



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

CAMPUS D'ALCOI

[Diseño del frontend del panel de control de un ayuntamiento]

MEMORIA PRESENTADA POR:
[Óscar Garrido Chueco]

GRADO DE [*Ingeniería Informática*]

Convocatoria de defensa: *[Junio 2017]*



Índice

	Página
Introducción	3
Concepto	4
Ejemplos base	5
Selección de Arquitectura	7
• Google Charts	8
• Google Maps	9
• Diagrama del proyecto	10
Proyecto	12
• Adquisición de datos	13
• Tráfico	15
• Historial	26
• Autobuses	34
• Información Meteorológica	38
• DashBoard	42
Valoraciones	48
Bibliografía	49



Introducción

Estamos en una época tecnológica, en la que cada día se intenta tener a los usuarios más conectados, y que es capaz de ofrecernos servicios multimedia a medida, sea cual sea nuestra necesidad. La conexión de todo con todos, eliminando fronteras e impedimentos a la información, tecnologías que nos permiten comunicarnos más fácilmente y obtener los recursos que necesitamos de la red de manera más cómoda y eficiente, son el día a día de nuestra sociedad.

Nos encontramos cada día con dispositivos conectados, coches conectados, electrodomésticos conectados, ciudades conectadas, términos como el "Big Data", la calidad de la información, el análisis de esa información, son términos muy sonados hoy en día y es el futuro de la tecnología de la información. A través de esa información recogida y realizando los diferentes estudios pertinentes, podemos mejorar infraestructuras, negocios y servicios.

De esta idea de conexión surge el concepto de este TFG, así como su idea principal, ser capaces de recoger distinta información de nuestra localidad, ser capaces de analizar la información que nos ofrecen los diferentes medios y plataformas, tratar esos datos, y mostrarlos de forma amena a diferentes instituciones, o usuarios, para así ofrecer la información que necesiten.



Concepto

El concepto de este TFG es facilitar una utilidad al ciudadano que le permita poder analizar la información de los diferentes dispositivos situados por Alcoi. Ser capaces de recoger, tratar y analizar esa información para poder mostrarla de una forma fácil y sencilla, una forma centralizada donde el usuario con pocas acciones pueda analizar una inmensa cantidad de datos de una forma rápida.

La Idea es ser capaces de generar una especie de DashBoard, es decir, un FROND END que muestre de una forma efectiva y eficaz, toda la información de la que disponemos, tratando toda la información de forma automatizada para el usuario.

El DashBoard o panel, constará de 4 apartados; La distinta información de tráfico de las diferentes entradas y salidas de Alcoi, generar diferentes Históricos para poder estudiar el tráfico en fechas pasadas de las diferentes entradas y salidas de la ciudad, mostrar la información de las diferentes paradas de Autobús y el tiempo restante en llegar de las diferentes líneas urbanas y por último, de mostrar el tiempo actual de la localidad, a través recoger los datos de los diferentes dispositivos repartidos por la ciudad y de mostrarlos a los diferentes usuarios de forma visual, para que pueda interactuar con ellos, o como método de información a nivel local de la localidad.

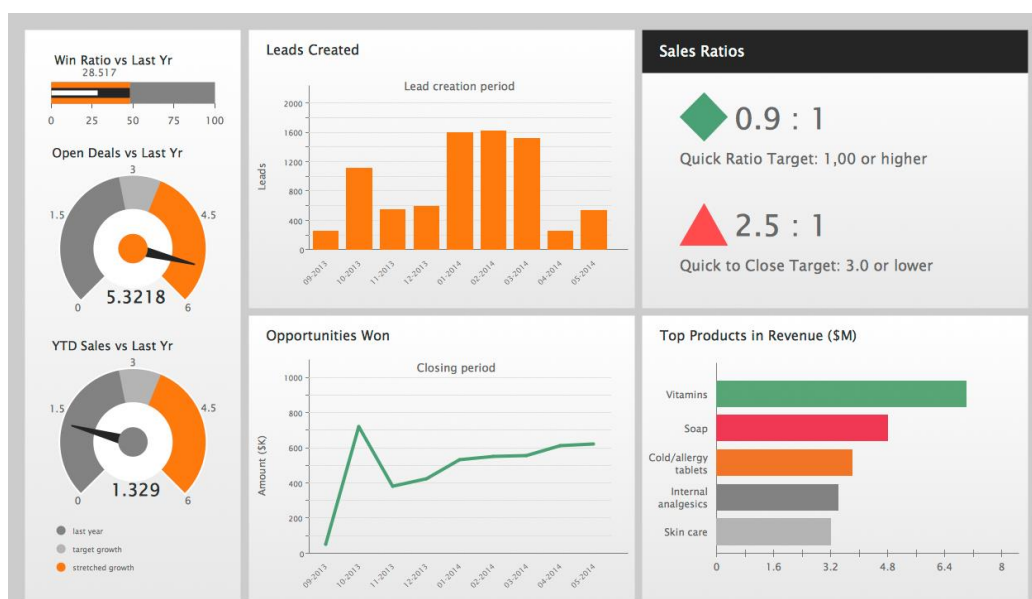
Agradecer los datos proporcionados por la Catedra Smart City, los proporcionados por AEMET de las estaciones meteorológicas, así como los datos de las paradas de Autobús proporcionados por Vectalia.com, y al Ajuntament D'Alcoi por proporcionar un Dominio y una infraestructura de servidor, con la cual, se ha podido confeccionar este proyecto. Gracias a todos.

Ejemplo base:

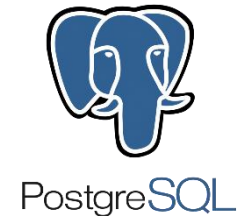
Partiendo de la idea de un Dashboard o la de un panel o sitio web, que muestre tal cantidad de información de forma gráfica y legible de las diferentes cámaras y dispositivos de Alcoi de los que disponemos, para cualquier usuario, se empezó a comparar ejemplos de diferentes Dashboards en la red.



Se encontraron una infinidad de Dashboards realizados en diferentes arquitecturas y servicios.



Pero todos derivaban a un tipo de arquitectura concreta, tratamiento de dato específico, con sus respectivos despliegues de servicios en la parte del servidor, y por lo tanto, la necesidad de disponer de servidor propio o con permiso total, para implementar las diferentes arquitecturas más novedosas. Como ejemplo, de arquitecturas en las que se ha basado este TFG para llegar a el concepto del mismo, tenemos a NodeJS, AngularJS y PostGreSQL.



No obstante, surgieron dificultades a la hora de adquirir documentación gratuita para realizar este TFG en las arquitecturas anteriormente mencionadas, como la dificultad recopilando información para realizar las diferentes funciones que se necesitaban para el enfoque del proyecto y la necesidad de salir de las diferentes plantillas por defecto que ofrecían las diferentes arquitecturas, por ello, se ha planteado la elección de una arquitectura diferente para este proyecto.

Selección de arquitectura.

La selección de la arquitectura del proyecto finalmente, se ha planteado desde el punto de vista académico, y de la no dependencia de lenguajes ni arquitecturas propietarias, tratando de conseguir el mayor grado de compatibilidad con arquitecturas OpenSource y facilitando la puesta en producción sobre servidores con configuraciones básicas.

Por ello se ha solicitado un dominio al Ajuntament D'Alcoi, con "phpMyAdmin" para así tratar los datos de forma deseada y con amplia compatibilidad en cualquier despliegue en cualquier servidor, y evitando la necesidad de utilizar lenguajes y arquitecturas propietarias en el lado del servidor.

Se utilizan finalmente lenguajes como JAVA, PHP, MYSQL, JavaScript, AJAX, HTML5, XML, CSS, JQuery y API de Google.




UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA
CAMPUS D'ALCOI



Dashboard Alcoi



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA
CAMPUS D'ALCOI



Si quiere ver el estado actual del Trafico Pulse [Aqui](#)



Si quieres ver el estado actual de los autobuses Pulse [Aqui](#)

Tabla de prediccion				
	2017-06-02	2017-06-03	2017-06-04	2017-06-05
Temperatura mínima y máxima	15 / 29	17 / 29	13 / 26	11 / 23
Sensación mínima y máxima	15 / 29	17 / 29	13 / 26	11 / 23
Humedad relativa mínima y máxima	40 / 65	35 / 65	45 / 100	35 / 100
Índice ultravioleta máxima	9	9	8	9

Fecha y Hora actual

06/02/2017 6:26pm
 Día actual del año:152
 Numero de días que quedan para finalizar el año: 213
 Cuantas semanas tiene el año:2017 - 52



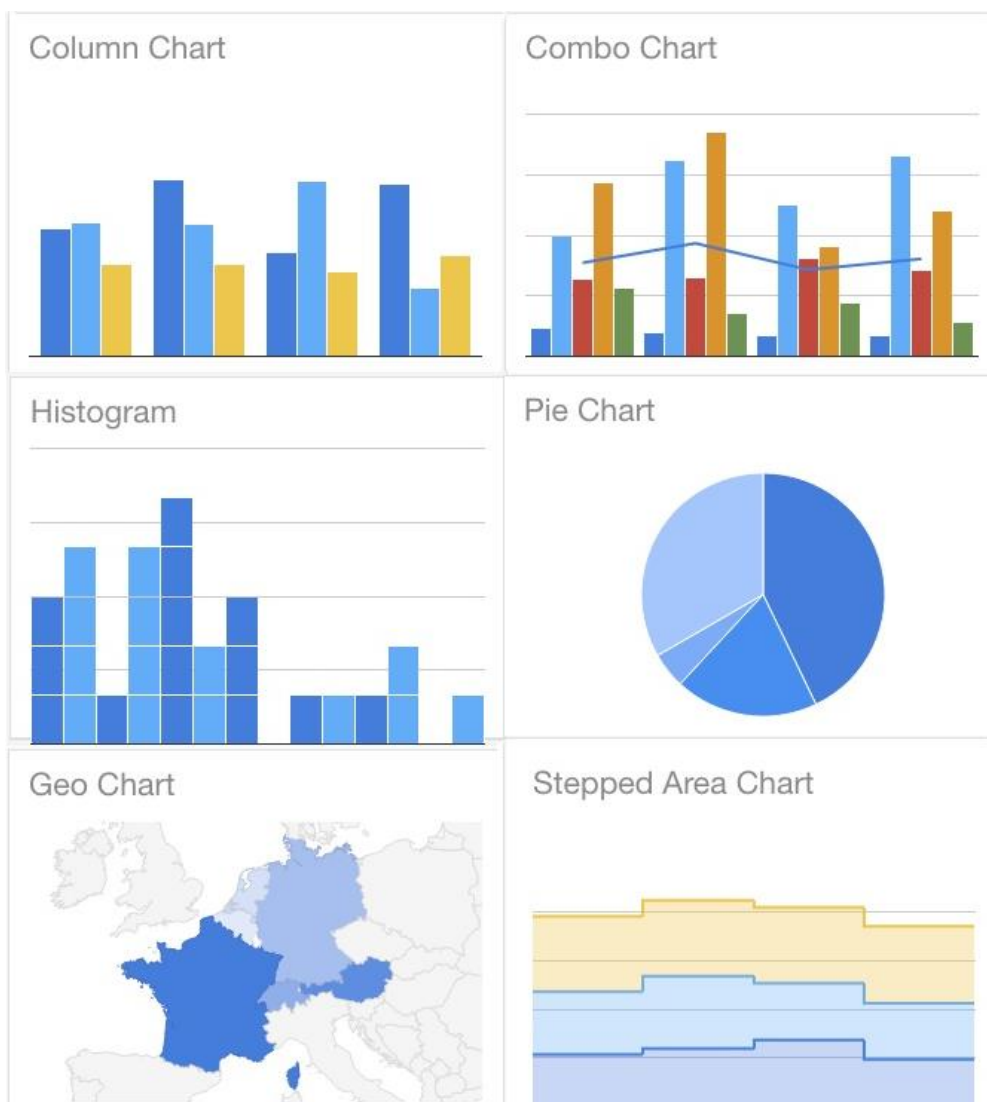
Si quiere ver el estado actual del Trafico Pulse [Aqui](#)

Imagen final "DashBoard Alcoi"

Google Charts.

Google Chart es una aplicación de Google para realizar estadísticas web, de fácil uso para desarrolladores de software web, usado en muchos campos como Google Analytics, se puede usar con diferentes formatos, Json, Javascript y plugins que se pueden integrar con varios lenguajes de programación.

Cargando esta utilidad desde la API de Google, podremos generar a través de los resultados de las consultas MySQL, los distintos gráficos con los que mostraremos los datos solicitados por los ciudadanos. Esta utilidad nos permite de una forma muy versátil, adaptar cada gráfico según nuestras necesidades y el tipo de información. Google Charts nos permite disponer de distintos modelos de gráficos. A continuación, mostramos algunos, aunque existen muchos más.



Google Maps.

Google Maps es un servidor de aplicaciones de mapas en la web que pertenece a Alphabet Inc. Ofrece imágenes de mapas desplazables, así como fotografías por satélite del mundo e incluso la ruta entre diferentes ubicaciones o imágenes a pie de calle con Google Street View.

