

	UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA			
	Documento	Código	Fecha	Revisión
	FORMATO HOJA DE RESUMEN PARA TRABAJO DE GRADO	F-AC-DBL-007	10-04-2012	A
Dependencia	Aprobado		Pág.	
DIVISIÓN DE BIBLIOTECA	SUBDIRECTOR ACADEMICO		1(155)	

RESUMEN – TRABAJO DE GRADO

AUTORES	ÁLVARO JAVIER DURÁN SANJUÁN JORGE LUIS PEINADO RODRÍGUEZ		
FACULTAD	INGENIERÍAS		
PLAN DE ESTUDIOS	INGENIERÍA DE SISTEMAS		
DIRECTOR	M.Sc. ALVEIRO ALONSO ROSADO GÓMEZ		
TÍTULO DE LA TESIS	CONFRONTACIÓN DE DOS TECNOLOGÍAS DE DESARROLLO DE APLICACIONES MÓVILES SOBRE LA PLATAFORMA ANDROID 4.0 DESDE LA PERSPECTIVA DE LOS ATRIBUTOS DE CALIDAD.		
RESUMEN (70 palabras aproximadamente)			
<p>En el desarrollo de aplicaciones móviles para Android principalmente existen dos opciones: tecnología de desarrollo nativa y tecnología de desarrollo basado en web. Este trabajo se ocupa de confrontar dichas tecnologías para determinar cuál es la mejor con respecto al rendimiento.</p> <p>La confrontación se llevó a cabo evaluando una aplicación que se construyó en las dos tecnologías mencionadas, haciendo uso del proceso de evaluación propuesto en el estándar ISO/IEC 14598-1 junto con el modelo de calidad del producto de ISO/IEC 9126-1. Los resultados permitieron concluir que una aplicación construida con tecnología de desarrollo nativa tiene mejor rendimiento que aquella que se desarrolla con tecnología basada en web.</p>			
CARACTERÍSTICAS			
PÁGINAS: 154	PLANOS:	ILUSTRACIONES:	CD-ROM: 1



CONFRONTACIÓN DE DOS TECNOLOGÍAS DE DESARROLLO DE
APLICACIONES MÓVILES SOBRE LA PLATAFORMA ANDROID 4.0 DESDE LA
PERSPECTIVA DE LOS ATRIBUTOS DE CALIDAD

ÁLVARO JAVIER DURÁN SANJUÁN
JORGE LUIS PEINADO RODRÍGUEZ

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA
FACULTAD DE INGENIERÍAS
INGENIERÍA DE SISTEMAS
OCAÑA
2014

CONFRONTACIÓN DE DOS TECNOLOGÍAS DE DESARROLLO DE
APLICACIONES MÓVILES SOBRE LA PLATAFORMA ANDROID 4.0 DESDE LA
PERSPECTIVA DE LOS ATRIBUTOS DE CALIDAD

ÁLVARO JAVIER DURÁN SANJUÁN
JORGE LUIS PEINADO RODRÍGUEZ

Director
M.Sc. ALVEIRO ALONSO ROSADO GÓMEZ

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA
FACULTAD DE INGENIERÍAS
INGENIERÍA DE SISTEMAS
OCAÑA
2014

CONTENIDO

pág.

INTRODUCCIÓN	15
1. TÍTULO	16
1.1 PLANTEAMIENTO DEL PROBLEMA	16
1.2 OBJETIVOS DE LA INVESTIGACIÓN.....	16
1.2.1 Objetivo General	16
1.2.2 Objetivos Específicos.....	16
1.3 JUSTIFICACIÓN	17
1.4 DELIMITACIONES	17
1.4.1 Delimitación geográfica	17
1.4.2 Delimitación temporal.....	17
1.4.3 Delimitación conceptual.....	17
1.4.4 Delimitación operativa	20
2. MARCO REFERENCIAL	22
2.1 MARCO HISTÓRICO.....	22
2.2 MARCO CONCEPTUAL.....	30
2.3 MARCO TEÓRICO.....	37
2.3.1 Guías no estudiadas.....	38
2.3.2 Enfoques de desarrollo de aplicaciones móviles.....	40
2.3.2.1 Clientes nativos.....	40
2.3.2.2 Clientes JME (Java Platform Micro Edition, Plataforma Java Micro Edición).	41
2.3.2.3 Clientes basados en web.....	41
2.3.2.4 Clientes basados en middleware.	41
2.3.3 Aplicación nativa o web según estudio de Global Intelligence Alliance.....	42
2.3.3.1 Aplicación Nativa.....	43
2.3.3.2 Aplicación Web.....	43
2.4 MARCO LEGAL	44
2.4.1 Artículo 71 de la Constitución Política de Colombia	44
2.4.2 Decreto 393 de 1991	45
2.4.3 Ley 1341 de 2009.....	45
3. DISEÑO METODOLÓGICO	46
3.1 TIPO DE INVESTIGACIÓN	46
3.2 POBLACIÓN	46
3.3 MUESTRA.....	46

3.4 TÉCNICA DE RECOLECCIÓN	47
3.5 INSTRUMENTO	48
3.6 ANÁLISIS DE INFORMACIÓN	49
3.7 DIAGNÓSTICO SITUACIONAL.....	55
4. PRESENTACIÓN DE RESULTADOS	56
4.1 CONFRONTAR DOS TECNOLOGÍAS DE DESARROLLO DE APLICACIONES MÓVILES SOBRE LA PLATAFORMA ANDROID 4.0 DESDE LA PERSPECTIVA DE LOS ATRIBUTOS DE CALIDAD	56
4.1.1 Definir el proceso de evaluación de los atributos de calidad del software.	
4.1.1.1 Establecimiento de los requisitos de la evaluación	59
4.1.1.2 Especificación de la evaluación.	67
4.1.1.3 Diseño de la evaluación.	74
4.1.2 Describir las tecnologías de desarrollo de aplicaciones móviles para Android.	77
4.1.2.1 Herramientas para el desarrollo de aplicaciones nativas para Android.	77
4.1.2.2 Herramientas para el desarrollo de aplicaciones basadas en web para Android.	84
4.1.3 Construir una aplicación móvil tanto en tecnología nativa como una basada en web que sirva como caso de estudio para el proceso de evaluación de los atributos de calidad del software	88
4.1.3.1 Descripción de la aplicación desarrollada.....	88
4.1.3.2 Herramientas de desarrollo empleadas.....	88
4.1.4 Efectuar el proceso de evaluación de los atributos de calidad del software tanto a la aplicación nativa como a la aplicación basada en web.	102
4.1.4.1 Ejecución de la evaluación.....	102
4.1.4.2 Resultados de la medición.....	103
4.1.4.3 Toma de datos.	105
4.1.4.4 Análisis de resultados.....	124
4.1.4.5 Conclusiones de la Evaluación de NewsApp.....	132
CONCLUSIONES	137
RECOMENDACIONES	139
BIBLIOGRAFÍA	140
INFOGRAFÍA	142
ANEXOS	149

LISTA DE CUADROS

	pág.
Cuadro 1. Beneficios e inconvenientes de los diferentes enfoques para el desarrollo de aplicaciones para dispositivos móviles	45
Cuadro 2. Datos del encuestado	49
Cuadro 3. Celulares organizados por marca y modelo	49
Cuadro 4. Cantidad de celulares básicos y smartphones	50
Cuadro 5. Persnas que tienen un dispositivo móvil diferente del celular	51
Cuadro 6. Cantidad de dispositivos móviles aparte del celular	51
Cuadro 7. Personas interesadas en desarrollar aplicaciones móviles	52
Cuadro 8. Personas vinculadas con el desarrollo de aplicaciones móviles	52
Cuadro 9. Personas que han realizado un curso de desarrollo de aplicaciones	53
Cuadro 10. Personas que han desarrollado una aplicación móvil	54
Cuadro 11. Desglose de la característica de calidad eficiencia	69
Cuadro 12. Requerimientos para la instalación de Little Eye	75
Cuadro 13. Herramientas para desarrollo nativo sobre la plataforma Android	83
Cuadro 14. Herramientas para desarrollo basado en web sobre la plataforma Android	86
Cuadro 15. Herramientas para desarrollo de interfaces gráficas para aplicaciones basadas en web sobre la plataforma Android	87
Cuadro 16. Historia de usuario de cargar noticias	92
Cuadro 17. Historia de usuario de ver detalle de la noticia	92
Cuadro 18. Tarea carga de titulares de noticias	93
Cuadro 19. Tarea ver detalle de la noticia	94
Cuadro 20. Historia de usuario lista de fuentes de noticias	96
Cuadro 21. Tarea de lista fuente de noticias	97
Cuadro 22. Diario de actividades de iteración 1	101
Cuadro 23. Diario de actividades de iteración 2	101
Cuadro 24. Especificaciones de hardware de Computador para monitoreo	102
Cuadro 25. Especificaciones de hardware de Sony Xperia S	103
Cuadro 26. Condiciones del teléfono previas a la evaluación	103
Cuadro 27. Guía de procedimiento para la toma de medidas en Little Eye	104
Cuadro 28. Evaluación de comportamiento temporal en NewsApp basada en web, métrica: tiempo de respuesta.	107

Cuadro 29. Evaluación de comportamiento temporal en NewsApp basada en web, métrica tiempo de respuesta del peor caso	108
Cuadro 30. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de memoria	109
Cuadro 31. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de memoria del peor caso	110
Cuadro 32. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de CPU	112
Cuadro 33. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de CPU del peor caso	114
Cuadro 34. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de batería	115
Cuadro 35. Evaluación de comportamiento temporal en NewsApp nativa, métrica: tiempo de respuesta	116
Cuadro 36. Evaluación de comportamiento temporal en NewsApp nativa, métrica: tiempo de respuesta del peor caso	117
Cuadro 37. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de memoria	118
Cuadro 38. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de memoria del peor caso	120
Cuadro 39. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de CPU	121
Cuadro 40. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de CPU del peor caso	123
Cuadro 41. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de batería	124
Cuadro 42. Resultados de la evaluación de NewsApp	132

LISTA DE FIGURAS

	pág.
Figura 1. Venta de celulares convencionales VS. Smartphones	18
Figura 2. Crecimiento de smartphones activos por país	19
Figura 3. Versiones de la plataforma Android para mayo de 2013	20
Figura 4. Número de aplicaciones por tienda oficial	25
Figura 5. Arquitectura de Android	32
Figura 6. Cantidad de celulares básicos y smartphones	50
Figura 7. Personas que tienen un dispositivo móvil diferente del celular	51
Figura 8. Personas interesadas en desarrollar aplicaciones móviles	52
Figura 9. Personas vinculadas con el desarrollo de aplicaciones móviles	53
Figura 10. Personas que han realizado un curso de desarrollo de aplicaciones móviles	53
Figura 11. Personas que han desarrollado una aplicación móvil	54
Figura 12. Comparativa de los modelos más utilizados de calidad de software	57
Figura 13. Proceso de Evaluación	58
Figura 14. Calidad más buscada en las aplicaciones móviles	62
Figura 15. Reacciones de los usuarios ante problemas en el rendimiento de una aplicación móvil	63
Figura 16. Principales problemas de las aplicaciones móviles	64
Figura 17. Arquitectura del lector RSS	67
Figura 18. Niveles de puntuación para las métricas	73
Figura 19. Número de aplicaciones por tienda oficial de la plataforma	77
Figura 20. Plataformas para desarrollo de aplicaciones basadas en web más usadas por los desarrolladores	90
Figura 21. Interfaz principal de NewsApp nativa de primer demo	94
Figura 22. Interfaz detalle de noticia de NewsApp nativa de primer demo	95
Figura 23. Interfaz principal de NewsApp basada en web de primer demo	95
Figura 24. Interfaz detalle de noticia de NewsApp basada en web de primer demo	96
Figura 25. Interfaz principal de NewsApp nativa de segundo demo	98
Figura 26. Interfaz lista titulares de noticias de NewsApp nativa de segundo demo	98
Figura 27. Interfaz detalle de noticia de NewsApp nativa de segundo demo	99
Figura 28. Interfaz principal de NewsApp basada en web de segundo demo	99
Figura 29. Interfaz lista titulares de noticias de NewsApp basada en web de segundo demo	100
Figura 30. Interfaz detalle de noticia de NewsApp basada en web de segundo demo	100
Figura 31. Interfaces de la aplicación NewsApp	104

Figura 32. Utilización máxima de memoria en NewsApp basada en web (MB)	111
Figura 33. Utilización máxima de CPU en NewsApp basada en web (%)	113
Figura 34. Utilización máxima de Memoria en NewsApp nativa (MB)	119
Figura 35. Utilización máxima de CPU en NewsApp nativa (%)	122
Figura 36. Tiempo de inicio promedio de NewsApp (s)	124
Figura 37. Tiempo de respuesta medio en la ejecución de una tarea (s)	124
Figura 38. Peor caso de tiempo de respuesta medio (s)	125
Figura 39. Consumo de memoria RAM en el inicio de la aplicación (MB)	125
Figura 40. Consumo de memoria RAM en el inicio de la aplicación con respecto al límite de memoria Dalvik (MB)	126
Figura 41. Consumo promedio de memoria RAM por tarea (MB)	126
Figura 42. Peak Native Memory (MB)	127
Figura 43. Límite de memoria de Dalvik para la aplicación (MB)	128
Figura 44. Peor caso promedio de consumo de memoria RAM por tarea (MB)	129
Figura 45. Consumo máximo de CPU en el inicio de la aplicación (%)	129
Figura 46. Promedio de uso máximo de CPU en la ejecución de una tarea (%)	130
Figura 47. Peor caso de uso máximo de CPU en la ejecución de una tarea (%)	130
Figura 48. Promedio de CPU utilizada por la aplicación durante el tiempo evaluado (%)	131
Figura 49. Total batería consumida durante el tiempo de la evaluación (mAh)	131
Figura 50. Tiempo de respuesta en NewsApp nativa VS. Tiempo de respuesta en NewsApp basada en web (s)	133
Figura 51. Utilización máxima de memoria en NewsApp nativa VS. Utilización máxima de memoria en NewsApp basada en web (kB)	134
Figura 52. Utilización máxima de CPU en NewsApp nativa VS. Utilización máxima de CPU en NewsApp basada en web (%)	135
Figura 53. NewsApp nativa VS. NewsApp basada en web en cuanto a utilización máxima de batería (mAh)	135

LISTA DE TABLAS

	pág.
Tabla 1. Métricas de comportamiento temporal: Tiempo de respuesta	70
Tabla 2. Métricas de utilización de Recursos: Utilización de recursos de memoria	71
Tabla 3. Métricas de utilización de Recursos: Utilización de recursos de CPU	72
Tabla 4. Métricas de utilización de Recursos: Utilización de recursos de batería	73

LISTA DE ANEXOS

	pág.
Anexo A. Instrumento de Recolección de información	149
Anexo B. Manual de instalación de NewsApp	152
Anexo C. Archivos .apk de NewsApp	155

INTRODUCCIÓN

En la actualidad existe un campo creciente en el desarrollo de software, se trata del mercado de las aplicaciones para dispositivos móviles que hoy representa grandes oportunidades en este sector. Aunque el desarrollo de dichas aplicaciones es relativamente nuevo, ya existen muchas herramientas que facilitan su construcción, y que además cobran alta importancia debido a la serie de plataformas móviles existentes, de las cuales, Android es la más sobresaliente, entre otras razones, porque está implantado fuertemente en el ámbito mundial.

El presente trabajo muestra la confrontación de dos tecnologías para el desarrollo de aplicaciones móviles sobre la plataforma Android 4.0 desde la perspectiva de los atributos de calidad del software. Las dos tecnologías que se comparan son: la tecnología nativa y la tecnología basada en web (Phonegap - jQuery Mobile), debido a su gran auge y a otras razones que se explican en este trabajo. Para efectos de la comparación se hizo uso de los estándares internacionales ISO/IEC 14598-1 y 9126-1, así como de una herramienta de monitorización y análisis de rendimiento llamada Little Eye.

Este informe de trabajo de grado se ha organizado de la siguiente manera: en el primer capítulo se describe el problema de donde partió el estudio, la justificación y los objetivos planteados; en el segundo capítulo se encuentran los aspectos teóricos, conceptuales y legales; en el tercer capítulo se presenta el diseño metodológico donde se hace un análisis de los datos recolectados, y en el cuarto capítulo se da a conocer el desarrollo de los objetivos propuestos al inicio de este trabajo. Finalmente se dan las conclusiones a las que se llegaron y una serie de recomendaciones.

1. TÍTULO

Confrontación de dos tecnologías de desarrollo de aplicaciones móviles sobre la plataforma Android 4.0 desde la perspectiva de los atributos de calidad.

1.1 PLANTEAMIENTO DEL PROBLEMA

El desarrollo de aplicaciones para dispositivos móviles es un tema ampliamente abordado por gran cantidad de sitios en internet tales como: blogs, foros y portales de empresas, asimismo, por una serie de autores que en su mayoría se enfocan principalmente en la programación misma de las aplicaciones, exponiendo técnicas, sugerencias, buenas prácticas y métodos, sin explicar con información respaldada por estudios, evaluaciones o referencias a otros autores, lo que respecta a explicar cuál es la modalidad de desarrollo más apropiada, si bien una aplicación con tecnología nativa o si debe ser basada en web.

La información referente a este asunto, generalmente, es producto de apreciaciones personales, experiencias particulares o artículos que no están argumentados técnicamente. Es así como estas tecnologías de desarrollo, nativo y basado en web, son largamente difundidas en cuanto a cómo emplearlas, pero no se cuenta con información técnica que permita determinar cuál deber ser la utilizada en unas condiciones específicas.

1.2 FORMULACIÓN DEL PROBLEMA

¿Cómo a través de los atributos de calidad del software determinar entre las tecnologías de desarrollo: nativo y basado en web, en la construcción de una aplicación móvil?

1.2 OBJETIVOS DE LA INVESTIGACIÓN

1.2.1 Objetivo General

Confrontar dos tecnologías de desarrollo de aplicaciones móviles sobre la plataforma Android 4.0 desde la perspectiva de los atributos de calidad.

1.2.2 Objetivos Específicos

- Definir el proceso de evaluación de los atributos de calidad del software.
- Describir las tecnologías de desarrollo de aplicaciones móviles para Android.

- Construir una aplicación móvil tanto en tecnología nativa como una basada en web que sirva como caso de estudio para el proceso de evaluación de los atributos de calidad del software.
- Efectuar el proceso de evaluación de los atributos de calidad del software tanto a la aplicación nativa como a la aplicación basada en web.

1.3 JUSTIFICACIÓN

La telefonía móvil ha revolucionado el mundo de las telecomunicaciones, dando la oportunidad de contar con su portabilidad, movilidad y usabilidad que tanto se requieren hoy en día. Los teléfonos celulares han ido incursionando de forma creciente en las actividades diarias, pasando de ser una herramienta de comunicación de voz únicamente, hasta brindar un sin número de posibilidades como si se tratara de un computador personal, originando el teléfono inteligente (smartphone, en inglés).

Ahora, la importancia de confrontar dos tecnologías de desarrollo de aplicaciones móviles, la nativa y la basada en web, radica en proveer técnicamente una confrontación argumentada y confiable que oriente al determinar a cuál tipo de aplicación debe recurrir un equipo de desarrollo. Mediante los atributos de calidad, que tienen amplia consideración en modelos de calidad del software pertenecientes a estándares internacionales, se proyecta conseguir una confrontación válida dentro del ámbito de ingeniería.

1.4 DELIMITACIONES

1.4.1 Delimitación geográfica

El desarrollo de este proyecto de grado tiene lugar en la Universidad Francisco de Paula Santander Ocaña. Es preciso mencionar que tanto los investigadores como el director del trabajo de grado residen en el municipio de Ocaña, departamento Norte de Santander.

1.4.2 Delimitación temporal

La realización del proyecto se enmarca en un tiempo aproximado de 44 semanas.

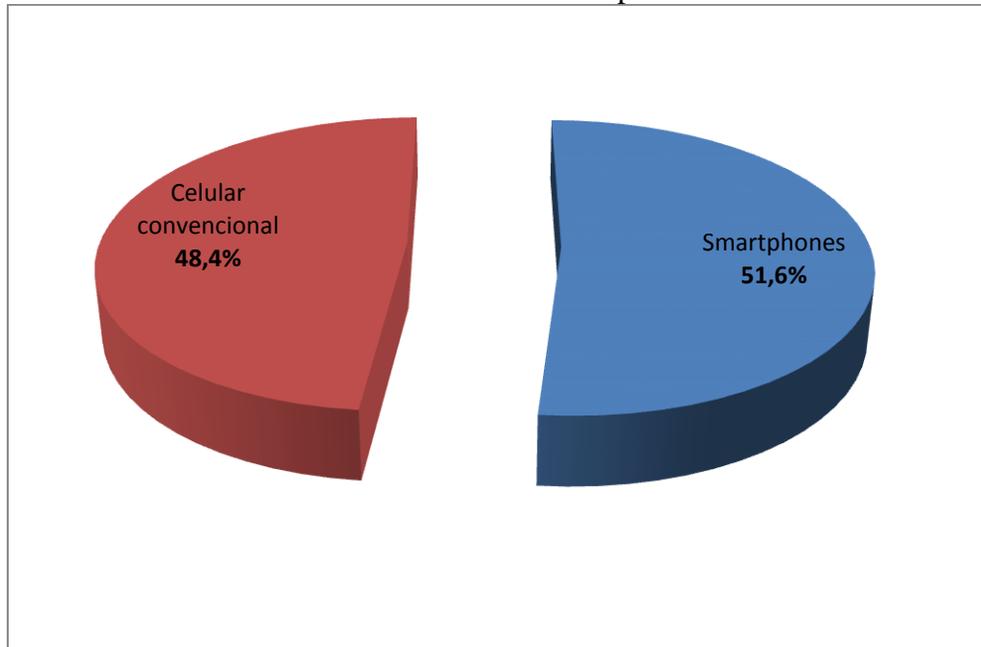
1.4.3 Delimitación conceptual

Aunque este trabajo pertenece al ámbito de desarrollo de software, debido a la magnitud de este campo es necesario precisar que se ocupa específicamente del desarrollo de aplicaciones móviles, que son denominadas de esa forma teniendo en cuenta que son

construidas para dispositivos móviles. Ahora, en cuanto a éstos últimos, con el nombre de dispositivos móviles es posible agrupar un conjunto de aparatos tales como: teléfonos móviles, organizadores y asistentes personales digitales, sistemas de navegación de automóviles, tablets, smartphones, entre otros; sin embargo, para efectos de este trabajo de grado se delimita a los smartphones o teléfonos inteligentes que tienen un rápido crecimiento en la población mundial.

Según la estadística se venden más smartphones que celulares convencionales (ver figura 1): “En total, entre enero y marzo se vendieron 418,6 millones de teléfonos móviles en todo el mundo, de los cuales el 51,6% fueron terminales inteligentes: 216,2 millones de unidades”¹, además, Colombia se situó como el país de mayor crecimiento de smartphones en el mundo durante 2012, según Flurry², como se ve en la figura 2.

Figura 1. Venta de celulares convencionales VS. Smartphones.

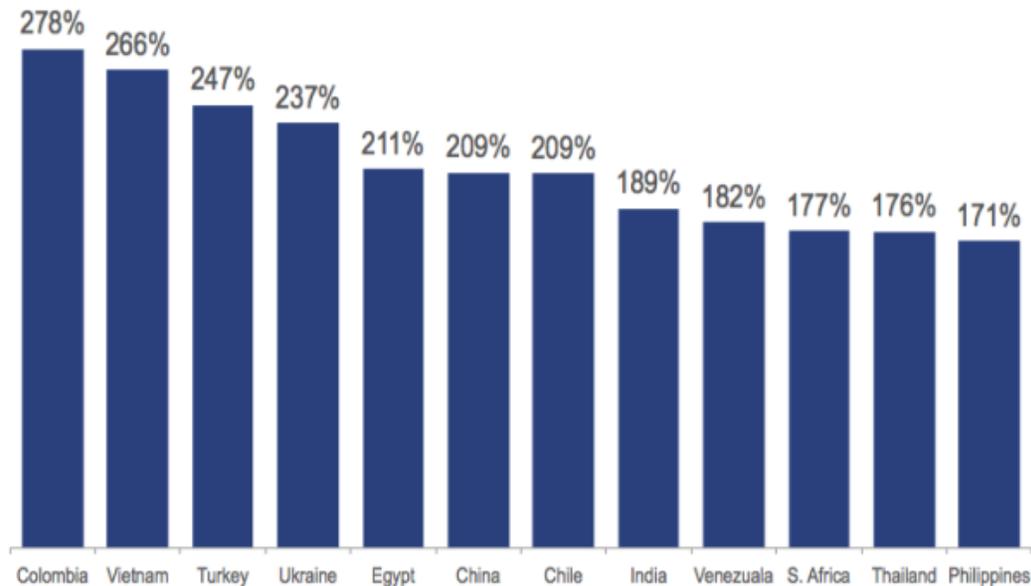


Fuente: Autores del proyecto.

¹ Ya se venden más smartphones que móviles en el mundo. [En línea]. Marzo de 2013. [Citado el 4 de mayo de 2013]. Disponible en internet: http://www.pactual.com/articulo/actualidad/noticias/12921/venden_mas_smartphones_que_moviles_mundo.html

² FLURRY. China Knocks Off U.S. to Become World's Top Smart Device Market. [En línea]. [Citado el 4 de mayo de 2013]. Disponible en internet: <http://blog.flurry.com/bid/94352/China-Knocks-Off-U-S-to-Become-Top-Smartphone-Tablet-Market>

Figura 2. Crecimiento de smartphones activos por país (%).



Fuente: Flurry Analytics. [Citado el 4 de mayo de 2013]. Disponible en internet: <http://blog.flurry.com/bid/94352/China-Knocks-Off-U-S-to-Become-Top-Smartphone-Tablet-Market>

Por otro lado, de los teléfonos inteligentes existe una serie de plataformas móviles y para este caso se utilizó Android por varias razones: primero, es el sistema operativo móvil más utilizado en el mundo³ con más de 750 millones⁴ de dispositivos activados.

Segundo, es una plataforma móvil de código abierto Linux⁵, y este tipo de software generalmente es más acogido por el ámbito académico, dado el caso de la facilidad para acceder a todas las herramientas de desarrollo de manera oficial por parte de Google, entre otras. Asimismo, la versión de Android que se empleó en este proyecto es la 4.0 conocida como Ice Cream Sandwich, teniendo en cuenta que los dispositivos con los que se contó en este trabajo de grado son Sony Xperia S con dicha versión de Android instalada y los autores del mismo no disponen de otros terminales para tal fin. Aparte de esto, Ice Cream Sandwich es una de las versiones más recientes de Android y según Google⁶ en este

³ COMSCORE. ComScore Reports March 2013 U.S. Smartphone Subscriber Market Share. [En línea]. Mayo de 2013. [Citado el 4 de mayo de 2013]. Disponible en internet: http://www.comscore.com/Insights/Press_Releases/2013/5/comScore_Reports_March_2013_U.S._Smartphone_Subscriber_Market_Share

⁴ Update from the CEO. [En línea]. Marzo de 2013. [Citado el 6 de mayo de 2013]. Disponible en internet: <http://googleblog.blogspot.com.es/2013/03/update-from-ceo.html>

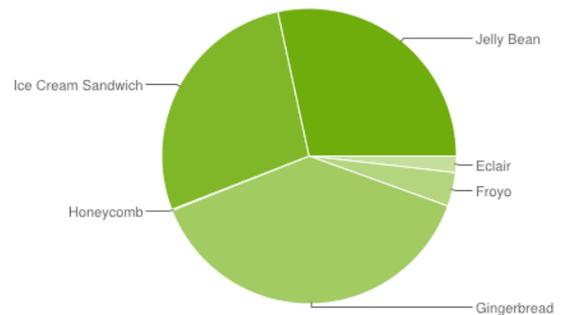
⁵ GOOGLE DEVELOPER ANDROID. Android, the world's most popular mobile platform. [En línea]. [Citado el 6 de mayo de 2013]. Disponible en internet: <http://developer.android.com/about/index.html>

⁶ GOOGLE DEVELOPER ANDROID. Platform versions. [En línea]. [Citado el 8 de mayo de 2013]. Disponible en internet: <http://developer.android.com/about/dashboards/index.html>

momento está presente en una importante cifra 27.5% de los dispositivos con esta plataforma como lo indica la figura 3.

Figura 3. Versiones de la plataforma Android para mayo de 2013

Version	Codename	API	Distribution
1.6	Donut	4	0.1%
2.1	Eclair	7	1.7%
2.2	Froyo	8	3.7%
2.3 - 2.3.2	Gingerbread	9	0.1%
2.3.3 - 2.3.7		10	38.4%
3.2	Honeycomb	13	0.1%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	27.5%
4.1.x	Jelly Bean	16	26.1%
4.2.x		17	2.3%



Fuente: Google Developer Android. [Citado el 4 de mayo de 2013]. Disponible en internet: <http://developer.android.com/about/dashboards/index.html>

De otra parte, las aplicaciones para dispositivos móviles, o simplemente aplicaciones móviles, son pequeños programas de software de las que existen básicamente tres tipos: 1. *Aplicación nativa*, o que es desarrollada en el lenguaje del sistema operativo a la que pertenece; 2. *Aplicación web*, a la que se accede desde un navegador de internet, o que tiene un aspecto muy similar a la nativa en muchos casos; y 3. *Aplicación híbrida*, que lleva este nombre puesto que es en esencia una aplicación web empaquetada como aplicación nativa.

Por último, tal como lo indica el título de este documento, este trabajo consiste en una confrontación que se llevó a cabo desde la perspectiva de los atributos de calidad del software, para lo cual es importante decir que dichos atributos pertenecen al área de calidad de software.

1.4.4 Delimitación operativa

Realizar un proyecto dentro del ámbito de la tecnología, y en este caso, los dispositivos móviles y el desarrollo de aplicaciones para los mismos, exige contar con información actualizada y veraz. Para el problema de estudio de este trabajo, hacer una búsqueda en internet con palabras clave como: “aplicación móvil”, “aplicación nativa y web”, la información disponible en la red puede llegar a ser abundante, pero en sitios informales u ofrecida por comunidades que no tienen reconocimiento académico. Sumado a esto, la biblioteca de la UFPSO no cuenta con libros dedicados a esta área.

Por otro lado, emprender un proyecto informático, implica para este caso, tener la disponibilidad de una serie de recursos tecnológicos como: computador, acceso a internet y smartphones para realizar pruebas, conocer características de Android, de las aplicaciones, de las funcionalidades del terminal, entre otras, en un dispositivo real y de especificaciones aceptables. Por lo anterior, los autores del proyecto se han visto en la necesidad de adquirir dos teléfonos inteligentes para dicha finalidad y por supuesto, la inversión de dinero que esto acarrea.

Asimismo, de acuerdo a la UFPSO⁷ únicamente se tienen dos experiencias asociadas a las aplicaciones móviles, puesto que AgroTIC fue desarrollada por el docente universitario Badwin Arévalo, dentro del programa Norte de Santander Vive Digital y de otra parte, el proyecto de realidad aumentada, denominado SAIIBRA, del estudiante de ingeniería de sistemas Carlos Lobo. Sin embargo, no hay proyectos anteriores de desarrollo de aplicaciones móviles en lo que respecta a trabajos de grado, por lo cual, son muy reducidas las posibilidades de obtener asesoría en asuntos particulares de los dispositivos móviles, plataformas móviles, smartphones y temas relacionados.

⁷ Norte de Santander Vive Digital y aplicaciones para dispositivos móviles. [Vídeo en línea]. [Citado el 6 de mayo de 2013]. Disponible en: [internet:http://www.ufpso.edu.co/medios/reproductor_video_conexiones.php?rutavideo=../videos/conexiones/2012_08\(30\)aplicativo.f4v](http://www.ufpso.edu.co/medios/reproductor_video_conexiones.php?rutavideo=../videos/conexiones/2012_08(30)aplicativo.f4v)

2. MARCO REFERENCIAL

2.1 MARCO HISTÓRICO

La telefonía móvil brinda esa posibilidad de comunicación independientemente de la posición geográfica del usuario, y esto ha producido una gran propagación de los dispositivos móviles.

En un comienzo estos aparatos sólo permitían la función de comunicación, pero poco a poco, y a pesar de las limitaciones en recursos hardware y software con que contaban, fueron incorporando pequeñas aplicaciones como la libreta de direcciones, el calendario, la calculadora y juegos. El ejemplo más memorable fue cuando Nokia implementó el clásico juego Snake en 1997, que se convirtió en un éxito masivo en todo el mundo. “Para 1997, bajo el nombre de Snake, esta idea llegaba a los bolsillos de millones de personas a través de sus teléfonos Nokia donde se fue gestando la locura por los juegos móviles entre los adolescentes de todo el mundo.”⁸

La telefonía móvil continúa avanzando, hoy en día existen teléfonos con una gran cantidad de herramientas y funcionalidades tales como soporte de correo electrónico, GPS, acceso a internet, pantalla táctil, teclado Qwerty, lectura de archivos en los tradicionales formatos ofimáticos, poseen un sistema operativo y muchas cosas más, generando lo que hoy se conoce como Smartphone o teléfono inteligente.

De acuerdo a lo anterior, es posible identificar claramente las dos partes por las que está compuesto un teléfono móvil en general, la primera conocida como el hardware y la segunda es el software.

Para el caso particular de este documento, es importante mencionar que en lo que respecta al hardware o parte física de los smartphones, esto se remonta a la fabricación de los teléfonos celulares básicos, y que mediante el avance tecnológico se fue incorporando mayor capacidad de procesamiento, memoria RAM y memoria interna o de almacenamiento. De este modo, se instalan cámaras tanto de fotografía como de vídeo de características sobresalientes, pantallas táctiles y de tamaño mayor a las 3 pulgadas; asimismo, se incluyen sensores como acelerómetro, giroscopio y de proximidad, como también disponibilidad de GPS, conexión WIFI y 4G, sin contar otras características. La parte de hardware de los smartphones sorprende y lo seguirá haciendo; mientras el tiempo transcurre los fabricantes de estos dispositivos continúan anunciando terminales con mayor innovación.

⁸ La evolución de Snake. [En línea]. [Citado el 8 de diciembre de 2012]. Disponible en internet: <http://conversaciones.nokia.com/2011/12/16/la-evolucion-de-snake/>

En cuanto al software para los dispositivos móviles, es posible mencionar dos grandes partes: El sistema operativo y las aplicaciones. El sistema operativo en un smartphone tiene la misma importancia que en un ordenador tradicional, por tal razón debe ser considerado con el mismo interés. Los sistemas operativos de mayor reconocimiento para las computadoras convencionales son: Windows, Mac OS, y una serie de distribuciones Linux. Ahora, hacer referencia a los sistemas operativos para smartphones también implica mencionar los S.O. para dispositivos móviles en general, que por supuesto, tienen su origen en los S.O. para computadores de sobremesa. Avanzando en la historia, como lo expresa Parsons⁹, se presentaron tres sistemas operativos con un fuerte dominio en el ámbito de dispositivos de bolsillo o asistentes personales (también conocidos como Palm): Palm OS, Windows Mobile y Symbian OS.

En la actualidad, para el caso de los smartphones, entre los sistemas operativos más destacados, se encuentran:

- **BlackBerry OS**, que pertenece a la empresa conocida como Research In Motion, que hoy en día es BlackBerry. Este sistema operativo solo se encuentra en los dispositivos que fabrica la misma marca.
- **Android**, la plataforma del gigante de internet, Google, que ha revolucionado el mercado, siendo un sistema libre y presente en terminales de grandes fabricantes como Samsung, Sony, HTC, LG y Motorola.
- **iOS**, perteneciente a la famosa empresa Apple. iOS cuenta con gran reconocimiento y está presente en los dispositivos fabricados por Apple.
- **Windows Phone**, sucesor de Windows Mobile. Los sistemas operativos bajo el nombre Windows están ampliamente extendidos en el mundo entero a causa de los tan conocidos Windows XP, Vista y Seven. Este sistema de Microsoft se encuentra en algunos smartphones de Samsung, HTC y los famosos terminales Lumia de Nokia.

Con respecto a las aplicaciones, cuando una persona adquiere su smartphone lo recibe con el sistema operativo y con una serie de aplicaciones de uso acostumbrado, además de algunas configuraciones que son establecidas entre el fabricante y probablemente el operador de telefonía móvil.

Para referirse a las tan conocidas aplicaciones para smartphones es necesario remontarse a los celulares más básicos, como ya se mencionó antes, que contaban con agenda de contactos, reloj, alarma, recordatorios, algunos incluían calculadora y la aplicación para enviar y recibir mensajes de texto. Con el transcurso del tiempo se comenzó a incluir algunos pequeños juegos muy elementales en ciertos modelos de teléfonos. El desarrollo de

⁹ PARSONS, June y OJA, Dan. Conceptos de Computación: Nuevas Perspectivas. Décima edición. México D.F. : Cengage, 2008, p. 196.

pequeños programas para los celulares avanzó rápidamente y aparecieron calendarios, reproductores de música, vídeo, aplicación para manejo de la cámara, del infrarrojo, del bluetooth, programa para radio, entre otras.

Todas esas pequeñas aplicaciones mencionadas, dieron lugar a las aplicaciones móviles de la actualidad. Hoy en día se habla de miles de aplicaciones en las diferentes plataformas, y que se ofrecen a los usuarios de manera oficial mediante la tienda en línea que corresponda. Una forma de revisar el crecimiento en el desarrollo de aplicaciones es a través de las tiendas en línea, en donde se puede encontrar una variedad importante de éstas para categorías como: entretenimiento, educación, salud, música, y otros campos más. A propósito de las tiendas en línea, Arroyo dice:

Existen cuatro tipos de tiendas de aplicaciones móviles:

De fabricantes de dispositivos, como BlackBerry o Apple, aunque en ambos casos también disponen de sistemas operativos propios, de sistemas operativos, como Android o Microsoft, de operadores de telefonía móvil, como la OVI Store de Nokia e Independientes, como GetJar o la más reciente Amazon Appstore.¹⁰

En general, las tiendas en línea más conocidas son: para el caso de Android: la Google Play Store, para iOS: la Appstore, asimismo, Microsoft cuenta con Windows Phone Store y BlackBerry tiene su BlackBerry World.

El número de aplicaciones en tiendas de sistemas operativos principales se describe a continuación (ver figura 4):

- La Google Play Store cuenta con más de 700.000 aplicaciones.¹¹
- La iOS AppStore tiene también unas 700.000 aplicaciones, lo que indica que la tienda de Google y la de Apple compiten con una cantidad similar¹².
- La tienda de Windows Phone ofrece unas 150.000 aplicaciones.¹³
- Y la BlackBerry World 100.000 aplicaciones¹⁴.

¹⁰ ARROYO, Natalia. Información en el móvil. Barcelona: UOC, 2011. p. 96.

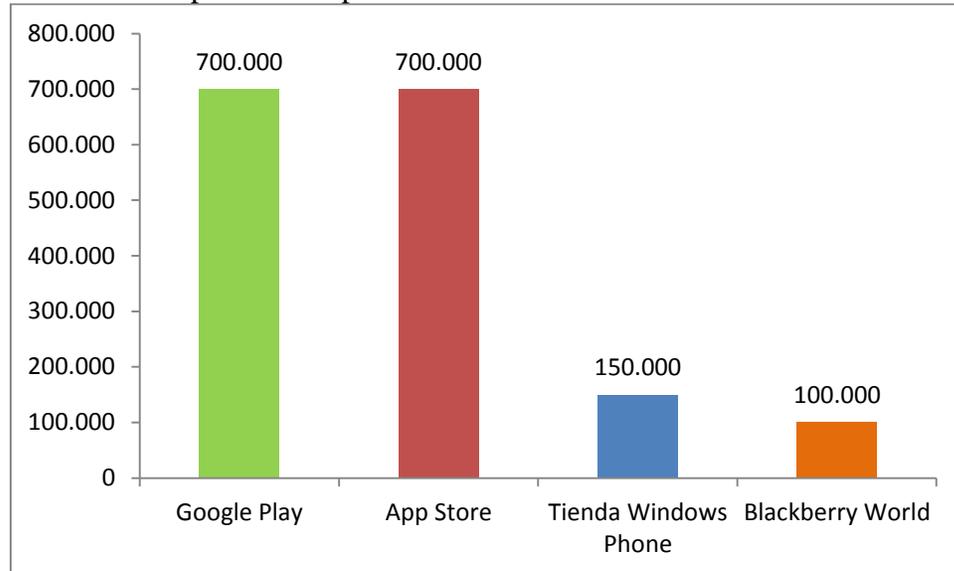
¹¹ WOMACK, Brian. Google Says 700,000 Applications Available for Android. [En línea]. [Citado 12 de diciembre de 2012]. Disponible en internet: <http://www.businessweek.com/news/2012-10-29/google-says-700-000-applications-available-for-android-devices>

¹² VALERO, Claudio. Android alcanza en número de aplicaciones a iOS. [En línea]. [Citado 12 de diciembre de 2012]. Disponible en internet: <http://www.adslzone.net/article9843-android-alcanza-en-numero-de-aplicaciones-a-ios.html>.

