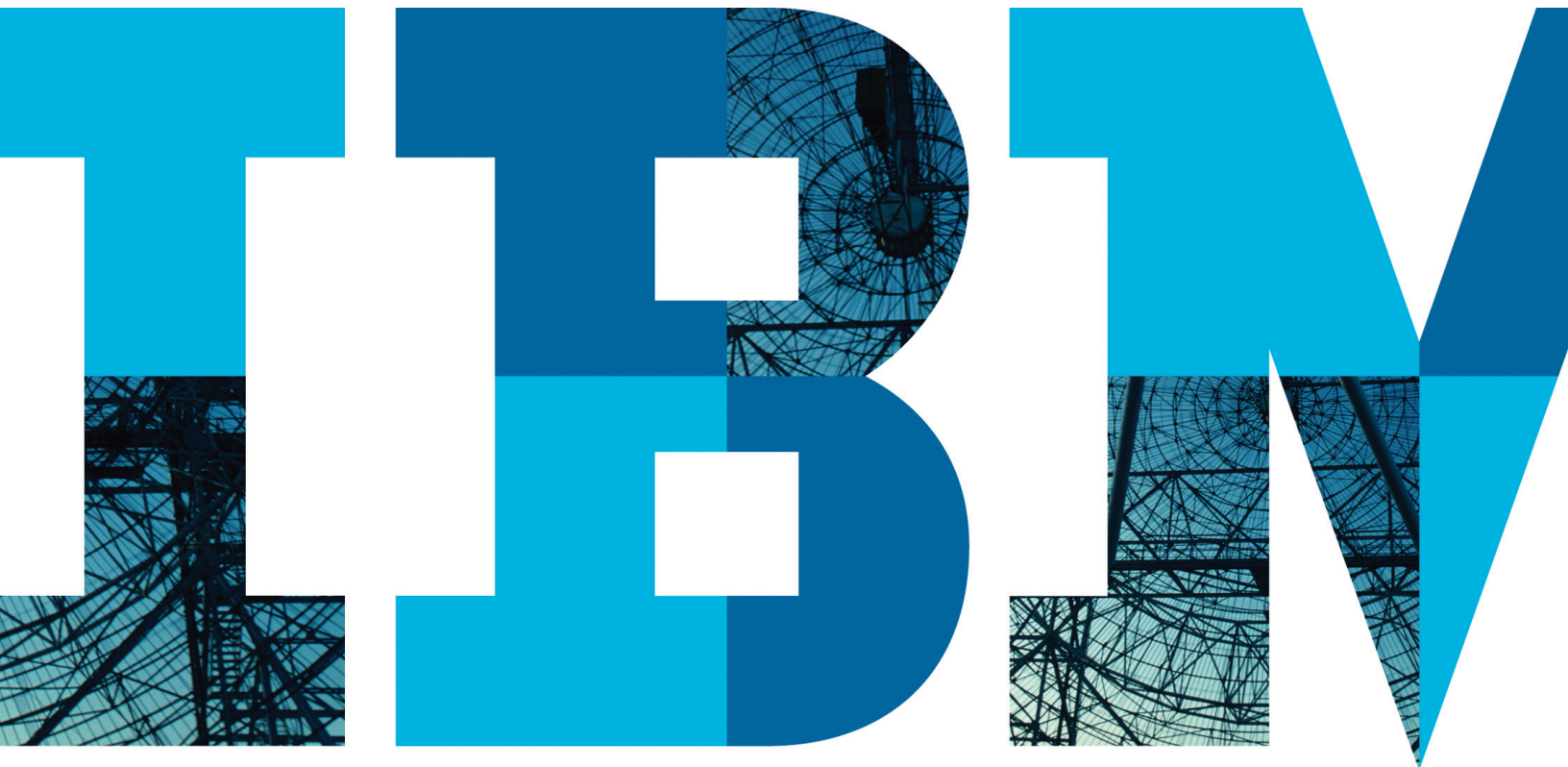


Fundamentos del desarrollo de aplicaciones móviles

Una guía para los equipos empresariales que trabajan en proyectos de aplicaciones móviles



Resumen ejecutivo

Todos los sectores han comenzado a comprender que los destinatarios de sus aplicaciones empresariales han pasado en su mayor parte de utilizar ordenadores personales o portátiles a emplear dispositivos móviles, como smartphones y tablets, para acceder a Internet y obtener la información que buscan. Sucede lo mismo si el destinatario de la aplicación es un cliente directo de la empresa (aplicaciones Business-to-Consumer o “B2C”) o un empleado o socio comercial (aplicaciones Business-to-Employee y Business-to-Business o “B2E” y “B2B”). El número de personas que usan dispositivos móviles, que pueden llevar consigo a cualquier lugar y son extremadamente sencillos de usar, como su medio principal de obtener información y solicitar servicios a través de Internet aumenta en todo el mundo.

Esta transformación crucial en el comportamiento del usuario final ha llevado a las empresas a crear canales móviles para sus aplicaciones existentes y a planificar nuevos tipos de aplicaciones capaces de explotar las características particulares de los dispositivos móviles del mercado. Al igual que en todos los grandes avances de las tecnologías de la información, los primeros años de este cambio han sido testigos de una frenética actividad destinada a satisfacer la demanda y generar presencia de mercado sin tener en cuenta aspectos más estratégicos, como el coste de desarrollo de las aplicaciones y su mantenimiento, calidad y seguridad. Una vez consolidado el mercado de las aplicaciones móviles y superada la fiebre inicial, estos aspectos del desarrollo de software reciben especial atención por parte de las personas de la empresa responsables de la planificación y la economía a largo plazo.

