***“Plataforma Mobile para el Sistema Integrado de Información Sanitaria Argentino”***

Integrantes

|  |  |  |
| --- | --- | --- |
| Legajo | Nombre | E-Mail |
| 138.373-5 | Santiago Panizza | panizzasantiago@gmail.com |
| 134.775-5 | Lucas Roca | lucasr28@gmail.com |
| 134.853-0 | Nicolás Chaikh | nicolaschaikh@gmail.com |
| 104698-6 | Sebastian Palotte | spalotte@gmail.com |
| 248.593-0 | Leonardo Oneto | leojava2007@gmail.com |

***Director de Cátedra****: Lic. Carlos Tomassino*

***Profesor a cargo del curso****:* [*Ing.*](http://www.campusvirtual.frba.utn.edu.ar/especialidad/user/view.php?id=35051&course=1) *Claudio Crescentini*

***Profesor a cargo del proyecto****: Ing. Federico Casuscelli; Nicolás Rodríguez*

***Controller:*** *Ing. Gabriela Salem*

**Historial de Revisión**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión | Descripción | Autor |
| 16/05/2014 | 1.0 | Versión inicial. | Leonardo Oneto |
| 17/05/2014 | 2.0 | Revisión general del documento. | Nicolas Chaikh, Lucas Roca. |

Contenido

[Objetivo 4](#_Toc388274784)

[Android 4](#_Toc388274785)

[Desarrollo 4](#_Toc388274786)

[Requisitos del sistema 4](#_Toc388274787)

[PhoneGap 4](#_Toc388274788)

[Windows phone 5](#_Toc388274789)

[Windows Phone 8/8.1 5](#_Toc388274790)

[Desarrollo 6](#_Toc388274791)

[Requisitos del sistema 6](#_Toc388274792)

[Windows Phone 7/7.5/7.8 6](#_Toc388274793)

[Cuadro comparativo 6](#_Toc388274794)

[Conclusión 7](#_Toc388274795)

[Glosario 9](#_Toc388274796)

[Aplicaciones nativas 9](#_Toc388274797)

[Aplicaciones web Mobile 9](#_Toc388274798)

[Aplicaciones híbridas 9](#_Toc388274799)

[Referencias y descargas 10](#_Toc388274800)

[Compatibilidad entre versiones de Windows Phone 10](#_Toc388274801)

[Windows Phone 8 10](#_Toc388274802)

[Android 10](#_Toc388274803)

[Android vs Windows Phone 10](#_Toc388274804)

[Aplicaciones híbridas 10](#_Toc388274805)

[Phonegap 10](#_Toc388274806)

[PhoneGap Build Service 10](#_Toc388274807)

Objetivo

El objetivo del presente documento es presentar una comparación entre 2 alternativas planteadas para el desarrollo Mobile, aplicaciones nativas o híbridas, en el marco del proyecto “Plataforma Mobile para el Sistema Integrado de Información Sanitario Argentino”. Para el siguiente análisis, las tecnologías definidas son Windows Phone y Android para el desarrollo nativo, y PhoneGap para el desarrollo de aplicaciones híbridas. Las conclusiones obtenidas a partir del mismo serán de ayuda para determinar la tecnología a utilizar para el desarrollo de la aplicación.

# Android

Es la plataforma de desarrollo para dispositivos móviles con el sistema operativo Android de Google. Para el desarrollo de las aplicaciones se utiliza Java como lenguaje de programación, Eclipse como IDE con un plugin de desarrollo (ADT) y el SDK de Android.

El entorno de desarrollo cuenta con herramientas que facilitan el diseño de interfaces gráficas mediante el “drag and drop” de controles, pero son algo rudimentarias y todavía necesitan mejorarse. Mucho del trabajo de diseño se realiza editando el XML generado. Una vez que se llegan a conocer bien los detalles del diseño de interfaces mediante la edición de los archivos XML, virtualmente, no hay límites en el diseño de las mismas.

