

APLICACIONES MÓVILES MULTIPLATAFORMA PARA CONSULTAS ACADÉMICAS EN INSTITUCIONES DE EDUCACIÓN SUPERIOR.

Janina Herrera Riofrío¹, Rene Guamán Quinche²

¹ Universidad Nacional de Loja, Ecuador, jtherrerar@unl.edu.ec

² Universidad Técnica de Manabí, Ecuador, eguman@utm.edu.ec

RESUMEN

El presente artículo se basa en un estudio realizado sobre las aplicaciones móviles multiplataforma para consultas académicas en Instituciones de Educación Superior, para lo cual se ha tomado como caso de estudio la Universidad Nacional de Loja (UNL). Durante el desarrollo de éste trabajo se realizó un análisis de las herramientas adecuadas para realizar la implementación de la aplicación, también se evaluó las metodologías de desarrollo de software para determinar la más adaptable al desarrollo de aplicaciones móviles. Luego se elaboró el análisis e implementación de la aplicación tomando como base el sistema operativo Android, posteriormente se realizó la adaptación y compilación para los sistemas operativos BlackBerry OS e iOS. Como resultado de ésta implementación se obtuvo una aplicación móvil multiplataforma la cual fue publicada en las respectivas tiendas de aplicaciones móviles, con la finalidad de determinar la utilidad de la aplicación y el grado de aceptación por parte de los usuarios de la misma. Finalmente se aplicó una encuesta a una muestra de estudiantes de la UNL y conjuntamente con las estadísticas generadas por las tiendas de aplicaciones permitieron comprobar que la aplicación móvil agiliza las consultas de información académica logrando así que los estudiantes conozcan esta información de manera oportuna y eficiente.

Palabras Clave: Aplicaciones móviles, información Académica, instituciones de educación superior.

ABSTRACT

This article is based on a study of multiplatform mobile applications for academic inquiries for Institutions of Higher Education. The National University of Loja (NUL) was used as a case study for this research. During the course of this research an analysis of the proper tools required was carried out for the implementation of this application tools. In addition software methodology development was also evaluated to determine the most compliant to mobile application development. Analysis and implementation of the application based on the Android operating system was performed, then an adaptation and compilation for Blackberry OS and iOS operating systems was carried out. As a result of this implementation a multiplatform mobile application was then distributed in respective mobile application stores in order to determine the usefulness of the application and the degree of acceptance by users of it. Finally, a survey was administered to a sample of students from the NUL and together with the statistics obtained from the mobile application stores permitted the verification that the mobile application did indeed streamline academic inquiries resulting in the fact that students found out this information in a timely and efficient manner.

KeyWords: Mobile Applications, Academic Information, Institutions of Higher Education.

1. INTRODUCCIÓN

Las Instituciones de Educación Superior (IES) generan gran cantidad de información la cual debe ser accesible por los estudiantes y personas que tengan interés en la misma. Los avances que han surgido en la Internet conjuntamente con los portales Web han constituido un gran aporte en este ámbito facilitando el acceso a la información de forma rápida sin necesidad de acudir a las IES.

Sin embargo la tecnología ha evolucionado cada vez más brindando nuevas alternativas tecnológicas, este es el caso de la tecnología móvil, la cual proporciona múltiples ventajas como son movilidad, facilidad de acceso, procesamiento de datos, entre otras. Si bien es cierto desde los dispositivos móviles se puede acceder a los portales Web que la mayoría de instituciones posee.

Sin embargo la mayoría de estos portales no se adaptan a las características de un dispositivo móvil ocasionando inconvenientes como lentitud en las consultas, dificultad para visualizar la información debido a que en algunos casos se muestra información poco relevante antes de mostrar la información que los estudiantes requieren consultar [1]. Actualmente existen una gran cantidad de aplicaciones móviles que han permitido a las IES transmitir información a sus estudiantes de manera rápida y de fácil acceso [2][3][4]. Estas razones constituyeron el punto de partida para realizar el análisis e implementación de una aplicación móvil para consultas académicas en la Universidad Nacional de Loja (UNL), la cual permite a los estudiantes registrarse mediante el usuario y contraseña proporcionado por la institución y realizar consultas de información personal, carreras y/ cursos, notas por módulo/ciclo, notas por unidad/taller e historial de matrículas. Además permite realizar consultas de las áreas y carreras que constituyen la IES. Además de ello considerando que las IES acogen una gran cantidad de población en donde se utiliza más de un sistema operativo móvil se estableció con requisito fundamental que la aplicación sea multiplataforma. Finalmente se realizó la publicación en las respectivas tiendas de aplicaciones móviles con la finalidad de que los estudiantes puedan hacer uso de la misma sin ningún tipo de restricción.

En desarrollo del presente artículo se relata los materiales y métodos utilizados para realizar el análisis e implementación de la aplicación móvil, cuyos resultados se presentan en la sección de diseño e implementación. A continuación se presenta los resultados obtenidos luego de realizar la implementación conjuntamente con la discusión de los mismos. Finalmente se menciona los trabajos relacionados al tema de estudio y se emite algunas conclusiones y trabajos futuros que pueden surgir a partir del trabajo realizado.

