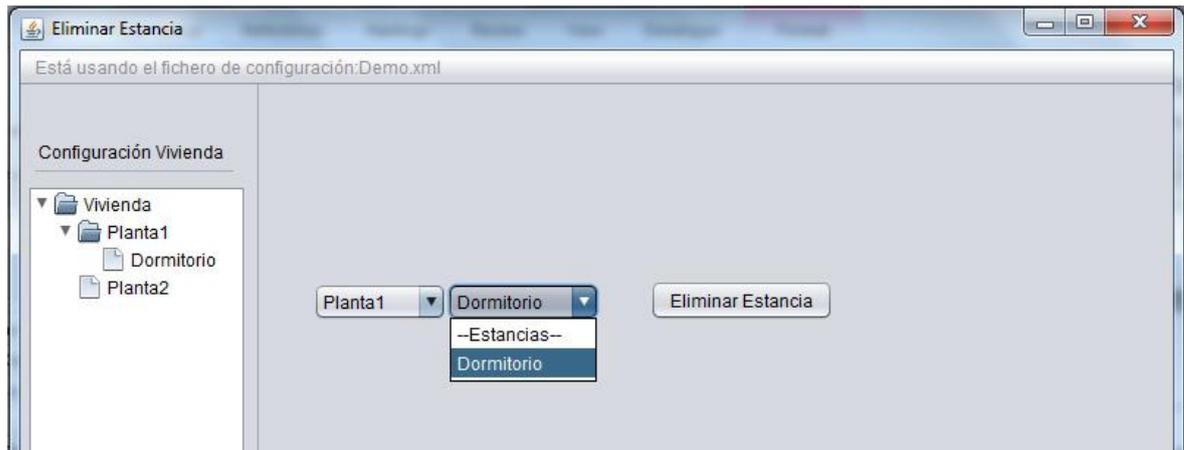


Figura A.23 Método para eliminar una Estancia existente

Una vez hecho esto, y tras pulsar en el botón “Eliminar Estancia” aparecerá una pregunta de confirmación, al ser esta una operación que conlleva cierto riesgo, ya que eliminar una estancia implica eliminar todos los dispositivos que se encuentran en ella. Si el usuario responde afirmativamente, la estancia habrá sido eliminada.

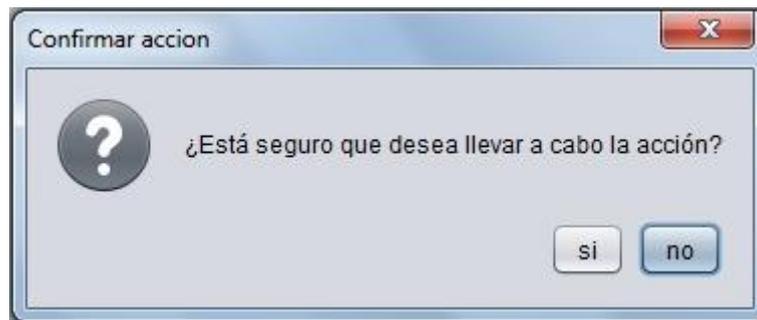


Figura A.24 Mensaje de advertencia mostrado al eliminar una estancia

1.2.5 Añadir actuadores/sensores

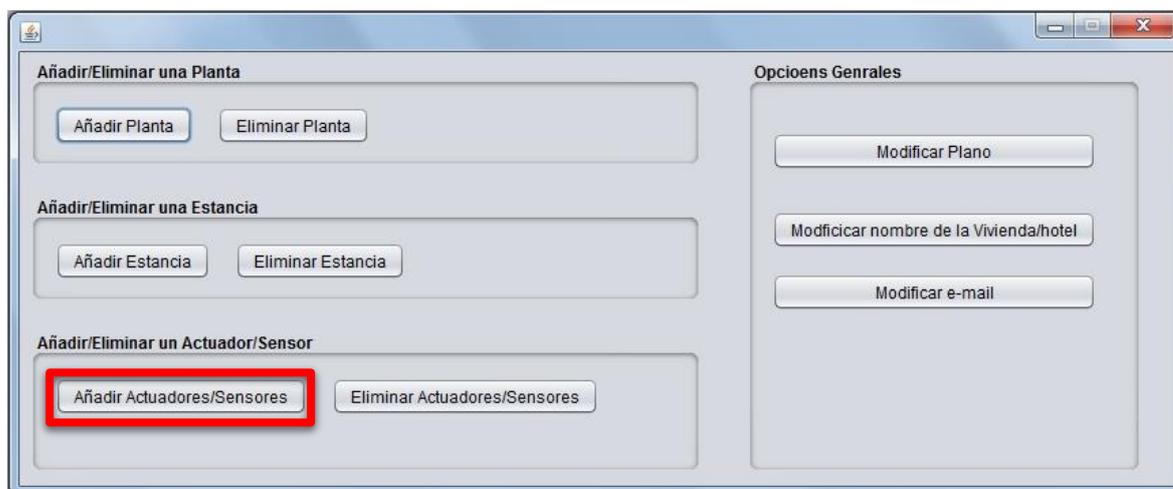


Figura A.25 Apartado “Añadir Actuadores/Sensores” del Módulo de Configuración

Cuando el usuario ya dispone de plantas y estancias en la configuración de su edificio, puede añadir actuadores y sensores dentro de cada estancia.

Una vez seleccionada esta opción en el “menú de configuración”, el usuario ha de elegir una planta y una estancia en concreto sobre la que se trabajará. A continuación aparecerá una imagen central que representa el plano de la estancia (dormitorio en el ejemplo). A ambos lados de dicha imagen, aparecerán los actuadores y los sensores, por Ej.: luces regulables, persianas, puertas, contadores, etc. que se representarán a través de iconos gráficos.

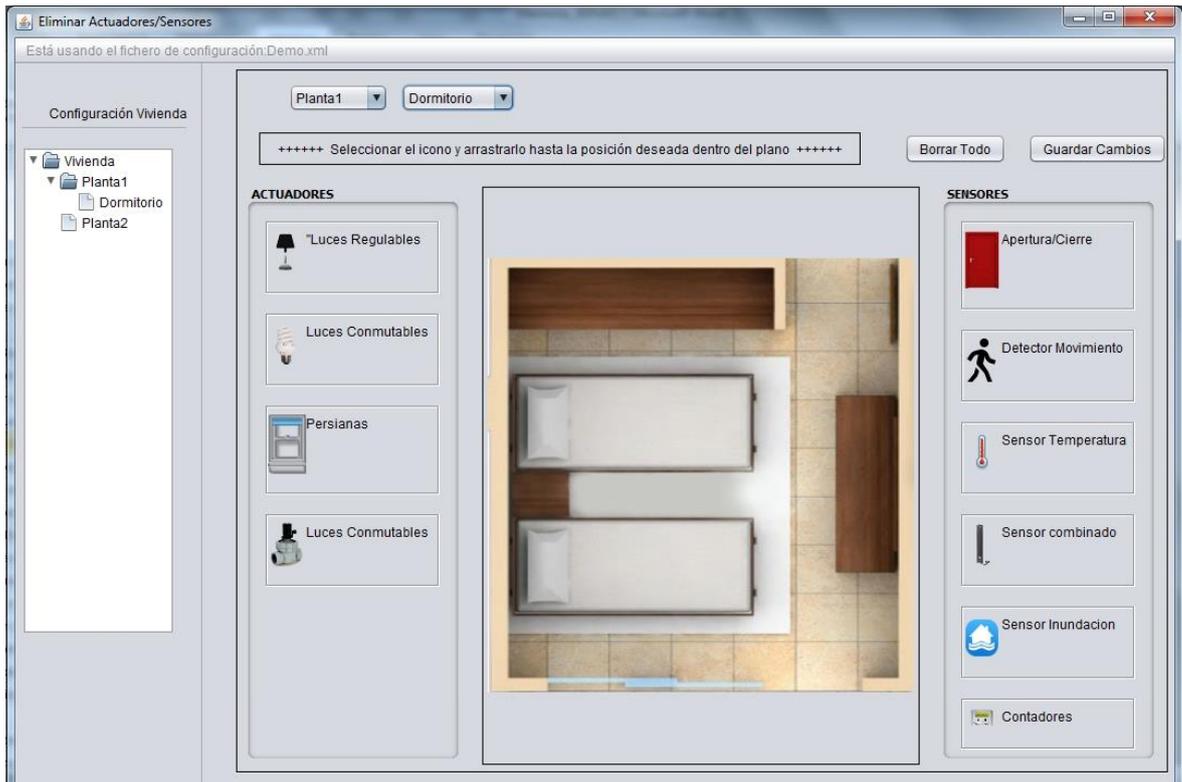


Figura A.26 Interfaz del Módulo de Configuración de Actuadores/Sensores

El sistema utilizado para colocar los dispositivos en el plano de la estancia, es el conocido como “Drag & Drop”, es decir, arrastrar y soltar.

El usuario debe pulsar, con el botón izquierdo del ratón, sobre el icono que desea utilizar, mantener pulsado el botón del ratón, moverse hasta la posición del plano de la estancia donde desea colocar el dispositivo y soltar el botón del ratón.

En el momento en el que se suelta el botón del ratón, aparecerá una ventana para introducir los parámetros correspondientes al elemento insertado.

Estos son: el tipo, subtipo, nombre y dirección de grupo del actuador/sensor añadido. El nombre es de libre elección, siempre que no sobrepase los 6 caracteres (limitación de la librería Calimero).

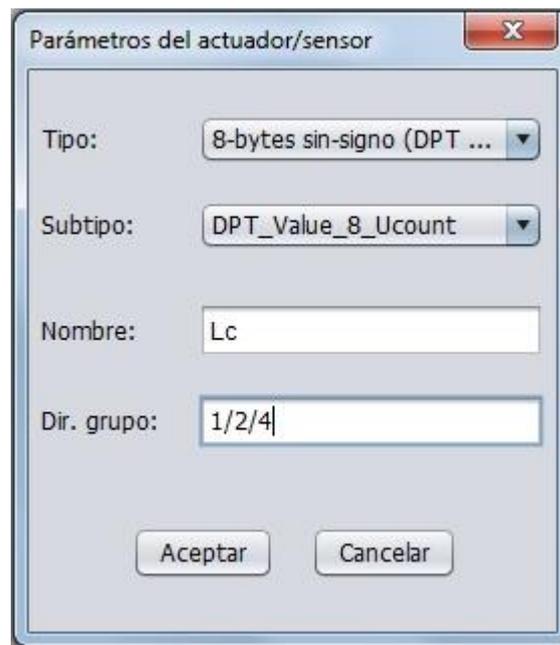


Figura A.27 Ventana de configuración de parámetros de actuadores/sensores

Si los parámetros han sido introducidos de forma correcta, el dispositivo seleccionado se colocará en la ubicación del plano de la estancia que se le haya indicado:

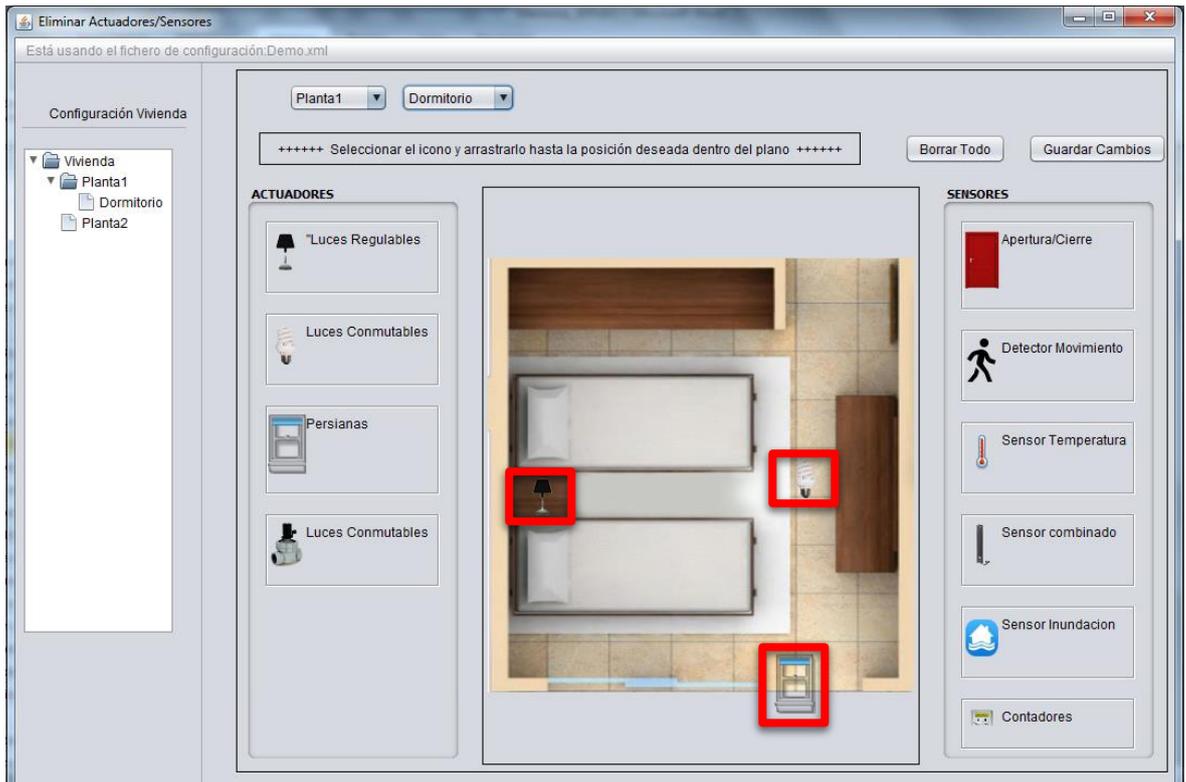


Figura A.28 Actuadores/Sensores colocados sobre el plano de la estancia

Cuando se ha terminado de insertar los distintos actuadores/sensores, el usuario tiene la posibilidad de guardar los cambios efectuados mediante el botón “Guardar Cambios” o bien puede deshacer dichos cambios mediante el botón “Borrar Todo”.

Este último botón elimina todos los cambios que se han realizado y que no hayan sido guardados con anterioridad.

