

UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

Grado en Ingeniería Eléctrica

PROYECTO

Fin de Grado

Título:
Monitorización de Sistemas de Energías
Renovables mediante protocolos IP

Autor: Joan Miquel Carbonell Crespí

Director: Manuel Alonso Castro Gil

Codirector:

Ponente:

Departamento: Ingeniería Eléctrica,
Electrónica y de Control

ESCUELA TÉCNICA SUPERIOR DE
INGENIEROS INDUSTRIALES

UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

TÍTULO: Monitorización de Sistemas de Energías Renovables
mediante protocolos IP

DEPARTAMENTO: Ingeniería Eléctrica, Electrónica y de Control

AUTOR: Joan Miquel Carbonell Crespí

DIRECTOR: Manuel Alonso Castro Gil

CODIRECTOR:

PONENTE:

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES

TÍTULO DEL PROYECTO:

Monitorización de sistemas de energías renovables mediante protocolos IP

(A rellenar por el tribunal calificador)

TRIBUNAL CALIFICADOR

PRESIDENTE:

VOCAL:

SECRETARIO:

Fecha de defensa:

CALIFICACIÓN:

Vocal

Presidente

Secretario

fdo:

fdo:

fdo:

Índice

RESUMEN	9
LISTA DE SÍMBOLOS	13
LISTA DE FIGURAS	15
CONTENIDOS DEL PROYECTO	
1. Introducción	17
2. Elementos del proyecto anterior	19
2.1. La instalación fotovoltaica y su software de gestión	19
2.1.1. Descripción de la instalación	19
2.1.2. El inversor Sunny Boy 1100E	20
2.1.3. El controlador Sunny Boy Control SBC	22
2.1.4. Sunny data control (sdc) y sunny data agent (sda)	24
2.2 La estación meteorológica	29
2.2.1 Descripción de la instalación	29
2.2.2 Software de control de la estación meteorológica	30
3. Funcionamiento del proyecto anterior	35
3.1 Software instalado	35
3.2 Arquitectura	35
3.3 Auditoría	36
3.3.1 Revisión de servicios y mantenimiento de los datos de las estaciones ..	36
3.3.2 Funcionalidades de la aplicación	38
3.3.3 Problemas de la aplicación	38
4. Nuevo proyecto	39
4.1 Brainboxes ethernet to serial	39
4.2 Posibilidades para el nuevo proyecto	41
4.3 Nueva versión en html5	42
4.3.1 Ventajas	42
4.3.2 Cambios	42
4.3.3 Interfaz de usuario	43

Proyecto de Fin de Grado

4.3.4 Resumen anual de valores meteorológicos y fotovoltaicos	49
4.3.5 Librería gráfica javascript: chart.js	50
4.3.6 Librería jxl y acceso a ficheros mdb	50
4.3.7 Lectura de valores instantáneos.	51
4.3.8 Plantilla Iridium para HTML5	52
4.3.9 Descripción del trabajo realizado	53
5. Conclusiones y posibles mejoras	55
6. Bibliografía	57
Anexos	
A1. Contenido del CDROM	59
A2. Instalación y mantenimiento	61
A2.1 Paquetes a instalar y configuraciones	61
A2.2 Instalación de servicios	62
A2.3 Copia de seguridad de los datos	64
A3. Código fuente de la aplicación	65
CV del autor	107

RESUMEN DEL PROYECTO

El siguiente proyecto tiene como finalidad la monitorización de las estaciones fotovoltaica y meteorológica de la Escuela Técnica Superior de Ingenieros Industriales de la Universidad Nacional de Educación a Distancia.

Para ello se ha procedido a intentar mejorar un proyecto anterior de monitorización, basado en aplicaciones Java de acceso al software de control de las estaciones, aunque en el nuevo proyecto se ha optado por utilizar HTML5 y tecnología JSP para poder acceder a través de cualquier navegador, sistema operativo o dispositivo.

Durante la ejecución del mismo, se ha procedido a la programación de aplicaciones JSP y al diseño de páginas HTML y JavaScript, en un servidor virtual de la escuela, en el cuál está instalado todo el software de control del fabricante de las diferentes estaciones, que junto con el servidor Tomcat, permite a los usuarios visualizar los datos. Su funcionamiento se describe con profundidad en capítulos posteriores.

Ya para acabar esta pequeña introducción, me gustaría añadir que la motivación para realizar este Proyecto de Fin de Grado ha sido debida a mi interés por la Energía Fotovoltaica. También me ha resultado interesante trabajar con una estación meteorológica, que se complementa muy bien con el análisis de datos fotovoltaicos.

Proyecto de Fin de Grado

ABSTRACT

The aim of this project is monitoring photovoltaics and meteorological stations at School of Industrial Engineering at *Universidad de Educación a Distancia (UNED)*.

As a result, it has been implemented an enhanced project to monitor both stations. The old version of the project was based on Java applets to access to the control of photovoltaics and meteo stations. In the new version HTML5 and JSP technologies have been used to let any browser, operating system or device monitor data from both stations.

In reference to the implementation, a JSP software has been developed and also HTML and JavaScript files have been designed in a virtual server installed in infrastructure data network at UNED. This station, working together with Tomcat and manufacturer controllers, let Internet users access to photovoltaic and meteo data. In following chapters, we talk in a deeply way about all this software.

And finally, I would like to mention my motivation to work with this project: I am really interested in Photovoltaic Energy. Also it has been interesting working with the meteorological workstation, which is an important complement to work with solar energy equipment.

Proyecto de Fin de Grado

LISTA DE SÍMBOLOS

ETSII	Escuela Técnica Superior de Ingenieros Industriales.
UNED	Universidad Nacional de Educación a Distancia.
SBC	Sunny Boy Control.
PC	<i>Personal Computer</i> . Ordenador Personal.
SDC	<i>Sunny Data Control</i> . Control de Datos de Sunny
TCP	<i>Transmission Control Protocol</i> . Protocolo de Control de Trasmisión.
IP	<i>Internet Protocol</i> . Protocolo de Internet.
RS232	<i>Recommended Standard 232</i> . Standard Recomendado 232
IGBT	<i>Isolated Gate Bipolar Transistor</i> . Transistor de Puerta Bipolar
FV	Fotovoltaica.
MPP	<i>Maximum Power Point</i> . Punto de Máxima Potencia
DC	<i>Direct Current</i> . Corriente Continua
RS485	<i>Recommended Standard 485</i> . Standard Recomendado 485
SAD	Sistema de Adquisición de Datos.
JSP	<i>Java Servlet Pages</i> .
HTML	<i>Hyper Text Markup Protocol</i> .
SDA	<i>Sunny Data Agent</i> .
IOS	<i>Iphone Operating System</i> .
SNMP	<i>Simple Network Management Protocol</i> . Protocolo Simple de Gestión de Red.
MRTG	<i>Multi Router Traffic Grapher</i> .
MIB	<i>Management Information Base</i> .
HTML5	<i>Hyper Text Markup Protocol v5</i> .
VPS	<i>Virtual Private Server</i> . Servidor Privado Virtual.

Proyecto de Fin de Grado

LISTA DE FIGURAS

Figura1. Conexionado de los inversores,el SBC y PC con SDC	19
Figura2. Sunny Boy 1100E	21
Figura3. Sunny Boy Control	23
Figura4. Selección de canales de datos de la estación FV.....	25
Figura5. Pantallas de visualización de datos de la estación FV	26
Figura6. Estructura de los ficheros de datos de la estación FV	27
Figura7. Interacción del SDC y el SDA con el servidor web	28
Figura8. Software Datagraph	31
Figura9. Conexión a la base de datos Access	32
Figura10. Acceso a través de Internet a los datos meteorológicos	32
Figura11. Listado de tablas de un fichero mdb de la estación meteorológica.....	33
Figura12. Valores y campos de la tabla de datos del fichero mdb	34
Figura13. Arquitectura de acceso a los datos	36
Figura14. Comprobación del estado de los servicios	37
Figura15. Brainboxes. Ethernet to serial	39
Figura16. Configuración del puerto COM virtual	40
Figura17. Pantalla inicial de la aplicación	43
Figura18. Pantalla de introducción de datos FV	44
Figura19. Pantalla de gráficos FV	45
Figura20. Pantalla de datos FV	46
Figura21. Pantalla de gráficos meteo	47
Figura22. Pantalla de datos meteo	48
Figura23. Pantalla de resumen anual de energía FV	49
Figura24. Licencia de la plantilla utilizada	52
Figura25. Principales ficheros de código de la aplicación.....	53

Proyecto de Fin de Grado

CONTENIDOS DEL PROYECTO

1.INTRODUCCIÓN

La Escuela Técnica Superior de Ingeniería Industrial (ETSII) de la Universidad Nacional de Educación a Distancia (UNED) dispone de una estación fotovoltaica y una estación meteorológica conectadas a un ordenador del departamento de Ingeniería Eléctrica, Electrónica y de Control, de forma que los datos recogidos en las mismas pueden ser monitorizados a través de un servidor con servlets y páginas jsp almacenadas en un servidor Apache-Tomcat, que interacciona con programas de gestión suministrados por las empresas fabricantes de ambas estaciones.

El proyecto anterior, realizado por José María Estepa Martínez y dirigido por el Dr. Manuel Alonso Castro Gil, titulado “**Rediseño del sistema de monitorización de energías renovables: acceso, estandarización y seguridad**” es accesible desde la siguiente dirección de Internet:

http://meteo.ieec.uned.es:8086/PFC_JMEstepa/

El objeto del nuevo proyecto, titulado “**Monitorización de Sistemas de Energías Renovables mediante protocolos IP**”, ha consistido en acceder de nuevo a las dos estaciones, pero realizando un sistema estándar accesible mediante todo tipo de dispositivos IP. La motivación para la realización del nuevo proyecto es debida al interés por la gestión y análisis de datos de las estaciones fotovoltaicas, para las cuales se debe tener en cuenta también las condiciones meteorológicas.

El contexto de trabajo en el que hemos implementado este nuevo proyecto ha consistido en el acceso a datos de hojas de cálculo y bases de datos generadas por el software de control del fabricante y mediante algoritmos implementados en páginas web dinámicas, hemos hecho posible el acceso a la información mediante el navegador de diferentes dispositivos IP.

El nuevo proyecto es accesible a través de Internet desde la siguiente dirección:

http://meteo.ieec.uned.es:8086/PFGr_JoanCarbonell

Para la realización del proyecto se ha utilizado, además de los elementos típicos de una instalación fotovoltaica complementada con la instalación meteorológica y cuyo equipamiento y funcionamiento describiremos en los siguientes puntos, un equipo conversor de puerto serie a puerto ethernet para hacer más versátil la comunicación entre el controlador de la estación fotovoltaica, Sunny Base Controller (SBC) y el software instalado en el PC, Sunny Data Control

Proyecto de Fin de Grado

(SDC), permitiendo además comunicación remota a través de TCP/IP entre ambos.

Además en el ordenador donde tenemos el software de control, se ha instalado un servidor web de aplicaciones para poder ejecutar los algoritmos de conversión.

2. ELEMENTOS DEL PROYECTO ANTERIOR

2.1. LA ESTACIÓN FOTOVOLTAICA Y SU SOFTWARE DE GESTIÓN

2.1.1. DESCRIPCIÓN DE LA INSTALACIÓN

El equipo está formado por siete módulos fotovoltaicos en serie y un ondulador de secuencias o inversor, del tipo Sunny Boy 1100E. La corriente continua producida por los generadores solares es convertida por el inversor en corriente alterna que se introduce directamente en la red eléctrica.

El control de la instalación se realiza con dos componentes adicionales, que se explicarán con detalle posteriormente:

- Controlador Sunny Boy Control (SBC)
- Un PC con el programa de Windows: Sunny Data Control (SDC).

Estos elementos permiten la monitorización y el registro continuo de los datos de trabajo de los inversores conectados.

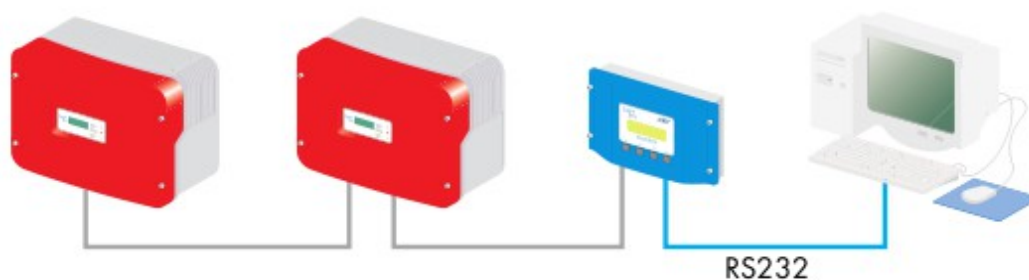


Figura1. Conexión de los inversores, el SBC y PC con SDC

2.1.2. EL INVERSOR SUNNY BOY 1100E

La tensión continua fotovoltaica se traslada por un montaje de puente IGBT (Insulated Gate Bipolar Transistor o Transistor Bipolar de Puerta Aislada) de alta frecuencia (16 kHz) a un circuito intermedio de corriente alterna. Desde allí se efectúa la alimentación directa a la red por un transformador toroidal de baja pérdida. El rango de tensión de entrada FV del Sunny Boy está dimensionado según configuración de la instalación hasta para 16 módulos fotovoltaicos estándar. La regulación de la corriente garantiza la forma sinusoidal de la curva con coeficiente mínimo de distorsión no lineal. El control de marcha garantiza la marcha completamente automática y la búsqueda y seguimiento del punto de trabajo de rendimiento máximo (MPP, Maximum Power Point o Punto de Máxima Potencia), evitándose pérdidas innecesarias en reserva y en marcha de alimentación.

Las magnitudes de entrada del Sunny Boy son las siguientes:

- Tensión máxima de entrada sin carga (VPV0): 400 V DC.
- Tensión mínima rango MPP (VMPPmin): 139 V DC.
- Tensión de entrada rango MPP (VPV): 139 V – 400 V DC.
- Corriente máxima de entrada (IPV max): 10,0 A.
- Potencia máxima de entrada (PPV): 1.210 W.
- Potencia máxima recomendada del generador: 1.500 W.
- Instalación de corte de todos los polos en el lado de entrada DC: Conectores de inserción MC (Multi-Contact).
 - Tensión de rizado (USS) < 10%.
 - Protección de personas mediante control de contacto a tierra (Riso > 1 M Ω).
 - Protección de sobretensión mediante varistores de control térmico.
 - Protección contra polarización inversa mediante diodos de cortocircuito.

Y las de salida (conexión a la red):

- Potencia nominal de salida (PACnom): 1.000 W.
- Potencia punta de salida (PACmax): 1.100 W.
- Resistencia al cortocircuito: de red por regulación de corriente.
- Rango de trabajo de la tensión de red (UAC): 198 – 251 V AC.
- Rango de trabajo de la frecuencia de red (fAC): 49,8 – 50,2 Hz.
- Instalación de corte de todos los polos en el lado de la red: Instalación automática de desconexión (ENS), doble.
 - Ángulo de desfase (φ) referido a onda fundamental de la corriente: 0o.
 - Rendimiento máximo (η_{max}): $\geq 93\%$.
 - Consumo de potencia en marcha < 4 W. En marcha nocturna < 0,1 W.
 - Rango de temperatura ambiente permitido: -25 oC a +60 oC.
 - Humedad relativa permitida: del 0% al 100% (clase 3K6).

Monitorización de estaciones FV y meteo



Figura2. Sunny Boy 1100E

La transferencia de datos entre el controlador (Sunny Boy Control) y el software instalado en el PC (Sunny Data Control) se hace por TCP/IP gracias al convertor Brainboxes serial to ethernet.

En realidad, el SBC tiene un conector serie RS232 conectado al convertor TCP/IP que será transparente a la comunicación, y finalmente en el ordenador donde tenemos el software SDC añadimos un puerto virtual COM asociado a un socket TCP y configuramos la comunicación de datos a través de ese puerto COM.

El Sunny Boy solo debe utilizarse en funcionamiento paralelo a la red. Para la desconexión segura en caso de separación de la red y para evitar la marcha aislada, el inversor está equipado de fábrica con un punto de desconexión automático. Este punto de desconexión de funcionamiento automático está equipado por razones de máxima seguridad con dos Instalaciones de Vigilancia de la red con órgano de conmutación respectivamente adjudicado en

serie. Cada una de estas instalaciones controla permanentemente la calidad de la red conectada por control de la tensión, frecuencia e impedancia. La construcción redundante y un autotest automático antes de cada conexión a la red aseguran la fiabilidad de funciones.

2.1.3. EL CONTROLADOR SUNNY BOY CONTROL (SBC)

El SBC trabaja como unidad central de registro de datos de medición y diagnóstico hasta para 50 inversores, apoya la puesta en marcha de la instalación FV, y ofrece la posibilidad de telediagnóstico por fax y módem. Por el contrario, el número de modelos de inversor diferentes está restringido a cuatro. La capacidad de almacenamiento para los datos adquiridos depende del número de inversores y de su configuración. El volumen de almacenamiento es dimensionado para poder guardar los valores diarios de energía de cada inversor simple durante al menos un año. La memoria restante es usada para el almacenamiento de los canales de medición. Según el número de canales de medición seleccionados y el intervalo de medición definido, resulta una capacidad de almacenamiento diferente de los datos medidos (número de días hasta que los valores más antiguos son sobrescritos).

Los canales de medición son los canales que se seleccionan para registrar sus datos en la memoria del sistema. Cualquier canal que un inversor Sunny Boy pueda medir se puede definir como un canal de medición. Para operar con la mayor profundidad de almacenamiento posible, el registro de datos se debe limitar a una selección de canales. Los canales de medición se almacenarán como el valor promedio de los valores en los intervalos de medición definidos. Por ejemplo, si se selecciona un canal y se define que el intervalo de medición sea 15 minutos, esto significa que todos los valores puntuales de ese canal se usarán para calcular el valor promedio de esos 15 minutos, y ese promedio y el tiempo correspondiente se registrarán en la memoria del sistema. Los canales de medición disponibles del Sunny Boy son los siguientes:

1. Vpv: Tensión de entrada FV.
2. Vpv-Setpoint: Tensión nominal FV del regulador Vpv interno.
3. Iac: Corriente de red.
4. Vac: Tensión de red.
5. Fac: Frecuencia de red.
6. Pac: Potencia suministrada a la red.
7. Zac: Impedancia de la red.
8. Riso: Resistencia de aislamiento.
9. Ipv: Corriente del generador FV.
10. E-Total: Suma total de la energía alimentada.
11. H-Total: Suma total de las horas de trabajo en marcha de alimentación.
12. Power On: Suma total de las conexiones a la red.
13. Serial Number: Número de serie del Sunny Boy.
14. Mode: Visualización del estado actual de trabajo.
15. Error: Visualización del tipo de fallo en estado de fallo.

Monitorización de estaciones FV y meteo

Cada registro de datos está sujeto a más o menos errores. Los valores de medición registrados por el inversor son necesarios para la dirección de marcha y para la regulación de la corriente alimentada. La reproducibilidad de los valores de medición está adaptada a estas funciones. El error máximo del registro de datos a una temperatura ambiente de 25°C se encuentra en un rango entre el 0,1% y el 4%. A otras temperaturas ambiente se debe tener en cuenta un coeficiente de error por temperatura.

El SBC presenta los siguientes conectores e interfaces externas:

- Un conector de comunicación con la línea de potencia, que es la opción estándar para la transmisión de datos con el inversor.
- Un puerto COM 1, opción alternativa de conexión con el inversor mediante un cable RS485.
- Un puerto COM 2, que se conecta con un PC con el SDC mediante un cable RS232.
- Un conector RELAIS OUT.

Se pueden adquirir y evaluar todos los datos de un SBC (datos de los canales de medición y valores de energía diarios) con un PC y el SDC. Además todas las configuraciones de un SBC pueden ser visualizadas y modificadas con el SDC. Esto es independiente del tipo de conexión del PC al SBC.

Durante su funcionamiento el SBC cambia al modo de ahorro de energía por la noche (después de no conseguir conectar con ningún dispositivo FV durante 15 minutos). Cada 15 minutos se despierta y chequea si puede conectar con algún dispositivo. Si no puede, vuelve al modo de ahorro de energía, en caso contrario reanuda su funcionamiento normal.