Cargando esta utilidad desde la API de Google, se mostrarán los distintos marcadores, gráficas y estadísticas, como el nivel de tráfico de la ciudad, dependiendo de los resultados de las consultas MySQL, para así mostrar los distintos marcadores e información según los datos obtenidos. Esta utilidad nos permite de una forma muy versátil, adaptar cada mapa según nuestras necesidades y el tipo de información que se desea mostrar.

Google Maps nos ofrece distintas vistas del mapa y forma de visualizar los datos de los diferentes marcadores, además de información como gasolineras, teatros, centros educativos, centros comerciales, tiendas, sitios de interés, etc.

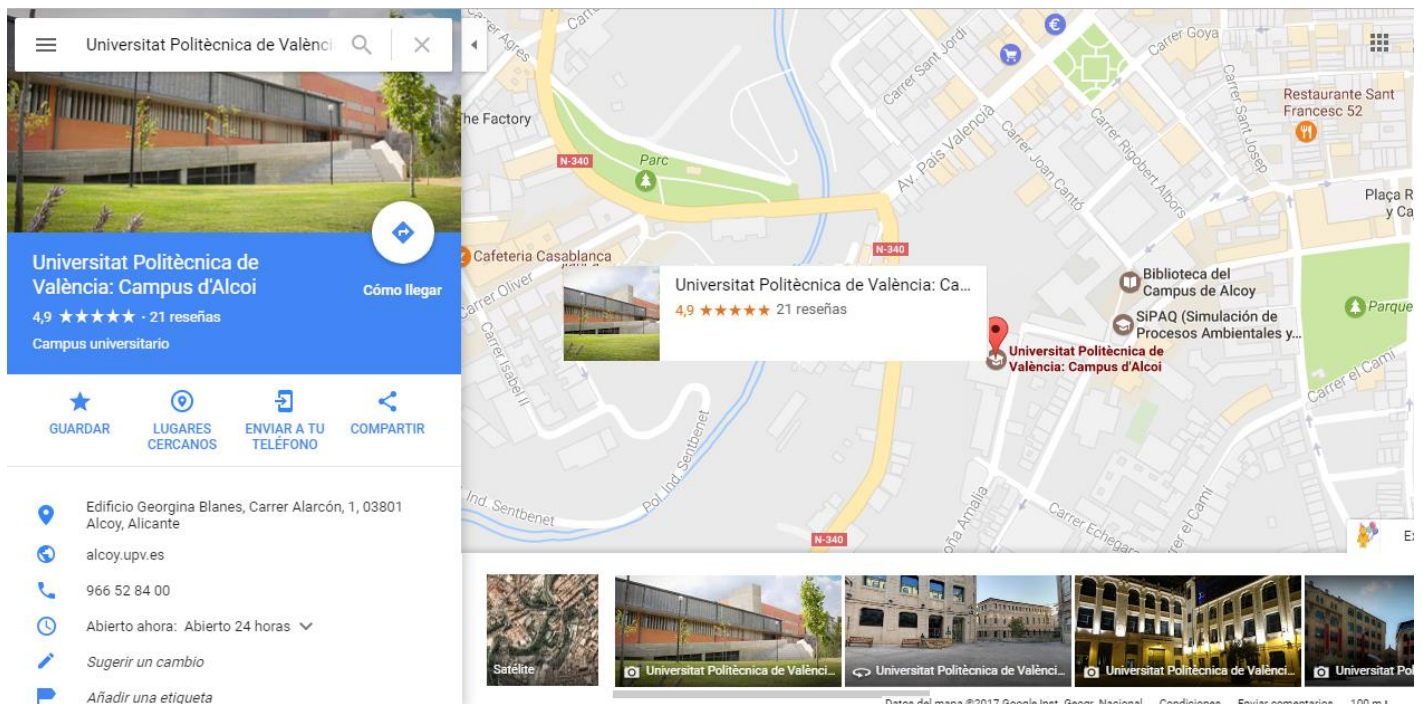


Diagrama de proyecto.

Vista de pantalla final de Contenidos:

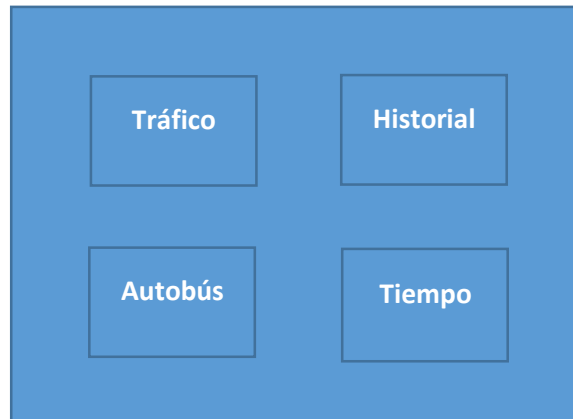


Diagrama general FROND END: CONTENIDOS TRAFICO e HISTORIAL:

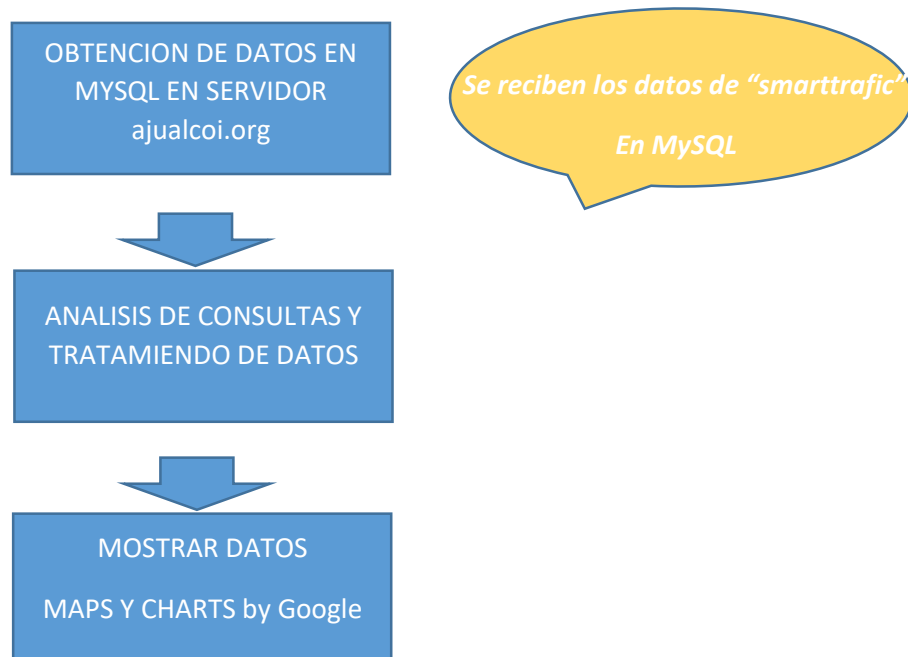
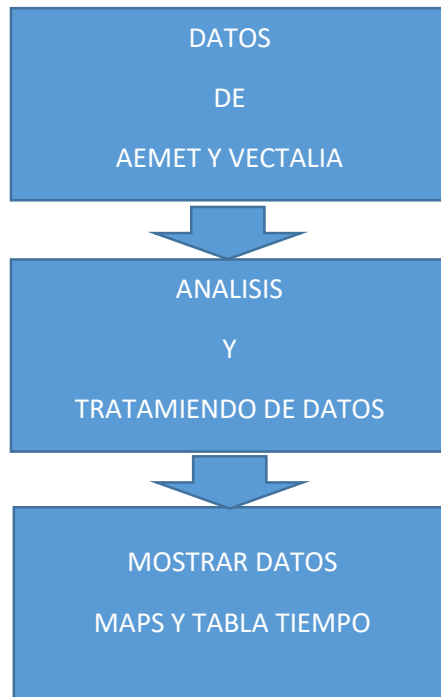
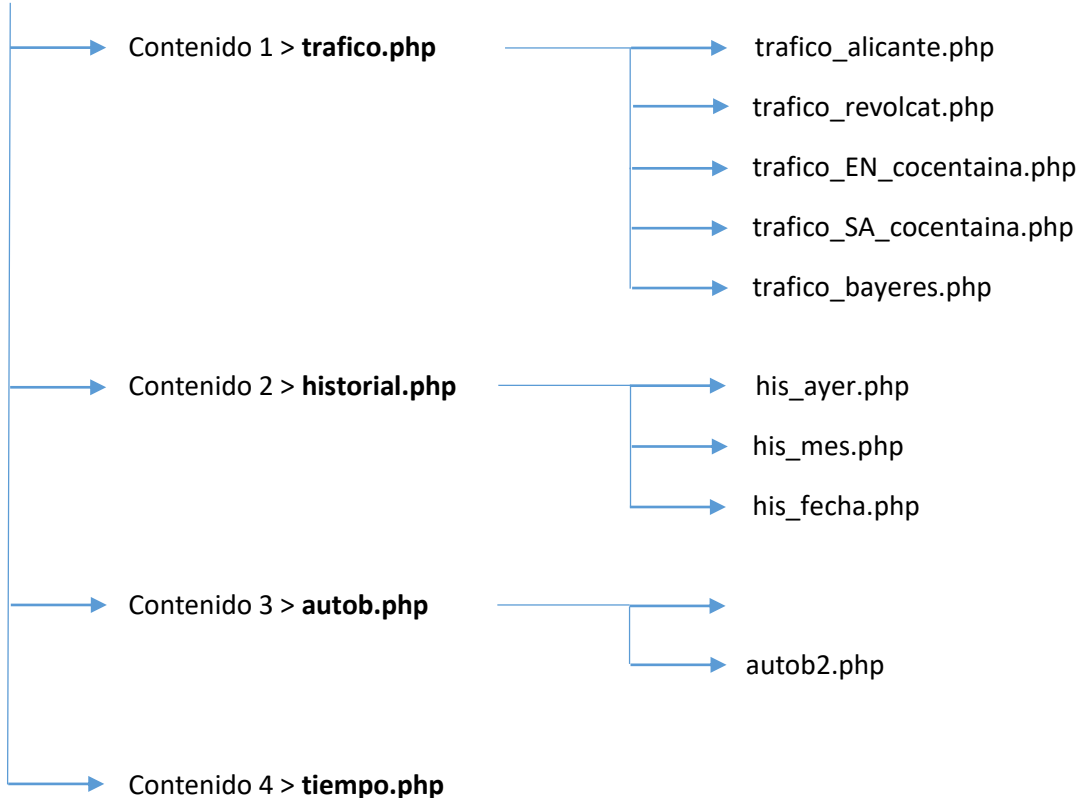


Diagrama general FROND END: CONTENIDOS AUTOBUS y TIEMPO:



Estructura general de ficheros:

index.html + CSS





Proyecto: DashBoard Alcoi

Partiendo del diagrama anteriormente expuesto, se han de localizar los datos relacionados con las cámaras en cada acceso de Alcoi, para analizar el tráfico y poder mostrarlo correctamente. Se cuenta con 5 cámaras, una en cada acceso de Alcoi, como son; el acceso de Alicante, el acceso al Revolcat, el acceso de Banyeres de Mariola, y la entrada y salida de Cocentaina con doble carril. Estos datos se adquieren a través de las cámaras, las cuales recogen los campos en orden correlativo: id registro (secuencial, se reinicia cada día), número reconocedor lpr, número cámara lpr, matrícula, fecha, hora, número de máscara, velocidad, dirección, coeficiente reconocimiento.

Quedando una estructura de datos: **2580,1,1,5490GLY,20-12-2016,10:11:31.462,1.0,0,Acercamiento,237**; almacenada en cada registro de la BD por cada coche que pasa por cualquier acceso, en una BD en PostgreSQL concreta llamada "smarttraffic" situada en un servidor externo.

A través de Manuel Llorca Alcón, Rafael Camps Jordà, responsable de las cámaras de los diferentes accesos de Alcoi, se proporcionó los datos necesarios para realizar la conexión SOCKET con la BD "smarttraffic" y migrar los datos a una BD en MYSQL.

