



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS**

**ESTUDIO ESTADÍSTICO COMPARATIVO ENTRE SENSORES
ANDROID Y WINDOWS PHONE APLICADO EN LA DETECCIÓN
DE MOVIMIENTOS TELÚRICOS.**

TESIS DE GRADO

**Previa la obtención del título de
INGENIERO EN SISTEMAS INFORMÁTICOS**

Presentado por:

**LUCY JOHANNA HONORES CHUCHUCA
JENNY GABRIELA VIZUETE SALAZAR**

RIOBAMBA-ECUADOR

2014

DEDICATORIA

Dedico esta tesis a mis padres por su esfuerzo y sacrificio por que siempre creyeron en mi y a pesar de mis errores nunca me dejaron de apoyar para que yo alcance mi meta.

A mis sobrinos que con sus ocurrencias me sacan sonrisas cuando me siento mas triste.

A todos mis amigos que siempre estuvieron a mi lado en este largo camino pero en especial a Jenny que ha sabido ser más que una amiga eres mi hermana gracias por tu apoyo incondicional.

Lucy Johanna Honores Chuchuca

DEDICATORIA

Agradezco en primer lugar a Dios por darme fortaleza, salud y vida para cumplir una de mis metas en la vida, a mis padres Rosa e Ignacio por su sacrificio, amor, comprensión diario para que sea una buena persona y una buena profesional, a mis hermanas, a mi hermano y a mis sobrinos quienes han sabido motivarme en los momentos de flaqueza, a mi gran amiga Johanna con quien he compartido lindas experiencias y a una persona especial que me ha brindado su apoyo incondicional en esta etapa de mi vida.

Jenny Gabriela Vizuite Salazar

NOMBRE	FIRMA	FECHA
Ing. Iván Menes Camejo DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
Ing. Jorge Huilca Palacios DIRECTOR DE LA ESCUELA DE INGENIERÍA EN SISTEMAS.
Ing. Diego Ávila Pesantez DIRECTOR DE TESIS
Ing. Paulina Vélez Núñez MIEMBRO DEL TRIBUNAL
Ing. Eduardo Tenelanda DIRECTOR (E) CENTRO DE DOCUMENTACIÓN
NOTA DE LA TESIS	

Nosotros, **Lucy Johanna Honores Chuchuca** y **Jenny Gabriela Vizuite Salazar** somos responsables de las ideas, doctrinas y resultados expuestos en esta tesis; y, el patrimonio intelectual de la Tesis de Grado pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

.....

.....

Lucy Johanna Honores Chuchuca

Jenny Gabriela Vizuite Salazar

ÍNDICE DE ABREVIATURAS

API:	Interfaz de programación de aplicaciones
CDMA:	Acceso múltiple por división de código
Cobol:	Lenguaje Común Orientado a Negocios
EJB:	Enterprise JavaBeans
GIF:	Graphics Interchange Format
GUI:	Interfaz Gráfica de Usuario
GPS:	Sistema de posicionamiento global
GSM:	Sistema global para las comunicaciones móviles
GHz:	Gigahercio
GB:	Gigabyte
HTC:	High Tech Computer
HTML:	HyperText Markup Language
IBM:	International Business Machines
IDE:	Entorno de Desarrollo Integrado
JSP:	Java Server Pages
OS:	Sistema Operativo
OLED:	Diodo orgánico de emisión de luz
PDA:	Asistente digital personal
PC:	Computadora personal
PHP:	Hypertext Pre-processor
Pdf:	Portable document format
RAM:	Random-access memory
RGB:	Red, Green, Blue
RIM:	Research In Motion
WiMAX:	Worldwide Interoperability for Microwave Access
XML:	Lenguaje de Mercado Extensible
XP:	Extreme Programming

INDICE

INTRODUCCIÓN	1
CAPÍTULO I	3
MARCO REFERENCIAL.....	3
1. TITULO DEL PROYECTO DE TESIS	3
2. ANTECEDENTES	3
3. PROBLEMATIZACIÓN.....	6
4. JUSTIFICACIÓN	6
4.1 Justificación Teórica.....	6
4.2 Justificación Práctica.....	8
5. OBJETIVOS	10
5.1. Objetivo General	10
5.2. Objetivos Específicos	10
6. HIPÓTESIS	10
7. MÉTODOS Y TÉCNICAS.....	11
7.1. Métodos	11
7.2. Técnicas.....	11
CAPÍTULO II	13
MARCO TEÓRICO.....	13

2.1. INTRODUCCIÓN A LOS DISPOSITIVOS MÓVILES DE ÚLTIMA GENERACIÓN.....	13
2.1.1. Smartphone. Características y funciones	13
2.1.2. Historia del Smartphone	15
2.1.3. La industria de los Smartphone	16
2.1.4. Industria de los Smartphone	18
2.2. SENSORES PARA TELÉFONOS INTELIGENTES	23
2.3. LIMITACIONES	31
CAPÍTULO III	32
3. ESTUDIO DE LAS PLATAFORMAS ANDROID Y WINDOWS PHONE PARA EL DESARROLLO DE LOS PROTOTIPOS	32
3.1. INTRODUCCIÓN	32
3.2. ESTUDIO DE LAS PLATAFORMAS DE LOS SMARTPHONES.....	33
3.3 ANÁLISIS DE LAS PLATAFORMAS SELECCIONADAS	35
3.3.1. ANDROID	35
3.3.1.1. CARACTERÍSTICAS	36
3.3.1.2. RESTRICCIONES	37
3.3.1.3. VERSIONES DE LA PLATAFORMA	38
3.3.1.4. ARQUITECTURA.....	42
3.3.1.5. SDK	43
3.3.1.6. SISTEMAS OPERATIVOS	46
3.3.1.7. LENGUAJES DE PROGRAMACIÓN.....	46
3.3.1.8. ENTORNOS DE PROGRAMACIÓN.....	46

3.3.1.9. DISTRIBUCIÓN DE APLICACIONES.....	47
3.3.1.10. TIENDA DE APLICACIONES	48
3.3.1.11. INSTALACIÓN DE ECLIPSE CON EL SDK DE ANDROID.....	48
3.3.2. WINDOWS PHONE	49
3.3.2.1. CARACTERÍSTICAS	51
3.3.2.2. RESTRICCIONES	52
3.3.2.3. VERSIONES DE LA PLATAFORMA	52
3.3.2.4. ARQUITECTURA.....	55
3.3.2.5. SDK	56
3.3.2.6. SISTEMAS OPERATIVOS	58
3.3.2.7. LENGUAJES DE PROGRAMACIÓN.....	59
3.3.2.8. ENTORNOS DE PROGRAMACIÓN.....	59
3.3.2.9. DISTRIBUCIÓN DE APLICACIONES.....	60
3.3.2.10. TIENDA DE APLICACIONES	62
3.3.2.11. INSTALACIÓN DE VISUAL STUDIO CON EL SDK DE WINDOWS PHONE 8.....	63
CAPÍTULO IV.....	69
METODOLOGIA DE DESARROLLO: METODOLOGIA XP	69
4.1 FASE DE PLANIFICACIÓN.....	70
4.1.1. Historias de usuario	70
4.1.2. Iteraciones	71
4.2 FASE DE DISEÑO.....	72
4.2.1 Riesgos.....	73

4.2.2 Tarjetas C.R.C.	76
4.3 FASE DE CODIFICACIÓN	79
4.4 FASE DE PRUEBAS	80
CAPÍTULO V	86
5.1 AMBIENTE DE PRUEBAS	86
5.2 DETERMINACION DE PARAMETROS PARA EL ESTUDIO ESTADISTICO COMPARATIVO	87
5.3 Análisis Comparativo	90
5.4. Comprobación de la Hipótesis	109
Conclusiones.....	112
Recomendaciones.....	114
ABSTRACT	¡Error! Marcador no definido.
CAPÍTULO VI	118
ANEXOS	124

ÍNDICE DE FIGURAS

Capítulo I

Figura I. 1 . Escenario del prototipo.....	9
--	---

Capítulo II

Figura II 1 Samsung Galaxy S4.....	19
------------------------------------	----

Figura II 2 Partes de un Smartphone “i Phone5”	20
--	----

Figura II 3 . Partes de un Smartphone Huawei Ascend P6	21
--	----

Figura II 4 BlackBerry Z30.....	22
---------------------------------	----

Figura II 5 Nokia Lumia 1520	23
------------------------------------	----

Figura II 6 Acelerómetro	24
--------------------------------	----

Figura II 7 Ejes del Acelerómetro	25
---	----

Figura II 8 Magnetómetro	27
--------------------------------	----

Figura II 9 Giroscopio	27
------------------------------	----

Figura II 10 Sensor de Luz	28
----------------------------------	----

Figura II 11 Sensor de Proximidad	29
---	----

Figura II 12 Sensor de Gestos.....	29
------------------------------------	----

Capítulo III

Figura III 1 Tipo de Sistemas Operativos de los Smartphone	34
--	----

Figura III 2 Arquitectura de la plataforma de Android	42
---	----

Figura III 3 Pantalla Windows Phone	51
---	----

Figura III 4 Arquitectura de Windows Phone	56
--	----

Figura III 5 Modelo de Aplicación. Conceptos	61
--	----

Figura III 6 Ciclo de vida de la aplicación	61
Figura III 7 Aislamiento y Ejecución de Aplicaciones	62
Figura III 8 Pantalla de instalación de visual Studio Ultimate	64
Figura III 9 Segunda Pantalla de instalación de visual Studio Ultimate	65
Figura III 10 Tercera Pantalla de instalación de visual Studio Ultimate	65
Figura III 11 Pantalla de Activación de Hyper - V	66
Figura III 12 Cuarta Pantalla de instalación de visual Studio Ultimate	67

Capítulo IV

Figura IV 1 Fases de la Metodología XP	70
Figura IV 2 Diagrama de componentes	78
Figura IV 3 Arquitectura física de la aplicación.....	78
Figura IV 4 Diagrama de Base de Datos	79
Figura IV 5 Modelo de Base de Datos.....	79
Figura IV 6 Pantalla sin datos de la aplicación en Android.....	81
Figura IV 7 Pantalla con datos de la aplicación en Android	81
Figura IV 8 Mensaje en la aplicación de Android	82
Figura IV 9 Pantalla sin datos de la aplicación en Windows Phone	82
Figura IV 10 Pantalla con datos de la aplicación en Windows Phone	83
Figura IV 11 Mensaje en la aplicación Windows Phone	83
Figura IV 12 Tiempo hasta la primera lectura en Android y Windows Phone	85

Capítulo V

Figura V 1 Representación de Resultados Parámetro 1: Codificación.....	96
--	----

Figura V 2 Representación de Resultados Parámetro 2: Lenguajes de Programación	99
Figura V 3 Representación de Resultados Parámetro 3: Instalación	101
Figura V 4 Representación de Resultados Parámetro 4: Rendimiento	105
Figura V 5 Diagrama General de Resultados.....	107
Figura V 6 Resultados Finales	110

ÍNDICE DE TABLAS

Capítulo II

Tabla II I Historia de los Smartphone.	15
---	----

Capítulo III

Tabla III I Versiones de la plataforma Android.....	38
---	----

Tabla III II Características de las Herramientas del SDK de Android	44
---	----

Tabla III III Versiones de Windows Phone 7	52
--	----

Tabla III IV Versiones de Windows Phone 8	54
---	----

Capítulo IV

Tabla IV I Tarea de Ingeniería - Definición de requerimientos	71
---	----

Tabla IV II Plan de Iteración No.1	72
--	----

Tabla IV III Plan de Iteración No.2.....	72
--	----

Tabla IV IV Plan de Iteración No.3	72
--	----

Tabla IV V Identificación de Riesgos.....	73
---	----

Tabla IV VI Criterios de valoración de la probabilidad de riesgos	73
---	----

Tabla IV VII Criterio de valoración del impacto de riesgos	74
--	----

Tabla IV VIII Determinación de exposición al riesgo	74
---	----

Tabla IV IX Valoración de determinación de exposición al riesgo	74
---	----

Tabla IV X Impacto Esperados.....	75
-----------------------------------	----

Tabla IV XI Priorización de los Riesgos	75
---	----

Tabla IV XII Hoja de Gestión de Riesgo 1	75
--	----

Tabla IV XIII Tarjeta CRC - Tabla sismo	77
---	----

Capítulo V

Tabla V I Parámetros para el Estudio Estadísticos.....	87
Tabla V II Indicadores de los parámetros de codificación	88
Tabla V III Indicadores del de Lenguaje de Programación.....	89
Tabla V IV Indicador de la instalación	89
Tabla V V Indicadores del Rendimiento	89
Tabla V VI Valoración de porcentajes	90
Tabla V VII Tabla de Valores Cualitativos y Cuantitativos del Indicador 1.1	92
Tabla V VIII Tabla de Valores Cualitativos y Cuantitativos del Indicador 1.2	92
Tabla V IX Escala de Valores del Indicador 1.3	93
Tabla V X Valorización de los indicadores del Parámetro Codificación	93
Tabla V XI Escala de Valores del Indicador 2.1	96
Tabla V XII Escala de valores del Indicador 2.2.....	97
Tabla V XIII Valorización de los indicadores del Parámetro Lenguajes de programación	97
Tabla V XIV Valorización de los indicadores del Parámetro Instalación	99
Tabla V XV Escala de valores del Indicador 3.1.....	101
Tabla V XVI Escala de valores del Indicador 3.2.....	102
Tabla V XVII Escala de valores del Indicador 3.3.....	102
Tabla V XVIII Escala de valores del Indicador 3.4.....	103
Tabla V XIX Valorización de los indicadores del Parámetro Rendimiento	103
Tabla V XX Parámetros Evaluados	106
Tabla V XXI Resultados en Porcentajes	107

Tabla V XXII Cuadro de Resultados Finales	109
Tabla V XXIII Valores y Porcentajes Finales	110

INTRODUCCIÓN

Hoy en día la tecnología ha ido avanzando tanto en hardware como el software y podemos ver que los usuarios se van adaptando a esta transformación y es que ellos son los que participan activamente en este proceso de cambio.

Los teléfonos móviles de última generación son dispositivos capaces de realizar conexiones web, y otros servicios de información inalámbricos. Transportan información a través de las redes de datos inalámbricas de empresas de telefonía móvil.

Una característica importante de los teléfonos inteligentes es que en la actualidad vienen incorporados con nuevas y actuales tecnologías como los sensores los cuales permiten la captura de los datos para después ser procesar y utilizar la información; para demostrar el uso de los mismos se ha desarrollado dos aplicaciones que utilizan los sensores de un teléfono inteligente con sistema operativo Android y otro con Windows Phone, estas aplicaciones permiten recolectar los datos de un sismo por medio de los sensores y emitir una alarma para el usuario acerca del movimiento sísmico.

El presente trabajo de investigación se ha dividido en capítulos donde el primero contiene el anteproyecto de tesis el cual ha servido como referencia para iniciar en este trabajo; el segundo capítulo trata acerca de los conceptos generales de

los términos que han sido de gran relevancia y poder aplicarlos en la creación de los componentes personalizados; el tercer capítulo trata acerca de los prototipos, las herramientas que se utilizó de software y hardware; el cuarto capítulo trata acerca de la metodología que se utilizó en el desarrollo de los prototipos y finalmente el quinto capítulo trata acerca de las pruebas realizadas para la comprobación de la hipótesis, identificación de variables, definición de indicadores.

Para la comprobación de la hipótesis se ha utilizado el método científico mediante el cual se ha avalado este trabajo de investigación, mediante este se ha obtenido que al utilizar los sensores usando Windows Phone 8.0 y un sensor en Android 4.0.4 se puede tener una mayor precisión en la captura de los datos con Windows Phone.

CAPÍTULO I

MARCO REFERENCIAL

1. TITULO DEL PROYECTO DE TESIS

Estudio estadístico comparativo entre sensores Android y Windows Phone aplicado en la detección de movimientos telúricos.

2. ANTECEDENTES

Si se piensa en algún dispositivo móvil, lo primero en lo que se viene a la mente es en un teléfono móvil, y como mucho en una PDA. Pero en la actualidad son varios los dispositivos móviles disponibles en el mercado. Este es el caso de los PCs portátiles, de los teléfonos con acceso a Internet, PocketPC's, etc.

Este hecho da lugar a una importante problemática para quien programa tales dispositivos, ya que cada uno de ellos tiene unas características particulares: cada uno dispone de una memoria determinada o ha de soportar un lenguaje y un entorno específicos.¹¹

El problema de los programadores de estos dispositivos reside, principalmente, en el momento en el que quieren estandarizar todos los dispositivos. Esto es muy difícil, ya que cada uno de ellos tiene unas aplicaciones diferentes, está implementado en un lenguaje distinto y utiliza técnicas de desarrollo y sistemas operativos dispares.

Los teléfonos inteligentes incluyen funciones intuitivas incorporadas que pueden requerir el uso de un giroscopio integrado o un acelerómetro. Un giroscopio te permite cambiar la orientación del dispositivo haciendo girar su pantalla. Un acelerómetro, por otra parte, rastrea la aceleración o percibe la vibración del dispositivo. Un giroscopio y un acelerómetro incorporado funcionan en combinación con el sistema operativo de un teléfono inteligente o con aplicaciones de software específicas para realizar estas y otras funciones.

Un acelerómetro es un elemento sensor que mide la aceleración, así como el ángulo de inclinación, la rotación, la vibración, el choque y la gravedad.

Para ofrecer funcionalidad en un teléfono inteligente, el software del acelerómetro debe traducir los datos proporcionados por el sensor. Los

teléfonos inteligentes utilizan varios tipos de acelerómetros, siendo el elemento sensor y el software las principales diferencias entre ellos.

Cuando se aplica a un teléfono inteligente, un acelerómetro puede cambiar automáticamente la orientación del dispositivo de vertical a horizontal.²¹

Aplicaciones Web Móviles

Una aplicación web móvil es una aplicación igual que la que podemos ejecutar en un PC de escritorio pero en el mundo móvil, con una versión más reducida del lenguaje.

Las características de las aplicaciones son:

- Accesibilidad a través de un navegador web, parecido a un navegador de PC de escritorio, pero con un código mucho más limitado.
- Las aplicaciones han de ser interactivas y dinámicas, ya que muchos móviles tienen el inconveniente de tener un teclado muy limitado y una pantalla reducida.
- La interfaz con el usuario ha de ser rica.
- El modelo de propagación por la red es el mismo que se utiliza para la web pero a través de la interfaz móvil.
- El móvil sólo se encarga de la interfaz gráfica con el usuario. Esto provoca que el dispositivo tenga capacidad nata de comunicación y siempre esté on-line para poder ejecutar una aplicación dinámica.
- La lógica de procesado pesado se realiza en el servidor y el resultado se manda al móvil que sólo se encarga de la interfaz gráfica.¹⁰

3. PROBLEMATIZACIÓN

Los teléfonos celulares de última generación poseen ciertas características las cuales permiten la captura y tratamiento de datos con diversas aplicaciones en distintos campos como: industriales, ambientales, entornos militares, etc.

Actualmente las personas buscamos aparatos electrónicos que sean fáciles de llevar por lo que nos vemos en la necesidad de crear dispositivos portables que faciliten nuestras tareas tanto en la casa como en nuestro trabajo para esto los dispositivos móviles son de gran utilidad ya que son aparatos pequeños que nos ayudan a crear aplicaciones en cualquier área pero en nuestra ciudad falta una aplicación que integre la utilización de un celular para la medición de movimientos telúricos para que pueda emitir una alarma a la persona que en ese momento cuenta con el teléfono inteligente

4. JUSTIFICACIÓN

4.1 Justificación Teórica

Hoy en día las características de los dispositivos móviles han hecho un mundo en continua evolución y desarrollo y está generando un mercado en expansión donde, en un futuro no muy lejano, se prevé un aumento de las aplicaciones. La mayor ventaja que supone la utilización de dispositivos móviles (DM's) es la eliminación del problema de cableado.

Este hecho hace que sea más cómodo el uso de los dispositivos porque no tenemos que depender de la existencia de una red eléctrica para su funcionamiento.

Con la nueva generación de dispositivos móviles, y una vez superados los problemas de estandarización, podremos acceder a la red en cualquier lugar y a cualquier hora. Esto supone otra de las grandes ventajas de esta tecnología, ya que era casi impensable hace unos años.

Los dispositivos móviles ofrece nuevas tecnologías, como la de poder enviar datos de diferentes clases ya sea voz, datos, imágenes o vídeo.

Esto conllevará a que el usuario será capaz en todo momento de enviar cualquiera de estas clases de datos siempre que lo desee, con la comodidad que supone el hecho de poder hacerlo desde el mismo dispositivo.

Bajo la denominación de sensores se engloba un conjunto de dispositivos con los que podremos obtener información del mundo exterior (en este conjunto no se incluye la cámara, el micrófono o el GPS). Los sensores se manipulan de forma homogénea. Son los dispositivos de entrada más novedosos que incorpora Android y con ellos podremos implementar formas atractivas de interacción con el usuario.

Los Smartphone actualmente llevan cada vez procesadores más avanzados y con mayor memoria RAM, también cuentan con sensores nuevos y mejorados.

Los cuales solo notamos en la descripción de la caja pero la mayoría del tiempo pasan desapercibidos hasta que comienzan a fallar.

Windows Phone soporta múltiples sensores que permiten aplicaciones para determinar la orientación y el movimiento del dispositivo. Estos sensores permiten el desarrollo de aplicaciones en donde el dispositivo físico en sí es un medio de entrada del usuario. Los usos típicos para esta funcionalidad incluyen juegos con control de movimiento y aplicaciones de realidad aumentada. Para muchos de estos fines, la API de movimiento que combina y procesa información proveniente de todos los sensores, es la forma más sencilla para obtener el movimiento y la información de orientación. Para aplicaciones que lo requieran, Windows Phone también proporciona Apis para recuperar datos de los sensores individuales. En este tema se proporciona una breve descripción de cada API sensor.

4.2 Justificación Práctica

Los teléfonos móviles de última generación son dispositivos capaces de realizar conexiones web, y otros servicios de información inalámbricos. Transportan información a través de las redes de datos inalámbricas de empresas de telefonía móvil.

Normalmente, la actividad de medir parámetros acerca de un movimiento telúrico se da de manera periódica o esporádica, la mayoría de las mediciones

de los fenómenos captados en el medio ambiente tal como la intensidad del sismo, la ciudad, la hora.

Se crea dos prototipos, en el primero se captura los datos de un sismo, para lo cual se utiliza un dispositivo móvil (celular de última generación Samsung GALAXI SII), el cual posee un sensor Android que nos permite la captura de los datos. En el segundo prototipo se captura de la misma manera los datos a través un dispositivo móvil Windows Phone el cual posee un sensor Windows Phone. En los dos casos se almacenarán los datos y se emitirá una alerta desde el celular para la persona.

Debemos tener en cuenta que la aplicación se desarrollará en una plataforma Android 4.0.4 y Windows Phone 8 y en la construcción de una mesa la cual simulará un movimiento de sismo esta nos ayudara en nuestro escenario (ver Figura I. 1).



Figura I. 1 . Escenario del prototipo

Fuente: Propia de las autoras

5. OBJETIVOS

5.1. Objetivo General

Realizar el estudio estadístico comparativo entre sensores Android y Windows Phone para la detección de movimientos telúricos.

5.2. Objetivos Específicos

- Estudiar el funcionamiento de los sensores en los Smartphone Android y Windows Phone.
- Determinar los parámetros necesarios para realizar el estudio comparativo entre los sensores de un Smartphone de una aplicación Android y otra Windows Phone.
- Comparar estadísticamente el desarrollo de un prototipo Android y un prototipo Windows Phone.
- Desarrollar una aplicación que detecte los movimientos telúricos para un dispositivo móvil.

6. HIPÓTESIS

Realizar un estudio estadístico comparativo entre los sensores de una aplicación Android y otra Windows Phone, permitirá escoger la mejor para desarrollo de una aplicación para la detección de movimientos telúricos.

7. MÉTODOS Y TÉCNICAS

7.1. Métodos

El método que se utilizara como referencia para la investigación es el método científico por lo que se ha establecido los siguientes puntos:

- El planteamiento del problema que en este caso acerca de los sensores Android y Windows Phone.
- El apoyo del proceso previo a la formulación de la hipótesis.
- Definición de las variables que permitirán el estudio estadístico comparativo de los sensores Android y Windows Phone.
- Análisis e interpretación de resultados.
- Proceso de comprobación de la hipótesis se utilizará estadística descriptiva
- Para el proceso de desarrollo del prototipo se utilizará la metodología de desarrollo XP.