¹³ SVETLIK, Joe. Windows Phone Store hits 150,000 apps, doubled in last year. [En línea]. [Citado el 3 de febrero de 2012] Disponible en internet: <http://crave.cnet.co.uk/software/windows-phone-store-hits-150000-apps-doubled-in-last-year-50010072/>

¹⁴ CIBERSUR.COM. 100.000 aplicaciones, el número redondo de BlackBerry para volver a la senda. [En línea]. [Citado el 3 de junio de 2013]. Disponible en internet: <http://www.cibersur.com/portada/013328/100000/aplicaciones/numero/redondo/blackberry/volver/send>

Figura 4. Número de aplicaciones por tienda oficial



Fuente: Autores del proyecto.

Entre las alternativas a las ya mencionadas tiendas en línea, se tiene:

- **Amazon Appstore**, la propuesta de Amazon para Android, en la que sobresale la posibilidad de descargar cada día una aplicación o un juego de pago de forma gratuita.
- **Panda App**, si bien la aplicación para Android no se encuentra en el Google Play, es posible acceder a ella mediante su portal web oficial.
- **Samsung Apps** es la tienda de aplicaciones que la empresa surcoreana dispone para todos los usuarios de smartphones de la marca.
- **OVI Store** una tienda de aplicaciones de Nokia.

Al hablar de las aplicaciones es necesario mencionar a quienes se proponen crearlas: los desarrolladores; éstos pertenecen a diferentes países, gracias al alcance de internet y de las nuevas tecnologías es posible encontrar aplicaciones provenientes de una gran variedad de orígenes alrededor del mundo, ya sean empresas dedicadas al desarrollo de éstas o sencillamente aficionados.

Los desarrolladores están motivados por historias de éxito como la de Angry Birds. Este juego para dispositivos móviles, lanzado en diciembre de 2009 por la compañía finlandesa Rovio, fue un hit absoluto pese a su trama simple y sus gráficos sencillos: sólo en la tienda de Apple se han vendido 12 millones de unidades, y la acogida ha sido tal que en sus diferentes versiones gratuitas ha tenido 1.000 millones de descargas, la mitad de ellas entre noviembre de 2011 y mayo de 2012 (las ediciones gratuitas le generan ingresos a Rovio por la venta de publicidad). Al igual que Angry Birds existen muchas aplicaciones que han alcanzado puestos importantes en las diferentes tiendas, por nombrar algunas: Google Maps, WhatsApp, Skype, Twitter y Facebook.

Colombia no se queda atrás e incursiona en este sector. Por esta razón, es común ver aplicaciones desarrolladas por colombianos para plataformas como iOS, Android, Windows Phone y BlackBerry. Entre ellas se pueden destacar juegos como Antártica, de Colombia Games; Soccer Stars Galaxy Team, del CDI Interactive; y Tejo, de AppSolution.

Para dar un ejemplo, Foonkie Monkey es una joven empresa colombiana creada por Camilo Páez y Juan Carlos Rosales que se dedica al desarrollo de aplicaciones para dispositivos móviles. El equipo de esta empresa consta de siete personas, en algunas oportunidades unos más o quizás menos, de acuerdo a la carga de trabajo. “Taxi Seguro” es una de las aplicaciones más descargadas en Colombia y se encuentra disponible para iPhone y BlackBerry. En Taxi Seguro el usuario ingresa los datos placa y nombre del conductor del taxi y esta información es enviada por GPS a los contactos que el usuario indique. Además de la aplicación ya mencionada, la empresa colombiana ha fabricado: Kid Rescue App (disponible sólo para iPhone), para ayudar a identificar niños que trabajan y lograr vincularlos al estudio; también “Knuckleheads” un juego disponible en la AppStore, entre otras. Cabe destacar que un número importante de las aplicaciones móviles de Foonkie Monkey están disponibles sólo para iPhone/iPad.

Tal como sucede con Foonkie Monkey, en Colombia hay una serie de empresas que desarrollan aplicaciones móviles, como: MDEPlus, KUBO MOBILE, Ingenios IT Business Solutions S.A.S, Kogi Mobile, WM Wireless & Mobile y algunas más.

Asimismo, es importante mencionar el trabajo que pertenece a las universidades, como el del Grupo W@P Colombia¹⁵, un semillero de investigación de la Universidad del Cauca, en donde se están desarrollando aplicaciones para celulares y tablet. Francisco Martínez, coordinador del semillero, explica que el grupo tiene 11 años de funcionamiento y la misión es formar nuevos talentos en el área de desarrollo de aplicaciones para dispositivos móviles. Según indica Martínez, actualmente cuentan con 200 personas que se han ido capacitando y tres nuevas empresas que se han formado gracias al semillero.

W@P Colombia ha desarrollado aplicaciones para celulares tradicionales y también para smartphones y tablets, entre las cuales hay aplicaciones de entretenimiento, herramientas para los ciudadanos como 'Pico y Placa' para Colombia, emisoras de radio y una tan autóctona que es útil para aprender a bailar salsa; Francisco Martínez destaca la creación de una aplicación para el mundial Sub 20 que se llevó a cabo en el país. “Luego los chicos de esta aplicación son los que forjaron uno de estos emprendimientos, y ya como empresa a partir de ese núcleo construyeron la aplicación para la Eurocopa, por ejemplo.” Cabe mencionar que los estudiantes y docentes que hacen parte del semillero W@P Colombia han recibido diferentes premios por las aplicaciones móviles que han desarrollado. La expectativa para los desarrolladores colombianos es bastante grande teniendo en cuenta que

¹⁵ PRADA CÉSPEDES, Edilma. Cauca pionero en el desarrollo de aplicaciones móviles en Latinoamérica. [En línea]. [Citado el 25 de Enero de 2012]. Disponible en internet: <http://www.agendapropia.com/index.php/dialogos/1676-cauca-pionero-en-el-desarrollo-de-aplicaciones-moviles-en-latinoamerica>

esta es una industria dinámica y en la que hay muchas oportunidades. Es así como Iván Gil, gerente de Ecosistemas de Nokia considera que Colombia es el primer país de Latinoamérica respecto a la calidad de las aplicaciones que desarrolla, por su oferta diferenciadora, ya que no son tan típicas y son muy útiles para el usuario.¹⁶

Sin embargo, el gerente de Colombia Games, Juan Nates, cree que el país aún tiene camino por recorrer en esta materia, pues la penetración de equipos móviles es inferior al 20%, y solo crecerá significativamente en dos años.

Debido a este crecimiento se decidió premiar a estas ideas innovadoras y el año inmediatamente anterior en un evento denominado The App Date Bogotá se escogió la mejor aplicación colombiana de 2012. La ganadora de la noche fue la aplicación La Rebaja Móvil, que permite hacer compras en ese establecimiento desde un dispositivo con Android. En el concurso participaron cerca de 200 aplicaciones.¹⁷

En lo que respecta a la ciudad de Ocaña, se tiene una aplicación móvil denominada AgroTIC, la cual permite consultar los precios actuales de los productos que se comercializan en los distintos municipios cobijados por el proyecto Norte de Santander Vive Digital, quien es el ente encargado del desarrollo de la aplicación a través de la Universidad Francisco de Paula Santander Ocaña, ejecutora del proyecto. El desarrollo de AgroTIC fue concebido utilizando los lenguajes HTML5, JavaScript, CSS y PHP, y luego compilado como aplicación nativa para Android.

El ingeniero de sistemas se hace presente en esta nueva era y hace uso de muchas herramientas que se ofrecen, pues así como existen tecnologías y lenguajes de programación para el desarrollo de software que corre en los ordenadores tradicionales, también existe una amplia gama de tecnologías, API's y frameworks para la programación en los teléfonos celulares.

De esta manera el desarrollador puede construir básicamente dos tipos de aplicaciones: nativas y web; aunque existe, por llamarlo de alguna manera, un tercer tipo de aplicación: la híbrida, ésta combina las dos anteriores ya que se construye usando tecnología web y luego es compilada para ejecutarse como una aplicación nativa. De aplicación híbrida se tiene el caso de la antes mencionada AgroTIC, desarrollada en la ciudad de Ocaña.

Históricamente las aplicaciones nativas surgieron antes que las web, puesto que las nativas se remontan a los programas preinstalados con los que venían los celulares comunes, además, mucho antes de que los teléfonos móviles tuvieran conectividad web¹⁸.

Hasta este punto se ha mostrado un panorama de las aplicaciones móviles, sin embargo, en adelante se expone lo relacionado con aspectos de calidad para las mismas.

¹⁶ EL PAÍS. Aplicaciones para móviles, un mercado que se abre camino en Colombia. [En línea]. [Citado el 12 de Enero de 2012]. Disponible en internet: <http://www.elpais.com.co/elpais/colombia/noticias/aplicaciones-para-moviles-mercado-abre-camino-colombia>.

¹⁷ COMUNICACIONES TAD_BOG. The App Date Awards. [En línea]. [Citado el 10 de Enero 2013]. Disponible en Internet: <http://bogota.theappdate.com/>

¹⁸ Aplicaciones para dispositivos móviles. [En línea] [Citado el 15 de Enero de 2013]. Disponible en internet: http://www.scholarium.co/index.php?option=com_content&view=article&id=65&Itemid=97

De acuerdo a un estudio realizado por Apigee¹⁹, el 96% de los norteamericanos encuestados, usuarios de aplicaciones móviles, dicen que hay frustraciones que los llevarían a lanzar una mala crítica a éstas, entre las que se encuentran: Aplicaciones congeladas, capacidad de respuesta lenta, demasiado uso de la batería, cantidad exagerada de publicidad; tales problemáticas muestran la relación entre las aplicaciones móviles y la necesidad de apuntar a los atributos de calidad en el desarrollo de éstas.

En lo que respecta al sector dedicado a la creación de aplicaciones móviles, para distintas empresas la necesidad de crear aplicaciones que cumplan una serie de exigencias concretas, es la clave para hacer publicidad a sus productos y hacerse notar entre la competencia. Específicamente, algunos hablan de realización de tests de calidad para las aplicaciones móviles. Dentro de este grupo de empresas se encuentran, por ejemplo: Accensit²⁰, que ofrece desarrollar aplicaciones móviles y entregarlas al cliente luego de haber implementado una serie de pruebas. También existen empresas que se ocupan del trabajo de control de calidad de las aplicaciones, de tal modo que el desarrollador de una aplicación contrata las pruebas de calidad de su producto. Es el caso de Albira Solutions, que afirma en su sitio web: “ALBIRA Solutions ha creado una solución completa de test de aplicaciones web y portales para móviles que le permite certificar que su aplicación o portal móvil funcionan correctamente en toda una gama de dispositivos móviles.”²¹

Sin embargo, en el caso anterior hay que notar que se indica una solución de test solamente para aplicaciones del tipo web. Tal como lo indica SlashMobility²²: “Actualmente existe cantidad de información dedicada a la fase de Testing, pero la mayor parte es demasiado generalista, basada en metodologías desactualizadas o enfocadas únicamente al paradigma web.” Con lo cual también es necesario observar que el proceso para asegurar la calidad de las aplicaciones móviles es de altísima importancia, pero que debe ampliarse.

Es por esto que Bull y Borland-Micro Focus definen un nuevo estándar de calidad de las aplicaciones móviles. De acuerdo a lo expuesto en el diario El Economista²³ en su edición virtual: “El testing de aplicaciones - explica Ricardo Montes, director de marketing y comunicación de Micro Focus - exige disponer de una cobertura de pruebas ad hoc.

¹⁹ APIGEE. Apigee Survey: Users Reveal Top Frustrations That Lead to Bad Mobile App Reviews. [En línea]. [Citado el 12 de enero de 2013]. Disponible en internet: <http://apigee.com/about/pressrelease/apigee-survey-users-reveal-top-frustrations-lead-bad-mobile-app-reviews>

²⁰ Pruebas de Aplicaciones Móviles. [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet : <http://www.accensit.com/index.php/desarrollo-de-aplicaciones-para-moviles/pruebas-de-aplicaciones-moviles.html>

²¹ Validación de aplicaciones móviles. [En línea]. [Citado el 20 de enero de 2012]. Disponible en Internet: <http://www.albirasolutions.com/es/servicios/tecnologias-de-la-informacion/control-de-calidad/validacion-de-aplicaciones-moviles>

²² Slash Session: Testing en aplicaciones móviles. [En línea]. [Citado el 21 de enero de 2013]. Disponible en internet: <http://slashmobility.com/blog/2011/09/410/>

²³ Bull y Borland-Micro Focus definen el nuevo estándar de calidad de las aplicaciones móviles [En línea]. [Citado el 25 de enero de 2013]. Disponible en internet : <http://www.economista.es/empresas-finanzas/noticias/4365236/10/12/Bull-y-BorlandMicro-Focus-definen-el-nuevo-estandar-de-calidad-de-las-aplicaciones-moviles.html>

Borland Silk Mobile permita disminuir el tiempo y costes en las tareas de pruebas, mejorar la calidad de las aplicaciones. La experiencia final del usuario y visualizar el impacto real del incremento de la complejidad”.

Bull y Borland-Micro Focus proponen una estrategia para lograr la máxima calidad de las aplicaciones móviles, aspecto que no es sencillamente una opción sino que ha pasado a ser una exigencia.

De otra parte, existen algunos documentos más elaborados, y que además no se ocupan de las aplicaciones móviles específicamente, como los realizados por El Laboratorio Aragonés de Usabilidad. Este instituto es una iniciativa de la Dirección General de Tecnologías para la Sociedad de la Información del Gobierno de Aragón, donde se abordan proyectos de I+D en el ámbito de la evaluación de la usabilidad y accesibilidad de interfaces, cuyos resultados aplica en el estudio de la respuesta del consumidor a diferentes estímulos, a la mejora de la usabilidad en los diseños y a la evaluación de sitios web.

Entre los proyectos de I+D desarrollados en el laboratorio se destaca WebA, una herramienta que se está utilizando en la actualidad para evaluar de modo semiautomatizado la accesibilidad y usabilidad de sitios web, tanto de instituciones públicas como de empresas privadas que lo solicitan al Laboratorio. La herramienta WebA dispone en la actualidad de módulos que permiten la preparación, cumplimentación y análisis de tests de satisfacción de usuarios, de heurísticas de usabilidad basadas en normas ISO y de accesibilidad basados en las WAI 1.0 del W3C y de análisis de la arquitectura de la información mediante técnicas de CardSorting con diferentes técnicas estadísticas.

Siguiendo con su línea de investigación, el instituto ha querido incursionar en el mundo de los dispositivos móviles teniendo en cuenta aspectos propios de estos aparatos como: pantalla más reducida, menos memoria y menos velocidad de proceso que los ordenadores convencionales, por lo que generan mayor complejidad en temas de usabilidad y accesibilidad. Por tanto, el instituto trabaja en WebA Mobile²⁴ (Herramienta de ayuda para el diseño y evaluación de sitios web para dispositivos móviles), un nuevo módulo de la aplicación WebA. Este módulo facilitará la evaluación de la usabilidad y de la accesibilidad mediante la cumplimentación de las pautas que recomiendan las Buenas Prácticas en la "W3C Mobile Web Initiative 1.0", además de permitir la cumplimentación y análisis de los correspondientes tests de Satisfacción de usuario. Para la evaluación, WebA Mobile contiene los procedimientos necesarios para realizar tanto la evaluación de la usabilidad como de la accesibilidad de las interfaces de usuario. Para ello combina distintos métodos de evaluación: automática, heurística, basada en normas y/o pautas.

Se espera que WebA Mobile sea en relación a la evaluación de la accesibilidad y usabilidad de sitios web para dispositivos móviles, lo que WebA es a la evaluación de la

²⁴ MENA TOBAR, Luis; LATORRE, Pedro y LAFUENTE LAPENA, Elena. WebA Mobile (Web Analysis Mobile): Herramienta de ayuda para el diseño y evaluación de websites para dispositivos móviles [En línea]. [Citado el 10 de diciembre de 2012]. Disponible en internet: http://www.aipo.es/info_art.php?id=33

accesibilidad y usabilidad de sitios web visualizados desde un computador. Cabe destacar que WebA Mobile facilitará la evaluación de la usabilidad mediante la comprobación de las normas ISO/IEC 18021, ISO 18789, garantizando el ajuste a estos estándares industriales, y a las pautas que recomienda la W3C en sus Buenas Prácticas en Web Móvil 1.0.

Ahora, los trabajos o alcances que se han relacionado hasta el momento pertenecen a un grupo en el que se engloban todos los asociados con lograr aspectos de calidad en los sistemas de software, pero Granollers y Lorés²⁵ proponen lo que ellos denominan un nuevo concepto: Esfuerzo de usabilidad. En su estudio, sobresale que únicamente se ocupan de un solo atributo de la calidad del software: la usabilidad.

2.2 MARCO CONCEPTUAL

El presente trabajo de grado aborda el desarrollo de aplicaciones móviles, y éstas son un tipo de software; por esta razón es necesario iniciar haciendo referencia a la definición de **software**. Campderrich Falgueras dice:

Un sistema de software, denominado también aplicación o simplemente software, es un conjunto integrado de programas que en su forma definitiva se pueden ejecutar, pero comprende también las definiciones de estructuras de datos (por ejemplo, definiciones de bases de datos) que utilizan estos programas y también la documentación referente a todo ello (tanto la documentación de ayuda en el uso del software para sus usuarios como la documentación generada durante su construcción, parte de la cual también servirá para su mantenimiento posterior).²⁶

Otra definición en las palabras de Niño Camazón, expresa: “Es la parte que no se puede tocar del ordenador (intangible), el software es un elemento lógico y se define como un conjunto de órdenes e instrucciones que al ejecutarse sirven para realizar alguna tarea (los programas sin ejecutar son simples archivos en disco).”²⁷

Ahora, un **software es un producto** y por tanto, como dice Sommerville²⁸ resulta de un proceso compuesto por una serie de actividades realizadas por ingenieros del software, que van ocasionando resultados y finalmente producir un producto de software.

²⁵ GRANOLLERS, Toni; LORÉS, Jesús. Esfuerzo de Usabilidad: un nuevo concepto para medir la usabilidad de un sistema interactivo basada en el Diseño Centrado en el Usuario [En línea]. [Citado el 17 de marzo de 2013]. Disponible en internet: <http://griho2.udl.es/publicacions/2004/EU%20i2004.pdf>

²⁶ CAMPDERRICH FALGUERAS, Benet. Ingeniería del Software. Barcelona : UOC, 2003. p. 115.

²⁷ NIÑO CAMAZÓN, Jesús. Sistemas Operativos monopuesto. Madrid : Editex, 2011. p. 16.

²⁸ SOMMERVILLE, Ian. Ingeniería del Software. Séptima edición. Madrid : Addison-Wesley, 2003. p. 7.

Cuando se habla de **desarrollo de software** se piensa principalmente en computadores tradicionales pero debe saberse que también existe construcción de programas para otros aparatos y dispositivos, hasta el punto de aplicaciones para los conocidos **dispositivos móviles**, de los que Arroyo²⁹ presenta la siguiente definición: "...aparato portátil desde el que se puede acceder a la web y diseñado para ser usado en movimiento. Se trata de un subgrupo de los dispositivos inalámbricos, que incluyen también a los dispositivos fijos pero que, a diferencia de ellos, prescinden de conexiones físicas como los ordenadores portátiles con WiFi. Por tanto, dos rasgos definen a los dispositivos móviles: la posibilidad de movilidad de quien lo emplea y la conexión a Internet".

Dentro de ese grupo de los dispositivos móviles, se hace necesario precisar con respecto al **smartphone**, del que LG Electronics dice: "El teléfono inteligente (smartphone en inglés) es un término comercial para denominar a un teléfono móvil que ofrece más funciones que un teléfono celular común. Casi todos los teléfonos inteligentes son móviles que soportan completamente un cliente de correo electrónico con la funcionalidad completa de un organizador personal."³⁰

A los smartphones se les considera un dispositivo que puede llegar a tener las mismas o similares funcionalidades de un computador convencional, tal es el caso de poseer un sistema operativo. De la misma forma como hay una gran variedad de **sistemas operativos** para computadores convencionales, también existe una gama de éstos para smartphones; sin embargo, este trabajo de grado se ocupa de la **plataforma Android**. A continuación se presenta una definición para éste, según la revista Gadgets:

Android es un sistema operativo orientado a dispositivos móviles, basado en una versión modificada del núcleo Linux. Inicialmente fue desarrollado por Android Inc. (una pequeña empresa basada en Palo Alto, California, que fue comprada después por Google), y en la actualidad lo desarrollan los miembros de la Open Handset Alliance (liderada por Google). Esta plataforma permite el desarrollo de aplicaciones por terceros, a través de herramientas proporcionadas por Google, y mediante los lenguajes de programación Java y C. El código fuente de Android está disponible bajo diversas licencias de software libre y código abierto destacando la versión 2 de la licencia de Apache.³¹

De la misma forma, Android se encuentra presente en varias versiones; en este trabajo se empleó la **versión 4.0 de Android** conocida como **Ice Cream Sandwich**, que es la sucesora de Android 3.0 Honeycomb. A propósito del sistema operativo Android, en la

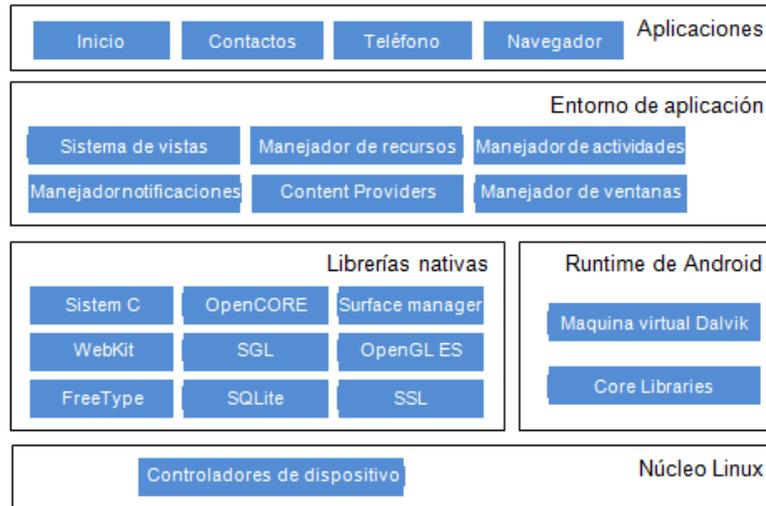
²⁹ ARROYO, Op. Cit., p. 14

³⁰ ¿Qué es un smartphone? [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: <http://www.lgmobile.cl/smartphones>

³¹ REVISTA GADGETS, Tecnología con Estilo. En : Sistema Operativo Android. México : Mina Editores, 2010. p. 60.

figura 5 se muestra la **arquitectura**³² del mismo. Como se puede ver está formada por cuatro capas. Una de las características más importantes es que todas las capas están basadas en software libre.

Figura 5. Arquitectura de Android.



Fuente: Universidad Politécnica de Valencia. [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: <http://www.androidcurso.com/index.php/tutoriales-android/31-unidad-1-vision-general-y-entorno-de-desarrollo/99-arquitectura-de-android>

El núcleo Linux

El núcleo de Android está formado por el sistema operativo Linux versión 2.6. Esta capa proporciona servicios como la seguridad, el manejo de la memoria, el multiproceso, la pila de protocolos y el soporte de drivers para dispositivos.

Esta capa del modelo actúa como capa de abstracción entre el hardware y el resto de la pila. Por lo tanto, es la única que es dependiente del hardware.

Runtime de Android

Está basado en el concepto de máquina virtual utilizado en Java. Dado las limitaciones de los dispositivos donde ha de correr Android (poca memoria y procesador limitado) no fue posible utilizar una máquina virtual Java estándar. Google tomó la decisión de crear una nueva, la máquina virtual Dalvik, que respondiera mejor a estas limitaciones.

Algunas características de la máquina virtual Dalvik que facilitan esta optimización de recursos son: que ejecuta ficheros Dalvik ejecutables (.dex) –formato optimizado para ahorrar memoria. Además, está basada en registros. Cada aplicación corre en su propio

³² UNIVERSIDAD POLITECNICA DE VALENCIA. Arquitectura de Android. [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: <http://www.androidcurso.com/index.php/tutoriales-android/31-unidad-1-vision-general-y-entorno-de-desarrollo/99-arquitectura-de-android>

proceso Linux con su propia instancia de la máquina virtual Dalvik. Delega al kernel de Linux algunas funciones como threading y el manejo de la memoria a bajo nivel.

También se incluye en el Runtime de Android el “core libraries” con la mayoría de las librerías disponibles en el lenguaje Java.

Librerías nativas

Incluye un conjunto de librerías en C/C++ usadas en varios componentes de Android. Están compiladas en código nativo del procesador. Muchas de las librerías utilizan proyectos de código abierto. Algunas de estas librerías son:

System C library: una derivación de la librería BSD de C estándar (libc), adaptada para dispositivos embebidos basados en Linux.

Media Framework: librería basada en PacketVideo's OpenCORE; soporta codecs de reproducción y grabación de multitud de formatos de audio vídeo e imágenes MPEG4, H.264, MP3, AAC, AMR, JPG y PNG.

Surface Manager: maneja el acceso al subsistema de representación gráfica en 2D y 3D.

WebKit: soporta un moderno navegador web utilizado en el navegador Android y en la vista webview. Se trata de la misma librería que utiliza Google Chrome y Safari de Apple.

SGL: motor de gráficos 2D.

Librerías 3D: implementación basada en OpenGL ES 1.0 API. Las librerías utilizan el acelerador hardware 3D si está disponible, o el software altamente optimizado de proyección 3D.

FreeType: fuentes en bitmap y renderizado vectorial.

SQLite: potente y ligero motor de bases de datos relacionales disponible para todas las aplicaciones.

SSL: proporciona servicios de encriptación Secure Socket Layer.

Entorno de aplicación

Proporciona una plataforma de desarrollo libre para aplicaciones con gran riqueza e innovaciones (sensores, localización, servicios, barra de notificaciones).

Esta capa ha sido diseñada para simplificar la reutilización de componentes. Las aplicaciones pueden publicar sus capacidades y otras pueden hacer uso de ellas (sujetas a las restricciones de seguridad). Este mismo mecanismo permite a los usuarios reemplazar componentes.

Una de las mayores fortalezas del entorno de aplicación de Android es que se aprovecha el lenguaje de programación Java. El SDK de Android no acaba de ofrecer todo lo disponible para su estándar del entorno de ejecución Java (JRE), pero es compatible con una fracción muy significativa de la misma.

Los servicios más importantes que incluye son:

Views: extenso conjunto de vistas, (parte visual de los componentes).

Resource Manager: proporciona acceso a recursos que no son en código.

Activity Manager: maneja el ciclo de vida de las aplicaciones y proporciona un sistema de navegación entre ellas.

Notification Manager: permite a las aplicaciones mostrar alertas personalizadas en la barra de estado.

Content Providers: mecanismo sencillo para acceder a datos de otras aplicaciones (como los contactos).

Aplicaciones

Este nivel está formado por el conjunto de aplicaciones instaladas en una máquina Android. Todas las aplicaciones han de correr en la máquina virtual Dalvik para garantizar la seguridad del sistema. Normalmente las aplicaciones Android están escritas en Java.

Por lo anterior, también es necesario definir lo que es un **sistema operativo**. Según Niño Camazón: “Es un programa o conjunto de programas que permite una comunicación simple y segura entre el usuario y el hardware, también se encarga de gestionar y optimizar los recursos hardware, tales como el procesador, la memoria y los dispositivos de entrada y salida.”³³

Una de las principales atracciones de los smartphones es la variedad de aplicaciones de las que los usuarios disponen en la respectiva **tienda online** que corresponda a la plataforma de su teléfono. Para el caso de Android se tiene la Google Play, que Motorola³⁴ define así: “Es un destino de entretenimiento digital, donde puede encontrar, disfrutar y compartir su música favorita, películas, libros y aplicaciones en la web y en su teléfono Android o Tablet. Google Play está totalmente basado en la nube para que toda su música, películas, libros y las aplicaciones se almacenen en línea, siempre disponibles para usted, así usted nunca tendrá que preocuparse de perderlos o de tener que moverlos de nuevo”.

³³ NIÑO CAMAZÓN, Op. cit., p. 34.

³⁴ ¿Qué es Google Play? [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: <http://www.lgmobile.cl/smartphones>

Desde esta tienda se puede acceder a una vasta serie de aplicaciones gratuitas y otras de pago que se pueden instalar en el smartphone dependiendo de si el smartphone cumple con los requerimientos que indica el respectivo desarrollador. En estas tiendas en línea (bien sea de Android o de otra plataforma) residen principalmente aplicaciones de un solo tipo, las que son conocidas como nativas. Como lo deja ver Arroyo³⁵, una **aplicación nativa** es aquella que debe ser desarrollada utilizando el lenguaje de programación propio de la plataforma para la cual se construye.

No obstante, las aplicaciones nativas no son el único tipo que existen para los smartphones. Se puede decir que en general hay tres tipos: las nativas, que ya se explicaron, las de **tipo web**, que como también lo presenta Arroyo, básicamente son aquellas capaces de funcionar en cualquier dispositivo con conectividad; y las híbridas que son prácticamente basadas en tecnologías web pero que luego se empaquetan para ser instaladas en el terminal como si se tratara de una aplicación nativa.

Respecto a las aplicaciones y las tiendas online de cada plataforma, Arroyo³⁶ dice: “La moda de las aplicaciones y el aprovechamiento de sus tiendas como un recurso de marketing pueden ser dos buenas razones que han llevado a los proveedores de contenidos a trasladar sus sitios web a aplicaciones.”

Este trabajo de grado plantea la confrontación de dos aplicaciones que fueron desarrolladas por los autores del mismo, siendo la primera de tipo nativa y la segunda, híbrida (que viene teniendo mucha relación con las de tipo web), las cuales después de haber sido construidas se han de someter a una evaluación de la calidad. Con respecto a lo anterior, Pressman dice: “Algunos desarrolladores de software continúan creyendo que la calidad de éste es algo en lo que se debe comenzar a preocupar sólo después de que se haya generado el código.”³⁷

Del aporte de Pressman se identifica que contrario a lo que muchos actualmente piensan, la gestión de la calidad del software, como técnicamente se denomina, es una actividad que debe ser tenida en cuenta durante todo el proceso de desarrollo. Sin embargo, para efectos de este trabajo, la evaluación de la calidad de los atributos se llevó a cabo una vez la aplicación móvil es un producto terminado.

Según Álvarez³⁸ para la mejor comprensión de lo que es la calidad en el software, existen los modelos de calidad. El estándar ISO/IEC 9126 -1 define **modelo de calidad** como: “conjunto de características, subcaracterísticas y las relaciones entre ellas que proveen la base para especificar requisitos de calidad con el fin de realizar un proceso de

³⁵ ARROYO, Op. Cit., p. 66.

³⁶ Ibid, p.66

³⁷ PRESSMAN, Roger. Ingeniería del Software: Un enfoque práctico. Sexta edición. México : McGraw-Hill, 2005. p. 767.

³⁸ ÁLVAREZ, Antonio. Los atributos de Calidad del Software. [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: <http://www.es.sogeti.com/PageFiles/173/SEPT-ANTONIO-ALVAREZ-SOGETI.pdf>

evaluación”³⁹. Dichas características a las que se refiere el estándar ISO/IEC 9126-1 son los **atributos de calidad del software** y que además distintos autores resaltan su gran importancia, como el caso de Granollers, Lorés y Cañas que dicen: “Actualmente, el atributo de calidad en la fabricación de cualquier producto toma una importancia relevante que determina si un proceso es mejor o peor que otro, y el desarrollo de sistemas interactivos no es una excepción.”⁴⁰

Como continúan diciendo Granollers, Lorés y Cañas⁴¹, el estándar ISO/IEC 9126-1: Software engineering-Product Quality define la calidad de una aplicación a partir de los atributos de funcionalidad, fiabilidad, usabilidad, eficiencia, mantenimiento y portabilidad.

Por otro lado, con respecto a los atributos de calidad del software, Llorens Fábregas expresa: “Si bien es difícil poder escribir una definición totalmente satisfactoria y general de calidad en sistemas, cada uno de nosotros puede, de manera intuitiva, definir cuáles son las “cosas” que caracterizan a un sistema de calidad o, en otras palabras, definir cuáles son los atributos que nos permiten decidir si un determinado sistema es de alta, mediana o baja calidad.”⁴²

Llorens Fábregas presenta los atributos de calidad del software, así:

1. **Adecuación al negocio.** Es el grado en que un sistema cumple con sus especificaciones, requerimientos y objetivos funcionales.
2. **Confiabilidad.** Es el grado de precisión con que un sistema realiza sus operaciones.
3. **Eficiencia.** Es la relación que guardan la cantidad y la complejidad de las funciones que un sistema realiza para sus usuarios, y la cantidad de recursos requeridos para realizarlas.
4. **Facilidad de operación.** Es una medida del esfuerzo requerido para operar un sistema, preparar sus datos de entrada e interpretar los resultados que produce. Normalmente esta cualidad se asocia con la “amigabilidad” del sistema.
5. **Facilidad de entrenamiento.** Es una medida del esfuerzo requerido para aprender a operar y a interpretar los resultados que produce un sistema.

³⁹ INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO/IEC 9126-1 (2000). Software engineering — Product quality — Part 1: Quality model. p. 8.

⁴⁰ GRANOLLERS, Toni; LORÉS VIDAL, Jesús y CAÑAS DELGADO, José Juan. Diseño de Sistemas Interactivos centrados en el usuario. Barcelona: UOC, 2005. p.121.

⁴¹ Ibid., p. 121.

⁴² LLORENS FÁBREGAS, Juan. Gerencia de Proyectos de Tecnología de Información. Caracas : CEC, S.A., 2005. p. 135.

6. **Integración.** Es la capacidad que tiene un sistema para integrarse con los restantes sistemas de la empresa, intercambiando y compartiendo datos con otras aplicaciones o usuarios.
7. **Disponibilidad.** Es una medida de la facilidad con que el usuario puede tener acceso a las aplicaciones y la información procesada por ellas. Normalmente, la disponibilidad de un sistema se asocia con los niveles de servicio o con la continuidad del servicio prestado por el sistema.
8. **Seguridad.** Es el grado de dificultad que encuentran las personas no autorizadas al tratar de tener acceso a las funciones, los componentes o los datos de un sistema.
9. **Integridad.** Es el grado de dificultad establecido para evitar que algún evento o persona, autorizada o no, pueda dañar, en forma voluntaria o accidental, los componentes o la información manejada por el sistema.
10. **Mantenibilidad.** Es una medida del esfuerzo requerido para localizar y corregir la fuente de error de un sistema.
11. **Flexibilidad.** Es una medida del esfuerzo requerido para mejorar o ampliar las funciones que realizan los componentes de un sistema.
12. **Facilidad de prueba.** Es una medida del esfuerzo requerido para probar un sistema y, de esa forma, asegurar que esté libre de errores y cumple con sus especificaciones.
13. **Reusabilidad.** Es la capacidad de adaptación de los componentes de un sistema para ser utilizados en otras aplicaciones.
14. **Portabilidad.** Es una medida del esfuerzo requerido para transferir un sistema, desde una plataforma de hardware o software a otra.⁴³

2.3 MARCO TEÓRICO

De acuerdo al objetivo general de este trabajo de grado, el conjunto de conocimientos que atañen al problema motivo de estudio, es el referente al concepto de aplicación móvil y los tipos que existen, así como la comparación entre éstos.

Ahora, del proceso de recopilar la información relevante se determinó que principalmente no se encuentra una teoría completa. Sin embargo, se hallan trabajos realizados por compañías asociadas al desarrollo de software y que se han interesado en comparar las

⁴³ Ibid., p. 135-136.

tecnologías de desarrollo de aplicaciones móviles, habiendo llevado a cabo estudios que han sido publicados mediante un libro blanco, como en el caso de Global Intelligence Alliance que dio a conocer un informe titulado *Native or Web Application? How Best to Deliver Content and Services to Your Audiences over the Mobile Phone* (¿Aplicación nativa o web? La mejor manera de llevar contenidos y servicios a sus audiencias a través del teléfono móvil, en español). El libro blanco antes mencionado, cabe destacarlo puesto que es un estudio con un cierto grado de apoyo científico y técnico que aporta variables importantes.

2.3.1 Guías no estudiadas

Con respecto a una serie de guías no estudiadas en las que se presentan ideas relacionadas con el problema de estudio, es necesario hacer referencia a tal conjunto de conocimientos que orientan dentro de la confrontación de las tecnologías de desarrollo de aplicaciones móviles. En Internet es posible realizar una búsqueda y fácilmente encontrar una gran cantidad de sitios que abordan la temática de las aplicaciones móviles y la comparación entre los tipos que existen, a saber, tres principalmente: aplicación nativa, web e híbrida. El libro blanco de la compañía Lionbridge⁴⁴, *Mobile Web Apps vs. Mobile Native Apps: How to Make the Right Choice*, es tomado como una referencia importante para lo que se describe a continuación:

La **aplicación nativa** es definida como aquella implementada en el lenguaje nativo del terminal y por tanto se diseña para un sistema operativo móvil específico y el firmware del dispositivo.

Al hablar de las ventajas de las aplicaciones nativas frente a las web, hay que resaltar que permiten aprovechar al máximo todos los recursos presentes en el dispositivo tanto de hardware como de software: cámara, GPS, sensores (acelerómetro), bluetooth, agenda de contactos, calendario, brújula, etc.

Ventajas:

- Mejor experiencia de usuario.
- Mayor rendimiento.

Desventajas:

- Coste de desarrollo más alto puesto que requieren mayor esfuerzo de desarrollo. Deben ser desarrolladas para cada plataforma.
- Son distribuidas por los market place (Tienda Online); en relación a esto las nativas deben cumplir con requisitos para ser validadas.

⁴⁴ Mobile Web Apps vs. Mobile Native Apps: How to Make the Right Choice. [En línea]. [Citado el 15 de marzo de 2013]. Disponible en internet: http://www.lionbridge.com/files/2012/11/Lionbridge-WP_MobileApps2.pdf

Las aplicaciones nativas, por su naturaleza, precisamente son las que sobresalen en rendimiento puesto que son escritas en lenguaje de la plataforma para la cual se desarrollan y por esa misma razón aquellas aplicaciones que se consideran pesadas, se estima que seguirán siendo del tipo nativo con tal de brindar el mejor desempeño en el dispositivo, aprovechando al máximo los recursos de procesamiento; tal es el caso de los juegos.

En cuanto a las aplicaciones se refiere, los usuarios de smartphones y tablets, en general, están bastante relacionados con la tienda online, por tanto se considera que la primera forma (y quizás la única para muchos) de instalar una aplicación en su dispositivo es entrando al market place, lo que en este sentido pone en clara desventaja a las aplicaciones web puesto que no pueden hacer parte de las tiendas online, por lo menos en la actualidad.

La **aplicación web** es aquella completamente desarrollada en HTML 5, hojas de estilo y JavaScript, y funcionan en un navegador.

Ventajas:

- Primeramente hay que destacar que por su naturaleza son multiplataforma.
- La tecnología web se considera más sencilla para desarrollar.
- No dependen de los market place, por lo que no es necesario someterse a controles para la publicación de la aplicación misma ni de las actualizaciones, de modo que llegan directamente a los usuarios.

Desventajas:

- Dependen de la conexión a internet.
- Su rendimiento es menor que la aplicaciones nativas.

Comparando la aplicación nativa frente a la web, es posible identificar que en algunos casos lo que para las nativas es una gran ventaja, para las de tipo web a termina siendo una clara oportunidad. Por ejemplo, se ha dicho que las aplicaciones nativas son distribuidas a través de las tiendas en línea y que por tanto requieren cumplir con una serie de exigencias para ser publicadas, mientras que las aplicaciones web por no poder pertenecer a los market place tienen la grandiosa posibilidad de entregar actualizaciones de forma transparente al usuario, además de que el desarrollador tiene todo el control al residir en el servidor.

