**UAH**

16

**Desarrollo de aplicaciones multiplataforma**

**TG1**

Contenido

[1. Autores del trabajo, planificación y entrega 3](#_Toc444537686)

[1.1 Autores 3](#_Toc444537687)

[1.2 Planificación 3](#_Toc444537688)

[1.3 Entrega 3](#_Toc444537689)

[2. Descripción del tipo de tecnología 3](#_Toc444537690)

[3. Fuentes de información (documentos) 3](#_Toc444537691)

[3.1 Fuentes sobre el tipo de tecnología en general 4](#_Toc444537692)

[3.1.1 Fuente de información 1 sobre el tipo de tecnología en general 4](#_Toc444537693)

[3.1.2 Fuente de información 2 sobre el tipo de tecnología en general 4](#_Toc444537694)

[3.1.n Fuente de información n sobre el tipo de tecnología en general 4](#_Toc444537695)

[3.2 Fuentes sobre la tecnología específica A 4](#_Toc444537696)

[3.2.1 Fuente de información 1 sobre la tecnología específica A 4](#_Toc444537697)

[3.2.2 Fuente de información 2 sobre la tecnología específica A 4](#_Toc444537698)

[3.2.n Fuente de información n sobre la tecnología específica A 4](#_Toc444537699)

[3.3 Fuentes sobre la tecnología específica B 4](#_Toc444537700)

[3.3.1 Fuente de información 1 sobre la tecnología específica B 4](#_Toc444537701)

[3.3.2 Fuente de información 2 sobre la tecnología específica B 4](#_Toc444537702)

[3.3.n Fuente de información n sobre la tecnología específica B 4](#_Toc444537703)

[4. Fuentes de información (cursos no gratuitos) 4](#_Toc444537704)

[4.1 Cursos no gratuitos sobre el tipo de tecnología en general 4](#_Toc444537705)

[4.1.1 Curso no gratuito 1 sobre el tipo de tecnología en general 4](#_Toc444537706)

[4.1.2 Curso no gratuito 2 sobre el tipo de tecnología en general 4](#_Toc444537707)

[4.1.n Curso no gratuito n sobre el tipo de tecnología en general 4](#_Toc444537708)

[4.2 Cursos no gratuitos sobre la tecnología específica A 4](#_Toc444537709)

[4.2.1 Curso no gratuito 1 sobre la tecnología específica A 4](#_Toc444537710)

[4.2.2 Curso no gratuito 2 sobre la tecnología específica A 4](#_Toc444537711)

[4.2.n Curso no gratuito n sobre la tecnología específica A 4](#_Toc444537712)

[4.3 Cursos no gratuitos sobre la tecnología específica B 5](#_Toc444537713)

[4.3.1 Curso no gratuito 1 sobre la tecnología específica B 5](#_Toc444537714)

[4.3.2 Curso no gratuito 2 sobre la tecnología específica B 5](#_Toc444537715)

[4.3.n Curso no gratuito n sobre la tecnología específica B 5](#_Toc444537716)

[5. Fuentes de información (cursos gratuitos) 5](#_Toc444537717)

[5.1 Cursos gratuitos sobre el tipo de tecnología en general 5](#_Toc444537718)

[5.1.1 Curso gratuito 1 sobre el tipo de tecnología en general 5](#_Toc444537719)

[5.1.2 Curso gratuito 2 sobre el tipo de tecnología en general 5](#_Toc444537720)

[5.1.n Curso gratuito n sobre el tipo de tecnología en general 5](#_Toc444537721)

[5.2 Cursos gratuitos sobre la tecnología específica A 5](#_Toc444537722)

[5.2.1 Curso gratuito 1 sobre la tecnología específica A 5](#_Toc444537723)

[5.2.2 Curso gratuito 2 sobre la tecnología específica A 5](#_Toc444537724)

[5.2.n Curso gratuito n sobre la tecnología especifica A 5](#_Toc444537725)

[5.3 Cursos gratuitos sobre la tecnología específica B 5](#_Toc444537726)

[5.3.1 Curso gratuito 1 sobre la tecnología específica B 5](#_Toc444537727)

[5.3.2 Curso gratuito 2 sobre la tecnología específica B 5](#_Toc444537728)

[5.3.n Curso gratuito n sobre la tecnología especifica B 5](#_Toc444537729)

[6. Ayudas para estudiar las tecnologías 5](#_Toc444537730)

