

Enfoques y Tendencias en el Desarrollo de Aplicaciones para Dispositivos Móviles.

Pablo Thomas , Lisandro Delia , Leonardo Corbalán , Germán Cáseres ,
Juan Fernández Sosa , Fernando Tesone , Verónica Aguirre , Verena Olsowy ,
Patricia Pesado 

Instituto de Investigación en Informática LIDI (III-LIDI)
Facultad de Informática – Universidad Nacional de La Plata
50 y 120 La Plata Buenos Aires

Centro Asociado a la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CIC)
526 e/ 10 y 11 La Plata Buenos Aires

{pthomas, ldelia, corbalan, gcaseres, jfernandez, ftesone, vaguirre, volsowy, ppesado}@lidi.info.unlp.edu.ar

RESUMEN

Se presenta una línea de investigación y desarrollo centrada en el estudio de aspectos de Ingeniería de Software aplicados a las diferentes tendencias en el desarrollo de Aplicaciones para Dispositivos Móviles.

Palabras claves: Plataformas para Dispositivos Móviles - Aplicaciones Nativas- Aplicaciones Híbridas – Aplicaciones Interpretadas – Aplicaciones por Compilación Cruzada – Aplicaciones Web Progresivas – Offline First. – Instant App

CONTEXTO

Esta línea de Investigación forma parte del Proyecto (2018-2021) “*Metodologías, técnicas y herramientas de Ingeniería de Software en escenarios híbridos. Mejora de proceso*”, en particular del subproyecto *Ingeniería de Software para escenarios híbridos* del Instituto de Investigación en Informática LIDI de la Facultad de Informática, acreditado por el Ministerio de Educación de la Nación.

Hay cooperación con Universidades de Argentina y se está trabajando con Universidades de Europa y Latinoamérica.

1. INTRODUCCIÓN

Los avances tecnológicos en la comunicación móvil, junto con los avances en el campo de la computación han dado lugar a la computación móvil, que puede definirse como un entorno de cómputo con movilidad física que brinda al usuario la capacidad de acceder a datos, información u otros objetos lógicos desde cualquier dispositivo en cualquier red, al mismo tiempo que va moviéndose, cambiando su ubicación geográfica [1].

Las características específicas de la computación móvil plantean nuevos desafíos en el desarrollo de software para este tipo de dispositivos. La necesidad de tratar con diversos estándares, protocolos y tecnologías de red; las capacidades limitadas, no sólo en cuanto a procesamiento, sino también, en cuanto a consumo, tamaño físico y capacidad de almacenamiento de los dispositivos; las restricciones de tiempo impuestas por un mercado altamente dinámico y la existencia de distintas plataformas de hardware y software son sólo algunas de las dificultades a las que se enfrentan los desarrolladores en esta área.

Las aplicaciones se generan en un entorno dinámico e incierto. En su mayoría se trata de aplicaciones pequeñas, no

críticas, destinadas a un gran número de usuarios finales que son liberadas en versiones rápidas para poder satisfacer las demandas del mercado. En otros casos las aplicaciones son de mayor tamaño, a veces para brindar movilidad a una parte de la funcionalidad de un sistema más grande, mientras que otras veces son el único punto de interacción del sistema.

Algunas iniciativas como *offline first* proponen nuevas maneras de abordar el desarrollo de aplicaciones móviles tomando como requerimiento no funcional la posibilidad de seguir brindando un servicio incluso cuando el dispositivo pierde conectividad con la red móvil.

Todas las particularidades previamente mencionadas hacen que el desarrollo de software para dispositivos móviles difiera considerablemente del tradicional. Ello conduce a nuevas prácticas y metodologías que promueven el crecimiento de la Ingeniería de Software como disciplina, acompañando este proceso de desarrollo tecnológico.

La existencia de una enorme competencia en el mercado de las aplicaciones móviles, promovida en parte por la proliferación de dispositivos móviles y el número de negocios que están migrando sus servicios a Internet, revela la importancia que implica para las empresas el posicionamiento en este mercado. Para conseguirlo, es necesario reducir al máximo el tiempo de desarrollo de las aplicaciones y al mismo tiempo hacer que éstas se ejecuten en el mayor número de dispositivos posible.

Este propósito se ve obstaculizado por la excesiva fragmentación de hardware y software existente, originada por el alto número de dispositivos distintos, con sus propios sistemas operativos y plataformas de desarrollo.