El tercer tipo es la **aplicación híbrida**, aquella que se caracteriza por ser una solución nativa que posee gran parte de contenido web.

- Se benefician de las ventajas de facilidad de implementación de las de tipo web.
- No debe ser desarrollada para cada plataforma en específico.

Puesto que no siempre son distribuidas a través de tienda online, algunas veces este tipo de aplicaciones son entregadas a los usuarios por el sitio web de la misma empresa a la cual pertenece. Se puede dar el caso de alguien que haya desarrollado una aplicación web y lastimosamente no cuenta con un sitio en Internet ampliamente difundido o simplemente no existe, por lo cual podría resultar más complicado dar a conocer dicha aplicación.

2.3.2 Enfoques de desarrollo de aplicaciones móviles

Uno de los aspectos más llamativos de los teléfonos inteligentes es la posibilidad de instalar una gran cantidad de aplicaciones que cumplen tareas específicas, poder realizar esas pequeñas tareas sin importar el lugar donde se encuentre el usuario es realmente interesante, pues mientras se encuentra en el autobús hay tiempo para revisar cuentas de correo, realizar una compra, hacer reservaciones en un restaurante, entre otras cosas. Hace unos años estas actividades requerían sentarse frente a un ordenador, pero hoy en día basta con llevar la mano al bolsillo y tomar el smartphone; por esta razón muchos le han dado al teléfono móvil el nombre de ordenadores de bolsillo.

Si bien es cierto que estos teléfonos son pequeñas computadoras, a la hora de desarrollar aplicaciones el programador debe tener en cuenta una serie de elementos o características propias de estos aparatos que particularizan el desarrollo de las mismas. Los teléfonos inteligentes no dejan de ser precisamente eso: teléfonos. Por lo tanto, operaciones como recibir y realizar llamadas tienen prioridad sobre cualquier proceso. De tal modo que características como sensores, cámara, mensajería, geo localización y vídeo conferencia son reunidas en las aplicaciones para mejorar la experiencia del usuario.

Por otra parte el programador debe considerar las capacidades hardware de estos dispositivos, aunque día a día estos aparatos mejoran en este aspecto, existen aún muchos con capacidades reducidas; también se enfrenta a una gran variedad de teléfonos con diferentes tamaños y resoluciones de pantalla, un factor importante al momento del diseño.

Es importante notar que a pesar de que los factores básicos del desarrollo de software se mantienen en la construcción de aplicaciones móviles, existen muchas características que lo convierten en un ámbito diferente. Debido a esto, han aparecido varias tecnologías de desarrollo que presentan herramientas y abordan el desarrollo de aplicaciones de forma distinta. Es deber del desarrollador conocer estas tecnologías para que luego de un análisis pueda elegir la que más se adapte a las necesidades de su proyecto. A continuación se presentan de forma breve algunos de estos enfoques de desarrollo de aplicaciones móviles, de acuerdo a lo expuesto por Paredes, Santacruz y Domínguez⁴⁵:

2.3.2.1 Clientes nativos. Bajo estas condiciones las aplicaciones se escriben en lenguajes de bajo nivel como C o ensamblador y compilados en un lenguaje de máquina para un grupo específico de procesadores y configuraciones hardware, y luego se ejecuta como código nativo en esos dispositivos móviles.

⁴⁵ PAREDES VELASCO, Maximiliano; SANTACRUZ VALENCIA, Liliana; DOMÍNGUEZ MATEOS, Francisco. Programación Multimedia y Dispositivos Móviles. Editorial Ra-Ma, 2012.p. 16.

El desarrollador se enfrenta ante una gran complejidad pues se entra a trabajar con los recursos de la máquina directamente, no obstante se logra un gran rendimiento; si esta es la prioridad, esta opción es la que logra mayores resultados en ese sentido.

Cabe aclarar que el término *desarrollo nativo* en el ámbito de desarrollo móvil se entiende de manera diferente, pues hace referencia al desarrollo de una aplicación para una plataforma en específico (Android, iOS, Windows Phone, BlackBerry OS).

2.3.2.2 Clientes JME (Java Platform Micro Edition, Plataforma Java Micro Edición). Es ampliamente conocido que Java brinda reconocidas opciones en el desarrollo de software, lo que genera confianza para usarla en esta área. En el mundo móvil esta tecnología mantiene su filosofía: las aplicaciones se escriben en Java y se compilan para ejecutarse en una máquina virtual Java (JVM), diseñada específicamente para dispositivos móviles.

2.3.2.3 Clientes basados en web. Un cliente basado en web se ejecuta dentro del navegador del dispositivo, desde el cual se accede a la página que el servidor web envía de la misma manera como ocurre en un computador tradicional. De una u otra forma este enfoque permite la posibilidad de llegar a más dispositivos dado que todos cuentan con al menos un navegador. Resulta importante la manera como se maneja la compatibilidad entre navegadores para que la aplicación funcione en la mayoría de éstos. El desarrollo de este tipo es más ágil debido a que se hace uso de herramientas de desarrollo web ampliamente difundidas (HTML, CSS, JavaScript), en este caso aplicado a los teléfonos móviles, no obstante se presenta dificultad a la hora de querer acceder a recursos del dispositivo (sensores, cámara, entre otros) por lo que se verá afectado el rendimiento. Cabe resaltar que esta tecnología ha mejorado mucho en este aspecto y que aún se encuentra en una etapa de desarrollo.

2.3.2.4 Clientes basados en middleware. El middleware es un software que ayuda a una aplicación a interactuar o comunicarse con otras aplicaciones, software, redes, hardware y/o sistemas operativos. El diseño de este tipo de aplicaciones utiliza un conjunto de herramientas y tiempos de ejecución para abstraer la aplicación y las tareas de adquisición de datos lejos de cualquier dispositivo.

El principal beneficio de este tipo de implementación es el desarrollo rápido y el mantenimiento del código específico del dispositivo. En el enfoque middleware se crea una aplicación utilizando un conjunto de herramientas de diseño para terceros, en un sistema operativo propietario. Además controla aspectos como: lo que se muestra en pantalla, los diálogos, los datos en el lado cliente, la gestión del estado de conexión del dispositivo y la gestión de los datos fuera de línea.

Para resumir, cada tecnología tiene sus ventajas y desventajas, teniendo en cuenta esto se puede determinar cuál de ellas ajusta mejor al desarrollo que se esté llevando a cabo. A continuación se presenta una tabla donde se mencionan las ventajas y desventajas de las tecnologías mencionadas anteriormente:

Cuadro 1. Beneficios e inconvenientes de los diferentes enfoques para el desarrollo de aplicaciones para dispositivos móviles

Arquitectura	Beneficios	Inconvenientes
Cliente nativo	<ul style="list-style-type: none"> - Aplicaciones sofisticadas y control sobre el entorno local - Capacidades multitarea sobre muchas plataformas 	<ul style="list-style-type: none"> - El más alto nivel de esfuerzo de desarrollo - Diferente código base para diferentes dispositivos
Java ME	<ul style="list-style-type: none"> - El mismo código base puede soportar múltiples aplicaciones - Aplicaciones más sofisticadas 	<ul style="list-style-type: none"> - Algunos límites de capacidad (no multitarea) - Requiere prueba y adaptación para las Plataformas
Navegador web	<ul style="list-style-type: none"> - Desarrollo rápido - No requiere mantenimiento del código cliente - Métodos Web 2.0 disponibles - Trabajo con un amplio rango de dispositivos móviles 	<ul style="list-style-type: none"> - Menos sensible y capas que las aplicaciones nativas
Middleware móvil	<ul style="list-style-type: none"> - Alto nivel de capacidad con reducido esfuerzo de desarrollo 	<ul style="list-style-type: none"> - Tasas adicionales de licencia - Curva de aprendizaje y esfuerzo de integración potencialmente grande.

Fuente: PAREDES VELASCO, Maximiliano; SANTACRUZ VALENCIA, Liliana; DOMÍNGUEZ MATEOS, Francisco. Programación Multimedia y Dispositivos Móviles.

2.3.3 Aplicación nativa o web según estudio de Global Intelligence Alliance

Por otra parte, el ya mencionado trabajo realizado por Global Intelligence Alliance⁴⁶, titulado Native or Web Application? How Best to Deliver Content and Services to Your Audiences over the Mobile Phone, en el que se aborda esta temática definiendo términos relevantes como se presenta a continuación:

⁴⁶ Native or Web Application? How Best to Deliver Content and Services to Your Audiences over the Mobile Phone. [En línea] [Citado el 12 de marzo de 2013] Disponible en internet: http://www.globalintelligence.com/insights-analysis/white-papers/native-or-web-application-how-best-to-deliver-cont/GIA%20Industry%20White%20Paper%202010_Native%20or%20Web%20App_How%20Best%20to%20Deliver%20Content%20and%20Services%20to%20Your%20Audiences%20over%20the%20Mobile%20Phone_Apr-10.pdf

2.3.3.1 Aplicación Nativa. Es una aplicación que se diseña específicamente para ser ejecutada en el sistema operativo del dispositivo y en el firmware de la máquina. Esta típicamente necesita ser adaptada/ajustada para dispositivos diferentes.

- Se destaca, además, que los costos de desarrollo y las actualizaciones son generalmente más altos para las aplicaciones nativas.

De la definición anterior se distingue que cada proyecto de desarrollo de aplicación nativa puede estar pensado para varias plataformas, pero lograr una versión nativa para cada una de ellas resulta altamente complicado y costoso.

Además, se dice que en el corto plazo las aplicaciones nativas probablemente seguirán siendo la interfaz preferida especialmente para aplicaciones más pesadas, por ejemplo, Juegos 3D y de pago por descarga de aplicaciones, gracias a las opciones de facturación integradas a más tiendas de aplicaciones nativas.

2.3.3.2 Aplicación Web. Es una aplicación en la cual todo o algunas partes son descargadas de la Web cada vez que se ejecuta. Esta puede funcionar en cualquier dispositivo compatible con tecnología web.

La plataforma web es particularmente útil para los servicios basados en suscripción, como las comunicaciones, las noticias y la información meteorológica, los servicios financieros y los centros comerciales, donde el diseño interactivo y análisis de usuario son más relevantes. Asimismo, se habla de la importancia de la forma de distribución abierta que tienen las aplicaciones web, es decir, independientes de las tiendas de aplicaciones.

Cabe señalar que Global Intelligence Alliance define *aplicación nativa* y *aplicación web* de una forma que se acerca mucho a los conceptos que se presentaron inicialmente.

El estudio por parte de Global Intelligence Alliance (GIA), también revela conclusiones importantes con respecto a los tipos de aplicaciones móviles desde varios enfoques:

- Se observa una mayor adopción y un mayor volumen de uso entre los usuarios en las aplicaciones nativas. El 30% de los encuestados ve el doble de volumen de uso para las aplicaciones nativas en comparación con las aplicaciones web.
- Las aplicaciones web tienen ventajas claras en cuanto a costos. El desarrollo de aplicaciones nativas tienen costos más altos debido a la necesidad de llevarlas a cada plataforma de sistema operativo y la rigurosidad del control de calidad.
- Las pequeñas empresas (menos de 50 trabajadores) tienen la aplicación nativa como interfaz principal.

- Las empresas más grandes con mayores presupuestos para el desarrollo tienden a aprovechar las ventajas de ambos tipos de aplicaciones.
- Por interfaz de usuario y aprovechamiento de hardware se prefiere la aplicación nativa.
- El control de distribución y las ventajas de costos son factores clave en el desarrollo de aplicaciones web. Tal vez sorprendentemente, el rápido tiempo de puesta en marcha y la libertad de facturación emergen como preocupaciones secundarias.
- A menudo, es más rápido el despliegue de aplicaciones web. El 60% de los encuestados ve más rápido el lanzamiento a través de una aplicación web.
- Las aplicaciones web son más propensas a ser desarrolladas dentro de la empresa que las aplicaciones nativas. Si bien el desarrollo interno móvil aparece generalmente dominante en todos los ámbitos.

Así pues, la revisión de la literatura relacionada con el problema objeto de estudio, da como resultado que a la fecha, empresas de desarrollo de software o que están asociadas a éste, se han interesado en definir con certeza los tipos de aplicaciones móviles que existen. Además, se identifica que particularmente, organizaciones como Global Intelligence Alliance han realizado estudios mediante entrevistas especializadas a desarrolladores, editores, agencias de diseño y expertos en la materia, para comparar la aplicación nativa frente a la web. Es decir, dicho estudio presenta las principales características de la aplicación nativa y de la web, vistas desde varias perspectivas. Es necesario notar que la información presentada en este marco resulta provechosa para la orientación de este trabajo de grado.

2.4 MARCO LEGAL

La normatividad en la que se apoyó la realización de este trabajo, se despliega a continuación:

2.4.1 Artículo 71 de la Constitución Política de Colombia

Este artículo otorga al Estado la responsabilidad de promover el desarrollo tecnológico e incentivar a quienes se dediquen a trabajar en este ámbito "... El Estado creará incentivos para personas e instituciones que desarrollen y fomenten la ciencia y la tecnología y las demás manifestaciones culturales y ofrecerá estímulos especiales a personas e instituciones que ejerzan estas actividades."⁴⁷

⁴⁷ CONSTITUCIÓN POLÍTICA DE COLOMBIA. Artículo 71. [En línea]. [Citado el 13 de noviembre de 2012]. Disponible en internet: http://www.secretariassenado.gov.co/senado/basedoc/cp/constitucion_politica_1991_pr002.html

Es de gran importancia lo que se acaba de citar puesto que es precisamente la Constitución Política que estando por encima de todas las leyes, ampara la actividad de desarrollo tecnológico.

2.4.2 Decreto 393 de 1991

Por el cual se dictan normas sobre asociación para actividades científicas y tecnológicas, proyectos de investigación y creación de tecnologías.⁴⁸

Este decreto ampara las actividades científicas y tecnológicas, proyectos de investigación y creación de tecnologías, de acuerdo a lo dicho en el artículo 1. Asimismo, el artículo 2 del mismo decreto, expone que dentro de los motivos de la asociación se encuentran: Adelantar proyectos de investigación científica así como también negociar, aplicar y adaptar tecnologías nacionales o extranjeras.

Por lo anterior, este proyecto de grado que se ocupa del ámbito de los dispositivos móviles y específicamente del desarrollo de aplicaciones para smartphones, está incluido dentro de las disposiciones del Estado.

2.4.3 Ley 1341 de 2009

Por la cual se definen principios y conceptos sobre la sociedad de la información y la organización de las Tecnologías de la Información y las Comunicaciones –TIC–, se crea la Agencia Nacional de Espectro y se dictan otras disposiciones.⁴⁹

De acuerdo al artículo segundo de esta ley, este proyecto de grado está alineado a una política de Estado: “La investigación, el fomento, la promoción y el desarrollo de las Tecnologías de la Información y las Comunicaciones son una política de Estado que involucra a todos los sectores y niveles de la administración pública y de la sociedad, para contribuir al desarrollo educativo, cultural, económico, social y político e incrementar la productividad, la competitividad, el respeto a los Derechos Humanos inherentes y la inclusión social”

De la misma manera, más adelante continúa: “Las Tecnologías de la Información y las Comunicaciones deben servir al interés general y es deber del Estado promover su acceso eficiente y en igualdad de oportunidades, a todos los habitantes del territorio nacional.”

⁴⁸ PRESIDENCIA DE LA REPÚBLICA DE COLOMBIA. Decreto 393 de 1991. [En línea]. [Citado el 15 de Noviembre de 2012]. Disponible en internet: http://www.colciencias.gov.co/sites/default/files/upload/reglamentacion/decreto_393_de_1991.pdf

⁴⁹ SENADO DE LA REPÚBLICA DE COLOMBIA. Ley 1341 de 2009. [En línea]. [Citado el 16 de noviembre de 2012]. Disponible en internet: http://www.secretariasenado.gov.co/senado/basedoc/ley/2009/ley_1341_2009.html

3. DISEÑO METODOLÓGICO

3.1 TIPO DE INVESTIGACIÓN

Este proyecto de grado se considera de tipo estudio aplicado. De acuerdo a Landeau⁵⁰: “...está encaminado a la resolución de problemas prácticos con un margen de generalización limitado. El aporte al conocimiento científico es secundario.” De lo que se puede ver que la presente propuesta se ajusta a un estudio que resolvió un asunto particular. Además, Landeau⁵¹ también dice: “El estudio aplicado se utiliza cuando el investigador se propone aplicar el conocimiento para resolver problemas de cuya solución depende el beneficio de individuos o comunidades. Es un tipo de estudio que se emplea con frecuencia en el contexto industrial, orientado a la producción de materiales, instrumentos sistemas, métodos, procedimientos y modelos.”

Con relación a lo expresado anteriormente, este trabajo está enfocado en dar una solución concreta teniendo como base el desarrollo de aplicaciones móviles y el conocimiento de los atributos de calidad del software para lograr confrontar dos tecnologías de desarrollo móvil, con lo cual se hace un aporte a la comunidad de desarrollo y más específicamente a aquellos que comienzan en este ámbito y desconocen cuál es la tecnología más apropiada para sus requerimientos.

3.2 POBLACIÓN

Al considerar el territorio al cual corresponde la realización de este trabajo de grado, las personas con las cuales se aplicó la técnica e instrumento de recolección de datos fueron los estudiantes de ingeniería de sistemas de la UFPSO teniendo en cuenta que ésta es la primera comunidad beneficiada.

De acuerdo al reporte entregado por la oficina de Admisiones, Registro y Control de la UFPSO, los datos con respecto a los estudiantes matriculados en ingeniería de sistemas, para el 25 de julio, segundo semestre de 2012, fecha en la que se realizó la solicitud, son: 255 estudiantes.

3.3 MUESTRA

Con lo explicado anteriormente es importante mencionar que por el tamaño considerable de la población, 255 estudiantes de ingeniería de sistemas, se calculó el tamaño de la muestra:

$$n = \frac{Z^2 \sigma^2 N}{e^2(N-1) + Z^2 \sigma^2}$$

⁵⁰ LANDEAU, Rebeca. Elaboración de trabajos de investigación. Caracas : Alfa, 2007. p. 54.

⁵¹ Ibid. p.54.

Donde:

n = el tamaño de la muestra.

N = tamaño de la población.

σ = Desviación estándar de la población.

Z = Valor obtenido mediante niveles de confianza.

e = Límite aceptable de error.

Para este análisis se han usado los siguientes valores:

$N = 255$, que corresponde al total de alumnos matriculados al plan de estudios de ingeniería de sistemas.

$Z = 1.65$

$\sigma = 0.5$

$e = 0.1$

Aplicando la fórmula descrita anteriormente con los valores presentados, obtenemos la muestra (n):

$n = 53.88 \approx 54$

Como se puede observar, el resultado se aproxima a 54 por tratarse de una cantidad de personas.

3.4 TÉCNICA DE RECOLECCIÓN

Luego de haber revisado las definiciones de algunas de las técnicas de recolección de datos como la observación y la encuesta, y teniendo en cuenta lo que dice Grasso:

... la encuesta permite obtener datos de manera más sistemática que otros procedimientos de recolección de datos. Hace posible el registro detallado de los datos, el estudiar una población a través de muestras con garantías de representatividad, la generalización de las conclusiones con conocimiento de los márgenes de error y el control de algunos factores que inciden sobre el fenómeno a observar, como por ejemplo las formas de efectuar las preguntas y el contexto en que están se formulan y contestan.⁵²

Se determinó la encuesta como técnica de recolección de datos para este proyecto, puesto que con relación a lo expresado por Grasso, es altamente importante lograr que la encuesta aplicada represente a la población de estudiantes de ingeniería de sistemas de la UFPSO.

⁵² GRASSO, Livio. Encuestas, Elementos para su diseño y análisis. Argentina : Encuentro. 2006. p. 13.

3.5 INSTRUMENTO

En lo que respecta al instrumento para la recolección de información, dice García⁵³: “Las herramientas más frecuentes para la realización de una encuesta son la entrevista y el cuestionario, ambas técnicas de interrogación.” Para este trabajo, como ya se dijo, la técnica de recolección empleada fue la encuesta. Puesto que las posibilidades de instrumento son el cuestionario y la entrevista, se descarta la última principalmente por el tamaño de la muestra; evidentemente, realizar una entrevista a cada estudiante, siendo éstos 54, implica una serie de actividades propias de este instrumento que desgastarían a los investigadores.

Así, el instrumento que se escogió para este trabajo fue el cuestionario, que además se ajusta a los requerimientos de este trabajo, tal como lo señala García⁵⁴ con respecto a sus características: “... las preguntas claras y concretas, presentadas en un orden rígido y preestablecido que no puede alterarse; ello generará respuestas cortas y de contenido limitado. La duración con frecuencia es corta y generalmente se realiza en un solo encuentro. La finalidad central es obtener datos de un amplio grupo de personas con respecto al problema de investigación.” Entonces, se identifica que la forma como se aplica el cuestionario es apropiado para este caso: un grupo de estudiantes de ingeniería de sistemas. Luego de haber diseñado y revisado el cuestionario, se procedió a aplicar la encuesta de forma grupal, es decir, mientras los estudiantes se encontraban en horas de clase, habiendo dado breves explicaciones con respecto al trabajo que se estaba adelantando y al cuestionario.

El cuestionario diseñado para este caso, es el que se aparece en el Anexo A de este trabajo. Dado que las personas que responden este cuestionario son estudiantes de ingeniería de sistemas, es posible hacer uso, en su redacción, de un lenguaje al alcance de dichos estudiantes, que sin ser muy cotidiano o escaso, tampoco tiende a ser complicado de comprender. Los principales resultados esperados son:

- a) Conocer qué tipo de celular poseen los encuestados.
- b) Determinar el nivel de utilización de smartphones.
- c) Identificar cuál es la plataforma móvil más utilizada entre los encuestados.
- d) Definir qué tan relacionados están los estudiantes encuestados con el desarrollo de aplicaciones móviles y si se han construido.
- e) Identificar el nivel de interés que tienen con respecto a este campo del desarrollo de software.

⁵³ GARCÍA CÓRDOBA, Fernando. El cuestionario: Recomendaciones metodológicas. Córdoba : Limusa, 2004. p. 22-23.

⁵⁴ Ibid., p. 23.

3.6 ANÁLISIS DE INFORMACIÓN

Los datos obtenidos mediante el cuestionario se organizaron en cuadros y a partir de ellas se elaboraron las gráficas correspondientes. El tipo de gráfica empleado fue circular. Por razones de una mejor presentación, se expone cada pregunta junto con los resultados obtenidos.

En el cuestionario la primera solicitud de información es con respecto a los datos del encuestado, representados en el cuadro 2:

Cuadro 2. Datos del encuestado.

	Edad (años)												Sexo	
	19	20	21	22	23	24	25	26	27	28	29	33	F	M
Cantidad	2	6	9	13	6	4	2	5	4	1	1	1	23	31

Fuente: Autores del proyecto

En el cuadro 2 se observa que la edad en la que se encuentra mayor número de estudiantes es 23 años. Ahora, la mayoría de los encuestados se encuentran entre los 19 a los 26 años. Además, se puede inferir que la carrera de Ingeniería de Sistemas es más preferida entre los hombres.

I. Indique las características de su celular.

Cuadro 3. Celulares organizados por marca y modelo.

MARCA	TOTAL	MODELO	Sistema Operativo	Cantidad
SAMSUNG	13	Young Duos	Android 2.3	2
		Galaxy Ace	Android 2.3	2
		B3310	No aplica	1
		Sghd900	No aplica	1
		Gts-3350 (chat 335)	No aplica	4
		No especificado	No aplica	3
BLACKBERRY	11	9300	BlackBerry OS	4
		9320	BlackBerry OS	1
		8520	BlackBerry OS	4
		9800	BlackBerry OS	1
		6530	BlackBerry OS	1
VERYKOOL	1	No especificado	No aplica	1
ALCATEL	3	OT 880a	No aplica	1
		No especificado	No aplica	2
NOKIA	8	5130	No aplica	1
		1110	No aplica	3
		C3	No aplica	2
		1208	No aplica	1
		No especificado	No aplica	1

Cuadro 3. (Continuación)

MARCA	TOTAL	MODELO	Sistema Operativo	Cantidad
SONY ERICSSON	6	W395	No aplica	2
		Yendo w150a	No aplica	1
		W200	No aplica	2
		Xperia Play r800	Android 2.3	1
BLU	1	No especificado	No aplica	1
MOVISTAR	1	No especificado	No aplica	1
IPRO	2	E71 – TG	No aplica	1
		I5	No aplica	1
ZTE	1	No especificado	No aplica	1
HUAWEI	1	HB511	No aplica	1
MOTOROLA	2	ZN200	No aplica	1
		EX108	No aplica	1
LG	1	No especificado	No aplica	1
HTC	1	One V	Android 4.0	1
TOTAL				54

Fuente: Autores del proyecto.

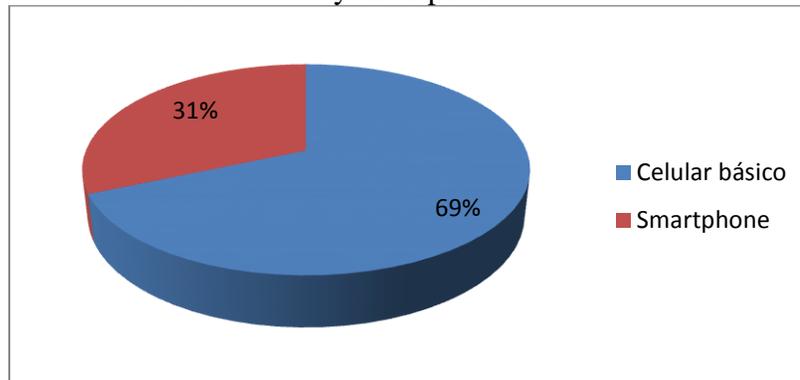
De lo anterior se puede afirmar que todos los encuestados utilizan celular. Sin embargo, para este trabajo resulta muy importante determinar la presencia de smartphones entre los encuestados, más que identificar qué tantos celulares básicos tienen. Para lo cual se construye el cuadro 4:

Cuadro 4. Cantidad de celulares básicos y smartphones.

	Celular básico	Smartphone
Cantidad	37	17

Fuente: Autores del proyecto.

Figura 6. Cantidad de celulares básicos y smartphones.



Fuente: Autores del proyecto

Como se muestra en la figura 6, el uso de smartphones entre los estudiantes de ingeniería de sistemas difícilmente alcanza el 31% frente al 69% de los encuestados que usan celulares básicos (o que no se catalogan como teléfonos inteligentes).

De otra parte, de ese 31% de smartphones, un 35.3% de esos teléfonos es de la plataforma Android y de los cuales la mayoría son de Samsung, mientras el 64.7% son del sistema de BlackBerry.

En lo que concierne a los teléfonos con Android, cabe notar que de los cuatro presentes en los resultados, 3 de ellos funcionan con la versión 2.3 de este sistema operativo, por su parte, solamente 1 cuenta con la 4.0 Ice Cream Sandwich.

La marca de fabricantes de celular más repetida es Samsung con un 24% de los teléfonos, luego aparece BlackBerry con un 20.4%, y el tercer lugar lo ocupa Nokia que alcanza 14.8% de los teléfonos. Por último, el cuarto fabricante que obtiene un porcentaje considerable es Sony Ericsson con 11,11%. De las otras marcas que aparecen relacionadas no se dan análisis puesto que tienen cifras poco significativas, además, para este trabajo no es importante resaltar el uso de celulares básicos, como ya se dijo antes.

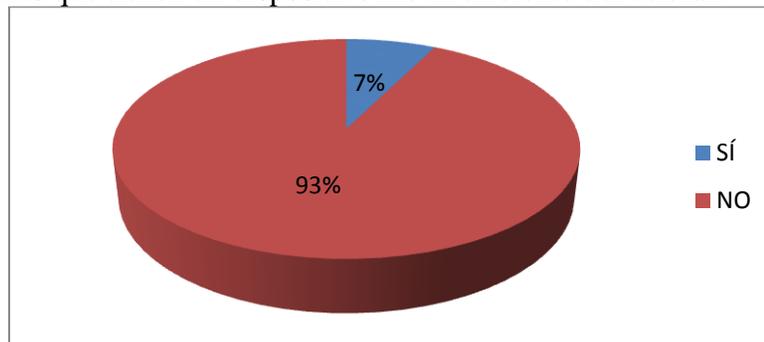
II. ¿Cuenta con otro dispositivo móvil además de su celular? ¿Cuál?

Cuadro 5. Personas que tienen un dispositivo móvil diferente del celular.

SÍ	NO
4	50

Fuente: Autores del proyecto.

Figura 7. Personas que tienen un dispositivo móvil diferente del celular.



Fuente: Autores del proyecto.

Cuadro 6. Cantidad de dispositivos móviles aparte del celular.

Dispositivo Móvil	Marca	Modelo	Cantidad
Smartphone			0
PDA (Asistente Personal Digital)			0
Tableta electrónica	Titan	7010	1
	Compumax	S9	1
Reproductor de música	Apple	iPod (Primera generación)	1
	Krono	MP4	1

Fuente: Autores del proyecto.

Del cuadro 6 hay que señalar que el porcentaje de los que poseen un segundo dispositivo móvil no es considerable puesto que de 54 estudiantes encuestados, únicamente 4 poseen un segundo dispositivo móvil.

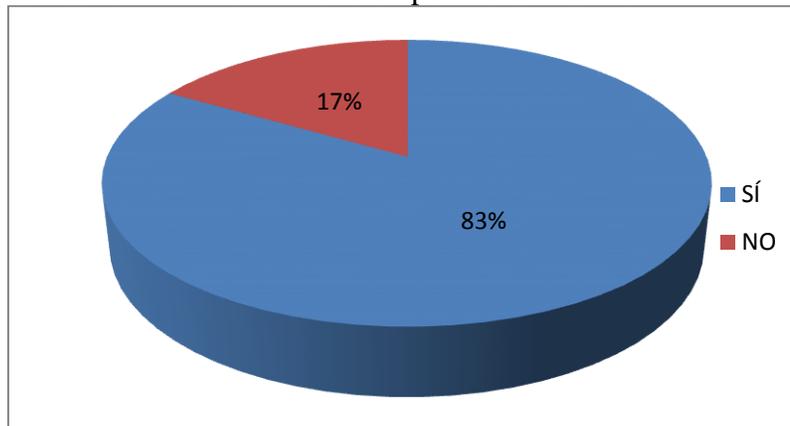
III. ¿Le interesa desarrollar aplicaciones para dispositivos móviles?

Cuadro 7. Personas interesadas en desarrollar aplicaciones móviles.

SÍ	NO
45	9

Fuente: Autores del proyecto.

Figura 8. Personas interesadas en desarrollar aplicaciones móviles.



Fuente: Autores del proyecto.

Un 83% de los encuestados manifiesta estar interesado en desarrollar aplicaciones móviles, por lo que se distingue una gran acogida de este campo entre los estudiantes de ingeniería de sistemas, lo cual también sugiere una gran expectativa entre futuros ingenieros (ver figura 8).

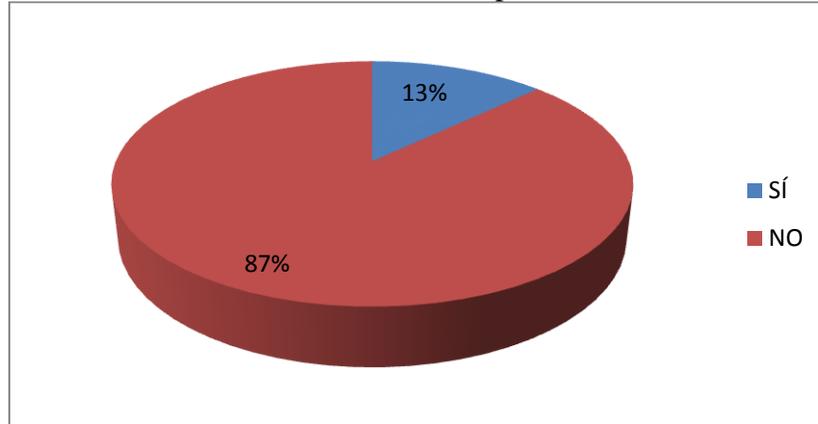
IV. ¿En alguna de las asignaturas ha tenido la oportunidad de tener por lo menos contacto inicial con el desarrollo de aplicaciones móviles?

Cuadro 8. Personas vinculadas con el desarrollo de aplicaciones móviles.

SÍ	NO
7	47

Fuente: Autores del proyecto.

Figura 9. Personas vinculadas con el desarrollo de aplicaciones móviles.



Fuente: Autores del proyecto.

Como muestra la figura 9, los resultados indican cómo los estudiantes de ingeniería de sistemas de la UFPSO manifiestan no tener por lo menos un contacto inicial con el desarrollo de aplicaciones móviles puesto que solamente 13% expresa lo contrario, y por esta razón se infiere que en las asignaturas de software de la carrera el trato al ámbito en cuestión es muy bajo.

Es decir, el nivel de interés de los estudiantes con respecto a desarrollar aplicaciones móviles es alto, no obstante, los mismos expresan que no están familiarizados con dicha área del desarrollo de software.

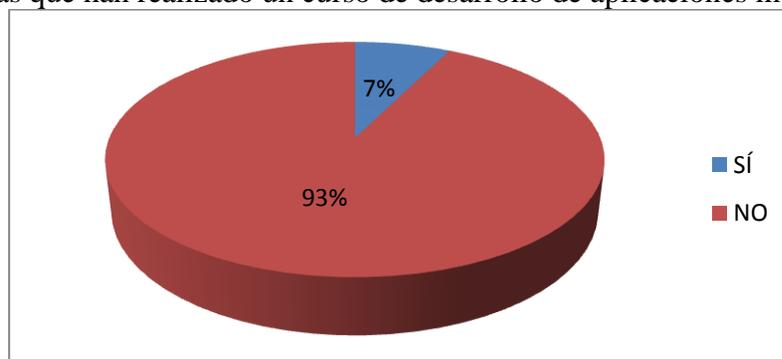
V. ¿Ha realizado algún curso para comenzar a desarrollar aplicaciones móviles?

Cuadro 9. Personas que han realizado un curso de desarrollo de aplicaciones móviles.

SÍ	NO
4 (INTERNET, SENA)	50

Fuente: Autores del proyecto.

Figura 10. Personas que han realizado un curso de desarrollo de aplicaciones móviles.



Fuente: Autores del proyecto.

Únicamente el 4% de los encuestados ha realizado un curso sobre desarrollo de aplicaciones móviles, lo cual es poco significativo. De acuerdo a los resultados se observa que el SENA ha ofrecido formación en este ámbito y por supuesto, las diferentes fuentes y materiales que se publican en sitios de Internet (ver figura 10).

Asociando los resultados de las preguntas V y VI es posible considerar que los estudiantes de ingeniería de sistemas en su mayoría no tienen una preparación específica para desarrollar aplicaciones móviles.

VI. ¿Ha desarrollado una aplicación para dispositivos móviles?

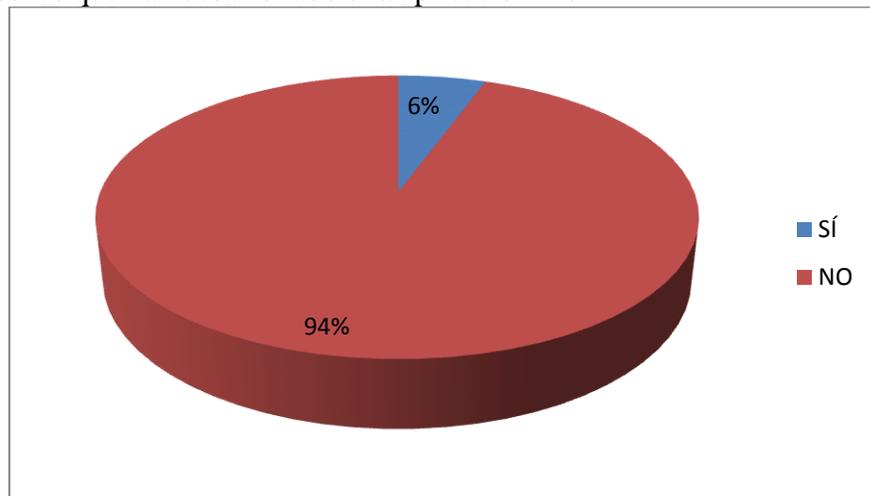
Cuadro 10. Personas que han desarrollado una aplicación móvil.

SÍ	NO
3	51

Fuente:

Autores del proyecto.

Figura 11. Personas que han desarrollado una aplicación móvil.



Fuente: Autores del proyecto.

Ahora, con la relación a la pregunta anterior, se ve reflejado que el porcentaje de quienes dicen haber desarrollado una aplicación móvil apenas alcanza un 6% que coincide prácticamente con el número de los encuestados que manifiestan haber realizado un curso para el desarrollo de aplicaciones móviles (ver figura 11).

3.7 DIAGNÓSTICO SITUACIONAL

Los datos analizados son importantes porque representan la opinión de los estudiantes de Ingeniería de Sistemas de la UFPSO, que es la primera población beneficiada con la realización de este trabajo de grado.

Luego de haber analizado los resultados, se identifica un respaldo a la escogencia de la plataforma Android sobre la que se llevó a cabo la confrontación de dos tecnologías de desarrollo de aplicaciones móviles, debido a que los encuestados manifiestan un interés considerable en el desarrollo de aplicaciones para dicha plataforma.

4. PRESENTACIÓN DE RESULTADOS

4.1 CONFRONTAR DOS TECNOLOGÍAS DE DESARROLLO DE APLICACIONES MÓVILES SOBRE LA PLATAFORMA ANDROID 4.0 DESDE LA PERSPECTIVA DE LOS ATRIBUTOS DE CALIDAD

A continuación se presentan los resultados obtenidos de acuerdo a los objetivos específicos planteados en el apartado 1.2.2 de este proyecto.

En este sentido, en la primera sección de este capítulo se expone lo relacionado con el planteamiento del proceso de evaluación de los atributos de calidad del software mediante el uso de los estándares ISO/IEC 14598 e ISO/IEC 9126 y la escogencia de la eficiencia como única característica de calidad a evaluar. En la segunda sección, se describen las tecnologías de desarrollo de aplicaciones móviles para Android, tanto para la nativa como para la basada en web. En la tercera sección se expone la construcción de la aplicación móvil con el nombre NewsApp que consiste en un lector de RSS; ésta se desarrolló en las dos tecnologías en cuestión. Finalmente, en la última sección se presenta la ejecución de la evaluación de calidad planteada a la aplicación NewsApp, que se empleó como caso de estudio para evaluar los atributos de calidad relacionados con la eficiencia.

4.1.1 Definir el proceso de evaluación de los atributos de calidad del software. En este trabajo se escogió el estándar ISO/IEC 9126 como modelo de calidad del producto software junto con el ISO/IEC 14598 como modelo de evaluación. En primer lugar, es necesario resaltar la importancia de los estándares ISO de acuerdo al alcance que este organismo internacional de normalización ha logrado mundialmente, tal como lo señala Covella: “La virtud de la organización ISO/IEC fue lograr el consenso necesario para un emprendimiento que tuviera alcance y reconocimiento internacional”⁵⁵. De acuerdo a cifras entregadas por la misma organización –ISO, en la actualidad cuenta con miembros de 163 países y 3368 organismos técnicos dedicados a observar el desarrollo de las normas ISO⁵⁶. En el caso particular de Colombia, el país es miembro titular de ISO mediante la participación del Instituto Colombiano de Normas Técnicas – ICONTEC.

ISO trabaja en la elaboración de normas para diferentes sectores entre los que se encuentra Tecnologías de ingeniería como también Electrónica, tecnologías de la información y las telecomunicaciones, ambos sectores relacionados con la calidad de software, y para los cuales se han elaborado una importante cifra de estándares internacionales.

Para este caso particular, se requiere llevar a cabo la evaluación de los atributos de la calidad del software, y el estándar ISO/IEC 14598 coincide con esta necesidad puesto que es un modelo de evaluación de calidad del software. Asimismo, de acuerdo a Vivanco “Para realizar el proceso de evaluación es necesario primeramente definir el modelo de

⁵⁵ COVELLA, Guillermo Juan. Medición y evaluación de calidad en uso de aplicaciones web. Magíster en Ingeniería de Software. La Plata. Universidad Nacional de La Plata. 2005. 135 p.

