

Monitorización y Sincronización de Recursos Compartidos en Aplicaciones Móviles Colaborativas Soportadas por MANETs

Gabriel Guerrero-Contreras, Kawtar Benghazi, José Garrido,
Carlos Rodríguez-Domínguez, Manuel Noguera

Resumen

Los sistemas colaborativos permiten que varias personas, cada una utilizando su propio dispositivo, trabajen en una tarea común compartiendo recursos. Esto conlleva habitualmente, mantener la consistencia entre las distintas réplicas de estos recursos. Además, gracias al gran avance en tecnologías móviles, actualmente existe un creciente interés para disponer y utilizar este tipo de aplicaciones en redes móviles. En este tipo de redes son frecuentes los problemas de desconexión y par-

ticiones en la red, lo cual dificulta la tarea de mantener la consistencia de los distintos recursos compartidos. En este trabajo se presenta una plataforma de servicios basada en el enfoque de arquitectura orientada a servicios. En dicha plataforma se ha desarrollado un servicio para gestionar la sincronización de las diversas réplicas existentes de los recursos compartidos en un entorno móvil, con el objetivo de facilitar el desarrollo de software colaborativo para este tipo de entornos.

Palabras clave

Sistemas Colaborativos, Sistemas Móviles, Arquitectura Orientada a Servicios, MANET.

1. Introducción

Gracias al desarrollo de las tecnologías de la comunicación e información (TICs), el concepto de trabajo cooperativo ha adquirido un nuevo significado y han surgido los términos de “Computer-Supported Cooperative Work” (CSCW) [1] y “Groupware” [2]. Mientras el concepto CSCW se refiere al campo que analiza la forma en la que las personas trabajan en grupos mediante el uso de las TICs, los sistemas Groupware se pueden definir como “sistemas de computación que dan soporte a grupos de personas que participan en una tarea común (u objetivo) y que proporcionan una interfaz a un entorno compartido.” [2]. Por tanto, una de las funciones principales del software colaborativo, es proveer de herramientas a sus usuarios para que puedan llevar a cabo las actividades de cooperación, coordinación y comunicación, necesarias para realizar el trabajo en común satisfactoriamente. Habitualmente, esto implica resolver la gestión de recursos compartidos (documentos y otra información) de forma consistente. Además, hoy en día existe un creciente interés para utilizar este tipo de software en redes móviles, caracterizadas por [3]: condicionamiento desigual, la tecnología disponible no es siempre la misma, depende del entorno; y escalabilidad localizada, un sistema bien diseñado, debe reducir de forma rigurosa la comunicación entre entidades distantes con el objetivo de no saturar la red de comunicación. Estas características están directamente relacionadas con las propiedades de continuidad y rendimiento en la conexión a redes (locales e Internet) y otros dispositivos, algo habitual cuando se utilizan redes “Mobile ad-hoc Networks” (MANETs) [4]. Una de las principales características de las redes MANETs consiste en que su infraestructura se configura de manera automática al detectarse dispositivos a su alrededor. Estas características complican la gestión de los recursos compartidos existentes en un entorno colaborativo.

Como solución al problema expuesto, actualmente existen distintas soluciones basadas en el enfoque peer-to-peer (p2p) [5, 6, 7], pero que resultan insuficientes para entornos colaborativos, donde los recursos compartidos son accedidos y modificados por varios usuarios de forma concurrente. En respuesta a este requisito, en este trabajo se presenta una plataforma de servicios basada en el enfoque de arquitectura orientada a servicios (“Service Oriented Architecture”, SOA) [8] en el cual se ha desarrollado un servicio para gestionar la sincronización de las diversas réplicas existentes de



los recursos compartidos de una aplicación colaborativa en un entorno móvil.

El resto del documento se organiza como sigue. La sección 2, identifica distintos trabajos para la gestión de recursos compartidos en redes móviles. En la sección 3, se describen los servicios propuestos para el soporte al desarrollo de aplicaciones móviles colaborativas. En la sección 4, se expone un caso de estudio para la edición de documentos compartidos. En la sección 5, se presenta una breve discusión acerca de la propuesta. Por último se recogen las principales conclusiones alcanzadas durante el desarrollo del trabajo.