7.2. Técnicas

Para recopilar la información que respalde el trabajo de investigación, se ha establecido las siguientes técnicas.

- Revisión de información en línea acerca de los dispositivos móviles de última generación.
- Revisión de artículos científicos acerca los sensores Android y Windows Phone.
- Observación

- La metodología a utilizar es Programación Extrema (XP) ya que en nuestro caso esta metodología se ajusta a nuestras necesidades y es la más óptima al momento del desarrollo de una aplicación por su flexibilidad a los cambios preocupándose por el aprendizaje de los desarrolladores, y un buen ambiente de trabajo.

CAPÍTULO II

MARCO TEÓRICO

2.1.INTRODUCCIÓN A LOS DISPOSITIVOS MÓVILES DE ÚLTIMA GENERACIÓN

2.1.1. Smartphone. Características y funciones

Un Smartphone o teléfonos inteligentes son dispositivos móviles que brindan mayores funcionalidades que un celular convencional, ya que presentan características y funcionalidades que permite trabajar como si estuvieras operando en una PC2.

Una de las principales características es que presentan un Sistema Operativo que permite instalar aplicaciones para facilitar el trabajo.

Algunas de los Sistemas Operativos más usados en teléfonos inteligentes actualmente son: ⁸

- Android: creado por Google.

- Windows Phone: creado por Microsoft.
- iPhone OS: creado por Apple.
- BlackBerry OS: creado por BlackBerry

Estos Sistemas Operativos se desarrollan para cada tipo de Smartphone, dependiendo de su fabricante y marca, teniendo en cuenta que cada uno brinda múltiples funcionalidades que los hacen estar en los primeros lugares del mercado mundial. Esto no quiere decir que las aplicaciones usadas por los dispositivos sean solo las instaladas en el tiempo de fabricación, ya que permite la instalación de aplicaciones desarrolladas por terceros, y no solo por sus fabricantes.

Son múltiples las aplicaciones soportadas por estos dispositivos, como son las hojas de cálculos que permite usarse en el entorno empresarial ya sea para actividades contables o de otro tipo. Presentan editores de textos como Word y Pdf que permiten, además, la lectura de los documentos.

Otra de las principales funciones de los Smartphone es la conexión a internet. Actualmente no se concibe un dispositivo móvil de última generación sin Wifi ni acceso a internet. También cuentan con Bluetooth.

Además cuentan con una pantalla táctil que facilita el manejo del dispositivo y la entrada de información. Presenta facilidades para la gestión de contactos, libreta de direcciones, gestión de la cuenta de correo, etc. Permite la reproducción de fotos y gran variedad de ficheros de música y videos en alta resolución, simultáneamente, sin que esto afecte la otra actividad que estás realizando. Presentan cámaras de buena o media resolución, dependiendo del

modelo. Por supuesto, igual que todos los teléfonos celulares permite el recibo y envió de llamadas y mensajes de textos.¹⁶

2.1.2. Historia del Smartphone

Los Smartphone desde que aparecieron por primera vez han pasado más de 20 años aquí presentamos una tabla en la que se describe cómo ha ido evolucionando los Smartphone con algunas de sus características principales.

Tabla II | Historia de los Smartphone.

Nombre	Año de lanzamiento	Características	Empresa que lo creó
Simón	1992	<ul style="list-style-type: none">• No tenía botones ni teclas.• Era un teléfono táctil.• Sus funcionalidades eran de envió de mensajes de textos, auto completamiento de palabras, conexión a internet a través de un modem, beeper, acceso a correo, agenda, envió y recepción de fax.• Las aplicaciones que tenía era para la gestión de direcciones, almanaque, reloj mundial, calculadoras, bloc de notas.• El tamaño era muy grande.	IBM (International Business Machines)
Palm Pilot	1996	<ul style="list-style-type: none">• Permitía el traslado de la información de un lugar a otro.• El Pilot 1000 contaba un procesador de 16MHz y una memoria de 128KB.•	Palm Computing
Nokia 9110 Communicator	1998	<ul style="list-style-type: none">• Poseía un teclado deslizable QWERTY.• Su pantalla no era color	Nokia's Communicator series
BlackBerry 5810	2002	<ul style="list-style-type: none">• capacidad de revisar correos electrónicos y navegar por Internet.• Era necesario utilizar auriculares.	RIM (Research In Motion).

Nombre	Año de lanzamiento	Características	Empresa que lo creó
Palm Treo 600	2003	<ul style="list-style-type: none"> • Soportar redes GSM y CDMA. • tenía 32MB de memoria RAM. • Procesador de 144 MHz. 	Handspring
iPhone	2007	<ul style="list-style-type: none"> • pantalla táctil. • Mejor experiencia en Internet 	Apple
Droid	2009	<ul style="list-style-type: none"> • Utiliza la plataforma Android. 	Motorola
EVO 4G	2010	<ul style="list-style-type: none"> • se distingue por su enorme pantalla táctil de 4,3 pulgadas, • Que ofrece una resolución de 800 x 400 pixeles. • Utiliza el sistema operativo Android. 	HTC
Samsung Galaxy S II i9100	2011	<ul style="list-style-type: none"> • sistema operativo de Google, Android 2.3 Gingerbread • Una pantalla más grande a 4.3 pulgadas y tecnología Super AMOLED Plus. • Cámara de 8 megapíxeles y agrega flash LED y captura de video a 1080p gracias a un procesador de doble núcleo de 1.2GHz. 	Samsung
Nokia Lumia 720	2013	<ul style="list-style-type: none"> • Pantalla Nokia ClearBlack • Pantalla Gorilla Glass 2 • Soporte multitouch • Sensor acelerómetro para auto rotación • Sensor de proximidad para auto apagado • Sensor giroscópico • 8GB memoria interna, 512MB RAM • Procesador Qualcomm MSM8227 dual-core 1GHz, GPU Adreno 305 	Microsoft

Fuente: Propia de las autoras

2.1.3. La industria de los Smartphone

Los Smartphone cuentan con varios Sistemas Operativos, los cuales influyen en el comportamiento de estos. A continuación se presentara las características de los usados en este proyecto.

Windows Phone

Creado por Microsoft, sucesor de Windows Mobile, Windows Phone direccionado a un mercado de consumo, no cuenta con las características de su antecesor, por lo que este requiere de nuevos software para su ejecución. Cuenta con una interfaz que integra varios servicios Microsoft. A inicio de 2013 sale a la venta Windows Phone 8, esta versión busca hacerle competencia a iOS de Apple o Android de Google, por sus nuevas funcionalidades, y es incompatible con su antecesor.

Android

Es un Sistema Operativo de desarrollo gratis, flexible, y por lo tanto, económico; creado por Google sobre las plataformas Linux, JAVA y Apache, y dirigido a establecer una normalización y compatibilidad en el desarrollo de aplicaciones móviles.

iPhone OS

Creado por Apple, iPhone OS es un Sistema Operativo con poco tiempo en el mercado, pero con gran aceptación por las características de interfaz que presenta. Posee una interfaz fluida y rápida, basada en la manipulación directa, a través de gestos multitáctiles.

BlackBerry OS

Tiene su mayor uso en el sector empresarial, por las múltiples funcionalidades de correo que brinda, como son: redacción, envió y recibo de todo tipo de

mensajes electrónicos. Posee funciones de calendarios, administración de contactos, notas, tareas y direcciones, y conexión a la red.

2.1.4. Industria de los Smartphone

Existen en la actualidad diversas compañías y empresas que fabrican cada marca de Smartphone. Cada una de ellas, cada año saca nuevos y mejorados productos para competir en el difícil y evolutivo campo del comercio de Smartphone. Algunas de las compañías son:¹⁸

Samsung

Es la mayor compañía destacada actualmente por el gran papel que desempeña en el comercio de la telefonía móvil. Produce celulares con Sistemas Operativos de gran utilización como Android (creado por Google), Windows Phone (creado por Microsoft) y en futuro desea utilizar su propio Sistema Operativo, Tizen. Su éxito actual es el Smartphone Samsung Galaxy S4, muestra características que lo hace ser la preferencia de todo el mundo, y en los primeros lugares de la lista de los mejores celulares del 2013.

El Samsung Galaxy S4 constituye la cuarta generación de esta familia de celulares. Con una pantalla de 5 pulgadas y resolución de 1920 y 1080 pixeles. Procesador dual-core de 1,6 GHz. Rendimiento de 2 GB y capacidad de memoria de 16, 32 y hasta 64 GB. De 7,9 mm de ancho y 130 gramos de peso. Resolución de cámara de 13 megapíxeles. Lanzado al mercado en mayo de 2013.

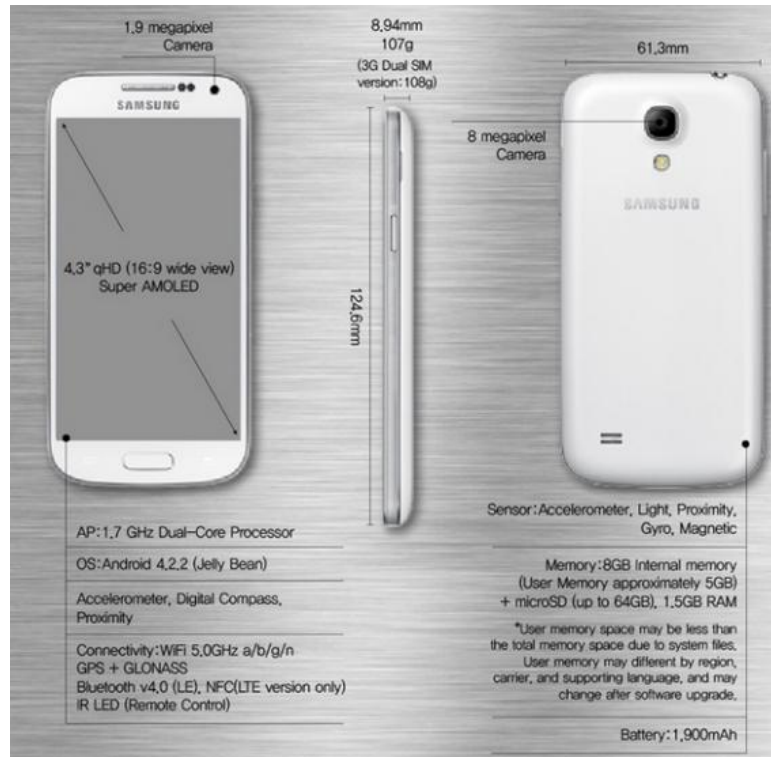


Figura II 1 Samsung Galaxy S4

Fuente: <http://blog.fnac.es/experto/smartphone-samsung-galaxy-s4-active/>

Apple

La compañía de la manzana ha sabido colocarse en los primeros lugares de la industria de Smartphone, con celulares como iPhone 4 y 4S es una de las más grandes empresas de telefonía móvil. Su más reciente creación, iPhone 5, tuvo gran aceptación, aunque ha empezado a disminuir su venta. En septiembre se espera el lanzamiento del iPhone 5S. Todos sus celulares presentan el Sistema Operativo creado por Apple, el iPhone OS.

iPhone 5 overview



Figura II 2 Partes de un Smartphone “i Phone5”

Fuente: <http://www.iclarified.com/24637/download-the-official-iphone-5-user-guide/comments>

Huawei

Es una empresa china que se ha situado en el tercer lugar dentro de las empresas de celulares. Con la fabricación de Smartphone de alta y media calidad que presentan Sistemas Operativos como Windows Phone y en un futuro Firefox OS. Su producto actual, Huawei Ascend P6, ha demostrado que es una empresa digna de ocupar el puesto otorgado, por las características y funcionalidades que posee y su acabado impecable.



Figura II 3 . Partes de un Smartphone Huawei Ascend P6

Fuente: <http://www.elandroidelibre.com/2013/06/huawei-ascend-p6-procesador-quad-core-pantalla-de-4-7-pulgadas-y-6-5-milímetros-de-grosor.html>

BlackBerry

Una de las empresas con más años en el mercado de la telefonía móvil, creadora del Sistema Operativo BlackBerry OS. Su creación más reciente el BlackBerry Z30 ha tenido gran acogida, con su nuevo diseño, que rompe con el estilo característico de esta empresa, el paso de teclado físico a pantalla táctil.

El BlackBerry Z30 sale al mercado en septiembre de 2013, después de algunos meses del lanzamiento del Z10, impulsando el camino de su fabricante, BlackBerry. Presenta el Sistema Operativo que los caracteriza, pero con nuevas mejoras y funcionalidades que rompe con lo que estamos acostumbrados a ver, y es la ausencia de teclas físicas. Con una pantalla táctil

de 5.0 pulgadas y 720 x 1280 píxeles. Procesador Qualcomm Snapdragon S4 Pro quad-core 1.7 GHz.

Rendimiento de 2 GB y memoria de 16 GB. Con un ancho de 9.4 mm y un peso de 137,5 gramos. Resolución de la cámara de 8 megapíxeles.



Figura II 4BlackBerry Z30

Fuente: <https://elpanaberry.wordpress.com/category/blackberry-z30/>

Nokia

Uno de los más antiguos en el mercado, su alianza con el Sistema Operativo de Microsoft, Windows Phone, ha demostrado muy buenos resultados. Su más reciente producto Nokia Lumia 1320 brinda funcionalidades mejoradas.

Esta empresa finlandesa tiene entre sus logros la fabricación de diversos celulares de las familias Lumia y Asha.

Es lanzado al mercado en octubre de 2013. Presenta una superficie de aluminio, con una cámara con mayor resolución. Cuenta con pantalla táctil OLED de 6.0 pulgadas de 1080 x 1920 píxeles. Rendimiento de 2GB de RAM y memoria de 32GB. Procesador quad-core 2.2GHz. 8,7mm de ancho y 209 gramos de peso. Cámara con resolución de 20 megapíxeles.



Figura II 5 Nokia Lumia 1520

Fuente: [http://www.siliconweek.com/actualidad/la-phablet-nokia-lumia-1520-ve-la-luz-en-chile-](http://www.siliconweek.com/actualidad/la-phablet-nokia-lumia-1520-ve-la-luz-en-chile-50462)

50462

2.2. SENSORES PARA TELÉFONOS INTELIGENTES

Los sensores de un dispositivo móvil son aquellos que permiten la interacción y obtención de datos con el exterior, estos no incluyen cámaras, micrófonos ni el Sistema de Posicionamiento Global. Entre los elementos que los componen se encuentran el acelerómetro, la orientación, luz, presión, campo magnético, giroscopio, proximidad y temperatura. Los sensores trabajan de forma

homogénea, lo que quiere decir que cada uno de ellos muestran valores escalares (x, y, z).⁹

Acelerómetro

El acelerómetro de un Smartphone es un chip de silicio puede ser de 2 ejes (X, Y) o 3 ejes (X, Y y Z) con su correspondiente parte electrónica capaz de procesar las señales.

Este tipo de acelerómetros usan una tecnología llamada MEMS y están basados en el traspaso térmico. Estos dispositivos miden la transferencia de calor causada por la aceleración. Gracias a que la masa de prueba en el diseño de estos acelerómetros son moléculas de gas, las partes mecánicas se pueden eliminar aligerando así su peso y su tamaño.⁵

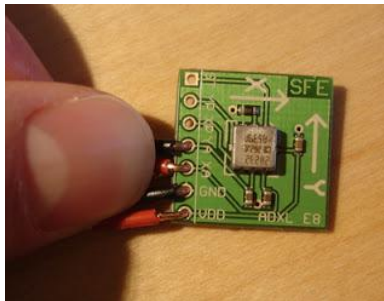


Figura II 6 Acelerómetro

Fuente: <http://www.poderpda.com/editorial/conociendo-los-principales-sensores-de-tu-smartphone/>

El acelerómetro mide las fuerzas aplicadas al dispositivo en un momento en el tiempo. Estas fuerzas se pueden utilizar para determinar en qué dirección se está moviendo el usuario del dispositivo. El valor de la aceleración se expresa como un vector 3-dimensional que representa los componentes de la aceleración en los ejes X, Y, y Z en unidades gravitacionales. La orientación

de la aceleración es con respecto al dispositivo de tal manera que $1g$ se aplica en el eje Z cuando el dispositivo está boca arriba sobre una mesa de nivel y $1g$ se aplica al eje Y cuando el dispositivo se coloca perpendicular a la mesa parte superior.

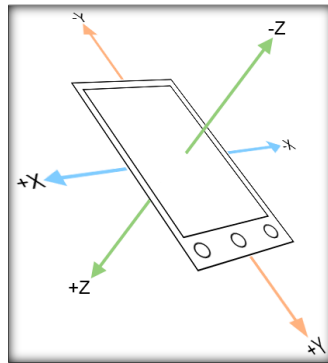


Figura II 7 Ejes del Acelerómetro

Fuente: <http://informmundo.blogspot.com/2013/04/los-sensores-con-los-que-cuenta-un.html>

El acelerómetro sensor detecta la fuerza de la gravedad, junto con cualesquiera fuerzas resultantes del movimiento del teléfono. El API movimiento combinado, consultado con el movimiento de clase, utiliza varios sensores del dispositivo para separar el vector de gravedad de la aceleración del dispositivo y le permite determinar fácilmente la actitud actual (guiñada, cabeceo y balanceo) del dispositivo.

Para un dispositivo móvil, un acelerómetro es un sensor de movimiento. Los A Smartphone en la actualidad tienen incorporada esta función por lo cual al mover el terminal la imagen de pantalla también se mueve. Así, cada vez que se mueva el teléfono de una posición la imagen se adapta a esta visión sea

esta vertical u horizontal; lo cual facilita la manipulación de imágenes y también para realizar la navegación por la web.

Este sensor nos resulta ser muy útil una vez aplicados a los juegos, porque permiten una interesante evolución, un ejemplo claro es: podemos conducir con el teléfono a modo de volante o guiar una bola por un laberinto como si fuera una caja real.

Otras aplicaciones prácticas de los sensores son la de responder a determinadas órdenes según lo que hagamos. Así, si entra una llamada o llevamos activo el reproductor y necesitamos silencio inmediato, basta con girar el dispositivo y se calla. También es factible programar una alarma y cuando suena apagarla sencillamente pasando la mano por encima (en este caso, el sensor va integrado en la cámara).⁴

Magnetómetro

La brújula, o magnetómetro, el sensor se puede utilizar para determinar el ángulo con el que el dispositivo es girado en relación con el polo norte magnético de la Tierra. Una aplicación también puede utilizar las lecturas del magnetómetro primas para detectar las fuerzas magnéticas alrededor del dispositivo.

Este sensor también puede trabajar con otros sensores MEMS, como acelerómetros o giroscopios de tres ejes para proporcionar soluciones con hasta nueve grados de libertad.

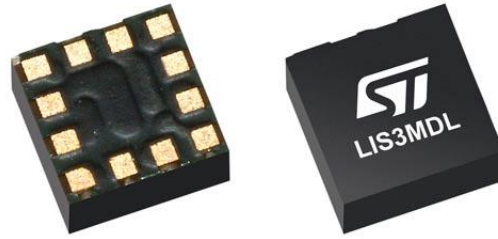


Figura II 8 Magnetómetro

Fuente: <http://www.tmbroadcast.es/index.php/magnetometro-monochip-de-tres-ejes-para-smartphones-y-dispositivos-de-navegacion/>

El sensor de la brújula no se requiere para todos los dispositivos Windows Phone. Es importante tener en cuenta esto al diseñar e implementar su aplicación.

Giroscopio

El giroscopio se encarga de medir el giro de un dispositivo en dirección diagonal gracias a la aceleración angular, algo de lo que no es capaz por sí solo el acelerómetro. Juntos, el acelerómetro y el giroscopio pueden detectar los cambios en la posición del dispositivo en 6 ejes.¹⁷



Figura II 9 Giroscopio

Fuente: <http://appliediario.com/2010/06/10/%C2%BFpara-que-sirve-el-girosco-pio-en-el-nuevo-iphone-4/>

El sensor de giroscopio no se requiere para todos los dispositivos Windows Phone. Es importante tener en cuenta esto al diseñar e implementar una aplicación. La aplicación siempre se debe comprobar para ver si el sensor está disponible y, o bien proporcionar un mecanismo alternativo de entrada.

Sensor de Luz

Este sensor detecta la intensidad de la luz del ambiente para ajustar el brillo de la pantalla, si se encuentra en un cuarto oscuro ajustara le brillo de la pantalla para que se pueda ver la información. Cuando nos encontremos con la luz fuerte ajustara los niveles lo que nos permita ahorrar la batería.

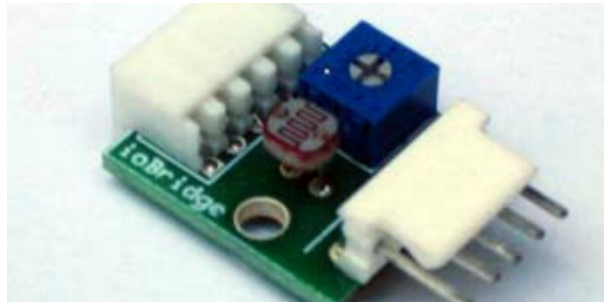


Figura II 10 Sensor de Luz

Fuente: <http://www.poderpda.com/editorial/conociendo-los-principales-sensores-de-tu-smartphone/>

Sensor de proximidad

Este sensor nos permite medir la distancia entre el equipo y nuestro cuerpo cuando acercamos el equipo a nuestro oreja para contestar una llamada el dispositivo apaga la luz de la pantalla para ahorrar la batería.

Si se llegara a tocar la pantalla por accidente mientras se está realizando una llamada, no pasa nada ya que el sensor de proximidad bloquea la pantalla para evitar errores, cuando alejamos el dispositivo de nuestro cuerpo automáticamente volverá a ser funcional la pantalla.



Figura II 11 Sensor de Proximidad

Fuente: <http://www.movilconsolas.com/flex-sensor-de-proximidad-smartphone-venom-iron-p-15303.html>

Sensor de gestos

Reconoce el movimiento de las manos usando rayos infrarrojos, del cual deriva las funciones Air Gesture.



Figura II 12 Sensor de Gestos

Fuente: <http://www.unlockmovil.com/auriculares-altavoces/flex-sensor-de-gestos-original-galaxy-s4-active-i9295-p-22819.html>

Sensor RGB Light

Sensor de luz que mezcla la intensidad del origen de la luz de los colores rojo, verde, azul y blanco.

Hall Sensor

Reconoce cuando la protección del teléfono esta puesta o no, usando las funciones S View Cover.

Temperature – Humidity Sensor

Determina los niveles de temperatura y humedad. Previene el sobrecalentamiento del teléfono apagándolo para evitar fallos.

Barometer

Es un sensor de presión que permite medir la presión atmosférica. Utiliza piezorresistencia.

Movimiento Combinado

Existen limitaciones físicas que pueden hacer que sea difícil determinar la orientación y el movimiento real de un dispositivo en los datos brutos de los sensores. Por ejemplo, las lecturas del acelerómetro incluyen la fuerza de gravedad aplicada al dispositivo, además de la fuerza resultante a partir del movimiento del dispositivo. El sensor del giroscopio mide la velocidad de rotación, no la posición, y por lo que está sujeto a la deriva. Además, hay

cálculos geométricos complejos que son necesarios para traducir los datos en bruto de estos sensores en la verdadera orientación del dispositivo. El Movimiento clase controla el cálculo sensor de bajo nivel y permite a las aplicaciones obtener fácilmente la actitud del equipo (guiñada, cabeceo y balanceo), la aceleración de rotación y aceleración lineal tanto por la gravedad y el movimiento del usuario. Para aplicaciones típicas que utilizan este tipo de datos, como aplicaciones de realidad aumentada, utilizando el movimiento de clase se recomienda. Si desea utilizar los datos del sensor de forma atípica o desea implementar su propio movimiento y cálculos de orientación, puede utilizar las clases de sensores individuales para obtener los datos brutos de cada sensor.

El API Motion puede utilizar dos configuraciones diferentes de sensores. Movimiento normal utiliza la brújula y el sensor de acelerómetro y es menos preciso que el movimiento mejorado, que utiliza la brújula, el acelerómetro y el giroscopio. Si su aplicación requiere la precisión de movimiento mejorada, usted debe comprobar para verificar que el dispositivo en el que se ejecuta la aplicación es compatible con el sensor de giroscopio.