[7. Recursos para implementar las tecnologías 6](#_Toc444537731)

[7.1 Recursos para implementar la tecnología A 6](#_Toc444537732)

[7.1.1 Recursos no gratuitos para implementar la tecnología A 6](#_Toc444537733)

[7.1.1 Recursos no gratuitos para implementar la tecnología A 6](#_Toc444537734)

[7.2 Recursos para implementar la tecnología B 6](#_Toc444537735)

[7.2.1 Recursos no gratuitos para implementar la tecnología B 6](#_Toc444537736)

[7.2.1 Recursos no gratuitos para implementar la tecnología B 6](#_Toc444537737)

[8. Conclusiones 6](#_Toc444537738)

# Autores del trabajo, planificación y entrega

## Autores

El grupo que desarrolla el trabajo que se presenta a continuación el es grupo 7 de laboratorio del turno de tarde; este grupo está formado por:

* Miguel Ángel Rodríguez Blanco (Coordinador del grupo).
* Luis Nueda García.
* Álvaro Laza Martín.
* Eduardo Dorado Pérez.

## Planificación

La planificación de proyecto ha sido realizada por el Coordinador del grupo utilizando la herramienta de diagramación GanttPro, tras una reunión con el grupo de trabajo en la que se han expuesto las tareas necesarias para la consecución del objetivo final y se ha realizado un reparto/asignación de tareas equitativo.

Este trabajo de planificación esta disponible en la URL:

[***https://app.ganttpro.com/shared/token/efc3eaab91ed6391cc072e2f6774710be06e949e7d796a915e91287d2e1aefce#!/app/home***](https://app.ganttpro.com/shared/token/efc3eaab91ed6391cc072e2f6774710be06e949e7d796a915e91287d2e1aefce#%21/app/home)

## Entrega

El repositorio creado para la puesta en común de los avances del grupo y en el que se ha guardado el resultado de este trabajo es:

[***https://github.com/MiguelRodriguezBlanco/TG1***](https://github.com/MiguelRodriguezBlanco/TG1)

En este repositorio se encuentra, además de los archivos subidos a los largo del desarrollo del trabajo los archivos requeridos en la rama máster:

* Trabajo terminado: del trabajo terminado con el nombre TG1\_final.ocx
* Presentación del trabajo: TG1\_final.pptx

# Desarrollo de aplicaciones multiplataforma

Cada vez son más necesarias las aplicaciones móviles tanto en el mundo empresarial como en el sector del entretenimiento. Sin embargo, el desarrollo de estas aplicaciones se ve condicionado por la segmentación de los dispositivos que obliga a desarrollar la misma aplicación varias veces en diferentes plataformas para cubrir una amplia cuota de mercado. Para ello se hace imprescindible:

* Conocer o aprender el lenguaje de cada plataforma.
* IDS de desarrollo diferentes.
* Utilizar las API’S NATVAS.
* Escribir la app más de una vez.
* Invertir tiempo en el desarrollo de cada una de ellas.

Todo esto dispara los costes de producción llegando incluso a no ser rentables, obligando a desarrolladores a definirse por una plataforma perdiendo cuota de potenciales usuarios con el fin de reducir costes.

La solución a este problema está en los “frameworks de desarrollo de aplicaciones móviles multiplataforma”. Con estas herramientas normalmente se desarrolla la aplicación en un lenguaje de programación diferente al que se utilizaría nativamente en las diferentes plataformas y con este código en lo que podríamos llamar un “lenguaje neutro” se llevan a cabo compilaciones específicas sobre cada una de las plataformas móviles. De esta forma, se desarrolla el código una única vez y puede distribuirse sin costes adicionales de programación en otras plataformas móviles.

Una de las principales ventajas de los Smartphones radica que el navegador HTML5 es común a todos, lo que permite crear una aplicación web que puede utilizarse desde el navegador de los móviles.

No obstante, el uso de frameworks multiplataforma de ninguna manera significa que todo quede unificado. Lógicamente siempre habrá que realizar pequeños cambios en el código desarrollado en cada uno de los frameworks para adaptarlo a las diferentes plataformas correctamente, pero obviamente, eso pequeños cambios siempre suponen menor esfuerzo y por consiguiente menores costes que el desarrollo de toda una aplicación.

