



최종보고서

비접촉 전류 센서를 이용한 무인 공용 세탁기 관리 시스템 (LA.MA.)

2018. 06. 25

한국외국어대학교

정보통신공학과

6팀(ITECTOR)



최종보고서: 비접촉 전류 센서를 이용한 무인 공용 세탁기 관리 시스템

문서 정보

구 분	소 속	성 명	날 짜	서 명
작성자	한국외국어대학교	신교선	2018. 06. 21	
	한국외국어대학교	김세용	2018. 06. 21	
	한국외국어대학교	김준영	2018. 06. 21	
	한국외국어대학교	김진우	2018. 06. 21	
	한국외국어대학교	김유진	2018. 06. 21	
	한국외국어대학교	유정현	2018. 06. 21	
검토자	한국외국어대학교	신교선	2018. 06. 24	
	한국외국어대학교	김세용	2018. 06. 24	
	한국외국어대학교	김준영	2018. 06. 24	
	한국외국어대학교	김진우	2018. 06. 24	
	한국외국어대학교	김유진	2018. 06. 24	
	한국외국어대학교	유정현	2018. 06. 24	
사용자				
승인자	한국외국어대학교	홍진표	2018. 06.	

머리말

본 문서는 무인 공용 세탁기 관리에 대한 서비스와 시스템 구성을 설명하고 있다. 비 접촉 전류센서와 각종 하드웨어를 이용하여 User(사용자)는 원격으로 세탁기의 사용 여부를 확인 할 수 있다. 관리자는 세탁기별 사용횟수, 일/월 별 수익 현황 등의 정보를 통해 자신 소유의 공용 세탁시설을 관리할 수 있다.



목 차

목 차	4
표 목 차	6
그 림 목 차	7
1. 개요	8
1.1 목적	8
1.2 관련 문서	9
1.3 용어 및 약어	9
2. 서비스 소개	10
2.1 서비스 개요	10
2.2 서비스 기능	11
2.2.1 시나리오	11
2.2.2 Application	12
3. 시스템 구성도	14
3.1 전체 시스템 구성도	14
3.2 세부 시스템 구성도	15
3.3 소프트웨어	15
3.3.1 Python	15
3.3.2 Android Studio	16
3.3.3 Amazon EC2	16
3.3.4 Amazon RDS	17
3.3.5 MySQL	17
3.3.6 Amazon Elastic Beanstalk	18
3.4 하드웨어	18
3.4.1 Arduino	18
3.4.2 Non-Invasive AC Current Sensor – 100A(비접촉 전류 센서)	19
3.4.3 Raspberry pi	19
3.4.4 Smart Phone	20
4. 네트워크 구성과 통신 방식	21
4.1 LM SensorNode – LM Gateway	21
4.2 LM Gateway – LM logic Server	21
4.3 LM Logic Server – LM database – LM Web Server	22
4.4 LM Application – LM Web Server	22

최종보고서: 비접촉 전류 센서를 이용한 무인 공용 세탁기 관리 시스템

5. 시스템 상세 설계	23
5.1 LM Application.....	23
5.2 LM SensorNode.....	24
5.3 LM Server(Web & Logic) + LM Database	26
6. 적용방안 및 기대효과 (SWOT 분석).....	27
7. 향후 계획.....	28
8. 프로젝트 세부추진계획 및 세부일정	29
9. 팀원 담당업무.....	29
10. Source Code.....	29

표 목 차

[Table 1] 관련 문서.....	9
[Table 2] 약어 및 용어.....	9
[Table 3] 프로젝트 추진 일정.....	30
[Table 4] 팀원 담당업무	29

