

# Adquisición Automatizada de Hechos para Sistemas Basados en Conocimiento Empleando Dispositivos Móviles

Matias Pirovano Varela, Gustavo Pereira, Iris Sattolo, Jorge Salvador Ierache

Instituto de Sistemas Inteligentes y Enseñanza Experimental de la Robótica (ISIER)  
Facultad de Informática, Ciencias de la Comunicación y Técnicas Especiales  
Universidad de Morón, Cabildo 134 (B1708JPD), Morón, Buenos Aires, Argentina  
jierache@unimoron.edu.ar

**Abstract.** Los sistemas basados en conocimiento existen desde hace varias décadas, y han sido utilizados durante ese periodo extensivamente en el desarrollo de sistemas expertos, pero recién con la aparición de los dispositivos móviles modernos se ha conseguido una renovada apreciación de ellos. Los dispositivos móviles permiten tener un experto en el tema que se necesite, cuando y donde se lo requiera. Este trabajo propone una arquitectura para la integración de sistemas basados en conocimiento integrados en dispositivos móviles que permita la adquisición de los datos que representan los hechos desde la nube minimizando la carga de datos por parte del usuario.

**Keywords:** Sistema Basado en Reglas, Dispositivo Móvil, Android, Sistemas Inteligentes.

## 1 Introducción

Este trabajo propone una arquitectura para una plataforma móvil que permita la ejecución de un Sistema Basado en Conocimiento (SBC) con la mínima intervención del usuario. Si bien actualmente existen SBC que corren en forma nativa sobre dispositivos móviles, estos requieren generalmente que el usuario ingrese datos para continuar con la ejecución del sistema. Estos no cuentan con una ejecución automatizada, con la mínima intervención del usuario en el proceso de carga de datos, es decir, que el sistema se encargue de buscar los datos sin solicitárselos al usuario, tanto en la nube u otros dispositivos asociados, como sensores o artefactos en el contexto del Internet de las cosas. En este orden se desarrolló un marco de trabajo que funcione sobre plataformas móviles, con el cual se pueda configurar la ejecución de un SBC, en lugar de requerir que un usuario ingrese la mayoría o la totalidad de los datos en la nube, el proceso de ejecución debe ser capaz de buscar la información necesaria desde otras fuentes de datos, reduciendo la carga de atención del usuario para el ingreso de los mismos.

## 1.1 Sistemas Basados en Conocimiento

Un SBC facilita la toma de decisiones a usuarios no expertos, utilizando el conocimiento de un experto o especialista humano [1]. Los SBC están diseñados para resolver problemas complejos utilizando reglas generadas de acuerdo al conocimiento extraído y educido del experto [2].

La arquitectura de un SBC puede dividirse en dos partes: a) una dependiente de la problemática a resolver, y b) otra independiente, formada por el Motor de Inferencia (MI). La parte dependiente es la Base de Conocimiento (BC) definida como la unión del conjunto de aserciones y reglas en los que se plasmó el conocimiento del experto. La BC está conformada por la Base de Hechos (BH) y la Base de Reglas (BR).

La BH tiene como función proveer al MI los datos sobre el problema, lo que le permite activar las reglas declaradas en la BR, mostrar los resultados obtenidos y generar una explicación de los pasos tomados.

## 1.2 Dispositivos Móviles

Los dispositivos móviles (celulares inteligentes, tablets, y similares) que han surgido en los últimos años tienen en su mayoría muy buena capacidad de procesamiento [3], lo que facilita la implantación de SBC en estos dispositivos.

La ventaja principal de poder ejecutar un SBC sobre un dispositivo móvil es justamente la movilidad del dispositivo. Esto genera la posibilidad de tener un experto siempre al alcance de la mano.

Para el desarrollo de SBC sobre dispositivos móviles, se debe tener en cuenta el sistema operativo, actualmente el mercado está concentrado en dos sistemas operativos, Android con el 80% e iOS con el 17% [4]. Programar para Android requiere un menor costo operativo, se posee una mayor penetración del mercado y permite una mayor variedad de desarrollos e integraciones entre aplicaciones.

De los varios SBC que existen, hay uno que se destaca sobre el resto, e2gDroid para Android [5]. e2gDroid es el único que posee un MI independiente y está preparado para ejecutar cualquier BC que el usuario le provea.

