JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지 Vol. 25, No. 1: 124~133, Jan. 2021

스마트팩토리에서 다중장치기반 실시간 장비 모니터링 및 이력관리 시스템 설계 및 구현

김동현[1] · 이재민[2] · 김종덕[3*]

Design and Implementation of Real Time Device Monitoring and History Management System based on Multiple devices in Smart Factory

Dong-Hyun Kim[1] · Jae-min Lee[2] · Jong-Deok Kim[3*]

[1]Research Professor, Dong-Nam Grand ICT R&D Center, Pusan National University, Busan, 46241 Korea
[2]Integrated course Student, Department of Information Convergence Engineering, Pusan National University, Busan, 46241 Korea
[3*]Professor, Department of Information Convergence Engineering, Pusan National University, Busan, 46241 Korea

요 약

스마트팩토리는 공장 내 장비에 센서를 부착하여 각종 데이터를 실시간으로 수집, 분석 및 모니터링함으로써 스스로 공정을 제어 및 통제하는 미래형 공장이다. 스마트팩토리에서 장비의 상태 및 이력을 현장에서 실시간으로 조회 및 생성하는 것은 매우 중요한 일이며 다양한 스마트 기기의 등장은 이것을 보다 효율적으로 수행할 수 있게 하고 있다. 본 논문은 실시간으로 장비의 상태 및 이력을 생성, 조회, 삭제 할 수 있는 다중장치 기반 시스템을 제안한다. 제안 시스템은 공장이라는 특수한 환경을 고려하여 안드로이드 시스템과 스마트 글래스 시스템을 동시에 사용한다. 스마트 글래스 시스템은 장비 인식을 위해 QR코드를 이용하였으며 음성 인식 기능을 이용하여 보다 효율적인 작업환경을 제공한다. 다중장치기반 실시간 장비 모니터링을 위한 시스템 구조를 설계하였으며 안드로이드 시스템, 스마트 글래스 시스템 및 웹 애플리케이션 서버를 구현함으로써 실용성을 보이겠다.

ABSTRACT

Smart factory is a future factory that collects, analyzes, and monitors various data in real time by attaching sensors to equipment in the factory. In a smart factory, it is very important to inquire and generate the status and history of equipment in real time, and the emergence of various smart devices enables this to be performed more efficiently. This paper proposes a multi device-based system that can create, search, and delete equipment status and history in real time. The proposed system uses the Android system and the smart glass system at the same time in consideration of the special environment of the factory. The smart glass system uses a QR code for equipment recognition and provides a more efficient work environment by using a voice recognition function. We designed a system structure for real time equipment monitoring based on multi devices, and we show practicality by implementing and Android system, a smart glass system, and a web application server.

키워드 : 장비모니터링 시스템, 장비이력관리 시스템, 스마트공장, 사물인터넷, 스마트글래스

Key word : Device monitoring system, Device history management system, Smart factory, Internet of things, Smart glass

Received 27 October 2020, **Revised** 9 November 2020, **Accepted** 10 November 2020

*** Corresponding Author** Jong-Deok Kim(E-mail:kimjd@pusan.ac.kr, Tel:+82-51-510-3519)
Professor, Department of Information Convergence Engineering, Pusan National University, Busan, 46241 Korea

Open Access http://doi.org/10.6109/jkiice.2021.25.1.124 print ISSN: 2234-4772 online ISSN: 2288-4165

ⓒThis is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(http://creativecommons.org/li-censes/by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright Ⓒ The Korea Institute of Information and Communication Engineering.

Ⅰ. 서 론

4차산업은 인공지능, 사물인터넷, 빅데이터, 모바일 등 첨단 정보통신기술이 경제, 사회 전반에 융합되어 혁신적인 변화가 나타나는 차세대 산업혁명으로 기존 산업과 서비스에 융합되거나 3D 프린팅, 로봇공학, 생명공학, 나노기술 등 여러 분야의 신기술과 결합하여 실세계 모든 제품, 서비스를 네트워크로 연결하고 사물을 지능화한다. 특히 생산공정에서 다양한 지능화 기술이 발전함에 따라 기계설비, 사람, 데이터를 서로 연결해 설비 운영의 효율화, 비용 절감, 새로운 제품과 서비스 제공이 가능하게 됨으로써 지능형 생산공장으로의 전환이 급속히 진행 중이다[1].

스마트팩토리는 설계 및 개발, 제조 및 유통 등 생산과정에 디지털 자동화 솔루션이 결합된 정보통신기술을 적용하여 생산성, 품질, 고객 만족도를 향상시키는 지능형 생산공장으로 공장 내 설비와 사물인터넷(IoT : Internet of Things)을 설치하여 공정 데이터를 실시간으로 수집하고, 이를 분석해 스스로 제어할 수 있게 만든 미래의 공장이다. 이를 위해서는 공장의 설비 단계부터 다양한 요소 기술을 고려하여야 하나 현실은 그렇지 못하다. 기존 설치된 설비에 다양한 센서를 부착하고 이를 활용한 공장 자동화 시스템에 관한 연구가 지속해서 진행 중이며 현재 사용되는 시스템의 활용도와 편의성을 높이기 위한 연구들이 병행되고 있다[2].

다양한 제품을 여러 공정 단계를 거쳐 생산하는 스마트팩토리에서 하나의 제품을 생산하기 위해서는 다수의 공정을 거치며 공정마다 다양한 장비들이 이용된다. 이러한 환경에서 장비의 상태를 실시간으로 모니터링 및 이력관리를 통한 상태 유지 관리는 매우 필요하며 장비의 상태를 빠르게 파악하고 대처함으로써 생산성 향상 및 품질을 보장할 수 있다. 그리고 이러한 관리를 보다 효율적으로 수행할 시스템은 언제나 필요하다. 그래서 사용의 편리성을 향상하고 언제 어디서나 장비의 상태를 모니터링 및 관리할 수 있는 시스템을 제안하고자 한다.