2. MATERIALES Y MÉTODOS

Durante el desarrollo del presente trabajo se utilizó algunos métodos de investigación científica que de describen a continuación.

Estudio de casos: Este método fue utilizado para realizar una exploración y análisis de casos de éxito sobre aplicaciones móviles de consultas de información académica a nivel internacional y a nivel de Ecuador.

Observación activa: Este método fue empleado para comprender a fondo el tema de estudio, así poder elaborar un buen análisis que permita implementar una solución que satisfaga las necesidades de los estudiantes de la Universidad Nacional de Loja(UNL).

Metodología de desarrollo de Software: La metodología utilizada en el desarrollo de la aplicación móvil fue SCRUM, misma que fue seleccionada por ser una metodología ágil y adaptable al desarrollo de aplicaciones móviles [5]. Además de ello se puede utilizar conjuntamente con la herramienta Trello, la cual proporciona un entorno interactivo que permite gestionar el desarrollo del proyecto, así como de visualizar los avances y los riesgos por lo que atraviesa durante todo el proceso de desarrollo.

A continuación se explicará más detalladamente como se desarrolló cada fase de dicha metodología.

a. Participantes en el proyecto

En esta fase se estableció el equipo que participó en el desarrollo del presente proyecto asignándoles a cada uno un rol de acuerdo a los roles que especifica la metodología Scrum [5] (Ver tabla I).

Tabla I: Equipo de trabajo

Rol	Nombre
Product Owner (Propietario del producto)	Universidad Nacional de Loja
Scrum Master (Facilitador)	Janina Herrera
Scrum Team (Equipo)	Janina Herrera Rene Guamán
Usuarios	Estudiantes de la Universidad Nacional de Loja.

b. Sprint 0: Planificación Inicial

En esta fase se realizó dos tareas fundamentales que son la especificación de requerimientos y a partir de los requerimientos encontrados se realizó la planificación del proyecto para su posterior ejecución.

- **Especificación de requerimientos**

Para desarrollar esta tarea primeramente se realizó un análisis de las necesidades y requerimientos de los estudiantes para el desarrollo de la aplicación móvil de consultas académicas. Posteriormente utilizó el estándar IEEE830 para documentar la especificación de requerimientos.

- **Planificación del proyecto**

Luego de realizar la especificación de requerimientos se procedió a elaborar la planificación del proyecto para ello se tomó como base los requerimientos encontrados dividiéndolos en Sprints (Iteraciones), también se asignó un tiempo estimado para cada Sprint. A continuación se detallan los Sprints definidos para el proyecto (Ver tabla II).

Tabla II: Sprints desarrollados en el proyecto

Nro. Sprint	Meta	Tareas	Tiempo
1	Análisis y diseño de la aplicación.	Se realizaron algunas tareas que corresponden al análisis y diseño de la aplicación móvil dentro de estas tareas se tiene: Diseñar la arquitectura de la aplicación, interfaz gráfica, elaborar el Modelado de casos de uso y diagrama de clases.	10 días
2	Autenticación en la aplicación móvil y visualización los datos personales del estudiante.	Se procedió a implementar los requerimientos asignados en la fase de planificación del proyecto, para ello se consideró al Sprint como un subproyecto en el que se realizaron las siguientes fases: <ul style="list-style-type: none"> • Análisis y diseño: tomando como base el análisis y diseño de la aplicación realizado en el Sprint 1, en esta fase se realizó nuevamente una revisión del análisis en especial de la interfaz y del diagrama de clases para determinar si lo antes realizado esta acorde con especificado en la fase de análisis de requerimientos. • Implementación: en esta fase se realizó la codificación de las funciones necesarias para dar cumplimiento a los requerimientos asignados. • Pruebas: en esta fase se realizó las pruebas funcionales de la aplicación para determinar si se cumplen los requerimientos establecidos por el 	10 días

		usuario y asignados a la iteración.	
3	Consulta del módulo actual e historial de matrículas por cada una de las carreras y/o cursos a los que pertenezca el estudiante.	Se procedió a implementar los requerimientos asignados a este Sprint, para ello se siguió las 3 fases descritas en el Sprint 2.	10 días
4	Consulta de información sobre las áreas, con sus respectivas carreras que oferta la UNL.	En este Sprint se realizó a implementar los requerimientos asignados a la misma, para ello se siguió las 3 fases descritas en el Sprint 2.	10 días
5	Compilar la aplicación para sistema operativo BlackBerry e iOS.	Se realizó la compilación de la aplicación móvil para sistema operativo BlackBerry e iOS. También se realizaron pruebas funcionales de la aplicación en estas dos plataformas.	5 días
6	Publicar la aplicación en las tiendas de Android, BlackBerry e iOS.	Finalmente en este Sprint se realizó la publicación de la aplicación móvil en las respectivas tiendas como Google Play, BlackBerry World y App Store, con la finalidad de que la aplicación este sea accesible para los estudiantes de la UNL.	5 días

3. DISEÑO E IMPLEMENTACIÓN

3.1 Análisis de frameworks para el desarrollo de aplicaciones móviles multiplataforma.

Los dispositivos móviles han evolucionado de tal forma que cada día existen más personas que usan estos dispositivos, de igual forma se han desarrollado diferentes sistemas operativos móviles como son Android, iOS, BlackBerry, Windows Mobile, entre otros. La variedad de sistemas operativos que existen obligan a los desarrolladores de aplicaciones móviles en elaborar una solución para cada una de las plataformas lo que resulta costoso tener que desarrollar la misma aplicación más de una vez puesto que para cada plataforma se debe conocer el lenguaje y las reglas que cada plataforma provee [1].

