



ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN

Titulación :

INGENIERO TÉCNICO DE TELECOMUNICACIÓN,
ESPECIALIDAD EN SONIDO E IMAGEN

Título del proyecto:

DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN
MÓVIL HÍBRIDA EMPLEANDO EL FRAMEWORK DE JQUERY
MOBILE Y PHONEGAP.

Adela Santamaría Gimeno

Marko Galarza Galarza

Pamplona, Septiembre 2016

Índice de contenido

1. Objetivo.....	2
2. Introducción.....	4
2.1 Breve descripción:.....	5
2.2 Uso del smartphone y de internet móvil:.....	5
2.3 Diferentes Tipos de Aplicaciones Móviles:	8
- Aplicaciones web:.....	8
- Aplicaciones nativas:.....	9
- Aplicaciones híbridas:.....	9
Nuestra elección:.....	10
3. Tecnologías utilizadas.....	11
3.1 HTML5:.....	12
3.2 CSS3:.....	13
3.3 JAVASCRIPT:.....	14
3.4 PHP:.....	14
3.5 JQUERY MOBILE:.....	15
3.6 PHONEGAP:.....	16
3.7 XAMPP:.....	17
3.8 GOOGLE MAPS:.....	17
4. Desarrollo.....	18
4.1 Esquema y maquetación:.....	19
4.2 Instalación de programas:	19
4.3 Estructura y desarrollo de la página paso a paso.....	20
4.3.1 HTML 5.....	21
4.3.2 CSS 3.....	22
4.3.3 Responsive Web Design.....	25
4.3.4 Vídeo.....	27
4.3.5 Geolocalización.....	28
4.3.6 Formulario de contacto.....	30
4.3.7 jQuery Mobile.....	32
4.3.8 PhoneGap.....	33
5. Líneas Futuras.....	36
6. Conclusiones.....	38
7. Bibliografía.....	40

1. Objetivo

1.Objetivo

El objetivo principal de este proyecto es desarrollar una aplicación para distintos dispositivos móviles.

El tema escogido para llevarla a cabo es el de los aparcamientos en nuestra ciudad, Pamplona. Abarca tanto desde los aparcamientos subterráneos, más conocidos por todos nosotros hasta los aparcamientos gratuitos que ofrece la ciudad, pasando también por las zonas de estacionamiento restringido y regulado. El Usuario podrá obtener información acerca de localizaciones, horarios, tarifas y más datos de interés, además de poder contactar con el creador para pedir mas información o hacer sugerencias.

Está dirigida a todo tipo de público, que puede estar o no familiarizado con la tecnología, por lo tanto el diseño tratará de ser sencillo e intuitivo para facilitar su uso.

La tecnología que queremos emplear para su desarrollo tecnologías web, es decir, Html 5, CSS3, y JavaScript, principalmente, para luego servirnos de los framework de jQuery Mobile y PhoneGap para transformar nuestra aplicación web en una aplicación híbrida. Más adelante veremos qué es cada una de estas tecnologías y conceptos y por qué los hemos elegido.

Como queremos que pueda ser empleada en todo tipo de dispositivos, usaremos un diseño adaptable (responsive web design) para que nos sea cómoda de ver y manejar según los diferentes tamaños de pantallas.

Tras detallar el proceso realizado hasta la obtención de dicha aplicación, relataremos posibles mejoras a tener en cuenta en un futuro así como las conclusiones obtenidas tras la realización.

2. Introducción

2.Introducción

2.1 Breve descripción:

El objetivo de este proyecto es conseguir una aplicación híbrida para distintos dispositivos móviles. Nos centraremos sobre todo en el sistema operativo de Android, ya que es muy difícil abarcar todos. No se ha contemplado la plataforma iOS ya que requiere del uso de equipos Mac y no ha sido posible disponer de uno para la realización del proyecto. El tema escogido para llevarla a cabo es el de los aparcamientos en nuestra ciudad, Pamplona.

2.2 Uso del smartphone y de internet móvil:

Pese a la crisis vivida en los últimos años, el sector de la telefonía móvil es un sector que no se ha visto gravemente afectado sino que está en auge. La tecnología juega un papel tan fundamental en nuestra sociedad, que ya no podemos concebir la idea de un mundo sin internet.

La utilización de los dispositivos móviles, tanto teléfonos como tabletas, también debe su crecimiento a las aplicaciones. Muchas empresas se dieron cuenta de que el futuro de su negocio estaba en internet. Rara es la empresa o negocio que hoy en día no posee una página web informando sobre sus productos. El mercado se va actualizando y cada vez son más y más los usuarios que se suman a la compra on-line, en lugar de tener que ir personalmente a la tienda a hacerlo. Las compras realizadas desde el móvil crece tres veces más rápido que el e-commerce y está desplazando a las compras que se realizan desde el equipo de escritorio en todo el mundo. Los usuarios de banca móvil no dejan de aumentar: A finales de 2014, 800 millones de personas accedían habitualmente a su banco a través del smartphone. Las perspectivas afirman que esta cifra se duplicará en los próximos 4 años.

El mundo de las aplicaciones está pues en constante crecimiento. Cada mes se lanzan 40.000 nuevas aplicaciones al mercado del móvil. Los sistemas Google Play Store y Apple Store ofrecieron en 2014 más de 2 millones de apps a los usuarios. Los usuarios de smartphones prefieren usar aplicaciones a navegar por Internet.

Según un estudio realizado por la AIMC (Asociación para la Investigación de los Medios de Comunicación), el móvil es el principal dispositivo de conexión a internet para 9 de cada 10 internautas.

Además en 2014 ya, el tiempo medio diario que un usuario pasa en el teléfono móvil superó al que pasa delante del televisor. Este año se vendieron 1.245 millones de smartphones en el mundo, lo que supuso un aumento del 28,4% respecto al año anterior.

Del total de líneas móviles activas en España, un 87% son smartphones. Lo que justifica que ya no empleamos el teléfono móvil sólo para llamar o enviar mensajes, sino que buscamos mucho más en nuestro dispositivo. El 100% de los españoles que se conectan a Internet lo hacen también desde su smartphone y el 90% de los usuarios se conecta todos o casi todos los días. Los consumidores se muestran cada vez, más dependientes de sus smartphones: El 40% de las personas mira la pantalla de su teléfono más de 50 veces al día mientras que el 70% mira su teléfono durante la primera media hora después de haberse despertado.

Los ingresos de las empresas gracias a la publicidad en teléfonos móviles han pasado de 400 millones de dólares en 2009 a 6.600 millones en el 2014 siendo las previsiones para 2016 de casi 12.000 millones. El 80% de las empresas planea aumentar sus presupuestos de marketing digital en los próximos 18 meses, incluyendo el marketing móvil.

En el mundo pues, el smartphone es el dispositivo más utilizado para acceder a Internet. Más del 70% de los internautas lo usan y Android es el sistema operativo que domina este mercado, con 3 de cada 4 personas, pero bajando ligeramente frente al año anterior. Le sigue de lejos Apple, que sin embargo ganó cuota de mercado el año pasado alcanzando casi el 20%.

	2011	2012	2013	2014
 Android	52,8%	70,4%	78,2%	76,6%
 iOS	23,0%	20,9%	17,5%	19,7%
 Windows Phone	1,5%	2,6%	3,0%	2,8%
 BlackBerry	8,1%	3,2%	0,6%	0,4%
Otros	14,6%	2,9%	0,8%	0,5%

Fig.1: cuota de penetración de los sistemas operativos en smartphones en el mundo.

Siguiendo con los datos curiosos, casi la mitad de las personas (46%) mira su teléfono móvil en los primeros 5 minutos tras levantarse y un 13% lo hace justo inmediatamente. Una hora después de haberse levantado, 9 de cada 10 españoles ya ha echado un vistazo a su teléfono.

Si comparamos el uso de smartphones y tablets, el móvil también es el dispositivo preferido para acceder a las redes sociales, informarse o hacer búsquedas en Internet. La tablet, por su parte, es el dispositivo preferido para leer y jugar. Además, un 41% de los usuarios afirma utilizar ambos dispositivos al mismo tiempo.

Las categorías de aplicaciones con más crecimiento en el último año en Apple Store fue la de negocios – con un aumento del 210% entre enero y octubre de 2014 – seguida por la sección de Comida y bebida debido al auge de las aplicaciones para pedir comida a domicilio. En GooglePlay, por su parte, fue la categoría de juegos la que

experimentó un mayor crecimiento, con un 260% de aumento en el mismo periodo, seguida de las aplicaciones para fotografía.

Los usuarios de tablets son más activos a la hora de descargar aplicaciones y activar permisos de las apps (ubicación o notificaciones) que los usuarios de smartphones. Las razones principales para descargar una aplicación son principalmente por conveniencia de acceso (un 65%) rapidez de acceso a la información (51%), conseguir contenidos más significantes (41%) o por fidelidad a la marca (40%).

En cuanto a la temática que ocupa nuestra aplicación, el 81% de los conductores está de acuerdo en que su próximo vehículo tendrá un alto nivel de conectividad y servicios para sus móviles. La tendencia apunta a que en el año 2020 el 90% de los vehículos dispondrá de conectividad.