제안하는 시스템은 공장이라는 환경을 고려하여 안드로이드 시스템과 스마트 글래스 시스템을 동시에 사용하는 다중장치 기반 시스템이며 QR(Quick Response) 코드 및 SMS(Short Message Service)와 같은 다양한 정보수집 방법을 통합하도록 설계하고 구현하였다. 구현한 시스템을 실제 현장에 적용하여 사용함으로써 사용의 편리성과 실용성을 확인하고자 한다[3].

본 논문의 이후 구성은 다음과 같다. 2장에서는 관련 연구로 설계 요구사항, Spring MVC 패턴, MyBatis, REST API, 그리고 OAuth 2.0에 대하여 설명하며 3장은 시스템 설계로 Web Application Server와 장비 모니터링 앱으로 안드로이드와 스마트 글래스를 구분하여 설명한다. 4장에서는 설계된 시스템을 각 시스템별로 구현하고 구현된 화면을 이용하여 시스템의 구성을 설명한다. 마지막으로 5장에서는 결론을 도출하고 향후 연구 방향을 제시한다.

Ⅱ. 관련 연구

2.1. 설계 요구사항

개발한 시스템을 적용할 현재 공장은 생산 관리 시스템(MES : Manufacturing Execution System) 기반의 장비 이력 관리 시스템을 보유하고 있다. 현재는 장비의 상태를 확인하기 위해 현장을 방문하여 수기로 이상 상태 및 수리 내용을 작성하고 PC만을 이용하여 내용을 입력하기 때문에 실시간 입력이 어려워 누락이 많은 실정이다.

제안 시스템은 기존 공장에서 사용 중인 생산 관리 시스템과 연계하여 생산 관리 시스템의 장비 이상 알림과 보고 사항을 상호 전달하여야 하며 이를 위한 SMS 및 음성 인식 기술이 필요하다. 공장 내부 설비에 대한 정보들은 QR코드 인식을 통해 조회, 변경 및 삭제가 가능하여야 하며 기존 사용 중인 메신저 플랫폼과 연동하여 사용의 편의성을 높인다. 작업자의 공정 효율을 높이기 위해서 안드로이드 시스템과 스마트 글래스 시스템을 같이 사용하는 다중장치 기반 장비 모니터링이 가능해야 한다. 스마트 글래스 시스템은 QR코드 인식을 통한 장비 인식이 가능해야 하며 음성을 통해 주변 작업자에게 다양한 메시지 전송이 가능해야 한다. 개발은 웹 애플리케이션 서버와 스마트 글래스 앱 그리고 안드로이드 앱으로 구분하여 개발한다.

2.2. Spring MVC 패턴

Spring Framework는 자바 플랫폼을 위한 오픈소스 애플리케이션 프레임워크로써 동적인 웹사이트를 개발하

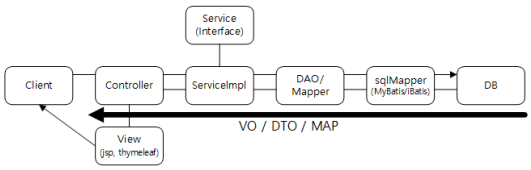


Fig. 1 The structure map of Spring MVC Pattern

기 위한 여러 가지 서비스를 제공한다[4]. MVC(Model-View-Controller) 패턴은 소프트웨어 디자인 패턴이다. 이 패턴은 사용자 인터페이스로부터 비즈니스 로직을 분리하여 애플리케이션의 시각적 요소나 그 이면에서 실행되는 비즈니스 로직을 서로 영향 없이 쉽게 고칠 수 있는 애플리케이션을 만들 수 있다. MVC 패턴에서 모델은 모든 뷰 템플릿에 렌더링할 수 있도록 데이터를 유지 관리하는 역할을 한다. View는 사용자에게 화면을 보여주기 위해서 모델의 데이터를 전달받아서 사용자가 읽을 수 있도록 표시하는 역할을 한다. Controller는 모델과 View 사이의 상호작용을 제어하는 역할을 한다. 그리고 모델을 만들고 모델을 View에 전달하는 역할을 한다. 그림 1은 Spring MVC 패턴의 구조도이다.

2.3. MyBatis

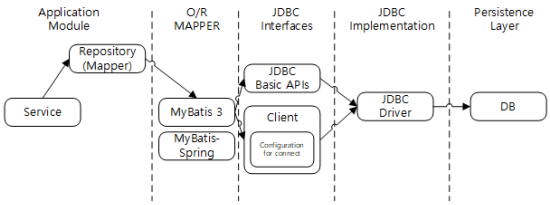


Fig. 2 The structure map of MyBatis component

MyBatis는 객체 지향 언어인 자바의 관계형 데이터베이스 프로그래밍을 쉽게 할 수 있게 도와주는 개발 프레임워크이다. JDBC(Java DataBase Connectivity)을 통해 데이터베이스에 액세스하는 작업을 캡슐화하고 일반 SQL 쿼리, 저장 프로시저 및 고급 매핑을 지원하며 모든 JDBC 코드 및 매개 변수의 중복 작업을 제거해준다. MyBatis에는 프로그램에 있는 SQL쿼리들을 한 구성 파일에 구성하여 프로그램 코드와 SQL을 분리할 수 있는 장점이 있다. 그림 2는 MyBatis의 컴포넌트 구조이다[5].

