

<http://dx.doi.org/10.7236/JIIBC.2016.16.5.137>

JIBC 2016-5-21

# TOS와 Mobile device 간의 펍섭 QoS를 지원하는 대량 커넥션 서비스 브로커 설계

## Design of a Service Broker for Large Scale Connections to Support Pubsub QoS between TOS and Mobile Devices

전영준\*, 황희정\*\*

Young-Jun Jeon\*, Hee-Joung Hwang\*\*

**요약** 2-Step 개방형 시스템은 힐링플랫폼과 개인건강 문서저장소 사이의 중계를 목적으로 제안되었다. 또한 TOS는 프로바이더(provider)의 건강문서 접근/요청 과정을 실시간으로 모니터링 하기 위해 대량 커넥션 기반의 pubsub서비스를 고려하여 설계되었다. TOS에서는 pubsub시의 통신 프로토콜의 용도로 WebSocket을 사용하고 있다. 그러나 힐링플랫폼의 사용자 단말인 모바일 기기의 저품질 무선 네트워크 운영 환경을 감안해볼 때, 전송 프로토콜 뿐 아니라 QoS를 지원하는 메시징 프로토콜 또한 추가될 필요가 있다. MQTT는 모바일 기기에 최적화된 경량 메시징 프로토콜로서 지속/저품질인 무선 네트워크 상황을 감안한 신뢰성 있는 메시징 QoS를 정의하고 있다. 본 논문에서는 힐링플랫폼의 사용자 단말인 모바일 기기를 고려하여 대량의 커넥션 및 펍섭시의 QoS를 지원하는 MQTT 프로토콜 기반의 메시지 브로커를 설계한다. 이를 위해, 우선 TOS와 MQTT 메시지 브로커 간의 모델을 설계하고, 제안 설계를 바탕으로 프로토타입을 구현한 후, 마지막으로 MQTT 클라이언트 툴을 사용한 load-test를 통해 기존연구와의 성능지표를 비교하도록 한다.

**Abstract** A two-step open system(TOS) was proposed to relay between a healing platform and a repository of personal health documents. TOS was designed by taking into consideration the pubsub service based on large scale connections to monitor a provider's access/request process for health documents in real time. TOS, however, uses WebSocket as a communication protocol in case of pubsub. Given the operational environment of low quality wireless networks for mobile devices that are user terminals in a healing platform, there is a need to add a messaging protocol to support QoS as well as a transmission protocol. As a light messaging protocol optimized for mobile devices, MQTT defines reliable messaging QoS to consider a wireless network situation of low speed/low quality. This study designed an MQTT protocol-based message broker to support QoS in case of large scale connections and pubsub by taking into consideration mobile devices that are user terminals in a healing platform. After designing a model between TOS and MQTT message broker, the study implemented a prototype based on the proposed design and compared it with its counterparts from previous studies based on the performance indicators in a load-test with the MQTT client tool.

**Key Words** : ICT Healing platform, TOS, MQTT, Vert.x

\*정회원, IDLE CO.,LTD.

\*\*정회원, 가천대학교 IT대학 컴퓨터공학과(교신저자)

접수일자 : 2016년 5월 24일, 수정완료 : 2016년 9월 6일

게재확정일자 : 2016년 10월 7일

Received: 24 May, 2016 / Revised: 6 September, 2016 /

Accepted: 7 October, 2016

\*\*Corresponding Author: hwanghj@gachon.ac.kr

Dept. of Computer Engineering College of Information Technology, Gachon University, Korea

