



Título del Proyecto:

WebApp
Panelistas Hornos
BSH

Autor:

Palmer Camps,
Carlos

Director:

Tomás Gironés,
Jesús

**TESINA PARA LA
OBTENCIÓN DEL TÍTULO DE:**

**Diploma de Especialización en
Desarrollo de Aplicaciones para
Android**

Septiembre del 2015

Introducción

Descripción del problema

Actualmente se está trabajando en el diseño y fabricación de un nuevo producto para hornos en BSH Montañana. El objetivo de las preseries que se fabricarán será, además de comprobar nuestros nuevos sistemas productivos, layouts y desarrollo, testear el correcto funcionamiento de los aparatos.

Para ello, además de pruebas que se definirán dentro de fábrica, se quiere contar con un número de usuarios panelistas a los que se les entregará un horno para instalarlo en su hogares, y poder recibir de su parte un feedback sobre el funcionamiento del aparato con el uso normal que un cliente final podría darle.

Los datos recibidos, serán utilizados para la mejora del producto antes del lanzamiento de la producción en serie del mismo.

Para el proyecto que actualmente se fabrica en BSH Montañana, no existe una web o sistema para recibir las valoraciones y perfiles del usuario. El flujo de información es meramente verbal.

El funcionamiento de dicha aplicación web será el siguiente:

Introducción de la información de usuario:

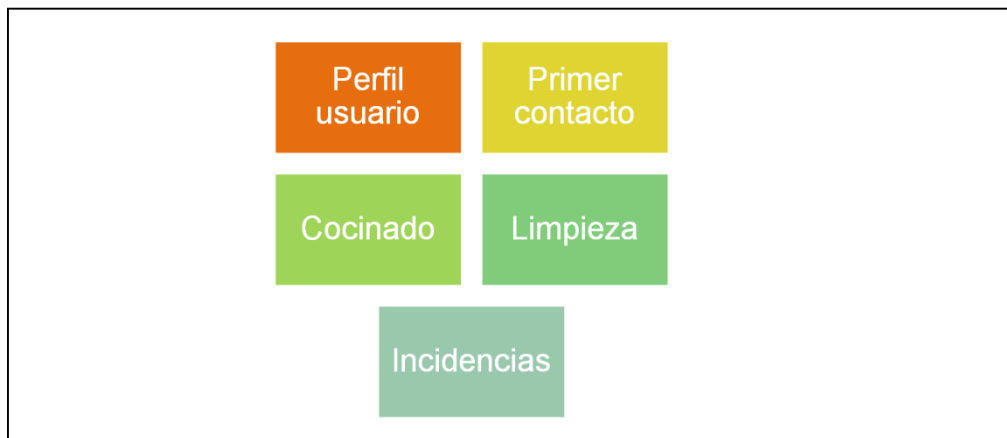
- Cada usuario será registrado inicialmente por el administrador de BSH (que se definirá internamente), incorporando:
 - Nombre y apellidos del usuario de panel
 - Aparato que se le entrega (VIB y SNR)
 - Email y teléfono del usuario
 - Fecha de entrega
 - Diferente user y password para cada usuario
- Cada panelista entrará en su sesión de la Web App con su usuario y contraseña. En caso de olvido, debe existir la opción de recuperación de contraseña que será enviada de forma automática a su email.
- Cada panelista tendrá que responder a un formulario que consta de diferentes módulos presentados a continuación.

En primer lugar, incluirá una ventana inicial para introducir el usuario y contraseña;



B/S/H/
Inicio <ul style="list-style-type: none">■ Log in/Log out■ User /Password■ Opción de recuperación de contraseña■ Dos tipos de usuarios<ul style="list-style-type: none">– Administrador (Pueda recibir estadísticas, modificar app)– Panelista

Tras el inicio de sesión aparecerá el menú principal (El panelista podrá entrar en cada uno de los bloques y tendrá la opción de regresar al menú nuevamente)



En cada uno de los módulos se presentarán varias preguntas que el panelista deberá ir respondiendo;


Incluirán desplegables o varias opciones para seleccionar (Ver ejemplos a continuación).

Qué te gusta hacer en tu tiempo libre: <input type="text" value="Elija un elemento"/>	<input type="radio"/> Masculino
<ul style="list-style-type: none">Elija un elemento.LeerIr al cineSalir con mis amigosIr de compras	<input checked="" type="radio"/> Femenino

El primer módulo es el perfil del usuario, se contemplan datos sencillos para crear un perfil de panelista

Perfil

- Edad (**Desplegable** con intervalo de edad)
- Personas en casa (**Desplegable**)
- Usos (**Desplegable**)
 - >1/semana
 - 1/semana
 - 1-2 /mes
- Uso frecuente del horno(**Desplegable**)
 - Cocinado
 - Precocinado
 - Gratinado
 - Asado
 - Descongelado
 - Repostería



El siguiente modulo corresponderá al primer contacto con el horno;

Primer contacto

- ¿Aprecia defectos en el aparato?
 - SI (Vincular a **incidencias**)
 - NO
- Instalación (**Desplegable**)
 - Bajo balda (Opcional; foto del encastre)
 - Columna (Opcional; foto del encastre)
- ¿Ha leído el manual? (**Desplegable**)
 - SI (**Incluye 3 sub-apartados**)
 - ❖ Claro (**Desplegable**; Valoración 0-5)
 - ❖ Legible (**Desplegable**; Valoración 0-5)
 - ❖ Funcional (**Desplegable**; Valoración 0-5)
 - NO
- Puerta (**Desplegable**)
 - De Carro
 - Abatible
- Accesorios (**3 sub-apartados para seleccionar**)
 - Bandeja esmaltada
 - Half-tray
 - Parrilla
 - Flex grill

- Primera valoración (Sub-apartados, si la valoración es igual o menor de 3 vincular a incidencias)
 - Maneta (**Desplegable**; Valoración 0-5)
 - Botones (**Desplegable**; Valoración 0-5)
 - Slider (**Desplegable**; Valoración 0-5)
 - Reloj (**Desplegable**; Valoración 0-5)
 - Puerta (**Desplegable**; Valoración 0-5)
 - Gradadas (**Desplegable**; Valoración 0-5)
 - Accesorios (**Desplegable**; Valoración 0-5)

El módulo de **cocinado** permitirá por un lado introducir lo que se vaya a cocinar con diferentes apartados con despleables u opciones. Por otro lado, se plantea la opción de introducir unas recetas concretas que todos los panelistas realicen y permitan obtener resultados comparativos.