2.3. LIMITACIONES

- **Acelerómetro:** Si se realiza un giro a velocidad constante sobre el eje z no se apreciaría ningún cambio en este sensor.
- **Sensor de campo Magnético:** Si el teléfono móvil se gira a favor de un eje imaginario de campo magnético no se aprecia ningún cambio en el sensor.

CAPÍTULO III

3. ESTUDIO DE LAS PLATAFORMAS ANDROID Y WINDOWS PHONE PARA EL DESARROLLO DE LOS PROTOTIPOS

3.1. INTRODUCCIÓN

La industria de los Smartphone es muy amplia y en el mercado se encuentra una gran gama de estos dispositivos, cada instante los teléfonos inteligentes tiene nuevas aplicaciones las cuales son de mucha ayuda para los usuarios, los Smartphone cuenta con un sistema operativo por lo cual es muy importante determinar de manera adecuada las plataformas que se van a utilizar en el desarrollo de los prototipos, se realizará una investigación de cada una de ellas para saber sus características.

Para determinar las plataformas en donde se desarrollaran los prototipos se tomará en cuenta las tendencias del mundo actual y el incremento de usuarios de Smartphone, así se podrá escoger las mejores plataformas y las que son

más utilizadas en el mercado por los usuarios referente a los teléfonos inteligentes.

3.2. ESTUDIO DE LAS PLATAFORMAS DE LOS SMARTPHONES

En la actualidad la elección de una plataforma Smartphone es uno de los grandes retos ya que se debe de encontrar una plataforma Smartphone que sea segura, fácil de configurar y administrar, y lo suficientemente flexible como para satisfacer las necesidades de las aplicaciones que se desea desarrollar.

También se debe analizar el estado actual del mercado móvil, y cuál es el nivel de penetración de cada sistema operativo.

Según un estudio por la empresa Gartner realizada en Agosto del 2013 las plataformas más utilizadas por los Smartphone son las siguientes: ¹⁴

- Android 79%
- iOS 14.2%
- Microsoft 3.3%
- BlackBerry 2.7%
- Bada 0.4%
- Sybian 0.3%
- Others 0.2%

Table 2
Worldwide Smartphone Sales to End Users by Operating System in 2Q13 (Thousands of Units)

Operating System	2Q13 Units	2Q13 Market Share (%)	2Q12 Units	2Q12 Market Share (%)
Android	177,898.2	79.0	98,664.0	64.2
iOS	31,899.7	14.2	28,935.0	18.8
Microsoft	7,407.6	3.3	4,039.1	2.6
BlackBerry	6,180.0	2.7	7,991.2	5.2
Bada	838.2	0.4	4,208.8	2.7
Symbian	630.8	0.3	9,071.5	5.9
Others	471.7	0.2	863.3	0.6
Total	225,326.2	100.0	153,772.9	100.0

Source: Gartner (August 2013)

Figura III 1 Tipo de Sistemas Operativos de los Smartphone

Fuente: <http://www.channelbiz.es/2013/08/14/samsung-smartphone-windows-blackberry/#>

Analizando la tabla los mejores sistemas operativos para desarrollar serian Android, iOS y Microsoft ya que son los más utilizados, pero se tomó en cuenta los siguientes factores para escoger los sistemas operativos.

- El primer sistema operativo que se escogió fue el Android ya que es de código abierto y lleva algunos años en el mercado y es el más utilizado por los usuarios como nos muestra el estudio de Strategy Analytics.¹⁵
- El segundo sistema operativo fue Windows Phone 8 por q es un sistema operativo respaldado por Microsoft y a pesar de estar poco tiempo en el mercado ha tenido gran aceptación por los usuarios de los Smartphone como publico la empresa Kantar Worldpanel a finales del 2013.¹³

3.3 ANÁLISIS DE LAS PLATAFORMAS SELECCIONADAS

A continuación se detallara cada una de las plataformas seleccionadas anteriormente.

3.3.1. ANDROID

Android es un Sistema Operativo en el cual su Software se basa en el núcleo de Linux. La idea fundamental por la cual se la creó desde un principio fue para dispositivos móviles, El sistema operativo Android controla dispositivos por medio de bibliotecas desarrolladas o adaptadas por Google a través del lenguaje de programación Java.

El S.O Android es una plataforma de código abierto. Esto significa, que cualquier programador o desarrollador puede crear aplicaciones con lenguaje de programación C u otros lenguajes, una vez desarrollada la aplicación se la envía a compilar y se la convierte a un formato propio para aplicaciones Android.

En un principio Android fue desarrollada por Google Inc. Tiempo después se unió Open Handset Alliance, una corporación de 48 compañías de Hardware, Software y telecomunicaciones, la idea desde un principio se basó en como promocionar los estándares de código abierto para este tipo de aplicaciones.

Google es la principal compañía que ha realizado la publicación del código fuente de Android mediante la licencia de Software Apache, que es una licencia de software libre y de código abierto para cualquier programador.

Android es una tecnología que facilita el trabajo a los desarrolladores, ya que se puede encontrar una amplia documentación oficial y no oficial. Esta plataforma funciona con cualquier dispositivo móvil inteligente, ya que es un sistema operativo completo que permite que sea compatible con un sin número de Smartphone. El sistema operativo Android tiene el privilegio de decir que es la primera plataforma verdaderamente abierta y completa para dispositivos móviles, tiene todo el sistema operativo para ejecutarse en un teléfono móvil pero sin los obstáculos de propiedad que tiene la innovación móvil.

Se puede decir que Android es una combinación de tres componentes:

- Un sistema libre, operativo de fuente abierta para dispositivos móviles.
- Una plataforma de desarrollo open-source para la creación de aplicaciones para dispositivos móviles, especialmente los dispositivos móviles, que ejecutan el sistema operativo Android.

3.3.1.1. CARACTERÍSTICAS

Android posee las siguientes características:

- Navegador integrado: basado en el motor de open Source Webkit.

- SQLite: base de datos para almacenamiento estructurado que se integra directamente con las aplicaciones.
- Multimedia: Soporte para medios con formatos comunes de audio, video e imágenes planas (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
- Máquina virtual Dalvik: Base de llamadas de instancias muy similar a Java.
- Telefonía GSM: dependiente del terminal.
- Bluetooth, EDGE, 3g y Wifi: dependiente del terminal.
- Cámara, GPS, brújula y acelerómetro: Dependiente del terminal
- Pantalla Táctil.⁷
- Máquina virtual de Java.
- Las aplicaciones escritas en Java pueden ser compiladas y ejecutadas en la máquina virtual de Dalvik, la cual es una especializada máquina virtual diseñada para uso en dispositivos móviles.
- Entorno de desarrollo (emulador, herramientas de depuración, perfiles de memoria y funcionamiento, plugin para Eclipse IDE).³

3.3.1.2. RESTRICCIONES

Para realizar una publicación de una aplicación en este sistema operativo es suficiente con pagar el valor de 25 dólares en Google CheckOut por una sola ocasión ya que por ser una plataforma de código abierto Android no posee ningún tipo de limitaciones.

3.3.1.3. VERSIONES DE LA PLATAFORMA

Existen diferentes versiones de Android que se detallan a continuación:

Tabla III I Versiones de la plataforma Android

VERSIÓN	DESCRIPCIÓN	FECHA LANZAMIENTO
Android 1.0 Nivel de API 1	<ul style="list-style-type: none"> • Primera versión de Android. • Nunca se utilizó comercialmente 	Septiembre 2008
Android Versión de API 3 Cupcake 1.5	<ul style="list-style-type: none"> • Teclado con predicción de texto • No posee teclado físico • Grabar audio y video avanzado • Aparecen los widgets y live folders • Soporte para Bluetooth estéreo • Las transiciones entre ventanas se realizan por animación • Basada en el kernel de Linux 2.6.27 	Abril de 2009
Android Versión Donut 1.6	<ul style="list-style-type: none"> • Búsqueda avanzada • Incorpora gestures y multi-touch • Síntesis de texto a voz • Soporta resolución de pantalla WVGA • Nuevo atributo XML • Mejoras en Android market • Mejora a la cámara Basada en el kernel de Linux 2.6.29 	Septiembre de 2009
Android 2.0/2.1 Nivel de API 5 Eclair	<ul style="list-style-type: none"> • API para manejar Bluetooth • Sincronizar • Mejora en la gestión de contactos • Ajustes a la cámara • Optimiza la velocidad del hardware • Mejor resolución de pantallas • Nueva interfaz del navegador • Soporte a HTML5 • Basada en el kernel de Linux 2.6.29 	Octubre 2009

VERSIÓN		DESCRIPCIÓN	FECHA LANZAMIENTO
Android 2.2 Froyo	Nivel de API 8	<ul style="list-style-type: none"> • Mejora de velocidad de las aplicaciones • Mejoras al navegador • Soporta flash 10.1 • Almacenamiento externo • Actualización automática de aplicaciones • Copia de seguridad • Mejora en el reconocimiento de voz • Mejora la conectividad • Mejora en la cámara • Basado en el kernel de Linux 2.6.32 	Mayo 2010
Android 2.3 Gingerbread	Nivel de API 9	<ul style="list-style-type: none"> • Mayor soporte en la resolución de pantallas • Nueva interfaz de usuario • Funcionalidad cortar, copiar y pegar • Capacidad multitáctil • Soporte nativo para varias cámaras • Reconocimiento facial • Gestión de la energía y control de aplicaciones • Soporte para la reproducción de video WebM/VP8 y codificación de audio AAC • Soporte en tecnología NFC • Soporte nativo para más sensores • Gestor de descargas • Basado en el kernel de Linux 2.6.25.7. 	Diciembre 2010
Android 3.0/3.1/3.2 Honeycomb	Nivel de API 11/12/13	<ul style="list-style-type: none"> • Disponía de un Escritorio 3D con widgets rediseñados, • Sistema multitarea mejorado. • Soporte para video chat mediante Google Talk. • Soporte a gran variedad de periféricos y conexiones USB como teclados, ratones. 	Febrero 2011

VERSIÓN	DESCRIPCIÓN	FECHA LANZAMIENTO
Android 3.0/3.1/3.2 Honeycomb Nivel de API 11/12/13	<ul style="list-style-type: none"> • Mejor soporte para redes Wi-Fi. • Basado en el kernel de Linux 2.6.36 	Febrero 2011
Android 4.0 Ice Cream Sandwich Nivel de API 14	<ul style="list-style-type: none"> • Se unifican las dos versiones anteriores (2.x para teléfonos y 3.x para tabletas) en una sola compatible con cualquier tipo de dispositivo. • Se introduce un nuevo interfaz de usuario totalmente renovado. • Nuevo API de reconocedor facial, permite entre otras muchas aplicaciones desbloquear el teléfono a su propietario. • También se mejora en el reconocimiento de voz. • Aparece un nuevo gestor de tráfico de datos por Internet. • Se mejora el API para comunicaciones por NFC y la integración con redes sociales. • Basada en el kernel de Linux 3.0.1 	Octubre 2011
Android 4.0.3 Nivel de API 15	<ul style="list-style-type: none"> • Redes sociales, calendario, revisor ortográfico, texto a voz y bases de datos entre otros. 	Diciembre 2011
Android 4.1 Jelly Bean Nivel de API 16	<ul style="list-style-type: none"> • Se incorporan varias técnicas, como: sincronismo vertical, triple búfer y aumentar la velocidad del procesador al tocar la pantalla. • Se mejoran las notificaciones con un sistema de información expandible personalizada. Los Widgets de escritorio pueden ajustar su tamaño y hacerse sitio de forma 	Julio 2012

VERSIÓN	DESCRIPCIÓN	FECHA LANZAMIENTO
Android 4.1 Jelly Bean Nivel de API 16	<ul style="list-style-type: none">• automática al situarlos en el escritorio. El dictado por voz puede realizarse sin conexión a Internet (de momento en inglés).• Basado en el kernel de Linux 3.1.10	Julio 2012
Android 4.2 Jelly Bean (Gummy Bear) Nivel de API 17	<ul style="list-style-type: none">• Los Widgets de escritorio pueden aparecer en la pantalla de bloqueo.• Se incorpora un nuevo teclado predictivo deslizante al estilo Swype.• Posibilidad de conectar dispositivo y TVHD mediante wifi (Miracast).• Mejoras menores en las notificaciones. Nueva aplicación de cámara que incorpora la funcionalidad Photo Sphere para hacer fotos panorámicas (en 360°). [28]	Noviembre 2012
Android 4.3 Jelly Bean (Gominola) Nivel de API 18	<ul style="list-style-type: none">• Soporte para Bluetooth de baja energía, mejora en la escritura, localización WiFi en segundo plano, Mejoras en seguridad.	Julio 2013
Android 4.4 KitKat Nivel de API 19	<ul style="list-style-type: none">• Agilización de componentes para la reducción de consumo de memoria.• Introducción de API's.• Mejoras en NFC.• Gestión de impresoras.• Integración de almacenamiento en la nube.• Caratulas en la pantalla de bloqueo.• Identificador de llamadas inteligentes.• Modo inmersivo a pantalla completa.• Mejoras de accesibilidad.	Noviembre 2013

Fuente: Propia de las autoras

3.3.1.4. ARQUITECTURA

Android usa Linux para los drivers de sus dispositivos, manejo de memoria, proceso de administración y la conectividad. Usualmente no se puede programar en esta capa directamente.

El siguiente nivel contiene las librerías nativas de Android. Todo esto se escribe en C/C++ internamente, pero puede ser llamado alrededor de una interfaz java.

La arquitectura interna de la plataforma Android, se compone por 4 componentes:

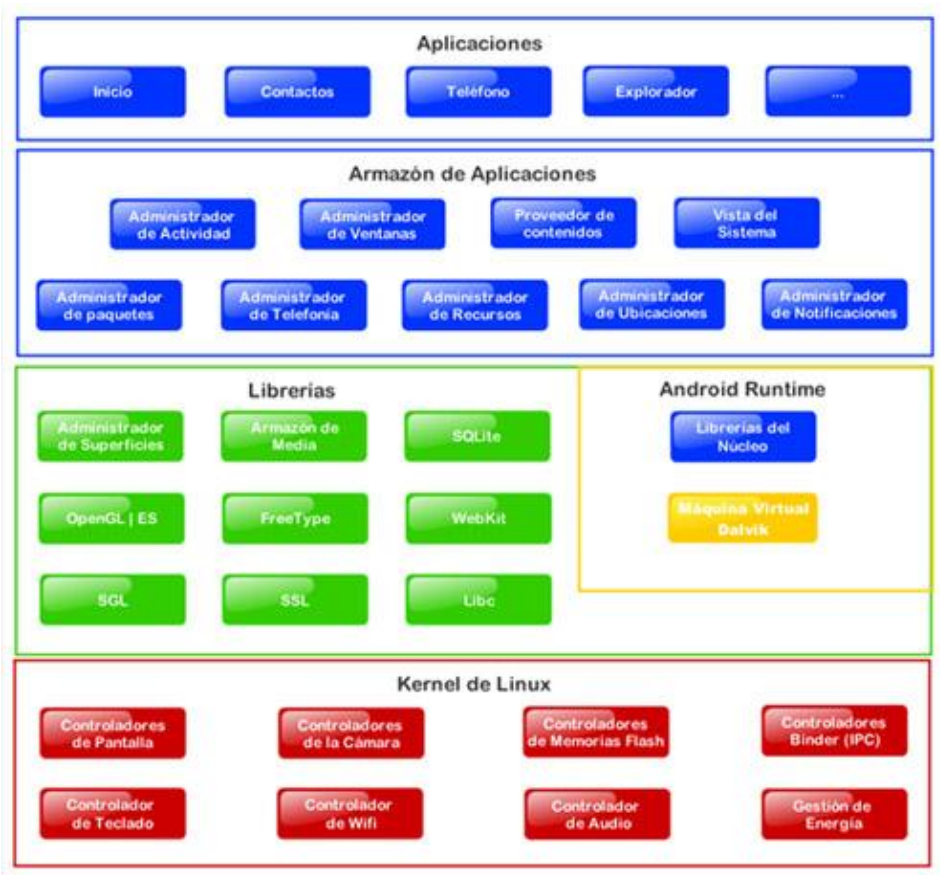


Figura III 2 Arquitectura de la plataforma de Android

Fuente: <http://www.configurarequipos.com/doc1107.html>

- **Aplicaciones:** las aplicaciones desarrolladas con la plataforma Android con tendrán como mínimo un cliente de email (correo electrónico), calendario, programa de SMS, mapas, navegador, contactos, y algunos otros servicios básicos.
- **Framework de aplicaciones:** Para las personas que se dedican a desarrollar aplicaciones para Android cuentan con acceso total al código fuente usado en las aplicaciones base o iniciales. Se ha adoptado esta medida para que no se creen un sin número de componentes de aplicaciones distintas, cuyo propósito sea el mismo, permitiendo de que los programas sean modificados, reemplazados o mejorados por cualquier usuario evitando que se realice desde cero una aplicación
- **Librerías:** El S.O. Android ha publicado sus librerías para que los programadores las utilicen, esto lo pueden hacer a través de su framework, estas han sido desarrolladas en C/C++; Android System C library son librerías de programación estándar, aquí algunas de ellas: librerías de medios, librerías de gráficos, 3D, SQLite, etc.
- **Runtime de Android:** Android agrega un gran número de librerías que ayudan para realizar las aplicaciones las cuales en su mayoría son basadas en el lenguaje de programación Java la. La Máquina Virtual cuenta con un compilador llamado dex (Dalvik Ejecutable) el cual se basa en registros, y corre clases compiladas por el compilador de Java.

3.3.1.5. SDK

El SDK de Android contiene un conjunto de herramientas de desarrollo de software las cuales son:

- Un debugger.
- Librerías.
- Un emulador.
- Documentación.
- Código de ejemplo.
- Tutoriales.

Además el SDK funciona en Sistemas Operativos Windows, Linux y Mac.

Cabe señalar que el SDK se ha ido publicando nuevas actualizaciones del SDK siendo la última la 18, la mejorando así algunos inconvenientes anteriores además se cuenta con nuevas funcionalidades y cambios en las APIs.

En cuanto a las herramientas del SDK se tiene:

Tabla III II Características de las Herramientas del SDK de Android

HERRAMIENTA	CARACTERISTICAS
Android Eclipse Views:	<p>Añade LogCat, un visor de logs del dispositivo que puede ser filtrado en 5 categorías:</p> <ul style="list-style-type: none">• V — Verbose (Prioridad más baja)• D — Debug• I — Info• W — Warning• E — Error• F — Fatal• S — Silent (Alta prioridad)
Emulador Console:	<ul style="list-style-type: none">• Contiene algunas funciones en tiempo real del emulador, como el envío de SMS,• Pueden ser accedidas usando la consola del emulador mediante telnet a local host y puerto, normalmente, 5554.
Android Debug Bridge (ADB):	<ul style="list-style-type: none">• Proporciona acceso a funciones del emulador, incluido el direccionamiento de puertos.

HERRAMIENTA	CARACTERISTICAS
Dalvik Debug Monitor Service (ddms):	<ul style="list-style-type: none"> • Herramienta integrada con la "Dalvik Virtual Machine". • Nos permite administrar los procesos que corren en una instancia de emulador/dispositivo y nos asiste en la depuración de ellos. • Podemos usar esta herramienta para terminar la ejecución de un proceso • Seleccionar un determinado proceso para depurar, generar reportes a partir de información de bitácoras, examinar el "heap" y la información de "thread, tomar fotografías de la pantalla del emulador/dispositivo y mucho más.
Android Debug Bridge (adb):	<ul style="list-style-type: none"> • Permite instalar nuestras aplicaciones (archivos ".apk") en una instancia de emulador/dispositivo. • Acceder a una instancia de emulador/dispositivo usando comandos de línea. • Se puede utilizar para enlazar un depurador estándar al código de una aplicación que esté corriendo en una instancia de emulador/dispositivo.
Android Asset Packaging Tool (aapt):	<ul style="list-style-type: none"> • Permite crear archivos ".apk", los cuales contienen las imágenes binarias de tu código y recursos de tus aplicaciones.
Android Interface Definition Language (aidl):	<ul style="list-style-type: none"> • Permite generar código para una interface de interproceso, tal como la que un servicio podría utilizar.
sqlite3:	<ul style="list-style-type: none"> • Esta herramienta ha sido incluida para comodidad de los desarrolladores. Nos provee acceso a los archivos de datos "SQLite" creados y usados por las aplicaciones Android.
Mksdcard:	<ul style="list-style-type: none"> • Esta herramienta nos ayuda a crear una imagen de disco que podemos usar con el emulador, para simular la presencia de una tarjeta de almacenamiento externa (tal como una tarjeta "SD").
Dx:	<ul style="list-style-type: none"> • La herramienta "dx" convierte los archivos de "bytecode"43 estándar (".class") en archivos "Android bytecode" (".dex").

Fuente: Propia de las autoras

3.3.1.6. SISTEMAS OPERATIVOS

El SDK para el desarrollo de aplicaciones en Android se encuentra disponible para los siguientes sistemas operativos:

- Linux (Todas las Distribuciones)
- Windows
- Mac OS X

3.3.1.7. LENGUAJES DE PROGRAMACIÓN

Actualmente la mayoría de las aplicaciones para Android están desarrolladas en Java y son ejecutadas en una máquina virtual diseñada para esta plataforma llamada Dalvik, el núcleo de Android está basado en el Kernel de Linux 2.6.

También se puede programar en Visual Basic este es más utilizado por las personas que recién empiezan en la programación, otro lenguaje es C# en visual Studio utilizando el SDK de Android, estos dos lenguajes no son gratuitos y sus licencias son un poco altas.

3.3.1.8. ENTORNOS DE PROGRAMACIÓN

Los IDE utilizados son NetBeans, Android Studio y Eclipse todos son gratuitos y de código abierto, la selección del IDE se basó en un estudio minucioso sobre las características que nos ofrecían cada uno como la estabilidad y compatibilidad en el entorno de desarrollo y el emulador o el terminal sin tener que realizar pasos intermedios.

- **NetBeans:** Es un entorno de desarrollo libre escrito en Java puede ser utilizado para otros lenguajes permite copilar, depurar y ejecutar programas.
- **Android Studio:** Es el nuevo entorno de desarrollo de Android, El IDE es un entorno de programación empaquetado como un programa de una aplicación.
- **Eclipse:** Es un entorno de desarrollo de código abierto multiplataforma que desarrolla aplicaciones de cliente enriquecido lo contrario a las aplicaciones de cliente liviano que se basa en navegadores.

3.3.1.9. DISTRIBUCIÓN DE APLICACIONES

Las aplicaciones se las puede distribuir de la siguiente manera:

- Instalando la aplicación (.apk) en cada uno de los dispositivos.
- Publicando en Google Play.

Pasos para la distribución de aplicaciones

- La primera vez que se accede al servicio se mostrará el asistente de nueva cuenta de desarrollador. Se debe de introducir los datos solicitados.¹²
- Se debe crear una cuenta como desarrolladores de Android para poder publicar aplicaciones deberemos abonar 25,00\$, es una cuota única, de por vida. Para pagar esta cuota se puede usar el servicio Google Checkout o para pagar la cuota con tarjeta.
- Una vez pagada la cuota, se puede subir la aplicación (.apk), se necesitan al menos 2 capturas de pantalla (8 capturas de pantalla como máximo por

tipo), más algunas imágenes de cómo es la aplicación y finalmente dar una descripción de la aplicación en los idiomas que se desee.

- Si además la aplicación está adaptada para tablets, se debe subir al menos una captura para tablets.

3.3.1.10. TIENDA DE APLICACIONES

Google Play en sus inicios fue llamada Android Market es una tienda de software en línea desarrollada por Google para los dispositivos Android su archivo ejecutable se llama Vending.apk. Permite gestionar la distribución de aplicaciones con características que le permiten controlar lo que los usuarios pueden descargar su aplicación, así como entregar versiones separadas de su aplicación sobre la base de ciertas características como versión de la plataforma.²⁴

En enero de 2012, Google Play disponía de más de 500.000 aplicaciones.

3.3.1.11. INSTALACIÓN DE ECLIPSE CON EL SDK DE ANDROID

Aquí se detallara las herramientas que se utilizó para la programación en Android. La instalación fueron realizadas en el entorno de desarrollo eclipse versión 4.2.1 (Juno).