2. Trabajos Relacionados

Las redes MANETs son útiles en ámbitos como el automovilístico (“Vehicular Ad-Hoc Network”) y en la gestión de desastres, donde la red de comunicaciones se ve afectada y no está disponible. Sin embargo, este tipo de redes presenta una serie de problemas, con respecto a la gestión de datos, que hay que resolver, tales como: localización de las fuentes de datos dentro de la red, protocolos de replicación y seguridad. En respuesta a esto, durante los últimos años han surgido una serie de propuestas enfocadas a tratar la gestión de datos en redes MANETs de forma eficiente y consistente, tales como DRIVE [5], MoGATU [6] y CHaMeLeoN [7]. Estos trabajos, tratan la compartición de datos a través del enfoque p2p. Pero, este enfoque no cubre completamente las necesidades de las aplicaciones colaborativas, donde varios usuarios pueden estar modificando el mismo recurso de forma concurrente, lo que hace que se necesiten otros enfoques.

En [9] se utiliza un enfoque SOA para afrontar la consistencia de datos en sistemas heterogéneos. El contexto en el que se presenta la solución, consta de diferentes aplicaciones donde cada una tiene un modelo local propio de la misma información. Lo que se pretende, es que cuando se haga una modificación en el modelo local de datos de una de las aplicaciones, esta modificación se propague al resto de modelos. La arquitectura propuesta se basa en un servicio de sincronización y un servicio de directorio. El servicio de directorio, mapea los distintos identificadores de los objetos de un modelo a otro y el servicio de sincronización, resuelve las posibles inconsistencias que puedan surgir entre las distintas modificaciones de la información. Esta solución presenta dos limitaciones: primero, no plantea la posibilidad de una desconexión por parte de alguna de las aplicaciones, ni permite las operacio-

nes sin conexión por lo que no es válida para un entorno móvil o ubicuo; y segundo, es una solución ad-hoc, por lo que resulta difícil adaptarla a otros dominios del problema.

Respecto a los sistemas colaborativos en entornos móviles, estos ofrecen beneficios tales como: adaptabilidad al usuario, facilidad de uso, interoperabilidad, recolección de información personalizada y movilidad. Gracias a estas características, los sistemas colaborativos móviles tienen una gran aceptación en campos tales como la salud o la educación. En parte, el éxito en estos campos se debe a que en estos entornos, tales como hospitales y colegios, se puede desplegar una infraestructura fija de soporte al sistema lo que facilita el desarrollo de sistemas colaborativos.

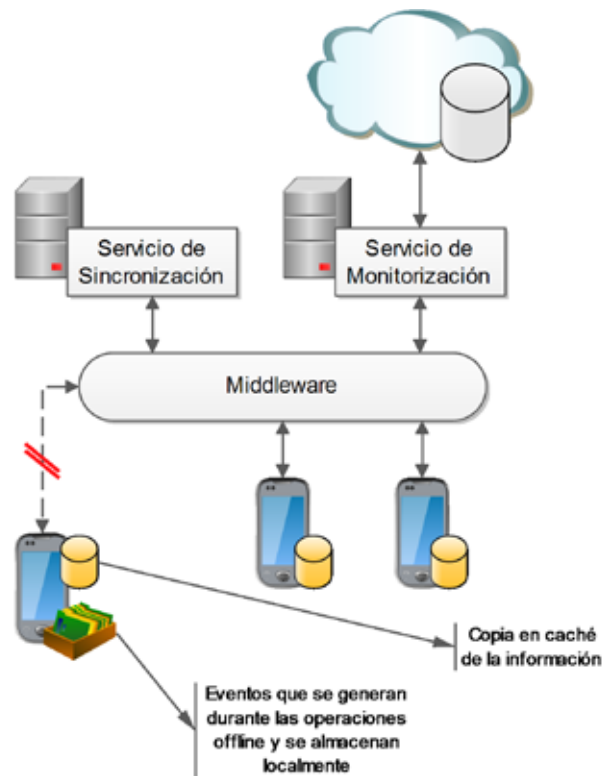
3. Servicios Para La Sincronización De Recursos Compartidos

En esta sección se presenta una propuesta basada en SOA para facilitar la gestión consistente de las diversas réplicas de un recurso compartido en una red MANET. Los servicios que se presentan se han diseñado con el objetivo de que pueden adaptarse a cualquier tipo de recurso compartido. En la Figura 1, se muestra un esquema de la plataforma. El principal componente es el Servicio de Sincronización, este servicio se encarga de mantener la consistencia de los recursos compartidos. Este servicio se basa en el servicio de monitorización, más básico. Este servicio se encarga de almacenar los cambios que se producen sobre los recursos compartidos en el sistema. Ambos servicios pueden estar replicados, para así mejorar su disponibilidad.