La alternativa más costosa para hacer frente a esta problemática consiste en el desarrollo específico de aplicaciones de manera nativa en cada una de las

plataformas existentes, utilizando entornos de desarrollo, lenguajes de programación y tecnologías propias de cada plataforma.

El desarrollo nativo de aplicaciones móviles posee ciertas ventajas como la posibilidad de acceder sin limitaciones a todas las características del dispositivo (cámara, GPS, acelerómetro y agenda, entre otras), el alto rendimiento, interfaces gráficas consistentes con el resto de la plataforma, la posibilidad de trabajar sin acceso a Internet y de correr procesos en segundo plano. Estas aplicaciones pueden distribuirse a través de las tiendas en línea correspondientes. Sin embargo, el precio de todas estas ventajas es alto: no es posible reutilizar el código fuente entre plataformas diferentes, el esfuerzo se multiplica y se elevan los costos de desarrollo, actualización y distribución de nuevas versiones.

Como alternativa al desarrollo nativo en múltiples plataformas, se presentan diversos enfoques de desarrollo que procuran optimizar la relación costo/beneficio compartiendo la misma base de código fuente entre las versiones para las distintas plataformas [2]. La construcción de aplicaciones Web Móviles constituye un ejemplo representativo de este enfoque. Estas aplicaciones se diseñan para correr dentro de un navegador, se desarrollan con tecnologías web bien conocidas (HTML, CSS y JavaScript), no necesitan adecuarse a ningún entorno operativo móvil específico; su puesta en marcha es rápida y sencilla.

Las desventajas de las aplicaciones Web Móviles recaen sobre su rendimiento. Los tiempos de respuesta se dilatan afectados por la interacción cliente-servidor y las restricciones de seguridad impuestas a la ejecución de código por medio del navegador limitan el acceso a todas las capacidades del dispositivo. Además, la experiencia de usuario dista de ser similar a la de las aplicaciones nativas, resultando menos atractiva para el usuario final.

El enfoque híbrido es una alternativa que posee también la ventaja de estar basado en tecnologías web estándar (HTML, Javascript y CSS) pero, a diferencia del anterior no funciona dentro de un navegador, sino en un contenedor web especial con mayor acceso a las características del dispositivo a través de una API específica.

Las aplicaciones híbridas permiten la reutilización de código en las distintas plataformas, el acceso al hardware del dispositivo, y la distribución a través de las tiendas de aplicaciones. Sin embargo, conservan algunas de las desventajas de las aplicaciones Web Móviles: la utilización de componentes no nativos en la interfaz perjudica la experiencia de usuario, y la ejecución se ve ralentizada por la carga asociada al contenedor web.

El enfoque interpretado se presenta como una alternativa en donde las aplicaciones son traducidas en su mayor parte a código nativo, mientras que un resto se interpreta en ejecución. Se implementan de forma independiente de las plataformas utilizando diversas tecnologías y lenguajes, tales como Javascript, Typescript y XML, entre otros.

La obtención de interfaces nativas constituye una de las principales ventajas de este tipo de aplicaciones, mientras que la definición de nuevas componentes suele tener un alto grado de complejidad debido a la necesidad de definir abstracciones compatibles con diferentes plataformas.

Finalmente, las aplicaciones generadas por compilación cruzada también constituyen un tipo de desarrollo multiplataforma. Estas aplicaciones se compilan de manera nativa creando una versión específica de alto rendimiento para cada plataforma destino.

Un nuevo concepto ha surgido en los últimos años denominado Aplicaciones Web Progresivas (PWA por sus siglas en inglés). Una PWA es una aplicación web que utiliza las últimas tecnologías

disponibles en los navegadores para ofrecer en dispositivos móviles una experiencia lo más parecida posible a la de una aplicación nativa.

Los objetivos que persiguen las PWA son: lograr el mayor rendimiento posible en dispositivos móviles, que la aplicación cargue de manera casi instantánea, que la interfaz de usuario se parezca lo máximo posible a una nativa, que se pueda trabajar sin conexión (*offline first*) y que se puedan enviar notificaciones a los usuarios, como en una aplicación nativa.

Una nueva tecnología, desarrollada recientemente por Google, denominada TWA (Trusted Web Activities) o Actividades Web de Confianza, permite integrar una PWA con una aplicación Android. Las TWA son ejecutadas desde un APK y distribuidas desde Google Play Store. Muestran en pantalla completa un navegador web dentro de una aplicación Android sin mostrar la interfaz del navegador.