⁵⁶ INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. Acerca de ISO, Nuestra historia. [En línea]. [Citado el 15 de julio de 2013] Disponible en internet: <http://www.iso.org/iso/home/about.htm>

calidad de software a utilizar...”⁵⁷ y en concordancia con el uso de estándares ISO, esta organización provee la norma ISO/IEC 9126, un modelo de calidad del software vinculado al antes mencionado, el estándar ISO/IEC 14598.

De acuerdo a Álvarez⁵⁸, los modelos de calidad más difundidos para especificar los atributos de calidad del software y las métricas, son: Boehm, McCall, ISO 9126, Paradigma GQM, Modelo de Gilb, CMM, Modelo FURSP, TMap y Modelo de Dromey.

En cuanto al hecho de optar por el estándar ISO/IEC 9126 es claro indicar que de acuerdo al cuadro comparativo de la figura 12, es el único modelo que ofrece métricas para cada uno de los atributos. Adicionalmente, cabe destacar el logro de este estándar al separar la calidad interna de la externa y la calidad en uso. Además, ISO/IEC 9126 es el modelo que apareció más recientemente, según lo indicado por el mismo autor.

Figura 12. Comparativa de los modelos más utilizados de calidad de software.

Modelo	Componentes	Niveles	Elementos de cada nivel	Métricas definidas?
McCall	* Factores * Criterios * Métricas	4: Subdivisión principal, Factores, Criterios, Métricas	* 14 Factores (3 en subdivisión principal + 11 en el segundo nivel) * 23 Criterios * Métricas no indicadas (definibles por el usuario)	NO
Boehm	* Características de alto nivel * Características Primitivas * Métricas	4: 2 capas de Características de alto nivel, 1 nivel de Primitivas, 1 nivel de métricas.	* 9 Características de Alto nivel * 15 Primitivas * Métricas no indicadas (definibles por el usuario)	NO
FURPS	* Atributos de calidad * Métricas	3: 2 niveles de Atributos de Calidad, 1 nivel de métricas	* 5 Atributos de calidad, primer nivel * 27 Atributos de calidad, segundo nivel * Métricas no indicadas (definibles por el usuario)	NO
GQM	* Objetivos * Cuestiones * Métricas	3: Goals, Questions, Metrics	* Objetivos: no indicados (definibles por el usuario) * Cuestiones: no indicados (definibles por el usuario) * Métricas: no indicadas (definibles por el usuario)	NO
Gilb	* Atributos de Alto nivel * Atributos de Bajo nivel * Métricas	No indicados (definibles por el usuario)	* Atributos de alto nivel: no indicados (definibles por el usuario) * Atributos de bajo nivel: no indicados (definibles por el usuario) * Métricas: no indicadas (definibles por el usuario)	NO
ISO/IEC 9126	* Características * Subcaracterísticas * Atributos de calidad * Métricas	* 2 niveles superiores (características y subcaracterísticas) * Niveles de atributos: no indicados (definibles por el usuario) * Niveles de métricas: asociados con cada nivel de atributos	* 6 Características * 27 Subcaracterísticas * Atributos de calidad: no indicados (definibles por el usuario) * Métricas: no indicadas (definibles por el usuario) pero se exponen las métricas estándar.	YES
IEEE 1061	* Factores * Subfactores * Métricas	No fijados (definibles por el usuario)	* Factores: no fijados (definibles por el usuario) * Subfactores: no fijados (definibles por el usuario) * Métricas: no fijados (definibles por el usuario)	NO
Dromey	* Atributos de alto nivel * Propiedades de calidad * Componentes	* 2 niveles superiores (Atributos de alto nivel y Propiedades) * Nivel de componentes: no indicados	* Atributos de alto nivel: no indicados (definibles por el usuario) * Propiedades de calidad: no indicadas (definibles por el usuario) * Componentes: no indicados (definibles por el usuario)	NO

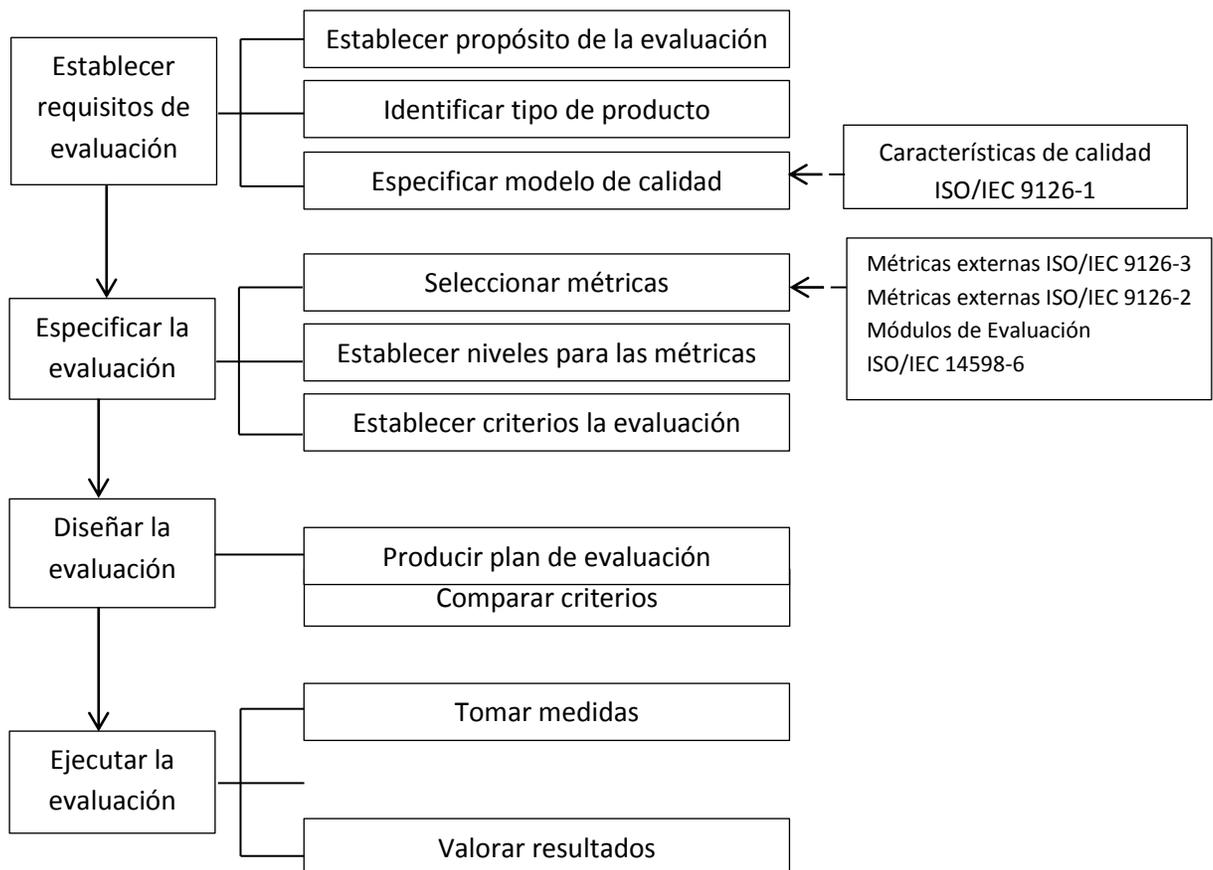
Fuente: ÁLVAREZ, Antonio. [En línea]. [Citado el 17 de junio de 2013]. Disponible en internet: <http://www.es.sogeti.com/PageFiles/173/SEPT-ANTONIO-ALVAREZ-SOGETI.pdf>

⁵⁷ VIVANCO, Andrés. Evaluación de la calidad del sistema integrado para casas de valores SICAV de la Bolsa de Valores de Quito utilizando la norma ISO/IEC 14598. [En línea]. [Citado el 8 de marzo de 2013] Disponible en internet: <http://repositorio.espe.edu.ec/handle/21000/5575>

⁵⁸ ÁLVAREZ, Op. cit., p. 8.

Teniendo en cuenta lo anterior, se planteó el proceso de evaluación de los atributos de calidad del software basados en la norma ISO/IEC 14598 como se muestra en la figura 13.

Figura 13. Proceso de Evaluación.



Fuente: ISO. International Standard ISO/IEC 14598-1. Ginebra : ISO/IEC, 1999. p. 9.

El proceso de evaluación descrito en la norma ISO/IEC 14598-1 puede ser ejecutado por un desarrollador, un adquisidor o un evaluador independiente, de acuerdo a esto, se eligió uno de estos roles:

Para este caso de estudio, es pertinente decir que los diseñadores y ejecutores de la evaluación de calidad son los mismos desarrolladores de la aplicación móvil, sin embargo, teniendo en cuenta que medir los productos intermedios desarrollados durante el ciclo de vida no es tenido en cuenta en este trabajo, no es aplicable el proceso para desarrolladores descrito en ISO/IEC 14598-3.

El proceso para adquirientes expuesto en ISO/IEC 14598-4 tampoco es aplicable para este caso, puesto que precisamente está destinado a aquellas organizaciones que tienen previsto adquirir un producto de software, lo cual no corresponde a este caso de estudio.

De otra parte, el proceso para evaluadores es el que se ajusta al propósito de este trabajo porque se prevé una evaluación que debe ser objetiva y además sobre el producto software terminado; por tanto, se hizo uso del estándar ISO/IEC 14598-5. Aunque en este caso particular la figura de evaluador independiente no corresponde con exactitud, porque como se ha dicho antes, los evaluadores de la calidad del producto software son los desarrolladores del mismo.

De esta manera, el estándar ISO/IEC 14598-5⁵⁹ describe en su numeral 6.4.1 que el proceso de evaluación comprende cinco actividades: establecimiento de los requerimientos de la evaluación, especificación de la evaluación, diseño de la evaluación, ejecución del plan de evaluación, y conclusión de la evaluación.

En este orden de ideas se puede establecer que el estándar ISO/IEC 14598-5 estructura el proceso de evaluación. A continuación se expone dicho proceso:

4.1.1.1 Establecimiento de los requisitos de la evaluación. El propósito de esta etapa es definir los objetivos de la evaluación. Esto se realiza a partir de los requerimientos del solicitante, constituidos por una descripción a gran escala de lo que el cliente quiere evaluar. Luego con el evaluador se realiza una definición del cubrimiento de la evaluación, donde se especifica qué es lo que se quiere evaluar y qué no.

En este caso no existe un solicitante que plantee los requerimientos; estos fueron definidos teniendo en cuenta los objetivos de la confrontación de tecnologías de desarrollo de aplicaciones móviles planteada, y con base a la norma ISO/IEC 9126-1. De esta manera, ahora se desglosa la explicación donde se indica los requerimientos que se definieron para la evaluación.

De acuerdo al estándar ISO/IEC 14598 es necesario especificar el modelo de calidad a emplear, que para este caso se escogió el propuesto por el estándar ISO/IEC 9126. Este modelo de calidad, establecido en la primera parte de dicho estándar (ISO/IEC 9126-1), estructura la calidad del software dividiéndola en tres partes: Calidad interna, Calidad externa y Calidad en uso. La calidad interna y externa son abordadas especificando seis características: funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad; mientras que la calidad en uso comprende cuatro características: efectividad, productividad, seguridad y satisfacción.

Para realizar la evaluación de calidad no se tuvo en cuenta la totalidad de características expuestas en el modelo de calidad de ISO/IEC 9126, de acuerdo a dos razones que se explican a continuación:

⁵⁹ INTERNATIONAL ORGANIZATION FOR STANDARDIZATION ISO/IEC 14598-5. Information technology - Software product evaluation. Primera edición. Ginebra : 1999. p. 9.

- **No es posible optimizar todos los atributos para un sistema.** Sommerville⁶⁰ dice: “Existe una amplia variedad de atributos de calidad del software potenciales a considerar en el proceso de planificación de la calidad. En general, no es posible optimizar todos los atributos para un sistema.”

Es importante enfatizar que Sommerville expresa que no todos los atributos de la calidad del software se pueden optimizar en un sistema, entendiéndose que el fin no es sencillamente alcanzar todos los atributos para llegar a un producto de calidad.

A propósito de lo anterior, Álvarez⁶¹ dice que generalmente es difícil identificar aquellos atributos de calidad del software para evaluar de forma correcta un sistema de software.

- **Considerar los atributos más importantes de acuerdo al sistema a evaluar.** Somerville⁶² expresa: “El conjunto específico de atributos que se espera de un sistema de software depende obviamente de su aplicación. Por lo tanto, un sistema bancario debe ser seguro, un juego interactivo debe tener capacidad de respuesta, un interruptor de un sistema telefónico debe ser Fiable, etcétera.” Además, Sommerville se refiere a la trascendencia de un atributo de calidad que resulta más relevante que otro en un software determinado, así:

El plan de calidad define los atributos de calidad más importantes para el producto a desarrollar. Puede ser que la eficacia sea primordial, por lo que será necesario sacrificar otros factores para alcanzarla. Esto se establece en el plan, y los ingenieros que trabajan en el desarrollo deben cooperar para lograrlo. El plan también define el proceso de evaluación de la calidad. Es una forma estándar de valorar si algún atributo de calidad, como la mantenibilidad o la robustez, está presente en el producto.⁶³

Por lo anterior, se han de considerar las características más significativas de acuerdo al tipo de software y al caso particular en que se ha de aplicar el modelo de calidad. Es importante abordar las características que vienen a ser más relevantes por tratarse de aplicaciones para dispositivos móviles. En este orden de ideas se presenta a continuación una serie de enfoques por varios autores para argumentar por qué se seleccionaron las características de calidad que hicieron parte de la evaluación.

En primer lugar, dentro del mundo móvil, El Cuera⁶⁴ considera que desde el inicio de la construcción o desarrollo es necesario probar las aplicaciones móviles teniendo en cuenta, precisamente, su propio contexto de movilidad y considerando el cumplimiento de los

⁶⁰ SOMMERVILLE. Op. cit. p.597.

⁶¹ ÁLVAREZ. Op. cit., p. 8.

⁶² SOMMERVILLE. Op. cit., p. 11.

⁶³ Ibid., p. 597.

⁶⁴ EL CUERA, Mamdouh. Pruebas de calidad en aplicaciones móviles [En línea]. [Citado el 3 de agosto de 2013]. Disponible en internet: <http://www.mtp.es/noticias/265-pruebas-calidad-aplicaciones-moviles/>

atributos de calidad. En concreto, se refiere a cuatro características de suma importancia para las aplicaciones móviles: funcionalidad, usabilidad, seguridad y rendimiento.

Ahora, como dice Vivanco⁶⁵: El usuario debe definir las características y subcaracterísticas a tener en cuenta en el proceso de evaluación de calidad. De acuerdo a lo anterior, el proceso de definición de la evaluación de calidad permite determinar las características y por ende, las métricas más relevantes de acuerdo al caso.

De otra parte, Google⁶⁶ a través de las mejores prácticas para el desarrollo de aplicaciones móviles Android, señala el énfasis que éstas hacen a algunos aspectos fundamentales propios de este ámbito. Las mejores prácticas son: Experiencia de usuario - Interfaz de usuario, entrada de usuario, rendimiento y también seguridad & privacidad.

De acuerdo a estas “Best Practices” (en inglés), Google apunta principalmente a tres características de calidad. Teniendo en cuenta la clasificación de características que presenta el modelo de calidad de ISO/IEC 9126, se logra reconocer que la usabilidad corresponde en las mejores prácticas a lo que es experiencia de usuario – interfaz de usuario y entrada de usuario. Además, las mejores prácticas para rendimiento coinciden con la característica Eficiencia en el modelo de calidad, y finalmente, respecto a seguridad & privacidad hay correspondencia con la subcaracterística Seguridad que pertenece a la Funcionalidad en el estándar ISO/IEC 9126. Así, las mejores prácticas de parte de Google guardan relación con las características de calidad presentes en ISO/IEC 9126.

Continuando con las mejores prácticas entregadas por Google acerca del rendimiento, éstas se ocupan de lograr un “diseño suave y sensible”, además, están enfocadas en la reducción del consumo de batería, que depende en gran medida de cómo se usan los recursos hardware del dispositivo, apartado que corresponde a la característica ‘eficiencia’ en el estándar ISO/IEC 9126. Asimismo, se menciona en estas buenas prácticas de rendimiento lo relacionado con la respuesta de la interfaz a la interacción con el usuario y evitar bloqueos, lo que notoriamente pertenece a la subcaracterística de calidad ‘comportamiento temporal’.

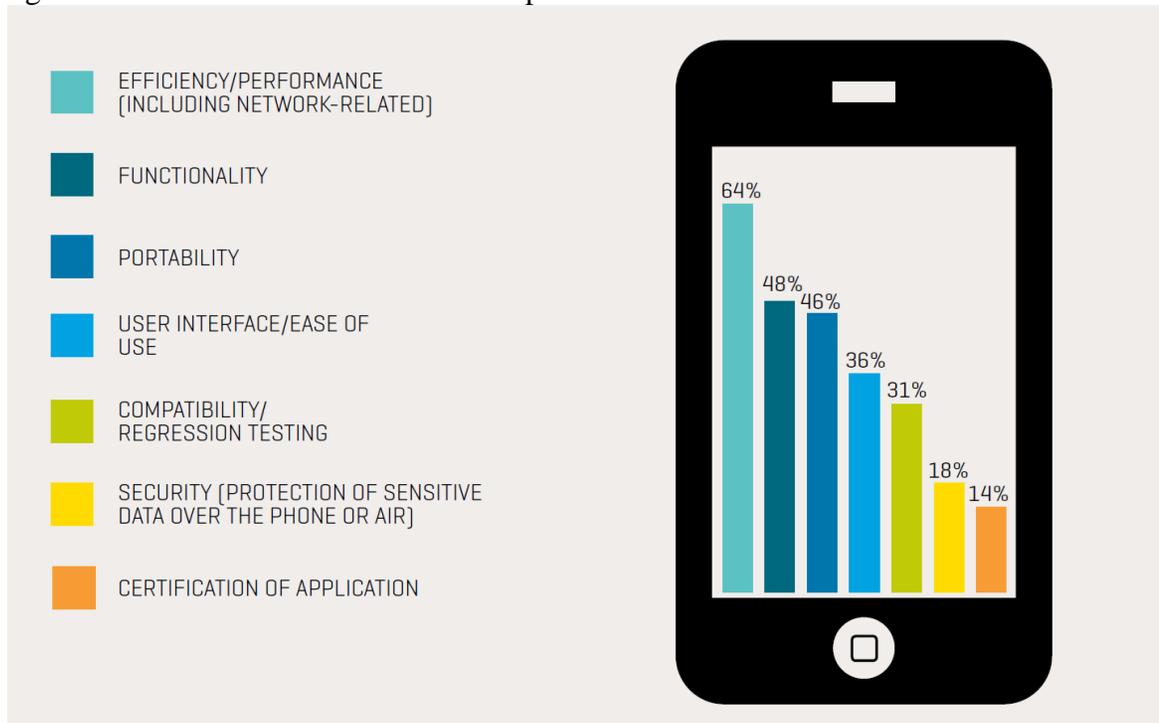
- **Escogencia de la eficiencia (rendimiento) como única característica de calidad evaluada.** Aunque se viene exponiendo que las características de calidad indispensables en las aplicaciones móviles son: funcionalidad, seguridad, usabilidad y rendimiento, en el contexto de las aplicaciones la eficiencia o también llamada rendimiento (como se aclara más adelante), se destaca notablemente y por tanto la única característica de calidad que se escogió en este estudio.

⁶⁵ VIVANCO. Op. cit., p. 21.

⁶⁶ GOOGLE DEVELOPER ANDROID. Training [En línea]. [Citado el 5 de agosto de 2013]. Disponible en internet: <http://developer.android.com/training/best-performance.html>

Según el Work Quality Report⁶⁷ 2012-2013, el rendimiento es la cualidad más buscada. En este informe elaborado por Capgemini, Sogeti y HP, un 64% de los encuestados afirma que el rendimiento es la cualidad más importante por encima de funcionalidad (48%), portabilidad (46%) e interfaz de usuario (36%). Lo anteriormente mencionado, se puede apreciar en la figura 14.

Figura 14. Calidad más buscada en las aplicaciones móviles.



Fuente: CAPGEMINI, SOGETI y HEWLETT-PACKARD. [En línea]. [Citado el 1 de agosto de 2013]. Disponible en internet: http://www.capgemini.com/sites/default/files/resource/pdf/World_Quality_Report_2012_-_2013.pdf

De acuerdo a un estudio realizado por Apigee⁶⁸, el 96% de los norteamericanos encuestados, usuarios de aplicaciones móviles, dicen que hay frustraciones que los llevarían a dar una mala crítica a éstas, entre las que se encuentran: Aplicaciones congeladas, capacidad de respuesta lenta, demasiado uso de la batería, cantidad exagerada de publicidad.

⁶⁷ CAPGEMINI, SOGETI y HEWLETT-PACKARD. World Quality Report 2012-2013. Cuarta Edición. [En línea]. [Citado el 1 de agosto de 2013]. Disponible en internet: http://www.capgemini.com/sites/default/files/resource/pdf/World_Quality_Report_2012_-_2013.pdf

⁶⁸ APIGEE. Op. cit.

El 98% de los encuestados hizo hincapié en las cuestiones de rendimiento. Cuando las aplicaciones no funcionan como se esperaba o como se ha anunciado, el 99% de los encuestados dijeron que tomarían medidas como: eliminar la aplicación inmediatamente el 44%, eliminar la aplicación si se congela durante más de 30 segundos el 38%, comentar a mis amigos y colegas lo mala que es el 32% y quejarse de la aplicación en Facebook y Twitter el 21%, como se muestra en la figura 15.

Figura 15. Reacciones de los usuarios ante problemas en el rendimiento de una aplicación móvil



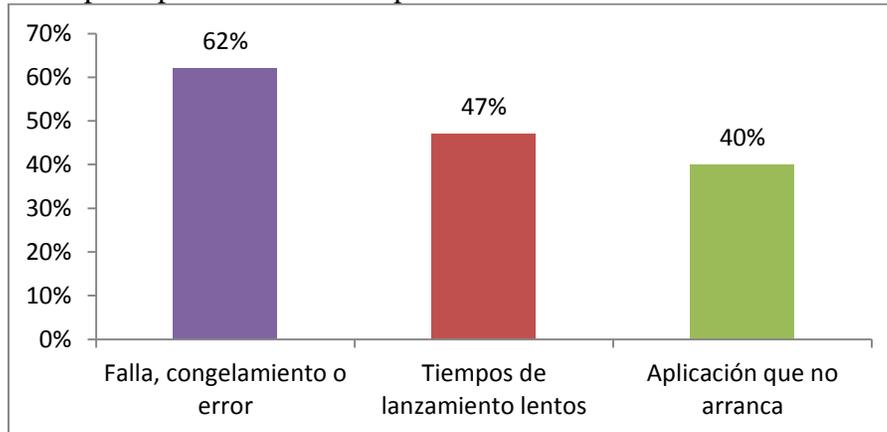
Fuente: Apigee. [En línea]. [Citado el 12 de enero de 2013]. Disponible en internet: https://blog.apigee.com/detail/users_reveal_top_frustrations_that_lead_to_bad_mobile_app_reviews_infographic

Y un implacable 18% admitió que elimina una aplicación inmediatamente si su dispositivo se congela por sólo 5 segundos. Si bien la mayoría dice que tomarían medidas, el 27% de los encuestados informó que iba a mantener la aplicación un poco más si había pagado por ella.

De acuerdo a Mobile Apps: What Consumers Really Need and Want⁶⁹. Los resultados de este estudio arrojaron que más de la mitad de los usuarios han experimentado problemas relacionados con el rendimiento de una aplicación móvil; entre los principales problemas se encuentran: el 62% reportan una falla, congelamiento o error, el 47% experimentó tiempos de lanzamiento lentos y el 40% ha probado una aplicación que simplemente no ha arrancado, como se muestra en la figura 16.

⁶⁹ COMPUWARE. Mobile Apps: What Consumers Really Need and Want. [En línea]. [Citado el 30 de agosto de 2013]. Disponible en internet: http://offers2.compuware.com/rs/compuware/images/Mobile_App_Survey_Report.pdf

Figura 16. Principales problemas de las aplicaciones móviles



Fuente: Autores del proyecto

De la misma forma, el estudio revela que el 79% de los usuarios intenta usar de nuevo la aplicación móvil una o dos veces más sino no funcionó la primera vez. Aparte de la insatisfacción de los usuarios con el rendimiento, el 48% posiblemente no usarían de nuevo la aplicación.

Según el estudio de SOASTA⁷⁰. Esta encuesta revela que casi nueve de cada diez estadounidenses (88%) experimentan sentimientos negativos frente a marcas que tienen aplicaciones móviles o sitios web con un bajo rendimiento. En este sentido, los encuestados manifiestan presentar sentimientos negativos, principalmente: Molestia (75%) y frustración (69%). Lo anterior permite inferir que los usuarios de aplicaciones móviles están muy atentos del rendimiento que éstas tienen. También, este mismo estudio encontró que para los encuestados, dentro de las cualidades más importantes en un sitio web o aplicación móvil, se destacan los productos rápidos y sin tiempo de espera (62%), así como los de rendimiento esperado (62%), con lo cual se ratifica el interés de los usuarios en el rendimiento de las aplicaciones móviles.

De acuerdo a los estudios anteriormente referenciados es posible notar que los usuarios manifiestan inconvenientes de las aplicaciones móviles relacionados con el rendimiento, lo que en el modelo de calidad de ISO/IEC 9126 corresponde a la característica de calidad 'Eficiencia'.

En definitiva, es posible identificar que el rendimiento de las aplicaciones móviles es un aspecto clave en la actualidad. En lo que respecta al Informe World Quality Report, es necesario señalar que éste indica cómo tradicionalmente se hacían grandes esfuerzos por lograr funcionalidad, sin embargo, hoy en día el rendimiento de las aplicaciones es lo que más interesa, tanto así que habla del rendimiento como "la clave del éxito"⁷¹. De la misma

⁷⁰ SOASTA. 88% of Americans Have Negative Feelings about Brands with Poorly Performing Websites and Mobile Apps. [En línea]. [Citado el 31 de agosto de 2013]. Disponible en internet: <http://www.soasta.com/press-releases/467413/>

⁷¹ CAPGEMINI, SOGETI y HEWLETT-PACKARD. Op. Cit., p. 25.

forma, los estudios realizados por: Apigee, SOASTA y Compuware ponen de manifiesto la importancia de esta característica por encima de otras, puesto que los usuarios de las aplicaciones móviles exponen un inconformismo muy significativo en aspectos asociados con el rendimiento de ellas.

Finalmente, el mismo Stephen Pierzchala, expresa: "El rendimiento es un factor crucial para proporcionar una experiencia de usuario de aplicaciones móviles de confianza, por lo que el rendimiento se debe considerar un factor clave en el proceso de diseño."⁷²

De otra parte, teniendo en cuenta que el término 'rendimiento' es el empleado tanto por El Cuera, las mejores prácticas (Best Practices) de Google, como también en los estudios referenciados (World Quality Report, Apigee, SOASTA y Compuware), y entendiendo que el estándar ISO/IEC 9126 establece 'Eficiencia' como una característica de calidad que se desglosa en: Comportamiento temporal (tiempos de respuesta) y Uso de recursos, se aclara que se toma como equivalente el uso de la palabra 'rendimiento' en lugar de 'eficiencia'.

En este orden de ideas, también se agrega lo expresado por Yorio, que hablando de los patrones de uso en el diseño de aplicaciones móviles, resalta uno de los aspectos fundamentales asociados con el rendimiento de éstas; se trata del tiempo de inicio de la aplicación que tiene correspondencia con la subcaracterística 'Comportamiento temporal' en el estándar ISO/IEC 9126.

Un tiempo de inicio corto es una característica importante para las aplicaciones móviles. Dado que los usuarios tienden a utilizar frecuentemente sus dispositivos móviles en sesiones de corta duración, se hace imperativo que la aplicación se encuentre lista para ser empleada en el menor tiempo posible. Puede ser algo incómodo el hecho de esperar que un procesador de texto o cualquier otra aplicación se tome algunos segundos para iniciar en un computadora de escritorio, pero comparado con el tiempo que se estará utilizando la misma, estos segundos de inicio no son significativos. Para una aplicación móvil donde el usuario quiere usarla por 20 segundos para verificar o actualizar una pieza de información, esperar un inicio de 6 segundos puede ser un tiempo completamente inaceptable. Por lo que podría concluir que una buena práctica es hacer que el tiempo de inicio de la aplicación sea despreciable frente al tiempo que tomará una sesión de uso. Las aplicaciones móviles tienen una sesión de uso corta por lo que proporcionalmente se requiere un tiempo de inicio corto.⁷³

⁷² COMPUWARE. Op. cit.

⁷³ YORIO, Darío. Identificación y Clasificación de Patrones en el Diseño de Aplicaciones Móviles. [En línea]. [Citado el 9 de agosto de 2013]. Disponible en internet: http://postgrado.info.unlp.edu.ar/Carreras/Magisters/Ingenieria_de_Software/Tesis/Yorio_Ruben.pdf

Asimismo, en concordancia a la literatura presentada en la sección de 2.3 Marco Teórico, es posible obtener conclusiones, una de ellas con respecto a las desventajas que tienen las aplicaciones móviles desarrolladas con base a tecnologías web, precisamente con respecto al menor rendimiento que éstas tienen en comparación con las aplicaciones del tipo nativo. Tal consideración es significativa porque indica la postura teórica existente en lo que se refiere a la confrontación entre las dos tecnologías de desarrollo de aplicaciones móviles.

Finalmente, El Cuera, señala: “En cuanto a las pruebas de rendimiento, tienen como objetivo ayudar a la empresa a prevenir errores que eviten el abandono de la aplicación debido a caídas o a un tiempo excesivo en su ejecución. Un alto consumo de batería, la velocidad a la que se desarrolla dependiendo del navegador utilizado o las características de cada sistema operativo, pueden ser factores que influyen de forma decisiva en este apartado”⁷⁴. En relación a lo antes expuesto, particularmente lo que se menciona como ‘tiempo excesivo en su ejecución’ está asociado con la subcaracterística de calidad ‘Comportamiento temporal’ en lo que concierne a eficiencia en ISO/IEC 9126.

De acuerdo a lo explicado, dentro del contexto de movilidad con el que se determinó el rendimiento como la única característica de calidad a evaluar, los requerimientos de la evaluación corresponden solamente a los atributos de calidad relacionados con tal característica. En este orden de ideas, los requerimientos no funcionales son los siguientes:

- a) Verificar el tiempo de respuesta ante una solicitud por parte del usuario.
- b) Comprobar el tiempo de espera que el usuario experimenta.
- c) Verificar la utilización de recursos de memoria RAM en el cumplimiento de una función.
- d) Verificar la utilización de CPU en el cumplimiento de una función.
- e) Inspeccionar el comportamiento del consumo de batería.

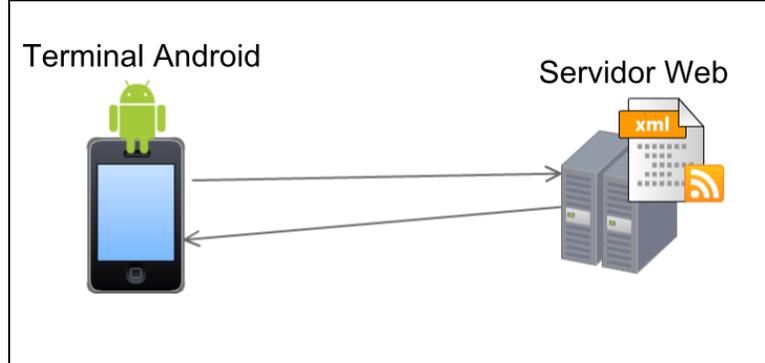
- **Descripción del producto.** En esta sección se plantea la descripción del software a evaluar; cabe mencionar que la aplicación fue construida con el objetivo de realizar un caso de estudio para comparar las tecnologías de desarrollo nativa y web desde la perspectiva de los atributos de calidad.

NewsApp. Aplicación móvil para ser instalada en smartphone Android con sistema operativo 4.0 Ice Cream Sandwich, preparada para ser ejecutada sin inconvenientes en una pantalla de resolución 1280 x 720 píxeles.

La aplicación es un sencillo lector o recopilador de contenidos (también conocido como agregador de noticias o lector de RSS). En este caso dicho lector presenta cinco fuentes de las cuales se puede tener acceso a las últimas noticias, obteniéndose el título de cada una junto con una breve descripción y el enlace para ver la noticia completa en el sitio web de la misma. En la figura 17 se aprecia la arquitectura de la aplicación.

⁷⁴ EL CUERA, Op. cit.

Figura 17. Arquitectura del lector RSS



Fuente: Autores del proyecto

4.1.1.2 Especificación de la evaluación. Una vez que se tienen los requerimientos de evaluación, los cuales fueron obtenidos en la etapa anterior, es preciso pasar a la próxima etapa, esta es la especificación de la evaluación. Esta etapa tiene como propósito establecer las mediciones a ser tomadas sobre los atributos de calidad que fueron definidos en la etapa anterior.

Para este caso de estudio en el que se tiene como finalidad confrontar dos tecnologías de desarrollo de aplicaciones móviles y teniendo en cuenta que el modelo de calidad establecido en el estándar ISO/IEC 9126-1, presenta tres vistas de calidad: interna, externa y calidad en uso, y debido a que es necesario efectuar el proceso de evaluación de calidad una vez el producto software esté terminado, se seleccionaron las métricas de la vista de calidad externa, que de acuerdo a la definición en el estándar ISO/IEC 9126: “Es la calidad cuando el software es ejecutado, la cual es típicamente medida y evaluada mientras se prueba en un ambiente simulado con datos simulados y usando métricas externas.”⁷⁵

El estándar ISO/IEC 9126⁷⁶ propone un modelo de calidad con una visión general aplicable a todo tipo de software. Por lo tanto, deja abierta la posibilidad de modificar y hacer consideraciones sobre las métricas definidas con el propósito de adaptarse al contexto al que pertenece el producto software a evaluar; además, es posible usar métricas que no están presentes en el estándar. La norma ISO/IEC 9126-2 expresa: “Los usuarios de este Informe Técnico Internacional pueden seleccionar o modificar y aplicar indicadores y medidas de este Informe Técnico Internacional o pueden definir parámetros específicos de la aplicación para su dominio de aplicación individual”⁷⁷. Por lo cual es claro que para este caso que se ocupa de las aplicaciones móviles, que además tienen particularidades que las convierten en

⁷⁵ INTERNATIONAL ORGANIZATION FOR STANDARDIZATION ISO/IEC TR 9126-2. Ginebra : 2002. p. 4.

⁷⁶ INTERNATIONAL ORGANIZATION FOR STANDARDIZATION ISO/IEC TR 9126-2. Ginebra : 2002. p. 4.

⁷⁷ Ibid., p. 9.

un tipo especial de software, se requiere tener en cuenta algunos cambios que se ajusten al contexto móvil.

Para este trabajo, donde el software es una aplicación para dispositivos móviles, de acuerdo a ese contexto de movilidad es preciso enfatizar en aspectos que proponen retos en el diseño de dichas aplicaciones.

- **El consumo de batería.** Según Yorio⁷⁸ el consumo de batería viene a ser muy importante puesto que obviamente la energía es limitada y requiere de administración apropiada, por lo anterior, el consumo de batería de una aplicación se considera fundamental para confrontar las dos tecnologías de desarrollo de aplicaciones móviles. Además, como se dijo antes, El Cuera⁷⁹ recalca que uno de los errores que debe evitarse es el alto consumo de batería por parte de una aplicación móvil. De la misma forma, se mencionó también que en la encuesta de Apigee, el 55% de los usuarios de aplicaciones móviles manifiestan tener problemas por el alto consumo de batería; finalmente, el estudio de Compuware señala que “...la satisfacción con los smartphones se ve afectada en gran medida por el rendimiento de la batería, en particular la duración de la vida de la batería antes de que se requiera recarga. Además, el estudio revela que la satisfacción con el rendimiento de la batería es, con mucho, el aspecto menos satisfactorio de los teléfonos inteligentes”⁸⁰.

Por todo lo anterior, se añadieron métricas para inspeccionar el comportamiento de consumo energía.

- **Consumo de CPU.** En este orden de ideas, también se requiere la consideración de elementos como CPU puesto que, en general, la capacidad de procesamiento de los dispositivos móviles en la actualidad es limitada aunque existen terminales gama alta que poseen características de CPU avanzadas. De acuerdo a lo anterior resulta altamente importante analizar el consumo que la aplicación hace de este recurso.

Así las cosas, las métricas de calidad que se abordaron son las que se indican en el cuadro 11 que muestra la característica eficiencia dividida en subcaracterísticas y las correspondientes métricas.

⁷⁸ YORIO. Op. cit., p.

⁷⁹ EL CUERA, Op. cit.

⁸⁰ J.D. Power and Associates Reports: Smartphone Battery Life has Become a Significant Drain on Customer Satisfaction and Loyalty. [En línea]. [Citado el 2 de diciembre de 2013]. Disponible en internet en: <http://www.prnewswire.com/news-releases/jd-power-and-associates-reports-smartphone-battery-life-has-become-a-significant-drain-on-customer-satisfaction-and-loyalty-142765065.html>

Cuadro 11. Desglose de la característica de calidad eficiencia

CARACTERÍSTICA	SUBCARACTERÍSTICA	MÉTRICA
Eficiencia	Comportamiento temporal	Tiempo de respuesta
		Tiempo de respuesta medio
		Tiempo de respuesta en el peor caso
	Utilización de recursos	Utilización máxima de memoria
		Utilización máxima de CPU
		Utilización máxima de batería

Fuente: Autores del proyecto

A continuación se presenta la especificación completa de cada una de las métricas mencionadas en el cuadro 11, en las tablas 1, 2, 3, y 4 que están compuestas por las siguientes columnas:

Nombre Métrica: nombre que identifica a la métrica.

Propósito de la métrica: Esta se expresa como la pregunta que debe responderse mediante la aplicación de la métrica.

Método de aplicación: Proporciona una descripción para llevar acabo la medición.

Medición, fórmulas y cálculos de elementos de datos: Proporciona la fórmula de medición y explica el significado de los elementos de datos utilizados.

NOTA: En algunas situaciones, se propone más de una fórmula para una métrica.

Interpretación del valor medido: proporciona el rango y los valores preferidos.

Tabla 1. Métricas de comportamiento temporal: Tiempo de respuesta

Nombre Métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculos de elementos de datos	Interpretación del valor medido
Tiempo de respuesta	¿Cuál es el tiempo necesario para completar una tarea específica?	Iniciar una tarea específica. Se mide el tiempo que tarda la muestra para completar la operación. Mantenga un registro de cada intento.	$T =$ (tiempo de obtener el resultado) - (tiempo de terminada la entrada de comandos)	$0 < T$ Cuanto más pronto es el mejor.
Tiempo de respuesta (el tiempo medio de respuesta)	¿Cuál es el tiempo promedio de espera que el usuario experimenta después de emitir una solicitud hasta que la solicitud es completada dentro de un sistema de carga especificada?	Ejecutar una serie de escenarios de tareas. Medir el tiempo que tarda en completar la operación seleccionada (s). Mantenga un registro de cada intento y calcule el tiempo medio para cada escenario.	$T = T_{\text{medio}}$ $T_{\text{medio}} = \sum (T_i) / N$, (para $i = 1$ a N) $T_i =$ tiempo de respuesta a una tarea	$0 < T$ Cuanto más pronto es el mejor.
Tiempo de respuesta (peor caso de tiempo de respuesta promedio)	¿Cuál es el límite absoluto de tiempo necesario en el cumplimiento de una función? En el peor de los casos, ¿el usuario todavía puede obtener respuesta en el plazo especificado? En el peor de los casos, ¿El usuario puede todavía obtener respuesta del software en un tiempo suficientemente corto como para ser tolerable para el usuario?	Calibrar la prueba. Emular una condición por la cual el sistema alcanza una situación de carga máxima. Ejecutar la aplicación y observar los resultados (s)	$T = T_{\text{max}}$ $T_{\text{max}} = \text{MAX} (T_i)$ (para $i = 1$ a N) $\text{MAX} =$ tiempo (T_i) de respuesta máximo entre las muestras $N =$ número de evaluaciones (tomas de muestra) $T_i =$ tiempo de respuesta para la i -ésima evaluación (muestra) En una situación de carga máxima.	$0 < T$ Cuanto más pronto es el mejor.

Fuente: ISO/IEC 9126-2. Ginebra : ISO/IEC, 2002.