Cocinado

- ¿Qué cocinas hoy? (Con **sub-apartados**, véase ejemplo)
 - Comida (A **rellenar** por el usuario)
 - Pre-calentamiento (**Desplegable**, SI/NO)
 - Función (**Desplegable**; con iconos)
 - Temperatura (**Desplegable**, ruleta?)
 - Tiempo (**Desplegable**)
 - Satisfacción (**Desplegable**)
- Receta del mes (Base de datos con recetas)

Comida	Pre-heat	Función	Temperatura (°C)	Tiempo (min)	Satisfacción
Pollo	Si/No	Grill	200	60	😊 😞
...					

El siguiente módulo se referirá a la **limpieza**, separando la interior (Cavidad) y la exterior (estética) y la de los accesorios.

Limpieza

INTERIOR - Cavidad

- Ecolítico
 - Satisfacción de la limpieza (**Desplegable**)
- Easy- to- clean
 - Especifique el producto de limpieza (A **rellenar** por el usuario)
 - Satisfacción de la limpieza (**Desplegable**)
- Pirolítico
 - ¿Ha retirado los residuos antes de pirolizar? (SI/NO)
 - Nivel (**Desplegable**; I, II ó III)
 - Satisfacción (**Desplegable**)
- Manual
 - Especifique el producto de limpieza (A **rellenar** por el usuario)

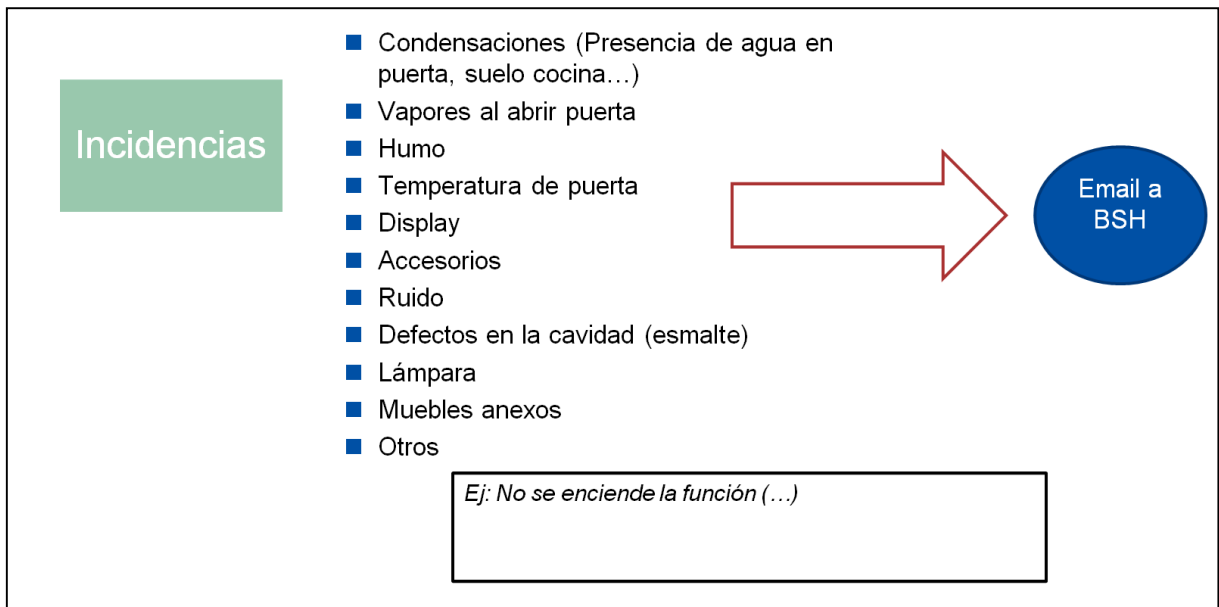
EXTERIOR - Estética

- Seleccione el producto de limpieza
 - Limpiacristales
 - Quitagrasas
 - Detergente
 - Otros

Sub-apartados para seleccionar el modo de limpieza usado según modelo de horno y su satisfacción



El último bloque se dedica a las **incidencias**, se incluirá sub-apartados con incidencias típicas y una opción de otros para que el usuario introduzca su incidencia (ver ejemplo), todas las incidencias se enviarán a un email especificado.



Objetivos

Se requiere para recibir esta información, un sistema Web App de sencillo mantenimiento, en el que tanto la introducción de datos, como el tratamiento posterior de éstos, sea de fácil o automático manejo.

Otro de los objetivos del propio sistema web era que fuera un sistema WebApp multilingüe. En este caso se ha realizado el sitio en dos idiomas, el español e inglés, para el cual se ha realizado una función que se encarga de realizar la función de las traducciones a partir de un diccionario con las palabras y sus respectivas traducciones en JSON.

Con la finalidad de obtener un sistema Web App que fuese multiplataforma y con la ventaja de nos aportan actualmente las tecnologías web, se ha optado por realizar el proyecto para que sea válido en cualquier plataforma, ya sea para Android o IOS así como para cualquier tipo de dispositivo sobre el que se lance la aplicación.

El objetivo básico de esta aplicación se basará en la cumplimentación de los distintos formularios por parte del usuario, y que luego se guardaran en la base de datos mediante el uso de unos servicios web definidos.

Finalmente se ha realizado una APP de Android y otra en IOS paquetizando el contenido de la aplicación web desarrollada dentro de la propia aplicación.

Motivación

Deseaba realizar un proyecto final que tuviese una finalidad en lugar de realizar una aplicación que solo valiese para un proyecto. Se me brindó la oportunidad de realizarla en una empresa que está relativamente cerca y por ello opté a realizar el proyecto en dicha empresa.