Las TWA representan un punto de inflexión para los desarrolladores que ahora pueden distribuir sus PWA en Google Play Store que ha dejado de ser una tienda exclusiva de aplicaciones nativas.

Desde 2017 los desarrolladores de Android tienen una nueva opción para hacer llegar sus apps a los usuarios finales. El concepto de *Instant App* permite ejecutar una funcionalidad específica de una aplicación sin necesidad de instalar la App completa.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

- Enfoques de desarrollo de Aplicaciones para Dispositivos Móviles.
- Metodologías y Técnicas de la Ingeniería de Software y su aplicación en el desarrollo de software para dispositivos móviles.
- Aplicaciones Nativas en Android [3].

- Aplicaciones Nativas en iOS [3].
- Aplicaciones Web Móviles.
- Aplicaciones Móviles Híbridas (PhoneGap [4], Ionic [5]).
- Aplicaciones Móviles Interpretadas (Appcelerator Titanium [6], NativeScript [7]).
- Aplicaciones Móviles generadas por compilación cruzada (Xamarin [8], Corona [9]).
- PWA, Instant App y Offline First.
- TWA (Trusted Web Activities),
- Análisis y estudio comparativo de requerimientos no funcionales, tales como rendimiento, consumo de energía, tamaño de software, entre otros, en los distintos enfoques de Aplicaciones Móviles.
- Experiencia de usuario en Aplicaciones Móviles generadas con distintos enfoques de desarrollo.
- Internet de las Cosas (IoT), Internet de Todo (IoE) y aplicaciones de sensado móvil o sensado urbano.
- Aplicaciones Móviles y su utilización en el proyecto de *Smart City*.
- El rol de los dispositivos móviles en aplicaciones de *crowdsourcing* o colaboración abierta distribuida.

3. RESULTADOS OBTENIDOS/ESPERADOS

Los resultados obtenidos/esperados se pueden resumir en:

- Se ha realizado un estudio [10] sobre el impacto que distintos frameworks de desarrollo multiplataforma tienen sobre el tamaño de la aplicación construida. Los resultados obtenidos en este estudio son relevantes dado que el espacio disponible se ha convertido en un recurso crítico para muchos usuarios.
- Se ha elaborado un detallado estudio comparativo [11] que contempla una

lista de 23 características de interés para la Ingeniería de Software y su impacto en 9 tecnologías para el desarrollo de aplicaciones móviles.

- Se ha presentado un minucioso estudio sobre la eficiencia energética y su relación con los enfoques de desarrollo [12]. Se estudió la autonomía de las baterías de los dispositivos móviles en aplicaciones con funciones multimedia (acceso a imágenes y reproducción de video) y con alta carga de procesamiento, generadas con diversos enfoques de desarrollo.
- Se ha estudiado [14] [15] la manera en que el enfoque de desarrollo utilizado afecta el rendimiento de aplicaciones que realizan procesamiento intensivo. Se extrajeron conclusiones considerando las plataformas iOS y Android utilizando varios frameworks de desarrollo multiplataforma.
- Se estudiaron de manera integral y conjunta dos tópicos importantes en el área de las aplicaciones móviles: los enfoques de desarrollo y los requerimientos no funcionales.[26] El objetivo planteado fue cuantificar el impacto que tiene la elección de un enfoque de desarrollo, y en particular un determinado *framework*, sobre tres de los requerimientos no funcionales más demandados en el área de las aplicaciones para dispositivos móviles: rendimiento, consumo de energía y uso de espacio de almacenamiento. Se analizaron enfoques nativos y multiplataformas. El estudio incluyó a 6 de los *frameworks* de desarrollo más conocidos y utilizados en el mercado: 1) Android SDK (enfoque nativo), 2) Cordova (enfoque híbrido), 3) Titanium (enfoque interpretado), 4) NativeScript (enfoque interpretado), 5) Xamarin (enfoque compilación cruzada) y 6) Corona (enfoque compilación cruzada). Se obtuvieron conclusiones que sirven de apoyo a las decisiones que los

desarrolladores de software deben realizar al momento de elegir el enfoque/*framework* más adecuado de acuerdo con el nivel de expectativas creadas sobre el rendimiento, consumo de energía y uso de espacio de almacenamiento.