## I. 서 론

ICT 힐링플랫폼에서는 다음과 같은 2-Step운용을 전제로 한다. Step1은 병원과 같은 데이터 공급자가 사용자 민감 정보를 제외한 개인의 건강정보나 라이프로그를 사용 동의를 받은 상태에서 힐링플랫폼을 통해 개인 저장소(클라우드 스토리지)<sup>[1][2]</sup>에 저장하는 것을 의미한다<sup>[3][4][5]</sup>. Step 2는 서비스 공급자가 모바일 플랫폼을 통해 개인 저장소로부터 사용자가 접근을 허용한 건강데이터의 사용을 의미한다<sup>[6][7]</sup>. 2-Step 개방형 시스템(TOS)<sup>[8]</sup>은 위와 같은 힐링플랫폼과 개인건강 문서 저장소사이의 데이터 중계를 목적으로 설계되었으며, 건강문서의 접근/요청 과정을 실시간으로 모니터링 하기 위해 대량 커넥션 기반의 pubsub(publish/subscribe) 서비스가 추가 고려되었다. 이전 연구인 TOS의 pubsub 서비스설계 및 검증 과정에서는 통신 프로토콜의 용도로 WebSocket을 사용하였으나, 모바일 기기 특유의 저속/저품질 네트워크를 감안해볼 때 통신프로토콜 뿐 아니라, 신뢰성 있는 메시징 프로토콜을 적용할 필요가 있다. 이와 관련해 MQTT는 모바일 기기에 최적화된 경량 메시징 프로토콜로서 저속/저품질의 모바일 네트워크상황을 고려한 신뢰성 있는 메시징 QoS<sup>[9][10]</sup>을 정의하고 있다. 따라서 본 논문에서는 힐링플랫폼의 사용자 단말인 모바일 기기를 고려하여 대량의 커넥션뿐 아니라 펍업(pubsub) 시의 QoS를 지원하는 MQTT<sup>[11]</sup> 프로토콜 기반의 메시지 브로커<sup>[12]</sup>를 설계하도록 한다. 이를 위한 구성은 다음과 같다. 우선 TOS와 메시지 브로커 간의 메시지 모델인 TOS-Broker를 설계하도록 하며, 메시지 브로커는 대량 커넥션 처리기술인 Vert.x<sup>[13]</sup>를 바탕으로 오픈소스 MQTT 브로커인 Mosquitto<sup>[14]</sup>를 활용하도록 한다. 이후, 설계된 모델의 적합성을 확인하기 위해서 MQTT 클라이언트 툴(eclipse mqtt)을 사용해 부하테스트를 실행하도록 한다. 최종 목표는 TOS가 수행하는 모니터링 pubsub의 가용성과 신뢰성을 동시에 만족할 수 있는 구체적인 방안을 제시하는 것이다.

## II. 관련 연구

### 1. MQTT (MQ Telemetry Transport)

MQTT는 경량의 Publish/Subscribe(pubsub) 메시징 프로토콜로서, 무선 및 저대역폭 네트워크에서의 사용을

목표로한다<sup>[11]</sup>. MQTT 메시지는 MQTT 메시징 브로커 서버를 통해 서로 교환된다. MQTT 메시징 모델에서 Publisher는 토픽을 발행하기 위한 목적으로 사용되며, Subscriber은 토픽을 구독하기 위한 목적으로 Broker 서버에 연결된다. 하나 이상의 PubSub 클라이언트는 브로커에 연결하여 구독내용에 해당하는 토픽을 발행하거나 구독할 수 있다. MQTT는 3단계의 QoS(Quality of service)를 제공하며 정의된 스펙은 다음과 같다. 0 : 메시지는 한번만 전달하며, 전달여부를 확인하지 않는다. Fire and Forget 타입이다. 1 : 메시지는 반드시 한번 이상 전달된다. 하지만 메시지의 핸드셰이킹 과정을 엄밀하게 추적하지 않기 때문에, 중복 전송될 가능성 있다. 2 : 메시지는 한번만 전달된다. 메시지의 핸드셰이킹 과정을 추적한다. 높은 품질을 보장하지만 성능의 희생이 따른다.

### 2. Vert.x

Vert.x<sup>[13]</sup>은 손쉽게 확장 가능하고 비동기 병렬 어플리케이션을 위한 차세대 소켓 통신 프레임워크로서 이벤트 기반 프로그래밍 모델을 제공하며 논블록킹 어플리케이션을 위한 비동기 프로그래밍 기술을 지원한다. Vert.x의 기본 실행 단위는 Vertical이며, Event-Loop 스레드에 의해 하나의 vert.x 인스턴스에서 동시에 여러개의 Vertical이 실행될 수 있다. Vert.x의 각각의 verticle이 스택 멤버, 전역변수 또는 다른 수단으로 verticle끼리 통신할 수 없도록 각자의 classloader를 가짐으로써 verticle들을 분리한다. Vert.x 인스턴스들은 같은 호스트 혹은 동일 네트워크상의 다른 호스트에서 동작할 수 있으며 IMDG(in-memory data grid)기반의 EventBus를 통해서로 통신이 가능하다<sup>[15]</sup>.

