

## Aplicación móvil en iOS para la gestión de la planeación y evaluación escolar

MACIAS-BRAMBILA, Hassem Rubén†\*, ZAMORA-RAMOS, Víctor Manuel, OSORIO-ÁNGEL, Sonia y ALTAMIRANO-MORELOS, Dan Emmanuel

*Universidad Autónoma de Guadalajara. Av. Patria 1201, Lomas del Valle, 45129 Zapopan, Jalisco*

Recibido Enero 09, 2017; Aceptado Marzo 27, 2017

### Resumen

El presente artículo describe el proceso de desarrollo tecnológico de un prototipo de aplicación móvil para la gestión de la planeación y evaluación escolar en la Universidad de Guadalajara, la cual se implementó para la plataforma iOS. El proceso de ingeniería de software en el desarrollo requirió de una metodología de desarrollo ágil la cual permitió la gestión de cambios adecuada al proyecto. Asimismo se contempla dentro del desarrollo la recolección de datos, la especificación de requerimientos, el diseño arquitectónico, el diseño semántico de datos, el diseño de la aplicación, la configuración del servidor y las pruebas. Esta aplicación tiene como objetivo fungir como un sistema integral de gestión de planeación de cursos, brindando a los docentes de la institución las herramientas necesarias para el perfeccionamiento y automatización de los procesos relacionados con el control administrativo de cursos, así mismo brindar una serie de herramientas para facilitar la evaluación de los estudiantes.

**Aplicación móvil, iOS, gestión de la planeación escolar, evaluación escolar**

### Abstract

This article describes the technological development process of a mobile application prototype for school planning and evaluation management at the University of Guadalajara, which was implemented for the iOS platform. The process of software engineering in development required an agile development methodology that allowed the appropriate change management to the project. Data collection, specification of requirements, architectural design, semantic data design, application design, server configuration and testing are also included within the development. This application aims to serve as an integral system of management of course planning, providing the institution's teachers with the necessary tools for perfecting and automating the processes related to the administrative control of courses, as well as providing a series of tools for Facilitate the evaluation of students.

**Mobile application, iOS, school planning management, school evaluation**

**Citación:** MACIAS-BRAMBILA, Hassem Rubén, ZAMORA-RAMOS, Víctor Manuel, OSORIO-ANGEL, Sonia y ALTAMIRANO-MORELOS, Dan Emmanuel. Aplicación móvil en iOS para la gestión de la planeación y evaluación escolar. *Revista de Tecnologías Computacionales*. 2017, 1-1: 53-61.

\*Correspondencia al Autor (Correo Electrónico: hassem.macias@academico.udg.mx.)

† Investigador contribuyendo como primer autor.

## Introducción

La implementación de las Tecnologías de la Información y Comunicación (TIC's) en los procesos del hecho educativo es necesaria, ya sea como un apoyo a las actividades del aula, para maximizar el alcance de los Recursos Educativos, como instrumentos que permitan la inclusión ó simplemente como agente de motivación y/o cambio. Es tanta la necesidad de la implementación de estas tecnologías, que las políticas públicas en el Plan Nacional de Desarrollo (2012-2018) o el Plan Sectorial de Educación (2012-2018), o el Plan de Desarrollo Institucional 2014-2030 de la Universidad de Guadalajara (UdG) contemplan acciones y estrategias que están dirigidas a la implementación de dichas tecnologías para los agentes educativos involucrados en la enseñanza y el aprendizaje.

Derivado de esta necesidad se planteó el desarrollo de una aplicación móvil con el objetivo de ofrecer una herramienta para la gestión de los procesos de la evaluación continua de la Universidad de Guadalajara, específicamente dentro del Centro Universitario de Ciencias Exactas e Ingenierías en la División de Computación y Electrónica. Esta aplicación será desarrollada para el Sistema Operativo iOS, específicamente el diseño y desarrollo, así como las pruebas se planean para un dispositivo como las tabletas de la marca Apple, para los diseños mini y Pro. Así mismo la aplicación contará con un modulo de conexión y sincronización de datos con iCloud lo cual permitirá almacenar los datos de una forma eficiente y segura

El desarrollo de la aplicación requirió de la planificación de las diversas actividades que conlleva un plan de desarrollo de software, teniendo un mejor control de tareas asignadas para el modelado y presentación de sistema.

