

IULE APPLICATION FOR SMART-PHONES FOR ACADEMIC ADMINISTRATION AT THE UNIVERSITY OF LEÓN

Alfonso Cendón, J.¹; Castejón Limas, M.¹; Rodríguez Delgado, G.¹; Alonso Juan, I.¹;
Fernández, L.²

¹ Universidad de León, ² ROBLES

The smartphones are increasingly presents in our daily lives, opening an important window of opportunity with its use. From the area of Project Engineering at the University of León is considered fundamental to provide a comprehensive management application accessible from mobile devices to the university community, for this reason has developed iule. This application has been implemented for IOS and Android operating systems using the most advanced programming technologies. The main functions of the application include full access to the academic and research information, academic staff information and consultation of the status of the various services offered on campus.

Keywords: *iULE; Software engineering; Projects, Computer science*

IULE APLICACIÓN PARA SMART-PHONES PARA LA GESTIÓN ACADÉMICA EN LA UNIVERSIDAD DE LEÓN

Los denominados Smartphone o teléfonos inteligentes se encuentran cada vez más presentes en nuestra vida diaria, abriendo un importante abanico de oportunidades con su utilización. Desde el área de Proyectos de Ingeniería de la Universidad de León se ha considerado fundamental el dotar a la comunidad universitaria de una aplicación para la gestión integral académica, accesible desde dispositivos móviles, propósito con el que se ha desarrollado iULE. Esta aplicación ha sido implementada para los sistemas operativos iOS y Android utilizando las más avanzadas tecnologías de programación. Entre las funciones principales de la aplicación destacan el acceso completo a toda la oferta formativa e investigadora de la universidad, a la información académica personal, así como la consulta del estado de los distintos servicios ofertados en el campus universitario.

Palabras clave: *iULE; Ingeniería del software; Proyectos; Informática*

1. Introducción

Los Smartphone están cada vez más presentes en nuestra vida diaria, cada día se pueden oír noticias sobre alguna aplicación nueva para los principales sistemas operativos de los Smartphone y también algún tipo de batalla entre empresas por quien tiene las patentes de cada pequeña parte de los dispositivos actuales.

El desarrollo de las aplicaciones para Smartphone es cada vez más extenso, Apple ya cuenta con más de ochocientas mil aplicaciones en su AppStore para iPhone e iPad, Android llegará en breve también a las ochocientas mil y Windows Phone está creciendo a pasos agigantados en el último año, pasando de ciento cincuenta mil aplicaciones (Apple, 2013) (Google, 2013).

Lo más importante en el desarrollo de aplicaciones para Smartphone es adaptar una idea o modelo de negocio complejo a una aplicación intuitiva y fácil de usar.

El prototipo implementado es un Framework para la gestión académica en la Universidad de León, pero que fácilmente y con unas simples modificaciones se puede adaptar a cualquier organización, obteniendo como resultado una aplicación personalizada para Smartphone en un espacio muy corto de tiempo.

2. Objetivos y metodología

El objetivo general de este proyecto es desarrollar un prototipo que permita crear un framework corporativo para Smartphone de manera simple y rápida, permitiendo a sus potenciales usuarios colocar a sus organizaciones en la vanguardia actual de las nuevas tecnologías y de la Web 2.0 mediante el acceso a su información corporativa a través de dispositivos móviles. Para el desarrollo de este proyecto se decidió desarrollar a modo de proyecto piloto un prototipo en la Universidad de León que permita realizar gestiones académicas.

Figura 1: Capturas de pantalla del prototipo iULE



Para el desarrollo de los objetivos anteriores se ha utilizado una metodología de Ingeniería del Software basada en el desarrollo de prototipos, mientras que el proyecto ha sido gestionando y planificado siguiendo lo establecido en el PMBOK (Project Management Body of Knowledge) del PMI (Project Management Institute) (Pressman, 2010) (PMI, 2013).

