



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA
CONSULTAR EL HORARIO DE LA FACULTAD DE INGENIERÍA DE LA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

Ariel Darío León Gálvez

Marcos Alejandro Sánchez Vásquez

Asesorado por el Ing. Edgar Estuardo Santos Sutuj

Guatemala, julio de 2015

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA
CONSULTAR EL HORARIO DE LA FACULTAD DE INGENIERÍA DE LA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

ARIEL DARÍO LEÓN GÁLVEZ

MARCOS ALEJANDRO SÁNCHEZ VÁSQUEZ

ASESORADO POR EL ING. EDGAR ESTUARDO SANTOS SUTUJ

AL CONFERÍRSELES EL TÍTULO DE

INGENIEROS EN CIENCIAS Y SISTEMAS

GUATEMALA, JULIO DE 2015

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Pedro Antonio Aguilar Polanco
VOCAL I	Ing. Angel Roberto Sic García
VOCAL II	Ing. Pablo Christian de León Rodríguez
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Narda Lucía Pacay Barrientos
VOCAL V	Br. Walter Rafael Véliz Muñoz
SECRETARIA	Inga. Lesbia Magalí Herrera López

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympto Paiz Recinos
EXAMINADOR	Ing. Edgar Estuardo Santos Sutuj
EXAMINADOR	Ing. José Alfredo González Díaz
EXAMINADOR	Ing. Marlon Francisco Orellana López
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA CONSULTAR EL HORARIO DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha febrero de 2015.



Ariel Darío León Gálvez



Marcos Alejandro Sánchez Vásquez

Guatemala, 27 de mayo de 2015

Ingeniero
Marlon Antonio Pérez Turk
Director
Escuela de Ciencias y Sistemas
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

Ingeniero Pérez Türk:

Me complace saludarle haciendo referencia al trabajo de graduación titulado "DESARROLLO E IMPLEMENTACION DE UNA APLICACIÓN MOVIL PARA CONSULTAR EL HORARIO DE LA FACULTAD DE INGENIERIA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA", desarrollado por los estudiantes universitarios Ariel Darío León Gálvez con número de carné 200312744 y Marcos Alejandro Sánchez Vásquez con número de carné 200312877, que como asesor apruebo el contenido del mismo.

Para su conocimiento y efectos, sin otro particular, me suscribo.



Atentamente,
Ing. Edgar Santos
Asesor

Edgar Santos
INGENIERO EN CIENCIAS Y SISTEMAS
Colegiado 5266



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 10 de Junio del 2015

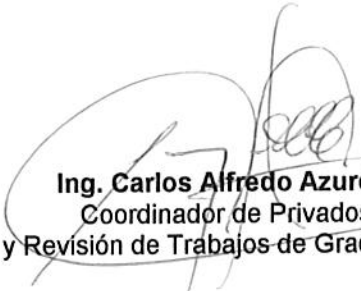
Ingeniero
Marlon Antonio Pérez Türk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación de los estudiantes **ARIEL DARIO LEÓN GÁLVEZ** con carné 2003-12744, y **MARCOS ALEJANDRO SÁNCHEZ VÁSQUEZ** con carné 2003-12877, titulado: "DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA CONSULTAR EL HORARIO DE LA FACULTAD DE INGENIERIA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA", y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación **“DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA CONSULTAR EL HORARIO DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**, realizado por los estudiantes **ARIEL DARÍO LEÓN GÁLVEZ** y **MARCOS ALEJANDRO SÁNCHEZ VÁSQUEZ**, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”

Ing. ~~Marlon~~ Antonio Pérez Türk
Director, Escuela de Ingeniería en Ciencias y Sistemas



Guatemala, 13 de julio de 2015



El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **DESARROLLO E IMPEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA CONSULTAR EL HORARIO DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, presentado por los estudiantes universitarios: **Ariel Darío León Gálvez y Marcos Alejandro Sánchez Vásquez**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.

9/07/15
Ing. Pedro Antonio Aguilar Polanco
Decano



Guatemala, julio de 2015

/cc

ACTO QUE DEDICO A:

- Dios** Por darme el regalo de la vida y la sabiduría necesaria para cada prueba superada.
- Mis padres** Elder León y Silvia Gálvez, por su amor incondicional a lo largo de la vida, sus consejos, ejemplo, sacrificio y confianza para brindarme educación.
- Mi esposa** Astrid Sánchez, por ser mi ayuda idónea y mayor motivación para seguir adelante en la recta final de mi carrera.
- Mi hermano** Romeo León, por su apoyo, compañía y cariño durante todos estos años.



Ariel Darío León Gálvez

ACTO QUE DEDICO A:

- Dios** Por darme la vida, las fuerzas y la voluntad hasta finalizar la carrera.
- Mi madre** Seidy Maricell Vásquez Mogollón, por su esfuerzo y sacrificio de sacarnos adelante a mi hermana y a mí, y darnos siempre lo mejor.
- Mi esposa** Kimberly Alvarado Cabrera, por su apoyo incondicional.
- Mi hijo** Ian Mateo, mi mayor privilegio y razón diaria de vivir.
- Mi hermana** Laura Michelle Laparra, por la compañía y motivación que me ha brindado.



Marcos Alejandro Sánchez Vásquez

AGRADECIMIENTOS A:

**Universidad de San
Carlos de Guatemala**

Por ser esta gran casa de estudios, fuente de conocimiento y superación.

Facultad de Ingeniería

Por proveer los medios que me permitieron completar estos años de estudio.

Mis amigos de la Facultad

Por su ayuda en cada semestre, fundamental para completar esta meta.

**Mis compañeros de
trabajo**

Por enseñarme a poner en práctica los conocimientos adquiridos en la carrera.

Mis asesores

Ing. Alvaro Donis e Ing. Edgar Santos, por su gran colaboración en la realización de este trabajo.


Ariel Darío León Gálvez

AGRADECIMIENTOS A:

**Universidad de San
Carlos de Guatemala**

Por permitirme el acceso a la educación superior.

Facultad de Ingeniería

Por formarme como profesional, proveyéndome el conocimiento para desempeñar mi labor.

Mis amigos

Gracias por todo su apoyo y amistad brindada durante la carrera.



Marcos Alejandro Sánchez Vásquez

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	VII
LISTA DE SÍMBOLOS	XI
GLOSARIO	XIII
RESUMEN.....	XVII
OBJETIVOS.....	XIX
INTRODUCCIÓN.....	XXI
1. ESTUDIO DE LA TECNOLOGÍA Y SU IMPACTO EN GUATEMALA.....	1
1.1. Modelo de investigación	1
1.1.1. Utilidad percibida	2
1.1.2. Facilidad de uso percibida	2
1.1.3. Intención de uso	2
1.1.4. Uso real del sistema	2
1.2. Relación con la tecnología.....	3
1.2.1. Utilidad y facilidad de uso percibida.....	3
1.2.2. Intención de uso	3
1.2.2.1. Encuesta.....	4
1.2.2.2. Resultados.....	4
1.2.3. Uso real del sistema	10
2. IDENTIFICACIÓN DEL PROBLEMA Y SOLUCIÓN.....	11
2.1. Antecedentes.....	11
2.2. Aplicaciones móviles	11
2.2.1. Tipos de aplicaciones móviles	12
2.2.1.1. Aplicaciones nativas	12

	2.2.1.2.	Aplicaciones web.....	13
	2.2.1.3.	Aplicaciones web nativas	13
	2.2.2.	Ventajas y desventajas de los tipos de aplicaciones móviles	13
	2.2.3.	Tendencias de las aplicaciones móviles.....	14
	2.2.4.	Tiendas de aplicaciones móviles.....	15
2.3.		Mercado objetivo	16
2.4.		<i>Benchmark</i> de la aplicación	18
3.		DISEÑO DE LA APLICACIÓN	23
3.1.		Interfaz de usuario.....	23
	3.1.1.	Inicio.....	23
		3.1.1.1. Componentes	24
	3.1.2.	Mi Horario.....	24
		3.1.2.1. Componentes	25
		3.1.2.2. Validaciones	26
	3.1.3.	Agregar a calendario	26
	3.1.4.	Horario de cursos	29
		3.1.4.1. Componentes	29
	3.1.5.	Pantalla de búsqueda de cursos	30
		3.1.5.1. Componentes	31
		3.1.5.2. Validaciones	31
	3.1.6.	Detalle curso	31
		3.1.6.1. Componentes	34
		3.1.6.2. Validaciones	34
3.2.		Usabilidad	35
	3.2.1.	Zonas pulsables	35
	3.2.2.	Uso de la pantalla.....	35
	3.2.3.	Navegación simple e intuitiva	36

3.2.4.	Encabezado.....	36
3.2.5.	Menú de navegación directa.....	37
3.2.6.	Botón de inicio.....	37
3.2.7.	Opción hacia atrás.....	38
3.2.8.	Botón de salir.....	38
4.	DOCUMENTACIÓN Y TUTORIAL DE PROGRAMACIÓN.....	39
4.1.	Herramientas.....	39
4.1.1.	PhoneGap.....	39
4.1.2.	Eclipse.....	40
4.1.3.	AndroidSDK.....	40
4.1.4.	Glassfish.....	40
4.1.5.	Java.....	41
4.2.	Hardware.....	41
4.2.1.	Servidor <i>backend</i>	42
4.2.2.	Dispositivos móviles.....	42
4.3.	Software.....	42
4.3.1.	jQuery Mobile.....	42
4.3.2.	SQLite.....	43
4.3.3.	HTML5.....	43
4.3.4.	JavaScript.....	44
4.3.5.	<i>Plugins</i>	44
4.3.5.1.	<i>Calendar</i>	45
4.3.5.2.	<i>Toast</i>	46
4.3.6.	Servicios REST.....	46
4.3.7.	JSON.....	48
4.3.8.	HTMLParser.....	48
4.4.	Tutorial de desarrollo y referencias.....	49
4.4.1.	Configuración de PhoneGap en Windows.....	49

4.4.2.	Configuración de AndroidSDK.....	52
4.4.3.	Parseo de horario.....	58
4.4.4.	Generación de servicio REST	60
4.4.4.1.	Listado de cursos	60
4.4.4.2.	Detalle de cursos.....	61
4.4.5.	Configuración de jQuery Mobile	62
4.4.6.	Creación de aplicación	63
4.4.6.1.	Instalación de NodeJS en Windows	63
4.4.6.2.	Creación de proyecto MiHorarioFIUSAC.....	65
4.4.7.	Desarrollo de páginas de la aplicación.....	67
4.4.8.	Base de datos local	67
4.4.8.1.	Abrir base de datos local	68
4.4.8.2.	Creación de la tabla	68
4.4.8.3.	Seleccionar registros de la tabla	70
4.4.8.4.	Borrar los registros de la tabla.....	71
4.4.8.5.	Verificación de existencia de registro ...	72
4.4.9.	Consumo de servicios REST.....	73
4.4.10.	Creación de un evento en calendario	75
4.5.	Consideraciones de implementación.....	77
4.5.1.	Generación de APK.....	77
4.5.1.1.	Almacén de claves	78
4.5.1.2.	Generar versión productiva	78
4.5.1.3.	Firma digital.....	79
4.5.1.4.	Empaquetado y compresión.....	79
4.5.2.	Publicación de la aplicación	80
4.5.2.1.	Cuenta de programador de Google	80
4.5.2.2.	Consola de programador de Google	80
4.5.2.3.	Subir APK.....	81

4.5.2.4.	Llenar la ficha de Play Store	83
4.5.2.5.	Ver en Play Store.....	86
4.5.3.	Publicación de servicios	87
4.5.3.1.	Generación de WAR.....	87
4.5.3.2.	Publicación en Glassfish.....	88
CONCLUSIONES		91
RECOMENDACIONES.....		93
BIBLIOGRAFÍA.....		95
APÉNDICE.....		97

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Modelo TAM	1
2.	Gráfica: tabla I	5
3.	Gráfica: tabla II	6
4.	Gráfica: tabla III	7
5.	Gráfica: tabla IV	8
6.	Gráfica: tabla V	9
7.	Gráfica: distribución de respuestas diarias	10
8.	Suscripciones por dispositivo	15
9.	Número total de aplicaciones por tienda	16
10.	Pantalla de Horario de Clases Plus	19
11.	Pantalla de Horario de Clase	20
12.	Pantalla Horario	21
13.	Pantalla MyClass Schedule	22
14.	Pantalla de inicio	23
15.	Pantalla de Mi Horario	25
16.	Mensaje de confirmación	26
17.	Agregar a calendario	27
18.	Creación de evento: título y fecha	28
19.	Creación de evento: lugar y detalles	28
20.	Pantalla horario de cursos	29
21.	Pantalla búsqueda de cursos	30
22.	Pantalla detalle curso	32
23.	Pantalla detalle examen	33

24.	Mensaje de agregado correctamente.....	34
25.	Mensaje de curso existente en Mi Horario	35
26.	Encabezado de la aplicación.....	36
27.	Menú de navegación directa	37
28.	Opciones del menú de navegación	37
29.	Botón de inicio.....	38
30.	Botón salir	38
31.	Fragmento de código: creación de evento	45
32.	Fragmento de código: mensaje <i>toast</i>	46
33.	Ruta de construcción	49
34.	Pantalla inicio de instalación de PhoneGap	50
35.	Pantalla de aceptación de licencia de PhoneGap	50
36.	Pantalla de selección de ruta de instalación de PhoneGap	51
37.	Pantalla de selección de menú de PhoneGap	51
38.	Pantalla de verificación de instalación de PhoneGap	52
39.	Términos y condiciones para instalar Android SDK	53
40.	Pantalla de inicio de instalación de Android SDK.....	54
41.	Pantalla de selección de carpeta de instalación de Android SDK.....	54
42.	Pantalla de menú de inicio de Android SDK.....	55
43.	Pantalla de finalización de instalación de Android SDK	55
44.	Pantalla SDK Manager.....	56
45.	Crear nuevo dispositivo Android	57
46.	Características de emulador de dispositivo Android	57
47.	Emulador de dispositivo Android.....	58
48.	Fragmento de código: estructura de horarios.....	59
49.	Fragmento de código: HTMLParser	60
50.	Fragmento de código: servicio de horarios	61
51.	Fragmento de código: servicio de detalle de horario.....	62
52.	Fragmento de código: inclusión de jQuery Mobile	62

53.	Pantalla de bienvenida de instalación de NodeJS.....	63
54.	Pantalla de selección de componentes de NodeJS.....	64
55.	Pantalla de aceptación de instalación de NodeJS.....	65
56.	Creación de proyecto con NodeJS	66
57.	Estructura del proyecto MiHorarioFIUSAC	66
58.	Estructura del <i>body</i> de páginas	67
59.	Código de apertura de base de datos	68
60.	Tabla dbMiHorario	70
61.	Código de la función <i>selectAll</i>	71
62.	Código para borrar un solo registro	72
63.	Código para eliminar todos los registros	72
64.	Código para verificar existencia de un registro.....	73
65.	Fragmento de código: consumo de servicio	74
66.	Fragmento de código: respuesta servicio horarios	74
67.	Fragmento de código: respuesta detalle horario	75
68.	Fragmento de código: función <i>addToCalendar</i>	77
69.	Consola de programador de Google	81
70.	Subir APK: versiones.....	81
71.	Subir APK: selección de archivo.....	82
72.	Ficha: información del producto.....	83
73.	Ficha: elementos gráficos – pantallas	84
74.	Ficha: elementos gráficos – icono	84
75.	Ficha: categorización.....	85
76.	Ficha: datos de contacto	85
77.	Ficha: política de privacidad	85
78.	Ver en Play Store	86
79.	Generación de WAR – exportar.....	87
80.	Generación de WAR – detalles	88
81.	Publicación en Glassfish – aplicaciones.....	89

82.	Publicación en Glassfish – selección de WAR	90
83.	Publicación en Glassfish – raíz de contexto.....	90

TABLAS

I.	Respuestas: ¿Has navegado por el sitio de la Facultad de Ingeniería de la Universidad de San Carlos desde un dispositivo móvil?.....	5
II.	Respuestas: ¿A qué escuela de la Facultad de Ingeniería perteneces?.....	6
III.	Respuestas: ¿Instalarías en tu dispositivo móvil una aplicación que te permita consultar el horario de cursos y personalizar tu horario?	7
IV.	Respuestas: ¿Qué dispositivo móvil utilizas para navegar por internet?.....	8
V.	Respuestas: ¿Qué sistema operativo tiene el dispositivo móvil seleccionado en la pregunta anterior?	9
VI.	Ventajas y desventajas de los tipos de aplicaciones móviles.....	13
VII.	Numeración de los periodos.....	69

LISTA DE SÍMBOLOS

Símbolo	Significado
DP	<i>Density pixel</i>
MB	Megabytes

GLOSARIO

ACID	Atomicity consistency isolation durability.
AJAX	Es una técnica de desarrollo web para aplicaciones interactivas.
APK	Application package.
Aplicación móvil	Es un programa que se puede descargar y acceder directamente desde un dispositivo móvil.
Base de datos	Son una colección de datos organizados y estructurados.
CSS3	Cascading style sheets 3 version.
CSV	Comma-separated value.
Dispositivo móvil	Son aparatos de pequeñas dimensiones con conexiones permanentes o limitadas a internet.
Eclipse	Es un entorno de desarrollo integrado de código abierto.
EDGE	Enhanced data rates for global evolution.