### 3. TOS (Two-Way Open System for Healing-Platform)

ICT 힐링플랫폼 상에서 개인화된 건강정보의 중계는 2-Step운용을 전제로 한다. Step1은 데이터 공급자가 사용자 민감 정보를 제외한 개인의 건강정보를 모바일 플랫폼을 통해 개인 저장소(클라우드 스토리지)<sup>[1][2]</sup>에 저장하는 것을 의미한다<sup>[4][5]</sup>. Step 2는 서비스 공급자가 모바일 플랫폼을 통해 개인 저장소로부터 건강데이터에 접근하는 것을 의미한다<sup>[7]</sup>. TOS<sup>[8]</sup>은 개인화된 건강 데이터 접근을 위한 2-Step 개방형 시스템은 힐링플랫폼과 개인 건강데이터 저장소간의 중계를 설계하였으며 2-Step 데

이더 처리과정을 실시간으로 전송(모니터링)하기 위한 대량 커넥션 기반의 pubsub 서비스를 고려하여 설계되었다. 또한 OAuth2 기반의 Restful<sup>[16]</sup> API 로 서비스하도록 설계되었으며 20여종의 API를 제공하고 있다.

### III. 커넥션 서비스 브로커의 설계

기존에 수행한 TOS 설계에서는 사용자별 개인 저장소와의 중계상황을 모니터링 하기 위해 대량 커넥션기술인 소켓프레임워크인 Vert.x 기술을 기반으로 개인화된 건강데이터처리의 모니터링이 적용되었다. 그림 1은 이를 위한 구조이며, 각 IMDB를 제외한 메인 API 서버, 커넥션 서버 및 변환 서버는 데이터 그리드를 통해 클러스터링 하는 것으로 설계되었다.

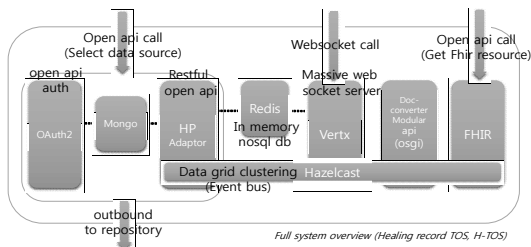


그림 1. TOS 개요 블록 다이어그램  
 Fig. 1. TOS overview block diagram

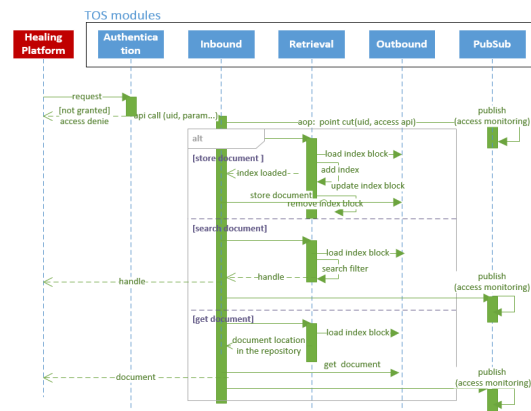


그림 2. TOS 모듈들 publish 요청 순차도  
 Fig. 2. Publish request sequence diagram of TOS modules

그림 2는 TOS에서 pubsub의 발행 시점을 나타낸다. 건강문서의 접근/요청 과정을 실시간으로 모니터링 하기

위해 대량 커넥션 기반의 pubsub(publish/subscribe) 서비스가 적용되는 시점을 나타낸다. 그림 2에서 모니터링 publish/subscribe는 다음과 같은 절차로 수행된다. 1.subscribe를 요청(topic수신 관제)하기 위해서는, Vert.x 서버에 접속해야 한다. 2.Vert.x 서버(연결서버)에 접속하기 위해서는 접속 인증에 해당하는 Token을 발급받는다. 3.Token은 TOS API를 통해 발급받는다. 4.발급받은 Token으로 Vert.x서버에 접속한다. 5.수신하고자 하는 토픽 변경시 TOS API 혹은 Vert.x서버를 통해 토픽 변경 등의 관제 명령을 전송한다. 본 연구의 대상인 메시지 브로커는 연결(세션)서버인 Vert.x 서버를 대상으로 수행되며, 메시지의 발송현황을 IMDG를 통해 IMDB (In-Memory Data Base)를 기록함으로써 추후 TOS API에서 메시지 브로커의 발송 상태 정보에 접근할 수 있다.