Dichas tareas fueron administradas correctamente a través de instrumentos de desarrollo, lineamientos y estándares que permiten asegurar la calidad en el desarrollo y planificación del software, una de ellas es el estándar 830 del Institute of Electrical and Electronics Engineers (IEEE) en su revisión de 1998. Además, se implementaron herramientas que facilitan el modelado arquitectónico de software y el modelado semántico de los datos siguiendo los lineamientos de modelado a través de diversos diagramas establecidos por el Lenguaje Unificado de Modelado (UML).

El desarrollo y proceso de pruebas de la aplicación contempla el uso del entorno de desarrollo Xcode el cual fue elegido porque ofrece soporte nativo para el desarrollo de aplicaciones iOS y lenguaje de programación Swift en su versión 3, el cual fue usado para el desarrollo de la misma. Asimismo, todas las pruebas realizadas durante el desarrollo del software están acompañadas de sus respectivos resultados y parámetros de prueba establecidos, además de la documentación correspondiente al proceso y planeación de pruebas, manuales de usuario, así como manuales técnicos que establecen un plan de capacitación para el uso y mantenimiento de la aplicación.

## Estado del Arte

La implementación de las TIC's en los procesos de gestión permite que sea más eficiente y eficaz, a través de su sistematización. Actualmente existen esfuerzos institucionales dentro de las Universidades, Institutos y Centros Educativos para modernizar dichos procesos incorporando al proceso de formación integral de los estudiantes las herramientas más avanzadas de tecnologías para complementar y fortalecer su desarrollo académico y asimismo apoyar a los docentes con las actividades administrativas que requieren sus labores en la institución.

Actualmente, existen herramientas informáticas que facilitan las tareas y actividades de planeación y valoración de los estudiantes, los cuales al ser aplicaciones comerciales cuentan con configuraciones establecidas, así como procedimientos y dinámicas rígidas, lo que obliga a las instituciones educativas que las adquieren a modificar sus procesos de administración, seguimiento, evaluación y consulta de acuerdo con las características de las aplicaciones disponibles en el mercado, por lo que el desarrollo de software a la medida ofrece una gama de posibilidades hacia la personalización de dichos procesos.

La creación de software contempla en el aseguramiento de la calidad de sus procesos el uso y aplicación de buenas prácticas en el desarrollo, pruebas y mantenimiento, así como la implementación de estándares, lo cuales aseguran que los procesos realizados son los adecuados. Es por eso que dentro de la etapa de ingeniería de requerimientos, especialmente para la elaboración del documento de especificación de requerimientos de software (ERS), existen estándares como el IEEE 830 revisión 1998, el cual provee un marco de desarrollo para el ERS y además, de acuerdo con la IEEE (2016) establece los requerimientos funcionales del modelo de negocio, provee un marco contextual: procedimientos, mecanismos y colaboración entre todos los actores involucrados y proporciona las directrices para el cumplimiento del estándar IEEE/IEA 12207 revisión 1997. Asimismo también se contempló el estándar ISO/IEC 15504.

La implementación de estos estándares dentro del marco de desarrollo es contemplado como solo una fase de la ingeniería de software, misma que es definida como la aplicación práctica del conocimiento científico al diseño y construcción de programas de computadora y a la documentación asociada requerida para desarrollar, operar y mantenerlos.

Este producto de software durante el proceso de desarrollo requiere de la implementación de herramientas, que asistirán a los analistas, diseñadores, programadores e ingenieros de prueba en sus actividades. Estas herramientas son denominadas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora), tales como, Microsoft Visio y Dia para el modelado arquitectónico del software y el semántico de los datos, así como Visual Studio aplicado a Xamarin para el desarrollo de la aplicación y SQLite como gestor de la base de datos.

