

Worker-Driven Service Development Tool for Smart Factory

Jin-Heung Lee*

*CTO, Company of the Daun Information & Communication, Busan, Korea

[Abstract]

Recently, many companies are interested in smart factory services. Because various smart factory services are provided by the combination of mobile devices, cloud computing, and IoT services. However, many workers turn away from these systems because most of them are not implemented from the worker's point of view. To solve this, we implemented a development tool that allows field workers to produce their own services so that workers can easily create smart factory services. Manufacturing data is collected in real time from sensors which are connected to manufacturing facilities and stored within smart factory platforms. Implemented development tools can produce services such as monitoring, processing, analysis, and control of manufacturing data in drag-and-drop. The implemented system is effective for small manufacturing companies because of their environment: making various services quickly according to the company's purpose. In addition, it is assumed that this also will help workers' improve operation skills on running smart factories and fostering smart factory capable personnel.

▶ **Key words:** Worker-Driven Service, Manufacturing data, Smart Factory Platform, Service Development tool, Open component

[요 약]

최근 모바일, 클라우드, 그리고 사물인터넷의 융합으로 다양한 스마트팩토리 서비스가 제공되고, 많은 기업에서도 관심을 가지고 있다. 그러나 대부분의 시스템은 근로자 관점에서 구현되지 않았기 때문에 근로자로부터 외면 받고 있다. 이에 본 논문은 스마트공장 서비스를 수요자들이 정의하여 사용할 수 있도록 서비스 제작을 현장 근로자가 직접할 수 있는 개발도구를 구현하였다. 서비스에 사용되는 제조데이터는 제조설비와 연결된 센서로부터 실시간으로 수집하여 스마트팩토리 플랫폼 내에 저장된다. 그리고 플랫폼에 저장된 제조데이터로부터 설비 모니터링, 공정상태분석, 설비 제어 등 다양한 스마트 공장 서비스를 근로자가 직접 드래그앤드롭 방식으로 매우 쉽게 만들 수 있다. 구현된 시스템은 특히 소규모 제조 기업에서 기업의 특정 목적에 맞게 수시로 서비스를 변경해야하는 환경에서 더욱더 큰 효과를 낼 것으로 예상된다. 또한, 현장 근로자의 스마트팩토리 운용 및 활용 능력 향상은 물론 중소기업의 스마트팩토리 인재 양성에 많은 도움이 될 것으로 기대된다.

▶ **주제어:** 근로자 주도 서비스, 제조데이터, 스마트팩토리 플랫폼, 서비스 개발도구, 오픈 컴포넌트

-
- First Author: Jin-Heung Lee, Corresponding Author: Jin-Heung Lee
 - Jin-Heung Lee (leejinheung@gmail.com), Company of the Daun Information & Communication
 - Received: 2020. 05. 19, Revised: 2020. 07. 24, Accepted: 2020. 07. 24.

I. Introduction

4차 산업혁명에 대한 관심이 높아지면서 제조영역에 대한 적용 시스템도 ‘스마트 매뉴팩처링’, ‘스마트팩토리’ 등 수많은 시스템과 적용 대상의 종류와 의미들도 다양하다. 제조업의 스마트화는 아직 구체화된 실체가 부족한 상황에서 스마트팩토리에 대한 정보와 구축에 대한 가이드가 정확하지 않다. 또한 스마트팩토리 구축은 회사의 도입 목적과 규모에 적합하게 수행되어야 하지만 기존 사례들은 독일 등 선진 제조국가의 적용 시스템을 그대로 도입함으로써 다양한 문제가 발생하고 있다. 또한, 국내 기업의 스마트팩토리에 대한 정확한 이해 부족과 초기 구축비용의 부담 등으로 정부의 2022년까지 스마트공장 3만개 구축 목표는 현장에서 비현실적인 목표로 인지되고 있다[1].