Instalación de Eclipse Juno

En la página oficial de eclipse se encuentran todas las versiones de eclipse, cuando termine la descarga no es necesario instalar nada pero se debe realizar los siguientes pasos:¹⁹

1. Se debe descomprimir el archivo y copiar la carpeta en la unidad C.
2. Se da clic en el ejecutable y se abre la aplicación y se puede empezar a utilizar eclipse.

Instalación del SDK de Android

El SDK es un conjunto de herramientas que permite crear, depurar y probar aplicaciones para el sistema Android, consta con una interfaz de programación de aplicaciones o bibliotecas API que permiten usar un lenguaje de programación determinado ²².

Cuando se termine la descarga se procede a instalar la ubicación de los archivos no tiene importancia, pero se debe tener en cuenta la ubicación para utilizarlos posteriormente.

Para la aplicación que se va a desarrollar de debe instalar los siguientes API (16, 17, 18, 19).

3.3.2. WINDOWS PHONE

Windows Phone es un sistema operativo para dispositivos móviles. Este se crea después de Windows Mobile, dirigido al sector consumista, diferenciándolo de su padre, que va dirigido al sector de las empresas. La compañía que patrocina este sistema operativo es Microsoft. Como va dirigido al sector consumista presenta una interfaz más atractiva con soporte para varias aplicaciones como Skype, Xbox, etc. Es uno de los sistemas operativos más comerciales, compitiendo con Android, iOS, entre otros.

Windows Phone se encuentra embebido en el núcleo de Windows Embedded CE 6.0 ²³.

Windows Phone presenta una interfaz Metro, esto quiere decir que su pantalla está compuesta de mosaicos interactivos que permiten la apertura y visualización de diferentes aplicaciones o accesos directos, esto se denomina “Live Tiles”. Estos “accesos directos” muestran información de la aplicación correspondiente, actualizándose en tiempo real. Los Live Tiles se puede personalizar y configurar a gusto pulsando sobre ellos y distribuyéndolos o eliminándolos de la pantalla. Los Live Tiles pueden presentar información de contactos, redes sociales, llamadas telefónicas, emails, SMSs, chats, etc., como se muestra en la Figura III 3.



Figura III 3 Pantalla Windows Phone

Fuente: http://es.wikipedia.org/wiki/Windows_Phone

3.3.2.1. CARACTERÍSTICAS

Las características de los dispositivos móviles Windows Phone varía dependiendo el modelo entre las más importantes tenemos las siguientes:

- Presenta un teclado inteligente y táctil que permite la predicción de palabras, corrección automática de faltas de ortografía y guardar nuevas palabras.
- Cuenta con el Internet Explorer 10 como explorador web, con la posibilidad de guardar favoritos, marcadores, abrir varias pestañas, etc.
- El buscador por defecto es el Bing, y para realizar búsquedas cuenta con diferentes aplicaciones Bing como son: Bing Music (para buscar canciones), Bing Vision (para buscar libros) y Bing Local Search (para localizar sitios de interés). Cuenta con un botón exclusivo para búsqueda.
- Windows Phone utiliza un mecanismo de agrupación denominado Hubs, que se basa en organizar las aplicaciones según la función que realiza. Por lo que quedan agrupados por Hubs de contactos, fotos, trabajo, Entretenimiento, etc. De esta forma quedan las aplicaciones para visualizar y editar fotos agrupadas en un solo hubs, y así sucesivamente.
- Permite la centralización de los contactos de diferentes redes sociales, como son Facebook, Twitter, LinkedIn, además de los propios del teléfono. Facilitando la actualización de estados, comentarios de fotos, etc.

3.3.2.2. RESTRICCIONES

La principal restricción es que los programas que permiten desarrollar estas aplicaciones no son de código abierto, son propiedad de la compañía Microsoft.

3.3.2.3. VERSIONES DE LA PLATAFORMA

El sistema Windows Phone utiliza un proceso de actualización similar a todos los demás sistemas operativos Windows.

Haciendo uso del Microsoft Update se descarga e instala la actualización en su dispositivo móvil. Existen diferentes versiones de Windows Phone que se detallan a continuación:²⁶

Tabla III III Versiones de Windows Phone 7

Versión	Descripción
Windows Phone 7.0 ("Photon")	Versión inicial
Windows Phone 7.1 ("Nodo")	1ra actualización del sistema operativo. Permitía copiar y pegar, redujo el tiempo de encendido, aumento el rendimiento y ventajas en la actualización con Facebook.
Windows Phone 7.5 ("Mango")	Promocionada el 25-05-2011 y puesta en funcionamiento el 27-11-2011. Aportó más de 500 nuevas funcionalidades como multitarea, un nuevo procedimiento de búsqueda, integración con otras redes sociales como Twitter y LinkedIn, y aplicaciones

Versión	Descripción
Windows Phone 7.5 ("Mango")	como Office y SkyDrive; ventajas en el Sistema de Posicionamiento Global (GPS) y en el navegador Internet Explorer 9.
Windows Phone 7.5.1 ("Refresh")	Llamado también como Tango. Uno de los requerimientos de Nokia en su trato con Microsoft. Dirigido a reducir las características del sistema operativo para adecuarlo a dispositivos de bajo costo. Promocionado en el MWC 2012 de Barcelona, ofrece nuevas funcionalidades pero también restricciones (para los dispositivos de gama baja).
Windows Phone 7.8	Puesto en funcionamiento en enero del 2013 se entrega como versión suplente de Windows Phone 8. Ofrece ventajas como la nueva pantalla de usuario y fondos personificados para la imagen de bloqueo. Última versión de WP7, ya que Microsoft se focalizo en WP8.

Fuente: Propia de las autoras

Windows Phone 8

Windows Phone 8 es la última versión de WP, disponible solo para nuevos terminales móviles. Las nuevas funcionalidades que ofrece son:

- Nuevas interfaces de inicio y bloqueo, mejor configurables

- Control infantil: un área disponible y controlado para los niños, independiente del resto del móvil
- Data Sense: medidor de consumo de recursos del dispositivo móvil
- Cartera: para guardar tus tarjetas de fidelización y crédito
- Nuevo navegador: Internet Explorer 10
- Skype integrado
- Núcleo Windows NT, con soporte para procesadores de varios núcleos
- Tarjeta de memoria microSD
- Actualizaciones en el dispositivo móvil (sin usar el ordenador)
- Captura de pantalla

Actualizaciones de Windows Phone 8

Tabla III IV Versiones de Windows Phone 8

Versión de WP8	Descripción
Windows Phone 8 Update 1 (GDR1) ('Pórtico')	Permite conservar la Wi-Fi activada aunque se bloquee el dispositivo móvil y otras ventajas.
Windows Phone 8 Update 2 (GDR2)	Distribuido a finales de agosto de 2013. Contiene Radio FM (en todos los modelos excepto en los Nokia Lumia 620 y Samsung Ativ S), soporta los protocolos CardDAV y CalDAV, mejoras HTML5 en Internet Explorer 10. Los modelo Nokia obtiene la actualización Amber, que contiene la interfaz Glance (un reloj que se muestra con la pantalla bloqueada, al estilo de los Nokia con Symbian), las aplicaciones Nokia Smart Cam y Nokia Camera, ventajas en los algoritmos de las fotos, en la autonomía y la opción flip to silence (que rechaza una llamada colocando boca abajo el dispositivo móvil).

Versión de WP8	Descripción
Windows Phone 8 Update 3 (GDR3)	Empieza su despliegue en enero del 2014. Soporta procesadores quad-core, pantallas FullHD 1080p con una nueva fila de Live Tiles, modo de conducción, nuevas características de accesibilidad, ventajas en la opción "compartir conexión", tonos configurables, bloqueo de rotación de pantalla, cierre de aplicaciones desde la opción multitarea y ventajas en las conexiones Wi-Fi y Bluetooth. Los modelos Nokia obtienen la actualización Lumia Black. Contiene ventajas en la interfaz Glance y aplicaciones como Nokia Storyteller, Refocus y App Folder, para insertar carpetas en la pantalla de inicio.

Fuente: Propia de las autoras

3.3.2.4. ARQUITECTURA

La arquitectura de Windows Phone cuenta con 4 capas:

- **Modelo de Aplicaciones:** Las aplicaciones se ejecutan mediante un fichero .xap, el cual contiene los recursos validados necesarios para la ejecución de la aplicación.
- **Modelo de UI:** Es el modelo de interfaz de usuario, el cual se compone de elementos y una sesión es el grupo de interacciones que realiza un usuario sobre la aplicación e incluso puede involucrar otras aplicaciones.
- **Integración con la nube:** Dispone de integración con servicios como Exchange, Google Mail, Hotmail, Xbox Live, SkyDrive, Facebook, Twitter, LinkedIn, y Bing.

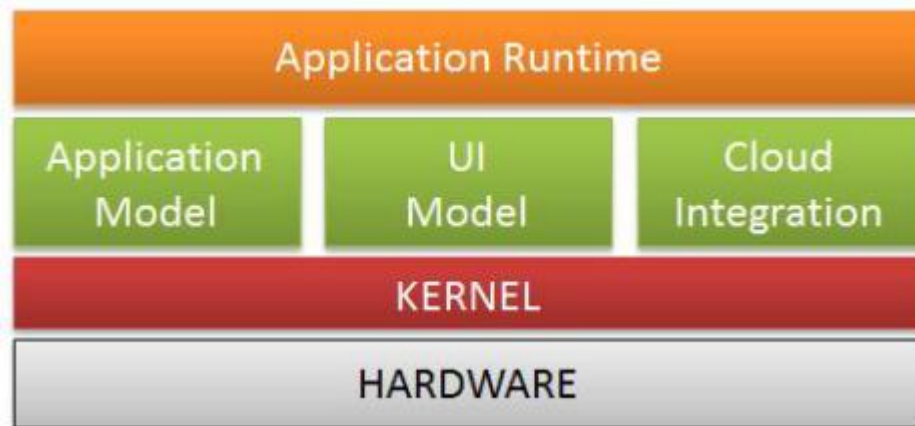


Figura III 4 Arquitectura de Windows Phone

Fuente: <http://tutocisc.bligoo.com/conocimientos-basicos-arquitectura-de-windows-phone-7>

3.3.2.5. SDK

En la actualidad existen 2 versiones del Kit de Desarrollo de Software (SDK) de Windows Phone. Estos son:²³

SDK 7.1

El SDK 7.1 de Windows Phone fue lanzado el 26 de septiembre del 2011 por la compañía Microsoft. Este permite el desarrollo aplicaciones y juegos para las versiones Windows Phone 7.0 y 7.5. Las características mínimas que debe tener el sistema para la ejecución del SDK son:

- Sistemas Operativos Windows 7 (x86 y x64), y Windows Vista (x86 y x64) con Service Pack 2, todas las ediciones excepto Starter Edition.
- Memoria RAM de 3GB o más.
- Espacio en HDD de 4 GB para instalar.

- Para la funcionalidad de Windows Phone Emulator necesita una tarjeta gráfica DirectX 10 o superior y un controlador WDDM 1.1.
- Es compatible con Visual Studio 2010 SP1.

El SDK 7.1 de Windows Phone provee las siguientes funcionalidades:

- Microsoft Visual Studio 2010 Express para Windows Phone
- Windows Phone Emulator
- Ensamblados de Windows Phone SDK 7.1
- Silverlight 4 SDK y DRT
- Extensiones de Windows Phone SDK 7.1 para XNA Game Studio 4.0
- Microsoft Expression Blend SDK para Windows Phone 7
- Microsoft Expression Blend SDK para Windows Phone OS 7.1
- Cliente de Servicios de datos de WCF para Windows Phone
- Microsoft Advertising SDK para Windows Phone

SDK 8.0

El SDK 8.0 de Windows Phone fue lanzado el 30 de octubre del 2012 por la compañía Microsoft. Este permite el desarrollo aplicaciones y juegos para las versiones Windows Phone 7.5 y 8. Las características mínimas que debe tener el sistema para la ejecución del SDK son:

- Sistemas Operativos Windows 8 y Windows 8 Pro. Versiones de cliente de Windows 8 de 64bits.
- Memoria RAM de 4GB o más.

- Espacio en HDD de 6.5 GB para instalar.
- CPU de 64 bits.
- Para la funcionalidad de Windows Phone Emulator requiere Windows 8 Pro Edition o versiones posteriores y un procesador que admita la traducción de direcciones de segundo nivel (SLAT)
- Si el sistema cumple con los requisitos de Sistema Operativo y Hardware pero no del Windows Phone Emulator, funcionara la ejecución del SDK pero no el Emulador.

El SDK 8 de Windows Phone provee las siguientes funcionalidades:

- Visual Studio Express 2012 para Windows Phone o puede funcionar como complemento de Visual Studio 2012 Professional, Premium o Ultimate.
- Windows Phone Emulator
- Usar sus conocimientos de programación y el código que ya tiene para generar aplicaciones administradas o con código nativo
- Contiene varios emuladores y herramientas adicionales para generar perfiles y probar la aplicación de Windows Phone en condiciones reales.

3.3.2.6. SISTEMAS OPERATIVOS

El SDK se encuentra disponible para los siguientes sistemas operativos:

- Windows Vista (x64 y x86) SP2
- Windows 7 (x64 y x86)
- Windows 8 x64

- Windows 8 Pro

3.3.2.7. LENGUAJES DE PROGRAMACIÓN

Las aplicaciones que se pueden encontrar en el Windows Phone Store son desarrolladas mediante:

- **Microsoft Silverlight**, la cual realiza aplicaciones usando XAML. Este software está compuesto por Microsoft .Net Compact Framework, que utiliza el CLR y la ejecución de código administrado tomado de la estructura .Net Framework. Contiene librerías que permiten la edición y visualización de ficheros, manejo de XML y gráficos, como la Base Class Library.
- **Microsoft XNA Framework**, la cual se utiliza para la creación de juegos. Es una implementación nativa de .Net Compact Framework. Facilita el control de dispositivos de entradas, gestión de audio y videos, utilización de modelos y texturas, utilización de ficheros sin importar el entorno de despliegue, realización de videojuegos on-line, todo esto mediante el uso de librerías.

3.3.2.8. ENTORNOS DE PROGRAMACIÓN

Microsoft Compact Framework

Es un framework de .Net que permite realizar aplicaciones para dispositivos móviles haciendo uso de librerías. Permite la programación en lenguajes C# o Visual Basic .Net. Tiene las siguientes características:

- Posee funciones como Common Language Runtime, Windows Communication Foundation, Language Integrated Query (LINQ), SoundPlayer, etc.
- Facilita la realización de interfaces a través del Windows Form, colocando tantos componentes visuales como se desee (botones, textos, ventanas, etc.).

Visual Studio .Net

Lenguaje de desarrollo dirigido a objetos que constituye un avance de Visual Basic usando como base el framework .Net. En sus inicios no era compatible con Visual Basic, por los cambios que se realizaron en este último, pero al tener instrucciones similares, .Net facilitó la implementación de aplicaciones más complejas con herramientas más actuales.

A medida que se fue mejorando el lenguaje Microsoft va proporcionando programas que cubran las nuevas necesidades.

3.3.2.9. DISTRIBUCIÓN DE APLICACIONES

La distribución de las aplicaciones y juegos de Windows Phone se realiza a través de su Marketplace.

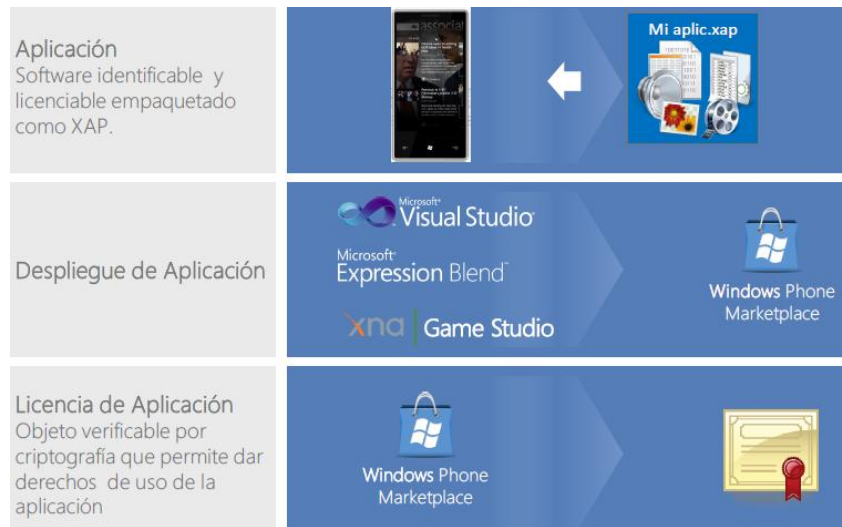


Figura III 5 Modelo de Aplicación. Conceptos

Fuente: Gómez, I., & Ortega, E. (2013). Windows Phone.

- El dispositivo móvil solo acepta paquetes .xap validados por Marketplace.
- Utiliza el manifiesto para la instalación del archivo .xap.
- Los usuarios pueden gestionar la instalación, actualización y desinstalación de la aplicación, y el Marketplace se encarga de la revocación.



Figura III 6 Ciclo de vida de la aplicación

Fuente: Gómez, I., & Ortega, E. (2013). Windows Phone.

- El dispositivo móvil solo ejecuta aplicaciones con una licencia válida de Marketplace.
- Las aplicaciones se ejecutan en un SandBox con cuentas de seguridad independientes durante la instalación y la ejecución.
- La política de asignación de recursos mantiene la aplicación activa viva.
- La política de gestión de recursos garantiza que el usuario pueda acceder a la pantalla de inicio para ejecutar otra aplicación.



Figura III 7 Aislamiento y Ejecución de Aplicaciones

Fuente: Gómez, I., & Ortega, E. (2013). Windows Phone.

3.3.2.10. TIENDA DE APLICACIONES

Windows Phone Store es la tienda de aplicaciones de todos los dispositivos móviles que presenten sistema operativo Windows Phone. Esta contiene diversas aplicaciones de juegos, reproducción de música y videos, redes sociales, descarga, trabajos, etc. que son supervisadas y probadas por la compañía Microsoft verificando que se cumpla con los requerimientos

necesarios. Actualmente cuenta con más de 215 mil aplicaciones y juegos. No permite aplicaciones con contenido sexual, que incite a la violencia, las drogas, fumar, etc.²³

3.3.2.11. INSTALACIÓN DE VISUAL STUDIO CON EL SDK DE WINDOWS PHONE 8

Aquí se describen los pasos necesarios para empezar a desarrollar la aplicación en Windows Phone. Los pasos de la instalación fueron desarrolladas en el entorno de Visual Studio 2012. El SDK de Windows Phone 8.0 que solo se lo puede instalar en una máquina que cuente con el sistema operativo de Windows 8 Pro de 64 bits, 4 GB de RAM y el procesador con soporte 'Second Level Address Translation' (SLAT) esto lo podremos comprobar aquí:

<http://j.mp/V8HqVK>

Instalación de Visual Studio 2012

La versión de Visual Studio 2012 que se utilizara fue la Ultimate para la instalación se ejecutó directamente de la página de Microsoft.

<http://www.microsoft.com/es-es/download/details.aspx?id=30678>

La instalación es muy sencilla y puede ser concretada en solo 3 pasos del asistente. El primero de ellos es la pantalla inicial, donde elegimos la ubicación

y debemos leer/aceptar los términos de usuario final. Hacemos clic en Next para continuar.

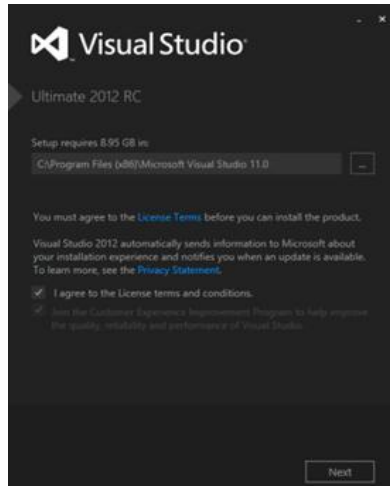


Figura III 8 Pantalla de instalación de visual Studio Ultimate

Fuente: Propia de las autoras

Luego se ofrecen las opciones de Lenguajes y Componentes. Mucho más fácil que VS2010, solo Nos ofrece componentes extra, entre los cuales se encuentran Blend, Silverlight y las herramientas Web. Recomendamos dejar marcadas todas las opciones. Hacemos clic en INSTALL para comenzar la instalación.



Figura III 9 Segunda Pantalla de instalación de visual Studio Ultimate

Fuente: Propia de las autoras

El proceso se inicia. Puede tardar entre 10 y 30 minutos como mínimo, al finalizar veremos una pantalla como esta. Damos clic en Launch y abrimos el Visual Studio 2012 por primera vez.

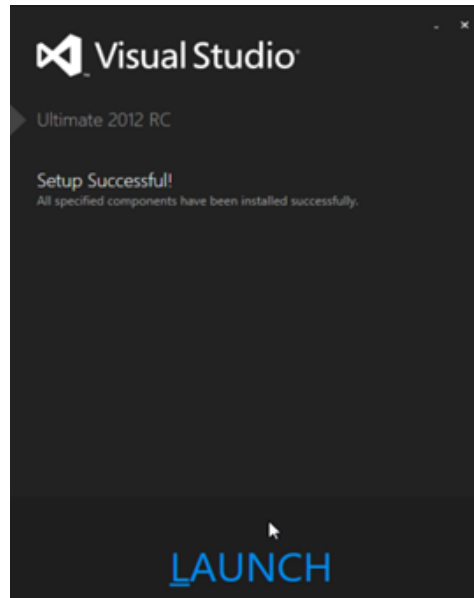


Figura III 10 Tercera Pantalla de instalación de visual Studio Ultimate

Fuente: Propia de las autoras

Una vez instalado Visual Studio seguimos con la configuración de Hyper V.

Activando Hyper V

Una vez que estemos seguros de cumplir con los requerimientos, procederemos a habilitar Hyper-V. Esto es necesario para poder virtualizar correctamente a Windows Phone 8.

Para ello buscaremos (con solo escribir) desde la pantalla de inicio “Agregar o desactivar características de Windows”, desde ahí habilitaremos a Hyper-V de la siguiente manera:

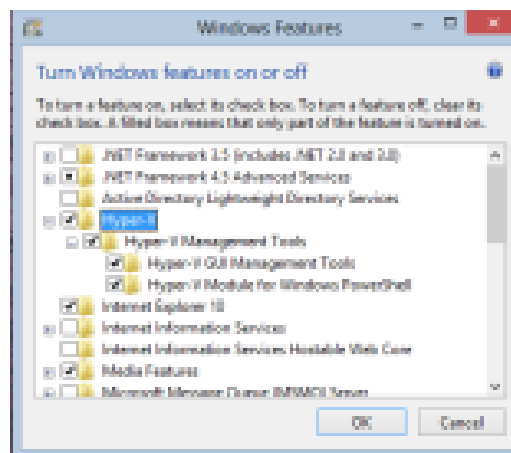


Figura III 11 Pantalla de Activación de Hyper - V

Fuente: Propia de las autoras

Se requerirá un reinicio para que los cambios se efectúen correctamente.

Descargar SDK de Windows Phone 8

Windows Phone 8.0 posee un SDK que nos proporciona un entorno de desarrollo completo que nos permite crear aplicaciones y juegos para Windows Phone 8.0 y Windows Phone 7.5. El SDK de Windows Phone ofrece una versión independiente de Visual Studio Express 2012 para Windows Phone. También puede funcionar como complemento de las ediciones

Professional, Premium o Ultimate de Visual Studio 2012. Con el SDK puedes usar los conocimientos y el código de programación que ya tienes para crear aplicaciones en código nativo o administrado. Además, el SDK contiene varios emuladores y otras herramientas para generar perfiles y realizar pruebas de las aplicaciones de Windows Phone en condiciones reales.

La instalación se la ejecutó directamente de la página de Windows:

<http://www.microsoft.com/en-us/download/details.aspx?id=35471>

Solo debemos brindar los datos de lugar de instalación, aceptar los términos y condiciones, y se procederá a la instalación, Es muy probable de que se requiera un reinicio.