### **Metodología de desarrollo**

El desarrollo del proyecto de la aplicación móvil para la gestión de la planeación y evaluación escolar para la plataforma iOS, fue dividido en cuatro fases, las cuales se describen a continuación:

#### **Recolección de datos**

Entrevista, encuestas y elaboración de plan de trabajo. De acuerdo a las prácticas establecidas en la Ingeniería de requerimientos se planea y establece una sesión de entrevistas con el cliente que solicita el software con el fin de reunir toda la información de manera abstracta relacionada al producto a desarrollar, se planea conocer los requisitos a cumplir, el modelo de negocio, aspectos básicos de la aplicación, sistemas en los que se usará, requisitos para ejecutarlo, compatibilidad, comportamiento con otros sistemas, aspectos de inicio de sesión, seguridad, manejo de los datos, toda información requerida para poder redactar los requisitos específicos para la aplicación.

Se redactó una lista de especificaciones y funcionalidades que se establecieron en un documento, dichas sentencias fueron esclarecidas por el cliente, una vez que se elaboró el reporte de entrevista y se recolectó toda la información necesaria organizada por categorías, dicho documento fue presentado al analista de software, posteriormente se eligieron personas afines al tipo de usuario de la aplicación para contestar encuestas sobre aspectos que se consideran que debería cumplir una aplicación de este tipo.

Todas las fechas de las entrevistas fueron planeadas y se realizaron en el tiempo acordado por los participantes del equipo desarrollador con base a la información recabada en las entrevistas se procedió a redactar una encuesta publica con fines de mejorar la calidad y aspecto de la aplicación. Después se desarrolló un planteamiento de las actividades y procesos para el desarrollo y documentación del software, así como el modelado mediante un diagrama de Gantt y un diagrama de Pert para el seguimiento y control de tareas asignadas en el grupo de trabajo.

**Especificación de requerimientos**

Una vez completadas cada una de las actividades de recolección de datos se procedió al llenado de la plantilla que ofrece el estándar IEEE 830 revisión 1998, este documento contiene la especificación de los requerimientos, en él se incluyen tanto requerimientos de usuario, requerimientos funcionales como no funcionales, los cuales permiten establecer el ¿qué va a hacer la aplicación? y ¿cómo lo va a hacer?, es decir; todo lo que necesita cumplir el software, sus características, limitantes operacionales, plan de negocio, perspectiva del producto, tipos de usuarios involucrados y requisitos de hardware, esta información es de gran importancia para el diseño y planificación de la aplicación.

A continuación se presenta un extracto del capítulo 3 del documento de Especificación de Requerimientos de Software (SRS):

**3 Requisitos específicos**

Número de requisito	RF01
Nombre de requisito	Toma de asistencia
Tipo	Requisito
Fuente del requisito	Descripción del proyecto
Prioridad del requisito	Alta/Esencial

**Descripción:** El programa deberá permitir la toma de asistencia de todos los alumnos registrados en la materia seleccionada.

Número de requisito	RF02
Nombre de requisito	Registro de actividades
Tipo	Requisito
Fuente del requisito	Descripción del proyecto
Prioridad del requisito	Alta/Esencial

**Descripción:** El programa deberá permitir registrar las actividades completadas por los alumnos, estas actividades pueden ser practicas o tareas, además de estar organizadas por fecha.

**Figura 1** Lista de requerimientos preliminares obtenidos de la entrevista para el desarrollo de la aplicación.

*Fuente: Elaboración propia*

**Diseño Arquitectónico de Software**

Se desarrollaron diversos diagramas que resultaron de gran utilidad para el cliente y el equipo de desarrollo del software, los diagramas fueron elaborados con el software Visio de Microsoft siguiendo los lineamientos de la versión 2.5 del Lenguaje Unificado de Modelado (UML). Así mismo se elaboró el diagrama de bloques el cual ofrece una vista gráfica del funcionamiento interno de la aplicación, de sus procesos, entradas y salidas, dicho diagrama fue mostrado al cliente para una revisión preliminar de las características del software a considerar. A continuación se presenta dicho diagrama:



**Figura 2** Diagrama de bloques para la aplicación.

*Fuente: Elaboración propia*

Una elemento importante a considerar durante el modelado del sistema son las tarjetas Clase – Responsabilidad -Colaboración (CRC) ya que permite representar de forma clara las conexiones y referencias existentes en las clases del sistema, sus funciones principales y comunicación con otras, logrando una perspectiva general del sistema, así como de sus componentes asociados, lo que permite que se establezcan las bases para realizar el diagrama de clases de acuerdo a UML 2.5, con cada uno de los módulos involucrados en el funcionamiento del sistema en donde se presentan por primera vez los atributos a utilizar con su respectivo dominio y tipo, además de las relaciones que existen entre ellas, las cuales se especifican en el diccionario de clases con el fin de establecer todos los detalles y características que no se modelaron en el diagrama, como recurso de utilidad para el equipo de desarrollo, se implementa el diagrama de objetos sirviendo como una guía de referencia ofreciendo una vista preliminar de los posibles datos que serán almacenados en las clases del sistema.