2.4. REST API

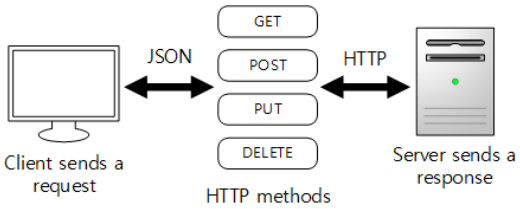


Fig. 3 The structure map of REST API

REST(REpresentational State Transfer)는 HTTP (HyperText Transfer Protocol) URI(Uniform Resource Identifier)를 통해 자원(Resource)을 명시하고, POST, GET, PUT, DELETE와 같은 HTTP Method를 통해 해당 자원에 대한 CRUD(Create, Read, Update, Delete) Operation을 적용하는 것을 의미한다. 즉, REST는 자원 기반의 구조 설계의 중심에 Resource가 있고 HTTP Method를 통해 Resource를 처리하도록 설계된 구조를 의미한다. 웹사이트의 이미지, 텍스트, DB 내용 등의 모든 자원에 고유한 ID인 HTTP URI를 부여한다. CRUD Operation의 Create는 생성 (POST), Read는 조회(GET), Update는 수정(PUT), Delete는 삭제(DELETE)를 의미한다. 그림 3은 REST API의 구성도이다[6].

2.5. OAuth 2.0

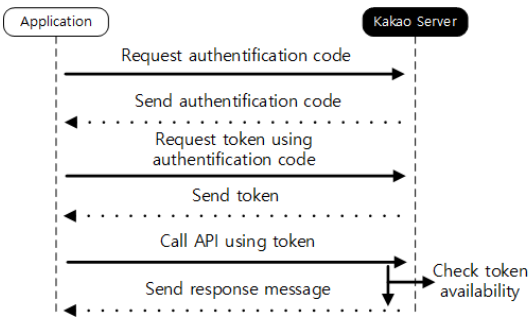


Fig. 4 Relationship diagram between application and kakao server

OAuth 2.0은 인터넷 사용자들이 비밀번호를 직접 제공하지 않고 다른 웹사이트에서 사용하는 자신의 비밀번호 정보를 대신 제공해 줄 수 있게 하는 접근 위임을 위한 개방형 표준이다. 그림 4는 애플리케이션과 카카오 시스템의 메시지 관계도이다[7][8].

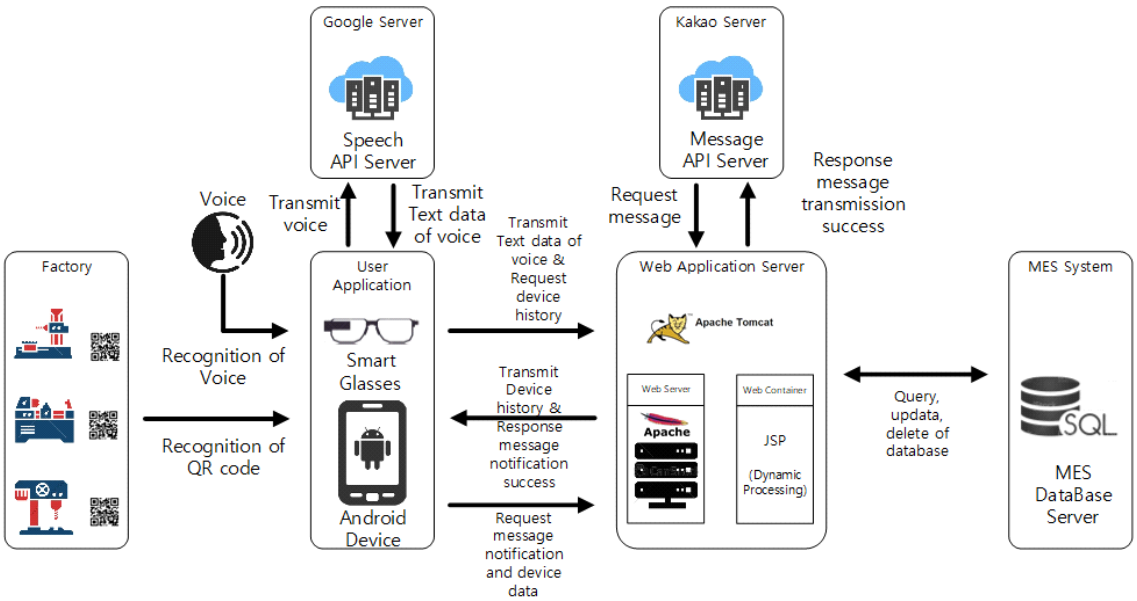


Fig. 5 The system bock diagram

Ⅲ. 시스템 설계

다중장치기반 실시간 장비 모니터링 및 이력관리 시스템의 구성 및 관계도는 그림 5와 같으며 공장의 장비는 QR코드를 이용하여 기본 정보를 제공한다. 사용자 애플리케이션으로 안드로이드 애플리케이션과 스마트 글래스 애플리케이션을 이용한다. 서비스 시나리오 및 동작 과정은 다음과 같다. 스마트 글래스 애플리케이션은 카메라로 QR코드를 수집하고 마이크를 이용하여 음성정보를 수집한다. 수집된 음성은 Google 서버로 전송하며 이것은 다시 문자로 수신한다. 수신된 음성정보는 Web Application Server로 전달된다. QR코드를 수집한 안드로이드 애플리케이션은 Web Application Server로 메시지 알림 및 장비 관련 데이터를 요청한다. Web application server는 메시지 알림 성공 응답 메시지와 장비 이력 데이터를 사용자 애플리케이션에 전달하며 Kakao API Server로 메시지 요청 및 메시지 전송 성공 응답을 확인한다. 그리고, 기존 회사의 MES 시스템을 통해 데이터베이스의 데이터를 조회, 수정, 삭제한다.