## Desarrollo

Se cuenta con la SDK de Android, este kit de desarrollo incluye herramientas para debuguear, bibliotecas de código, un emulador de Android, documentación, ejemplos de código y tutoriales.

## Requisitos del sistema

No tiene grandes requisitos de software y hardware. Se necesita tener instalada la JVM y puede ejecutarse en diversas plataformas, como Linux, Windows y Mac OS.

# PhoneGap

La creación de aplicaciones para cada dispositivo - iPhone, Android, Windows Mobile y otros, requiere diferentes frameworks, IDEs, SDKs y lenguajes. PhoneGap es una herramienta que permite escribir el código una única vez, mediante el uso de tecnologías web estándar (HTML5, CSS y JavaScript).

Según el sitio web PhoneGap, cuenta con muchas descargas y una creciente comunidad de desarrolladores y aplicaciones implementadas con este (más de 1 millón de descargas, utilizado por más de 400.000 desarrolladores y miles de aplicaciones construidas usando PhoneGap).

Nitobi fue el creador original y es uno de los mayores contribuyentes. En octubre de 2011, Adobe adquirió Nitobi y continúa su desarrollo hasta el año 2012, luego, en Octubre de ese año el código fue contribuido a la Apache Software Foundation (ASF) bajo el nombre “Apacha Corbova”.

A través de la ASF, se asegura la administración abierta del proyecto y el futuro desarrollo de PhoneGap. Siempre será libre y de código abierto bajo la licencia Apache, Versión 2.0.

Para el desarrollo de PhoneGap no se encuentra con una IDE específica, sin embargo, se puede utilizar el Eclipse o el WebStorm de JetBrains.

PhoneGap no proporciona mucha asistencia para el desarrollo de interfaces gráficas. Se puede utilizar sólo HTML y CSS, pero conviene usar algún framework como jQuery Mobile o Sencha Touch. Éstas brindan paletas de controles y resuelven mucho del trabajo que habría que realizar si se comenzara a desarrollar desde cero con HTML y CSS.

Cuenta con un servicio (PhoneGap Build cloud service) en la nube que permite a los desarrolladores compilar sus aplicaciones móviles fácilmente sin necesidad SDKs, compiladores y hardware. Si se utiliza una cuenta gratuita, se puede hostear una sola aplicación privada, el resto se tiene que almacenar en un repositorio de GitHub con visibilidad pública. Aunque también existe la posibilidad de usar el servicio de manera privada con un costo de U$10 mensuales, permitiendo alojar hasta 25 aplicaciones.

# Windows phone

## Windows Phone 8/8.1

Su versión RTM (para fabricantes) fue lanzada el 14 de septiembre de 2012 y fue liberado a los consumidores el 29 de octubre de 2012. Poco tiempo después de que saliera Windows Phone 7.8.

Están basados en la arquitectura de Windows NT, o sea, estaría construido sobre el mismo Kernel que Windows 8 y siguientes. Esto le permite ejecutar sobre procesadores con dos o más núcleos, cosa que no se puede en Windows CE. Por este cambio de kernel, es que **las aplicaciones para WP8 no son compatibles con WP7**. Aparentemente, no habrá por parte de Microsoft actualizaciones sobre las distintas versiones de WP7, esto afecta aquellos equipos que lo tengan instalado.

Microsoft ofrece una plataforma de desarrollo integrada que reduce los tiempos de startup del proyecto. Brinda facilidad para hacer deploy de las aplicaciones en dispositivos móviles, posibilidad de testear las aplicaciones en una amplia variedad de imágenes del Emulador de Windows Phone y realizar test unitarios. El lenguaje de desarrollo puede ser C# o VB .NET.

La suite de desarrollo Visual Studio ofrece al igual que Android herramientas para facilitar el diseño de interfaces gráficas mediante el “drag and drop” de controles, a diferencia de este, están más desarrolladas y mejoradas, facilitando el trabajo del diseñador.