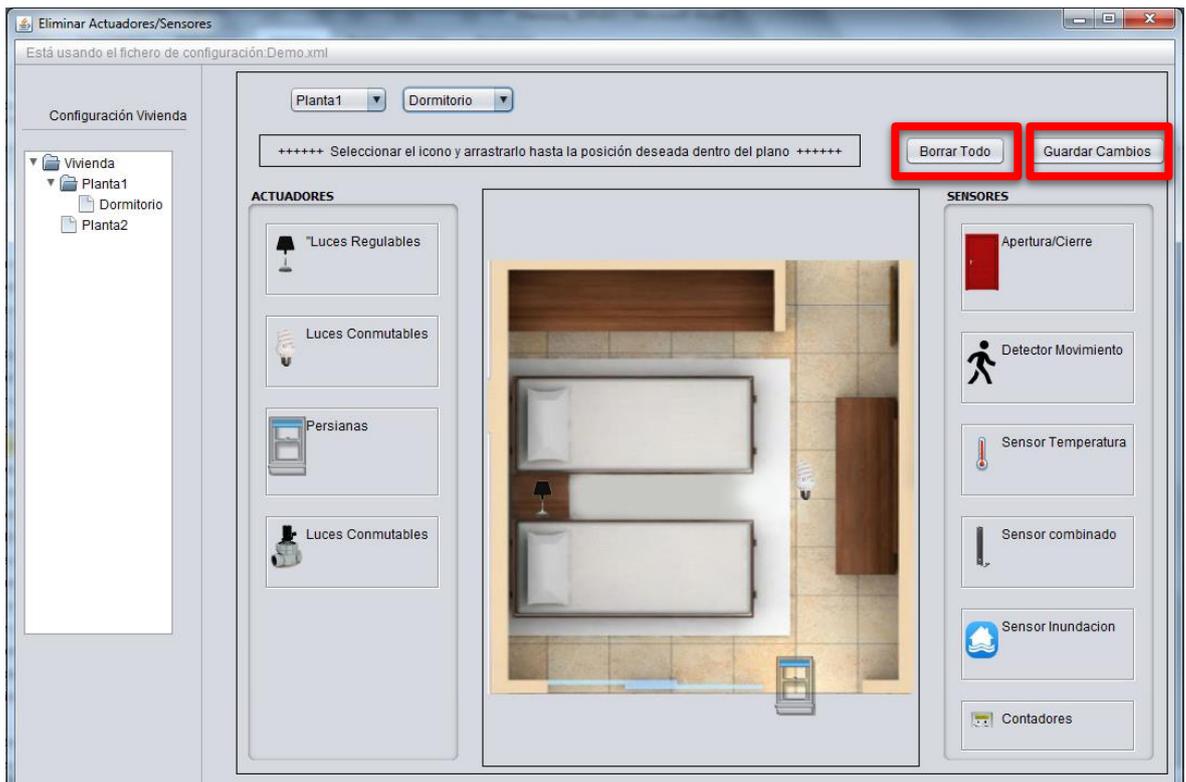


Figura A.29 Botones “Guardar Cambios” y “Borrar todo”

1.2.6 Eliminar actuadores/sensores

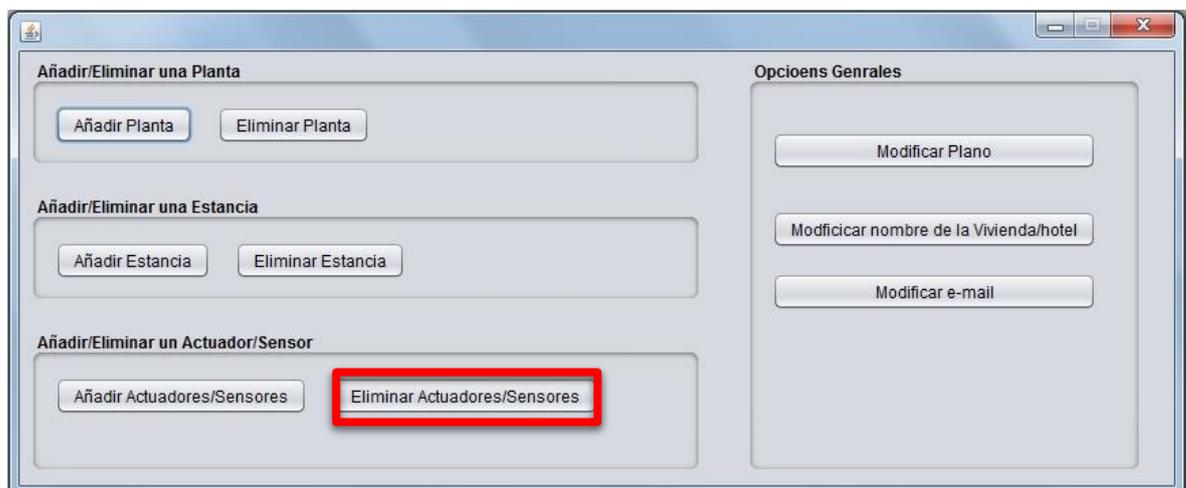


Figura A.30 Apartado “Eliminar Actuadores/Sensores” del Módulo de Configuración

Si el usuario desea eliminar algún actuador/sensor, una vez seleccionada esta opción en el “menú de configuración”, a continuación deberá seleccionar la planta en la que se ubica la estancia, así como la estancia en la que se ubican los actuadores / sensores. Por último, para eliminar un dispositivo bastará con pulsar en la X de color rojo que aparecerá

al lado de cada dispositivo.

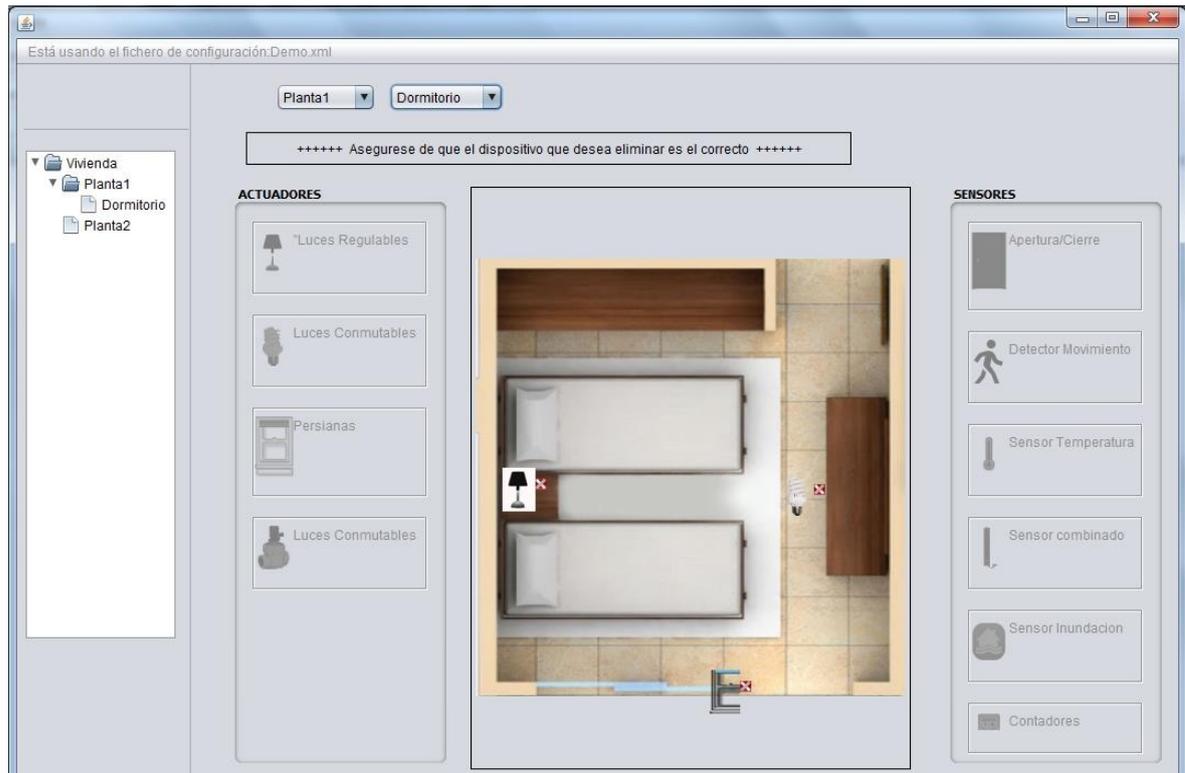


Figura A.31 Módulo “Eliminar Actuadores/Sensores” del Módulo de Configuración

En este caso, al no ser tan destructiva la operación, no se realizará una pregunta de confirmación, simplemente aparecerá un mensaje informativo para que el usuario tenga conocimiento de que se ha eliminado un dispositivo.

1.2.7 Modificar Plano

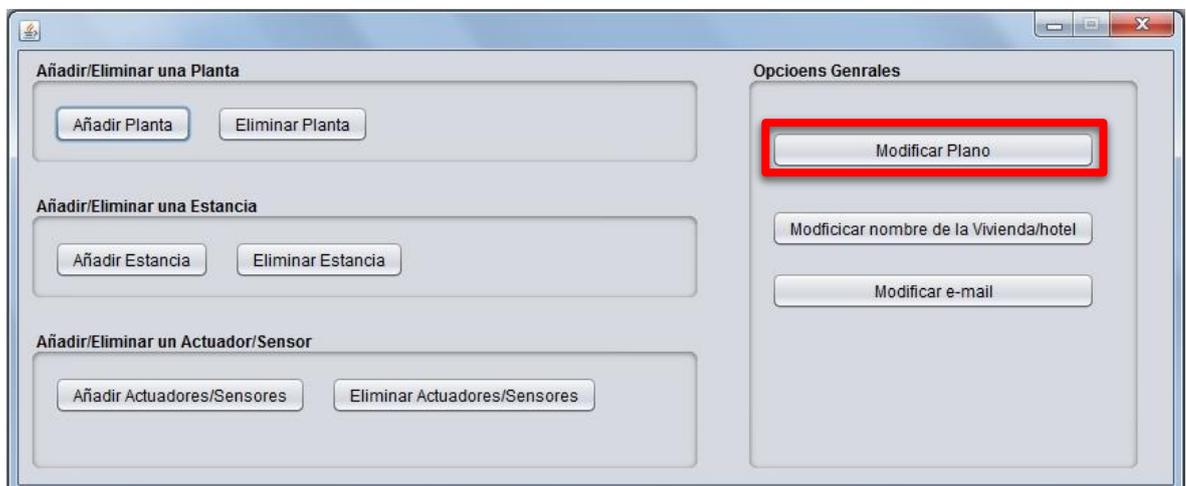


Figura A.32 Apartado “Modificar Plano” del Módulo de Configuración

En este apartado podremos modificar un plano subido con anterioridad.

Para ello accederemos al módulo haciendo click en el botón “Modificar Plano” y una vez dentro del módulo seleccionaremos la planta y la estancia correspondiente, a la vez que cargaremos un nuevo plano desde nuestros archivos.

Una vez hecho esto, pulsamos el botón “Modificar Planos” y la modificación será realizada.



Figura A.33 Módulo “Modificar Plano” del Módulo de Configuración

1.2.8 Modificar nombre vivienda/hotel

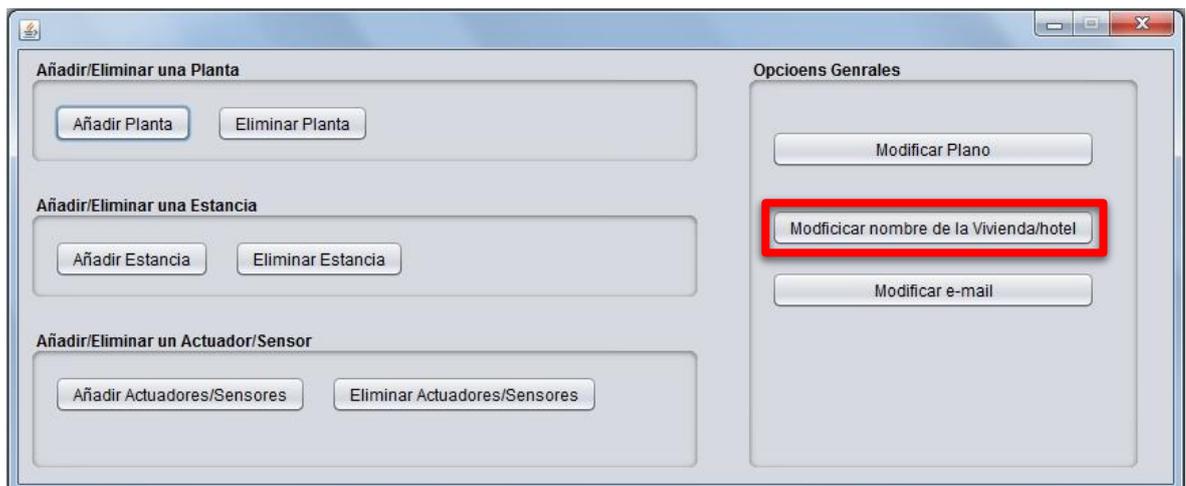


Figura A.34 Apartado “Modificar nombre de la Vivienda/Hotel” del Módulo de Configuración

Esta opción simplemente cambia el nombre de la vivienda u hotel que se está configurando. Para modificarlo, una vez seleccionada esta opción en el “menú de configuración”, el usuario deberá escoger el nuevo nombre, y darle al botón “Aceptar”.

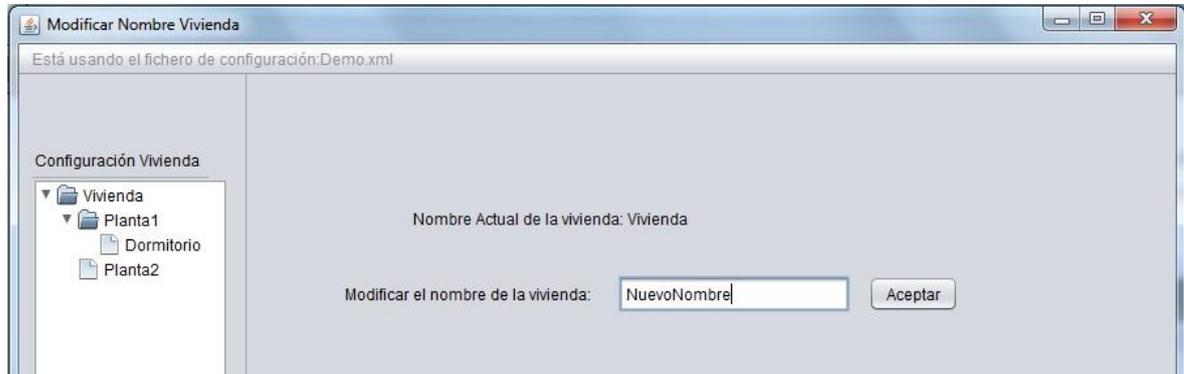


Figura A.35 Módulo “Modificar nombre de la Vivienda/Hotel” del Módulo de Configuración

Una vez introducido el nuevo nombre veremos que el mismo aparece tanto en el árbol informativo de la izquierda como en el campo que nos muestra el nombre actual.

1.2.9 Modificar dirección e-mail

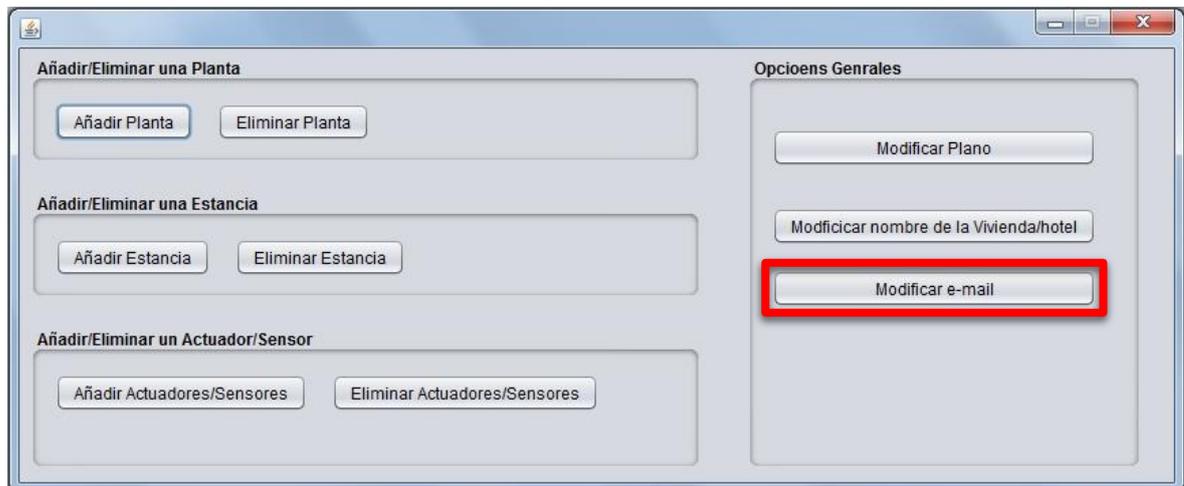


Figura A.36 Apartado “Modificar Email” del Módulo de Configuración

Esta opción permite modificar la dirección de correo electrónica (email) a la cual se enviarán los mensajes de alerta generados por la aplicación.