그 림 목 차

[Figure 1] 시장 동향 관련 기사1	10
[Figure 2] 시장 동향 관련 기사2	10
[Figure 3] 어플리케이션 서비스	12
[Figure 4] 전체 시스템 구성도	14
[Figure 5] 세부 시스템 구성도	15
[Figure 6] Python 3.6	15
[Figure 7] Android Studio	16
[Figure 8]Amazon EC2	16
[Figure 9] Amazon RDS	17
[Figure 10] MySQL Database	17
[Figure 11] Amazon Elastic Beanstalk	18
[Figure 12] Arduino Uno	18
[Figure 13] 비접촉 AC 전류센서	19
[Figure 14] Raspberry pi3.0	19
[Figure 15] Galaxy S7 & iPhone 6s	20
[Figure 16] LM SensorNode – LM Gateway	21
[Figure 17] LM Gateway – LM logic Server	21
[Figure 18] LM Logic Server – LM database – LM Web Server	22
[Figure 19] LM Application – LM Web Server	22
[Figure 20] LM Application	23
[Figure 21] 회로 모식도	24
[Figure 22] 회로 구성	24
[Figure 23] 비접촉 전류센서를 연결한 아두이노 회로도	24
[Figure 24] 비접촉 전류센서 실제 측정 사진	24
[Figure 25] 시작 후 10분 세탁기 전류 측정 그래프	25
[Figure 26] 종료 10분 전 세탁기 전류 측정 그래프	25
[Figure 27] 세탁기 전류 측정 그래프	25
[Figure 28] 시작 알고리즘	26
[Figure 29] 종료 알고리즘	26
[Figure 30] LM Server(Web & Logic) + LM Database	26

1. 개요

본 장에서는 비접촉 전류 센서를 이용한 세탁기 가동 유무를 확인하는 어플리케이션인 'LAMA'에 대한 목적, 참고자료 용어 및 약어를 제시한다.

1.1 목적

2018년 5월 16일부터 2018년 6월 25일까지 진행된 해당 프로젝트가 실현하고자 하는 바는 현재 무인으로 운영되어지는 공용 세탁기 및 건조기를 관리하고 이용함에 있어서 어플리케이션 하나로 편리함과 효율성을 제공하는 것이다.

무인 공용 세탁시설은 1인 가구가 증가하는 시대의 흐름에 맞춘 획기적인 시스템이다. 단 돈 3,000원을 넣고 일정 시간을 기다리면 세탁이 된다. 장마나 한파 등과 같은 기상악화로 인한 빨래의 어려움을 해결하는 한편, '빨리빨리'를 좋아하는 현대인들의 특성을 정확히 공략한 아이디어다. 하지만, 이러한 효율적인 세탁 시스템에도 불구하고, 군대에서, 또는 기숙사에서, 자취를 하면서 등 한 번씩 공용 세탁기 이용 경험이 있는 팀원들은 다소 편리하지 않은 상황을 느낀 적이 있었다.

가장 불편하다고 생각했던 점은 세탁기의 가용 여부를 모른다는 것이었다. 이로 인해, 이전 사람의 세탁기 사용시간이 끝날 때까지 기다려야 하거나, 사용 시간이 끝났는데도 세탁물을 가지러 오지 않는 사람 때문에 세탁기를 이용할 수 없는 불편한 상황이 발생된다. 팀원들은 이러한 점을 개선하여 사람들이 좀 더 편리하게 무인 세탁기를 이용할 수 있도록 하고 싶었다.

공용 세탁기 관리자의 입장에서 불편한 점이 있지 않을까 생각해보았다. 무인으로 돌아가는 세탁기들을 실시간으로 관리하는 것이 어려울 수 있다. 이용자들이 겪을 수 있는 불편함을 사전에 방지하고, 이에 대해 빠르게 조치를 취할 수 있다면, 세탁기 이용의 순환율을 높여 효과적으로 세탁시설을 운영할 수 있을 것이다.

즉, 매 시간 세탁기를 관찰하지 않아도 되는 무인 환경의 장점은 살리고, 세탁기의 문제를 빠르게 해결하는 것이 어려운 단점을 극복하는 관리시스템을 제공하는 것이 본 프로젝트의 목적이다.