IBM goza de una sólida reputación como colaborador prudente y responsable en el desarrollo de software para empresas de numerosos sectores. Un reciente informe de IBM Global Services, titulado “Establishing an effective application strategy for your mobile enterprise”¹, resume en términos generales las recomendaciones de IBM para la planificación, el desarrollo, el despliegue y la gestión de las aplicaciones móviles. El presente

informe cubre uno de los temas del informe citado y versa sobre una estrategia completa para el desarrollo de aplicaciones móviles empresariales. La técnica descrita combina las mejores prácticas para la gestión en colaboración del ciclo de vida del software con los nuevos requisitos específicos que conlleva la creación de aplicaciones móviles. El contenido de este informe tiene como fin generar valor para todos los roles implicados en los proyectos de desarrollo de aplicaciones móviles empresariales: los arquitectos que planifican los proyectos móviles; los equipos de desarrollo que toman decisiones sobre la implementación; los directores de proyecto que establecen los detalles de las actividades del proyecto; las organizaciones de prueba que verifican las nuevas aplicaciones y los ejecutivos que necesitan comprender cómo estas nuevas aplicaciones móviles encajan con las aplicaciones y procesos de desarrollo existentes.

Retos particulares del desarrollo de aplicaciones móviles

La creación de aplicaciones destinadas a ejecutarse en dispositivos como smartphones y tablets conlleva requisitos y retos particulares.

Formatos y tecnología de entrada

El primer y más obvio aspecto de las aplicaciones móviles es que el formato de visualización y la interacción con el usuario son muy diferentes con respecto a tipos anteriores de software. Los smartphones normalmente ofrecen tan sólo una superficie de cuatro pulgadas en la que visualizar el contenido de la aplicación y tienen una densidad de píxeles y una resolución de pantalla menores que los de las pantallas de los ordenadores personales (PC), que tienden hacia tamaños más grandes y mayor número de píxeles en pantalla. Incluso los tablets tienen pantallas más pequeñas que los PC, especialmente en comparación con los grandes monitores de pantalla plana utilizados en los PC de sobremesa más modernos.

Lo reducido del formato implica que la cantidad de datos presentados al usuario y su diseño deben ser diferentes con respecto a las aplicaciones destinadas a un PC. Sólo es posible mostrar simultáneamente un volumen de datos muy inferior, por lo que deben ser los datos “precisos” más relevantes para las necesidades del usuario en esa parte de la aplicación.

Otra diferencia física obvia para las aplicaciones móviles es que los mecanismos de entrada de datos son diferentes. Los dispositivos móviles han introducido el uso de “gestos” como un método de entrada efectivo y popular que prescinde del teclado. Acciones como tocar, pasar y pellizcar deben ser planificadas y tenidas en cuenta para que la experiencia del usuario sea satisfactoria. Estos mecanismos de entrada táctil han resultado ser tan populares que ahora están siendo introducidos en sistemas operativos de PC, como la versión “Lion” de OS X, de Apple, y el SO Windows 8 “Metro”. Además de la entrada táctil, los dispositivos móviles son un medio natural para la entrada por voz. De hecho, la entrada de datos mediante teclado es probablemente el mecanismo menos efectivo y popular para los usuarios de aplicaciones móviles.

Además de la entrada directa por parte de los usuarios, los dispositivos móviles pueden recibir información de otras fuentes, como la localización geográfica indicada por su sistema GPS y los datos de la cámara, elementos normalmente integrados en el dispositivo. Estas formas de entrada propias deben ser tenidas en cuenta durante el diseño y desarrollo de aplicaciones móviles, ya que ofrecen nuevos y valiosos mecanismos para lograr que las aplicaciones sean más potentes y útiles que otras con posibilidades de entrada más limitadas.

Facilidad de uso y diseño de la interacción con el usuario

Diversos factores hacen necesario prestar especial atención a la facilidad de uso y el diseño de la interacción con el usuario en las aplicaciones móviles. Uno es la diferencia en el formato y los métodos de entrada de información. Es mucho más difícil y lento planificar cómo visualizar solamente los datos necesarios que simplemente presentar todos los datos posibles y dejar que el

usuario busque lo que necesita. Una analogía con la palabra escrita sería que es más difícil redactar un resumen conciso que todo un informe. El diseñador de la aplicación móvil debe pensar en el espacio disponible en pantalla. Cuando una aplicación necesita presentar una gran cantidad de datos, con múltiples capas de detalle, normalmente es mejor utilizar un método de descubrimiento progresivo que permita al usuario “profundizar” en niveles cada vez más detallados y centrados en menos elementos específicos.

La gran variedad de métodos de entrada disponibles en los dispositivos móviles también lleva a diseñar desde un principio cómo identificar y usar formas más eficientes de introducir información que limitarse a “rellenar un formulario”, como es la norma en las aplicaciones web y de PC tradicionales. Debe evitarse la introducción frecuente de datos mediante el teclado para reducir la frustración del usuario, especialmente debido a lo pequeño de los teclados táctiles y su falta de respuesta física. Encontrar formas distintas de reunir y suministrar información en una aplicación sin necesidad del teclado supone un reto considerable.

