

# Diseño de un componente de comunicación para app móviles

Adriana Elizabeth Martin<sup>1</sup>, Susana Beatriz Chavez<sup>2</sup>, Sergio Rafael Flores<sup>3</sup>, A. Sara Zogbe<sup>4</sup>,

Departamento e Instituto de Informática - F.C.E.F. y N. - U.N.S.J.  
Av. Ignacio de la Roza 590 (O), Complejo Universitario "Islas Malvinas",  
Rivadavia, San Juan, CPA: J5402. San Juan, Tel 0264 4234129

<sup>1</sup>arianamartinsj@gmail.com; <sup>2</sup>schavez@iinfo.unsj.edu.ar; <sup>3</sup>sergiorflores@gmail.com;  
<sup>4</sup>sarazogbe@yahoo.com.ar

## Abstrac

El avance de las comunicaciones ha cambiado drásticamente la forma en que las personas y las máquinas interactúan entre sí, permitiendo el acceso instantáneo a información y servicios en tiempo real.

Se propuso un modelo de comunicación entre app móviles, que permitiera evaluar la disponibilidad de canales de comunicación, de manera de garantizar que un mensaje llegue a destino. Para lo cual se decidió trabajar sobre una plataforma con soporte a la programación reactiva. Esto obligó a analizar y entender qué propone el paradigma reactivo para que el desarrollo del software móvil sea una solución competitiva, haciendo uso de Apps reactivas.

**Palabras claves:** Paradigma Reactivo, Dispositivos móviles, IoT, Java, Android SO.

## 1. Introducción

Debido a que los procesadores multinúcleo se han convertido en un estándar, se han creado múltiples niveles de abstracción para simplificar la concurrencia y permitir un desarrollo más simple.

Para interactuar continuamente con su entorno, las apps reactivas deben adaptarse a la carga a la que se enfrentan, utilizando una mayor capacidad computacional cuando es necesario. Esto significa que debe hacer uso eficiente del hardware en un solo dispositivo (de uno o más núcleos), y pueda funcionar a través de varios nodos de cómputo a su disposición, dependiendo de la carga.

Se está desarrollando un componente de comunicación que se encarga de identificar todas las alternativas posibles de comunicación, que se encuentran disponibles en el hardware del dispositivo, con el objeto de garantizar la conectividad necesaria.

En un modelo de concurrencia, ya no hay memoria real compartida, los diferentes núcleos de un dispositivo se envían entre sí fragmentos de datos explícitamente, tal como lo hacen las computadoras conectadas en una red. Esto ocurre en lugar de ocultar el aspecto del paso del mensaje a través de variables marcadas como compartidas.

Un enfoque más disciplinado, consiste en mantener el estado local de una entidad concurrente y propagar datos o eventos a través de mensajes entre ellos.

Los protocolos de IoT, tanto para el sector Doméstico como para el Industrial, están resueltos siempre y cuando haya conexión real de internet (datos o wifi). Cuando estas conexiones no están disponibles, no es posible mantener la comunicación, solo queda esperar a que se restablezca la misma, con las consecuencias que esto pudiera acarrear. Por este motivo es que se propone el diseño de un componente (driver) que se encargue de guiar en forma automática la comunicación de acuerdo a un esquema de jerarquía y en forma transparente al usuario.

En la Figura 1 se muestra cómo funciona el sistema de comunicación en forma normal.