Además, con el objetivo de proveer una plataforma de desarrollo de aplicaciones móviles colaborativas completa, estos servicios se han desplegado sobre un middleware para entornos ubicuos. Se ha utilizado el middleware Bluerose [10, 11]. Este middleware se basa en una arquitectura dirigida por eventos, lo cual resulta muy interesante para trabajar en entornos móviles debido a que reduce el acoplamiento en la comunicación de un entorno heterogéneo.



Figura 1. Un dispositivo trabajando sin conexión dentro del sistema.



3.1 Servicio de Monitorización

El servicio de monitorización es un servicio básico, su función es almacenar los eventos que se generan a causa de modificaciones en los recursos compartidos. Esta información se puede utilizar con distintos objetivos, como por ejemplo llevar un control de versiones de un documento. En el caso del servicio de sincronización, esta información resulta fundamental para poder ejecutar los algoritmos de sincronización, ya que se necesita saber lo que ha ocurrido en el sistema mientras el dispositivo estaba desconectado. El servicio de monitorización se ha diseñado de forma que es capaz de monitorizar cualquier acción que pueda ocurrir en el sistema sin que sea necesario modificarlo. Esto es posible gracias a que las acciones en el sistema se representan como eventos y el servicio de monitorización es capaz de almacenar cualquier evento, sin conocer a priori su estructura.

3.2 Servicio de Sincronización

El servicio de sincronización es el principal componente del modelo. Este servicio se basa en el servicio de monitorización. De esta forma, se pueden afrontar los problemas de las redes móviles (desconexiones y particiones de la red) y además permitir que los dispositivos desconectados puedan seguir trabajando. Para poder realizar operaciones sin conexión, la aplicación cliente, como se muestra en la Figura 1, tiene que ir almacenando en caché la información con

la que está trabajando el usuario, para que cuando se produzca una desconexión el usuario pueda continuar trabajando. Por otra parte, también tiene que almacenar de forma local los eventos que genera mientras no dispone de conexión, para que al recuperar ésta pueda enviar la información al servicio de sincronización.

El servicio de sincronización no puede ser independiente del dominio del problema, ya que los algoritmos de sincronización son muy dependientes del tipo de recurso compartido. Por ello, se ha diseñado el servicio de forma genérica, permitiendo que pueda extenderse, mediante herencia, con los algoritmos de sincronización concretos de un recurso en particular.

4. Caso De Estudio: Edición Colaborativa De Documentos

Para probar y mostrar la validez de la plataforma que se ha presentado en este trabajo, se ha especializado el servicio de sincronización para el caso en el que los recursos compartidos son documentos de texto, es decir, se ha especializado el servicio de sincronización en un servicio de repositorio de documentos. El sistema exhibe el comportamiento que se describe a continuación (véase Figura 2).

El servicio de repositorio mantiene el conjunto completo de los documentos

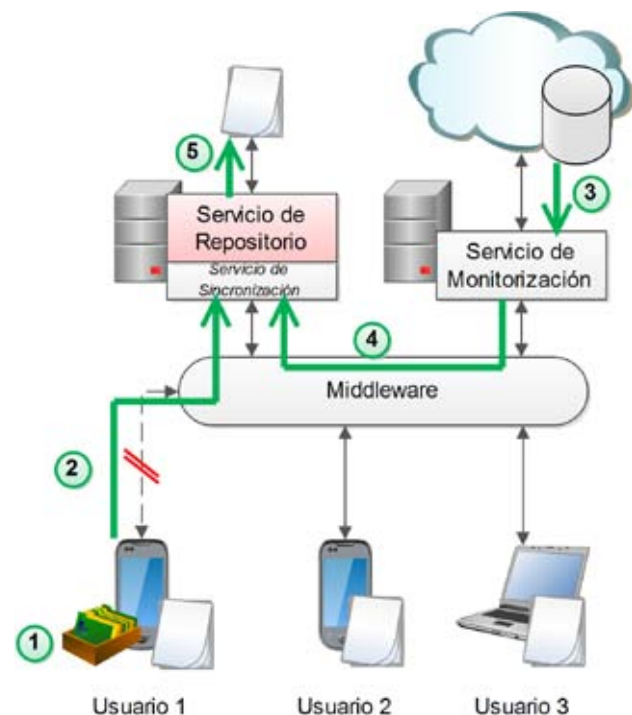


Figura 2. Comportamiento del servicio de repositorio cuando se produce una re-conexión.