Figura3. Sunny Boy Control

2.1.4. EL SOFTWARE DE CONTROL SUNNY DATA CONTROL (SDC) Y EL SUNNY DATA AGENT (SDA)

Sunny Data Control es un programa de SMA Technologies AG con el que se puede controlar el almacenamiento de los datos de la planta fotovoltaica y visualizarlos.

Sunny Data Control consulta los datos de su dispositivo de comunicación (Sunny Boy Control) y los guarda como archivos de Microsoft Excel o archivos CSV.

Con Sunny Data Control, por ejemplo, puede configurar una visión completa de todos los datos del inversor y mostrar los datos en formato Excel.

Para mostrar los datos de funcionamiento actuales dispone de distintas modalidades de visualización. Para manejar la gran cantidad de información disponible, puede combinar estas modalidades en función del tamaño de su instalación y de sus requisitos en lo relativo a la claridad y el valor informativo de la visualización.

Dispone de las siguientes modalidades de visualización de los valores puntuales.

- Vista panorámica (con campos individuales para cada equipo)
- Información dispositivo individual
- Información corta
- Vista general

A cada modo de visualización le puede asignar los canales de medición deseados. Cada canal puede aparecer en más de un modo de visualización. La selección de canales puede ajustarse de forma individual para cada equipo o cada tipo de equipo.

Monitorización de estaciones FV y meteo

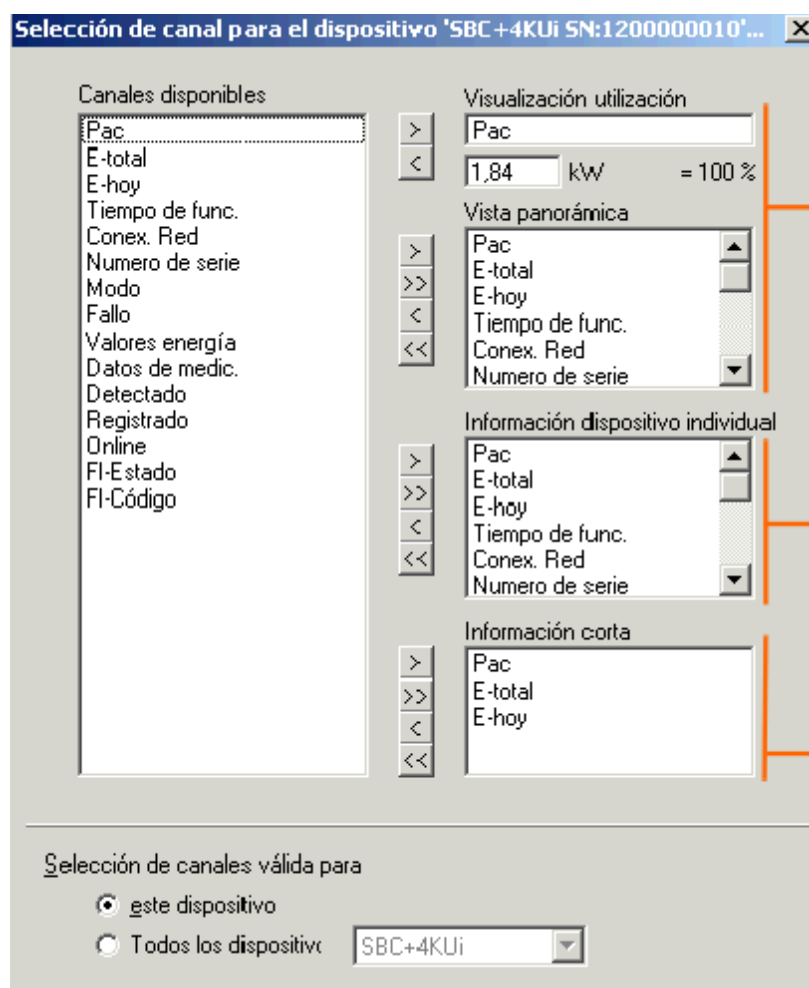


Figura4. Selección de canales de datos de la estación FV

<u>SunBC-17 SN:106367711</u>		<u>WR11E-04 SN:654407248</u>	
Pac	0.000 kW	Vpv	--- V
E-Total	14891.990 kWh	Vpv-Setpoint	--- V
E-today	5.633 kWh	Iac	--- mA
Operating Time	89993.663 h	Vac	--- V
Power On	283	Fac	--- Hz
Serial Number	106367711	Pac	--- W
Mode	waiting	Zac	--- Ohm
Error	---	Riso	--- kOhm
Energy Values	374 days	Ipv	--- mA
Measuring Data	5216 cycles	E-Total	--- kWh
Detected	1 devices	h-Total	--- h
Registered	1 devices	Power On	---
Modem-Status	deactivated	Serial Number	---
		Mode	---
		Error	---

Figura5. Pantallas de visualización de datos de la estación FV

Para cada Sunny Boy Control existe un subdirectorío cuyo nombre se compone de las letras "SBC" y el número de serie del equipo (p.ej. SBC119010325).

Este subdirectorío contiene los archivos con los datos de medición guardados. Los datos de medición de los diferentes canales se clasifican por días y se guardan por meses en los subdirectoríos de los equipos correspondientes en archivos Excel separados.

El número de archivos queda determinado por el período de medición. Si por ejemplo se registraron datos de medición con el Sunny Boy Control durante un año entero, su disco duro contendrá doce archivos con los valores mensuales.

Monitorización de estaciones FV y meteo

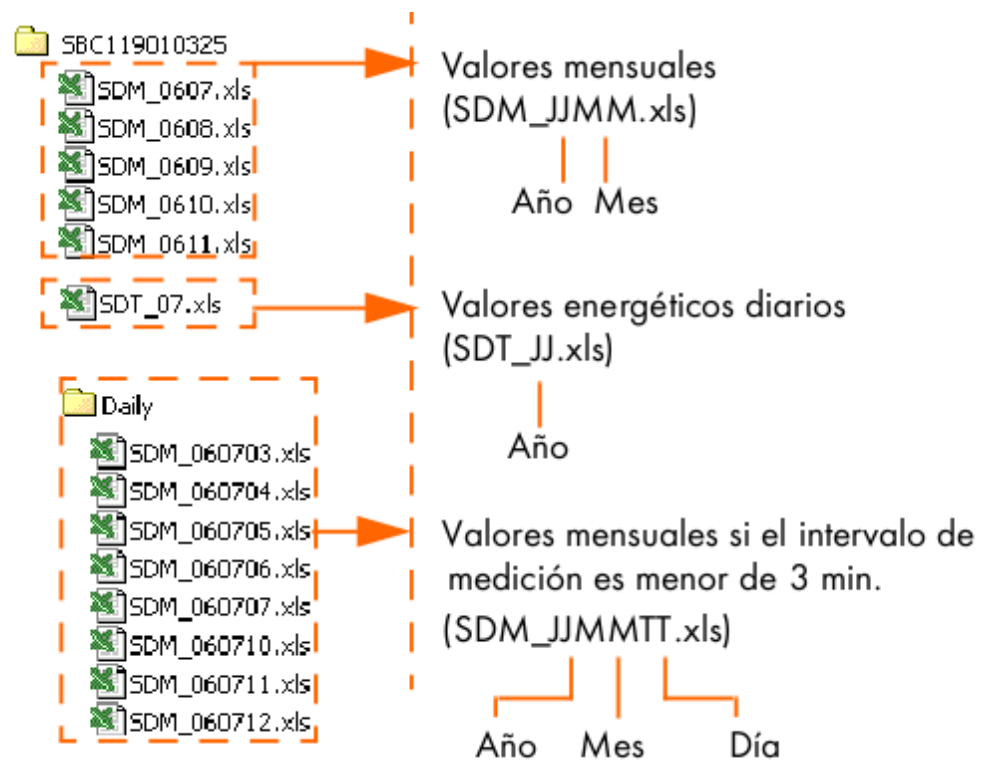


Figura6. Estructura de los ficheros de datos de la estación FV

Proyecto de Fin de Grado

El agente SDA actúa como intermediario de comunicación entre el Sunny Data Control y los applets de Java en el servidor web. El agente SDA puede gestionar tres conexiones de servidor, por lo tanto tres plantas fotovoltaicas se pueden comunicar simultáneamente a través del mismo servidor.

Estos dos programas deberían instalarse como servicios del sistema, ya que al arrancar la máquina las aplicaciones web deben acceder a la información que generan y es necesario que arranquen con el sistema operativo.

El SDA puede configurarse para cambiar los puertos de trabajo por los que servirá los applets, ello se consigue accediendo al PC del SDA por el puerto 18501 a través de un navegador.

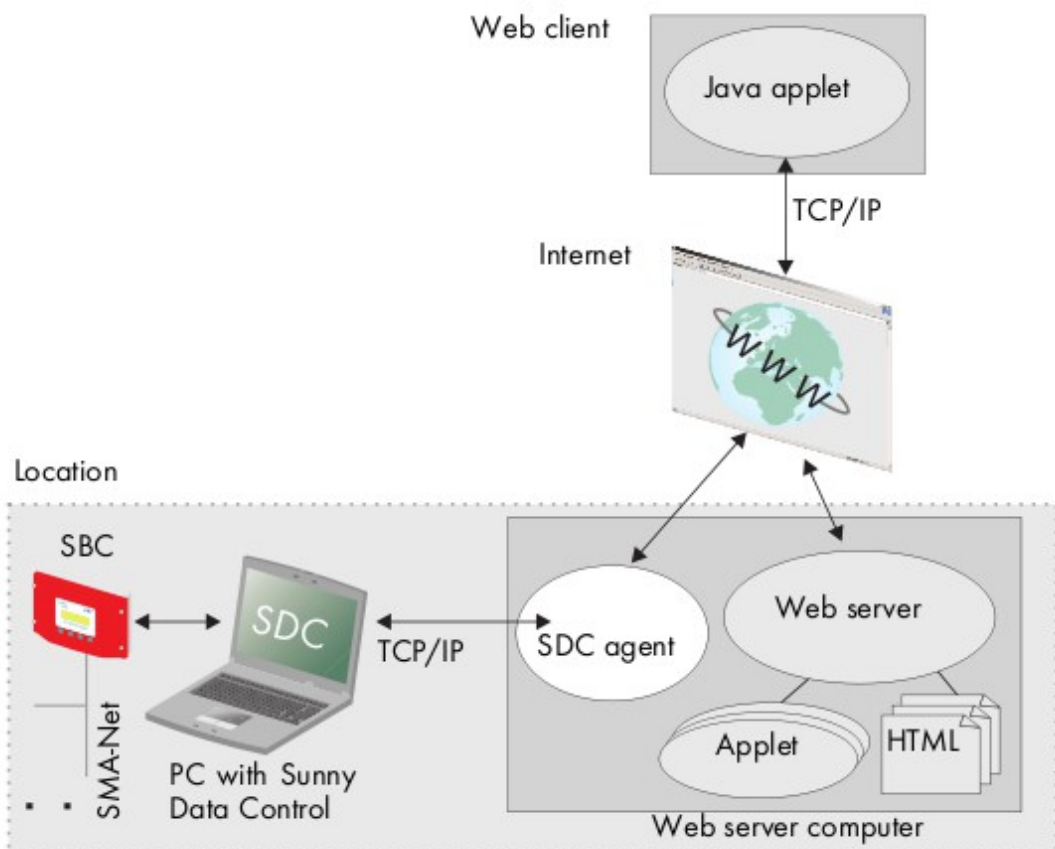


Figura7. Interacción del SDC y el SDA con el servidor web

2.2 LA ESTACIÓN METEOROLÓGICA

2.2.1 DESCRIPCIÓN DE LA INSTALACIÓN

La estación meteorológica de la serie 3000 de Geonica tiene un alto grado de funcionalidad , estando formada por diversos componentes hardware que describimos a continuación:

Canal físico. Corresponde con cada una de las entradas físicas de las que dispone el equipo y que permiten la conexión de las señales procedentes de los sensores. Dichas señales pueden proceder de un sensor externo o de otro sistema de medida o dispositivo, y pueden ser de naturaleza analógica o digital.

Canal analógico. Los canales analógicos, antes de procesarse, necesitan ser digitalizados. Para ello se utiliza un conversor analógico-digital que está incluido en el SAD. La conversión de la señal a digital será tanto más fiel cuanto mayor sea la resolución y la precisión de dicha conversión. Con una resolución de 19 bits más signo (20 bits), que es la que utiliza este modelo, se tiene la posibilidad de discriminar una parte entre 524.288 partes de la señal de entrada, tanto para señales positivas como negativas. Esta gran resolución evita tener que amplificar las señales de entrada, eliminando así los errores que se derivan del proceso de amplificación presentes en otros equipos de menor resolución.

Canal digital. A diferencia de los analógicos, los canales digitales no necesitan ser convertidos y por tanto, la información que aportan puede ser tratada directamente por el microprocesador del SAD. Estos canales digitales pueden ser de cuatro tipos distintos:

- Frecuencia-periodo.
- Acumuladores-contadores.
- De estado.
- Canales inteligentes.

Canal lógico. Corresponde con cada una de las variables o parámetros que se desea medir. En condiciones normales cada canal físico se corresponde con una variable a medir y por tanto con un canal lógico.

Periodo de almacenamiento: Permite al usuario definir cada cuánto tiempo desea registrar en la memoria de almacenamiento del sistema, los cálculos y datos estadísticos obtenidos por el equipo.

El SAD de la ETSII de la UNED, dispone de sensores para registrar los siguientes parámetros:

- Velocidad y dirección del viento.
- Temperatura del aire.
- Humedad relativa.
- Presión atmosférica.
- Radiación solar.
- Lluvia.

2.2.2 SOFTWARE DE CONTROL DE LA ESTACIÓN METEOROLÓGICA

Para el caso de la estación meteorológica tenemos una configuración similar, el SAD (Sistema de Adquisición de Datos) se ha conectado a través del puerto serie 232 con un Device Server que la conecta a la red Ethernet de la Escuela.

Se ha configurado como servidor en el puerto 30000. De esta forma, la estación es visible a través de Internet y accesible por el programa Teletrans W3K en la dirección 62.204.201.170:30000.

El programa Teletrans W3K permite la recogida de datos que se envían a una Base de Datos Access y que serán después recogidos por una aplicación instalada en el servidor Apache Tomcat formada por servlets y páginas jsp.