Datos "smarttraffic" en PostgreSQL:

- i. nom bd: smarttraffic
- ii. user bd: upvuser
- iii. pass bd: alc0yUPV016US
- iv. ip servidor bd: 5.40.185.131
- v. port bd: 5433

Puerto de escucha del socket en el servidor: 16092

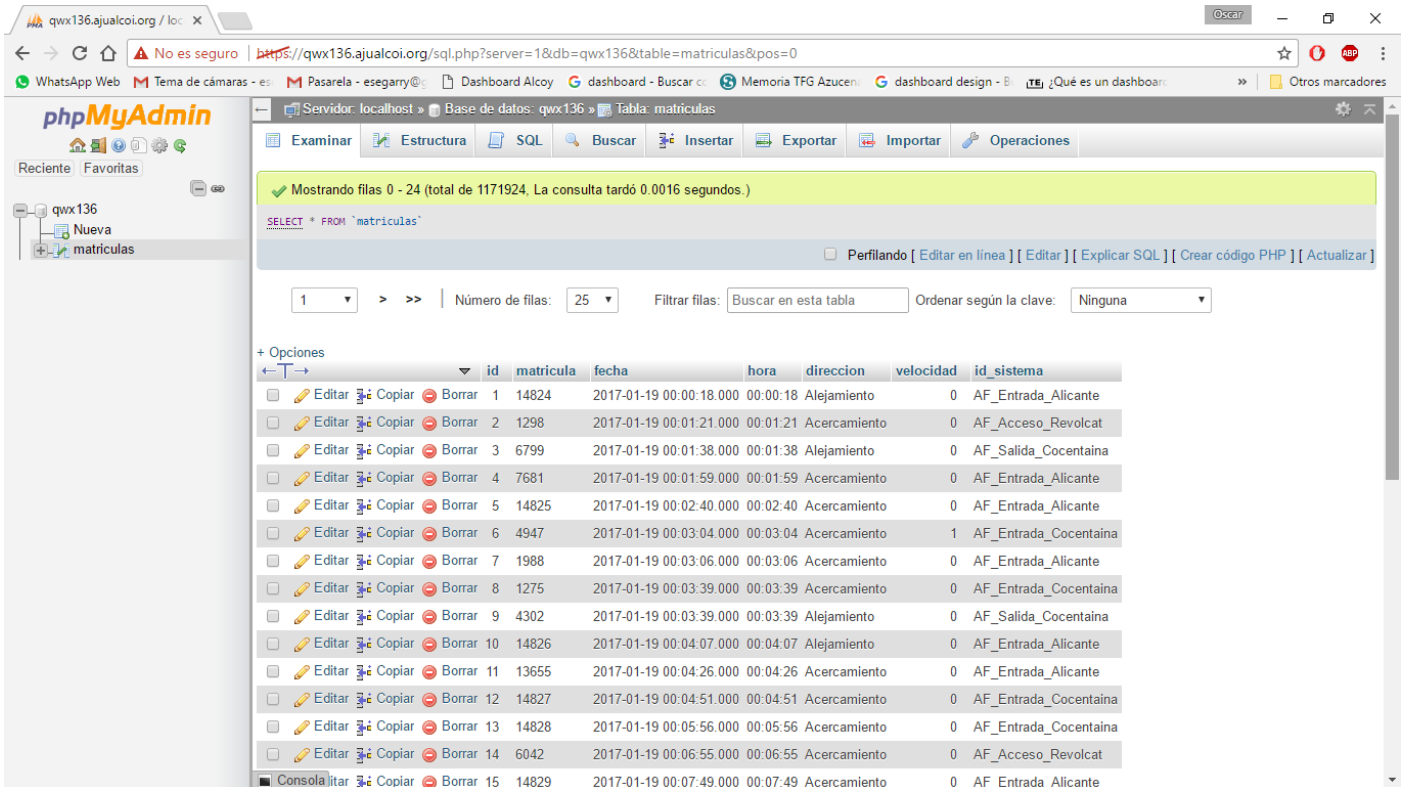
- i. 192.168.0.11 ALC01
- ii. 192.168.0.12 ALC02 Alacant
- iii. 192.168.0.13 ALC03 Cocentaina
- iv. 192.168.0.14 ALC04 Banyeres

Adquisición de datos

Debido a la seguridad requerida para acceder al servidor y leer la BD "smartraffic", y en cumplimiento de la LOPD, no está permitido trabajar directamente con los datos de la BD, ya que se debían dar los respectivos accesos a través de las diferentes capas de seguridad y, por otra parte, porque no se puede tratar los datos de las matrículas reales de cada automóvil por la LOPD.

Se utilizó una pasarela que transfiriera los datos de la BD "smartraffic" con arquitectura PostgreSQL del servidor externo, a una BD en MYSQL que se habilitó en el servidor del Ajuntament D'Alcoi, para así solventar por una parte el problema de acceso a la BD de "smartraffic", ya que se necesitaba acceso para poder tratar los datos, y por otro lado que se encargara de modificar la matrícula de cada registro y ser capaz de poner una máscara con un ID único e irrepitable para cada matrícula, para así poder tratar los datos sin infringir la ley de protección de datos.

La pasarela se desplegó en el lado del servidor externo, para que mediante conexión SOCKET y modificando antes los datos del campo matrícula, se inyectaran los datos de "smartraffic" a la BD de MYSQL del dominio del Ajuntament D'Alcoi. Una vez se empezó a recibir los datos de "smartraffic" a la BD "matriculas", con la máscara actuando sobre el campo matrículas, ya se pudo empezar a tratar los datos de una manera más segura, y sin problemas a lo que la LOPD se refiere.



The screenshot shows the phpMyAdmin interface for a database named 'qwx136'. The table 'matriculas' is selected, and the SQL query 'SELECT * FROM `matriculas`' is displayed. The table contains 15 rows of data, each with columns for 'id', 'matricula', 'fecha', 'hora', 'direccion', 'velocidad', and 'id_sistema'. The data includes various matricula numbers and timestamps from 2017-01-19.

	id	matricula	fecha	hora	direccion	velocidad	id_sistema
<input type="checkbox"/>	1	14824	2017-01-19 00:00:18.000	00:00:18	Alejamiento	0	AF_Entrada_Alicante
<input type="checkbox"/>	2	1298	2017-01-19 00:01:21.000	00:01:21	Acercamiento	0	AF_Acceso_Revolcat
<input type="checkbox"/>	3	6799	2017-01-19 00:01:38.000	00:01:38	Alejamiento	0	AF_Salida_Cocentaina
<input type="checkbox"/>	4	7681	2017-01-19 00:01:59.000	00:01:59	Acercamiento	0	AF_Entrada_Alicante
<input type="checkbox"/>	5	14825	2017-01-19 00:02:40.000	00:02:40	Acercamiento	0	AF_Entrada_Alicante
<input type="checkbox"/>	6	4947	2017-01-19 00:03:04.000	00:03:04	Acercamiento	1	AF_Entrada_Cocentaina
<input type="checkbox"/>	7	1988	2017-01-19 00:03:06.000	00:03:06	Acercamiento	0	AF_Entrada_Alicante
<input type="checkbox"/>	8	1275	2017-01-19 00:03:39.000	00:03:39	Acercamiento	0	AF_Entrada_Cocentaina
<input type="checkbox"/>	9	4302	2017-01-19 00:03:39.000	00:03:39	Alejamiento	0	AF_Salida_Cocentaina
<input type="checkbox"/>	10	14826	2017-01-19 00:04:07.000	00:04:07	Alejamiento	0	AF_Entrada_Alicante
<input type="checkbox"/>	11	13655	2017-01-19 00:04:26.000	00:04:26	Acercamiento	0	AF_Entrada_Alicante
<input type="checkbox"/>	12	14827	2017-01-19 00:04:51.000	00:04:51	Acercamiento	0	AF_Entrada_Cocentaina
<input type="checkbox"/>	13	14828	2017-01-19 00:05:56.000	00:05:56	Acercamiento	0	AF_Entrada_Cocentaina
<input type="checkbox"/>	14	6042	2017-01-19 00:06:55.000	00:06:55	Acercamiento	0	AF_Acceso_Revolcat
<input type="checkbox"/>	15	14829	2017-01-19 00:07:49.000	00:07:49	Acercamiento	0	AF_Entrada_Alicante

Ya se dispone de forma indirecta de todos los datos de la BD “smarttraffic” del servidor externo, con el campo de “matricula” modificado a través de una máscara única, con la información de cada acceso, por lo tanto, ya se puede realizar la siguiente fase, para efectuar el contenido de trafico.php y historial.php analizando y construyendo las diferentes consultas MySQL.

Contenido 1: Tráfico.

Ya se han obtenido los datos de los diferentes accesos de la ciudad en la BD MySQL proporcionada por el Ajuntament D'Alcoi, y en este apartado se ha de realizar un estudio de los coches que entran y salen por cada acceso durante cada hora, y que a través de un tipo de "semáforo" visualizado en un mapa de Google de la localidad, analice la densidad de tráfico y muestre al usuario, el nivel de tráfico actual, la media de coches por minuto, y una gráfica del tráfico de cada hora de cada acceso. Se utilizan diferentes recursos en diferentes lenguajes de programación, como son PHP/JS/MYSQL y HTML.

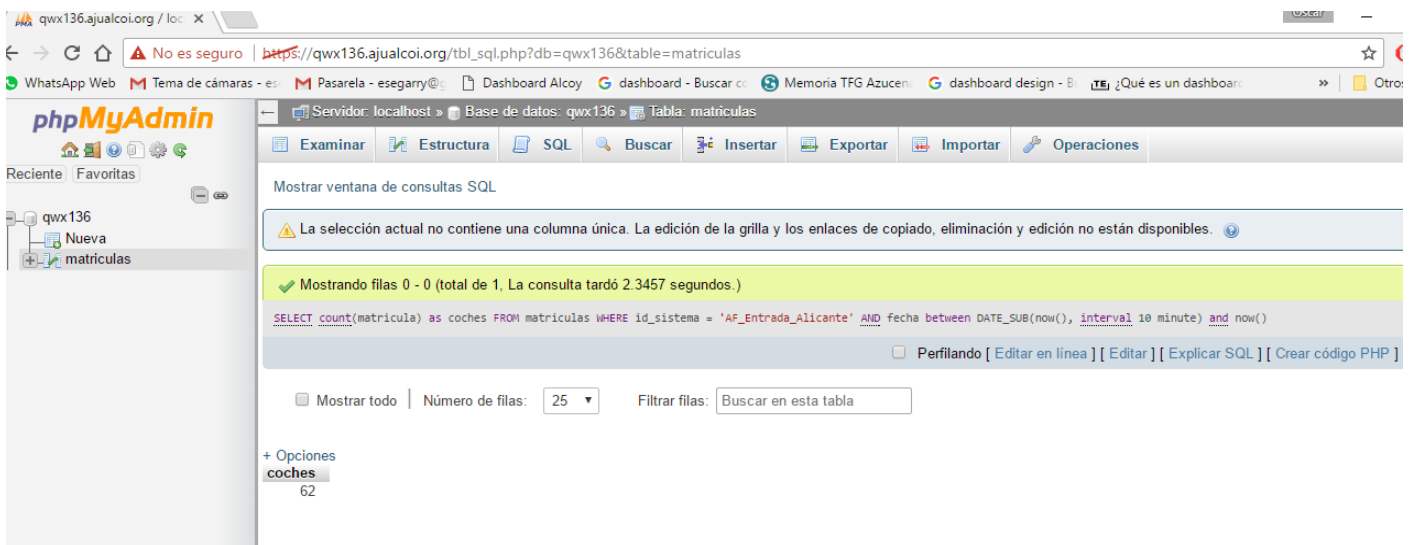
Por un lado, se ha de tener clara la conexión a la BD y la consulta MySQL que se ha de realizar para que muestre los datos que se desean de la densidad de tráfico, una vez realizada, se tratan los datos para mostrarlos correctamente.

Imagen de conexión

```
<?php
// Conectamos base de datos
// Datos del servidor
$nombrebdd = "qwx136";
$servidor = "qwx136.ajualcoi.org";
$usuario = "qwx136";
$password = "MyDash2016";
//conexion

$conexion = mysql_connect($servidor,$usuario,$password)or die("Error:".mysql_error());
mysql_select_db($nombrebdd,$conexion)or die("Error:".mysql_error());
//variables
```

Imagen del resultado de la consulta en phpMyAdmin donde se muestran el número de coches que han pasado los últimos 10 min por la entrada de Alicante.





A través de la consulta:

```
SELECT count(matricula) as coches FROM matriculas WHERE id_sistema = 'AF_Entrada_Alicante' AND direccion='Acercamiento' AND fecha between DATE_SUB(now(), interval 10 minute) and now();
```

Se calculan los coches que han pasado en este caso por el acceso 'AF_Entrada_Alicante' dirección 'Acercamiento' desde los últimos 10 min hasta ahora. Realizando la misma consulta con diferente id_sistema y diferente dirección, se obtienen los datos de los coches que han entrado y salido por los diferentes accesos de Alcoi.

```
//preparamos la consulta

$$SQLDatos = "SELECT count(matricula) as coches FROM matriculas WHERE id_sistema = 'AF_Entrada_Alicante'AND direccion='Acercamiento' AND fecha between DATE_SUB(now(), interval 10 minute) and now()";

$$SQLDatos_1 = "SELECT count(matricula) as coches FROM matriculas WHERE id_sistema = 'AF_Entrada_Alicante'AND direccion='Alejamiento' AND fecha between DATE_SUB(now(), interval 10 minute) and now()";

$$SQLDatos1 = "SELECT count(matricula) as coches FROM matriculas WHERE id_sistema = 'AF_Entrada_Cocentaina' AND fecha between DATE_SUB(now(), interval 10 minute) and now()";

$$SQLDatos2 = "SELECT count(matricula) as coches FROM matriculas WHERE id_sistema = 'AF_Salida_Cocentaina' AND fecha between DATE_SUB(now(), interval 10 minute) and now()";

$$SQLDatos3 = "SELECT count(matricula) as coches FROM matriculas WHERE id_sistema = 'AF_Acceso_Revolcat' AND direccion='Acercamiento' AND fecha between DATE_SUB(now(), interval 10 minute) and now()";

$$SQLDatos3_1 = "SELECT count(matricula) as coches FROM matriculas WHERE id_sistema = 'AF_Acceso_Revolcat' AND direccion='Alejamiento' AND fecha between DATE_SUB(now(), interval 10 minute) and now()";

$$SQLDatos4 = "SELECT count(matricula) as coches FROM matriculas WHERE id_sistema = 'AF_Acceso_Banyeres' AND direccion='Acercamiento' AND fecha between DATE_SUB(now(), interval 10 minute) and now()";

$$SQLDatos4_1 = "SELECT count(matricula) as coches FROM matriculas WHERE id_sistema = 'AF_Acceso_Banyeres' AND direccion='Alejamiento' AND fecha between DATE_SUB(now(), interval 10 minute) and now()";
```