## 1.3 Trabajos Previos

La integración entre los dispositivos móviles y los SBC comenzaron hace años con un sistema funcionando en un servidor externo y utilizando el dispositivo únicamente como pantalla para mostrar resultados [6], luego, al incrementar la capacidad de procesamiento de los dispositivos móviles, fue posible crear aplicaciones nativas.

Algunos trabajos han tratado la implementación de SBC en el área pedagógica, con el fin de expandir la actual educación a distancia [7]; la integración con la medicina y el procesamiento móvil [8], [9]; o creando un sistema de asistencia al conductor de vehículos (Artemisa) [10]. Otros trabajos han dejado abiertas líneas de investigación referidas a la integración de SBC con el modelado basado en agentes, combinándolos mediante una arquitectura llamada ARTIMIS [11]; mientras que otros definieron un marco de trabajo genérico que plantea una solución para la comunicación entre los

agentes y sus usuarios humanos [12]. Otro modelo de agentes presentado es ANDROMEDA, una plataforma orientada a embeber agentes en el sistema operativo Android [13]. También se deben tener en cuenta a Drools, librería que permite implementar un MI [14], [15]; y d3web, un SBC web [16].

## 2 Comparación de Sistemas Basados en Conocimiento Existentes

Se investigaron las distintas soluciones existentes para dispositivos móviles presentándose en la tabla 1 una síntesis comparativa de las características deseadas, enunciadas en la misma, que muestra la situación actual con respecto al desarrollo de un SBC en dispositivos móviles.

**Tabla 1.** Comparación de soluciones de SBC en dispositivos móviles.

Características de aplicaciones	e2gDroid	Artemisa	Drools	d3web
Compatibilidad de Sistema Operativo	Android	Android	Android	Android / iOS
Corre nativo al SO	Sí	Sí	Sí	No
Requiere desarrollo para ser utilizado	No	No	Sí <sup>1</sup>	Sí <sup>2</sup>
Requiere conexión activa con Internet	No	Sí	No	Sí
Aplicación independiente de la base de conocimiento	Sí	No	No	Sí
Permite captura de datos automatizada	No	Sí	No	No

Del análisis presentado se desprende como síntesis que ninguna opción es una solución completa a la problemática que se investiga, en términos de alcanzar una solución que brinde una ejecución de la adquisición de datos automatizada, siendo las detalladas sólo soluciones parciales.

e2gDroid es el único que demuestra la posibilidad de tener un sistema basado en conocimiento funcionando en un dispositivo móvil y a la vez ser independiente de la base de conocimiento que se ejecuta, pero no permite la captura de datos automatizada.

## 3 Solución Propuesta

El primer punto planteado en el desarrollo, fue la selección de Android como sistema operativo. Si bien iOS posee un ambiente más controlado, una menor cantidad de errores en el sistema operativo y una menor fragmentación de versiones, Android es una solución más económica, posee un mayor mercado local y permite un mayor número de desarrollos posibles.

Se tomó como punto de inicio para el desarrollo la flexibilidad y madurez de e2gDroid. La arquitectura del SBC funciona de forma local en el dispositivo móvil,

<sup>1</sup> Dado que Drools es una librería, es necesario programar una aplicación que las emplee.

<sup>2</sup> d3web requiere la instalación y configuración del servidor web donde va a estar alojado.

pero también puede actualizar las BC desde un repositorio central remoto, simplificando el mantenimiento de las mismas. También es capaz adquirir las respuestas por sí mismo, es decir, además de esperar que el usuario ingrese los valores para continuar con la ejecución, la aplicación obtiene los datos consultando fuentes externas. Estas fuentes pueden ser direcciones web o dispositivos a los cuales se comunicará.

### 3.1 Arquitectura

Luego de relevar las arquitecturas existentes [17], se optó por utilizar una arquitectura híbrida. En esta arquitectura el proceso se ejecuta sobre el dispositivo móvil, conservando conexiones a servidores y dispositivos externos. Estas conexiones permiten la actualización y adquisición de la BC y de la Base de Automatización de captura de datos, o simplemente Base de Automatización, (BA). Esta nueva base contiene la descripción de a dónde ir a encontrar el dato que alimenta a la BH de la BC. En la figura 1 se detalla la arquitectura propuesta.