que existen en el sistema y se encarga de integrar en ellos, de forma consistente, los distintos cambios que realizan los usuarios. El servicio de monitorización, mientras tanto, almacena toda la información acerca de las modificaciones que se realizan en los documentos. La aplicación cliente se encarga de mantener en caché una copia reciente de los documentos con los que el usuario está trabajando. Cuando se produce la desconexión de un dispositivo (Usuario 1 en la Figura 2), la aplicación cliente comienza a trabajar con la copia en caché de los documentos y a almacenar localmente los eventos que el usuario genera mediante las modificaciones en estos documentos (Figura 2, punto 1) para posteriormente volver a sincronizarse con el resto del sistema cuando se recupere la conexión (sección 3.2). Al recuperar la conexión, la aplicación solicita al servicio de repositorio sincronizarse con el resto del sistema (Figura 2, punto 2). Para ello le proporciona el periodo de tiempo en el cual ha estado desconectada y los eventos que el usuario ha generado durante la desconexión, es decir, le informa sobre los cambios que el usuario ha realizado en los recursos compartidos durante ese periodo de tiempo. Cuando el servicio de repositorio recibe esta petición, solicita al servicio de monitorización las modificaciones sobre los documentos que se han realizado en el sistema durante el periodo de desconexión del dispositivo. El servicio de monitorización consulta esta información (Figura 2, punto 3) y la envía al servicio de repositorio (Figura 4, punto 4). Una vez tiene por una parte el conjunto de eventos que ha generado el usuario desconectado y por otra el conjunto de eventos que se ha generado en el sistema, el servicio de repositorio contrasta esos dos conjuntos de acciones en busca de inconsistencias y aplica los cambios en los documentos (Figura 2, punto 5), pudiéndose dar los tres casos siguientes:

- El documento que ha modificado el usuario desconectado no ha sido modificado por otro usuario en el sistema. En este caso, el servicio de repositorio aplica los cambios directamente en el documento.
- El documento que ha modificado el usuario desconectado ha sido modificado por otro u otros usuarios pero no son modificaciones conflictivas. En este caso el servicio de repositorio integra automáticamente los cambios de los distintos usuarios en el mismo documento. En nuestro caso, hemos considerado que una modificación conflictiva se da cuando dos o más usuarios han modificado la misma posición (byte) del fichero. También caben otras opciones, como considerar modificaciones conflictivas a nivel de palabra o frase.



- El documento que ha modificado el usuario desconectado ha sido modificado por otro u otros usuarios y son modificaciones conflictivas. En este caso el servicio de repositorio crea una versión paralela del documento con los cambios del usuario que estaba desconectado, por lo tanto, se mantienen dos o más versiones del documento, dejando que sean los propios usuarios los que resuelvan el conflicto.

Respecto al servicio de repositorio diseñado hay que destacar que los algoritmos de sincronización implementados son independientes de la aplicación, por lo que este servicio puede ser reutilizado por cualquier aplicación colaborativa que maneje documentos de texto.

5. Discusión

Con el objetivo de avanzar en una mayor aceptación en cuanto al uso de aplicaciones colaborativas en un entorno de computación móvil, la propuesta que se ha presentado, la cual resulta de la combinación de un conjunto de técnicas de computación distribuida y arquitecturas orientadas a servicios, persigue abordar los siguientes beneficios: interoperabilidad, adaptabilidad, reusabilidad, escalabilidad, bajo acoplamiento con la plataforma y facilitar las operaciones sin conexión. Por otra parte, pueden darse ciertos problemas como saturación en la red de comunicaciones para el caso de la edición colaborativa de documentos (sección 4) dado que se genera un evento por cada nuevo carácter que se introduce o se elimina en un documento. Como solución a esto, se puede estudiar trabajar a nivel de palabra, con lo que se reduciría el número de eventos generados. Además, utilizar un enfoque SOA conlleva una serie de problemáticas que hay que resolver. Entre otras cuestiones, hay que tener en cuenta que una vez que el servicio esté integrado y en funcionamiento, antes de realizar cualquier cambio, debe realizarse un estudio acerca del impacto que puede producir el cambio en el sistema, lo cual puede limitar la evolución del mismo; además, conforme el sistema crece, puede resultar complicado cumplir con todos los estándares en los que se basa SOA.