El precio no es exactamente 1 a 3, realmente sería del orden de 1.5 a 3 o 2 a 3, dependiendo de la complejidad de la interfaz de usuario lo que supone un ahorro considerable. Pero más allá del simple precio, está el ahorro en tiempo. En Nativo, podemos tardar un mes en desarrollar para cada plataforma, con lo que al final tardaríamos 3 meses en total. Con un framework multiplataforma finalizaríamos el trabajo en 1 mes y medio, lo que nos permite disponer de tiempo extra para atender el lanzamiento, recoger datos y preparar nuevas versiones, con lo que logramos ser más agiles en responder a nuestros usuarios y corregir bugs, que además corregimos una vez para las tres plataformas, al tener el código compartido

Lógicamente, estas plataformas intermedias de desarrollo no están pensadas para desarrollar aplicaciones punteras que explotan al 100% el potencial de un dispositivo. Estos frameworks están orientados a desarrollar aplicaciones con unos requerimientos técnicos normales, es decir, aplicaciones que no necesitan de implementar sistemas complejos de cómputo, sin necesidad de trabajar en OpenGL con aplicaciones numerosos gráficos y entornos 3D… Básicamente estos frameworks, están pensados para desarrollar juegos 2D, o aplicaciones de carácter empresarial, guías turísticas…

Este enfoque, aunque ofrece grandes ventajas, también presenta algunos inconvenientes:

* El código debe ser interpretado por un navegador web: html y javascript; este proceso de interpretación supone una pérdida de rendimiento frente a una aplicación nativa ya compilada.
* Existe una dependencia de un agente externo no controlado: el navegador web. Dependiendo de la plataforma y su versión, el HTML será interpretado de una forma u otra, por lo que no es posible garantizar al 100% el resultado final.
* Se pierden las peculiaridades de cada plataforma; al seguir el enfoque de “lo mismo para todos”, se crea una aplicación que no se adapta a ninguna plataforma, y por tanto, se diferencia de las aplicaciones específicas de la plataforma lo que hace que resulte extraña para un usuario habituado a un tipo de uso, lo que provoca reticencias por su parte que podrían provocar que el usuario no la utilice o busque un sustituto más similar a lo que está habituado.

En cuanto a la actualización de nuestras aplicaciones, l disponer de proyecto común con lógica y código compartido y un proyecto por cada plataforma: iOS, Android y Windows Phone, es posible actualizar cualquier plataforma por separado del resto sin ningún problema. Las aplicaciones son totalmente nativas, así que las actualizaciones de los sistemas operativos afectarán de la misma forma que a una aplicación escrita de forma nativa. No existe diferencia en este aspecto.

# Fuentes de información (documentos)

## Desarrollo de aplicaciones multiplataforma

Con el fin de adquirir conocimientos acerca del desarrollo de aplicaciones multiplataformas se navegado donde se han localizado los siguientes artículos que han sido de gran utilidad:

### <http://apprende.es/el-desarrollo-multiplataforma-en-los-dispositivos-moviles/>

<http://necsia.es/frameworks-multiplataforma-para-el-desarrollo-movil/>

<http://apprende.es/el-desarrollo-multiplataforma-en-los-dispositivos-moviles/>

<http://es.slideshare.net/erickstark1/desarrollo-de-apps-multiplataforma>

<http://e-growing.com/desarrollo-de-aplicaciones-moviles-nativo-o-multiplataforma/>

<http://asiermarques.com/2015/desarrollo-aplicaciones-multiplataforma/>

<https://www.witcamp.com/post/2013-05-12-aplicaciones-web-apps-nativas-interpretadas-web-movil-que-hacer>

## Ionic

### Fuente de información 1 sobre la tecnología específica A

### Fuente de información 2 sobre la tecnología específica A

### 3.2.n Fuente de información n sobre la tecnología específica A

## Apache Cordova

### <https://cordova.apache.org/>

Página oficial de Apache Cordova. Esta fuente de información nos da una primera información sobre el framework, además de incluir las plataformas y herramientas que usa. La primera parte que nos ofrece esta fuente es una guía rápida de inicio, en la que nos muestra 5 pasos básicos para empezar a desarrollar con Apache Cordova.

Tras esto no muestra cuales son las plataforma que soporta el desarrollo (Android, iOS, Windows, Ubuntu…)

Y tras esto, una serie de herramientas framework JavaScript con las que podemos hacer uso de su tecnología, seguido de unas app conocidas que han sido desarrolladas con dicha tecnología.