Otra de las motivaciones fue el de desarrollar una sistema WebApp multiplataforma que se adaptase a todos los requisitos que se me pedían ya que anteriormente nunca se me hubiese ocurrido realizarlo así, y gracias a toda la tecnología web que disponemos actualmente se ha podido realizar.

Situación de... / Tecnologías utilizadas

HTML5

HTML5 es el futuro más inmediato de la Web: muchos de los sitios y aplicaciones Web más importantes del momento implementan algunas de sus características, como la reproducción de vídeo, las API que permiten integrarse con el sistema operativo o la aceleración gráfica. Muchas veces vemos verdaderas virguerías en forma de experimento.

Además, hay que tener en cuenta que HTML5 es especialmente interesante para ofrecer experiencias de usuario impresionantes en dispositivos que, por defecto y casi por definición, no admiten la instalación de complementos de terceros, como móviles y tabletas. El navegador Web de iOS (en iPhone e iPad), sin ir más lejos, únicamente admite HTML5 (y sus tecnologías relacionadas).

Pero el futuro prometedor, según muchos fabricantes, va más allá de la Web, y pasa por utilizar HTML5 como una plataforma completa de desarrollo de aplicaciones, en conjunción siempre con CSS3 y JavaScript

De hecho, algunos sistemas operativos han ido más allá. El sistema operativo de la renacida Palm, web OS, ya permitía que las aplicaciones fueran desarrolladas principalmente con tecnologías Web.

Ubuntu Phone también ofrece un SDK para desarrollar e integrar aplicaciones Web en el sistema operativo. Y, llevando el concepto al extremo, Firefox OS hace que toda la interfaz gráfica de usuario sea desarrollada mediante tecnologías Web, pasando a ser el sistema operativo una especie de Linux con Gecko.

CSS3

La versión 3 de CSS sigue el mismo camino, pero esta vez con un mayor compromiso. La especificación de HTML5 fue desarrollada considerando CSS a cargo del diseño. Debido a esta consideración, la integración entre HTML y CSS es ahora vital para el desarrollo web y esta es la razón por la que cada vez que mencionamos HTML5 también estamos haciendo referencia a CSS3, aunque oficialmente se trate de dos tecnologías completamente separadas.

En este momento las nuevas características incorporadas en CSS3 están siendo implementadas e incluidas junto al resto de la especificación en navegadores compatibles con HTML5.

En definitiva, CSS3 nos deja varias cosas muy buenas: la opacidad, con la que yo ya he estado jugando y se hacen verdaderas maravillas, la posibilidad de poder usar cualquier fuente en nuestros textos, la facilidad para crear cajas redondeadas y algunos efectos espectaculares como sombras y marcos.

BOOTSTRAP

Los últimos años y con la aparición de la web 2.0 Internet ha cambiado y se ha transformado para dar acogida a todas las necesidades de sus usuarios, y por esa razón los sitios web también ha tenido que cambiar mucho.

Hace unos 3 o 4 años, ahí por 2011 se empezó a hablar de los sitios web responsive o adaptables a todo tipo de pantallas y dispositivos fuese cual fuese su tamaño, esta capacidad de adaptación de los sitios web se consiguió utilizando técnicas CSS avanzadas para su desarrollo o utilizando frameworks CSS como por ejemplo Bootstrap.

Vamos a empezar por el principio, Bootstrap es un framework CSS desarrollado inicialmente (en el año 2011) por Twitter que permite dar forma a un sitio web mediante librerías CSS que incluyen tipografías, botones, cuadros, menús y otros elementos que pueden ser utilizados en cualquier sitio web.

Bootstrap es una excelente herramienta para crear interfaces de usuario limpias y totalmente adaptables a todo tipo de dispositivos y pantallas, sea cual sea su tamaño.

Además, Bootstrap ofrece las herramientas necesarias para crear cualquier tipo de sitio web utilizando los estilos y elementos de sus librerías.

Desde la aparición de Bootstrap 3 el framework se ha vuelto bastante más compatible con desarrollo web responsive, entre otras características se han reforzado las siguientes:

- Soporte bastante bueno (casi completo) con HTML5 y CSS3, permitiendo ser usado de forma muy flexible para desarrollo web con unos excelentes resultados.
- Se ha añadido un sistema GRID que permite diseñar usando un GRID de 12 columnas donde se debe plasmar el contenido, con esto podemos desarrollar responsive de forma mucho más fácil e intuitiva.

- Bootstrap 3 establece Media Queries para 4 tamaños de dispositivos diferentes variando dependiendo del tamaño de su pantalla, estas Media Queries permiten desarrollar para dispositivos móviles y tablets de forma mucho más fácil.
- Bootstrap 3 también permite insertar imágenes responsive, es decir, con solo insertar la imagen con la clase “img-responsive” las imágenes se adaptaran al tamaño.

Todas estas características hacen que Bootstrap sea una excelente opción para desarrollar webs y aplicaciones web totalmente adaptables a cualquier tipo de dispositivo.

Bootstrap es compatible con la mayoría de navegadores web del mercado, y más desde la versión 3, actualmente es totalmente compatible con los siguientes navegadores:

- Google Chrome (en todas las plataformas).
- Safari (tanto en iOS como en Mac).
- Mozilla Firefox (en Mac y en Windows).
- Internet Explorer (en Windows y Windows Phone).
- Opera (en Windows y Mac).

JQUERY

jQuery es una biblioteca de JavaScript que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC.

Es la biblioteca de JavaScript más utilizada.

Es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privados.

jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

AJAX

AJAX, acrónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*).

Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se solicitan al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página, aunque existe la posibilidad de configurar las peticiones como síncronas de tal forma que la interactividad de la página se detiene hasta la espera de la respuesta por parte del servidor.

JavaScript es el lenguaje interpretado en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante *XMLHttpRequest*, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dado que está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

JSON

JSON, acrónimo de *JavaScript Object Notation*, es un formato ligero para el intercambio de datos.

Es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

La simplicidad de JSON ha dado lugar a la generalización de su uso, especialmente como alternativa a XML en AJAX.