Tabla 2. Métricas de utilización de Recursos: Utilización de recursos de memoria

Nombre Métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculos de elementos de datos	Interpretación del valor medido
Utilización máxima de memoria	¿Cuál es el límite absoluto de memoria necesaria en el cumplimiento de una función?	Iniciar una tarea específica. Se mide la memoria consumida para completar la operación. Mantenga un registro de cada intento.	$M =$ Cantidad de memoria consumida	$0 \leq M$ Cuanto más pequeño es el mejor.
Utilización máxima de memoria el consumo medio de memoria)	¿Cuál es el consumo promedio de memoria en la ejecución de una tarea?	Calibrar las condiciones de ensayo. Mantenga un registro de cada intento.	$M = M_{\text{medio}}$ $M_{\text{medio}} = \sum (Ti) / N$, (para $i = 1$ a N) $M_i =$ memoria consumida por una tarea	$0 \leq M$ Cuanto más pequeño es el mejor.
Utilización de memoria (peor caso de consumo promedio de memoria)	¿Cuál es el consumo promedio de memoria en una situación de carga máxima?	Calibrar las condiciones de ensayo. Mantenga un registro de cada intento.	$M = M_{\text{max}}$ $M_{\text{max}} = \text{MAX} (M_i)$ (para $i = 1$ a N) $\text{MAX} =$ memoria (M_i) máxima consumida entre las muestras $N =$ número de evaluaciones (tomas de muestra) $M_i =$ memoria consumida para la i -ésima evaluación (muestra) En una situación de carga máxima.	$0 \leq M$ Cuanto más pequeño es el mejor.

Fuente: ISO/IEC 9126-2. Ginebra : ISO/IEC, 2002.

Tabla 3. Métricas de utilización de Recursos: Utilización de recursos de CPU

Nombre Métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculos de elementos de datos	Interpretación del valor medido
Utilización máxima de CPU	¿Cuál es el límite absoluto CPU necesaria en el cumplimiento de una función?	Iniciar una tarea específica. Se mide la CPU consumida para completar la operación. Mantenga un registro de cada intento.	C = Cantidad de CPU consumida	$0 \leq C$ Cuanto más pequeño es el mejor.
Utilización máxima de CPU (el consumo medio de CPU)	¿Cuál es el consumo promedio de CPU en la ejecución de una tarea?	Calibrar las condiciones de ensayo. Mantenga un registro de cada intento.	C = Cmedio $C_{medio} = \sum (C_i) / N$, (para $i = 1$ a N) C_i = CPU consumida por una tarea	$0 \leq C$ Cuanto más pequeño es el mejor.
Utilización de CPU (peor caso de consumo promedio de CPU)	¿Cuál es el consumo promedio de CPU en una situación de carga máxima?	Calibrar las condiciones de ensayo. Mantenga un registro de cada intento.	C = Cmax $C_{max} = \text{MAX} (C_i)$ (para $i = 1$ a N) $\text{MAX} = \text{CPU} (C_i)$ máxima consumida entre las muestras N = número de evaluaciones (tomas de muestra) C_i = CPU consumida para la i-ésima evaluación (muestra) En una situación de carga máxima.	$0 \leq C$ Cuanto más pequeño es el mejor.

Fuente: Autores del proyecto.

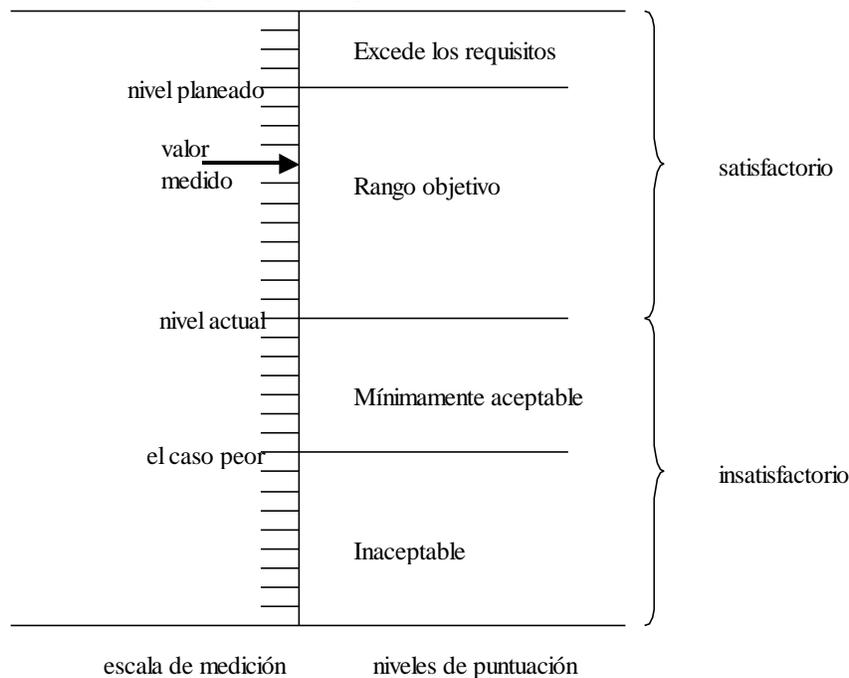
Tabla 4. Métricas de utilización de Recursos: Utilización de recursos de batería

Nombre Métrica	Propósito de la métrica	Método de aplicación	Medición, fórmula y cálculos de elementos de datos	Interpretación del valor medido
Utilización máxima de batería	¿Cuál es la cantidad de energía consumida en el cumplimiento de una función?	Iniciar una tarea específica. Se mide la energía consumida para completar la operación. Mantenga un registro de cada intento.	B= Cantidad de energía consumida	$0 \leq B$ Cuanto más pequeño es el mejor.
Utilización máxima de energía (el consumo medio de energía)	¿Cuál es el consumo promedio de energía en la ejecución de una tarea?	Calibrar las condiciones de ensayo.	$B = B_{medio}$ $B_{medio} = \sum (B_i) / N$, (para $i = 1$ a N) $B_i =$ Energía consumida durante la ejecución de la evaluación.	$0 \leq B$ Cuanto más pequeño es el mejor.

Fuente: Autores del proyecto.

Establecer niveles de puntuación para las métricas. En este punto la norma ISO/IEC 14598 expone que las características cualitativas pueden medirse cuantitativamente, usando métricas de calidad, de esta manera propone una escala que puede verse en la figura 18.

Figura 18. Niveles de puntuación para las métricas.



Fuente: ISO/IEC 14598. Ginebra : ISO/IEC, 1999.

En la figura 18 se muestra que esta escala no establece rangos cuantitativos por el hecho de ser un proceso general para la evaluación de un producto software; teniendo en cuenta esto para este proyecto no se usó dicha escala, sino que se llevó a cabo interpretando los resultados de las mediciones con el uso de los siguientes criterios:

Para las métricas del comportamiento temporal: $0 < T$, Cuanto más pronto es el mejor, donde $T =$ tiempo.

Para la utilización de recursos: $0 \leq R$, Cuanto más pequeño es el mejor, donde $R =$ recurso, que puede ser: memoria RAM, CPU, batería.

Establecer criterios para la valoración. Al ejecutar la evaluación sobre un producto software deben definirse ciertos criterios que se tendrán en cuenta durante dicho proceso. Conforme a las indicaciones de ISO/IEC 14598⁸¹ debe definirse un modelo de calidad para determinar los requerimientos de la evaluación, para esto se seleccionó el modelo de calidad formulado en el estándar ISO/IEC 9126, no obstante deja abierta la posibilidad de hacer uso de otro modelo de calidad si existe una razón en particular.

4.1.1.3 Diseño de la evaluación. Esta etapa consiste en especificar los métodos, procedimientos y herramientas necesarias en la evaluación.

Descripción de la herramienta de evaluación. Para la medición se seleccionó el software denominado Little Eye, una herramienta de monitoreo que puede analizar diversos tipos de datos relacionados con el rendimiento de una aplicación para dispositivos Android. Little Eye es desarrollado por la empresa que lleva el mismo nombre.

Little Eye permite medir el consumo de recursos de una aplicación determinada, obteniendo estadísticas sobre cantidad de energía, datos, CPU, memoria RAM y espacio en disco que la aplicación está utilizando. Se monitorea en tiempo de ejecución, y se presentan los resultados mediante gráficas que se actualizan mientras es realizado el monitoreo, puesto que Little Eye captura el video de la pantalla del teléfono, permitiendo observar las acciones que se llevan a cabo en el mismo. Una vez finalizada la evaluación del rendimiento, es posible acceder a la grabación del monitoreo y analizar cualquier punto de interés en la línea de tiempo. Todo el proceso anterior se realiza sin escribir una sola línea de código.

Finalmente, también cabe indicar que Little Eye ofrece la opción de generar un reporte detallado de la evaluación realizada que incluye gráficas y datos de importancia muy bien organizados.

A continuación se presenta la especificación de los detalles técnicos y algunas características de Little Eye en el cuadro 12.

⁸¹ INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. Op. cit., p. 16.

Cuadro 12. Requerimientos para la instalación de Little Eye

Última versión	2.2.1.0	
Licencia	Pro Trial License por seis meses	
Plataformas compatibles	Windows	XP, Vista (32 bit, 64 bit), 7 (32 bit, 64 bit), 8 (32 bit, 64 bit)
	Linux	Ubuntu 12.04 (32 bit, 64 bit) y podría funcionar en todas las demás plataformas de Linux.
	Mac OS X	Snow Leopard, Lion, Mountain Lion.
Versiones de Android soportadas	Android 2.3; 3.0; 3.1; 3.2; 4.0; 4.1; 4.2; 4.3	
Requisitos para la instalación y ejecución	<ul style="list-style-type: none"> ▪ JRE o JDK (Java 6 o superior) ▪ SDK de Android ▪ Habilitar la opción de depuración USB en el terminal Android ▪ Instalar el controlador de su teléfono si es necesario. 	
Sitio web	www.littleeye.co	

Fuente: Autores del proyecto.

Por último, esta herramienta de monitoreo es usada por empresas como Intel, Qualcomm, GE Healthcare, Capgemini, entre otras. Además, es importante decir que la compañía Little Eye⁸² en 2013 fue galardonada con el premio TiE50, la conferencia más grande del mundo para los empresarios.

Funcionamiento de Little Eye. Little Eye⁸³ es un software de escritorio que se descarga y se instala en el computador. Esta herramienta comprende dos componentes principales que se describen brevemente a continuación.

En primer lugar, hay un agente que se ejecuta en el dispositivo móvil Android, que corre al lado de la aplicación en análisis. Esta parte de Little Eye se ejecuta en un proceso independiente, sin afectar de ninguna manera la aplicación que está siendo monitoreada. La recolección de datos (una vez por segundo) es completamente externa, de modo que no se interviene en el código de la aplicación en cuestión. Una vez que se recoge dichos datos, éstos son enviados al segundo componente instalado en la estación de trabajo, que después de recibir los datos recogidos desde el dispositivo móvil, lleva a cabo procesos estadísticos

⁸² LITTLE EYE. Little Eye wins 2013 TiE “Top Startup” Award. [En línea]. [Citado el 2 de junio de 2103]. Disponible en internet: <http://www.littleeye.co/blog/2013/06/02/littleeye-wins-tie-top-startup-award/>

⁸³ LITTLE EYE. How Little Eye Works Under The Hood. [En línea]. [Citado el 2 de junio de 2103]. Disponible en internet: <http://www.littleeye.co/blog/2013/07/24/how-little-eye-works-under-the-hood-part-1/>

con dichos datos para generar una serie de gráficas que representan el consumo de recursos (memoria RAM, CPU, batería, datos, almacenamiento en disco).

Tal como se ha mencionado, Little Eye se ejecuta en un proceso por separado lo que garantiza que si se presenta un bloqueo de Little Eye en el dispositivo móvil, no se afecta el funcionamiento de la aplicación monitoreada, y viceversa.

Procedimiento de instalación y ejecución de Little Eye

- I. Descargar Little Eye del sitio oficial (www.littleeye.co) e instalar en la estación de trabajo.
- II. Emplear la versión de prueba del software o utilizar una versión registrada según sea el caso. La versión 2.2.1.0 puede ser usada por seis meses con todas las funcionalidades.
- III. Conectar el teléfono Android mediante el cable USB. Se debe tener en cuenta los requerimientos que se especifican en ____ para garantizar que el procedimiento sea exitoso.
- IV. Seleccionar de la lista de aplicaciones instaladas en el teléfono, aquella que se desea monitorear. Dicha lista presentada por Little Eye, se obtiene automáticamente al conectar el dispositivo vía USB.
- V. Iniciar monitoreo dando clic al botón Start monitoring.

Una vez hecho lo anterior, Little Eye lanza en el teléfono la aplicación seleccionada. Cabe mencionar que el teléfono puede seguirse usando normalmente sin que esto afecte el monitoreo que se ha iniciado. Además, durante y después del proceso de monitoreo es posible observar el rendimiento de la aplicación y revisar cualquier detalle que sea necesario analizar. Al final es posible generar un reporte de la evaluación de rendimiento realizada.

Para esta evaluación los procedimientos de Little Eye empleados son:

- I. Energía consumida por la aplicación
 - a. Energía total de la app
 - Por CPU
 - Por Wifi
 - Por pantalla
 - Por GPS (Experimental)
 - Por 3G
 - b. Energía total del sistema
- II. Consumo de memoria
 - a. Consumo de memoria por aplicación – memoria de Dalvik
 - b. Consumo de memoria por aplicación – memoria nativa
- III. CPU
 - a. Porcentaje de ciclos de CPU consumidos a nivel de usuario.
 - b. Porcentaje de ciclos de CPU consumidos a nivel de sistema.

Finalmente, la etapa de ejecución de la evaluación y conclusión de la misma corresponden al cuarto objetivo específico del presente trabajo.

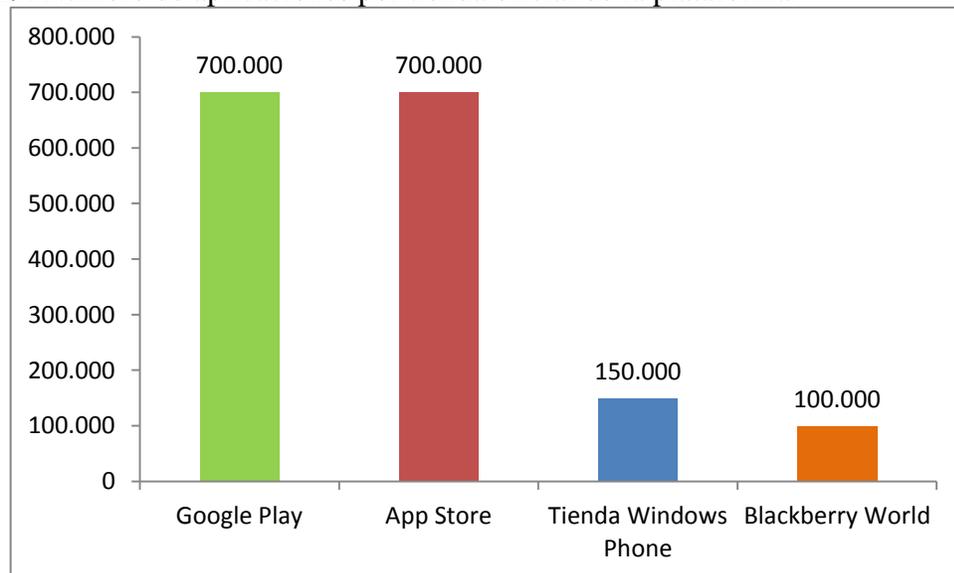
4.1.2 Describir las tecnologías de desarrollo de aplicaciones móviles para Android.

Para llevar a cabo este objetivo fue necesario realizar algunas actividades principales: recopilar información referente al desarrollo de aplicaciones móviles Android, revisar la información recopilada y organizarla por categoría, y sintetizar la información encontrada. Por lo anterior y de acuerdo a la información expuesta en el apartado 2.2 Marco Conceptual, de este trabajo, las tecnologías de desarrollo de aplicaciones móviles para Android se clasificaron en dos grandes ramas: por una parte están las herramientas de desarrollo de aplicaciones nativas y por otro lado, las herramientas de desarrollo de aplicaciones basadas en web, ambas para Android.

4.1.2.1 Herramientas para el desarrollo de aplicaciones nativas para Android. El gran auge de los dispositivos móviles, tanto teléfonos inteligentes como tabletas, ha disparado exponencialmente el desarrollo de aplicaciones para estos dispositivos, complementariamente al desarrollo de páginas web y software de escritorio.

Esto se ve reflejado al visitar las tiendas de aplicaciones de los fabricantes de dispositivos que lideran este mercado y que también poseen un sistema operativo propio. A continuación se puede observar algunas cifras en la figura 19.

Figura 19. Número de aplicaciones por tienda oficial de la plataforma.



Fuente: Autores del proyecto.

En la figura 19 se ve claramente que Google Play Store⁸⁴ y la AppStore⁸⁵ lideran con más de 700.000 aplicaciones cada una; por otro lado, Windows Phone ofrece unas 150.000 aplicaciones⁸⁶ y la BlackBerry World 100.000 aplicaciones.

Como se ve, las cifras son altas: sin tener en cuenta las aplicaciones de las tiendas de operadores de telefonía móvil, como la OVI Store de Nokia y algunas tiendas independientes, como GetJar o la más reciente Amazon Appstore.

Cuando se inicia en esta área, el desarrollador debe decidir dos elementos muy importantes: La plataforma para la cual desarrollará y las herramientas que usará para construir la aplicación. Para el caso de la plataforma se puede observar cuatro sistemas operativos sólidos en el mercado, como lo son: Android de Google, iOS de Apple, Windows Phone de Microsoft y BlackBerry OS que pertenece a la empresa conocida como Research In Motion, que hoy en día es BlackBerry.

Para el caso de este proyecto se utilizará Android por varias razones: primero, es el sistema operativo móvil más utilizado en el mundo⁸⁷, con más de 750 millones⁸⁸ dispositivos activados. Segundo, es una plataforma móvil de código abierto Linux⁸⁹, y este tipo de software generalmente es mayormente acogido por el ámbito académico, dado el caso de la facilidad para acceder a todas las herramientas de desarrollo de manera oficial.

A continuación se abordan algunas herramientas que hay disponibles para el desarrollo con tecnología nativa en la plataforma Android.

Basic4Android. Basic4Android es una plataforma de programación para aplicaciones Android cuyo lenguaje base de programación es Visual Basic, el cual está orientado a aquellas personas que inician en el mundo de la programación de una manera más gráfica y no tan abstracta. No es el mismo lenguaje de Microsoft, pero su sintaxis es la misma, por lo tanto posee sus mismas ventajas e inconvenientes.

Algunas de sus características son⁹⁰:

⁸⁴ WOMACK, Brian. Google Says 700,000 Applications Available for Android. Op., cit.

⁸⁵ CHAFFIN, Bryan. Google Matches Apple App Store with 700,000 Android Apps. [En línea]. [Citado el 3 de febrero de 2012] Disponible en internet: <http://www.macobserver.com/tmo/article/google-matches-apple-app-store-with-700000-android-apps>

⁸⁶ SVETLIK, Joe. Windows Phone Store hits 150,000 apps, doubled in last year. Op., cit.

⁸⁷ COMSCORE. ComScore Reports March 2013 U.S. Smartphone Subscriber Market Share. [En línea]. [Citado el 4 de mayo de 2013]. Disponible en internet: http://www.comscore.com/Insights/Press_Releases/2013/5/comScore_Reports_March_2013_U.S._Smartphone_Subscriber_Market_Share

⁸⁸ Update from the CEO. [En línea]. [Citado el 6 de mayo de 2013]. Disponible en internet: <http://googleblog.blogspot.com.es/2013/03/update-from-ceo.html>

⁸⁹ GOOGLE DEVELOPER ANDROID. Android, the world's most popular mobile platform. [En línea]. [Citado el 6 de mayo de 2013]. Disponible en internet: <http://developer.android.com/about/index.html>

⁹⁰ BASIC4ANDROID. Características Generales. [En Línea]. [Citado el 12 de mayo de 2013]. Disponible en Internet: <http://www.basic4ppc.com/android/why.html>

- Herramienta simple y poderosa para el desarrollo de aplicaciones Android nativas y sin mucho tiempo de programación.
- IDE y lenguaje de programación completo 100% enfocada en el desarrollo de Android.
- Compila a código de bytes nativo. No se necesitan bibliotecas en tiempo de ejecución.
- El rendimiento es similar a las aplicaciones escritas en Java.
- Soporta todos los teléfonos Android y tabletas desde Android 1.6 y superior de Android 4.x.

Finalmente es de resaltar que esta plataforma no es gratuita. En la página oficial de Basic4Android* se encuentra toda la documentación de este ambiente de programación para Android.

Xamarin. Esta herramienta es desarrollada por la empresa Xamarin, hace uso de los lenguajes C# y .NET, muy práctico para desarrolladores familiarizados con Visual Studio. Xamarin.Android trabaja con un lenguaje nativo para Android ya que no tiene un intérprete y su aprendizaje es relativamente sencillo.

Al igual que Basic4Android, Mono Xamarin.Android no es gratuito. Algunas características⁹¹ importantes de esta herramienta son:

- Aprovecha toda la potencia de C# y .NET.
- Controlar XML es fácil y rápido gracias a la incorporada en el XDocument clase sólo uno de los miles de .NET API disponible cuando utiliza Xamarin.Android.
- Gestione fácilmente las pulsaciones de botón y otros eventos de la interfaz de usuario.
- Proporciona acceso completo a la API de Android.
- Se puede hacer uso de Visual Studio o del editor propio Studio Xamarin.

En la página oficial de Xamarin* se encuentra todo lo relacionado con esta herramienta.

Android SDK. Google proporciona a los desarrolladores un completo kit de desarrollo para aplicaciones nativas, este es conocido como SDK (software development kit), el cual provee bibliotecas API y las herramientas de desarrollo necesarias para crear, probar y depurar aplicaciones para Android.

También se dispone de Android Development Tools (ADT), un plugin para el IDE Eclipse que está diseñado para dar un ambiente potente a la construcción de aplicaciones.

El ADT amplía las capacidades de Eclipse que permiten configurar rápidamente nuevos proyectos para Android, crear interfaces de usuarios, agregar los paquetes basados en la

⁹¹ XAMARIN. ¿Por qué te va a encantar Xamarin.Android? [En Línea]. [Citado el 12 de mayo de 2013]. Disponible en Internet: <http://xamarin.com/monoforandroid>

* <http://www.basic4ppc.com>

API Framework Android, depurar sus aplicaciones utilizando las herramientas del SDK de Android, e incluso exportar los archivos .apk con el fin de distribuir la aplicación.

Según Google⁹²: El desarrollo en Eclipse con ADT es muy recomendable y es la manera más rápida para empezar. Con la configuración del proyecto guiada que ofrece, así como la integración de herramientas, editores de XML personalizados, y el panel de resultados de depuración, ADT le da un impulso increíble en el desarrollo de aplicaciones Android.

Android Studio. Esta herramienta está basada en el IntelliJ IDEA Community Edition, un IDE (Integrated Development Environment) para Java y es el primer IDE dedicado para este sistema operativo móvil.

Android Studio ofrece la posibilidad de ir escribiendo a la vez que se puede obtener una vista previa en tiempo real, ahorrando tiempo y esfuerzo, además ofrece la posibilidad de contactarse con traductores profesionales que ayudan con la internacionalización de la interfaz. También ofrece gráficos estadísticos sobre las ventas de la aplicación y consejos para optimizarla. Toda la información relacionada se encuentra en la página web de Android Studio^{**}.

Android NDK. El NDK es un conjunto de herramientas que permite implementar partes de la aplicación utilizando lenguajes de código nativo como C y C++, solamente en el caso que sea necesario programar en dichos lenguajes y no porque simplemente sean de preferencia.

El NDK permite instalar bibliotecas escritas en C y otros lenguajes, una vez compiladas para ARM o código x86 nativo. Los programas Java corriendo en la máquina virtual Dalvik (Dalvik VM) pueden llamar a clases nativas por medio de la función System.loadLibrary, que forma parte de las clases estándar Java en Android.

Lo antes expuesto muestra las opciones a la hora de elegir herramientas de desarrollo para crear aplicaciones nativas para este sistema operativo móvil.

En el momento de desarrollo una parte muy significativa y que en ocasiones consume mucho tiempo, son las interfaces de usuario, más si se analiza un caso especial, donde una sola persona desarrolla la aplicación y cumple diferentes roles durante el proceso, ser el diseñador de interfaces, es uno de ellos. Debido a esto y como se mencionó anteriormente, en el caso de Eclipse se cuenta con el plugin ADT y recientemente se añadió Android

⁹² GOOGLE DEVELOPER ANDROID. ADT Plugin. [En Línea]. [Citado el 12 de mayo de 2013]. Disponible en Internet: <http://developer.android.com/tools/sdk/eclipse-adt.html#notes>

* <http://xamarin.com>

** <http://developer.android.com/sdk/installing/studio.html>

Studio que agiliza este proceso; no obstante cabe resaltar una herramienta muy expuesta en la web, se trata de DroidDraw .

DroidDraw. Esta es una herramienta escrita en Java para el diseño de interfaces de usuarios con tan sólo arrastrar y soltar componentes. DroidDraw dispone de una versión web y una aplicación multiplataforma solamente con descargar y ejecutar el fichero .jar.

Esta herramienta genera el código XML de las pantallas, también permite exportar un archivo .apk para instalar en el dispositivo físico o en el emulador. Es válido resaltar que esta herramienta es open-source, y se puede conocer más en su sitio web* .

En la búsqueda de herramientas para desarrollo nativo en la plataforma Android puede verse que existen algunas que sólo se limitan a la elaboración de interfaces, otras que van más allá y ofrecen todo un entorno de desarrollo haciendo uso de otros lenguajes, pero enfocados en realizar aplicaciones que se ejecuten de manera nativa.

Conociendo estas herramientas sólo queda mencionar algo no menos importante: el lenguaje de programación. En el desarrollo de aplicaciones móviles de forma nativa para Android se hace uso del lenguaje Java. Cabe aclarar que no se trata de J2ME, que es la versión de Java orientada a dispositivos móviles. Jeff Friesen⁹³ menciona:

Los desarrolladores utilizan diferentes ediciones de la plataforma Java para crear programas Java que se ejecutan en equipos de escritorio, navegadores web, servidores Web, dispositivos móviles de información (por ejemplo, teléfonos celulares) y los dispositivos integrados (como la televisión set-top boxes):

- Java Platform, Standard Edition (Java SE): La plataforma Java para el desarrollo de aplicaciones, que son programas independientes que se ejecutan en escritorios. Java SE también se utiliza para desarrollar applets, que son programas que se ejecutan en el contexto de un navegador web.
- Java Platform, Enterprise Edition (Java EE): La plataforma Java para el desarrollo de aplicaciones y servlets orientados a empresas, que son programas de servidor que se ajustan a API Servlet de Java EE. Java EE se construye en la parte superior de Java SE.
- Java Platform, Micro Edition (Java ME): La plataforma Java para MIDlets en desarrollo, que son programas que se ejecutan en el dispositivo móvil, y Xlets, que son programas que se ejecutan en dispositivos integrados.

Los desarrolladores también utilizan una edición especial creada por Google de la plataforma Java de creación de aplicaciones para Android que se ejecutan en dispositivos Android habilitados. Esta edición es conocida como la plataforma

* <http://www.droiddraw.org/>

⁹³ FRIESEN, Jeff. Java for Android Development. España: Anaya Multimedia-Anaya Interactiva, 2010. p. 5.

Android. La plataforma Android de Google se compone principalmente de bibliotecas del núcleo de Java (en parte basado en Java SE) y una máquina virtual conocida como Dalvik, especialmente modificada del kernel de Linux.

De esta manera se tiene todo un entorno, oficial y libre, de desarrollo de aplicaciones nativas para el sistema operativo Android. Adicionalmente se encuentra una amplia documentación con tutoriales y ejemplos para el diseño, desarrollo y distribución de las aplicaciones en el sitio web de Developer Android*.

En el cuadro 13, se pueden observar las características básicas de las herramientas que ofrecen un entorno para el desarrollo de estas aplicaciones.

* <http://developer.android.com/tools/sdk/eclipse-adt.html#notes>

Cuadro 13. Herramientas para desarrollo nativo sobre la plataforma Android.

Herramienta	Licencia	Lenguaje de programación	Dispositivos de destino	Propietario	IDE	Compatibilidad
Basic4Android	Propietario	VisualBasic	Android	Anywhere software	Basic4Android	Windows
XAMARIN	Propietario	C #.	iOS, Android, Windows y Mac	Xamarin	Xamarin Studio	Windows
ANDROID SDK	Libre	Java	Android	Google Inc	Eclipse, Android Studio	Windows, Linux, Mac OS X
ANDROID NDK	Libre	C	Android	Google Inc	Eclipse, Android Studio	Windows, Linux, Mac OS X

Fuente: Autores del proyecto.

4.1.2.2 Herramientas para el desarrollo de aplicaciones basadas en web para Android. Existe actualmente una variedad de herramientas de software disponibles para facilitar el desarrollo de aplicaciones móviles. Entre éstas se encuentran frameworks, plugins, librerías y similares, todas ellas para el desarrollo de aplicaciones móviles basadas en tecnologías de tipo web.

En el presente documento se hace referencia a unas cuentas herramientas de las que se encuentran en Internet, teniendo en cuenta que la lista de éstas es muy extensa. Según Developer Economics 2013⁹⁴, se han identificado más de 100 CPTs (Cross-platform Tool – Herramienta de plataforma cruzada), categoría que engloba todas aquellas herramientas que facilitan el desarrollo de aplicaciones móviles multiplataforma, que son basadas en tecnologías web pero que se enfocan en conseguir aplicaciones que corran sobre smartphones, tablets, entre otros dispositivos y pertenecientes a las distintas plataformas móviles.

A continuación se presenta una clasificación realizada de acuerdo al análisis hecho a las distintas herramientas disponibles. Dicha clasificación se elaboró para dar claridad y evitar la confusión que se puede generar debido a la extensa cantidad de frameworks, plugins o librerías, y que en general se tiene poca documentación de ellos.

Tipo 1: Framework para facilitar la creación de las interfaces de la aplicación móvil. Hacen uso de HTML5, CSS, JavaScript. Disponen de temas, elementos de interfaz. Están diseñados para dispositivos con pantalla táctil. Se tienen:

- jQuery Mobile: elaborado específicamente para interfaces de apps móviles.
- joApp: funciona en iOS, Android, BlackBerry, webOS, Chrome OS. Compatibilidad con PhoneGap.
- SenchaTouch: Similar a jQuery Mobile, pero sólo es con Javascript y CSS.
- jQTouch: No es framework, es un plugin de jQuery. Se centra en los dispositivos de Apple. Para móviles compatibles con WebKit.
- LungoJS: para iOS, Android, Blackberry y WebOS.
- Dojo Mobile: Compatibilidad con PhoneGap.

Tipo 2: Framework para empaquetar y generar instalador, se les llama contenedores. La aplicación se desarrolla con HTML, CSS, JavaScript. La finalidad principalmente es brindar acceso a recursos de hardware del dispositivo mediante librerías propias del framework, y además, permite empaquetar la aplicación para generar el instalador de la plataforma deseada. Algunos soportan más plataformas que otras.

⁹⁴ Developer Economics 2013: The Tools Repost. [En línea]. Citado el 13 de diciembre de 2013. Disponible en internet En: <http://www.developereconomics.com/phonegap-and-appcelerator-lead-developer-mindshare-across-tens-of-cpts/>

- Phonegap: es un sistema (no se considera un framework) para crear aplicaciones usando exclusivamente HTML5, CSS3 y Javascript, ejecutadas dentro en un componente WebKit del móvil. Empaquetador de aplicaciones web en aplicaciones nativas.

Tipo 3: Framework para escribir en un lenguaje diferente al nativo y generar el instalador. La aplicación se desarrolla en un lenguaje en particular, y el framework funciona como plataforma intermedia para generar el instalador.

- AppceleratorTitanium: Se desarrolla con JavaScript. Compila el Javascript a nativo. Soporta iOS y Android. Accede directamente al hardware. Vistas con JavaScript. La aplicación producto es muy similar a una nativa. No contiene HTML ni CSS.
- RhodesRhoMobile: Se desarrolla con Ruby, se compila a nativo. Soporta Android, iOS, Windows Phone, BlackBerry, Symbian. Interfaces con HTML. Es modelo vista controlador.

En el cuadro 14 se presenta una comparativa a las herramientas antes descritas con aspectos de gran importancia.

Cuadro 14. Herramientas para desarrollo basado en web sobre la plataforma Android.

Fuente: Autores del proyecto.

NOMBRE	DESARROLADOR	PLATAFORMAS DE DESTINO	LICENCIA	ACCESO AL HARDWARE	TECNOLOGÍAS PARA INTERFAZ GRÁFICA	TECNOLOGÍAS PARA LA FUNCIONALIDAD
RHOMOBILE (RHOELEMENTS)	Motorola	<ul style="list-style-type: none"> - Windows Embedded Handheld - Windows CE - Windows Phone 7 - Apple iOS - Android - BlackBerry 	MIT	<ul style="list-style-type: none"> - GPS - Contactos - Calendario - Cámara - Mapas - Bluetooth - NFC 	<ul style="list-style-type: none"> - HTML - CCS 	- Ruby
PHONEGAP	Adobe	<ul style="list-style-type: none"> - Windows Phone - Apple iOS - Android - BlackBerry OS - WebOS - Bada - Symbian 	Apache 2.0	<ul style="list-style-type: none"> - Acelerómetro - Cámara - Brújula - Contactos - Historial - Geolocalización - Notificación (alerta, sonido y vibraciones) - Almacenamiento 	<ul style="list-style-type: none"> - HTML - CSS 	- JavaScript
APPCELERATOR TITANIUM	Appcelerator	<ul style="list-style-type: none"> - Apple iOS - Android - BlackBerry 	Apache 2.0	Acceso total al dispositivo	<ul style="list-style-type: none"> - JavaScript - Es posible utilizar HTML, CSS pero no es lo habitual. 	- JavaScript

Cuadro 15. Herramientas para desarrollo de interfaces gráficas para aplicaciones basadas en web sobre la plataforma Android.

NOMBRE	DESARROLLADOR	PLATAFORMAS DE DESTINO	COMPATIBILIDAD PARA GENERAR APLICACIÓN BASADA EN WEB	LICENCIA	TECNOLOGÍAS
JQUERY MOBILE	jQuery	- iOS - Android - BlackBerry OS - WebOS - Bada - Symbian - Windows Phone - MeeGo - Palm webOS - Symbian	- PhoneGap	MIT	- HTML - CSS - JavaScript
JQT (jQTouch)	David Kaneda y mantenido por Jonathan Stark	- iOS - Android - WebOS	- PhoneGap	MIT	- HTML - CSS - JavaScript
SENCHA TOUCH	Sencha	- Android - iOS - BlackBerry OS - Windows Phone	- PhoneGap	- GPL v3 - Comercial	- HTML - CSS - JavaScript
IJU	IJU	- iOS - Android - WebOS - Bada		MIT	- HTML - CSS - JavaScript
THE M PROJECT		- iOS - Android - BlackBerry - WebOS	- PhoneGap	MIT	- HTML - CSS - JavaScript
LUNGO.JS	LungoJs	- iOS - Android - BlackBerry - WebOS	- PhoneGap	GPLv3	- HTML - CSS - JavaScript
ZEPTO.JS		- iOS - Android - BlackBerry - WebOS		MIT	- HTML - CSS - JavaScript
DOJO	Dojo Foundation	- iOS - Android - BlackBerry	- PhoneGap	BSD AFL	- HTML - CSS - JavaScript

Fuente: Autores del proyecto.

4.1.3 Construir una aplicación móvil tanto en tecnología nativa como una basada en web que sirva como caso de estudio para el proceso de evaluación de los atributos de calidad del software.

4.1.3.1 Descripción de la aplicación desarrollada. Se construyó una aplicación móvil con el nombre “NewsApp” tanto en la versión nativa como en la basada en web; ambas se desarrollaron buscando la mayor similitud entre ellas.

NewsApp es un sencillo lector o recopilador de contenidos (también conocido como lector de RSS). Este lector presenta en la primera interfaz cinco opciones de fuentes RSS, entre las que el usuario selecciona la que desee, luego aparece una segunda interfaz con los titulares de las noticias que ofrece la fuente escogida. Al pulsar sobre uno de los titulares de la lista, se muestra una tercera interfaz donde además del título de la noticia, se indica una breve descripción de la misma y el enlace para visualizar la noticia completa en su sitio web.

Para el caso de la versión nativa de NewsApp, esta se construyó con base en una aplicación desarrollada en un curso de inicios en programación Android de la comunidad de aprendizaje Maestros del Web.⁹⁵ Asimismo, la versión basada en web de NewsApp, se construyó con base al tutorial para el desarrollo de un lector simple de RSS⁹⁶. Cabe resaltar que en ambos casos se hicieron modificaciones importantes al código fuente original.

Por último, cabe decir que el manual de instalación de NewsApp se presenta en el Anexo B de este mismo trabajo, asimismo, en el Anexo C se hace referencia a los archivos .apk de la versión nativa de NewsApp y el de la versión basada en web.

4.1.3.2 Herramientas de desarrollo empleadas

- **Aplicación nativa.** Luego de analizar las características de las herramientas se escogió el SDK de Android y el IDE Eclipse con el plugin ADT, por razones que se detallan a continuación:

Teniendo en cuenta que el SDK de Android es la herramienta oficial de este sistema operativo desarrollado y apoyado Google por lo tanto cualquier actualización en el sistema operativo inmediatamente será reflejada en estas herramientas para su manejo.

También se puede notar que estas herramientas son open-source, muy adecuado, como ya se conoce, para trabajos en el ambiente académico por la facilidad de obtener los recursos,

⁹⁵ Curso Android: Construir un lector de feeds simple. [En línea]. [Citado el 10 de Octubre de 2013]. Disponible en internet en: <http://www.maestrosdelweb.com/editorial/curso-android-construir-lector-de-feeds/>

⁹⁶ CAMDEN, Raymond. PhoneGap RSS Reader. [En línea]. [Citado el 10 de Octubre de 2013]. Disponible en internet en: <http://www.raymondcamden.com/index.cfm/2012/1/24/PhoneGap-RSS-Reader--Part-3>

además el hecho de ser libre genera una gran comunidad de desarrolladores en esta área que darán apoyo para el aprendizaje de dichas herramientas.

- **Aplicación basada en web.** Se eligió a jQuery Mobile como framework para la elaboración de las interfaces de usuario, debido a:

Su característica de ser multiplataforma lo ubica como un framework ampliamente utilizado. Además, es compatible con una vasta serie de dispositivos táctiles.

Se basa en tecnologías web, que por supuesto ya son ampliamente conocidas y manejadas, lo que facilita sustancialmente el desarrollo con este framework.

Es un framework que se deriva del reconocido jQuery, y por tanto, sugiere un significativo nivel de confianza.

Además, Phonegap se seleccionó para empaquetar y generar el instalador de la aplicación, por las siguientes razones:

Es la plataforma más usada por los desarrolladores de acuerdo a un estudio realizado por Developer Economics⁹⁷ en 2013. Como se aprecia en la figura 20, el 34% de los encuestados manifiestan preferir Phonegap por encima de Appcelerator (21%), Adobe AIR (19%), Sencha (12%).

Es compatible con las principales plataformas móviles.

Está lanzado bajo licencia Apache, lo que lo hace un proyecto muy atractivo, confiable y cercano a la comunidad académica.

Su documentación es muy completa.

Es compatible con jQuery Mobile lo cual permite una integración completa entre estas herramientas.

⁹⁷ Developer Economics 2013. [En línea]. [Citado el 10 de Octubre de 2013]. Disponible en internet en: <http://www.developereconomics.com/phonegap-and-appcelerator-lead-developer-mindshare-across-tens-of-cpts/>

Figura 20. Plataformas para desarrollo de aplicaciones basadas en web más usadas por los desarrolladores.



Fuente: Developer Economics 2013. [En línea]. [Citado el 10 de Octubre de 2013]. Disponible en internet en: <http://www.developereconomics.com/phonegap-and-appcelerator-lead-developer-mindshare-across-tens-of-cpts/>

De otra parte, se determinó el uso de Eclipse como entorno de desarrollo por las mismas razones que dieron lugar a emplearse en el desarrollo nativo, puesto que es la recomendación oficial de Google tanto por el plugin ADT como por la facilidad de editor de archivos XML y demás necesarios en un proyecto Phonegap para Android.