El objetivo de Apache Cordova es el desarrollo de aplicaciones web multiplataforma a través de tecnología web (HTML5, CSS3 y JavaScript), con el objetivo de desarrollar aplicaciones híbridas (nativas/basadas en web)

### <https://cordova.apache.org/docs/en/latest/guide/overview/>

Documentación de Apache Cordova. Ofrece un guía con varios apartados con los que podemos entender, configurar y empezar a desarrollar nuestras aplicaciones.

Esta página contiene un índice por el que podemos ir navegando y conociendo los distintos aspectos y métodos que componen Apache Cordova, desde una primera vista y muestra de información de la arquitectura y las características generales a cómo desarrollar para las distintas plataformas existentes o desarrollar y trabajar con la API de plugins.

### <https://en.wikipedia.org/wiki/Apache_Cordova>

Definición, historia y plataformas soportadas de Apache Corodova.

Apache Cordoba nace, a través de la empresa Nitobi, que crea un framework de desarrollo de aplicaciones móviles. Esta empresa es comprada por Adobe Systems, que decide cambiar el nombre de la aplicación a “PhoneGap”, y tras un tiempo denominarla “Apache Callback” para después lanzar una versión de código abierto de la misma denominada Apache Cordoba.

Como se menciona anteriormente, Apache Cordova permite el desarrollo de aplicaciones para dispositivos móviles a través de las tecnologías CSS3, HTML5 y JavaScript, permitiendo desarrollar para el conjunto de plataformas disponibles a la vez, en vez de tener que crear un aplicación desde cero dependiendo de los sistemas (Android, iOS…)

Existen numerosas versiones y frameworks que utilizan la tecnología de Apache Cordava, una de ellas PhoneGap, aunque también debemos incluir Ionic, Monaca o Intel XDK entre otros.

### <http://www.campusmvp.es/recursos/post/PhoneGap-o-Apache-Cordova-que-diferencia-hay.aspx>

Distinción entre Apache Cordova y PhoneGap.

A la hora de realizar una búsqueda sobre Apache Cordova, nos aparecen numerosas opciones de PhoneGap, y es posible que sin un estudio previo del ecosistema de Cordova, podamos creer que son dos tecnologías distintas, pero suplementarias.

Esta fuente de información nos muestra la historia del nacimiento de Apache Cordova, desde que la empresa Nitobi crea el framework “PhoneGap” para el desarrollo de aplicaciones móviles multiplataforma, hasta que Apache compra la empresa Nitobi y decide lanzar un proyecto de código abierto denominado Apache Cordova.

Como resumen, nos indica que PhoneGap, en la actualidad, es una distribución de las varias que existen en la actualidad de Apache Cordova

### <http://blog.ckgrafico.com/que-es-apache-cordova/>

Explicación sobre Apache Cordova, su comportamiento en los distintos sistemas operativos, y breve explicación sobre PhoneGap.

Las aplicaciones desarrolladas en Apache Cordova se pueden usar en numerosas plataformas (Android, Blackberry, iOS, Ubuntu, Windows Phone…) teniendo en cada una de ellas unas características que como desarrolladores tenemos que tener en cuenta.

Con este framework, a la hora de desarrollar nuestra aplicación, podemos crear un conjunto de hojas de estilo (CSS), que en función del sistema en el que nos encontremos, este muestre una configuración distinta una a otra, en función de las características o expectativas del cliente.

Se menciona que PhoneGap es una distribución de Apache Cordova y que en función de los gustos de cada desarrollador se puede elegir entre una distribución u otra.

Tras esto nos indica una serie de características que se ha encontrado en la implantación de aplicaciones en distintos sistemas operativos.

### <https://voylinux.com/que-es-y-que-no-apache-cordova/>

Distinción entre las posibilidades que nos ofrece Apache Cordova a la hora de desarrollar y el desarrollo de la UI, que tendrá que ser realizado por cada desarrollador.

Lo que conseguimos con Apache Cordova es generar el código con el que la aplicación funciona (hablando en términos de desarrollo web, estamos haciendo referencia al back-end). Estamos usando el navegador web de nuestro dispositivo como motor para ejecutar nuestra aplicación.

Gracias a Apache Cordova, podemos acceder a distintas funcionalidades de nuestro teléfono a través de la APIs que contiene. Finalmente la aplicación se guarda como una app nativa, ya traducida al sistema en el que nos encontremos.