** Cifras extraídas de “Informe Mobile de Ditrendia 2015”*

2.3 Diferentes Tipos de Aplicaciones Móviles:

Una vez visto el auge de las aplicaciones y el uso del smartphone, pasamos a conocer los distintos tipos de aplicaciones que existen y a decantarnos por uno para nuestra aplicación:

- Aplicaciones web:

Es una aplicación desarrollada en lenguajes muy conocidos por la mayoría de los programadores, como pueden ser HTML , CSS y JavaScript. Una de sus principales ventajas es su sencillez o facilidad de programar. Otra de sus principales ventajas es que puede ser utilizada en cualquier dispositivo independientemente del sistema operativo que este utilice.

Se ejecutan dentro del propio navegador, por lo que precisan de conexión internet. La ventaja es que no necesita instalación, por lo que no necesitamos tener gran espacio libre en el almacenamiento de nuestro dispositivo. Además el usuario siempre dispone de la versión más actualizada.

Inconveniente: no aparecen en las tiendas de aplicaciones, lo que dificulta su comercialización y distribución; pero tampoco necesitan de ninguna aprobación externa para publicarse.

Son una buena opción si lo que queremos es adaptar una web que ya tuviéramos a un formato móvil.

- Aplicaciones nativas:

Son aquellas que se desarrollan específicamente para un sistema operativo determinado, por lo que si quieres que tu aplicación este disponible para todas las plataformas (iOS, Android, Windows Mobile...) deberás crear una app para cada lenguaje de dichos sistemas operativos.

La principal ventaja que ofrecen es la posibilidad de acceder a todas las características del dispositivo: Cámara , acelerómetro, Gps, agenda... por lo que la experiencia de usuario será más positiva. Otra ventaja es que no precisan de conexión a internet para poder utilizarlas. La descarga e instalación se puede realizar a través de las tiendas de aplicaciones, lo que facilita su distribución. El principal inconveniente es que tiene unos costos y tiempos de desarrollo elevados.

- Aplicaciones híbridas:

Son la combinación de los dos tipos de aplicaciones explicados anteriormente. Recoge lo mejor de cada una de ellas, pero también hereda los contras de ambas.

Se crean también utilizando lenguajes web como HTML, CSS y JavaScript, lo que permite su uso en diferentes plataformas, pero también ofrecen la posibilidad de acceder a gran parte de las características hardware del dispositivo.

Otra ventaja es que es posible comercializarlas en las tiendas de aplicaciones.

Nuestra elección:

Con esta información sopesamos qué tipo de aplicación nos conviene mas en este caso y por tanto qué tipo de programación vamos a emplear en su desarrollo. Debemos tener en cuenta a qué público va dirigida, si el diseño será complejo o sencillo, el coste que queremos asumir, si queremos acceso a todas las funcionalidades del dispositivo, etc.

En nuestro caso hemos elegido una app híbrida. Nuestro objetivo era conocer los lenguajes web (en mi caso desde cero) y estudiar un poco más a fondo el marco de trabajo de PhoneGap. La idea era desarrollarla para la plataforma Android y además vamos a poder acceder algunas funcionalidades del dispositivo como puede ser el GPS.

Los principales dispositivos a los que está destinada nuestra aplicación son principalmente dispositivos móviles, más en concreto smartphones (teléfonos inteligentes).

3. Tecnologías utilizadas

3. Tecnologías utilizadas:

3.1 HTML5:

Es la 5ª revisión del lenguaje estándar para web (HyperText Markup Language) . Pretende reemplazar el actual Html, corrigiendo problemas con los que los desarrolladores web se encuentran y rediseñar el código, con nuevas funcionalidades, aunque a día de hoy puede presentar un problema, por el tema de compatibilidades con los diferentes navegadores.



Elimina etiquetas no necesarias e introduce :

- Canvas: permite dibujar formas y responder a la interacción del usuario sin instalar plugins, mediante las funciones de una API (Interfaz de Programación de Aplicaciones)
- Aplicaciones Web Offline: mediante una API es posible desarrollar aplicaciones web sin necesidad de estar conectados a Internet.
- Geolocalización: Por medio de una API también es posible localizar usuarios geográficamente.
- Mejoras en formularios: sin tener que emplear obligatoriamente JavaScript.

El principal cambio en semántica se debe a la desaparición de las etiquetas `<div>` `</div>`, en versiones anteriores empleadas para casi todas las secciones y ahora sustituidas por:

- `<header>` `</header>`
- `<nav>` `</nav>`
- `<section>` `</section>`
- `<article>` `</article>`
- `<aside>` `</aside>`
- `<footer>` `</footer>`
- `<audio>` y `</video>`

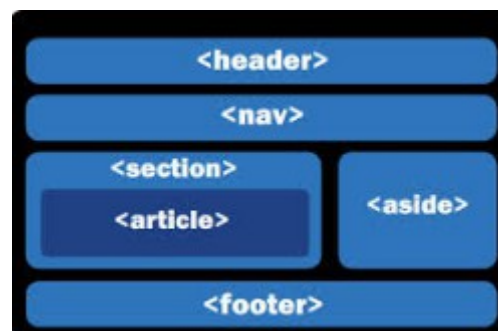


Fig.2: Estructura Semántica Html5

3.2 CSS3:

Hojas de estilo, permite dar aspecto y formato a una web (el diseño). En su tercera versión añade nuevas funciones capaces de crear animaciones, efectos de texto, transiciones, etc. y sustituir por completo el uso de flash.



Permite a los diseñadores mantener un control mucho más preciso sobre la apariencia de las páginas, y evitar así la necesidad de recurrir a ciertos trucos o apañíos, con lo que se evita el problema de modificar el código fuente de páginas o aplicaciones web.

Algunas de las mejoras más importantes son:

- Creación de bordes
- Fondos
- Color
- Texto y fuentes
- Degradados
- Animaciones

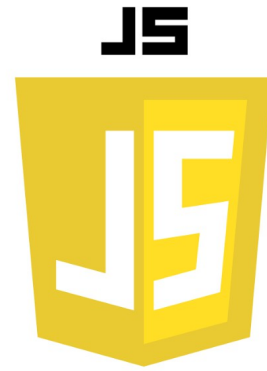
Pero al igual que ocurre con HTML5 no todo son ventajas, ya que algunos navegadores no tienen la capacidad de mostrarlas y tenemos que recurrir a ciertos prefijos en determinados navegadores para ver las mejoras introducidas:

- moz-
- webkit-
- o-
- ms-

Prefijo	Familia de navegadores a los que aplica
-webkit-	Chrome, Safari, Android, iOS 
-moz-	Firefox 
-o-	Opera 
-ms-	Microsoft Internet Explorer 

3.3 JAVASCRIPT:

Es un lenguaje de programación orientado a objetos. Permite mejoras en la interfaz de usuario. Con este lenguaje podemos acceder a la estructura del DOM de la web y modificar su contenido en tiempo real e interactuar dentro de la propia página.



El principal objetivo es dotar de dinamismo las páginas web, incorporando efectos, animaciones, acciones tras ejecutar acciones en la aplicación (como pulsar un botón).

Aunque no es el mismo tipo de código, interactúa con el código Html5 y puede incluso añadirse dentro de él, siempre y cuando se etiquete y se identifique dicha parte de código como JavaScript, para que el navegador la pueda interpretar correctamente.

3.4 PHP:

Es un lenguaje de código abierto del lado del servidor. Permite la conexión con servidores de bases de datos como MySQL. Es de relevante utilidad, y en el proyecto se ha empleado principalmente para el envío de correos a través de un formulario.



Cuenta con la ventaja de ser compatible con todos los navegadores.

3.5 JQUERY MOBILE:

Es el más nuevo de los productos de la familia jQuery. Es una librería que nos permite el desarrollo rápido de sitios web para móviles. Este framework nos provee de



ciertas herramientas que nos hacen la tarea de crear una página mucho más sencilla, usable y accesible. Tiene un tamaño muy reducido: toda la librería comprimida no ocupa más de 12 K.

Tiene una serie de componentes de interfaz de usuario y una serie de utilidades para crear páginas orientadas para móvil, responder a interacción en pantallas táctiles, etc. Con muy poco código Javascript en jQuery Mobile se pueden crear sitios dinámicos que se comportan muy bien cuando los usas en la mayoría de los fabricantes de dispositivos. Pero es necesario tener jQuery para poder usar jQuery Mobile.

Nos permite generar aplicaciones cuya apariencia será siempre la misma independientemente del dispositivo desde el que acceda un usuario siempre que este usuario acceda desde un dispositivo que acepte HTML5. Antes un desarrollador tenía que programar según para qué dispositivo en concreto y eso alargaba los tiempos de desarrollo y mantenimiento de los sitios webs; ahora con JQueryMobile, evitamos conocer la lógica específica de cada dispositivo y nos centramos en la programación para un solo fin, el navegador de un teléfono móvil.

3.6 PHONEGAP:

Producido por Nitobi, y comprado posteriormente por Adobe Systems. Es una distribución de Apache Cordova. La aplicación se



llamó en un principio "PhoneGap", y posteriormente "Apache Callback". Ambos sistemas tienen funciones prácticamente idénticas, la diferencia principal entre Apache Cordova y PhoneGap es que el segundo tiene acceso a servicios de compilación en la nube proporcionados por Adobe Creative Cloud.

Apache Cordova es un software de código abierto y tanto este como PhoneGap pueden ser utilizados libremente en cualquier aplicación sin necesidad de atribución o licencias de ningún tipo.

Las posibilidades de hacer una aplicación móvil se amplían gracias al uso de PhoneGap Build, un poderoso compilador en la nube.