또한, 정부 주도의 전략적인 스마트팩토리 보급과 달리 중소 제조기업들은 생산공정에서 사용되는 다양한 제조데이터들이 생산품의 품질과 매우 밀접한 연관성이 있다는 것을 인지하면서도 제조데이터의 통합관리의 어려움과 생산 근로자들의 여러 불편한 문제로 많은 기업들이 오늘의 변화를 꺼리고 있다[2]. 반면, 4차 산업을 적극적으로 받아들이는 글로벌 기업들은 산업데이터 플랫폼을 구축하고 기업들이 서로 선호하는 데이터 구조와 공통의 거버넌스 규칙 하에서 데이터의 연결 및 교환을 가능하게 하는 가상 환경을 구축하여 새로운 데이터 주도 서비스 및 혁신적 비즈니스 프로세스를 창출하는 새로운 생태계를 주도하고 있다[3]. 그 외 많은 기업들이 이미 제조현장에서 발생하는 다양한 제조데이터를 수집, 저장하고, 분석된 데이터를 기반으로 생산성을 높이고 여러 공장 자동화 시스템에서 효율성을 높이기 위해 노력 중이다.

이에 본 연구에서는 생산 현장에서 발생하는 제조데이터를 사물인터넷 플랫폼에 실시간으로 저장하고, 저장된 제조데이터를 생산현장에서 직접 제조서비스를 기획, 디자인하여 근로현장에서 필요로 하는 서비스를 그때그때 간편하게 제작하여 사용하고 이를 작업자들과 공유할 수 있는 시스템을 구현한다. 2장에서는 스마트 서비스 관점에서 구현에 직접적으로 활용할 수 있는 스마트팩토리 아키텍처와 마이크로 그리드용 공통플랫폼 기반의 어플리케이션 개발 및 배포를 위한 PaaS(Platform as a Service) 형태의 서비스 개발도구에 관한 문헌연구를 수행하였으며, 3장에서는 사물인터넷 기반의 제조데이터 수집 플랫폼을 구성하고, 수집된 제조데이터를 현장 근로자가 직접 서비스를 설계하고 구현할 수 있는 Open API 기반의 스마트팩토리 서비스 저작도구를 개발하였다. 결론에서는 본 연구

를 통하여 스마트팩토리 관련 기술의 학문 및 산업적 영향을 기술하고, 본 성과가 관련 산업에 줄 수 있는 전략적인 방향성을 제공하였다.

II. Preliminaries

1. Related works

1.1 Service based smart factory architecture

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

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

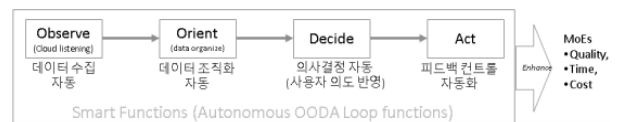


Fig. 1. Smart Functions

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.

최근 IEC(International Electrical Committee)에서는 독일의 RAMI4.0을 IEC PAS 63088 규격으로 채택하였고, 그림2와 같이 계층구조에서 상호호환성이 중요함을 강조하고 있다[5]. 스마트공장은 기존의 산업설비에 ICT를 융합함으로써 공장 내·외부의 다양한 사물 및 서비스와 연결

이 가능하게 되었다. 이에 따라 제조사, 고객, 서비스 제공 업체의 요구사항을 만족시킬 수 있는 상호호환성 기술을 필요로 한다. 스마트공장의 근본적인 목적은 가상으로 표현하고 연결한 기술 객체 간의 협력과 협업을 쉽게 하는 것이다. 이러한 맥락에서 조직에 가치 있는 기술 객체는 물리적인 유형의 객체만을 의미할 뿐만 아니라 아이디어, 저장소 및 소프트웨어와 같은 무형의 객체도 포함한다. 이를 위하여 전체 생애 주기 동안 기술적인 객체에 대한 디지털 기술 규칙을 만들고 기술 객체와 기술 객체의 개발, 생산 및 사용에서부터 처분에 이르기까지 모든 측면을 표현해야 하며 이때 상호호환성은 필수적이다.