Metodología de desarrollo. Las metodologías denominadas ágiles son las que mejor se adecúan para el caso de aplicaciones móviles. Blanco, Camarero, Fumero, Warterski y Rodríguez⁹⁸ señalan: “[...] las metodologías ágiles poseen ciertas propiedades que las hacen totalmente aplicables al dominio del software en los móviles.”, de acuerdo a esto se hizo uso de la metodología ágil XP en la construcción de NewsApp. A propósito de esta metodología, Bautista dice:

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes,

⁹⁸ BLANCO, Paco; CAMARERO, Julio; FUMERO, Antonio, WERTERSKI, Adam y RODRÍGUEZ, Pedro. Metodología de desarrollo ágil para sistemas móviles. Introducción al desarrollo con Android y el iPhone. [En línea]. [Citado el 5 de diciembre de 2013]. Disponible en internet: http://www.adamwesterski.com/wp-content/files/docsCursos/Agile_doc_TemasAnv.pdf

simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico⁹⁹.

- **Etapas de desarrollo de XP.** La programación extrema engloba un conjunto de reglas que se ejecutan dentro de cuatro actividades estructurales: planeación, diseño, codificación y pruebas.

Planeación: Esta actividad comienza atendiendo a los clientes, para entender el contexto del negocio y establecer las características que se requieren. Según Pressman¹⁰⁰ estas características se transforman en requerimientos del negocio que se especifican mediante Historias de Usuario; las cuales recogen la interacción hablada entre desarrolladores y usuarios. Una vez se tienen las Historias de Usuario, el equipo de desarrollo las divide en tareas, estima el esfuerzo, recursos requeridos para su implementación, se genera el plan de entregas, las iteraciones, la rotación de parejas y las reuniones diarias.

Diseño. En esta etapa son evaluadas las historias de usuario por el equipo del proyecto para dividir las en tareas, cada tarea representa una característica distinta del sistema.

Desarrollo. Durante el desarrollo se lleva a cabo la programación en pareja, la unidad de pruebas y la integración del código, es altamente importante la disponibilidad del cliente para que pueda resolver cualquier duda que se presente.

Pruebas. Las pruebas se deben realizar a cada característica del sistema de manera que se pueden analizar cada una de las clases y métodos presentes.

Para efectos de este proyecto no se realizó la etapa de pruebas que propone la Metodología XP, teniendo en cuenta que la Aplicación NewsApp fue sometida a una evaluación desde el punto de vista de los atributos de calidad propuestos por la norma ISO/IEC 9126.

⁹⁹ BAUTISTA, José. Programación Extrema XP. [En línea]. [Citado el 10 de octubre de 2013]. Disponible en Internet en: http://ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html

¹⁰⁰ PRESSMAN, Op. cit., p.62.

Planeación inicial:

Cuadro 16. Historia de usuario de cargar noticias.

Historia de Usuario	
Número: 1	Usuario: Cliente.
Nombre historia: Cargar noticias.	
Prioridad en negocio: Alta (Alta/Media/baja)	Riesgo en desarrollo: Alta (Alta/Media/baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Equipo XP	
Descripción: Esta es la pantalla principal de la aplicación la cual luego de que ésta inicie muestra en una lista los titulares de las noticias ofrecidas por la fuente RSS: http://noticiasunolaredindependiente.com/feed/	
Observaciones: No hay	

Fuente: Autores del proyecto.

Cuadro 17. Historia de usuario de ver detalle de la noticia.

Historia de Usuario	
Número: 2	Usuario: Cliente.
Nombre historia: Ver detalle de la noticia.	
Prioridad en negocio: Alta. (Alta/Media/baja)	Riesgo en desarrollo: Media. (Alta/Media/baja)
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Equipo XP	

Fuente: Autores del proyecto.

Cuadro 17. (Continuación)

Historia de Usuario
<p>Descripción:</p> <p>Luego de pulsar sobre uno de los titulares aparece una interfaz donde el usuario podrá leer, además del titular de la noticia, un pequeño detalle de ésta, y también se encontrará el enlace “Ver más” para acceder a la página web donde se muestra la noticia completa.</p>
<p>Observaciones: No hay</p>

Fuente: Autores del proyecto.

Iteración 1

Historias realizadas en esta iteración: H1 y H2

Cuadro 18. Tarea carga de titulares de noticias.

Tarea	
Número tarea: 1	Número historia: 1
Nombre tarea: Carga de titulares de noticias	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 1 de septiembre	Fecha fin: 22 de septiembre
Programador responsable: Equipo XP	
<p>Descripción:</p> <p>Diseño de la interfaz principal para mostrar los titulares por medio de una lista.</p> <p>Se debe crear un método que permita la lectura de los titulares desde la fuente RSS para su posterior presentación en la pantalla del teléfono.</p>	

Fuente: Autores del proyecto.

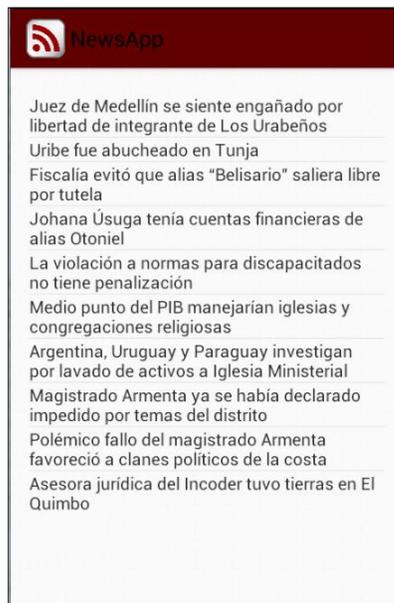
Cuadro 19. Tarea ver detalle de la noticia.

Tarea	
Número tarea: 2	Número historia: 2
Nombre tarea: Ver detalle de la noticia.	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 23 de septiembre	Fecha fin: 10 de octubre
Programador responsable: Equipo XP	
<p>Descripción:</p> <p>Construir una interfaz donde se muestra el detalle de la noticia luego de haberla seleccionado, además, debe aparecer un enlace “Ver más” para ir a la página web y ampliar la información de dicha noticia.</p>	

Fuente: Autores del proyecto.

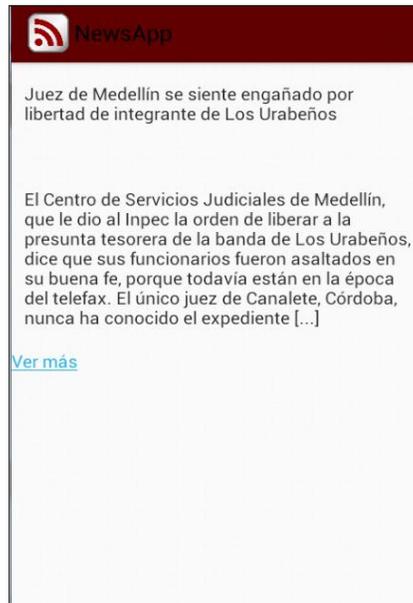
Demo: Versión Nativa

Figura 21. Interfaz principal de NewsApp nativa de primer demo.



Fuente: Autores del proyecto.

Figura 22. Interfaz detalle de noticia de NewsApp nativa de primer demo.



Fuente: Autores del proyecto.

Demo: Versión basada en web

Figura 23. Interfaz principal de NewsApp basada en web de primer demo.



Fuente: Autores del proyecto.

Figura 24. Interfaz detalle de noticia de NewsApp basada en web de primer demo.



Fuente: Autores del proyecto.

Iteración 2

Aparecen algunos cambios por lo tanto se crea una nueva historia de Usuario.

Cuadro 20. Historia de usuario lista de fuentes de noticias.

Historia de Usuario	
Número: 3	Usuario: Cliente.
Nombre historia: Lista de fuentes de noticias.	
Prioridad en negocio: Media. (Alta/Media/baja)	Riesgo en desarrollo: Baja. (Alta/Media/baja)
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Equipo XP	

Fuente: Autores del proyecto.

Cuadro 20. (Continuación)

Historia de Usuario
<p>Descripción:</p> <p>Esta es ahora la pantalla principal de la aplicación, la cual luego de que la aplicación inicie muestra una lista con cinco fuentes RSS: Noticias Uno, Ministerio de Educación, Acac, Ciencia y Tecnología, Sgoliver.net Blog.</p>
<p>Observaciones: No hay</p>

Fuente: Autores del proyecto.

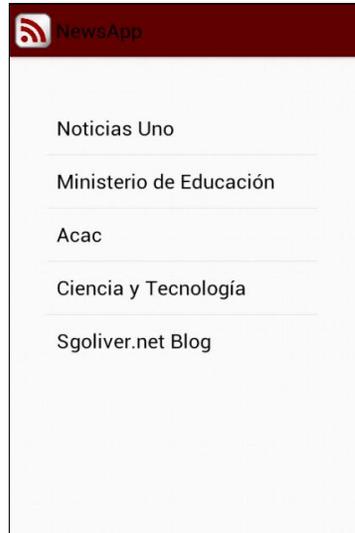
Cuadro 21. Tarea de lista fuente de noticias.

Tarea	
Número tarea: 3	Número historia: 3
Nombre tarea: Lista de fuentes de noticias.	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 11 de octubre	Fecha fin: 20 de Octubre
Programador responsable: Equipo XP	
<p>Descripción: Se crea una nueva interfaz que ahora será la inicial, donde se podrá elegir entre cinco fuentes RSS, que son: Noticias Uno, Ministerio de Educación, Acac, Ciencia y Tecnología, Sgoliver.net Blog.</p>	

Fuente: Autores del proyecto.

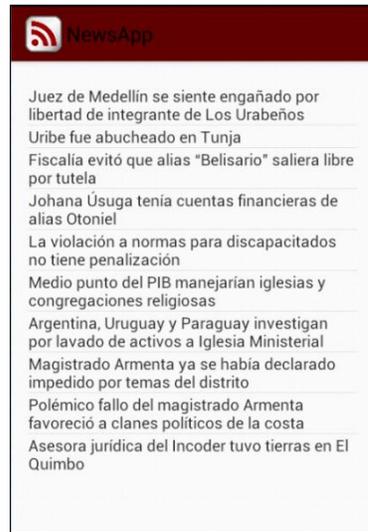
Demo 2 (versión final): Versión nativa.

Figura 25. Interfaz principal de NewsApp nativa de segundo demo.



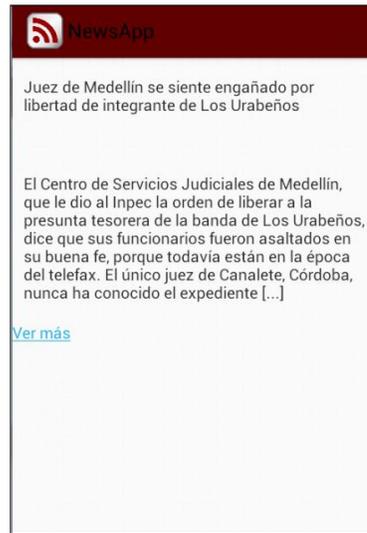
Fuente: Autores del proyecto.

Figura 26. Interfaz lista titulares de noticias de NewsApp nativa de segundo demo.



Fuente: Autores del proyecto.

Figura 27. Interfaz detalle de noticia de NewsApp nativa de segundo demo.



Fuente: Autores del proyecto.

Figura 28. Interfaz principal de NewsApp basada en web de segundo demo.



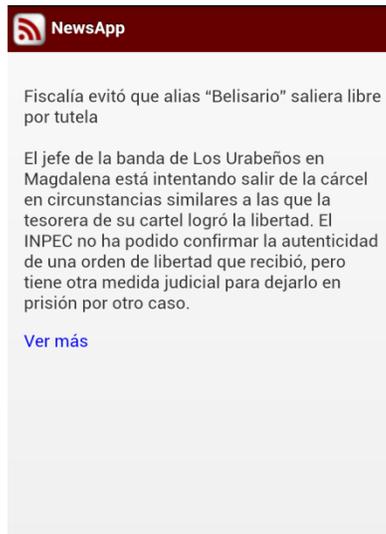
Fuente: Autores del proyecto.

Figura 29. Interfaz lista titulares de noticias de NewsApp basada en web de segundo demo.



Fuente: Autores del proyecto.

Figura 30. Interfaz detalle de noticia de NewsApp basada en web de segundo demo.



Fuente: Autores del proyecto.

Diario de Actividades NewsApp

Nombre: Alveiro Rosado

Equipo: XP

Rol desempeñado: Director

Cuadro 22. Diario de actividades de iteración 1.

Fecha (día/mes)	Actividad Realizada	Tiempo dedicado (en horas)	Observaciones
15/08	Primera reunión con el cliente	2	Aclarando la idea del proyecto.
22/08	Revisión con el grupo de las historias de usuario	2	Se aclararon dudas.
11/10	Revisión demo 1	2	Las cosas marchan bien.
10/10	Reunión con el cliente	2	Se realizó una nueva historia de usuario.
11/10	Reunión con el equipo XP	2	Socialización de la nueva historia de usuario.
18/10	Revisión versión final	2	Aprobado.

Fuente: Autores del proyecto.

Nombre: Álvaro Javier Durán Sanjuán, Jorge Luis Peinado Rodríguez

Equipo: XP

Rol desempeñado: Programadores

Cuadro 23. Diario de actividades de iteración 2.

Fecha (día/mes)	Actividad realizada	T(h)	Observaciones
15/08	Reunión de presentación del equipo XP	0.5	Primera toma de contacto con los miembros que van a componer el equipo.
15/08	Reunión entre el cliente y equipo XP para conocer las necesidades de la empresa y las tareas que se deben realizar	1,5	

Cuadro 23. (Continuación)

Fecha (día/mes)	Actividad realizada	T(h)	Observaciones
27/09	Lectura historias de usuario	3	
29/09	Resolución de dudas con el cliente	1	
1/09	Asignación de tareas e inicio de primera iteración.	1	
10/09	Análisis demo 1	2	
10/10	Reunión con el cliente	2	Aparece un nuevo requerimiento.
11/10	Asignación de tareas e inicio iteración dos.	2	
18/10	Análisis versión final	2	Aprobado

Fuente: Autores del proyecto.

4.1.4 Efectuar el proceso de evaluación de los atributos de calidad del software tanto a la aplicación nativa como a la aplicación basada en web. El desarrollo de este objetivo consiste en la última etapa del proceso de evaluación definido en el estándar ISO/IEC 14598.

4.1.4.1 Ejecución de la evaluación. Conforme al plan de evaluación, se tomaron medidas de comportamiento temporal (Tiempo de respuesta) y utilización de recursos (CPU, memoria RAM, batería).

Como se mencionó en el desarrollo del objetivo 1, apartado 4.1.1.3, la herramienta para la toma de medidas, es el software Little Eye.

De los recursos materiales disponibles para este proyecto, el computador de escritorio en el que se llevó a cabo el monitoreo de la aplicación móvil NewsApp, es el que tiene las características de hardware que se especifican en el cuadro 24.

Cuadro 24. Especificaciones de hardware de Computador para monitoreo.

COMPUTADOR PARA MONITOREO	
Procesador	AMD Phenom II X6 2.7 Ghz
Memoria RAM	4 GB
Disco duro	1 TB
Pantalla	Samsung 21"
SO	Windows 7 Professional 64 bits

Fuente: Autores del proyecto.

Así también, el teléfono celular que se empleó en la evaluación es el Sony Xperia S, con especificaciones que se detallan en el cuadro 25:

Cuadro 25. Especificaciones de hardware de Sony Xperia S.

SMARTPHONE SONY XPERIA S (LT26i)	
Procesador	Dual-core de 1,5 GHz
Chipset	Qualcomm MSM8260 Snapdragon
GPU	Adreno 220
Memoria RAM	1 GB
Memoria interna	32 GB
Pantalla	720 x 1280 píxeles 4.3 pulgadas
Sensores	Acelerómetro, giroscopio, proximidad, brújula
Navegador	Compatible HTML5
Versión SO	Android 4.0 Ice Cream Sandwich

Fuente: Autores del proyecto.

También se verificaron las condiciones en el teléfono, antes de ejecutar la evaluación, con el fin de obtener información con respecto al uso de memoria RAM, almacenamiento interno y estado de la batería como se indica en el cuadro 26.

Cuadro 26. Condiciones del teléfono previas a la evaluación.

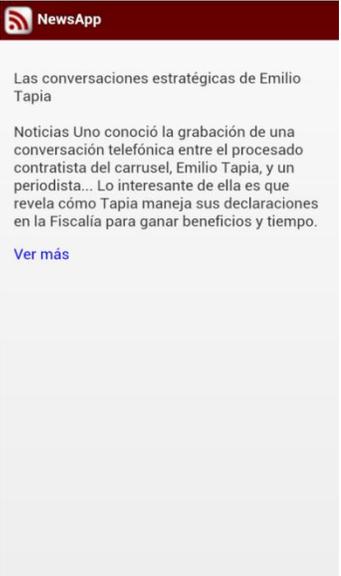
Monitoreo de NewsApp			
Condiciones iniciales en el teléfono			
Uso de memoria RAM	330MB usada; 308MB libre	Intensidad señal WiFi	Aceptable
Batería	100%	Velocidad de descarga*	0.96 Mbps
Brillo de la pantalla	50%	Almacenamiento interno	25 GB libres
Aplicaciones en ejecución	Solamente NewsApp y Little Eye App Monitor		

Fuente: Autores del proyecto.

4.1.4.2 Resultados de la medición. Con el fin de indicar claramente en qué parte de NewsApp se llevaba a cabo cada tarea y sus respectivas mediciones, se le dio identificador a cada interfaz de la aplicación, como se ve en la figura 31.

* Velocidad de descarga medida con la aplicación móvil OOKLA Speed Test.

Figura 31. Interfaces de la aplicación NewsApp.

Pantalla A	Pantalla B	Pantalla C
Lista de fuentes RSS	Lista de titulares	Descripción de la noticia
		

Fuente: Autores del proyecto.

El proceso de toma de medidas se realizó siguiendo una guía que se muestra en el cuadro 30, éste consta de tres columnas: la primera ordena la secuencia de pasos; en la segunda se indica la tarea a realizar y en la tercera se describe cómo ejecutar la tarea respectiva. Además, este procedimiento permitió evaluar las 5 fuentes RSS con que cuenta NewsApp y se siguió tanto en la evaluación para la versión nativa como para la basada en web.

Cuadro 27. Guía de procedimiento para la toma de medidas en Little Eye.

No.	Tarea	Descripción
0.	Ejecutar Little Eye	Abrir Little Eye en el computador de escritorio donde se realizará el monitoreo.
1.	Inicio de la aplicación	Seleccionar NewsApp en la lista de aplicaciones que presenta Little Eye y dar clic en Start monitoring. Little Eye ejecutará NewsApp y Little Eye App Monitor en el teléfono e iniciará el monitoreo en el computador de escritorio.

Cuadro 27. (Continuación)

1.1	Esperar que transcurran 20 segundos	Espera para dar tiempo a que la aplicación inicie totalmente y se estabilice dentro del sistema.
2.	Seleccionar un elemento de la pantalla A	Pulsar sobre una de las fuentes RSS de la lista.
2.1	Esperar que transcurran 30 segundos	Espera para dar tiempo a que la aplicación cargue la lista de titulares y se estabilice dentro del sistema, para luego ejecutar la siguiente tarea.
3.	Seleccionar un elemento de la pantalla B	Pulsar sobre uno de los titulares de la lista.
3.1	Esperar que transcurran 20 segundos	Espera para dar tiempo a que la aplicación cargue la descripción de la noticia y se estabilice dentro del sistema, para luego ejecutar la siguiente tarea.
4.	Volver a la pantalla B	Pulsar el botón atrás del teléfono.
4.1	Esperar que transcurran 20 segundos	Espera para dar tiempo a que la aplicación cargue la pantalla B, para luego ejecutar la siguiente tarea.
5.	Seleccionar un elemento de la pantalla B (diferente al escogido en el paso 3)	Pulsar sobre uno de los titulares de la lista.
5.1	Esperar que transcurran 20 segundos	Espera para dar tiempo a que la aplicación cargue la descripción de la noticia y se estabilice dentro del sistema, para luego ejecutar la siguiente tarea.
6.	Volver a la pantalla B	Pulsar el botón atrás del teléfono.
6.1	Esperar que transcurran 20 segundos	Espera para dar tiempo a que la aplicación cargue la pantalla B, para volver a la pantalla A.
7.	Volver a la pantalla A	Pulsar el botón atrás del teléfono.
7.1	Esperar que transcurran 20 segundos	Espera para dar tiempo a que la aplicación cargue la pantalla A y se estabilice dentro del sistema.
8.	Repetir el proceso	Volver al punto 2 y seguir los pasos hasta evaluar todas las fuentes RSS con que cuenta la aplicación.

Fuente: Autores del proyecto.

4.1.4.3 Toma de datos. El registro de los datos obtenidos a partir de Little Eye, se realizó mediante técnica de inspección en las gráficas presentadas por dicha herramienta.

De acuerdo a ISO/IEC 9126 que plantea evaluar la aplicación en una situación donde ésta se vea exigida, denominada peor caso, se analizó que el comportamiento cuando se accede a la fuente RSS “Ministerio de Educación” corresponde a tal caso puesto proporciona una lista de noticias que supera los 200 ítems.

A continuación se presentan los resultados en cuadros que especifican el valor obtenido en cada medición. De acuerdo a la guía de procedimiento explicada en el cuadro 27, por cada fuente RSS que se evaluó, se seleccionaron dos titulares de noticia de la lista, por lo cual en el cuadro de los resultados aparece dos veces “Selección de un titular de noticia” en la columna “Tarea”; asimismo, aparece dos veces “Volver atrás, a la pantalla B” porque el procedimiento así lo requiere.

Por otro lado, las mediciones de la subcaracterística comportamiento temporal se llevaron a cabo inspeccionando la línea de tiempo y la captura del video que Little Eye presenta durante la monitorización, la cual permitió observar el instante en el que se ejecutó la tarea en el teléfono y el instante en que finalizó, de modo que se operó una resta entre estas dos cifras y se obtuvo el tiempo de respuesta en cada caso.

En cuanto a las mediciones correspondientes a la utilización de memoria, éstas se obtuvieron analizando el periodo de tiempo que tardó en completarse una tarea y calculando la diferencia entre la cantidad de memoria ocupada en el instante en que la tarea finalizó y la cantidad de memoria al momento de iniciar la misma tarea.

Por otro lado, para las mediciones de utilización de CPU, haciendo uso del lapso de tiempo que tardó una tarea en ser completada, se identificó el pico más elevado registrado en dicho periodo que se interpreta como el uso máximo de CPU en la ejecución de la respectiva tarea.

Por último, las mediciones de la utilización de batería se obtuvieron del reporte generado por Little Eye, en el que aparece el total de mAh consumidos durante el periodo del monitoreo a la aplicación. En este sentido, se llevaron a cabo 4 intentos, donde cada uno de ellos tardó 5 minutos y 5 segundos; al finalizar cada intento, se registró el total de batería consumida y se calculó el promedio de dicho valor.

Cuadro 28. Evaluación de comportamiento temporal en NewsApp basada en web, métrica: tiempo de respuesta.

Aplicación	NewsApp Basada en web				
Subcaracterística	Comportamiento temporal			Métrica: Tiempo de respuesta	
Mediciones					
Tiempo de inicio (segundos)	Primer intento	Segundo intento	Tercer intento	Cuarto intento	Promedio
	4	6	5	4	4.75
Tiempo de respuesta a una tarea (segundos)	Tarea	Noticias Uno	Acac	Ciencia y Tecnología	Sgoliver.net Blog
	Selección de la fuente RSS	3	7	5	6
	Selección de un titular de noticia	3	5	6	4
	Volver atrás, a la pantalla B	3	2	3	3
	Selección de un titular de noticia	1	5	3	4
	Volver atrás, a la pantalla B	3	4	3	3
Tiempo de respuesta medio a una tarea (segundos)	Volver atrás, a la pantalla A	5	3	3	3
	Selección de la fuente RSS	5.25			
	Selección de un titular de noticia	3.9			
	Volver atrás, a la pantalla B	2.9			
	Volver atrás, a la pantalla A	3.5			

Fuente: Autores del proyecto

Cuadro 29. Evaluación de comportamiento temporal en NewsApp basada en web, métrica tiempo de respuesta del peor caso.

Aplicación	NewsApp Basada en web		
Subcaracterística	Comportamiento temporal	Métrica: Tiempo de respuesta	Peor caso
Mediciones			
Tiempo de respuesta a una tarea(segundos)	Tarea	Min. de Educación	
		Primer intento	Segundo intento
	Selección de la fuente RSS	15	23
	Selección de un titular de noticia	21	7
	Volver atrás, a la pantalla B	6	7
	Selección de un titular de noticia	27	6
	Volver atrás, a la pantalla B	11	8
	Volver atrás, a la pantalla A	12	13
Tiempo de respuesta medio a una tarea (segundos)	Selección de la fuente RSS	19	
	Selección de un titular de noticia	15.25	
	Volver atrás, a la pantalla B	8	
	Volver atrás, a la pantalla A	12.5	

Fuente: Autores del proyecto.

Cuadro 30. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de memoria.

Aplicación	NewsApp Basada en web				
Subcaracterística	Utilización de recursos			Métrica: Memoria	
Mediciones					
Consumo de memoria RAM en el inicio de la aplicación (kB)	Primer intento	Segundo intento	Tercer intento	Cuarto intento	Promedio
		11044	6536	7016	11044
Consumo de memoria RAM por tarea (kB)	Tarea	Noticias Uno	Acac	Ciencia y Tecnología	Sgoliver.net Blog
	Selección de la fuente RSS	28844	8824	96	24
	Selección de un titular de noticia	18472	28	52	1368
	Volver atrás, a la pantalla B	272	4	4	4
	Selección de un titular de noticia	76	2576	12	548
	Volver atrás, a la pantalla B	12	4	12	8
	Volver atrás, a la pantalla A	272	8	8	4
Consumo promedio de memoria RAM por tarea (kB)	Selección de la fuente RSS				9447
	Selección de un titular de noticia				2891.5
	Volver atrás, a la pantalla B				40
	Volver atrás, a la pantalla A				73
Peak Native memory (MB)	153.70				

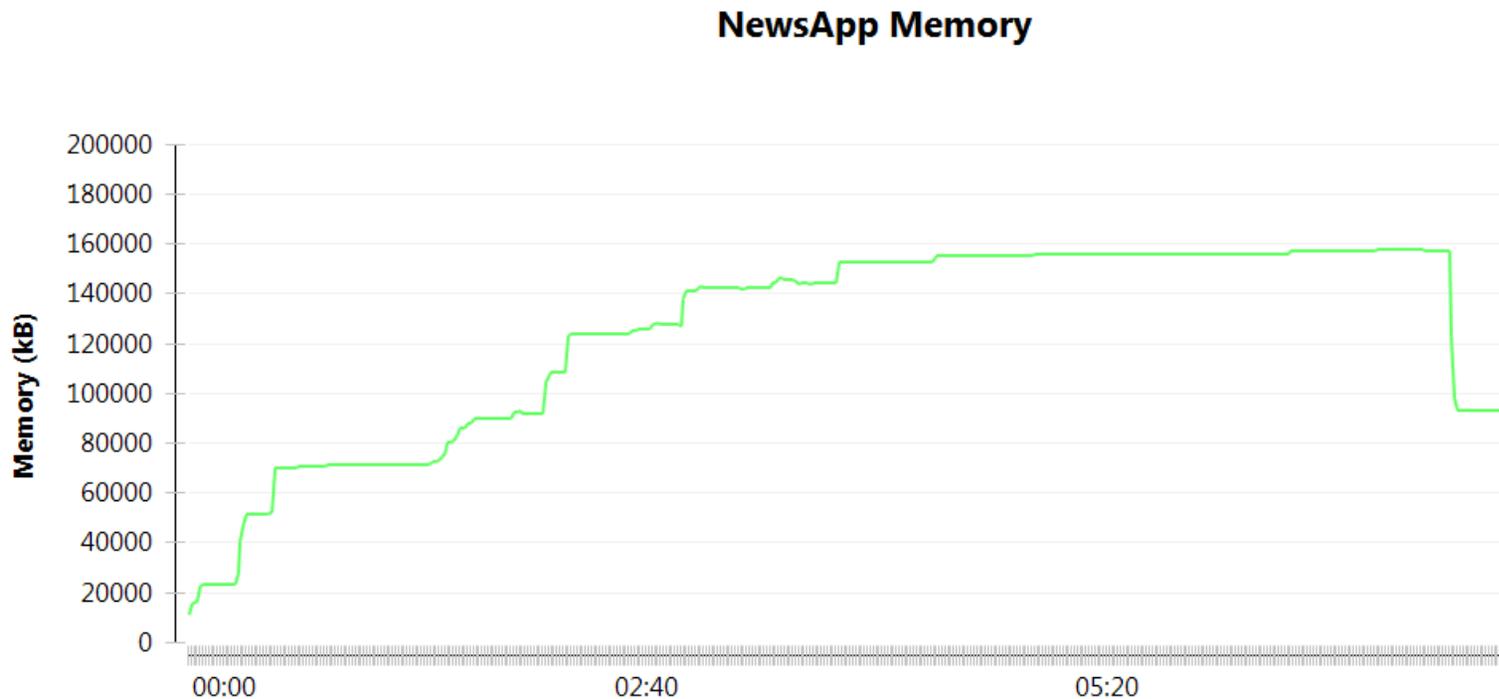
Fuente: Autores del proyecto.

Cuadro 31. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de memoria del peor caso.

Aplicación	NewsApp Basada en web		
Subcaracterística	Utilización de recursos	Métrica: Memoria	Peor caso
Mediciones			
Consumo de memoria RAM por tarea (kB)	Tarea	Min. de Educación	
		Primer intento	Segundo intento
	Selección de la fuente RSS	18616	20696
	Selección de un titular de noticia	33756	33756
	Volver atrás, a la pantalla B	8	8
	Selección de un titular de noticia	31832	28741
	Volver atrás, a la pantalla B	20	8
	Volver atrás, a la pantalla A	1080	1346
Peor caso promedio de consumo de memoria RAM por tarea (kB)	Selección de la fuente RSS	19656	
	Selección de un titular de noticia	32021	
	Volver atrás, a la pantalla B	11	
	Volver atrás, a la pantalla A	1217	

Fuente: Autores del proyecto.

Figura 32. Utilización máxima de memoria en NewsApp basada en web.



Fuente: Little Eye

La figura 32, generada por Little Eye, corresponde al uso de memoria RAM por la aplicación basada en web, durante el periodo de tiempo que tardó el monitoreo. El eje Y corresponde a la cantidad de memoria RAM utilizada en kB, contra el eje X que representa el tiempo transcurrido en segundos. La curva verde describe la utilización de memoria que hizo la aplicación.

Cuadro 32. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de CPU.

Aplicación	NewsApp Basada en web				
Subcaracterística	Utilización de recursos			Métrica: CPU	
Mediciones					
Uso máximo de CPU en el inicio de la aplicación (%)	Primer intento	Segundo intento	Tercer intento	Cuarto intento	Promedio
	35	58	58	79	57.5
Uso máximo de CPU en la ejecución de una tarea (%)	Tarea	Noticias Uno	Acac	Ciencia y Tecnología	Sgoliver.net Blog
	Selección de la fuente RSS	36	46	38	32
	Selección de un titular de noticia	41	39	43	30
	Volver atrás, a la pantalla B	35	34	33	33
	Selección de un titular de noticia	26	42	39	34
	Volver atrás, a la pantalla B	35	37	37	39
	Volver atrás, a la pantalla A	36	39	32	33
Promedio de uso máximo de CPU en la ejecución de una tarea (%)	Selección de la fuente RSS	38			
	Selección de un titular de noticia	36.75			
	Volver atrás, a la pantalla B	35.4			
	Volver atrás, a la pantalla A	35			
Promedio de CPU utilizada por la aplicación (%)	12,58%				

Fuente: Autores del proyecto.

Los datos del cuadro 32 fueron tomados de la figura 33 de Little Eye.

Cuadro 33. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de CPU del peor caso.

Aplicación	NewsApp Basada en web		
Subcaracterística	Utilización de recursos	Métrica: CPU	Peor caso
Mediciones			
Uso máximo de CPU en la ejecución de una tarea (%)	Tarea	Min. de Educación	
		Primer intento	Segundo intento
	Selección de la fuente RSS	49	55
	Selección de un titular de noticia	53	53
	Volver atrás, a la pantalla B	53	53
	Selección de un titular de noticia	52	53
	Volver atrás, a la pantalla B	47	51
	Volver atrás, a la pantalla A	56	59
Promedio de uso máximo de CPU en la ejecución de una tarea (%)	Selección de la fuente RSS	52	
	Selección de un titular de noticia	52.75	
	Volver atrás, a la pantalla B	51	
	Volver atrás, a la pantalla A	57.5	

Fuente: Autores del proyecto.

Cuadro 34. Evaluación de utilización de recursos en NewsApp basada en web, métrica: utilización máxima de batería.

Aplicación	NewsApp Basada en web				
Subcaracterística	Utilización de recursos	Métrica: utilización máxima de batería			
Mediciones					
Total batería consumida durante el tiempo de la evaluación	Tarea	Consumo de batería (mAh)			
		Primer intento	Segundo intento	Tercer intento	Cuarto intento
	El tiempo de la evaluación fue 5 minutos y 5 segundos.	41.94	41.78	41.36	40.84

Fuente: Autores del proyecto.

Cuadro 35. Evaluación de comportamiento temporal en NewsApp nativa, métrica: tiempo de respuesta.

Aplicación	NewsApp Nativa				
Subcaracterística	Comportamiento temporal			Métrica: Tiempo de respuesta	
Mediciones					
Tiempo de inicio (segundos)	Primer intento	Segundo intento	Tercer intento	Cuarto intento	Promedio
	1	1	1	1	1
Tiempo de respuesta a un evento (segundos)	Tarea	Noticias Uno	Acac	Ciencia y Tecnología	Sgoliver.net Blog
	Selección de la fuente RSS	2	4	4	5
	Selección de un titular de noticia	2	3	2	2
	Volver atrás, a la pantalla B	2	1	1	2
	Selección de un titular de noticia	2	3	2	2
	Volver atrás, a la pantalla B	2	2	3	3
	Volver atrás, a la pantalla A	2	2	2	2
Tiempo de respuesta medio (segundos)	Selección de la fuente RSS	3.75			
	Selección de un titular de noticia	2.25			
	Volver atrás, a la pantalla B	2			
	Volver atrás, a la pantalla A	2			

Fuente: Autores del proyecto.

Cuadro 36. Evaluación de comportamiento temporal en NewsApp nativa, métrica: tiempo de respuesta del peor caso.

Aplicación	NewsApp Nativa		
Subcaracterística	Comportamiento temporal	Métrica: Tiempo de respuesta	Peor caso
Mediciones			
Tiempo de respuesta a un evento (segundos)	Tarea	Min. de Educación	
		Primer intento	Segundo intento
	Selección de la fuente RSS	10	10
	Selección de un titular de noticia	6	2
	Volver atrás, a la pantalla B	2	2
	Selección de un titular de noticia	3	3
	Volver atrás, a la pantalla B	2	2
Tiempo de respuesta medio (segundos)	Volver atrás, a la pantalla A	2	2
	Selección de la fuente RSS	10	
	Selección de un titular de noticia	3.5	
	Volver atrás, a la pantalla B	2	
	Volver atrás, a la pantalla A	2	

Fuente: Autores del proyecto.

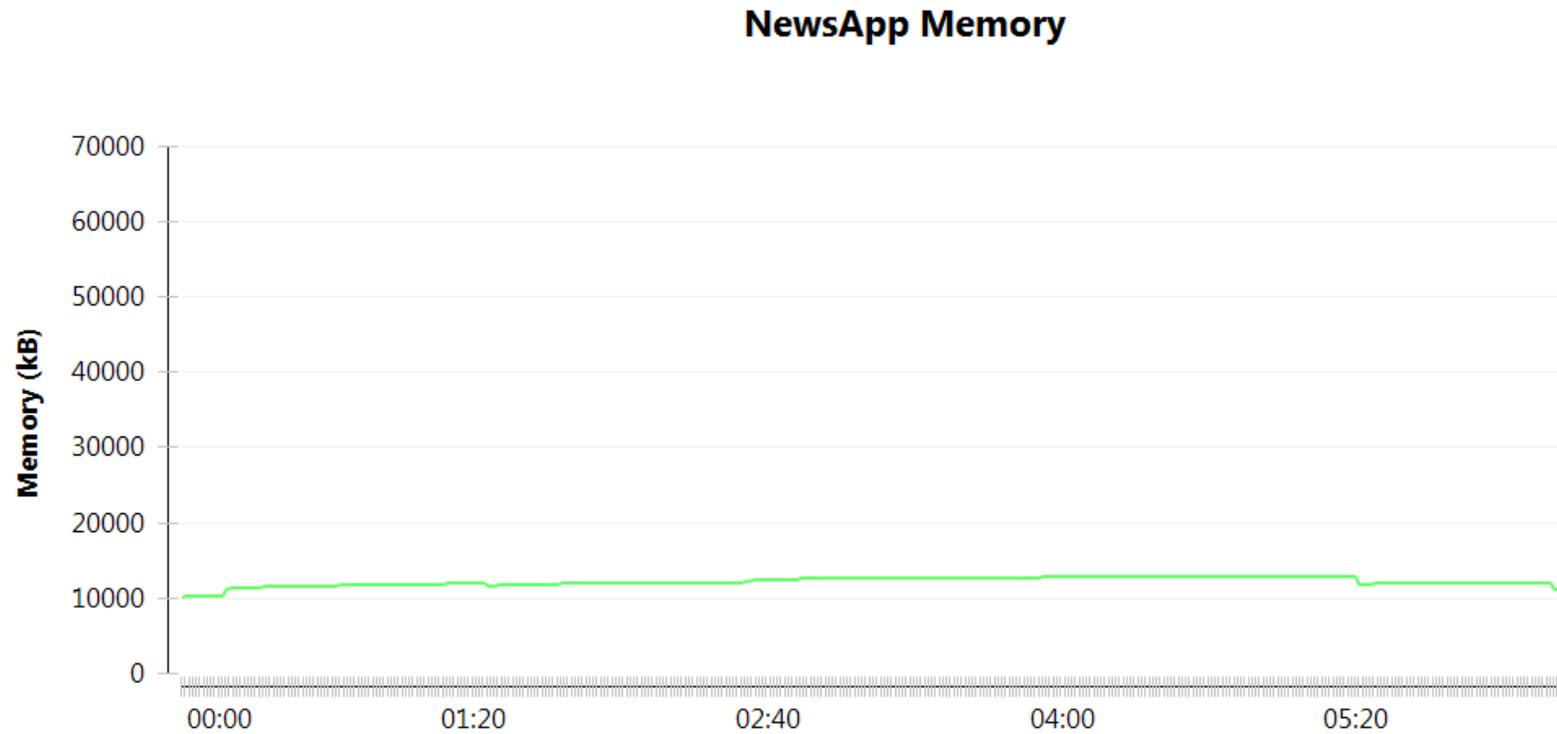
Cuadro 37. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de memoria.

Aplicación	NewsApp Nativa				
Subcaracterística	Utilización de recursos			Métrica: utilización máxima de memoria.	
Mediciones					
Consumo de memoria RAM en el inicio de la aplicación (kB)	Primer intento	Segundo intento	Tercer intento	Cuarto intento	Promedio
	308	308	576	348	385
Consumo de memoria RAM por tarea (kB)	Tarea	Noticias Uno	Acac	Ciencia y Tecnología	Sgoliver.net Blog
	Selección de la fuente RSS	924	64	136	0
	Selección de un titular de noticia	248	136	8	112
	Volver atrás, a la pantalla B	0	0	0	0
	Selección de un titular de noticia	144	100	0	0
	Volver atrás, a la pantalla B	0	0	0	0
	Volver atrás, a la pantalla A	0	0	0	0
Consumo promedio de memoria RAM por tarea (kB)	Selección de la fuente RSS	281			
	Selección de un titular de noticia	96			
	Volver atrás, a la pantalla B	0			
	Volver atrás, a la pantalla A	0			
Peak Native memory (MB)	34.20				

Fuente: Autores del proyecto.

Los datos del cuadro 37 fueron tomados de la figura 34.

Figura 34. Utilización máxima de Memoria en NewsApp nativa.



Fuente: Little Eye.