Lo que no hace Apache Cordova es la parte de la interfaz de usuario, que deberá ser configurada por el desarrollador a través de las hojas de estilo (CSS), aunque algunas de las distribuciones que existen para Apache Cordoba cuentan con un elemento de UI, como punto de partida para el diseño de nuestra app.

### <http://es.slideshare.net/RevistaSG/desarrollo-de-apps-mviles-con-apache-cordova>

Conceptos básicos y creación breve de aplicación móvil.

Esta presentación comienza indicándonos que para el año 2020, existirá una media de 4.3 dispositivos móviles por cápita, dándonos a entender que contaremos con numerosos dispositivos con los que realizaremos distintas operaciones. Estos dispositivos no tienen por qué contar con el mismo sistema operativo, por lo que contar con sistemas que permitan desarrollar la misma funcionalidad pero para distintos entornos será necesario.

La línea de comandos e Cordova funciona como una terminal, y tendremos que hacer uso de Node.js para poder desarrollar. Tras esto se muestran varios archivos útiles, pudiendo observar su estructura y el directorio en el que se encuentran (desarrollo de plugins).

# Cursos no gratuitos

## Cursos no gratuitos sobre desarrollo de aplicaciones multiplataforma

## Cursos no gratuitos: Ionic

### Curso no gratuito 1 sobre la tecnología específica A

### Curso no gratuito 2 sobre la tecnología específica A

### 4.2.n Curso no gratuito n sobre la tecnología específica A

## Cursos no gratuitos: Apache Cordova

### <http://www.netmind.es/curso/desarrollo-de-aplicaciones-moviles-con-apachecordova/>

Desarrollo aplicaciones moviles con Apache/Cordova. Comprender las tecnologías HTML5/CSS3 y JavaScript junto con el uso de jQuery, BootStrap.

Debido a la existencia actual de numerosas plataforma para desarrollar (iOS, Android, Windows Phone...) y a que para poder acceder a las funcionalidades del dispositivo (localización, cámara...) es necesario el desarrollo de apliaciones nativas, el uso de Apache Cordova es una virtud, ya que permite usar los recursos del dispositivo debido a las API que contiene, haciendo posible la creación de aplicaciones para diversas plataformas a través de tecnologías Web (HTML5, CSS3 y JavaScript).

El objetivo del curso es comprender y usar las API que contiene Apache Cordova para el desarrollo de aplicaciones a través de las tecnologías Web anteriormente citadas, junto con el uso de jQuery, Bootstrap y lenguajes de programación como C# ó C++

### <http://www.itmaster.com.ar/cursos/curso-phonegap>

Curso de Apache Cordova y PhoneGap. Crear aplicaciones móviles usando HTML5 para iOS, Android, Windows Phone y BlackBerry. Cordova 5.0

Aprender el desarrollo de aplicaciones multiplataformas a través de Apache Cordova y el framework PhoneGap a través de tecnologías Web y que puedan hacer uso de los recursos con los que el dispositivo cuenta (GPS, cámara, micrófono, contactos...)

### <http://atsistemas.com/eventos-atsistemas/desarrollo-de-aplicaciones-moviles-hibridas-con-apache-cordova-3/>

Desarrollo de aplicaciones móviles híbridas con Apache Cordova. Conceptos iniciales de Apache Cordova, comandos, funcionalidades...

Apache Cordova cuenta con un conjunto de librerias JavaScript disponibles desde cualquier dispositivo, independiente de la plataforma en la que se encuentre. Actualmente está disponible para las plataformas iOS, Android, BlackBerry, Windows Phone, Palm WebOS, Bada y Symbian

# Cursos gratuitos

## Cursos gratuitos para el desarrollo de aplicaciones multiplataforma

### Curso gratuito 1 sobre el tipo de tecnología en general

### Curso gratuito 2 sobre el tipo de tecnología en general

### Curso gratuito n sobre el tipo de tecnología en general

## Cursos gratuitos: Ionic

### Curso gratuito 1 sobre la tecnología específica A

### Curso gratuito 2 sobre la tecnología específica A

### Curso gratuito n sobre la tecnología especifica A

## Cursos gratuitos: Apache Cordova

### <https://cordova.apache.org/docs/en/latest/guide/cli/index.html>

Creación de la primera aplicación con Cordova. Curso sobre la creación de una aplicación Cordova y su configuración para distintas plataformas móviles a través de la interface de comandos.

Creación de la aplicación en HTML/JS y como lanzarla en distintas plataformas nativas móviles a través de la interfaz de línea de comandos de Apache Cordova.