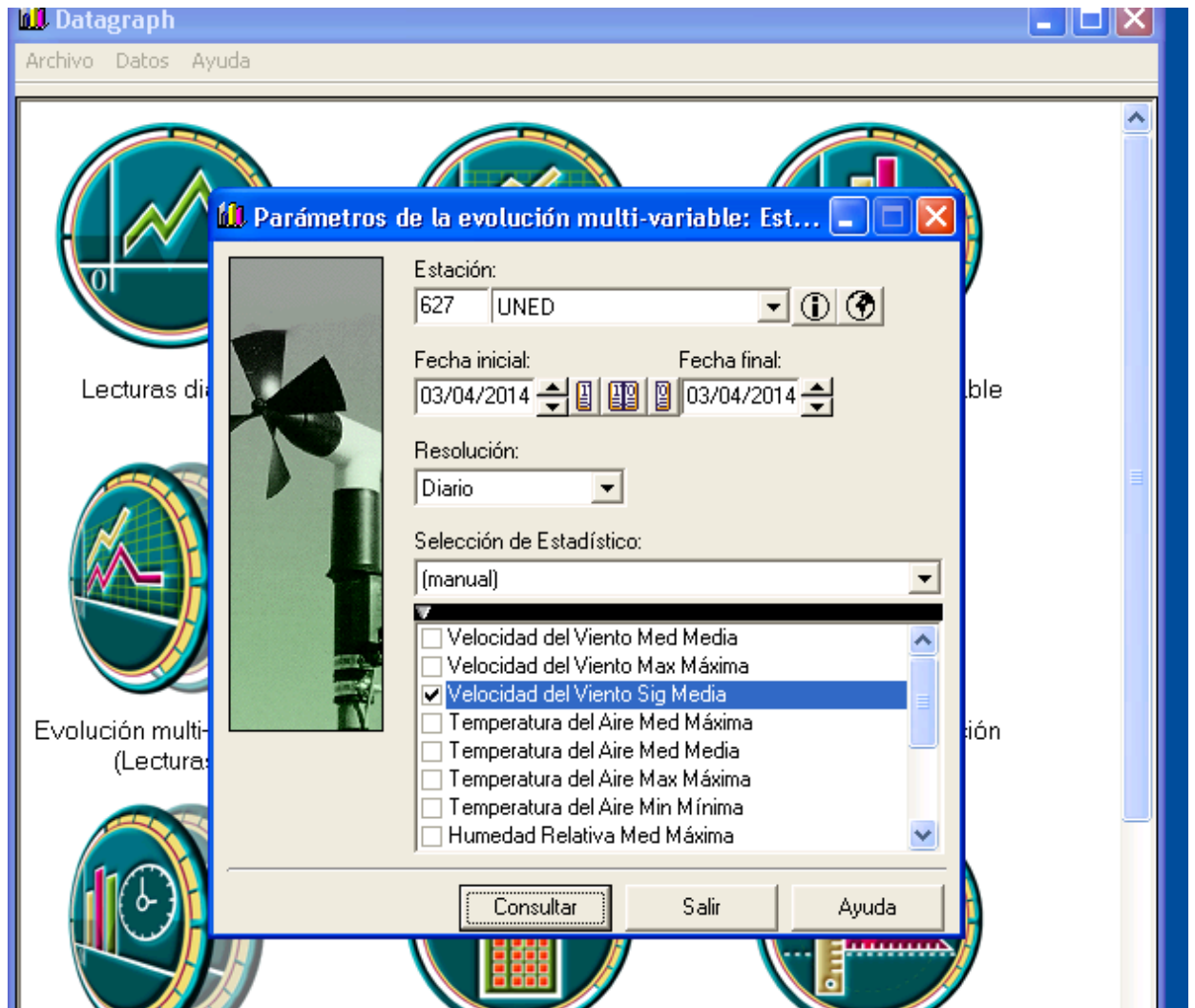


Figura8. Software Datagraph

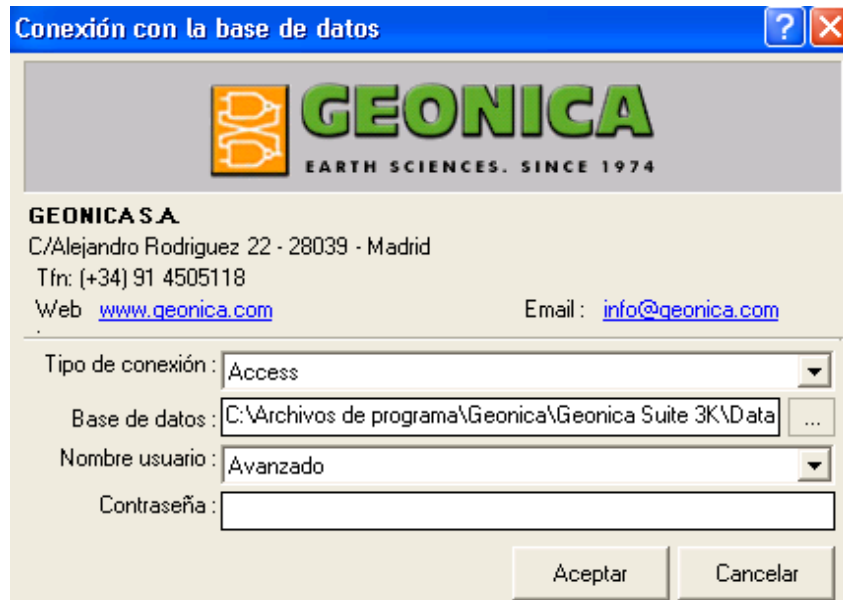


Figura9. Conexión a la base de datos Access

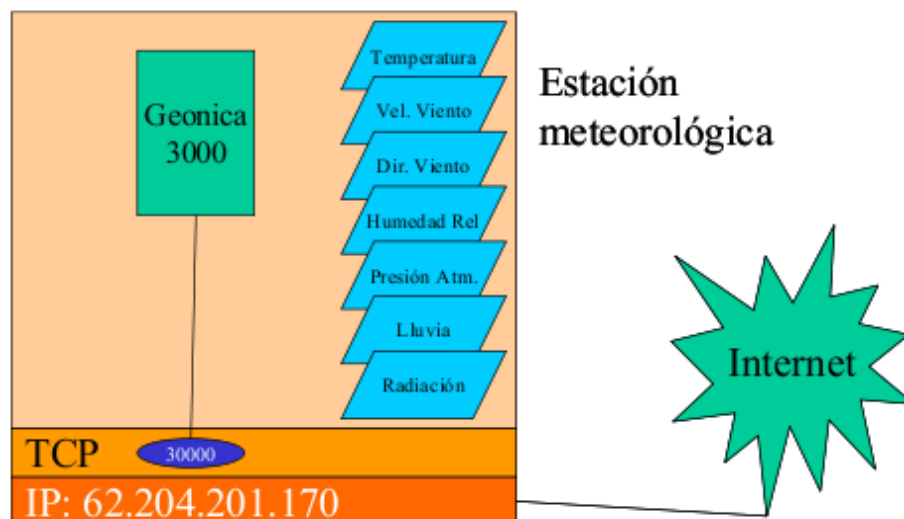
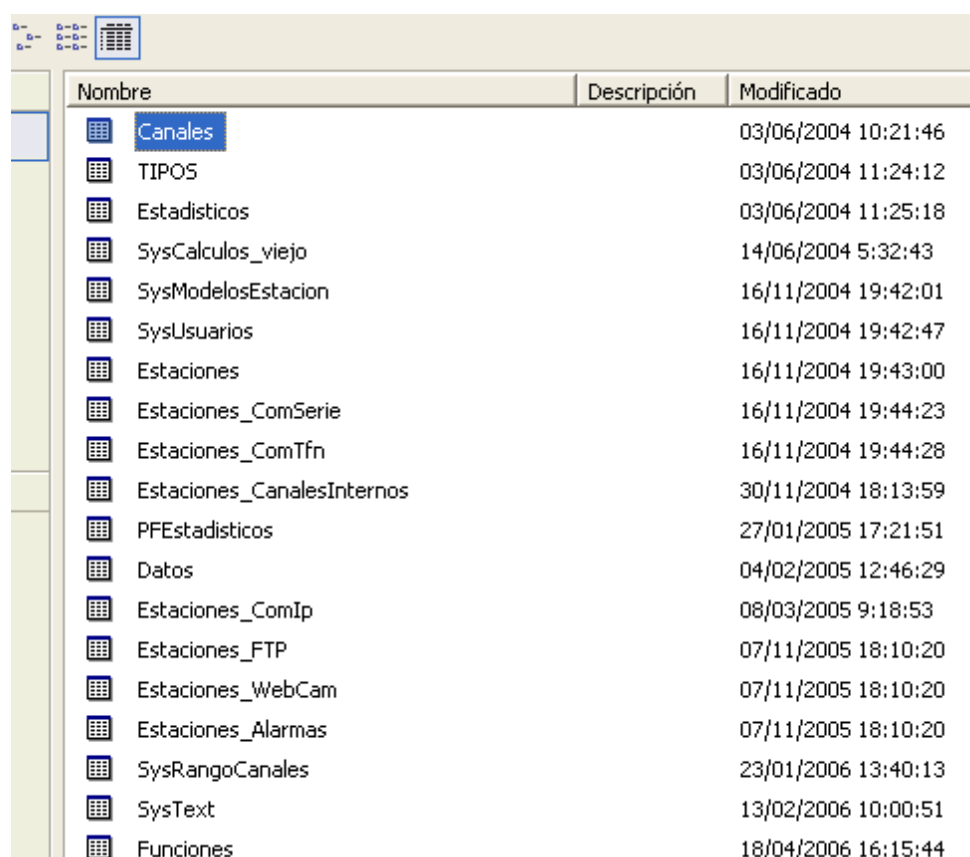


Figura10. Acceso a través de Internet a los datos meteorológicos (Obtenida del PFC de JMEstepa)

Monitorización de estaciones FV y meteo

El software Geonica3000, lo que hace es crear una base de datos mdb para cada año diferente, y en esta base de datos, en diferentes tablas, va guardando valores de medición.



Nombre	Descripción	Modificado
Canales		03/06/2004 10:21:46
TIPOS		03/06/2004 11:24:12
Estadisticos		03/06/2004 11:25:18
SysCalculos_viejo		14/06/2004 5:32:43
SysModelosEstacion		16/11/2004 19:42:01
SysUsuarios		16/11/2004 19:42:47
Estaciones		16/11/2004 19:43:00
Estaciones_ComSerie		16/11/2004 19:44:23
Estaciones_ComTfn		16/11/2004 19:44:28
Estaciones_CanalesInternos		30/11/2004 18:13:59
PFEstadisticos		27/01/2005 17:21:51
Datos		04/02/2005 12:46:29
Estaciones_ComIp		08/03/2005 9:18:53
Estaciones_FTP		07/11/2005 18:10:20
Estaciones_WebCam		07/11/2005 18:10:20
Estaciones_Alarmas		07/11/2005 18:10:20
SysRangoCanales		23/01/2006 13:40:13
SysText		13/02/2006 10:00:51
Funciones		18/04/2006 16:15:44

Figura11. Listado de tablas de un fichero mdb de la estación meteorológica

Proyecto de Fin de Grado

Numero de estación	Fecha	Parámetro	Cálculo	Valor
627	01/01/2010	1	1	12,53125
627	01/01/2010	1	4	12,53568
627	01/01/2010	1	5	12,52161
627	01/01/2010	2	1	84,66759
627	01/01/2010	4	2	0
627	01/01/2010	4	4	0
627	01/01/2010	5	1	9,894301E-07
627	01/01/2010	5	4	1,039676E-06
627	01/01/2010	6	1	2,720844
627	01/01/2010	6	4	2,762299
627	01/01/2010	6	5	2,689972
627	01/01/2010	7	1	3,364986
627	01/01/2010	7	4	6,183334
627	01/01/2010	7	6	1,17725
627	01/01/2010	8	1	253,0242
627	01/01/2010	8	4	250
627	01/01/2010	8	6	23,33092
627	01/01/2010	15	1	930,0002

Figura12. Valores y campos de la tabla de datos del fichero mdb

En nuestra aplicación nos hemos decidido por presentar valores de lluvia, radiación y velocidad y temperatura del viento, esto en las diferentes tablas se corresponde que los siguientes campos:

Lluvia: tabla datos, campo valor, con parámetro=4 (lluvia) y función=2(acumulado).

Radiación: tabla datos, campo valor, con parámetro=5 (radiación) y función=1(media).

Velocidad del viento: tabla datos, campo valor, parámetro=7 (vel viento) y función=1(media).

Temperatura viento: tabla datos, campo valor, parámetro=6 (temp viento) y función=1(media).

3. FUNCIONAMIENTO DEL PROYECTO ANTERIOR

3.1 SOFTWARE INSTALADO

Para poder controlar la instalación del generador fotovoltaico y de la estación meteorológica necesitamos, además de los equipos hardware anteriormente comentados (controlador SBC para instalación FV, sistema SAD para estación meteorológica y los Device Server para acceso TCP/IP) un ordenador con sistema operativo Windows con el siguiente software instalado:

Drivers de los Device Server.

Software Sunny Data Control para poder generar archivos xls de información de la instalación FV.

Sunny Data Agent. Consiste en unos applets java para poder acceder a los datos fotovoltaicos desde páginas web.

Geonica Suite 3000, que es el software de la estación meteorológica que genera gráficos con Datagraph-W3K y que genera bases de datos en formato Access con el software Teletrans-W3K

Apache Tomcat. Servidor web y contenedor de servlets y páginas jsp y html para permitir el acceso remoto a toda la información de las dos estaciones.

Es importante mencionar que todo este software tiene que estar asociado a servicios de sistema para que arranquen automáticamente al arrancar el ordenador. Además, al disponer de acceso TCP/IP a los sistemas de adquisición de datos, podemos tener todo el sistema de recogida de datos y monitorización web en un servidor remoto.

3.2 ARQUITECTURA

Nuestro sistema accesible desde Internet con un ordenador con Java instalado consta de un servidor Apache-Tomcat que almacena servlets, applets y páginas jsp que además interaccionan con el SD agent (applets del SDC de la estación fotovoltaica) y con Geonica Suite 3000 (software de la estación meteorológica)



Figura13. Arquitectura de acceso a los datos (Obtenida del PFC de JMEstepa)

3.3 AUDITORÍA

3.3.1 REVISIÓN DE SERVICIOS Y MANTENIMIENTO DE LOS DATOS DE LAS ESTACIONES

Para que la aplicación de monitorización funcione correctamente, en el apartado de servicios en herramientas administrativas, debemos asegurarnos de que el sistema operativo al arrancar ha iniciado los servicios de:

Apache-Tomcat (para poder acceder via web a la aplicación), SDAgent y SDControl (para monitorizar los datos fotovoltaicos) y W3k (para monitorizar los datos meteorológicos)

Monitorización de estaciones FV y meteo

Y en cuanto al mantenimiento de datos, para la estación fotovoltaica el SDC se encarga de generar ficheros xls que mostrarán las aplicaciones web, y para la estación meteorológica el programa Geonica se encarga de abrir la base de datos Access y guardar sus datos en ella.

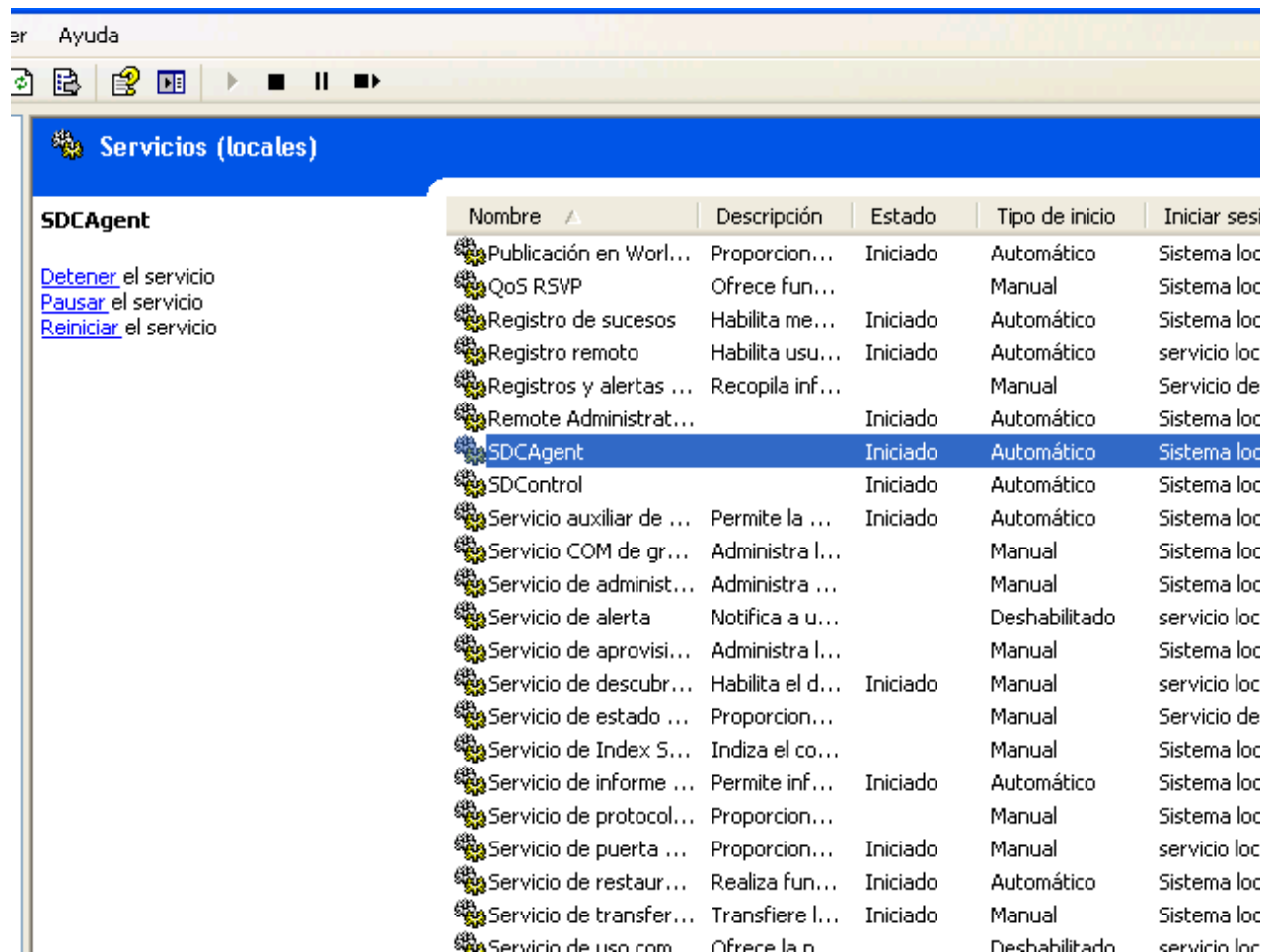


Figura14. Comprobación del estado de los servicios

Para ejecutar todo este software de los fabricantes como servicio del sistema operativo se utiliza el NTServiceInstaller, que describiremos detalladamente en el anexo A2.2

3.3.2 FUNCIONALIDADES DE LA APLICACIÓN

En la aplicación del proyecto anterior se diseñó un entorno web basado en monitorización de datos y gráficos, tanto de la estación fotovoltaica, como de la estación meteorológica.

En la pantalla inicial aparecen los datos actuales más importantes, y a la izquierda la posibilidad de visualizar todos los datos históricos, tanto en formato gráfico, como en diferentes formatos de datos. La gestión de estos datos almacenados en formato de hoja de cálculo xls por parte del SDC es el objetivo principal del proyecto

3.3.3 PROBLEMAS DE LA APLICACIÓN

La aplicación funciona de una manera muy correcta e intuitiva, aunque dando algunos errores, más por la pérdida de algunos históricos en los datos xls de las estaciones, que por errores de programación. A pesar de ello, al estar basada en applets de java (aunque con algunas funcionalidades basadas en tecnología jsp) tenemos el problema de que no funciona con sistemas modernos basados en Android e IOS de Apple. Por lo tanto, hay que reprogramar las funcionalidades de nuevo para solventar este problema.

Además de lo comentado anteriormente en este apartado, comento algunos problemas que vamos a intentar solucionar:

Estación fotovoltaica. En la visualización gráfica no queda gravado el parámetro elegido a visualizar pudiendo confundir al usuario que analiza gráficos.

Los valores en tiempo real tardan bastante, mejor visualizarlos al principio y no desde el menú.

En el apartado de datos históricos, es mejor poner datos entre dos valores. Además faltan períodos de tiempo de visualización, aunque la aplicación avisa y funciona correctamente. También hay gráficos horizontales que confunden un poco.

Estación meteorológica. La representación de datos y gráficos funciona bastante bien, aunque hay períodos no registrados. También es verdad que para la estación meteorológica no hay programadas tantas funcionalidades

4. NUEVO PROYECTO

4.1 BRAINBOXES ETHERNET TO SERIAL

Este equipo forma parte de uno de los cambios que se hace en la nueva versión del proyecto, técnicamente no ha sido un cambio complicado, pero ha aportado mucha flexibilidad a la hora de instalar el nuevo sistema de gestión del generador fotovoltaico.



Figura15. Brainboxes. Ethernet to serial

Contiene 2 puertos RS232 y un conector ethernet. Por una lado conectamos el controlador SBC y el conector ethernet lo conectamos a la red de cableado estructurado de la escuela. Previamente lo configuramos con una ip de la red.

Proyecto de Fin de Grado

En el ordenador con el SDC instalado instalamos los drivers y al añadir el puerto COM virtual y le informamos de la IP que contendrá un puerto COM. Finalmente en el ordenador configuramos el software SDC de la estación fotovoltaica para recibir comunicación a través de ese puerto COM virtual.

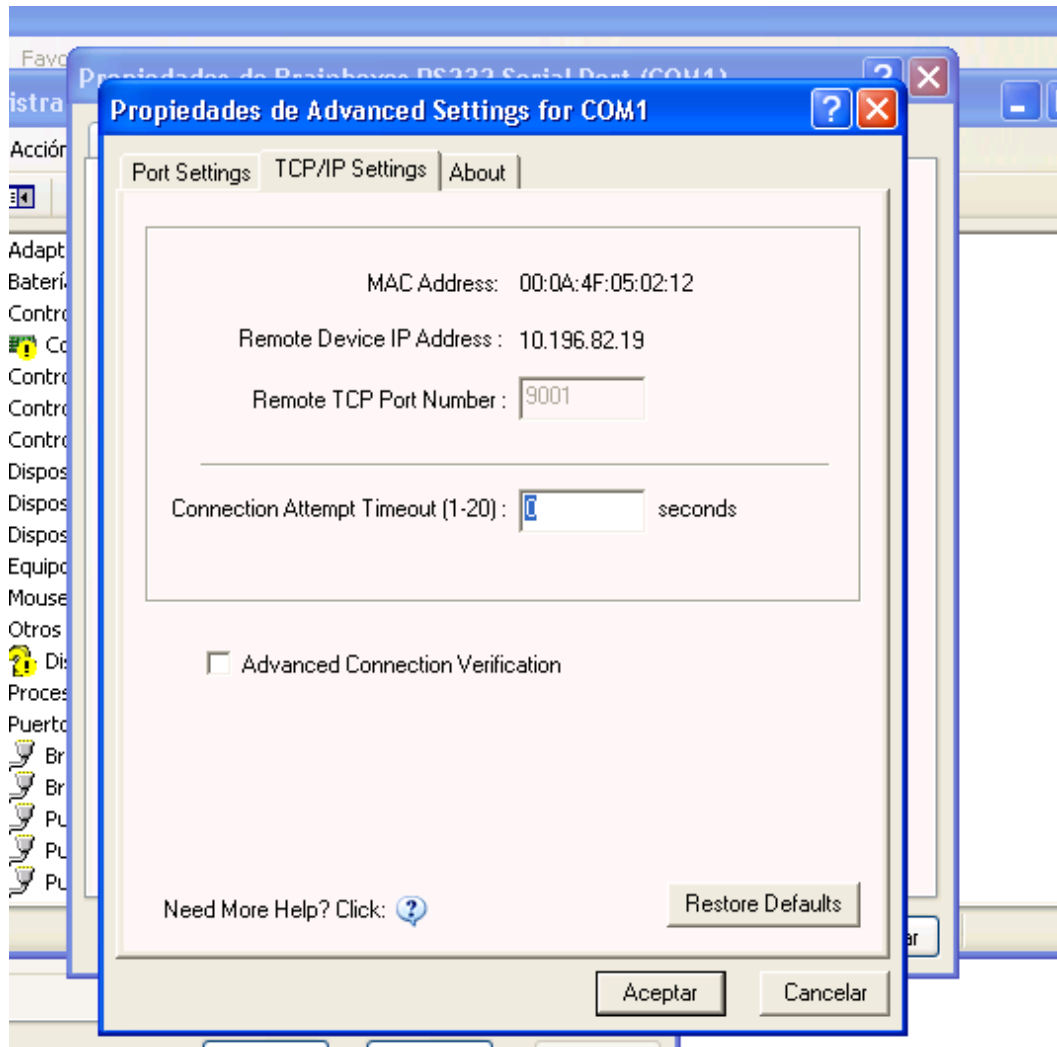


Figura16. Configuración del puerto COM virtual

4.2 POSIBILIDADES PARA EL NUEVO PROYECTO

Debido a que el proyecto está basado en la ejecución de código java para acceder a los datos, no es posible acceder a ellos a través de plataformas Android, ya que aunque las aplicaciones Android se programan con instrucciones Java, la máquina virtual es muy diferente a la de las plataformas para PC, con lo que se tendría que adaptar el software para visualizar los datos en terminales Android.