Las etapas en las que se ha dividido el proyecto son (Pressman, 2010):

1. Primera etapa. Definición de especificaciones. Esta primera etapa tiene por objeto auditar toda la información relativa a la idea del desarrollo del prototipo, para determinar y justificar la necesidad del mismo. A continuación se realizará un análisis previo con el fin de establecer la amplitud y el calendario del proyecto, estimándose el esfuerzo necesario y el tiempo de desarrollo, e identificando los procesos involucrados en el mismo. Después se construirá el prototipo inicial, no necesariamente operativo, construyéndose macro-modelos de actividad para cada uno de los procesos identificados en la actividad anterior. La intención es disponer de la información necesaria para recabar la aprobación para comenzar el desarrollo. Como final de esta fase, y de todas las demás fases, se emitirá un informe del estado del proyecto.
2. Segunda etapa. Diseño conceptual. El objetivo de esta fase es construir un modelo de información que refleje el esquema conceptual del prototipo. Se realizarán entrevistas a los usuarios finales, y se estudiará y diseñará el primer prototipo operativo, determinando sus puntos fuertes, y sus puntos débiles, documentando todas sus funcionalidades. En esta fase también se prepararán los planes de implantación, formación y pruebas, y se desarrollará tanto el manual de usuario, como el manual técnico.
3. Tercera etapa. Desarrollo del prototipo. Esta fase tiene por objeto la construcción del primer prototipo operativo de la aplicación. Esta fase a su vez está formada de otras dos fases: una fase de desarrollo técnico, y otra para el desarrollo de la documentación asociada. Al final de esta fase se dispondrá de un prototipo totalmente funcional y operativo, que será sometido a múltiples pruebas en la siguiente fase para comprobar su validez.
4. Cuarta etapa. Pruebas de usuario. En esta fase se realizarán todas las pruebas necesarias para validar el prototipo desarrollado en la fase anterior. Si como resultado de estas pruebas se detectara la necesidad de modificar el prototipo, para corregir defectos o para añadirle funcionalidad, se volvería a la etapa anterior y se realizarán todas las iteraciones necesarias hasta que el usuario final se sienta satisfecho con el prototipo, y compruebe que responde a las especificaciones que se habían alcanzado inicialmente. Las pruebas en esta fase serán de dos tipos: pruebas dirigidas, donde los desarrolladores guían y asesoran al usuario durante las mismas, y pruebas no dirigidas, donde el usuario actúa libremente y sin la presencia de los desarrolladores.
5. Quinta etapa. Implantación. En esta fase se ejecutará un mini proyecto piloto en el que usuarios finales puedan probar las funcionalidades del prototipo en un entorno real.
6. Sexta etapa. Auditoría y seguimiento. Se realizará una auditoría del rendimiento y de la calidad de la aplicación, determinando los mecanismos necesarios para realizar modificaciones, y labores de mantenimiento en la aplicación.

Para terminar este apartado, destacar las características principales que deberá de cumplir el prototipo aquí planificado:

- El prototipo debe ser funcional.
- El prototipo deber ser creado rápidamente.
- El prototipo debe evolucionar constantemente.
- El prototipo debe obtener retroalimentaciones para su mejora.

- El prototipo debe tener un coste bajo de desarrollo.