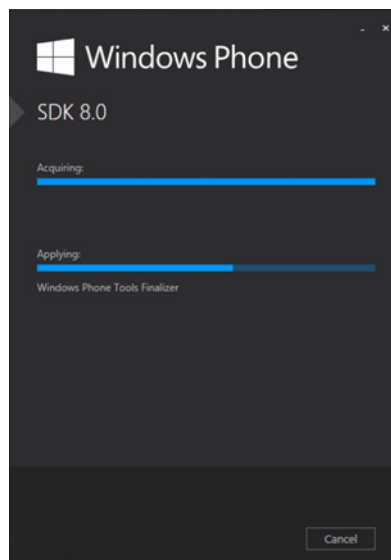


Figura III 12 Cuarta Pantalla de instalación de visual Studio Ultimate

Fuente: Propia de las autoras

La instalación del SDK tarda al menos dos horas, si todo se ha completado correctamente podremos disfrutar de las herramientas de desarrollo de Windows Phone.

CAPÍTULO IV

METODOLOGIA DE DESARROLLO: METODOLOGIA XP

La metodología de Programación Extrema (XP, por sus siglas en inglés, eXtreme Programming) es una metodología ágil para el desarrollo de softwares. Creada para proyectos donde los requisitos no sean precisos o se modifiquen con facilidad, por lo que en esta metodología es la más adecuada para nuestro proyecto de tesis ya que las modificaciones son vistas como algo normal, a diferencia de las restantes. Esta metodología está estructurada para adaptarse a los cambios de manera rápida, reduciendo los gastos, en cualquier momento del proceso de desarrollo del proyecto de tesis¹.

Las fases de la metodología XP se muestra en la siguiente figura:

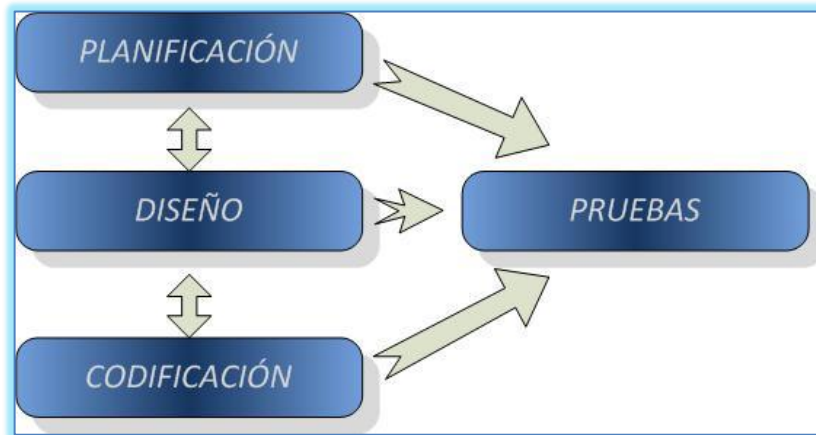


Figura IV 1 Fases de la Metodología XP

Fuente: http://wiki.monagas.udo.edu.ve/index.php/Metodolog%C3%ADas_SCRUM_y_XP

4.1 FASE DE PLANIFICACIÓN

En esta fase se determinarán las herramientas a utilizar en la realización de las aplicaciones. Se analiza varias posibles soluciones y se escoge la más adecuada para lograr un proyecto de calidad. Se estructura los prototipos necesarios para la implementación de los requerimientos básicos de las aplicaciones móviles. Teniendo en cuenta los riesgos que se pueden presentar en la elaboración de las aplicaciones.

4.1.1. Historias de usuario

Es muy importante definir las historias de usuario para que podamos planificar el tiempo entre una y otra historia de usuario. El tiempo de desarrollo es ideal para una historia de usuario es entre 1 y 3 semanas.

En esta fase se entrega al cliente una **Historial de Usuario** a realizar en las aplicaciones, para que así el cliente y el programador tengan una idea clara del propósito final y lograr una mejor comunicación. (ver Anexo 1)

Tabla IV I Tarea de Ingeniería - Definición de requerimientos

TAREA DE INGENIERÍA	
Número de la Tarea: 01	
Nombre de la tarea: Definición de requerimientos	
Tipo de tarea: Creación	Puntos Estimados:
Fecha Inicio: 4 Noviembre 2013	Fecha Fin: 22 Noviembre 2013
Programador Responsable: Johanna Honores, Jenny Vizuite.	
Descripción: Definir los requerimientos para desarrollar el sistema.	

Fuente: Propia de las autoras

4.1.2. Iteraciones

En esta fase se definen las iteraciones a cumplir en el proceso de desarrollo, Cada iteración tiene una duración de 1 mes a 2 meses, en las cuales se realizan las pruebas funcionales para cada una de las historias a ser implementadas.

Al final de cada iteración se debe comprobar que todas las historias estén implementadas.

Las iteraciones quedan de la siguiente manera:

Tabla IV II Plan de Iteración No.1

No.	Historia de Usuario	Nivel de Complejidad
01	Captura de los parámetros X, Y, y Z en WP y Android	Alta
02	Captura de los parámetros latitud y longitud en WP y Android	Alta
Total de Tiempo		2 meses

Fuente: Propia de las autoras

Tabla IV III Plan de Iteración No.2

No.	Historia de Usuario	Nivel de Complejidad
03	Calculo de la magnitud del sismo (algoritmo) en WP y Android	Alta
Total de Tiempo		2 meses

Fuente: Propia de las autoras

Tabla IV IV Plan de Iteración No.3

No.	Historia de Usuario	Nivel de Complejidad
04	Salida de gráficos estadísticos en WP y Android	Media
Total de Tiempo		1 mes

Fuente: Propia de las autoras

4.2 FASE DE DISEÑO

En el desarrollo de nuestro proyecto de tesis según la metodología X.P. es aconsejable realizar diseños simples y sencillos. Lo cual nos ayudará a realizar

las tareas de una manera menos complicada posible para conseguir un diseño fácilmente entendible e implementable que a la larga costará menos tiempo y esfuerzo desarrollar.

4.2.1 Riesgos

A continuación se realiza el análisis de riesgos (ver Anexo 2).

Identificación de Riesgos

Tabla IV V Identificación de Riesgos

ID	DESCRIPCIÓN DEL RIESGO
R1	Tiempo con el que se cuenta para la elaboración de las aplicaciones del proyecto de tesis no es suficiente para concluir satisfactoriamente con el desarrollo de la solución
R2	Definición de requerimiento ambigua para las aplicaciones
R3	Aparición de nuevos requerimientos cuando las aplicaciones están en las fases finales
R4	Modificación del cronograma de actividades
R5	Falta de conocimiento en la utilización de las herramientas de desarrollo de las aplicaciones
R6	Las aplicaciones no cubren todas las expectativas del usuario

Fuente: Propia de las autoras

Análisis de los Riesgos

- Criterios de valoración de la probabilidad

Tabla IV VI Criterios de valoración de la probabilidad de riesgos

RANGO PROBABILIDAD	DESCRIPCIÓN	VALOR
1 % - 33 %	Baja	1
34 % - 67 %	Media	2
68 % - 99 %	Alta	3

Fuente: Propia de las autoras

- Criterios de valoración del impacto

Tabla IV VII Criterio de valoración del impacto de riesgos

Impacto	Costo	Retraso	Impacto técnico	valor
Bajo	< 1 %	1 semana	Ligero efecto en el desarrollo de las aplicaciones	1
Moderado	< 5 %	2 semanas	Moderado efecto en el desarrollo de las aplicaciones	2
Alto	< 10 %	1 mes	Severo efecto en el desarrollo de las aplicaciones	3
Crítico	>10 %	Más de 1 mes	Proyecto no puede ser terminado	4

Fuente: Propia de las autoras

- Determinación de exposición al riesgo

Tabla IV VIII Determinación de exposición al riesgo

Exposición al riesgo	Valor	Color
Baja	1 o 2	Verde
Media	3 o 4	Amarillo
Alta	>= 6	Rojo

Fuente: Propia de las autoras

Tabla IV IX Valoración de determinación de exposición al riesgo

Impacto Probabilidad	Bajo = 1	Moderada = 2	Alta = 3	Crítico = 4
Alta = 3	3	6	9	12
Media = 2	2	4	6	8
Baja = 1	1	2	3	4

Fuente: Propia de las autoras

- Tabla de impactos esperados

Tabla IV X Impacto Esperados

Identificación	Probabilidad		Impacto		Exposición al riesgo		
	%	Valor	Probabilidad	Valor	Impacto	Valor	Exposición
R1	50	2	Media	3	Alto	9	Alto
R2	40	2	Media	2	Medio	6	Alto
R3	10	1	Bajo	3	Alto	3	Medio
R4	10	1	Bajo	4	Crítico	4	Medio
R5	15	2	Medio	3	Alto	3	Medio
R6	5	1	Baja	4	Medio	2	Bajo

Fuente: Propia de las autoras

Priorización de los Riesgos

Tabla IV XI Priorización de los Riesgos

ID	EXPOSICIÓN	
	Valor	Exposición
R1	9	Alto
R2	6	Alto
R3	3	Medio
R4	4	Medio
R5	2	Bajo
R6	2	Bajo

Fuente: Propia de las autoras

- Plan de Reducción, Supervisión y Gestión del Riesgo

Tabla IV XII Hoja de Gestión de Riesgo 1

HOJA DE GESTIÓN DEL RIESGO			
ID DEL RIESGO: R1			
Probabilidad: 50 Valor: 2	Impacto: Alto Valor: 3	Exposición: Alta Valor: 9	Prioridad: Alta Valor: 9
Descripción: Tiempo con el que se cuenta para la elaboración de las aplicaciones del proyecto de tesis no es suficiente para concluir satisfactoriamente con el desarrollo de la solución			

REFINAMIENTO: Causas: <ul style="list-style-type: none">• Retraso en la ejecución de las actividades planificadas• Falta de cumplimiento del cronograma• Reducción del tiempo Consecuencias: <ul style="list-style-type: none">• Afecta la calidad de las aplicaciones• El proyecto de tesis no podrá ser entregado en la fecha planificada
REDUCCIÓN Y SUPERVISIÓN: <ul style="list-style-type: none">• Elaborar planes para cumplir las fechas planificadas• Adquirir nuevos horarios que permitan lograr el cumplimiento de las actividades
SUPERVISIÓN: <ul style="list-style-type: none">• Seguimiento continuo en el desarrollo del proyecto de tesis• Incrementar procesos de seguridad en cuanto a la verificación del cronograma
GESTIÓN: Plan de Contingencia: <ul style="list-style-type: none">• Realizar la factibilidad técnica antes del desarrollo del proyecto• Incrementar procesos de seguridad en cuanto a la verificación del cronograma
ESTADO ACTUAL Fase de reducción iniciada <input checked="" type="checkbox"/> Fase de Supervisión iniciada <input type="checkbox"/> Gestionando el riesgo <input type="checkbox"/>
RESPONSABLES: Johanna Honores Jenny Vizuite

Fuente: Propia de las autoras

4.2.2 Tarjetas C.R.C.

Las tarjetas C.R.C nos ayudan a representar objetos; la clase a la que pertenece el objeto se encuentra en la parte de arriba de la tarjeta, en la columna a la izquierda se encuentran las responsabilidades u objetivos que debemos cumplir con el objeto y a la derecha, están las clases que colaboran con cada responsabilidad.

Aquí empieza la implementación del sistema y se determina la estructura del modelo de datos. Se realiza las tarjetas CRC (clase, responsabilidad y colaboración). (Ver Anexo 3)

Tabla IV XIII Tarjeta CRC - Tabla sismo

Sismo	
Recoge el código (serial) del sismo.	param_lecturas
Recoge la fecha en que ocurre el sismo.	
Recoge los parámetros de latitud y longitud del sismo.	
Recoge los parámetros X, Y, y Z del sismo.	
Calcula la intensidad del sismo.	

Fuente: Propia de las autoras

Diagrama de Componentes

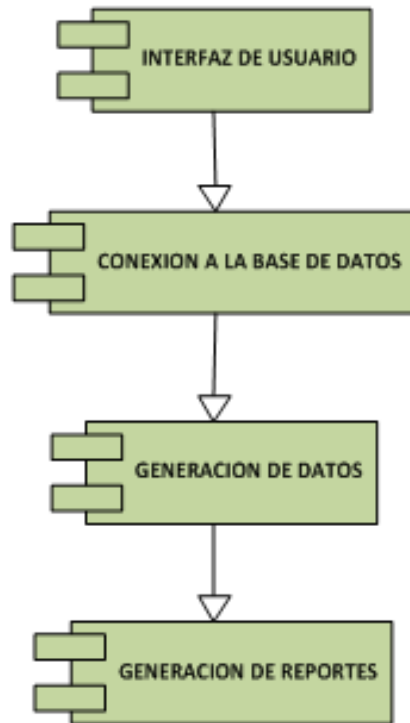


Figura IV 2 Diagrama de componentes

Fuente: Propia de las autoras

Arquitectura Física de la Aplicación

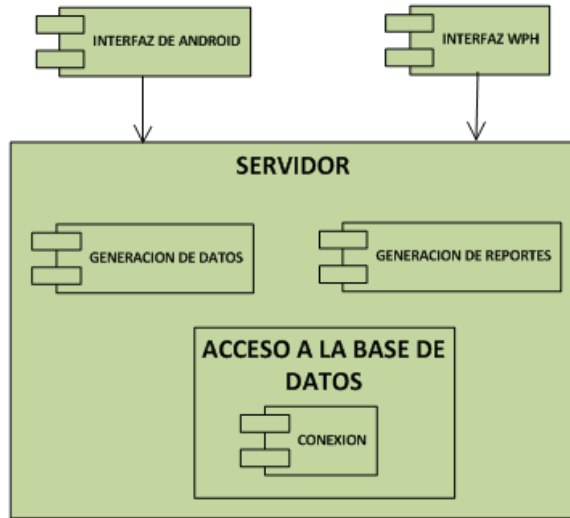


Figura IV 3 Arquitectura física de la aplicación

Fuente: Propia de las autoras

Diseño de la Base de Datos

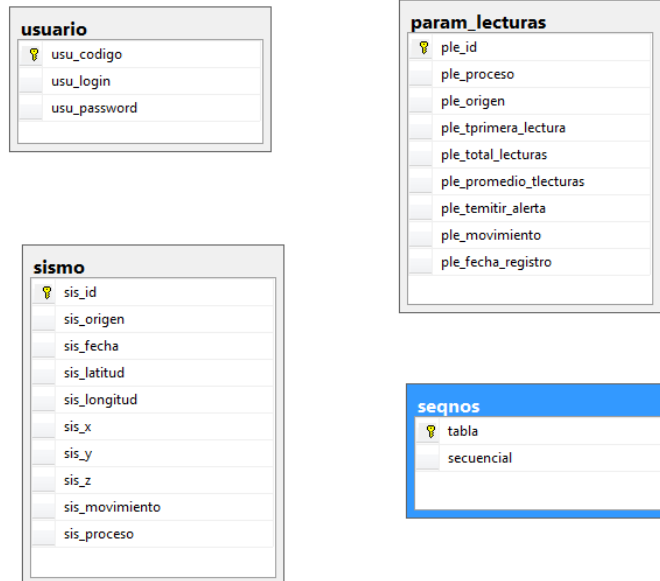


Figura IV 4 Diagrama de Base de Datos

Fuente: Propia de las autoras

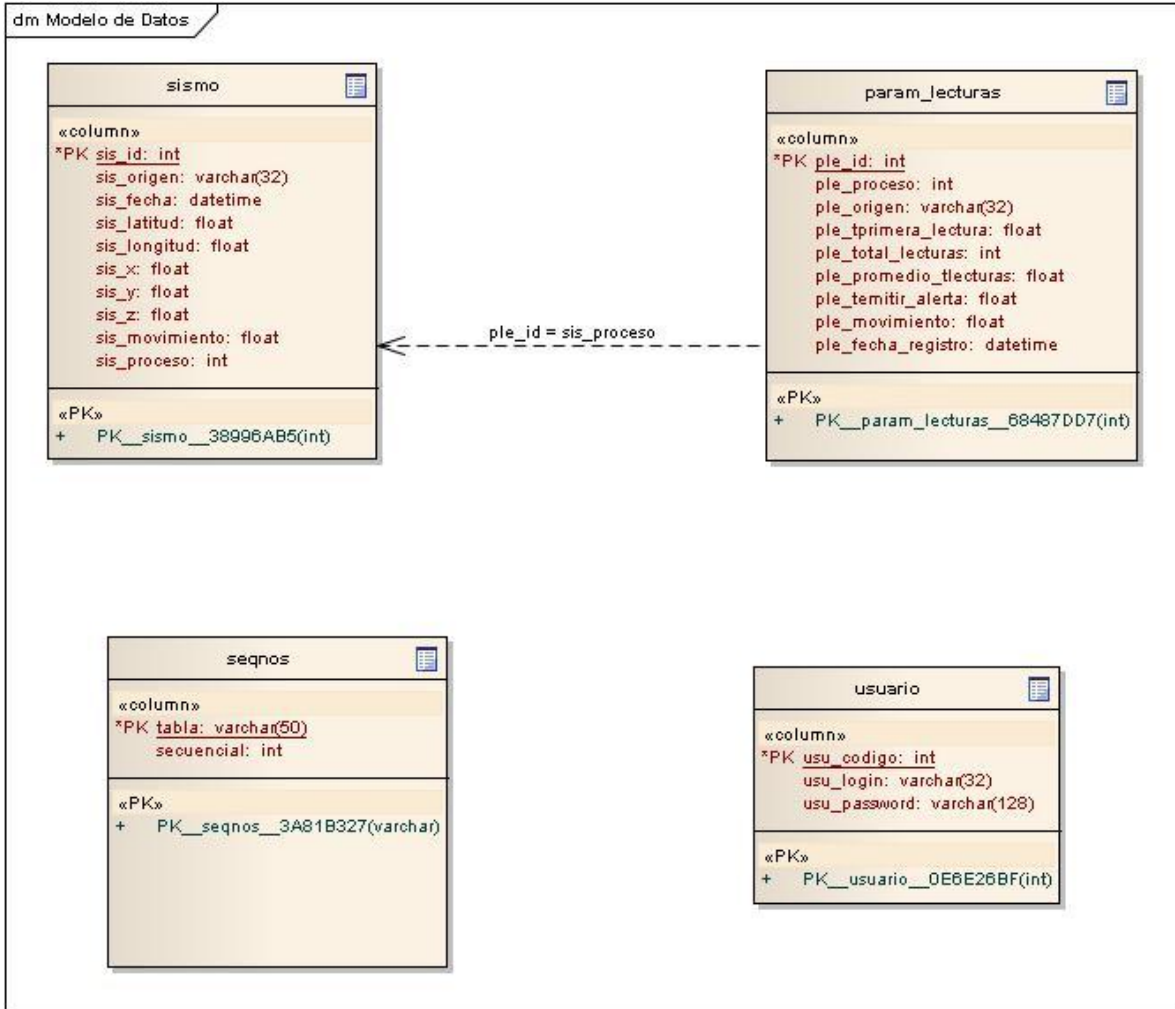


Figura IV 5 Modelo de Base de Datos

Fuente: Propia de las autoras

4.3 FASE DE CODIFICACIÓN

Se debe definir estándares de programación aceptado e implantado por todo el equipo de programación es una práctica que no solo se recomienda en la

metodología XP si no en todo trabajo de desarrollo de aplicaciones; estos nos permitirá en un futuro modificar fácilmente el código implementado de ser necesario.

Estándares del código

La estandarización del código se la realizó desde el mismo momento en que se inició la codificación. Debido que el equipo de desarrollo había estado trabajando por largo tiempo juntos.

4.4 FASE DE PRUEBAS

En esta fase se realizan las pruebas con las aplicaciones para movimientos telúricos instalados en los Smartphone para el análisis posterior teniendo en cuenta las limitaciones del ambiente de pruebas.

A continuación se presentan las pantallas capturadas con las aplicaciones en ejecución:

- **Pantalla de Android.-** En esta pantalla se puede observar todos los valores que la aplicación utilizara para la detección del movimiento antes de iniciar la aplicación todos los valores se encuentran en cero.

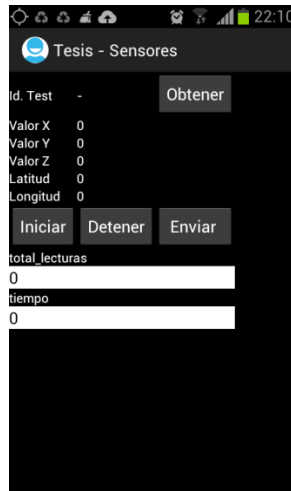


Figura IV 6 Pantalla sin datos de la aplicación en Android

Fuente: Propia de las autoras

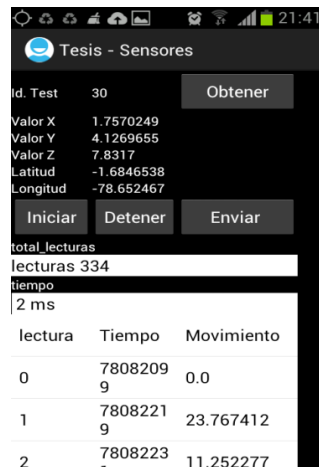


Figura IV 7 Pantalla con datos de la aplicación en Android

Fuente: Propia de las autoras

Después de obtener el id se debe mover al Smartphone para poder capturar los datos los valores que antes se encontraban en cero ahora ya tienen los valores correspondientes al movimiento que se realizó también se puede observar la latitud y la longitud de donde se generó el sismo.

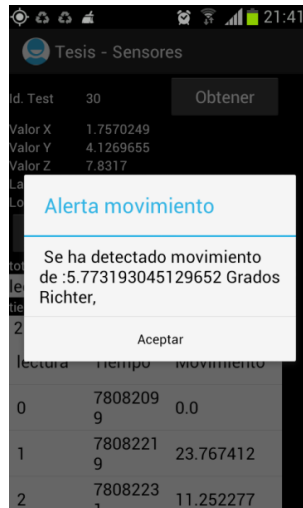


Figura IV 8 Mensaje en la aplicación de Android

Fuente: Propia de las autoras

Aquí se muestra la intensidad del sismo en la escala de Richter.

- **Pantalla de Windows Phone** Al igual que la aplicación para Android antes de obtener el id los valores estarán en cero.



Figura IV 9 Pantalla sin datos de la aplicación en Windows Phone

Fuente: Propia de las autoras

Después de obtener el id se debe mover el Smartphone para que empiece la captura de datos.



Figura IV 10 Pantalla con datos de la aplicación en Windows Phone

Fuente: Propia de las autoras

Después de generar el movimiento la aplicación emitirá una alarma con la intensidad del sismo que se generó.



Figura IV 11 Mensaje en la aplicación Windows Phone

Fuente: Propia de las autoras

Reportes

Después de obtener los datos del sismo se envían a través de un web service a la base de datos para generar los reportes para realizar el estudio estadístico comparativo entre los sensores Android y Windows phone esto nos facilita al momento de comparar cual sensor es mejor al momento de capturar los datos Ver Anexo 4.