Para ello, una vez seleccionada esta opción en el “menú de configuración”, se deberá

insertar la nueva dirección de correo y pulsar el botón “Modificar e-mail”.

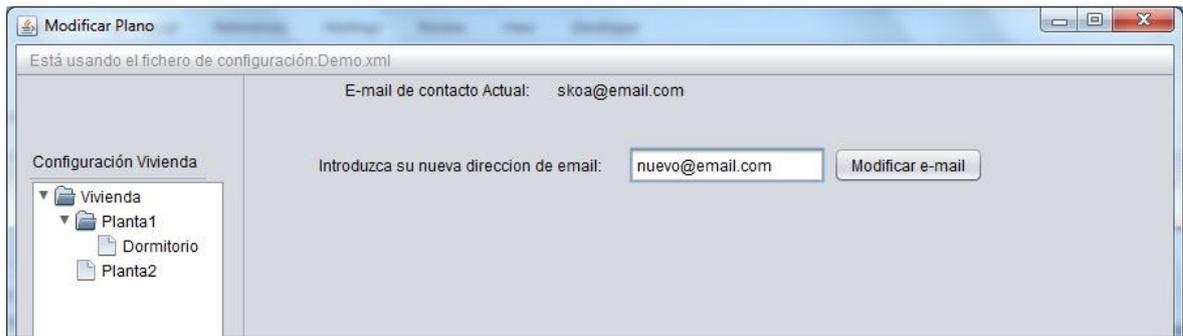


Figura A.37 Módulo “Modificar Email” del Módulo de Configuración

1.2.10 Finalización de configuración del fichero

Después de modificar todos los parámetros del archivo de configuración .XML, una vez de vuelta en la interfaz principal de la aplicación, habrá que asegurarse de guardar el fichero de los contadores y el proyecto global; para ello, seleccionaremos en el menú *Opciones* → *Contadores* → *Guardar contadores en fichero*.

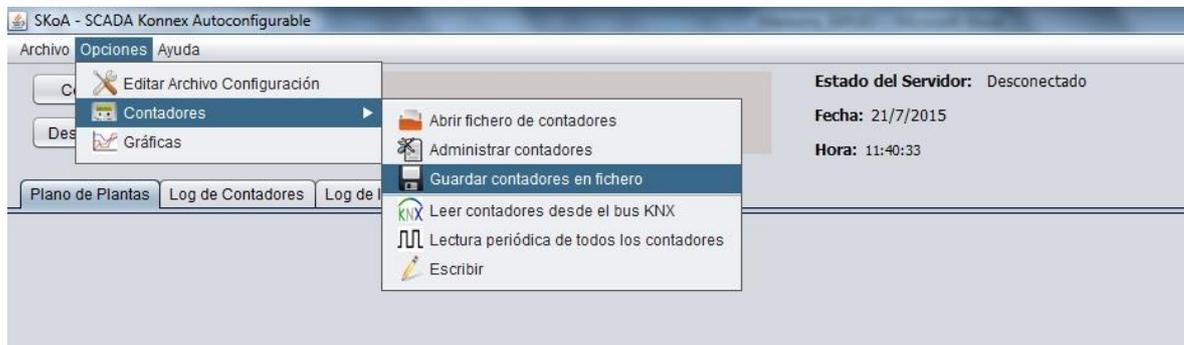


Figura A.38 Menú Contadores

A continuación tendremos que introducir un nombre para el fichero de contadores con la extensión .dat



Figura A.39 Método de guardado de fichero .dat

Por último tendremos que guardar el proyecto global. Para lo cual solo tendremos que seleccionar la opción del menú *Archivo* → *Guardar Proyecto*:

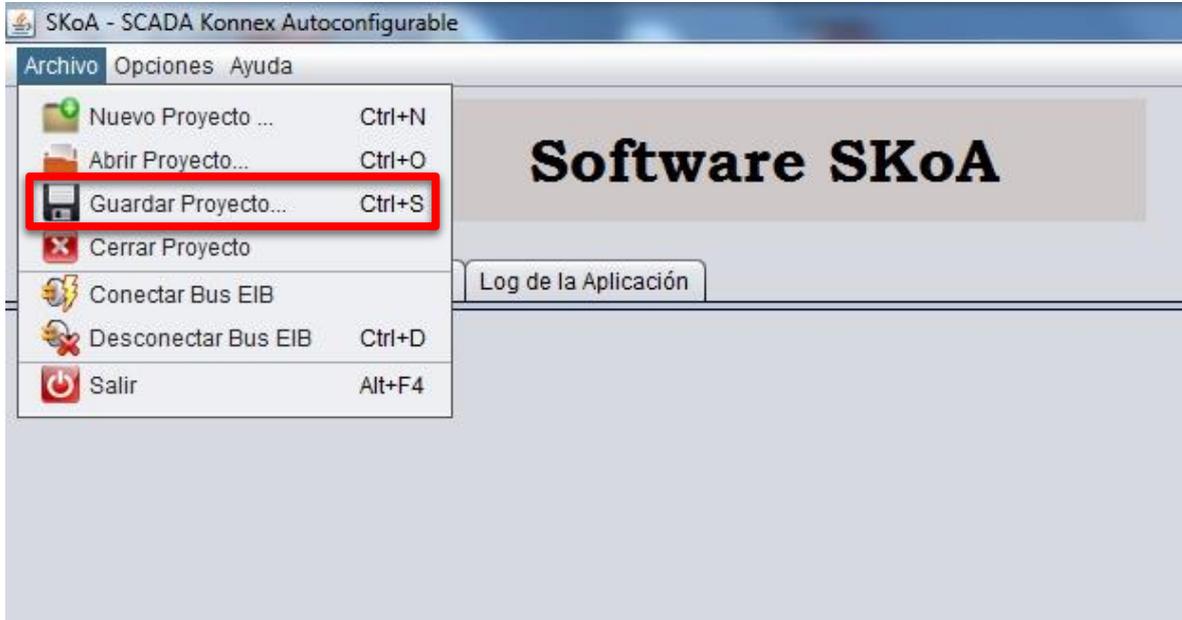


Figura A.40 Opción "Guardar Proyecto"

3.3 Abrir un proyecto existente

Una vez que se ha creado un proyecto, se han añadido las diferentes plantas estancias, actuadores, sensores, etc. se está en condiciones de abrirlo para su control y monitorización. Todas las configuraciones anteriormente comentadas se traducen en la escritura de un fichero de configuración con extensión .XML. Al abrir un proyecto, la aplicación interpreta dicho fichero y muestra de manera gráfica a través de la pestaña “Planos de planta” situada en la interfaz principal, el estado del edificio correctamente configurado. Para ello, hay que seleccionar la opción *Archivo* → *Abrir Proyecto*:

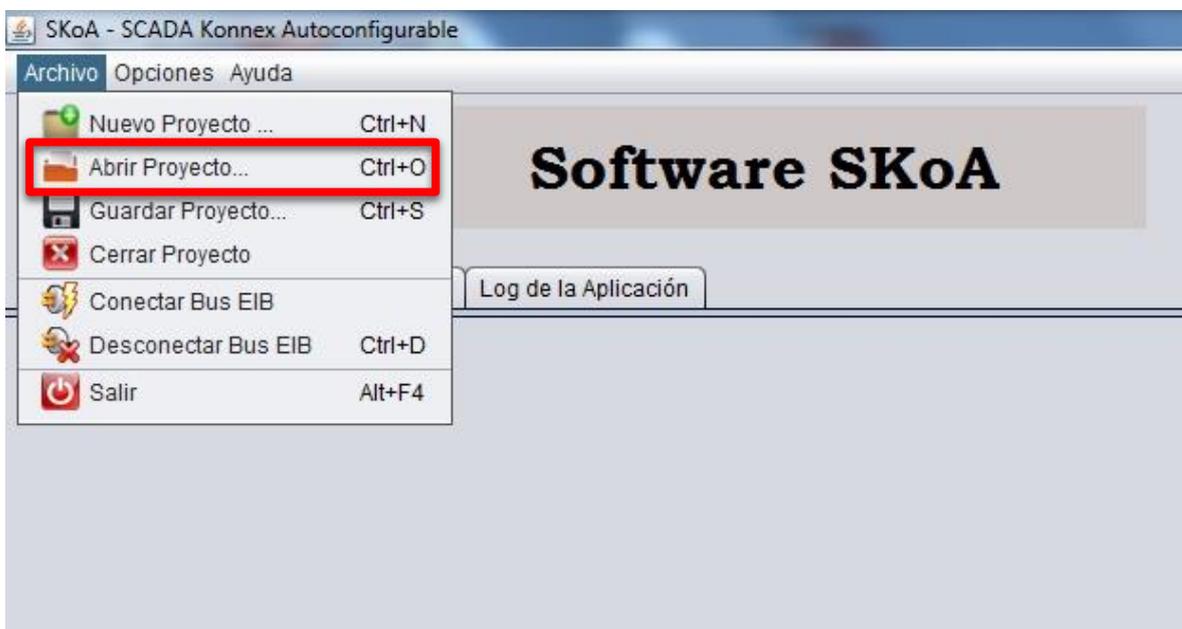


Figura A.41 Opción “Abrir Proyecto”

A continuación se abrirá un menú para seleccionar el fichero de configuración que deseamos abrir:

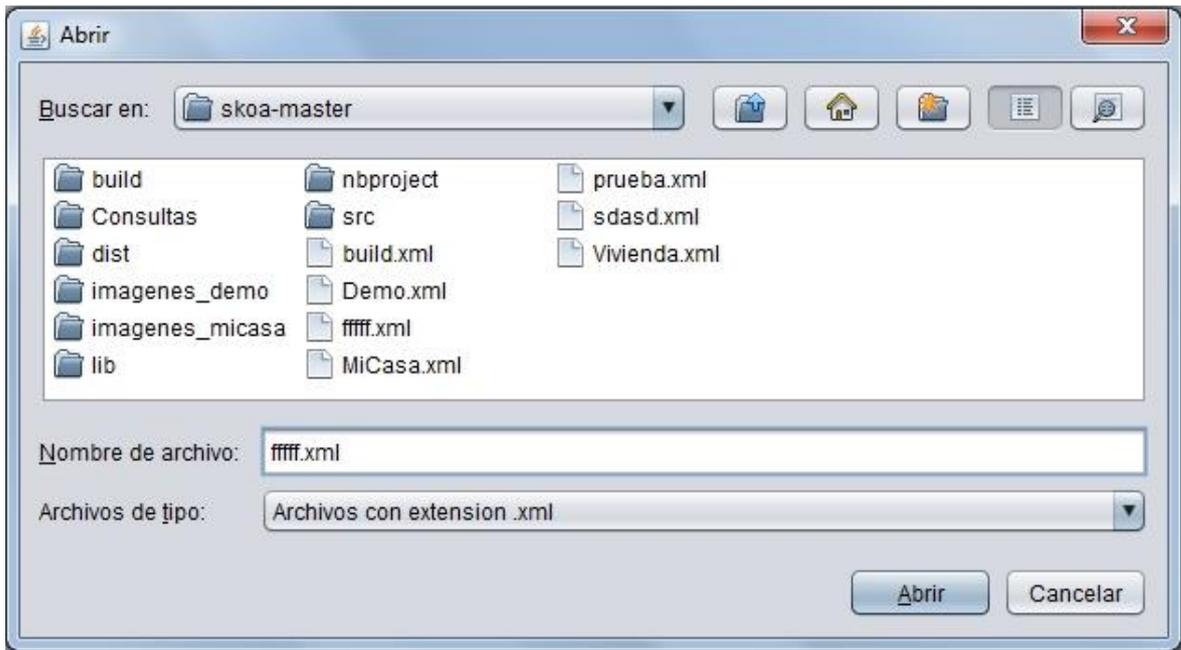


Figura A.42 Seleccionar un proyecto existente

Posteriormente veremos en la pestaña “Plano de Plantas” el contenido del fichero de configuración:

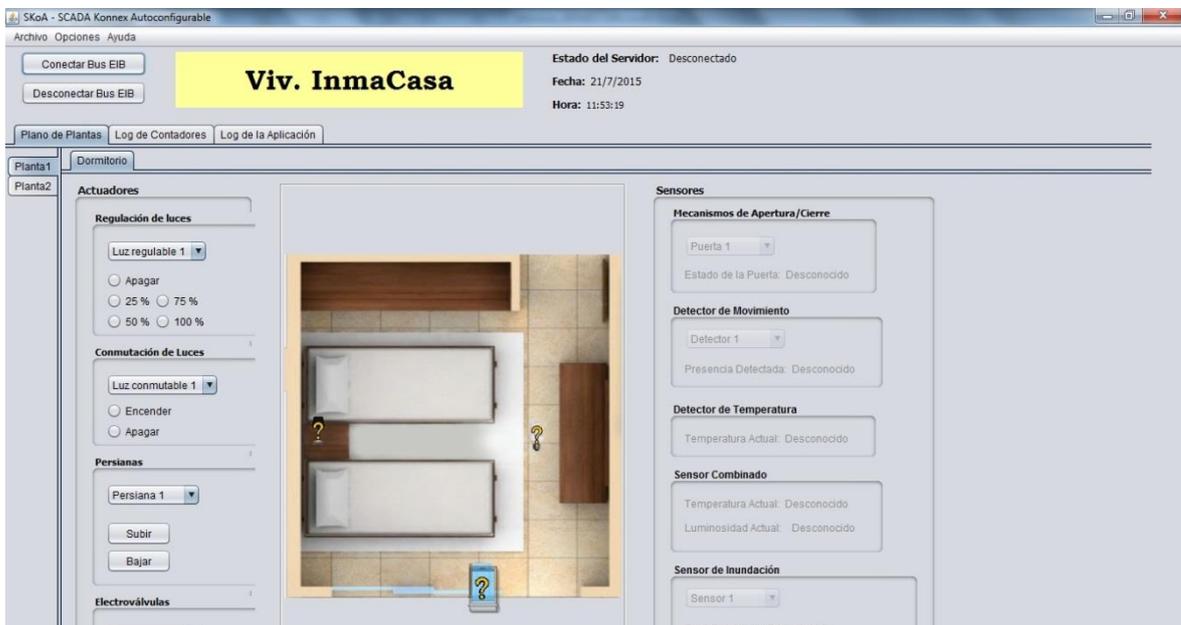


Figura A.43 Proyecto abierto en Plano de Plantas

Para poder comunicarnos con los dispositivos configurados tendremos que cargar a su vez el fichero de contadores (ver punto 5.1 *Abrir fichero base de datos*).

3.4 Editar un proyecto existente

Para editar un archivo de configuración XML existente, tendremos que seleccionar la opción del menú *Opciones* → *Editar Archivo Configuración*:

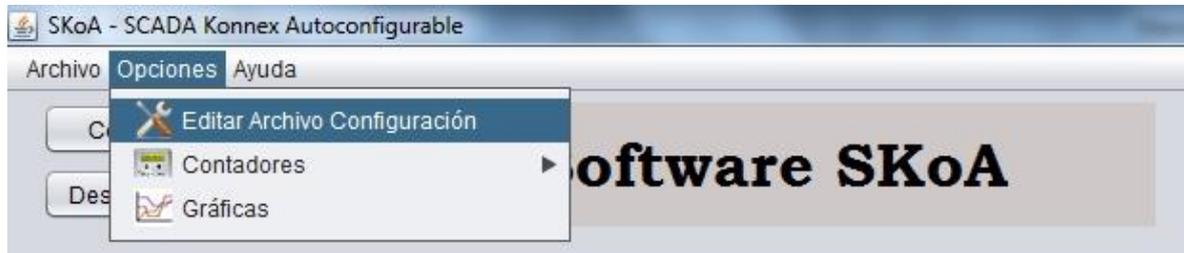


Figura A.44 Menú Opciones

A continuación veremos la ventana de selección de un archivo de configuración XML y posteriormente podremos continuar con la modificación de la configuración previa del fichero (continuar a partir de la **Figura A.10** del punto **3.2 “Fichero de Configuración XML”**).