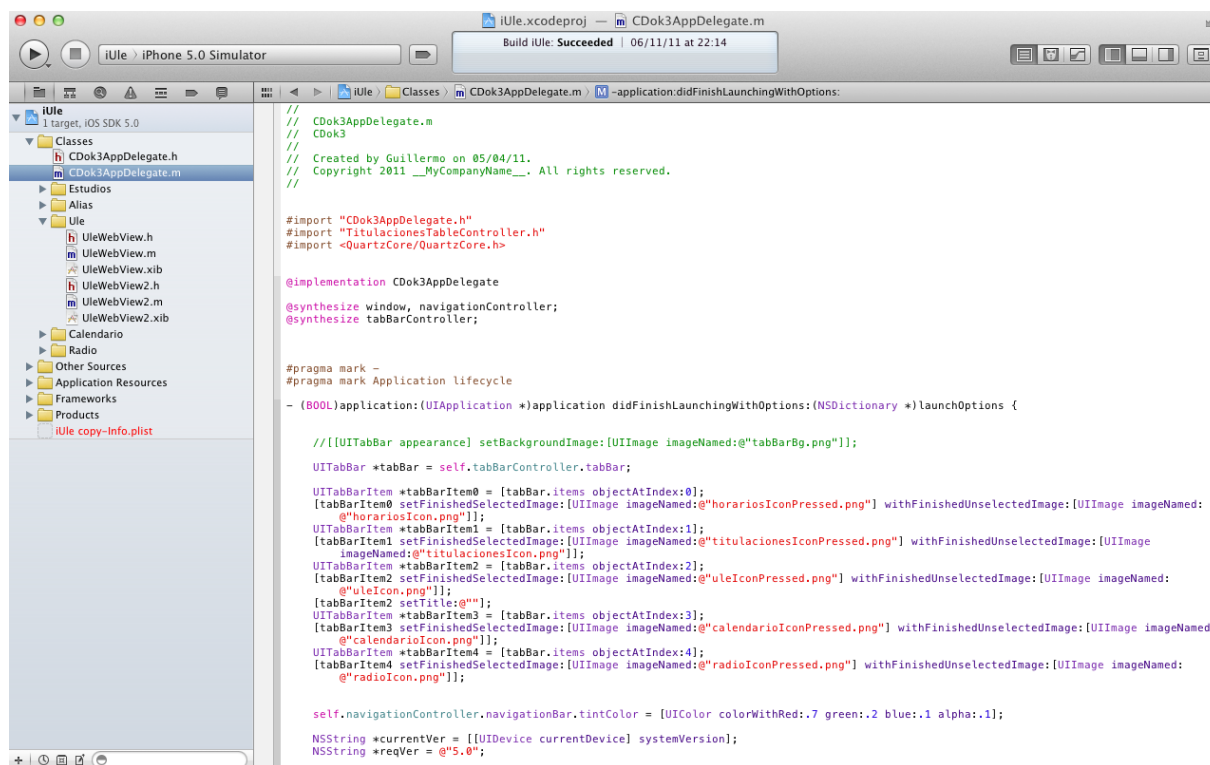
2.1. Motivación

La motivación de los promotores de este prototipo (Área de Proyectos de Ingeniería de la Universidad de León) para desarrollar y poner en marcha esta idea, y su posterior comercialización fue muy elevada, motivo por el cual se tomó la decisión de presentar el proyecto al III Concurso de Prototipos de la ULE (Universidad de León), ya que en el caso de resultar ganadores de este concurso, la Fundación General de la Universidad de León y de la Empresa (FGULEM) colaboraría en la elaboración de un plan de negocio del proyecto, permitiendo así, conocer las posibilidades reales de éxito de la comercialización de este prototipo, y en el caso de un resultado positivo, iniciar todos los procedimientos para crear una empresa que pudiera explotarlo.

3. Especificaciones técnicas

El prototipo ha sido implementado utilizando el entorno de desarrollo Xcode bajo plataforma Mac OS, ya que en principio fue desarrollado pensando en su funcionamiento en dispositivos de la marca Apple tales como iPhone, iPod e iPad. En su codificación se han utilizado lenguajes de programación orientados a objetos como Objective-C que es un súper-conjunto de C basado en el envío de mensajes a instancias de objetos, en lugar de llamadas directas a métodos, concepto clave a la hora de programar una aplicación ágil para terminales escasos en recursos como son los terminales móviles. Objective-C al ser un súper-conjunto de C proporciona que una gran parte del prototipo este también escrita en C junto con el uso de librerías escritas en C++ debido a la versatilidad de este (Kochan, 2004).

Figura 2: IDE de desarrollo Xcode



iULE es un software dedicado a la plataforma Apple, pero su arquitectura esta desacoplada de tal modo que el desarrollo de la aplicación de este framework en cualquier otra plataforma sería rápida, ágil y limpia, siendo tan solo necesario desarrollar un UI dedicado a cada cliente conectando el mismo mediante su controlador, al modelo de datos, convirtiendo así la aplicación en nativa y ganando eficacia y rendimiento frente a aplicaciones Cross-Platform que se cargan con funcionalidades dedicadas en todas sus plataformas.