- En [27] se analizaron las principales características de otro enfoque de desarrollo para aplicaciones móviles denominado Aplicaciones Web Progresivas o PWA, por sus siglas en inglés. Dado su estrecha relación con las aplicaciones web móviles resultó de interés realizar un detallado estudio comparativo entre ambos enfoques explicitando diferencias y similitudes. Se concluyeron una serie de ventajas y desventajas de un enfoque respecto del otro.
- Se realizó el lanzamiento de la versión para iOS de "*Informática UNLP*", una aplicación móvil presentada en [13] desarrollada para la comunidad de la Facultad de Informática de la Universidad Nacional de La Plata. Las primeras versiones de esta aplicación fueron liberadas para la plataforma Android. En la actualidad también está disponible en la tienda App Store para iPhone y iPad. "*Informática UNLP*" mejora la comunicación entre alumnos y docentes. Incluye una cartelera virtual donde los alumnos reciben información de interés sobre sus cursos. También permite consultar información de la ocupación de las aulas en tiempo real utilizando tecnología de realidad aumentada
- Se continúa con el desarrollo de nuevas características de "*Informática UNLP*", una aplicación multiplataforma en continua evolución y crecimiento.
- Se continúa con el estudio de PWA al que se agrega el análisis de los alcances de la tecnología TWA (Trusted Web Activities), recientemente aparecida. TWA es la estrategia implementada por Google para integrar una PWA con una

aplicación Android que puede ser entonces distribuida desde Google Play Store. Actualmente se está realizando un análisis detallado de la combinación PWA+TWA, teniendo en cuenta características no funcionales, de desarrollo, y de la gestión de proyectos de software.

- El Centro de Innovación y Transferencia Tecnológica (CiyTT) de la Facultad de Informática de la UNLP fue sede de una muestra de Ciencia y Tecnología 2019 en la que se exhibieron diferentes desarrollos tecnológicos innovadores. La muestra consistió de 10 estaciones, distribuidas a lo largo del centro, las cuales contenían distintos proyectos que podían ser visitados por el público en general. Para realizar una visita guiada, se desarrolló una PWA denominada "InnovApp". Su funcionalidad principal consistió en describir, por medio de texto y contenido multimedia (imágenes, audio y video), cada uno de los proyectos dentro de las diferentes estaciones.
- Se esperan resultados concretos al estudiar las posibilidades que surgen de la utilización de los sensores de los dispositivos móviles en aplicaciones de Internet de las Cosas (IoT), Internet de Todo (IoE) y particularmente aplicaciones de sensado móvil, también conocido como sensado urbano.
- Se plantea analizar la utilidad de las Aplicaciones Móviles en el contexto de las ciudades inteligentes, haciendo foco especialmente en aquellas que implementan la colaboración abierta distribuida (*crowdsourcing*).
- Analizar las ventajas y desventajas de las "*Instant App*" de Android. Resulta de interés su estudio y posterior comparación con las aplicaciones nativas tradicionales.
- Promover el avance sostenido y continuo de la formación de los miembros involucrados en ella.

4. FORMACIÓN DE RECURSOS HUMANOS

Los integrantes de esta línea de investigación dirigen Tesinas de Grado y Tesis de Postgrado en la Facultad de Informática, y Becarios III-LIDI en temas relacionados con este proyecto. Además, participan en el dictado de asignaturas/cursos de grado y postgrado de la Facultad de Informática de la UNLP.