Se efectúan todas las consultas pertinentes para cada acceso y dirección.

```
$result = mysql_query($SQLDatos,$conexion)or die("Error:".mysql_error());
$info = mysql_fetch_array($result, MYSQL_NUM);
mysql_free_result($result);
$result_1 = mysql_query($SQLDatos_1,$conexion)or die("Error:".mysql_error());
$info_1 = mysql_fetch_array($result_1, MYSQL_NUM);
mysql_free_result($result_1);
$result1 = mysql_query($SQLDatos1,$conexion)or die("Error:".mysql_error());
$info1 = mysql_fetch_array($result1, MYSQL_NUM);
mysql_free_result($result1);
$result2 = mysql_query($SQLDatos2,$conexion)or die("Error:".mysql_error());
$info2 = mysql_fetch_array($result2, MYSQL_NUM);
mysql_free_result($result2);
$result3 = mysql_query($SQLDatos3,$conexion)or die("Error:".mysql_error());
$info3 = mysql_fetch_array($result3, MYSQL_NUM);
mysql_free_result($result3);
$result3_1 = mysql_query($SQLDatos3_1,$conexion)or die("Error:".mysql_error());
$info3_1 = mysql_fetch_array($result3_1, MYSQL_NUM);
mysql_free_result($result3_1);
$result4 = mysql_query($SQLDatos4,$conexion)or die("Error:".mysql_error());
$info4 = mysql_fetch_array($result4, MYSQL_NUM);
mysql_free_result($result4);
$result4_1 = mysql_query($SQLDatos4_1,$conexion)or die("Error:".mysql_error());
$info4_1 = mysql_fetch_array($result4_1, MYSQL_NUM);
mysql_free_result($result4_1);
```

Se calcula la media a través de las variables de cada acceso, para especificar la media por minuto y en el caso de doble carril, calcularlo correctamente.

```
$finfo[0]=$finfo[0]/10;  
$finfo_1[0]=$finfo_1[0]/10;  
$finfo1[0]=$finfo1[0]/10;  
$finfo1[0]=$finfo1[0]/2;  
$finfo2[0]=$finfo2[0]/10;  
$finfo2[0]=$finfo2[0]/2;  
$finfo3[0]=$finfo3[0]/10;  
$finfo3_1[0]=$finfo3_1[0]/10;  
$finfo4[0]=$finfo4[0]/10;  
$finfo4_1[0]=$finfo4_1[0]/10;  
  
mysql_close($conexion);
```

Para mostrar los datos, se ha optado por utilizar la API de Google y sus aplicaciones Charts y Maps, por su gran poder a la hora de modificar sus características, por su gran documentación, y por ser OpenSource y soportado por la mayoría de navegadores y Motores de BD.

A través del código base de Google Maps, se realiza un script que identifique la cantidad de tráfico en cada momento y respecto a esa cantidad de automóviles, imprima un marcador diferente, siendo el marcador rojo si pasan más de 20 coches por minuto, siendo amarillo si pasan de 11 a 20 coches por minuto y siendo el marcador color verde, si el tráfico es de 10 coches o menos por minuto pasado, realizando la media con los valores de los 10 últimos minutos.

Se ha debido comentar mediante “echo” el script de Google Maps en formato PHP, para que el paso de los “results” sea correcto.

A continuación, mostramos el código que utilizamos para los posicionar con coordenadas los diferentes marcadores y posteriormente, la función que identifica que marcador poner, que dependerá del valor de la consulta.

```

echo "function myMap(){";
echo "var myCenter = new google.maps.LatLng(38.702438,-0.481251);";
echo "var estacion1a = new google.maps.LatLng(38.684420,-0.479158);";
echo "var estacion1 = new google.maps.LatLng(38.687485,-0.476497);";
echo "var estacion2 = new google.maps.LatLng(38.713358,-0.466864);";
echo "var estacion3 = new google.maps.LatLng(38.711640,-0.461424);";
echo "var estacion4 = new google.maps.LatLng(38.695212,-0.465137);";
echo "var estacion4a = new google.maps.LatLng(38.693647,-0.469557);";
echo "var estacion5 = new google.maps.LatLng(38.68451,-0.4993187);";
echo "var estacion5a = new google.maps.LatLng(38.688549,-0.5016457);";

echo "var fine1 = new google.maps.LatLng(38.706950,-0.4721487);";
echo "var fine2 = new google.maps.LatLng(38.7063647,-0.4734216);";
echo "var fine3 = new google.maps.LatLng(38.696371,-0.485587);";
echo "var fine4 = new google.maps.LatLng(38.6943255,-0.4773754);";
echo "var fine5 = new google.maps.LatLng(38.6950656,-0.4741219);";

echo "var mapCanvas = document.getElementById(\"map\");";
echo "var mapProp ={";
echo "center: myCenter,";
echo "scrollwheel: false,";
echo "zoom: 14,";
echo "zoomControl: false,";
echo "rotateControl : false,";
echo "mapTypeControl: false,";
echo "streetViewControl: false,";
echo "scaleControl: false,";
echo "disableDefaultUI: true,";
echo "};";
echo "var map = new google.maps.Map(mapCanvas, mapProp);";
  
```

Ahora depende del valor de la variable, imprimimos un marcador u otro.

```

echo "if($finfo[0]>20){";
echo "var marker1 = new google.maps.Marker({";
echo "position:estacion1,";
echo "icon: \"semaforo-rojo-opcion-8.png\",";
echo "animation: google.maps.Animation.DROP";
echo "});";
echo "marker1.setMap(map);";
echo "}else if($finfo[0]>10){";
echo "var marker1 = new google.maps.Marker({";
echo "position:estacion1,";
echo "icon: \"semaforo-amarillo-opcion-8.png\",";
echo "animation: google.maps.Animation.DROP";
echo "});";
echo "marker1.setMap(map);";
echo "}else{";
echo "var marker1 = new google.maps.Marker({";
echo "position:estacion1,";
echo "icon: \"semaforo-verde-opcion-8.png\",";
echo "animation: google.maps.Animation.DROP";
echo "});";
echo "marker1.setMap(map);";
echo "};";
  
```

De esta manera se leen los datos de las variables \$finfoX y se comparan mediante un IF, así el script imprime el marcador que se ajuste a la densidad de tráfico actual conforme la condición del IF. Repetimos el mismo proceso con las diferentes entradas de Alcoi, siendo cada una de ellas un marcador.

Asignamos a cada marcador su evento a través de las funciones de google Maps, y también le decimos que muestra para cada marcador un infowindow.

```

echo "google.maps.event.addListener(marker1,'click',function() {";
echo "infowindow1.open(map, marker1);";
echo "});";

echo "google.maps.event.addListener(marker1a,'click',function() {";
echo "infowindow1a.open(map, marker1a);";
echo "});";

echo "google.maps.event.addListener(marker2,'click',function() {";
echo "infowindow2.open(map, marker2);";
echo "});";

echo "google.maps.event.addListener(marker3,'click',function() {";
echo "infowindow3.open(map,marker3);";
echo "});";

echo "google.maps.event.addListener(marker4,'click',function() {";
echo "infowindow4.open(map,marker4);";
echo "});";

echo "google.maps.event.addListener(marker4a,'click',function() {";
echo "infowindow4a.open(map,marker4a);";
echo "});";

echo "google.maps.event.addListener(marker5,'click',function() {";
echo "infowindow5.open(map,marker5);";
echo "});";

```

En cada infowindow de cada marcador, se muestra la media de coches durante el último minuto y un enlace a la gráfica que mostrará el tráfico trascurrido durante las horas del día actual. Existirá una gráfica por acceso.

```

echo "var infowindow1 = new google.maps.InfoWindow({";
echo "content: \"Entrada desde Alicante: <br> Media de coches por minuto: $finfo[0] <br> <a href=./trafico_alicante.php target=_blank> info";
echo "});";
echo "var infowindow1a = new google.maps.InfoWindow({";
echo "content: \"Salida hacia Alicante: <br> Media de coches por minuto: $finfo_1[0] <br> <a href=./trafico_alicante.php target=_blank>";
echo "});";
echo "var infowindow2 = new google.maps.InfoWindow({";
echo "content: \"Entrada desde Cocentaina: <br> Media de coches por minuto: $finfo1[0] <br> <a href=./trafico_EN_cocentaina.php";
echo "});";
echo "var infowindow3 = new google.maps.InfoWindow({";
echo "content: \"Salida hacia Cocentaina: <br> Media de coches por minuto: $finfo2[0] <br> <a href=./trafico_SA_cocentaina.php";
echo "});";
echo "var infowindow4 = new google.maps.InfoWindow({";
echo "content: \"Entrada desde Revolcat: <br> Media de coches por minuto: $finfo3[0] <br> <a href=./trafico_revolcat.php target=_blank>";
echo "});";
echo "var infowindow4a = new google.maps.InfoWindow({";
echo "content: \"Salida hacia Revolcat: <br> Media de coches por minuto: $finfo3_1[0] <br> <a href=./trafico_revolcat.php target=_blank>";
echo "});";
echo "var infowindow5 = new google.maps.InfoWindow({";
echo "content: \"Entrada desde Banyeres: <br> Media de coches por minuto: $finfo4[0] <br> <a href=./trafico_banyeres.php target=_blank>";
echo "});";
echo "var infowindow5a = new google.maps.InfoWindow({";
echo "content: \"Salida hacia Banyeres: <br> Media de coches por minuto: $finfo4_1[0] <br> <a href=./trafico_banyeres.php target=_blank>";
echo "});";

```


Esto dejará el siguiente resultado por marcador, posicionado en su coordenada del mapa de Alcoi y con su respectivo color. Al pulsar sobre el marcador podemos acceder a la gráfica del tráfico transcurrido durante el día de hoy por cada salida pulsando el link "+ info", para ello se realiza diferentes ficheros PHP uno por salida, ya que necesitamos tratar los datos antes de imprimir con Google Charts la respectiva gráfica por acceso.



Se modifica la consulta MYSQL, ya que ahora interesan los coches que entran y salen cada hora y que el resultado lo cuadre agrupándolo por horas, que será la gráfica que se imprima al pulsar en "+ info" en el infowindow de cada marcador.

Para ello se realiza una consulta que cree una tabla con COALESCE con el resultado de dos consultas independientes acercamiento/alejamiento, se adquieren las variables \$ano, \$mes y \$dia a través del sistema.

```

$SQLDatos = "SELECT t1.hora, t1.Entran, COALESCE(t2.Salida,0) AS Salen FROM (SELECT HOUR(fecha) as hora, COUNT(matricula) as Entran FROM
matriculas WHERE (id_sistema = 'AF_Entrada_Alicante') AND (direccion = 'Acercamiento') AND fecha LIKE ('%".$ano."-%".$mes."-%".$dia."%') group by
HOUR(fecha)) t1 LEFT JOIN (SELECT HOUR(fecha) as hora, COUNT(matricula) as Salida FROM matriculas WHERE (id_sistema = 'AF_Entrada_Alicante')
AND (direccion = 'Alejamiento') AND fecha LIKE ('%".$ano."-%".$mes."-%".$dia."%') group by HOUR(fecha)) t2 ON t1.hora = t2.hora";
  
```

Consulta que se ejecuta al pulsar "+info"

Se deben tratar los datos con la función "obtenerDatos" para que el API de Google interprete correctamente la gráfica, si no es incapaz de pintarla correctamente.

```
$result = mysqli_query($conexion,$SQLDatos)or die("Error:".mysqli_error());
$datosObtenidos=obtenerDatos($result);
$datosJSON= json_encode($datosObtenidos,JSON_NUMERIC_CHECK);
mysqli_close($conexion);

function obtenerDatos($result){
    $numFilas = mysqli_num_rows($result); //obtenemos número de filas
    $numCols = mysqli_num_fields($result); //obtenemos número de columnas
    $columnas = array();
    //preparamos las cabeceras de tabla de datos
    while ($finfo = mysqli_fetch_field($result)) { //obtenemos diferente informacion del resultado de la consulta
        $columnas = array_merge($columnas, array($finfo->name)); //obtenemos el nombre del campo de cada columna
    }
    $datos[0] =$columnas; //pasa el nombre de todas la columnas a la primera fila de los datos
    $i=1;
    //vamos leyendo fila a fila los resultados y se preparan en un array
    while ($filas = $result->fetch_array(MYSQLI_NUM)) {
        $columnas = array();
        $j=0;
        while ($j<$numCols)
        {
            $columnas = array_merge($columnas, array($filas[$j]));
            $j++;
        }
        $datos[$i] = $columnas; //pasamos todas la columnas a una fila de datos
        $i++;
    }
    mysqli_free_result($result); //liberamos el resultset
    //print_r ($datos);
    return $datos; //devolvemos el array completo con cabecera y datos
}
```

Código obtenerDatos, necesario para tratar los datos de cualquier gráfica.