3.1 Lenguajes de programación

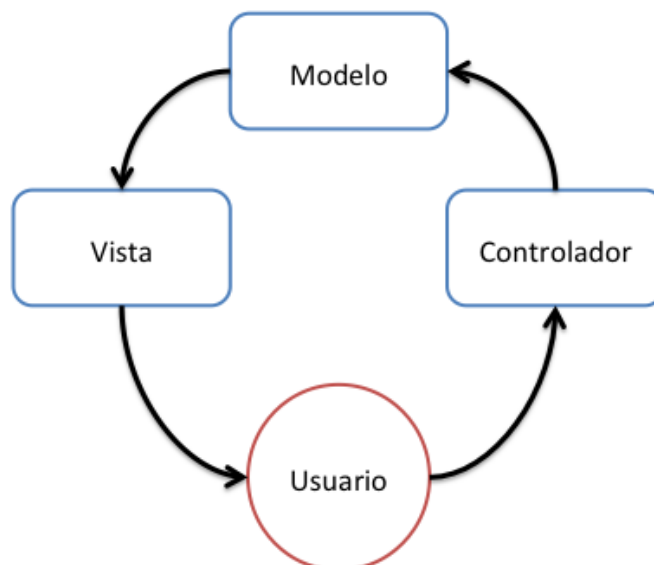
Tal y como se señaló anteriormente el prototipo iULE ha sido desarrollando utilizando los lenguajes de programación más ágiles para el uso en terminales móviles tales como Objective-C, C y C++ (Mark, Nutting, & LaMarche, 2011).

Objective-C es un lenguaje de programación orientado a objetos creado como un súper-conjunto de C para que implementase un modelo de objetos parecido al de Smalltalk. Originalmente fue creado por Brad Cox y la corporación StepStone en 1980. En 1988 fue adoptado como lenguaje de programación de NEXTSTEP y en 1992 fue liberado bajo licencia GPL para el compilador GCC. Actualmente se usa como lenguaje principal de programación en Mac OS X, iOS y GNUstep (Frederick & Lal, 2010).

El modelo de programación orientada a objetos de Objective-C se basa en enviar mensajes a instancias de objetos. Esto es diferente al modelo de programación al estilo de Simula, utilizado por C++ y ésta distinción es semánticamente importante. En Objective-C uno no llama a un método; uno envía un mensaje, y la diferencia entre ambos conceptos radica en cómo el código referido por el nombre del mensaje o método es ejecutado. En un lenguaje al estilo Simula, el nombre del método es en la mayoría de los casos atado a una sección de código en la clase objetivo por el compilador, pero en Smalltalk y Objective-C, el mensaje sigue siendo simplemente un nombre, y es resuelto en tiempo de ejecución: el objeto receptor tiene la tarea de interpretar por sí mismo el mensaje. Una consecuencia de esto es que el mensaje del sistema que pasa no tiene chequeo de tipo: el objeto al cual es dirigido el mensaje (conocido como receptor) no está inherentemente garantizado a responder a un mensaje, y si no lo hace, simplemente lo ignora y retorna un puntero nulo (Hillegass & Preble, 2011) (Kochan, 2004).

Figura 3: Arquitectura del prototipo (Modelo Vista Controlador)

Arquitectura **iULE**



La Programación orientada a objetos al estilo Smalltalk permite que los mensajes no posean implementación - por ejemplo, toda una colección de objetos pueden enviar un mensaje sin temor a producir errores en tiempo de ejecución. El envío de mensajes tampoco requiere que un objeto sea definido en tiempo de compilación.

Sin embargo, se debe notar que debido a la sobrecarga de la interpretación de los mensajes, un mensaje en Objective-C toma, en el mejor de los casos, tres veces más tiempo que una llamada a un método virtual en C++.

3.2 Arquitectura

Desarrollar aplicaciones para dispositivos móviles es muy distinto al desarrollo de aplicaciones de escritorio pese a que pueda resultar muy similar, sin embargo pensar que el desarrollo móvil pueda ser semejante al web es acertado ya que ambos comparten la misma base de arquitectura que componen tanto una aplicación web como una móvil, se trata del patrón Modelo-Vista-Controlador (MVC) (Frederick & Lal, 2010).

MVC es un patrón o modelo de abstracción de desarrollo de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos. El patrón de llamada y retorno MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.