Framework	Conjunto estandarizado de conceptos, prácticas y criterios para enfrentar un tipo de problemática.
GSM	Global system for mobile.
Hardware	Son todas las partes físicas de un sistema informático.
HSPA	High speed packet access.
HTML5	HyperText markup language 5 version
iOS	iPhone operating system.
JavaScript	Es un lenguaje de programación interpretado.
JQuery mobile	Es un <i>framework</i> optimizado para dispositivos táctiles.
JSON	JavaScript object notation.
PhoneGap	<i>Framework</i> para desarrollar aplicaciones móviles.
Play Store	Es la tienda de aplicaciones en línea desarrollada por Google.
Plugin	Es una aplicación que se relaciona con otra para darle una funcionalidad nueva.

REST	Representational state transfer.
SDK	Software development kit.
Sistema operativo	Es un conjunto de programas que permite la administración eficaz de un ordenador.
Software	Es el soporte lógico de un sistema informático.
TAM	Technology acceptance model.
WAP	Wireless application protocol.
WAR	Web application archive.

RESUMEN

La Facultad de Ingeniería de la Universidad de San Carlos de Guatemala cuenta con un sitio web donde cada semestre se publican los horarios de los cursos que se van a impartir y el calendario de exámenes de cada uno de estos.

Con la evolución de los dispositivos móviles y una de sus principales características que es siempre estar conectado a internet se han vuelto una herramienta de mucha utilidad en las diferentes actividades de la vida cotidiana, para dichos dispositivos se encuentran un sinnúmero de aplicaciones que permiten realizar una tarea específica que facilita las actividades diarias que se realizan.

Ya que el sitio de la Facultad de Ingeniería es uno de los más visitados a los principios de cada semestre y que no siempre se cuenta con un computador a la mano, se desarrolla una aplicación móvil para dispositivos Android que por medio de un *web service* accede al sitio oficial de Ingeniería, donde se publican los horarios y los muestra de una manera amigable en la pantalla del dispositivo móvil. Además es capaz de guardar los cursos que el estudiante llevará durante el semestre y de calendarizar los exámenes finales de dicho periodo.

OBJETIVOS

General

Desarrollar e implementar una versión del horario de la Facultad de Ingeniería que sea de fácil uso en un dispositivo móvil.

Específicos

1. Facilitar la consulta del horario de la Facultad de Ingeniería desde dispositivos móviles con sistema operativo Android.
2. Permitir el almacenamiento de los cursos seleccionados en el dispositivo móvil para que estos estén disponibles para posteriores consultas.
3. Proveer la misma información que se obtiene en el sitio web actual de la Facultad de Ingeniería.

INTRODUCCIÓN

Con el avance de la tecnología, en el mercado cada día se ven más dispositivos móviles con funcionalidades que hace algunos años estaban limitadas a los computadores. Este creciente mercado de dispositivos móviles viene acompañado con el desarrollo de aplicaciones móviles.

Las aplicaciones móviles permiten realizar varias cotidianas sin necesidad de tener un computador a la mano, como: administrar agendas, envío y recepción de correos, acceso a internet, juegos, entre otras. Esto es de gran utilidad, ya que permite que la persona esté conectada en cualquier lugar y en cualquier momento.

Al finalizar la implementación de la aplicación móvil, el estudiante podrá acceder, desde cualquier dispositivo móvil, de una manera fácil y amigable, al horario del semestre en curso, teniendo también la opción de guardar en el dispositivo los cursos que el estudiante seleccione y consultarlos en cualquier momento.

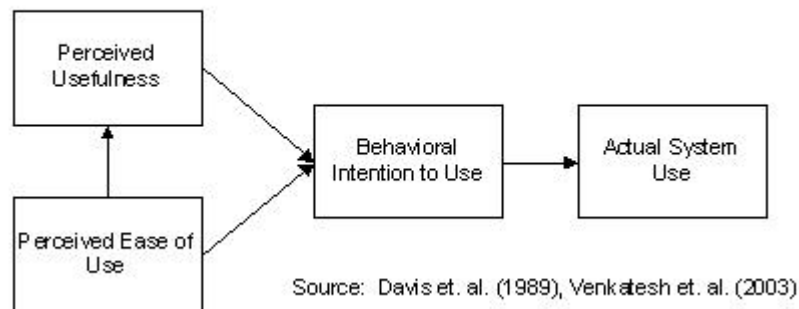
1. ESTUDIO DE LA TECNOLOGÍA Y SU IMPACTO EN GUATEMALA

1.1. Modelo de investigación

Se utilizará el modelo TAM (*technology acceptance model*) para evaluar la intención de uso y el uso real del sistema, estas variables dependen directamente de la utilidad y la facilidad de uso percibidas de la aplicación.

Figura 1. Modelo TAM

Diagram/schematic of theory



Fuente: INNOVAR. *Revista de Ciencias Administrativas y Sociales*. p. 192.

El modelo TAM postula que la utilidad y la facilidad de uso percibida determinan la intención de uso de un individuo, sirviendo esta como mediador del uso real del sistema.

1.1.1. Utilidad percibida

Determina qué tan útil le ha resultado la aplicación al usuario, esto se verá afectado según la misma cumpla con el objetivo y el alcance ofrecidos. El término “percibida” implica que aun cuando la aplicación cumpla con el alcance, determinados usuarios pueden clasificarla como no útil de acuerdo a lo que ellos perciben como útil.

1.1.2. Facilidad de uso percibida

Determina qué tan fácil le ha resultado al usuario interactuar con la aplicación. Esto se verá afectado según el diseño, que debe estar orientado al usuario, conociéndolo a detalle para asegurar que el producto satisfaga las necesidades específicas por las que fue creado. Es de gran importancia realizar varios conjuntos de pruebas con un determinado grupo de usuarios.

1.1.3. Intención de uso

Determina si un usuario está dispuesto a utilizar por primera vez, o bien, continuar utilizando la aplicación. Esto se verá afectado según la relación entre la facilidad de uso y utilidad percibida.

1.1.4. Uso real del sistema

Determina el uso real de la aplicación, esta variable no se ve afectada por la percepción del usuario, ya que puede ser medida objetivamente considerando datos como la cantidad de nuevos usuarios.

1.2. Relación con la tecnología

Utilizando los datos proporcionados por la consola de desarrolladores de Google, el sistema de valoración de Play Store y una encuesta virtual, es posible determinar el valor de variables clave como la utilidad y facilidades de uso percibidas, la intención de uso y el uso real de la aplicación.

1.2.1. Utilidad y facilidad de uso percibida

Es posible evaluar la utilidad y facilidad de uso percibida de la aplicación analizando los datos obtenidos del sistema de *rating* y comentarios del Play Store de Android. Al publicar una aplicación en Play Store, el usuario tiene la posibilidad de evaluar la aplicación en un rango de 0 a 5 estrellas y, opcionalmente, dejar un comentario de la razón de la valoración.

Es posible navegar dentro de los comentarios de la aplicación y ver los siguientes datos: valoración de estrellas, usuario que hizo el comentario, marca de tiempo, dispositivo y versión de la aplicación. Al navegar dentro de los comentarios es posible aplicar filtros por dispositivo o versión.

Google también permite descargar informes diarios en formato CSV desde Google Cloud Storage.

1.2.2. Intención de uso

La intención de uso se obtuvo utilizando una encuesta virtual, a continuación se describen las preguntas realizadas y los resultados obtenidos.

1.2.2.1. Encuesta

Para evaluar la intención de uso de la aplicación, se realizó una encuesta electrónica, la cual fue enviada por correo electrónico y compartida en redes sociales, obteniendo un total de 257 respuestas como muestra de la población estudiantil. La encuesta se hizo utilizando la herramienta Google Forms para una fácil distribución y procesamiento de resultados.

La encuesta contenía las siguientes preguntas:

- ¿Has navegado por el sitio de la Facultad de Ingeniería de la Universidad de San Carlos desde un dispositivo móvil?
- ¿A qué escuela de la Facultad de Ingeniería perteneces?
- ¿Instalarías en tu dispositivo móvil una aplicación que te permita consultar el horario de cursos y personalizar tu horario?
- ¿Qué dispositivo móvil utilizas para navegar por internet?
- ¿Qué sistema operativo tiene el dispositivo móvil seleccionado en la pregunta anterior?

1.2.2.2. Resultados

Como se puede observar en la tabla I, el 81,3 % de la muestra de estudiantes encuestada ha utilizado un dispositivo móvil para navegar por el sitio consultado. Esto representa el mercado objetivo de la aplicación, ya que

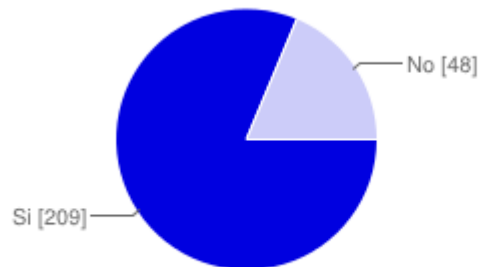
sería a estos estudiantes a los que se beneficiará con una forma adicional para consultar sus horarios, en una aplicación diseñada para su dispositivo.

Tabla I. **Respuestas: ¿Has navegado por el sitio de la Facultad de Ingeniería de la Universidad de San Carlos desde un dispositivo móvil?**

Respuesta	Cantidad	Porcentaje
Si	209	81,3
No	48	18,7

Fuente: elaboración propia.

Figura 2. **Gráfica: tabla I**



Fuente: elaboración propia.

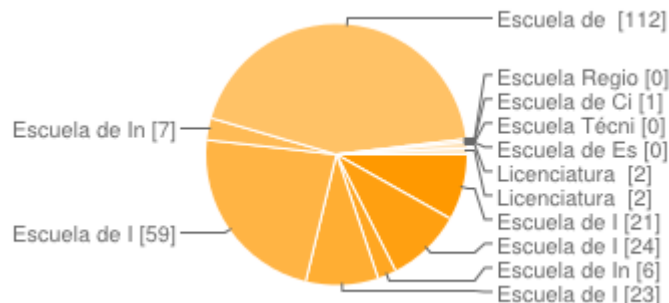
En la tabla II se ve la distribución de los estudiantes encuestados de acuerdo a la escuela a la que pertenecen.

Tabla II. **Respuestas: ¿A qué escuela de la Facultad de Ingeniería perteneces?**

Respuesta	Cantidad	Porcentaje
Ingeniería Mecánica	6	2,3
Ingeniería Mecánica Eléctrica	23	8,9
Ingeniería Mecánica Industrial	59	23
Ingeniería en Industrias Agropecuarias y Forestales	7	2,7
Ingeniería en Ciencias y Sistemas	112	43,6
Regional de Ingeniería Sanitaria y Recursos Hidráulicos – ERIS	0	0
Ciencias	1	0,4
Técnica	0	0
Estudios de Postgrado	0	0
Licenciatura en Física Aplicada	2	0,8
Licenciatura en Matemáticas	2	0,8
Ingeniería Mecánica	6	2,3
Ingeniería Mecánica Eléctrica	23	8,9

Fuente: elaboración propia.

Figura 3. **Gráfica: tabla II**



Fuente: elaboración propia.

En la tabla III se presenta la pregunta clave de la encuesta, ya que con esta se puede determinar la intención de uso de la aplicación. Se obtuvo un 93,4 % de estudiantes dispuestos a instalar la aplicación por primera vez, lo

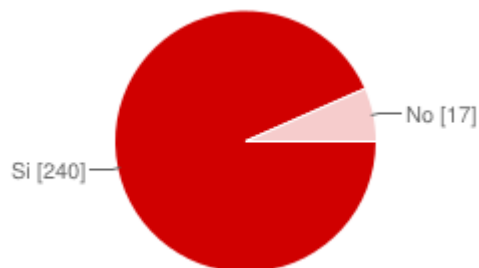
cual en combinación con la demás variables del modelo, conducirá a la aceptación final de la aplicación.

Tabla III. **Respuestas: ¿Instalarías en tu dispositivo móvil una aplicación que te permita consultar el horario de cursos y personalizar tu horario?**

Respuesta	Cantidad	Porcentaje
Si	240	93,4
No	17	6,6

Fuente: elaboración propia.

Figura 4. **Gráfica: tabla III**



Fuente: elaboración propia.

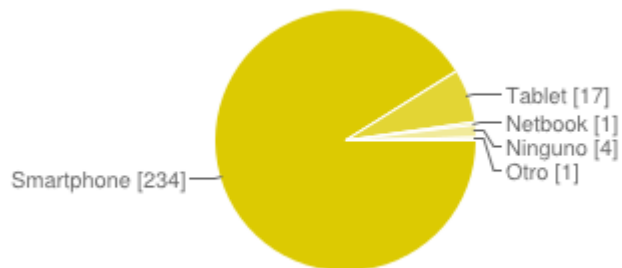
En la tabla IV se observa que la mayoría de la población encuestada utiliza el dispositivo *smartphone* como medio de navegación. Por lo tanto, es correcto el enfoque de diseño a los dispositivos más pequeños y con posibilidad de expansión hacia los dispositivos de mayor resolución.

Tabla IV. **Respuestas: ¿Qué dispositivo móvil utilizas para navegar por internet?**

Respuesta	Cantidad	Porcentaje
<i>Smartphone</i>	234	91,1
<i>Tablet</i>	17	6,6
<i>Netbook</i>	1	0,4
Ninguno	4	1,6
Otro	1	0,4

Fuente: elaboración propia.

Figura 5. **Gráfica: tabla IV**



Fuente: elaboración propia.

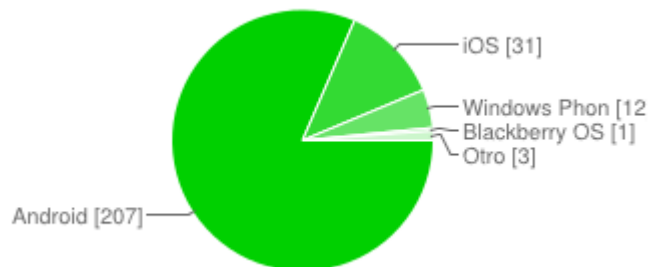
En la tabla V se observa que la mayoría de la población encuestada utiliza el sistema operativo Android. Esto indica que el alcance de la aplicación, desplegada únicamente para este sistema operativo, es correcto y podrá ser utilizado por la mayor cantidad de estudiantes.

Tabla V. **Respuestas: ¿Qué sistema operativo tiene el dispositivo móvil seleccionado en la pregunta anterior?**

Respuesta	Cantidad	Porcentaje
Android	207	80,5
iOS	31	12,1
Windows Phone	12	4,7
Blackberry OS	1	0,4
Otro	3	1,2

Fuente: elaboración propia.

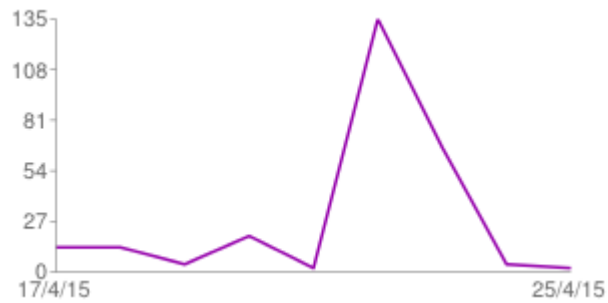
Figura 6. **Gráfica: tabla V**



Fuente: elaboración propia.

Las respuestas fueron recibidas entre el 17 y el 25 de abril de 2015, obteniendo la mayor cantidad de estas al ser publicada la encuesta en las redes sociales de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala.

Figura 7. **Gráfica: distribución de respuestas diarias**



Fuente: elaboración propia.

1.2.3. **Uso real del sistema**

Para determinar el uso real del sistema en complemento a las estadísticas de descarga e instalación provistas por el Play Store, se implementarán bitácoras internas en la aplicación que permitan almacenar la cantidad de búsquedas agrupadas por curso y período.

2. IDENTIFICACIÓN DEL PROBLEMA Y SOLUCIÓN

2.1. Antecedentes

El gran avance que han tenido las aplicaciones móviles desde su aparición en los años 90 se debe gracias a la constante innovación de la tecnología WAP (*wireless application protocol*) y la transmisión de datos (EDGE). Con estas tecnologías ha crecido el desarrollo de aplicaciones móviles, que son programas diseñados para ser ejecutados en un teléfono inteligente, tabletas y otros dispositivos móviles.

Las aplicaciones móviles permiten realizar varias actividades cotidianas sin necesidad de tener un computador a la mano, como: administrar agendas, envío y recepción de correos, acceso a internet, juegos, entre otras. Esto es de gran utilidad, ya que permite que las personas estén conectadas en cualquier lugar y en cualquier momento.

2.2. Aplicaciones móviles

Las aplicaciones móviles son programas informáticos diseñados para ser ejecutados en dispositivos móviles. Estas aplicaciones deben tener en consideración las limitaciones que tienen los dispositivos, como lo son la batería, procesador, memoria y los distintos tamaños de pantallas que son menos poderosos que los de un computador.

2.2.1. Tipos de aplicaciones móviles

No todas las aplicaciones móviles tienen las mismas características, ni son del mismo tipo, por lo que, para elegir en qué tipo de aplicación se quiere desarrollar, es necesario tomar en cuenta lo siguiente:

- Tipo de información a brindar
- Forma en que se va a desarrollar
- Uso de los recursos del dispositivo móvil
- Sistema operativo del dispositivo
- Tiempo con que se cuenta
- Dinero con que se cuenta

Para cubrir estas características existen tres tipos de aplicaciones móviles, las nativas, las web y las híbridas.

2.2.1.1. Aplicaciones nativas

Es una aplicación creada para funcionar en un sistema operativo específico. Si se quiere que una aplicación móvil pueda ser utilizada en los diferentes sistemas operativos, se debe compilar en cada lenguaje del sistema operativo donde se quiera que esté disponible. Estas aplicaciones permiten que el funcionamiento sea fluido y estable para el sistema operativo en el que fue creada.