También existe otro motivo más sutil para dedicar especial atención al diseño de las aplicaciones móviles. El modo en el que los usuarios interactúan con los dispositivos móviles y las aplicaciones ejecutadas en éstos es diferente del modo en el que interactúan con PC de sobremesa e incluso portátiles. Los usuarios de un dispositivo móvil habitualmente sostienen el dispositivo en la mano mientras interactúan con la realidad de su entorno físico. Estos usuarios normalmente no pueden dedicar su atención a la aplicación móvil durante mucho tiempo sin tener que prestar atención a su entorno. El modelo de interacción para los usuarios de aplicaciones móviles es breve, intermitente, en “ráfagas”, lo que significa que deben completar con suma rapidez la tarea de la aplicación antes de desviar la atención.

Todos estos factores hacen precisa una mayor inversión en el diseño orientado al usuario de las aplicaciones móviles en las primeras etapas del proyecto de desarrollo. En condiciones ideales, estos aspectos de la facilidad de uso y el diseño deberían ser codificados en los requisitos de la aplicación móvil y luego vinculados a los elementos a entregar en las etapas posteriores, junto con las pruebas que validan que la interacción con el usuario y la “consumibilidad” de la aplicación son lo más satisfactorias posible.

Elección de la tecnología de implementación

Existe una amplia variedad de opciones para la implementación de aplicaciones móviles en el mercado. No existe una respuesta perfecta en cuanto a la mejor opción de implementación de una aplicación móvil y todas tienen sus ventajas y desventajas. Por lo tanto, el reto para el equipo de desarrollo móvil consiste en comprender los compromisos entre las tecnologías y tomar una decisión en función de los requisitos específicos de la aplicación. El informe de IBM Global Services anteriormente mencionado incluye una descripción concisa de las opciones de implementación, junto con una tabla comparativa. El presente informe incluye algunas consideraciones complementarias.

La elección de la tecnología de implementación de un proyecto móvil afecta a las demás decisiones relacionadas con el desarrollo de la aplicación. Puede limitar las opciones en cuanto a herramientas de desarrollo e influirá probablemente en los roles y la estructura del equipo, así como en el modo de probar y verificar la aplicación y en la forma en la que es distribuida y suministrada al usuario. Por lo tanto, la elección de un método de implementación para una aplicación móvil es una decisión inicial que debe tomarse con sumo cuidado.

Implementación de aplicaciones nativas

Una implementación “nativa” significa que la aplicación está siendo escrita utilizando el lenguaje y las interfaces de programación correspondientes al sistema operativo móvil de un tipo de dispositivo concreto. Por ejemplo, una implementación nativa para iPhone será escrita utilizando el lenguaje Objective-C y las interfaces de programación de aplicaciones (API) para iOS que Apple suministra y a las que da soporte.

La implementación de aplicaciones nativas tiene la ventaja de ofrecer la máxima fidelidad con ese dispositivo móvil. Debido a que las API empleadas son de bajo nivel y son específicas para el dispositivo al que está destinada la aplicación, ésta puede beneficiarse plenamente todas las características y servicios del dispositivo.

Las implementaciones nativas de las aplicaciones móviles no pueden ser transferidas a ningún otro sistema operativo móvil. Una aplicación iOS nativa debe ser reescrita por completo para que pueda ejecutarse en un dispositivo Android. Esto convierte a esta opción en un modo muy costoso de implementar una aplicación móvil empresarial.

Aplicaciones web

Los smartphones y tablets más recientes incluyen navegadores web avanzados y es perfectamente factible implementar una aplicación móvil empresarial consistente en una aplicación web estándar unida a hojas de estilo especiales para adaptarla al formato móvil y aproximarse a la apariencia y control del dispositivo móvil. Las aplicaciones móviles implementadas con este método son compatibles con una amplia variedad de dispositivos, ya que la compatibilidad de los navegadores web con JavaScript y HTML5 es bastante homogénea. Existen varias bibliotecas de widgets Web 2.0 comerciales y de código abierto que son de ayuda para este método. El modelo de programación web para la implementación de aplicaciones móviles tiene además la ventaja de que las empresas ya cuentan con desarrolladores formados en los lenguajes y las técnicas de desarrollo de aplicaciones web.

La desventaja de la implementación de aplicaciones web puras es que dichas aplicaciones no tienen acceso a las funciones y características que se ejecutan directamente en el dispositivo móvil, como la cámara, la lista de contactos, etc. Sin embargo, si su aplicación móvil no depende de servicios locales que se ejecuten en el dispositivo, el uso de una aplicación web pura puede ser suficiente. A medida que la especificación HTML5 se

consolide y sea más ampliamente admitida por los navegadores web móviles, muchos de los servicios locales de los dispositivos móviles estarán al alcance de las aplicaciones web puras mediante este estándar de programación W3C.

Otro aspecto que diferencia las aplicaciones web y las aplicaciones nativas es el modo en el que la aplicación es distribuida e instalada en el dispositivo. Las aplicaciones nativas deben ser descargadas e instaladas desde alguna clase de tienda de aplicaciones o “App Store”, como la tienda iTunes de Apple o el Android Market de Google. El mecanismo de distribución de la tienda de aplicaciones tiene la ventaja de facilitar la localización de la aplicación móvil mediante algoritmos de búsqueda. Las empresas agradecen en ocasiones la visibilidad y la opinión del usuario final que las tiendas de aplicaciones móviles proporcionan.