4. IULE

Tal y como se señaló anteriormente para la puesta en marcha de este proyecto se tomó la decisión de desarrollar un proyecto piloto que permitiera poner en marcha en un ámbito real de trabajo el prototipo diseñado.

Para llevarlo a cabo se decidió utilizar a la Universidad de León como cliente, y a los alumnos de la asignatura Dirección de Proyectos del Grado en Ingeniería Informática de la Universidad de León como usuarios finales de la misma, por lo que además por un lado se dotó a la Universidad de León de una nueva herramienta informática y por otro los alumnos de dicha asignatura pudieron completar su currículum con la participación en el desarrollo y las pruebas de un proyecto real (Alfonso Cendón, Alonso Juan, & Rodríguez, 2013).

Figura 4: Captura de pantalla del prototipo iULE en la tienda oficial de Apple (App Store)



Las distintas funcionalidad integradas en el prototipo iULE permiten acceder a toda la información relacionada con la Universidad de León, desde los horarios de las clases, hasta las fechas de los exámenes, pasando por la consulta de las notas de una determinada asignatura, el expediente académico o los días y las horas de tutoría de un determinado profesor.

El prototipo IULE puede ser descargado para su prueba de la tienda online oficial de Apple, el App Store (Alfonso Cendón, Alonso Juan, & Rodríguez, 2013).

5. Modelo de negocio del prototipo

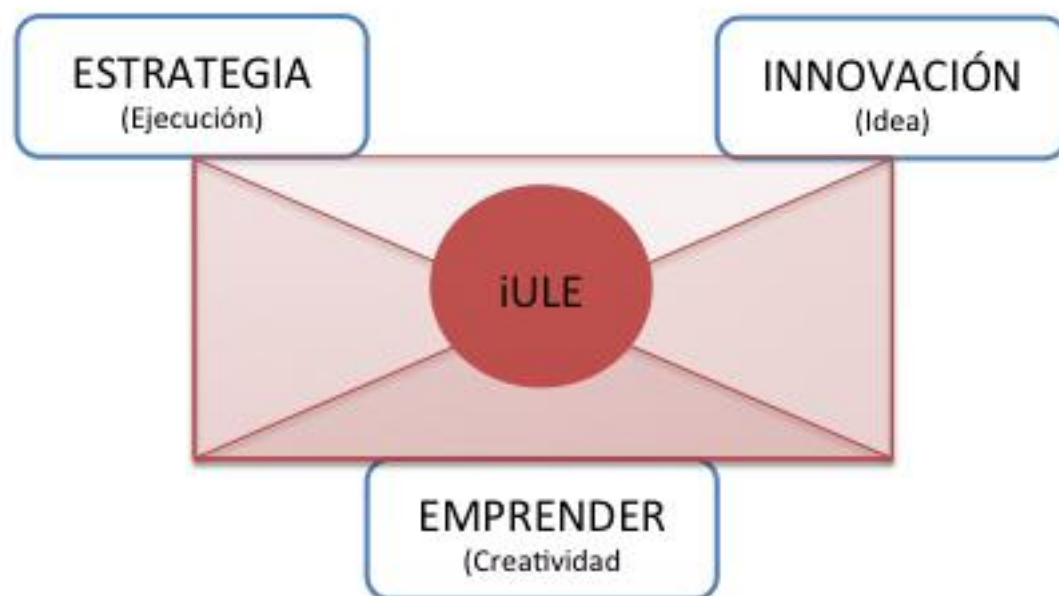
A continuación se detalla el modelo de negocio propuesto para este prototipo, entendiéndolo no ya como el proyecto piloto iULE, sino como un Framework Corporativo personalizable listo para salir al mercado, considerado que la mejor opción para su comercialización es mediante el licenciamiento del mismo, basado en el siguiente modelo:

- 1º Año: Cuota de entrada (Personalización del Framework y licencia anual).
- 2º Año y sucesivos: Licencia anual.

El primer año se pagaría una cuota de entrada que correspondería al concepto de personalización de la aplicación de acuerdo a las necesidades de la organización, así como adaptación a su modelo de identidad corporativa, en los años sucesivos se pagaría una licencia anual que incluiría el mantenimiento de la aplicación, así como la incorporación de todas las mejoras que se vayan desarrollando desde la empresa.