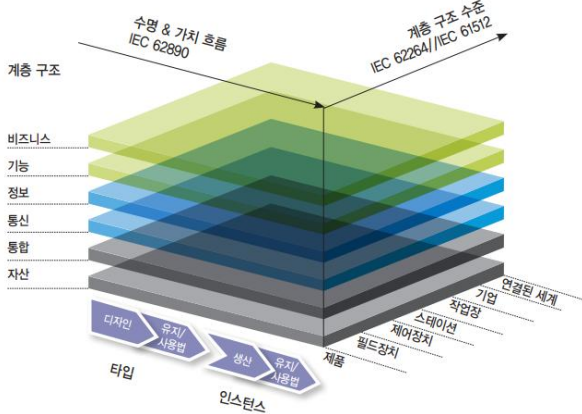


Fig. 2. Reference Architecture Model Industry 4.0(RAMI4.0)

1.2 Service development tool

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 상호운용성 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[6].

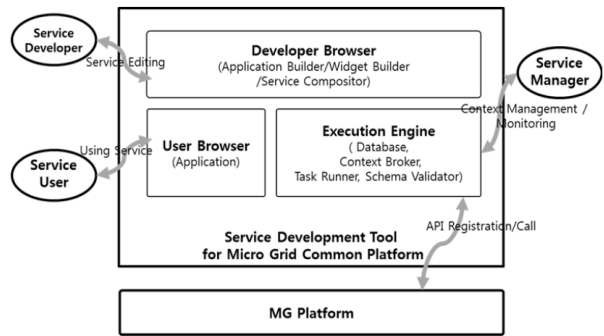


Fig. 3. Service Development Tool for Micro-Grid Platform

마이크로그리드용 애플리케이션 개발도구는 응용서비스 개발을 위하여 개발자 및 사용자 브라우저와 실행엔진으로 구성되며 Open API 관리, 애플리케이션을 제작, 실행, 저장 등의 관리, 그리고 애플리케이션의 라이프사이클 관리와 모니터링 기능을 수행한다.

III. The Proposed Development Tool

본 연구에서 제안하는 개발도구는 사물인터넷 기반 제조 데이터 수집 플랫폼에서 생산현장에서 필요로 하는 스마트 공장 서비스를 현장에서 직접 개발하고 적용가능한 스마트팩토리 서비스플랫폼과 현장에 설치되어 제조데이터를 수집하는 디바이스 및 이를 사용하기 위한 API를 개발한다. 그리고, 수집된 제조데이터를 현장 근로자가 직접 서비스를 기획하고 제작할 수 있는 개발도구를 설계, 제작하였다.

1. System configuration information

1.1 proposed smart factory platform

제안 시스템은 모바일 기반 공통플랫폼과 스마트팩토리 응용 플랫폼, 제조설비에 연결된 센서로부터 제조데이터를 수집하는 하드웨어, 그리고 구성된 시스템을 이용하여 스마트팩토리 서비스를 개발하는 인터페이스로 구성된다.

제안된 시스템에서 응용플랫폼은 IoT 센서 데이터를 수집하는 공통플랫폼에 리소스를 등록하거나 등록된 리소스에 대한 사용자 처리 요청을 수행한다. 이때 공통플랫폼은 one2M2M 표준을 기반으로 개발된 모바일 플랫폼을 사용하였으며, 표준에 따라 사물인터넷 통신, IoT 리소스 처리 및 관리 기능을 구현하였다. 모바일 기반의 공통플랫폼은 시스템 내부 데이터 저장 역할을 수행하며, 이미지 뷰어 및 동영상 스트리밍과 같은 파일 기반 사용자 서비스 등의 기능을 확장하였다.