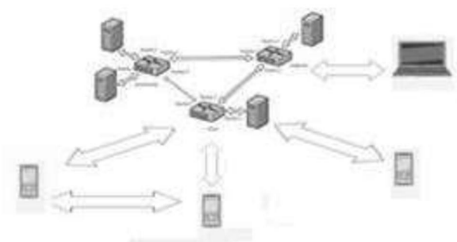


Figura 1. Comunicación normal

Cuando la comunicación se interrumpe debido a diversos factores, como por ejemplo fuera de cobertura, falta de crédito, o por cualquier otro motivo, existen otras formas de comunicación que siguen vigentes sin que hayan sido diseñadas para este propósito, SMS o llamadas de voz. Incluso, algunas compañías proveen de servicio gratuito al sistema de comunicación WhatsApp, que, si bien están destinados a un fin específico, se podrían utilizar para proveer de conectividad de manera temporal al móvil. De esa manera, un móvil puede hacer de anfitrión para proveer la comunicación, por ejemplo: como se muestra en la Figura 2 el móvil M1 tiene comunicación con un sistema que a su vez se comunica con el móvil M3. Si se rompe la comunicación de M1, el driver busca un anfitrión como por ejemplo el móvil M2 y accede a través de él al sistema

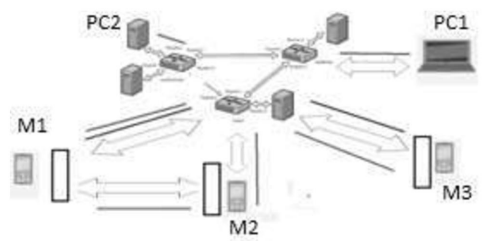


Figura 2. Comunicación a través de un anfitrión

Se está implementando un driver que se comunica directamente con el sistema operativo, tratando de discernir el mejor camino cuando se pierde la conexión natural de datos o wifi. La figura 3 muestra un esquema general de las funciones del driver.

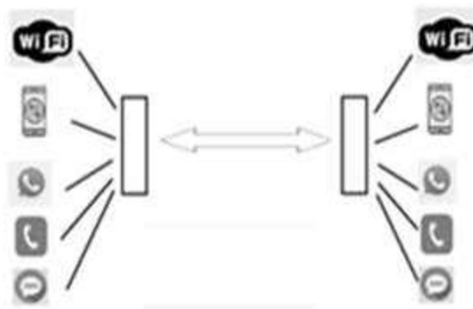


Figura 3. Funciones del Driver propuesto

## 2. MOTIVACIÓN

El objetivo de este trabajo es evaluar todas alternativas disponibles para manipular la comunicación entre aplicaciones que se originan en los distintos dispositivos, y construir un prototipo basado en una arquitectura reactiva donde la resiliencia y la comunicación asíncrona permitan que los sistemas estén exentos de errores, de modo que la comunicación siempre sea posible.

## 3. CONTEXTO

El presente trabajo se enmarca dentro del proyecto de investigación: Modelo de Sistema de Comunicación en Programación Reactiva, que ha sido aprobado por CICITCA , UNSJ. y está en desarrollo para el período 2020-2022.

Las unidades ejecutoras para dicho proyecto son el Departamento e Instituto de Informática de la FCEFYN de la UNSJ.

## 4. APORTES DE TRABAJO

Con esta propuesta de trabajo se espera contribuir a la profundización y consolidación del conocimiento en esta área temática por parte de cada uno de los integrantes de este proyecto

Desde el 2020 se realizaron diferentes publicaciones afines a la propuesta.

El prototipo propuesto se encuentra en un estado avanzado de desarrollo e implementación.

Se establecieron comunicaciones punto a punto entre de los distintos sistemas de comunicación sin hacer eco con el dispositivo, es decir, generando mensajes internos, enviándolos y recibiendo los en forma transparente al usuario. Evitando de esa forma la intervención del usuario en la comunicación. El entorno de desarrollo utilizado es Android Studio, con el lenguaje de programación nativo java.

Se está avanzando en la comunicación de las apps a través del canal de voz, por medio de una llamada, para lo cual se lograron las siguientes acciones:

- a) Establecer la llamada con el móvil destino, es decir, se realiza la llamada y el móvil destino responde
- b) Sincronizar la comunicación con el móvil destino, es decir, que sepa que no es una llamada de voz común, sino que se trata de una comunicación entre máquinas. Esto se puede realizar por

medio de una escucha de las notificaciones de llamadas, si es el número programado, se captura la llamada, y se envía un ok para sincronizar (en otra frecuencia).