1.2 관련 문서

출판사	문서 제목
이지스퍼블리싱	Do It! 안드로이드 앱 프로그래밍
엔써북	무한상상 DIY 아두이노로 만드는 사물인터넷

[Table 1] 관련 문서

1.3 용어 및 약어

용어 및 약어	풀이	비고
AWS	Amazon Web Service – 클라우드에서는 컴퓨팅 파워, 스토리지 옵션, 네트워킹 및 데이터 베이스 같은 다양한 인프라 서비스를 제공	
Irms	전류의 실효값 (Root Mean Square)	
LAMA	본 프로젝트의 어플리케이션으로써 Laundry Mate의 약어	

[Table 2] 약어 및 용어

2. 서비스 소개

본 장에서는 LAMA 서비스에 대한 개요와 더불어 서비스 기능을 기술한다.

2.1 서비스 개요

셀프빨래방 문화 빠르게 자리잡다

창업은 경쟁자 많아 신중하게

2018년 04월 24일(화) 16:10 1010호 [영천시민뉴스] [강원북서](#)



© 영천시민뉴스

셀프빨래방이 지역에서도 성업 중에 있다. 현재 3곳에서 운영 중인 셀프빨래방은 고객이 빨래감을 가지고 들어가면 500원 동전을 교환해 이불을 비롯한 각종 크고 작은 빨래를 세탁기에 넣고 정도에 따라 (40분) 기다린 뒤 마치면 바로 옆 건조대를 옮겨 건조를 시작하는데, 건조대에서도 동전이 필요하다. 빨래에서 건조까지 약 1만 원의 비용(양에 따라 차이)이 들어간다고 한다. 지난 주말 완산동 빨래방을 찾은 고객들은 “이제 두 번 이용했는데, 보기와는 달리 빨래가

한파에 코인빨래방 ‘복적복적’

#. 30대 직장인 이병기 씨는 올 겨울 통패딩을 사복처럼 입고 다니고 있다. 하지만 이 씨는 늘 신경 쓰이는 것이 있다. 항상 입고 다니다 보니 회식때 옷에 땀 고기냄새와 얼룩이 고민이다. 이 씨는 “냄새와 얼룩 때문에 다른 옷을 입으려 했지만 연일 날씨가 너무 추워 계속 패딩만 고집하게 된다”며 “내일 또 입어야 하는데 빨래방에 갈수도 없다”고 했다.

연일 계속되는 한파로 통패딩을 입는 빈도가 늘어났다. 하지만 매일 세탁기에 빨거나 세탁소에 맡길 수도 없는 노릇이다. 이에 한파 속 셀프 빨래방은 문전성시를 이루고 있다.



한파가 연일 계속되면서 세탁기 통파사고 등으로 인해 코인빨래방을 찾는 시민들이 늘고 있다. [영남뉴스]

[Figure 1] 시장 동향 관련 기사 1

[Figure 2] 시장 동향 관련 기사 2

과거에는 자취생 밀집지역에서나 볼 수 있었던 셀프 빨래방이 최근 1인 가구 시대에 맞춰 그 점포수가 눈에 띄게 증가하고 있다. 그러나 증가하는 추세임에도 불구하고, 사용자가 빨래를 하러 갔을 때 이미 세탁기를 사용중인 사람들 때문에 이용할 수 있는 세탁기가 없는 경우도 종종 발생한다. 이는 빨래가 어려운 기상 환경을 마주하는 겨울철과 여름철에 더욱 빈번하다. 한파로 인해 가정집의 세탁기가 동파되는 경우, 그리고 여름철 장마로 인해 많은 사람들이 건조기와 세탁기를 동시에 이용하는 경우 등이다.