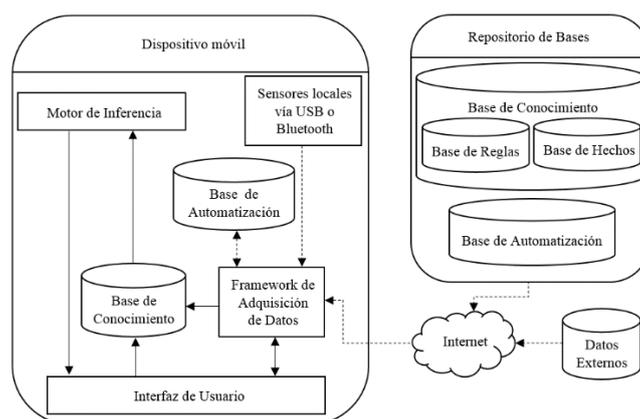


Fig. 1. Arquitectura conceptual del SBC para dispositivos móviles.

El Motor de Inferencia (MI) de la aplicación corre localmente en el dispositivo, actuando sobre la copia local de la BC. El framework de adquisición de datos cumple un doble rol, mantiene actualizadas las BC y BA, y la automatización de la captura de datos del exterior. La ventaja principal de la arquitectura radica en la centralización del mantenimiento de las BC, permitiendo que los cambios impacten en simultáneo en cada dispositivo móvil que utilice las mismas. La adquisición de datos desde sensores locales vía bluetooth o usb se encuentra prevista en futuras líneas de trabajo.

### 3.2 Base de Conocimiento

Para la configuración de la adquisición de datos automatizada se desarrolló un marco de trabajo o framework que facilite la configuración de la captura de cada dato. Estos

datos representan los hechos que componen la BH. Se define así la BA donde se detallan los datos para la búsqueda automática en la nube y la actualización de los valores. Para las BC, se adoptó el formato de bases definido por e2gDroid [18], para las BA propuestas se utilizó el lenguaje XML, estándar para la transferencia de datos [19].

Ejemplo de una BR de un SBC especializado en predecir si va a llover y recomendar el uso de paraguas. Se detallan tres preguntas al usuario, una regla de inferencia y el atributo objetivo.

```
REM Usar o no paraguas

RULE [Va a llover próximamente?]
If [nublado] = True and
[presión_actual] < [presión_3horas]
Then [recomendación] = "Llevar paraguas"

GOAL [recomendación]

PROMPT [nublado] YesNo
"Está nublado?"

PROMPT [presión_actual] Numeric
"Cuál es la presión atmosférica actual?"

PROMPT [presión_3horas] Numeric
"Cuál fue la presión atmosférica hace 3 horas?"
```

Ejemplo de una configuración de automatización de captura de datos para la BH del SBC, donde se detalla las rutas URL para la búsqueda automática de los valores de los atributos presión\_actual y presión\_3horas, y el valor por defecto para el atributo nublado.

```
<atributo>
<nombre>nublado</nombre>
<valor>True</valor>
</atributo>
<atributo>
<nombre>presión_actual</nombre>
<url>http://infinnitem.dyndns.pro/data/presion_actual.html
</url>
</atributo>
<atributo>
<nombre>presión_3horas</nombre>
<url>http://infinnitem.dyndns.pro/data/presion_3horas.html
</url>
</atributo>
```

### 3.3 Desarrollo de la Aplicación

El desarrollo efectuado para la construcción de un SBC se realizó en el lenguaje de programación Android [20], que funciona de forma local en un dispositivo móvil con Android (desarrollado para versión del S.O. 2.3.3 en adelante, probado en un Samsung I9300 con Android 4.1.2). La aplicación desarrollada permite: a) la explotación independiente de BC mediante el MI de la aplicación; b) la obtención de hechos a través de los datos provistos por el usuario como del exterior, en este último caso en forma automática y transparente al usuario al ser capturados desde la nube; c) el mantenimiento de las bases (BC y BA) se realice contra un repositorio central; d) revisar la trazabilidad de las reglas utilizadas.

Para correr el SBC se debe seleccionar y cargar la BC a utilizar. El sistema luego intentará adquirir los hechos a través de lo configurado en la BA, y de ser necesario generará preguntas al usuario hasta alcanzar un resultado. Para esto se realizan sucesivas búsquedas hacia atrás revisando cada regla, hasta encontrar un camino lógico que puede completar. Al finalizar todo el proceso, se mostrará en pantalla el resultado obtenido. En la figura 2 se compilan distintas capturas de pantallas de la aplicación corriendo las bases de ejemplo anteriormente detalladas. La primera captura muestra la pantalla de selección de BC; la segunda ejemplifica una pregunta realizada al usuario; y en la tercera se observa el resultado obtenido, con el agregado de una búsqueda relacionada realizada en Internet.