El lado negativo, especialmente en las tiendas públicas, es que se interponen directamente entre la empresa y sus destinatarios. Todas las actualizaciones de las aplicaciones móviles deben pasar por la tienda, por lo que puede ser difícil controlar y gestionar a distancia la aplicación móvil suministrada a través de los mecanismos de la tienda. Las aplicaciones web no se distribuyen a través de una tienda de aplicaciones: el usuario simplemente introduce la dirección web de la aplicación en el navegador móvil y la aplicación es suministrada por Internet. Actualizar la aplicación web móvil es tan sencillo como actualizar el servidor o los servidores que alojan la aplicación. La siguiente vez que el usuario acceda a la página web, la nueva versión de la aplicación se descarga en su dispositivo.

Implementación de aplicaciones móviles híbridas

La implementación de aplicaciones móviles híbridas es una suerte de compromiso entre la implementación nativa y la implementación web pura.

Las aplicaciones móviles se escriben utilizando lenguajes y técnicas de programación web estándar, como HTML5 y JavaScript. Sin embargo, la aplicación se empaqueta en un formato instalable nativamente y distribuido a través del mecanismo de la tienda de aplicaciones.

Las aplicaciones híbridas están vinculadas a bibliotecas nativas adicionales que permiten a la aplicación acceder a las funciones nativas del dispositivo desde la base de código simple. Debido a que el grueso de una aplicación híbrida es implementado con una tecnología que no es específica de ningún dispositivo concreto, la mayoría de su código puede ser reutilizado en muchos sistemas operativos móviles diferentes, aunque es posible integrar pequeños segmentos de código nativo en la aplicación híbrida. De este modo el desarrollador puede decidir qué parte de la implementación de la aplicación está compuesta por una base de código compartida común y cuál consiste en una personalización específica para el dispositivo.

También puede elegirse la proporción del código que debe ser empaquetada como una aplicación “nativa” inestable y suministrada desde la tienda de aplicaciones y cuál es la parte que debe ser descargada desde la red. Los primeros elementos de la aplicación que son visualizados pueden ser empaquetados para su instalación directa en el dispositivo, de forma que se carguen rápidamente cuando el usuario inicie la aplicación. Otros elementos más dinámicos pueden ser estructurados como páginas web gestionadas en un servidor y siempre proporciona la versión más reciente de la aplicación al acceder a ellos.

Muchos analistas tienen la firme convicción de que lo económico de la reutilización de código y la flexibilidad en el desarrollo de las aplicaciones favorecerán el método híbrido a largo plazo para las aplicaciones móviles empresariales normales.

Compilación y suministro de aplicaciones móviles

Debido a la fuerte motivación de las empresas para introducir aplicaciones móviles en el mercado rápidamente, los proyectos de desarrollo normalmente tienen plazos muy estrictos. Es común que el tiempo transcurrido desde la concepción a la entrega sea de unos pocos meses. Esta presión tiene como consecuencia la adopción de métodos de desarrollo ágil para la mayoría de los proyectos.

Un elemento importante de las prácticas de desarrollo ágiles es la integración y las compilaciones continuas. Los cambios en la aplicación efectuados por los desarrolladores deben ser procesados inmediatamente en todos los sistemas operativos móviles en los cuales debe ejecutarse. Si la aplicación móvil es una implementación híbrida o nativa, deben activarse varias compilaciones diferentes de la aplicación cada vez que un desarrollador efectúa un cambio en ella. La instalación y configuración de compilaciones para cada entorno móvil compatible será diferente del resto y es muy probable que sea necesario proporcionar un pequeño “parque” de servidores de compilaciones para administrar las compilaciones de la aplicación móvil destinadas a varios sistemas operativos.

Pruebas

Otra área en la que el desarrollo de aplicaciones móviles plantea un enorme reto son las pruebas. Las pruebas de aplicaciones móviles son mucho más complejas en comparación con las aplicaciones tradicionales. Al contrario que las aplicaciones web y de PC convencionales, la variedad de dispositivos móviles compatibles y de niveles de versiones es abrumadora. Es muy común ver matrices de pruebas para proyectos móviles que contienen cientos o incluso miles de permutaciones de combinaciones de dispositivos, nivel de SO móvil, operadores de red, lugares y orientación de los dispositivos.

Existen numerosas variables en la ecuación de las pruebas móviles que no son relevantes en otras clases de software. El mismo modelo de dispositivo puede funcionar de manera sutilmente distinta cuando se conecta a una red móvil diferente. Además, la calidad de la conexión de red puede tener un profundo impacto en el comportamiento de una aplicación móvil. Incluso el movimiento del dispositivo en sí puede ser un factor importante para el comportamiento de la aplicación, ya que algunas de ellas emplean específicamente dicho movimiento.

La mayoría de las aplicaciones móviles tienen una arquitectura multinivel en la que el código ejecuta en el dispositivo el cliente “frontal” de los datos y servicios suministrados por sistemas “internos” tradicionales de nivel medio y centros de datos. Unas

pruebas efectivas y completas de las aplicaciones móviles deben abarcar todos los niveles de la aplicación, además del código del dispositivo móvil. La configuración y disponibilidad de las versiones de prueba de los servicios de nivel medio e internos puede plantear retos enormes en cuanto al coste y la complejidad de las pruebas de aplicaciones móviles.