La figura 34, generada por Little Eye, corresponde al uso de memoria RAM por la aplicación nativa, durante el periodo de tiempo que tardó el monitoreo. El eje Y corresponde a la cantidad de memoria RAM utilizada en kB, contra el eje X que representa el tiempo transcurrido en segundos. La curva verde describe la utilización de memoria que hizo la aplicación.

Cuadro 38. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de memoria del peor caso.

Aplicación	NewsApp Nativa		
Subcaracterística	Utilización de recursos	Métrica: Memoria	Peor caso
Mediciones			
Memoria RAM consumida por evento (kB)	Tarea	Min. de Educación	
		Primer intento	Segundo intento
	Selección de la fuente RSS	196	220
	Selección de un titular de noticia	144	144
	Volver atrás, a la pantalla B	16	16
	Selección de un titular de noticia	108	132
	Volver atrás, a la pantalla B	4	8
	Volver atrás, a la pantalla A	0	0
Peor caso promedio de consumo de memoria RAM por tarea (kB)	Selección de la fuente RSS	208	
	Selección de un titular de noticia	132	
	Volver atrás, a la pantalla B	11	
	Volver atrás, a la pantalla A	0	

Fuente: Autores del proyecto

Cuadro 39. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de CPU.

Aplicación	NewsApp Nativa					
Subcaracterística	Utilización de recursos				Métrica: CPU	
Mediciones						
Uso máximo de CPU en el inicio de la aplicación (%)	Primer intento	Segundo intento	Tercer intento	Cuarto intento	Promedio	
	2	3	3	2	2.5	
Uso máximo de CPU en la ejecución de una tarea (%)	Tarea	Noticias Uno	Min. de Educación	Acac	Ciencia y Tecnología	Sgoliver.net Blog
	Selección de la fuente RSS	10	31	16	8	8
	Selección de un titular de noticia	4	3	6	1	2
	Volver atrás, a la pantalla B	4	3	2	4	3
	Selección de un titular de noticia	2	2	1	2	4
	Volver atrás, a la pantalla B	1	3	3	5	2
	Volver atrás, a la pantalla A	3	5	3	5	1
Promedio de uso máximo de CPU en la ejecución de una tarea (%)	Selección de la fuente RSS	10.5				
	Selección de un titular de noticia	2.75				
	Volver atrás, a la pantalla B	3				
	Volver atrás, a la pantalla A	3				
Promedio de CPU utilizada por la aplicación (%)	0.73					

Fuente: Autores del proyecto.

Los datos del cuadro 39 fueron tomados de la figura 35.

Figura 35. Utilización máxima de CPU en NewsApp nativa.



Fuente: Little Eye

La figura 35, generada por Little Eye, corresponde al uso de CPU por la aplicación nativa, durante el periodo de tiempo que tardó el monitoreo. El eje Y corresponde al porcentaje de CPU utilizada, contra el eje X que representa el tiempo transcurrido en segundos. La curva verde describe la utilización de CPU que hizo la aplicación.

Cuadro 40. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de CPU del peor caso.

Aplicación	NewsApp Nativa		
Subcaracterística	Utilización de recursos	Métrica: CPU	Peor caso
Mediciones			
CPU utilizada por evento (%)	Tarea	Min. de Educación	
		Primer intento	Segundo intento
	Selección de la fuente RSS	31	29
	Selección de un titular de noticia	3	3
	Volver atrás, a la pantalla B	3	3
	Selección de un titular de noticia	1	2
	Volver atrás, a la pantalla B	2	2
	Volver atrás, a la pantalla A	5	3
Promedio de CPU utilizada por evento (%)	Selección de la fuente RSS	30	
	Selección de un titular de noticia	2.25	
	Volver atrás, a la pantalla B	2.5	
	Volver atrás, a la pantalla A	4	

Fuente: Autores del proyecto.

Cuadro 41. Evaluación de utilización de recursos en NewsApp nativa, métrica: utilización máxima de batería.

Aplicación	NewsApp Basada Nativa					
Subcaracterística	Utilización de recursos	Métrica: Batería				
Mediciones						
Promedio batería consumida durante el tiempo de la evaluación	Tarea	Consumo de batería (mAh)				
	El tiempo de la evaluación fue 5 minutos y 5 segundos.	Primer intento	Segundo intento	Tercer intento	Cuarto intento	Promedio
		25.43	26.06	25.36	26.74	25.9

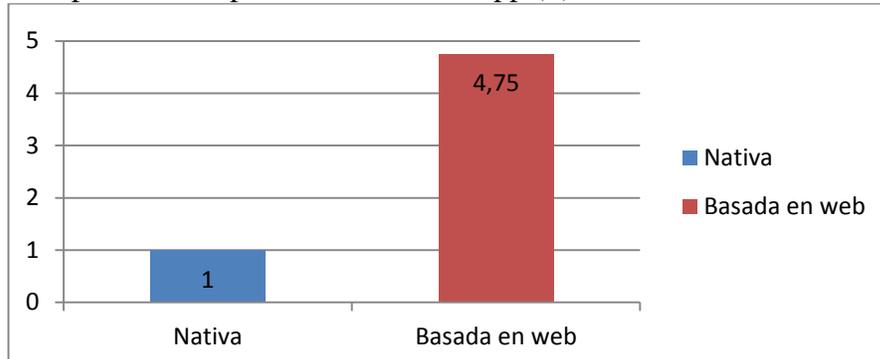
Fuente: Autores del proyecto.

4.1.4.4 Análisis de resultados. Conforme a las mediciones propuestas en las métricas para esta evaluación, se elaboraron las gráficas respectivas.

Subcaracterística: Comportamiento Temporal

Métrica: Tiempo de respuesta

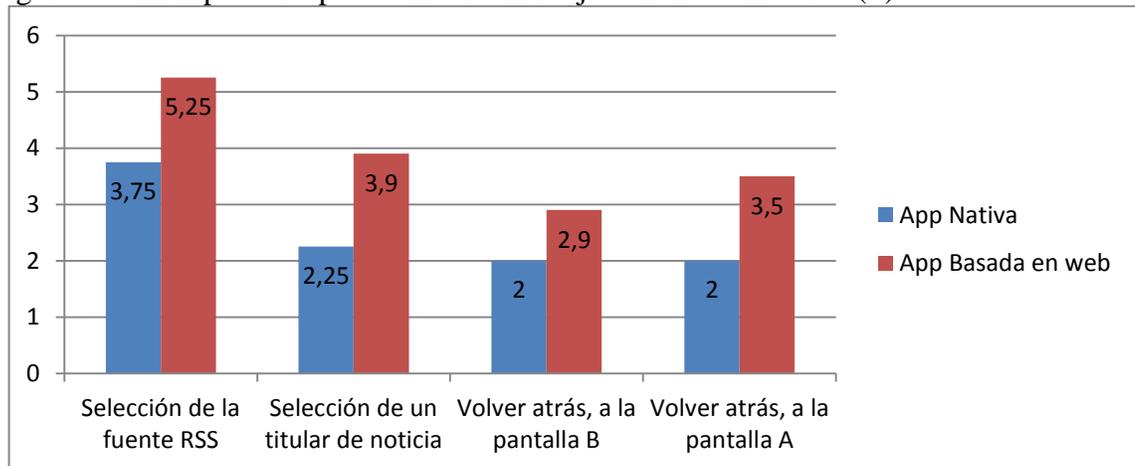
Figura 36. Tiempo de inicio promedio de NewsApp (S).



Fuente: Autores del proyecto.

La figura 36 muestra el tiempo promedio de la aplicación NewsApp en ambas versiones, y se observa la notable diferencia entre tales tiempos. NewsApp basada en web necesitó 3,75 segundos más para su inicio.

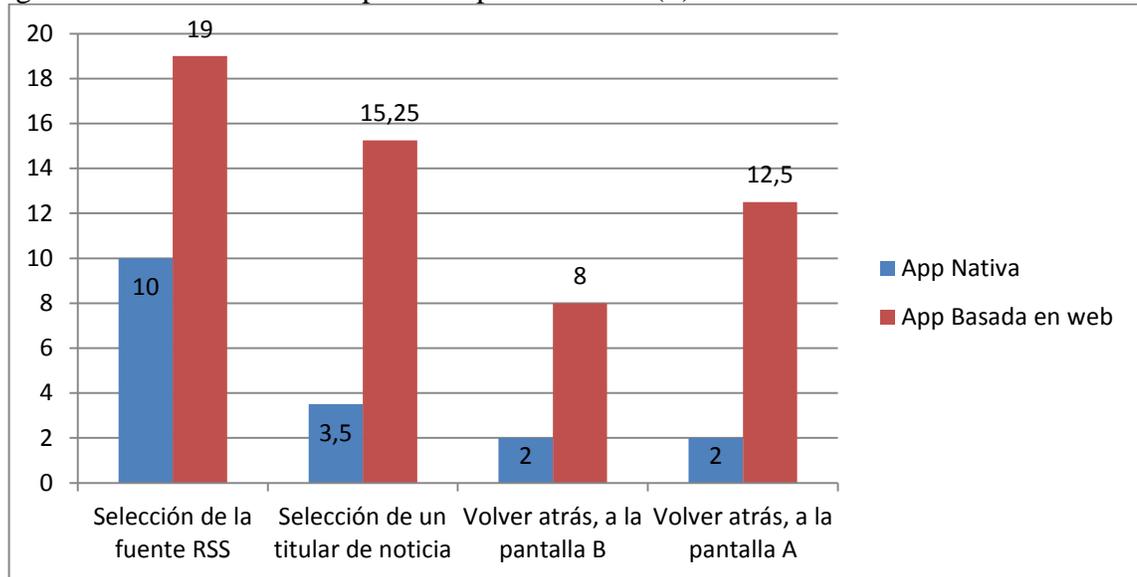
Figura 37. Tiempo de respuesta medio en la ejecución de una tarea (S).



Fuente: Autores del proyecto.

Como se puede evidenciar en la figura 37, en la ejecución de todas las tareas, NewsApp nativa siempre invirtió menos tiempo.

Figura 38. Peor caso de tiempo de respuesta medio (S).



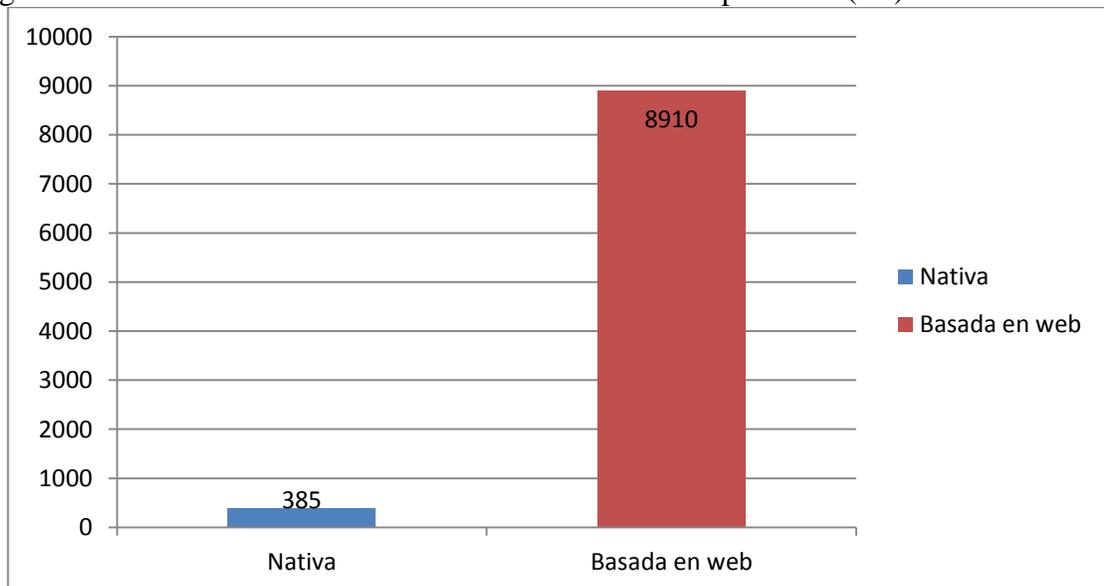
Fuente: Autores del proyecto.

En el peor caso de tiempo de respuesta medio, como se muestra en la figura 38, NewsApp basada en web en todas las tareas empleó más tiempo en ejecutarlas. La tarea que más requiere tiempo para ser ejecutada en NewsApp basada en web, es la selección de un titular de noticia.

Subcaracterística: Utilización de recursos

Métrica: Utilización máxima de memoria RAM

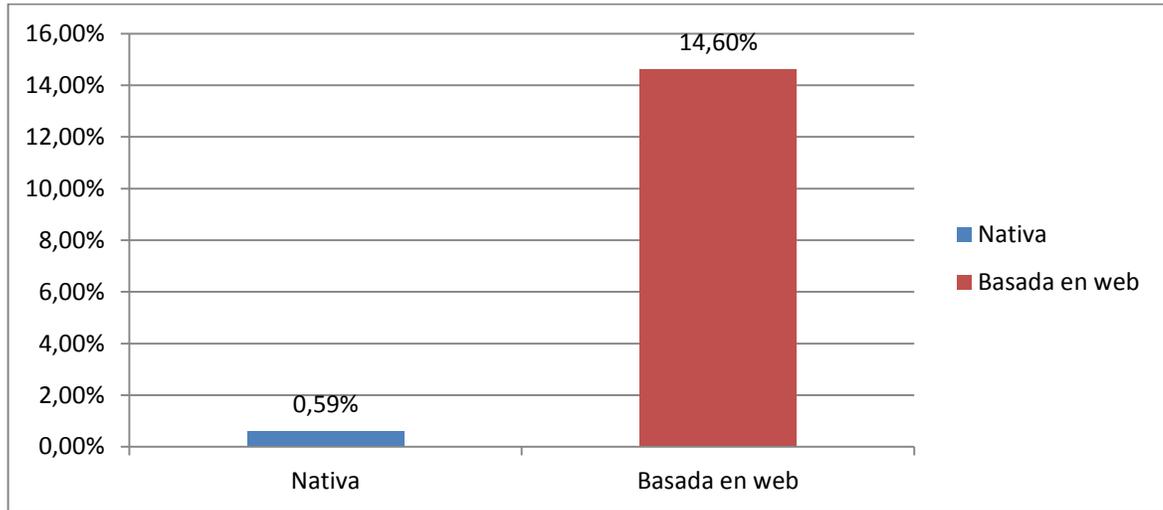
Figura 39. Consumo de memoria RAM en el inicio de la aplicación (kB)



Fuente: Autores del proyecto.

La figura 39 presenta el consumo de memoria RAM en el inicio de la aplicación, pero, las cifras están dadas en kB, razón por la que se elaboró la figura 40 que muestra dicho consumo en porcentaje.

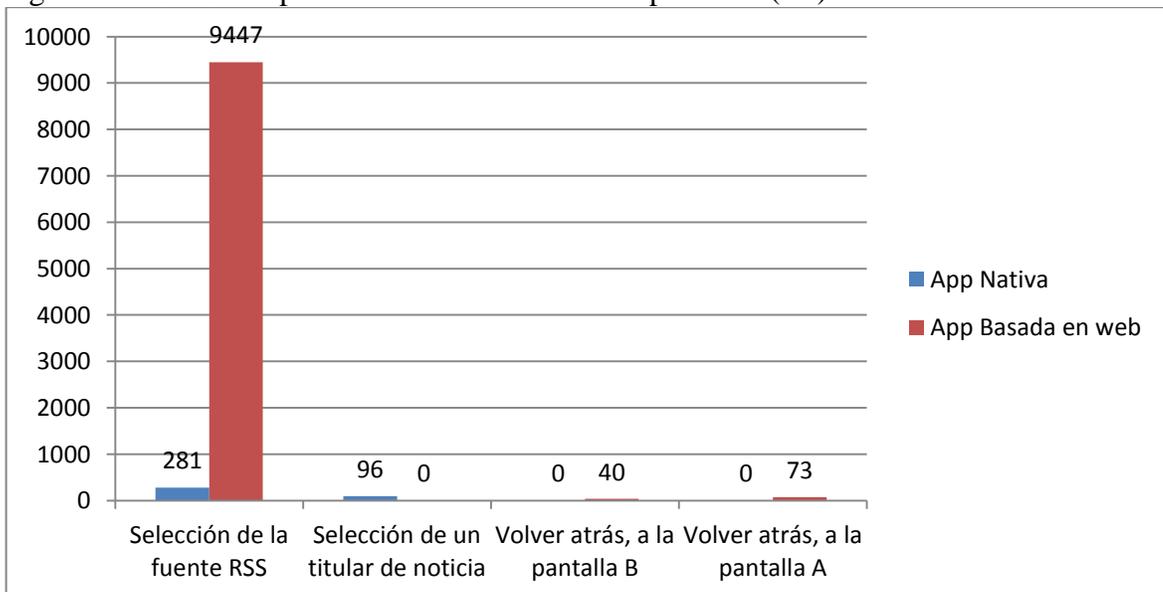
Figura 40. Consumo de memoria RAM en el inicio de la aplicación con respecto al límite de memoria Dalvik.



Fuente: Autores del proyecto.

La figura 40 muestra cómo al iniciar la aplicación, la versión basada en web consumió 14,6% memoria RAM de las 64 MB (65536 kB) asignada por Dalvik, mientras que la versión nativa no superó el 1% de este límite de memoria.

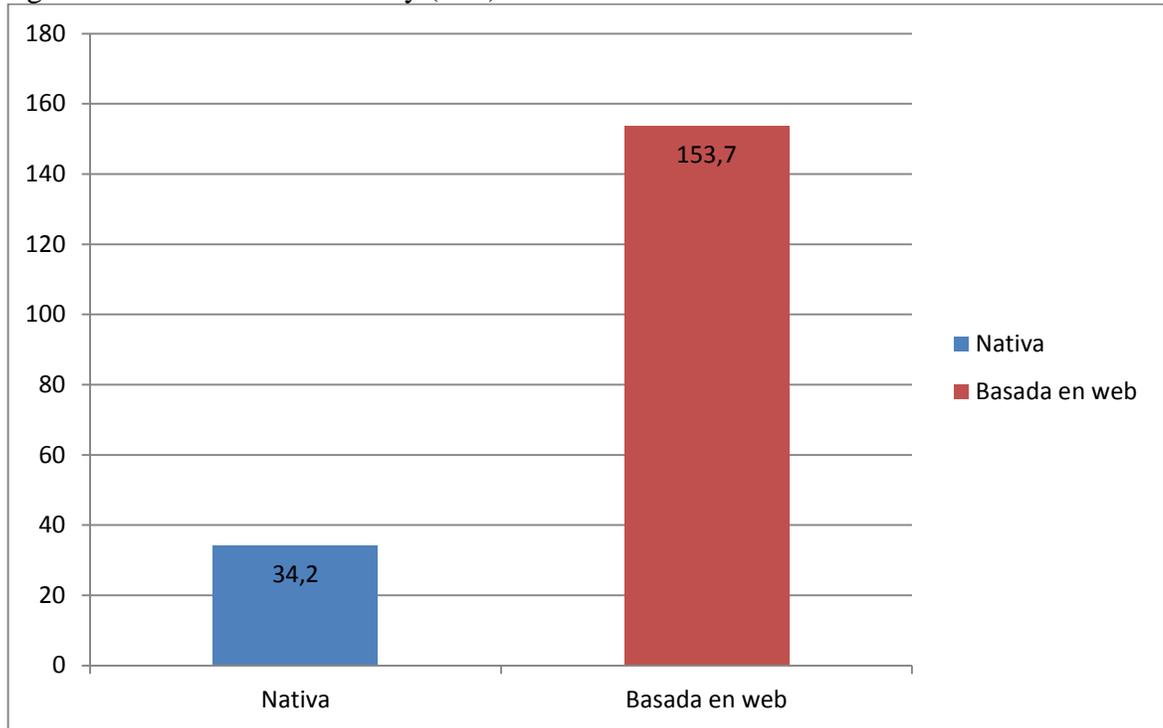
Figura 41. Consumo promedio de memoria RAM por tarea (kB).



Fuente: Autores del proyecto.

En la mayoría de las tareas, NewsApp basada en web ocupó más memoria RAM que la versión nativa. La selección de la fuente RSS es la tarea que ocupa más memoria RAM en ambas versiones de la aplicación, sin embargo, la versión nativa se comporta mejor ejecutando esta tarea (ver figura 41).

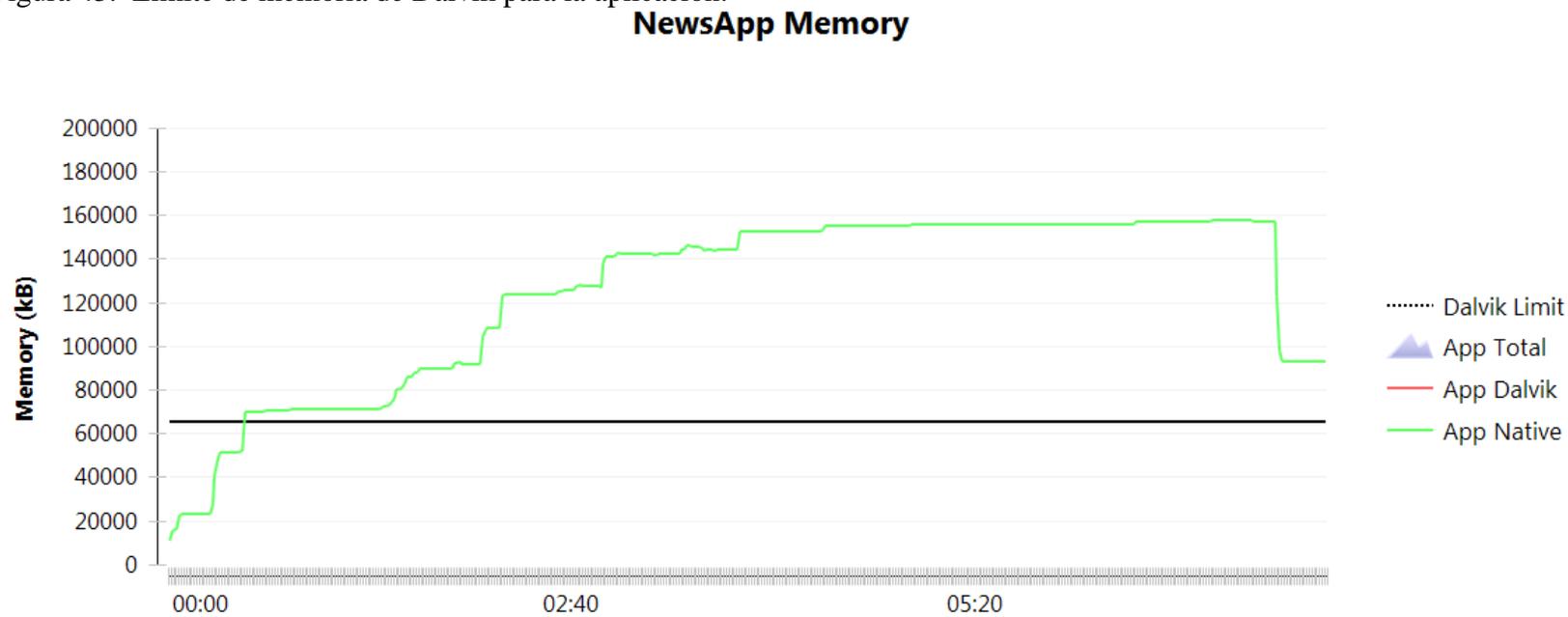
Figura 42. Peak Native Memory (MB).



Fuente: Autores del proyecto.

Se observa en la figura 42 que NewsApp basada en web superó el límite de memoria que Dalvik le asignó que para este caso Little Eye indicó que era de 64 MB.

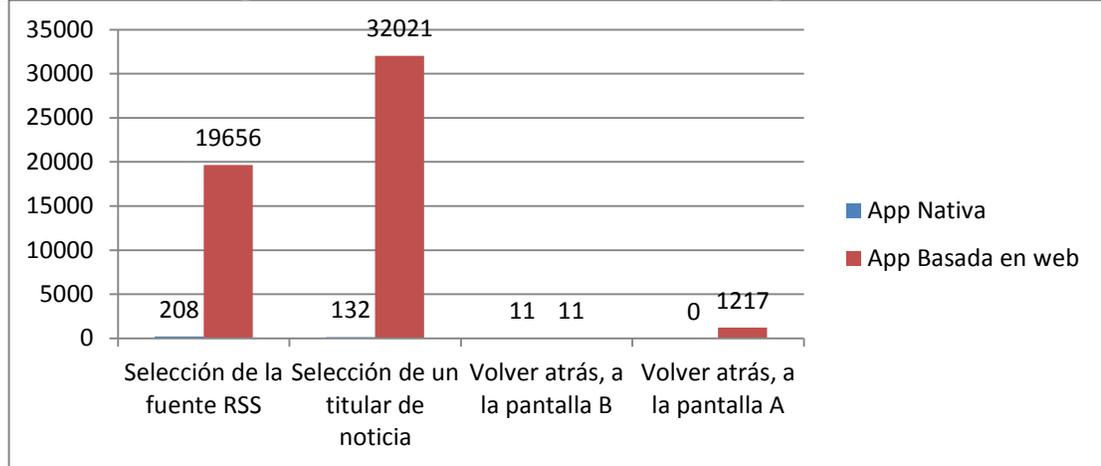
Figura 43. Límite de memoria de Dalvik para la aplicación.



06:50 -- Dalvik Limit: 65536kB, App Total: 173628kB, App Dalvik: 16796kB, App Native: 156832kB

Fuente: Little Eye

Figura 44. Peor caso promedio de consumo de memoria RAM por tarea (kB).



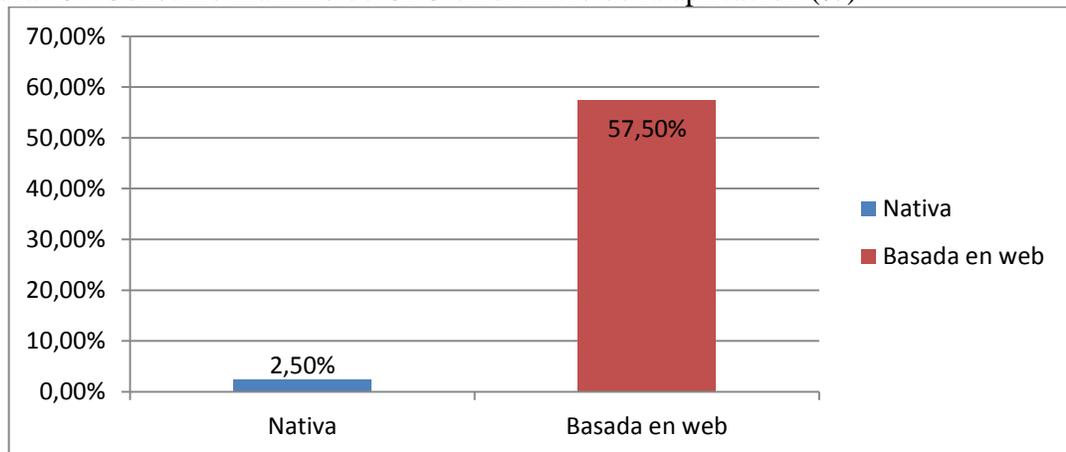
Fuente: Autores del proyecto.

La figura 44 indica que en 3 de las 4 tareas, NewsApp basada en web, requirió una cantidad de tiempo mucho mayor para ejecutarlas; únicamente en “Volver atrás, a la pantalla B” ambas versiones de la aplicación necesitaron el mismo tiempo. Para NewsApp basada en web la tarea que tardó más tiempo ejecutándose, fue “Selección de un titular de noticia”.

Subcaracterística: Utilización de recursos

Métrica: Utilización máxima de CPU

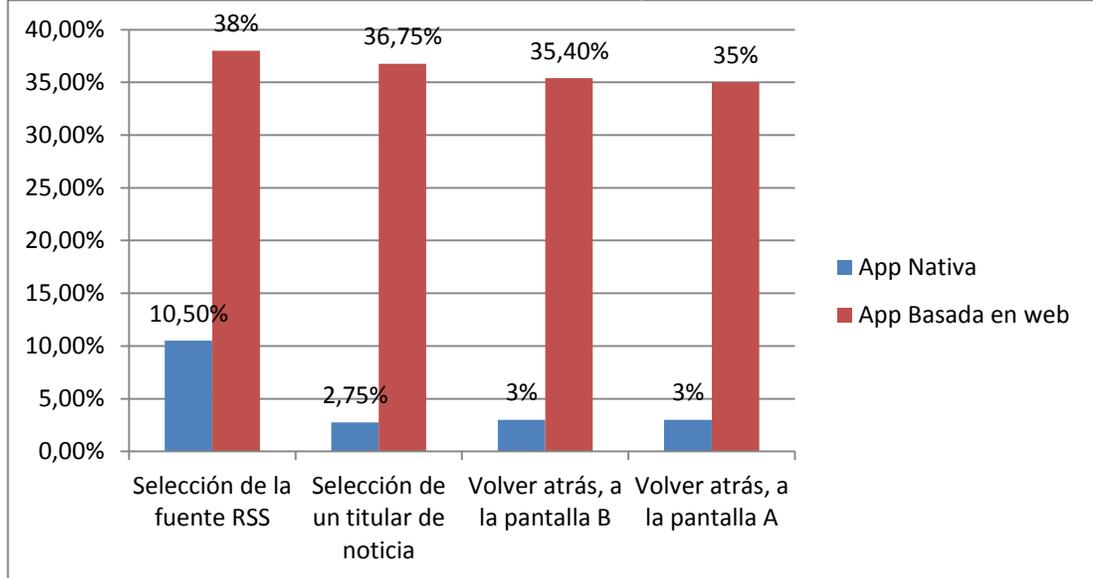
Figura 45. Consumo máximo de CPU en el inicio de la aplicación (%).



Fuente: Autores del proyecto.

Al iniciar la aplicación el consumo máximo de CPU que se registró en la versión basada en web superó en un 55% al consumo de la nativa, como se indica en la figura 45.

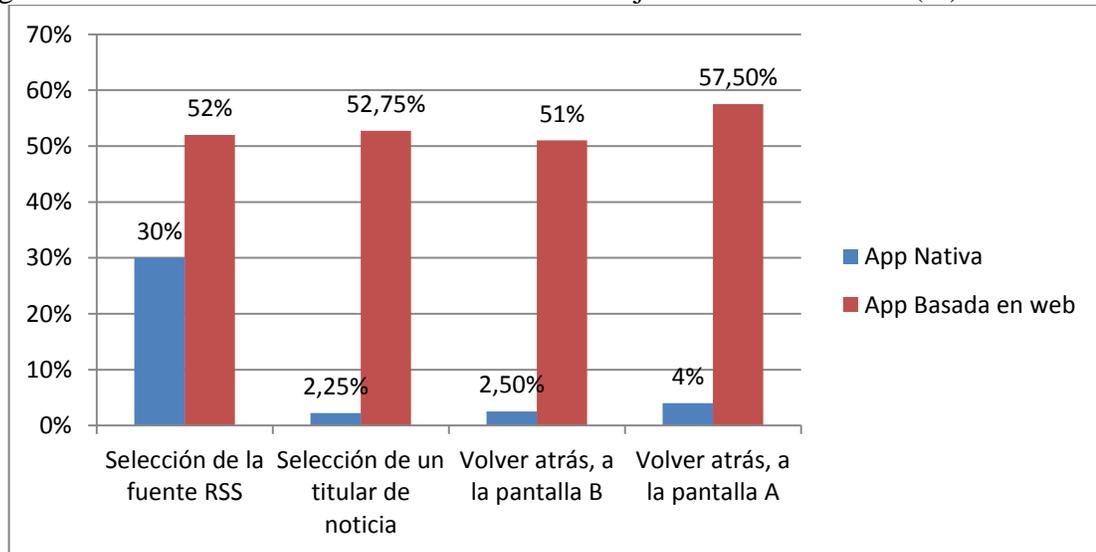
Figura 46. Promedio de uso máximo de CPU en la ejecución de una tarea (%).



Fuente: Autores del proyecto.

La versión basada en web de NewsApp, en la ejecución de todas las tareas alcanzó a usar un máximo de CPU que superó el 30%, mientras que la nativa en ninguna de las mismas tareas superó el 11% (ver figura 46).

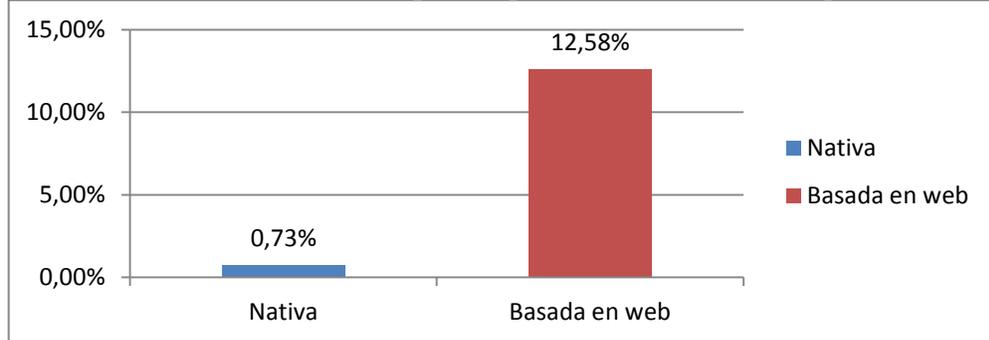
Figura 47. Peor caso de uso máximo de CPU en la ejecución de una tarea (%).



Fuente: Autores del proyecto.

La figura 47 indica que la versión basada en web de NewsApp, en la ejecución de todas las tareas alcanzó a usar un máximo de CPU que superó el 50%, mientras que la nativa en ninguna de las mismas tareas superó el 30%.

Figura 48. Promedio de CPU utilizada por la aplicación durante el tiempo evaluado (%).



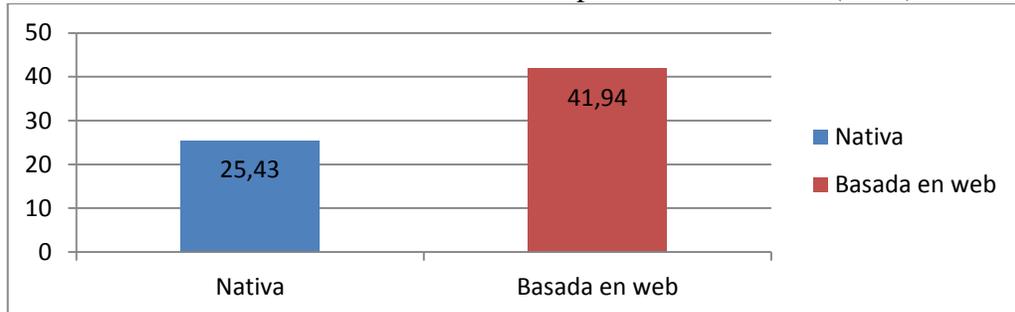
Fuente: Autores del proyecto.

En la figura 48 se aprecia que NewsApp nativa no alcanzó a usar un promedio de CPU del 1% durante el tiempo de la evaluación, mientras que la versión basada en web, utilizó el 12,58% de CPU.

Subcaracterística: Utilización de recursos

Métrica: Utilización máxima de batería

Figura 49. Total batería consumida durante el tiempo de la evaluación (mAh)



Fuente: Autores del proyecto.

En la figura 49, NewsApp nativa durante el tiempo que tardó la evaluación consumió 25,43 mAh, es decir, 16,51 mAh menos que la versión basada en web.

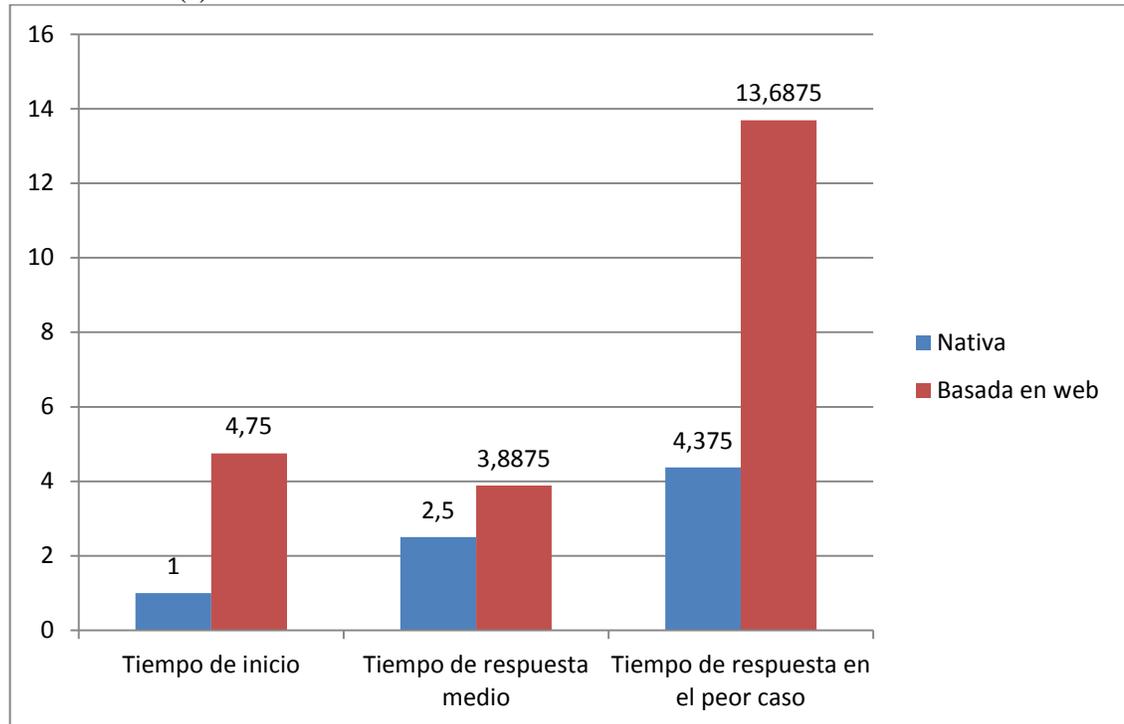
4.1.4.5 Conclusiones de la Evaluación de NewsApp. A continuación se organizan los resultados que permitieron realizar las conclusiones de la evaluación como se muestra en el cuadro 42.

Cuadro 42. Resultados de la evaluación de NewsApp.

NewsApp				Basada en web	Nativa	Criterio
Característica	Subcaracterística	Métrica	Mediciones	Valor medido	Valor medido	
Eficiencia	Comportamiento temporal	Tiempo de respuesta (S)	Tiempo de inicio	4,75	1	0 < T Cuanto más pronto Es el mejor.
			Tiempo de respuesta medio	3,8875	2,5	
			Tiempo de respuesta en el peor caso	13,6875	4,375	
	Utilización de recursos	Utilización máxima de memoria (kB)	Utilización de memoria en el inicio	8910	3,85	0 ≤ M Cuanto más pequeño es el mejor
			Utilización de memoria por tarea	3112,875	94,25	
			Utilización de memoria por tarea peor caso	13226,25	87,75	
		Utilización Máxima de CPU (%)	Utilización de CPU en el inicio	57,5	2,5	0 ≤ C Cuanto más pequeño es el mejor
			Utilización de CPU por tarea	36,2875	4,8125	
			Utilización de CPU por tarea peor caso	53,31	9,6875	
	Utilización máxima de batería (mAh)	Utilización de batería durante la evaluación	41,48	25,9	0 ≤ B Cuanto más pequeño es el mejor	

Fuente: Autores del proyecto.

Figura 50. Tiempo de respuesta en NewsApp nativa VS. Tiempo de respuesta en NewsApp basada en web (s).