Otra herramienta de gran utilidad para la comprensión del funcionamiento y comportamiento de sistema son los casos de uso, estos ofrecen una descripción del comportamiento del sistema, el plan de negocio empleado, además del comportamiento del sistema con factores externos tales como empleo de software adicional o los diferentes tipos de usuarios que utilizarán el sistema.

Esta información detallada permite a su vez la definición de los aspectos dinámicos del sistema, como lo es la comunicación entre las instancias durante cada uno de los procesos definidos en los casos de uso, para lo cual se diseñaron los diagramas de secuencia y colaboración, los que permiten al equipo desarrollador definir el proceso de comunicación y por lo tanto los mensajes que son enviados y recibidos por dichas instancias.

A continuación se presenta el diagrama de secuencia del caso de uso crear curso.

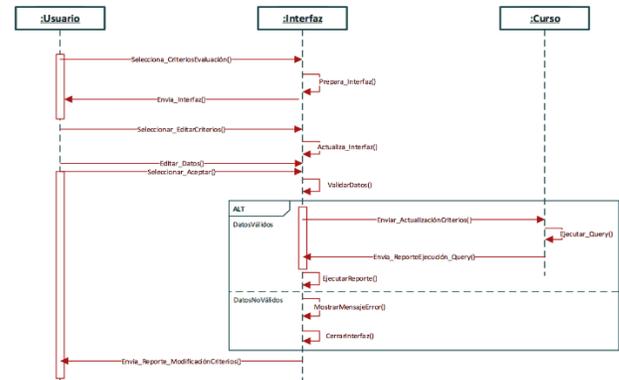


Figura 3 Diagrama de secuencia para el evento editar criterios de evaluación

Fuente: Elaboración propia

Otra elemento para el modelado de las interacciones del sistema es el diagrama de estados, el cual permite visualizar la ejecución de cada uno de los procesos del sistema, representar los sucesos respecto al tiempo en un evento, la respuesta y acciones del sistema en determinada circunstancia, explica todos los posibles estados por los que el sistema puede transitar.

Asimismo se modeló mediante el diagrama de despliegue y de componentes la estructura de dependencia con otros módulos propios o externos del sistema, restricción de archivos, distribución y almacenamiento de datos, además de las conexiones a servidores o internet.

### Diseño semántico de datos

Esta fase está enfocada a la vista y manejo de los datos del sistema, se incluyó la elaboración del diagrama Entidad-Relación y el Modelo Relacional para la base de datos.

En la creación de la base de datos se especificaron las condiciones y restricciones de integridad haciendo uso de llaves primarias a los códigos únicos que representan una entidad y llaves foráneas para especificar una relación entre dos tablas, estas restricciones son de utilidad para identificar los registros entre los catálogos de datos, además de asegurar que la información sea almacenada de forma segura evitando errores imprevistos o eliminaciones accidentales de registros que tengan una relación con otros, con base al diseño conceptual se procede a crear un diccionario de datos en donde se especifica el nombre y descripción de la tabla, además del nombre de los atributos, tamaño, tipo de dato, descripción del contenido del atributo y el número de requerimiento en donde se hace mención en el documento de especificación de requerimientos IEEE 830-1998.

Una vez completado el diseño conceptual de la base de datos se procedió a la creación de un script con sentencias SQL para generar la base de datos y permitir al grupo de desarrollo de la aplicación conocer las restricciones y así diseñar las consultas necesarias para cada módulo dentro de la aplicación.