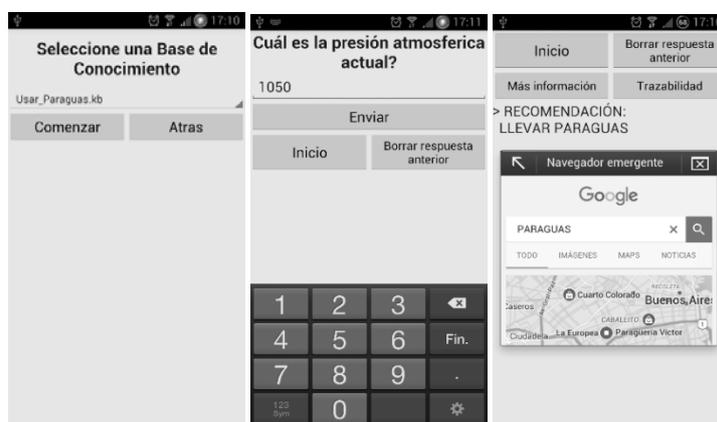


Fig. 2. Capturas de pantalla del nuevo SBC en ejecución.

Respecto al mantenimiento de las bases, el mismo se realiza desde una aplicación web con acceso al repositorio central, desde la cual se pueden crear, actualizar y borrar BC, como así también BA. La figura 3 muestra una imagen de la aplicación desarrollada para la gestión de la BC vía web. En el lateral izquierdo se pueden ver las BC publicadas para su selección, mientras que en sector central se observa el cuadro de edición de la BC seleccionada. En la figura 4 se captura el mantenimiento de la BA mediante la aplicación web. La aplicación posee una lista de BC en la izquierda y en la

parte central se ingresan las declaraciones, como ser una URL a un atributo en la BC seleccionada, en este caso presión atmosférica correspondiente a la BC Usar Paraguas.



Fig. 3. Captura de pantalla de la aplicación web del repositorio central – BC.

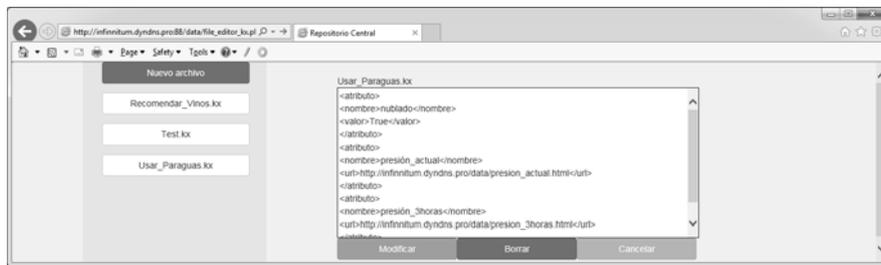


Fig. 4. Captura de pantalla de la aplicación web del repositorio central – BA.

Se presenta en la figura 5 el diagrama de componentes, en el cual se visualizan las interacciones entre las distintas interfaces del sistema. Mediante la aplicación móvil se puede interactuar con el MI y con el módulo de actualización de las bases (BC y BA). Tanto el MI como el módulo de actualización requieren para su funcionamiento interactuar con las BC residentes en el móvil. El módulo de actualización realiza la actualización extrayendo la última versión de las bases desde el repositorio remoto. Por el otro lado, utilizando el sistema web el usuario le da mantenimiento al repositorio de las BC y al de las BA.

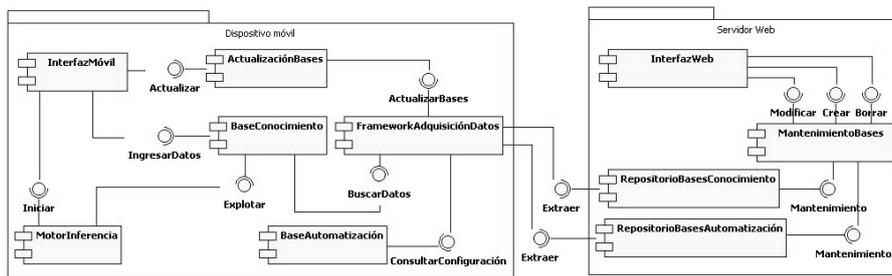


Fig. 5. Diagrama de componentes de la solución propuesta.



