

근로자 주도 스마트팩토리 서비스 구성 방법

이진흥^o

^o다운정보통신(주) 기술연구소

e-mail: jhlee@daun.co.kr^o

Development of Worker-Driven Smart Factory Service

Jin-Heung Lee^o

^oDaun Information & Communication Co.

● 요약 ●

본 논문은 생산 현장에서 필요로 하는 다양한 스마트팩토리 서비스를 현장 근로자가 직접 기획, 설계, 구현 및 적용 가능한 서비스 플랫폼을 제안한다. 이를 위하여 오픈 하드웨어 개발 도구 등을 활용한 IoT 기반 제조데이터 수집과 이를 활용하여 서비스 화면을 구성할 수 있는 개발도구를 설계하고 구현하였으며, 구현된 프로그램으로부터 제조데이터 기반의 다양한 현장 서비스를 근로자가 직접 만들고 배포할 수 있다.

키워드: 스마트팩토리(smart factory), 근로자 주도(worker-driven), 서비스플랫폼(service platform)

I. Introduction

최근 전략적인 스마트팩토리 보급과는 달리 중소형 제조 생산업체들은 생산공정에서 사용되는 수많은 제조데이터들이 생산품의 품질과 매우 밀접한 연관성이 있다는 것을 인지하면서도 제조데이터의 통합관리와 생산 근로자들의 불편함으로 인하여 많은 기업들이 이러한 변화를 꺼리고 있다.

반면 글로벌 기업들은 산업데이터플랫폼을 구축하고, 기업들이 서로 선호하는 데이터 구조와 공통의 거버넌스 규칙 하에서 데이터의 연결 및 교환을 가능하게 하는 가상환경을 구축하여 새로운 데이터 주도 서비스 및 혁신적 비즈니스 프로세스를 창출하는 새로운 비즈니스 생태계를 주도하고 있다[1]. 많은 선진 기업들이 이미 제조현장에서 수집, 분석되는 데이터를 기반으로 생산성을 제고하고 공급망을 효율화하기 위해 노력 중이다.

본 논문에서는 이러한 제조데이터를 생산현장에서 다양한 효율성을 높이고자 노력하는 근로자들이 직접 서비스를 디자인하고, 필요한 서비스를 만들어서 동료들과 공유할 수 있는 시스템을 구현하였다.

그러나 최근의 스마트팩토리 시스템은 이러한 공장 시스템 환경에 적용하는 방법으로 제조업의 비즈니스 목표들을 다양한 IT 기술을 활용하여 향상시킬 수 있도록 구성하고, 기존 공장들이 가지고 있는 설비 자동화를 넘어 스마트 기술 기반의 보다 높은 성과를 달성하는 스마트 공장 시스템을 제공한다[3]. 이러한 스마트 기능을 이루기 위한 과정을 Observe, Orient, Decide, Act의 4단계로 구성할 수 있으며, 각 과정을 위해 요구되는 기술을 스마트 기술로 제공할 수 있다.

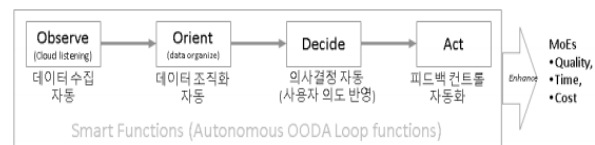


Fig. 1. Smart Functions

Observe 단계에서는 사전 정의된 외부 및 내부 시스템의 데이터 및 신호를 일정 주기 또는 실시간으로 자동 수집하여 서비스 시스템 내에 저장할 수 있어야 한다. Orient 단계에서는 수집한 데이터로부터 비즈니스 목표(비용, 시간, 품질)를 위하여 분석 가능한 형태로 데이터를 구조화하는 작업을 수행하고 설정된 분석 엔진에 따라 달라질 수 있어야 한다. Decide 단계에서는 사전에 정의된 비즈니스 목표를 반영하기 위한 최적의 행동을 결정할 수 있다. 마지막으로 Act 단계에서는 앞선 단계에서 결정된 최적 행동을 실제 시스템에 반영한다.

II. Related works

2.1 스마트팩토리 서비스 및 아키텍처

스마트팩토리 서비스는 오늘날 빅데이터 시스템과 연계하여 다양한 산업에 적용하는 추세이다. 그러나 중소제조 환경에서는 기존의 운용 중인 제조설비 데이터를 수집하고 이로부터 의사결정을 판단하는데 어려움이 많이 있다.