2.2.1.2. Aplicaciones web

Son aplicaciones que pueden ser programadas en lenguajes como HTML, JavaScript, CSS, entre otras. Estas aplicaciones pueden funcionar en cualquier dispositivo sin importar el sistema operativo, ya que se accede desde el navegador.

2.2.1.3. Aplicaciones web nativas

Es una mezcla de las dos anteriores y son aplicaciones programadas en lenguajes conocidos como los anteriores, por lo que funcionan en cualquier sistema operativo y permiten utilizar al máximo el hardware del dispositivo móvil.

2.2.2. Ventajas y desventajas de los tipos de aplicaciones móviles

A continuación se realiza una tabla comparativa de las ventajas y desventajas de los tipos de aplicaciones móviles.

Tabla VI. **Ventajas y desventajas de los tipos de aplicaciones móviles**

Tipo de aplicación móvil	Ventajas	Desventajas
Aplicación nativa	<ul style="list-style-type: none">- Acceso completo al dispositivo.- Puede ser descargada desde una tienda de aplicaciones.- Envío de notificaciones.	<ul style="list-style-type: none">- El desarrollo es más caro.- No se puede reutilizar el código.

Continuación de la tabla VI.

Aplicación web	<ul style="list-style-type: none"> - El código es reutilizable. - Desarrollo más económico. - No necesita aprobación externa para publicarse. 	<ul style="list-style-type: none"> - Requiere de conexión a internet. - No se utiliza al máximo el hardware del dispositivo. - Menor tiempo de respuesta de la aplicación.
Aplicación web nativa	<ul style="list-style-type: none"> - Se puede distribuir en tiendas de aplicaciones. - El mismo código se puede utilizar en las diferentes plataformas. - Se puede acceder al hardware del dispositivo. 	<ul style="list-style-type: none"> - Mejor experiencia de usuario. - El diseño de la aplicación no se relaciona con el del sistema operativo.

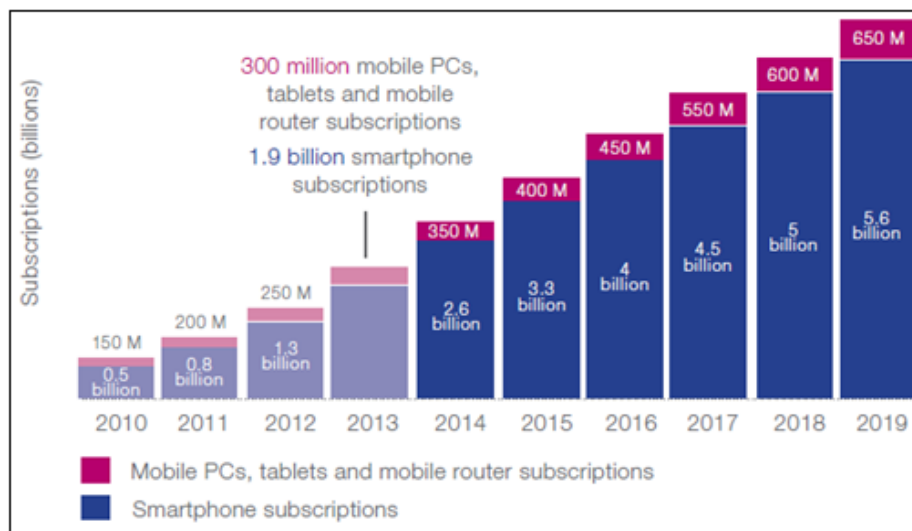
Fuente: elaboración propia.

2.2.3. Tendencias de las aplicaciones móviles

Actualmente las aplicaciones móviles están ganando más terreno, ya que cada vez más personas cuentan con un teléfono inteligente o una tableta, simplemente por el hecho de que un dispositivo móvil cuenta con internet las 24 horas del día, los 7 días de la semana. Esto permite que las personas estén conectadas y comunicadas todo el tiempo. Mientras este número de dispositivos siga subiendo, el número de aplicaciones móviles será cada vez más.

Según GSMA, en el 2020, la mitad de la población mundial se conectará a internet desde el móvil. Este incremento se deberá, en gran parte a los países en desarrollo, ya que se pasarán de 1,5 billones de usuarios en 2013, a 3 billones en 2020; lo que significa que se duplicará el número de personas que contarán con un dispositivo móvil y lo utilizarán para la transmisión de datos.

Figura 8. Suscripciones por dispositivo



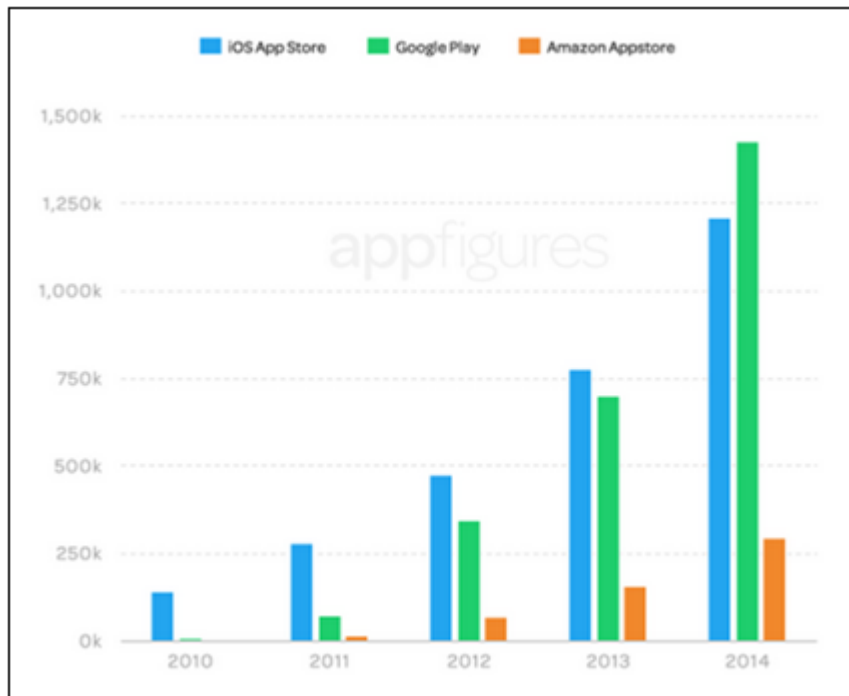
Fuente: RODRÍGUEZ, Paco. *XatakaMovil*. ww.xatakamovil.com. Consulta: 13 de abril de 2015.

2.2.4. Tiendas de aplicaciones móviles

Las aplicaciones móviles pueden encontrarse gratis o de pago en las distintas plataformas de distribución. Entre estas plataformas están la AppStore de Apple, Play Store de Android y BlackBerry App World de BlackBerry, por mencionar algunas. En las plataformas se pueden encontrar un sinnúmero de aplicaciones que se ajusten a las necesidades del cliente, para tenerlas disponibles en cualquier momento.

En la figura 9 se observa cómo, en los últimos años, ha crecido, de manera realmente rápida, el número de aplicaciones disponibles en las principales tiendas de aplicaciones móviles, ya que, en tan solo 4 años, se ha logrado multiplicar por más de 10 el número de las aplicaciones, siendo Google Play la que ha crecido más significativamente.

Figura 9. **Número total de aplicaciones por tienda**



Fuente: RANGER, Steve. *ZDNet*. <http://www.zdnet.com/>. Consulta: 6 de marzo de 2015.

2.3. **Mercado objetivo**

La Facultad de Ingeniería de la Universidad de San Carlos de Guatemala es una de las 10 facultades con más población estudiantil, alberga alrededor de 12 000 estudiantes cada semestre, cuenta con 12 programas de pregrado. Por tal razón, se tomó la decisión de crear una aplicación móvil para ver los horarios de una manera fácil y amigable, desde cualquier dispositivo móvil. De esta manera, los estudiantes podrán consultar y seleccionar los cursos que van a llevar durante el semestre y así poder crear su horario, el cual quedará guardado en el dispositivo móvil para ser consultado en cualquier momento.

Desde la aplicación se podrán consultar los horarios de las distintas escuelas de la Facultad de Ingeniería, las cuales son:

- Escuela de Civil
 - Ingeniería Civil

- Escuela de Mecánica
 - Ingeniería Mecánica

- Escuela de Mecánica Industrial
 - Ingeniería Industrial
 - Ingeniería Mecánica Industrial

- Escuela de Mecánica Eléctrica
 - Ingeniería Eléctrica
 - Ingeniería Electrónica
 - Ingeniería Mecánica Eléctrica

- Escuela de Química
 - Ingeniería Química
 - Ingeniería Ambiental

- Escuela de Ciencias y Sistemas

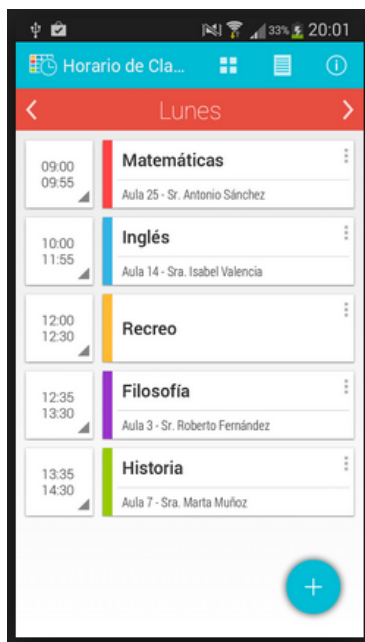
- Ingeniería en Ciencias y Sistemas
- Escuela de Ciencias – Departamento de Matemática
 - Licenciatura en Matemática Aplicada
- Escuela de Ciencias – Departamento de Física
 - Licenciatura en Física Aplicada

2.4. Benchmark de la aplicación

Para Android existen diferentes aplicaciones para agendar los horarios, sin embargo, no se ha encontrado alguna aplicación móvil que brinde el horario de cursos de alguna universidad o colegio. A continuación se nombran algunas de las aplicaciones que existen en la tienda de Android para agendar horarios:

- Horario de Clases Plus: tiene un peso de 2,7 MB y requiere un sistema operativo Android 4.0+, esta aplicación permite gestionar y administrar las actividades escolares y una agenda al estudiante.
 - Características
 - Diario en cada asignatura
 - Diario general
 - Poder incluir en Google Calendar exámenes y trabajos
 - Creación de avisos visuales por curso
 - Control de notas por curso

Figura 10. **Pantalla de Horario de Clases Plus**



Fuente: *Labeldroid*. <https://play.google.com>. Consulta: 13 de abril de 2015.

- Horario de clase: tiene un peso de 2,3 MB y requiere un sistema operativo Android 3.0+, es una aplicación para la creación de horarios simples y cómodos, con una mayor visibilidad del horario gracias a su interface intuitiva.
 - Características
 - Verificación de las horas de clase
 - Notas sobre clases
 - Horarios de exámenes y tareas

Figura 11. **Pantalla de Horario de Clase**

	lun	mar	mié	jue	vie
09:00					
1	Historia H123	Biología G220	Economía E401	Matemáticas G107	Inglés E225
09:50					
10:00					
2					
10:50					
3	Matemáticas G105	Inglés E223		Historia H125	Economía E401
11:00					
11:50					
12:00					
4					
12:50					
13:00					
5					

Materia Matemáticas
Prof. Antonio
Aula G107

Fuente: *NewBi*. <http://play.google.com>. Consulta: 13 de abril de 2015.

- Horario: tiene un peso de 2,8 MB y requiere un sistema operativo Android 4.0+, es una aplicación para manejar de manera fácil los horarios de la escuela desde el celular o *tablet*.
 - Características
 - Manejo de grados
 - Manejo de cursos y exámenes
 - Se pueden adjuntar imágenes por curso o examen
 - Calendarizar exámenes y tareas

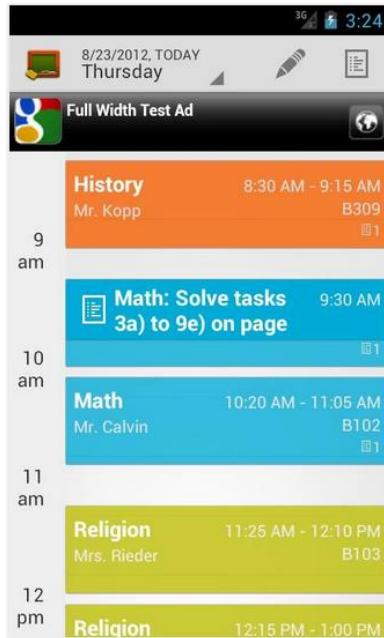
Figura 12. **Pantalla Horario**



Fuente: *Moritz/seke*. <https://play.google.com>. Consulta: 13 de abril de 2015.

- MyClass schedule: tiene un peso de 2,8 MB y requiere un sistema operativo Android 4.0+, aplicación que ayuda a mantener un horario organizado, con colores personalizados por clase y alertas de exámenes y tareas no terminadas.
 - Características
 - Horarios con colores personalizables
 - Manejo de cursos y tareas
 - Horarios con diferente frecuencia de las clases
 - Guarda las notas de los cursos

Figura 13. **Pantalla MyClass Schedule**



Fuente: *Sebastian Mayer*. <http://play.google.com>. Consulta: 13 de abril de 2015.

3. DISEÑO DE LA APLICACIÓN

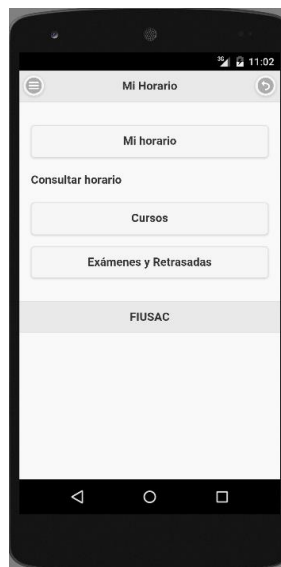
3.1. Interfaz de usuario

El acceso a las pantallas donde se muestra la información se ha realizado con no más de dos pasos intermedios.

3.1.1. Inicio

Esta es la pantalla mostrada al recién iniciar la aplicación, o bien, al presionar el botón de inicio desde cualquiera de las otras pantallas de la aplicación.

Figura 14. Pantalla de inicio



Fuente: elaboración propia.

3.1.1.1. Componentes

- Botón Mi Horario
Acceso a la pantalla de Mi Horario
- Botón de cursos
Acceso a la pantalla de los horarios de cursos
- Botón de exámenes y retrasadas
Acceso a la pantalla de los horarios de exámenes

3.1.2. Mi Horario

A estas pantallas se accede desde el menú principal, pulsando solamente un botón.

Aquí se muestran todos los cursos que han sido agregados a Mi Horario y se han almacenado en el dispositivo.

Figura 15. **Pantalla de Mi Horario**



Fuente: elaboración propia.

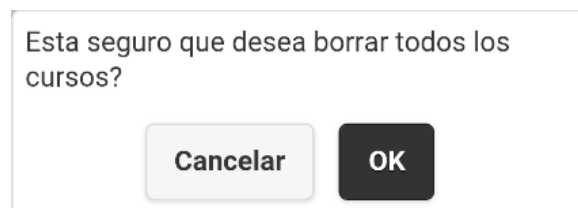
3.1.2.1. **Componentes**

- Lista cursos: despliega un listado de los cursos agrupados por periodo. Este listado contiene todos los datos del detalle del curso o examen para que, al momento de entrar a consultar Mi Horario, este contenga toda la información en una sola pantalla.
- Botón de borrar todos: permite borrar todos los cursos o exámenes almacenados en el dispositivo.
- Botón de borrar curso: permite borrar del dispositivo únicamente el curso o examen que se está seleccionando.

3.1.2.2. Validaciones

Al momento de seleccionar la opción de borrar todo, se muestra un mensaje de alerta para confirmar si realmente se desea borrar todos los cursos o exámenes que estén almacenados en el dispositivo.

Figura 16. **Mensaje de confirmación**



Fuente: elaboración propia.

3.1.3. Agregar a calendario

En la pantalla de Mi Horario, cuando se despliegan los elementos correspondientes a exámenes o retrasadas, se muestra el botón 'Agregar a calendario', este permite crear un evento en el calendario del dispositivo que corresponda al horario del examen seleccionado.

Figura 17. **Agregar a calendario**



Fuente: elaboración propia.

El evento se crea interactivamente, el estudiante tiene la posibilidad de modificar la información que considere pertinente, sin embargo, todos los datos relevantes del curso se muestran con valores por defecto.

Figura 18. **Creación de evento: título y fecha**

The screenshot shows a mobile application interface for creating an event. At the top, there are two tabs: "Añadir evento" (selected) and "Añadir tarea". Below the tabs are "Cancelar" and "Guardar" buttons. The main form has the following sections:

- Título:** A text input field containing "Exámen: REDES DE COMPUT," and a star icon to its right.
- De:** A date and time selector showing "Vie 16/01/2015" and "18:00".
- A:** A date and time selector showing "Vie 16/01/2015" and "18:00".
- Zona horaria:** A dropdown menu showing "(GMT-6:00) Hora estándar central".
- Todo el día:** A checkbox that is currently unchecked.
- Repetir:** A section with "Repetir" text, "Evento único" below it, and a blue right-pointing arrow icon.
- Recordatorio:** A section with "Recordatorio" text and a green plus icon.

Fuente: elaboración propia.