### **Diseño de interfaces**

Para el diseño de la colorimetría a emplear, logotipo de la Universidad y recursos gráficos para la aplicación se utilizó el software Adobe Illustrator CC, en cuanto al manejo de objetos y organización de pantallas se usó Adobe Photoshop CC estableciendo como unidad de medida pixeles y una resolución de 2048 x 1536 pixeles con una densidad de 264 pixeles por pulgada.

Las pantallas generadas fueron el inicio de sesión, icono de aplicación, banner publicitario para la tienda de aplicaciones App Store, iconos de pestañas, iconos de alumnos, ventanas genéricas para registro de datos, eliminación, modificación y consultas, además de los botones encargados de dichas acciones.

### **Configuración del servidor**

Para el manejo de los datos en la aplicación se habilitó la herramienta que proporciona el entorno de desarrollo Xcode llamada Core Data el cual permite establecer una conexión con el servidor de iCloud y sincronizar los datos en la nube, para ello fue necesario habilitar la opción en la creación del proyecto y habilitar el documento .xcdatamodeld dentro de la carpeta del mismo.

### **Diseño de vistas**

Para este apartado se crearon vistas preliminares del producto final con ayuda de la herramienta Adobe Photoshop CC para modelar las pantallas haciendo uso de los recursos gráficos y lineamientos de diseño como colores, tipo y tamaño de fuente, botones, iconos y márgenes previamente establecidos. El diseño de las pantallas se basa en los recursos de diseño disponibles en el entorno de desarrollo Xcode, así como en las restricciones de espaciado entre objetos.

### **Programación y conexión vista-controlador**

Para la creación del modelo vista-controlador se implementó la herramienta storyboard dentro del entorno de desarrollo Xcode, dicha herramienta permite la creación y diseño de ventanas, así como sus restricciones de tamaño de objetos, textos, disposición pantalla horizontal y vertical, así como la posibilidad de adaptarse a la función de pantalla dividida reorganizando el contenido de la pantalla.

Una vez diseñado el storyboard, se procede a crear clases encargadas de cada pantalla del sistema y vincularlas a cada una de ellas permitiendo la conexión entre ventanas y animaciones.

Posteriormente se crean las clases del sistema siguiendo una metodología de programación orientada a objetos con encapsulamiento de las variables, métodos para la inicialización de variables y herencia entre clases. Estas clases se encargarán de todos los datos del sistema, de la extracción de los datos obtenidos de las clases vista-controlador y de la comunicación con la base de datos usando métodos para consulta de información, inserción de registros, modificación, eliminación, y creación de copias de seguridad en iCloud.

Como parte de una buena práctica de programación las clases del sistema se documentan con información de ayuda a los programadores para futuras mejoras, optimizaciones o corrección de errores en la aplicación.

### **Pruebas y mantenimiento de software**

Conforme a la planeación y diagramas de seguimiento de actividades al final del desarrollo de los módulos del sistema se realizó una serie de pruebas para asegurar la calidad del módulo en el cual se estaba trabajando, una vez concluido el desarrollo y ejecución de las pruebas unitarias del sistema se elaboró un documento de plan de pruebas en donde se incluye el historial de todas las versiones que ha tenido el sistema, así como las aprobaciones de cada una de ellas, se describe el alcance de las pruebas, funciones a no probar y el enfoque que tiene cada una de ellas describiendo detalladamente su proceso, hasta qué punto se acepta la prueba o si será rechazada.

Para la aplicación se implementarán pruebas funcionales y pruebas de stress, al término de cada prueba se describió el resultado y los errores encontrados con el fin de preparar una solución, esto asegura la calidad final del producto.

Posteriormente se redactó un documento con la descripción del proceso y las técnicas para dar seguimiento al software, en un plan de mantenimiento, con el que se pretende mejorar la calidad del producto final, ya que el mantenimiento se puede hacer combinando herramientas de software, métodos y técnicas sujetas a los antecedentes de la aplicación de software desarrollado, además de que brindará material de consulta y de resolución de problemas respecto a la aplicación. En dicho documento se especificaron todos los procesos para dar seguimiento al software, así como las fases y procesos de implementación de módulos o funcionalidades extra que serían agregadas en un futuro.