Muchos proyectos móviles empiezan utilizando métodos de prueba manuales. Este es el modo más obvio de comenzar las pruebas rápidamente. Sería preciso adquirir todos los dispositivos móviles a los que se prevé suministrar la aplicación y pagar a alguien, probablemente un grupo de personas, para que siga un guión por escrito con instrucciones que describen las pruebas a realizar en cada dispositivo y con cada compilación de la aplicación. Estas pruebas manuales son excesivamente costosas y poco eficaces. Aun así, las pruebas manuales juegan un papel importante al proporcionar un mecanismo para obtener opiniones vitales acerca de la facilidad de uso de la aplicación.

En lugar de adquirir los dispositivos móviles pueden emplearse simuladores y emuladores para las pruebas. Siguiendo este método, un programa de software ejecutado en un ordenador de sobremesa sustituye al dispositivo real. El uso de simuladores y emuladores para las pruebas de aplicaciones móviles puede ser valioso en tareas como las pruebas de la unidad de desarrollo. Algunos emuladores son excelentes, pero otros no reproducen el dispositivo con la misma eficacia. Por este motivo, ya sea en las pruebas manuales o automáticas, es esencial efectuar algún tipo de prueba en el dispositivo móvil real.

Existen soluciones de pruebas de aplicaciones móviles que emplean un programa agente en el dispositivo con el que un script de prueba puede interactuar en una ejecución automática. Este método tiene la misma flexibilidad que utilizar dispositivos físicos reales o emuladores en las pruebas, con la eficacia que aporta la automatización. Sin embargo, la organización de pruebas corre con el coste de configurar los dispositivos a probar e instalar el agente de pruebas en ellos.

Otra estrategia para afrontar el reto de estas pruebas es hacer uso de lo que podría denominarse como una “nube de dispositivos”. Una nube puede proporcionar recursos que en realidad son dispositivos móviles en lugar de ordenadores convencionales. En lugar de “alquilar” una máquina virtual Linux para realizar pruebas durante unos días o unas horas, puede alquilarse un modelo y versión específicos de un dispositivo móvil. Este método ahorra a la empresa el coste de adquirir quizá cientos de dispositivos y gestionarlos durante las pruebas.

¿En qué se asemeja el desarrollo móvil al desarrollo de otras clases de software?

Aunque existen aspectos propios del desarrollo de aplicaciones móviles, muchos de los roles y tareas que componen el ciclo de vida general del desarrollo son idénticos a sus equivalentes en el desarrollo empresarial de otras clases de software. En el informe “Measuring Agility and Architectural Integrity”², Walker Royce describe técnicas y prácticas clave para el suministro efectivo de software empleando principios ágiles y de pruebas previas. Las prácticas de suministro de software del informe de Walker son idóneas para los proyectos de desarrollo móvil.

Ciclo de vida completo del proyecto

El ciclo de vida de un proyecto de desarrollo de software generalmente presenta unas pautas similares con independencia del tipo de software creado. Comienza con la decisión empresarial, tomada a partir de un análisis, de invertir en la aplicación. Se recogen y detallan los requisitos de la aplicación. Dichos requisitos son desglosados a su vez en historias de usuario y elementos de trabajo para las funciones, todo ello reunido en un plan de trabajo con las iteraciones y las versiones que deben ser completadas para suministrar la aplicación. Los elementos de trabajo son asignados a los miembros del equipo que actúan en diversos roles y utilizan distintas herramientas para completarlos y entregar al proyecto el resultado. La aplicación resultante se prueba y se certifica que cumple los requisitos. El proceso exacto y el ciclo de vida seguidos por los proyectos de software en una empresa concreta está adaptado a las metas y políticas de esa empresa.

Esto es igual para las aplicaciones móviles. El desarrollo de aplicaciones móviles generalmente se caracteriza por emplear equipos pequeños, usar la infraestructura existente y crear aplicaciones altamente interactivas. Los métodos ágiles y los principios de pruebas previas son idóneos para este caso. Aunque los requisitos específicos del desarrollo de aplicaciones móviles son seguramente distintos de los de otro software, las herramientas y procesos para recopilar, detallar y comunicar los requisitos son los mismos. En las aplicaciones móviles es igualmente necesario vincular los requisitos con los cambios del código que los implementa. En otras palabras: el flujo del proyecto y la necesidad de integración y trazabilidad en su transcurso son idénticos para los proyectos de desarrollo de software móvil y de otras clases.

Integración de múltiples herramientas

Existen muy pocos proyectos, en el caso de que exista alguno, de desarrollo de software que puedan ser completados con una sola herramienta de desarrollo. La mayoría de los proyectos precisan de una amplia variedad de herramientas de distintos proveedores, diseñadas para satisfacer las necesidades de una tarea o un rol específicos dentro del ciclo de vida general del proyecto.

El desarrollador de código de una aplicación móvil, por ejemplo, puede encontrar que una sencilla herramienta de construcción de código, adaptada a la plataforma móvil en la que debe ejecutarse la aplicación, es suficiente para sus necesidades. Sin embargo, esa herramienta carece de funciones que faciliten la coordinación y la colaboración necesarias cuando un equipo de desarrollo ágil trabaja en la aplicación.

Al integrar la herramienta de construcción de código de ese desarrollador en una plataforma de desarrollo colaborativo, los equipos ágiles pueden mejorar su eficiencia y la calidad.

Necesidad de colaboración en el equipo

Las aplicaciones móviles normalmente son creadas por un equipo pequeño con diversas habilidades y conocimientos. Un equipo típico puede consistir en un par de desarrolladores de la lógica empresarial fundamental y los servicios web, un par de desarrolladores de la interfaz de usuario (IU), un diseñador de la experiencia del usuario, algunos testers y un líder o jefe de equipo.