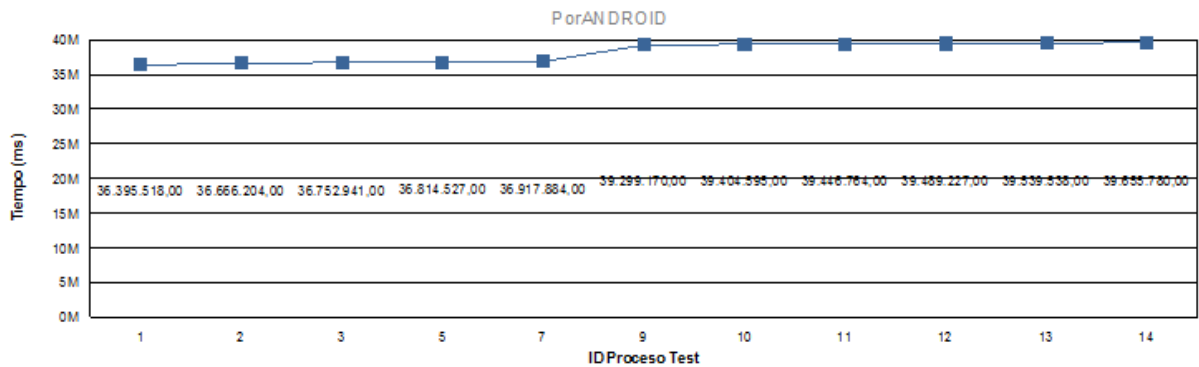
Se debe considerar que los tiempos están tomados en milisegundos para todos los reportes:

FECHA: 4/21/2014

PROMEDIO ANDROID: 38216559

PROMEDIO WINDOWS PHONE: 38442416

TIEMPO HASTA PRIMERA LECTURA



TIEMPO HASTA PRIMERA LECTURA

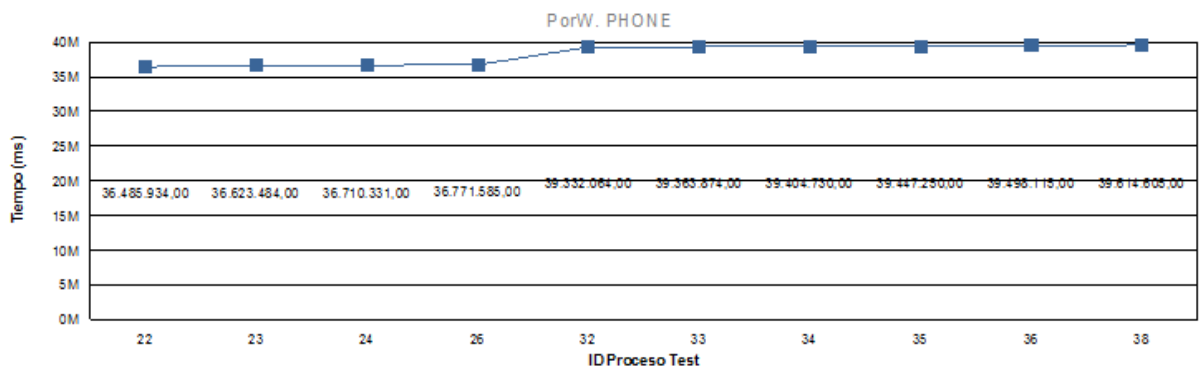


Figura IV 12 Tiempo hasta la primera lectura en Android y Windows Phone

Fuente: Propia de las autoras

En la Figura IV.12 se puede decir que en Android tomo menos tiempo para capturar la primera lectura con respecto a Windows Phone.

CAPÍTULO V

RESULTADOS Y COMPARACIÓN ESTADÍSTICA

En este capítulo se realizará el estudio estadístico comparativo entre los sensores Android y Windows phone, este trabajo de investigación nos ayudará a determinar cuál sensor es mejor al momento de capturar los datos en un movimiento telúrico bajo ciertos parámetros que se los definirán más adelante.

Es muy importante tener en cuenta las limitaciones que posee el sensor de un Smartphone por lo cual se ha creado un ambiente de pruebas que nos ayudará con nuestro proyecto de tesis.

5.1 AMBIENTE DE PRUEBAS

Para este proyecto de tesis se ha tenido en cuenta las siguientes limitaciones para realizar las pruebas que a continuación se las enlista:

- Se ha utilizado una mesa que simula un movimiento sísmico de diferente intensidad que se va graduando con un potenciómetro. El tipo de movimiento que simula la mesa es un movimiento pendular Ver Anexo 5.
- El algoritmo que se creó para el cálculo de la intensidad del sismo captura valores mayores a 5 grados en la escala de Richter en un determinado tiempo.

Al momento de capturar los datos los sensores de Windows phone y Android nos proporcionan una magnitud estas son: Para Windows

5.2 DETERMINACION DE PARAMETROS PARA EL ESTUDIO ESTADISTICO COMPARATIVO

Para este proyecto de tesis se ha establecido cuatro parámetros con sus respectivos indicadores los cuales nos ayudarán al momento de realizar el estudio estadístico comparativo entre los sensores Android y Windows Phone.

Tabla V I Parámetros para el Estudio Estadísticos

Parámetro	Indicadores
Codificación	• Tiempo de desarrollo
	• LDC
	• No. De componentes
Lenguajes de programación	• No. De lenguajes de programación
	• Curva de aprendizaje de los lenguajes de programación
Instalación	• Tiempo de instalación
Rendimiento	• Tiempo hasta la primera lectura
	• Total de lecturas

Parámetro	Indicadores
	<ul style="list-style-type: none">• Tiempo hasta emitir la alerta• Movimiento detectado

Fuente: Propia de las autoras

5.2.1. Codificación

Se describe el nivel de complejidad en el desarrollo y codificación de cada aplicación para los dos Smartphone.

Tabla V II Indicadores de los parámetros de codificación

INDICADORES		DESCRIPCIÓN
Indicador 1.1	Tiempo de desarrollo	Se detalla el tiempo que se tardó en desarrollar las aplicaciones.
Indicador 1.2	LDC	Se describe las líneas de código que se utilizó para desarrollar las aplicaciones
Indicador 1.3	Número de componentes	Se detalla el número de componentes que se utilizó para el desarrollo de cada aplicación.

Fuente: Propia de las autoras

5.2.2. Lenguajes de Programación

Se describe los lenguajes de programación que se tiene para las plataformas de los Smartphone seleccionadas anteriormente y cuál es la más fácil de aprender.

Tabla V III Indicadores del de Lenguaje de Programación

INDICADORES		DESCRIPCIÓN
Indicador [2.1]	No. De lenguajes de programación.	Describe los lenguajes de programación que se tiene para el desarrollo de las aplicaciones.
Indicador [2.2]	Curva de aprendizaje de los lenguajes de programación.	Describe el tiempo empleado en comprender el lenguaje de programación su estructura y sintaxis.

Fuente: Propia de las autoras

5.2.3. Instalación

Se describe la facilidad de instalación y el tiempo utilizado.

Tabla V IV Indicador de la instalación

INDICADORES		DESCRIPCIÓN
Indicador [3.1]	Tiempo de instalación	Tiempo utilizado en la instalación de cada plataforma.

Fuente: Propia de las autoras

5.2.3. Rendimiento

Describe el rendimiento que tiene las aplicaciones cuando captura los parámetros en cada uno de los Smartphone para la detección de los movimientos telúricos.

Tabla V V Indicadores del Rendimiento

INDICADORES		DESCRIPCIÓN
Indicador 4.1	Tiempo hasta la primera lectura	Se detalla el tiempo que tardó el sensor al capturar los datos hasta la primera lectura.

INDICADORES		DESCRIPCIÓN
Indicador 4.2	Total de lecturas	Describe el total de lecturas que captura en sensor al producirse el movimiento telúrico
Indicador 4.3	Tiempo hasta emitir la alerta	Muestra el tiempo que tarda hasta emitir la alarma.
Indicador 4.4	Movimiento detectado	Indica la intensidad del movimiento que se detectó con ayuda del sensor de los Smartphone

Fuente: Propia de las autoras

5.3 Análisis Comparativo

Cada dispositivo se probara en el mismo escenario la comparación de los parámetros permitirá escoger cual es el mejor sensor. Algunos de los resultados se los hará midiendo el tiempo empleado en la captura de datos.

La calificación de cada parámetro se determinara de acuerdo a la siguiente escala, la cual nos permitirá determinar cuál es el mejor sensor para la detección de movimientos telúricos.

Tabla V VI Valoración de porcentajes

Regular	Bueno	Muy Bueno	Excelente
<70%	>=70% y <80%	>=80% y <95%	>=95%

Fuente: Propia de las autoras

Para evaluarlos indicadores se tomara la valoración entre uno y cuatro.

Tabla V. I. Valores Cualitativos y Cuantitativos

Valor Cualitativo		Valor Cuantitativo	Valor Representativo
Insuficiente	No Satisfactorio	1	☆
Parcial	Poco Satisfactorio	2	☆☆
Suficiente	Satisfactorio	3	☆☆☆
Excelente	Muy Satisfactorio	4	☆☆☆☆

Fuente: Propia de las autoras

Para realizar la comparación se utilizara la siguiente nomenclatura:

A= Representa el puntaje obtenido por Android.

W = Representa el puntaje obtenido por Windows Phone.

P = Representa el puntaje sobre el cual será evaluado el parámetro.

Ca = Representa el puntaje alcanzado por Android en el parámetro.

Cw = Representa el puntaje alcanzado por Windows Phone en el parámetro.

Pe = Representa el puntaje por el cual es evaluado el parámetro.

Pa = Calificación porcentual obtenida por Android.

Pw = Calificación porcentual obtenida por Windows Phone.

Las fórmulas que se utilizaran en el proceso del análisis comparativo son las siguientes:

$$Ca = \sum A$$

$$Cw = \sum W$$

$$Ct = \sum P$$

$$Pa = \left(\frac{ca}{pe} \right) * 100 \%$$

$$Pw = \left(\frac{cw}{pe} \right) * 100 \%$$

5.3.1. Codificación

- **Indicador 1.1: Tiempo de Desarrollo**

En este indicador se valorizará tomando en consideración el tiempo en meses que se ha tardado en desarrollar las aplicaciones para los dos Smartphone.

Tabla V VII Tabla de Valores Cualitativos y Cuantitativos del Indicador 1.1

Valorización		
Meses	Valor Cuantitativo	Valor Cualitativo
1 mes	4	Muy Satisfactorio
2 meses	3	Satisfactorio
4 meses	2	Poco Satisfactorio
Más de 5 meses	1	No Satisfactorio

Fuente: Propia de las autoras

- **Indicador 1.2: LDC**

En este índice se valorizará mediante el número de líneas de código que se utilizó para cada aplicación de los Smartphone para la detección de movimientos telúricos.

Tabla V VIII Tabla de Valores Cualitativos y Cuantitativos del Indicador 1.2

Valorización		
LDC	Valor Cuantitativo	Valor Cualitativo
100 – 700	4	Muy Satisfactorio
701 – 1010	3	Satisfactorio
1011 – 2000	2	Poco Satisfactorio
Más de 2001	1	No Satisfactorio

Fuente: Propia de las autoras

- **Indicador 1.3: No. de Componentes**

Se valorizará según el número de componentes que se utilizó para la creación de las aplicaciones en cada uno de los Smartphone.

Tabla V IX Escala de Valores del Indicador 1.3

Valorización		
N° Componentes	Valor Cuantitativo	Valor Cualitativo
1 – 2	3	Muy Satisfactorio
3 – 4	2	Satisfactorio
Más de 5	1	Poco Satisfactorio

Fuente: Propia de las autoras

5.3.1.1. Valoración

Tabla V X Valorización de los indicadores del Parámetro Codificación

Indicador	Android		Windows Phone	
	Valor Cuantitativo	Valor Cualitativo	Valor Cuantitativo	Valor Cualitativo
Tiempo de desarrollo	2	Poco Satisfactorio	4	Muy Satisfactorio
LDC	3	Poco Satisfactorio	4	Muy Satisfactorio

Indicador	Android		Windows Phone	
	Valor Cuantitativo	Valor Cualitativo	Valor Cuantitativo	Valor Cualitativo
Número de componentes	2	Poco Satisfactorio	3	Muy Satisfactorio

Fuente: Propia de las autoras

5.3.1.2. Interpretación

Tiempo de desarrollo

En este indicador podemos observar que el tiempo nos hemos demorado en el desarrollo de cada una de las aplicaciones por lo cual se la ha asignado la siguiente calificación: para Windows Phone el valor de 4 que representa Muy satisfactorio y en Android el valor de 3 que significa satisfactorio.

LDC

En el desarrollo de las aplicaciones se ha creado muchas líneas de código las mismas que se las ha puntuado con la siguiente calificación: para Windows Phone con un valor de 4 que significa muy satisfactorio y en Android tiene una puntuación de 3 que es poco satisfactorio.

Número de componentes

Este indicador permite determinar cuántos componentes se ha utilizado para el desarrollo de cada una de las aplicaciones para los Smartphone al momento

de la detección de un movimiento telúrico para lo cual se tiene: para Windows Phone es 3 lo cual significa muy satisfactorio y en Android su valor es de 2 que es satisfactorio.

5.3.1.3. Calificación

Cálculo de Porcentajes

$$Ca = \sum A \qquad Cw = \sum w \qquad Pe = \sum P$$

$$Pa = (ca/pe) * 100 \%$$

$$Pw = (cw/pe) * 100 \%$$

$$Ca = 2+3+2 = 7$$

$$Cw = 4+4+3 = 11$$

$$Pe = 4+4+4 = 12$$

$$Pa = \left(\frac{7}{12}\right) * 100\% = 58,33 \%$$

$$Pw = \left(\frac{11}{12}\right) * 100\% = 91,66 \%$$

5.3.1.4. Representación de Resultados

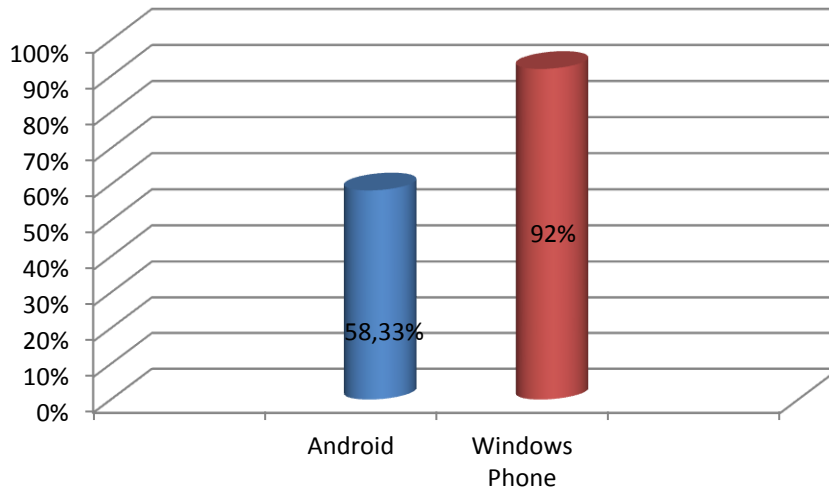


Figura V 1 Representación de Resultados Parámetro 1: Codificación

Fuente: Propia de las autoras

5.3.2. Lenguajes de Programación

- **Indicador 2.1: No. de Lenguajes de Programación**

En este parámetro se evaluará la cantidad de lenguajes de programación para cada uno de las plataformas.

Tabla V XI Escala de Valores del Indicador 2.1

Número de lenguajes	Valoración	
	Cuantitativa	Cualitativa
Hasta 1	1	No Satisfactorio
Hasta 2	2	Poco Satisfactorio
Hasta 3	3	Satisfactorio
4 0 mas	4	Muy Satisfactorio

Fuente: Propia de las autoras

- **Indicador 2.2: Curva de Aprendizaje**

Este indicador nos permitirá medir el tiempo empleado en comprender las estructuras de los lenguajes de cada plataforma.

Tabla V XII Escala de valores del Indicador 2.2

Tiempo en minutos	Valoración	
	Cuantitativa	Cualitativa
41 a 50	1	No Satisfactorio
31 a 40	2	Poco Satisfactorio
21 a 30	3	Satisfactorio
10 a 20	4	Muy Satisfactorio

Fuente: Propia de las autoras

5.3.2.1. Valoración

Tabla V XIII Valorización de los indicadores del Parámetro Lenguajes de programación

Indicadores	Android		Windows Phone	
	Valor Cualitativo	Valor Cuantitativo	Valor Cualitativo	Valor Cuantitativo
No. De lenguajes de programación	Poco Satisfactorio	2	Poco Satisfactorio	2
Curva de aprendizaje de los lenguajes de programación	No Satisfactorio	1	Muy Satisfactorio	4

Fuente: Propia de las autoras

5.3.2.2. Interpretación

Número De lenguajes de programación

En Android se tiene java y C/C++por lo que obtiene una calificación de 2 puntos equivalentes a poco satisfactorio, Windows Phone tiene los lenguajes C# y Visual Basic obteniendo la misma calificación de Android. Podemos decir que las dos plataformas nos ofrecen pocos lenguajes de programación.

Curva de aprendizaje de los lenguajes de programación

Este indicador nos permite medir el tiempo requerido para comprender los lenguajes de programación. En Android no llevo 45 minutos poder entender la estructura para poder realizar las aplicaciones obteniendo una calificación de 1 equivalente a no satisfactorio, mientras que en Windows phone no llevo aproximadamente 10 minutos para comprender su código por lo cual obtuvo una calificación de 4 equivalente a muy satisfactorio.

5.3.2.3. Calificación

Cálculo de Porcentajes

$$Ca = \sum A \qquad Cw = \sum w \qquad Pe = \sum P$$

$$Pa = \left(\frac{ca}{pe} \right) * 100 \%$$

$$Pw = \left(\frac{cw}{pe} \right) * 100 \%$$

$$Ca = 2+1 = 3$$

$$Cw = 2+4 = 6$$

$$Pe = 4+4 = 8$$

$$Pa = \left(\frac{3}{8} \right) * 100 \% = 37,5 \%$$

$$Pw = \left(\frac{6}{8}\right) * 100 \% = 75 \%$$

5.3.2.4. Representación de Resultados

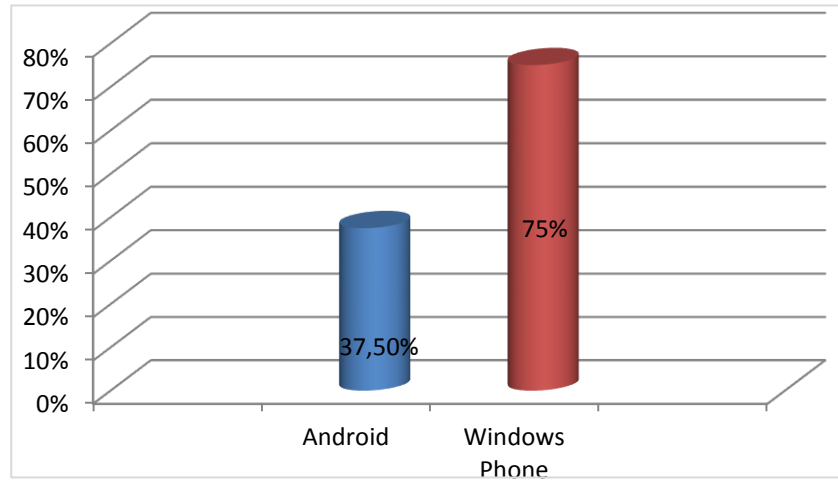


Figura V 2 Representación de Resultados Parámetro 2: Lenguajes de Programación

Fuente: Propia de las autoras

5.3.3. instalación

- **Indicador 3.1: Tiempo de Instalación**

Este indicador medirá el tiempo utilizado en la instalación de la aplicación en cada dispositivo.

5.3.3.1. Valoración

Tabla V XIV Valorización de los indicadores del Parámetro Instalación

Indicadores	Android		Windows Phone	
	Valor Cualitativo	Valor Cuantitativo	Valor Cualitativo	Valor Cuantitativo
Tiempo de instalación	No Satisfactorio	1	Muy Satisfactorio	4

Fuente: Propia de las autoras

5.3.3.2. Interpretación

Tiempo de instalación

En este indicador se medirá en tiempo de instalación de cada aplicación, en Android la instalación duro aproximadamente 50 segundos obteniendo una calificación de 1 equivalente a no satisfactorio, mientras que en Windows phone la instalación se la realizo en 10 segundos mucho más rápido por lo cual obtuvo una calificación de 4 equivalente a muy satisfactorio.

5.3.3.3. Calificación

Cálculo de Porcentajes

$$Ca = \sum A \qquad Cw = \sum w \qquad Pe = \sum P$$

$$Pa = \left(\frac{ca}{pe} \right) * 100 \%$$

$$Pw = \left(\frac{cw}{pe} \right) * 100 \%$$

$$Ca = 1 = 1$$

$$Cw = 4 = 4$$

$$Pe = 4 = 4$$

$$Pa = \left(\frac{1}{4} \right) * 100 \% = 25 \%$$

$$Pw = \left(\frac{4}{4} \right) * 100 \% = 100 \%$$

5.3.3.4. Representación de Resultados

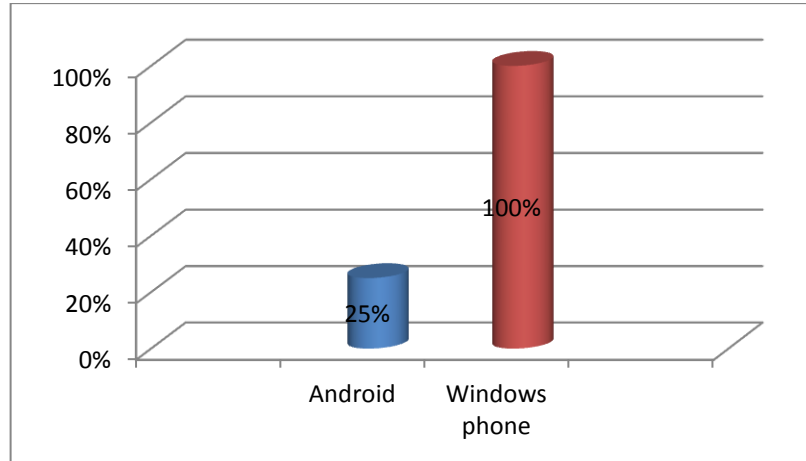


Figura V 3 Representación de Resultados Parámetro 3: Instalación

Fuente: Propia de las autoras

5.3.4. Rendimiento

- **Indicador 3.1: Tiempo Hasta la Primera Lectura**

En este indicador se valorizará tomando en consideración el tiempo en que se ha tardado en capturar los datos hasta la primera lectura en los dos Smartphone.

Tabla V XV Escala de valores del Indicador 3.1

Meses	Valorización	
	Valor Cuantitativo	Valor Cualitativo
10 – 20	4	Muy Satisfactorio
21 – 30	3	Satisfactorio
31 -50	2	Poco Satisfactorio
Más de 50	1	No Satisfactorio

Fuente: Propia de las autoras

- **Indicador 3.2: Total de Lecturas**

Se este indicador se valorizará de acuerdo al número de lecturas que captura cada sensor de los dos Smartphone al momento que ocurre un sismo

Tabla V XVI Escala de valores del Indicador 3.2

Valorización		
Total de lecturas	Valor Cuantitativo	Valor Cualitativo
0 – 100	4	Muy Satisfactorio
101 – 200	3	Satisfactorio
201 – 300	2	Poco Satisfactorio
Más de 301	1	No Satisfactorio

Fuente: Propia de las autoras

- **Indicador 3.3: Tiempo Hasta Emitir la Alarma**

En este índice se valorizará mediante el tiempo (milisegundos) que tarda cada aplicación para emitir la alerta cuando detecte un sismo en cada uno de los Smartphone.

Tabla V XVII Escala de valores del Indicador 3.3

Valorización		
Tiempo (ms)	Valor Cuantitativo	Valor Cualitativo
1 – 10	4	Muy Satisfactorio
10 – 20	3	Satisfactorio
30 – 40	2	Poco Satisfactorio
Más de 41	1	No Satisfactorio

Fuente: Propia de las autoras

- **Indicador 3.4: Movimiento**

Se valorizará según la intensidad del sismo capturado a través de las aplicaciones en cada uno de los Smartphone.

Tabla V XVIII Escala de valores del Indicador 3.4

Valorización		
Movimiento	Valor Cuantitativo	Valor Cualitativo
4 – 5	1	Bajo
6 – 7	2	Medio
Más de 7	3	Alto

Fuente: Propia de las autoras

5.3.4.1. Valoración

Tabla V XIX Valorización de los indicadores del Parámetro Rendimiento

Indicador	Android		Windows Phone	
	Valor Cuantitativo	Valor Cualitativo	Valor Cuantitativo	Valor Cualitativo
Tiempo hasta la primera lectura	2	Poco Satisfactorio	3	Satisfactorio
Total de lecturas	3	Satisfactorio	4	Muy satisfactorio
Tiempo hasta emitir la alerta	2	Poco Satisfactorio	3	Satisfactorio
Movimiento detectado	2	Satisfactorio	3	Alto

Fuente: Propia de las autoras

5.3.4.2. Interpretación

Tiempo hasta la primera lectura: El tiempo que se emplea para la recolección de los primeros datos utilizando las aplicaciones para los dos Smartphone es muy importante ya que el sensor de Windows phone realiza la

captura de los datos más rápido hasta la primera lectura por eso se le asignado una valoración 3 que equivale a Satisfactorio y en el otro dispositivo ósea en Android se le ha dado un valor de 2 que equivale a Poco Satisfactorio.

Total de lecturas: Este indicador nos permite identificar cuál de los dos sensores recolecto más lecturas desde que inició el movimiento sísmico, por lo que se determinó la siguiente calificación: para Windows Phone se puntuó con una valoración de 4 que significa Muy Satisfactorio y para Android se la calificó con un valor de 3 que significa Satisfactorio.