Una de las supuestas ventajas de JSON sobre XML como formato de intercambio de datos en este contexto es que es mucho más sencillo escribir un analizador sintáctico (parser) de JSON. En JavaScript, un texto JSON se puede analizar fácilmente usando la función `eval()`, lo cual ha sido fundamental para que JSON haya sido aceptado por parte de la comunidad de desarrolladores AJAX, debido a la ubicuidad de JavaScript en casi cualquier navegador web.

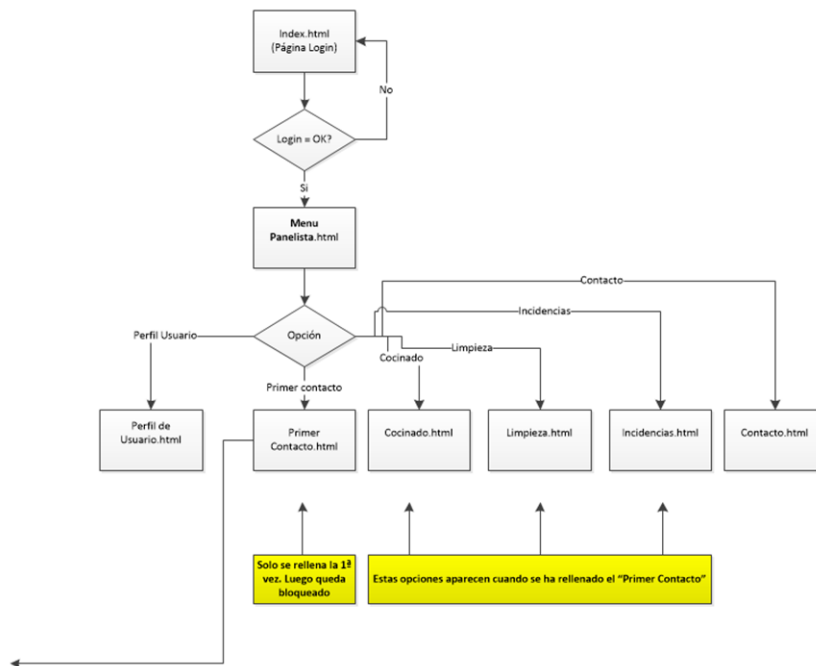


Una de las principales ventajas es su facilidad de uso, ya se basa en atributos clave – valor. Por ejemplo en mi caso, que era el de traducir términos, tendremos la clave a traducir y luego los idiomas en los cuales queremos tener dicha traducción.

```
"home": [{  
    "en": "HOME",  
    "es": "INICIO"  
}]
```

Arquitectura de la aplicación

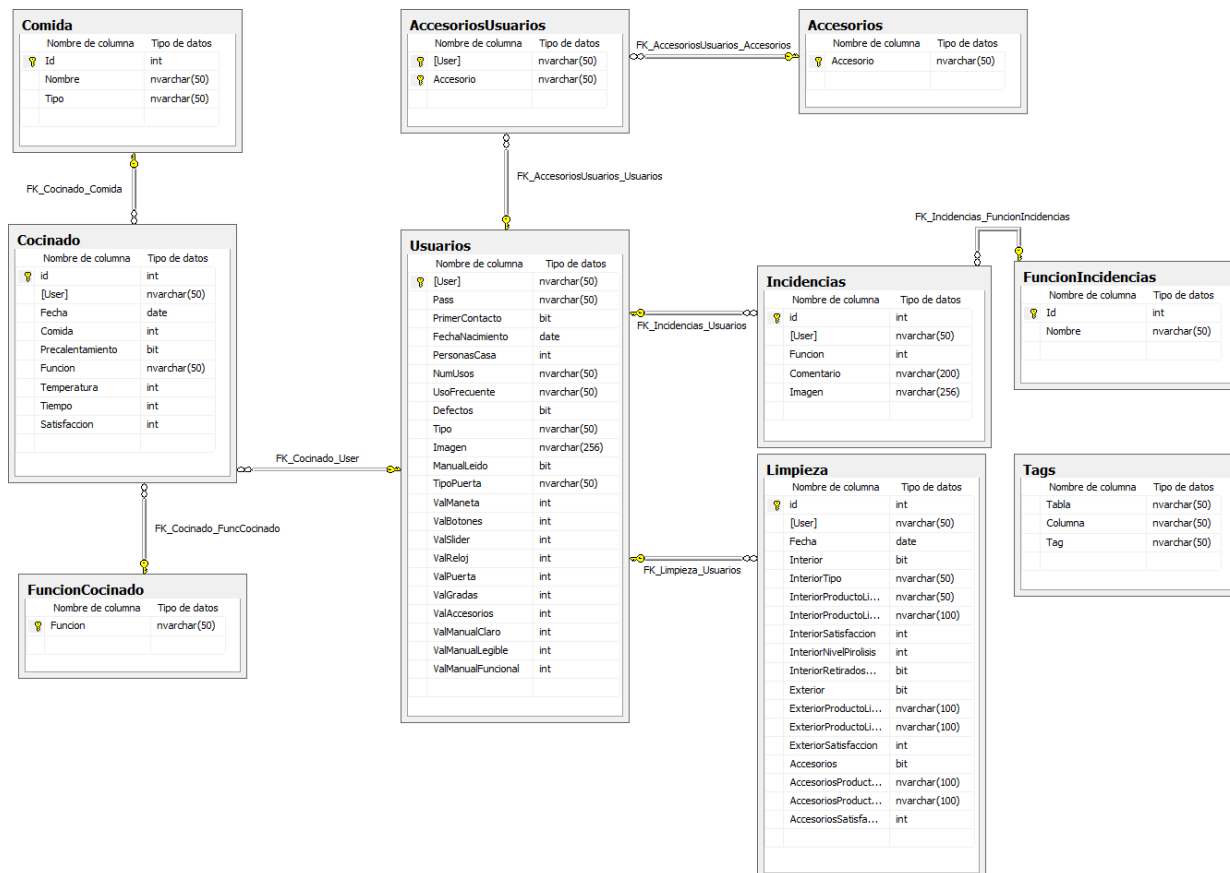
Esquema del diseño





Modelo de datos

Esquema de la base de datos, incluyendo relaciones entre las tablas.



Tendremos la tabla principal de los usuarios, y luego a cada usuario le asociamos el cocinado, las incidencias, limpieza el primer contacto y su perfil de usuario.