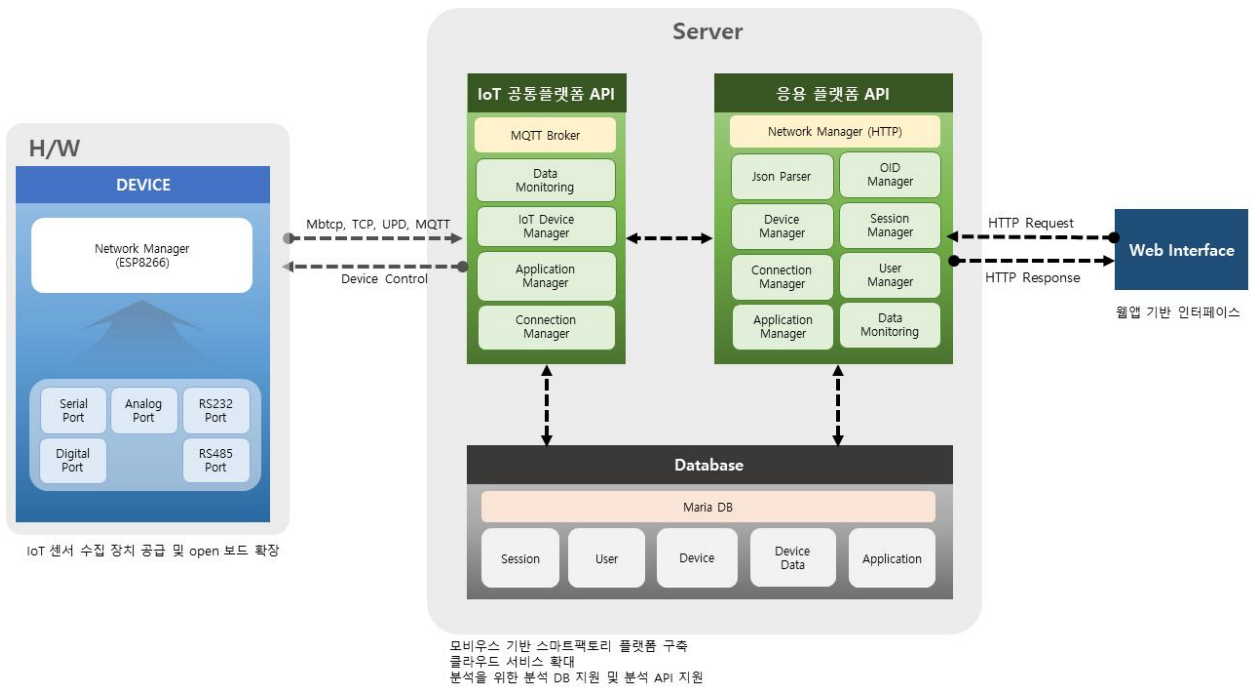


Fig. 4. Proposed system component

구현된 시스템은 사물인터넷 기반의 제조데이터를 수집하는 디바이스로부터 제조데이터를 서버에 저장하고, 현장의 근로자가 저장된 제조데이터를 활용하는 서비스 프로그램을 제작, 등록, 공유, 이용하는 기능을 제공하는 시스템으로, 센서연결장치, 사용자 인터페이스 등을 이용하여 응용플랫폼에서 공통플랫폼에 저장된 데이터를 이용하는 서비스를 수행한다.

1.2 design of openAPI

구현된 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

Table 1. System message

200 REQUEST SUCCESS	
리소스 생성에 대한 성공	
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..!" }

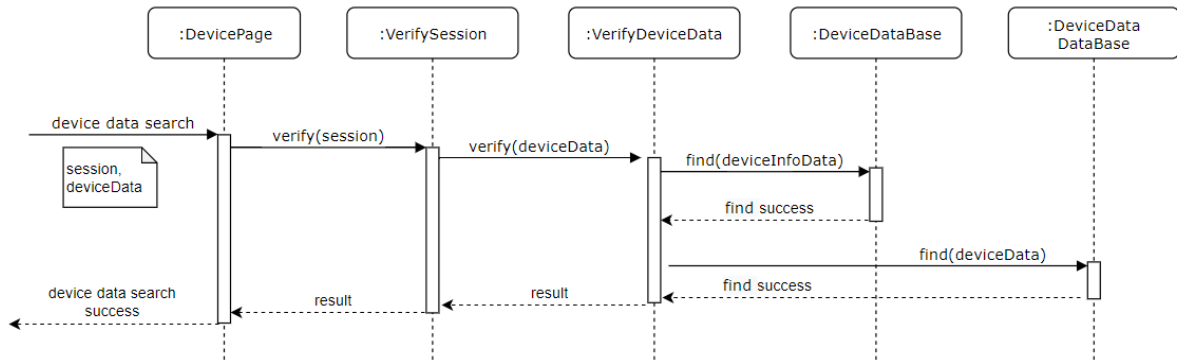


Fig. 5. Device data search

각각의 API들은 요청/응답 구조로 설계하여 해당 API의 파라미터에 따라 리턴 되는 응답 메시지로 해당 정보를 전달한다. 그림5는 디바이스 데이터 조회에 대하여 응답 메시지 흐름도를 나타낸다. 제안 시스템은 서비스 개발에 따른 API 응답코드를 다음 표1과 같이 정의하였다.