Una alternativa para este tipo de inconvenientes es desarrollar aplicaciones Web Mobile mediante el uso de frameworks como son jQuery Mobile, Sencha Touch, entre otros. Sin embargo estas tecnologías no permiten acceder a funciones propias de cada dispositivo como cámara, contactos, etc. Además estas aplicaciones no pueden distribuirse mediante las tiendas de aplicaciones de cada plataforma.

Para dar solución a estas necesidades se ha desarrollado Frameworks como Phonegap, Titanium Appcelerator, entre otros [6], los cuales permite desarrollar aplicaciones híbridas multiplataforma. Analizando estos Frameworks se puede concluir que tanto Phonegap como Titanium permite desarrollar la aplicación móvil multiplataforma para consultas académicas en la Universidad Nacional de Loja. Sin embargo analizando las características de cada uno de estos Frameworks se puede evidenciar que para el desarrollo de la aplicación móvil antes mencionada resulta factible utilizar Phonegap puesto que se trata de una aplicación de consulta de datos en donde es primordial el diseño de la interfaz, misma que se la puede realizar eficientemente mediante el uso de tecnologías que nos brinda Phonegap como son HTML5, CSS3, JavaScript y la integración con Frameworks como jQuery Mobile y Bootstrap, permitiendo obtener una interfaz más amigable e intuitiva [6].

3.2 Evaluar el Servicio web del Sistema de Gestión Académico (SGA)

Para el desarrollo de la aplicación móvil se utilizará un Servicio Web SOAP, el cual proporciona algunos métodos de consulta de datos personales, académicos y estadísticos. Las respuestas del Servicio Web son

en XML formateado con JSON para facilitar el intercambio de datos; en este caso la consulta de información al SGA.

El Servicio Web (WS) se encuentra alojado en un servidor de la UNL [7]. Para acceder a esta biblioteca de métodos es necesario autenticarse con el respectivo usuario y contraseña previamente registrados en el (WS).

3.3 Análisis y diseño de la Aplicación

3.3.1 Requerimientos funcionales

En la Tabla III se presenta los requerimientos funcionales establecidos para la aplicación móvil.

Tabla III: Requerimientos Funcionales

Nombre	Descripción
Ingresar a la aplicación.	La aplicación móvil permitirá al Estudiante ingresar a la aplicación mediante en número de cédula y clave utilizados en el Sistema de Gestión Académico.
Visualizar datos personales.	La aplicación móvil permitirá al Estudiante visualizar sus datos personales (cédula/DNI, nombres, email, teléfono, dirección).
Consultar carreras y/o cursos.	La aplicación móvil permitirá al Estudiante consultar la(s) carrera(s) y/o curso(s) al que pertenezca.
Consultar el módulo actual con su respectiva información.	La aplicación móvil permitirá al Estudiante consultar el módulo actual en el que se encuentra matriculado por cada de una de la(s) carrera(s) y/o curso(s) al que pertenezca.
Consultar el historial de matrículas.	La aplicación móvil permitirá al Estudiante consultar el historial de matrículas por cada de una de la(s) carrera(s) y/o curso(s) al que pertenezca.
Consultar notas por cada carrera y/o curso, módulo y unidad.	La aplicación móvil permitirá al Estudiante consultar notas por cada carrera y/o curso, modulo y unidad.
Consultar áreas y carreras de la UNL.	La aplicación móvil permitirá al Usuario consultar información de las áreas que tiene la UNL, así como las carreras pertenecientes a cada área.

3.3.2 Arquitectura de la aplicación y Diagrama de casos de uso

La Figura 1 muestra la arquitectura de la aplicación móvil en la cual se especifica la tecnología que se utilizó el desarrollo de la misma y en la Figura 2 se presenta el diagrama de los casos de uso implementados en el desarrollo de la misma.



Figura 1. Arquitectura de la aplicación

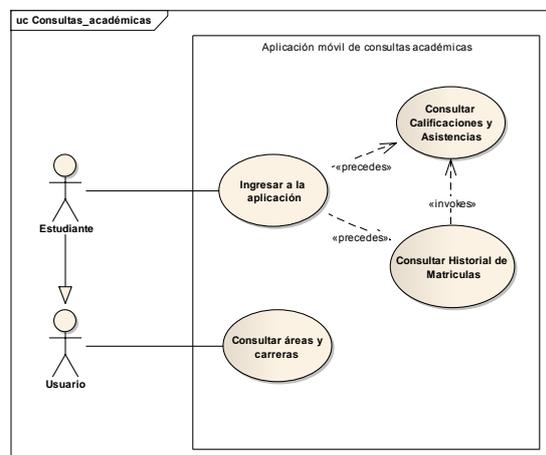


Figura 2: Diagrama de casos de uso

3.3.3 Diagrama de Clases

La figura 3 presenta el diagrama de clases elaborado para la implementación de la aplicación móvil, cada clase corresponde a un archivo JavaScript y los métodos corresponden a las funciones implementadas.