Al tratar los datos para que el API de Google sea capaz de imprimir correctamente la gráfica, con la función **obtenerDatos**, se recogen los datos de la consulta, identifica los diferentes campos y pone como posición 0 del ARRAY, las cabeceras de la consulta, pintándola google en la posición X del eje de cada gráfica.



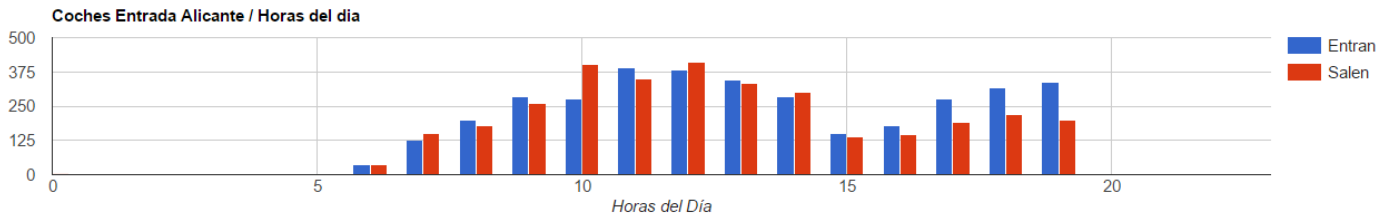
Una vez estudiada la consulta, y tratado el "result" correspondiente con la función "obtenerDatos" para que se devuelva la estructura de tabla que se necesita para que se imprima correctamente la gráfica, se muestran los datos con Google Charts.

```
echo "<script type=\"text/javascript\" src=\"https://www.gstatic.com/charts/loader.js\"></script>";
echo "<script src=\"https://ajax.googleapis.com/ajax/libs/jquery/2.2.0/jquery.min.js\"></script>";
echo "<script type=\"text/javascript\">";
echo "google.charts.load('current', {packages: ['corechart']});";
echo "google.charts.setOnLoadCallback(drawBasic);";
echo "";
echo "function drawBasic() {";
echo "";
echo "    var respuesta=$datosJSON;";
echo "    var data = new google.visualization.arrayToDataTable(respuesta);";
echo "";
echo "var options = {";
echo "title: 'Coches Entrada Alicante / Horas del dia',";
echo "hAxis: {";
echo "title: 'Horas del Día',";
echo "format: '',";
echo "viewWindow: {";
echo "min: [00,, ],";
echo "max: [23,, ]";
echo "};";
echo "},";
echo "vAxis: {";
echo "title: 'Numero de coches'";
echo "};";
echo "};";
echo "var chart = new google.visualization.ColumnChart(document.getElementById('chart_div'));";
echo "";
echo "chart.draw(data, options);";
echo "};";
echo "</script>";
echo "</head>";
echo "<body>";
```

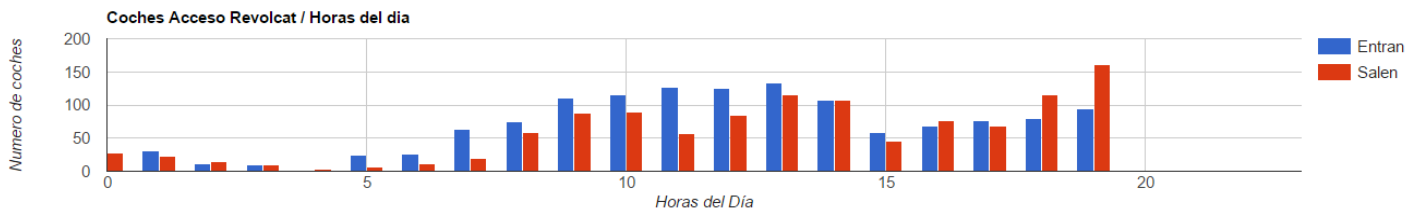
Función para imprimir la gráfica de cada acceso correctamente después de tratar los datos con "obtenerDatos".

Este fichero PHP el cual recoge los datos de conexión, la consulta, la función "obtenerDatos" y el script del mapa de Google, se repetirá por cada acceso a la ciudad, imprimiendo cada uno de ellos su respectiva gráfica.

Gráfica +info Entrada Alicante (trafico_alicante.php):



Gráfica +info Acceso Revolcat (trafico_revolcat.php):



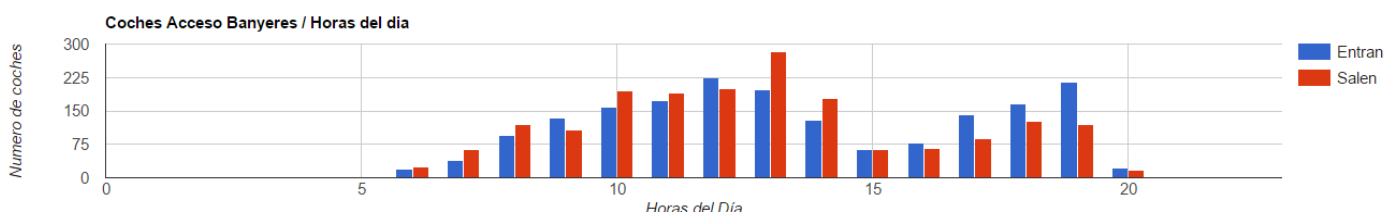
Gráfica +info Entrada Cocentaina (trafico_EN_cocentaina.php):



Gráfica +info Salida Cocentaina (trafico_SA_cocentaina.php):



Gráfica +info Acceso a Banyeres (trafico_bayeres.php):



Una vez se posicionan correctamente los marcadores y con su respectiva media correctamente calculada, se bloquean los controles de Zoom y la UI ESTÁNDAR de Google Maps, para limitar el movimiento del usuario.

```
echo "var mapCanvas = document.getElementById(\"map\");";  
echo "var mapProp ={";  
echo "center: myCenter,";  
echo "scrollwheel: false,";  
echo "zoom: 14,";  
echo "zoomControl: false,";  
echo "rotateControl : false,";  
echo "mapTypeControl: false,";  
echo "streetViewControl: false,";  
echo "scaleControl: false,";  
echo "disableDefaultUI: true,";  
echo "}";
```

Se ha implementado también un GIF de Carga, para hacer más amena la espera de la carga de trafico.php, ya que antes de mostrar el mapa, debe realizar las diferentes consultas de cada acceso, realizar los cálculos de media, e imprimir correctamente el marcador asociado, si pasan más de 20 coches por minuto imprimir rojo marcador, si pasan más de 10 coches marcador amarillo y si 10 coches o menos, marcador en verde.

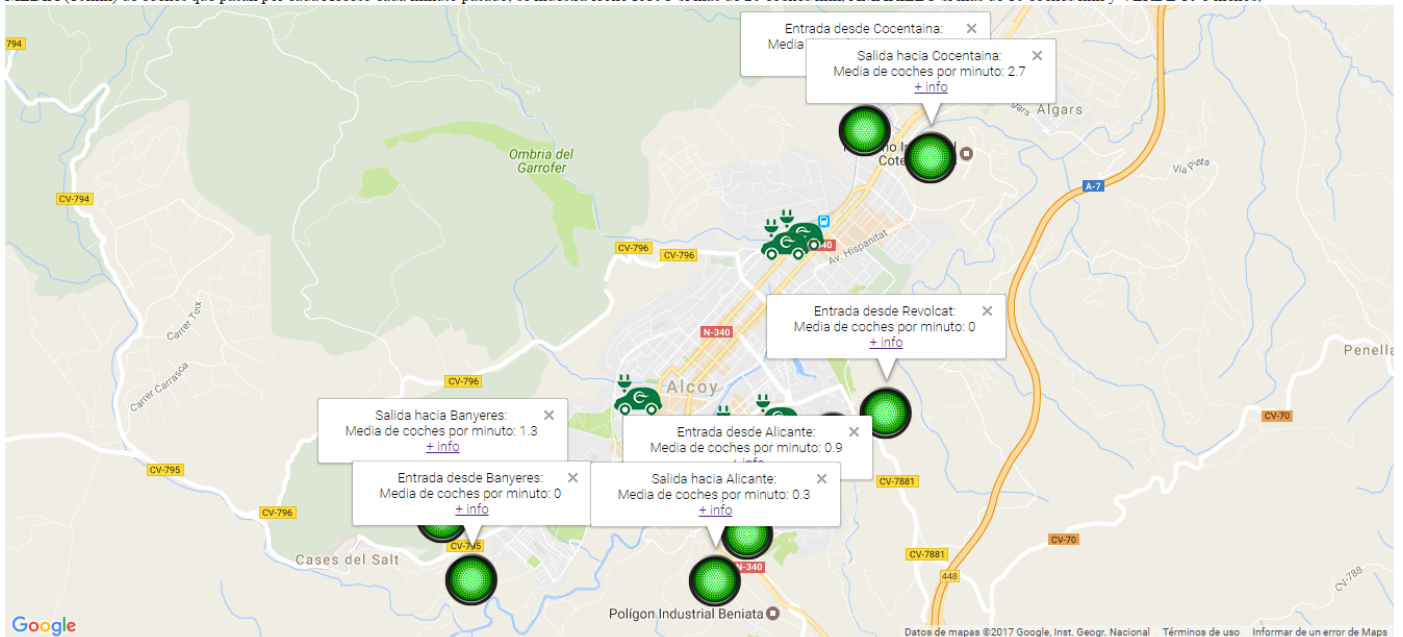
```
echo "<script type='text/javascript' src='http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js'></script>";  
echo "<script type='text/javascript'>";  
echo "function efecto(){";  
echo "$('#fotocargando').hide();";  
echo "$('#map').fadeIn(1500);";  
echo "}";  
echo "</script>";  
  
echo "</head>";  
  
echo "<body onload='efecto();'>";  
echo "<div id='fotocargando' style='width:100%;height:100%;text-align: center;'>";  
echo "<img src='./loading.gif'>";  
echo "</div>";  
echo "<div id='map' style='width:100%;height:95%;text-align: center;'></div>";  
  
echo "</body>";
```

Para poder mostrar correctamente el mapa de Google, nos hace falta generar una KEY de la API de Google Maps.

```
echo "}]";  
echo "</script>";  
echo "<script src='\"https://maps.googleapis.com/maps/api/js?key=AIzaSyAwItg2TfpikBdHrBKUKTRiLYjgUy3sGs&callback=myMap'></script>";  
echo "</html>";  
?>
```

Ya disponemos de trafico.php con sus respectivas consultas y sus respectivas gráficas por acceso.

MEDIA (10min) de coches que pasan por cada Acceso cada minuto pasado, se muestra icono ROJO si mas de 20 coches/min, AMARILLO si mas de 10 coches/min y VERDE 10 o menos:



Esta es la vista final de trafico.php



Contenido 2: Históricos de tráfico

En este apartado, se ofrece al usuario otro tipo de estadísticas de los diferentes accesos de Alcoi, no solo el agrupado por horas del día actual, sino también de días anteriores, meses anteriores o de una fecha concreta.

Para ello se ha generado un fichero PHP llamado **historial.php**, con un menú compuesto de Radio Buttons, capaz de recoger las diferentes variables que el usuario marque, y generar la consulta conforme a sus necesidades, dando una flexibilidad y una capacidad para extraer la información, muy completa.

Para realizar este contenido se han utilizado lenguajes como como PHP, MYSQL, HTML y JavaScript y paso de variables entre ficheros PHP de **historial.php** a los ficheros **his_ayer.php**, **his_mes.php** e **his_fecha.php**.

Como a través del contenido anterior de tráfico, ya se dispone de los datos en la BD externa en MYSQL, alojada en el dominio proporcionado por el Ajuntament D'Alcoi, ya es seleccionar y estudiar las diferentes consultas, para que ofrezcan y muestren los datos, a través de los gráficos de Google Chart y su API.

Se parte de ofrecer un menú para que el usuario pueda especificar el tipo de consulta que quiere realizar. Para ello, se ha hecho uso de métodos GET y POST a través de un FORM compuesto de radios, que serán capaces de seleccionar el acceso del cual queremos realizar la gráfica, y por otro lado la fecha en la que se comprende la gráfica.