### Desarrollo

Se debe utilizar la SDK de Windows Phone 8.0 que permite la creación de aplicaciones y juegos para Windows Phone 8.0 y Windows Phone 7.5. Incluye:

* La edición de Visual Studio Express 2012 para Windows Phone o también puede funcionar como complemento para las ediciones Visual Studio 2012 Professional, Premium o Ultimate.
* Windows Emulator, juntos con varias imágenes para distintos dispositivos.
* Herramientas adicionales para debug y pruebas de una aplicación de Windows en condiciones reales.

### Requisitos del sistema

* Sistema operativos compatibles: Windows 8 , Windows 8 Pro
* Tipo de sistema operativo: Versiones de cliente de Windows 8 64 - bit ( x64)

Hardware:

* Procesador de 64 bits (x64), que tenga soporte para virtualización Hyper-V de Microsoft.

## Windows Phone 7/7.5/7.8

**Windows Phone 7** fue la primera publicación de los SO Windows Phone. La primera versión fue liberada mundialmente el 21 de Octubre de 2010. Después siguieron las versiones 7.5 “Mango” liberada en Septiembre de 2011 y finalmente, la versión 7.8 en Junio de 2012. Siendo esta la versión final de la serie. Todas estas versiones están basadas en la arquitectura de Windows CE.

# Cuadro comparativo

|  |  |  |  |
| --- | --- | --- | --- |
| **Plataforma** | **Windows Phone** | **Android** | **Phone gap** |
| **Store** | Windows Phone Marketplace. Cuenta desarrollador: U$49. | Google Play. Para publicar apps se necesita una Cuenta que cuesta U$25. | Se tiene que distribuir la aplicación para cada uno por separado. |
| **Dispositivos** | Hay varias opciones, Nokia, Samsung, HTC, LG. Disp. De gama alta. Mayor precio. | Muchos dispositivos (Samsung, Motorola, LG, etc.). Amplia variedad (desde gama baja a alta). Cualquiera puede adquirir uno a menor precio. | Todos los anteriores. Incluyendo también dispositivos con iOS. Tendríamos la mayor cobertura. |
| **IDE** | Visual Studio Express. Mayores requisitos. Solo Windows. | Eclipse + SDK Android. Multiplataforma. | No hay uno específico. Se puede utilizar el Eclipse o el WebStorm de JetBrains. |
| **Lenguaje** | C# | Java | HTML, CSS y Javascript. |
| **Emulador** | Windows Phone Emulator. | Android Emulator. | Si se quiere probar la aplicación en un emulador, habría que usar ambos. Windows Phone y Android Emulator. |
| **Diseño de Interfaces** | Lineamientos de diseño más restrictivos. Herramientas “drag and drop” que facilitan la tarea. | Lineamientos de diseño menos restrictivos. Herramienta “drag and drop” no tan buena. Hay que escribir las interfaces en XML. | PhoneGap no proporciona mucha asistencia para el desarrollo de interfaces gráficas. Se puede utilizar solamente HTML y CSS, pero conviene usar algún framework como jQuery Mobile o Sencha Touch que allanan bastante esta labor. |
| **Mantenimiento** | Proyecto separado para Windows Phone | Proyecto separado para Android. | Un solo proyecto, los mismos archivos fuentes para todas las plataformas. |
| **Performance** | Óptima | Óptima | Buena, en los dispositivos actuales se acerca a las aplicaciones implementadas en forma nativa. |
| **Tipo de Aplicación** | Nativa | Nativa | Híbrida |

# Conclusión

Surgen dos tipos de aplicaciones que podemos desarrollar, nativas e híbridas.

En el caso de optar por una aplicación nativa, se obtendría la mejor performance para el Sistema Operativo seleccionado. Se posee acceso al hardware de los dispositivos sin restricciones y existen entornos de desarrollo específicos para cada plataforma con todas las herramientas necesarias. Dado las características del proyecto, existe la desventaja de tener que realizar el desarrollo en dos plataformas distintas, configurando los ambientes por partida doble, controlando ambos desarrollos en paralelo complicando el futuro mantenimiento y ampliación del aplicativo.