본 논문은 Web application server, 안드로이드 애플리케이션, 그리고 스마트 글래스 애플리케이션으로 나누어 설계한다.

3.1. Web Application Server

Web Application Server의 구조는 Spring MVC, MyBatis, REST API를 통합적으로 사용하는 구조로 설계하였다.

Spring MVC 구조는 사용자 인터페이스로부터 비즈니스 로직을 분리하여 애플리케이션의 시각적 요소나 그 이면에서 실행되는 비즈니스 로직을 서로 영향 없이 쉽게 고칠 수 있는 애플리케이션을 만들 수 있어서 서버의 Framework로 이용하였다. MyBatis는 프로그램에 있는 SQL 쿼리들을 한 구성 파일에 구성하여 프로그램 코드와 SQL을 분리하여 수정 용이성을 높일 수 있어서 활용하였다. 그리고 REST API를 사용해서 안드로이드 앱과 스마트 글래스에서 모두 접근할 수 있도록 설계하였다.

3.2. 장비 모니터링 앱 및 QR코드 인식

3.2.1. 안드로이드 스마트폰용 모바일 애플리케이션

MES 시스템과 연동을 위해 기존 PC Web으로 구현되어 있는 생산 관리 시스템 모니터링 프로그램 기능 중 필요한 기능들을 모바일 애플리케이션으로 개발하여 작업자들이 스마트폰으로도 장비 상태 모니터링이 가능하게 설계하였다. 그리고 장비 고유 QR코드를 부여하여 스마트폰으로 인식하면 바로 장비 이력을 조회할 수 있도록 설계하였다.

그림 6은 안드로이드 애플리케이션의 화면 흐름도이

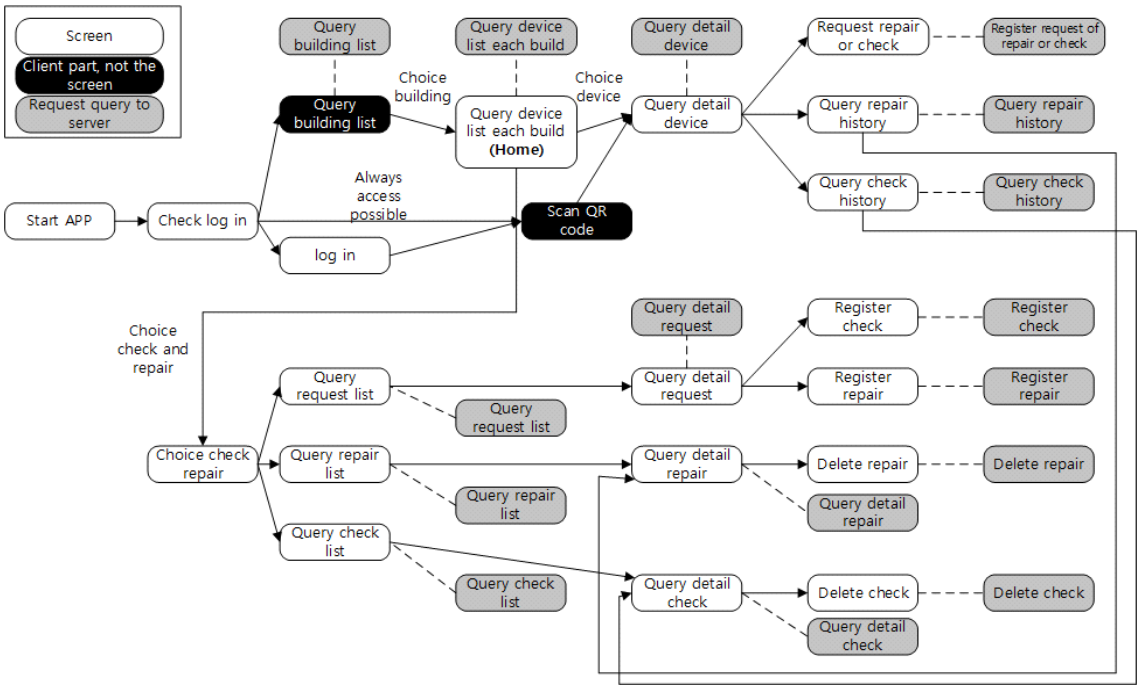


Fig. 6 Block diagram that added query request to android application screen flowchart

며 부가적으로 데이터 조회 시점을 추가하였다. 장비 조회를 위해 QR코드를 이용하며 기본적으로 설비, 점검, 수리를 위한 데이터 생성, 조회, 삭제가 가능하도록 설계하였다. 데이터베이스는 MES의 기존 데이터를 이용하여 연동할 수 있도록 설계하였다.