Figura 19. **Creación de evento: lugar y detalles**

The screenshot shows a mobile application interface for creating an event, continuing from the previous screen. It features the same top navigation and buttons. The main form has the following sections:

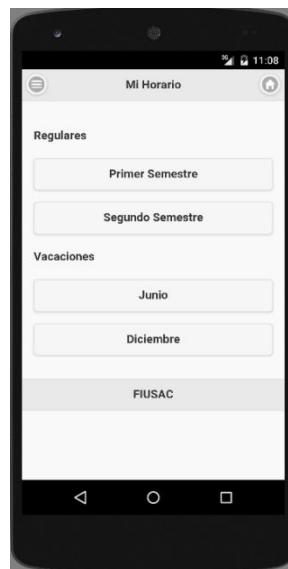
- Ubicación:** A text input field containing "T-3 305,310" and a location pin icon to its right.
- Descripción:** A text input field containing "Jornada: Vespertina".
- Imágenes:** A section with "Imágenes" text and a green plus icon.

Fuente: elaboración propia.

3.1.4. Horario de cursos

Para la consulta de horarios, se accede pulsando solamente dos botones. Primero se pasa por la opción de qué horario se desea consultar (cursos o exámenes y retrasadas), luego se selecciona el periodo que se quiere consultar (primer semestre, segundo semestre, junio o diciembre).

Figura 20. **Pantalla horario de cursos**



Fuente: elaboración propia.

3.1.4.1. Componentes

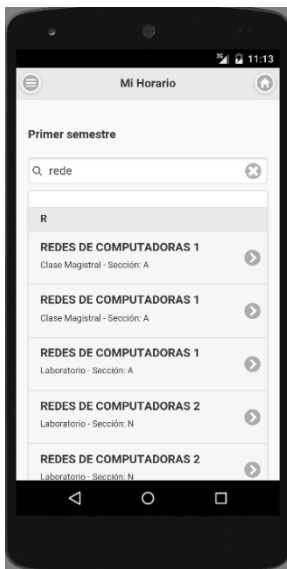
- Botón primer semestre: acceso a la consulta de cursos de primer semestre.
- Botón segundo semestre: acceso a la consulta de cursos de segundo semestre.

- Botón vacaciones junio: acceso a la consulta de cursos de vacaciones del mes de junio.
- Botón vacaciones diciembre: acceso a la consulta de cursos de vacaciones del mes de diciembre.

3.1.5. Pantalla de búsqueda de cursos

Esta pantalla permite consultar los horarios de los cursos para el periodo seleccionado en la pantalla anterior, los resultados mostrados son obtenidos directamente de la página oficial de la Facultad de Ingeniería.

Figura 21. **Pantalla búsqueda de cursos**



Fuente: elaboración propia.

3.1.5.1. Componentes

- Lista cursos: muestra el listado de cursos que coincidan con la palabra ingresada en el filtro.
- Filtro de búsqueda: campo de entrada de texto para escribir el nombre del curso que se desea consultar horario.
- Acceso a detalle del curso: cada elemento de la lista de cursos representa un acceso directo al detalle del curso seleccionado.

3.1.5.2. Validaciones

La búsqueda inicia hasta ingresar al menos 4 caracteres, esto permite descartar preposiciones y artículos que sobrecargarían la cantidad de resultados obtenidos, haciendo mal uso del ancho de banda del usuario final. Se descartan los espacios en blanco al inicio y al final del nombre de curso ingresado.

3.1.6. Detalle curso

A esta pantalla se accede al pulsar el curso que la búsqueda devolvió, según los filtros ingresados en el filtro de búsqueda.

La funcionalidad de la pantalla consiste en mostrar el detalle de la selección realizada en la pantalla anterior, dependiendo de cuál fue la búsqueda (curso o exámenes), devuelve ciertos valores en el detalle:

- Campos del detalle de curso
 - Código del curso
 - Nombre del curso
 - Sección
 - Salón
 - Hora
 - Días
 - Catedrático

Figura 22. **Pantalla detalle curso**



Fuente: elaboración propia.

- Campos del detalle exámenes y retrasadas
 - Código del curso
 - Nombre del curso
 - Jornada
 - Salón
 - Hora
 - Días

Figura 23. **Pantalla detalle examen**



Fuente: elaboración propia.

3.1.6.1. Componentes

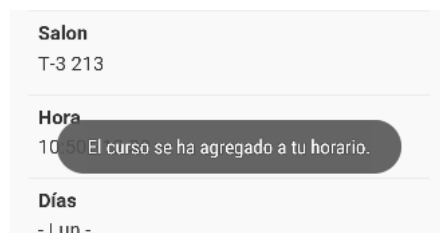
- Detalle curso: tabla que lista todo el detalle del curso o del examen.
- Agregar a Mi Horario: botón que permite agregar a Mi Horario el curso o examen que se está viendo.

3.1.6.2. Validaciones

Al presionar el botón de Agregar a Mi Horario se mostrará uno de los siguientes dos mensajes:

- El curso se ha agregado a Mi Horario, cuando este curso no exista en Mi Horario.

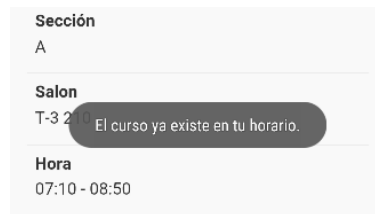
Figura 24. **Mensaje de agregado correctamente**



Fuente: elaboración propia.

- El curso ya existe en Mi Horario, cuando el curso ya existe en Mi Horario.

Figura 25. **Mensaje de curso existente en Mi Horario**



Fuente: elaboración propia.

3.2. Usabilidad

En el desarrollo de aplicaciones móviles, aparte de la funcionalidad y utilidad de las aplicaciones, existe otra característica importante como la usabilidad de la aplicación.

En esta aplicación se han utilizado las siguientes características para hacer la experiencia del usuario lo más cómoda posible.

3.2.1. Zonas pulsables

Tener zonas pulsables grandes es una de las claves más importantes en Android, Google recomienda que las zonas pulsables tengan un mínimo de 48 dp para que sean cómodas de pulsar.

3.2.2. Uso de la pantalla

Las pantallas de los dispositivos móviles tienen la limitación del tamaño, por lo tanto se debe de aprovechar al máximo dicha pantalla, es por eso que es

indispensable mostrar solo la información necesaria en cada momento, sin saturar de información.

3.2.3. Navegación simple e intuitiva

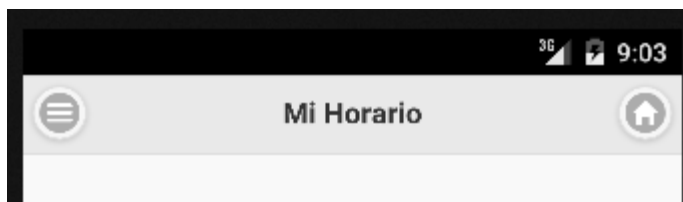
Los dispositivos móviles tienen menos características en cuanto a hardware se refiere, por lo tanto una aplicación móvil debe ser lo más simple y liviana para que funcione con rapidez.

En esta aplicación se ha minimizado la entrada de datos por parte del usuario.

3.2.4. Encabezado

En la parte superior de esta aplicación siempre van a estar disponibles los menús de navegación directa y el botón de inicio.

Figura 26. **Encabezado de la aplicación**



Fuente: elaboración propia.

3.2.5. Menú de navegación directa

Ubicado en la parte superior izquierda, desplegable lateral, permite saltar de pantalla en cualquier momento, sin necesidad de tener que ir al menú principal.

Figura 27. Menú de navegación directa



Fuente: elaboración propia.

Figura 28. Opciones del menú de navegación

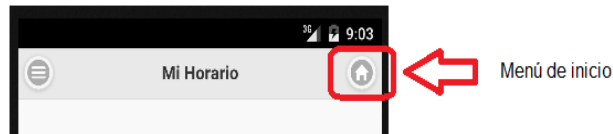


Fuente: elaboración propia.

3.2.6. Botón de inicio

Ubicado en la parte superior derecha, permite regresar al menú de inicio de manera rápida.

Figura 29. **Botón de inicio**



Fuente: elaboración propia.

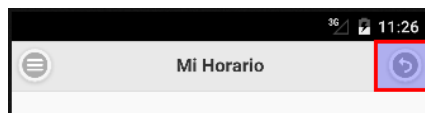
3.2.7. **Opción hacia atrás**

Todo dispositivo móvil Android posee el botón de regresar en la parte inferior. La aplicación diseñada es capaz de ir a la página anterior al pulsar el botón de atrás del dispositivo móvil, o bien, salir de la aplicación si se presiona dos veces.

3.2.8. **Botón de salir**

Al estar dentro de la pantalla de inicio, el botón de inicio se sustituye por el botón salir, el cual cerrará la aplicación.

Figura 30. **Botón salir**



Fuente: elaboración propia.

4. DOCUMENTACIÓN Y TUTORIAL DE PROGRAMACIÓN

4.1. Herramientas

Se utilizaron las siguientes herramientas durante el desarrollo e implementación de la aplicación.

4.1.1. PhoneGap

“Es un *framework* para el desarrollo de aplicaciones móviles, permite a los programadores desarrollar aplicaciones para dispositivos móviles utilizando herramientas genéricas. Dentro de las herramientas genéricas que utiliza PhoneGap podemos encontrar JavaScript, HTML5 y CSS3.”¹

Se escogió PhoneGap como herramienta para el desarrollo de la aplicación móvil por las siguientes razones:

- Es un *framework* multiplataforma, compatible con los principales sistemas operativos móviles por ejemplo iOS, Android, BlackBerry OS, entre otras. Con PhoneGap se desarrolla un solo código que puede ser empaquetado para que funcione en cada uno de los sistemas operativos que se deseen.
- Se programa con lenguajes de última generación, como pueden ser HTML5 y CSS3.

¹ PhoneGap. <http://es.wikipedia.org/wiki/PhoneGap>. Consulta: 23 de abril de 2015.

- Permite desarrollar, de manera rápida, aplicaciones sencillas que no necesiten muchos recursos.
- Una vez empaquetado puede ser subido a las diferentes tiendas disponibles.

4.1.2. Eclipse

Es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma.

Eclipse es un IDE muy completo y adaptable, permite configurar cada ambiente de desarrollo y adaptarlo a las necesidades, además, posee una gran variedad de *plugins* que pueden ser adaptados a los proyectos.

4.1.3. AndroidSDK

Es un kit de desarrollo que incluye las herramientas necesarias para programar aplicaciones que funcionen en un sistema operativo Android. Incluye un conjunto de API provistas por Google, para el control de las funciones del dispositivo móvil, la integración de servicios, depurar y un emulador para testear las aplicaciones.

4.1.4. Glassfish

“Es un servidor de aplicaciones que implementa tecnologías definidas en la plataforma Java EE y que permiten ejecutar código que siguen esta

especificación. Está basado en un código fuente donado por Sun y Oracle Corporation".²

4.1.5. Java

Es un lenguaje de programación orientado a objetos que fue diseñado para tener la menor cantidad de dependencias en la implementación. La principal finalidad es escribir solamente una vez el código y que este se pueda ejecutar en cualquier dispositivo. Por lo general, las aplicaciones Java son compiladas en *bytecode* (clase Java) que puede ser ejecutada en cualquier máquina virtual Java, sin importar la arquitectura de la computadora. Entre las principales características de Java están:

- Orientado a objetos
- Independiente de la plataforma
- Soporte para trabajo en red
- Fácil de usar

4.2. Hardware

El hardware requerido para el funcionamiento de la aplicación se limita al servidor *backend*, donde se encuentra publicado el servicio web que permite realizar la consulta de los diferentes horarios mostrados en la aplicación, y a los dispositivos móviles utilizados durante las pruebas de la misma.

² *Glassfish*. <http://es.wikipedia.org/wiki/GlassFish>. Consulta: 25 de abril de 2015.

4.2.1. Servidor *backend*

Servidor virtual con las siguientes características:

- Windows Server 2003
- 8GB de memoria RAM
- Procesador Intel Xeon

4.2.2. Dispositivos móviles

Las pruebas de la aplicación se realizaron en los siguientes dispositivos:

- Samsung Galaxy SIII Mini
- Samsung Galaxy S5 Mini
- Huawei P6
- Samsung Galaxy S5

4.3. Software

Se utilizaron los siguientes programas durante el desarrollo e implementación de la aplicación.

4.3.1. jQuery Mobile

La interfaz gráfica de usuario fue desarrollada utilizando el *framework* jQuery Mobile (jQM) versión 1.4.5. Este *framework* permite el desarrollo de páginas *responsive* que pueden ser empaquetadas como aplicaciones móviles utilizando PhoneGap.

Tiene optimizaciones para la captura de eventos de contacto (*touch*), navegación con transiciones por Ajax y gran variedad de componentes prefabricados, lo cual permite reducir el tiempo de desarrollo.

4.3.2. SQLite

Es una base de datos de código abierto que soporta bases de datos relacionales, está embebida en dentro de cada dispositivo Android e implementa la mayor parte de del estándar sql, incluyendo transacciones de bases de datos atómicas, consistencia de bases de datos, aislamiento y durabilidad (ACID, por sus siglas en ingles). Entre las principales razones para elegir SQLite se encuentran las siguientes:

- Tamaño: tiene poca memoria y usa una única librería para acceder a la base de datos.
- Rendimiento: realiza operaciones de manera eficiente y es más rápido que MySql y PostgreSQL por ejemplo.
- Estabilidad: es compatible con ACID.
- Costo: es de dominio público, lo que significa que puede ser usada para cualquier propósito de manera gratuita.

4.3.3. HTML5

Es la quinta revisión del lenguaje más básico de la web. Establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos, básicamente es un sistema para formatear el *layout* de las páginas. Con HTML5 los navegadores como Firefox, Chrome, Explorer, Safari, entre otros, pueden saber cómo mostrar una determinada página, saber dónde están

los elementos, dónde poner las imágenes, dónde ubicar el texto. Dentro de las principales novedades de HTML5 se pueden encontrar:

- Se puede reducir el uso de *plugins*
- Los usuarios pueden acceder a sitios web de manera *offline*
- Inserción de multimedia en los sitios web

4.3.4. JavaScript

“Es un lenguaje de programación interpretado, se define como orientado a objetos basado en prototipos, débilmente tipado y dinámico, este lenguaje se utiliza principalmente del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.”³ Una de las ventajas de JavaScript es que es soportado por todos los navegadores, dentro de las principales características, se encuentran:

- Es simple de utilizar
- Maneja objetos dentro de la página
- Responde a eventos en tiempo real

4.3.5. Plugins

Un *plugin* es un paquete de código inyectado que permite a las aplicaciones, construidas con PhoneGap, comunicarse con la plataforma nativa donde son ejecutadas.

³ *JavaScript*. <http://es.wikipedia.org/wiki/JavaScript>. Consulta: 26 de abril de 2015.

Los *plugins* deben ser instalados dentro de la carpeta del proyecto. Al ser instalados, sus dependencias y permisos son inyectados automáticamente al construir la aplicación.

Para facilitar la instalación de *plugins* se utilizó la herramienta Plugman, esta debe ser instalada con el siguiente comando: `npm install -g plugman`.

4.3.5.1. **Calendar**

Este *plugin* permite la interacción con el calendario del dispositivo donde se encuentre instalada la aplicación. Es utilizado para crear los eventos que corresponden a los exámenes cuando el estudiante elija esta opción. Se utiliza la opción de creación interactiva para permitir al estudiante modificar los datos que considere necesarios antes de guardar el evento en el calendario.

Comando de instalación:

Cordova plugin add nl.xservices.plugins.Calendar

Figura 31. **Fragmento de código: creación de evento**

```
var startDate = new Date(anio, mes, arrayDia[1], arrayHora[0],
    arrayHora[1], 0, 0, 0);
var endDate = new Date(anio, mes, arrayDia[1], parseInt(arrayHora[0]) + 1,
    arrayHora[1], 0, 0, 0);
var title = "Exámen: " + curso;
var eventLocation = salon;
var notes = "Jornada: " + jornada;
var success = function(message) {
    console.debug("Evento agregado con éxito: " + JSON.stringify(message));
};
var error = function(message) {
    console.debug("Error: " + message);
};

window.plugins.calendar.createEventInteractively(title, eventLocation,
    notes, startDate, endDate, success, error);
```

Fuente: elaboración propia.

4.3.5.2. **Toast**

Este *plugin* permite mostrar mensajes temporales de la plataforma nativa del dispositivo donde se encuentre instalada la aplicación. Estos mensajes son utilizados para presentar al estudiante información no invasiva, sin interferir con el resto del flujo de la aplicación.

Comando de instalación:

```
cordova plugin add nl.x-services.plugins.toast
```

Figura 32. **Fragmento de código: mensaje *toast***

```
function mostrarToast() {  
    var now = (new Date()).getTime();  
    if((now-lastToast) > 10000){  
        window.plugins.toast  
        .showShortCenter('No hay horarios publicados que coincidan con tu búsqueda.');
```

Fuente: elaboración propia.

4.3.6. **Servicios REST**

Por sus siglas en inglés, *representational state transfer*, son servicios web con una arquitectura cliente servidor, no poseen estado y utilizan métodos de invocación uniforme, esto permite obtener mejor rendimiento, escalabilidad y facilidad de mantenimiento.

Los servicios REST siguen los siguientes principios para mantenerse simples, ligeros y rápidos:

- Identificación de recursos por URI
- Interfaz uniforme
- Mensajes autodescriptivos
- Interacciones con estado a través de vínculos

En Java se pueden crear servicios REST utilizando la interfaz de programación JAX-RS, la cual provee útiles anotaciones para construir un servicio REST de forma rápida y práctica. A continuación se describen algunas de las anotaciones utilizadas para la creación de estos servicios.