Luego para obtener determinadas etiquetas para rellenar los campos que tendremos en algunos de los “Selects” disponemos de una tabla llamada “Tags” que es la encargada mediante servicios web los datos que contendrán los selects.

SERVICIOS WEB

- TAGS

Para facilitar el multiidioma de la aplicación, los textos de los “combobox” de la aplicación se guardarán como etiquetas en la base de datos en la tabla “Tags”. De esta forma, para rellenar un “combobox”, habría que seleccionar de la tabla “Tags” las etiquetas correspondientes para posteriormente asignarles la traducción ya desde la aplicación.

- LOGIN

Esta realizará el login de la app utilizando un usuario y contraseña.

- LOGOUT

Función usada para cerrar la sesión del usuario logeado.

- SET USUARIO

Registra o actualiza, en el caso de ya existir el usuario, la información de un usuario en la base de datos.

- GET INCIDENCIAS

Obtiene de la base de datos las incidencias introducidas por un usuario.

- SET INCIDENCIA

Inserta en la base de datos una incidencia introducida por un usuario.

- GET ACCESORIOS USUARIO

Obtiene de la base de datos los accesorios que tiene asignado un usuario.

- SET ACCESORIOS USUARIO

Inserta en la base de datos un accesorio asignado un usuario.

- GET ACCESORIOS

Obtiene de la base de datos los accesorios disponibles.

- GET COCINADO

Obtiene de la base de datos los alimentos cocinados por un usuario.

- SET COCINADO

Inserta en la base de datos un alimento cocinado por un usuario.



- GET COMIDAS

Obtiene de la base de datos los tipos de comida disponibles.

- GET FUNCIONES COCINADO

Obtiene de la base de datos las funciones de cocinado.

- GET FUNCIONES INCIDENCIAS

Obtiene de la base de datos las funciones de incidencias.

- GET LIMPIEZA

Obtiene de la base de datos los históricos de limpieza de un usuario.

- SET LIMPIEZA

Guarda en la base de datos un registro de limpieza de un usuario.



Vistas

B/S/H/

[ENTRAR](#) [RECUPERAR CONTRASEÑA](#)

AUTIS INGENIEROS

AUTIS SALIR INICIO PERFIL USUARIO PRIMER CONTACTO COCINADO LIMPIEZA INCIDENCIAS CONTACTO

B/S/H/

PERFIL USUARIO

PRIMER CONTACTO

COCINADO

LIMPIEZA


INCIDENCIAS

CONTACTO

AUTIS INGENIEROS




AUTIS SALIR INICIO **PERFIL USUARIO** PRIMER CONTACTO COCINADO LIMPIEZA INCIDENCIAS CONTACTO



PERFIL USUARIO

EDAD	PERSONAS EN CASA
13/09/2015	2
USOS	USO FRECUENTE DEL HORNO
1-2/MES	REPOSTERIA


GUARDAR VOLVER ATRÁS

AUTIS INGENIEROS 

AUTIS SALIR INICIO **PERFIL USUARIO** **PRIMER CONTACTO** COCINADO LIMPIEZA INCIDENCIAS CONTACTO

PRIMER CONTACTO CON EL HORNO

¿APRECIA DEFECTOS EN EL APARATO?	TIPO DE INSTALACIÓN
SI	COLUMNA

 ENVIAR INCIDENCIA

¿HA LEÍDO EL MANUAL?	TIPO DE PUERTA	ACCESORIOS
SI	CARRO	BANDEJA SIMULTANEA BANDEJA PLANA BANDEJA PROFUNDA PARRILLA FIJA MEDIA BANDEJA PARRILLA

CLARO 1
LEGIBLE 2
FUNCIONAL 3

PRIMERA VALORACIÓN

MANETA	BOTONES	SLIDER	RELOJ
5	4	4	5

PUERTA **GRADAS** **ACCESORIOS**



COCINADO

¿QUÉ COCINAS HOY?

COMIDA	PRECALENTAMIENTO	FUNCIÓN
PUJÓN DE PESCADO	SI	CALOR INFERIOR
SATISFACCIÓN	TEMPERATURA	TIEMPO
5	TEMPERATURA	TIEMPO

GUARDAR VOLVER ATRÁS

RECETAS DEL MES

COMIDA	FECHA	PRECALENTAMIENTO	FUNCIÓN	TEMPERATURA	TIEMPO	SATISFACCIÓN
PUJÓN DE PESCADO	01/09/2015	NO	GRILL	190	20	2
PUJÓN DE VERDURAS	01/09/2015	NO	GRILL	190	25	0
CABALLA	01/09/2015	NO	GRILL	190	25	0
COSEBRO	01/09/2015	NO	GRILL	190	25	0
SARDINAS	01/09/2015	NO	GRILL	190	25	0



INCIDENCIAS

FUNCIÓN	COMENTARIOS	IMAGEN
CONDENACIONES		SELECC. IMAGEN

ENVIAR INCIDENCIA VOLVER ATRÁS

MIS INCIDENCIAS

INCIDENCIA	ID	COMENTARIO	IMAGEN
CONDENACIONES	#1	COMENTARIO 1	
CONDENACIONES	#1	COMENTARIO 1	
CONDENACIONES	#1	COMENTARIO INCIDENCIA	
HURDO	#5	NO ELIMINA EL HURDO	
OFICIAL	#11	OFICIAL	
LAMPARA	#9	NO FUNCIONA LA LAMPARA	
MUEBLES AMERIC	#10	DEFECTO EN EL MUEBLE AMERIC	