3.2.2. 스마트 글래스 애플리케이션

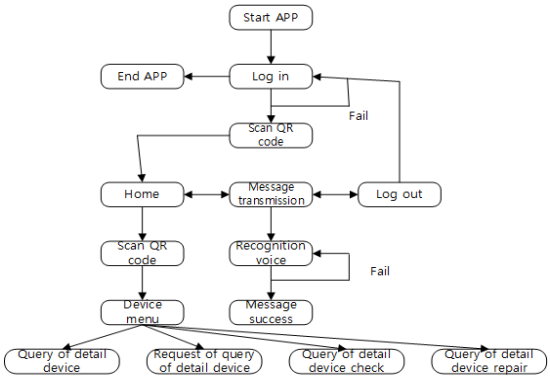


Fig. 7 The system bock diagram

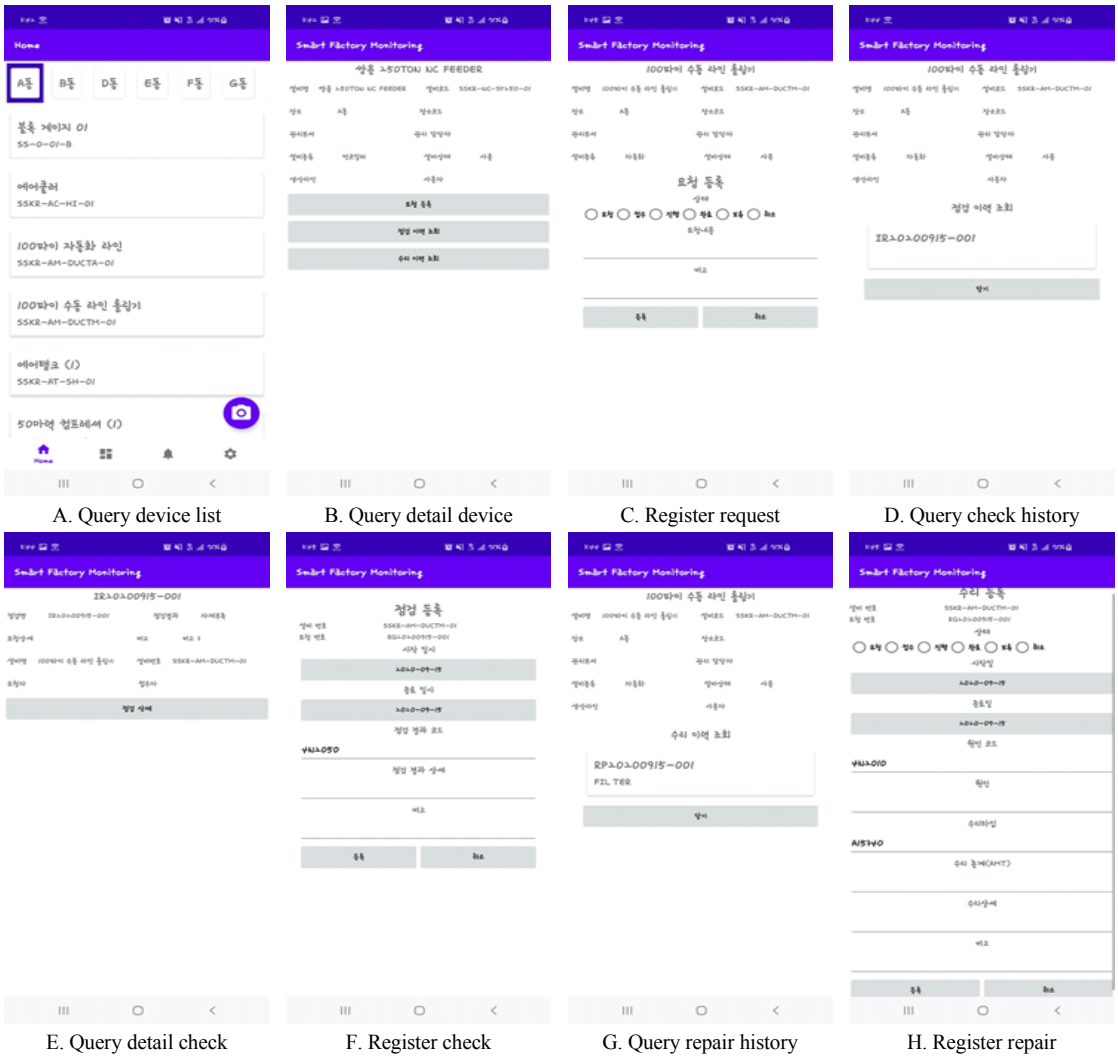
작업자가 양손을 사용 중이거나 장갑을 착용하는 등 스마트폰 조작에 제약이 있을 때도 편리하게 설비 정보를 조회하고 명령할 수 있도록 스마트 글래스 애플리케이션을 설계하였다. 스마트 글래스용 애플리케이션은 안드로이드 스마트폰용 모바일 애플리케이션 기능을 기반으로 설계하였으며, 키보드 입력이 불가능한 특성 때문에 설비 수정과 관련된 기능은 제거했다[9]. 그리고 양손을 사용하지 않더라도 QR코드를 인식하여 로그인 및 설비 조회를 할 수 있도록 하였으며, 음성 인식을 통해 원하는 상대에게 카카오톡 메시지를 보낼 수 있도록 설계하였다. 그림 7은 스마트 글래스 애플리케이션의 시스템 흐름도이다.

Ⅳ. 시스템 구현

4.1. Web application server

Spring MVP 구조를 기반으로 데이터에 맞게 각 컴포넌트들을 구현하였고, 컴포넌트에는 Controller, DAO, Mapper, Service가 포함된다.

Web application server에서 Controller는 요청이 들어온 URL 요청에 대한 응답을 처리하는 메소드이다.



A. Query device list B. Query detail device C. Register request D. Query check history

E. Query detail check F. Register check G. Query repair history H. Register repair

Fig. 8 Implemented screen shot of android application

@RequestMapping 이라는 Spring의 Annotation을 사용하여 이를 구현하였다. 요청에 대한 응답은 @ResponseBody Annotation을 정의하여 JSONArrary 객체로 반환하도록 하였다.