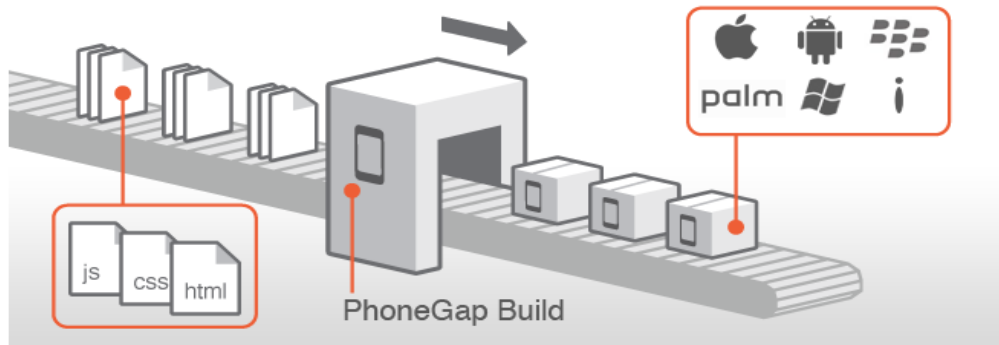
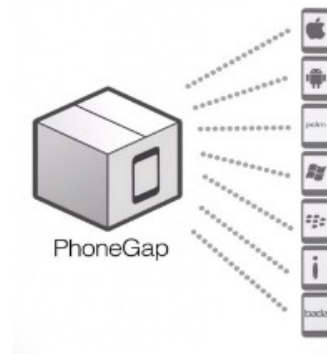


Fig.3: Esquema funcionamiento PhoneGap Build

Es el complemento perfecto para todo lo que significa el framework también conocido como Apache Cordova, pues el acceso a esta poderosa herramienta resulta ser una tarea de poca complejidad.

El framework trabaja con siete plataformas, pero solo compila seis:

- iOS
- Android
- BlackBerry OS
- Symbian
- Web OS
- Windows Phone



Por la complejidad de cada sistema operativo, en este proyecto sólo nos vamos a centrar en Android.

Es un paquete de librerías que permite empaquetar aplicaciones HTML5 de manera que puedan ser usadas como apps para móviles, pero también tiene sus limitaciones.

3.7 XAMPP:

Es un servidor independiente que nos brinda la opción de crear un servidor local e independiente en nuestro ordenador con el que poder probar nuestra aplicación antes de subirla a un servidor web.



3.8 GOOGLE MAPS:

Es un servicio de Google que nos proporciona mapas desplazables de todo el mundo y la opción de poder añadir rutas o marcadores. Es una herramienta de mucha importancia en esta aplicación.



4. Desarrollo

4. Desarrollo:

4.1 Esquema y maquetación:

Primero se elabora un esquema o croquis de cómo se pretende que sea nuestra aplicación: de cuántas páginas va a constar, qué información va a ofrecer, cómo será la navegación, etc.

Primero se empieza por crear una página web y posteriormente se transformará en aplicación.

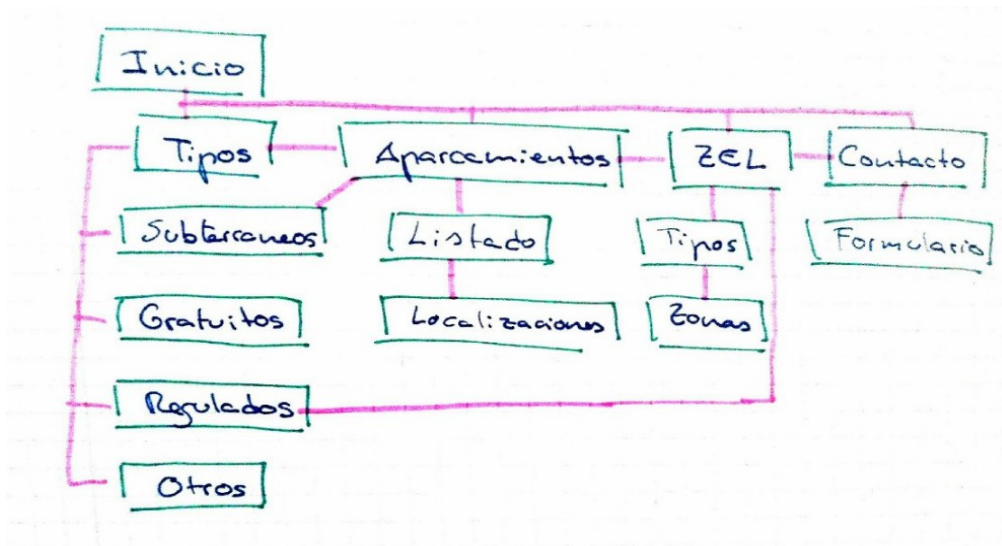


Fig.4: esquema inicial de la aplicación

4.2 Instalación de programas:

A continuación y con la idea ya en la cabeza, se lleva a cabo la documentación sobre todas las tecnologías que se van a emplear para llevar a cabo el proyecto, que como ya se ha comentado anteriormente, prácticamente todas son nuevas para mi, lo que supone un reto aún mayor.

Pese a la recomendación del tutor por usar DreamWeaver para empezar, se decide emplear un editor de texto para comenzar a programar, ya que así se va a aprender más y mejor que con las facilidades que ofrece el DreamWeaver de poder hacerse directamente diseñando en lugar de programando. En mi caso se he escogido el Notepad++.

El nombre escogido para la aplicación ha sido Pamking, un juego de palabras entre Pamplona y Parking, que es de lo que va la aplicación como anteriormente ya se ha comentado.

El logo se ha creado con el mismo Paint, para evitar problemas de copyright, aunque la intención real de esta aplicación no es comercializarla (a día de hoy ya hay alguna con temática similar) sino ver todas las posibilidades que nos ofrece el nuevo estándar de HTML5 y PhoneGap.

Es por ello que la aplicación contiene por ejemplo vídeo, una de las novedades de HTML5, pese a no ser estrictamente necesario y aumentar así el peso de la aplicación.

4.3 Estructura y desarrollo de la página paso a paso

4.3.1 HTML 5

Se va a empezar mostrando una imagen de una de las páginas de la aplicación para posteriormente ir detallando cómo se ha ido llegando hasta ahí.

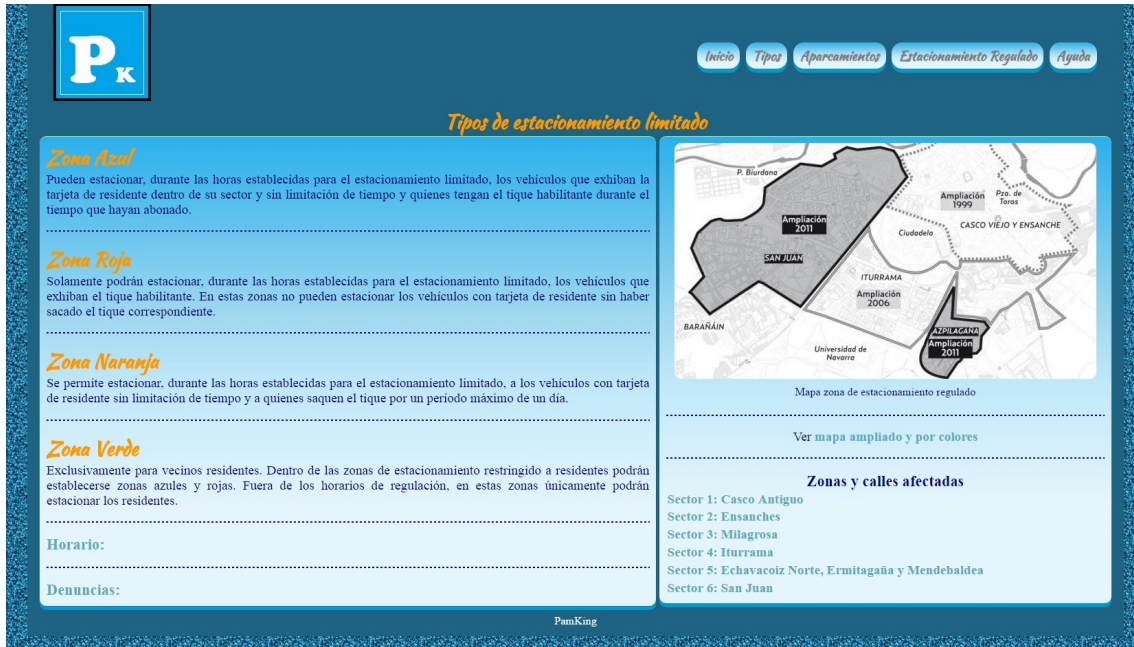


Fig.5: Captura de pantalla de la aplicación web

Nada más empezar con el código, surge la necesidad de codificar correctamente los ficheros para mostrar los caracteres acentuados de forma correcta. Esto puede realizarse con los comandos “á” “é” por ejemplo, y nos saldría en pantalla “á” y “é” respectivamente. Posteriormente se comprueba que eso no es necesario si ponemos el formato de codificación en español, que por defecto viene en inglés. Esto se consigue con el siguiente comando en la cabecera: <meta charset="utf-8"/>

```

1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <meta charset="utf-8"/>
5     <meta name="description" content="Futura aplicación móvil"/>
6     <meta name="viewport" content="width=device-width,initial-scale=1"/>
7
8     <link rel="stylesheet" href="css/estilos.css"/>
9     <link rel="stylesheet" href="css/sliderman.css"/>
10
11
12     <script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1/jquery.min.js" >/script>
13     <!-- <script type="text/javascript" src="js/acordeon.js">/script -->
14     <script type="text/javascript" src="js/sliderman.1.3.7.js">/script>
15
16     <link rel="shortcut icon" type="image/x-icon" href="img/vintagecar.ico"/>
17     <link rel="author" type="text/plain" href="humans.txt"/>
18     <link rel="sitemap" type="application/xml" title="Sitemap" href="sitemap.xml"/>
19     <title>PamKing</title>
20   </head>
21   <body>
22     <header>
23       <h1>
24         <a href="index.html">
25           
26         </a>
27       </h1>
28       <nav>
29         <ul>
30           <li><a href="index.html" class="botoncillo">Inicio</a></li>
31           <li><a href="tipos.html" class="botoncillo">Tipos</a></li>
32           <li><a href="lista_sub.html" class="botoncillo">Aparcamientos</a></li>
33           <li><a href="lista_zon.html" class="botoncillo">Estacionamiento Regulado</a></li>
34           <li><a href="contacto.html" class="botoncillo">Ayuda</a></li>
35         </ul>
36       </nav>
37     </header>
38