CONTACTO



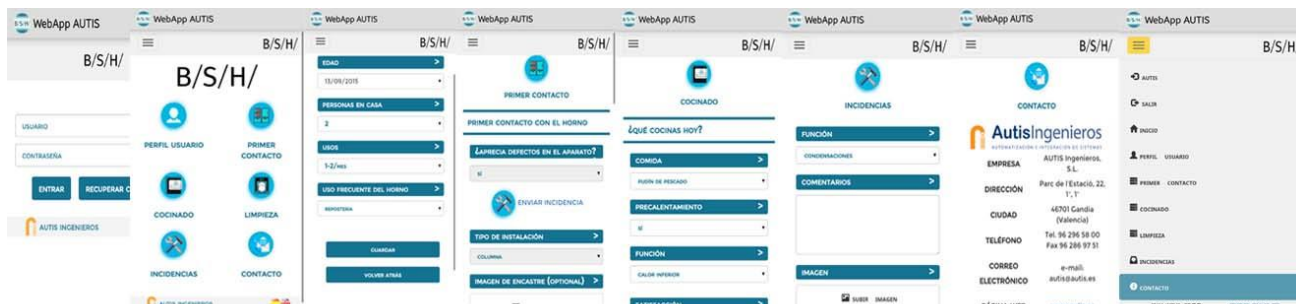
AUTOMATIZACIÓN E INTEGRACIÓN DE SISTEMAS

EMPRESA: AUTIS Ingenieros, S.L.
 DIRECCIÓN: Parc de l'Estació, 22, 1^a, 1^a
 CIUDAD: 46701 Gandia (Valencia)
 TELÉFONO: Tel. 96 296 58 00
 Fax: 96 286 97 51
 CORREO ELECTRÓNICO: e-mail: autis@autis.es
 PÁGINA WEB: www.autis.es





Versión móvil



Conclusiones

En conclusión he quedado satisfecho de la aplicación creada, he aprendido bastante más de cómo hacer sistemas Web Apps que utilicen servicios web, y que además puedan servir para más de una plataforma.

Aparte he tenido que bajar alguno que otro modulo para mejorar la presentación en los dispositivos móviles.

No obstante me hubiese gustado más hacer una aplicación completamente en Android para mejorar los conocimientos adquiridos durante todo el diploma.

Por otro lado he tenido que investigar sobre el funcionamiento de Bootstrap, el cual me ha gustado mucho utilizarlo ya que es una herramienta que facilita bastante el desarrollo de WebApps al desarrollador y por ello creo que me será de gran utilidad haberlo aprendido para seguir desarrollando WebApps con este framework.

Uno de los problemas con los que me he encontrado en el proyecto, ha sido al final y es que funciona sobre todos los navegadores perfectamente menos en el Chrome y el Webview nativo de Android, que para que funcione perfectamente debe estar corriendo sobre un servidor, ya que estos navegadores por motivos de seguridad no dejan ejecutar Cookies de JQuery ni abrir ficheros .json cuando las rutas de ejecución son del tipo **file:///C:/Users/-----/autisv3/index.html** o rutas que se denominan locales.

Anexos

La aplicación de EOX5 deberá contemplar el multiidioma.

Es por ello que se deberá gestionar con algún mecanismo externo a la BBDD la traducción de los textos. Por defecto en la BBDD se guardarán los Tags de los textos.

Por ejemplo: Para la tabla Comidas se devuelven campos con el formato {"Id":3,"Nombre":"Fish pudding","Tipo":"Starters"}

A la hora de rellenar el desplegable con estos textos, se deberá revisar el texto configurado en la app y realizar la traducción basándose en un fichero que funcionará como un diccionario y que podrá crecer en idiomas tanto como se requiera en un futuro.

2. TAGS

Descripción:

Para facilitar el multiidioma de la aplicación, los textos de los “combobox” de la aplicación se guardarán como etiquetas en la base de datos en la tabla “Tags”. De esta forma, para rellenar un “combobox”, habría que seleccionar de la tabla “Tags” las etiquetas correspondientes para posteriormente asignarles la traducción ya desde la aplicación.

URL: http://data.psfview.com:8000/FuncionesEOX5/Get_Tags

Parámetros de Entrada (POST):

- Tabla: Tabla en la cual se aplica la etiqueta.
- Columna: columna a la cual se le aplica la etiqueta.

Parámetros de Salida:

- Tag: Vector de etiquetas

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Get_Tags

Tabla: Usuarios

Columna: NumUsos

Resultado Ejemplo:

["MAS_1_SEMANA","1_SEMANA","1_2_MES"]



3. LOGIN

Descripción:

Esta realizará el login de la app utilizando un usuario y contraseña.

URL:

<http://data.psfview.com:8000/FuncionesEOX5/Login>

Parámetros de Entrada (POST):(de momento no se van a utilizar)

- User: Usuario.
- Pass: Contraseña.

Parámetros de Salida:

- User.
- Pass.
- PrimerContacto (true o false).
- FechaNacimiento.
- PersonasCasa.
- NumUsos (Tipo Tag).
- UsoFrecuente (Tipo Tag).
- Defectos (true o false).
- Tipo (Tipo Tag).
- Imagen: Path de la imagen.
- ManualLeido (true o false).
- TipoPuerta (Tipo Tag).
- ValManeta.
- ValBotones.
- ValSlider.
- ValReloj.



- ValPuerta.
- ValGradas.
- ValAccesorios.

Ejemplo Llamada:

<http://data.psfview.com:8000/FuncionesEOX5/Login>

User: Autis

Pass: autis1234

Resultado Ejemplo:

```
{ "User": "Autis", "Pass": "autis1234", "PrimerContacto": false, "FechaNacimiento": "2015-09-02", "PersonasCasa": 2, "NumUsos": "MAS_1_SEMANA", "UsoFrecuente": "COCINADO", "Defectos": false, "Tipo": "BAJO_BALDA", "Imagen": "C:\\Ruta\\Imagen.jpg", "ManualLeido": false, "TipoPuerta": "CARRO", "ValManeta": 4, "ValBotones": 5, "ValSlider": 5, "ValReloj": 4, "ValPuerta": 3, "ValGradas": 5, "ValAccesorios": 4, "ValManualClaro": 4, "ValManualLegible": 4, "ValManualFuncional": 4 }
```

4. LOGOUT

Descripción:

Función usada para cerrar la sesión del usuario logeado.

URL:

<http://data.psfview.com:8000/FuncionesEOX5/Logout>

Parámetros de Salida:

- status: "true" si hay error, "false" si no lo hay.
- code: Código del error.
- source: Descripción del error.