본 어플리케이션은 무인 공용 세탁시설을 이용할 시 발생할 수 있는 이러한 불편한 점을 개선하는 서비스이다. 사용 가능한 세탁기가 없어서 이용하지 못하는 경우에 있어서, 사용자가 이 어플리케이션을 사용하게 되면 가던 길을 되돌아 오거나 세탁이 끝나기만을 기다리는 헛수고를 줄일 수 있다. 이것은 기존의 셀프 빨래방 시스템에 한층 더 효율적이고 편리한 방법을 제공할 것이다.

2.2 서비스 기능

2.2.1 시나리오

시나리오1:

시험이 3일 남은 K군. 그 동안 밀린 빨래를 하기 위해 자취방 근처에 있는 무인세탁소로 향했다. K군은 빨랫감을 넣고 세탁이 될 때까지 50분을 기다려야 했다. 도서관으로 돌아온 K군은 앉아서 시험 공부를 시작했다. 50분이 지났지만 공부에 집중을 하다가 K군은 빨래를 꺼내야 하는 것을 잊어버렸다.

이 때, K군이 본 어플리케이션의 사용자 모드에서 해당 세탁기의 Push 알림 받기를 설정한다면, 세탁기 가동이 끝나기 5분 전에 가동 종료 알림을 받을 수 있다. 그럼 K군은 해당 알림으로 빨래 꺼내는 것을 잊어버리지 않을 것이고, 다른 사람들이 K군으로 인해 다음 세탁기 사용을 기다려야 하는 상황도 해결될 것이다.

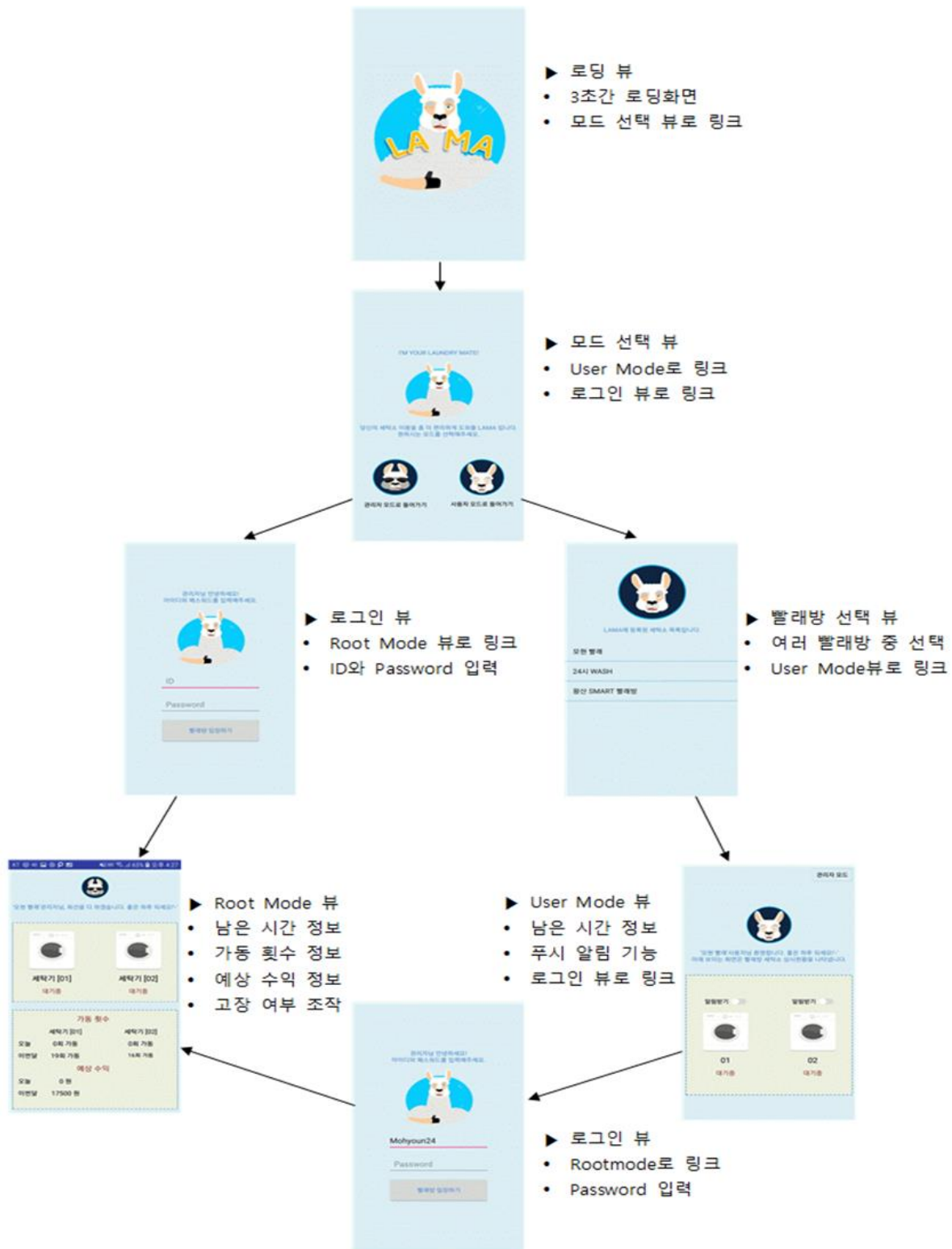
시나리오2:

매일 같은 유니폼을 입고 출근해야 하는 J양. 그녀는 오늘 점심에 커피를 먹다가 유니폼에 그만 쏟고 말았다. 아, 내일도 이 유니폼을 입어야만 한다. 하지만 오늘 빨래를 해도 내일은 마르지 않는 상황. 그녀는 저녁에 일을 마친 후 유니폼을 들고 바로 자취방 근처의 무인세탁소로 향했다. 하필 사람이 많이 붐비고 있었다. 그래서 세탁기를 바로 사용할 수 없었고, 그녀는 근처 커피숍에 가서 다른 사람들의 세탁기 사용이 끝나는 것을 하염없이 기다리기로 했다.

이 때 그녀가 본 어플리케이션을 이용한다면, 무인 공용 세탁시설의 세탁기 가동 현황을 실시간으로 확인할 수 있다. 그러면 J양은 하염없이 기다리지 않아도 된다. 언제쯤 어떤 세탁기를 사용해야 할지 예측할 수 있기 때문이다. 또한, 해당 세탁소가 아닌 다른 무인 세탁소의 세탁기를 사용할 수도 있다. 그러면 J양은 더 넓은 선택권을 가지고서 세탁기 종료까지 기다리는 시간을 절약 할 수 있다.

최종보고서: 비접촉 전류 센서를 이용한 무인 공용 세탁기 관리 시스템

2.2.2 Application



[Figure 3] 어플리케이션 서비스

최종보고서: 비접촉 전류 센서를 이용한 무인 공용 세탁기 관리 시스템

모드 선택 뷰는 User mode(사용자 모드)와 Root mode(관리자 모드) 중 하나를 선택하는 뷰이다. User mode(사용자 모드)를 선택하면, 빨래방 선택 뷰로 전환되고, 해당 뷰에서 원하는 빨래방을 선택할 수 있다. Root mode(관리자 모드)를 선택하면 로그인 뷰로 전환되고, ID와 비밀번호를 입력하여, 그 값이 일치하면 Root mode뷰로 전환된다.