```

Fig.6: Ejemplo de código Html5

4.3.2 CSS 3

Como puede observarse nuestra hoja de estilos la tenemos en un documento externo, es una de las opciones para hacerlo. La otra sería incluirlo dentro del código Html pero así me parece más cómodo y limpio.

También se crea un *sitemap.xml* que es un archivo en el cual se puede enumerar las páginas de las que consta nuestro sitio web para informar a los motores de búsqueda (Google por ejemplo), sobre la organización y contenido del mismo, para que se pueda rastrear el sitio de una manera más inteligente.

El menú de navegación contenido dentro de la etiqueta `<nav>` con un formato de lista, es muy intuitivo, pero además gracias a css3 se le ha podido dar múltiples valores para hacer unos botones más funcionales y llamativos:

```

155  nav a.botoncillo{
156      background: navy; /*border: 1px solid*/
157      background-image: linear-gradient(to bottom, rgba(42,176,237,1) 0%,
158      rgba(159,216,239,1) 38%, rgba(191,232,249,1) 52%, rgba(228,245,252,1) 78%);
159      border-radius: 15px;
160      box-shadow: inset 0px 1px 0px #49B7F2, 0px 6px 0px #09c;
161      color: grey;
162      display:block;
163      font-family: "kaushan_scriptregular", Helvetica, Verdana;
164      font-size:20px bold;
165      padding:5px 10px ;
166      position: relative;
167      text-decoration:none;
168      transition:all 0.5s ease-in;
169  }
170
171  nav a.botoncillo:hover {
172      /*border: 0.1em dashed navy;*/
173      background-image: linear-gradient(to bottom, rgba(228,245,252,1)
174      22%, rgba(191,232,249,1) 48%, rgba(159,216,239,1) 62%, rgba(42,176,237,1) 100%);
175      color:navy;
176  }
177
178  nav a.botoncillo:active {
179      /*background: /*#6fba26;*/
180      background-image: linear-gradient(to bottom, rgba(228,245,252,1)
181      22%, rgba(191,232,249,1) 48%, rgba(159,216,239,1) 62%, rgba(42,176,237,1) 100%);
182      box-shadow: inset 0px 1px 0px #3d6f0d, inset 0px -2px 0px #136E9F;
183      color: grey;
184      text-shadow: 0px 1px 1px rgba(255,255,255,0.3);
185      top:7px;
186  }
187
188  nav a.botoncillo::before {
189      background-color: navy;
190      content:"";
191      display:block;
192      position:absolute;
193      width:100%;
194      height:100%;
195      padding-left:2px;
196      padding-right:1px;
197      padding-bottom:4px;
198      left:-2px;
199      top:5px;
200      z-index:-1;
201      border-radius: 15px;
202      box-shadow: 0px 1px 1px #000033;
203  }
204
205  nav a.botoncillo:active::before {
206      top:-2px;
207  }

```

Cascade Style Sheets File

Fig.7: ejemplo de código css para unos botones



Fig.8: Ejemplo de dichos botones

Como podemos observar en ambas imágenes se han tenido en cuenta múltiples factores y valores. Por ejemplo en esta última teníamos el ratón puesto sobre “aparcamientos” para poder observar qué sucede al hacer “hover”: se invierten los colores para parecer que el botón esté seleccionado, se cambia el color de la fuente, etc.

Como podemos observar, no tenemos una fuente de letra muy común, eso es porque html5 y css3 nos ofrecen la posibilidad de poner una que nos guste e “instalarla”

```

6  @font-face {
7      font-family: "kaushan_scriptregular";
8      src: url("../fonts/kaushanscript-regular-webfont.woff");
9      src: url("../fonts/kaushanscript-regular-webfont.woff?#iefix") format("embedded-opentype"),
10     url("../fonts/kaushanscript-regular-webfont.woff") format('woff'),
11     url("../fonts/kaushanscript-regular-webfont.ttf") format('truetype'),
12     url("../fonts/kaushanscript-regular-webfont.svg#kaushan_scriptregular") format("svg");
13     font-weight: normal;
14     font-style: normal;
15 }
16
17 /*poner otra fuente*/
18

```

Fig.9: Css para poner otro tipo de fuente diferente a los habituales.

Se pueden extraer ésta y muchas más de los recursos que nos ofrece por ejemplo google web fonts o www.fontsquirrel.com.

En la anterior imagen también se puede observar cómo se escriben comentarios en el código para después entenderlo o encontrar algo más fácilmente:

- Para comentar dentro del lenguaje de Html, se tienen que escribir los siguientes caracteres: <!-- comentario -->

- Para comentar en Css, existen los caracteres: `/* comentario */`

- Para comentar en Php, se tienen 2 opciones, los de una línea: `//comentario` y los multilinea: `/* comentario*/`

- Para comentar en el lenguaje Javascript, se pueden utilizar todos los métodos del lenguaje Php y Html, es decir:

```
<!-- comentario -->
```

```
//comentario
```

```
/* comentario*/
```

4.3.3 Responsive Web Design



Fig.10: Ejemplo de los mismos botones en pantalla más pequeña

Como se puede observar en la imagen anterior, se trata de los mismos botones que en la Figura 8, pero vistos desde una pantalla de menores dimensiones. Pueden verse así porque se ha trabajado con un diseño adaptable (Responsive Web Design). Esto quiere decir que para que sea visible en cualquier dispositivo (Smartphone, Tablet..) se tiene que adaptar el contenido a las dimensiones del dispositivo que se use en el momento. Por lo tanto, el contenido (imágenes, vídeos, texto...) se tendrán que hacer más grandes o más pequeños y no tendrán que perder calidad, observándose como si estuvieran en con el tamaño original. Para ello hay que tener en cuenta la elección del zoom en los navegadores móviles y usar la meta-etiqueta “viewport” en la cabecera de nuestro código:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

También se han tenido en cuenta las “media queries”, que son unas sentencias que aplicar a nuestro diseño si se cumplen unos ciertos requisitos que nosotros les pongamos:

```
403
404 @media screen and (max-width:600px){
405     form{
406         width:70%;
407     }
408
409     div#mapa{
410         width: 300px;
411         height:200px;
412     }
413
414     nav li{
415         display: inline-block;
416         margin: 2px;
417         width: 98%;
418     }
419
420     section#principal, article#descripcion-tipos, article#servicio2{
421         margin-bottom: 0.2em;
422         margin-top: 0.2em;
423     }
424
425     section#principal, aside, #descripcion-tipos, #multimedia-tipos, #servicio2, #servicio3{
426         display: block;
427         width: 95%;
428     }
429
430     #img-servicios{
431         display:none;
432     }
433
434     div.portafolio article {
435         display: block;
436         margin: .5em;
437         padding: 0;
438     }
439 }
```

Cascade Style Sheets File

Fig.11: Ejemplo de media queries en nuestro código css



Fig.11b: Ejemplo web adaptativa

Como se aprecia en la imagen también debemos tener en cuenta que si un dispositivo está “rotado”, el ancho de su pantalla también cambia.

Este método aún se encuentra en fase de desarrollo, no paran de salir novedades al respecto y hay que saber cual es la que más nos conviene en cada momento, para poder resolver las complicaciones que surjan.

4.3.4 Vídeo



Fig.12: Imagen de un vídeo insertado en la aplicación.

Como ya se ha comentado anteriormente, una de las novedades de HTML5 es la posibilidad de incluir contenido multimedia directamente en el código tanto para audio como para vídeo, sin necesidad de tener que recurrir a nada externo, aunque posteriormente con JavaScript se puede controlar la reproducción o la pausa, el volumen, etc.

```

50  <article id="multimedia-tipos">
51
52  <h3>
53  Lo complicado de aparcar en Pamplona.
54  </h3>
55
56  <video id="video-tipos" controls preload >
57  <source src="vdo/APARCAR.mp4" type="video/mp4" />
58  <source src="vdo/APARCAR.webm" type="video/webm" />
59  <source src="vdo/APARCAR.ogv" type="video/ogg" />
60  </video>
61  </article>

```

Fig.13: código HTML5 de introducción de vídeo

Comentar la importancia de subir el video en los diferentes formatos que se muestran , porque dependiendo del navegador en el que se abra soportará uno u otro formato. Resaltar también la importancia de añadir los controles, para poder parar la reproducción cuando queramos sin que llegue a ser molesta por no poder controlarla.