La segunda opción es realizar un desarrollo de una aplicación hibrida haciendo uso de alguna de las herramientas disponibles. En nuestro caso elegimos PhoneGap que es una herramienta de código abierto, gratuita, que tiene una comunidad de desarrolladores en crecimiento y varios casos de éxito de aplicaciones que fueron desarrolladas con la misma. La principal ventaja de este enfoque, es que el mismo código puede ser exportado para compilarse en múltiples plataformas (además de Android, iOS y Windows Phone, esta BlackBerry entre otros). Se pueden acceder a los recursos hardware de los dispositivos como la cámara, acelerómetro, etc., mediante APIs. Sin embargo, aunque permite el uso de estos recursos la performance nunca será igual a un control nativo de los mismos. Existen ciertos dispositivos que no están soportados, pero se cuenta con mecanismos para desarrollar plugins que soportan nuevo hardware (aunque no sería necesario para este proyecto). Se trabaja con tecnologías web estándar, lo cual facilita la tarea de encontrar RRHH capacitados (desarrolladores) para el proyecto, contra una aplicación nativa donde se requiere contar con programadores que conozcan diferentes lenguajes y entornos de desarrollo para cada plataforma desarrollada.

Como contra de implementar una aplicación hibrida podemos mencionar que es menos performante debido a que existe una capa intermedia (API de PhoneGap) que es la que permite el funcionamiento en múltiples plataformas.

Para el tipo de aplicación que se desarrollará en este proyecto, podemos decir que el enfoque de implementación de una aplicación híbrida es el mejor, teniendo en cuenta los beneficios de mantenibilidad del código, soporte para todas las plataformas Mobile, flexibilidad para poder extender la aplicación, utilización de tecnologías web estándar (utilizadas para desarrollo de aplicaciones web).

Dadas las características del aplicativo a desarrollar no creemos que la perdida de performance al utilizar un enfoque de desarrollo hibrido sea determinante.

# Glosario

## Aplicaciones nativas

Construidas para una plataforma en particular, utilizando su SDK, herramientas y lenguaje provistos por el proveedor de la plataforma (Por ejemplo: Xcode/Objective-C para iOS, Eclipse/Java para Android, Visual Studio/C# para Windows Phone)

## Aplicaciones web Mobile

Son aplicaciones web que ejecutan código en el servidor (PHP, ASP .NET, Node.js, etc.) que devuelven una página HTML a la cual se han aplicado estilos (CSS) para que se vean bien en las pantallas de los dispositivos móviles (Smartphone y tablets).

## Aplicaciones híbridas

Como las aplicaciones nativas se ejecutan en el dispositivo y fueron desarrolladas utilizando tecnologías web (HTML5, CSS y JavaScript). Se ejecutan dentro de un contenedor nativo, y aprovechan el motor del navegador del dispositivo (pero no el navegador o browser) para “renderizar” el HTML y procesar el código JavaScript localmente. Pueden acceder a funciones de los dispositivos como el acelerómetro, la cámara y el almacenamiento local, cosa que no se puede con aplicaciones Web Mobile.

# Referencias y descargas

http://en.wikipedia.org/wiki/Second\_Level\_Address\_Translation

## Compatibilidad entre versiones de Windows Phone

http://msdn.microsoft.com/en-us/library/windowsphone/develop/hh202996(v=vs.105).aspx

## Windows Phone 8

<http://es.wikipedia.org/wiki/Windows_Phone_8>

http://dev.windowsphone.com/en-us/

## Android

<http://en.wikipedia.org/wiki/Android_software_development>

http://developer.android.com/index.html

## Android vs Windows Phone

<http://www.codeproject.com/Articles/594184/Android-vs-Windows-Phone>

## Aplicaciones híbridas

<http://blogs.telerik.com/appbuilder/posts/12-06-14/what-is-a-hybrid-mobile-app->

## Phonegap

<http://phonegap.com/about/>

http://www.smashingmagazine.com/2014/02/11/four-ways-to-build-a-mobile-app-part3-phonegap/

## PhoneGap Build Service

https://build.phonegap.com