3.5 Cerrar Proyecto

Mediante la opción *Archivo* → *Cerrar Proyecto* se cierra el proyecto actual y se liberan todos los recursos utilizados por la aplicación, de manera que se puede abrir a continuación otro proyecto o crear uno nuevo.

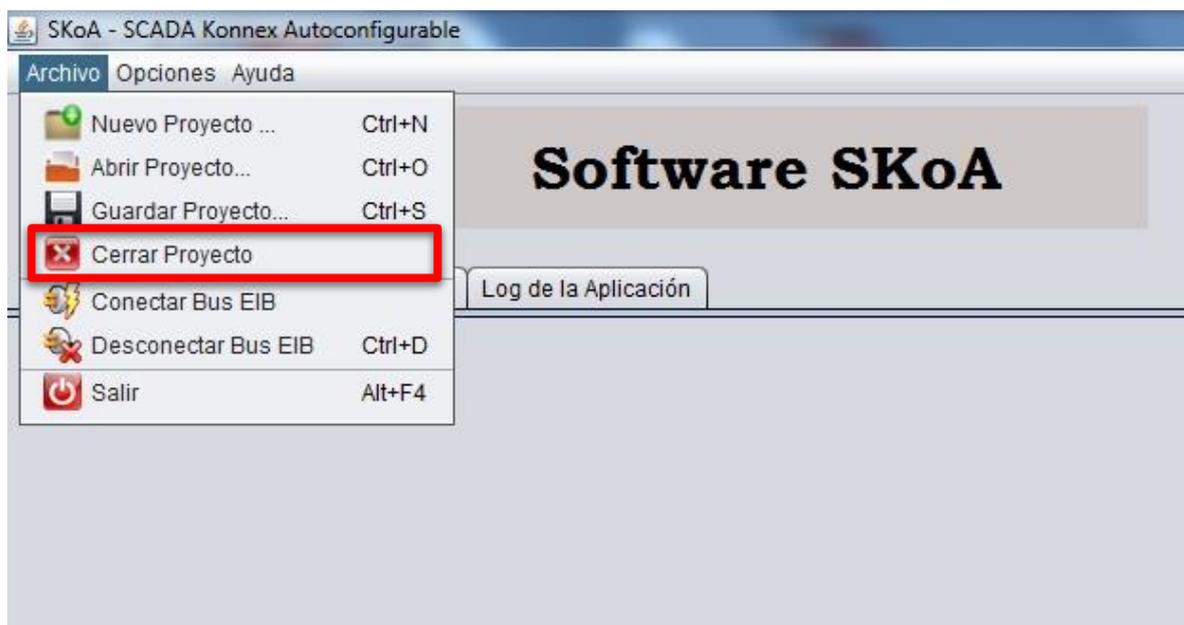


Figura A.45 Opción “Cerrar Proyecto”

4. Procesamiento Estadístico

En este apartado procederemos a explicar las funciones de Procesamiento Estadístico, aplicación secundaria del software SKoA, que muestra datos de diferentes parámetros de forma gráfica. Mediante una sencilla guía haremos un repaso a todas aquellas funciones a las que el usuario tiene acceso al utilizar esta aplicación

4.1 Guía rápida de las funciones principales de la aplicación “Procesamiento Estadístico”

En esta sección se describe de manera general cada una de las ventanas con las que se podrá encontrar el usuario.

4.1.1 Interfaz principal del módulo de procesamiento estadístico

Es la ventana que se encuentra el usuario al seleccionar este módulo desde la interfaz de la aplicación SKoA (interfaz principal).

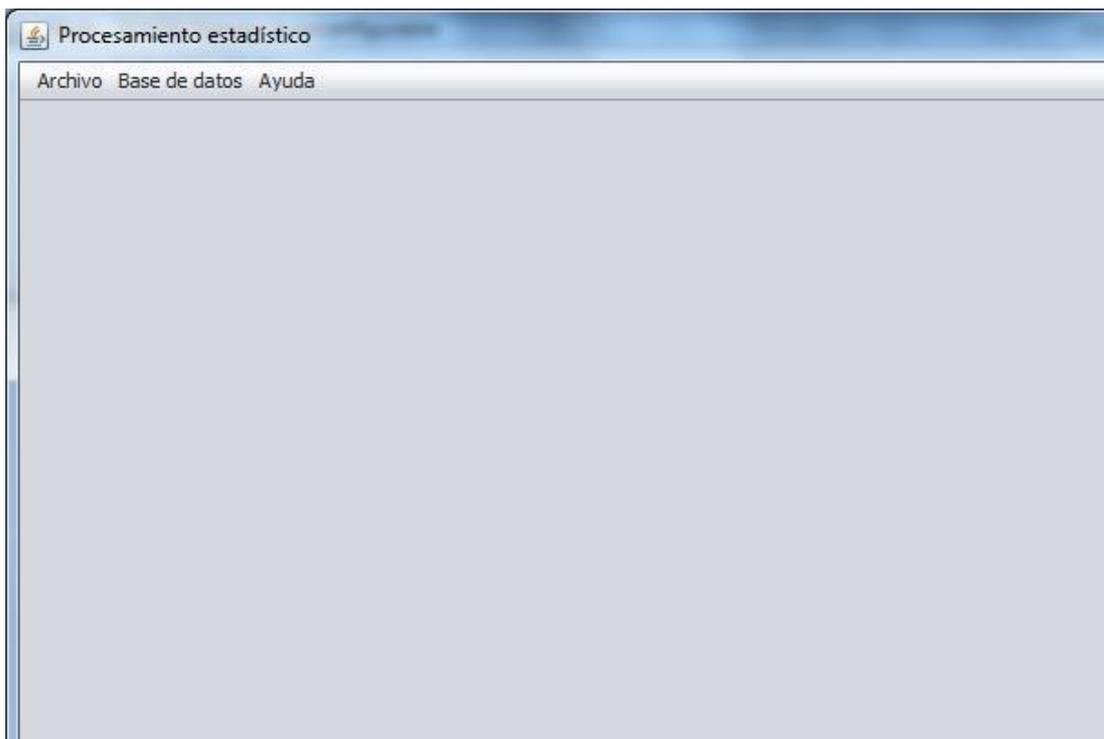


Figura A.46 Interfaz Procesamiento Estadístico

En la parte superior se puede observar el menú principal, con las siguientes opciones y acciones correspondientes:

▪ **Archivo:**

- ❖ Nuevo
- ❖ Salir

▪ **Base de datos:**

- ❖ Actualizar

▪ **Ayuda:**

- ❖ Acerca de...

4.1.2 Apertura de la aplicación “Procesamiento Estadístico”

Para abrir el módulo de procesamiento estadístico, en primer lugar debemos abrir la aplicación SKoA. Una vez nos encontremos con su interfaz principal, navegando a través de las opciones de la barra menú, seleccionamos la opción “Gráficas”, y directamente se abre una nueva ventana que se corresponde con la interfaz principal del módulo que buscamos. Cabe destacar que no es necesario haber realizado ningún paso previo al abrir la aplicación SKoA.

En la siguiente figura, se observa los pasos anteriormente explicados:



Figura A.47 Método de apertura de Procesamiento Estadístico

4.1.3 Crear un nuevo proyecto

Al seleccionar la opción “Archivo” y a continuación, la acción “Nuevo”, queremos indicar que deseamos comenzar un procesamiento estadístico.

De esta manera es como se comienza con un nuevo procesamiento estadístico donde podemos introducir las consultas que deseemos y generar sus correspondientes gráficas, que además, se visualizarán en la pantalla.

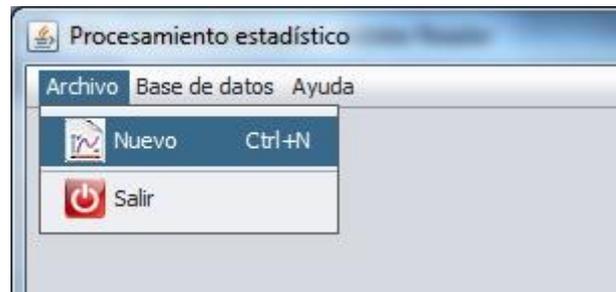


Figura A.48 Nuevo proyecto en Procesamiento Estadístico

Inmediatamente, aparece una ventana donde se nos pregunta cuántas gráficas deseamos visualizar. En realidad se refiere a cuántos procesamientos estadísticos vamos a realizar a la vez. La aplicación nos da la opción de elegir entre 1 o 4 gráficas. Si elegimos 4 gráficas, tenemos la posibilidad de realizar 1, 2, 3 o 4 procesamientos a la vez, puesto que al terminar la introducción de los datos de un análisis podemos comenzar a generar las gráficas o empezar a introducir los datos del siguiente análisis.

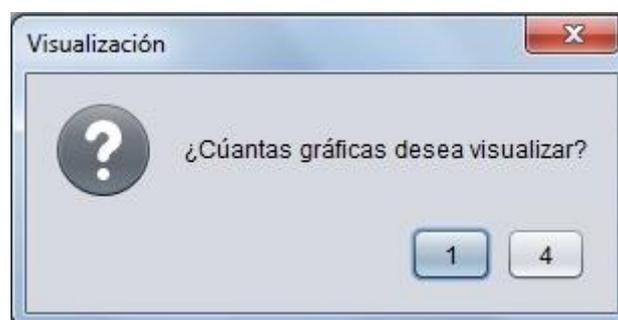


Figura A.49 Cuadro de diálogo número de gráficas

Dependiendo del número seleccionado, la aplicación nos redirige a su correspondiente vista:

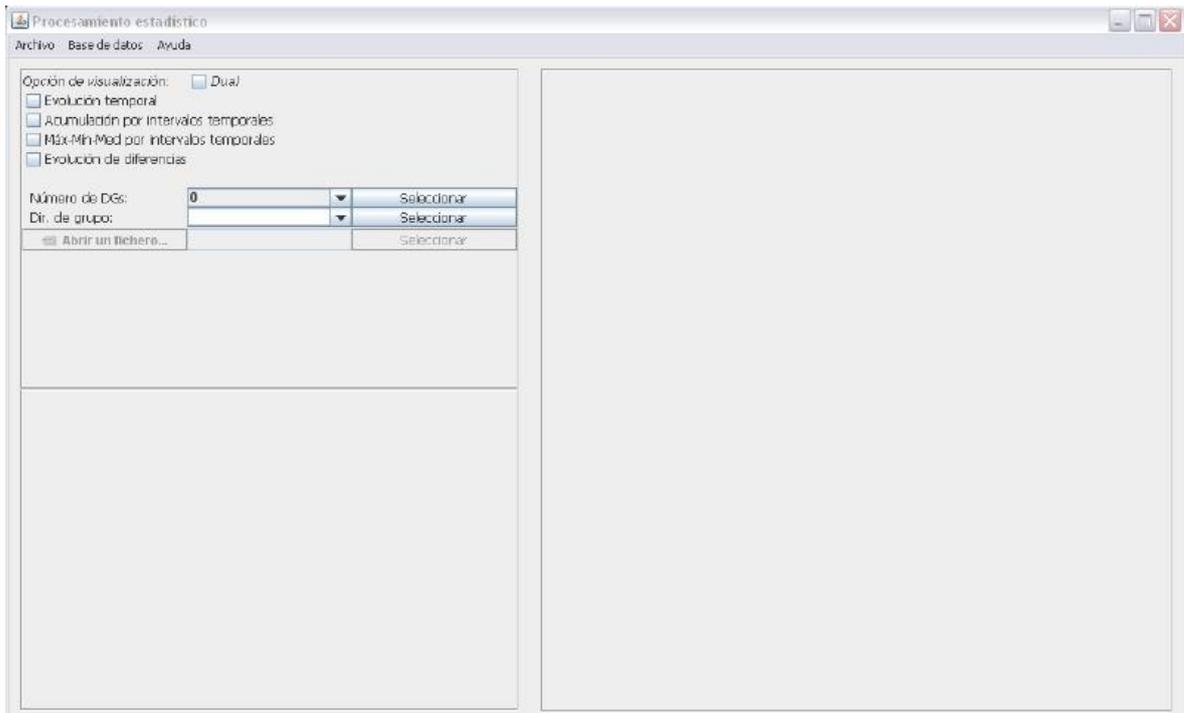


Figura A.50 Vista correspondiente para una gráfica

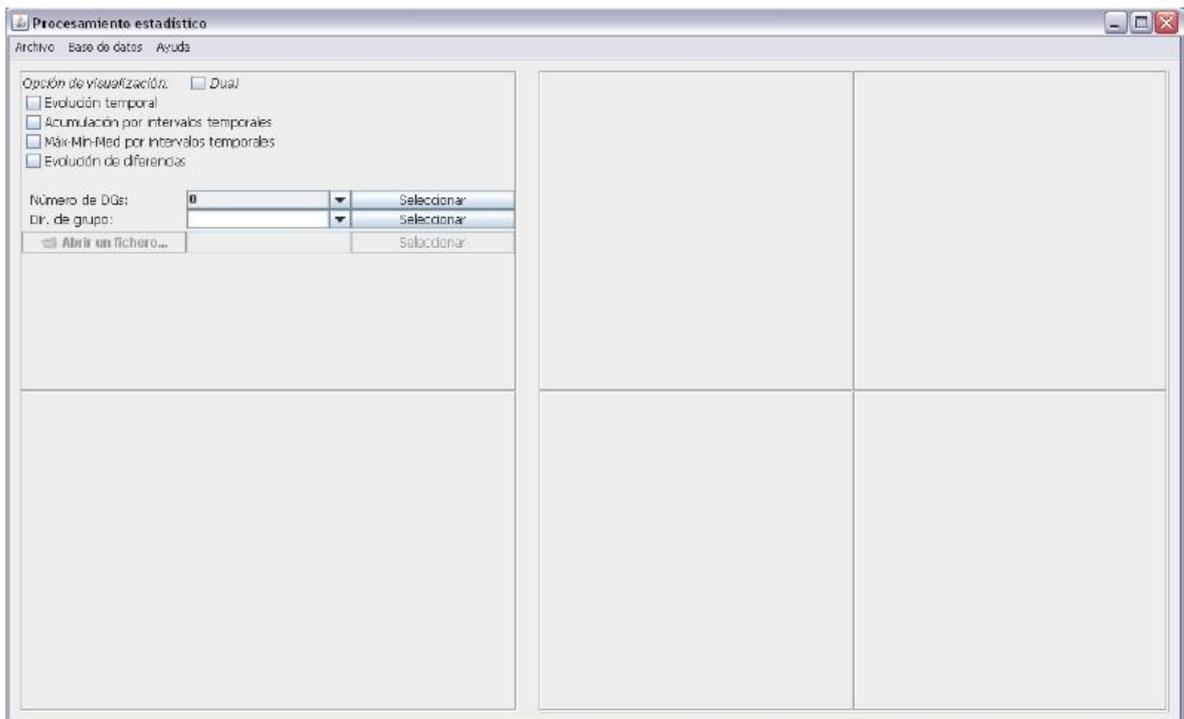


Figura A.51 Vista correspondiente para cuatro gráficas

4.1.4 Selección de tipo de análisis

Una vez estemos listos, en la parte superior izquierda de la ventana resultante del paso anterior, tenemos los tipos de análisis que nos ofrece el módulo de procesamiento estadístico.