- `@Path`: es un enlace relativo hacia la clase principal del servicio.
- `@GET`: permite asociar un método a la petición de HTTP GET.
- `@POST`: permite asociar un método a la petición de HTTP POST.
- `@PUT`: permite asociar un método a la petición de HTTP PUT.
- `@DELETE`: permite asociar un método a la petición de HTTP DELETE.
- `@HEAD`: permite asociar un método a la petición de HTTP HEAD.
- `@PathParam`: permite obtener parámetros definidos en el URI de la clase y los nombres deben coincidir con los especificados en la anotación `@Path`.
- `@QueryParam`: permite obtener parámetros desde URI, no es necesario especificar los nombre en `@Path`.
- `@Consumes`: especifica el tipo de datos MIME que el servicio recibe como entrada.
- `@Produces`: especifica el tipo de datos MIME que el servicio produce como salida.
- `@Provider`: se utiliza para especificar parámetros en tiempo de ejecución de JAX-RS.

4.3.7. JSON

“Es un formato ligero para el intercambio de datos, es un subconjunto de la notación literal de objetos JavaScript que no requiere el uso de XML, la simplicidad del formato JSON ha dado lugar a la generalización de uso ya que es mucho más fácil construir un analizador sintáctico.”⁴

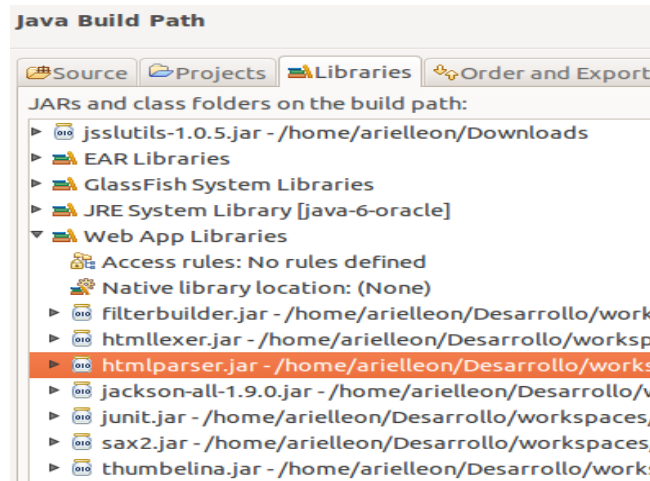
4.3.8. HTMLParser

Es una librería hecha en Java para analizar gramáticamente el contenido HTML de una página web, puede realizarse de forma linear o anidada. Es principalmente utilizada para transformación o extracción, utilizando filtros, etiquetas y JavaBeans. Es un paquete robusto, rápido y sometido a diversas pruebas.

Esta librería resuelve fundamentalmente dos casos de uso, la extracción y la transformación de datos desde páginas HTML. En este caso, se utilizarán exclusivamente las funciones de extracción. Para poder utilizar esta librería es necesario colocarla dentro de la ruta de construcción del proyecto de servicios.

⁴ JSON. <http://es.wikipedia.org/wiki/JSON>. Consulta: 29 de abril de 2015.

Figura 33. Ruta de construcción



Fuente: elaboración propia

4.4. Tutorial de desarrollo y referencias

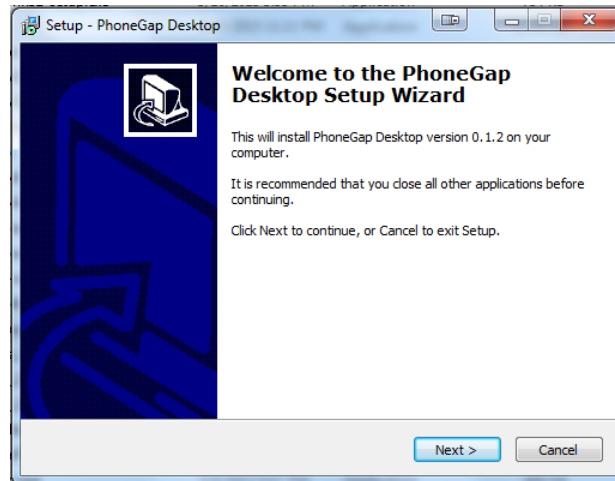
A continuación se detalla cómo se configuran las herramientas para desarrollar aplicaciones móviles, así como las funciones desarrolladas para la aplicación Mi Horario.

4.4.1. Configuración de PhoneGap en Windows

Primero, debe descargarse el instalador desde la página oficial de PhoneGap (<http://phonegap.com/install/>). Luego, se siguen los siguientes pasos:

- Se ejecuta el instalador y se le da siguiente a la página de bienvenida.

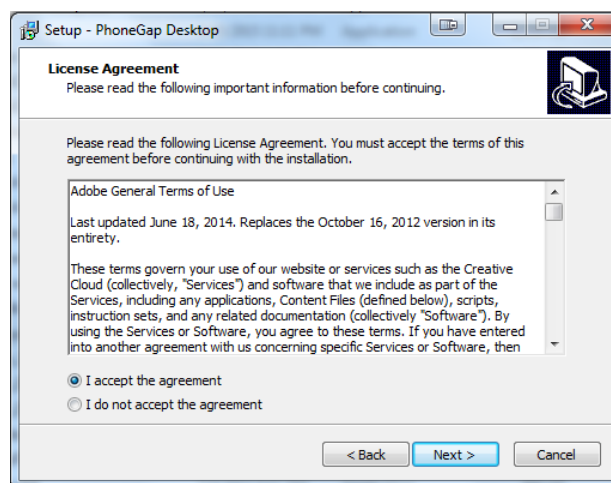
Figura 34. **Pantalla inicio de instalación de PhoneGap**



Fuente: elaboración propia.

- Se selecciona que se aceptan los términos de licencia de PhoneGap y se le da siguiente.

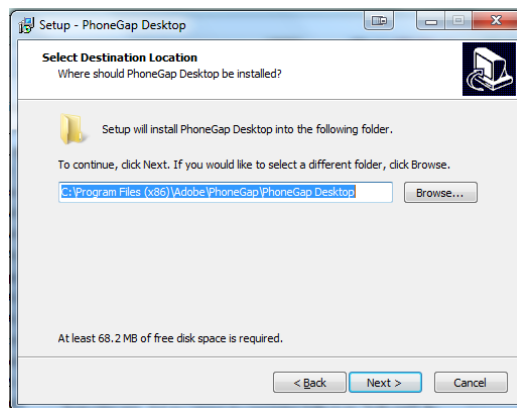
Figura 35. **Pantalla de aceptación de licencia de PhoneGap**



Fuente: elaboración propia.

- Se selecciona la ruta donde se quiere que PhoneGap sea instalado y se le da siguiente.

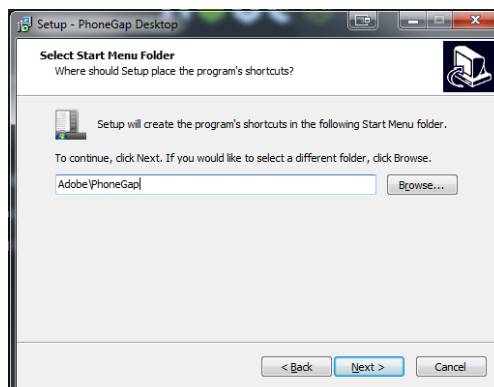
Figura 36. **Pantalla de selección de ruta de instalación de PhoneGap**



Fuente: elaboración propia.

- Se selecciona la ruta donde se quiere que ponga el menú de PhoneGap y se le da siguiente.

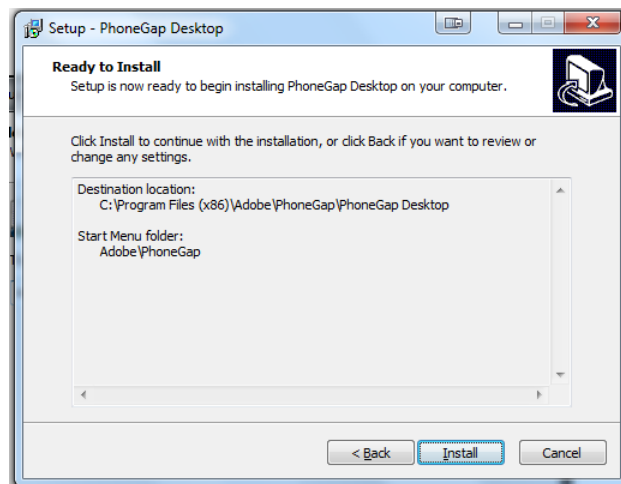
Figura 37. **Pantalla de selección de menú de PhoneGap**



Fuente: elaboración propia.

- Se verifica que todos los datos ingresados estén correctos y se presiona instalar.

Figura 38. **Pantalla de verificación de instalación de PhoneGap**



Fuente: elaboración propia.

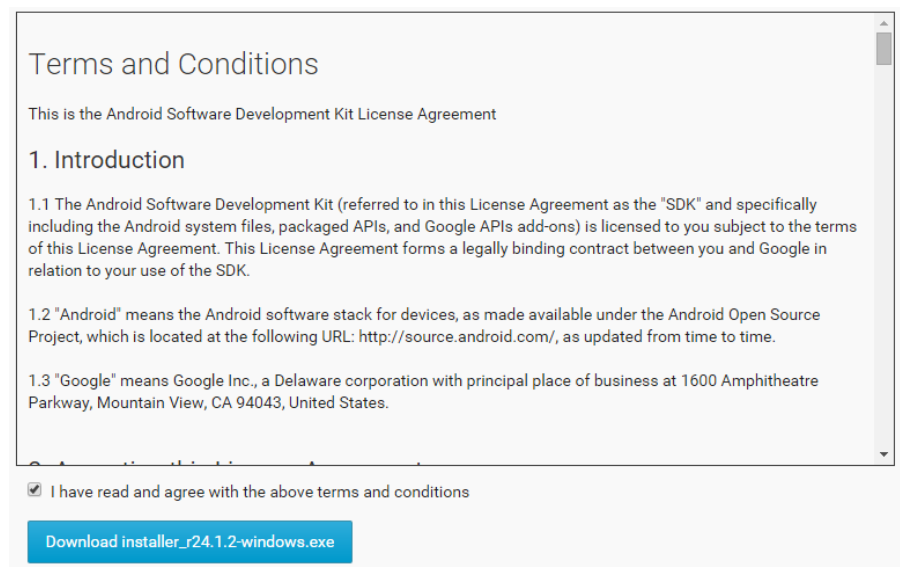
4.4.2. Configuración de AndroidSDK

Para poder instalar el Stand-alone de Android SDK en el computador se necesitan los siguientes requerimientos mínimos.

- Microsoft Windows 8/7/Vista/2003 de 32 o 64 bits
- 2 GB de memoria RAM, como mínimo
- 400 GB de disco duro
- 1280 x 800, resolución mínima de monitor
- Java Development Kit 7 instalado

- Desde el sitio oficial de Android SDK (<https://developer.android.com/sdk>), escoger la opción de descargar Stand-alone Android SDK y aceptar los términos de licencia.

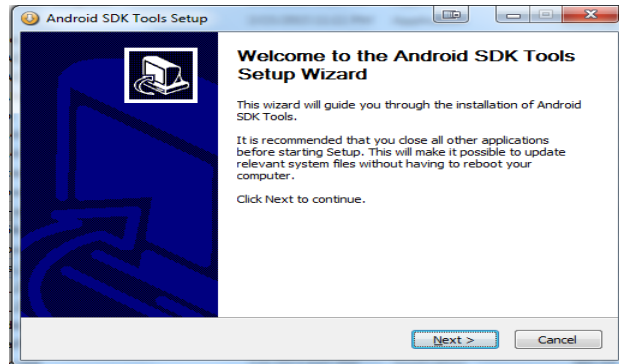
Figura 39. **Términos y condiciones para instalar Android SDK**



Fuente: elaboración propia.

- Ir a la carpeta donde se guardó el instalador descargado y darle doble clic para iniciar la instalación.

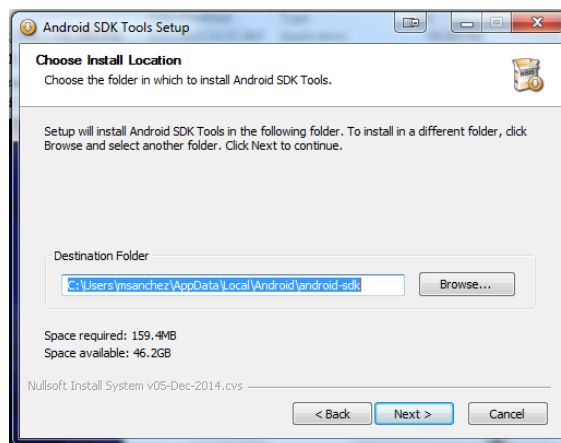
Figura 40. **Pantalla de inicio de instalación de Android SDK**



Fuente: elaboración propia.

- Seleccionar el directorio donde se quiere instalar el Android SDK.

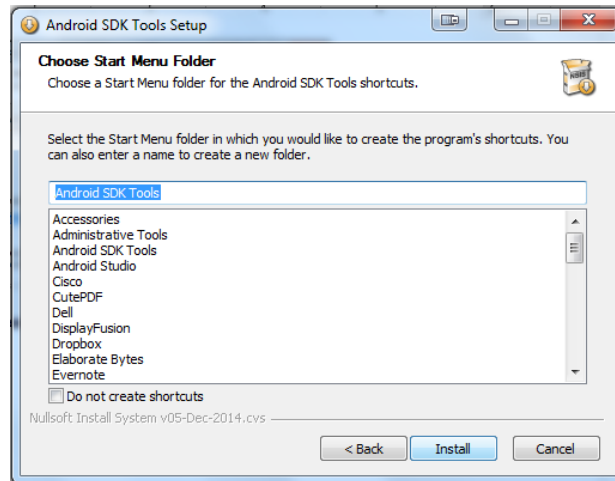
Figura 41. **Pantalla de selección de carpeta de instalación de Android SDK**



Fuente: elaboración propia.

- Seleccionar donde va a quedar el menú de inicio de Android SDK.

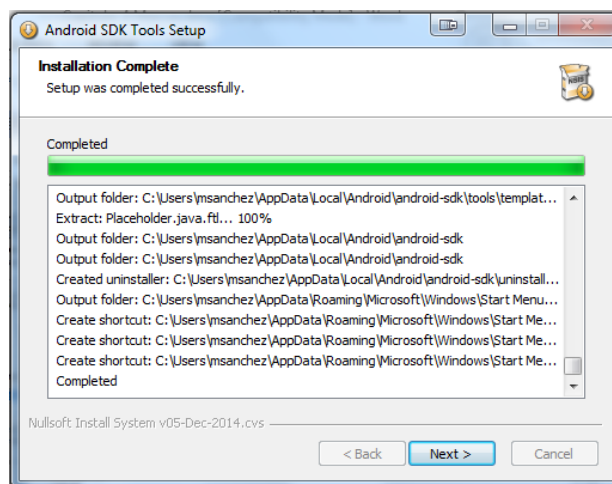
Figura 42. **Pantalla de menú de inicio de Android SDK**



Fuente: elaboración propia.

- Pantalla de finalización de la instalación de Android SDK.

Figura 43. **Pantalla de finalización de instalación de Android SDK**



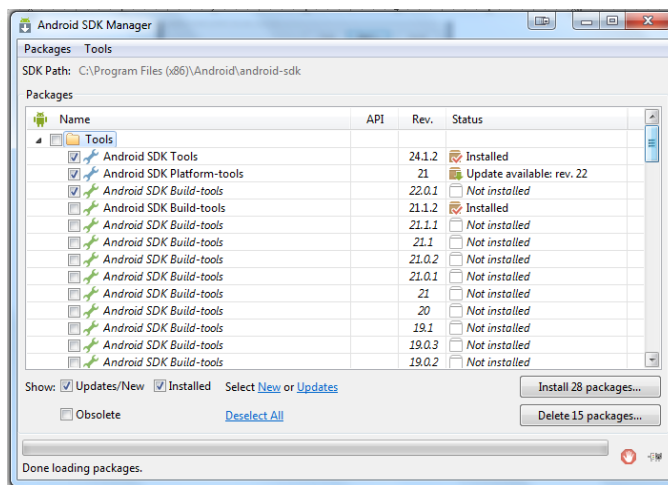
Fuente: elaboración propia.

- En el menú donde se instaló el Android SDK, se ejecuta el SDK Manager. Desde esta pantalla se puede seleccionar qué paquetes son los que se quieren descargar e instalar. Dentro de los paquetes requeridos que se deben de instalar están los siguientes:

- SDK Tools
- SDK Platform-Tools
- SDK Platform

Aparte de los paquetes requeridos, se deben seleccionar una o varias versiones de Android sobre las cuales se quieren crear los emuladores para hacer las pruebas.

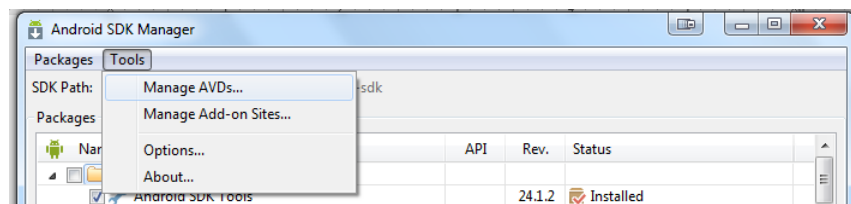
Figura 44. **Pantalla SDK Manager**



Fuente: elaboración propia.

- Después de instalados todos los paquetes necesarios, se va a herramientas y luego manejar dispositivos Android en el menú del SDK Manager.