A continuación, se muestra el código que efectúa el menú FORM con sus respectivos Radio Buttons, y la respectiva recogida de variables para la posterior ejecución de la consulta MySQL. Se ha añadido una tabla que agrupe mejor las secciones y que permita una mejor visualización para el usuario.

```
<html>
<head>
<meta charset="utf-8">
<title>Historial de Trafico</title>
</head>

<body>

<table width="280" cellspacing="1" cellpadding="3" border="0" bgcolor="#1E679A" align="center">
<tr>
<td><font color="#FFFFFF" face="arial, verdana, helvetica">
<div style="text-align:center;">
<b>Selecciona un Acceso</b>
</div>
</font></td>
</tr>
<tr>
<td bgcolor="#ffffcc">
<font face="arial, verdana, helvetica">
<form action="historial.php" method="post">
<input type="radio" name="v3" value="AF_Entrada_Alicante"> Alicante<br><br>
<input type="radio" name="v3" value="AF_Acceso_Revolcat"> Acceso Revolcat<br><br>
<input type="radio" name="v3" value="AF_Entrada_Cocentaina"> Entrada Cocentaina<br><br>
<input type="radio" name="v3" value="AF_Salida_Cocentaina"> Salida Cocentaina<br><br>
<input type="radio" name="v3" value="AF_Acceso_Banyeres"> Acceso Banyeres<br><br>
</font>
</td>
</tr>
</table>
<table width="280" cellspacing="1" cellpadding="3" border="0" bgcolor="#1E679A" align="center">
<tr>
<td><font color="#FFFFFF" face="arial, verdana, helvetica">
<div style="text-align:center;">
<b>Selecciona un tiempo</b>
</div>
</font></td>
</tr>
<tr>
<td bgcolor="#ffffcc">
<font face="arial, verdana, helvetica">
<input type="radio" name="fecha" value="dia"> Dia anterior<br><br>
<input type="radio" name="fecha" value="mes"> Mes anterior<br><br>
<input type="radio" name="fecha" value="fechax"> Fecha concreta<br><br>
Desde <input type="date" name="v1" size="20"><br><br>
Hasta <input type="date" name="v2" size="20"><br><br>
<div style="text-align:center;">
<input type="submit" name="enviar" value="Enviar"/>
</div>
</form>
</font>
</td>
</tr>
</table>
<?php
```

Selecciona un Acceso

- Alicante
- Acceso Revolcat
- Entrada Cocentaina
- Salida Cocentaina
- Acceso Banyeres

Selecciona un tiempo

- Dia anterior
- Mes anterior
- Fecha concreta

Desde

Hasta

Imagen de resultado final de historial.php utilizando FORM y Radio Buttons.

Cuando el usuario selecciona la entrada y el tiempo para la consulta, se recogen a través del botón "Enviar" los diferentes parámetros y se envía a los diferentes ficheros PHP. Para evitar realizar ficheros PHP innecesarios, se pasa como variable el acceso, y el tiempo, reutilizando así los diferentes ficheros existentes mediante el paso de variables.

```
if (isset($_POST['enviar'])) {
    //Si el checkbox condiciones tiene valor
    if (isset($_POST['v3']) && $_POST['v3'] == 'AF_Entrada_Alicante'){
        if (isset($_POST['fecha']) && $_POST['fecha'] == 'dia'){
            header("Location: his_ayer.php?v3=$_v3");
        }
        <script language="JavaScript">
            top.location.href = "his_ayer.php?v3=$_v3";
        </script>
    }
    if (isset($_POST['fecha']) && $_POST['fecha'] == 'mes'){
        header("Location: his_mes.php?v3=$_v3");
    }
    <script language="JavaScript">
        top.location.href = "his_mes.php?v3=$_v3";
    </script>
    if (isset($_POST['fecha']) && $_POST['fecha'] == 'fechax'){
        header("Location: his_fecha.php?v1=$_v1&v2=$_v2&v3=$_v3");
    }
    <script language="JavaScript">
        top.location.href = "his_fecha.php?v1=$_v1&v2=$_v2&v3=$_v3";
    </script>
}
```

A través del código anterior, y según las variables marcadas, se abre su PHP correspondiente pasándole las variables que el usuario a marcado. Este mismo código será exactamente igual para el resto de accesos, cambiando el nombre de la entrada. Una vez concluido el paso de variables mediante JavaScript, se han realizado 3 ficheros PHP, uno para cada tiempo de consulta, ya que hay que modificar la consulta MySQL en cada una de ellas. Se han reutilizado los ficheros evitando crear 3 ficheros PHP por entrada.

El fichero "his_ayer.php" mostrará una gráfica del tráfico del día anterior en la entrada seleccionada, se realiza la conexión, se recoge la fecha del sistema actual, y se resta un día, para imprima el día pasado.

```

//conexion
$conexion = mysqli_connect($servidor,$suserbbdd,$spassbbdd,$nombrebdd)or die("Error:".mysqli_error());

//variables
$aux_contador_horas=0;
$ano = date("Y");
$mes = date("m");
$dia = date("d");

$sv3= $_GET['v3'];
$dia--;

//preparamos la consulta
$SQLDatos = "SELECT t1.hora, t1.Entran, COALESCE(t2.Salida,0) AS Salen FROM (SELECT HOUR(fecha) as hora, COUNT(matricula) as Entran FROM matriculas WHERE (id_sistema = '". $sv3. "') AND (direccion ='Acercamiento') AND fecha LIKE ('%". $ano. "-" . $mes. "-" . $dia. "%') group by HOUR(fecha)) t1 LEFT JOIN (SELECT HOUR(fecha) as hora, COUNT(matricula) as Salida FROM matriculas WHERE (id_sistema = '". $sv3. "') AND (direccion ='Alejamiento') AND fecha LIKE ('%". $ano. "-" . $mes. "-" . $dia. "%') group by HOUR(fecha)) t2 ON t1.hora = t2.hora";

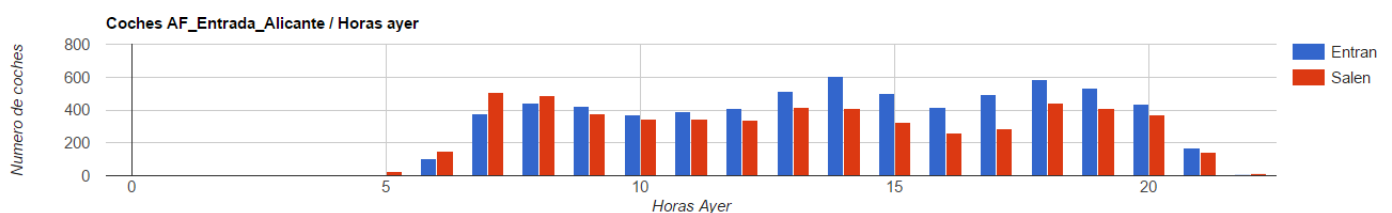
$result = mysqli_query($conexion,$SQLDatos)or die("Error:".mysqli_error());
$datosObtenidos=obtenerDatos($result);
$datosJSON= json_encode($datosObtenidos,JSON_NUMERIC_CHECK);

mysqli_close($conexion);
  
```

Se recoge la variable V3 que especifica el acceso marcado en historial.php, para completar la consulta y que muestre el tráfico agrupado en horas de la salida seleccionada el día anterior, este mismo fichero se utilizará con el resto de salidas, cuando se seleccione la gráfica del día anterior.

Una vez realizada la consulta, se deben tratar los datos con la función "obtenerDatos" (mostrada en el apartado "trafico.php") para que la API de Google Maps imprima correctamente la gráfica.

Gráfica seleccionando "Entrada Alicante y Día anterior".



El fichero "his_mes.php" mostrará una gráfica del tráfico del mes anterior en la entrada seleccionada, realizamos la conexión, recogemos la fecha del sistema actual, y se resta un mes, para obtener correctamente los datos del mes pasado.

```
//variables
$aux_contador_horas=0;
$ano = date("Y");
$mesi = date("m");
$mes = date("m");
$día = date("d");

$V3= $_GET['v3'];
$mesi--;

//preparamos la consulta
$SQLDatos = "SELECT t1.día, t1.Entran, COALESCE(t2.Salida,0) AS Salen FROM (SELECT DAY(fecha) as día, COUNT(matricula) as Entran FROM matriculas WHERE (id_sistema = '". $V3. "') AND (direccion ='Acercamiento') AND fecha BETWEEN '2017-".$mesi."-1' AND '2017-".$mes."-1' group by DAY(fecha)) t1 LEFT JOIN (SELECT DAY(fecha) as día, COUNT(matricula) as Salida FROM matriculas WHERE (id_sistema = '". $V3. "') AND (direccion ='Alejamiento') AND fecha BETWEEN '2017-".$mesi."-1' AND '2017-".$mes."-1' group by DAY(fecha)) t2 ON t1.día = t2.día";

$result = mysqli_query($conexion,$SQLDatos)or die("Error:".mysqli_error());
$datosObtenidos=obtenerDatos($result);
$datosJSON= json_encode($datosObtenidos,JSON_NUMERIC_CHECK);

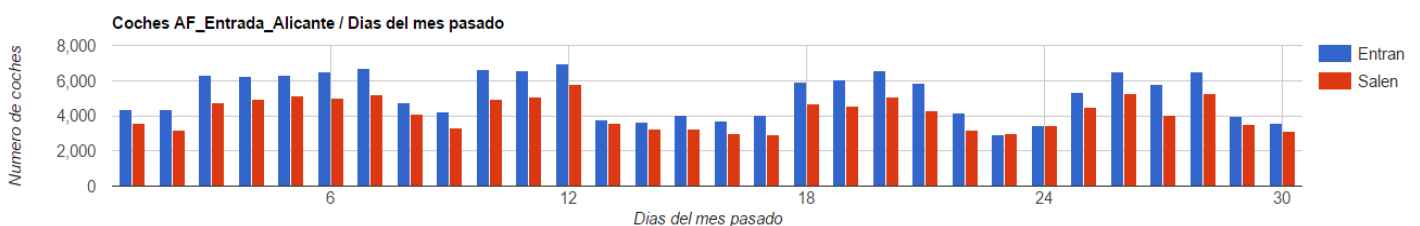
mysqli_close($conexion);

function obtenerDatos($result){
    $numFilas = mysqli_num_rows($result); //obtenemos número de filas
    $numCols = mysqli_num_fields($result); //obtenemos número de columnas
    $columnas = array();
}
```

Se recoge la variable V3 que especifica el acceso marcado en historial.php, para completar la consulta y que muestre el tráfico agrupado en días de la salida seleccionada del mes anterior, este mismo fichero se utilizará con el resto de salidas, cuando se seleccione la gráfica del mes anterior.

Una vez realizada la consulta, se deben tratar los datos con la función "obtenerDatos" (mostrada en el apartado "trafico.php") para que la API de Google Maps imprima correctamente la gráfica.

Gráfica seleccionando "Entrada Alicante y Mes anterior".



El fichero "his_fecha.php" mostrará una gráfica del tráfico comprendido entre las fechas seleccionadas y de su respectivo acceso, realizamos la conexión, y modificando la consulta, en este caso no necesitamos ninguna fecha del sistema.

```

$v1 = $_GET['v1'];
$v2 = $_GET['v2'];
$v3 = $_GET['v3'];

//preparamos la consulta
$SQLDatos = "SELECT t1.dia, t1.Entran, COALESCE(t2.Salida,0) AS Salen FROM (SELECT DAY(fecha) as dia, COUNT(matricula) as Entran FROM
matriculas WHERE (id_sistema = '". $v3. "') AND (direccion ='Acercamiento') AND fecha BETWEEN '". $v1. "' AND '". $v2. "' group by DAY(fecha)) t1
LEFT JOIN (SELECT DAY(fecha) as dia, COUNT(matricula) as Salida FROM matriculas WHERE (id_sistema = '". $v3. "') AND (direccion ='Alejamiento')
AND fecha BETWEEN '". $v1. "' AND '". $v2. "' group by DAY(fecha)) t2 ON t1.dia = t2.dia";

$result = mysqli_query($conexion,$SQLDatos)or die("Error:".mysqli_error());
$datosObtenidos=obtenerDatos($result);
$datosJSON= json_encode($datosObtenidos,JSON_NUMERIC_CHECK);

mysqli_close($conexion);

function obtenerDatos($result){
    $numFilas = mysqli_num_rows($result); //obtenemos número de filas
    $numCols = mysqli_num_fields($result); //obtenemos número de columnas
  
```

Se recoge la variable V3 que especifica el acceso marcado en historial.php, la variable V1 que especifica la fecha DESDE y la variable V2 que especifica la fecha HASTA, para completar la consulta y que muestre el tráfico agrupado en días de la salida seleccionada, del intervalo de fecha seleccionado. Este mismo fichero se utilizará con el resto de salidas, cuando se seleccione el Radio Button "Fecha concreta" de historial.php.

Una vez realizada la consulta, se deben tratar los datos con la función "obtenerDatos" (mostrada en el apartado "trafico.php") para que la API de Google Maps imprima correctamente la gráfica.

Gráfica seleccionando "Entrada Alicante y Fecha concreta".

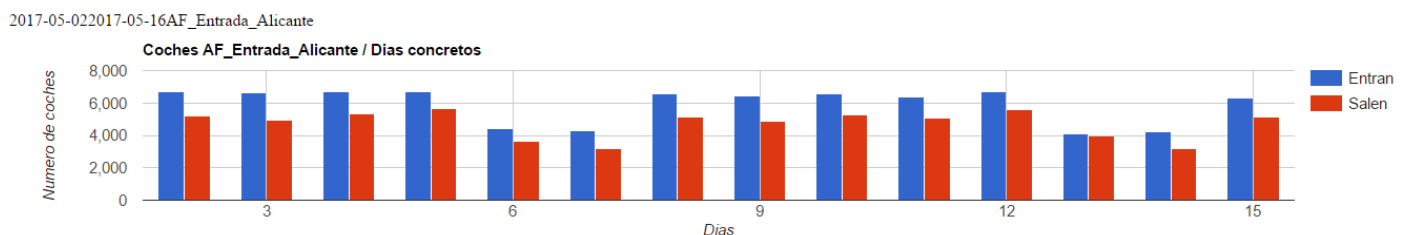


Imagen final historial.php

Selecciona un Acceso

- Alicante
- Acceso Revolcat
- Entrada Cocentaina
- Salida Cocentaina
- Acceso Banyeres

Selecciona un tiempo

- Dia anterior
- Mes anterior
- Fecha concreta

Desde

Hasta

mayo de 2017

lu.	ma.	mi.	ju.	vi.	sá.	do.
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Al pulsar en el botón "Enviar" se abrirá una nueva ventana con la gráfica y valores seleccionados, pasados por JS, V3, V2 y V1.