4.3.5 Geolocalización



Fig.14: Ejemplo de ruta a trazar para llegar a un aparcamiento

Otra de las novedades más importantes es la geolocalización. En nuestra app se ha implementado con ayuda de JavaScript y aparte de encontrar nuestra ubicación en cada momento, y la ubicación del parking que se le señale, se marca la ruta a seguir para llegar hasta él.

```

35 <section id="contenido" >
36 <h2>Aparcamiento Subterráneo Plaza de Toros</h2>
37 <article id="descripcion-tipos">
38 <h3>Mapa interactivo:</h3>
39 <div id="mapa" style="width: 700px; height: 500px;">
40 --Aquí va nuestro mapa--
41 </div>
42 <script type="text/javascript">
43 //NAMESPACE --> google.maps.ALGO
44 var divMapa = document.getElementById('mapa');
45 navigator.geolocation.getCurrentPosition( fn_ok, fn_mal );
46 function fn_mal() { }
47 function fn_ok( rta ){
48     var lat = rta.coords.latitude;
49     var lon = rta.coords.longitude;
50
51     var gLatLon = new google.maps.LatLng( lat, lon );
52     var objConfig = {
53         zoom: 17,
54         center: gLatLon
55     }
56     var gMapa = new google.maps.Map( divMapa, objConfig );
57     var objConfigMarker = {
58         position: gLatLon,
59         map: gMapa,
60         title: "ud está aquí"
61     }
62     var gMarker = new google.maps.Marker( objConfigMarker );
63
64     var gCoder = new google.maps.Geocoder( );
65     var objInformacion = {
66         address: ' Calle de Emilio Arrieta s/n, 31002 Pamplona, Navarra'
67     }
68     gCoder.geocode( objInformacion, fn_coder );
69
70     function fn_coder( datos ){
71         var coordenadas = datos[0].geometry.location; //obj LatLng
72         var config = {
73             map: gMapa,
74             position: coordenadas,
75             title: 'Aparcamiento Plaza de Toros'
76         }
77         var gMarkerDV = new google.maps.Marker( config )
78     }
79     var objConfigDR = {
80         map: gMapa
81     }
82     var objConfigDS = {
83         origin:gLatLon, //LatLng - String domicilio
84         destination:objInformacion.address,
85         travelMode:google.maps.TravelMode.DRIVING
86     }
87
88     var ds = new google.maps.DirectionsService( ); // obtener coordenadas

```

Fig.15: Ejemplo de código para la geolocalización

La configuración en firefox ha requerido niveles de seguridad altos. Es por ello que se lanza una ventana requiriendo la confirmación del usuario como puede observarse en la imagen:

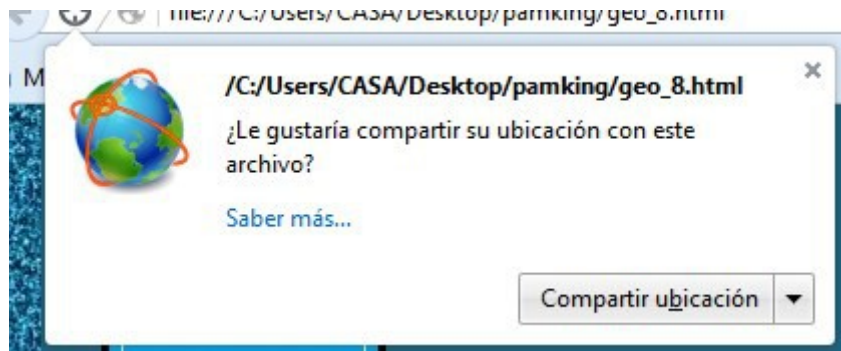


Fig.16: Permiso para ubicar

En cambio en Google Chrome no ha sido posible la configuración. Tras consultar varios foros como <https://developers.google.com/web/updates/2016/04/geolocation-on-secure-contexts-only?hl=es> se averigua que es debido a que la última actualización (la versión 50), no soporta la API HTML5 de Geolocalización en páginas no seguras (que no utilicen conexiones cifradas HTTPS), por medidas de seguridad.

4.3.6 Formulario de contacto

A screenshot of a web page with a dark blue background. At the top left is a logo with the letters 'P' and 'K'. To its right are navigation buttons: "Inicio", "Tipos", "Aparcamientos", "Estacionamiento Regulado", and "Ayuda". The main content is a white-bordered box titled "CONTACTO" in orange. It contains four input fields: "Nombre", "Correo", "Teléfono", and a larger text area with the placeholder "Escriba aquí su mensaje". Below the text area is a blue button labeled "ENVIAR". At the bottom of the page, it says "PamKing - Formulario de contacto".

Fig. 16: Formulario de contacto

Otra de las novedades que ofrece HTML5 es la posibilidad de realizar formularios. Hasta ahora se venían haciendo directamente en PHP por ejemplo, ahora ya se pueden hacer directamente en el código HTML5.

```
<section id="contenido">
  <article id="descripcion-tipos">
    <form action="enviar.php" method="post">
      <h2>CONTACTO</h2>
      <br>
      <input type="text" name="nombre" placeholder="Nombre" required>
      <input type="text" name="correo" placeholder="Correo" required>
      <input type="text" name="telefono" placeholder="Teléfono" required>
      <textarea name="Mensaje" placeholder="Escriba aquí su mensaje" required></textarea>
      <input type="submit" value="ENVIAR" id="boton">
    </form>
    <br>
  </article>
</section>

<footer>PamKing - Formulario de contacto
</footer>
```

Fig.17: Ejemplo de nuestro código para el formulario de contacto

Pese a poder escribir dicho código en HTML, para poder enviar el formulario se recurre a PHP:

```
1 <?php
2 $destino = "aladha@hotmail.com";
3 $nombre = $_POST["nombre"];
4 $correo = $_POST["correo"];
5 $telefono = $_POST["telefono"];
6 $mensaje = $_POST["mensaje"];
7 $contenido = "Nombre: " . $nombre . "\nCorreo: " . $correo . "\nTeléfono: " . $telefono . "\nMensaje: " . $mensaje;
8 mail($destino,"Contacto", $contenido);
9 header("Location:gracias.html");
10 ?>
```

Fig.18: Archivo “enviar.php”

4.3.7 jQuery Mobile

Una vez ya finalizada nuestra página web se pasa a introducir jQuery Mobile, para conseguir una aplicación más sólida, más sobria, con el formato que le otorga dicho marco de trabajo.

Lo primero que tenemos que tener en cuenta para instalar este framework, es descargar jQuery en su página oficial:

http://docs.jquery.com/Downloading_jQuery#Current_Release

Crear una subcarpeta llamada “js” dentro de tu proyecto PhoneGap y guardar allí el archivo de jQuery. Una vez descargado jQuery, debemos descargar el plugin de jQuery Mobile desde su sitio web oficial: <http://jquerymobile.com/download/>

Crear una carpeta con el nombre “jQueryMobile”, descomprimir el archivo .zip que acabamos de descargar y copiar allí todos los archivos que contiene. Al final deberíamos tener una estructura similar a esta:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Banking</title>
5 <meta name="viewport" content="width=device-width, initial-scale=1">
6 <link rel="stylesheet" href="http://code.jquery.com/mobile/1.2.0/jquery.mobile-1.2.0.min.css" />
7 <script src="http://code.jquery.com/jquery-1.8.2.min.js"></script>
8 <script src="http://code.jquery.com/mobile/1.2.0/jquery.mobile-1.2.0.min.js"></script>
9 </head>
10 <body>
11
12 <div data-role="page">
13
14 <div data-role="header">
```

Fig.19: Archivo con jQuery Mobile

4.3.8 PhoneGap

Por último ya sólo queda utilizar PhoneGap para conseguir nuestra aplicación.

PhoneGap Build es un poderoso compilador en la nube, y es mucho más sencillo de utilizar que PhoneGap, porque no precisa descargarse nada adicional. Solo se debe crear una cuenta, donde se puede escoger entre cuatro tipos diferentes. Se debe crear un fichero .zip con toda nuestra aplicación e incluir en ella un archivo *config.xml* con la siguiente información, para que PhoneGap la pueda emplear, como el autor, la descripción de la aplicación, habilitar funcionalidades del dispositivo, la imagen de la descarga, etc

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <widget xmlns = "http://www.w3.org/ns/widgets"
3       xmlns:gap = "http://phonegap.com/ns/1.0"
4       id = "com.phonegap.example"
5       version = "1.0.0">
6
7 <name>Pamking</name>
8 <icon src = "img/pk.jpg" gap:platform="android" gap:density="ldpi" />
9 <gap:splash src="img/splash.png" gap:platform="android" gap:density="ldpi" />
10 <gap:splash src="img/splash.png" gap:platform="android" gap:density="mdpi" />
11 <gap:splash src="img/splash.png" gap:platform="android" gap:density="hdpi" />
12 <gap:splash src="img/splash.png" gap:platform="android" gap:density="xhdpi" />
13
14 <description> Aparcamientos Pamplona</description>
15 <preference name="permissions" value="none"/>
16
17 <author email="aledha@hotmail.com" href="http://phonegap.com"> Adela Santamaria </author>
18
19 <content src="index.html"/>
20
21 <access origin="*" />
22
23 <platform name="android">
```

Fig.20: Ejemplo de archivo *config.xml*

También es importante reseñar que se tiene que incluir un archivo principal llamado *index.html*.