5. BIBLIOGRAFÍA

1. Hongxing Li, Guochu Shou, Yihong Hu, Zhigang Guo. Mobile Edge Computing: Progress and Challenges. 2016 4th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud). Oxford UK.
2. Spyros Xanthopoulos, Stelios Xinogalos, *A Comparative Analysis of Cross-platform Development Approaches for Mobile Applications*, BCI' 2013, Greece
3. Tracy, K.W., *Mobile Application Development Experiences on Apple's iOS and Android OS*, Potentials, IEEE, 2012.
4. <http://phonegap.com/>
5. <https://ionicframework.com/>
6. <http://www.appcelerator.com/>
7. <https://www.nativescript.org/>
8. <http://xamarin.com/>
9. <https://coronalabs.com/>
10. J. Fernández Sosa, P. Thomas, L. Delía, G. Cáseres, L. Corbalán, F. Tesone, A. Cuitiño, P. Pesado. *“Mobile Application Development Approaches: A Comparative Analysis on the Use of Storage Space”*. XV Workshop de Ingeniería de Software - XXIV Congreso Argentino de Ciencias de la Computación CACIC 2018. Tandil, Argentina. 8 al 12 de octubre de 2018.
11. L. Delia, P. Thomas, L. Corbalan, J. Fernandez Sosa, A. Cuitiño, G. Cáseres, P. Pesado. *“Development Approaches for Mobile Applications: Comparative Analysis of Features”* SAI - Computing Conference 2018. London, United Kingdom. 10 al 12 de Julio de 2018.
12. L. Corbalan, J. Fernandez Sosa, A. Cuitiño, L. Delía, G. Cáseres, P. Thomas, P. Pesado. *“Development Frameworks for Mobile Devices: A Comparative Study about Energy Consumption”*. 5th IEEE/ACM International Conference on Mobile Software Engineering and Systems. Gothenburg, Sweden. 27 y 28 de Mayo del 2018.
13. Fernández Sosa J., Cuitiño A., Thomas P., Delia L., Caseres G., Corbalán L., Pesado P. *“Informática UNLP” la App de la Facultad de Informática*. CACIC 2017. La Plata, del 9 al 13 de Octubre de 2017.
14. Delia L., Galdamez N., Corbalan L., Thomas P., Pesado P. *Approaches to Mobile Application Development: Comparative Performance Analysis* SAI Computing Conference (SAI), 2017. Londres, del 18 al 20 de Julio de 2017.
15. Delia L., Galdamez N., Corbalan L., Thomas P., Pesado P., *Un Análisis comparativo de rendimiento en Aplicaciones Móviles Multiplataforma*, CACIC 2015, UNNOBA Junín, Octubre 2015.
16. Delia L., Galdamez N., Corbalan L., Thomas P., Pesado P., *Multi-Platform Mobile Application Development Analysis*, IEEE Ninth International Conference on Research Challenges in Information Science, May 2015, Athens, Greece, ISBN 978-1-4673-6630-4
17. Delia L., Galdamez N., Thomas P., Pesado P., *Un Análisis Experimental de Tipo de Aplicaciones para Dispositivos Móviles*, CACIC 2013, CAECE Mar del Plata, Octubre 2013
18. Digital Possibilities. Mobile Development Frameworks Overview <http://digital-possibilities.com/mobile-development-frameworks-overview/>

19. Markus Falk. Mobile Frameworks Comparison Chart, <http://www.markus-falk.com/mobile-frameworks-comparison-chart/>
20. Anup Kumar y Bin Xie, Handbook of Mobile Systems Applications and Services. Editorial CRS Press, ISBN 978-1-4398-0152-9, Año 2012.
21. Sambasivan, D.; John, N.; Udayakumar, S.; Gupta, R., Generic framework for mobile application development, Internet (AH-ICI), 2011 Second Asian Himalayas International Conference on Computing & Processing (Hardware/Software).
22. Choi, Y.; Yang, J.-S.; Jeong, J., Application framework for multi platform mobile application software development, Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference on Computing & Processing (Hardware/Software)
23. Anthony Wasserman, Carnegie Mellon Silicon Valley, Software Engineering Issues for Mobile Application Development, 2° Workshop on Software Engineering for Mobile Application Development, MobiCASE '11, Santa Monica, California, USA, October 2011.
24. Jemel, M., Serhrouchni, A. Content protection and secure synchronization of HTML5 local storage data. Consumer Communications and Networking Conference (CCNC), 2014 IEEE 11th, Las Vegas, NV, USA.
25. Tim A. Majchrzak, Andreas Biørn-Hansen, Tor-Morten Grønli, ***Progressive Web Apps: the Definite Approach to Cross-Platform Development*** ?, Proceedings of the 51st Hawaii International Conference on System Sciences j, 2018, ISBN: 978-0-9981331-1-9
26. Corbalán, L., Thomas, P., Delía, L., Cáseres, G., Sosa, J. F., Tesone, F., & Pesado, P. (2019, June). A Study of Non-Functional Requirements in Apps for Mobile Devices. In *Conference on Cloud Computing and Big Data* (pp. 125-136). Springer, Cham
27. Aguirre, V., Ortu, A., Delía, L. N., Thomas, P. J., Corbalán, L. C., Cáseres, G., & Pesado, P. M. (2019). PWA para unificar el desarrollo Desktop, Web y Mobile. In *XXV Congreso Argentino de Ciencias de la Computación (CACIC)*(Universidad Nacional de Río Cuarto, Córdoba, 14 al 18 de octubre de 2019).