Tiempo hasta emitir la alerta: Al evaluar este indicador se puede apreciar que el tiempo que tarda en emitir la alerta en el sensor al detectar un movimiento telúrico en Windows phone se califica con un valor de 3 que significa Satisfactorio y en Android un valoración de 2 que quiere decir poco satisfactorio.

Movimiento detectado: en la evaluación de este indicador podemos ver que el movimiento que captura el sensor de Windows phone se le asigna una calificación de 3 que significa alto mientras que en el sensor de Android se lo valora con 2 que lo interpretamos como medio.

5.3.4.3. Calificación

Cálculo de Porcentajes

$$Ca = \sum A \qquad Cw = \sum w \qquad Pe = \sum P$$

$$Pa = (ca/pe) * 100 \%$$

$$Pw = (cw/pe) * 100 \%$$

$$Ca = 2+3+2+2 = 9$$

$$Cw = 3+4+3+3 = 13$$

$$Pe = 4+4+4+4 = 16$$

$$Pa = \left(\frac{9}{16}\right) * 100\% = 56,25 \%$$

$$Pw = \left(\frac{13}{16}\right) * 100\% = 81,25 \%$$

5.3.3.4. Representación de Resultados

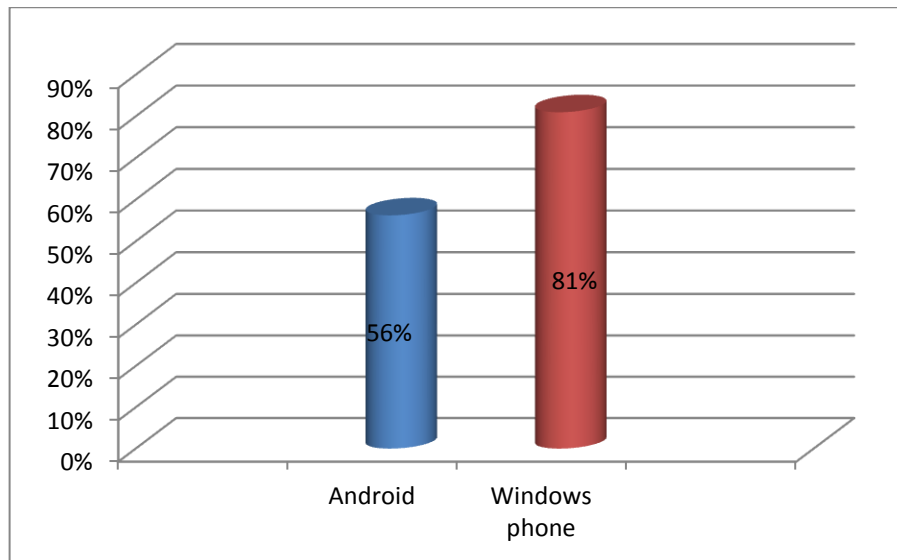


Figura V 4 Representación de Resultados Parámetro 4: Rendimiento

Fuente: Propia de las autoras

5.3.5. Puntajes Alcanzados

Una vez que se ha evaluado todos los parámetros para el estudio estadístico comparativo entre los sensores Android y Windows Phone tenemos el siguiente:

Tabla V XX Parámetros Evaluados

PARAMETROS	INDICADORES	ANDROID	WINDOWS PHONE
CODIFICACION	Tiempo de desarrollo	2	4
	LDC	3	4
	No. De componentes	2	3
LENGUAJES DE PROGRAMACION	No. De Lenguajes de Programación	2	2
	Curva de aprendizaje	1	4
INSTALACIÓN	Tiempo de instalación	1	4
RENDIMIENTO	Tiempo hasta la primera lectura	2	3
	Total de lecturas	3	4
	Tiempo hasta emitir la alarma	2	3
	Movimiento	2	3
Totales		20	34

Fuente: Propia de las autoras

5.3.6. Diagrama General de Resultados

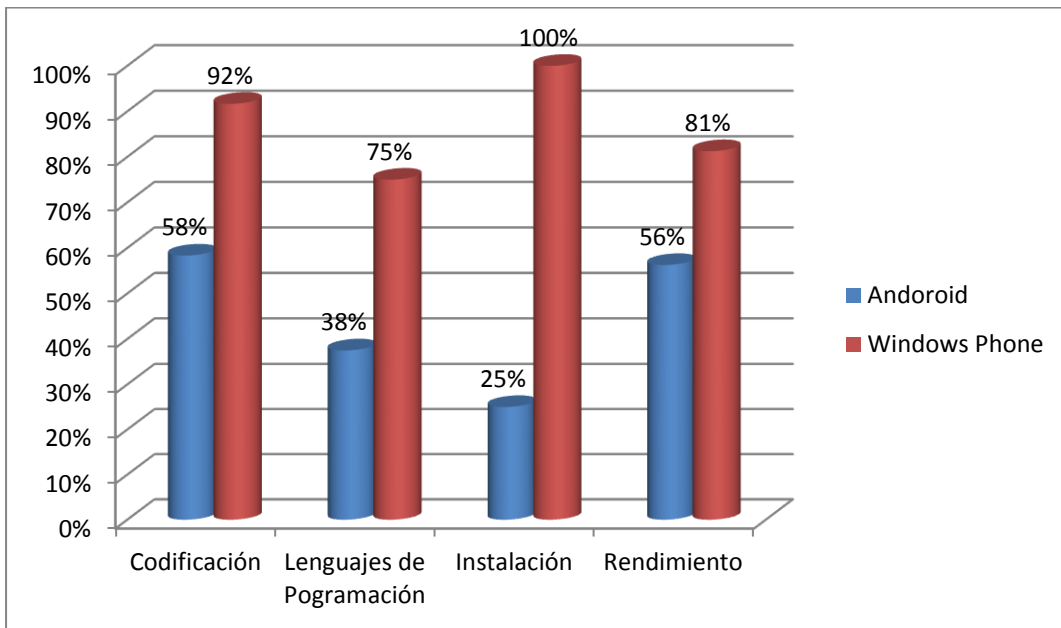


Figura V 5 Diagrama General de Resultados

Fuente: Propia de las autoras

Tabla V XXI Resultados en Porcentajes

	Codificación	Lenguajes de Programación	Instalación	Rendimiento
Android	58.33	37,5	25	56,25
Windows Phone	91.66	75	100	81,25

Fuente: Propia de las autoras

5.3.7. Interpretación

Con los porcentajes que se obtuvo de la evaluación de cada parámetro podemos conocer de manera global el comportamiento de cada sensor en la detección de movimientos telúricos.

El parámetro Codificación nos permitió medir la facilidad, tiempo y el número de componentes para la realización de las aplicaciones en donde se pudo observar que Windows phone tiene una gran ventaja ante Android.

El lenguaje de Programación es el parámetro que nos permitió medir cual aplicación fue más fácil de desarrollar en donde se destacó Windows phone ya que C# es fácil de aprender ya que no tiene una sintaxis de programación tan compleja como java.

La instalación de las aplicaciones es un parámetro muy importante ya que nos permitió medir la complejidad al momento de instalar la aplicación en los Smartphone en donde Windows phone presenta un óptimo proceso de instalación destacándose sobre Android.

El Rendimiento nos permito medir que sensor es mejor dependiendo del tiempo y en la captura de datos, Windows phone presenta un sensor más sensible que el sensor de Android.

5.4. Comprobación de la Hipótesis

La hipótesis planteada es:

H1: Realizar un estudio estadístico comparativo entre los sensores de una aplicación Android y otra Windows Phone, permitirá escoger la mejor para desarrollo de una aplicación para la detección de movimientos telúricos.

5.4.1. Resultados Finales

Tabla V XXII Cuadro de Resultados Finales

PARAMETROS	INDICADORES	ANDROID	WINDOWS PHONE
CODIFICACION	Tiempo de desarrollo	2	4
	LDC	3	4
	No. De componentes	2	3
LENGUAJES DE PROGRAMACION	No. De Lenguajes de Programación	2	2
	Curva de aprendizaje	1	4
INSTALACIÓN	Tiempo de instalación	1	4
RENDIMIENTO	Tiempo hasta la primera lectura	2	3
	Total de lecturas	3	4
	Tiempo hasta emitir la alarma	2	3
	Movimiento	2	3
Totales		20	34

Fuente: Propia de las autoras

Tabla V XXIII Valores y Porcentajes Finales

TOTAL		
	Valor	Porcentaje
Android	20	50 %
Windows Phone	34	85 %

Fuente: Propia de las autoras

5.4.2. Representación Gráfica

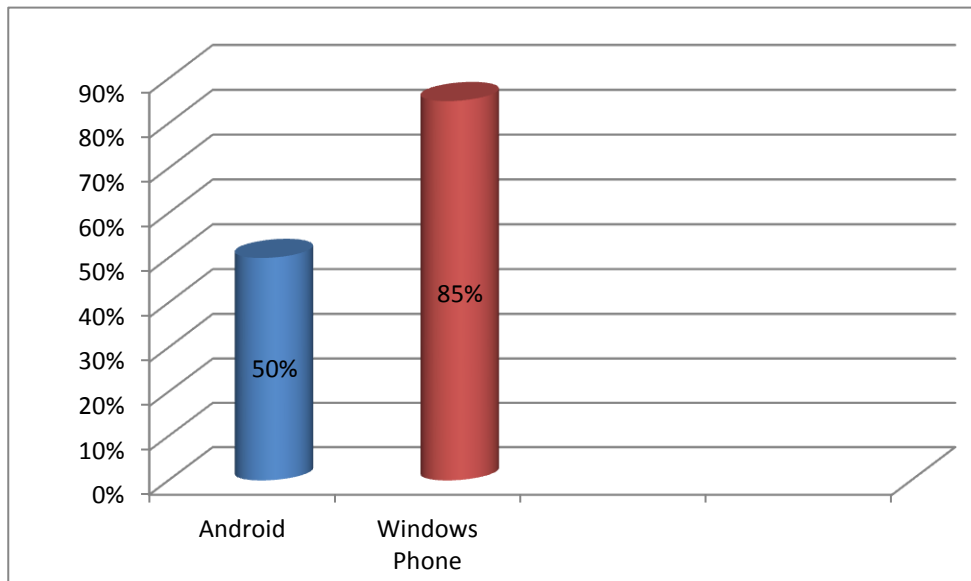


Figura V 6 Resultados Finales

Fuente: Propia de las autoras

Una vez que se han obtenido los resultados como se observa en Tabla V.XXIV y por observación directa se concluye que en la hipótesis planteada: el sensor de Windows Phone obtuvo el valor más alto de 34 puntos mientras que Android obtuvo 20 puntos, por lo que el sensor de Windows Phone superó en un 35 % al

sensor de Android, El sensor de Windows Phone es más sensible con un 85 % que equivale a muy bueno. Por lo que se concluye que el sensor de Windows Phone es mejor al momento de capturar los datos de un movimiento sísmico.

Conclusiones

- El sensor de Android funciona con aceleración y el de Windows phone con desplazamiento esto se dedujo, por los diferentes valores que toma al cambiar de orientación los dos móviles, por ejemplo, el de Android en el eje z devuelve +9.8 (el valor de la aceleración de la gravedad), al cambiar de orientación al revés -9.8 es decir la aceleración negativa. En el caso del WPH, nos devuelve el desplazamiento y esto se deduce porque los valores son demasiado pequeños, muy ajenos a la aceleración, por lo que varían con forme al movimiento y no a la orientación del dispositivo.
- Con los resultados obtenidos se concluye que el sensor de Windows phone es más sensible para la detección de sismos por que obtuvo un 85 % comparado con el sensor de Android que obtuvo un porcentaje de 50 % por lo que en este estudio el sensor del Smartphone Nokia Lumia 720 es mejor al momento de capturar los datos que el Samsung Galaxy SII.
- Para la captura de los datos en los Smartphone se utilizó una ecuación logarítmica la cual nos permitió obtener un valor que se aproxima a la escala de Richter.
- Mediante el estudio que se realizó para el desarrollo de las aplicaciones el Smartphone que utiliza Windows Phone es más fácil de utilizar porque requirió menos componentes y menos líneas de código al momento de

desarrollar la aplicación; con un porcentaje de 92% mientras que en Android se obtuvo el 58,33%.

- Para la instalación de las aplicaciones en los Smartphone el tiempo que se demora en la instalación en el celular Nokia Lumia 720 es más rápido con un porcentaje de 100% comparado con el tiempo de instalación en el Smartphone Samsung Galaxy SII con un 25%.

Recomendaciones

- Se recomienda realizar un estudio más profundo acerca de los tipos de movimientos que puede tener un Smartphone por que los sensores que se encuentran en este tipo de teléfonos capturan valores según como fue desarrollado el sensor.
- Es recomendable establecer bien los parámetros para la comparación de los sensores para poder escoger cual es el mejor en la detección de sismos.
- Al momento de realizar las pruebas los dos Smartphone deben de iniciarse simultáneamente para poder determinar cuál Smartphone captura más datos en un intervalo de tiempo.
- Una vez concluido el estudio es aconsejable utilizar el sensor del Smartphone Nokia Lumia 720 antes que el sensor del Samsung Galaxy SII por que el sensor del Smartphone Nokia Lumia se aproxima a datos más reales.

RESUMEN

Investigación para el estudio estadístico comparativo entre acelerómetros android y Windows phone aplicado en la detección de movimientos telúricos desarrollado en la Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica de la Escuela de Ingeniería en Sistemas.

Se aplicó el método científico usando la técnica de observación directa, para desarrollar los prototipos se utilizó metodología Programación Extrema (XP), se implementaron las aplicaciones utilizando dos plataformas: una de software libre llamada eclipse y el Software Development Kit (SDK) para Android, Visual estudio 2012 con SDK para Windows Phone, bajo sistemas operativos Windows 7 de 64 bits y Windows 8 pro de 64 bits respectivamente. Se estableció parámetros comparativos para determinar cuál sensor es mejor en el momento de recolectar datos en un movimiento sísmico.

Se realizó dos aplicaciones con las mismas características en dos plataformas: eclipse con SDK para Android, Visual Studio 2012 con SDK para Windows phone.

De acuerdo al análisis comparativo según 4 parámetros: codificación, Lenguajes de programación, instalación y rendimiento obtuvimos un 85 % para el sensor Windows phone comparado con un 50 % para el sensor Android.

Concluimos que Windows Phone es mejor capturando datos de un movimiento sísmico que un sensor Android.

Se recomienda realizar estudios acerca del funcionamiento del acelerómetro en un Smartphone, esto permitirá capturar con más exactitud los datos recolectados.

Palabras Claves: ESTUDIO COMPARATIVO, ACELERÓMETRO ANDROID, ACELERÓMETRO WINDOWS PHONE, DETECCIÓN DE SISMOS.

SUMMARY

This is a Research for comparative statistical study between Android and Windows phone accelerometers applied in detecting earthquakes which has been developed in the Escuela Superior Politécnica de Chimborazo, Computer and Electronics Faculty, School of Engineering System.

The scientific method was applied using the technique of direct observation, to develop prototypes, methodology Extreme Programming (XP) was used, by using two platforms applications were implemented: a free software called Eclipse and Development Kit (SDK) for Android software, Visual Studio 2012 SDK for Windows Phone, Windows 7 64-bit operating system and Windows 8 pro 64-bit respectively. Parameters were established to determine which sensor is better at the time of collecting data in an earthquake .

Two applications with the same characteristics were performed on two platforms: will eclipse with Android SDK, Visual Studio 2012 with Windows phone SDK.

In Accord with the comparative analysis and according to four parameters these results were found: codification, programming language, installation and performance we obtained an 85 % for Windows phone sensor, compared with a 50% for the Android sensor.

It is concluded that Windows Phone software is better than Android when collecting data in an earthquake.

It is recommend a study about the performance of the accelerometer in a smartphone, which will allow collecting data accurately.

CAPÍTULO VI

BIBLIOGRAFÍA

1. **Fernández, J.**, Introducción a las Metodologías Ágiles., España., Barcelona-Universidad Oberta de Catalunya., 2013., Pp 19.
2. **Gargenta, M.**, Learning Android., Nueva York–Estados Unidos., O’Reilly., 2011., Pp: 260- 280.
3. **Roche, K.**, Benning Java Google App Engine., Nueva York–Estados Unidos., O’Reilly., 2009., Pp: 330- 420.
4. **ACELERÓMETRO**
<http://tomatek.wordpress.com/que-es-acelerometro/>
2013-11-12
5. **ACELERÓMETRO ANDROID**
<http://www.europapress.es/portaltic/software/google->
2013-11-14

6. ARQUITECTURA DE ANDROID

<http://www.configurarequipos.com/doc1107.html>

2013-12-12

7. CARACTERÍSTICAS DE ANDROID

<http://www.configurarequipos.com/doc1107.html>

2013-12-10

8. CARACTERÍSTICAS DE LOS SMARTPHONE

<http://www.areatecnologia.com/un-smartphone.htm>

2013-10-2

9. DEFINICIÓN DE SENSORES

<http://www.androidcurso.com/index.php/android/36>

2013-10-6

10. DISEÑO DE UN ENTORNO MÓVIL

<http://rita.det.uvigo.es/201011/uploads/ITA.2010.V5.N4>

2013-06-20

11. DISPOSITIVOS MÓVILES

<http://exa.unne.edu.ar/depar/areas/informatica/>

2013-05-31

12. DISTRIBUCIÓN DE APLICACIONES

<https://support.google.com/googleplay/android-deve>

2013-12-20

13. ESTUDIO DE KANTAR WORLDPANEL

<http://www.kantarworldpanel.com/global/News/news-ar>

2014-01-08

14. ESTUDIO DE PLATAFORMAS SMARTPHONE

<http://www.channelbiz.es/2013/08/14/samsung>

2014-11-08

15. ESTUDIO DE STRATEGY ANALYTICS

<http://blogs.strategyanalytics.com/WSS/post.aspx>

2014-01-08

16. FUNCIONES DE LOS SMARTPHONE

<http://histinf.blogs.upv.es/2012/12/03/smartphones/>
2013-10-20

17. GIROSCOPIO

<http://www.poderpda.com/editorial/conociendo-los>
2013-10-21

18. INDUSTRIA DE LOS SMARTPHONE

<http://www.expansion.com/2013/04/01/empresas>
2013-10-21

19. INSTALACIÓN DE ECLIPSE

<http://proyectosbeta.net/2012/11/instalar-la-ultima>
2013-12-20

20. INSTALACIÓN DEL PLUGIN DE ANDROID

<http://developer.android.com/sdk/installing/installing>
2013-12-21

21. MÓVILES DE ÚLTIMA GENERACIÓN

<http://www.consumoastur.es/opencms/consumo>

2013-06-13

22. SDK DE ANDROID

<http://developer.android.com/sdk/index.html>

2013-12-21

23. SISTEMA OPERATIVO WINDOWS PHONE

<https://www.windowsphone.com/es-ec>

2013-12-21

24. TIENDA DE APLICACIONES

<http://developer.android.com/google/playindex.html>

2013-12-20

25. VERSIONES DE ANDROID

<http://www.androidcurso.com/index.php/curso-android>

2013-12-10

26. VERSIONES DE WINDOWS PHONE

<http://windowsphoneapps.es/noticias/windows-phone>

2013-10-20

ANEXOS

ANEXOS 1. HISTORIAS DE USUARIO

Captura de los parámetros X, Y, y Z en WP y Android

Tabla IV.II. Historia de Usuario – Captura de los parámetros X, Y, y Z en WP y Android

HISTORIA DE USUARIO	
Historia Nro.: 01	Nombre de la historia: Captura de los parámetros X, Y, y Z en WP y Android
Usuario:	Iteración Asignada: 1
Prioridad en negocio: Alta	
Riesgo en Desarrollo: Alta	
Programador Responsable: Johanna Honores, Jenny Vizuite.	
Descripción: Se realiza una interfaz para mostrar los parámetros recogidos por el sensor Acelerómetro de X, Y, y Z al ocurrir el movimiento (sismo).	
Observaciones: Estos parámetros se registran con un id único generado por el sistema.	

Fuente: Propia de las autoras

Captura de los parámetros latitud y longitud en WP y Android

Tabla IV.III. Historia de Usuario - Captura de los parámetros latitud y longitud en WP y Android

HISTORIA DE USUARIO	
Historia Nro.: 02	Nombre de la historia: Captura de los parámetros latitud y longitud en WP y Android
Usuario:	Iteración Asignada: 1
Prioridad en negocio: Alta	
Riesgo en Desarrollo: Alta	
Programador Responsable: Johanna Honores, Jenny Vizuite.	
Descripción: Se realiza una interfaz para mostrar los parámetros de latitud y longitud recogidos por el GPS del Smartphone en el momento del movimiento (sismo).	
Observaciones: Estos parámetros se registran con un id único generado por el sistema.	

Fuente: Propia de las autoras

Calculo de la magnitud del sismo (algoritmo) en WP y Android

Tabla IV. IV. Historia de Usuario - Calculo de la magnitud del sismo (algoritmo) en WP y Android

HISTORIA DE USUARIO	
Historia Nro.: 03	Nombre de la historia: Calculo de la magnitud del sismo (algoritmo) en WP y Android
Usuario:	Iteración Asignada: 2
Prioridad en negocio: Alta	
Riesgo en Desarrollo: Alta	
Programador Responsable: Johanna Honores, Jenny Vizuite.	
Descripción: Se procede al cálculo del valor de la magnitud del sismo, usando los parámetros X, Y y Z. Posteriormente se muestra en una interfaz emitiendo una alerta sonora.	
Observaciones:	

Fuente: Propia de las autoras

Salida de gráficos estadísticos en WP y Android

Tabla IV. V. Historia de Usuario - Salida de gráficos estadísticos en WP y Android

HISTORIA DE USUARIO	
Historia Nro.: 04	Nombre de la historia: Salida de gráficos estadísticos en WP y Android
Usuario:	Iteración Asignada: 3
Prioridad en negocio: Alta	
Riesgo en Desarrollo: Media	
Programador Responsable: Johanna Honores, Jenny Vizuite.	
Descripción: A través de una página web se muestra una serie de gráficos que permite la comparación de los 2 Smartphone a través de los datos recogidos en cada uno de ellos.	
Observaciones:	

Fuente: Propia de las autoras

ANEXO 2 TABLAS DE HOJAS DE RIESGOS

TABLA IV.XVII. Hoja de Gestión de Riesgo 2

HOJA DE GESTIÓN DEL RIESGO			
ID DEL RIESGO: R2			
Probabilidad: 40 Valor: 2	Impacto: Medio Valor: 2	Exposición: Medio Valor: 2	Prioridad: Alta Valor: 6
Descripción: Definición de requerimiento ambigua para las aplicaciones			
REFINAMIENTO: Causas: <ul style="list-style-type: none">• Mala comunicación con el cliente• Falta de coordinación entre los miembros del equipo• Acumulación de Información Consecuencias: <ul style="list-style-type: none">• Afecta el desarrollo de las aplicaciones• El proyecto no podrá ser entregado en la fecha planificada			
REDUCCIÓN Y SUPERVISIÓN: <ul style="list-style-type: none">• Establecer acuerdos entre los miembros del equipo que lleven al mejoramiento del desarrollo de las aplicaciones.• Reorganización de información			
SUPERVISIÓN: <ul style="list-style-type: none">• Seguimiento continuo en el cumplimiento de requerimientos• Evitar modificaciones frecuentes de los requerimientos establecidos.			
GESTIÓN: Plan de Contingencia: <ul style="list-style-type: none">• Establecer claramente los requerimientos con el usuario• Establecer las limitaciones del proyecto de tesis.			
ESTADO ACTUAL Fase de reducción iniciada <input checked="" type="checkbox"/> Fase de Supervisión iniciada <input type="checkbox"/> Gestionando el riesgo <input type="checkbox"/>			
RESPONSABLES: Johanna Honores Jenny Vizquete			