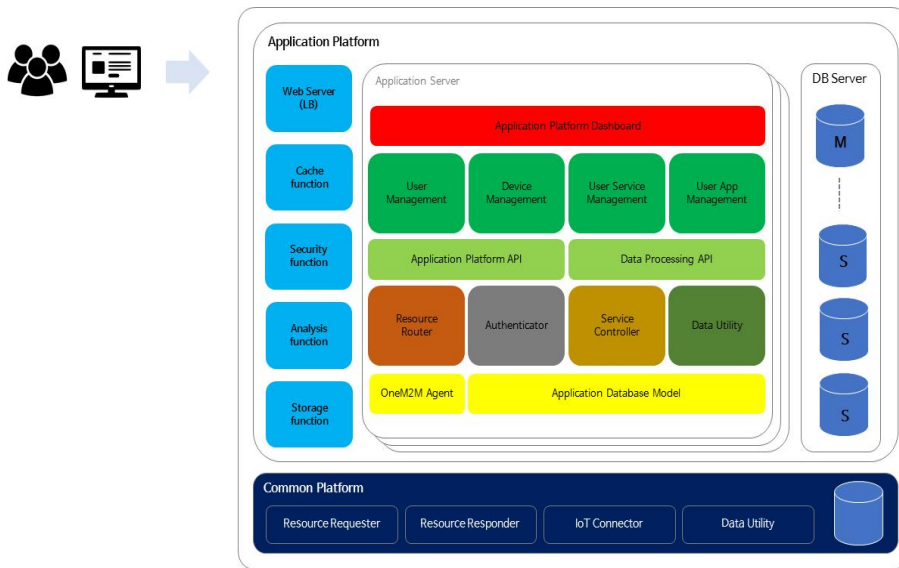


Fig. 3. System Platform Architecture

2.2 마이크로그리드용 공통플랫폼 서비스 개발도구

마이크로그리드는 소규모 독립형 전력망 내에서 전력 정보를 공유하여 에너지 효율을 최적화하는 지능형 전력망 기술로서 풍력, 태양광 등 신재생에너지원과 에너지 저장장치 등을 에너지 관리시스템으로 제어하여 안정성과 효율성을 갖춘 소규모 전력망 기술이다. 마이크로그리드 내의 발전, 판매, 시장 등 다양한 영역의 시스템의 통합운영을 지원하고 장치 및 시스템 간 상호 운용성 제공과 Third-party 사업자에게 개방형 API 기능을 통해 전력 정보를 제공 등 비즈니스 모델을 지원하는 공통플랫폼을 개발하고 있다.

마이크로그리드용 개발도구는 공통플랫폼에서 제공하는 Open API를 이용하여 마이크로그리드 운영 및 관리 그리고 모니터링 서비스를 제공하는 웹 기반 어플리케이션 개발환경과 실행환경을 제공한다. 이를 위해 개발도구는 크게 웹브라우저 상에서 구동되는 부분과 실행엔진에서 구동되는 부분으로 나뉘어져 구성되어 있다[2].

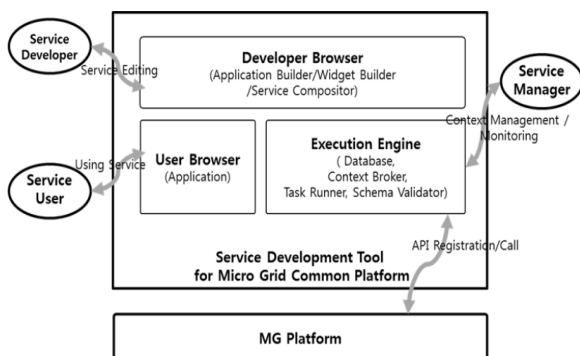


Fig. 2. Service Development Tool for Micro-Grid Pplatform

III. The System Implementation

제안된 시스템은 사물인터넷 기반 데이터 수집 플랫폼에서 서비스

를 개발하고 적용하기 위한 응용플랫폼, IoT 디바이스 사용을 위한 API, 현장 근로자가 직접 서비스를 기획하고 쉽게 개발을 도와주는 개발 플랫폼을 구성하고 직접 구현하였다.

3.1 응용플랫폼 구성

구성된 시스템에서 응용플랫폼은 IoT 센서 데이터를 수집하는 공통플랫폼에 리소스를 등록하거나 등록된 리소스에 대한 사용자 처리 요청을 수행한다. 이때 공통플랫폼은 oneM2M 표준을 기반으로 개발된 모비우스 플랫폼으로 구성하였으며 표준에 따라 사물인터넷 통신, IoT 리소스 처리 및 리소스 관리 기능을 구현하였다. 모비우스 기반의 공통플랫폼은 시스템 내부 데이터 저장 역할을 수행하며, 이미지 뷰어 및 동영상 스트리밍과 같은 파일 기반 사용자 서비스 관련 기능을 확장하였다.