Contenido 3: Tiempo autobuses urbanos Alcoi.

En este apartado se realiza lo mismo que en el apartado de tráfico, pero con las diferentes paradas de autobuses urbanos de Alcoi, en la que se recoja también el tiempo que falta para que llegue cada autobús. En este tipo de contenido se adquieren los datos a través de serie de SCRIPTS conectados a VECTALIA que, a través del "parseo" de datos, se recogen los diferentes datos de cada parada.

Se hace uso de los datos recogidos por el SCRITP, llamado "autob.php" que se encarga de recoger el código de la parada, el nombre, y la coordenada en la que se encuentra cada parada, y otro SCRIPT llamado "autob2.php" que recoge el tiempo de le queda a cada línea para llegar a esa parada, y procedemos a imprimir la información en el mapa de google.

Se define el mapa de Google y se le marca su centro.

```
$contador_mapa_1=1;
echo "<html>";
echo "<style>";
echo "*{ margin: 0px; padding: 0px; }";
echo "</style>";
echo "<body>";
echo "<div id=\"map\" style=\"width:100%;height:100%\"></div>";
echo "<script>";
  echo "function myMap(){";
  echo "var myCenter = new google.maps.LatLng(38.701552, -0.478876);";
```

Se procede a posicionar los marcadores de cada parada, recorriendo el número total de paradas.

```
while($contador_mapa_1<=81)
{
  echo "var estacion$contador_mapa_1 = new google.maps.LatLng($cord[$contador_mapa_1]);";
  $contador_mapa_1++;
}
echo "var mapCanvas = document.getElementById(\"map\");";
echo "var mapProp ={";
echo "center: myCenter,";
echo "zoom: 14,";
echo "};";
echo "var map = new google.maps.Map(mapCanvas, mapProp);";
```

Se procede a poner la imagen a los marcadores de cada parada, recorriendo el número total de paradas.

```
$contador_mapa_1=1;
while($contador_mapa_1<=81)
{
echo "var marker$contador_mapa_1 = new google.maps.Marker({" ;
echo "position:estacion$contador_mapa_1," ;
echo "icon: \"imagenes/autobuss.png\"," ;
echo "animation: google.maps.Animation.DROP";
echo "});";
echo "marker$contador_mapa_1.setMap(map);";

$contador_mapa_1++;
}
```

Se procede a crear el infowindow de cada marcador, de cada parada, recorriendo el número total de paradas.

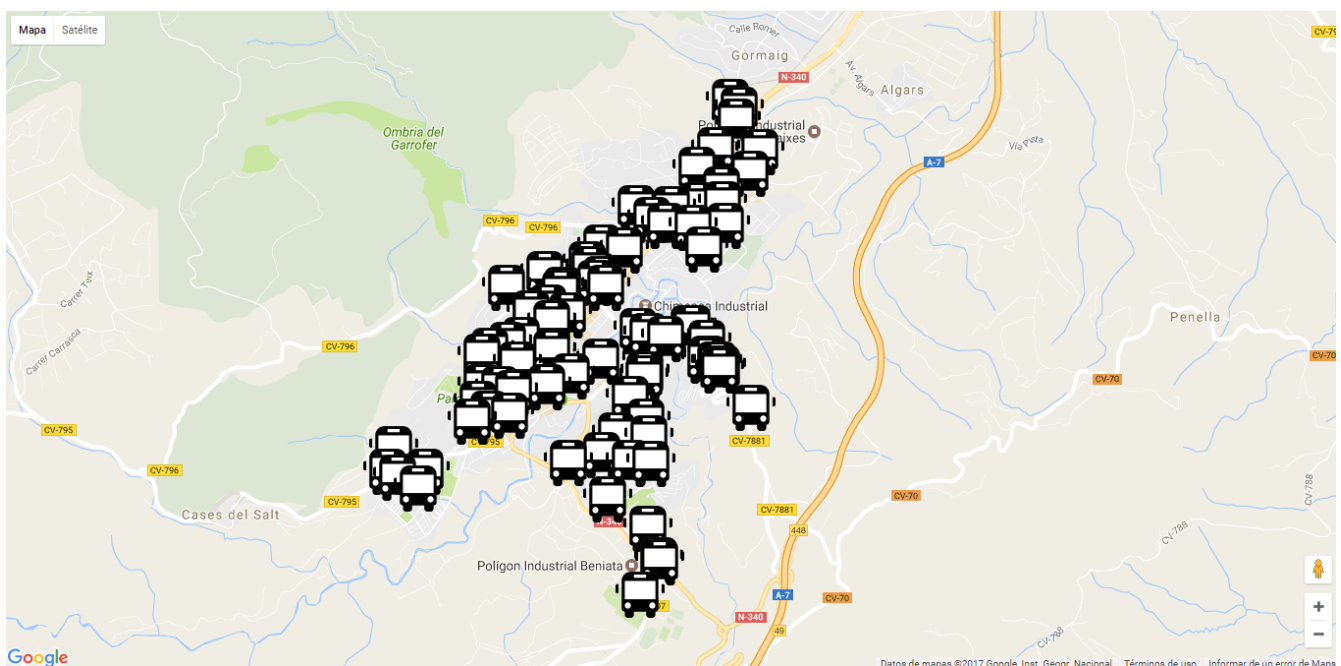
```
$contador_mapa_1=1;
while($contador_mapa_1<=81)
{
echo "var infowindow$contador_mapa_1 = new google.maps.InfoWindow({" ;
echo "content: \"<center>$nombre[$contador_mapa_1] <br> Cuando llega el siguiente autobus <br> Pulsa <a href='./autob2.php?codigo=$codigo[$contador_mapa_1]' target='popup' onClick='window.open(this.href, this.target,width=380,height=480); return false;'>Aquí</a></center>\"";
echo "});";
$contador_mapa_1++;
}
```

Por último, se genera el evento "Click" para cada marcador de cada parada, recorriendo el número total de paradas.

```
$contador_mapa_1=1;
while($contador_mapa_1<=81)
{
echo "google.maps.event.addListener(marker$contador_mapa_1,'click',function() {" ;
echo "infowindow$contador_mapa_1.open(map, marker$contador_mapa_1);";
echo "});";
$contador_mapa_1++;
}
echo "};";
echo "</script>";
echo "<script src='https://maps.googleapis.com/maps/api/js?key=AIzaSyAwItg2Tfp1KBdHdBKUKTRfLYjgUy3sGs&callback=myMap'></script>";
echo "</body>";
echo "</html>";
?>
```


En este caso, A través de WHILEs que recorren hasta el número total de paradas, se imprimen los marcadores, coordenadas, infowindows, y evento del marcador, de todas las paradas, de forma recurrente y ahorrando código, al ser el mismo icono y lectura de datos de fichero plano, podemos usar este método de impresión de marcadores a diferencia del contenido "trafico.php".

Imagen **autob.php** con marcadores de cada parada.



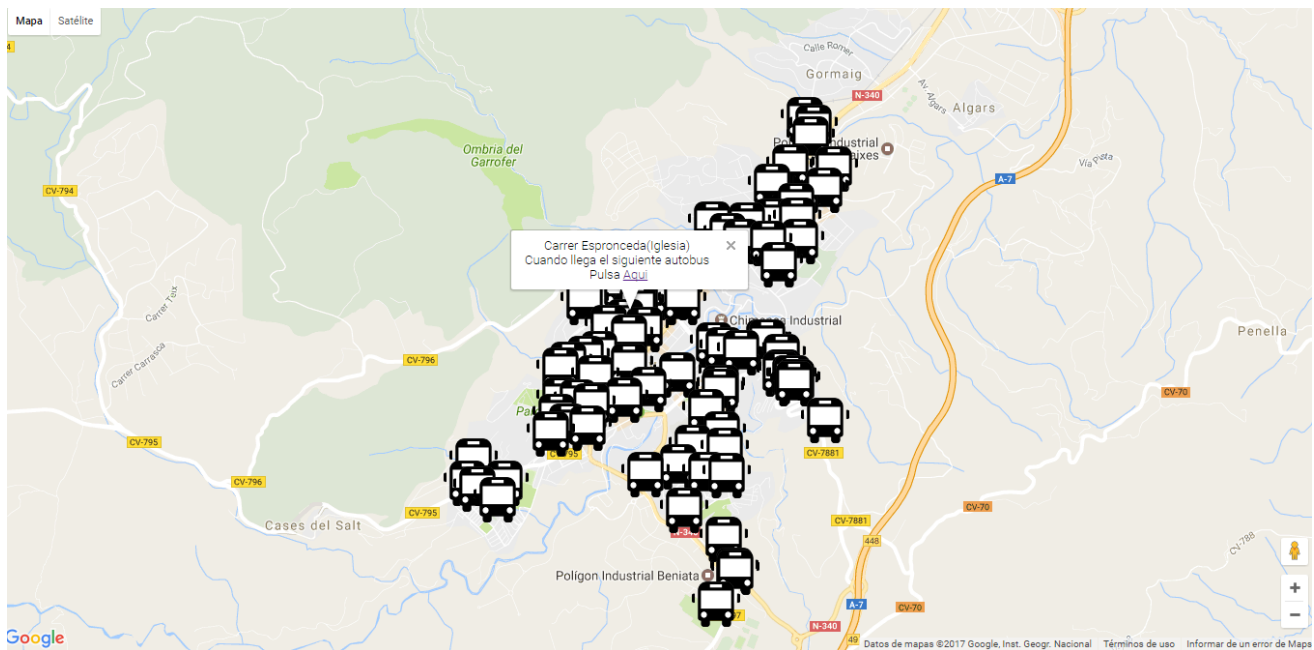
En este caso no limitamos el zoom ni la UI Estándar de Google, para que el usuario pueda ampliar y seleccionar correctamente la parada que elija.

Cuando en el infowindow de cada parada de Autobús se pulsa en "Aquí", que se encarga de adquirir los tiempos por línea con **autob2.php**, y los mostrará en una nueva ventana según la parada.

Imagen al pulsar "Pulsa Aquí".

Línea: 2
Tiempo de espera: 11 minutos

Imagen final **autob.php**



Ya se dispone de la vista del mapa de Alcoi, con la posición de las diferentes paradas de Autobús, y con su respectiva infowindow que al accionar en "Pulsa Aquí", abre una nueva ventana con los tiempos restantes de las líneas correspondientes por parada.

Contenido 4: Información meteorológica

En este apartado la intención es mostrar un cuadro meteorológico de la localidad, para ello se recogen los datos directamente de AEMET a través de su fichero XML de la localidad de Alcoi "localidad_03009.xml". Acto seguido se carga el fichero XML, se seleccionan los datos y se muestran en una tabla.

Imagen de carga de datos en tabla:

```
<?php
$xml = simplexml_load_file('http://www.aemet.es/xml/municipios/localidad_03009.xml');

?>
</br></br></br>
<center>
<table border="1">
  <tr>
    <th colspan="5">Tabla de prediccion</th>
  </tr>
  <tr>
    <td>&nbsp;</td>
    <td>
      <?php
      foreach($xml->prediccion->dia[0]->attributes() as $a => $b)
      {
        echo $b;
      }
      ?>
    </td>
    <td>
      <?php
      foreach($xml->prediccion->dia[1]->attributes() as $a => $b)
      {
        echo $b;
      }
      ?>
    </td>
    <td><?php
      foreach($xml->prediccion->dia[2]->attributes() as $a => $b)
      {
        echo $b;
      }
    </td>
  </tr>
</table>
?>
```