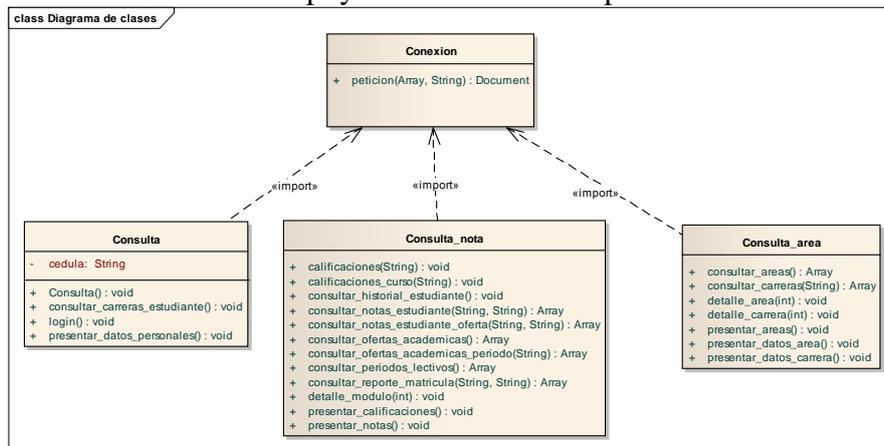


Figura 3. Diagrama de Clases

3.4 Implementar la aplicación móvil para consultas de información académica.

3.4.1 Creación de un proyecto con PhoneGap para Android

Para crear el proyecto para Android se ha utilizado el IDE Eclipse en el cual se realizaron los siguientes pasos:

- Descargar PhoneGap versión 2.9 desde la página oficial.
- Crear un proyecto de Android en Eclipse, para ello se ha como versión mínima la 2.3 para que la aplicación función desde esta versión hasta la última versión de Android.
- En el directorio assets del proyecto se creó un directorio llamado www.
- Dentro del directorio www se creó los directorios js, css e img.
- Luego de la carpeta descargada de Phonegap/Android/lib se copió los siguientes archivos:
 - El archivo cordova.js se lo copia en el directorio www/js.
 - El archivo cordova.jar se lo copia en el directorio libs.
 - La carpeta xml se la copia en el directorios res.
- Luego se debe añadir la librería cordova.jar a la ruta de construcción del proyecto Android para ello se debe presionar clic derecho en la librería, luego en Build Path y finalmente en Add to Build Path.
- Para que el proyecto se pueda ejecutar como una aplicación nativa se debe ubicar el siguiente código en el archivo MainActivity.java (ver Tabla IV).

Tabla IV: Código de MainActivity.java

```
import org.apache.cordova.*;
import android.os.Bundle;
public class MainActivity extends DroidGap {
    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        super.loadUrl(Config.getStartUrl());
    }
}
```

Luego se añadió las los archivos js y css de los Frameworks Front End en los respectivos directorios creados. Para ello primeramente se descargó dichos archivos desde la página oficial de jQuery Mobile [8], jQuery [9] y Bootstrap [10]. Luego de haber creado y configurado el proyecto se procedió a implementar la aplicación para ello todos los archivos creados se ubicaron en el directorio www en sus respectivos directorios.

3.4.2 Conexión con el Servicio Web del SGA.

Para realizar la conexión con el Servicio Web (WS) se ha creado una función en JavaScript llamada **peticion (url, parámetros)**, la cual recibe como parámetros la URL según el método del servicio web que se vaya a consultar y los parámetros de dicho método. Este método se ha implementado de forma genérica para que pueda ser utilizado en todas las consultas hacia el WS. Dentro de esta función se ha utilizado la función \$.ajax(), quedando el método de la siguiente manera (ver Tabla V).

Tabla V: Petición al Servicio Web.

```
function peticion(url, parámetros){
var respuesta;
$.ajax( {
    type: "POST",
    url:url,
    data: parametros,
    contentType: "application/x-www-form-urlencoded; charset=utf8",
    dataType: "xml",
    async: false,
    beforeSend: function(xhr) {
    xhr.setRequestHeader("Authorization", "Basic " +
    btoa("sgamovil"+":" + "jtSGAmov"));},
    success: function (data,estado,jpxhr) {
        if(data!=null){
            respuesta=data;
        }else{
            alert("Servicio de consultas no disponible");
            window.location="index.html"; } },
    error: function (error) {
        alert("Servicio de consultas no disponible");
        window.location="index.html"; } });
return respuesta; }
```

La función \$.ajax() posee algunos parámetros que se deben considerar al momento de implementar esta función. A continuación se explicará el valor asignado a cada uno de ellos.