Ejemplo Llamada:

<http://data.psfview.com:8000/FuncionesEOX5/Logout>



Resultado Ejemplo:

```
{"status":false,"code":0,"source":""}
```

5. SET USUARIO

Descripción:

Registra o actualiza, en el caso de ya existir el usuario, la información de un usuario en la base de datos.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Set_Usuario

Parámetros de Entrada (POST):

- User.
- Pass.
- PrimerContacto (true o false).
- FechaNacimiento.
- PersonasCasa.
- Usos (Tipo Tag).
- UsoFrecuente (Tipo Tag).
- Defectos (true o false).
- Tipo (Tipo Tag).
- Imagen: Path de la imagen.
- ManualLeido (true o false).
- TipoPuerta (Tipo Tag).
- ValManeta.
- ValBotones.
- ValSlider.
- ValReloj.
- ValPuerta.



- ValGradas.
- ValAccesorios.
- ValManualClaro
- ValManualLegible
- ValManualFuncional

Parámetros de Salida:

- affected rows: Número de filas afectadas en la base de datos.
- error out: Estructura de error.

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Set_Usuario

User: 'Autis'

Pass: 'autis1234'

PrimerContacto: 'TRUE'

FechaNacimiento: '2015-09-02'

PersonasCasa: 'MAS_6'

Usos: '1_SEMANA'

UsoFrecuente: 'COCINADO'

Defectos: 'FALSE'

Tipo: 'COLUMNA'

Imagen: 'C:\Ruta\Imagen.jpg'

ManualLeido: 'FALSE'

TipoPuerta: 'CARRO'

ValManeta: 5

ValBotones: 4

ValSlider: 5

ValReloj: 3



ValPuerta:4

ValGradas:3

ValAccesorios:4

ValManualClaro:4

ValManualLegible:4

ValManualFuncional:4

Resultado Ejemplo:

```
{"affected rows":1,"error out":{"status":false,"code":0,"source":""}}
```

6. GET INCIDENCIAS

Descripción:

Obtiene de la base de datos las incidencias introducidas por un usuario.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Get_Incidencias

Parámetros de Entrada (POST):

- User: Usuario.

Parámetros de Salida (Vector):

- User.
- FuncionIncidencias (Estructura).
 - id.
 - Nombre (Tipo Tag).
- Comentario.
- Imagen: Path de la imagen.

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Get_Incidencias

User: Autis



Resultado Ejemplo:

```
[{"User":"Autis","FuncionIncidencias":{"Id":1,"Nombre":"CONDENSACIONES"},"Comentario":"comentario1","Imagen":"C:\\Ruta\\imagen.jpg"}]
```

7. SET INCIDENCIA

Descripción:

Inserta en la base de datos una incidencia introducida por un usuario.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Set_Incidencia

Parámetros de Entrada (POST):

- User.
- Funcion: Id que identifica un tipo de incidencia en la tabla "FuncionIncidencias".
- Comentario.
- Imagen: Path de la imagen.

Parámetros de Salida:

- affected rows: Número de filas afectadas en la base de datos.
- error out: Estructura de error.

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Set_Incidencia

User:'Autis'

Funcion:'1'

Comentario:'Comentario Incidencia'

Imagen: 'C:\\Ruta\\Imagen.jpg'

Resultado Ejemplo:

```
{"affected rows":1,"error out":{"status":false,"code":0,"source":""}}
```



8. GET ACCESORIOS USUARIO

Descripción:

Obtiene de la base de datos los accesorios que tiene asignado un usuario.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Get_AccesoriosUsuario

Parámetros de Entrada (POST):

- User: Usuario.

Parámetros de Salida (Vector):

- User.
- Accesorio (Tipo Tag).

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Get_AccesoriosUsuario

User: 'Autis'

Resultado Ejemplo:

```
[{"User":"Autis","Accesorio":"BANDEJA_ESMALT"}, {"User":"Autis","Accesorio":"PARRILLA"}]
```

9. SET ACCESORIOS USUARIO

Descripción:

Inserta en la base de datos un accesorio asignado un usuario.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Set_AccesoriosUsuario

Parámetros de Entrada (POST):

- User.
- Accesorio (Tipo Tag).



Parámetros de Salida:

- affected rows: Número de filas afectadas en la base de datos.
- error out: Estructura de error.

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Set_AccesoriosUsuario

User: 'Autis'

Accesorio:'FLEX_GRILL'

Resultado Ejemplo:

```
{"affected rows":1,"error out":{"status":false,"code":0,"source":""}}
```

10. GET ACCESORIOS

Descripción:

Obtiene de la base de datos los accesorios disponibles.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Get_Accesorios

Parámetros de Entrada:

Parámetros de Salida (Vector):

- Vector Strings (Tipo Tag)

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Get_Accesorios

Resultado Ejemplo:

```
["BANDEJA_ESMALT","FLEX_GRILL","PARRILLA"]
```



11. GET COCINADO

Descripción:

Obtiene de la base de datos los alimentos cocinados por un usuario.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Get_Cocinado

Parámetros de Entrada (POST):

- User: Usuario.

Parámetros de Salida (Vector):

- User.
- Date: Año-Mes-Día.
- Comida (Estructura).
 - id.
 - Nombre (Tipo Tag).
 - Tipo (Tipo Tag).
- Precalentamiento (true o false).
- Funcion.
- Temperatura.
- Tiempo.
- Satisfaccion (Tipo Tag).

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Get_Cocinado

User: 'Autis'

Resultado Ejemplo:



```
[{"User":"Autis","Date":"2015-09-01","Comida":{"Id":1,"Nombre":"Mero","Tipo":"Pescado"},"Precalentamiento":false,"Funcion":"GRILL","Temperatura":190,"Tiempo":20,"Satisfaccion":2}]
```

12. SET COCINADO

Descripción:

Inserta en la base de datos un alimento cocinado por un usuario.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Set_Cocinado

Parámetros de Entrada (POST):

- User.
- Fecha: Año-Mes-Día.
- Comida: Id que identifica un tipo de comida en la tabla "Comida".
- Precalentamiento (true o false).
- Funcion.
- Temperatura.
- Tiempo.
- Satisfaccion (Tipo Tag).

Parámetros de Salida (Vector):

- affected rows: Número de filas afectadas en la base de datos.
- error out: Estructura de error.