Application Server 파트는 사물인터넷 기반의 제조데이터를 수집하는 현장의 근로자가 스마트팩토리 서비스를 위하여 실제 서버 프로그램을 활용하기 위한 서버를 동작하는 기능으로서, 구성된 Device 및 User App Management 등의 기능으로 응용플랫폼 내부에서 공통플랫폼에 저장된 사용자디바이스서비스 등의 관리 서비스를 수행한다. 또한, Data Processing API와 같은 API를 제공하여 공통플랫폼의 데이터와 응용플랫폼에서 구현된 서비스와 연결하고 응용플랫폼 내부 데이터에 대한 요청 및 내부함수 동작 요청 등을 제공한다.

3.2 Open API 구현

구성된 시스템에서 사용하는 응용플랫폼 API 개발을 위하여 우선 공통 응답코드를 API 요청 성공, API 요청 자체 실패, API 요청 성공했지만 내부 요청 처리 실패로 구분하여 정의하였다. 또, 현장 근로자가 직접 서비스를 제작하고 공유하기 위하여 필요한 기본 API들을 아래와 같이 분류하여 정의한다.

- 로그인 API: 세션 생성, 갱신, 삭제
- 사용자 API: 사용자 등록, 조회, 수정, 삭제
- 디바이스 API: 디바이스 등록, 디바이스 조회, 디바이스 데이터 조회, 수정, 삭제
- 사용자 애플리케이션 API: 애플리케이션 등록, 조회, 수정

200 REQUEST SUCCESS	
서버에서 알 수 없는 문제가 발생함	
body	{ message : "<성공 메시지>" data : "<요청 결과 데이터>" }
400 BAD REQUEST	
해당 API 호출에 대한 파라미터가 잘못되었음	
body	{ statusCode : 400, error : 'Bad Request', message : "<파라미터 에러 메시지>" }
401 UNAUTHORIZED	
해당 API 호출에 대한 권한이 없음	
body	{ statusCode : 401, error : 'Unauthorized', message : "<권한 에러 메시지>" }
403 FORBIDDEN	
해당 API 호출이 금지됨	
body	{ statusCode : 403, error : 'Forbidden', message : "<접근금지 에러 메시지>" }
404 NOT FOUND	
해당 API 의 결과값이 없음	
body	{ statusCode : 404, error : 'Not found', message : "<호출 결과 메시지>" }
500 INTERNAL SERVER ERROR	
서버에서 알 수 없는 문제가 발생함	
body	{ statusCode : 500, error : 'Internal Server Error', message : "<서버 에러 메시지>" }
501 NOT IMPLEMENTED	
해당 API 리소스가 서버에서 구현되어 있지 않음	
body	{ statusCode : 501, error : 'Not Implemented', message : "이 리소스는 사용되지 않습니다..!" }

각각의 API들은 요청/응답 구조로 설계하여 해당 API의 파라미터에 따라 리턴되는 응답 메시지로 해당 정보를 전달한다. 그림4는 디바이스 데이터 조회에 대하여 응답 메시지 흐름도를 나타낸다. 제안 시스템은 서비스 개발에 따른 API 응답코드를 다음 표와 같이 정의하였다.

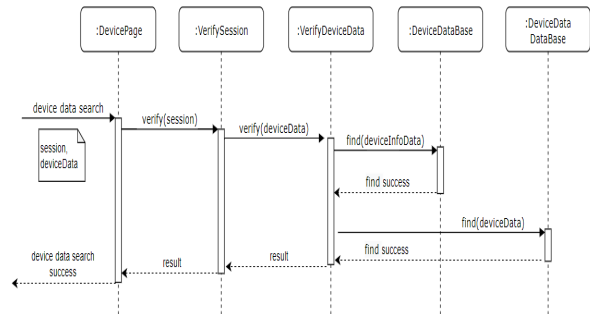


Fig. 4. Device Data Search

3.3 서비스 개발도구 개발

설치된 제조데이터 수집 장치와 API를 이용하여 근로자가 쉽게 서비스 앱 화면을 구성하고 공동 작업자와 공유하여 사용할 수 있는 플랫폼을 제공한다.