Creo que sería interesante intentar implementar un sistema para visualizar y monitorizar los datos fotovoltaicos y meteorológicos para dispositivos Android, tal vez no orientados a navegador, ya que en el caso de los smartphones las pantallas suelen ser pequeñas.

Otra segunda opción sería implementar la visualización mediante plataformas SNMP ya preparadas para acceder al agente SNMP mediante clientes ya programados. Por ejemplo tenemos Nagios y MRTG. Este protocolo está más pensado para monitorizar tráfico de redes, pero creo que se podría adaptar para leer datos de las estaciones.

En este caso la programación del proyecto consistiría en generar la información para ser guardada en variables MIB, accesibles a través del protocolo SNMP, que previamente leeríamos de los datos del SDC y de Geonica 3000.

Otras posibilidades, para lectura de datos fotovoltaicos sin utilizar el SBC y el ordenador Windows con SDC, consiste en leer directamente desde el inversor programando con la librería YASDI (Yet Another SMA Data Implementation).

YASDI se puede descargar de la web oficial de SMA, y consiste en una librería escrita en lenguaje C que permite la lectura de datos de dispositivos SMA.

Podemos obtener más información en:

<http://www.sma-iberica.com/es/productos/software/yasdi.html>

Una de las características importantes de YASDI, consiste en que no se puede leer del controlador SDC, con lo cual se leerán los valores directamente desde el inversor. Para ello, será necesario conectar un bus RS485 y con un conversor a RS232 conectarlo al ordenador que ejecuta el programa escrito en C y que utiliza las librerías. Estas librerías también tienen su versión Linux, con lo que se podría utilizar un microordenador, tipo Rasp Berry Pi, para acceder y guardar los datos

Por último, y opción finalmente elegida, vamos a intentar reprogramar el proyecto para hacerlo en versión html5. Así será compatible con todo tipo de sistemas. Esta opción la describimos a continuación.

4.3 NUEVA VERSIÓN EN HTML5

4.3.1 VENTAJAS

La principal ventaja del nuevo sistema es que es totalmente compatible con cualquier sistema operativo y cualquier arquitectura, ya que al estar basado en HTML5, al aplicación se carga con cualquier ordenador. En HTML5 básicamente nos dedicamos a representar gráficos y mediante aplicaciones JSP ejecutadas por el servidor Tomcat, procesamos estos datos para que sean representables en formato web

4.3.2 CAMBIOS

Los principales cambios que hacemos respecto del proyecto anterior son:

Eliminación de las aplicaciones que se ejecutan mediante applets, reprogramando los algoritmos en entorno servidor.

Representación gráfica de datos en HTML5

4.3.3 INTERFAZ DE USUARIO

Pantalla inicial

Al ejecutar la aplicación en:

http://meteo.ieec.uned.es:8086/PFGr_JoanCarbonell

Nos aparece la pantalla inicial con diferentes opciones, donde para cada estación (FV y meteo) tenemos la opción de visualizar gráficos y datos de diferentes períodos de tiempo.

Se ha apostado por un diseño sencillo y más compatible con dispositivos Android o Apple más preparados para webs más sencillas.



VIENTO:3.6M/S TEMP:28.7°C HUM:22% PRES:936.6MB RAD:442W/M2 LLUV:0.0MM
CIUDAD UNIVERSITARIA-MADRID (SPAIN) COORDENADAS:40° 27' 5.5"N 3° 44' 13.5"W
FECHA Y HORA LOCAL: 29/05/2015 15:55:18 GMT +1

Figura17. Pantalla inicial de la aplicación

Pantallas de introducción de datos

Desde la parte izquierda tenemos las opciones de visualizar datos o gráficos meteorológicos o fotovoltaicos, después de acceder a la opción deseada nos aparece un menú con las fechas de los datos que queremos visualizar:

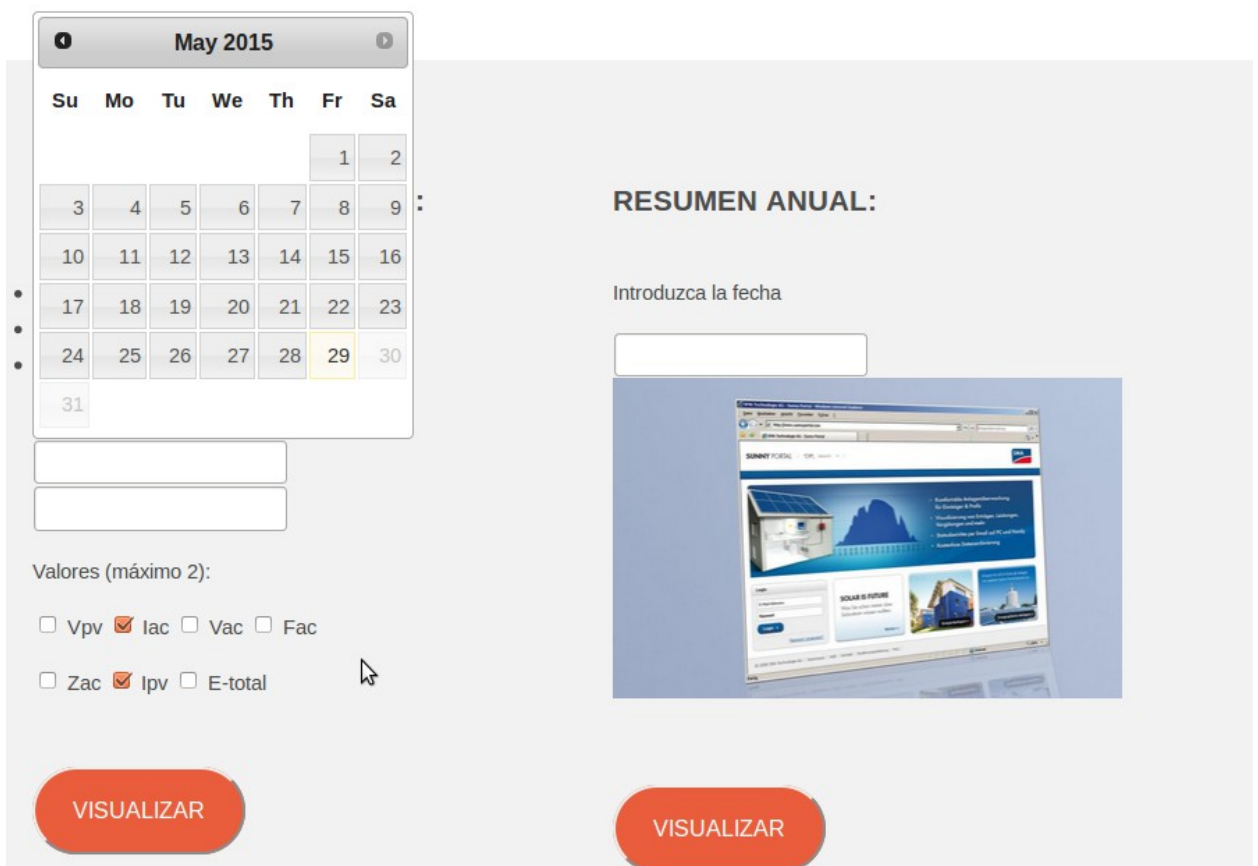
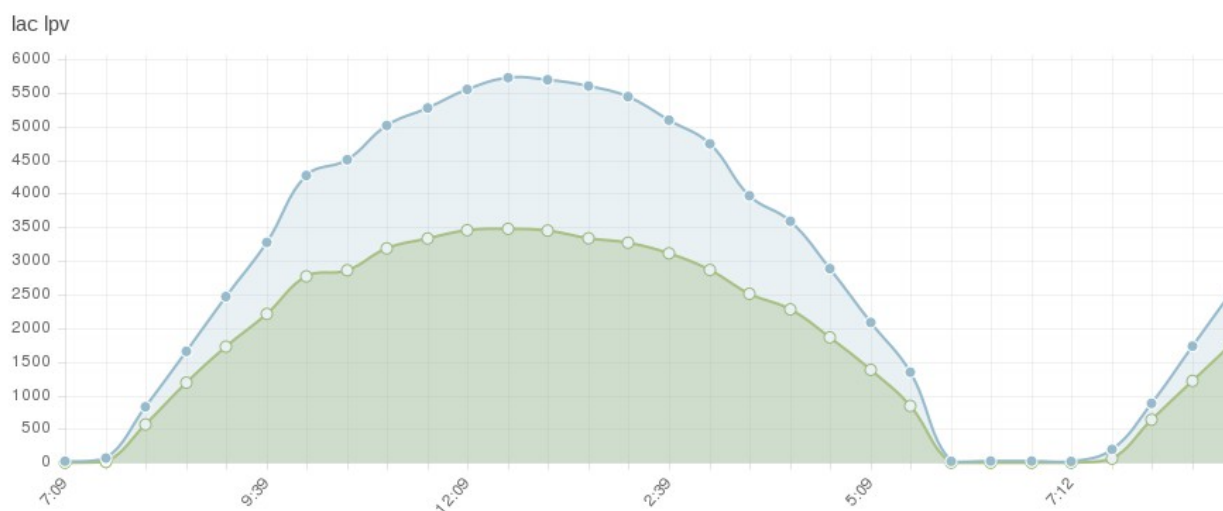


Figura18. Pantalla de introducción de datos FV

Gráficos FV

Si a la pantalla anterior le damos a la opción de visualizar, obtenemos gráficos del período elegido:



descarga de valores en formato xls entre 2015/04/01 y 2015/04/08

- [ficheroFV.xls](#)

Figura19. Pantalla de gráficos FV

Vemos en la figura anterior que tenemos la opción de visualizar hasta dos variables para poder comparar valores, además la aplicación nos crea un fichero con los datos elegidos entre los dos períodos de tiempo.

Datos FV

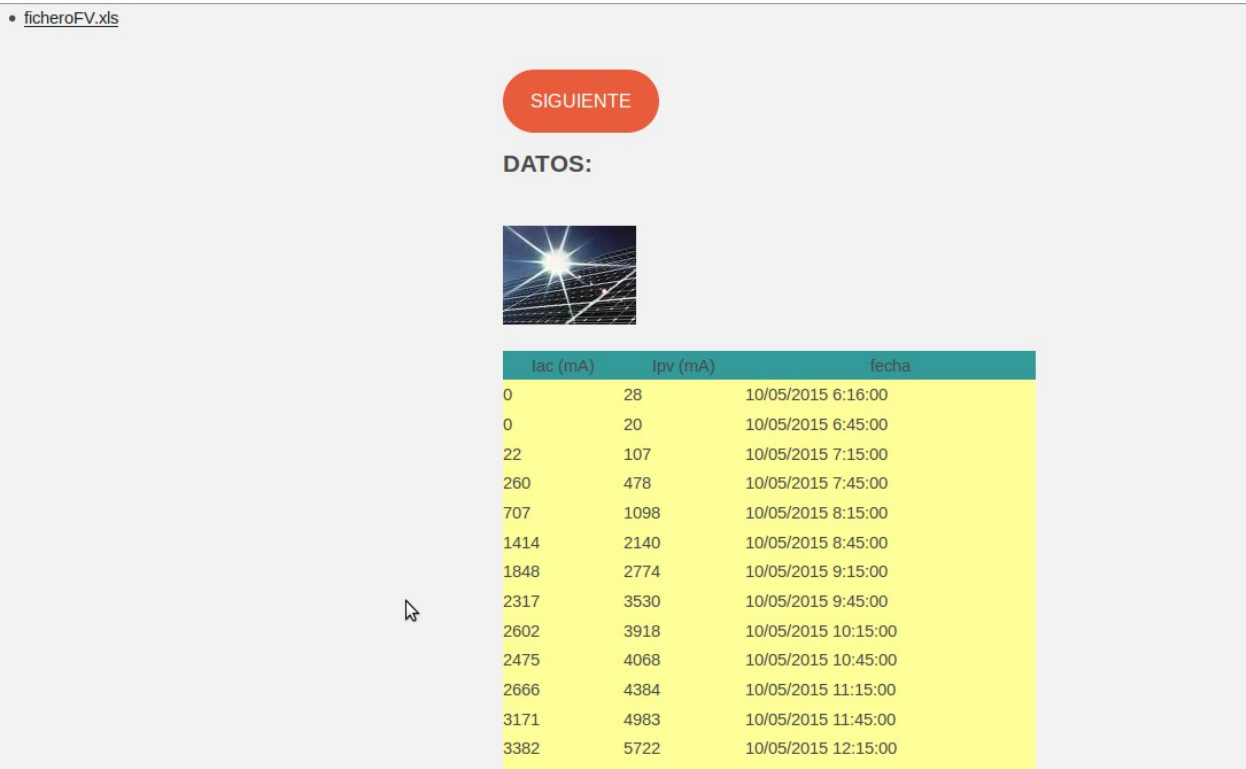
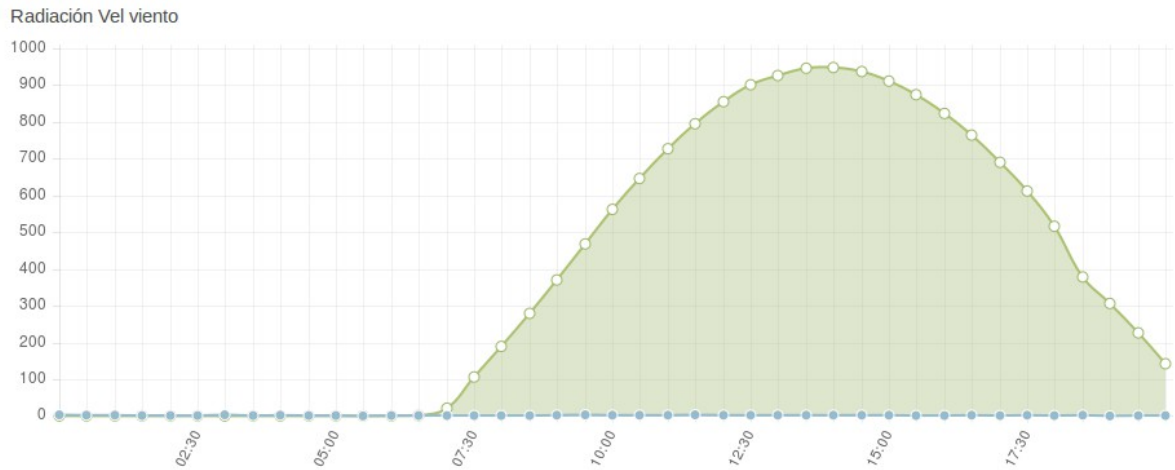


Figura20. Pantalla de datos FV

Gráficos Meteo



descarga de valores en formato xls entre 2015/05/17 y 2015/05/20

- [ficheroMeteo.xls](#)

SIGUIENTE

Figura21. Pantalla de gráficos meteo

Datos Meteo

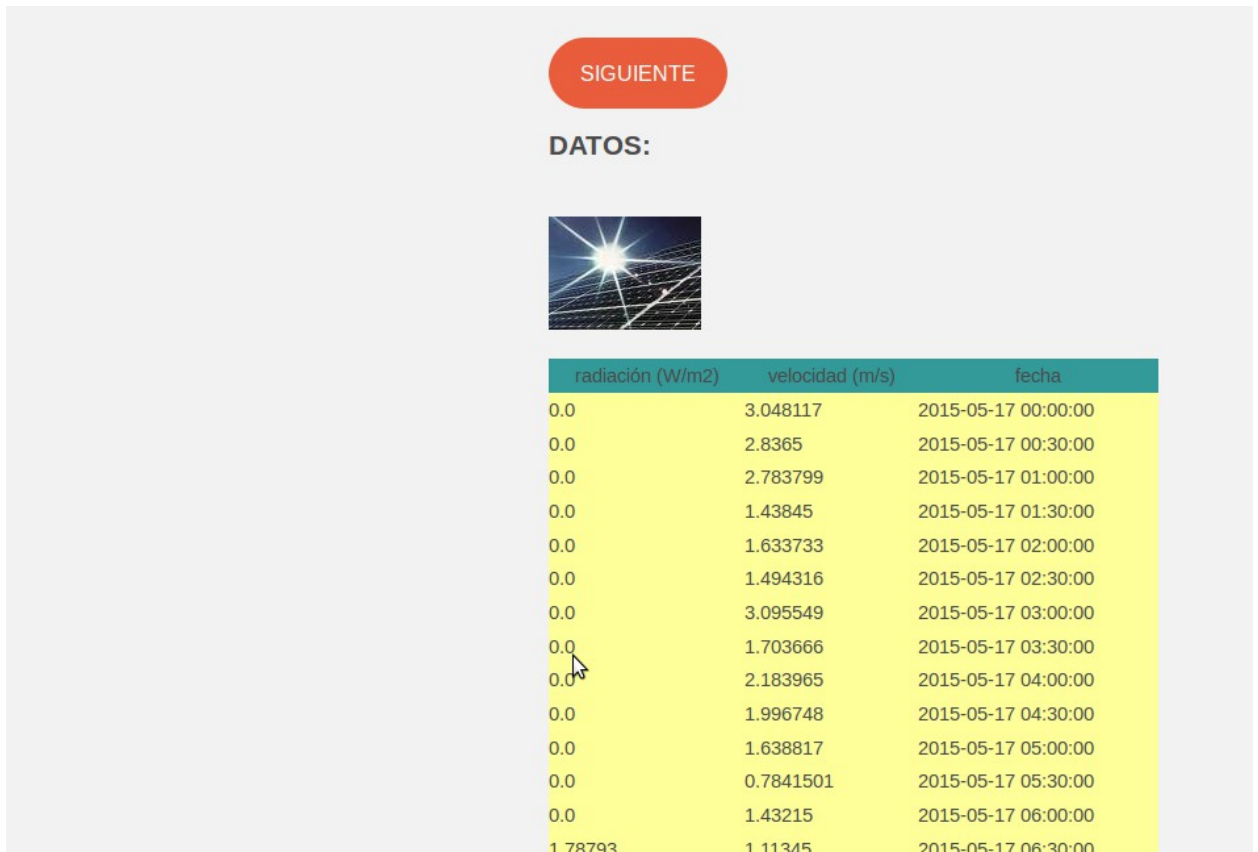


Figura22. Pantalla de datos meteo

4.3.4 RESUMEN ANUAL DE VALORES METEOROLÓGICOS Y FOTOVOLTAICOS

También se puede acceder a visualizar los valores de energía de la estación fotovoltaica durante un año entero. La opción está disponible desde la pantalla central. Para esta opción sólo aparecerán los valores de energía, en caso de querer analizar todos los demás parámetros debe realizarse mes a mes.

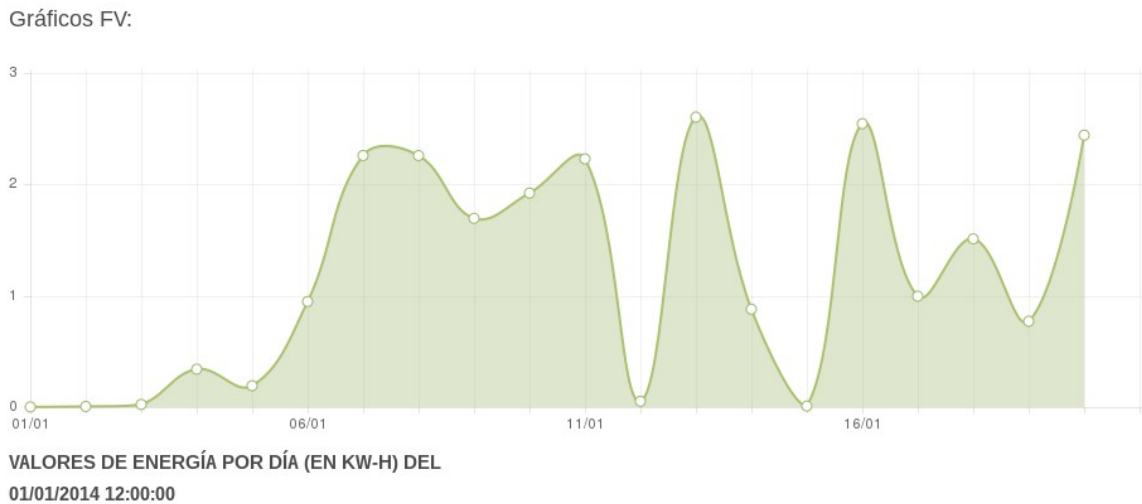


Figura23. Pantalla de resumen anual de energía FV

4.3.5 LIBRERÍA GRÁFICA JAVASCRIPT: CHART.JS

Para la visualización gráfica de los diferentes valores se ha optado por utilizar la librería gráfica Chart.js, basada en JavaScript. Es un proyecto Open Source con licencia cc-by (permite copia y adaptación para cualquier propósito, incluso comercial)

Es una librería muy fácil de utilizar y muy bien documentada en:
<http://www.chartjs.org/docs/#>

En esta página web podemos utilizar muchos ejemplos de cómo representar datos en Java Script. En concreto se ha utilizado el modelo Line Chart para representar gráficamente los datos.