### 1. TOS-Broker 모델 설계

그림 3은 본 논문이 제안하는 메시지 브로커 모델인 TOS-Broker의 블록 다이어그램이다. 그림 2의 우측 pubsub 블록은 그림 3의 QoS 0 publish에 해당하며, 그림 3의 중단 MQTT 브로커는 그림 1의 vetx 블록을 대체한다. MQTT 브로커는 선택적으로 브리지 모드를 사용할 수 있으며, 이는 A 브로커에서 수신한 publish 메시지를 동일 망 혹은 동일 장치의 B 브로커에게 전송하기 위해 사용된다. 통상 TOS에서 관제 이력을 나타내는 publish와 통합관제 환경이 구축된 관리자 화면등의 우선 상황에서는 QoS를 0으로 지정하여도 무방하나, wireless 환경인 모바일 사용자는 QoS 1 혹은 QoS 2를 지정하여 신뢰성있는 메시지 전달을 지원할 필요가 있다.

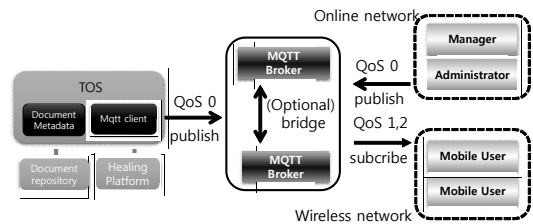


그림 3. QoS를 지원하는 브로커 및 TOS의 블록다이어그램  
 Fig. 3. Block diagrams of TOS and brokers that support QoS

TOS-Broker모델에서 Broker의 실제적인 연결기술을 제공하는 vert.x는 비동기 기반의 프로세스로 수행하기 때문에 TOS와 데이터 그리드인 Hazelcast<sup>[13]</sup>로 event 처

리하며 이와 관련한 순차도는 그림 4가 나타내고 있다. 그림 5는 기존 연구인 대량 커넥션에 MQTT 브로커인 Mosquitto를 적용하기 위한 클래스 다이어그램이다. 제안된 Broker는 Vert.x의 통신 프로토콜을 바탕으로 Mosquitto의 MQTT 메시지 인코딩/디코딩 기술이 적용되어 socket 및 메시지 큐잉을 처리하는 세션클래스로 구성되어 있다.

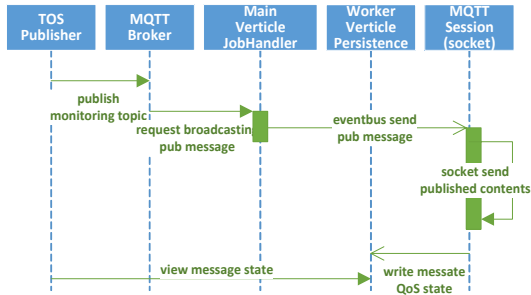


그림 4. TOS와 MQTT 세션간의 pubsub 순차도  
Fig. 4. Pubsub diagrams between TOS and MQTT session.

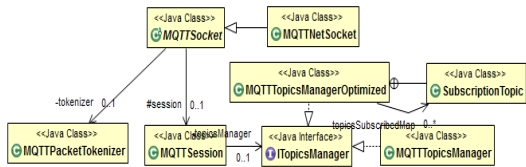


그림 5. MQTT Broker의 클래스 다이어그램  
Fig. 5. Class diagram of MQTT Broker.

메시지 브로커에 도착한 publish 메시지는 vert.x의 verticle에 전송되어 연결서버에 접속중인 연결(세션)에 해당 메시지를 브로드캐스팅을 요청한다. 이때 각 연결은 그림 4와 같이 별도의 세션을 생성하여 브로드 캐스팅한 콘텐츠를 파싱하여 실제 자신의 subscribe topic에 해당하는지 확인한다.

확인결과 자신의 topic 리스트에 부합하는 메시지가

면, vert.x의 커넥션 write를 통해 해당 publish contents를 발송한다. 이때 각 커넥션들은 mqtt 프로토콜로 클라이언트에 메시지를 발송하며, 발송결과인 QoS 상태를 반환한다. 최종 QoS 결과는 worker verticle persistence인 IMDB인 기록된다. 위 과정은 non-block 기반의 프로세스로 설계되었으며, 이는 연결서버 내에서 blocking-queue 등의 blocking thread기술 사용시 single thread 기반인 vert.x 플랫폼에 freezing이 발생하는 위험을 방지하기 위함이다.

## 2. 메시지 브로커 로드 테스트

부하테스트는 그림 6과 같이 단일 서버급장비를 통해 수행되었으며 클라이언트는 windows2종 linux 4종으로 구성하여 각기 다른 os를 탑재한 pc가 사용되었다. 서버와 클라이언트 사이는 L2 스위치가 사용되었으며 서버와 L2 구간은 1Gbit 대역폭 회선을 설정하고, L2와 클라이언트는 100Mbit 대역폭 회선이 설정되었다. 그림 6의 실행 스크립트의 의미는 100byte payload 자동 생성하여 각 클라이언트에 MQTT QoS 1로 100회씩 Publish 하도록 명령하는 것이다.