- **type:** Se utilizó el método “POST”, puesto que es más seguro para enviar los datos de consulta al servidor.
- **url:** Se le asigna el valor de URL recibida como parámetro en la función.
- **data:** Se le asigna el valor de parámetros recibidos en la función. Posteriormente se explicará cómo se debe construir la URL y la especificación de parámetros.
- **contentType:** Este parámetro toma el valor de “application/x-www-form-urlencoded; charset=utf8”, el primer dato es el valor por defecto que se debe asignar, y la codificación utf8 se especifica puesto que las peticiones **XMLHttpRequest** siempre deben ser codificadas con utf8.
- **dataType:** Este parámetro toma el valor “xml”, puesto que el servicio web retorna los datos con una estructura xml.
- **async:** Este parámetro se le asigna el valor “false”, lo cual indica que se va a realizar una aplicación síncrona. En este tipo de consultas se debe utilizar peticiones síncronas puesto que la aplicación depende de los datos solicitados al servidor para continuar con la ejecución de la aplicación.
- **beforeSend:** Aquí se utiliza una función para realizar la autenticación al Servicio Web, lo que permite esta función es modificar la cabecera de la petición indicando el usuario y la contraseña respectiva. Dentro de esta función se hace uso del método **btoa (String autenticación)** el cual permite encriptar las credenciales utilizadas para que la autenticación se realice de manera segura.
- **success:** Aquí se llama a una función la cual se ejecutará si la petición se ejecuta correctamente. La función recibe un parámetro llamado data el cual tendrá los datos que devuelve la petición, por esta razón se asigna esta variable a la variable respuesta que es el valor que retorna la función.
- **error:** Aquí se llama una función la cual se va a ejecutar si la petición realizada al servicio web falla. En esta función se presenta un mensaje indicativo y se re direcciona la aplicación a la pantalla principal de la misma.

3.4.3 Implementar las funciones específicas para la consulta de información académica.

Luego de haber implementado el método petición el cual servirá para realizar las consultas al Servicio Web se procedió a implementar los métodos especificados en el diagrama de clases (Ver figura 3).

A continuación se explica el proceso de implementación de los métodos. Primeramente se recupera los valores ingresados por el usuario de la siguiente manera:

```
var cedula = document.getElementById("cedula").value;
```

Después de recuperar los datos se verifica que ninguno de los valores sea cadena vacía, en caso de serlo se presenta un mensaje al estudiante, caso contrario se procede a validar los datos. Para realizar la validación se utilizó un método del servicio web llamado **sgaws_validar_estudiante**, el cual recibe como parámetros la cedula y la clave del estudiante y devuelve true si los datos son correctos, caso contrario devuelve false. Para realizar la llamada al método se debe primeramente especificar la url donde está ubicado el servicio web, seguido del nombre del método quedando de la siguiente manera:

```
var url= "http://ws.unl.edu.ec/sgaws/wsvalidacion/sgaws_validar_estudiante";
```

Luego se debe especificar los parámetros, los cuales deben ir en un arreglo de tipo clave/valor de la siguiente manera: **var parámetros={cedula:cedula,clave:clave};**

A continuación se llama a la función **petición(url, parámetros)**, explicada anteriormente la cual retorna los datos que devuelve el servidor. Estos datos son retornados en una estructura XML, por lo que es necesario extraer los datos de la siguiente manera:

```
var data=peticion(url,parametros);
```

```
var respuesta= data.getElementsByTagName("result")[0].childNodes[0].nodeValue;
```

Este proceso fue realizado en todas las consultas realizadas al servidor mediante el WS, cabe recalcar que la URL y los parámetros están en función del método que se requiera utilizar. En los métodos que retorna un String dentro de la estructura XML se utilizó el método `JSON.parse()` para convertir el String devuelto como respuesta a un Objeto JSON y poder extraer la información de manera eficiente para presentarla en páginas HTML que forman parte de la aplicación.

3.4.4 Compilar la aplicación para sistema operativo Blackberry OS e iOS.

a. Compilar la aplicación para Blackberry.

La aplicación ha sido compilada para BlackBerry OS para ello primeramente se ha instalado las herramientas BlackBerry WebWorks y PhoneGap. Luego de ello se ejecutó los siguientes pasos:

- **Crear el proyecto para BlackBerry OS**

Del proyecto construido de PhoneGap para Android se ha copiado el contenido del directorio `www`, posteriormente se ha creado un archivo llamado `config.xml` en donde se establecerán algunas configuraciones de proyecto como son nombre de la aplicación, icono, entre otros (ver Tabla VI).

Tabla VI: Archivo `config.xml` para BlackBerry OS

```
<?xml version='1.0' encoding='utf8'?>
<widget id="UNL" version="0.0.1"
xmlns="http://www.w3.org/ns/widgets"
xmlns:cdv="http://cordova.apache.org/ns/1.0"
xmlns:rim="http://www.blackberry.com/ns/widgets">
  <name>UNL</name>
  <author>Janina Herrera</author>
  <description>Universidad Nacional de Loja</description>
  <content src="index.html" />
  <preference name="Orientation" value="default" />
  <access uri="*" subdomains="true" />
  <preference name="websecurity" value="disable" />
  <icon src="img/icono.png" />
</widget>
```

Es muy importante verificar que el tamaño del icono no sea demasiado grande para que no presente problemas al momento de compilar la aplicación. En la aplicación realizada en tamaño del icono es de 13.6KB.

Luego de haber especificado las configuraciones se comprime se comprime todos los archivos en formato .zip para poder ser compilada.

- **Solicitar y registrar las claves para firmar la aplicación**

Para solicitar las claves se debe ingresar a la siguiente dirección [<https://www.blackberry.com/SignedKeys/codesigning.html>], en donde se selecciona la versión de Sistema operativo para el cual se va a publicar; en este caso se realizó para BlackBerry OS.