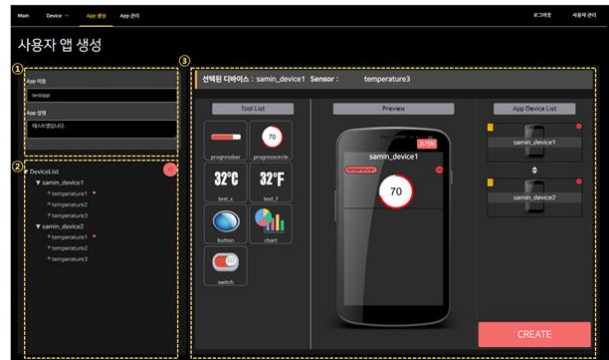


Fig. 5. Development Tools

현장 근로자는 공장 내에 설치된 제조데이터 수집 센서를 연결하고 연결된 센서 데이터에 대한 어플리케이션 화면을 생성한다. 이를 위하여 근로자는 ①생성할 어플리케이션의 기본정보(예를들어, App 이름, App 세부설명내용)를 입력하고 ②생산설비에 설치된 센서데이터 수집 장치를 목록에서 확인하고 사용할 센서 데이터와 연결한다. 그리고 ③연결된 센서 데이터의 App 화면 배치를 통하여 센서 데이터를 연결하고 제어할 수 있는 앱을 제작한다. 제작된 화면은 미리보기를 통하여 확인하고, 최종 확인 완료된 App 실행 파일을 어플리케이션 등록 틀에 저장한다.

3.4 개발 시스템 연동 테스트

개발 시스템의 통합 테스트를 위하여 우리는 우레탄 가공공장을 대상으로 제조데이터를 연결하고, 연결된 제조데이터를 통합 모니터링 하는 시스템을 그림4와 같이 구성하였다. 구성된 화면을 근로자가 직접 담당 근로자가 직접 관리해야하는 제조데이터를 확인하는 앱 화면을 구성하였다.

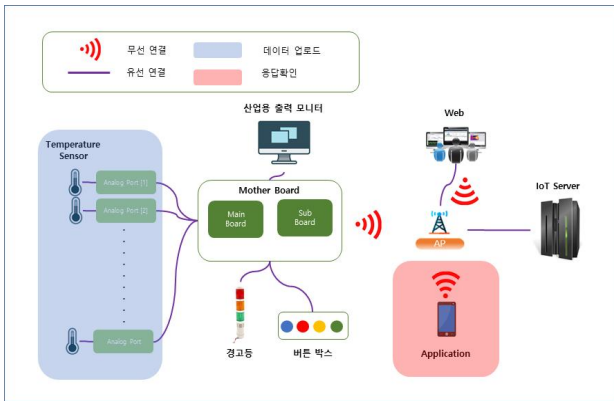


Fig. 6. Test Factory Diagram

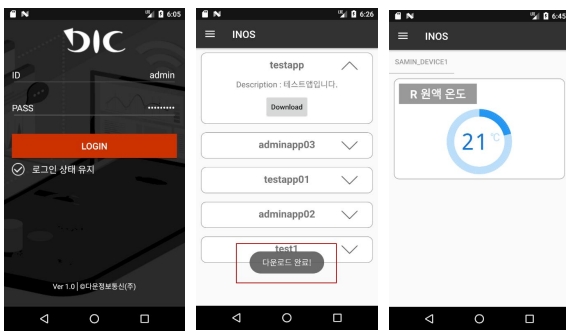


Fig. 7. App Screens



Fig. 8. Monitoring Screen

IV. Conclusions

본 논문은 제조 기업들이 현재 사용중인 노후화된 센서 설비를 기반으로 스마트팩토리 네트워크를 구성하고, 구성된 네트워크로부터 수집되는 제조데이터를 근로자가 쉽게 확인, 제어할 수 있도록 해당 서비스를 직접 개발 가능한 시스템을 구성하였다. 위 시스템 구성을 위하여 모비우스 기반의 제조데이터 수집 플랫폼과 이를 사용하기 위한 응용플랫폼을 새로 구성하고, 구성된 플랫폼 내에 근로자가 쉽게 사용 가능한 개발플랫폼을 포함하였다. 또한 포함된 개발플랫폼의 기능들을 연결하는 API를 설계, 제공함으로써 향후 다양한 응용 서비스를 제작할 수 있다.

ACKNOWLEDGEMENT

본 연구는 중소벤처기업부의 국가융복합단지 연계 지역기업 사용화 R&D 사업(P0009977, 스마트팩토리 네트워크 최적화를 위한 인터페이스 계층 분석 및 취약성 점검 시스템 개발)의 지원을 받아 수행된 논문이다.

REFERENCES

- [1] Working Group 2 Digital Industrial Platforms, Digitising European Industry, 2017.
- [2] Hyun-Hee Lee and Dae-Sang You "Service Development Tool for Micro Grid Common Platform" Journal of the Korean Institute of Information and Communication Engineering, Vol. 19, No. 6, pp. 1455-1461, 2015.
- [3] Hee-Je Lee and Joong-Yoon Lee, "Smart Service System-based Architecture Design of Smart Factory", Journal of KOSSE, Vol. 13, No. 2, pp. 57-64, 2017.