Las aplicaciones creadas por este método se pueden compartir, dando una página de acceso público, donde se pone a disposición cada uno de los instaladores compilados de la aplicación, además de un código QR, para facilitar el acceso a los enlaces.

Una de las ventajas de este servicio es que es gratuito, pero la contra es que esto es así solo para una única aplicación, si quisiéramos tener más se tendría que pagar por ello.

La versión gratuita de PhoneGap limita el tamaño de las aplicaciones a 50 MB, lo que ha obligado a revisar y optimizar el tamaño de los ficheros utilizados en el proyecto, principalmente imágenes y vídeo.

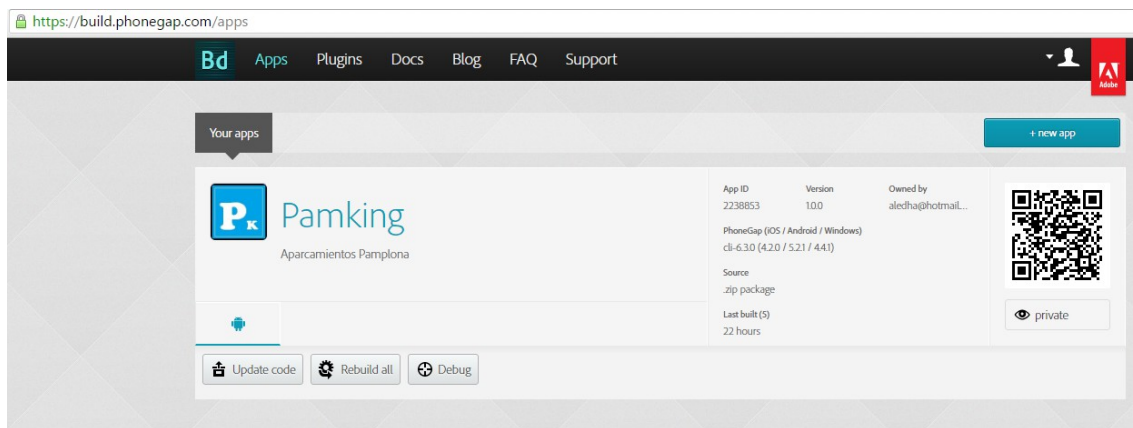


Fig.21: Resultado final tras compilarla en PhoneGap Build

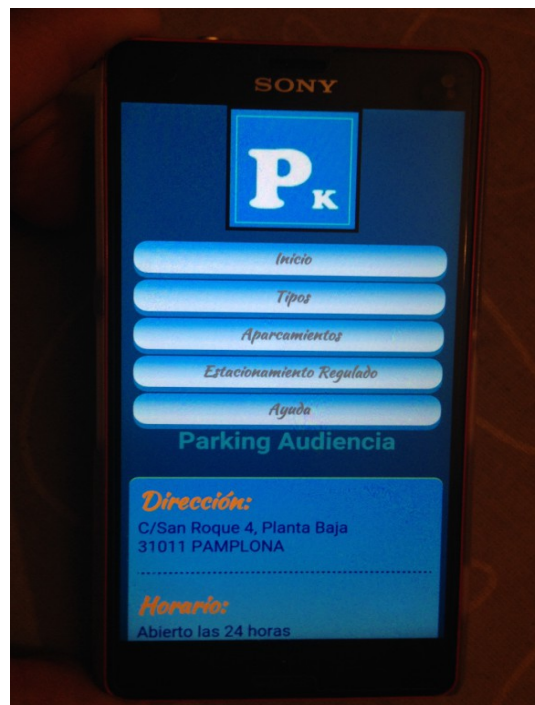


Fig.22: aplicación ya en mi móvil

5. Líneas Futuras

5. Lineas Futuras:

A continuación se exponen algunas de las posibles mejoras que se podrían realizar en un futuro:

- Desarrollarla también para iOS.
- Utilizar PhoneGap u otro framework para conseguir una aplicación híbrida con mejor rendimiento y más características nativas (que pueda ocupar más de 50 MB y tener acceso a la cámara por ejemplo).
- Mejorar la geolocalización, ofreciendo por ejemplo la posibilidad de escoger el parking más cercano, todos en un mismo mapa, en lugar de individualizados.
- Publicar la aplicación en la Play Store.
- Añadir una sección que calcule el importe total de la estancia en un Parking o Zona restringida.
- Crear una galería con fotos propias en lugar de usar algunas encontradas a través de Google.

6. Conclusiones

6. Conclusiones:

A través de la utilización de HTML5, se han conseguido los objetivos planteados y desarrollar una aplicación híbrida que pueda ser instalada en un dispositivo móvil y compatible con todos los navegadores.

Combinando pues HTML5, CSS3, JavaScript y jQueryMobile podemos conseguir aplicaciones web muy funcionales y dinámicas.

jQuery Mobile aporta funcionalidades, reduciendo considerablemente el trabajo y el código, de manera que se obtiene una aplicación atractiva y sencilla.

El uso de PhoneGap Build nos ha dado la oportunidad de acercarnos al mundo de la aplicación nativa sin tener que aprender los lenguajes específicos de cada plataforma.

7. Bibliografía

7. Bibliografía:

1. HTML5 y CSS3. Alexis Goldstein. Anaya Multimedia 2011
2. <http://www.w3schools.com/>
3. <http://www.desarrolloweb.com/>
4. <http://jquerymobile.com/> --> Página oficial de JQuery Mobile
5. <http://jquery.com/>
6. <https://build.phonegap.com/> --> Página oficial de PhoneGap Build
7. <http://www.phonegapspain.com/>
8. MANUALES HTML5, PHP Y JAVASCRIPT --> Varios autores

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN



INGENIERO TÉCNICO DE TELECOMUNICACIÓN,
ESPECIALIDAD EN SONIDO E IMAGEN



DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL HÍBRIDA EMPLEANDO EL FRAMEWORK DE JQUERY MOBILE Y PHONEGAP



Adela Santamaría Gimeno
Tutor: Marko Galarza
Pamplona, 9 Septiembre 2016

Índice

1. INTRODUCCIÓN
2. OBJETIVO
3. TECNOLOGÍAS EMPLEADAS
4. DESARROLLO
5. CONCLUSIONES
6. LÍNEAS FUTURAS



INTRODUCCIÓN

	2011	2012	2013	2014
 ANDROID	52,8%	70,4%	78,2%	76,6%
 iOS	23,0%	20,9%	17,5%	19,7%
 Windows Phone	1,5%	2,6%	3,0%	2,8%
 BlackBerry	8,1%	3,2%	0,6%	0,4%
Otros	14,6%	2,9%	0,8%	0,5%



TIPOS DE APLICACIONES

- APLICACIONES WEB
- APLICACIONES NATIVAS
- APLICACIONES HÍBRIDAS



APLICACIONES WEB

- Sencillez y facilidad de programar
- Usada en cualquier dispositivo
- No necesita de instalación
- Siempre se dispone la versión más actual
- Precisan de conexión a internet
- No aparecen en tiendas de aplicaciones



APLICACIONES NATIVAS

- Desarrollo específico para un S.O.
- Acceso a todas las características del dispositivo
- No precisan conexión a internet
- Descarga a través de tiendas de apps
- Costos y tiempos de desarrollo elevados



APLICACIONES HÍBRIDAS

- Combinación de ambos tipos anteriores
- Ventajas de las dos, pero también los contras de ambas



OBJETIVOS

- Aplicación
- Tema
- Tecnologías
- Diseño



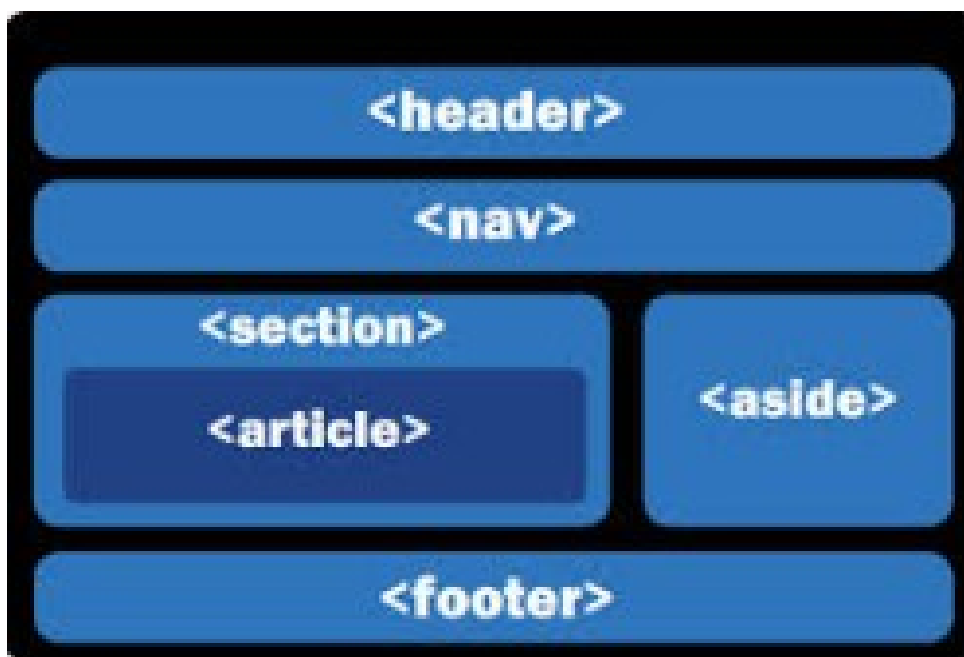
TECNOLOGÍAS EMPLEADAS

- HTML5
- CSS3
- JAVASCRIPT
- PHP
- JQUERYMOBILE
- PHONEGAP
- XAMPP
- GOOGLE MAPS



HTML 5

HTML



CSS 3

CSS



Prefijo	Familia de navegadores a los que aplica	
-webkit-	Chrome, Safari, Android, iOS	
-moz-	Firefox	
-o-	Opera	
-ms-	Microsoft Internet Explorer	