Al término del desarrollo de la aplicación es necesario proveer al usuario y al equipo de desarrolladores que se encargará de dar mantenimiento a la aplicación su respectivo manual en los cuales se incluye el resumen de todo el contenido y estructuración de la aplicación, además describe cuál es su propósito y forma de operar, estos documentos se elaboraron con el fin de especificar qué elementos o componentes del producto final estarán disponibles para modificaciones o mejoras posteriores por parte del equipo de desarrollo, la funcionalidad de la aplicación cumple con los requisitos descritos en el documento IEEE 830-1998 de especificación de requisitos de software, desde los requerimientos mínimos de software y de hardware para su instalación.

Además se incluyeron capturas de pantalla del proceso de búsqueda y descarga de la tienda de aplicaciones App store, el proceso de ingreso a la aplicación, así como cada una de las funcionalidades que realiza el sistema, en el caso del manual técnico se incluyen toda la documentación y referencias de ayuda para la aplicación

## Resultados

Durante el proceso de pruebas de la aplicación se seleccionaron usuarios aleatoriamente para llevar a cabo las pruebas, en las pruebas de stress los resultados fueron satisfactorios demostrando la velocidad y fiabilidad del sistema en situaciones donde se requiere más consumo de recursos de hardware, esta prueba se realizó en tres dispositivos iOS distintos, el primer dispositivo de pruebas fue un iPad Air con sistema operativo iOS 9 en donde solo presento un rendimiento inferior al sobrepasar los 63,000 registros, el segundo dispositivo fue un iPad Air 2 con sistema operativo iOS 10 en este dispositivo se notó un mayor rendimiento y no se presentó disminución en el rendimiento incluso al concluir con la carga total de datos insertados, por último el tercer dispositivo fue un iPad Pro 9.7 con sistema operativo iOS 10 el cual en todo el proceso de pruebas mostro un rendimiento fluido al presentar grandes cargas de datos.

Se concluye que el sistema soporta los principales modelos de iPad y los dos últimos sistemas operativos más recientes, en tanto a las pruebas funcionales solo se presentaron errores en la extracción y exportación de datos desfasando los márgenes del documento, dichos errores fueron notificados al equipo de desarrollo del sistema para llegar a la solución.

## Conclusiones

La implementación de la tecnología en las actividades inherentes de la docencia permite que los procesos de gestión de los cursos sean exactos y se disminuye el tiempo que el profesor invierte en dichos procesos, así mismo que crea un puente que acerca al profesor a la tecnología, esto derivado de la brecha digital que en algunos casos específicos es mayor entre el docente y la tecnología, que entre los estudiantes y la misma. El desarrollo de esta aplicación permite además estandarizar los reportes que el docente entrega como evidencia del trabajo de programación y planeación académica, lo cual abona a los lineamientos que los organismos acreditadores de los programas educativos observan.

## Referencias

- Arias, Ángel & Florez Gonzalo, Enrique (2016) "Curso de programación con iOS". IT Campus Academy: España.
- Bucarek, James (2006) "Beginning Xcode" EUA: Wiley Publishing, Inc.
- Kendall, Kenneth E. & Kendall, Julie E. (2011) "Análisis y Diseño de Sistemas, 8va edición". México: Pearson Education.
- Institute of Electrical and Electronics Engineers. "Estandar IEEE 830 revisión 1998". <https://standards.ieee.org/findstds/estandar/830-1998.html> . Recuperado 03 de Marzo del 2017.
- Sommerville, Ian (2011) "Ingeniería de Software, 9na edición". México: Pearson Education.
- Pressman, Roger S. (2010) "Ingeniería de Software, un enfoque práctico". México: McGraw Hill.

Universidad de Guadalajara.  
[http://www.cgti.udg.mx/noticias/apps\\_moviles-en-la-udeg](http://www.cgti.udg.mx/noticias/apps_moviles-en-la-udeg) . Recuperado el 10 de mayo de 2017.

Zamora Ramos, Victor M.; Macías Brambila Hassem R.; Osorio Angel, Sonia & López Perales, Víctor L. (2016) “Aplicación móvil para la optimización de procesos de la industria restaurantera”. Revista de Estrategias del Desarrollo Empresarial, volumen 2, numero 4. ECORFAN. España.