Lo que se ha hecho en el proyecto actual es adaptar el código Java Script y ejecutarlos desde páginas jsp con código java. En uno de los anexos podemos encontrar el código fuente de toda la aplicación.

4.3.6 LIBRERÍA JXL Y ACCESO A FICHEROS MDB

Para representar los datos fotovoltaicos ha sido necesario leer datos de ficheros xls, por lo que ha sido necesario acceder a los ficheros xls creados por el SDC. Para ellos se ha utilizado la librería JexcelApi.

Los métodos más importante son:

```
Workbook archivo = Workbook.getWorkbook(new File(s));  
Sheet hoja = archivo.getSheet(int num_hoja);  
String dato=hoja.getCell(int fila,int columna).getContents();
```

Mientras que para acceder a los datos meteorológicos, guardados en bases de datos Access, se ha utilizado el driver: `Class.forName("sun.jdbc.odbc.JdbcOdbcDriver")`.

Los métodos más importantes utilizados son:

```
String s="jdbc:odbc:Driver={Microsoft Access Driver (*.mdb)};DBQ=" + fichero.mdb;  
Connection c = java.sql.DriverManager.getConnection(s, user, pwd);  
Statement orden = c.createStatement();  
ResultSet rs = orden.executeQuery(string orden_sql);  
String rs.getString(int num);
```

4.3.7 LECTURA DE VALORES INSTANTÁNEOS.

Para la lectura de los valores instantáneos, debido a que el software del SDC está basado en applets java, que además se ejecutan con una versión de java algo antigua, no es posible su lectura desde HTML5.

En el caso de la estación meteorológica estos valores se leen directamente desde un archivo en el servidor llamado DATOSMETEO_Inst.LOG, que está en el directorio de instalación de Geonica3000

4.3.8 PLANTILLA IRIDIUM PARA HTML5

Para el diseño de la aplicación se ha utilizado una plantilla HTML5 con algo de código JavaScript que hemos descargado de la web:

<http://templated.co/>

Entre varias plantillas hemos elegido la plantilla Iridium. Todas estas plantillas, como se indica en la web, tienen licencia cc by 3.0, con lo cual se puede utilizar siempre que mencionemos al autor.



Licencia Creative Commons 3.0

License



Figura24. Licencia de la plantilla utilizada.

4.3.9 DESCRIPCIÓN DEL TRABAJO REALIZADO

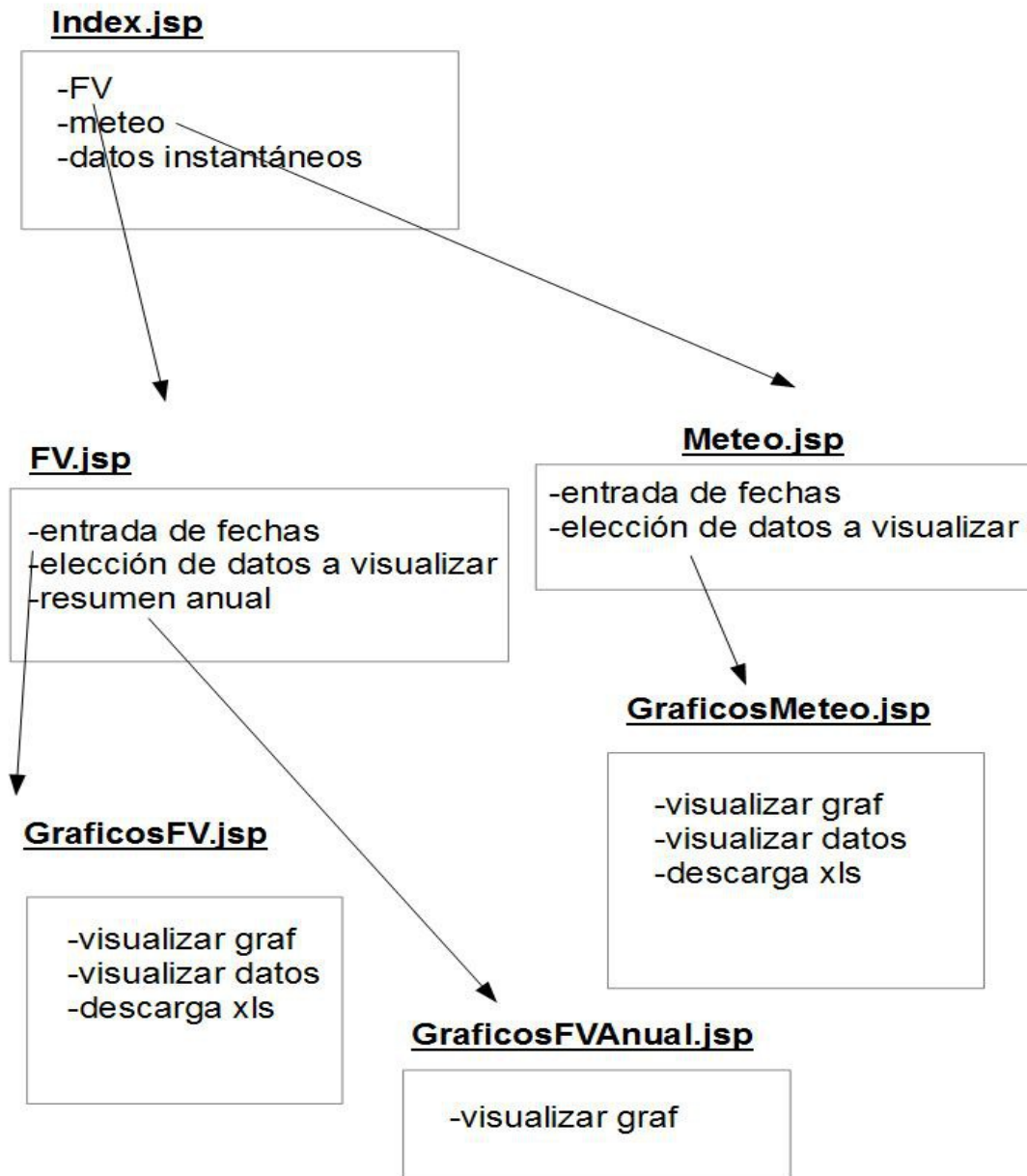


Figura25. Principales ficheros de código de la aplicación.

Proyecto de Fin de Grado

5. CONCLUSIONES Y POSIBLES MEJORAS

Para futuras aplicaciones se podría considerar la posibilidad de leer todos los datos a través de la librería YASDI para instalaciones FV sencillas, para así evitar tener un SBC y un ordenador dedicado a la lectura de datos con el SDC. Existen en el mercado miniordenadores basados en Linux para ejecutar el programa de lectura de datos directamente desde el inversor.

Estos miniordenadores tienen la capacidad de instalar servidores tipo tomcat para visualizar los datos mediante servicios web, o incluso hay versiones de Nagios para presentarlos por SNMP.

La gestión de los datos y su conversión a ficheros xls o bases de datos SQL sería tal vez algo complicada, pero posible, gracias a los algoritmos de lectura y escritura de estos ficheros basados en Java y su posibilidad de ejecutarlos desde aplicaciones JSP. Además estos minicomputadores se pueden complementar con discos duros o memoria USB para guardar todos los datos. La principal ventaja es el poco espacio y consumo de estos equipos para monitorizar los datos a través de Internet, incluso se pueden complementar con Virtual Private Servers (VPS) en cloud computing para minimizar la instalación del usuario.

Otra posibilidad para el futuro podría ser realizar un análisis de la eficiencia de la estación FV a partir de los datos de radiación y temperatura de la estación meteorológica.

Y finalmente, como conclusiones podríamos comentar lo siguiente:

- Se puede integrar la gestión de las estaciones renovables con sistemas de monitorización remota.
- Posible integración con instalaciones domóticas.
- Fácil implementación con protocolos IP: HTML5, SNMP...
- Grandes posibilidades de utilización de tecnologías de software libre con licencias Creative Commons.

Proyecto de Fin de Grado

6. BIBLIOGRAFÍA

Proyectos anteriores

http://meteo.ieec.uned.es:8086/PFC_JMEstepa/

http://meteo.ieec.uned.es/www_Usumeteo4/homepage.html

Libros

B.Eckel, “Thinking in Java”, Prentice Hall 2002.

M.L.Liu. “Computación distribuida. Fundamentos y Aplicaciones”, Addison Wesley 2004.

J.L.Hevia y A.Esteban, “Tecnologías de Servidor con J2EE”, Eidos 2005.

S.Allamaraju, C.Beust y otros, “Programación Java Server con J2EE”, Addison Wesley 2001.

S.Bodoff, E.Armstrong, Jennifer Ball, Debbie Carson, Ian Evans y Dale Green, “J2EETutorial”, Addison Wesley 2004.

Manuales

SMA Regelsysteme, “Sunny Boy. Descripción Técnica”.

SMA Regelsysteme, “Sunny Boy Control and Control Plus. User Manual”.

SMA Regelsysteme, “Sunny Data Control. Operating Instructions”.

Geonica, “Manual Meteodata Rev.03”.

Geonica, “Manual Teletrans W3K”.

Recursos de Internet

<http://www.chartjs.org/>

Febrero 2015. Librerías de visualización de datos en JavaScript

<http://www.sma-iberica.com/es.html>

Diciembre 2014. Documentación de manuales SMA

<http://www.desarrolloweb.com/manuales/>

Septiembre 2014. Manuales de programación JSP y servlets

<http://templated.co/>

Abril 2015. Web de plantillas HTML5 con licencia cc 3.0

Proyecto de Fin de Grado

ANEXOS

A1. CONTENIDO DEL CD-ROM

En el CD del proyecto, adjuntamos lo siguientes archivos:

- Memoria del proyecto (en formato pdf)
- Presentación del proyecto (en formato pdf)
- Manuales de las estaciones FV y meteorológica.
- Carpeta \PFGr_JoanCarbonell con todos los ficheros de código de proyecto, que si se copian en el directorio webapps de apache-tomcat ya permite la ejecución de la aplicación. Se incluye también el fichero PFGr_JoanCarbonell.war que si se copia, el mismo servidor Tomcat se encarga de descomprimir.
- Plantilla Iridium.
- Fichero PFGr_JoanCarbonell.conf con los directorios del software FV y meteo.
- Fichero jxl.jar para leer y escribir ficheros xls en Java.
- Fichero Chart.js-master.zip para los gráficos en JavaScript.

Proyecto de Fin de Grado

A2. INSTALACIÓN Y MANTENIMIENTO

A2.1 PAQUETES A INSTALAR Y CONFIGURACIONES

-Drivers de los Device Server.

-Software Sunny Data Control para poder generar archivos xls de información de la instalación FV.

-Geonica Suite 3000, que es el software de la estación meteorológica que genera gráficos con Datagraph-W3K y que genera bases de datos en formato Access con el software Teletrans-W3K

-Apache Tomcat. Servidor web y contenedor de servlets y páginas jsp y html para permitir el acceso remoto a toda la información de las dos estaciones.

-Configuración del software SDC y Geonica3000 para que arranquen como servicio.

-Instalación del proyecto que contiene la aplicación con los ficheros html y jsp para poder visualizar los datos desde Internet. Este fichero debería estar en webapps dentro del directorio del servidor Tomcat.

A2.2 INSTALACIÓN DE SERVICIOS

Para poder ejecutar como servicios los distintos programas de los fabricantes de las estaciones procederemos a efectuar lo siguiente:

1.- Descargar las herramientas del kit de recursos de Microsoft Windows. Este programa una vez descomprimido proporciona los programas **instsrv.exe** y **srvany.exe**

2.- Crear desde la línea de comandos el servicio tecleando la instrucción:
instsrv <nombre_servicio> <archivo.exe> srvany.exe

Por ejemplo, si tratáramos de ejecutar como servicio TeleTrans-w3k.exe (el programa que pide datos a la estación meteorológica) que está instalado en c:\dirw3k, la instrucción sería:
Instsrv W3K c:\dirw3k\TeleTrans-w3k.exe srvany.exe

3.- Añadir parámetros al servicio (CUIDADO: ESTE PASO PUEDE SER PELIGROSO PARA EL EQUIPO. Conviene realizar una copia de seguridad del registro y cuidar los parámetros que se modifican)

Ejecutar el comando regedit

HKEY_LOCAL_MACHINE-SYSTEM-CURRENTCONTROLSET-SERVICES

Desplazarse hasta la clave del servicio (ej:W3K) y hacer los siguientes ajustes:

Modificar la clave ImagePath para que su contenido sea el camino absoluto para ejecutar srvany.exe

Añadir la clave (carpeta) Parameters

Añadir un valor de cadena en la clave Parameters llamado Application con el valor del camino completo hasta el programa correspondiente.

En nuestro ejemplo Application=c:\dirw3k\TeleTrans-w3k.exe

Añadir un valor de cadena en la clave Parameters llamado AppDirectory con la carpeta donde se debe ejecutar el programa. En nuestro ejemplo:

AppDirectory=[c:/dirw3k](#)

4.- Configurar el servicio.

Accedemos al servicio mediante Inicio-Panel de Control-Herramientas Administrativas-Servicios.

Marcamos el servicio (en el ejemplo W3K) y con el botón derecho accedemos a sus propiedades. En la pestaña de “Iniciar sesión” marcamos la casilla “Permitir que el servicio interactue con el escritorio”.

Arrancamos el servicio y se mostrará el programa como si se ejecutara de forma normal.

Monitorización de estaciones FV y meteo

Configurarlo para que se ejecute tal como deseamos

Parar el servicio

Acceder a sus propiedades y desmarcar la casilla de “Permitir que el servicio interactue con el escritorio”.

Indicar que el servicio se ejecute de forma automática

A2.3 COPIA DE SEGURIDAD DE LOS DATOS

Para poder garantizar el correcto funcionamiento futuro de la aplicación sería necesario efectuar copia de seguridad de los datos de las estaciones de vez en cuando.

En concreto es necesario hacer copia de los ficheros xls en el directorio:

C:/Archivos de programa/SMA Regelsysteme/Sunny DataControl/Plants/DIEEC/
SBC106367711/

Y de los ficheros mdb en:

C:/Archivos de programa/Geonica/Geonica Suite 3K/DataBase/

A3. CÓDIGO FUENTE DE LA APLICACIÓN

Index.jsp

Fichero inicial de la aplicación con los enlaces a las diferentes funcionalidades

```
<!DOCTYPE HTML>
<!--
    Iridium by TEMPLATED
    templated.co @templatedco
    Released for free under the Creative Commons Attribution 3.0 license (templated.co/license)
-->
<html>
    <head>
        <title>PFG Joan Carbonell</title>
        <meta http-equiv="content-type" content="text/html; charset=utf-8" />
        <meta name="description" content="" />
        <meta name="keywords" content="" />
        <link href='http://fonts.googleapis.com/css?family=Arimo:400,700' rel='stylesheet'
type='text/css'>
        <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
        <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
        <script src="js/skel.min.js"></script>
        <script src="js/skel-panels.min.js"></script>
        <script src="js/init.js"></script>
        <script language="JavaScript" type="text/JavaScript">
<!--
function MM_reloadPage(init) { //reloads the window if Nav4 resized
    if (init==true) with (navigator) { if ((appName=="Netscape")&&(parseInt(appVersion)==4)) {
        document.MM_pgW=innerWidth; document.MM_pgH=innerHeight; onresize=MM_reloadPage; }}
    else if (innerWidth!=document.MM_pgW || innerHeight!=document.MM_pgH) location.reload();
    }
MM_reloadPage(true);
//-->
    </script>
    <style type="text/css">
<!--
.Estilo1 {font-size: 14px}
.Estilo2 {font-size: 18px}
-->
    </style>
    <noscript>
        <link rel="stylesheet" href="css/skel-noscript.css" />
        <link rel="stylesheet" href="css/style.css" />
        <link rel="stylesheet" href="css/style-desktop.css" />
    </noscript>
    <!--[if lte IE 8]><link rel="stylesheet" href="css/ie/v8.css" /><![endif]-->
```

Proyecto de Fin de Grado

```
<!--[if lte IE 9]><link rel="stylesheet" href="css/ie/v9.css" /><![endif]-->
</head>
<body class="homepage">

    <!-- Header -->
    <div id="header">
        <div class="container">

            <!-- Logo -->
            <div id="logo">
                <h1><a href="#" class="Estilo2">Est FV y Meteo ETSII
</a></h1>
                <p class="Estilo2">&nbsp;</p>
            </div>

            <nav id="nav">
                <ul>
                    <li class="active"><a href="index.jsp">Inicio</a></li>
                    <li><a href="/FV.jsp">FV</a></li>
                    <li><a href="/meteo.jsp">Meteo</a></li>
                    <li><a href="/licencia.jsp">Iridium template</a></li>
                </ul>
            </nav></div>
        </div>

        <div id="main"><div class="container"><div class="row">
            <div id="content" class="8u skel-cell-
important"><section><header></header></section></div>

            <%@page import="java.io.*"%>
            <%
//buscamos el directorio meteorológico
FileReader fr=new FileReader("C:/PFGGr_JoanCarbonell.conf");
BufferedReader br= new BufferedReader(fr);
String lin="";
String directorio="";
lin=br.readLine();
while (lin!=null)
{
    String datos[]=lin.split("=");
    if (datos[0].equals("Directorio_Meteorologico"))
    {
        directorio=datos[1];
        break;
    }
    lin=br.readLine();
}
FileReader fr2=new FileReader(directorio + "DATOSMETEO_Inst.LOG");
BufferedReader br2= new BufferedReader(fr2);
```

Monitorización de estaciones FV y meteo

```
lin="";
String ult_lin="";
lin=br2.readLine();
while (lin!=null)
{
    ult_lin=lin;
    lin=br2.readLine();
}
String datos[]=ult_lin.split(";");
%>
                                <div id="sidebar" class="4u"><section><header>
                                    <div id="Layer1" style="position:absolute; width:859px;
height:101px; z-index:1; left: 185px; top: 38px;">
                                        <h2 align="justify" class="Estilo1">
                                            <%
out.print(" viento:" + datos[1] + "m/s temp:" + datos[3] + "°C hum:" + datos[4]);
out.println("% pres:" + datos[5] + "mb rad:" + datos[6] + "w/m2 lluv:" + datos[7] + "mm");
out.print("Ciudad Universitaria-Madrid (Spain) ");
out.println("Coordenadas:40° 27' 5.5"N 3° 44' 13.5"W");
out.println("fecha y hora local: " + datos[0] + " GMT +1");
                                            %>
                                        </h2>
                                            <p align="justify" class="Estilo1">&nbsp;</p>
                                        </div>
                                    </header></section></div>
                                </div>
                            </div>
                        </div>
                    <div id="featured"><div class="container"><div class="row"><div class="4u"></div>
                        </div>
                    </div>
                <div id="footer">
                    <div class="container">
                        <div class="row">
                            <div class="4u">
                                <section>
                                    <h2>&nbsp;</h2>
                                </section></div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </body>
    </html>
```