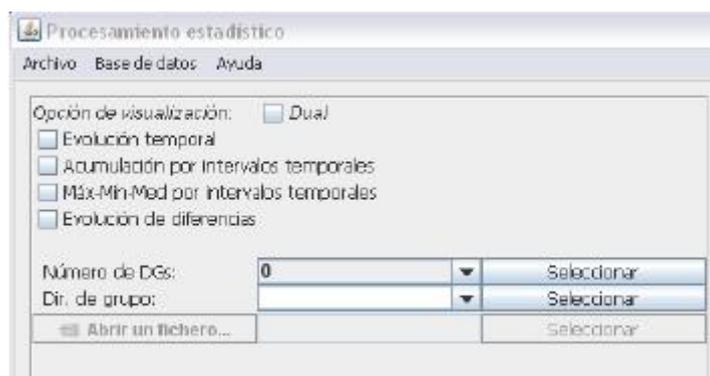


Figura A.52 Opciones de tipo de análisis

Lo primero que podemos hacer es seleccionar el tipo de análisis a realizar o la opción de visualización. La única opción de visualización es “Dual” que significa si queremos que la gráfica resultante tenga doble eje. Si seleccionamos esta opción, la aplicación directamente nos limita a 2 el número de direcciones de grupo (DGs) que podemos seleccionar. Sino se selecciona, en la gráfica sólo tendrá un eje.

Respecto a los tipos de análisis, tenemos 4 posibles. Recordar que si previamente hemos seleccionado la opción de visualización “Dual” y ahora cambiamos el tipo de análisis, la aplicación deselecciona la opción “Dual”, por lo que si la queremos se debe seleccionar de nuevo.

Una vez seleccionado el tipo de análisis y la opción de visualización (si es el caso), pasamos a seleccionar las direcciones de grupo o ficheros que queramos procesar. Antes debemos elegir el número de direcciones de grupo si la aplicación lo permite, puesto que según el tipo de análisis o la opción “Dual” se limita.

Si seleccionamos direcciones de grupo, debemos saber que estaremos extrayendo datos de dichas dirección de grupo directamente de la base de datos. En cambio, si seleccionamos ficheros, estaremos reutilizando los datos resultantes de procesamientos

previos. Sólo podremos utilizar ficheros de procesamientos generados anteriormente por este software, para mantener el formato y estructura que se requiere de los datos que se pueden procesar en este módulo.

A medida que vayamos introduciendo los datos, se nos muestran cuántos datos llevamos seleccionados, cuáles, y cuántos debemos introducir en total.

Una vez hayamos introducido todos los ficheros y direcciones de grupo que especificamos mediante el número de DGs, la interfaz se modifica mostrando nuevas opciones.

Opción de visualización: Dual

Evolución temporal

Acumulación por intervalos temporales

Máx-Mín-Med por intervalos temporales

Evolución de diferencias

Número de DGs: 2 Seleccionar

Dir. de grupo: 2/5/13 1.3.5 Seleccionar

Abrir un fichero... Seleccionar

Seleccionadas (2 de 2): Desplegar para ver

Fechas: Intervalo Completo

Figura A.53 Campos de tipo de análisis

Debemos elegir entre “Intervalo” o “Completo”.

“Intervalo” se refiere a indicar desde cuando y hasta cuando queremos procesar los datos.

“Completo” indicaría procesar todos los datos disponibles en la base de datos o en los ficheros, según sea el caso.

Si elegimos “Intervalo”, se debe especificar fecha inicial y fecha final (y rango, sólo

para el tipo de análisis de acumulación por intervalos temporales y máx.-mín.-media por intervalos temporales).

Opción de visualización: Dual

Evolución temporal

Acumulación por intervalos temporales

Máx-Min-Med por intervalos temporales

Evolución de diferencias

Número de DGs: 2 Seleccionar

Dir. de grupo: 2/5/13 1.3.5 Seleccionar

Seleccionar

Seleccionadas (2 de 2): Desplegar para ver

Fechas: Intervalo Completo

Fecha inicial: 2010-03-01 <aaaa-mm-dd> ó

Fecha final: 2010-03-12 <aaaa-mm-dd hh:mm>

Opción de visualización: Dual

Evolución temporal

Acumulación por intervalos temporales

Máx-Min-Med por intervalos temporales

Evolución de diferencias

Número de DGs: 1 Seleccionar

Dir. de grupo: 2/4/31 1.1.7 Seleccionar

Seleccionar

Seleccionadas (1 de 1): Desplegar para ver

Fechas: Intervalo Completo

Fecha inicial: 2010-03-01 <aaaa-mm-dd> ó

Fecha final: 2010-03-12 <aaaa-mm-dd hh:mm>

Rango: 12 en horas

Figura A.54 Opción Intervalo

Si elegimos “Completo”, se debe especificar nada o rango sólo para el tipo de análisis de acumulación por intervalos temporales y máx.-mín.-media por intervalos temporales.

Opción de visualización:		<input checked="" type="checkbox"/> Dual
<input checked="" type="checkbox"/> Evolución temporal		
<input type="checkbox"/> Acumulación por intervalos temporales		
<input type="checkbox"/> Máx-Min-Med por intervalos temporales		
<input type="checkbox"/> Evolución de diferencias		
Número de DGs:	2	Seleccionar
Dir. de grupo:	2/5/13 1.35	Seleccionar
Abrir un fichero...		Seleccionar
Seleccionadas (2 de 2):	Desplegar para ver	
Fechas:	<input type="checkbox"/> Intervalo	<input checked="" type="checkbox"/> Completo
		AÑADIR DATOS

Opción de visualización:		<input type="checkbox"/> Dual
<input type="checkbox"/> Evolución temporal		
<input checked="" type="checkbox"/> Acumulación por intervalos temporales		
<input type="checkbox"/> Máx-Min-Med por intervalos temporales		
<input type="checkbox"/> Evolución de diferencias		
Número de DGs:	1	Seleccionar
Dir. de grupo:	2/4/31 1.17	Seleccionar
Abrir un fichero...		Seleccionar
Seleccionadas (1 de 1):	Desplegar para ver	
Fechas:	<input type="checkbox"/> Intervalo	<input checked="" type="checkbox"/> Completo
Rango:	12	en horas
		AÑADIR DATOS

Figura A.55 Opción Completo

4.1.5 Añadir datos

Una vez hayamos completado el formulario del procesamiento que deseamos realizar, debemos confirmar los datos pulsando sobre el botón “Añadir datos”.

A continuación, se mostrará un resumen de los datos introducidos en la parte inferior izquierda de la ventana, y se resetea el formulario por si se quiere introducir un nuevo análisis, si se escogió la opción de visualizar más de una gráfica.

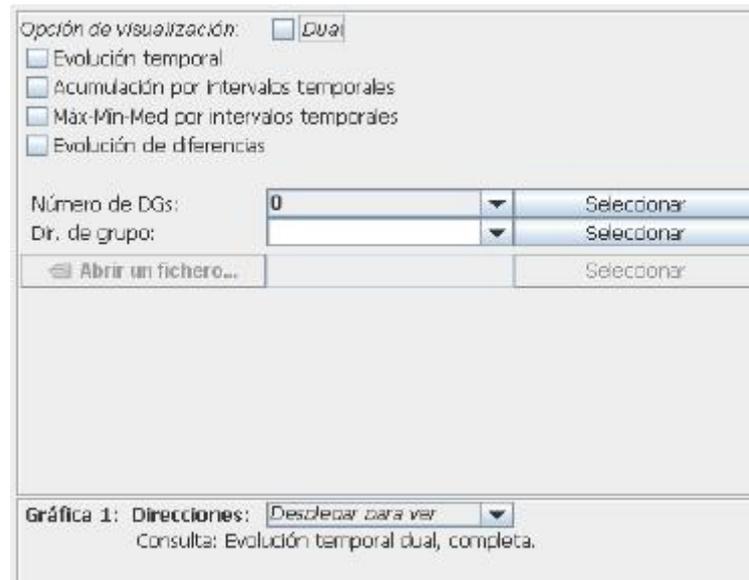


Figura A.56 Añadir datos

4.1.6 Generar gráficas

Al haber introducido los datos de, como mínimo, un procesamiento, al final de los datos confirmados (parte inferior izquierda de la ventana) se muestra el botón de “Generar gráficas”, con el que se ejecutan en paralelo todos los procesos estadísticos insertados. Al terminar todos los procesos, se colocan las gráficas resultantes en la parte derecha.

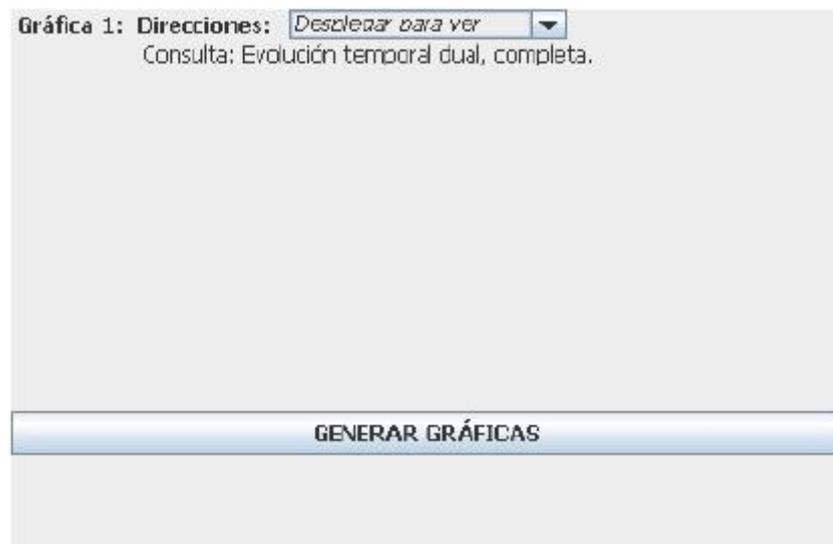


Figura A.57 Generar Gráficas

4.1.7 Salir

Al seleccionar la opción “Archivo” y a continuación, la acción “Salir”, queremos indicar que deseamos salir del módulo de procesamiento estadístico. Esta acción se puede realizar en cualquier momento, incluso en medio de la generación de las gráficas.



Figura A.58 Salir

4.1.8 Actualizar

El usuario tiene la opción de actualizar la base de datos “manualmente”. Para ello debe seleccionar la opción “Base de datos” y a continuación la acción “Actualizar”. De esta manera, el usuario actualiza la base de datos con todos los ficheros almacenados en el directorio Logs.

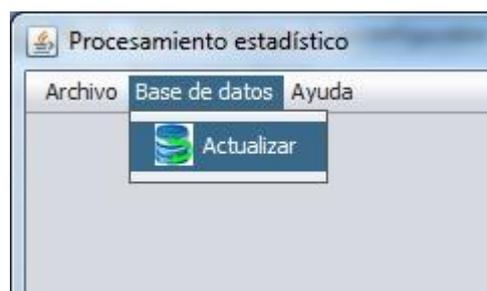


Figura A.59 Actualizar base de datos

Al tratarse de una operación de riesgo, puesto que no se controla si ya han sido introducidos esos ficheros en la base de datos, la aplicación muestra una ventana para que se confirme dicha acción.

4.1.9 Acerca de...

El usuario puede comprobar la versión del módulo de procesamiento estadístico así como quién lo creó. Para ello debe seleccionar la opción “Ayuda” y a continuación la acción “Acerca de...”.

5. Menú Contadores

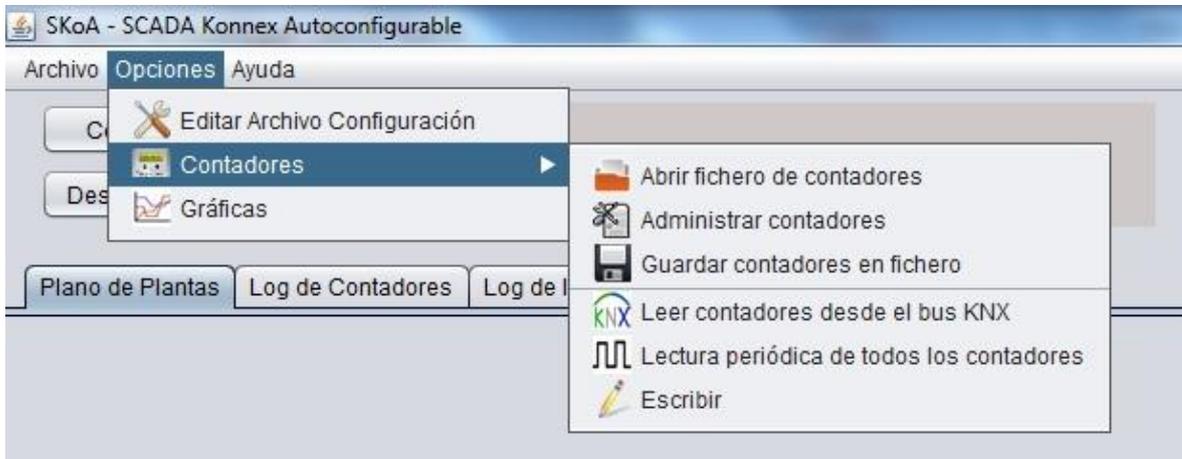


Figura A.60 Menú Contadores interfaz principal

En el menú de la interfaz principal de control y monitorización, Opciones -> Contadores, nos podremos encontrar las siguientes posibilidades:

5.1 Abrir fichero base de datos

Cuando deseamos cargar en la aplicación un fichero de configuración previamente establecido, tendremos que cargar a su vez el fichero de contadores asociado. El cual cargará en la aplicación de una manera rápida y eficaz todos los contadores especificados por el fichero. La estructura del fichero se verá más adelante.

5.2 Administrar contadores

A través de esta opción se pueden añadir contadores, eliminarlos, o editarlos. A medida que se van añadiendo contadores, éstos aparecerán en una lista en la ventana de la izquierda, además de diversa información asociada a ellos. Para añadir un contador, hay que indicar su tipo, subtipo y dirección de grupo. Además, se puede insertar un nombre para identificar al contador asociado. El nombre no puede superar los 6 caracteres de longitud (limitación de la librería Calimero).

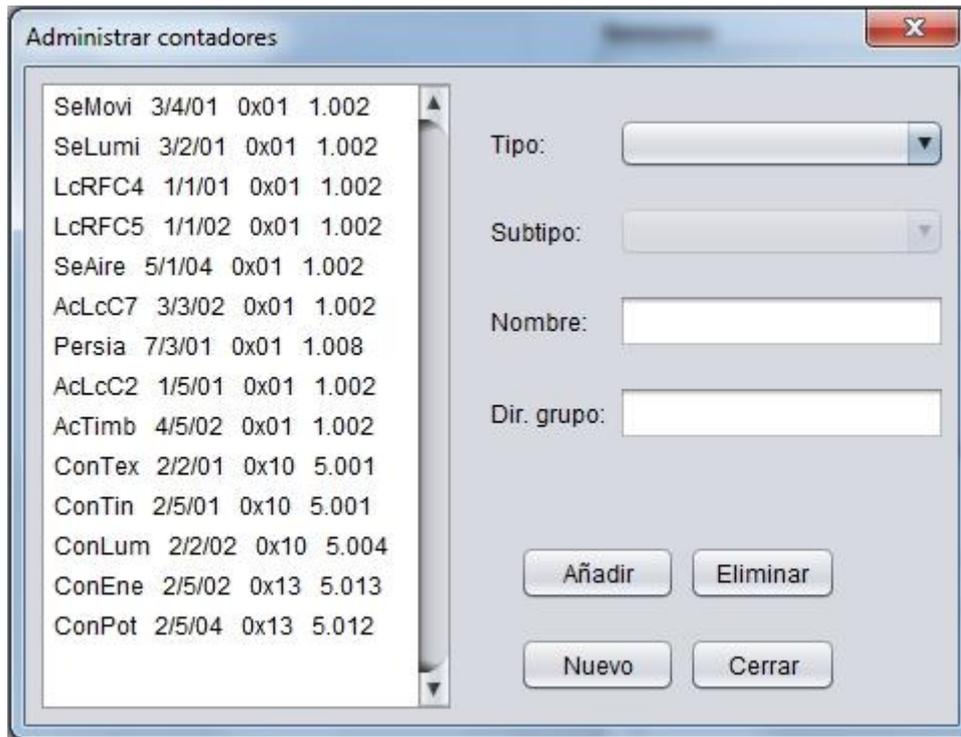


Figura A.61 Ventana Administrar Contadores

Para editar un contador, basta con seleccionarlo en la lista, modificar los parámetros que se desean, y pulsar en el botón “Modificar”.