Figura 5: Esquema del modelo de negocio propuesto

Modelo de Negocio **iULE**



Se ha considerado que este modelo de negocio es el más adecuado para este prototipo después de analizar numerosos casos de éxito basados en esta filosofía, teniendo a su vez en cuenta que es un modelo muy utilizado en este tipo de aplicaciones.

También se han establecido una serie de servicios que se podrían ofrecer ligados al prototipo y que se detallan a continuación:

- Características avanzadas. El prototipo en su estado actual cuenta con las características necesarias para su funcionamiento básico, pero existe una amplia lista de mejoras que podrían incluirse en una cartera de servicios bajo demanda.
- Desarrollos a medida. Tal y como se señaló a lo largo de este documento, el prototipo tiene numerosos ámbitos de aplicación, por este motivo resulta un servicio muy atractivo la posibilidad de personalizar su desarrollo adaptándolo a las necesidades específicas de cada usuario.

Los sectores de aplicabilidad de este prototipo son principalmente empresas e instituciones que quieran dotar a sus respectivas organizaciones de un sistema corporativo moderno y ágil, utilizando la última tecnología móvil para establecer la mejor comunicación, Corporación-Cliente y Cliente-Corporación.

Por este motivo se ha considerado como una importante oportunidad de negocio el crear una "Spin Off" de la Universidad de León para poder desarrollar el modelo de negocio propuesto anteriormente.

6. Conclusiones

Actualmente las empresas y las organizaciones están demandando aplicaciones corporativas para dispositivos móviles con el objetivo de reducir su brecha digital e iniciarse en muchos casos en el nuevo paradigma de comunicación de la llamada Web 2.0, en el que la comunicaciones Corporación-Cliente y Cliente-Corporación son de vital importancia, estableciéndose así un importante nicho de mercado que este prototipo pretende cubrir.

La puesta en marcha del proyecto piloto iULE utilizando a usuarios finales ha permitido establecer los requisitos y las características necesarias para definir específicamente el prototipo final con el objetivo de llevar a cabo su futura comercialización.

Este prototipo permite desarrollar rápida y eficazmente aplicaciones corporativas para dispositivos móviles, lo que supone una importantísima ventaja competitiva sobre todas aquellas empresas que se dedican a realizar aplicaciones mediante largos y costosos procesos de desarrollo.

El modelo de negocio propuesto para el prototipo está basado en el licenciamiento, estableciendo una cuota de entrada en la que se incluye la personalización del prototipo, más una cuota anual de mantenimiento y de incorporación de nuevas mejoras.

7. Referencias

- Alfonso Cendón, J., Alonso Juan, I., & Rodríguez, G. (2013). iULE Aplicación para la Universidad de León. [Recuperado 17 abril 2013]. Disponible en <https://itunes.apple.com/es/app/ule/id469616665?mt=8>
- Apple. (2013). Apple Store. [Recuperado 17 abril 2013]. Disponible en <http://www.apple.com/es/iphone/>
- Frederick, G., & Lal, R. (2010). Beginning Smartphone Web Development: Building JavaScript, CSS, HTML and Ajax-based Applications for iPhone, Android, Palm Pre, BlackBerry, Windows Mobile and Nokia S60. U.S.A.: Apress.
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (2009). Elements of reusable object-oriented software. Addison Wesley Profesional Computing Series.

- Google. (2013). Google Play. [Recuperado 17 abril 2013]. Disponible en <https://play.google.com/store>
- Hillegass, A., & Preble, A. (2011). Cocoa Programming for Mac OS X (4th Edition). Addison-Wesley.
- Kochan, S. (2004). Programming in Objective-C. U.S.A: Sams Publishing.
- Mark, D., Nutting, J., & LaMarche, J. (2011). Beginning iPhone 4 Development iOS SDK. U.S.A.: Apress.
- PMI. (2013). A Guide to the Project Management Body of Knowledge (PMBOK® Guide). Project Management Institute.
- Pressman, R. (2010). Ingeniería del Software (7ª Edición). Madrid: McGraw-Hill.