FV.jsp

Recoge los rangos de fechas y los valores a representar de la estación fotovoltaica.

```
<!DOCTYPE HTML>
<!--
    Iridium by TEMPLATED
    templated.co @templatedco
    Released for free under the Creative Commons Attribution 3.0 license (templated.co/license)
-->
<html>
  <head>
    <title>PFG Joan Carbonell</title>
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
    <link href='http://fonts.googleapis.com/css?family=Arimo:400,700' rel='stylesheet'
type='text/css'>
    <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
    <script src="js/skel.min.js"></script>
    <script src="js/skel-panels.min.js"></script>
    <script src="js/init.js"></script>
    <noscript>
      <link rel="stylesheet" href="css/skel-noscript.css" />
      <link rel="stylesheet" href="css/style.css" />
      <link rel="stylesheet" href="css/style-desktop.css" />
    </noscript>
    <!--[if lte IE 8]><link rel="stylesheet" href="css/ie/v8.css" /><![endif]-->
    <!--[if lte IE 9]><link rel="stylesheet" href="css/ie/v9.css" /><![endif]-->
  </head>
  <body>

    <!-- Header -->
    <div id="header">
      <div class="container">

        <!-- Logo -->
        <div id="logo">
          <h1><a href="#">Est FV y meteo</a></h1>
        </div>

        <!-- Nav -->
        <nav id="nav">
          <ul>
            <li><a href="index.jsp">Inicio</a></li>
            <li class="active"><a href="FV.jsp">FV</a></li>
          </ul>
        </nav>
      </div>
    </div>
  </body>
</html>
```

Monitorización de estaciones FV y meteo

```
<li><a href="meteo.jsp">meteo</a></li>
</ul>
</nav>
</div>
</div>

<!-- Main -->
<div id="main">
  <div class="container">
    <div class="row">
      <!-- Sidebar -->
      <div id="sidebar" class="4u">
        <header>
          <h2>Est FV en ETSII-UNED</h2>
        </header>
        <p class="text">Visualizaci&oacute;n de datos y gr&aacute;ficos FV.</p>
      </div>
    </div>
  </div>
  <!-- Content -->
</div>
</div>

<!-- Footer -->
<div id="featured">
  <div class="container">
    <div class="row">
      <div class="4u">
        <h2>Visualizaci&oacute;n de datos: </h2>

<%@page import="java.io.*"%>
<% int contador=0; %>

<form name="form2" method="post" onSubmit="" target="_top">
<li class="style">valores/p&aacute;gina
  <input name="valor_pag" type="radio" value="20" checked>
  20
  <input name="valor_pag" type="radio" value="30" >
  30
  <input name="valor_pag" type="radio" value="40" >
  40</li>
<li class="style">valores cada
  <input name="tiempo" type="radio" value="3 minutos" checked>
  3 min
  <input name="tiempo" type="radio" value="6 minutos" >
```

```
6 min </li>
<li class="style">
  <input name="tiempo" type="radio" value="9 minutos" >
  9 min
  <input name="tiempo" type="radio" value="30 minutos" >
  30 min
  <input name="tiempo" type="radio" value="60 minutos" >
  60 min</li>
<p class="style">Rango de fechas:</p>
<span class="style">
<%
String ua = request.getHeader( "User-Agent" );
%>
</span>
<link rel="stylesheet" href="//code.jquery.com/ui/1.11.3/themes/smoothness/jquery-ui.css">
<span class="style">
<script src="//code.jquery.com/jquery-1.10.2.js"></script>
<script src="//code.jquery.com/ui/1.11.3/jquery-ui.js"></script>
</span>
<link rel="stylesheet" href="/resources/demos/style.css">
<span class="style">
<script>
$(function() {
$("#calendar1").datepicker({
  maxDate: "+0d",
  onClose: function(selectedDate)
  {
    $(this).datepicker("option","dateFormat","yy/mm/dd");
    $(this).datepicker("option","maxDate",-1d);
    $("#calendar2").datepicker("option","minDate",selectedDate);
    var mes=$(this).datepicker('getDate').getMonth();
    var any=$(this).datepicker('getDate').getFullYear();
    var dia=0;
    if ((mes==0)||(mes==2)||(mes==4)||(mes==6)||(mes==7)||(mes==9)||(mes==11))
      dia=31;
    else if (mes==1) dia=28;
    else dia=30;
    var fecha = new Date();
    var fecha_act = new Date();
    var dia_act=fecha_act.getDate();
    var mes_act=fecha_act.getMonth();
    var any_act=fecha_act.getFullYear();
    if ((mes==mes_act)&&(any==any_act)) dia=dia_act;
    //var any_act=fecha_act.getFullYear();
    fecha.setFullYear(any,mes,dia); //aÃ±o, mes y dia, mes: 0..11
    $("#calendar2").datepicker("option","maxDate",fecha);
    $("#calendar2").datepicker("option","dateFormat","yy/mm/dd");
  }
})

```

Monitorización de estaciones FV y meteo

```
});
$("#calendar2").datepicker();
$("#calendar3").datepicker(
{
    dateFormat: "yy/mm/dd",
    changeYear: "true",
    maxDate: "-1y",
    minDate: "2006/01/01"
}
);
});

</script>
<%if (ua.matches(".*Android.*"))
{
    out.print("<input type='text' id='calendar1' name='cal1' onfocus='blur()'>");
    out.print("<input type='text' id='calendar2' name='cal2' onfocus='blur()'></p>");
}
else
{
    out.println("<input type='text' id='calendar1' name='cal1' onfocus='blur()'>");
    out.println("<input type='text' id='calendar2' name='cal2' onfocus='blur()'></p>");
}
%>

<script>
var num=0;
function cuenta(elem)
{
    if (elem.checked)
    {
        if (num < 2)
            num=num+1;
        else if (num==2)
            elem.checked=false;
    }
    else
        num=num-1;
}
</script>
<p>Valores (mximo 2): </p>
<p><span class="style">
    <input name="col1" type="checkbox" value="Vpv" onclick="cuenta(this)" />
    Vpv
    <input name="col2" type="checkbox" value="Iac" onclick="cuenta(this)" />
    Iac
    <input name="col3" type="checkbox" value="Vac" onclick="cuenta(this)" />
    Vac
    <input name="col4" type="checkbox" value="Fac" onclick="cuenta(this)" />
```

Proyecto de Fin de Grado

```
Fac </span></p>
<p><span class="style">
  <input name="col5" type="checkbox" value="Zac" onclick="cuenta(this)" />
  Zac</span><span class="style">
  <input name="col6" type="checkbox" value="Ipv" onclick="cuenta(this)" />
  Ipv
  <input name="col7" type="checkbox" value="E-total" onclick="cuenta(this)" />
  E-total </span></p>
<p>
  <input name="Submit" type="submit" onclick = "this.form.action='./graficosFV.jsp?contador=<
%out.print(contador);%>" class="button" value="Visualizar">
  </p>
</form>

</div>
<div class="4u">
  <h2>Resumen anual:</h2>
  <p>Introduzca la fecha </p>
  <form name="form3" method="post" onSubmit=""
target="_top">
  <%
  if (ua.matches(".*Android.*"))
  out.print("<input type='text' id='calendar3' name='cal3'
onfocus='\"blur()\">");
  else
  out.print("<input type='text' id='calendar3' name='cal3'
onfocus='\"blur()\">");
  %>
  <div align="left">
    <p><a href="#" class="image full"></a>
    </p>
  </div>
  <input name="Submit" type="submit" onclick =
"this.form.action='./graficosFVAnual.jsp?contador=<%out.print(contador);%>" class="button"
value="Visualizar">
  </form>
</div>
<div class="4u"></div>
</div>
<div id="copyright"></div>
</body>
</html>
```


graficosFV.jsp

Representa gráficamente los valores elegidos en FV.jsp y también presenta estos valores en una tabla de datos. Todo esto se consigue leyendo los valores del fichero xls que nos crea el SDC.

```
<!DOCTYPE HTML>
<!--
    Iridium by TEMPLATED
    templated.co @templatedco
    Released for free under the Creative Commons Attribution 3.0 license (templated.co/license)
-->
<html>
    <head>
        <title>PFG Joan Carbonell</title>
        <meta http-equiv="content-type" content="text/html; charset=utf-8" />
        <meta name="description" content="" />
        <meta name="keywords" content="" />
        <link href='http://fonts.googleapis.com/css?family=Arimo:400,700' rel='stylesheet'
type='text/css'>
        <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
        <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
        <script src="js/skel.min.js"></script>
        <script src="js/skel-panels.min.js"></script>
        <script src="js/init.js"></script>
        <style type="text/css">
<!--
.Estilo1 {font-size: 18px}
-->
    </style>
    <noscript>
        <link rel="stylesheet" href="css/skel-noscript.css" />
        <link rel="stylesheet" href="css/style.css" />
        <link rel="stylesheet" href="css/style-desktop.css" />
</noscript>
        <!--[if lte IE 8]><link rel="stylesheet" href="css/ie/v8.css" /><![endif]-->
        <!--[if lte IE 9]><link rel="stylesheet" href="css/ie/v9.css" /><![endif]-->
    </head>
    <body>

        <!-- Header -->
        <div id="header">
            <div class="container">

                <!-- Logo -->
                <div id="logo">
                    <h1><a href="#">Est FV y meteo</a></h1>
                </div>
            </div>
        </div>
    </body>
</html>
```

Proyecto de Fin de Grado

```
<!-- Nav -->
<nav id="nav">
    <ul>
        <li><a href="index.jsp">Inicio</a></li>
        <li class="active"><a href="FV.jsp">FV</a></li>
        <li><a href="meteo.jsp">meteo</a></li>
    </ul>
</nav>
</div>
</div>
<span class="Estilo1">
<!-- Main -->
</span>
<div id="main">
    <div class="container">
        <div class="row">
            <span class="Estilo1">
                <!-- Sidebar -->
            </span>
            <div id="sidebar" class="4u"><span
class="Estilo1"><section><header></header></section></span><section>
                <p class="text Estilo1">Gr&aacute;ficos FV: </p>
<%@ page import="java.io.*" %>
<%@ page import="java.util.*" %>
<%@ page import="jxl.*" %>
<%@ page import="jxl.write.*" %>
<%@ page import="java.string.*" %>
<%
try
    {
        // leemos la fecha
        String cal1, cal2, contador,valor_pag,tiempo;
        String op1,op2,op3,op4,op5,op6,op7;
        String fichero="";
        cal1=(String)request.getParameter("cal1");
        cal2=(String)request.getParameter("cal2");
        op1=(String)request.getParameter("col1");
        op2=(String)request.getParameter("col2");
        op3=(String)request.getParameter("col3");
        op4=(String)request.getParameter("col4");
        op5=(String)request.getParameter("col5");
        op6=(String)request.getParameter("col6");
        op7=(String)request.getParameter("col7");

        if (op1==null) op1="";
        else out.println(op1 + " ");
```

Monitorización de estaciones FV y meteo

```
if (op2==null) op2="";
else out.println(op2 + " ");
if (op3==null) op3="";
else out.println(op3 + " ");
if (op4==null) op4="";
else out.println(op4 + " ");
if (op5==null) op5="";
else out.println(op5 + " ");
if (op6==null) op6="";
else out.println(op6 + " ");
if (op7==null) op7="";
else out.println(op7 + " ");

int xls=0;
contador=(String)request.getParameter("contador");
if (contador.equals("0")) xls=1;
valor_pag=(String)request.getParameter("valor_pag");
tiempo=(String)request.getParameter("tiempo");

int valor_pagina=0, incremento=0;
valor_pagina=Integer.parseInt(valor_pag);
if (tiempo.equals("3 minutos")) incremento=1;
else if (tiempo.equals("6 minutos")) incremento=2;
else if (tiempo.equals("9 minutos")) incremento=3;
else if (tiempo.equals("30 minutos")) incremento=10;
else if (tiempo.equals("60 minutos")) incremento=20;
fichero="SDM_" + cal1.substring(2,4);
fichero=fichero + cal1.substring(5,7) + ".xls";
// ya tenemos año y mes
// Buscamos el directorio en C:/PFGr_JoanCarbonell.conf

FileReader fr=new FileReader("C:/PFGr_JoanCarbonell.conf");
BufferedReader br= new BufferedReader(fr);
String lin="";
String directorio="";
lin=br.readLine();
while (lin!=null)
{
    String datos[]=lin.split("=");
    if (datos[0].equals("Directorio_Fotovoltaico"))
    {
        directorio=datos[1];
        break;
    }
    lin=br.readLine();
}
String s=directorio + fichero;
Workbook arxiu = Workbook.getWorkbook(new File(s));
Sheet full = arxiu.getSheet(0);//primera hoja
```

```
int numColumnes = full.getColumns();
int numFiles = full.getRows();
String data,data_ant;
int col1=0,col2=0;
// leemos los valores elegidos
if (op1.equals("Vpv"))
{
    col1=2;
    if (op2.equals("Iac")) col2=4;
    else if (op3.equals("Vac")) col2=5;
    else if (op4.equals("Fac")) col2=6;
    else if (op5.equals("Zac")) col2=8;
    else if (op6.equals("Ipv")) col2=10;
    else if (op7.equals("E-total"))col2=11;
}
else if (op2.equals("Iac"))
{
    col1=4;
    if (op3.equals("Vac")) col2=5;
    else if (op4.equals("Fac")) col2=6;
    else if (op5.equals("Zac")) col2=8;
    else if (op6.equals("Ipv")) col2=10;
    else if (op7.equals("E-total"))col2=11;
}
else if (op3.equals("Vac"))
{
    col1=5;
    if (op4.equals("Fac")) col2=6;
    else if (op5.equals("Zac")) col2=8;
    else if (op6.equals("Ipv")) col2=10;
    else if (op7.equals("E-total"))col2=11;
}
else if (op4.equals("Fac"))
{
    col1=6;
    if (op5.equals("Zac")) col2=8;
    else if (op6.equals("Ipv")) col2=10;
    else if (op7.equals("E-total")) col2=11;
}
else if (op5.equals("Zac"))
{
    col1=8;
    if (op6.equals("Ipv")) col2=10;
    else if (op7.equals("E-total")) col2=11;
}
else if (op6.equals("Ipv"))
{
    col1=10;
```

Monitorización de estaciones FV y meteo

```
        if (op7.equals("E-total")) col2=11;
    }
    else if (op7.equals("E-total"))
        col1=11;

    String aux1,aux2;
    float valors1[] = new float[numFiles-7];
    float valors2[] = new float[numFiles-7];
//recogemos datos

    int inicio=0,compt=0;
    inicio=inicio+7;

    // leemos a partir de la fecha, situamos inicio
    int dia_ent=Integer.parseInt(cal1.substring(8,10));
    data=full.getCell(0,inicio).getContents();
    //data=data.replace('/', ' ');
    int dia_ent_r=Integer.parseInt(data.substring(0,2));
    while (dia_ent>dia_ent_r)
    {
        inicio++;
        data=full.getCell(0,inicio).getContents();
        dia_ent_r=Integer.parseInt(data.substring(0,2));
    }
    // situamos final con num_files
    dia_ent=Integer.parseInt(cal2.substring(8,10));
    data=full.getCell(0,numFiles-1).getContents();
    dia_ent_r=Integer.parseInt(data.substring(0,2));
    while (dia_ent<dia_ent_r)
    {
        numFiles--;
        data=full.getCell(0,numFiles-1).getContents();
        dia_ent_r=Integer.parseInt(data.substring(0,2));
    }

    // ALERTAAA!
    if (inicio>numFiles) inicio=numFiles;
    // ho he afegit 3/6/15

    inicio=inicio + Integer.parseInt(contador);

    int conta=Integer.parseInt(contador);
    conta=conta + incremento*valor_pagina;
    contador=Integer.toString(conta);

    String fechas[] = new String[40];
    char c;
    String cad_fecha="";
    for (int i=0;i<40;i++) fechas[i]="";
```

```
for (int fila = inicio; ((fila < numFiles)&&(fila<(valor_pagina)*incremento+inicio)); fila=fila +
incremento)
    {
    if (col1!=0)
    {
        data = full.getCell(col1-1,fila).getContents();
        //cambiamos la , por .
        data=data.replace(',','.');
        // afegit dia 3 de Juny 2015, ja que l'inversor dona un error a una dada
        if (data.equals("")) valors1[compt]=0;
        else valors1[compt]=Float.parseFloat(data);
    }
    s=full.getCell(0,fila).getContents().substring(10,16);
    c=s.charAt(5);
    if (c==':') fechas[compt]=full.getCell(0,fila).getContents().substring(10,15);
    else fechas[compt]=full.getCell(0,fila).getContents().substring(10,16);
    if (col2!=0)
    {
        data = full.getCell(col2-1,fila).getContents();
        //cambiamos la , por .
        data=data.replace(',','.');
        valors2[compt]=Float.parseFloat(data);
    }
    compt++;
    }
String labels="labels : [\\"";
String dades1="data : [";
String dades2="data : [";
%>

<script src='Chart.min.js'></script>
<!-- line chart canvas element -->
<canvas id="buyers" width="900" height="350"></canvas>
<script>
<% labels=labels + fechas[0]; %>
var buyerData = {
    <% for (int i=0;i<compt; i++)
        {
            if ((i==(i/5)*5)&&(i!=0)) // mÃltiplos de 5
            {
                labels=labels + fechas[i] + "\",\\"";
            }
            else labels=labels + "\",\\"";
            dades1=dades1 + Integer.toString((int)valors1[i]) + ",";
            dades2=dades2 + Integer.toString((int)valors2[i]) + ",";
        }
    labels=labels + "\",\"";
    dades1=dades1 + "]"";
```

Monitorización de estaciones FV y meteo

```
        dades2=dades2 + "];
        out.println(labels);%>

datasets : [
{
    label: "Vac",
    fillColor : "rgba(172,194,132,0.4)",
    strokeColor : "#ACC26D",
    pointColor : "#fff",
    pointStrokeColor : "#9DB86D",
    legendTemplate: "Vac",
    //data : [203,156,99,251,305,247,23]
    <%out.println(dades1);%>
},
{
    label: "Iac",
    fillColor : "rgba(151,187,205,0.2)",
    strokeColor : "rgba(151,187,205,1)",
    pointColor : "rgba(151,187,205,1)",
    pointStrokeColor : "#fff",
    pointHighlightFill : "#fff",
    pointHighlightStroke : "rgba(151,187,205,1)",
    //data : [203,156,99,251,305,247,23]
    <%out.println(dades2);%>
}
]
}
// get line chart canvas
var buyers = document.getElementById('buyers').getContext('2d');
// draw line chart
new Chart(buyers).Line(buyerData);
</script>

</section>                                </div>

<!-- Content -->
<div id="content" class="8u skel-cell-important"><section><header>
</header></section></div>
</div>
</div>
</div>

<!-- Footer -->
<div id="featured">
    <div class="container">
        <%
String fecha;
if (xls==1)
{
```

Proyecto de Fin de Grado

```
// creamos el fichero xls
WritableWorkbook libro=Workbook.createWorkbook(new
File("./webapps/PFGr_JoanCarbonell/valores/ficheroFV.xls"));
WritableSheet hoja=libro.createSheet("Datos",0);
Label cab=new Label(0,0,"valores entre " + cal1 + " y " + cal2 + " de " +op1+"
"+op2+" "+op3+" "+op4+" "+op5+" "+op6+" "+op7);
hoja.addCell(cab);
Label cab2=new Label(1,2,"fecha");
hoja.addCell(cab2);
String option="";
if (col1==2) option="Vpv (V)";
else if (col1==4) option="Iac (mA)";
else if (col1==5) option="Vac (V)";
else if (col1==6) option="Fac (Hz)";
else if (col1==8) option="Zac (Ohm)";
else if (col1==10) option="Ipv (mA)";
else if (col1==11) option="E-total (Kw-h)";
Label cab3=new Label(2,2,option);
hoja.addCell(cab3);
option="";
if (col2==2) option="Vpv (V)";
else if (col2==4) option="Iac (mA)";
else if (col2==5) option="Vac (V)";
else if (col2==6) option="Fac (Hz)";
else if (col2==8) option="Zac (Ohm)";
else if (col2==10) option="Ipv (mA)";
else if (col2==11) option="E-total (Kw-h)";
Label cab4=new Label(3,2,option);
hoja.addCell(cab4);

// guardamos datos
int fila2=0;
for (int fila = inicio;fila < numFiles; fila=fila + incremento)
{
fila2++;
fecha=full.getCell(0,fila).getContents();
Label etiqueta1=new Label(1,fila2+2,fecha);
hoja.addCell(etiqueta1);
if (col1!=0)
{
data = full.getCell(col1-1,fila).getContents();
Label etiqueta2=new Label(2,fila2+2,data);
hoja.addCell(etiqueta2);
}
}
if (col2!=0)
{
data = full.getCell(col2-1,fila).getContents();
Label etiqueta3=new Label(3,fila2+2,data);
}
```