Creación de tabla con los datos seleccionados del XML AEMET

```

        echo $b;
    }
    ?></td>
<td>
    <?php
    foreach($xml->prediccion->dia[3]->attributes() as $a => $b)
    {
        echo $b;
    }
    ?>
</td>
</tr>
<tr>
<td>Temperatura m&iacute;nima y m&aacute;xima</td>
<td>
    <?php echo "'".$xml->prediccion->dia[0]->temperatura->minima." / ".$xml->prediccion->dia[0]->temperatura->maxima."'"; ?>
</td>
<td>
    <?php echo "'".$xml->prediccion->dia[1]->temperatura->minima." / ".$xml->prediccion->dia[1]->temperatura->maxima."'"; ?>
</td>
<td>
    <?php echo "'".$xml->prediccion->dia[2]->temperatura->minima." / ".$xml->prediccion->dia[2]->temperatura->maxima."'"; ?>
</td>
<td>
    <?php echo "'".$xml->prediccion->dia[3]->temperatura->minima." / ".$xml->prediccion->dia[3]->temperatura->maxima."'"; ?>
</td>
</tr>
<tr>
<td>Sensaci&oacute;n m&iacute;nima y m&aacute;xima</td>
<td>
    <?php echo "'".$xml->prediccion->dia[0]->sens_termica->minima." / ".$xml->prediccion->dia[0]->sens_termica->maxima."'"; ?>
</td>
<td>
    <?php echo "'".$xml->prediccion->dia[1]->sens_termica->minima." / ".$xml->prediccion->dia[1]->sens_termica->maxima."'"; ?>
</td>
<td>
    <?php echo "'".$xml->prediccion->dia[2]->sens_termica->minima." / ".$xml->prediccion->dia[2]->sens_termica->maxima."'"; ?>
</td>
<td>
    <?php echo "'".$xml->prediccion->dia[3]->sens_termica->minima." / ".$xml->prediccion->dia[3]->sens_termica->maxima."'"; ?>
</td>
</tr>
<tr>
<td>Humedad relativa m&iacute;nima y m&aacute;xima</td>
<td>
    <?php echo "'".$xml->prediccion->dia[0]->humedad_relativa->minima." / ".$xml->prediccion->dia[0]->humedad_relativa->maxima."'";
?>
</td>
<td>
    <?php echo "'".$xml->prediccion->dia[1]->humedad_relativa->minima." / ".$xml->prediccion->dia[1]->humedad_relativa->maxima."'";
?>
</td>
<td>
    <?php echo "'".$xml->prediccion->dia[2]->humedad_relativa->minima." / ".$xml->prediccion->dia[2]->humedad_relativa->maxima."'";
?>
</td>
<td>
    <?php echo "'".$xml->prediccion->dia[3]->humedad_relativa->minima." / ".$xml->prediccion->dia[3]->humedad_relativa->maxima."'";
?>
</td>
</tr>
<tr>
<td>Indice ultravioleta m&aacute;xima</td>
<td>
    <?php echo "'".$xml->prediccion->dia[0]->uv_max."'"; ?>

```

Creación de tabla con los datos seleccionados del XML AEMET



```
</td>
<td>
  <?php echo "". $xml->prediccion->dia[1]->uv_max.""; ?>
</td>
<td>
  <?php echo "". $xml->prediccion->dia[2]->uv_max.""; ?>
</td>
<td>
  <?php echo "". $xml->prediccion->dia[3]->uv_max.""; ?>
</td>
</tr>
</table>

<h1>Fecha y Hora actual</h1>

<?php
echo date('m/d/Y g:ia');
$fecha = date('m/d/Y');
echo "<br>Día actual del a&ntilde;o:";
echo date('z',strtotime($fecha));
$numero = date('z',strtotime($fecha));
$resta = 365 - $numero;
echo "<br>Numero de días que quedan para finalizar el a&ntilde;o: ".$resta;

$ano=date('Y');

function NumeroSemanasTieneUnAño($ano){
    $date = new DateTime;

    # Establecemos la fecha segun el estandar ISO 8601 (numero de semana)
    $date->setISODate("$ano", 53);
$ano=date('Y');

function NumeroSemanasTieneUnAño($ano){
    $date = new DateTime;

    # Establecemos la fecha segun el estandar ISO 8601 (numero de semana)
    $date->setISODate("$ano", 53);

    # Si estamos en la semana 53 devolvemos 53, sino, es que estamos en la 52
    if($date->format("W")==="53")
        return 53;
    else
        return 52;
}

echo "<br>Cuantas semanas tiene el a&ntilde;o:";
for($i=0;$i<1;$i++){
    echo $ano." - ".NumeroSemanasTieneUnAño($ano);
}

?>
```

Creación de tabla con los datos seleccionados del XML AEMET

Una vez se obtienen, se imprime en una tabla los diferentes datos, para que muestre la información de los diferentes días de la situación meteorológica.

Tabla de prediccion				
	2017-05-27	2017-05-28	2017-05-29	2017-05-30
Temperatura mínima y máxima	13 / 28	13 / 28	13 / 24	13 / 25
Sensación mínima y máxima	13 / 28	13 / 28	13 / 24	13 / 25
Humedad relativa mínima y máxima	35 / 80	30 / 85	40 / 95	35 / 85
Indice ultravioleta máxima	8	9	9	8

Fecha y Hora actual
05/27/2017 10:47pm
Dia actual del año:146
Numero de dias que quedan para finalizar el año: 219
Cuantas semanas tiene el año:2017 - 52

Tabla final con los datos seleccionados del XML de AEMET

Contenido 5: Dashboard

Ahora en la parte final del proyecto, es unir los 4 contenidos en un único panel, para que el usuario acceda de forma fácil y sencilla a los distintos contenidos que se ofrecen.

Se ha utilizado HTML, PHP y CSS, dejando la parte de CSS un poco dinámica, para poder efectuar las diferentes modificaciones oportunas en el futuro para adaptarla de forma responsive.

Se ha realizado una página HTML, que se estructura en 4 apartados, uno para cada contenido, aplicando a través del CSS un movimiento aleatorio a cada contenido, cada vez que se carga la página. Al pulsar en cada cuadro, se abrirá el respectivo contenido, ofreciendo así al usuario toda la información de la que disponemos en este TFG.

El código HTML de la página sería el siguiente:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <!-- Librerías -->
    <meta name="autor" content="Oscar Garrido Chueco" />
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
    <meta name="copyright" content="#" />
    <meta name="robots" content="index, follow" />
    <meta name="keywords" content="dashboard, alcoi" />
    <meta name="description" content="Dashboard Alcoi" />

    <link href="estilo/estilo.css" rel="stylesheet" type="text/css"><!-- hoja de estilo -->

    <title>Dashboard Alcoi</title><!-- titulo -->
  </head>
  <?php
  $nombres = array("autob.php", "tiempo.php", "trafico.php", "historial.php");
  $aleatorio = rand(0, 3);
  <div class="aux_div_dos"><!-- Parte superior segunda barra -->
    <div class="separador_cent_izq">&nbsp;</div>
    <div class="central_medio">&nbsp;</div>
    <div class="separador_cent_der">&nbsp;</div>
  </div><!-- fin Parte superior segunda barra -->
  <div class="separador_blanco">&nbsp;</div><!-- degradado de negro a color web -->
  <div class="aux_div_tres"><!-- cuerpo de la web -->
    <div class="separador_noticias_izq">&nbsp;</div>
    <div class="central_noticias" >

      <div class="arriba_izq"><?php include $nombres[$aleatorio]; $aleatorio++;?></div>
      <div class="arriba_dere"><?php if ($aleatorio > 3 ){ $aleatorio = 0; }include $nombres[$aleatorio]; $aleatorio++;?></div>
      <div class="bajo_izq"><?php if ($aleatorio > 3 ){ $aleatorio = 0; }include $nombres[$aleatorio]; $aleatorio++;?></div>
      <div class="bajo_dere"><?php if ($aleatorio > 3 ){ $aleatorio = 0; }include $nombres[$aleatorio]; $aleatorio++;?></div>

    </div>
    <div class="separador_noticias_der">&nbsp;</div>
  </div><!-- fin cuerpo de la web -->

  </div><!-- fin contenedor web global -->

</body>
</html>
```



El código CSS de la página index.php sería el siguiente(estilo.css):

```

*{
    margin: 0px;
    padding: 0px;
}
html,body{
height:100%;
width:100%;
background-color: #F0F0F0;
}
/* divs de apollo */
.aux_div
{
    width: 100%;
    height: 86px;
    background-image: url("../imagenes/barra_sup.png");
}
.aux_div_dos
{
    width: 100%;
    height: 44px;
    background-image: url("../imagenes/barra_sup_dos.png");
}
.aux_div_tres
{
    width: 100%;
    height:100%;
    overflow: hidden;
}
/* fin divs de apollo */
.separador_blanco
{
    width: 100%;
    height: 17px;
    background-image: url("../imagenes/barra_sup_tres.png");
}

```

Contenido de CSS de index.html DashBoard Alcoi 1



```
/* Parte superior */  
.separador_sup_izq  
{  
    width: 20%;  
    float: left;  
}  
.central_superior  
{  
    height:100%;  
    width:60%;  
    float: left;  
}  
.separador_sup_der  
{  
    width: 20%;  
    float:right;  
}  
.titulo  
{  
    width:100%;  
    height:100%;  
    font-size: 45px;  
    color: #F8811D;  
  
    text-align:center;  
    vertical-align: middle;  
    padding-top: 5px;  
    /*padding-bottom: 20px;*/  
}
```

Contenido de CSS de index.html DashBoard Alcoi 2



```
/* Parte superior fin*/
/* Menu */
.separador_cent_izq
{
    width: 5%;
    float: left;
}
.central_medio
{
    width: 90%;
    display: inline;
}
.separador_cent_der
{
    width: 5%;
    float:right;
}
/* Menu fin*/
/* bloque central */
.separador_noticias_izq
{
    width: 5%;
    height: 100%;
    float: left;
}
.central_noticias
{
    width: 90%;
    height: 100%;
    float: left;
    text-align: center;
}
```

Contenido de CSS de index.html DashBoard Alcoi 3



```
.separador_noticias_der
{
    width: 5%;
    height: 100%;
    float:right;
}
/* bloque central fin */
/* Los cuadro div del cuadro central */
    .arriba_izq{
        width: 49%;
        height: 45%;
        float: left;
        border-radius: 15px;
        margin: 2px auto;
        padding-top: 2px;
        padding-left: 2px;
        background-color: #E5E8E8;
        border: 1px solid black;
    }
    .arriba_dere{
        width: 49%;
        height: 45%;
        float: right;
        border-radius: 15px;
        margin: 2px auto;
        padding-top: 2px;
        padding-left: 2px;
        background-color: #E5E8E8;
        border: 1px solid black;
    }
}
```

Contenido de CSS de index.html DashBoard Alcoi 4

Se obtiene un SITE web compuesto por un panel y 4 apartados, el cual cada uno realiza una redirección a su respectiva sección de una forma cómoda y eficaz.

Imagen final `index.php` con los 4 contenidos:



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA
CAMPUS D'ALCOI



Dashboard Alcoi



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA
CAMPUS D'ALCOI



Si quiere ver el estado actual del Trafico Pulse [Aqui](#)



Si quieres ver el estado actual de los autobuses Pulse [Aqui](#)

Tabla de prediccion				
	2017-06-02	2017-06-03	2017-06-04	2017-06-05
Temperatura mínima y máxima	15 / 29	17 / 29	13 / 26	11 / 23
Sensación mínima y máxima	15 / 29	17 / 29	13 / 26	11 / 23
Humedad relativa mínima y máxima	40 / 65	35 / 65	45 / 100	35 / 100
Índice ultravioleta máxima	9	9	8	9

Fecha y Hora actual

06/02/2017 6:26pm
 Día actual del año:152
 Numero de dias que quedan para finalizar el año: 213
 Cuantas semanas tiene el año:2017 - 52



Si quiere ver el estado actual del Trafico Pulse [Aqui](#)

Imagen Final DashBoard Alcoi

Valoraciones

Valorando el proyecto desde el punto de vista lectivo, ha sido muy variado respecto a los lenguajes utilizados y muy completo a la hora de combinar las diferentes arquitecturas conforme a nuestras necesidades, para finalmente poder analizar, trata y mostrar los diferentes contenidos de los que disponemos.

Este TFG aporta una utilidad simple y sencilla de mostrar una gran cantidad de datos, de forma clara y completa para el usuario o instituciones, pudiendo así realiza los estudios pertinentes y las posibles mejoras de la ciudad. La usabilidad del proyecto es alta, puesto que se pone a disposición de todos los usuarios a través de la URL: <http://ajualcoi.org/dashboard/>

Además, al estar realizado en código abierto, y sin necesidad de infraestructura en la parte del servidor, es fácilmente migrable a cualquier otro servidor, localidad y servicios.

El estudio de las diferentes fases del proyecto y las diferentes disciplinas tanto en el FRON END como en el BACK END han ayudado a conocer mejor las diferentes funciones a un lado y otro del servidor, para así poder realizar nuestro DashBoard sin necesidad de utilizar plantillas estándar o similares de lenguaje propietario como AngularJS, NodeJs y derivados.

En resumidas cuentas personalmente, ha sido un proyecto donde he podido investigar y aprender diferentes formas de tratamiento de datos, alternar entre los diferentes lenguajes para adaptarlo a las necesidades del TFG, y creo que es un utilidad muy buena con posible continuación en el futuro y de una alta usabilidad por los diferentes usuarios y estamentos, y al estar en un dominio como el de www.ajualcoi.org, disponible para cualquier usuario del mundo.

Bibliografía

Información general:

<https://stackoverflow.com/>

Conexión socket:

<https://www.programarya.com/Cursos/Java-Avanzado/Socket>

<http://nereida.deioc.ull.es/~cleon/doctorado/doc06/doc06/html/node9.html>

Consultas MySQL:

<http://www.w3resource.com/mysql/comparision-functions-and-operators/coalesce-function.php>

https://www.w3schools.com/sql/sql_isnull.asp

Paso de variables JS:

<http://www.devservice.es/blog/como-pasar-variables-javascript-a-php/>

<http://www.forosdelweb.com/f13/pasar-variable-php-funcion-javascript-1005947/>

Forma, POST, GET y Radio Buttons:

https://www.w3schools.com/tags/att_form_method.asp

https://www.w3schools.com/tags/ref_httpmethods.asp

Google Maps:

<https://developers.google.com/maps/documentation/javascript/?hl=es-419>

<https://productforums.google.com/forum/#!forum/maps>