1.3 user interface

본 논문에서 제안된 스마트팩토리 서비스 개발도구의 사용자 인터페이스는 다음과 같은 구조로 주요 기능들이 구현되었다. 사용자는 그림6과 같이 디바이스 관리 기능에서 제조현장의 기계들과 연결된 제조데이터 수집 장치들을 그룹화하고 디바이스들로부터 수집되는 제조데이터를 활용할 수 있는 기능이 제공된다. 또한, 애플리케이션 관리 기능을 통하여 저장된 제조데이터를 이용한 서비스를 제작하기 위하여 시각화를 위한 컴포넌트들을 이용할 수 있고, 현장 근로자가 직접 컴포넌트를 수정, 제작하여 적용할 수 있다. 이렇게 제작된 서비스들은 통합관리 화면의 대시보드로 구성하여 공장 내 전체 제조데이터 수집 현황과 생산설비의 다양한 정보들을 관리할 수 있다.

2. Implement the smart factory service program

2.1 device management

구현된 개발도구는 제조설비와 연결된 제조데이터 수집 장치로부터 측정된 센서 데이터를 실시간으로 전달, 저장, 등록하고, 등록된 제조데이터를 서비스 목적에 따라 센서를 그룹화하고 측정된 데이터를 여러 시각화 컴포넌트에 의해 다양하게 표현된다.

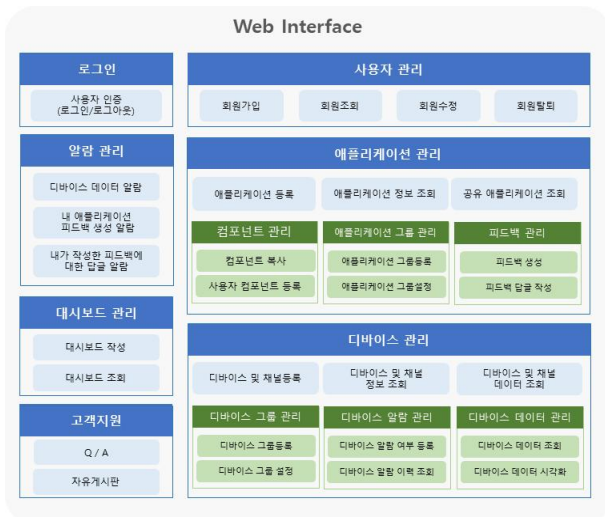


Fig. 6. User interface

Fig. 7. Device registration

디바이스 등록은 관리자가 생산설비에 센서를 설치하고 제조데이터를 저장, 사용할 수 있게 연결하는 기능이다. 그림7과 같이 디바이스 식별을 위한 OID(Object ID) 등 디바이스 기본정보를 입력하고, 타입별로 센서를 등록, 관리 한다. 디바이스 상세 관리 화면에서 디바이스 이름 등 등록된 정보를 수정하고 삭제할 수 있으며, 그림8과 같이

■ 그래프 정보 입력 : 선택한 센서에서 주기적으로 수집되는 제조데이터를 시각화된 그래프로 나타내기 위한 정보를 입력한다.

■ 그래프 모양 선택 : 기본 제공 컴포넌트와 사용자 정의 컴포넌트로부터 해당 그래프를 표현하거나, 새로운 컴포넌트를 제작하여 적용할 수 있다.

2.3 component management

제안된 개발도구에서 사용되는 컴포넌트는 구글 차트를 이용하여 기본 제공 컴포넌트를 구현하였다. 그리고 기본 컴포넌트로부터 사용자가 다양한 수정을 통하여 사용자 정의 컴포넌트를 제작하여 사용할 수 있다.