En vista de los plazos existentes para la entrega de aplicaciones móviles, normalmente muy reducidos, incluso un equipo pequeño debe funcionar con la máxima eficiencia. Cualquier retraso causado por un malentendido o la falta de comunicación entre los miembros del equipo puede desbaratar todo el calendario de entregas.

Los proyectos de aplicaciones móviles destinados a varios sistemas operativos exigen compartir y reutilizar el código. Un desarrollador puede especializarse en Android y otro en iOS. Es esencial conocer claramente el trabajo que cada miembro del equipo debe realizar y cuándo deben entregarlo. Los requisitos, plazos, planes, etc. del proyecto son comunes en este caso, mientras que solamente el código fuente, las pruebas y las compilaciones son diferentes.

Gestión de cambios integrada y control de versión de software

Todos los cambios en el código asociados con un elemento de trabajo concreto deben ser reunidos en un “conjunto de cambios” o lista de archivos de código fuente modificados que se entrega de una sola vez para que toda la modificación del código pueda ser seguida de forma unitaria. En condiciones ideales, el proceso de reunir un conjunto de cambios debe ser lo más sencillo posible, de modo que no interrumpa la concentración de los desarrolladores en la lógica que están creando.

El proceso de control de versiones y fusión/rollback también debe ser automático e intuitivo. Todo momento en el que un desarrollador deba cambiar de contexto de trabajo para realizar alguna tarea representa un punto de interrupción y un posible “bache” en el proceso de desarrollo.

Necesidad de trazabilidad en todo el proyecto

El enfoque habitual del desarrollo de software en equipos ágiles consiste en definir múltiples iteraciones breves en las que se implementa y valida un pequeño conjunto de mejoras en la aplicación. Una iteración ágil normal tiene una duración de entre dos y cuatro semanas. El líder puede trabajar con el equipo para correlacionar los elementos de trabajo de una lista de elementos pendientes con cada iteración y asignar dichos elementos a cada desarrollador.

Cuando los desarrolladores eligen los elementos de trabajo y comienzan a avanzar en ellos, su esfuerzo debe quedar registrado automáticamente para que el líder del equipo pueda seguirlo y visualizarlo. Esto permite conocer y visualizar en un panel de control qué ha sido completado, en qué se está trabajando en ese momento y qué queda por hacer. Todos los miembros del equipo deben poder ver cómo progresa la iteración y el estado de los elementos de trabajo previstos para ésta.

Cuando los testers del equipo inicien las pruebas funcionales de la aplicación móvil, necesitarán abrir elementos de trabajo en el proyecto de desarrollo compartido para los defectos que descubran. Si el caso de prueba que ha fallado está vinculado con un conjunto de cambios o característica concreta del plan del proyecto, entonces la información sobre el código que ha sido modificado y que probablemente esté asociado con ese fallo puede ser introducida automáticamente en los datos del defecto. Si, además, el conjunto de cambios está vinculado con el requisito original que motivó la modificación del código, existe trazabilidad para todo el ciclo de vida del proyecto, desde los requisitos originales hasta el caso de prueba que verificó que el requisito se ha cumplido en la aplicación.

Esta clase de “panorámica” del proyecto y de trazabilidad de un extremo al otro es muy importante para cualquier proyecto de desarrollo de software, pero es especialmente relevante para los equipos de desarrollo de aplicaciones móviles que trabajan con plazos estrictos y emplean métodos ágiles. Este tipo de equipos

de desarrollo ligero no pueden perder tiempo buscando detalles acerca de si se ha verificado y cumplido un requisito en particular y en que momento se hizo.

Elección de una solución de desarrollo de aplicaciones móviles

La metodología de IBM para el desarrollo de aplicaciones móviles combina años de experiencia en procesos de desarrollo de software empresarial general con nuevas herramientas y técnicas específicas para los dispositivos móviles y sus bases de software subyacentes.

Gracias a su extensa experiencia en el diseño y despliegue de software empresarial en diversos sectores, IBM ofrece soluciones personalizadas para las necesidades de desarrollo de los proyectos de aplicaciones móviles. La misma IBM ha evolucionado para emplear métodos ágiles en miles de proyectos en los que han participado decenas de miles de desarrolladores.

Gestión colaborativa del ciclo de vida (CLM) para la plena visibilidad del proyecto

Sin integración durante todo el ciclo de suministro de aplicaciones móviles, los equipos de desarrollo trabajan aislados en silos. Cuando se forman silos, la efectividad del suministro de productos se resiente. Los equipos de diseño de software deben trabajar eficazmente y gestionar en colaboración todo los productos destinados a trabajar en el ciclo de vida con el fin de suministrar soluciones para aplicaciones móviles empresariales atractivas que respondan a los cambios en las necesidades y los estándares del mercado.

La solución IBM Rational para el desarrollo de aplicaciones móviles empresariales ofrece una plataforma de gestión integrada del ciclo de vida que admite tareas en colaboración y ayuda a vincular los diversos artefactos desarrollados durante el ciclo de vida del producto. Esta solución también establece flujos de trabajo para el suministro y dispone de capacidades de gestión de tareas con el fin de completar eficazmente el proyecto de desarrollo de la aplicación móvil. Se complementa con la integración de capacidades específicas para móviles en las fases del proyecto en las que tales capacidades son necesarias.