Luego de haber llenado los datos y aceptado los términos de la licencia, se procede a enviar el formulario con la solicitud. Una vez realizada la solicitud en el transcurso de una hora llegarán al correo electrónico indicado 3 archivos .csi.

Para generar las claves se debe abrir la consola de comandos y ubicarse en el directorio **bin** de la carpeta de instalación de BlackBerry WebWorks, en este caso se encuentra en:

C:\Program Files (x86)\Research In Motion\BlackBerry WebWorks SDK 2.3.1.5\bin

En esta dirección se deben colocar los 3 archivos .csi que se recibieron en el correo electrónico. Luego se debe ejecutar en siguiente comando para cada uno de los archivos csi:

java -jar SignatureTool.jar nombearchivo.csi

La primera vez que se ejecuta el comando antes indicado, es necesario colocar el PIN que se registró al momento de solicitar las claves y una contraseña, misma que servirá para compilar posteriormente la aplicación. Una vez que se haya realizado el proceso para los 3 archivos, se debe verificar que en el directorio **bin** donde se ubicaron los archivos csi se hayan generado dos archivos **sigtool.csk** y **sigtool.db**.

- **Compilar la aplicación**

Luego de haber creado el proyecto y registrado las claves para compilar la aplicación se debe ubicar en el directorio de instalación de BlackBerry WebWorks, ejemplo:

C:\Program Files (x86)\Research In Motion\BlackBerry WebWorks SDK 2.3.1.5\bin

Luego ejecutar el siguiente comando:

bbwp rutaarchivo\archivo.zip -g contraseña -o rutaaparalaaplicacióncompilada

Si el archivo .zip se encuentra en el mismo directorio no es necesario especificar la ruta del archivo.

Una vez ejecutado el comando se firman todos los archivos que se generan durante el proceso. Al finalizar la compilación se debe verificar que en la ruta especificada se hayan creado las carpetas **StandardInstall** y **OTAInstall**, las cuales son necesarias para publicar la aplicación en la respectiva tienda.

b. **Compilar la aplicación para iOS**

Para compilar la aplicación para iOS es necesario hacer uso de una computadora Mac e instalar las herramientas Xcode, Phoneygap y Apache Córdoba. Luego de ello ejecutar los siguientes pasos:

- **Crear el proyecto:** Para crear el proyecto para iOS se ejecutó los siguientes comandos (Ver figura 8).

```
Mac-Pro-de-Janina:Desktop janinaherrera$ cordova create SGA_IOS
Creating a new cordova project with name "HelloCordova" and id "io.cordova.hellocordova"
at location "/Users/janinaherrera/Desktop/SGA_IOS"
Downloading cordova library for www...
Download complete
Mac-Pro-de-Janina:Desktop janinaherrera$ cd SGA_IOS/
Mac-Pro-de-Janina:SGA_IOS janinaherrera$ cordova platform add ios
Creating ios project...
```

Figura 8: Crear proyecto iOS.

- **Agregar los archivos de la aplicación:** Dentro del proyecto creado se debe copiar el directorio **www** del proyecto Android. En este archivo se encuentra la implementación realizada para dar solución al tema planteado.
- **Crear el archivo config.xml:** Dentro del proyecto se debe crear el archivo config.xml, en el cual se indique el nombre de la aplicación, la ruta donde se encuentra el icono y algunas configuraciones necesarias para el correcto funcionamiento de la misma. La Tabla VII muestra el código del archivo config.xml.

Tabla VII: Código de config.xml para iOS

```
<?xml version='1.0' encoding='utf8'?>
<widget id="io.cordova.hellocordova" version="0.0.1"
xmlns="http://www.w3.org/ns/widgets"
xmlns:cdv="http://cordova.apache.org/ns/1.0">
<name>UNL</name>
<description>
    Universidad Nacional de Loja
</description>
<author email="jtherrera@unl.edu.ec"
href="http://cordova.io">
    Janina Herrera
</author>
<content src="index.html" />
<access origin="*" />
<icon src="www/img/icono.png" />
</widget>
```

- **Compilar la aplicación:** Para compilar la aplicación y poder emularla en XCODE, se debe ejecutar el siguiente comando: **cordova build**.
- **Emular la aplicación:** Para emular la aplicación primeramente se debe acceder al directorio **NombreAplicación/platforms/ios** y dar doble clic al archivo **.xcodeproj**.

4. RESULTADOS

En el presente trabajo de investigación se realizó el análisis e implementación de una aplicación móvil para consultas académicas en la Universidad Nacional de Loja (UNL), para ello se utilizó la metodología de desarrollo de Software SCRUM y la herramienta Trello para gestionar el proceso de desarrollo del proyecto. Para la implementación se utilizó el Framework para desarrollo de aplicaciones móviles multiplataforma PhoneGap.

Como resultado de este trabajo se obtuvo una aplicación móvil para los sistemas operativos Android, BlackBerry OS e iOS, la cual fue publicada en las respectivas tiendas de aplicaciones móviles como son: Google Play [11], BlackBerry World [12] y App Store [13].