5.3 Guardar Contadores en Fichero

Una vez que se han agregado una serie de contadores, ya sea a través de la opción “Opciones → Contadores → Administrar Contadores” o bien mediante la opción de añadir actuadores (ver punto 3.2.5 “Añadir actuadores/sensores” del apartado 3.2 Fichero de Configuración XML) es posible guardar un listado de los mismos en un fichero .dat.

De esta forma, los contadores permanecerán en el fichero que se comportará como una base de datos.

5.4 Leer Contadores desde el Bus KNX

Una vez que se han añadido contadores (al menos uno) es posible realizar una lectura de los mismos a través del bus KNX. Para ello, un requisito indispensable es haber establecido previamente una conexión con el bus. De lo contrario, y a pesar de que se intente realizar una lectura, no aparecerá la ventana de lectura de contadores (no aparecerá ninguna ventana de hecho).

Al seleccionar esta opción se mostrará una ventana en la que se podrá seleccionar un contador de entre todos los contadores insertados a través de una lista desplegable.

También aparecerá información acerca del contador, mostrando su tipo y subtipo. Para realizar una lectura, bastará con pulsar en el botón “Leer”. Hay un botón que sirve para actualizar los contadores insertados, en caso de que no se muestren todos en la lista desplegable en un primer momento. Será suficiente con pulsar en el botón “Actualizar contadores” una vez para, posteriormente, seleccionar el contador apropiado en la lista desplegable.

Inmediatamente tras pulsar en el botón “Leer”, en la pestaña “Log de contadores” situada en la interfaz principal aparecerá información acerca del contador. El formato de la información para los contadores es siempre el mismo:

5.5 Lectura Periódica de todos los Contadores

La aplicación se encarga de realizar una lectura periódica de todos los contadores insertados en la base de datos. También es necesario haber establecido previamente una conexión con el bus EIB.

Se mostrará una ventana en la que se podrá apreciar un listado con todos los contadores insertados en la base de datos (también se mostrará su dirección de grupo, tipo y subtipo) y que se tendrán en cuenta a la hora de realizar la lectura periódica. El usuario ha de rellenar tres campos:

- **Intervalo de muestreo corto:** Se mide en minutos, e indica cada cuanto

tiempo se realizarán lecturas de los contadores.

- **Intervalo de muestre largo:** Se mide en horas, e indica cada cuanto tiempo se realizará una copia de seguridad del fichero log de la aplicación y del fichero con las medidas obtenidas por los contadores.
- **Nombre del fichero de medidas:** En él se guardarán todos los valores proporcionados por los contadores, obtenidos éstos a su vez por el programa a través de la lectura periódica.

Una vez rellenados estos tres parámetros, el usuario ha de pulsar en el botón “Activar” para que se active el proceso de lectura periódica de los contadores.

Puede detener dicho proceso en cualquier momento si pulsa en el botón “Desactivar”. Al hacerlo, también se realizará una copia de seguridad de los ficheros comentados anteriormente (fichero log y fichero de medidas). Además de almacenarse la información de los contadores en el fichero especificado, se puede consultar en todo momento dicha información en la pestaña “Log de los contadores”



Figura A.62 Logs de Contadores

El log muestra el valor de cada uno de ellos por separado y finalmente muestra en una línea todos los valores obtenidos. Esta línea será la que vaya almacenando en el fichero log “medidas_SKoA.log” (por defecto con ese nombre, aunque se puede cambiar).

5.6 Escribir

A través de esta opción es posible mandar una señal de escritura mediante el bus KNX. Así pues, podremos controlar diferentes dispositivos y realizar diversas acciones como encender una luz, bajar una persiana, cerrar una electroválvula, etc. Es otra forma de poder mandar una señal a los actuadores sin necesidad de realizarlo mediante la interfaz gráfica de la vivienda.

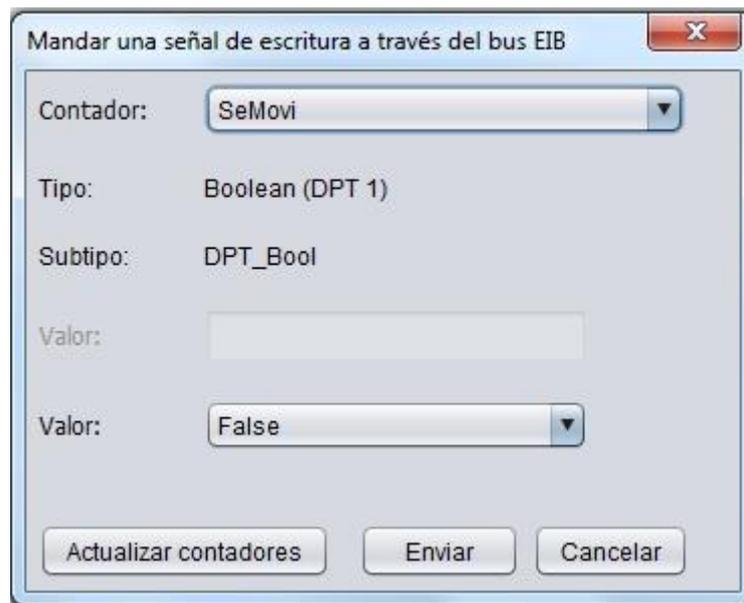


Figura A.63 Vista ventana Escribir

6. Control y Monitorización de los Dispositivos Domóticos

Una vez que se ha abierto un proyecto, en la pestaña “Planos de planta” veremos de manera gráfica el estado de las diferentes plantas y estancias del edificio domótico. Las plantas se distribuirán a su vez en pestañas verticales, mientras que las estancias lo harán a través de pestañas horizontales. Para interactuar con una estancia determinada, simplemente tendremos que seleccionar la pestaña correspondiente a la planta en la que se encuentra y posteriormente la pestaña que se corresponde con la estancia que deseamos controlar.

Podremos actuar sobre los actuadores (subir persianas, cerrar electroválvulas, encender luces regulables, etc.) y también consultar el estado de los sensores (puertas, termostatos, contadores de energía, etc.)



Figura A.64 Módulo de Control y Monitorización

En la imagen anterior, se observa el estado de la estancia “Dormitorio” correspondiente a la planta “Planta 1”. Podemos observar varias cosas en esta imagen:

- La luz regulable se encuentra al 75%..
- La luz conmutable se encuentra encendida.
- La persiana se encuentra abierta.

Si una estancia no dispone de algunos elementos domóticos, la interfaz correspondiente a su manejo/monitorización se desactiva, de manera que el usuario sólo puede actuar/consultar aquellos elementos de los que dispone la estancia. Por ejemplo, en la figura anterior, está desactivado el sensor de temperatura.

7. Iconos Utilizados

El hecho de representar el estado de algunos dispositivos a través de iconos es simplemente visual, ya que también se muestra por escrito cómo se encuentran éstos en todo momento. A continuación se muestran los diferentes iconos y su significado:

➤ Luces regulables

Estado	Desconocido	Apagado	Al 25 %	Al 50%	Al 75%	Al 100%
Icono		 Negro	 Azul	 Amarillo	 Naranja	 Rojo

Tabla 3.2 Iconos Luces Regulables

➤ Luces conmutables

Estado	Desconocido	Apagado	Encendido
Icono			

Tabla 3.4 Iconos Luces Conmutables

➤ **Persianas**

Estado	Desconocido	Cerrada	Subida
Icono			

Tabla 3.6 Iconos Persianas

➤ **Electroválvulas**

Estado	Desconocido	Cerrada	Abierta
Icono			

Tabla 3.8 Iconos Electroválvula

➤ **Puertas**

Estado	Desconocido	Cerrada	Abierta
Icono			

Tabla 3.10 Iconos Puerta

➤ **Detector de movimiento**

Estado	Desconocido	Presencia no detectada	Presencia detectada
Icono			

Tabla 3.12 Iconos Detector Movimiento

➤ **Sensor de temperatura**

Estado	Presente en la estancia
Icono	

Tabla 3.14 Icono Sensor Tempemperatura

➤ **Sensor combinado**

Estado	Presente en la estancia
Icono	

Tabla 3.15 Icono Sensor Combinado

➤ **Sensor de inundación**

Estado	Presente en la estancia	Fuga detectada	Fuga no detectada
Icono			

Tabla 3.17 Icono Sensor Inundación

➤ **Contadores**

Estado	Presente en la estancia
Icono	

Tabla 3.18 Icono Contadores

➤ **Iconos de menú**

Icono	Nomenclatura
	Abrir fichero contadores
	Abrir proyecto
	Acerca de
	Administrar contadores
	Ver configuración actual
	Conectar Bus EIB
	Editar archivo de configuración
	Cerrar Proyecto
	Desconectar Bus EIB
	Escribir

	Gráficas
	Guardar Proyecto
	Nuevo Proyecto
	Salir

Tabla 3.19 Iconos de Menú

8. Menú Ayuda

7.1 Ver Configuración Actual

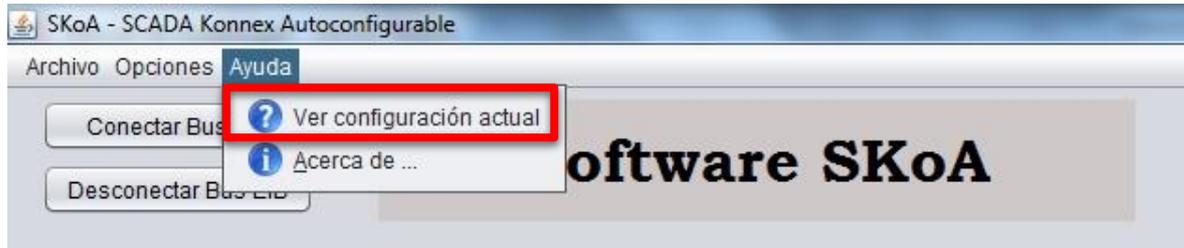


Figura A.65 Menú Ayuda

Esta opción muestra la configuración del fichero “MCA.cnf”, que contiene algunos parámetros importantes para el manejo de la aplicación.

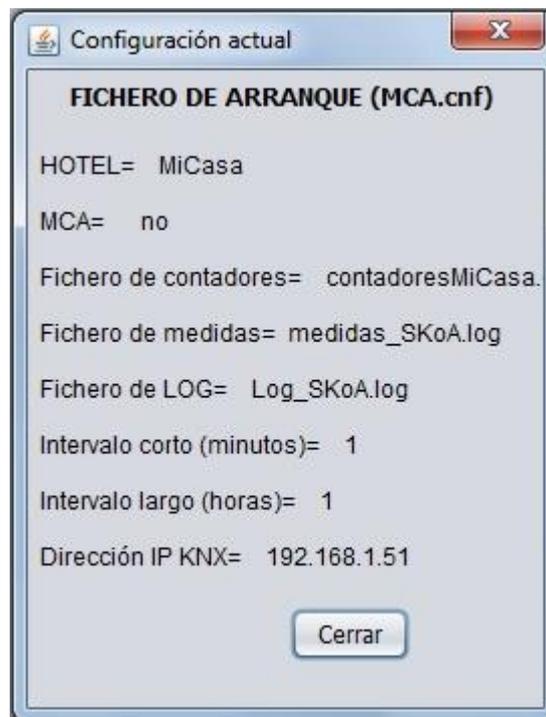


Figura A.66 Vista Configuración Actual

El fichero MCA.cnf almacena determinados parámetros que utilizará el programa en caso de querer automatizar el proceso de lectura de contadores sin necesidad de la intervención humana. Estos parámetros son:

- **Hotel:** Carece de importancia, describe al hotel.
- **MCA:** Si tiene el valor “no” significa que el uso de la aplicación será manual, mientras que si tiene el valor “si” el usuario no ha de efectuar nada, tan solo ejecutar la aplicación. El programa será el encargado de iniciar los intervalos de lectura y backup, conectarse al bus, cargar los contadores, configurar los envíos periódicos y obtener las medidas de dichos contadores, todo ello de forma automática.
- **Fcontadores:** Es el fichero en el que se almacenarán los contadores.
- **Fmedidas:** Es el fichero en el que se almacenarán las medidas proporcionadas por los contadores.
- **Flog:** Es el fichero en el que se almacenará la actividad de la aplicación.
- **Intervalo:** Medido en minutos, indica cada cuanto se realizará una lectura de los contadores.
- **Intervalo_largo:** Medido en horas, indica cada cuanto se realizará una copia de seguridad de los archivos log (medidas de los contadores y actividad de la aplicación).
- **IPknx:** La dirección IP del bus a la cual hay que conectarse.

7.2 Acerca de

A través del menú *Ayuda* → *Acerca de*, podemos comprobar la versión actual de la aplicación SKoA así como sus autores:

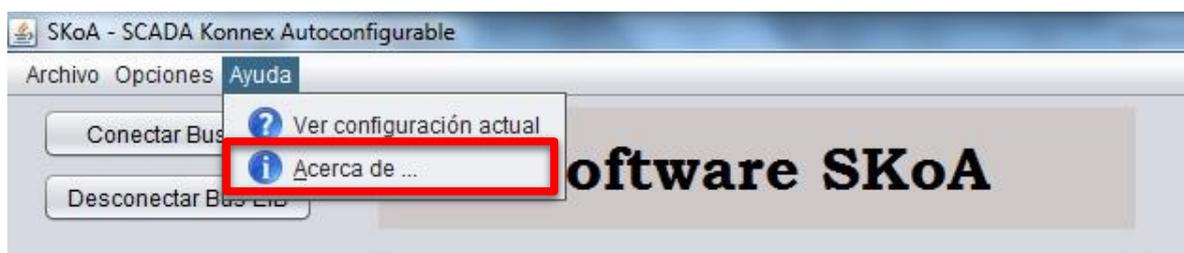


Figura A.67 Acerca de



Figura A.68 Vista "Acerca de"

9. Glosario

- **Completo:** Opción de Fechas que permite al usuario elegir todos los datos disponibles a lo largo del tiempo.
- **DGs:** Sigla en plural formada por las iniciales de “Direcciones de Grupo”.
- **Dual:** Opción para visualizar dos ejes en las gráficas, o lo que es lo mismo, un eje dual.
- **Fichero Contadores.dat:** Posee un listado de todos los contadores
- **Fichero de Configuración XML:** Contiene toda la información necesaria para que la aplicación pueda mostrar en la pestaña “Planos de planta” la configuración del edificio domótico.
- **Fichero info.txt:** Guarda información relativa a los dispositivos domóticos ubicados en las diferentes estancias del hotel. Esta información se guarda al seleccionar la opción “Guardar Proyecto” del menú principal, y es importante para poder monitorizar los diferentes sensores. La información se almacena en dos vectores (líneas 1 y 3) mientras que en las líneas 2 y 4 se almacenan los índices de los vectores por si se añade más información posteriormente.
- **Fichero Log_SKoA.log:** Almacena la información relativa a toda actividad producida en la aplicación. Tiene el mismo comportamiento que la pestaña “Log de la aplicación”, pero en este caso la información se almacena en un fichero.
- **Fichero MCA.cnf:** Almacena determinados parámetros que utilizará el programa en caso de querer automatizar el proceso de lectura de contadores sin necesidad de la intervención humana.
- **Fichero Medidas_SKoA.log:** Almacena las diferentes medidas de todos los contadores insertados en la base de datos al realizarse una lectura periódica. insertados en la base de datos.
- **Intervalo:** Opción de Fechas que permite al usuario establecer la fecha inicial a partir de la cual se desean los datos y la fecha final hasta donde se desean los datos.
- **Rango:** Número de horas que componen cada uno de los intervalos temporales contenidos en los datos a analizar. Únicamente se utiliza para las consultas de

acumulación por intervalos temporales y de máximo mínimo y media por intervalos temporales.