2.4 service scenario using proposed tool

본 연구에서 구현한 제조데이터 수집 플랫폼 및 스마트팩토리 개발 도구는 생산설비에서 생성되는 다양한 제조데이터를 저장하고, 현장 근로자가 필요한 응용서비스를 직접 제작할 수 있는 서비스를 제공한다. 서비스 형태는 근로자가 소유한 모바일 앱 형태로 모바일 폰과 컴퓨터에 동일하게 표현될 수 있다. 스마트공장 현장의 비전문적인 근로자들이 본 연구에서 구현된 서비스 개발도구를 이용하여 스마트공장 서비스를 제작하고, 수집된 제조데이터를 다양하게 현장에 맞는 서비스를 기획하고, 쉽게 제작하여 쓸 수 있다. 그리고 스마트공장 내에서 수집되는 제조데이터를 안전하게 저장, 식별, 관리하고, 허가된 사용자(또는 서비스 앱)에게만 데이터가 전달할 수 있도록 그림10과 같이 제조데이터 전송 절차를 적용하였다.

3. complexity

구현된 개발도구의 효율성 검증을 위하여 우리는 표2에서 나타난 세가지 공장 외에 기계부품공장 등 다섯 곳의 현장을 대상으로 현장 부서에서 필요한 앱을 직접 구현하고, 구현된 앱의 복잡도를 측정하기 위해 SourceMonitor를 이용하였다. SourceMonitor는 구현된 소스 코드의 양이나 모듈의 복잡성을 측정하는 도구로 코드의 복잡도, 분기율, 구문수, 함수호출수 등 소스 코드의 품질 지표를 계산한다. 컴포넌트의 복잡도는 소프트웨어 컴포넌트 안에서 조건들 사이의 내부 관계를 측정한 것으로, 시스템에서의 잠재적인 스트레스 지점을 식별한다. 아래 표와 같이 구현된 앱은 현장의 근로자가 자신의 업무와 관련 있는 서비스를 본 논문에서 구현한 개발 도구를 이용하여 제작한 것으로 실적관리, 재고현황 모니터링 등의 업무를 시각화 하였다. 표에 나타난 앱의 평균 복잡도는 각각 3.14, 3.41, 그리고 3.56로 계산되어 프로그램이 정한 정상 범위 2.0~4.5

Table 2. App service

분류	앱 화면	구현 서비스
의류 공장		공장 근로자의 업무시간 및 개인별 작업량 수집 및 모니터링하여 직원별 실적관리, 작업시간관리, 팀별 계획대비 달성률을 모니터링하는 서비스
금형 공장		자재관리를 위하여 실시간 재고현황 및 설비별 생산량 측정 서비스
식품 가공 공장		숙련도가 낮은 초급 작업자들의 실수를 최소화하기 위하여 실시간 설비 가동 정보 모니터링 및 개인별 생산량 모니터링 서비스

사이의 복잡도를 가지는 것으로 계산되었다. 표3에서 평균복잡도 등 소스코드 품질을 위한 주요 측정지표들의 값을 타내고 있다.

앞의 결과와 같이 본 논문에서 구현된 개발도구를 사용하여 근로자가 간단하게 관련 스마트공장 서비스를 구현할 수 있고, 구현된 소스코드의 가독성이 높아 향후 수정, 분석 등이 쉬운 형식으로 구현되는 것을 확인하였다. 그러나 branches 수가 평균 55.56으로 정상범위 밖에 있어, 이 부분에 대한 추가 연구를 통하여 수정, 보완되어야 한다.

Table 3. complexity analysis

분류	평균라인수	평균복잡도	최대복잡도	평균깊이
1	1029	3.03	7	1.57
2	958	3.14	7	1.97
3	1334	3.41	6	1.79
4	1394	3.31	7	1.7
5	1698	3.56	7	1.83
평균	1282	3.29	6.8	1.77

IV. Conclusions

본 논문은 스마트공장 서비스를 생산현장 근로자들이 설계, 제작하고 이를 관련 근로자들과 공유하여 활용하기 위한 스마트팩토리 서비스 개발도구를 구현하였다. 구현된

개발도구는 크게 제조데이터를 수집하는 디바이스를 관리하고, 애플리케이션 제작 및 공장 설비의 통합 관리와 같은 서비스 개발을 도와주고, 제작된 앱 프로그램을 서로 공유하고 사용하기 위한 기능들을 포함한다. 이 개발 도구를 통하여 제조 기업들은 현장에서 가동중인 생산설비로부터 제조데이터 수집 플랫폼을 구성하고, 플랫폼에 저장된 제조데이터를 근로자가 쉽게 확인, 이용하여 원하는 서비스를 직접 제작할 수 있다.