Esta aplicación permite a los estudiantes autenticarse mediante el usuario y contraseña proporcionada por la Unidad de Telecomunicaciones e información de la UNL. Una vez que ingresa el estudiante puede visualizar sus datos personales, así como las carreras y/cursos a los que pertenece. Por cada carrera y/o curso el estudiante puede consultar el historial de matrículas, las notas y asistencias de cada módulo/ciclo. También permite consultar las notas detalladas por cada unidad o taller. Además de ello permite consultar información de las áreas y carreras de la UNL. La figura 4 muestra algunas consultas realizadas en un dispositivo Android.

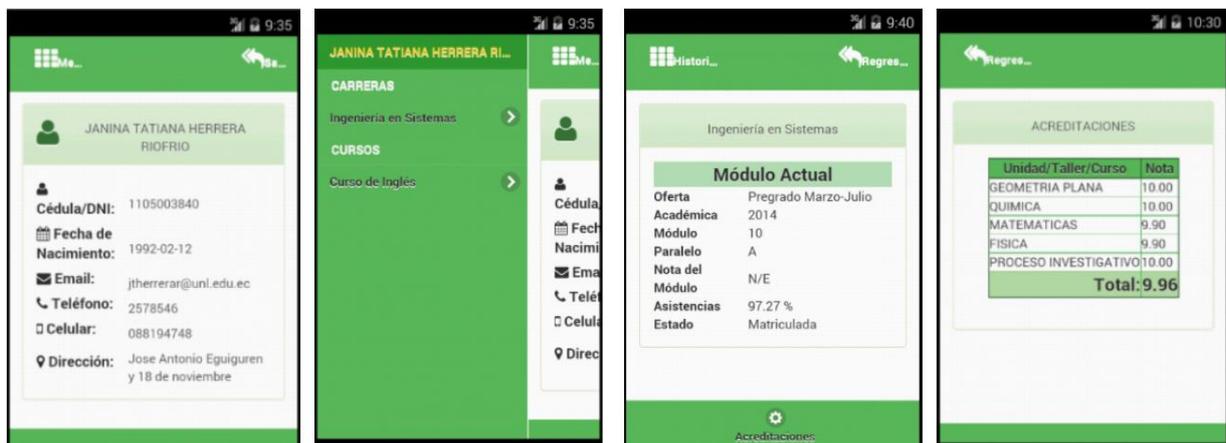


Figura 4. Consultas realizadas en la aplicación.

El funcionamiento de la aplicación es el idéntico en las tres plataformas para las cuales fue implementada, esto fue posible gracias al uso del Framework PhoneGap, en el cual se empleó tecnologías comunes para los tres sistemas operativos como son HTML5, CSS, JavaScript y un Servicio Web (WS) para realizar las consultas al servidor de la UNL y presentar la información académica en la aplicación móvil. El WS constituyó un componente fundamental en la implementación de la aplicación puesto que permitió la interoperabilidad entre la aplicación móvil y el Sistema de Gestión Académico de la UNL [14][15].

El rendimiento de la aplicación depende de la versión de sistema operativo del dispositivo móvil, según las pruebas realizadas y una encuesta aplicada a una muestra de 50 estudiantes de la UNL se pudo determinar que el 67% de encuestados concuerdan que el rendimiento de la aplicación fue excelente, contrastando estos resultados con las estadísticas que proporcionan las tiendas de aplicaciones se pudo determinar que la mayoría de usuarios instalaron la aplicación en las versiones más recientes de cada sistema operativo.

El 94% de encuestados consideran que la aplicación móvil les facilita la consulta de información académica, puesto que les permite acceder a su información desde cualquier lugar donde tengan acceso a internet. Además el estar disponible para los 3 sistemas operativos más utilizados en la Universidad proporciona a los estudiantes una alternativa de consulta sin restricciones en cuanto al sistema operativo del dispositivo móvil.

La aplicación móvil resulta también más eficiente puesto que la interfaz de usuario se adapta a las características del dispositivo móvil como se puede evidenciar en la figura 5, a diferencia de las aplicaciones Web en las cuales se presentan algunos inconvenientes para los usuarios al momento de visualizar la información que requieren consultar.

5. TRABAJOS RELACIONADOS

En relación a las aplicaciones móviles multiplataforma para consultas académicas, el trabajo presentado por Vera Rendón en [1], explica que debido a la incompatibilidad, velocidad y calidad de servicio que proporciona el portal de la Universidad Politécnica Salesiana cuando los estudiantes acceden mediante un dispositivo móvil, es necesario la implementación de una aplicación móvil nativa la cual facilite a los estudiantes el proceso de consulta. Aunque en dicho trabajo no se realiza la publicación de la aplicación móvil, es un punto de referencia importante puesto que se presentan los beneficios que brinda PhoneGap para el desarrollo de aplicaciones móviles multiplataforma. En [16] se presenta un estudio sobre los beneficios que proporcionan los Web Service (WS) como son la interoperabilidad entre aplicaciones y su uso en el desarrollo de aplicaciones móviles. En [17] se presenta un estudio sobre el uso de la Metodología SCRUM para el desarrollo de aplicaciones móviles, en la cual los investigadores concluyen que esta metodología es aplicable a proyectos pequeños y medianos con entornos cambiantes; este es el caso de las aplicaciones móviles en las cuales es de vital importancia que los usuarios finales conozcan los avances del proyecto y así poder realizar cambios en tiempo de desarrollo para evitar cambios significativos cuando el proyecto haya finalizado y publicado en las tiendas de aplicaciones.