6. Conclusiones Y Trabajo Futuro

En este trabajo se ha presentado una plataforma de servicios para el soporte al desarrollo de aplicaciones móviles colaborativas. Esta plataforma se basa en un servicio principal de sincronización y otro de monitorización más básico. Estos servicios se han desarrollado de forma general para que puedan ser aplicados a cualquier dominio, a diferencia de las soluciones ad-hoc. Para ofrecer una plataforma de desarrollo completa, los servicios se han desplegado sobre un middleware para entornos ubicuos y redes MANET. La plataforma expuesta, está orientada especialmente a soportar las operaciones sin conexión. Por otra parte, para conseguir escalabilidad localizada y solucionar problemas propios de los entornos ubicuos, se han utilizado protocolos de replicación en los servicios, mejorando además la disponibilidad de estos. Por último, el uso de la plataforma propuesta proporciona beneficios tales como reutilización, escalabilidad, disponibilidad, interoperabilidad, bajo acoplamiento con la plataforma y además facilita las operaciones sin conexión.

Respecto a las líneas de trabajo futuro, se plantea realizar un estudio más profundo de los requisitos generales de sincronización que pueden tener distintos tipos de recursos compartidos y ofrecer estos algoritmos en el servicio de sincronización. También, se plantea aplicar esta plataforma de servicios en ámbitos como la gestión de información del contexto y herramientas para realizar anotaciones semánticas en la web. Además se pretende integrar los servicios presentados en este trabajo en una plataforma mayor junto a otros servicios, como el servicio de localización que se presenta en el siguiente trabajo [12].

7. Agradecimientos

Este trabajo está financiado por el Ministerio de Economía y Competitividad del Gobierno de España bajo el Proyecto de Investigación con Referencia TIN2012-38600 y por la beca de Iniciación a la Investigación del plan propio de la Universidad de Granada.

Referencias

- [1] **I. Greif**, "Computer-supported cooperative work: a book of readings," 1988.
- [2] **C. A. Ellis, S. J. Gibbs, and G. Rein**, "Groupware: some issues and experiences," *Communications of the ACM*, vol. 34, no. 1, pp. 39--58, 1991.
- [3] **M. Satyanarayanan**, "Pervasive computing: Vision and challenges," *Personal Communications, IEEE*, vol. 8, no. 4, pp. 10--17, 2001.
- [4] **M. Bansal, R. Rajput, and G. Gupta**, "Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations," 1999.
- [5] **B. Xu and O. Wolfson**, "Data management in mobile peer-to-peer networks," in *Databases, Information Systems, and Peer-to-Peer Computing*, pp. 1--15, Springer, 2005.
- [6] **F. Perich, A. Joshi, and R. Chirkova**, "Data management for mobile ad-hoc networks," in *Enabling Technologies for Wireless E-Business*, pp. 132--176, Springer, 2006.
- [7] **S. Ghandeharizadeh, A. Helmy, B. Krishnamachari**, F. Bar, and T. Richmond, "Data management techniques for continuous media in ad-hoc networks of wireless devices," in *Encyclopedia of Multimedia*, 2006.
- [8] **C. M. MacKenzie, K. Laskey, F. McCabe, P. F. Brown, R. Metz, and B. A. Hamilton**, "Reference model for service oriented architecture 1.0," *OASIS Standard*, vol. 12, 2006.
- [9] **E. Svensson, C. Vetter, and T. Werner**, "Data consistency in a heterogeneous it landscape: a service oriented architecture approach," in *Enterprise Distributed Object Computing Conference, 2004. EDOC 2004. Proceedings. Eighth IEEE International*, pp. 3--8, IEEE, 2004.
- [10] **C. Rodríguez-Domínguez, K. Benghazi, J. L. Garrido, and A. Valenzuela**, "A platform supporting the development of applications in ubiquitous systems: the collaborative application example of mobile forensics," in *Proceedings of the 13th International Conference on Interacción Persona-Ordenador*, p. 41, ACM, 2012.
- [11] **C. Rodríguez-Domínguez, K. Benghazi, M. Noguera, J. L. Garrido, M. L. Rodríguez, and T. Ruiz-López**, "A communication model to integrate the request-response and the publish-subscribe paradigms into ubiquitous systems," *Sensors*, vol. 12, no. 6, pp. 7648--7668, 2012.
- [12] **T. Ruiz-López, C. Rodríguez-Domínguez, M. Noguera, and J. L. Garrido**, "Towards a reusable design of a positioning system for aal environments," in *Evaluating AAL Systems Through Competitive Benchmarking. Indoor Localization and Tracking*, pp. 65--79, Springer, 2012.