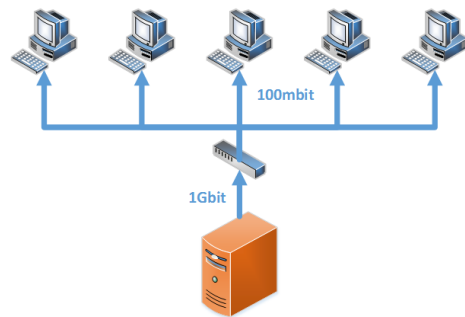


그림 6. 로드테스트 모델(Dell R430 - PowerEdge 6core 8gb ram: publish -h 192.168.0.156 -p 1883 -q 1 -mco 1 -mmc 100 -sid 100000 -pl 100 -log)

Fig. 6. LoadTest model

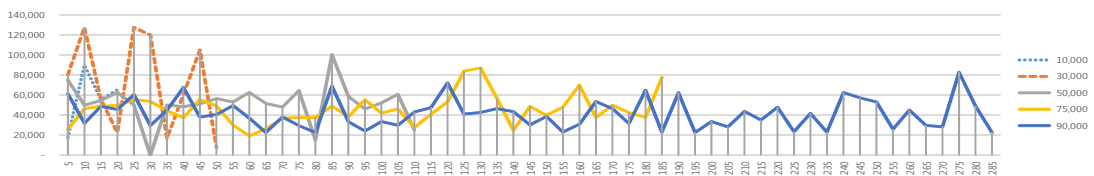


그림 7. TOS-브로커의 TPS 그래프 (10,000~90,000 concurrent connection)  
Fig. 7. TPS graph of TOS-Broker

표 1. 동접세션별 subscribe tps  
 Table 1. Per concurrent session, subscribe tps

Concurrent connection	Runtime (sec)	Send message(sum)	tos-broker avarage tps	tos-websocket avarage tps
10,000	20	1,020,101	57,022	10,000
30,000	50	3,068,597	72,337	30,000
50,000	110	5,100,101	51,604	50,000
75,000	185	7,650,101	46,194	75,000
90,000	285	9,330,101	41,249	90,000

그림 7은 그림 4의 프로세스를 바탕으로 그림 6의 구성하에 수행된 클라이언트 동접 연결별 TPS지표이다. 표 1은 그림 7의 결과를 평균치로 환산한 것이며 지난 TOS 연구의 성능지표를 그림 6의 구성으로 재실행하여 비교한 결과이다. 표 1의 connection은 동시 접속중인 클라이언트 커넥션 수이다. run time (단위 초)는 실험 시작 후 경과한 시간이다. send message는 클라이언트 전송한 메시지 수이다. TPS(Transaction Per Second)는 단위 초당 요청처리 능력이다. 수행 결과 그림 7처럼 제안 설계의 MQTT 브로커인 TOS-Broker 모델은 상/하한에 변위가 있으며, 이전 연구와 다른 특성을 나타내었다. 기존 연구는 목표 TPS에 맞추어 작업스케줄을 직접 조절하는 것과 달리 TOS-Broker는 10k(kilo)~90k의 동접 연결에 대해 최대성능을 내도록 큐잉함으로써 40k~50k 사이에서의 TPS를 나타내었다. 결과적으로 50k를 넘는 동접자 상황에서는 물리적인 큐잉 자원(메모리 등)의 요구한계로 인해 기존 연구 대비 저하된 성능을 나타내었으나, 기존 연구결과에서 제시하지 못한 QoS 및 메시지 큐잉이 적용된 점에서 기존 연구와의 차별점이 있다.

#### IV. 결 론

본 논문이 제안한 펍십 QoS를 지원하는 대량 커넥션 서비스 브로커 설계는 다음과 같은 특징이 있다. 우선 TOS의 메시지 수신 대상인, 모바일 기기 특유의 저속/저품질 네트워크를 감안하여 신뢰성 있는 메시징 프로토콜이 적용되었다. 다음으로 대량의 모바일 기기 접속을 고려한 대량 커넥션 유지기술이 적용되었다. 이를 위해 TOS와 메시지 브로커 간의 모델이 설계되었으며, 메시지 브로커는 대량 커넥션 처리기술인 Vert.x 를 바탕으로 오픈소스 MQTT 브로커인 Mosquitto를 활용하였다. 또한 설계된 모델의 적합성을 확인하기 위해서 MQTT 클

라이언트 (Eclipse mqtt client)를 사용해 부하테스트를 진행하였다. 테스트를 통해 기존 연구 대비 동시연결 성능은 다소 저하된 것으로 나타났으나, 기존에 제공하지 못하였던 QoS를 반영함으로써 pubsub 운용시의 신뢰성을 향상하였다. 결과적으로 본 설계를 통해, TOS 가 수행하는 모니터링 pubsub시의 성능 가용성과 메시지 신뢰성을 동시에 만족하는 구체적인 방안을 제시하였다.