REST의 CRUD에 맞게 Create는 생성 (POST), Read는 조회(GET), Update는 수정(PUT), Delete는 삭제(DELETE)로 구현하였다. DAO(Data Access Object)는 데이터베이스를 사용해 데이터를 조회하거나 조작하는 기능을 전담하도록 만든 객체이다.

본 시스템에서는 생성자와 getter/setter 메소드를 정의하여 데이터베이스를 사용해 데이터를 조회하거나 조작하도록 구현하였다. Mapper 인터페이스는 MyBatis 매핑 XML에 기재된 SQL 문을 호출하기 위한 인터페이스이다.

JAVA의 Mapper의 경우 Interface이기 때문에 함수명에 대한 정의만 존재하고, SQL 쿼리문은 Mapper.xml에서 구현되었다. MyBatis Mapper는 프로그램에 있는 SQL 쿼리문을 한 구성 파일에 구성하여 프로그램 코드와 분리하여 .xml 파일에 해당 interface에 대한 SQL 쿼리문을 구현하였다.

본 시스템에서는 Mapper Interface의 메소드와 같은 이름으로 SQL 쿼리문을 구현하였고, 입력 형식은 Hashmap으로 통일되게 설정하였다. 그리고, 출력 형식은 데이터와 일치하는 DAO로 출력하였다.

Service Interface는 비즈니스 로직을 정의하고 ServiceImpl에 비즈니스 로직을 상세 구현하는 역할을 한다. 본 시스템의 ServiceImpl에서는 데이터 정제를 할 필요가 없으므로 Mapper의 메소드를 사용하는 방식으로 구현하였다.

4.2. 장비 모니터링 앱 및 QR코드 인식

4.2.1. 안드로이드 스마트폰용 모바일 애플리케이션

안드로이드 애플리케이션 개발을 위한 개발 환경으로 장비는 Galaxy S10과 S8 Plus를 사용하였고 각 장비의 OS는 Android 10과 Android 9(Pie)를 사용하였다. 각 장비의 SDK는 API 29와 28을 사용하였다.

현재 사용 중인 PC 기반의 생산 관리 시스템 모니터링 프로그램의 쿼리문을 기반으로 애플리케이션을 구현하였으며, 최초 앱을 실행하면 사원 계정으로 로그인을 한다. 로그인에 성공하면 메인 화면으로 동별 설비 목록 조회 화면과 QR코드를 인식할 수 있는 카메라 실행 플로팅 버튼이 출력된다.

플로팅 버튼을 누르면 바로 QR코드를 인식할 수 있도록 카메라가 실행되며, 이력을 조회하고 싶은 설비의 QR코드를 인식하게 되면 '설비 상세 조회 화면'이 보이게 된다. QR코드 인식을 하지 않아도 메인 화면에서 원하는 설비를 선택하면 '설비 상세 조회 화면'으로 이동하며 이 화면에서는 해당 설비에 대한 정보들을 상세하게 볼 수 있다. 그리고 '요청 등록', '점검 이력 조회', '수리 이력 조회' 버튼을 활성화한다.

'요청 등록' 버튼을 누르면 어떤 요청을 할 것인지 상태를 선택할 수 있게 되며 이에 대한 요청 내용을 입력하고 등록할 수 있다. '점검 이력 조회' 버튼을 누르면 해당 설비의 점검 이력을 조회하게 되며, 특정 점검 이력을 누르게 되면 해당 점검에 대한 상세 이력을 볼 수 있는 화면으로 전환된다. 여기서 '점검 삭제' 버튼을 누르면 해당 점검 기록이 삭제된다. '수리 이력 조회' 버튼을 누르면 해당 설비의 수리 이력을 조회하게 되며, 특정 수리 이력을 누르게 되면 해당 수리에 대한 상세 이력을 볼 수 있는 화면으로 전환된다. 여기서도 '수리 삭제' 버튼을 누르면 해당 수리 기록이 삭제된다.

메인 화면 하단의 내비게이션 바 두 번째 탭을 누르면 설비 수리 및 점검을 등록할 수 있는 목록이 출력되며, 등록하고 싶은 설비를 누르면 '수리 등록'과 '점검 등록'을 할 수 있다. 수리와 점검의 원인 코드와 수리 타입은 기업체의 데이터를 기반으로 선택할 수 있게 되며 모바일 환경에서 쉽게 설비들을 관리할 수 있도록 플랫폼을 구현하였다. 기존 PC Web으로 구현된 생산 관리 시스템 모니터링 프로그램 기능 중 필요한 기능들을 모바일 애플리케이션으로 구현하여 작업자들이 스마트폰으로도 장비 상태 점검 모니터링이 가능하게 하였다. 그리고 장비 고유 QR코드를 부여하여 스마트폰으로 인식하면 바로 장비 이력을 조회할 수 있도록 구현하였다. 그림 8은 안드로이드 스마트폰용 애플리케이션의 구동 화면이다.