Anexo II:

Código Fuente PILEV APP

Index.html

```
<!doctype html>
<html lang="en" ng-app="app">
<head>
  <meta charset="utf-8">
  <meta name="apple-mobile-web-app-capable" content="yes">
  <meta name="mobile-web-app-capable" content="yes">
  <allow-navigation href="http://*/*" />
  <title>Pilve View</title>

  <link rel="stylesheet" href="lib/onsen/css/onsenui.css">
  <link rel="stylesheet" href="styles/app.css"/>
  <link rel="stylesheet" href="styles/onsen-css-components.css">

  <script src="lib/onsen/js/angular/angular.js"></script>
  <script src="lib/onsen/js/onsenui.js"></script>
  <script src="lib/angular-animate/angular-animate.js"></script>
  <script src="cordova.js"></script>
  <script src="js/app.js"></script>
</head>

<body ng-controller="AppController">

  <!-- Loading -->
  <ons-page ng-show="isSplashSHow">
    <ons-toolbar fixed-style>
      <div class="center" class="title-app">Pilve Visor</div>
    </ons-toolbar>

    <div class="spinner">
      <div class="dot1"></div>
      <div class="dot2"></div>
    </div>
  </ons-page>

  <!-- error coccnection -->
  <ons-page class="error" ng-show="error">
```

```

<ons-toolbar fixed-style>
  <div class="left">
    <div class="center" class="title-app">Pilve Visor</div>
  </div>
  <div class="center">Pilve Visor</div>
</ons-toolbar>

<ons-row>
  <ons-col>
    <ons-icon icon="fa-exclamation-triangle" size="80px">

    <h1> Problemas de coneccion</h1>
    <p>Lo sentimos no hay coneccion a internet</p>
  </ons-col>
</ons-row>
</ons-page>

<!-- Master - navigator-->
<ons-navigator var="navi" ng-show="!isSplashShow && !error" >
  <ons-page ng-controller="MasterController">

    <ons-toolbar fixed-style>
      <div class="center">Pilve Visor</div>
      <div class="right">
        <ons-button ng-click="showDetailTotal()">
          <span>Totales</span>
          <ons-icon icon="ion-stats-bars">
        </ons-button>
      </div>
    </ons-toolbar>

    <div class="last-update">Ultima actualización:
    {{lastUpdate}}</div>

    <ons-list>
      <ons-list-item class="item-master" ng-repeat="item in items" ng-
      click="showDetail($index)">
        <ons-row>
          <ons-col width="80px">
            <div class="item-img-circle">
              {{item.title}}</span>
            </ons-col>
          </ons-row>
        </ons-list-item>
      </ons-list>
    </ons-page>

```

```

</ons-navigator>

<!-- Details -->
<ons-template id="detail.html" >
  <ons-page ng-controller="DetailController">

    <ons-toolbar>
      <div class="left"><ons-back-button>Atras</ons-back-button></div>
      <div class="center"> Detalles </div>
    </ons-toolbar>

    <div class= "item-img-circle">
      
    </div>

    <h1 class="item-detail-title">{{item.title2}}</h1>

    <ons-carousel swipeable style="height: 60%; width: 100%;" initial-
index="0" auto-scroll var="mCarousel">

      <ons-carousel-item class="today">

        <ons-list-item class="item">
          <ons-row>
            <ons-col>
              <span class="item-detail-label">Energia:</span>
              <span
class="item-detail-
value">{{item.today.energy}}</span>
            </ons-col>
          </ons-row>
        </ons-list-item>

        <ons-list-item class="item">
          <ons-row>
            <ons-col>
              <span class="item-detail-label">Co2:</span>
              <span
class="item-detail-
value">{{item.today.co2}}</span>
            </ons-col>
          </ons-row>
        </ons-list-item>
        <ons-list-item class="item">
          <ons-row>
            <ons-col>
              <span class="item-detail-label">Arboles:</span>
              <span
class="item-detail-
value">{{item.today.trees}}</span>
            </ons-col>
          </ons-row>

```

```

    </ons-list-item>
    <div class="item-detail-cover">
      <span class="item-detail-cover-label">Ayer</span>
      <ons-icon icon="fa-arrow-right" size="40px">
    </div>
  </ons-carousel-item>

  <ons-carousel-item class="yesterday">
    <ons-list-item class="item">
      <ons-row>
        <ons-col>
          <span class="item-detail-label">Energía:</span>
          <span class="item-detail-
value">{{item.yesterday.energy}}</span>
        </ons-col>
      </ons-row>
    </ons-list-item>
    <ons-list-item class="item">
      <ons-row>
        <ons-col>
          <span class="item-detail-label">Co2:</span>
          <span class="item-detail-
value">{{item.yesterday.co2}}</span>
        </ons-col>
      </ons-row>
    </ons-list-item>
    <ons-list-item class="item">
      <ons-row>
        <ons-col>
          <span class="item-detail-label">Arboles:</span>
          <span class="item-detail-
value">{{item.yesterday.trees}}</span>
        </ons-col>
      </ons-row>
    </ons-list-item>
    <div class="item-detail-cover">
      <ons-icon icon="fa-arrow-left" size="40px">
      <span class="item-detail-cover-label">Hoy</span>
    </div>
  </ons-carousel-item>

</ons-carousel>

</ons-page>
</ons-template>

</body>
</html>

```


App.js

```
(function(){
  'use strict';

  var md5 = function(s){function L(k,d){return(k<<d)|(k>>>(32-
d));}function K(G,k){var
I,d,F,H,x;F=(G&2147483648);H=(k&2147483648);I=(G&1073741824);d=(k&107374
1824);x=(G&1073741823)+(k&1073741823);if(I&d){return(x^2147483648^F^H);}
if(I|d){if(x&1073741824){return(x^3221225472^F^H);}else{return(x^1073741
824^F^H);}}else{return(x^F^H);}}function
r(d,F,k){return(d&F)|((~d)&k);}function
q(d,F,k){return(d&k)|(F&(~k));}function p(d,F,k){return(d^F^k);}function
n(d,F,k){return(F^(d|(~k)));}function
u(G,F,aa,Z,k,H,I){G=K(G,K(K(r(F,aa,Z),k),I));return
K(L(G,H),F);}function
f(G,F,aa,Z,k,H,I){G=K(G,K(K(q(F,aa,Z),k),I));return
K(L(G,H),F);}function
D(G,F,aa,Z,k,H,I){G=K(G,K(K(p(F,aa,Z),k),I));return
K(L(G,H),F);}function
t(G,F,aa,Z,k,H,I){G=K(G,K(K(n(F,aa,Z),k),I));return
K(L(G,H),F);}function e(G){var Z;var F=G.length;var x=F+8;var k=(x-
(x%64))/64;var I=(k+1)*16;var aa=Array(I-1);var d=0;var
H=0;while(H<F){Z=(H-
(H%4))/4;d=(H%4)*8;aa[Z]=(aa[Z]|(G.charCodeAt(H)<<d));H++;}Z=(H-
(H%4))/4;d=(H%4)*8;aa[Z]=aa[Z]|(128<<d);aa[I-2]=F<<3;aa[I-
1]=F>>>29;return aa;}function B(x){var
k="",F="",G,d;for(d=0;d<=3;d++){G=(x>>>(d*8))&255;F=""+G.toString(16);k
=k+F.substr(F.length-2,2);}return k;}function
J(k){k=k.replace(/rn/g,"n");var d="";for(var F=0;F<k.length;F++){var
x=k.charCodeAt(F);if(x<128){d+=String.fromCharCode(x);}else{if((x>127)&&
(x<2048)){d+=String.fromCharCode((x>>6)|192);d+=String.fromCharCode((x&6
3)|128);}else{d+=String.fromCharCode((x>>12)|224);d+=String.fromCharCode
(((x>>6)&63)|128);d+=String.fromCharCode((x&63)|128);}}}return d;}var
C=Array();var P,h,E,v,g,Y,X,W,V;var S=7,Q=12,N=17,M=22;var
A=5,z=9,y=14,w=20;var o=4,m=11,l=16,j=23;var
U=6,T=10,R=15,O=21;s=J(s);C=e(s);Y=1732584193;X=4023233417;W=2562383102;
V=271733878;for(P=0;P<C.length;P+=16){h=Y;E=X;v=W;g=V;Y=u(Y,X,W,V,C[P+0]
,S,3614090360);V=u(V,Y,X,W,C[P+1],Q,3905402710);W=u(W,V,Y,X,C[P+2],N,606
105819);X=u(X,W,V,Y,C[P+3],M,3250441966);Y=u(Y,X,W,V,C[P+4],S,4118548399
);V=u(V,Y,X,W,C[P+5],Q,1200080426);W=u(W,V,Y,X,C[P+6],N,2821735955);X=u(
X,W,V,Y,C[P+7],M,4249261313);Y=u(Y,X,W,V,C[P+8],S,1770035416);V=u(V,Y,X,
W,C[P+9],Q,2336552879);W=u(W,V,Y,X,C[P+10],N,4294925233);X=u(X,W,V,Y,C[P
+11],M,2304563134);Y=u(Y,X,W,V,C[P+12],S,1804603682);V=u(V,Y,X,W,C[P+13]
,Q,4254626195);W=u(W,V,Y,X,C[P+14],N,2792965006);X=u(X,W,V,Y,C[P+15],M,1
236535329);Y=f(Y,X,W,V,C[P+1],A,4129170786);V=f(V,Y,X,W,C[P+6],z,3225465
664);W=f(W,V,Y,X,C[P+11],y,643717713);X=f(X,W,V,Y,C[P+0],w,3921069994);Y
=f(Y,X,W,V,C[P+5],A,3593408605);V=f(V,Y,X,W,C[P+10],z,38016083);W=f(W,V,
Y,X,C[P+15],y,3634488961);X=f(X,W,V,Y,C[P+4],w,3889429448);Y=f(Y,X,W,V,C
[P+9],A,568446438);V=f(V,Y,X,W,C[P+14],z,3275163606);W=f(W,V,Y,X,C[P+3],
y,4107603335);X=f(X,W,V,Y,C[P+8],w,1163531501);Y=f(Y,X,W,V,C[P+13],A,285
0285829);V=f(V,Y,X,W,C[P+2],z,4243563512);W=f(W,V,Y,X,C[P+7],y,173532847
```

```

3);X=f(X,W,V,Y,C[P+12],w,2368359562);Y=D(Y,X,W,V,C[P+5],o,4294588738);V=
D(V,Y,X,W,C[P+8],m,2272392833);W=D(W,V,Y,X,C[P+11],l,1839030562);X=D(X,W
,V,Y,C[P+14],j,4259657740);Y=D(Y,X,W,V,C[P+1],o,2763975236);V=D(V,Y,X,W,
C[P+4],m,1272893353);W=D(W,V,Y,X,C[P+7],l,4139469664);X=D(X,W,V,Y,C[P+10
],j,3200236656);Y=D(Y,X,W,V,C[P+13],o,681279174);V=D(V,Y,X,W,C[P+0],m,39
36430074);W=D(W,V,Y,X,C[P+3],l,3572445317);X=D(X,W,V,Y,C[P+6],j,76029189
);Y=D(Y,X,W,V,C[P+9],o,3654602809);V=D(V,Y,X,W,C[P+12],m,3873151461);W=D
(W,V,Y,X,C[P+15],l,530742520);X=D(X,W,V,Y,C[P+2],j,3299628645);Y=t(Y,X,W
,V,C[P+0],U,4096336452);V=t(V,Y,X,W,C[P+7],T,1126891415);W=t(W,V,Y,X,C[P
+14],R,2878612391);X=t(X,W,V,Y,C[P+5],O,4237533241);Y=t(Y,X,W,V,C[P+12],
U,1700485571);V=t(V,Y,X,W,C[P+3],T,2399980690);W=t(W,V,Y,X,C[P+10],R,429
3915773);X=t(X,W,V,Y,C[P+1],O,2240044497);Y=t(Y,X,W,V,C[P+8],U,187331335
9);V=t(V,Y,X,W,C[P+15],T,4264355552);W=t(W,V,Y,X,C[P+6],R,2734768916);X=
t(X,W,V,Y,C[P+13],O,1309151649);Y=t(Y,X,W,V,C[P+4],U,4149444226);V=t(V,Y
,X,W,C[P+11],T,3174756917);W=t(W,V,Y,X,C[P+2],R,718787259);X=t(X,W,V,Y,C
[P+9],O,3951481745);Y=K(Y,h);X=K(X,E);W=K(W,v);V=K(V,g);}var
i=B(Y)+B(X)+B(W)+B(V);return i.toLowerCase();}

var extractNumber = function(str){ return
parseInt(str.match(/\d+/)[0])}
var insertString = function (str, index, value) {return str.substr(0,
index) + value + str.substr(index);}

var module = angular.module('app', ['onsen','ngAnimate']);
var onsenComponents;

module.controller('AppController', function($scope, $timeout,$http) {
    $scope.isSplashSHow = true;
    $scope.error = false;
    $scope.lastUpdate = '';

    $scope.hideSlapash = function(){
        $timeout(function() {
            $scope.isSplashSHow = false;
        }, 2000);
    }

    $scope.setLastUpdate = function(lastUpdate){
        $scope.lastUpdate = lastUpdate;
    }

    $scope.setError = function(error){
        $scope.error = error;
        console.log("setError : " + $scope.error)
    }

});

```

```

/*
  Modulo contralos de la vista details
*/

module.controller('DetailController', function($scope,$dataRequeest) {

  $scope.item = $dataRequeest.selectedItem;
  $scope.title = $dataRequeest.selectedItem.title;
});

/*
  Modulo contralos de la vista master
*/

module.controller('MasterController', function($scope, $dataRequeest)
{
  $dataRequeest.get()
  .then(function(data) {

    $scope.items = $dataRequeest.items = data.items;
    $scope.setLastUpdate( data.lastUpdate );

    /*
      Funcion que muestra los detalles de los hoteles
    */
    $scope.showDetail = function(index) {
      var selectedItem = data.items[index];

      $dataRequeest.selectedItem = selectedItem;
      $scope.navi.pushPage('detail.html', {title :
selectedItem.title});
    };

    /*
      Funcion que muestra los detalles t totales
    */
    $scope.showDetailTotal = function() {
      var selectedItemTotal = $dataRequeest.getConsumeTotal();
      selectedItemTotal.title = "Datos totales de HOY del GRUPO
LOPESAN"

      $dataRequeest.selectedItem = selectedItemTotal;
      $scope.navi.pushPage('detail.html', {title :
selectedItemTotal.title});
    };

    $scope.hideSlapash();

  }, function(error) {
    console.log("request error : " + $scope.error)
    $scope.setError(true);
  });
});