#### References

- [1] Frank E, Gillett, "The Personal Cloud", Forrester Research, 2009.
- [2] Dropbox Developers, <https://www.dropbox.com/developers>
- [3] Seokjin Im, HeeJoung Hwang, "Development of Smart Health Client based on Real-Time Health Information Sharing Framework", The Journal of The Institute of Internet, Broadcasting and Communication (IIIBC), Vol.14, No.3, pp.131-137, 2014.6.
- [4] Kim, Il Kon., "Health Innovation for Aging Society", Healthcare informatics research Vol.21, No.2 pp.63-66. 2015.
- [5] International Organization for Standardization. Health informatics - Person-owned document repository for PHR applications and health information exchange. Geneva: International Organization for Standardization; (ISO/AWI TR 20055), 2015.
- [6] JeMin Song, MyungSic Kim, KyeongJa Jeong and MoonSun Shin, "RBAC-based health care service platform for individual recommended health information service", Journal of the Korea Academia-Industrial cooperation Society(JKAIS), Vol.15, No.3, pp.1740-1748, 2014.
- [7] YoungJun Jeon, HeeJoung Hwang, "Design of Two-Step Open System for Personalized Health Data Access", The Journal of The Institute of Internet, Broadcasting and Communication (IIIBC) Vol.15, No.4, pp.177-183, 2015.8.
- [8] YoungJun Jeon, HeeJoung Hwang, "Design of

Dynamic Buffer Assignment and Message model for Large-scale Process Monitoring of Personalized Health Data”, The Journal of The Institute of Internet, Broadcasting and Communication (JIIBC), Vol.15, No.6, pp.187-193, 2015.12.

- [9] S. Behnel, L. Fiege and G. Muhl, "On Quality-of-Service and Publish-Subscribe", Proc. 26th International Conference on Distributed Computing Systems Workshops, pp.20-20, 2006.7.
- [10] P. Bellavista, A. Corradi and A. Reale, "Quality of Service in Wide Scale Publish/Subscribe Systems", in IEEE Communications Surveys & Tutorials, Vol.16, No.3, pp.1591-1616, 2014.
- [11] MQ Telemetry Transport, <http://mqtt.org>
- [12] M. Collina, G. E. Corazza and A. Vanelli-Coralli, "Introducing the QEST broker: Scaling the IoT by bridging MQTT and REST", 2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications - (PIMRC), pp.36-41, 2012.9.
- [13] Vertx framework, <http://vertx.io/>
- [14] Mosquitto message broker, <http://mosquitto.org>
- [15] Hazelcast: In-Memory Data Grid, <https://hazelcast.com/>
- [16] Roy Fielding, "Architectural Styles and the Design of Network-based Software Architectures", Dissertation of Doctor of Philosophy in Information and Computer Science, University of California, IRVINE, 2000.

## 저자 소개

### 전 영 준(정회원)



- 2003년 2월 : 인천대학교 컴퓨터공학과(공학사)
- 2005년 8월 : 인천대학교 컴퓨터공학과(공학석사)
- 2010년 2월 : 인천대학교 컴퓨터공학과(공학박사)
- 2013년 3월 ~ 현재 : IDLE CO, LTD.

<주관심분야 : Software Engineering, u-Health, Big Data, Medical Informatics, OSGi>

### 황 희 정(정회원)



- 2000년 9월 : 인하대학교 컴퓨터공학과(공학석사)
- 2008년 2월 : 인천대학교 컴퓨터공학과(공학박사)
- 2000년 10월 ~ 현재 : 가천대학교 IT 대학 컴퓨터공학과

<주관심분야 : Software Engineering, u-Health, Big Data, Medical Informatics, Ubiquitous Computing.>

※ 이 논문은 2016년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No.B0101-16-0247, 개인 건강정보 기반 개방형 ICT 힐링 플랫폼 기술 개발)