4.2.2. 스마트 글래스용 애플리케이션

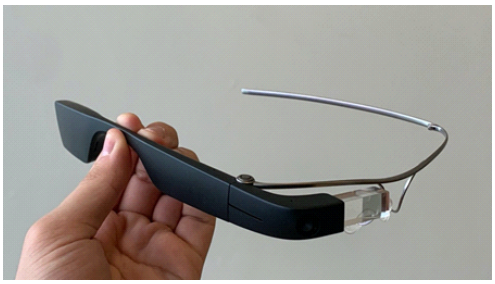
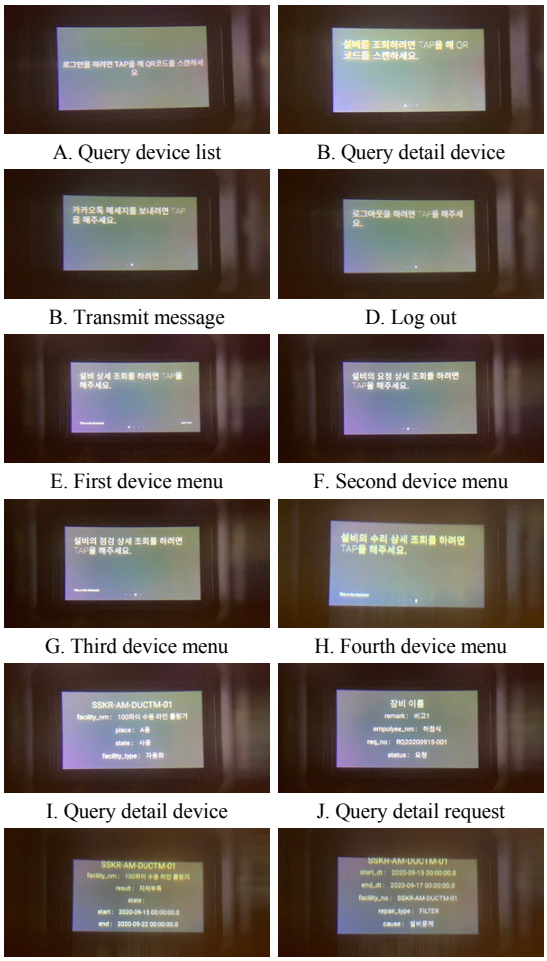


Fig. 9 Google Glass Enterprise Edition 2 used for development

스마트 글래스 애플리케이션을 개발하기 위해 구글 글래스 엔터프라이즈 에디션 2를 사용하였다. 구글 글래스 엔터프라이즈 에디션 2의 스펙으로 AP는 1.7GHz Quad-Core Qualcomm XR1, RAM은 3GB, ROM은 32GB, 카메라는 800만 화소, Display는 640 x 360, 배터리는 820mAh, OS는 Android 8.0 Oreo, Sensor는 6-Dof Gyro Sensor, 그리고 무게는 46g이다[10]. 그림 9는 실제 구현된 스마트 클래스이다.

스마트 글래스 애플리케이션은 작업자가 양손을 사용 중이거나 스마트폰 조작이 어려운 상황을 고려하였기 때문에 정보 입력보다는 정보 조회 위주로 구현하였으며 정보 입력은 음성을 인식할 수 있도록 구현하였다. 그림 10은 스마트 글래스 애플리케이션의 주요 실행 화면이다.



A. Query device list B. Query detail device
B. Transmit message D. Log out
E. First device menu F. Second device menu
G. Third device menu H. Fourth device menu
I. Query detail device J. Query detail request
K. Query detail check L. Query detail repair

Fig. 10 Implemented screen shot of Google Glass application

4.2.3. 음성 인식

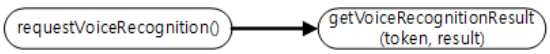


Fig. 11 Execution process of voice recognition

안드로이드 스마트폰용 애플리케이션과 스마트 글래스용 애플리케이션 모두 Google Speech Recognizer 라이브러리를 사용하였다. requestVoiceRecognition() 함수를 실행해 각 디바이스에서 음성 입력을 받을 수 있도록 구현하였다. 음성을 입력받은 후, 안정적인 동작을 위해 예외처리 코드를 이용하여 정상적으로 입력 받은 경우만 결과 함수를 실행하도록 구현하였다. 그리고 getVoiceRecognitionResult() 함수를 실행해 음성 인식 결과를 사용자도 확인하도록 화면에 출력시켰다. 이후 안드로이드 스마트폰용 애플리케이션과 같은 과정으로 서버에 메시지를 전송하도록 구현했다. 그림 11은 음성 인식 실행 과정이며 그림 12는 스마트 글래스 음성 인식 메뉴와 음성 인식 중인 화면이다.

A. Voice recognition menu B. Screen shot during voice recognition

Fig. 12 Voice recognition function of Google Glass application

Ⅴ. 성능분석

개발된 시스템은 안드로이드 앱, 스마트 글래스 앱 그리고 웹 애플리케이션 서버로 구성된다. 표1은 안드로이드 앱, 스마트 글래스 앱 그리고 웹 애플리케이션 서버에 사용된 주요 기능과 동작 상황을 확인한 것이다. 안드로이드 앱의 Android_A는 설비, 요청, 점검, 수리 관련 정보 조회, 추가, 삭제 요청 기능이고 Android_B는 로그인을 위한 QR코드 생성 기능이다. Android_C는 설비에 부착된 코드를 인식하기 위한 QR코드 인식 기능이고 Android_D는 카카오톡 로그인 관리 및 메시지 전송을 위한 토큰 관리 기능이다. 마지막으로 Android_E는 음성 인식을 통한 카카오톡 메시지 전송 기능이며 5가지 주요 기능은 모두 이상 없이 동작하였다.

스마트 글래스 앱의 Smart Glass_A는 설비, 요청, 점검, 수리 관련 정보 조회 요청 기능이고 Smart Glass_B는 스마트 글래스 로그인과 설비에 부착된 코드를 인식하기 위한 QR코드 인식 기능이다. Smart Glass_C는 음성 인식을 통한 카카오톡 메시지 전송 기능이며 3가지 기능 모두 이상 없이 동작하였다.