Monitorización de estaciones FV y meteo

```
        hoja.addCell(etiqueta3);
    }
}
libro.write();
libro.close();
}
%>
        descarga de valores en formato xls entre <%out.print(cal1 + " y " +cal2);%>
        <li><a href="/valores/ficheroFV.xls">ficheroFV.xls</a></li>
        <div class="row">
            <div class="4u">
                <h2>&nbsp;</h2>
                <p>&nbsp;</p>
            </div>
            <div class="4u">
                <p><a href="/graficosFV.jsp?contador=<%out.print(contador);
%>
&cal1=<%out.print(cal1);%>&cal2=<%out.print(cal2);%>&col1=<%out.print(op1);%>&col2=<
%out.print(op2);%>
&col3=<%out.print(op3);%>&col4=<%out.print(op4);%>&col5=<%out.print(op5);%>&col6=<
%out.print(op6);%>
&col7=<%out.print(op7);%>
&valor_pag=<%out.print(valor_pag);%>&tiempo=<%out.print(tiempo);%>"
class="button">Siguiete</a></p>
                <h2>Datos: </h2>
                <p></p>
                <div align="left">
            </div>
        </div>
        <% // Imprimimos la cabecera
String opcion="";
if (col1==2) opcion="Vpv (V)";
else if (col1==4) opcion="Iac (mA)";
else if (col1==5) opcion="Vac (V)";
else if (col1==6) opcion="Fac (Hz)";
else if (col1==8) opcion="Zac (Ohm)";
else if (col1==10) opcion="Ipv (mA)";
else if (col1==11) opcion="E-total (Kw-h)";

out.println("<table width=\"464\" border=\"1\">");
out.println("<tr bgcolor=\"#339999\">");
out.println("<th scope=\"col\"><span class=\"Estilo5\">\" + opcion + \"</span></th>");

if (col2==2) opcion="Vpv (V)";
else if (col2==4) opcion="Iac (mA)";
else if (col2==5) opcion="Vac (V)";
else if (col2==6) opcion="Fac (Hz)";
```

```
else if (col2==8) opcion="Zac (Ohm)";
else if (col2==10) opcion="Ipv (mA)";
else if (col2==11) opcion="E-total (Kw-h)";
if (col2!=0) out.println("<th scope=\"col\"><span class=\"Estilo5\">" + opcion + "</span></th>");
out.println("<th scope=\"col\"><span class=\"Estilo5\">" + "fecha" + "</span></th>");
out.println("</tr>");
// imprimimos datos
for (int fila = inicio; ((fila < numFiles)&&(fila<(valor_pagina)*incremento+inicio)); fila=fila +
incremento)
{
    fecha=full.getCell(0,fila).getContents();
    if (col1!=0)
    {
        data = full.getCell(col1-1,fila).getContents();
        out.println("<tr bgcolor=\"#FFFF99\">");
        out.println("<td><span class=\"Estilo11\">" + data + "</td>");
    }
    if (col2!=0)
    {
        data = full.getCell(col2-1,fila).getContents();
        out.println("<td><span class=\"Estilo11\">" + data + "</td>");
    }
    out.println("<td><span class=\"Estilo11\">" + fecha + "</td>");
    out.println("</tr>");
}
out.println("</table>");
arxiu.close();
}
catch (Exception e)
{
    throw new ServletException("problema abriendo el fichero", e);
}
finally
{
    ;
}
%>
```

```
<p>&nbsp;</p>
</div>
<div class="4u"></div>
</div>
</div>
</div>
<div id="copyright"></div>
</body>
</html>
```

graficosFVAnual.jsp

Se representan los valores de energía en un resumen anual.

```
<!DOCTYPE HTML>
<!--
    Iridium by TEMPLATED
    templated.co @templatedco
    Released for free under the Creative Commons Attribution 3.0 license (templated.co/license)
-->
<html>
  <head>
    <title>Iridium by TEMPLATED</title>
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
    <link href='http://fonts.googleapis.com/css?family=Arimo:400,700' rel='stylesheet'
type='text/css'>
    <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
    <script src="js/skel.min.js"></script>
    <script src="js/skel-panels.min.js"></script>
    <script src="js/init.js"></script>
    <style type="text/css">
<!--
.Estilo1 {font-size: 18px}
-->
  </style>
  <noscript>
    <link rel="stylesheet" href="css/skel-noscript.css" />
    <link rel="stylesheet" href="css/style.css" />
    <link rel="stylesheet" href="css/style-desktop.css" />
  </noscript>
  <!--[if lte IE 8]><link rel="stylesheet" href="css/ie/v8.css" /><![endif]-->
  <!--[if lte IE 9]><link rel="stylesheet" href="css/ie/v9.css" /><![endif]-->
</head>
<body>

  <!-- Header -->
  <div id="header">
    <div class="container">

      <!-- Logo -->
      <div id="logo">
        <h1><a href="#">Est FV y meteo</a></h1>
      </div>

      <!-- Nav -->
```

Proyecto de Fin de Grado

```
<nav id="nav">
    <ul>
        <li><a href="index.jsp">Inicio</a></li>
        <li class="active"><a href="FV.jsp">FV</a></li>
        <li><a href="meteo.jsp">meteo</a></li>
    </ul>
</nav>
</div>
</div>
<span class="Estilo1">
<!-- Main -->
</span>
<div id="main">
    <div class="container">
        <div class="row">
            <span class="Estilo1">
                <!-- Sidebar -->
            </span>
            <div id="sidebar" class="4u"><span
class="Estilo1"><section><header></header></section></span><section>
                <p class="text Estilo1">Gr&aacute;ficos FV: </p>
                <%@ page import="java.io.*" %>
                <%@ page import="java.util.*" %>
                <%@ page import="jxl.*" %>
                <%@ page import="java.string.*" %>
                <%
                try
                {
                    // leemos la fecha
                    String cal3, contador,any;
                    String fichero="";
                    cal3=(String)request.getParameter("cal3");
                    contador=(String)request.getParameter("contador");
                    int valor_pagina=20, incremento=1;
                    any=(String)request.getParameter("cal3");

                    fichero="SDT_" + any.substring(2,4) + ".xls";

                    //String directorio="C:/Archivos de programa/SMA Regelsysteme/Sunny Data
Control/Plants/DIEEC/SBC106367711/";
                    // Buscamos el directorio en C:/PFGr_JoanCarbonell.conf

                    FileReader fr=new FileReader("C:/PFGr_JoanCarbonell.conf");
                    BufferedReader br= new BufferedReader(fr);
                    String lin="";
```

Monitorización de estaciones FV y meteo

```
String directorio="";
lin=br.readLine();
while (lin!=null)
{
    String datos[]=lin.split("=");
    if (datos[0].equals("Directorio_Fotovoltaico"))
    {
        directorio=datos[1];
        break;
    }
    lin=br.readLine();
}
String s=directorio + fichero;
Workbook arxiu = Workbook.getWorkbook(new File(s));
Sheet full = arxiu.getSheet(0);
int numColumnes = full.getColumns();
int numFiles = full.getRows();

String data,data_ant;

String aux1,aux2;
float valors[] = new float[numFiles-7];
//recogemos datos
int inicio=0,compt=0;
inicio=inicio+7;
data_ant=0 + "";
inicio=inicio + Integer.parseInt(contador);
String fechas[] = new String[20];
for (int i=0;i<20;i++) fechas[i]="";
for (int fila = inicio; ((fila < numFiles)&&(fila<(valor_pagina)*incremento+inicio)); fila=fila +
incremento)
{
    data = full.getCell(1,fila).getContents();
    //cambiamos la , por .
    data=data.replace(',','.');
    valors[compt]=Float.parseFloat(data);
    fechas[compt]=full.getCell(0,fila).getContents().substring(0,5);
    compt++;
}

String labels="labels : [\\"";
String dades="data : [";
%>

<script src="./canvas/Chart.min.js"></script>
<canvas id="buyers" width="900" height="300"></canvas>
<script type="text/javascript">
// line chart data
<% labels=labels + fechas[0]; %>
```

```
var buyerData = {
    <% for (int i=0;i<compt; i++)
        {
            if ((i==(i/5)*5)&&(i!=0)) // múltiplos de 5
                {
                    labels=labels + fechas[i] + "\",\",";
                }
            else labels=labels + "\",\",";
            dades=dades + Float.toString(valors[i]) + ",";
        }
    labels=labels + "\"],";
    dades=dades + "]"";
    out.println(labels);%>

    datasets : [
    {
        fillColor : "rgba(172,194,132,0.4)",
        strokeColor : "#ACC26D",
        pointColor : "#fff",
        pointStrokeColor : "#9DB86D",
        //data : [203,156,99,251,305,247,23]
        <%out.println(dades);%>
    }
    ]
}
// get line chart canvas
var buyers = document.getElementById('buyers').getContext('2d');
// draw line chart
new Chart(buyers).Line(buyerData);
</script>

<h3>
<%
if (inicio < 365)
    out.print("Valores de Energía por día (en Kw-h) del " + full.getCell(0,inicio).getContents());
%>
</h3>

</section>                                </div>

<!-- Content -->
<div id="content" class="8u skel-cell-important"><section><header>
</header></section></div>
</div>
</div>
</div>
<!-- Footer -->
```

Monitorización de estaciones FV y meteo

```
<div id="featured">
  <div class="container">
    <div class="row">
      <div class="4u">
        <h2>&nbsp;</h2>
        <p>&nbsp;</p>
      </div>
      <div class="4u">
        <p>&nbsp;</p>
      </div>
      <div class="4u">
        <%
inicio=inicio - 7 + 20;
%>
        <p><a href="/graficosFVAnual.jsp?contador=<%out.print(inicio);%>
&cal3=<%out.print(cal3);%>" class="button">Siguiete</a></p>
      </div>
    </div>
  </div>
</div>

<%
arxiu.close();
}
catch (Exception e)
{
;
}
finally
{
;
}
%>

<div id="copyright"></div>

</body>
</html>
```

meteo.jsp

Recoge los rangos de fechas y los valores a representar de la estación meteorológica.

```
<!DOCTYPE HTML>
<!--
    Iridium by TEMPLATED
    templated.co @templatedco
    Released for free under the Creative Commons Attribution 3.0 license (templated.co/license)
-->
<html>
    <head>
        <title>PFG Joan Carbonell</title>
        <meta http-equiv="content-type" content="text/html; charset=utf-8" />
        <meta name="description" content="" />
        <meta name="keywords" content="" />
        <link href='http://fonts.googleapis.com/css?family=Arimo:400,700' rel='stylesheet'
type='text/css'>
        <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
        <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
        <script src="js/skel.min.js"></script>
        <script src="js/skel-panels.min.js"></script>
        <script src="js/init.js"></script>
        <noscript>
            <link rel="stylesheet" href="css/skel-noscript.css" />
            <link rel="stylesheet" href="css/style.css" />
            <link rel="stylesheet" href="css/style-desktop.css" />
        </noscript>
        <!--[if lte IE 8]><link rel="stylesheet" href="css/ie/v8.css" /><![endif]-->
        <!--[if lte IE 9]><link rel="stylesheet" href="css/ie/v9.css" /><![endif]-->
    </head>
    <body>

        <!-- Header -->
        <div id="header">
            <div class="container">

                <!-- Logo -->
                <div id="logo">
                    <h1><a href="#">Est FV y meteo</a></h1>
                </div>

                <!-- Nav -->
                <nav id="nav">
                    <ul>
                        <li><a href="index.jsp">Inicio</a></li>
                        <li><a href="FV.jsp">FV</a></li>
                    </ul>
                </nav>
            </div>
        </div>
    </body>
</html>
```


Monitorización de estaciones FV y meteo

```
<li class="active"><a href="meteo.jsp">meteo</a></li>
</ul>
</nav>
</div>
</div>

<!-- Main -->
<div id="main">
  <div class="container">
    <div class="row">
      <!-- Sidebar -->
      <div id="sidebar" class="4u">
        <section>
          <header>
            <h2>Est meteorol&oacute;gica en
            ETSII-UNED</h2>
            </header>
            <p class="text">Visualizaci&oacute;n de datos y gr&aacute;ficos
            meteorol&oacute;gicos.</p>
            <p class="text">&nbsp;</p>
          </section>
        </div>
      <!-- Content -->
    </div>
  </div>
</div>
</div>

<!-- Footer -->
<div id="featured">
  <div class="container">
    <div class="row">
      <div class="4u">
        <h2>Visualizaci&oacute;n de datos: </h2>

<%@page import="java.io.*"%>

<% //String graf=request.getParameter("graf");
int contador=0; %>

<form name="form2" method="post" onSubmit="" target="_top">
<li class="style">valores/p&aacute;gina
  <input name="valor_pag" type="radio" value="20" checked>
  20
```

```
<input name="valor_pag" type="radio" value="30" >
30
<input name="valor_pag" type="radio" value="40" >
40</li>
<li class="style">valores cada
  <input name="tiempo" type="radio" value="10 minutos" checked>
10 min</li>
<li class="style">  <input name="tiempo" type="radio" value="20 minutos" >
  20 min
  <input name="tiempo" type="radio" value="30 minutos" >
30 min
  <input name="tiempo" type="radio" value="60 minutos" >
60 min</li>
<p class="style">Rango de fechas:</p>
<span class="style">
<%
String ua = request.getHeader( "User-Agent" );
%>
</span>
<link rel="stylesheet" href="//code.jquery.com/ui/1.11.3/themes/smoothness/jquery-ui.css">
<span class="style">
<script src="//code.jquery.com/jquery-1.10.2.js"></script>
<script src="//code.jquery.com/ui/1.11.3/jquery-ui.js"></script>
</span>
<link rel="stylesheet" href="/resources/demos/style.css">
<span class="style">
<script>
$(function() {
$("#calendar1").datepicker({
  maxDate: "+0d",
  onClose: function(selectedDate)
  {
    $(this).datepicker("option","dateFormat","yy/mm/dd");
    $(this).datepicker("option","maxDate",-1d);
    $("#calendar2").datepicker("option","minDate",selectedDate);
    var mes=$(this).datepicker('getDate').getMonth();
    var any=$(this).datepicker('getDate').getFullYear();
    var dia=0;
    if ((mes==0)||(mes==2)||(mes==4)||(mes==6)||(mes==7)||(mes==9)||(mes==11))
      dia=31;
    else if (mes==1) dia=28;
    else dia=30;
    var fecha = new Date();
    var fecha_act = new Date();
    var dia_act=fecha_act.getDate();
    var mes_act=fecha_act.getMonth();
    var any_act=fecha_act.getFullYear();
    if ((mes==mes_act)&&(any==any_act)) dia=dia_act;
  }
});

```

Monitorización de estaciones FV y meteo

```
        fecha.setFullYear(any,mes,dia); //añ±o, mes y dia, mes: 0..11
        $("#calendar2").datepicker("option","maxDate",fecha);
        $("#calendar2").datepicker("option","dateFormat","yy/mm/dd");
    }
});
$("#calendar2").datepicker();
});

</script>

<%if (ua.matches(".*Android.*"))
{
    out.print("<input type='text' id='calendar1' name='cal1' onfocus='blur()'>");
    out.print("<input type='text' id='calendar2' name='cal2' onfocus='blur()'></p>");
}
else
{
    out.println("<input type='text' id='calendar1' name='cal1' onfocus='blur()'>");
    out.println("<input type='text' id='calendar2' name='cal2' onfocus='blur()'></p>");
}
%>

<script>
var num=0;
function cuenta(elem)
{
    if (elem.checked)
    {
        if (num < 2)
            num=num+1;
        else if (num==2)
            elem.checked=false;
    }
    else
        num=num-1;
}
</script>

<p>Valores (mximo 2): </p>
<p><span class="style">
    <input name="col1" type="checkbox" value="4" onclick="cuenta(this)">
    Lluvia
    <input name="col2" type="checkbox" value="5" onclick="cuenta(this)">
    Radiacin Solar</span></p>
<p><span class="style">
    <input name="col3" type="checkbox" value="7" onclick="cuenta(this)">
    Veloc viento
    <input name="col4" type="checkbox" value="6" onclick="cuenta(this)">
    Temp viento </span></p>
```

Proyecto de Fin de Grado

```
<p>
  <input name="Submit" type="submit" onclick = "this.form.action='./graficosMeteo.jsp?contador=<
%out.print(contador);%>" class="button" value="Visualizar">
  </p>
</form>

      </div>
      <div class="4u">
        <h2>&nbsp;</h2>
        <div align="left">
          <div align="left">
            <p><a href="#" class="image full"></a>
            </p>
          </div>
        </div>
        <p>&nbsp;</p>
        <p>&nbsp;</p>
      </div>
    <div class="4u"></div>
  </div>
</div>

  <div id="copyright"></div>

</body>
</html>
```

graficosMeteo.jsp

Representa gráficamente los valores elegidos en meteo.jsp y también presenta estos valores en una tabla de datos. Todo esto se consigue leyendo los valores de la base de datos mdb que nos crea Geonica 3k.

```
<!DOCTYPE HTML>
<!--
    Iridium by TEMPLATED
    templated.co @templatedco
    Released for free under the Creative Commons Attribution 3.0 license (templated.co/license)
-->
<html>
  <head>
    <title>PFG Joan Carbonell</title>
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
    <link href='http://fonts.googleapis.com/css?family=Arimo:400,700' rel='stylesheet'
type='text/css'>
    <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
    <script src="js/skel.min.js"></script>
    <script src="js/skel-panels.min.js"></script>
    <script src="js/init.js"></script>
    <style type="text/css">
<!--
.Estilo1 {font-size: 18px}
-->
  </style>
  <noscript>
    <link rel="stylesheet" href="css/skel-noscript.css" />
    <link rel="stylesheet" href="css/style.css" />
    <link rel="stylesheet" href="css/style-desktop.css" />
  </noscript>
    <!--[if lte IE 8]><link rel="stylesheet" href="css/ie/v8.css" /><![endif]-->
    <!--[if lte IE 9]><link rel="stylesheet" href="css/ie/v9.css" /><![endif]-->
  </head>
  <body>

    <!-- Header -->
    <div id="header">
      <div class="container">

        <!-- Logo -->
        <div id="logo">
```

```
<h1><a href="#">Est FV y meteo</a></h1>
</div>

<!-- Nav -->
<nav id="nav">
    <ul>
        <li><a href="index.jsp">Inicio</a></li>
        <li><a href="FV.jsp">FV</a></li>
        <li class="active"><a href="meteo.jsp">meteo</a></li>
    </ul>
</nav>
</div>
</div>

<span class="Estilo1">
<!-- Main -->
</span>
<div id="main">
    <div class="container">
        <div class="row">

            <span class="Estilo1">
                <!-- Sidebar -->
            </span>
            <div id="sidebar" class="4u"><span
class="Estilo1"><section><header></header></section></span><section>
                <p class="text Estilo1">Gr&acute;ficos meteorol&oacute;gicos: </p>
                <%@ page import="java.io.*" %>
                <%@ page import="java.util.*" %>
                <%@ page import="java.sql.*" %>
                <%@ page import="jxl.*" %>
                <%@ page import="jxl.write.*" %>

                <%
                Vector v = new Vector();
                Vector v2 = new Vector();
                int numCols=0;
                try
                {
                    // leemos la fecha
                    String cal1,cal2,mes,dia,dia2,conta,valor_pag,tiempo;
                    String Nump1,Nump2,Nump3,Nump4;
                    String NumParam1,NumParam2,NumFun1,NumFun2;
                    String fichero="";
                    String cadenaBD="";
                    // recogemos la fecha
                    cal1=(String)request.getParameter("cal1");
                    cal2=(String)request.getParameter("cal2");
```