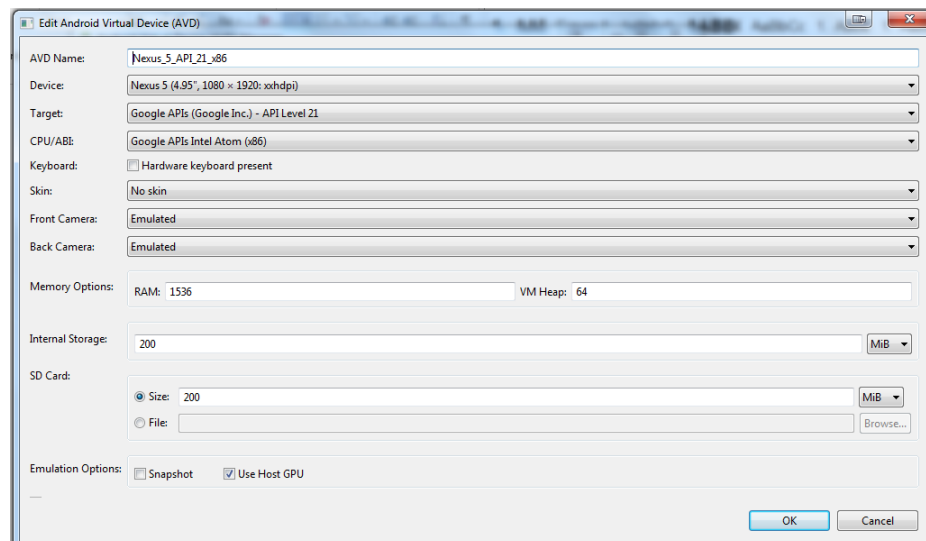
Figura 45. **Crear nuevo dispositivo Android**



Fuente: elaboración propia.

- Se selecciona crear un nuevo dispositivo Android y se ingresan todas las especificaciones de hardware y software que tendrá el emulador.

Figura 46. **Características de emulador de dispositivo Android**



Fuente: elaboración propia.

- Después de creado el emulador de dispositivo Android, se puede proceder a ejecutarlo.

Figura 47. **Emulador de dispositivo Android**



Fuente: elaboración propia.

4.4.3. Parseo de horario

Es utilizado para obtener, en línea y en tiempo real, los diferentes horarios de cursos y exámenes publicados en el sitio web de la Facultad de Ingeniería. Se utilizan las librerías `htmllexer.jar` y `htmlparser.jar` de Apache, estas permiten obtener el contenido de las páginas donde se publican los horarios.

Los horarios de la Facultad son publicados en tablas HTML con la estructura que puede observarse en la figura 48.

Figura 48. **Fragmento de código: estructura de horarios**

```
app.addToCalendar = function(idPeriodo, dia, curso, jornada, salon, hora) {  
  
    var hoy = new Date();  
    var anio = hoy.getFullYear();  
    var mes = 11;  
  
    var arrayDia = dia.split(" ");  
    var arrayHora = hora.split(":");  
  
    var startDate = new Date(anio, mes, arrayDia[1], arrayHora[0],  
        arrayHora[1], 0, 0, 0);  
    var endDate = new Date(anio, mes, arrayDia[1], parseInt(arrayHora[0]) + 1,  
        arrayHora[1], 0, 0, 0);  
    var title = "Examen: " + curso;  
    var eventLocation = salon;  
    var notes = "Jornada: " + jornada;  
    var success = function(message) {  
        console.debug("Evento agregado con éxito: " + JSON.stringify(message));  
    };  
    var error = function(message) {  
        console.debug("Error: " + message);  
    };  
  
    window.plugins.calendar.createEventInteractively(title, eventLocation,  
        notes, startDate, endDate, success, error);  
}
```

Fuente: elaboración propia.

De esta forma, se puede almacenar en una estructura de datos propia los diferentes horarios publicados en la página de la Facultad, recorriendo cada fila y cada celda como se puede observar en la figura 49.

Figura 49. Fragmento de código: HTMLParser

```
ArrayList<Horario> listaHorarios = null;
try {
    Parser parser = new Parser(url);
    NodeList list = parser.parse(null);
    NodeList trs = list.extractAllNodesThatMatch(new TagNameFilter("tr"),true);

    listaHorarios = new ArrayList<Horario>();

    for(int i=0;i<trs.size();i++) {
        Horario nuevoHorario = new Horario();
        NodeList nodes = trs.elementAt(i).getChildren();
        NodeList tds = nodes.extractAllNodesThatMatch(new TagNameFilter("td"),true);
        boolean ingresar = false;
        for(int j=0;j<tds.size();j++){

            TagNode tag1 = new TagNode();
            tag1 = (TagNode) tds.elementAt(j);

            if(tag1.getAttribute("class")!=null && (tag1.getAttribute("class").equals("filaPar") || t.
("class").equals("filaImpar"))){
                ingresar = true;

                NodeList nodos = tds.elementAt(j).getChildren();
                NodeList fonts = nodos.extractAllNodesThatMatch(new TagNameFilter("font"),true);
                if(fonts.size()>0){
                    TagNode tag = new TagNode();
                    tag = (TagNode) fonts.elementAt(0);
                    nuevoHorario.setColor(tag.getAttribute("color"));
                }
            }
        }
    }
}
```

Fuente: elaboración propia.

4.4.4. Generación de servicio REST

Se utilizaron servicios REST como interfaz de comunicación entre la aplicación y el servidor *backend* enviando mensajes en formato JSON.

Los servicios publicados fueron los siguientes:

4.4.4.1. Listado de cursos

Este es el servicio que permite obtener el listado de cursos de un determinado periodo y que coincida con el parámetro de búsqueda enviado.

El *path* de consumo es el siguiente:

<http://servidor/HorarioUsac/servicios/horarios?id=1&q=red>

- Contexto de aplicación: HorarioUsac
- Servicio: servicios
- Método: horarios
- Parámetros: id, identificador del período a consultar; q, criterio de búsqueda. Se retornan los cursos que pertenezcan al período id que contengan en el nombre del curso la cadena q.

Figura 50. **Fragmento de código: servicio de horarios**

```

@ApplicationPath("/servicios")
@Path("horarios")
public class Horarios extends Application {

    @GET
    public String listaHorario(@QueryParam("id")

```

Fuente: elaboración propia.

4.4.4.2. Detalle de cursos

Este es el servicio que permite obtener el detalle de un curso o examen determinado, que coincida con el identificador único y periodo enviado como parámetro.

El *path* de consumo es el siguiente:

<http://servidor/HorarioUsac/servicios/horarios/detalleHorario?p=1&ic=1181>

- Contexto de aplicación: HorarioUsac
- Servicio: servicios
- Método: detalleHorario

- Parámetros: p, identificador del período a consultar; ic, identificador único de curso. Se retornan el detalle del curso o examen que pertenezca al periodo p cuyo identificador sea igual a ic.

Figura 51. **Fragmento de código: servicio de detalle de horario**

```
@Path("detalleHorario")
@GET
public String detalleHorario(@QueryParam("p") int periodo,
    ObjectWriter ow = new ObjectMapper().writer().withDefaultPrettyPrinter(),
    Horario detalleCurso = new Horario());
```

Fuente: elaboración propia.

4.4.5. Configuración de jQuery Mobile

Para integrar jQuery Mobile únicamente es necesario copiar los siguientes *scripts* dentro del proyecto e incluirlos dentro de las páginas donde se desea utilizarlos.

- jquery.mobile-1.4.5.min.js
- jquery-1.11.2.min.js
- jquery.mobile-1.4.5.min.css

Figura 52. **Fragmento de código: inclusión de jQuery Mobile**

```
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1.0,
  <meta charset="utf-8">
  <link rel="stylesheet" href="css/jquery.mobile-1.4.5.min.css">
  <script src="scripts/jquery-1.11.2.min.js"></script>
  <script src="scripts/jquery.mobile-1.4.5.min.js"></script>
</head>
```

Fuente: elaboración propia.

4.4.6. Creación de aplicación

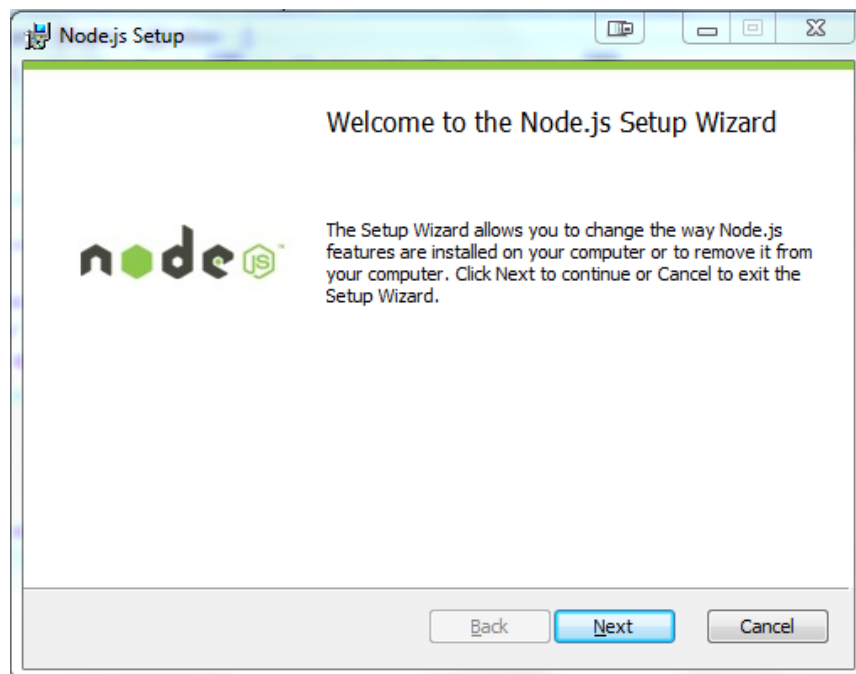
Para crear el proyecto donde se va a desarrollar la aplicación, se debe de tener instalado NodeJS.

4.4.6.1. Instalación de NodeJS en Windows

Se descarga el instalador de NodeJS desde el sitio oficial (<https://nodejs.org/>), y los siguientes pasos:

- Se hace clic en siguiente en la pantalla de bienvenida del instalador.

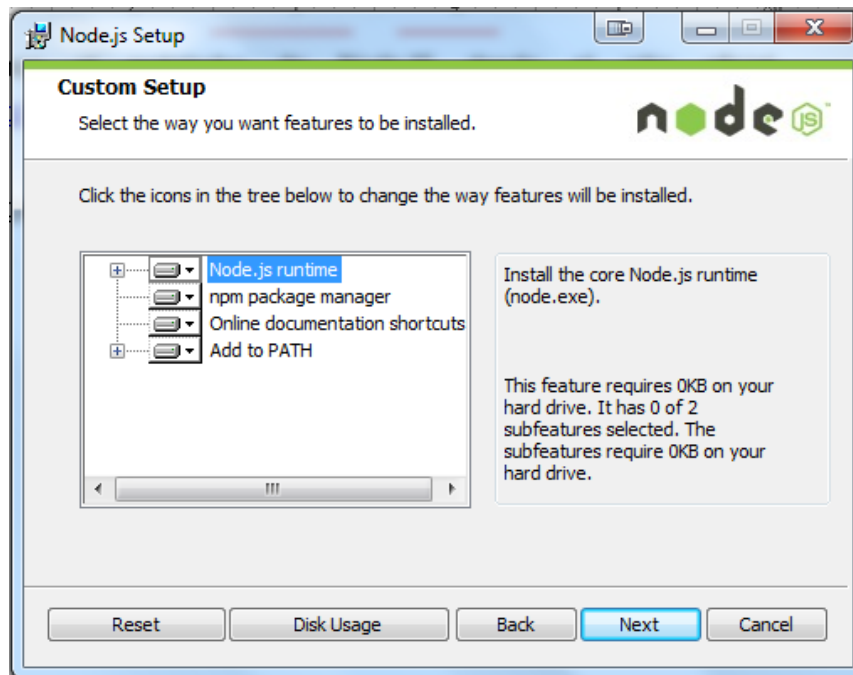
Figura 53. **Pantalla de bienvenida de instalación de NodeJS**



Fuente: elaboración propia.

- Se seleccionan los componentes que se quieren instalar.

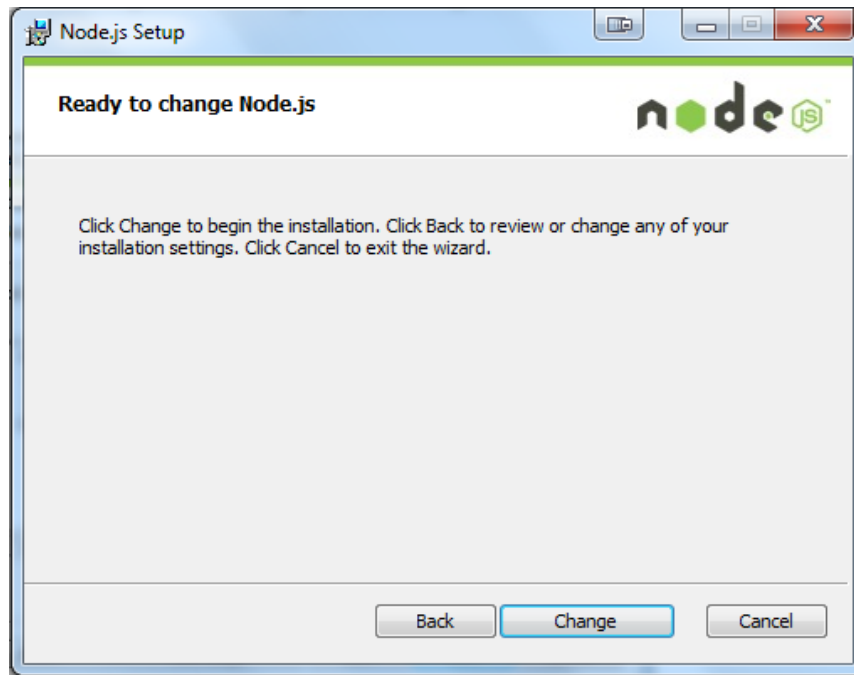
Figura 54. **Pantalla de selección de componentes de NodeJS**



Fuente: elaboración propia.

- Se presiona siguiente para finalizar la instalación de NodeJS.

Figura 55. **Pantalla de aceptación de instalación de NodeJS**

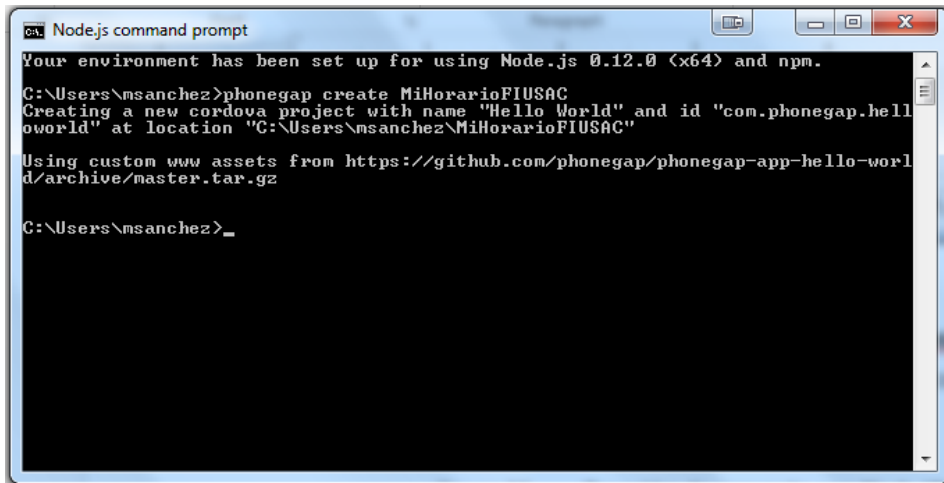


Fuente: elaboración propia.

4.4.6.2. Creación de proyecto MiHorarioFIUSAC

Se abre la consola de NodeJS, se escribe el siguiente comando *phonegap create MiHorarioFIUSAC* y se presiona enter.

Figura 56. Creación de proyecto con NodeJS

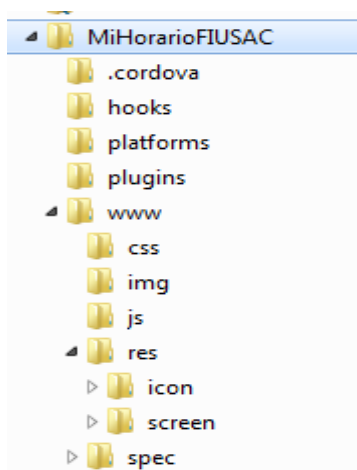


```
Node.js command prompt
Your environment has been set up for using Node.js 0.12.0 <x64> and npm.
C:\Users\msanchez>phonegap create MiHorarioFIUSAC
Creating a new cordova project with name "Hello World" and id "com.phonegap.hello-world" at location "C:\Users\msanchez\MiHorarioFIUSAC"
Using custom www assets from https://github.com/phonegap/phonegap-app-hello-world/archive/master.tar.gz
C:\Users\msanchez>_
```

Fuente: elaboración propia.

Creado el proyecto, se va a la dirección donde fue creado y se debería tener una estructura similar a la que se presenta en la figura 57.

Figura 57. Estructura del proyecto MiHorarioFIUSAC



Fuente: elaboración propia.

4.4.7. Desarrollo de páginas de la aplicación

En el *body* de las páginas se cuenta con la siguiente estructura:

- *Head*: en esta parte se coloca el encabezado de las páginas
- *Content*: contiene el cuerpo de cada una de las páginas
- *Footer*: muestra el pie de página de las páginas

Figura 58. Estructura del *body* de páginas

```
<div data-role="header" data-position="fixed">  
  
</div><!-- /header -->  
  
<div role="main" class="ui-content">  
  
</div>  
<!-- /content -->  
  
<div data-role="footer" data-theme="a">  
  
</div>  
<!-- /footer -->
```

Fuente: elaboración propia.