## 4 Pruebas de la Solución

Se efectuaron distintas pruebas con el fin de comprobar la arquitectura propuesta y su integración con la Base de Conocimiento (BC) y la Base de Automatización (BA). Estas pruebas se efectuaron en dominios particulares: predicción de clima, recomendación de lugares turísticos, diagnóstico de fallas de PC y maridaje de vinos. El desarrollo de pruebas se realizó con un set de siete casos para cada BC, con distintas configuraciones y objetivos puntuales. En la tabla 2 se detalla el set empleado para cada uno de los casos mencionados.

**Tabla 2.** Pruebas efectuadas sobre la solución propuesta.

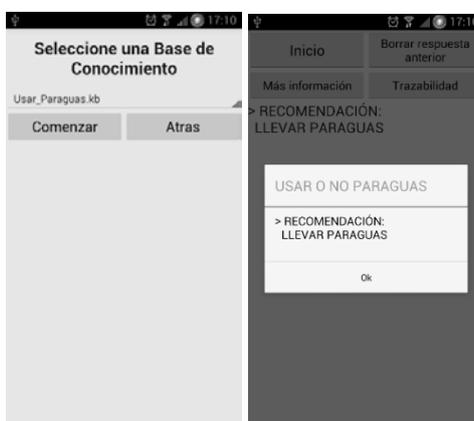
Prueba	Objetivo
Comparación entre Motores de Inferencia (MI)	Probar el MI desarrollado contra uno existente. Se utilizó el mismo caso en el nuevo SBC y en e2gDroid.
Automatización de un solo atributo	Lograr una ejecución con una BA configurada con un valor fijo para un solo atributo.
Automatización de dos atributos	Obtener una ejecución con una BA que tenga dos atributos y sus respectivos valores fijos.
Automatización de la totalidad de los atributos	Ejecutar de forma totalmente automatizada, agregando a la configuración del caso de prueba anterior un valor fijo para el tercer y último atributo de la BC.
Automatización con URL	Conseguir el mismo resultado anterior pero que un atributo, en vez de tener un valor fijo, dependa de la búsqueda en una URL.
Automatización con URL no existente	Evaluar la tolerancia a fallos sin la URL configurada no existe. Se ejecutó la prueba anterior modificando el entorno para que la URL sea un recurso inválido.
Automatización con URL y borrado	Comprobar la capacidad del SBC de explorar otras rutas cambiando las respuestas obtenidas.

Continuando con el ejemplo presentado en la sección 3.2, se detalla en la tabla 3 un caso de prueba utilizando una BC que recomienda el uso de un paraguas dependiendo de la situación climática, logrando una ejecución totalmente automatizada, de tal forma que el usuario no ingrese ningún valor de forma manual. La Base de Hechos (BH) se confecciona con un valor fijo y con búsquedas de valores en dos direcciones web declaradas en la BA. La ejecución de la prueba se puede apreciar en la figura 7<sup>3</sup>. Se observa que se logra el resultado deseado, sin la necesidad de ninguna pantalla intermedia que pregunte al usuario el valor de ningún atributo.

<sup>3</sup> Para ver la ejecución completa referirse a <https://goo.gl/z5EqxK> (<https://www.youtube.com/playlist?list=PL3WKnmIEYB91sJ5XvqbGRQxaDIXSsJEsH>)

**Tabla 3.** Caso de prueba detallado.

Pregunta	Respuesta	Origen	Configuración
¿Está nublado?	True	Base de hechos	Valor fijo: True
¿Cuál es la presión atmosférica actual?	1050	Base de automatización	Dirección URL: http://infinitum.dyndns.pr o/data/presion_actual.html
¿Cuál fue la presión atmosférica hace 3 horas?	1100	Base de automatización	Dirección URL: http://infinitum.dyndns.pr o/data/presion_3horas.html
Recomendación	Llevar paraguas		



**Fig. 7.** Capturas de pantalla del caso de prueba en ejecución.

## 5 Conclusión y Futuras Líneas de Trabajo

La arquitectura propuesta contribuye a la implementación de aplicaciones en dispositivos móviles que brinden servicios útiles, facilitando la integración de sistemas basados en conocimiento y la actualización de datos, como así también el mantenimiento de la base de conocimiento. En el marco de las futuras líneas de trabajo se prevé la integración y experimentación con sensores y artefactos en el contexto del Internet de las cosas.