전체 시스템은 모비우스 기반의 제조데이터 수집 플랫폼과 저장된 다양한 제조데이터를 응용서비스에서 쉽게 사용하기 위한 응용플랫폼을 구성하고, 구성된 스마트팩토리 플랫폼을 이용하여 근로자가 쉽게 서비스를 제작할 수 있는 개발도구를 포함한다. 또한 서비스 개발 및 다양한 오픈 하드웨어를 사용하기 위해 플랫폼 접근 API를 제공하여 향후 여러 응용 서비스를 위하여 현장에서 추가, 확장하여 사용할 수 있게 구현하였다.

본 연구 결과를 실제 공장에서 사용하는 스마트공장 서비스를 구현하여 근로자의 이용 편의성 및 코드 복잡도를 계산하였다. 향후 중소 제조기업의대표 산업군을 대상으로 서비스 실증을 거쳐 근로자 주도 서비스 개발이 얼마나 효율적인지 검토하고, 근로자가 쉽게 제작할 수 있는지 사용성 평가를 통한 검증이 필요하다. 또한 제작된 서비스의 소스코드 품질 지표와 효과적인 관리를 수행을 위한 소스코드 품질의 정적분석 및 검증 방안에 대하여 논의될 필요가 있다.

ACKNOWLEDGEMENT

This work was supported by the Technology development Program(P0009977) funded by the Ministry of SMEs and Startups(MSS, Korea)

REFERENCES

- [1] J.P. Park, "Analysis on Success Cases of Smart Factory in Korea: Leveraging from Large, Medium, and Small Size Enterprises", *Journal of Digital Convergence*, Vol. 15, No. 5, pp.107-115, 2017. DOI: 10.14400/JDC.2017.15.5.107
- [2] T.W. Chang, S.I. Sung, J.C. Lee "Survey Analysis on Small and Medium-Sized Suppliers of Smart Factory and Improvement Plan for Them", *Entrun Journal of Information Technology*, Vol. 17, No. 1, pp.77-88, 2019.
- [3] Working Group 2 Digital Industrial Platforms, *Digitising European Industry*, 2017.
- [4] Heeje Lee, Joongyoon Lee, "Smart Service System-based Architecture Design", *Journal of KOSSE*, Vol. 13, No. 2, pp.57-64, 2017.
- [5] IEC, IEC PAS 63088:2017 "Smart manufacturing - Reference architecture model industry 4.0(RAMI4.0)", 2017.
- [6] H.H. Lee and D.S. You, "Service Development Tool for Micro Grid Common Platform", *Journal of the Korea Institute of Information and Communication Engineering*, Vol. 19, No. 6, pp. 1455-1461, 2015.
- [7] Haluk Demirkan, Charlie Bess, Jim Spohrer, Ammar Rayes, Don Allen, Yassi Moghaddam, "Innovations with Smart Service Systems: Analytics, Big Data, Cognitive Assistance, and the Internet of Everything", *Communications of the Association for Information Systems*, Vol. 37, pp.733-752, 2015. ISSN:1529-3181
- [8] H.R. Chong and K.H. Bae, M.K. Lee, H.M. Kwon, S.H. Hong, "Quality Strategy for Building a Smart Factory in the Fourth Industrial Revolution", *Journal of Korean Society Quality Management*, Vol. 48, No. 1, pp. 87-105, 2020. DOI: 10.7469/JKSQM.2020.48.1.87
- [9] Mobius, <http://developers.iotocean.org/archives/module/mobius>
- [10] Google chart tools, <https://developers.google.com/chart/>

Authors



Jin-Heung Lee received the B.S. degree in Information and Communication Engineering from Dongseo University in 1998 and M.S and Ph.D. degree in Information Security from Pukyong National University in 2000 and 2007

respectively. Dr. Lee is a research associate professor at Dongseo University from 2001 to 2015. Since 2015 he has been working for Daun Information and Communication Co. He is interested in network security, automatic control system management, IoT platform, smart factory, and others.