4.4.8. Base de datos local

Como se mencionó anteriormente, la base de datos que se utiliza en el proyecto es SQLite. En el documento de JavaScript, *bdMiHorario.js*, se encuentran todos los métodos implementados que son utilizados en el proyecto.

4.4.8.1. Abrir base de datos local

Al momento de querer insertar, consultar o eliminar datos de las tablas se ejecuta la función *app.openDb*.

Figura 59. Código de apertura de base de datos

```
app.openDb = function() {  
  app.db = window.openDatabase("PushAndSync", "1.0", "PushAndSync", 500000);  
}
```

Fuente: elaboración propia.

4.4.8.2. Creación de la tabla

En el proyecto se cuenta con una sola tabla que se llama bdMiHorario. Al entrar a la aplicación por primera vez se crea, las siguientes veces está lista para ser consultada.

La tabla cuenta con los siguientes campos:

- Id: campo de tipo entero que es autoincremental y es la llave primaria de la tabla.
- Periodold: campo entero que contiene el número del periodo.

Tabla VII. **Numeración de los periodos**

Periodo	Descripción
1	Cursos primer semestre
2	Cursos segundo semestre
14	Exámenes finales primer semestre
15	Exámenes finales segundo semestre
3	Primera retrasada primer semestre
4	Primera retrasada segundo semestre
7	Segunda retrasada primer semestre
8	Segunda retrasada segundo semestre

Fuente: elaboración propia.

- **Código:** campo de tipo texto que contiene el código del curso.
- **Curso:** campo de tipo texto que contiene el nombre del curso.
- **Sección:** campo de tipo texto que contiene la sección del curso.
- **Salón:** campo de tipo texto que contiene el salón del curso.
- **Hora:** campo de tipo texto que contiene la hora en que se imparte el curso.
- **Días:** campo de tipo texto que contiene los días en que se imparte el curso.
- **Catedrático:** campo de tipo texto que contiene el nombre del catedrático que da el curso.

Figura 60. **Tabla dbMiHorario**

bdMiHorario	
id	int
periodold	int
codigo	text
curso	text
seccion	text
salon	text
hora	text
dias	text
catedratico	text
added_on	datetime

Fuente: elaboración propia.

4.4.8.3. Seleccionar registros de la tabla

Para seleccionar los datos que están en las tablas, se utiliza la función *selectAll*.

Figura 61. Código de la función *selectAll*

```
function selectAll() {
  app.openDb();
  app.createTable();
  app.refresh();
}

app.refresh = function() {
  var renderTodo = function (row) {

    var periodoName = "";
    if (row.periodoId==1)
      periodoName = "Cursos Primer Semestre";
    else if (row.periodoId==2)
      periodoName = "Cursos Segundo Semestre";
    else if (row.periodoId==14)
      periodoName = "Exámenes Finales Primer Semestre";
    else if (row.periodoId==15)
      periodoName = "Exámenes Finales Segundo Semestre";
    else if (row.periodoId==3)
      periodoName = "Primera Retrasada Primer Semestre";
    else if (row.periodoId==4)
      periodoName = "Primera Retrasada Segundo Semestre";
    else if (row.periodoId==7)
      periodoName = "Segunda Retrasada Primer Semestre";
    else if (row.periodoId==8)
      periodoName = "Segunda Retrasada Segundo Semestre";
    else if (row.periodoId==2)
      periodoName = "Segundo Semestre";
  }
}
```

Fuente: elaboración propia.

4.4.8.4. Borrar los registros de la tabla

Para el borrado existen dos funciones, una cuando se desea borrar un solo registro y otra cuando se desean borrar todos los registros.

- Para borrar solamente un registro se utiliza la función *deleteTodo*

Figura 62. **Código para borrar un solo registro**

```
app.deleteTodo = function(id) {  
  var db = app.db;  
  db.transaction(function(tx) {  
    tx.executeSql("DELETE FROM bdMiHorario WHERE ID=?", [id],  
                 app.onSuccessRefresh(),  
                 app.onError);  
  });  
}
```

Fuente: elaboración propia.

- Para eliminar todos los registros se utiliza la función “*deleteTodoAll*”

Figura 63. **Código para eliminar todos los registros**

```
app.deleteTodoAll = function() {  
  var db = app.db;  
  db.transaction(function(tx) {  
    tx.executeSql("DELETE FROM bdMiHorario2 ", [],  
                 app.onSuccess,  
                 app.onError);  
  });  
}
```

Fuente: elaboración propia.

4.4.8.5. Verificación de existencia de registro

Antes de cada inserción en nuestra tabla, se verifica que el registro que se quiere agregar no esté ya guardado en la tabla con la función *exists*.

Figura 64. Código para verificar existencia de un registro

```
function exists(periodoId,codigo){
    var db = app.db;
    var x;
    db.readTransaction(function (tx) {
        tx.executeSql('SELECT COUNT(*) AS c
                    FROM bdMiHorario
                    where periodoId = ' + periodoId +
                    ' and codigo="' + codigo + '"', [], function (tx, r) {
            return r.rows.item(0).c;
        });
    });
}
```

Fuente: elaboración propia.

4.4.9. Consumo de servicios REST

El consumo de los servicios REST se realizó utilizando una petición AJAX del *framework* jQuery. Esto permite obtener asincrónicamente los resultados en formato JSON que, posteriormente, es procesado para la manipulación y despliegue de los datos obtenidos.

- Función utilizada: *\$.ajax*
- Parámetros: *url, dataType, crossDomain, data*
- Respuesta: *result*

Figura 65. **Fragmento de código: consumo de servicio**

```
$.ajax({
  url: url,
  dataType: "json",
  crossDomain: true,
  data: {
    q: $input.val().trim()
  }
})
.then( function ( result ) {
```

Fuente: elaboración propia.

El resultado obtenido es almacenado en la variable *result* y guarda la respuesta en formato JSON, esta respuesta tiene los formatos presentados en las figura 66 y 67.

Figura 66. **Fragmento de código: respuesta servicio horarios**

```
[ {
  "nombreCurso" : "REDES DE COMPUTADORAS 1",
  "idCurso" : 1093,
  "tipo" : 1,
  "detalle" : "A "
}, {
  "nombreCurso" : "REDES DE COMPUTADORAS 1",
  "idCurso" : 1094,
  "tipo" : 1,
  "detalle" : "A "
}, {
  "nombreCurso" : "REDES DE COMPUTADORAS 1",
  "idCurso" : 1095,
  "tipo" : 2,
  "detalle" : "A "
}, {
  "nombreCurso" : "REDES DE COMPUTADORAS 2",
  "idCurso" : 1096,
  "tipo" : 1,
  "detalle" : "N "
}, {
  "nombreCurso" : "REDES DE COMPUTADORAS 2",
  "idCurso" : 1097,
  "tipo" : 2,
  "detalle" : "N "
}, {
  "nombreCurso" : "TELECOMUNICACIONES Y REDES LOCALES",
  "idCurso" : 1216,
  "tipo" : 1,
  "detalle" : "N "
}, {
  "nombreCurso" : "TELECOMUNICACIONES Y REDES LOCALES",
  "idCurso" : 1217,
  "tipo" : 1,
  "detalle" : "N "
} ]
```

Fuente: elaboración propia.

Figura 67. Fragmento de código: respuesta detalle horario

```
{
  "codigoCurso" : "0775",
  "nombreCurso" : "SISTEMAS DE BASES DE DATOS 2",
  "seccion" : "B",
  "edificio" : "T-3",
  "salon" : "309",
  "horaInicio" : "12:30",
  "horaFin" : "14:10",
  "lunes" : "-",
  "martes" : "-",
  "miercoles" : "-",
  "jueves" : "-",
  "viernes" : "-",
  "sabado" : "X",
  "catedratico" : "STAFF",
  "color" : "#0000FF",
  "jornada" : "",
  "dia" : "",
  "tipo" : 2
}
```

Fuente: elaboración propia.

4.4.10. Creación de un evento en calendario

Para realizar la creación del evento correspondiente a un examen o retrasada en el calendario personal del dispositivo, se creó la función *addToCalendar*, la cual recibe los siguientes parámetros:

- *idPeriodo*: necesario para calcular el mes al cual pertenece el examen
- *dia*: día de la semana en que se realizará el examen
- *curso*: nombre del curso a examinarse
- *jornada*: descripción de la jornada donde se imparte el curso
- *salón*: salón donde se realizará el examen
- *hora*: necesario para crear el evento en hora exacta

Adicional a estos parámetros, se obtiene la fecha actual creando un nuevo objeto de tipo *date* (*new Date()*) y, posteriormente, invocando el método *getFullYear()*.

Al agregar el horario, se muestra la ventana de creación de eventos predeterminada del dispositivo, para darle al estudiante la oportunidad de configurar personalmente la descripción o recordatorios del evento. Sin embargo, los campos aparecen llenos por defecto con la información más relevante del examen.

Para esto fue utilizada la función *createEventInteractively* del *plugin* Calendar, la cual recibe los siguientes parámetros:

- *title*: título del evento, se envía el texto “Examen” concatenado al nombre del curso.
- *eventLocation*: lugar del evento, se envía el salón y edificio donde se realizará el examen.
- *Notes*: se envía la jornada del curso.
- *startDate*: fecha y hora de inicio del examen.
- *endDate*: ya que el horario publicado no cuenta con hora de fin, por defecto se envía la fecha de inicio más una hora.
- *success*: función de retorno en caso de éxito.
- *error*: función de retorno en caso de error.

Figura 68. **Fragmento de código: función *addToCalendar***

```
app.addToCalendar = function(idPeriodo, dia, curso, jornada, salon, hora) {  
  
    var hoy = new Date();  
    var anio = hoy.getFullYear();  
    var mes = 11;  
  
    var arrayDia = dia.split(" ");  
    var arrayHora = hora.split(":");  
  
    var startDate = new Date(anio, mes, arrayDia[1], arrayHora[0],  
        arrayHora[1], 0, 0, 0);  
    var endDate = new Date(anio, mes, arrayDia[1], parseInt(arrayHora[0]) + 1,  
        arrayHora[1], 0, 0, 0);  
    var title = "Examen: " + curso;  
    var eventLocation = salon;  
    var notes = "Jornada: " + jornada;  
    var success = function(message) {  
        console.debug("Evento agregado con éxito: " + JSON.stringify(message));  
    };  
    var error = function(message) {  
        console.debug("Error: " + message);  
    };  
  
    window.plugins.calendar.createEventInteractively(title, eventLocation,  
        notes, startDate, endDate, success, error);  
}
```

Fuente: elaboración propia.

4.5. Consideraciones de implementación

Para poder subir una aplicación a Google Play es necesario seguir una serie de lineamientos, tanto para la generación del instalable (APK) como para la publicación. A continuación se describen los pasos a seguir para cada tarea.

4.5.1. Generación de APK

Para la generación del APK que se va a subir a Google Play, se deben seguir los siguientes pasos.

4.5.1.1. Almacén de claves

El primer paso es generar un almacén de claves (*keystore*), el cual se utilizará para firmar digitalmente la aplicación. Esto se realiza con el programa del Java Development Kit (JDK) llamado Keytool.

El comando a ejecutar es el siguiente:

```
keytool -genkey -v -keystorehorario-fiusac-key.keystore -alias arielleon -keyalg RSA -keysize 2048 -validity 10000
```

Parámetros:

- *genkey*: opción de creación del almacén
- *keystore*: nombre del almacén
- *alias*: nombre alternativo del almacén
- *keyalg*: algoritmo de encriptación
- *keysize*: tamaño de la llave
- *validity*: duración en días

Este paso se ejecuta únicamente una vez, es decir, al tener creado el almacén de claves ya no será necesario volverlo a generar y solo se invoca en los comandos posteriores.

4.5.1.2. Generar versión productiva

El siguiente paso es generar una versión productiva de la aplicación con PhoneGap, para esto, en el directorio del proyecto se ejecuta el siguiente comando:

phonegap build android--release

Esto genera un archivo llamado CordovaApp-release-unsigned.apk, el archivo productivo, es decir que no tiene habilitadas las opciones de depuración, pero aún no cuenta con la firma digital, necesaria para subirse a Google Play.

4.5.1.3. Firma digital

Para firmar digitalmente el archivo generado, se utiliza el programa Jarsigner del Java Development Kit (JDK), es en este punto donde se utiliza el almacén de claves creado en el primer paso. Para firmar digitalmente el archivo se ejecutan los siguientes comandos:

```
jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore  
horario-fiusac-key.keystoreCordovaApp-release-unsigned.apkarielleon
```

```
jarsigner -verify -verbose -certs CordovaApp-release-unsigned.apk
```

4.5.1.4. Empaquetado y compresión

El último paso para la generación del archivo final (APK), que se podrá subir al *store*, es empaquetar y comprimir el archivo firmado utilizando el programa Zipalign del Java Development Kit (JDK), esto se realiza ejecutando el siguiente comando:

```
zipalign -v 4 CordovaApp-release-unsigned.apkMiHorarioUSAC.apk
```

4.5.2. Publicación de la aplicación

Para la publicación la aplicación en el Google Play se deben seguir los siguientes pasos.

4.5.2.1. Cuenta de programador de Google

Para publicar aplicaciones Android en Google Play es necesario contar con una cuenta de programador de Google, esto se realiza siguiendo los siguientes pasos:

- Registrarse
- Aceptar el Acuerdo de distribución para programadores
- Pagar la tarifa de registro
- Completar los datos de la cuenta

4.5.2.2. Consola de programador de Google

Al tener activa la cuenta de programador, se puede acceder a la consola de programador. Desde aquí se puede subir la aplicación, administrar las versiones de prueba, recolectar estadísticas y comentarios.

Figura 69. **Consola de programador de Google**

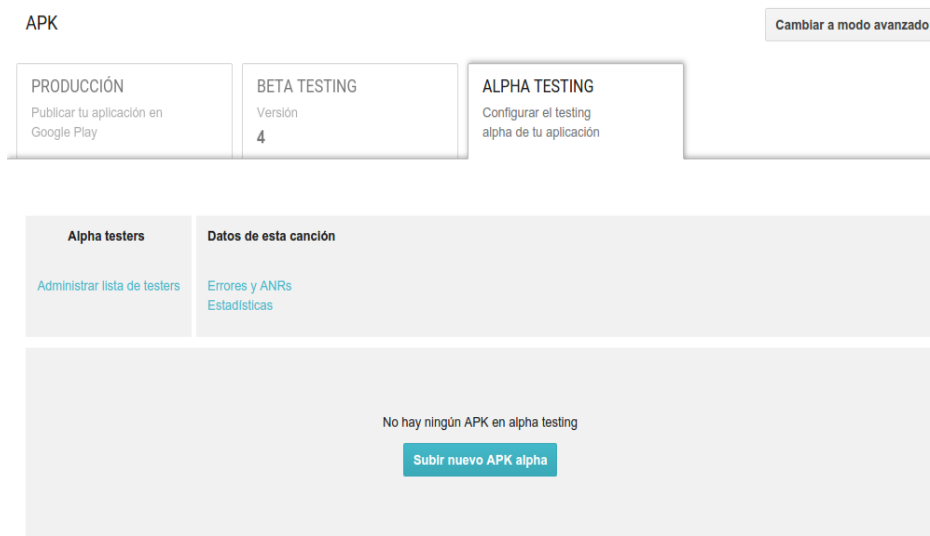


Fuente: elaboración propia.

4.5.2.3. **Subir APK**

Ya dentro de la consola de programador, el primer paso es subir el APK, Google proporciona tres etapas para administrar las versiones y pruebas. Se elige si se quiere subir una versión alpha, beta o productiva.

Figura 70. **Subir APK: versiones**



Fuente: elaboración propia.

Luego de decidir el tipo de versión, se hace clic en el botón Subir nuevo APK, esto muestra la siguiente pantalla donde se puede seleccionar o arrastrar el archivo APK firmado que se creó en la etapa de generación (ver figura 71).

Figura 71. **Subir APK: selección de archivo**



Fuente: elaboración propia.

Es importante administrar, en tiempo de compilación, el código de versión cada vez que se suba una nueva versión a Google Play, para esto se puede ir aumentando manualmente un correlativo en el archivo config.xml del directorio donde se encuentra el proyecto de PhoneGap. Esto se configura con la siguiente etiqueta:

```
<widget id="com.fiusac.mihorario" versionCode="4" version="0.0.4"
```

4.5.2.4. Llenar la ficha de Play Store

El siguiente paso es llenar la información de la aplicación que se está subiendo, esta ficha cuenta con varias secciones, las cuales deben ser completadas ingresando todos los campos obligatorios, marcados con asterisco. Las secciones de información son las siguientes:

- Información del producto
- Elementos gráficos
- Categorización
- Datos de contacto
- Política de privacidad

A continuación se muestra la ficha ya completa (ver figuras 72-77).

Figura 72. **Ficha: información del producto**

INFORMACIÓN DEL PRODUCTO Para publicar la aplicación, debes rellenar los campos marcados con *

Español (Latinoamérica) – es-419 Gestionar traducciones ▼

Título*
Español (Latinoamérica) – es-419
17 de 30 caracteres

Descripción breve*
Español (Latinoamérica) – es-419
61 de 80 caracteres

Descripción completa*
Español (Latinoamérica) – es-419
347 de 4000 caracteres

Fuente: elaboración propia.

Figura 73. Ficha: elementos gráficos – pantallas

ELEMENTOS GRÁFICOS

Si no añades imágenes localizadas para cada idioma, se usarán las imágenes del idioma predeterminado.