Sobre los autores

Gabriel Guerrero Contreras es Ingeniero en Informática desde el año 2012 por la Universidad de Granada, España, con Máster en Desarrollo de Software desde 2013. Actualmente es investigador del Grupo Modelling & Development of Advanced Software Systems (MYDASS) del departamento de Lenguajes y Sistemas Informáticos de la Universidad de Granada. Su investigación se centra en sistemas ubicuos y aplicaciones sensibles al contexto.

gjuerrero@ugr.es

Kawtar Benghazi es actualmente profesora contratada doctora de la Universidad de Granada y doctora por la misma universidad desde el año 2009. Sus temas de interés se centran en la especificación formal y diseño de soluciones software aplicadas a dominios o ámbitos específicos, particularmente e-salud, calidad de vida y computación ubicua. Asimismo, investiga y trabaja en soluciones software basadas en arquitecturas orientadas a servicios.

benghazi@ugr.es

José Luis Garrido es actualmente Profesor del Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Granada. Obtuvo su doctorado en Informática en 2003. Es miembro de comités, coeditor y revisor en varias conferencias, talleres, revistas y libros científicos. Sus publicaciones y líneas de investigación se enmarcan dentro de la ingeniería de requisitos, arquitecturas de sistemas/software, computación ubicua y sistemas sensibles al contexto, y sistemas cooperativos y distribuidos. También es socio cofundador de la Empresa EVERYWARE TECHNOLOGIES.

jgarrido@ugr.es

Carlos Rodríguez Domínguez es Ingeniero en Informática desde 2009, con Máster en Desarrollo de Software desde 2010. Ha cursado sus estudios en la Universidad de Granada, España. Actualmente es investigador del departamento de Lenguajes y Sistemas Informáticos de la Universidad de Granada, donde también está realizando su doctorado. Forma parte del Grupo Modelling & Development of Advanced Software Systems (MYDASS), y es miembro de comités y revisor de varias conferencias, talleres y revistas científicas. Ha participado como desarrollador y como investigador en diversos proyectos de I+D+I, publicando en revistas y congresos nacionales e internacionales. Su investigación gira en torno a la comunicación y coordinación en sistemas ubicuos y aplicaciones sensibles al contexto. También es socio cofundador y director técnico de Everyware Technologies.

carlosrodriguez@ugr.es

Manuel Noguera es profesor Contratado Doctor en la Universidad de Granada, (España) y se doctoró en dicha universidad en 2009. Es investigador principal y responsable de desarrollo de CloudRehab, una plataforma móvil de telerehabilitación que utiliza tecnología cloud para personas con daño cerebral adquirido. Sus temas de trabajo se orientan hacia métodos de especificación, diseño e implementación de arquitecturas de sistemas y software, groupware e ingeniería dirigida por ontologías. Actualmente, participa en varios proyectos de investigación y transferencia tecnológica en el ámbito de mhealth, deporte y bienestar, y es miembro del Comité de Gestión de la Acción COST europea AAPELE (Algorithms, Architectures and Platforms for Enhanced Living Environments).

mnoguera@ugr.es



© Leonardo Infante

Monitorización y Sincronización de Recursos Compartidos en Aplicaciones Móviles Colaborativas Soportadas por MANETs

Los sistemas colaborativos permiten que varias personas, cada una utilizando su propio dispositivo, trabajen en una tarea común compartiendo recursos. Esto conlleva habitualmente, mantener la consistencia entre las distintas réplicas de estos recursos. Además, gracias al gran avance en tecnologías móviles, actualmente existe un creciente interés para disponer y utilizar este tipo de aplicaciones en redes móviles. En este tipo de redes son frecuentes los problemas de desconexión y particiones en la red, lo cual dificulta la tarea de mantener la consistencia de los distintos recursos compartidos. En este trabajo se presenta una plataforma de servicios basada en el enfoque de arquitectura orientada a servicios. En dicha plataforma se ha desarrollado un servicio para gestionar la sincronización de las diversas réplicas existentes de los recursos compartidos en un entorno móvil, con el objetivo de facilitar el desarrollo de software colaborativo para este tipo de entornos.