Las herramientas de desarrollo ágil en colaboración para equipos, como el software IBM® Rational® Team Concert™, permiten definir múltiples iteraciones breves dentro del proyecto general. Los elementos de trabajo pueden ser transferidos fácilmente de la lista de elementos pendientes del proyecto al plan de una iteración concreta.

Cuando los desarrolladores modifican los archivos dentro de las herramientas de desarrollo de código para aplicaciones móviles integradas en el sistema de control de versiones, se actualiza y mantiene automáticamente un conjunto de cambios. Los desarrolladores no tienen que hacer nada más para generar el conjunto de cambios aparte de modificar los archivos en los que deben trabajar.

Los conjuntos de cambios pueden ser compartidos entre los miembros del equipo antes de ser plenamente integrados en el flujo de código principal. De este modo, un conjunto de cambios creado por el desarrollador del servicio web que modifique el formato de los datos suministrados por el servicio puede ser compartido con el desarrollador de la IU que trabaja en la lógica que visualiza los nuevos datos sin que esto afecte al resto del equipo. Una vez que coincidan los cambios en el código de la IU y el código del servicio web y se consideren listos, pueden ser integrados en una tarea sincronizada en el flujo de código principal para que el resto del equipo pueda emplearla.

La estrategia IBM Mobile Enterprise suministra el tiempo de ejecución de la aplicación móvil

La combinación de una plataforma de gestión colaborativa del ciclo de vida integrada con herramientas de desarrollo de código específicamente destinadas a aplicaciones móviles es un componente fundamental de la completa solución de aplicaciones móviles de IBM. Sin embargo, algunos de los retos del desarrollo de aplicaciones móviles empresariales no pueden ser solucionados con herramientas y prácticas. Con el fin de ofrecer la capacidad necesaria para un solo modelo común de programación de aplicaciones móviles, ofrecemos tiempos de ejecución de software IBM Mobile Enterprise.

La solución de desarrollo de aplicaciones móviles empresariales de IBM combina las potentes capacidades de desarrollo en equipo encarnadas en IBM Rational Collaborative Lifecycle Management con las herramientas móviles y las capacidades de tiempo de ejecución de la solución IBM Mobile Enterprise.

Un método de pruebas completo

Las pruebas de aplicaciones móviles en múltiples niveles implican el uso y coordinación de más de una capacidad de ejecución de pruebas para garantizar la calidad de una aplicación. El software IBM Rational Quality Manager es una excelente elección para vincular y gestionar los diversos motores de ejecución de pruebas requeridos para las pruebas móviles. Nuestro entorno de pruebas permite poner en práctica una estrategia de pruebas de integración iniciales que puede ayudar a eliminar graves problemas al principio del ciclo de vida y mejorar la gestión económica. El informe de Walker Royce, “Measuring Agility and Architectural Integrity”, describe con detalle la técnica para verificar en primer lugar los problemas más complejos.

Como se menciona en un apartado anterior del presente informe, existen varias técnicas para probar y validar las aplicaciones móviles utilizadas actualmente. Los responsables de un proyecto de desarrollo efectivo deben emplear todas las técnicas aplicables contra su aplicación móvil porque cada una de ellas tiene sus virtudes y sus puntos débiles. No existe ninguna respuesta ideal a las pruebas de aplicaciones móviles y ninguna de estas técnicas excluye a las demás. La estrategia de pruebas más efectiva equilibra el uso de todas las formas de ejecución de pruebas móviles y recopila los resultados de cada una en una “medida de calidad” general de la aplicación.

Algunos ejemplos de técnicas de pruebas compatibles con la estrategia de IBM:

Pruebas manuales

Las pruebas manuales son el método más común utilizado actualmente en el sector. También son, sin embargo, las más lentas, proclives a errores y costosas. El uso de soluciones que organicen los casos de prueba, guíen al verificador durante la ejecución y almacenen los resultados pueden reducir su coste considerablemente. El software Rational Quality Manager ofrece estas capacidades.

Emuladores y simuladores

Los emuladores incluyen todos los kits de desarrollo del sistema operativo móvil nativo. Existen simuladores disponibles de fuentes como IBM, que ofrece simuladores móviles como parte de las herramientas de desarrollo de la solución IBM Mobile Enterprise.

Virtualización de protocolos, aislamiento del nivel de aplicación

Debido a que las aplicaciones móviles tienen una arquitectura multinivel, el proceso de configurar la infraestructura que sustenta la ejecución de pruebas del código en el dispositivo móvil puede ser lento y costoso. Es posible minimizar el coste y los retrasos en el despliegue empleando la solución IBM Rational. Los equipos de pruebas pueden ahorrarse la configuración de complejos entornos de middleware para ejecutar pruebas de código en dispositivos móviles. La solución Rational puede emular el nivel medio y los servicios y protocolos internos de forma que la ejecución de la prueba pueda centrarse en el nivel de cliente de la aplicación móvil iniciada en el dispositivo en sí.

Instrumentación y agentes en el dispositivo

En última instancia es preciso sustituir la verificación funcional manual por alguna forma de prueba automática del código que se ejecuta en el dispositivo móvil. Existen distintos métodos creados por diversos fabricantes. Un método típico consiste en incluir algún tipo de código adicional en el dispositivo en el que

deben realizarse las pruebas automáticas. Dicho código actúa como un agente local “en el dispositivo” que introduce entradas de usuario automáticas en la aplicación y monitoriza el comportamiento resultante de tales entradas. Esta técnica de automatización de las pruebas funcionales móviles es un buen complemento para las demás técnicas descritas en este documento y puede ser muy efectivo si se combina con éstas.