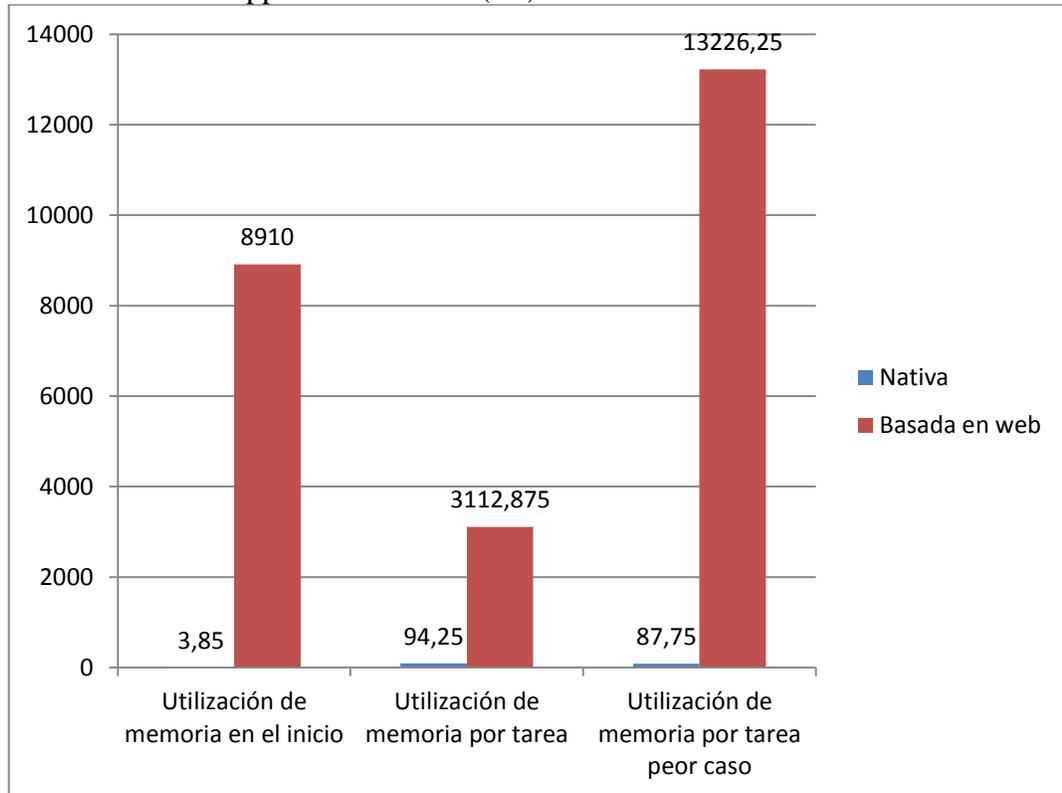
Fuente: Autores del proyecto.

En la figura 50 se observa claramente que la versión nativa de NewsApp tiene un mejor tiempo de respuesta con respecto a la basada en web. En el caso específico del tiempo de inicio de la aplicación, que es un factor altamente importante de acuerdo a lo explicado en el apartado 4.1.1.1 de este trabajo, NewsApp basada en web tarda 3,75 segundos más que la nativa en iniciar, es decir, ésta última es aproximadamente 4 veces más rápida que la basada en web; a propósito, según Yorio¹⁰¹, si la aplicación tarda 6 segundos en iniciar, se puede considerar inaceptable.

Se nota que el tiempo de respuesta medio no supera los 2 segundos de diferencia entre ambas versiones de NewsApp; además, se puede decir que estos tiempos son aceptables teniendo en cuenta lo que se concluyó en el estudio de Apigee, donde se indica que un tiempo de respuesta inferior a los 5 segundos es tolerable para el usuario; ahora, la versión nativa de NewsApp se mantuvo por debajo de dicho margen de tiempo aún en el peor caso, mientras que la versión basada en web lo sobrepasó notablemente.

¹⁰¹ YORIO, Op. cit., p.24.

Figura 51. Utilización máxima de memoria en NewsApp nativa VS. Utilización máxima de memoria en NewsApp basada en web (kB).

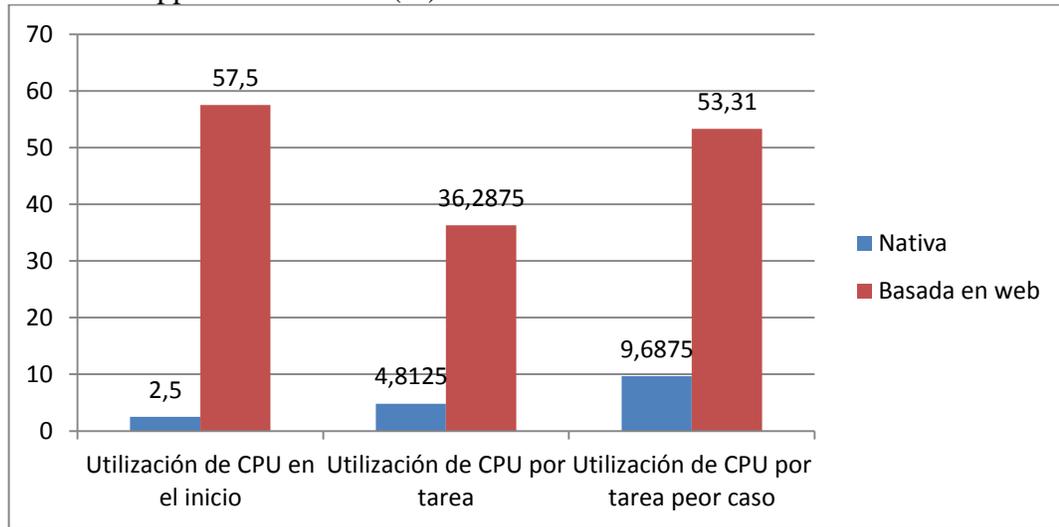


Fuente: Autores del proyecto.

Según la figura 51, la utilización máxima de memoria en el inicio de la aplicación, para el caso de NewsApp nativa

NewsApp nativa consumió menor cantidad de memoria RAM que la versión basada en web tanto al inicio de la misma, como también por tarea ejecutada y en el peor caso; en otras palabras, en los tres casos mencionados la utilización de memoria de NewsApp nativa es insignificante frente al uso de este recurso en la otra versión.

Figura 52. Utilización máxima de CPU en NewsApp nativa VS. Utilización máxima de CPU en NewsApp basada en web (%).

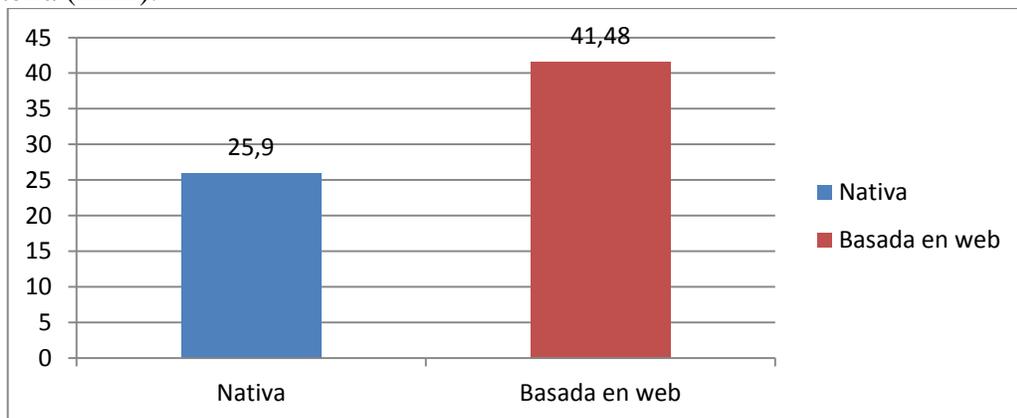


Fuente: Autores del proyecto.

En la figura 52 se aprecia que NewsApp basada en web tanto en el inicio de la aplicación como en el peor caso, hace una utilización de CPU por encima del 50%. Los picos de uso de CPU a los que se llegó en la versión basada en web son tan elevados que podrían generar problemas de rendimiento en el sistema.

Al iniciar la NewsApp nativa, se alcanzó a utilizar únicamente un 2,5% de CPU.

Figura 53. NewsApp nativa VS. NewsApp basada en web en cuanto a utilización máxima de batería (mAh).



Fuente: Autores del proyecto.

Como indica la figura 53, NewsApp nativa consumió menos cantidad de miliamperios que la versión basada en web, de modo que ésta última consumió 1,6 veces más de batería que la nativa.

Teniendo en cuenta que la herramienta Little Eye usa como modelo de comparación el Samsung Galaxy Nexus, que posee una batería de 1750 mAh, idéntica a la del Sony Xperia S usado en la evaluación, se determinó:

NewsApp basada en web consumió el 2,37% del total de la batería, mientras que NewsApp nativa utilizó el 1,48%.

CONCLUSIONES

La confrontación de las dos tecnologías de desarrollo de aplicaciones móviles: nativa y basada en web, fue necesario soportarla en directrices que hicieron del presente trabajo un documento técnico, serio y de contenido apoyado en la realización de un procedimiento que se basó en los criterios de los estándares internacionales ISO/IEC 14598 e ISO/IEC 9126, de los cuales el primero, proveyó el modelo para el proceso de evaluación, y el segundo se empleó como modelo de calidad del producto. De acuerdo a las características de calidad que plantea el modelo ISO/IEC 9126, y considerándolas en el contexto de los smartphones y las aplicaciones móviles, se determinó la eficiencia como única característica a evaluar.

En el desarrollo de aplicaciones móviles para Android se encuentran dos alternativas claramente identificadas: el desarrollo nativo, en el que se hace uso del SDK de Android, Eclipse como IDE y el plugin ADT, que son las oficialmente recomendadas por Google y además son open source, adecuadas para el ámbito académico. La segunda alternativa es el desarrollo basado en tecnologías web, para el que existe una vasta serie de opciones, sin embargo, el uso de jQuery Mobile para la construcción de interfaces gráficas, Phonegap como empaquetador y la utilización de Javascript para la lógica de programación, son las técnicas que predominan para el desarrollo de aplicaciones de este tipo.

En el proceso de evaluación se seleccionó la eficiencia, también denominada rendimiento, como única característica de calidad a evaluar porque los estudios de Work Quality Report¹⁰², SOASTA¹⁰³, Apigee¹⁰⁴ y Mobile Apps: What Consumers Really Need and Want¹⁰⁵, ponen de manifiesto la importancia de esta característica por encima de otras, puesto que los usuarios de las aplicaciones móviles exponen un inconformismo muy significativo en aspectos asociados con el rendimiento de ellas, además, particularmente el informe World Quality Report, indica cómo tradicionalmente se hacían grandes esfuerzos por lograr funcionalidad, no obstante, hoy en día el rendimiento de las aplicaciones es lo que más interesa, tanto así que se habla del rendimiento como “la clave del éxito”¹⁰⁶. Asimismo, Stephen Pierzchala, expresa: "El rendimiento es un factor crucial para proporcionar una experiencia de usuario de aplicaciones móviles de confianza, por lo que el rendimiento se debe considerar un factor clave en el proceso de diseño."¹⁰⁷

En la evaluación de la eficiencia se analizaron: el tiempo de respuesta, métrica de comportamiento temporal y utilización máxima de memoria, de CPU y de batería, métricas pertenecientes a la subcaracterística utilización de recursos. Los resultados de esta

¹⁰² CAPGEMINI, SOGETI y HEWLETT-PACKARD. Op. cit., p.25.

¹⁰³ SOASTA. Op. cit.

¹⁰⁴ APIGEE, Op. cit.

¹⁰⁵ COMPUWARE. Op.cit.

¹⁰⁶ CAPGEMINI, SOGETI y HEWLETT-PACKARD. Op. cit., p.24.

¹⁰⁷ COMPUWARE. Mobile Apps vs. Mobile Websites -- and the Winner Is? [En línea]. [Citado el 30 de agosto de 2013]. Disponible en Internet en: http://www.compuware.com/en_us/about/press-releases/2013/3/mobile-apps-vs-mobile-websites---and-the-winner-is.html

evaluación sugieren que: el tiempo de respuesta es menor en la aplicación nativa que en la versión basada en web y la utilización de recursos tanto de memoria como de CPU, también es menor en la versión nativa de la aplicación móvil. En cuanto a la utilización de batería, cabe resaltar que el uso de este recurso, en ambas versiones de la aplicación, es el que presenta una diferencia más pequeña.

Tras la confrontación de las dos tecnologías de desarrollo de aplicaciones móviles mediante los atributos de la calidad relacionados con la eficiencia, se confirmó que el desarrollo nativo genera aplicaciones de mejor rendimiento que el desarrollo basado en tecnologías web, tal como se dijo en el marco teórico de este trabajo.

RECOMENDACIONES

La confrontación de las dos tecnologías de desarrollo de aplicaciones móviles en el presente trabajo, se llevó a cabo únicamente desde la perspectiva de los atributos de calidad relacionados con la eficiencia, sin embargo, resulta importante para trabajos futuros considerar una evaluación a las aplicaciones donde se incluya usabilidad, funcionalidad y seguridad.

Es recomendable desarrollar una aplicación caso de estudio que implemente el uso de recursos de hardware del dispositivo, como: cámara, sensores, GPS y almacenamiento interno, como también elementales tales como el calendario y la agenda de contactos.

En lo que respecta al análisis de rendimiento y monitorización sería conveniente tener acceso a otros dispositivos Android con diferentes especificaciones de hardware en cuanto a pantalla, memoria y CPU, y distintas versiones del sistema operativo. Además, llevar a cabo dicho procedimiento en un ambiente controlado.

Teniendo en cuenta que la organización ISO brinda un modelo de calidad del producto (ISO/IEC 9126) y un modelo de evaluación (ISO/IEC 14598), resultaría altamente útil contar con una herramienta de software para la toma de medidas que esté diseñada de acuerdo a dichos estándares internacionales.

Se recomienda desarrollar la aplicación basada en web usando un framework diferente a jQuery Mobile para la construcción de interfaces gráficas, pero manteniendo la utilización de Phonegap.

BIBLIOGRAFÍA

ARROYO, Natalia. Información en el móvil. Barcelona: UOC, 2011. 112 p.

CAMPDERRICH FALGUERAS, Benet. Ingeniería del Software. Barcelona: UOC, 2003. 320 p.

GARCÍA CÓRDOBA, Fernando. El cuestionario: Recomendaciones metodológicas. Córdoba, México: Limusa, 2004. 120 p.

GRANOLLERS, Toni; LORÉS VIDAL, Jesús y CAÑAS DELGADO, José Juan. Diseño de Sistemas Interactivos centrados en el usuario. Barcelona: UOC, 2005. 280 p.

GRASSO, Livio. Encuestas elementos para su diseño y análisis. Argentina: Encuentro, 2006. 184 p.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO/IEC 14598-1. Information technology - Software product evaluation. Part 1: General Overview. Primera edición. Ginebra : 1999. 18 p.

_____. ISO/IEC 14598-5. Information technology - Software product evaluation. Part 5: Process for evaluators. Primera edición. Ginebra : 1999. 35 p.

_____. ISO/IEC 9126-1: Information technology—Software product quality—Part 1: Quality model. Final draft. Ginebra: 2000. 34 p.

_____.ISO/IEC 9126-2: Software engineering –Product quality – Part 2: External metrics. Technical Report Type 2. Ginebra: 2002. 106 p.

LANDEAU, Rebeca. Elaboración de trabajos de investigación. Caracas: Alfa, 2007. 197 p.

LLORENS FÁBREGAS, Juan. Gerencia de Proyectos de Tecnología de Información. Caracas: CEC, S.A., 2005. 284 p.

NIÑO CAMAZÓN, Jesús. Sistemas Operativos monopuesto. Madrid: Editex, 2011. 312 p.

PAREDES VELASCO, Maximiliano; SANTACRUZ VALENCIA, Liliana; DOMÍNGUEZ MATEOS, Francisco. Programación Multimedia y Dispositivos Móviles. CFGS. Ra-Ma, 2012. 220 p.

PARSONS, June y OJA, Dan. Conceptos de Computación: Nuevas Perspectivas. Décima edición. México D.F.: Cengage, 2008, 848 p.

PRESSMAN, Roger. Ingeniería del Software: Un enfoque práctico. Sexta edición. McGraw-Hill. 2005. 980 p.

REVISTA GADGETS, Tecnología con estilo. En: Sistema operativo Android. 2010. México: Mina Editores. 220 p.

SOMMERVILLE, Ian. Ingeniería del Software. Séptima edición. Madrid: Addison-Wesley, 2003. 712 p.

INFOGRAFÍA

ÁLVAREZ, Antonio. Los atributos de Calidad del Software. [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: <http://www.es.sogeti.com/PageFiles/173/SEPT-ANTONIO-ALVAREZ-SOGETI.pdf>

APIGEE. Apigee Survey: Users Reveal Top Frustrations That Lead to Bad Mobile App Reviews. [En línea]. [Citado el 12 de enero de 2013]. Disponible en internet: <http://apigee.com/about/pressrelease/apigee-survey-users-reveal-top-frustrations-lead-bad-mobile-app-reviews>

Aplicaciones para dispositivos móviles. [En línea]. [Citado el 15 de Enero de 2013] Disponible en internet: http://www.scholarium.co/index.php?option=com_content&view=article&id=65&Itemid=97

Bull y Borland-Micro Focus definen el nuevo estándar de calidad de las aplicaciones móviles [En línea]. [Citado el 25 de enero de 2013]. Disponible en internet: <http://www.eleconomista.es/empresas-finanzas/noticias/4365236/10/12/Bull-y-BorlandMicro-Focus-definen-el-nuevo-estandar-de-calidad-de-las-aplicaciones-moviles.html>

BASIC4ANDROID. Características Generales. [En Línea]. [Citado el 12 de mayo de 2013]. Disponible en Internet: <http://www.basic4ppc.com/android/why.html>

BAUTISTA, José. Programación Extrema XP. [En línea]. [Citado el 10 de octubre de 2013]. Disponible en Internet en: http://ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html

BLANCO, Paco; CAMARERO, Julio; FUMERO, Antonio, WERTERSKI, Adam y RODRÍGUEZ, Pedro. Metodología de desarrollo ágil para sistemas móviles. Introducción al desarrollo con Android y el iPhone. [En línea]. [Citado el 5 de diciembre de 2013]. Disponible en internet: http://www.adamwesterski.com/wp-content/files/docsCursos/Agile_doc_TemasAnv.pdf

CAMDEN, Raymond. PhoneGap RSS Reader. [En línea]. [Citado el 10 de Octubre de 2013]. Disponible en internet en: <http://www.raymondcamden.com/index.cfm/2012/1/24/PhoneGap-RSS-Reader--Part-3>

CAPGEMINI, SOGETI y HEWLETT-PACKARD. World Quality Report 2012-2013. Cuarta Edición. [En línea]. [Citado el 1 de agosto de 2013]. Disponible en internet: http://www.capgemini.com/sites/default/files/resource/pdf/World_Quality_Report_2012_-_2013.pdf

CHAFFIN, Bryan. Google Matches Apple App Store with 700,000 Android Apps. [En línea]. [Citado el 3 de febrero de 2012] Disponible en internet: <http://www.macobserver.com/tmo/article/google-matches-apple-app-store-with-700000-android-apps>

FLURRY. China Knocks Off U.S. to Become World's Top Smart Device Market. [En línea]. [Citado el 4 de mayo de 2013]. Disponible en internet: <http://blog.flurry.com/bid/94352/China-Knocks-Off-U-S-to-Become-Top-Smartphone-Tablet-Market>

CIBERSUR.COM. 100.000 aplicaciones, el número redondo de BlackBerry para volver a la senda. [En línea]. [Citado el 3 de junio de 2013]. Disponible en internet: <http://www.cibersur.com/portada/013328/100000/aplicaciones/numero/redondo/blackberry/volver/senda>

COMPUWARE. Mobile Apps vs. Mobile Websites -- and the Winner Is? [En línea]. [Citado el 30 de agosto de 2013]. Disponible en Internet en: http://www.compuware.com/en_us/about/press-releases/2013/3/mobile-apps-vs-mobile-websites---and-the-winner-is.html

_____. Mobile Apps: What Consumers Really Need and Want. [En línea]. [Citado el 30 de agosto de 2013]. Disponible en internet: http://offers2.compuware.com/rs/compuware/images/Mobile_App_Survey_Report.pdf

COMSCORE. ComScore Reports March 2013 U.S. Smartphone Subscriber Market Share. [En línea]. Mayo de 2013. [Citado el 4 de mayo de 2013]. Disponible en internet: http://www.comscore.com/Insights/Press_Releases/2013/5/comScore_Reports_March_2013_U.S._Smartphone_Subscriber_Market_Share

COMUNICACIONES TAD_BOG. The App Date Awards. [En línea]. [Citado el 10 de Enero 2013]. Disponible en Internet: <http://bogota.theappdate.com/>

CONSTITUCIÓN POLÍTICA DE COLOMBIA. Artículo 71. [En línea]. [Citado el 13 de noviembre de 2012]. Disponible en internet: http://www.secretariassenado.gov.co/senado/basedoc/cp/constitucion_politica_1991_pr002.html

Curso Android: Construir un lector de feeds simple. [En línea]. [Citado el 10 de Octubre de 2013]. Disponible en internet en: <http://www.maestrosdelweb.com/editorial/curso-android-construir-lector-de-feeds/>

EL CUERA, Mamdouh. Pruebas de calidad en aplicaciones móviles [En línea]. [Citado el 3 de agosto de 2013]. Disponible en internet: <http://www.mtp.es/noticias/265-pruebas-calidad-aplicaciones-movilesl>

EL PAÍS. Aplicaciones para móviles, un mercado que se abre camino en Colombia. [En línea]. [Citado el 12 de Enero de 2012]. Disponible en internet: <http://www.elpais.com.co/elpais/colombia/noticias/aplicaciones-para-moviles-mercado-abre-camino-colombia>.

GOOGLE DEVELOPER ANDROID. ADT Plugin. [En Línea]. [Citado el 12 de mayo de 2013]. Disponible en Internet: <http://developer.android.com/tools/sdk/eclipse-adt.html#notes>

_____. Android, the world's most popular mobile platform. [En línea]. [Citado el 6 de mayo de 2013]. Disponible en internet: <http://developer.android.com/about/index.html>

_____. Platform versions. [En línea]. [Citado el 8 de mayo de 2013]. Disponible en internet: <http://developer.android.com/about/dashboards/index.html>

_____. Training. [En línea]. [Citado el 5 de agosto de 2013]. Disponible en internet: <http://developer.android.com/training/best-performance.html>

GRANOLLERS, Toni; LORÉS, Jesús. Esfuerzo de Usabilidad: un nuevo concepto para medir la usabilidad de un sistema interactivo basada en el Diseño Centrado en el Usuario

[En línea]. [Citado el 17 de marzo de 2013]. Disponible en internet: <http://griho2.udl.es/publicacions/2004/EU%20i2004.pdf>

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. Acerca de ISO, Nuestra historia. [En línea]. [Citado el 15 de julio de 2013] Disponible en internet: <http://www.iso.org/iso/home/about.htm>

J.D. Power and Associates Reports: Smartphone Battery Life has Become a Significant Drain on Customer Satisfaction and Loyalty. [En línea]. [Citado el 2 de diciembre de 2013]. Disponible en internet en: <http://www.prnewswire.com/news-releases/jd-power-and-associates-reports-smartphone-battery-life-has-become-a-significant-drain-on-customer-satisfaction-and-loyalty-142765065.html>

La evolución de Snake. [En línea]. [Citado el 8 de Diciembre de 2012]. Disponible en internet: <http://conversaciones.nokia.com/2011/12/16/la-evolucion-de-snake/>

LITTLE EYE. How Little Eye Works Under The Hood. [En línea]. [Citado el 2 de junio de 2103]. Disponible en internet: <http://www.littleeye.co/blog/2013/07/24/how-little-eye-works-under-the-hood-part-1/>

_____. Little Eye wins 2013 TiE “Top Startup” Award. [En línea]. [Citado el 2 de junio de 2103]. Disponible en internet: <http://www.littleeye.co/blog/2013/06/02/littleeye-wins-tie-top-startup-award/>

MENA TOBAR, Luis; LATORRE, Pedro y LAFUENTE LAPENA, Elena. WebA Mobile (Web Analysis Mobile): Herramienta de ayuda para el diseño y evaluación de websites para dispositivos móviles [En línea]. [Citado el 10 de diciembre de 2012]. Disponible en internet: http://www.aipo.es/info_art.php?id=33

Ministerio TIC presenta Apps.co, su aceleradora para emprendimientos. [En línea]. [Citado el 20 de Noviembre de 2012]. Disponible en internet: <http://www.mintic.gov.co/index.php/mn-news/71-ministerio-en-medios/1324-ministerio-tic-presenta-appsco-su-aceleradora-para-emprendimientos>

Mobile Web Apps vs. Mobile Native Apps: How to Make the Right Choice. [En línea]. [Citado el 15 de marzo de 2013]. Disponible en internet: http://www.lionbridge.com/files/2012/11/Lionbridge-WP_MobileApps2.pdf

Native or Web Application? How Best to Deliver Content and Services to Your Audiences over the Mobile Phone. [En línea]. [Citado el 12 de marzo de 2013]. Disponible en internet: http://www.globalintelligence.com/insights-analysis/white-papers/native-or-web-application-how-best-to-deliver-cont/GIA%20Industry%20White%20Paper%202010_Native%20or%20Web%20App_How%20Best%20to%20Deliver%20Content%20and%20Services%20to%20Your%20Audiences%20over%20the%20Mobile%20Phone_Apr-10.pdf

Norte de Santander Vive Digital y aplicaciones para dispositivos móviles. [Vídeo en línea]. [Citado el 6 de mayo de 2013]. Disponible en internet: [http://www.ufpso.edu.co/medios/reproductor_video_conexiones.php?rutavideo=../videos/conexiones/2012_08\(30\)aplicativo.f4v](http://www.ufpso.edu.co/medios/reproductor_video_conexiones.php?rutavideo=../videos/conexiones/2012_08(30)aplicativo.f4v)

PRADA CÉSPEDES, Edilma. Cauca pionero en el desarrollo de aplicaciones móviles en Latinoamérica. [En línea]. [Citado el 25 de Enero de 2012]. Disponible en internet: <http://www.agendapropia.com/index.php/dialogos/1676-cauca-pionero-en-el-desarrollo-de-aplicaciones-moviles-en-latinoamerica>

PRESIDENCIA DE LA REPÚBLICA DE COLOMBIA. Decreto 393 de 1991. [En línea]. [Citado el 15 de Noviembre de 2012]. Disponible en internet: http://www.colciencias.gov.co/sites/default/files/upload/reglamentacion/decreto_393_de_1991.pdf

Pruebas de Aplicaciones Móviles. [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: <http://www.accensit.com/index.php/desarrollo-de-aplicaciones-para-moviles/pruebas-de-aplicaciones-moviles.html>

¿Qué es Google Play? [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: https://motorola-global-es-latam.custhelp.com/app/answers/prod_answer_detail/a_id/82111/p/7901,8196/action/auth

¿Qué es un smartphone? [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: <http://www.lgmobile.cl/smartphones>

SENADO DE LA REPÚBLICA DE COLOMBIA. Ley 1341 de 2009. [En línea]. [Citado el 16 de noviembre de 2012]. Disponible en internet: http://www.secretariasenado.gov.co/senado/basedoc/ley/2009/ley_1341_2009.html

SlashSession: Testing en aplicaciones móviles. [En línea]. [Citado el 21 de enero de 2013]. Disponible en internet: <http://slashmobility.com/blog/2011/09/410/>

SOASTA. 88% of Americans Have Negative Feelings about Brands with Poorly Performing Websites and Mobile Apps. [En línea]. [Citado el 31 de agosto de 2013]. Disponible en internet: <http://www.soasta.com/press-releases/467413/>
Steve Jobs proclaims the post-PC era has arrived. [En línea]. [Citado el 18 de Noviembre de 2012]. Disponible en internet: <http://www.techrepublic.com/blog/hiner/steve-jobs-proclaims-the-post-pc-era-has-arrived/4701>

SVETLIK, Joe. Windows Phone Store hits 150,000 apps, doubled in last year. [En línea]. [Citado el 3 de febrero de 2012] Disponible en internet: <http://crave.cnet.co.uk/software/windows-phone-store-hits-150000-apps-doubled-in-last-year-50010072/>

UNIVERSIDAD POLITECNICA DE VALENCIA. Arquitectura de Android. [En línea]. [Citado el 16 de enero de 2013]. Disponible en internet: <http://www.androidcurso.com/index.php/tutoriales-android/31-unidad-1-vision-general-y-entorno-de-desarrollo/99-arquitectura-de-android>

Update from the CEO. [En línea]. Marzo de 2013. [Citado el 6 de mayo de 2013]. Disponible en internet: <http://googleblog.blogspot.com.es/2013/03/update-from-ceo.html>

VALERO, Claudio. Android alcanza en número de aplicaciones a iOS. [En línea]. [Citado 12 de diciembre de 2012]. Disponible en internet: <http://www.adslzone.net/article9843-android-alcanza-en-numero-de-aplicaciones-a-ios.html>.

Validación de aplicaciones móviles. [En línea]. [Citado el 20 de enero de 2012]. Disponible en Internet: <http://www.albirasolutions.com/es/servicios/tecnologias-de-la-informacion/control-de-calidad/validacion-de-aplicaciones-moviles>

VIVANCO, Andrés. Evaluación de la calidad del sistema integrado para casas de valores SICAV de la Bolsa de Valores de Quito utilizando la norma ISO/IEC 14598. [En línea]. [Citado el 8 de marzo de 2013] Disponible en internet: <http://repositorio.espe.edu.ec/handle/21000/5575>

WOMACK, Brian. Google Says 700,000 Applications Available for Android. [En línea]. [Citado 12 de Diciembre de 2012]. Disponible en internet: <http://www.businessweek.com/news/2012-10-29/google-says-700-000-applications-available-for-android-devices>

XAMARIN. ¿Por qué te va a encantar Xamarin.Android? [En Línea]. [Citado el 12 de mayo de 2013]. Disponible en Internet: <http://xamarin.com/monoforandroid>

Ya se venden más smartphones que móviles en el mundo. [En línea]. Marzo de 2013. [Citado el 4 de mayo de 2013]. Disponible en internet: http://www.pcactual.com/articulo/actualidad/noticias/12921/venden_mas_smartphones_que_moviles_mundo.html

YORIO, Darío. Identificación y Clasificación de Patrones en el Diseño de Aplicaciones Móviles. [En línea]. [Citado el 9 de agosto de 2013]. Disponible en internet: http://postgrado.info.unlp.edu.ar/Carreras/Magisters/Ingenieria_de_Software/Tesis/Yorio_Ruben.pdf

ANEXOS

Anexo A. Instrumento de Recolección de información

CUESTIONARIO

Datos del Encuestado	
Edad	Sexo
<input type="checkbox"/> 15-19	<input type="checkbox"/> Mujer
<input type="checkbox"/> 20-24	<input type="checkbox"/> Hombre
<input type="checkbox"/> 25-29	
<input type="checkbox"/> 30-34	

La presente encuesta va dirigida a los estudiantes del programa de Ingeniería de Sistemas. Le agradecemos que dedique unos minutos a responder esta pequeña encuesta, será muy importante para el desarrollo de un proyecto de grado de la misma carrera.

1. Indique las características de su celular:

Marca	
Modelo	
Sistema Operativo	

2. ¿Cuenta con otro dispositivo móvil además de su celular? Sí ___ No ___ ¿Cuál?

Dispositivo Móvil	Marca	Modelo
Smartphone		
PDA (Asistente Personal Digital)		
Tableta electrónica		
Otro:		

3. ¿Le interesa desarrollar aplicaciones para dispositivos móviles?

Sí ___ No ___

4. ¿En alguna de las asignaturas ha tenido la oportunidad de tener por lo menos contacto inicial con el desarrollo de aplicaciones móviles?

Sí ___ No ___

¿En cuál asignatura?

_____.

5. ¿Ha realizado algún curso para comenzar a desarrollar aplicaciones móviles?
 Sí ____ No ____
 ¿Por cuál medio o institución? _____.

6. ¿Ha desarrollado una aplicación para dispositivos móviles?
 Sí ____ No ____
 Para cuál plataforma (o sistema operativo). Seleccione una:

<input type="checkbox"/>	Android
<input type="checkbox"/>	BlackBerry OS
<input type="checkbox"/>	Symbian
<input type="checkbox"/>	iOS (iPhone, iPad)
<input type="checkbox"/>	Windows Phone
<input type="checkbox"/>	Otro:

Solamente si su respuesta es afirmativa, continúe con las siguientes preguntas.
 Explique brevemente en qué consiste dicha aplicación

¿Qué framework o herramienta software utilizó?
 _____.

¿De qué tipo es la aplicación?
 Nativa ____ Web ____

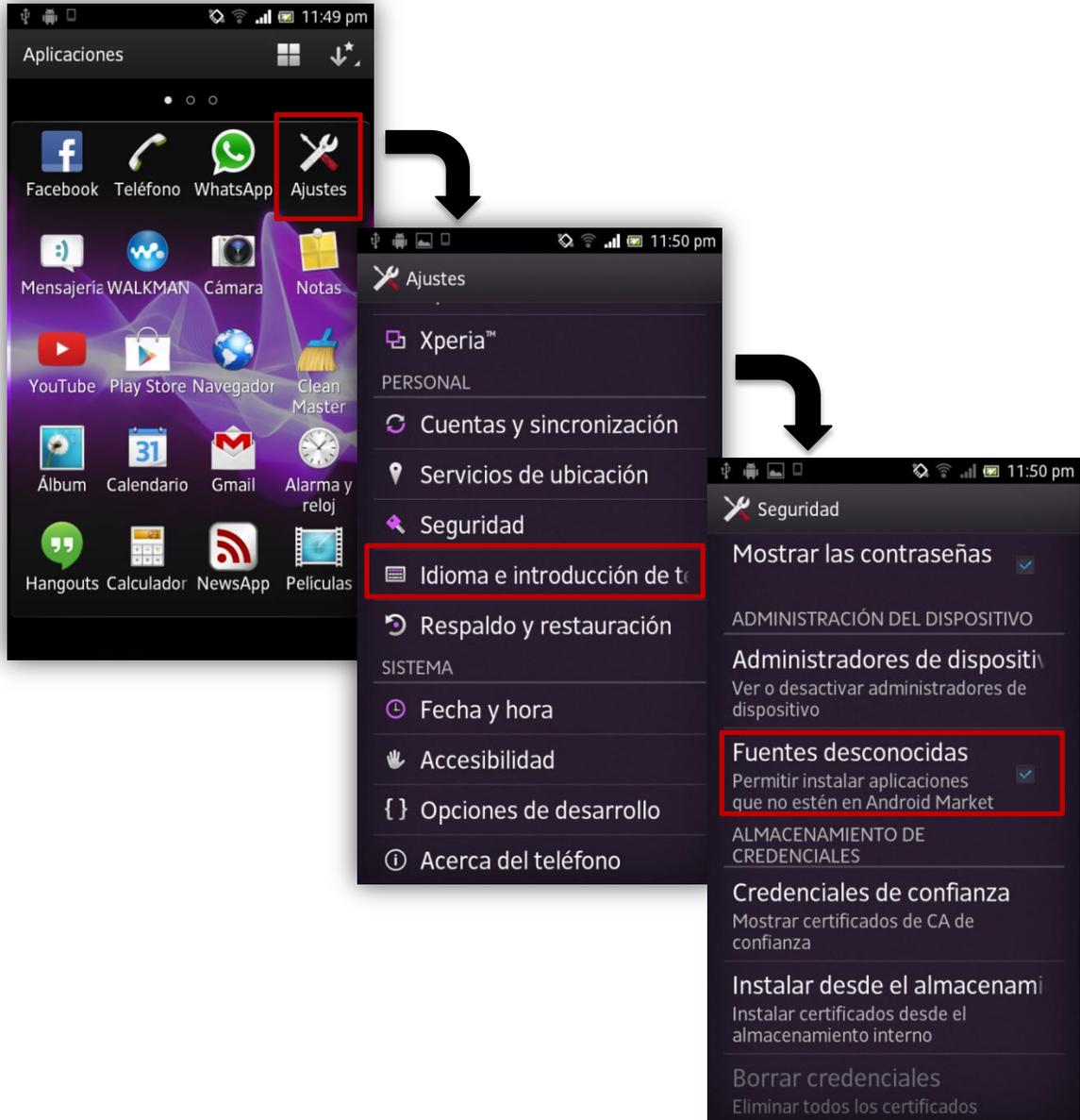
7. Para cuál de las siguientes tecnologías le gustaría desarrollar en una próxima oportunidad:

<input type="checkbox"/>	Windows Phone (Ejemplo: Nokia Lumia 900, Samsung Focus S)
<input type="checkbox"/>	BlackBerry OS (Ejemplo: BlackBerry Curve 9320, BlackBerry Bold 9790)
<input type="checkbox"/>	Symbian (Ejemplo: Nokia C3, Nokia N97)
<input type="checkbox"/>	iOS (Ejemplo: iPhone 5, iPad 4)
<input type="checkbox"/>	Android (Ejemplo: Samsung Galaxy S3, Sony Xperia S)
<input type="checkbox"/>	Otro:

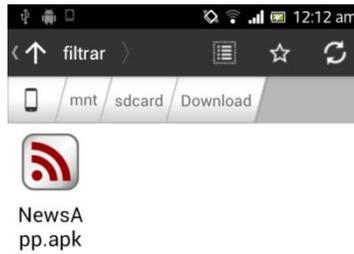
Anexo B. Manual de instalación de NewsApp

Para la instalación de NewsApp (NewsApp.apk) en el terminal Android, se emplea el siguiente procedimiento (probado en Android 4.0 Ice Cream Sandwich):

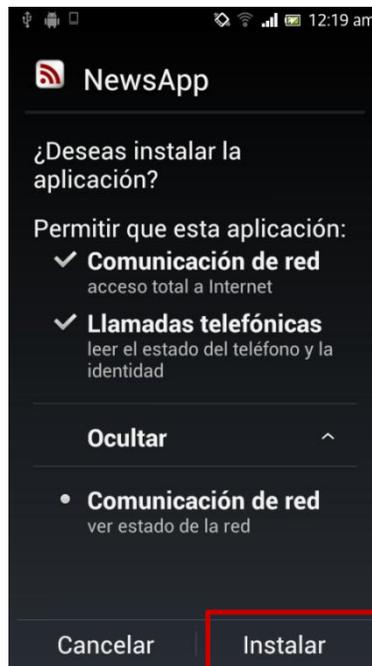
1. Antes de cualquier otro paso es indispensable hacer un cambio en los ajustes del teléfono, para esto, vaya a al menú de *Ajustes* > *Seguridad* y habilitar la opción “Fuentes desconocidas (Permitir instalar aplicaciones que no estén en Android Market)”



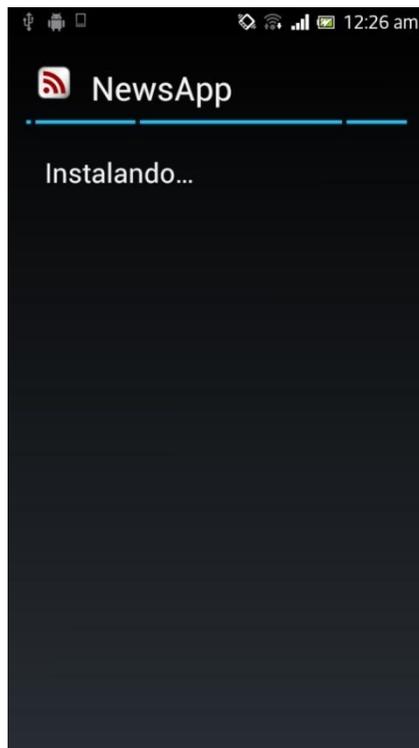
2. Guardar el fichero **NewsApp.apk** en el directorio que el usuario desee.



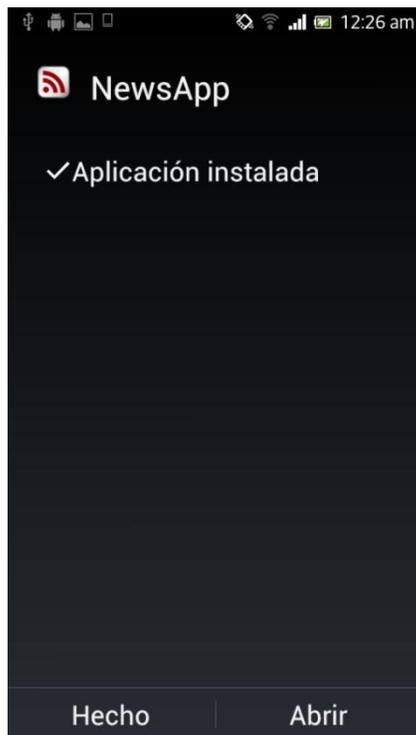
3. Una vez realizado el paso anterior, acceder a la carpeta donde se encuentra **NewsApp.apk** y ejecutarlo. Se debe aceptar los permisos requeridos.



Entonces aparece lo siguiente:



Por último, pulsar en “Abrir” o “Hecho”; la aplicación se instaló exitosamente.



Anexo C. Archivos .apk de NewsApp

Los archivos .apk de la aplicación NewsApp en sus dos versiones: nativa y basada en web, se encuentran en el CD incluido en este trabajo.