- c) El texto a enviar se debe modular con una frecuencia de 42KHz, luego tomar los datos, pasarlo a voz y modularlos.
- d) Una vez sincronizado enviar la señal modulada.
- e) En el dispositivo destino tomar la señal modulada con un formato de voz y aplicar el proceso de demodulación
- f) El dato estaría disponible

La variedad de dispositivos con capacidades diferentes es enorme y está al alcance de la mano.

Resta configurar, adaptar, y poner a punto el prototipo propuesto: SiCo (Sistema de Comunicación).

De esta manera se contribuirá a la profundización y consolidación del conocimiento de esta área temática.

En cuanto a la movilidad de los dispositivos, los recursos pueden variar sin depender de los mismos, sino de los mecanismos de comunicación desde 4G pasando por GSM, solo WhatsApp, solo mensajería de texto o simplemente llamada de voz, es por ello que se necesita disponer de un mecanismo de selección automática del mejor sistema de comunicación disponible.

## **5. FORMACIÓN DE RECURSOS HUMANOS**

El equipo de trabajo está compuesto por 4 docentes-investigadores de la línea de investigación presentada que figuran en este trabajo y Alumnos avanzados de las Carreras de Licenciatura en Sistemas de Información, Y Licenciatura en Ciencias de la Computación en estado de tesis; pertenecientes a la Universidad Nacional de San Juan.

Se está trabajando con dos tesinas en el ámbito de la programación para dispositivos móviles. Se espera iniciar otra tesina de grado en el área motivo de la presente propuesta de investigación.

## **6. INVESTIGACIONES FUTURAS**

Por el momento la aplicación se está desarrollando bajo Android, pero se prevé a futuro implementar este componente en IOS.

## **7. BIBLIOGRAFÍA CONSULTADA**

- Adam L. Davis: Reactive Streams in Java\_ Concurrency with RxJava, Reactor, and Akka Streams-Apress (2018)
- How the Actor Model Meets the Needs of Modern, Distributed Systems: <https://doc.akka.io/docs/akka/current/typed/guide/actors-intro.html>
- José I. Rodríguez M. Tesis Doctoral “Metamodelo para la integración del Internet de las cosas y Redes sociales” Universidad Oviedo 2017
- González García Cristian, “MIDGAR: Plataforma para la generación dinámica de aplicaciones distribuidas basadas en la integración de redes de sensores y dispositivos electrónicos IoT,” UNIVERSIDAD DE VIEDO, 2013.

- L. Atzori, A. Iera, G. Morabito, and M. Nitti, “The Social Internet of Things (SIoT) – When social networks meet the Internet of Things: Concept, architecture and network characterization,” *Comput. Networks*, vol. 56, no. 16, pp. 3594–3608, Nov. 2012.
- C. González García, B. C. Pelayo G-Bustelo, J. Pascual Espada, and G. Cueva-Fernandez, “Midgar: Generation of heterogeneous objects interconnecting applications. A Domain Specific Language proposal for Internet of Things scenarios,” *Comput. Networks*, vol. 64, pp. 143– 158, May 2014.
- R. Roman, J. Zhou, and J. Lopez, “On the features and challenges of security and privacy in distributed internet of things,” *Comput. Networks*, vol. 57, no. 10, pp. 2266–2279, Jul. 2013.
- J. Pascual Espada, O. Sanjuán Martínez, B. C. Pelayo GBustelo, and J. M. Cueva Lovelle, “Virtual Objects on the Internet of Things,” *Int. J. Interact. Multimed. Artif. Intell.*, vol. 1, no. 4, p. 23, 2011.
- B. Xu, L. Da Xu, H. Cai, C. Xie, J. Hu, and F. Bu, “Ubiquitous Data Accessing Method in IoT-based Information System for Emergency Medical Services,” *IEEE Trans. Ind. Informatics*, vol. 3203, no. c, pp. 1–1, 2014. 100
- La Ciberseguridad en la Industria 4.0 <https://www.incibe-cert.es/blog/ciberseguridad-industria-4-0>