## 6 Referencias

1. Jackson, Peter (1998). *Introduction To Expert Systems* (3 ed.), Addison Wesley, ISBN 9780201876864.

2. Cornelius T. Leondes (2002). *Expert systems: the technology of knowledge management and decision making for the 21st century*, ISBN 9780124438804.
3. Smart Insights (2016). *Mobile marketing statistics 2016*, <<http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/>>.
4. Statista (2016). *Smartphone OS global market share 2009-2015*, <<http://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems>>.
5. eXpertise2Go (2016), <<http://www.expertise2go.com>>.
6. Sandeep Kumar and Ravi Bhushan Mishra (2010). *Web-based expert systems and services. The Knowledge Engineering Review*. doi:10.1017/S0269888910000020.
7. Asabere, Enguah (2012). *Integration of Expert Systems in Mobile Learning*, University of Cape Coast, Cape Coast, Ghana, <[http://esjournals.org/journaloftechnology/archive/vol2no1/vol2no1\\_8.pdf](http://esjournals.org/journaloftechnology/archive/vol2no1/vol2no1_8.pdf)>.
8. Lopes, Abreu, Batista (2008). *Innovation in Telemedicine: en Expert Medical Information System Based on SOA, Expert Systems and Mobile Computing*, Federal University of Maranhão, Brasil, <[http://link.springer.com/chapter/10.1007/978-1-4020-8741-7\\_22](http://link.springer.com/chapter/10.1007/978-1-4020-8741-7_22)>.
9. Mitchell, Sharma, Modi, Simpson, Thomas, Hill, Goyal (2011). *A Smartphone Client-Server Teleradiology System for Primary Diagnosis of Acute Stroke*, Estados Unidos de América, <<http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3221380>>.
10. Corcoba Magaña, Muñoz-Organero (2011). *Artemisa: using and android device as an eco-driving assistant*, Universidad Carlos III de Madrid, Madrid, España, <[http://e-archivo.uc3m.es:8080/bitstream/10016/13091/1/android\\_JMTC\\_2011.pdf](http://e-archivo.uc3m.es:8080/bitstream/10016/13091/1/android_JMTC_2011.pdf)>.
11. Mestizo Gutierrez, Guerra Hernandez, Parra Loera (2008). *Desarrollo de un Centro de Ayuda Inteligente mediante el uso de Tecnologías de Internet*, Veracruz, México, <<http://cdigital.uv.mx/bitstream/123456789/32358/1/mestizogutierrezsonia.pdf>>.
12. France Telecom — CNET (1997). *ARTIMIS: Natural Dialogue Meets Rational Agency*, Francia, <<http://www.ijcai.org/Past%20Proceedings/IJCAI-97-VOL2/PDF/035.pdf>>.
13. Aguero, Rebollo, Carrascosa, Julian (2009). *Developing intelligent agents on the Android platform*, Universidad Politécnica de Valencia, Valencia, España, <[http://www.gt-ia.upv.es/sma/tools/Andromeda/archivos/downloads/Description\\_of\\_Andromeda.pdf](http://www.gt-ia.upv.es/sma/tools/Andromeda/archivos/downloads/Description_of_Andromeda.pdf)>.
14. StackOverflow (2011). *Integration of Drools (Expert System) with Android Projects*, <<http://stackoverflow.com/questions/8646934>>.
15. Verlaenen, Kris (2011). *jBPM5 lightweight? Running on Android!*, <<http://kverlaen.blogspot.com.ar/2011/03/jbpm5-lightweight-running-on-android.html>>.
16. d3web (2016). *Main*, <<http://www.d3web.de>>.
17. Albin, Stephen (2003). *The Art of Software Architecture: Design Methods and Techniques*, ISBN 9780471228868.
18. eXpertise2Go (2016), *e2gRuleWriter decision table software: e2gRuleEngine knowledge base generator*, <<http://www.expertise2go.com/e2g3g/e2g3gdoc/e2gRuleWriterRef.htm>>.
19. W3C (2016). *Extensible Markup Language (XML) 1.1 (Second Edition)*, <<https://www.w3.org/TR/2006/REC-xml11-20060816>>.
20. Android (2016). *Android Developers*, <<https://developer.android.com>>.