Monitorización de estaciones FV y meteo

```
mes=cal1.substring(5,7);
dia=cal1.substring(8,10);
dia2=cal2.substring(8,10);

Nump1=(String)request.getParameter("col1");
Nump2=(String)request.getParameter("col2");
Nump3=(String)request.getParameter("col3");
Nump4=(String)request.getParameter("col4");

if (Nump1==null) Nump1="";
if (Nump2==null) Nump2="";
if (Nump3==null) Nump3="";
if (Nump4==null) Nump4="";
int NumParametro1=0,NumParametro2=0;
if (Nump1.equals("4"))
{
    NumParametro1=4;
    if (Nump2.equals("5")) NumParametro2=5;
    else if (Nump3.equals("7")) NumParametro2=7;
    else if (Nump4.equals("6")) NumParametro2=6;
}
else if (Nump2.equals("5"))
{
    NumParametro1=5;
    if (Nump3.equals("7")) NumParametro2=7;
    else if (Nump4.equals("6")) NumParametro2=6;
}
else if (Nump3.equals("7"))
{
    NumParametro1=7;
    if (Nump4.equals("6")) NumParametro2=6;
}
else if (Nump4.equals("6")) NumParametro1=6;

int xls=0;

conta=(String)request.getParameter("contador");
if (conta.equals("0")) xls=1;
valor_pag=(String)request.getParameter("valor_pag");
tiempo=(String)request.getParameter("tiempo");

int valor_pagina=0, incremento=0;
valor_pagina=Integer.parseInt(valor_pag);
if (tiempo.equals("10 minutos")) incremento=1;
else if (tiempo.equals("20 minutos")) incremento=2;
else if (tiempo.equals("30 minutos")) incremento=3;
else if (tiempo.equals("60 minutos")) incremento=6;

Calendar fecha_actual=new GregorianCalendar();
```

```
int any_act=fecha_actual.get(Calendar.YEAR);

if (cal1.substring(0,4).equals(Integer.toString(any_act)))
    fichero="MeteoStation.mdb";
else
    fichero="MeteoStation" + cal1.substring(0,4) + ".mdb";
int numMes=Integer.parseInt(cal1.substring(5,7));

//int NumParametro=Integer.parseInt(NumParam);
int NumFuncion1=0,NumFuncion2=0;
// para meteo hay que hacer una reconversión
if (NumParametro1==4)
    NumFuncion1=2;
else if (NumParametro1==5)
    NumFuncion1=1;
else if (NumParametro1==7)
    NumFuncion1=1;
else if (NumParametro1==6)
    NumFuncion1=1;
if (NumParametro2==4)
    NumFuncion2=2;
else if (NumParametro2==5)
    NumFuncion2=1;
else if (NumParametro2==7)
    NumFuncion2=1;
else if (NumParametro2==6)
    NumFuncion2=1;
NumParam1=Integer.toString(NumParametro1);
NumParam2=Integer.toString(NumParametro2);
NumFun1=Integer.toString(NumFuncion1);
NumFun2=Integer.toString(NumFuncion2);
int contador=Integer.parseInt(conta);

//buscamos el directorio meteorológico
FileReader fr=new FileReader("C:/PFGr_JoanCarbonell.conf");
BufferedReader br= new BufferedReader(fr);
String lin="";
String directorio="";
lin=br.readLine();
while (lin!=null)
{
    String datos[]=lin.split("=");
    if (datos[0].equals("Directorio_Meteorologico"))
    {
        directorio=datos[1];
        break;
    }
    lin=br.readLine();
}
```


Monitorización de estaciones FV y meteo

```
}

cadenaBD=directorío + fichero;

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
String s="jdbc:odbc:Driver={Microsoft Access Driver (*.mdb)};DBQ="+ cadenaBD;
Connection c = java.sql.DriverManager.getConnection(s, "", "");
Statement orden = c.createStatement();
String sql="";
ResultSet rs;
ResultSetMetaData metaDatos;
sql="SELECT Datos.Valor, Datos.Fecha FROM Datos" +
    " WHERE MONTH([Datos.Fecha])=" + numMes + "And DAY([Datos.Fecha])>=" + dia + " AND
DAY([Datos.Fecha])<=" + dia2 + "AND Datos.NumParametro="
    + NumParam1 + "AND Datos.NumFuncion="+ NumFun1 + " ORDER BY Datos.Fecha";

rs = orden.executeQuery(sql);
metaDatos = rs.getMetaData();
numCols = metaDatos.getColumnCount();
while (rs.next()) {
    for (int i = 1; i <= numCols; i++) {
        v.addElement(rs.getString(i));
    }
}
rs.close();
orden.close();
c.close();
String Valores[][] = new String[v.size() / numCols][numCols];
for (int fila = 0; fila < (v.size()/numCols); fila++)
{
    for (int i = 0; i < numCols; i++)
    {
        Valores[fila][i] = (String)v.elementAt(fila * numCols + i);
        if (Valores[fila][i].isEmpty()) Valores[fila][i]="0";
    }
}
String Valores2[][] = new String[v.size() / numCols][numCols];
if (NumParametro2==0)
    for (int fila = 0; fila < (v.size()/numCols); fila++)
        for (int i=0; i < numCols; i++) Valores2[fila][i]="0";
c = java.sql.DriverManager.getConnection(s, "", "");
sql="SELECT Datos.Valor, Datos.Fecha FROM Datos" +
    " WHERE MONTH([Datos.Fecha])=" + numMes + "And DAY([Datos.Fecha])>=" + dia + " AND
DAY([Datos.Fecha])<=" + dia2 + "AND Datos.NumParametro="
    + NumParam2 + "AND Datos.NumFuncion="+ NumFun2 + " ORDER BY Datos.Fecha";

ResultSet rs2;
ResultSetMetaData metaDatos2;
Statement orden2 = c.createStatement();
```

```
rs2 = orden2.executeQuery(sql);
metaDatos2 = rs2.getMetaData();
numCols = metaDatos2.getColumnCount();
while (rs2.next()) {
    for (int i = 1; i <= numCols; i++) {
        v2.addElement(rs2.getString(i));
    }
}

rs2.close();
orden2.close();
c.close();

if (NumParametro2!=0)
{
for (int fila = 0;fila < (v2.size()/numCols); fila++) {
    for (int i = 0; i < numCols; i++)
    {
        Valores2[fila][i] = (String)v2.elementAt(fila * numCols + i);
        if (Valores2[fila][i].isEmpty()) Valores2[fila][i]="0";
    }
}
}

String palabra1="",palabra2="";
if (NumParametro1==4) palabra1="Lluvia ";
else if (NumParametro1==5) palabra1="Radiación ";
else if (NumParametro1==7) palabra1="Vel viento ";
else if (NumParametro1==6) palabra1="Temp viento ";
if (NumParametro2==4) palabra2="Lluvia ";
else if (NumParametro2==5) palabra2="Radiación ";
else if (NumParametro2==7) palabra2="Vel viento ";
else if (NumParametro2==6) palabra2="Temp viento ";
out.print(palabra1 + palabra2);
%>

<script src='Chart.min.js'></script>
<!-- line chart canvas element -->
<canvas id="buyers" width="900" height="350"></canvas>
<script>
<%
String labels="labels : [\\"";
String dades1="data : [";
String dades2="data : [";
%>
var buyerData = {
```

Monitorización de estaciones FV y meteo

```
<% float valor_f;
String fecha;
int max_print=valor_pagina*incremento;
if ((v.size()-contador)/numCols<valor_pagina) max_print=((v.size()-
contador)/numCols);
for (int i=contador; ((i<(max_print + contador
-1))&&(i<v.size()/numCols)); i=i+incremento)
{
if ((i==(i/5)*5)&&(i!=0)) // múltiplos de 5
{
fecha=Valores[i][1].substring(11,16);
labels=labels + fecha + "\",\",";
}
else labels=labels + "\",\",";
valor_f=Float.parseFloat(Valores[i][0]);
dades1=dades1 + Integer.toString((int)valor_f) + ",";
}
labels=labels + "\",\",";
if ((v.size()/numCols)>= (max_print + contador -1))
valor_f=Float.parseFloat(Valores[max_print + contador -1][0]);
else valor_f=Float.parseFloat(Valores[(v.size()/numCols)-1][0]);
//out.print(valor_f);
dades1=dades1 + Integer.toString((int)valor_f) + "\",";

if ((v.size()-contador)/numCols<valor_pagina) max_print=((v.size()-
contador)/numCols);
for (int i=contador; ((i<(max_print + contador
-1))&&(i<v.size()/numCols)); i=i+incremento)
{
valor_f=Float.parseFloat(Valores2[i][0]);
dades2=dades2 + Integer.toString((int)valor_f) + ",";
}
if ((v.size()/numCols)>= (max_print + contador -1))
valor_f=Float.parseFloat(Valores2[max_print + contador -1][0]);
else valor_f=Float.parseFloat(Valores2[(v.size()/numCols)-1][0]);
dades2=dades2 + Integer.toString((int)valor_f) + "\",";

out.println(labels);%>
//labels : ["January","February","March","April","May","June","prova"],
datasets : [
{
fillColor : "rgba(172,194,132,0.4)",
strokeColor : "#ACC26D",
pointColor : "#fff",
pointStrokeColor : "#9DB86D",
//data : [203,156,99,251,305,247,23]
<% out.println(dades1);%>
},
{
```

```
fillColor : "rgba(151,187,205,0.2)",
strokeColor : "rgba(151,187,205,1)",
pointColor : "rgba(151,187,205,1)",
pointStrokeColor : "#fff",
pointHighlightFill : "#fff",
pointHighlightStroke : "rgba(151,187,205,1)",
//data : [203,156,99,251,305,247,23]
<% out.println(dades2);%>
}
]
}
// get line chart canvas
var buyers = document.getElementById('buyers').getContext('2d');
// draw line chart
new Chart(buyers).Line(buyerData);
</script>

</section>                                </div>

<!-- Content -->
<div id="content" class="8u skel-cell-important"><section><header>
</header></section></div>
</div>
</div>
</div>

<!-- Footer -->
<div id="featured">
  <div class="container">
    <%
if (xls==1)
{
  // creamos el fichero xls
  WritableWorkbook libro=Workbook.createWorkbook(new
File("./webapps/PFGr_JoanCarbonell/valores/ficheroMeteo.xls"));
  WritableSheet hoja=libro.createSheet("Datos",0);

  Label cab=new Label(0,0,"valores entre " + cal1 + " y " + cal2 + " de "+ palabra1 +
palabra2);
  hoja.addCell(cab);
  Label cab2=new Label(1,2,"fecha");
  hoja.addCell(cab2);
  Label cab3=new Label(2,2,palabra1);
  hoja.addCell(cab3);
  Label cab4=new Label(3,2,palabra2);
  hoja.addCell(cab4);
  // guardamos datos
  int i=0,total=0;
```

Monitorización de estaciones FV y meteo

```
total=v.size()/(numCols*incremento);
int fila=0;
for (i=0; (i<total); i++)
{
    Label etiqueta1=new Label(3,i+3,Valores2[fil][0]);
    Label etiqueta2=new Label(2,i+3,Valores[fil][0]);
    Label etiqueta3=new Label(1,i+3,Valores[fil][1]);
    hoja.addCell(etiqueta1);
    hoja.addCell(etiqueta2);
    hoja.addCell(etiqueta3);
    fila=fila+incremento;
}
libro.write();
libro.close();
}
%>
descarga de valores en formato xls entre <%out.print(cal1 + " y " +cal2);%>
<li><a href="/valores/ficheroMeteo.xls">ficheroMeteo.xls</a></li>
<div class="row">
    <div class="4u">
        <h2>&nbsp;</h2>
        <p>&nbsp;</p>
    </div>
<div class="4u">
<%
contador=contador + incremento*valor_pagina;
conta=Integer.toString(contador);
%>
<p><a href="/graficosMeteo.jsp?contador=<%out.print(conta);
%>
&cal1=<%out.print(cal1);%>&cal2=<%out.print(cal2);%>&col1=<%out.print(Nump1);%>&col2=<
%out.print(Nump2);%>
&col3=<%out.print(Nump3);%>&col4=<%out.print(Nump4);%>&valor_pag=<%out.print(valor_pag);
%>&tiempo=<%out.print(tiempo);%>" class="button">Siguiente</a></p>
<h2>Datos: </h2>
<p></p>
<div align="left">
</div>
<% // Imprimimos la cabecera
out.println("<table width=\"464\" border=\"1\">");
out.println("<tr bgcolor=\"#339999\">");
String cabecera[]={ "lluvia (mm)", "lluvia (mm)", "fecha" };
if (NumParametro1==5) cabecera[0]="radiación (W/m2)";
else if (NumParametro1==7) cabecera[0]="velocidad (m/s)";
else if (NumParametro1==6) cabecera[0]="temperatura (°C)";
if (NumParametro2==5) cabecera[1]="radiación (W/m2)";
else if (NumParametro2==7) cabecera[1]="velocidad (m/s)";
```

Proyecto de Fin de Grado

```
else if (NumParametro2==6) cabecera[1]="temperatura (°C)";
out.println("<th scope=\"col\"><span class=\"Estilo5\">" + cabecera[0] + "</span></th>");
if (NumParametro2!=0) out.println("<th scope=\"col\"><span class=\"Estilo5\">" + cabecera[1] +
"</span></th>");
out.println("<th scope=\"col\"><span class=\"Estilo5\">" + cabecera[2] + "</span></th>");
out.println("</tr>");
// imprimimos datos
max_print=valor_pagina*incremento;
contador=contador - incremento*valor_pagina;
for (int fila=contador; (fila<((v.size())/numCols) && (fila<contador + max_print)); fila=fila+incremento)
{
    out.println("<tr bgcolor=\"#FFFF99\">");
    out.println("<td><span class=\"Estilo11\">" + Valores[fila][0] + "</td>");
    if (NumParametro2!=0) out.println("<td><span class=\"Estilo11\">" + Valores2[fila][0] + "</td>");
    out.println("<td><span class=\"Estilo11\">" + Valores[fila][1] + "</td>");
    out.println("</tr>");
}
out.println("</table>");
contador=contador + incremento*valor_pagina;
conta=Integer.toString(contador);
}
catch (Exception e)
{
    ;
}
finally
{
}
%>
```

<p> </p>

</div>

<div class="4u"></div>

</div>

</div>

</div>

<div id="copyright"></div>

</body>

</html>

licencia.jsp

Se describe la licencia cc3.0 del diseñador de la plantilla Iridium.

```
<!DOCTYPE HTML>
<!--
    Iridium by TEMPLATED
    templated.co @templatedco
    Released for free under the Creative Commons Attribution 3.0 license (templated.co/license)
-->
<html>
  <head>
    <title>Iridium by TEMPLATED</title>
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
    <link href='http://fonts.googleapis.com/css?family=Arimo:400,700' rel='stylesheet'
type='text/css'>
    <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
    <script src="js/skel.min.js"></script>
    <script src="js/skel-panels.min.js"></script>
    <script src="js/init.js"></script>
    <style type="text/css">
<!--
.Estilo1 {font-size: 18px}
-->
  </style>
  <noscript>
    <link rel="stylesheet" href="css/skel-noscript.css" />
    <link rel="stylesheet" href="css/style.css" />
    <link rel="stylesheet" href="css/style-desktop.css" />
  </noscript>
  <!--[if lte IE 8]><link rel="stylesheet" href="css/ie/v8.css" /><![endif]-->
  <!--[if lte IE 9]><link rel="stylesheet" href="css/ie/v9.css" /><![endif]-->
</head>
<body>

  <!-- Header -->
  <div id="header">
    <div class="container">

      <!-- Logo -->
      <div id="logo">
        <h1><a href="#">Est FV y meteo</a></h1>
      </div>

      <!-- Nav -->
    <nav id="nav">
```

Proyecto de Fin de Grado

```
<ul>
  <li><a href="index.jsp">Inicio</a></li>
  <li><a href="FV.jsp">FV</a></li>
  <li><a href="meteo.jsp">meteo</a></li>
  <li class="active"><a href="/licencia.jsp">Iridium
template</a></li>
</ul>
</nav>
</div>
</div>
<span class="Estilo1">
<!-- Main -->
</span>
<div id="main">
  <div class="container">
    <div class="row">
      <span class="Estilo1">
        <!-- Sidebar -->
      </span>
      <div id="sidebar" class="4u"><span
class="Estilo1"><section><header></header></section></span><section>
<p class="text Estilo1">Licencia Creative Commons 3.0 </p>
<header>
</header>
<p class="text Estilo1">En esta aplicación se ha utilizado la plantilla
Iridium y se ha introducido código JSP para procesar datos fotovoltaicos
y meteorológicos</p>
<li><a href="http://templated.co">enlace a la web de descarga</a></li>
</section>
</div>
<!-- Content -->
<div id="content" class="8u skel-cell-important"><section><header>
</header></section></div>
</div>
</div>
<!-- Footer -->
<div id="featured">
  <div class="container">
```


Monitorización de estaciones FV y meteo

```
<div class="row">
  <div class="4u">
    <h2>&nbsp;</h2>
    <p>&nbsp;</p>
  </div>
  <div class="4u">
    <p>&nbsp;</p>
  </div>
  <div class="4u">
    </div>
  </div>
  </div>
  </div>
  </div>
  <div id="copyright"></div>
</body>
</html>
```

Proyecto de Fin de Grado

CV del autor



Datos Personales:

Nombre: *Joan Miquel Carbonell Crespí*
Lugar de nacimiento: *Muro(Mallorca)*
mail: [*joancc20@gmail.com*](mailto:joancc20@gmail.com)

Estudios Universitarios:

- Grado en Ingeniería Eléctrica en ETSII de la UNED
- Ingeniería de Telecomunicaciones en ETSETB de Barcelona
- Ingeniería Técnica de Sistemas Electrónicos en EUPVG de Barcelona

Experiencia Profesional:

- 3 años trabajando como profesor de FP en un Instituto de Imagen y Sonido
- 1 año trabajando como profesor de Electrónica en la Universidad E.U.E.T.I. (UPC) .
- 8 años trabajando en networking, diseñando redes y servidores para un ISP: BITel S.A. en Palma de Mallorca.
- 12 años trabajando en un Instituto de Electrónica como profesor de sistemas operativos, lenguajes de programación, domótica, radicomunicaciones y redes telemáticas.
- 10 años trabajando de manera autónoma en proyectos de telecomunicaciones y proyectos domóticos.

Lenguajes:

Castellano, Catalán e Inglés.

Proyectos desarrollados:

Sistemas.

- Servidores Windows con Active Directory (Win2k i Win2k3) con estaciones windows y linux
- Servidores Linux con servicios de red: apache, mail, ...
- LTSP (implementación de un servidor con Linux Terminal Server Project) con terminales Windows y Linux
- Servidor WinRadius para accounting en una red VoIP implementando AAA en un gatekeeper CISCO (con IOS h323 para enumeración con VoIP)
- Servidor MRTG para monitorización con SNMP
- Servidor SNMP con NNM (HPOV) y CiscoWorks con unos 200 nodos gestionados

Networking.

- Red VoIP diseñada con más de 100 routers Cisco de diferentes modelos (1750, 2621, 3640, 837). Enrutamiento dinámico, gw de voz con AS5300. Líneas RDSI, ADSL. Seguridad con FW PIX. SNMP con HPOV. Programación de centralitas. VPN's, encapsulamiento de X.25 sobre IP, data link switching para SNA para máquinas AS400.
- Red VoIP con 2 routers 1750 y centralitas Ericsson.
- ISP diseñado con seguridad perimetral centralizada con Fw CheckPoint.
- Diseño de Firewall con soluciones FortiNet (FortiGate y FortiAnalyzer)
- LAN diseñada con tagging, trunking y routing. (con equipamiento de switching Cisco)
- Red Wireless con productos Cisco (Aironet) y LinkSys.
- Diseño LAN con AP's dlink y salida balanceada a Internet con source routing

Desarrollo.

- Diseño de páginas web con HTML
- Diseño de páginas dinámicas con PHP y mysql
- Software de gestión con Builder C++ (entorno Windows) y con Kdevelop y Designer-QT (entorno Linux)
- Diseño web con Java servlets y JSPs.

Infraestructuras de Telecomunicaciones.

- ICTs
- Cableado estructurado.
- Domótica: Integración mediante KNX, Crestron y tecnología IP de red de RTV, datos, wifi y controles de iluminación, climatización, persianas y red de seguridad.