웹 애플리케이션 서버의 Web Application_A는 설비, 요청, 점검, 수리 관련 정보 조회, 추가, 삭제 요청에 대한 응답 기능이며 Web Application_B는 카카오 토큰 유효시간 조회, 토큰 재발급, 카카오 메시지 전송에 대한 요청의 응답 기능이다. Web Application_C는 카카오 서버로 토큰 유효시간 조회 요청과 토큰 재발급 요청 기능이며 Web Application_D는 카카오 서버로 메시지 전달

요청 기능이다. 4가지 주요 기능 모두 이상 없이 동작하였다.

Table. 1 Working status of developed functions according to category

Category	Development content	Working
Android Application	Android_A	O
	Android_B	O
	Android_C	O
	Android_D	O
	Android_F	O
Smart Glass Application	Smart Glass_A	O
	Smart Glass_B	O
	Smart Glass_C	O
Web Application Server	Web Application_A	O
	Web Application_B	O
	Web Application_C	O
	Web Application_D	O

Ⅵ. 결 론

제조업에서는 고객 주문, 제품개발 단계 이후 제품 생산에서의 품질(Q), 원가(C), 납기(D)를 중요하게 관리한다. 제조업에서의 설비 모니터링은 설비 이슈로 인한 생산 차질과 납기지연, 설비 수리 등 작업 중지로 인한 제조원가 상승, 설비상태 변경과 같은 작업조건 변화에 따른 품질 문제 발생 등을 사전에 모니터링하고 예방하기 위한 시스템이다. 휴대가 간편한 스마트폰이나 스마트 글래스를 통한 설비 모니터링 시스템은 실제 현장에서 활용도가 높다. 또한, 카카오톡 메신저 연동을 통한 알림 기능 역시 설비 모니터링 앱의 한계점을 보완하고 작업자들에게 설비 모니터링 시스템에 대한 친밀도를 높여 시스템의 활용도가 높아진다. 특히 스마트 글래스는 작업자가 양손을 사용 중이거나 장갑을 착용하는 등 스마트폰 사용에 제약이 있을 때도 사용이 가능한 특징을 가지고 있다. 본 논문은 실시간 장비 모니터링을 위해 스마트폰과 스마트 글래스를 동시에 사용 가능한 다중 장치 기반 실시간 장비 모니터링 및 이력관리 시스템을 설계 및 구현하였다. 스마트 글래스 앱을 통해서 설비 정보를 편리하게 조회하고 음성 인식을 통한 간단한 메시지를 전달하여 불필요한 스마트폰 및 데스크톱 PC 조작 시간을 줄일 수 있었다. 작업환경을 고려하여 작업자의 효율을 극대화할 수 있는 다중 인터페이스를 제공한 것이다. 구현된 시스템을 실제 공장에 적용함으로써 사용의 편리성과 실용성을 확인하였으며 다양한 시스템을 스마트 글래스 기반으로 개발할 수 있는 기반을 마련했다 할 수 있다. 향후 스마트팩토리를 위한 다양한 응용에 관한 연구가 필요하다.

ACKNOWLEDGEMENT

"This work was supported by a 2-Year Research Grant of Pusan National University"

References

[1] H. G. Kim, "A Study on the Factors Influencing on the Intention to Continuously Use a Smart Factory," *Journal of The Korean Industrial Information System Research*, vol. 25, no. 2, pp. 73-85, Apr. 2020.

[2] J. J. Oh, S. J. Choi, and J. S. Kim, "Development of Multiple Wireless Communication Controller for Smart Factory Construction," *Journal of The Korean Institute of Electronic Material Engineers*, vol. 30, no. 9, pp. 602-608, Sep. 2017.

[3] J. H. Jung and D. H. An, "Development of Remote Control System based on CNC Cutting Machine for Gradual Construction of Smart Factory Environment," *KIPS Transactions on Computer and Communication System*, vol. 8, no. 12, pp. 297-304, Aug. 2019.

[4] Spring Framework Documentation[Internet]. Available: https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/.

[5] MyBatis [Internet]. Available: https://blog.mybatis.org/.

[6] REST API? [Internet]. Available: https://www.redhat.com/ko/topics/api/what-is-a-rest-api.

[7] OAuth 2.0 [Internet]. Available: https://oauth.net/2/.

[8] kakao developers [Internet]. Available: https://developers.kakao.com/docs/latest/ko/getting-started/app.

[9] Glass Enterprise Edition 2 [Internet]. Available: https://developers.google.com/glass-enterprise/guides/design-guidelines.

[10] GLASS [Internet]. Available: https://www.google.com/glass/tech-specs/.

김동현(Dong-Hyun Kim)

2013년 부산대학교 전자전기컴퓨터공학과 (공학박사)
2013년 ~ 현재 : 부산대학교 동남권 그랜드 ICT 연구센터 연구교수
※관심분야 : 무선통신, 사물인터넷, 무선 멀티미디어 방송 시스템, LPWAN, 강화학습, SDR, 대규모 근거리 광대역 통신망

이재민(Jae-Min Lee)

2017년 부산대학교 정보컴퓨터공학부 (공학사)
2017년 3월 ~ 현재 : 부산대학교 정보융합공학과 석박사통합과정
※관심분야 : 무선 이동통신, 인공 신경망, 최적화, LPWAN, 사물인터넷, 대규모 근거리 광대역 통신망, 강화학습

김종덕(Jong-Deok Kim)

2003년 서울대학교 컴퓨터공학과 (공학박사)
2004년 ~ 현재 : 부산대학교 정보융합공학과 교수
2016년 ~ 현재 : 부산대학교 동남권그랜드 ICT 연구센터 부센터장
※관심분야 : 무선통신, 사물인터넷, 무선 멀티미디어 방송 시스템, LPWAN, 인공 신경망, 최적화, 무선랜