6. CONCLUSIONES Y TRABAJO FUTURO

El análisis de requerimientos permitió conocer a fondo las necesidades de los usuarios para así realizar una aplicación acorde a sus expectativas. Además la definición de criterios de aceptación para cada uno de ellos constituyó el punto de partida para la ejecución de pruebas funcionales al finalizar el proyecto. El uso del Framework PhoneGap permitió implementar la aplicación móvil con las mismas funcionalidades para las plataformas Android, BlackBerry e iOS y distribuirla en las respectivas tiendas de aplicaciones móviles, facilitando a los usuarios el acceso y uso de la misma. La aplicación móvil agiliza las consultas de información académica en la Universidad Nacional de Loja, puesto que permite a los estudiantes consultar su información desde cualquier lugar y de manera sencilla, sin necesidad de acceder mediante un navegador Web.

La utilización de un Servicio Web (WS) para acceder a la información del Sistema de Gestión Académico permitió realizar las consultas de manera eficiente y segura, puesto que utilizó el protocolo SOAP (*Simple Object Access Protocol*) para el intercambio de la información y el método `btoc()` para encriptar los datos de autenticación al WS.

La aplicación móvil permitió testear el Servicio Web del Sistema de Gestión Académico ayudando así a corregir algunos Bugs detectados en el mismo. La aplicación soporta una cantidad de 500 a 700 usuarios aproximadamente que ingresen a la aplicación de manera simultánea, variando el tiempo de respuesta en función de dicho número de usuarios.

7. AGRADECIMIENTOS

La presente investigación forma parte del Trabajo de Titulación previo a la obtención del título de Ingeniero en Sistemas en la Universidad Nacional de Loja cuya problemática versa sobre el tema: “Aplicación móvil para consultas académicas en la Universidad Nacional de Loja”.

Los autores agradecen a las autoridades de la Universidad Nacional de Loja (UNL); el Área de la Energía, las Industrias y los Recursos Naturales no renovables; a la planta docente de la Carrera de Ingeniería en Sistemas y a la Unidad de Telecomunicaciones e Información de la UNL, por el apoyo brindado para la elaboración de la presente investigación.

8. REFERENCIAS BIBLIOGRÁFICAS

1. J. Vera, Aplicaciones nativas de dispositivos móviles para acceso a la información personal de la página web de los estudiantes de la Universidad Politécnica Salesiana, Universidad Politécnica Salesiana, Guayaquil Ecuador, 2012.
2. La Universidad de Salamanca crea una aplicación móvil para notificar a los alumnos la publicación de notas, <http://www.usal.es/webusal/node/34592>, Fecha publicación: 12/09/2013.
3. Nuntiatum USAL en Google Play, <https://play.google.com/store/apps/details?id=es.usal.notificaciones.cliente.android&hl=es>, Última actualización: 20 de febrero de 2014.
4. Consulta de notas UPNA en Google Play, <https://play.google.com/store/apps/details?id=com.sigma.mobile.target.upna>, Última actualización: 28 de diciembre de 2012.
5. R. Pressman. Ingeniería de Software: Un enfoque práctico, McGrawHill, Quinta Edición, pp. 5778.
6. L. Corral, A. Sillitti, and G. Succi, "Mobile Multiplatform Development: An Experiment for Performance Analysis," *Procedia Comput. Sci.*, vol. 10, pp. 736–743, Jan. 2012.
7. Servicio Web de la Universidad Nacional de Loja, <http://ws.unl.edu.ec/>.
8. jQuery Mobile, <http://jquerymobile.com/>.
9. jQuery, jquery.com/.
10. Bootstrap, getbootstrap.com/.
11. Aplicación móvil UNL en Google Play, <https://play.google.com/store/apps/details?id=org.unl.sga&hl=es419>, Fecha de actualización: 28 de Julio del 2014.
12. Aplicación móvil UNL en Blackberry World, <http://appworld.blackberry.com/webstore/content/58391263/?lang=en&countrycode=EC>, Fecha de actualización: 31 de Julio del 2014.
13. Aplicación móvil UNL en App Store, <https://itunes.apple.com/es/app/unl/id894356480?mt=8>, Fecha de actualización: 02 de Octubre del 2014.
14. K. Mohamed and D. Wijesekera, "Performance Analysis of Web Services on Mobile Devices," *Procedia Comput. Sci.*, vol. 10, pp. 744–751, Jan. 2012.
15. J. Bertram and C. Kleiner, "Secure Web Service Clients on Mobile Devices," *Procedia Comput. Sci.*, vol. 10, pp. 696–704, Jan. 2012.
16. K. Chancusi, M. Vergara, M. Campaña, Método Ágil SCRUM, aplicado a la implantación de un Sistema Informático para el proceso de recolección masiva de información con tecnología móvil.
17. N. Rodríguez, A. Martin, A. Valenzuela, S. Chávez, Consumo de Web Service desde dispositivos móviles heterogéneos.