Nubes de dispositivos

El coste de adquirir, configurar y gestionar todas las combinaciones diferentes de dispositivos móviles para pruebas resulta prohibitivo hasta para los proyectos mejor financiados. Una técnica que puede solucionar este problema es emplear una “nube de dispositivos”. Este método reduce de modo efectivo el coste que supone adquirir la enorme variedad de tipos de dispositivos que existen y que posiblemente sean utilizados por los usuarios de la aplicación móvil una vez entre en producción. IBM ofrece integración entre la solución de gestión general de pruebas móviles, el software Rational Quality Manager, y diversos Business Partners que disponen de nubes de dispositivos.

Conclusión

Al aumentar el número de empresas de todos los sectores que necesitan una versión móvil de sus aplicaciones empresariales, surge la necesidad de un método empresarial para el desarrollo de aplicaciones móviles. IBM ha creado ese método.

El método de IBM para el desarrollo de aplicaciones móviles hace hincapié en cinco aspectos clave:

- Simplificar la experiencia del usuario de la aplicación móvil.
- Realizar la integración desde un principio para mejorar la gestión económica del proyecto de la aplicación móvil.
- Asegurar la trazabilidad de los requisitos para verificar su cumplimiento.
- Poner en práctica métodos extremadamente ágiles.
- Impulsar la evolución de suites de pruebas de regresión automáticas para acelerar el despliegue y reducir el coste de los cambios.

Todo esto permite utilizar herramientas y tecnología específicas para móviles con la misma eficacia y rigor que para el desarrollo de otras clases de aplicaciones empresariales.

Acerca del autor

Leigh Williamson es IBM Distinguished Engineer y trabaja en el laboratorio de Austin, Texas (EE.UU.), desde 1988, participando en importantes proyectos de software de IBM, como OS/2, DB2®, AIX®, OpenDoc, Java, Component Broker y WebSphere® Application Server. Actualmente es miembro del equipo del IBM Rational Software Chief Technology Officer, donde influye en la dirección estratégica de los productos de la marca Rational y lidera la definición de soluciones para el desarrollo de aplicaciones móviles. Leigh se licenció en Ciencias Informáticas por la Nova University y tiene un máster en Ingeniería Informática por la University of Texas.

Información adicional

Para obtener más información acerca de las soluciones IBM Rational para el desarrollo de aplicaciones móviles, póngase en contacto con su representante de marketing IBM o IBM Business Partner o visite:

ibm.com/software/solutions/mobile-enterprise/

Además, IBM Global Financing puede ayudarle a adquirir las soluciones de TI que su empresa necesita del modo más rentable y estratégico posible. Colaboramos con clientes que dispongan de crédito para personalizar una solución de financiación de TI acorde con sus objetivos empresariales, hacer posible una gestión de efectivo adecuada y mejorar el coste total de propiedad. IBM Global Financing es la solución más inteligente para financiar inversiones en TI vitales e impulsar su empresa hacia delante. Para obtener más información, visite: ibm.com/financing



© Copyright IBM Corporation 2012

IBM Corporation
Software Group
Route 100
Somers, NY 10589

Producido en los Estados Unidos de América
Abril 2012

IBM, el logo IBM, ibm.com, Rational y Rational Team Concert son marcas registradas de International Business Machines Corporation en numerosas jurisdicciones de todo el mundo. Otros nombres de productos o servicios pueden ser marcas registradas de IBM o de otras compañías. Existe una lista actualizada de marcas registradas de IBM disponible en la web, en el apartado "Copyright and trademark information" de ibm.com/legal/copytrade.shtml

Linux es una marca registrada de Linus Torvalds tanto en los Estados Unidos como en otros países.

Microsoft, Windows, Windows NT y el logo Windows son marcas registradas de Microsoft Corporation tanto en los Estados Unidos como en otros países.

Java y todas las marcas y logos basados en Java son marcas o marcas registradas de Oracle o sus filiales.

Este documento se considera actualizado en la fecha inicial de su publicación y puede ser modificado por IBM en cualquier momento. No todas las ofertas están disponibles en todos los países en los que opera IBM.

LA INFORMACIÓN PROPORCIONADA EN ESTE DOCUMENTO SE DISTRIBUYE "TAL CUAL", SIN GARANTÍA ALGUNA, YA SEA EXPRESA O IMPLÍCITA. IBM RECHAZA EXPRESAMENTE TODA GARANTÍA DE COMERCIALIZACIÓN, IDONEIDAD PARA UN FIN CONCRETO O CONFORMIDAD LEGAL. Los productos IBM están garantizados de acuerdo con los términos y condiciones de los contratos con arreglo a los cuales son facilitados.

¹ Informe IGS, "Establishing an effective application strategy for your mobile enterprise", ibm.com/common/ssi/cgi-bin/ssialias?subtype=WH&infotype=SA&appname=GTSE_EN_OS_USEN_C&htmlfid=ENW03007USEN&attachment=ENW03007USEN.PDF

² Walker Royce, informe "Measuring Agility and Architectural Integrity", www.ijsi.org/ijsi/ch/reader/view_abstract.aspx?file_no=i92



Recicle este documento.