Fuente: Propia de las autoras

TABLA IV.XVIII. Hoja de Gestión de Riesgo 3

HOJA DE GESTIÓN DEL RIESGO			
ID DEL RIESGO: R3			
Probabilidad: 10 Valor: 1	Impacto: Bajo Valor: 1	Exposición: Alto Valor: 3	Prioridad: Medio Valor: 3
Descripción: Aparición de nuevos requerimientos cuando las aplicaciones están en las fases finales			
REFINAMIENTO:			
Causas: <ul style="list-style-type: none"> • Mala comunicación con el cliente • Falta información sobre los requerimientos de las aplicaciones • Retraso en la entrega del proyecto de tesis 			
Consecuencias: <ul style="list-style-type: none"> • Afecta el desarrollo de las aplicaciones • El proyecto no podrá ser entregado en la fecha planificada 			
REDUCCIÓN Y SUPERVISIÓN:			
<ul style="list-style-type: none"> • Establecer acuerdos entre los miembros del equipo que lleven al mejoramiento del desarrollo de las aplicaciones. • Reorganización de información 			
SUPERVISIÓN:			
<ul style="list-style-type: none"> • Seguimiento continuo en el cumplimiento de requerimientos • Evitar modificaciones frecuentes de los requerimientos establecidos. 			
GESTIÓN:			
Plan de Contingencia: <ul style="list-style-type: none"> • Establecer claramente los requerimientos con el usuario desde el principio • Mantener una buena comunicación con el usuario para evitar cambios en el futuro. 			
ESTADO ACTUAL			
Fase de reducción iniciada		<input checked="" type="checkbox"/>	
Fase de Supervisión iniciada		<input type="checkbox"/>	
Gestionando el riesgo		<input type="checkbox"/>	
RESPONSABLES:			
Johanna Honores			
Jenny Vizquete			

Fuente: Propia de las autoras

TABLA IV.XIX. Hoja de Gestión de Riesgo 4

HOJA DE GESTIÓN DEL RIESGO			
ID DEL RIESGO: R4			
Probabilidad: 10 Valor: 2	Impacto: Bajo Valor: 1	Exposición: Crítico Valor: 4	Prioridad: Medio Valor: 4
Descripción: Modificación del cronograma de actividades			

<p>REFINAMIENTO: Causas:</p> <ul style="list-style-type: none"> • Aparición de nuevas tareas • Falta de coordinación entre los miembros del equipo • Retraso en la entrega de las tareas programadas <p>Consecuencias:</p> <ul style="list-style-type: none"> • Afecta el desarrollo de las aplicaciones • El proyecto no podrá ser entregado en la fecha planificada
<p>REDUCCIÓN Y SUPERVISIÓN:</p> <ul style="list-style-type: none"> • Establecer acuerdos entre los miembros del equipo que lleven al mejoramiento del desarrollo de las aplicaciones. • Reorganización del cronograma de actividades
<p>SUPERVISIÓN:</p> <ul style="list-style-type: none"> • Seguimiento continuo en el cumplimiento de las tareas programadas. • Evitar modificaciones frecuentes de los requerimientos establecidos.
<p>GESTIÓN: Plan de Contingencia:</p> <ul style="list-style-type: none"> • Establecer claramente los requerimientos con el usuario • Establecer las limitaciones del proyecto de tesis.
<p>ESTADO ACTUAL</p> <p>Fase de reducción iniciada <input checked="" type="checkbox"/></p> <p>Fase de Supervisión iniciada <input type="checkbox"/></p> <p>Gestionando el riesgo <input type="checkbox"/></p>
<p>RESPONSABLES: Johanna Honores Jenny Vizúete</p>

Fuente: Propia de las autoras

TABLA IV.XX. Hoja de Gestión de Riesgo 5

HOJA DE GESTIÓN DEL RIESGO			
ID DEL RIESGO: R5			
Probabilidad: 50 Valor: 2	Impacto: Medio Valor: 2	Exposición: Alto Valor: 3	Prioridad: Medio Valor: 3
Descripción: Falta de conocimiento en la utilización de las herramientas de desarrollo de las aplicaciones			
<p>REFINAMIENTO: Causas:</p> <ul style="list-style-type: none"> • Falta de conocimiento sobre las herramientas de desarrollo de las aplicaciones • Incompatibilidad entre las herramientas seleccionadas para las aplicaciones <p>Consecuencias:</p> <ul style="list-style-type: none"> • Afecta el desarrollo del sistema 			

<ul style="list-style-type: none"> El proyecto no podrá ser entregado en la fecha planificada
REDUCCIÓN Y SUPERVISIÓN: <ul style="list-style-type: none"> Establecer acuerdos entre los miembros del equipo para seleccionar las herramientas más adecuadas para el desarrollo de las aplicaciones Auto educarse para conocer sobre las herramientas.
SUPERVISIÓN: <ul style="list-style-type: none"> Seguimiento acerca de la selección correcta de las herramientas de desarrollo Realizar pruebas de compatibilidad acerca de las herramientas de desarrollo
GESTIÓN: Plan de Contingencia: <ul style="list-style-type: none"> Establecer desde un principio herramientas adecuadas para el desarrollo de las aplicaciones Establecer las limitaciones del proyecto de tesis.
ESTADO ACTUAL Fase de reducción iniciada <input checked="" type="checkbox"/> Fase de Supervisión iniciada <input type="checkbox"/> Gestionando el riesgo <input type="checkbox"/>
RESPONSABLES: Johanna Honores Jenny Vizuete

Fuente: Propia de las autoras

TABLA IV.XXI. Hoja de Gestión de Riesgo 6

HOJA DE GESTIÓN DEL RIESGO			
ID DEL RIESGO: R6			
Probabilidad: 50 Valor: 2	Impacto: Bajo Valor: 1	Exposición: Medio Valor: 4	Prioridad: Bajo Valor: 2
Descripción: Las aplicaciones no cubren todas las expectativas del usuario			
REFINAMIENTO:			
Causas: <ul style="list-style-type: none"> Mala recolección de información al momento de definir los requerimientos. Las aplicaciones no fueron terminadas al tiempo acordado 			
Consecuencias: <ul style="list-style-type: none"> Afecta el desarrollo de las aplicaciones El proyecto no podrá ser entregado en la fecha planificada 			
REDUCCIÓN Y SUPERVISIÓN: <ul style="list-style-type: none"> Establecer acuerdos entre los miembros del equipo que lleven al mejoramiento del desarrollo de las aplicaciones. Reorganización de información 			

SUPERVISIÓN:	
<ul style="list-style-type: none"> • Seguimiento continuo en el cumplimiento de requerimiento definidas anteriormente junto con el usuario • Evitar modificaciones frecuentes de los requerimientos establecidos. 	
GESTIÓN:	
Plan de Contingencia:	
<ul style="list-style-type: none"> • Establecer claramente los requerimientos con el usuario • Establecer las limitaciones del proyecto de tesis. 	
ESTADO ACTUAL	
Fase de reducción iniciada	<input checked="" type="checkbox"/>
Fase de Supervisión iniciada	<input type="checkbox"/>
Gestionando el riesgo	<input type="checkbox"/>
RESPONSABLES:	
Johanna Honores	
Jenny Vizquete	

Fuente: Propia de las autoras

ANEXOS 3. TABLAS CRC

Tabla IV. XXIII. Tarjeta CRC - Tabla param_lecturas

param_lecturas
Recoge el valor de la primera lectura.
Recoge el valor del total de lecturas.
Recoge el valor del promedio de lecturas.
Recoge el valor del tiempo que se demora para emitir la alerta.
Recoge el valor de la magnitud del sismo.

Fuente: Propia de las autoras

Tabla IV. XXIV. Tarjeta CRC - Tabla usuario

Usuario
Permite el acceso de varios usuarios a través del logueo de sus credenciales (usuario y contraseña).

Fuente: Propia de las autoras

Anexo 4. Reportes

4/21/2014

4/21/2014

PROMEDIO ANDROID: 91

PROMEDIO WINDOWS PHONE: 203

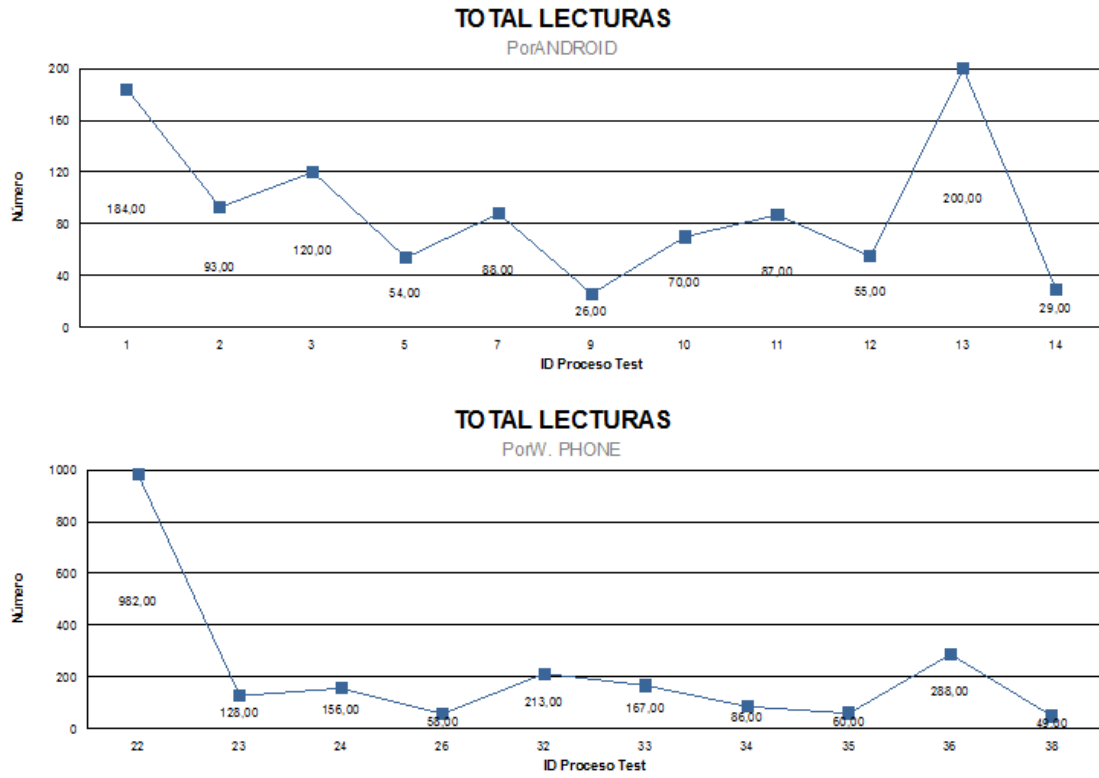


Figura IV. 13. Total de lecturas entre Android y Windows Phone

Fuente: Propia de las autoras

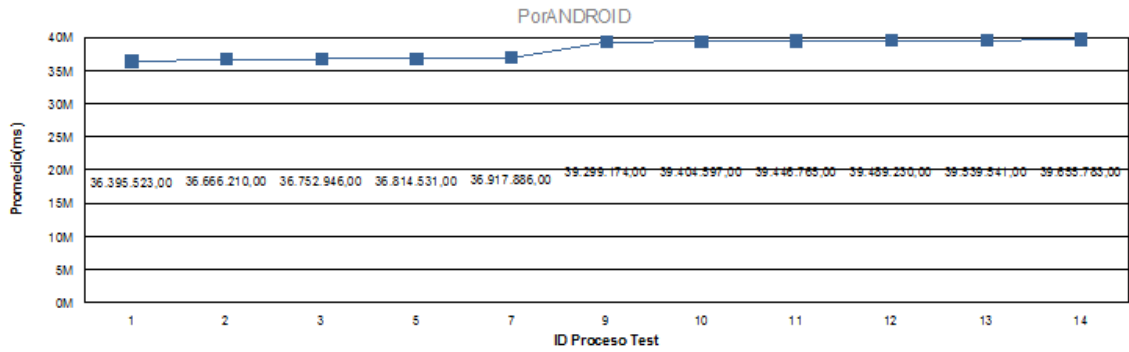
4/21/2014

4/21/2014

PROMEDIO ANDROID: 38216562

PROMEDIO WINDOWS PHONE: 38454119

TIEMPO HASTA EMITIR ALERTA



TIEMPO HASTA EMITIR ALERTA

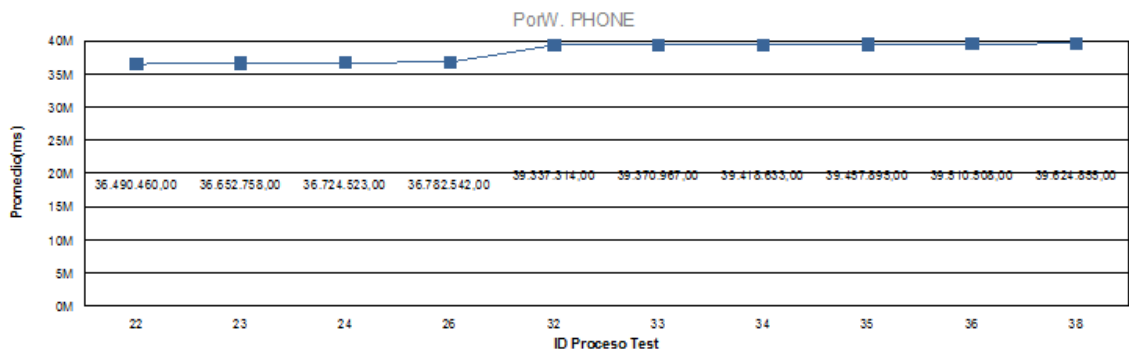


Figura IV. 14. Tiempo hasta emitir la alerta entre Android y Windows Phone

Fuente: Propia de las autoras

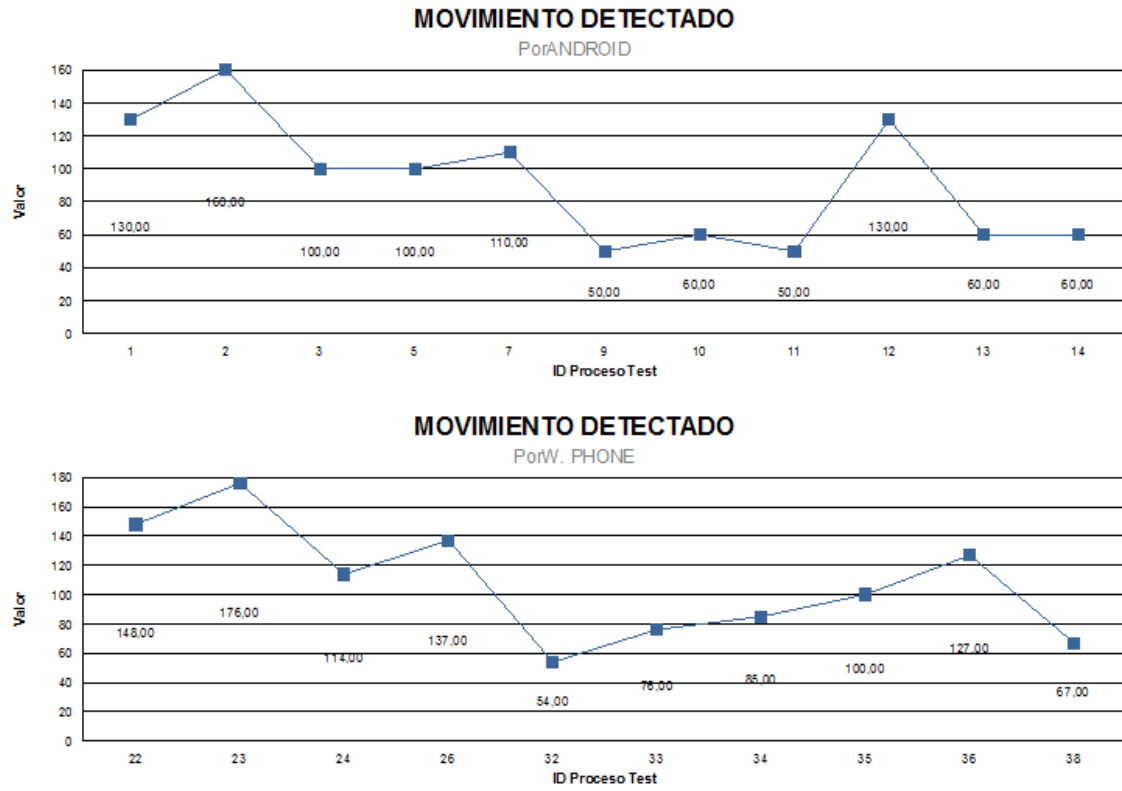


Figura IV. 15. Movimiento detectado en Android y Windows Phone

Fuente: Propia de las autoras

Anexo 5.

MESA VIBRATORIA



El objetivo principal de la mesa vibratoria es simular un movimiento sísmico, el cual podemos controlar la intensidad del sismo.

Elementos:

- Motor de máquina de
- 1 resistencia de $3.3\text{ k } \frac{1}{2}\text{ w}$
- 1 resistencia de $10\text{ k } \frac{1}{2}\text{ w}$
- 1 resistencia de $27\text{ k } \frac{1}{2}\text{ w}$
- 2 condensadores no polarizados de $0.1\text{ uF } /100\text{ v}$
- 1 diodo diac
- 1 triac BT137 -600 F
- 1 potenciómetro de 50 k
- Madera triplex
- 1 enchufe
- 1 barra de metal de 30 cm
- 4 ruedas locas

El presente circuito es conocido también como atenuador de luz para focos incandescentes o filamento, cuyo funcionamiento se basa en el rendimiento del triac BT137F o similar y el diodo bidireccional DIAC el que controla la cantidad

de potencia que es transmitida a través del triac para controlar la intensidad de la luz, el diodo DIAC no tiene polaridad.

Este circuito puede tener muchas aplicaciones como crear diferentes tipos de ambientes, fabricar lámparas con efectos luminosos, controladores de revoluciones pequeños motores de AC como motores de batidores, y muchas otras cosas más.



MANUAL TÉCNICO

ESTUDIO ESTADÍSTICO COMPARATIVO ENTRE SENSORES ANDROID Y WINDOWS PHONE APLICADO EN LA DETECCIÓN DE MOVIMIENTOS TELÚRICOS

Aplicación para Android

Para la realización de la aplicación de Android se utiliza las siguientes herramientas:

- Eclipse Juno
- SDK Android
- API 16, 17, 18, 19
- Motor de base de datos Microsoft SQL Server 2012

En la sección “3.3.1.11. INSTALACIÓN DE ECLIPSE CON EL SDK DE ANDROID” del informe de tesis se encuentra detallada la información de la instalación.

Con estas herramientas se elaboró el apk para Android en la versión 4.0.4.

Aplicación para Windows Phone

Para la aplicación de Windows phone se utilizó las siguientes herramientas:

- Visual Studio 2012
- SDK para Windows Phone 8.0
- Motor de base de datos Microsoft SQL Server 2012

En la sección “3.3.2.11. INSTALACIÓN DE VISUAL STUDIO CON EL SDK DE WINDOWS PHONE 8” del informe de tesis se encuentra detallada la información

Con estas herramientas se elaboró el xap para la versión Windows Phone 8.0.

PROCESOS

Aquí se describirá la manera en que se utiliza el sensor del Smartphone para la captura de datos y la utilización del GPS para mostrar la latitud y longitud del sismo.

Código para la intensidad del sismo para Android

```
if (tiempo > 0) {  
  
    Float my0 = Math.abs(prevY * tiempo*tiempo)/2;  
    Float mx0 = Math.abs(prevX * tiempo*tiempo)/2;  
    Float mz0 = Math.abs(prevZ * tiempo*tiempo)/2;  
  
    Float myf = Math.abs(curY * tiempo*tiempo)/2;  
    Float mxf = Math.abs(curX * tiempo*tiempo)/2;  
    Float mzf = Math.abs(curZ * tiempo*tiempo)/2;  
  
    Float mov_x =Math.abs(mxf-mx0) ;  
    Float mov_y =Math.abs(myf-my0) ;  
    Float mov_z =Math.abs(mzf-mz0) ;  
  
    Float movement=Float.parseFloat("0.0");  
movement =(float) Math.pow(( Math.pow(mov_x,2) + Math.pow(mov_y,2) + Math.pow(mov_z,2)),0.5);  
  
    Double grados = Logaritmo(movement,2);  
  
    TextView objTxtIdLimite = (TextView) findViewById(R.id.txtLimite);  
    TextView objTxtIdMinimo = (TextView) findViewById(R.id.txtValorMinimo);
```

```

        listAdapterX.add(""+contador) ;
        listAdapterY.add(""+tiempoHastaPrimeraLecturaTest);
        listAdapterZ.add(movement.toString());

        ListView lstX = (ListView) findViewById(R.id.LstX);
        ListView lstY = (ListView) findViewById(R.id.LstY);
        ListView lstZ = (ListView) findViewById(R.id.LstZ);

        lstX.setAdapter( listAdapterX );
        lstY.setAdapter( listAdapterY );
        lstZ.setAdapter( listAdapterZ );

        try
        {

                catch (Exception ex)
                {
Rastro.EscribirEnConsola(nombreClase, "E", ex.getMessage(),"cmdActivar_Onclick");
Toast.makeText(getApplicationContext(), "El Valor Límite o Valor" +
"r Mínimo no son correctos, el detalle es: " + ex.getMessage() ,Toast.LENGTH_SHORT).show();
                return;
                }

                if ((movement > 50) ) {

                        this.IniciarAcelerometro = false;

AlertDialog.Builder alertDialog = new AlertDialog.Builder(MainActivity.this);
alertDialog.setTitle(" Alerta movimiento ");

String mensaje=" Se ha detectado movimiento de :" + grados +" Grados Richter, ";

                fecha = GregorianCalendar.getInstance();
                horas = fecha.get(Calendar.HOUR_OF_DAY);
                minutos = fecha.get(Calendar.MINUTE);
                segundos = fecha.get(Calendar.SECOND);
                msecond = fecha.get(Calendar.MILLISECOND);
                tiempoHastaEmitirAlarma = (long)( horas * 3600000 + minutos * 60000 +
segundos *1000 + msecond);

                tiempo_alarma=tiempoHastaEmitirAlarma - tiempoHastaPrimeraLecturaTest;

                objTxtIdLimite.setText("lecturas "+contador);
                objTxtIdMinimo.setText(" "+tiempo_alarma+" ms");
                tiempoPromedioLecturas = tiempo_alarma;

                alertDialog.setMessage(mensaje);

                alertDialog.setNegativeButton("Aceptar", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
                        dialog.dismiss();
                }
                });

```

Código para la latitud y la longitud del Sismo Android

```

public class MyLocationListenerNetwork implements LocationListener
{

```

```

@Override
public void onLocationChanged(Location loc)
{
    nlat = loc.getLatitude();
    nlng = loc.getLongitude();

    TextView objTxtIdLatitud = (TextView) findViewById(R.id.txtLatitud);
    Double latitud = loc.getLatitude();
    objTxtIdLatitud.setText(nlat.toString());

    TextView objTxtIdLongitud = (TextView) findViewById(R.id.txtLongitud);
    Double longitud = loc.getLongitude();
    objTxtIdLongitud.setText(nlng.toString());

    Log.d("LAT & LNG Network:", nlat + " " + nlng);
}

```

Código para la intensidad del sismo de Windows Phone 8.0

```

private void CurrentValueChanged(MotionReading e)
{
    long tiempo = 10;
    if (!recibirDataSensor)
        return;
    if (motion.IsDataValid || this.simularMovimiento)
    {
        if (this.primeralectura)
        {
            try
            {
                horaTestActual = DateTime.Now;
            }
            catch (Exception ex) {
                MessageBox.Show("error: " + ex.ToString());
            }

            this.lecturaAnterior = horaTestActual;
            primeralectura = false;
        }
        else
        {
            this.lecturaAnterior = horaTestActual;
        }
        totalLecturasTest++;
        this.totalLecturasTest= Math.Round(totalLecturasTest);
        curX = e.DeviceAcceleration.X * 100;
        curY = e.DeviceAcceleration.Y * 100;
        curZ = e.DeviceAcceleration.Z * 100;
        horaTestActual = DateTime.Now;
        long horas = (long)horaTestActual.Hour;
        long minutos = (long)horaTestActual.Minute;
        long segundos = (long)horaTestActual.Second;
        this.tiempoHastaPrimeralecturaTest = horas * 3600000 + minutos * 60000 +
segundos * 1000 + (long)horaTestActual.Millisecond;
        this.txtValorX.Text = "X: " + curX.ToString("0.00000");
        this.txtValorY.Text = "Y: " + curY.ToString("0.00000");
        this.txtValorZ.Text = "Z: " + curZ.ToString("0.00000");

        if (prevX == 0 && prevY == 0 && prevZ == 0)
        {

```



```
    }  
}  
  
void watcher_PositionChanged(object sender, GeoPositionChangedEventArgs<GeoCoordinate> e)  
{  
    this.txtLatitud.Text = e.Position.Location.Latitude.ToString("0.00000");  
    this.txtLongitud.Text = e.Position.Location.Longitude.ToString("0.00000");  
}
```