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Set_Cocinado

User: 'Autis'

Fecha:'2015-09-01'

Comida:'1'



Precaentamiento:'FALSE'

Funcion:'GRILL'

Temperatura:'190'

Tiempo:'25'

Satisfaccion:'3'

Resultado Ejemplo:

```
{"affected rows":1,"error out":{"status":false,"code":0,"source":""}}
```

13. GET COMIDAS

Descripción:

Obtiene de la base de datos los tipos de comida disponibles.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Get_Comidas

Parámetros de Entrada:

Parámetros de Salida (Vector):

- id.
- Nombre (Tipo Tag).
- Tipo (Tipo Tag):Pescado, Carne, Pasta....

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Get_Comidas

Resultado Ejemplo:

```
[{"Id":1,"Nombre":"Mero","Tipo":"Pescado"}, {"Id":2,"Nombre":"Pollo","Tipo":"Carne"}]
```



14. GET FUNCIONES COCINADO

Descripción:

Obtiene de la base de datos las funciones de cocinado.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Get_FuncionesCocinado

Parámetros de Entrada:

Parámetros de Salida (Vector):

- Vector Strings (Tipo Tag)

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Get_FuncionesCocinado

Resultado Ejemplo:

["CALOR_INF","GRILL","GRILL_SUP_AMPL"]

15. GET FUNCIONES INCIDENCIAS

Descripción:

Obtiene de la base de datos las funciones de incidencias.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Get_FuncionesIncidencias

Parámetros de Entrada:

Parámetros de Salida (Vector):

- Id.
- Nombre (Tipo Tag).

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Get_FuncionesIncidencias



Resultado Ejemplo:

```
[{"Id":1,"Nombre":"CONDENSACIONES"}, {"Id":2,"Nombre":"VAPORES_ABRIR"}, {"Id":3,"Nombre":"HUMO"}]
```

16. GET LIMPIEZA

Descripción:

Obtiene de la base de datos el historicos de limpieza de un usuario.

URL:

http://data.psfview.com:8000/FuncionesEOX5/Get_Limpieza

Parámetros de Entrada (POST):

- User: Usuario.

Parámetros de Salida (Vector):

- User.
- Fecha.
- Interior (true o false): Indica si el guardado contiene información de la limpieza interior.
- InteriorTipo (Tipo Tag).
- InteriorProductoLimpieza (Tipo Tag).
- InteriorProductoLimpiezaOtros.
- InteriorSatisfaccion.
- InteriorNivelPirolisis.
- InteriorRetiradosResiduos.
- Exterior (true o false): Indica si el guardado contiene información de la limpieza exterior.
- ExteriorProductoLimpieza (Tipo Tag).
- ExteriorProductoLimpiezaOtros.
- ExteriorSatisfaccion .



- Accesorios (true o false): Indica si el guardado contiene información de la limpieza de Accesorios.
- AccesoriosProductoLimpieza (Tipo Tag).
- AccesoriosProductoLimpiezaOtros.
- AccesoriosSatisfaccion.

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Get_Limpieza

User: 'Autis'

Resultado Ejemplo:

```
[{"User":"Autis2","Date":"2015-09-16","Interior":false,"Interior Tipo":"Int Tipo","Interior Producto Limpieza":"Int Prod Limpieza","Interior Producto Limpieza Otros":"Int Prod Limpieza Otros","Interior Satisfaccion":5,"Interior Nivel Pirolysis":5,"Interior Retirados Residuos":true,"Exterior":true,"Exterior Producto Limpieza":"Ext Prod Limpieza","Exterior Producto Limpieza Otros":"Ext Prod Limpieza Otros","Exterior Satisfaccion":4,"Accesorios":false,"Accesorios Producto Limpieza":"Acc Prod Limpieza","Accesorios Producto Limpieza Otros":"Acc Prod Limpieza Otros","Accesorios Satisfaccion":3}]
```

17. SET LIMPIEZA

Descripción:

Guarda en la la base de datos un registro de limpieza de un usuario.

URL:http://data.psfview.com:8000/FuncionesEOX5/Set_Limpieza

Parámetros de Entrada (POST):

- User.
- Fecha.
- Interior (true o false): Indica si el guardado contiene información de la limpieza interior.
- InteriorTipo (Tipo Tag).
- InteriorProductoLimpieza (Tipo Tag).
- InteriorProductoLimpiezaOtros.



- InteriorSatisfaccion.
- InteriorNivelPirolisis.
- InteriorRetiradosResiduos.
- Exterior (true o false): Indica si el guardado contiene información de la limpieza exterior.
- ExteriorProductoLimpieza (Tipo Tag).
- ExteriorProductoLimpiezaOtros.
- ExteriorSatisfaccion .
- Accesorios (true o false): Indica si el guardado contiene información de la limpieza de Accesorios.
- AccesoriosProductoLimpieza (Tipo Tag).
- AccesoriosProductoLimpiezaOtros.
- AccesoriosSatisfaccion.

Parámetros de Salida (Vector):

- affected rows: Número de filas afectadas en la base de datos.
- error out: Estructura de error.

Ejemplo Llamada:

http://data.psfview.com:8000/FuncionesEOX5/Set_Limpieza

User : 'Autis2',
Fecha : '2015-09-16',
Interior : 'TRUE',
InteriorTipo : 'Int Tipo',
InteriorProductoLimpieza : 'Int Prod Limpieza',
InteriorProductoLimpiezaOtros : 'Int Prod Limpieza Otros',
InteriorSatisfaccion : 5,
InteriorNivelPirolisis : 5,



InteriorRetiradosResiduos : 'TRUE',
Exterior : 'TRUE',
ExteriorProductoLimpieza : 'Ext Prod Limpieza',
ExteriorProductoLimpiezaOtros: 'Ext Prod Limpieza Otros',
ExteriorSatisfaccion : 4,
Accesorios : 'FALSE',
AccesoriosProductoLimpieza : "",
AccesoriosProductoLimpiezaOtros : "",
AccesoriosSatisfaccion : ""

Resultado Ejemplo:

```
{"affected rows":1,"error out":{"status":false,"code":0,"source":""}}
```