JAVASCRIPT

JS



- Lenguaje de programación orientado a objetos
- Principal objetivo: dotar de dinamismo
- Interactúa con el código HTML5 y puede incluirse dentro de él, debidamente etiquetado



PHP



- Es un lenguaje de código abierto
- Permite la conexión con servidores de BBDD
- Es compatible con todos los navegadores



JQUERY MOBILE

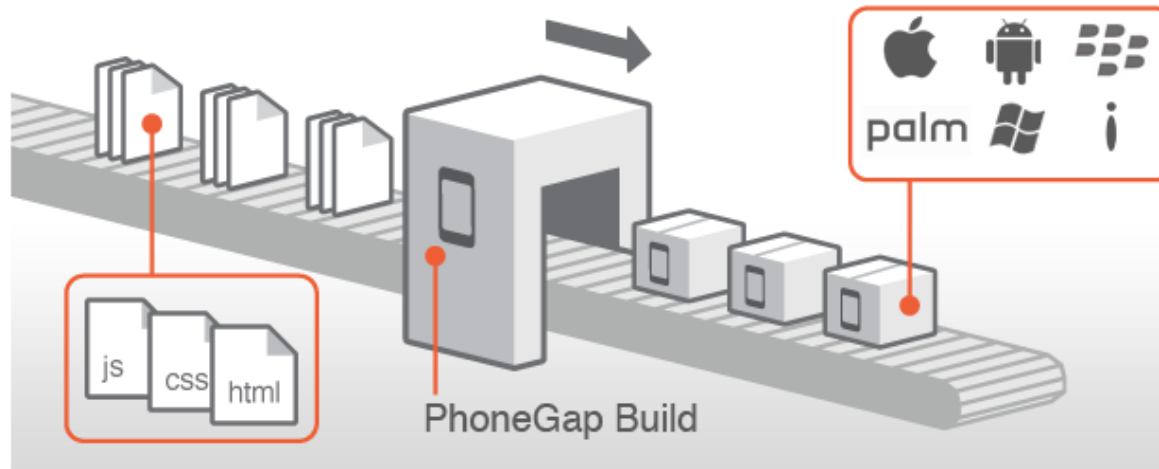


- Es un framework JavaScript para el desarrollo de sitios webs para móviles

- Temas personalizados
- Tamaño reducido
- Facilidad de uso
- Múltiples plataformas
- Soporta las nuevas etiquetas HTML 5



PHONEGAP



XAMPP

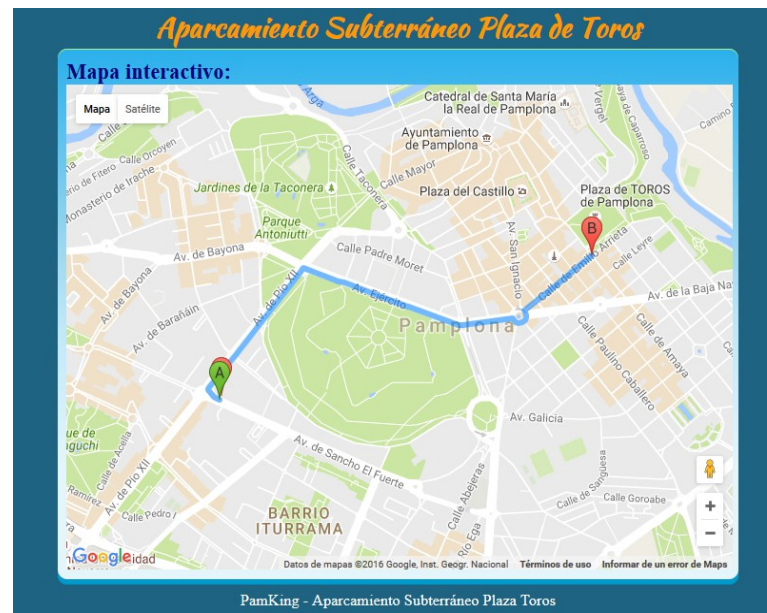


XAMPP

- Servidor independiente
- Opción de crear un servidor local en nuestro ordenador
- Permite poder probar nuestra aplicación antes de subirla a un servidor web.



GOOGLE MAPS

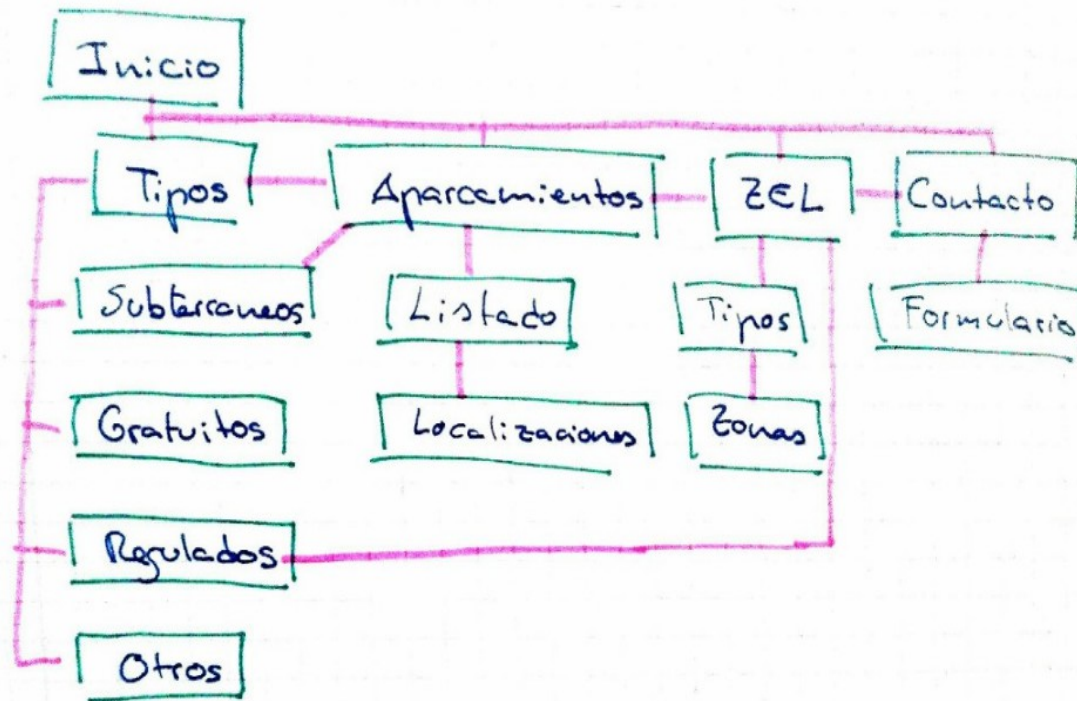


DESARROLLO

- ESQUEMA
- PROGRAMAS
- ESTRUCTURA



ESQUEMA



PROGRAMAS

- DreamWeaver
- Notepad++
- Paint
- conversores vídeo
- Filezilla



ESTRUCTURA

Paso a paso:

- HTML5
- CSS3
- RESPONSIVE WEB DESIGN
- VIDEO
- GEOLOCALIZACION
- FORMULARIO CONTACTO
- JQUERY MOBILE
- PHONEGAP



HTML5

```
index.html x estilos.css x tipos.html x lista_sub.html x lista_zon.html x contacto.html x enviar.php x gracias.html x bici.html x gratis.html x geo_10.html x
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <meta charset="utf-8"/>
5     <meta name="description" content="Futura aplicación móvil"/>
6     <meta name="viewport" content="width=device-width,initial-scale=1"/>
7
8     <link rel="stylesheet" href="css/estilos.css"/>
9     <link rel="stylesheet" href="css/sliderman.css"/>
10
11
12     <script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1/jquery.min.js"> </script>
13     <!-- <script type="text/javascript" src="js/acordeon.js"></script> -->
14     <script type="text/javascript" src="js/sliderman.1.3.7.js"></script>
15
16     <link rel="shortcut icon" type="image/x-icon" href="img/vintagecar.ico"/>
17     <link rel="author" type="text/plain" href="humans.txt"/>
18     <link rel="sitemap" type="application/xml" title="Sitemap" href="sitemap.xml"/>
19     <title>PamKing</title>
20   </head>
21   <body>
22     <header>
23       <h1>
24         <a href="index.html">
25           
26         </a>
27       </h1>
28       <nav>
29         <ul>
30           <li><a href="index.html" class="botoncillo">Inicio</a></li>
31           <li><a href="tipos.html" class="botoncillo">Tipos</a></li>
32           <li><a href="lista_sub.html" class="botoncillo">Aparcamientos</a></li>
33           <li><a href="lista_zon.html" class="botoncillo">Estacionamiento Regulado</a></li>
34           <li><a href="contacto.html" class="botoncillo">Ayuda</a></li>
35         </ul>
36       </nav>
37     </header>
38
```