```

```

        $scope.mylist = [];
        $scope.hideSlapash();
    });

});

/*
    Modulo que genera los datos
*/

module.factory('$dataRequeest', function($http, $q) {

    /*
        Funcion para obtener la imagne correspondiente al hotl
    */
    function getHotleImg(nameHotel){
        var imgs = {
            'hotel baobab': 'baobab.jpg',
            'hotel faro': 'faro.jpg',
            'hotel continental': 'continental.jpg',
            'hotel interclub1': 'interclub1.jpg',
            'hotel buenaventura': 'buenaventura.jpg'
        }

        return imgs[nameHotel];
    }

    /*
        Funcion para realizar la extracion de los datos
    */
    function scraping(data){

        //obtenemos un array con las lineas del archivo en miniscula
        var lines = data.toLowerCase().split("\n")

        var itemsTpm = []; //valores encontrados durante el scraping
        var positionKeys = []; //para contener la posicion en la que
        estan guardados los hoteles

        //recorremos las lineas
        for(var i = 5; i < lines.length ; i+=5 ){

            //nos aseguramos que la linea no esta vacia
            if(lines[i] === ""){
                continue;
            }

            //obtenemos los datos
            var title = (lines[i]+'');
            title = title.substring(title.indexOf('hotel'));

```

```

        var title2 =
(lines[i]+'').substring((lines[i]+'').indexOf('datos'),(lines[i]+'').ind
exOf('totales'));
        title2 = title2 + "del " + title;

var energy = (lines[i+2]+'').substring('Energia: '.length );
var co2 = (lines[i+3]+'').substring('Energia: '.length );
var trees = (lines[i+4]+'').substring('Arboles: '.length );

var values = {
    'energy':energy,
    'co2':co2,
    'trees':trees
}

//verificamos si el item ya fue agregado anteriormente
if(positionKeys.indexOf(title) < 0 ) {
    var item = {
        'title': title,
        'title2':title2,
        'img':getHotleImg(title),
        'today':values,
        'yesterday':null
    }

    itemsTpm.push(item);
    positionKeys.push(title);

}

}

var lastUpdate = (lines[1]+'').substring('Ultima actualizacion:
.length );
return {items: itemsTpm, lastUpdate: lastUpdate}
}

/*
Funcion que realiza el request
*/
function get() {
    var deferred = $q.defer();
    $http.get('http://pilve.iusiani.ulpgc.es/PILEV1.1-
war/ficherosTXT/datosLopesan.txt')
    .success(function(data, status, headers, config) {

```

```

        var items = [];
        var scrapingItems = scraping(data);
        deferred.resolve(scrapingItems);
    })
    .error(deferred.reject);
    return deferred.promise;
}

/*
Funcion que realiza el request
*/
function getConsumeTotal() {
    var items = this.items;

    var itemToday = { 'energy':0,'co2':0, 'trees':0 }
    var itemYesterday = { 'energy':0,'co2':0,'trees':0}

    for(var i = 0; i< items.length ; i++ ){
        itemToday.energy += extractNumber(items[i].today.energy);
        itemToday.co2 += extractNumber(items[i].today.co2);
        itemToday.trees += extractNumber(items[i].today.trees);

        itemYesterday.energy +=
extractNumber(items[i].yesterday.energy);
        itemYesterday.co2 += extractNumber(items[i].yesterday.co2);
        itemYesterday.trees += extractNumber(items[i].yesterday.trees);
    }

    itemToday.energy = itemToday.energy + " kWh"
    itemToday.co2 = itemToday.co2 + " kilos"
    itemToday.trees = itemToday.trees + " arboles"

    itemYesterday.energy = itemYesterday.energy + " kWh"
    itemYesterday.co2 = itemYesterday.co2 + " kilos"
    itemYesterday.trees = itemYesterday.trees + " arboles"

    return {today: itemToday,yesterday: itemYesterday};
}

return {
    'get':get,
    'getConsumeTotal':getConsumeTotal
};
});
})();

```

App.css

```
-webkit-animation: bounce 2.0s infinite ease-in-out;
animation: bounce 2.0s infinite ease-in-out;
}

.dot2 {
  top: auto;
  bottom: 0px;
  -webkit-animation-delay: -1.0s;
  animation-delay: -1.0s;
}

@-webkit-keyframes rotate { 100% { -webkit-transform: rotate(360deg) }}
@keyframes rotate { 100% { transform: rotate(360deg); -webkit-transform:
rotate(360deg) }}

@-webkit-keyframes bounce {
  0%, 100% { -webkit-transform: scale(0.0) }
  50% { -webkit-transform: scale(1.0) }
}

@keyframes bounce {
  0%, 100% {
    transform: scale(0.0);
    -webkit-transform: scale(0.0);
  } 50% {
    transform: scale(1.0);
    -webkit-transform: scale(1.0);
  }
}

.title-app{
  font-size: 16px;
  font-weight: bold;
}

.item-img-circle{
  margin: 0 auto;
  text-align:center;
}
```

```
.item-img-circle img{
  background-size: cover;
  background-repeat: no-repeat;
  background-position: center center;
  -webkit-border-radius: 99em;
  -moz-border-radius: 99em;
  border-radius: 99em;
  border: 5px solid #eee;
  box-shadow: 0 3px 2px rgba(0, 0, 0, 0.3);
}

/* mater*/
.error {
  color:red;
}

.error ons-row{
  margin-top: 100px; text-align: center;
}

.error h1{
  color:red;
}

.error p{
  color: #999;
  font-size: 15px;
}

.last-update{
  font-size: 15px;
  margin: 15px 10px;
  text-align: right;
  padding: 4px;
  font-weight: bold;
}

.item-master {
  padding: 40px;
  line-height: 1;
  height: 60px;
}

.item-master-thum {
  background-color: #ccc;
  width: 40px;
  height: 40px;
  border-radius: 100%;
  -webkit-border-radius: 100%;
  margin-top: 10px;
}
```

```
.item-master-title{
  font-size: 14px;
  font-weight: 500;
  text-transform: capitalize;
}

/**details*/
.item-detail {
  padding: 40px;
  line-height: 1;
}

.item-detail-title{
  font-size: 18px;
  text-transform: capitalize;
  font-weight: bold;
  margin-bottom: 10px;
  text-align: center;
}

.item-detail-label{
  line-height: 1.2;
  font-size: 16px;
  font-weight: bold;
}
.item-detail-value{
  line-height: 1.2;
  font-size: 14px;
}

.item-detail-cover-label {
  font-size: 20px;
  font-weight: 500;
  margin-right: 10px;
  margin-left: 10px;
}

.today .item-detail-cover {
  text-align: right;
```

```
}  
.yesterday .item-detail-cover {  
  text-align: left;  
}  
  
ons-carousel-item.today{  
  display: table;  
  text-align: left;  
}  
  
ons-carousel-item.yesterday{  
  display: table;  
  text-align: right;  
}
```


PLIEGO DE CONDICIONES

Pliego de condiciones

PL.1 Introducción

En el presente Pliego de Condiciones se exponen las condiciones bajo las que se ha desarrollado este Proyecto Fin de Carrera. A continuación se muestran el conjunto de herramientas hardware y software empleadas durante la realización del mismo.

PL.2 Condiciones hardware

Para elaborar este PFC se ha requerido el uso del material hardware siguiente:

Equipo	Modelo	Características
Ordenador de sobremesa	Gigabyte GA-EP43-DS3L	<ul style="list-style-type: none">– Procesador Intel Core 2 Duo a 3.00 GHz– 2 GB de memoria RAM– 1 TB de disco duro
Ordenador portátil	Samsung NP-R522H	<ul style="list-style-type: none">– Procesador Intel Core i2 a 2.16 GHz– 4 GB de memoria RAM– 500 GB de disco duro
Maleta Transportable KNX	-	<ul style="list-style-type: none">– Instalación domótica para pruebas de laboratorio

Tabla PL.1.- Recursos hardware utilizados

PL.3 Condiciones software

Los elementos software imprescindibles para el diseño, implementación y verificación del código desarrollado se muestran en la siguiente tabla:

Software	Versión
S.O. Windows	Windows 7 Professional 64 bits
Java	Java 8 de 64 bits
NetBeans	NetBeans IDE 8.0.2
Sublime Text	v.3
PhoneGap	v.2.9.1
ETS3	v.3.1.5
Microsoft Office	2013

Tabla PL.2.- Recursos software utilizados

PRESUPUESTO

Presupuesto

El presupuesto del presente proyecto se ha basado en las últimas listas de Honorarios Orientativos publicadas por el Colegio Oficial de Ingenieros Técnicos de Telecomunicación. Hay que señalar que estos listados se han venido publicando hasta el año 2008 pero debido a la modificación de la ley de Colegios Profesionales, mediante Ley 25/2009 de 22 de diciembre, no es posible seguir divulgando estas tablas, por lo que actualmente el COITT no puede elaborar baremos de honorarios, ni siquiera orientativos, salvo que sean con finalidad de tasar costas en procedimientos judiciales.

En este presupuesto se tratarán los siguientes conceptos:

- Trabajo tarifado por tiempo empleado.
- Amortización del inmovilizado material.
 - Amortización del material hardware.
 - Amortización del material software.
- Redacción del proyecto.

Por último se deben aplicar los impuestos correspondientes para el cálculo del **presupuesto total** del proyecto.

P.1 Trabajo tarifado por tiempo empleado

Se contabilizan los gastos que corresponden a la mano de obra según salario correspondiente a la hora de trabajo de un ingeniero técnico. El COITT propone utilizar la siguiente fórmula:

$$H (\text{€}) = (65 \cdot H_n + 78 \cdot H_e) \cdot Ct \quad (\text{P.1})$$

siendo:

- H los honorarios por tiempo.
- H_n las horas normales trabajadas dentro de la jornada laboral.
- H_e las horas especiales trabajadas.

- Ct coeficiente reductor por tramo.

Se estima que para el desarrollo del presente proyecto se ha destinado un total de 8 meses de trabajo, lo que implica que se necesiten 32 semanas, siendo el número de horas trabajadas por semana de 40 horas semanales. Esto hace un total de:

$$\text{Tiempo empleado} = 40 \text{ horas/semana} \cdot 32 \text{ semanas} = 1280 \text{ horas} \quad (\text{P.2})$$

El COITT recomienda aplicar a las horas de trabajo los coeficientes reductores por tramos de la tabla P.1

Horas Trabajadas	Coeficientes (C)
Hasta 36	1
Exceso de 36 a 72	0,9
Exceso de 72 a 108	0,8
Exceso de 108 a 144	0,7
Exceso de 144 a 180	0,65
Exceso de 180 a 360	0,60
Exceso de 360 a 510	0,55
Exceso de 510 a 720	0,50
Exceso de 720 a 1080	0,45
Exceso de 1080	0,40

Tabla P.3.- Coeficientes reductores para trabajo tarifado

Aplicando los coeficientes reductores de la tabla P.1, y puesto que se exceden las 1080 horas, se debe aplicar un coeficiente de reducción de 0,40. Considerando que no se ha trabajado horas especiales se obtiene, según P.1, que los honorarios son:

$$H = (65 \cdot 1280 + 78 \cdot 0) \cdot 0,40 = 33.280,00 \text{ €} \quad (\text{P.3})$$

Por tanto, el trabajo tarifado por tiempo empleado asciende a la cantidad de *treinta y tres mil doscientos ochenta euros con cero céntimos*.

P.2 Amortización del inmovilizado material

Dentro de este concepto hay que considerar tanto el hardware como el software empleado.

Se estipula que el coste de amortización para un periodo de 3 años, utilizando un sistema de amortización lineal o constante en el que se supone que el inmovilizado material se desprecia de forma constante a lo largo de su vida útil, la cuota de amortización anual se calcula haciendo uso de la P.4:

$$Cuota\ anual = \frac{\text{valor de adquisición} - \text{valor residual}}{\text{número de años de vida útil}} \quad (P.4)$$

El valor residual es el valor teórico que se supone tendrá el elemento en cuestión después de su vida útil.

P.2.1 Amortización del material hardware

Puesto que el proyecto se ha desarrollado durante 8 meses y el coste de amortización estipula un periodo de 3 años, los costes serán los derivados de los **8 meses de uso**.

En la Tabla P.2 se muestra el hardware necesario para la realización del proyecto, indicando para cada elemento su valor de adquisición, valor residual y coste de amortización.

Amortización del material hardware			
Elemento	Valor de adquisición	Valor residual	Coste de amortización
Ordenador de sobremesa	760,00 €	125,00 €	141,11 €
Portatil Samsung NP-R522H	629,00 €	100,00 €	117,55 €
Maleta Transportable KNX	3000 €	500,00 €	555,55 €
Total de amortización del hardware:			814,21 €

Tabla P.4.- Precios y coste de amortización del hardware

El costo de amortización del hardware asciende a *ochocientos catorce euros y veintiún céntimos*.

P.2.2 Amortización del material software

Al igual que se hizo en el caso del material hardware, los costes de amortización del material software serán los derivados de los primeros 8 meses y el coste de amortización estipula un período de 3 años.

La tabla P.3 muestra los elementos de software necesarios para la realización del proyecto, así como su valor de adquisición, valor residual y coste de amortización.

Amortización del material software			
Elemento	Valor de adquisición	Valor residual	Coste de amortización
Windows 7 Professional 64 bits	175,00 €	43,75 €	29,17 €
Java 8 (64bits)	0 €	-	-
NetBeans IDE 8.0.2	0 €	-	-
Sublime Text 3	0 €	-	-
PhoneGap	0 €	-	-
ETS3	250,00 €	50,00 €	44,44 €
MS Office 2013	269,00 €	70,00 €	44,22 €
Total de amortización del software:			117,83 €

Tabla P.5.- Precios y costes de amortización de software

Por tanto, el coste de amortización del software asciende a la cantidad de *ciento diecisiete euros y ochenta y tres céntimos*.

P.3 Redacción del proyecto

En esta sección se consignan los costes relacionados con la redacción del proyecto y demás materiales fungibles utilizados, como son: material de papelería, discos DVD-R, tinta de impresora láser y encuadernación. Estos costes están representados en la siguiente tabla:

Redacción del proyecto			
Concepto	Cantidad	Precio/Unidad	Importe
Fotocopias	618	0,04 €	24,72 €
Anillado	3	7,00 €	21,00 €
Discos DVD	3	1,25 €	3,75 €
Total redacción del proyecto:			49,47 €

Tabla P.6.- Gastos derivados de la redacción del proyecto

El coste total de la redacción de este proyecto asciende a *cuarenta y nueve euros con cuarenta y siete céntimos*.

P.4 Presupuesto final

A las sumas parciales calculadas hasta ahora hay que añadir el 7% del IGIC, quedando el presupuesto total como sigue:

Presupuesto Final	
Concepto	Importe
Trabajo tarifado por tiempo trabajado	33.280,00 €
Amortización del material hardware	814,21 €
Amortización del material software	117,83 €
Redacción del proyecto	49,47 €
Total (sin IGIC)	34.261,51 €
IGIC (7%)	2.398,30 €
Total	36.659,81 €

Tabla P.7.- Presupuesto final

Por tanto, el presupuesto final, incluyendo impuestos, asciende a la cantidad de *treinta y seis mil seiscientos cincuenta y nueve euros con ochenta y un céntimos*.

En Las Palmas de Gran Canaria, a 21 de julio de 2015

Fdo: Sergio Martín Medina

Autor del proyecto