Guía de descargar (Node.js) e instalación a través de las distintas plataformas (dependiendo de la que usemos como desarrollador) y como crear, construir y probar nuestra propia apliación

### <https://ccoenraets.github.io/cordova-tutorial/index.html>

Tutorial de Apache Cordova. Contiene varias apartados, desde la creación de un proyecto, a la configuración del directorio de trabajo o la creación de vistas.

El tutorial comienza con los objetivos del curso, que son entro otros crear y construir un proyecto a través de la interfaz de línea de comandos que nos proporciona Apache Cordova, a como diferenciar entre las distintas estrategias de almacenamiento de datos o cómo usar los recursos que dispone nuestro dispositivo (Geolocalización, cámara o contactos entro otros)

De la misma forma nos comenta una serie de requisitos para completar el curso (tener un editor de código, un navegador actualizado y una conexión a internet junto con conocimiento de HTML y JavaScript)

Inicia con la descarga de Node.js

### <https://msdn.microsoft.com/es-es/library/dn771545.aspx>

Introducción a Visual Studio Tools para Apache Cordova. Guía para desarrollar aplicaciones mediante Visual Studio Tools para Apache Cordova. Instalación y configuración de Visual Studio, ejecutar aplicaciones en diferentes plataformas o incluso hasta la configuración de seguridad de Cordova 5.

Guía completa sobre el uso y configuración de aplicaciones basadas en Apache Cordova desarrolladas a través de Visual Studio. Gracias a Visual Studio se pueden desarrollar aplicaciones en las siguiente plataformas:

* Androir 2.2.3 y posterior
* iOS 6,7 y 8
* Windows 8 y 8.1
* Windows Phone 8 y 8.1

Tras mencionar las distintas ventajas que supone usar esta herramienta para el uso de aplicaciones basadas en Apache Cordova, nos ofrece varios ejemplos ubicados en directorios de GitHub.

Tras esto aparece una guía a modo de tutorial que nos indica paso a paso como descargar e instalar Visual Studio, configurarlo y comenzar a desarrollar nuestra aplicación

### <https://netbeans.org/kb/docs/webclient/cordova-gettingstarted.html>

Creación de aplicación usando el template de Apache Cordova a través de NetBeans. Instalación de Cordova con el IDE NetBeans, Creación de proyecto y debug.

Este tutorial comienza con la instalación de Apache Cordova y configuración posterior, para que podemos crear una Aplicación Cordova a través de NetBeans. Debemos de añadir los archivos JavaScript jquery y jquey-mobile para su correcto desarrollo.

Tras esto se creará el proyecto y podremos añadir simuladores de los distintos entornos para comenzar a desarrollar y probar nuestra aplicación, modificando los archivos HTML o JavaScript para la configuración que creamos oportuna.

### <http://weblog.west-wind.com/posts/2015/Jan/06/Using-Cordova-and-Visual-Studio-to-build-iOS-Mobile-Apps>

Uso de Visual Studio y Cordova para la creación de aplicaciones móviles en iOS.

Creación de aplicación para iOS a través de la herramienta Visual Studio. En este tutorial, su autor nos muestra cómo crear un proyecto en Visual Studio para la creación de una aplicación haciendo uso de Apache Cordova.

La clave principal del proyecto es el documento index.html, que es la página principal de la aplicación cuando inicia. Los scripts que aparecen en el documento hacen referencia a las distintas funcionalidades. Cordova.js es la puerta de enlace a Cordova, para la integración a la plataforma que estemos usando.

En la última parte del tutorial se muestra información sobre el desarrollo en plataformas Android y Windows Phone

### <http://www.tutorialspoint.com/cordova/index.htm>

Curso de Cordova que muestra bastantes aspectos claves y la mayoría de los plugins que puede usar. Contiene aspectos teóricos y prácticos

Tutorial completo sobre el desarrollo de aplicaciones a través de Apache Cordova, que cuenta con información sobre instalación y configuración, y de numerosos de los plugins que cuenta Cordova, como el de la batería, sistema de archivos o vibración, permitiendo obtener mayores conocimientos sobre las posibilidades que Apache Cordova nos ofrece y pudiendo realizar aplicaciones mas completas

### <http://events.linuxfoundation.org/sites/events/files/slides/16_9_Hazems_Cordova.pdf>

Desarrollo de aplicaciones nativas móviles JavaScript usando Apache Cordova. Introducción, configuración, línea de comnados, APIs, jQuery y demo.