CSS 3

Inicio

Tipos

Aparcamientos

Estacionamiento Regulado

Ayuda



```
155 nav a.botoncillo{
156     background: navy;/*kitar lue*/
157     background-image: linear-gradient(to bottom, rgba(42,176,237,1) 0%,
158     rgba(159,216,239,1) 38%,rgba(191,232,249,1) 52%,rgba(228,245,252,1) 78%);
159     border-radius: 15px;
160     box-shadow: inset 0px 1px 0px #49B7F2, 0px 6px 0px #09c;
161     color: grey;
162     display:block;
163     font-family: "kaushan_scriptregular", Helvetica, Verdana;
164     font-size:20px bold;
165     padding:5px 10px ;
166     position: relative;
167     text-decoration:none;
168     transition:all 0.5s ease-in;
169 }
170
171 nav a.botoncillo:hover {
172     /*border: 0.1em dashed navy;*/
173     background-image: linear-gradient(to bottom, rgba(228,245,252,1)
174     22%,rgba(191,232,249,1) 48%,rgba(159,216,239,1) 62%,rgba(42,176,237,1) 100%);
175     color:navy;
176 }
177
178 nav a.botoncillo:active {
179     /*background: /*#6fba26;*/
180     background-image: linear-gradient(to bottom, rgba(228,245,252,1)
181     22%,rgba(191,232,249,1) 48%,rgba(159,216,239,1) 62%,rgba(42,176,237,1) 100%);
```


RESPONSIVE WEB DESIGN



```
404 @media screen and (max-width:600px){
405     form{
406         width:70%;
407     }
408
409     div#mapa{
410         width: 300px;
411         height:200px;
412     }
413
414     nav li{
415         display: inline-block;
416         margin: 2px;
417         width: 98%;
418     }
419 }
```

```
<meta name="viewport"
content="width=device-width,
initial-scale=1.0" />
```

VIDEO

8:18



50
51
52
53
54
55
56
57
58
59
60
61

```
<article id="multimedia-tipos">
  <h3>
    Lo complicado de aparcar en Pamplona.
  </h3>
  <video id="video-tipos" controls preload >
    <source src="vdo/APARCAR.mp4" type="video/mp4" />
    <source src="vdo/APARCAR.webm" type="video/webm" />
    <source src="vdo/APARCAR.ogv" type="video/ogg" />
  </video>
</article>
```



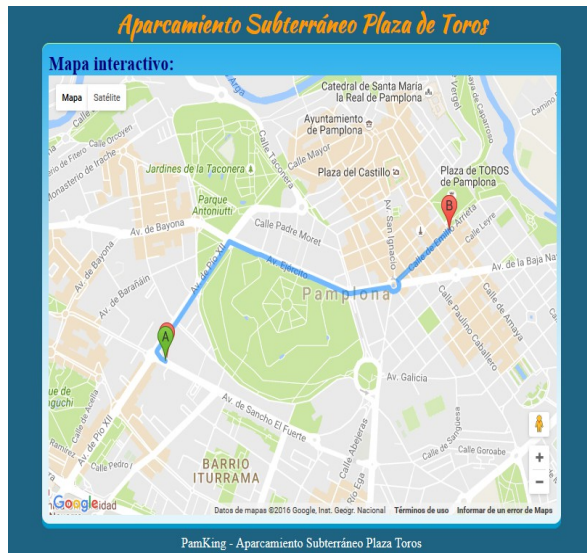
[Inicio](#) [Tipos](#) [Aparcamientos](#) [Estacionamiento Regulado](#) [Ayuda](#)

Lo complicado de aparcar en Pamplona.



0:04 / 3:10

GEOLOCALIZACIÓN



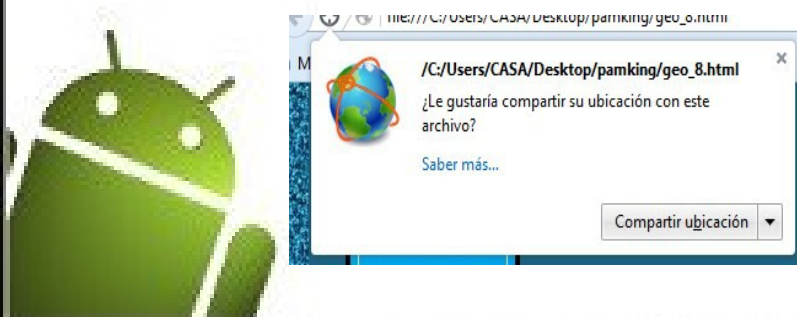
```
<!--Mapa interactivo-->
<div id="mapa" style="width: 700px; height: 500px;">
  --Aquí va nuestro mapa--
</div>
<script type="text/javascript">
  //NAMESPACE --> google.maps.ALGO
  var divMapa = document.getElementById('mapa');
  navigator.geolocation.getCurrentPosition( fn_ok, fn_mal );
  function fn_mal(){}
  function fn_ok( rta ){
    var lat = rta.coords.latitude;
    var lon = rta.coords.longitude;

    var gLatLon = new google.maps.LatLng( lat, lon );
    var objConfig = {
      zoom: 17,
      center: gLatLon
    }
    var gMapa = new google.maps.Map( divMapa, objConfig );
    var objConfigMarker = {
      position: gLatLon,
      map: gMapa,
      title: "ud está aquí"
    }
    var gMarker = new google.maps.Marker( objConfigMarker );

    var gCoder = new google.maps.Geocoder( );
    var objInformacion = {
      address: ' Rincón de la Aduana s/n, 31001 Pamplona, Navarra'
    }
    gCoder.geocode( objInformacion, fn_coder );

    function fn_coder( datos ){
      var coordenadas = datos[0].geometry.location; //obj LatLng

```



FORMULARIO CONTACTO

```
<form action="enviar.php" method="post">
  <h2>CONTACTO</h2>
  <br>
  <input type="text" name="nombre" placeholder="Nombre" required>
  <input type="text" name="correo" placeholder="Correo" required>
  <input type="text" name="telefono" placeholder="Teléfono" required>
  <textarea name="Mensaje" placeholder="Escriba aquí su mensaje" required></textarea>
  <input type="submit" value="ENVIAR" id="boton">
</form>
```

```
<?php
  $destino = "aledha@hotmail.com";
  $nombre = $_POST["nombre"];
  $correo = $_POST["correo"];
  $telefono = $_POST["telefono"];
  $mensaje = $_POST["mensaje"];
  $contenido = "Nombre:" . $nombre . "\nCorreo: " .
  mail($destino, "Contacto", $contenido);
  header("Location: gracias.html");
?>
```

CONTACTO

Nombre

Correo

Teléfono

Escriba aquí su mensaje

ENVIAR

PamKing - Formulario de contacto

JQUERY MOBILE

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Parking</title>
5 <meta name="viewport" content="width=device-width, initial-scale=1">
6 <link rel="stylesheet" href="http://code.jquery.com/mobile/1.2.0/jquery.mobile-1.2.0.min.css" />
7 <script src="http://code.jquery.com/jquery-1.8.2.min.js"></script>
8 <script src="http://code.jquery.com/mobile/1.2.0/jquery.mobile-1.2.0.min.js"></script>
9 </head>
10 <body>
11
12 <div data-role="page">
13
14 <div data-role="header">
```



PHONEGAP








```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <widget xmlns = "http://www.w3.org/ns/widgets"
3       xmlns:gap = "http://phonegap.com/ns/1.0"
4       id = "com.phonegap.example"
5       version = "1.0.0">
6
7 <name>Pamking</name>
8 <icon src = "img/pk_.jpg" gap:platform="android" gap:density="ldpi" />
9 <gap:splash src="img/splash.png" gap:platform="android" gap:density="ldpi" />
10 <gap:splash src="img/splash.png" gap:platform="android" gap:density="mdpi" />
11 <gap:splash src="img/splash.png" gap:platform="android" gap:density="hdpi" />
12 <gap:splash src="img/splash.png" gap:platform="android" gap:density="xhdpi" />
13
14 <description> Aparcamientos Pamplona</description>
15 <preference name="permissions" value="none" />
16
17 <author email="aledha@hotmail.com" href="http://phonegap.com"> Adela Santamaria </author>
18
19 <content src="index.html" />
20
21 <access origin="*" />
22
23 <platform name="android">
```

phonegap.com/apps

phonegap.com/apps

Bd Apps Plugins Docs Blog FAQ Support

Your apps + new app

 Pamking Aparcamientos Pamplona	App ID 2238853	Version 1.0.0	Owned by aledha@hotmail...	
	PhoneGap (iOS / Android / Windows) cli-6.3.0 (4.2.0 / 5.21 / 4.41)	Source .zip package	Last built (5) 22 hours	
 Update code	 Rebuild all	 Debug		



LINEAS FUTURAS

- DESARROLLO PARA iOS
- USO DE OTROS FRAMEWORKS PARA MEJORAR RENDIMIENTO
- MEJORAR GEOLOCALIZACIÓN
- PLAY STORE
- AÑADIR SECCIONES
- GALERÍA PROPIA



CONCLUSIONES

- OBJETIVOS CONSEGUIDOS
- WEB FUNCIONAL Y DINÁMICA
- JQUERY MOBILE FUNCIONALIDADES
- PHONEGAP BUILD ACERCAMIENTO



AGRADECIMIENTOS

GRACIAS

- TUTOR: MARKO GALARZA

- FAMILIA Y AMIGOS

- UNIVERSIDAD PÚBLICA DE NAVARRA