[Más información sobre los archivos de imágenes](#)

Capturas de pantalla *

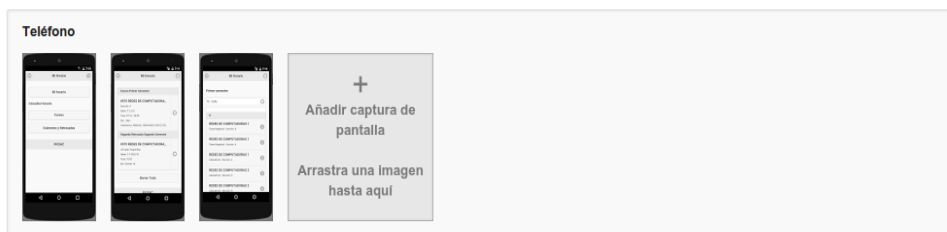
Español (Latinoamérica) – es-419 (predeterminado)

Archivo JPEG o PNG de 24 bits (no alfa). Longitud mínima para los laterales: 320 píxeles. Longitud máxima para los laterales: 3840 píxeles

Se necesitan al menos 2 capturas de pantalla (8 capturas de pantalla como máximo por tipo). Arrastra el dedo para cambiar el orden o para moverte por los diferentes tipos.

Para que tu aplicación aparezca destacada en la lista de aplicaciones diseñadas para tablets de Play Store, debes subir al menos una captura para tablets de 7 pulgadas y otra para tablets de 10 pulgadas. Si ya has subido capturas de pantalla, asegúrate de moverlas al área correspondiente que aparece a continuación.

[Más información sobre cómo las capturas de pantalla de tablets se mostrarán en las fichas de Google Play Store.](#)



Fuente: elaboración propia.

Figura 74. Ficha: elementos gráficos – icono



Fuente: elaboración propia.

Figura 75. **Ficha: categorización**

CATEGORIZACIÓN


Tipo de aplicación *







Categoría *

Clasificación de contenido *

[Más información sobre la clasificación de contenido](#)

Nueva clasificación de contenido *

 **CLASIFICACIÓN APLICADA**
Enviada: 28 de mar. 21:22
[Ver detalles](#) [Más información](#)

Fuente: elaboración propia.

Figura 76. **Ficha: datos de contacto**

DATOS DE CONTACTO

Sitio web

Correo electrónico *
Indica una dirección de correo electrónico en la que los usuarios puedan ponerse en contacto contigo. Esta dirección se mostrará públicamente con tu aplicación.

Teléfono

Fuente: elaboración propia.

Figura 77. **Ficha: política de privacidad**

POLÍTICA DE PRIVACIDAD *

Si quieres proporcionar la URL de la política de privacidad de esta aplicación, indícala a continuación.

Política de privacidad

No indicar por ahora ninguna política de privacidad [Más información](#)

Fuente: elaboración propia.

Al completar esta información, la aplicación entra en el período de aprobación, el cual suele demorar algunas horas antes la que la aplicación sea

visible al público o al conjunto de *testers* asignados a las versiones alpha o beta.

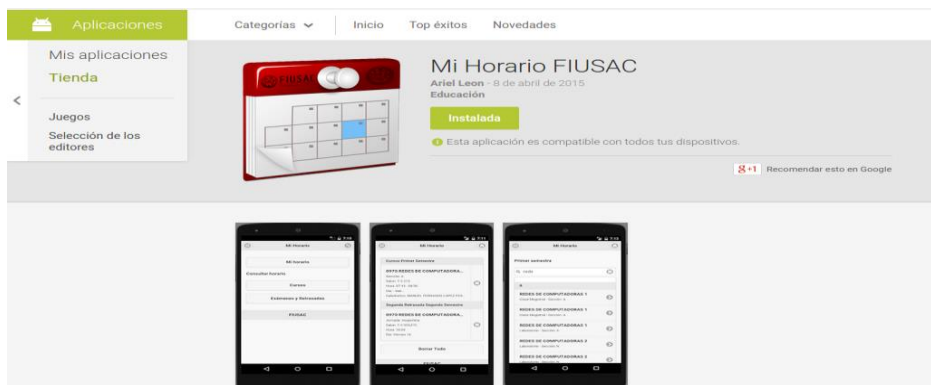
Las siguientes secciones fueron omitidas del proceso, por ser una aplicación gratuita o no utilizar la funcionalidad mencionada:

- Precio y distribución
- Productos integrados en la aplicación
- Servicios y API

4.5.2.5. Ver en Play Store

Al superar el proceso y período de aprobación de Google, la aplicación queda disponible para instalarse en el dispositivo móvil.

Figura 78. Ver en Play Store



Fuente: elaboración propia.

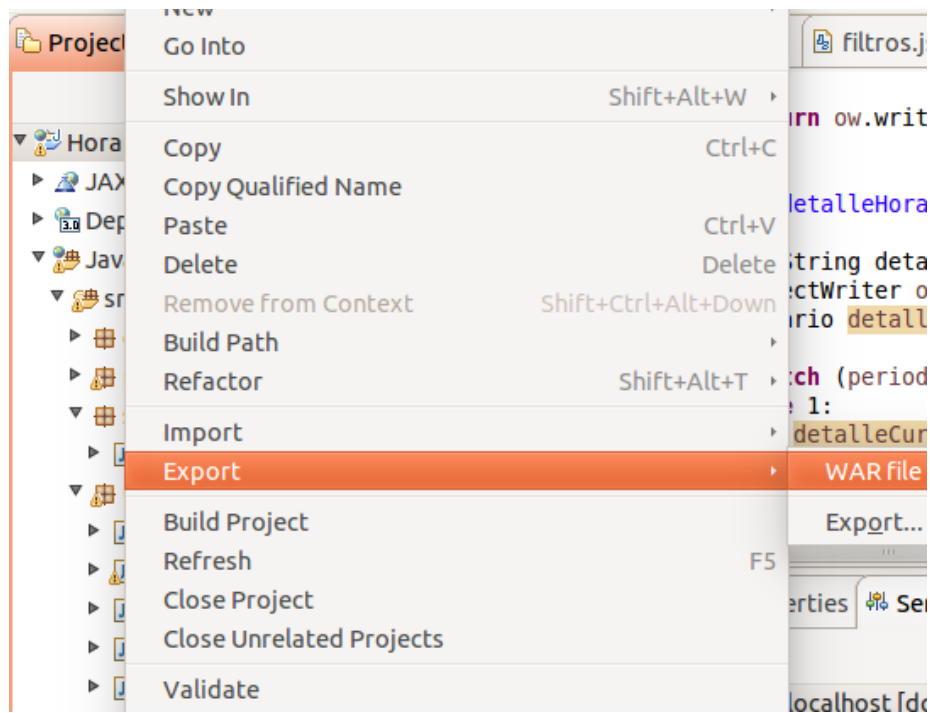
4.5.3. Publicación de servicios

Para la publicación de los servicios de la aplicación se deben seguir los siguientes pasos.

4.5.3.1. Generación de WAR

La generación del archivo WAR, que contiene la aplicación encargada de publicar los servicios REST consumidos por la aplicación, se realiza desde Eclipse. Se hace clic derecho en la carpeta del proyecto, luego se posiciona en el menú Export y se selecciona la opción Generar WAR.

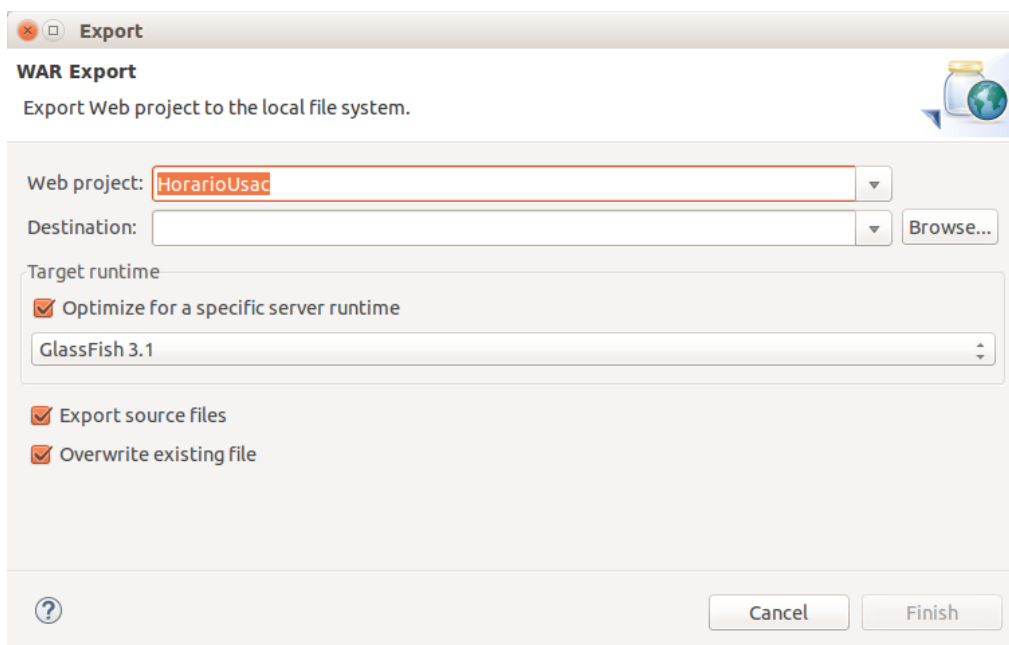
Figura 79. Generación de WAR – exportar



Fuente: elaboración propia.

El siguiente paso es seleccionar dónde se almacenará, en el sistema de archivos local, el archivo generado, el entorno de ejecución para que sea optimizado, si se desea exportar el código fuente y reemplazar el archivo si ya existe alguno con el mismo nombre.

Figura 80. **Generación de WAR – detalles**

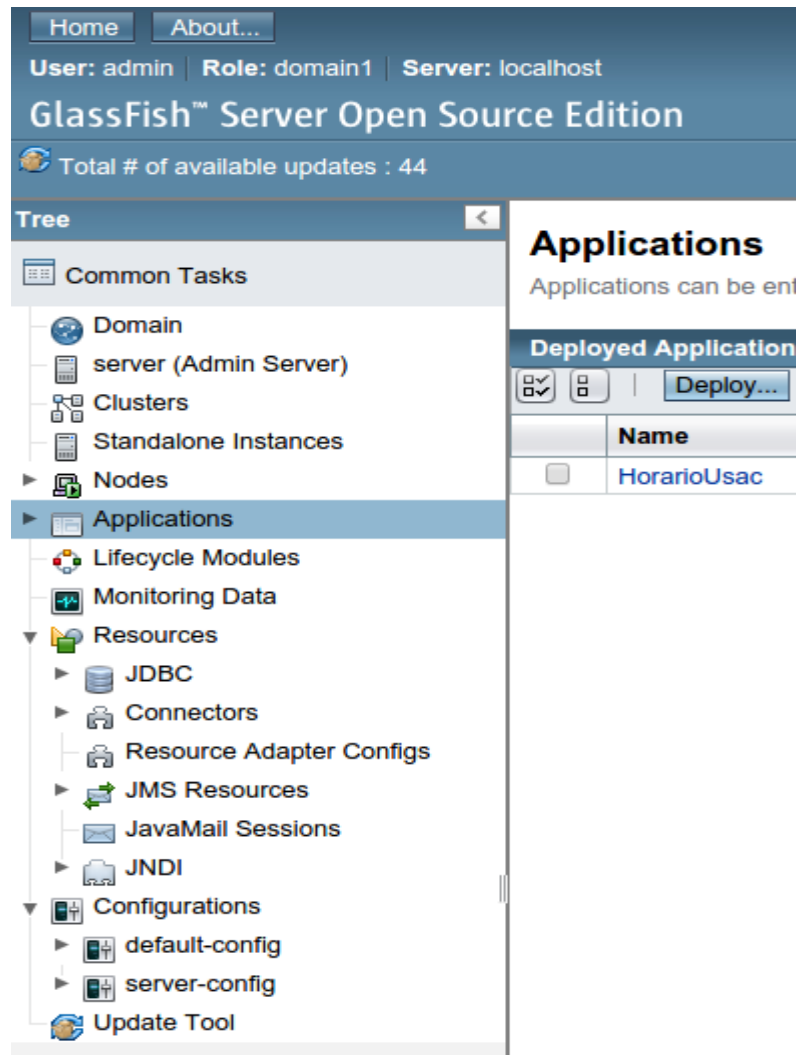


Fuente: elaboración propia.

4.5.3.2. **Publicación en Glassfish**

Para finalizar el proceso de publicación de los servicios REST, se ingresa a la consola administrativa del servidor de aplicaciones Glassfish y se selecciona la opción Applications.

Figura 81. **Publicación en Glassfish – aplicaciones**



Fuente: elaboración propia.

Luego de esto, se presiona el botón Deploy lo cual mostrará la siguiente pantalla para seleccionar el archivo WAR generado en la sección anterior (ver figura 82).

Figura 82. **Publicación en Glassfish – selección de WAR**

Deploy Applications or Modules
Specify the location of the application or module to deploy. An application can be in a packaged file or specified as a directory.

Location: **Packaged File to Be Uploaded to the Server**
 No file chosen

Local Packaged File or Directory That is Accessible from GlassFish Server

Type: *

Fuente: elaboración propia.

El siguiente paso es ingresar la raíz de contexto de la aplicación, con la palabra `HorarioUsac`, dejar el resto de valores seleccionados por defecto y presionar el botón OK.

Figura 83. **Publicación en Glassfish – raíz de contexto**

Type: *

Context Root:
Path relative to server's base URL.

Application Name: *

Virtual Servers:
Associates an Internet domain name with a physical server.

Status: **Enabled**
Allows users to access the application.

Precompile JSPs:
Precompiles JSP pages during deployment.

Run Verifier:
Verifies the syntax and semantics of the deployment descriptor. Verifier packages must be installed.

Force Redeploy:
Forces redeployment even if this application has already been deployed or already exists.

Keep State:
Retains web sessions, SF/SB instances, and persistently created EJB timers between redeployments.

Libraries:
A comma-separated list of library JAR files. Specify the library JAR files by their relative or absolute paths. Specify relative paths relative to instance-root/lib/app/lib. The libraries are made available to the application in the order specified.

Description:

Fuente: elaboración propia.

CONCLUSIONES

1. El sitio de la Facultad de Ingeniería es uno de los más visitados para la consulta de horarios de los semestre, por tal motivo se creó una aplicación móvil que es amigable al usuario, en la cual se pueden consultar dichos cursos.
2. La aplicación móvil permite que el estudiante guarde, dentro de su dispositivo, los cursos que llevará durante el semestre.
3. La aplicación móvil permite calendarizar y crear eventos para los exámenes finales de cada semestre.
4. La aplicación móvil está siempre disponible para su descarga en el Play Store de Android.

RECOMENDACIONES

1. Escalar el desarrollo de la aplicación a otras plataformas como iOS y Windows Phone, para beneficiar a más estudiantes.
2. Monitorear el estado de las publicaciones de horarios en los períodos críticos, como cambios de ciclo y cursos de vacaciones.

BIBLIOGRAFÍA

1. Android SDK. *Documentación*. [en línea]. <<https://developer.android.com/sdk/index.html>>. [Consulta: marzo de 2015].
2. Eclipse. *Documentación*. [en línea]. <<https://eclipse.org/>>. [Consulta: marzo de 2015].
3. GlassFish. *Servidor de Aplicaciones*. [en línea]. <<https://glassfish.java.net/es/>>. [Consulta: marzo de 2015].
4. HTML5. *Documentación*. [en línea]. <http://www.w3schools.com/html/html5_intro.asp>. [Consulta: marzo de 2015].
5. jQueryMobile. *Documentación*. [en línea]. <<http://api.jquerymobile.com>>. [Consulta: febrero de 2015].
6. PhoneGap. *Desarrollo*. [en línea]. <<http://phonegap.com/>>. [Consulta: febrero de 2015].
7. SQLite. *Documentación*. [en línea]. <<https://www.sqlite.org/>>. [Consulta: marzo de 2015].

APÉNDICE

Encuesta - Horario FIUSAC

*Obligatorio

¿Has navegado por el sitio de la Facultad de Ingeniería de la Universidad de San Carlos desde un dispositivo móvil? *

- Sí
- No

¿A que escuela de la Facultad de Ingeniería perteneces? *

- Escuela de Ingeniería Civil
- Escuela de Ingeniería Química
- Escuela de Ingeniería Mecánica
- Escuela de Ingeniería Mecánica Eléctrica
- Escuela de Ingeniería Mecánica Industrial
- Escuela de Ingeniería en Industrias Agropecuarias y Forestales
- Escuela de Ingeniería en Ciencias y Sistemas
- Escuela Regional de Ingeniería Sanitaria y Recursos Hidráulicos - ERIS
- Escuela de Ciencias
- Escuela Técnica
- Escuela de Estudios de Postgrado
- Licenciatura en Física Aplicada
- Licenciatura en Matemáticas

¿Instalarías en tu dispositivo móvil una aplicación que te permita consultar el horario de cursos y personalizar tu horario? *

- Sí
- No

¿Que dispositivo móvil utilizas para navegar por internet? *

- Smartphone
- Tablet
- Netbook
- Ninguno
- Otro:

¿Que sistema operativo tiene el dispositivo móvil seleccionado en la pregunta anterior?

- Android
- IOS
- Windows Phone
- Blackberry OS
- Otro:

Enviar

Nunca envíes contraseñas a través de Formularios de Google.

Con la tecnología de
 Google Forms

Este contenido no ha sido creado ni aprobado por Google.
[Informar sobre abusos](#) - [Condiciones del servicio](#) - [Otros términos](#)

Fuente: elaboración propia.