Tutorial en formato pdf que incluye información básica de Apache Cordova, línea de comandos, algunas de las APIs más importantes (dispositivo, almacenamiento, cámara, notificaciones, media y geolocalización) y la integración con jQuery.

# Ayudas para estudiar las tecnologías

# Recursos para implementar las tecnologías

## Recursos para implementar Ionic

### Recursos no gratuitos para implementar la tecnología A

### 7.1.1 Recursos no gratuitos para implementar la tecnología A

## Recursos para implementar Apache Cordova

### [Monaca](https://monaca.io/):

Herramienta basada en Apache Cordova que hace que el proceso de desarrollo de aplicaciones híbridas basadas en Cordova más simple, y que cuenta con herramientas de desarrollo en la nube.

La aplicación cuenta con herramientas diseñadas para utilizar la nube y cargar el desarrollo de la app. El desarrollo en la nube permite dar flexibilidad dado que no es necesario una configuración requerida. Cuenta con herramientas para el diseño, desarrollo, backend, testing y debugging, y permite usar los frameworks Ionic ó PhoneGap

Cuenta con varias opciones de productos:

* **Monaca** **Cloud**: Proporciona un entorno de desarrollo basado en la web como servicio, sin alguna configuración
* **Monaca Visual Studio**: Desarrollo multiplataforma. Desarrollo completo de modelos híbridos.
* **Monaca Dubugger**: Permite lanzar la aplicación en tiempo real en un dispositivos, guardando el código fuente
* **Monaca LocalKitt**: Entorno de desarrollo que soporta el uso de varias herramientas Monaca a la vez, consiguiendo mayor seguridad y tranquilidad al combinar las distintas herramientas que proporciona Monaca
* **Onsen UI**: Interfaz de usuario responsiva, que permite el uso de Cordova y PhoneGap y que está optimizada para tecnologías web
* **Monaca Enterprise**: Aplicación móvil de desarrollo y framework de gestión que ofrece servicios de seguridad y eficiencia para el desarrollo y gestión de apps, incluyendo herramientas como Visual Studio, Eclipse o Dreamweaver junto con características de seguridad de encriptación y almacenamientos

Y un plan de precio, con distintas opciones

* Basis: Gratuito, con la posibilidad de incluir 3 proyectos online, 250NB de almacenamiento, IDE online, demo de desarrollo local y foro
* Developer: 160$ año. 10 proyectos online, 1GB de almacenamiento, desarrollo local con monaca CLI y posibilidad de realizar plugins en Cordova.
* Gold: 400$ año. 100 proyectos online, 10GB de almacenamiento, equipo de desarrollo, CLI, monaca para Visual Studio y posibilidad de realizar plugins en Cordova
* Platinum: 650$ año. Proyectos ilimitados y 20 GB de almacenamiento

### [Telerik](http://www.telerik.com/platform" \l "overview):

Plataforma de desarrollo que cuenta con numerosas herramientas de desarrollo (.NET, PHP, AJAX) y que cuenta con una parte de Apache Cordoba.

La herramienta de desarrollo móvil cuenta una estrategia para cada punto del proyecto, aportando una solución completo a todos el alcance.

Las partes en las que divide el desarrollo son:

* Diseño: Permite crear el diseño de la app de forma visual, con pantallas prediseñadas y editables
* Desarrollo: Desarrollo de aplicaciones móviles multiplataforma con tecnologías Web (HTML, CSS y JavaScript) basadas en Apache Cordova y PhoneGap
* Conexión con Bases de Datos: Posibilidad de conectarse a más de 20 gestores de bases de datos y sistemas ERP
* Test: Test y debug de la aplicación para los entornos iOS, Android y Windows Phone
* Gestión: Control de acceso a la aplicación y permite desactivar de forma remota la app.
* Estadísticas: Obtención de estadísticas y métricas para mejoras la funcionabilidad y rendimiento de la aplicación.

Existen modalidades de pago:

* Developer: 39$ mes. Ideal para desarrolladores que se inician en el desarrollo de apps móviles. Soporte web limitadp
* Professional: 79 $ mes. Desarrolladores profesionales. Soporte web limitado
* Business: 149 $ mes. Para equipos de desarrollo y distribución. Soporte web ilimitado
* Enterprise: Información privada

# Conclusiones