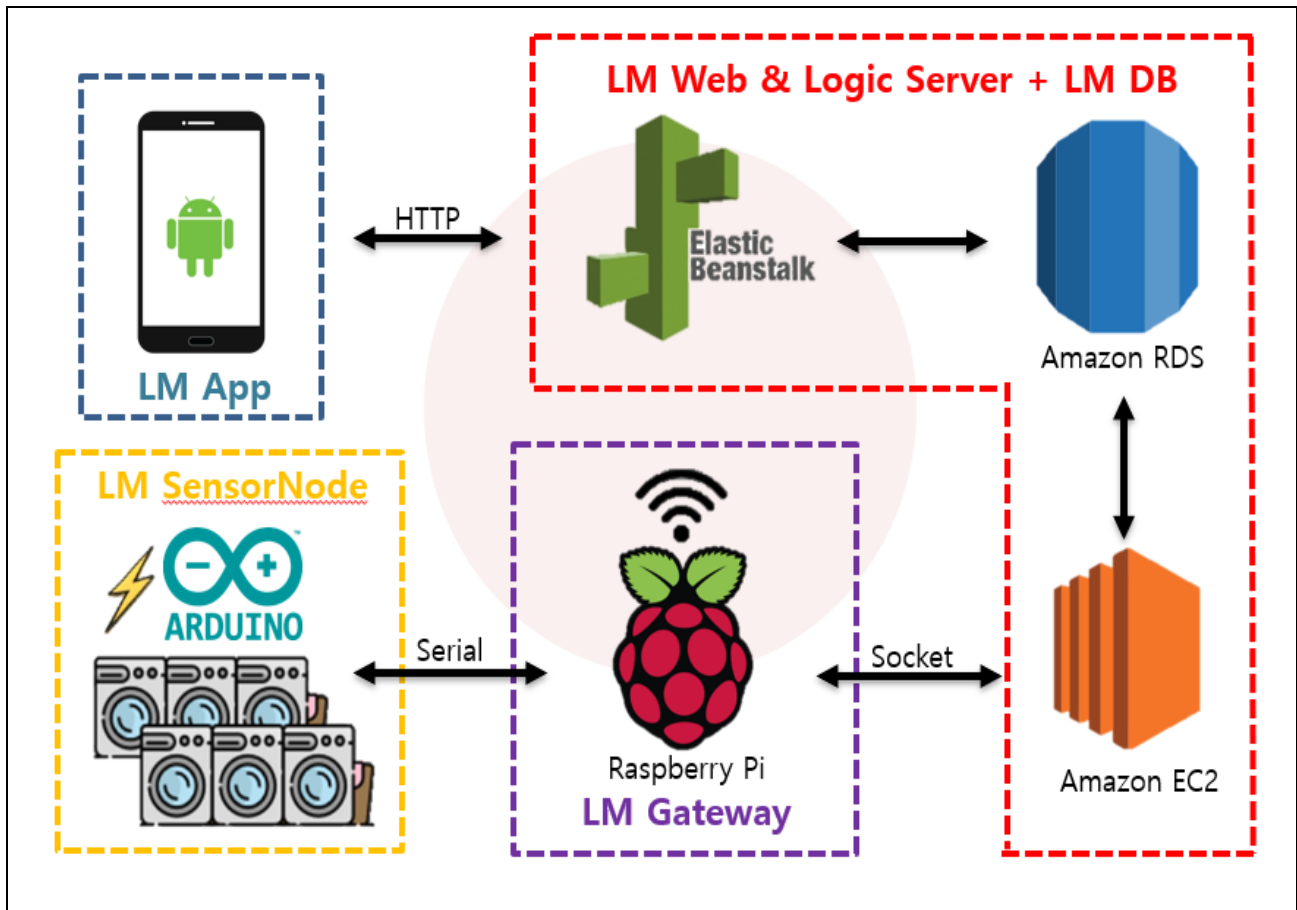
User mode(사용자 모드)뷰에서는 선택된 세탁소의 세탁기별 남은 시간과 세탁기별 가동현황을 확인 할 수 있다. 원하는 경우, '알림받기' 스위치를 통하여, 가동 중인 세탁기의 종료시간이 5분 남았음을 알려주는 푸쉬 알림 서비스를 이용할 수 있다.

Root mode(관리자 모드)뷰에서는 해당 세탁소의 세탁기별 남은 시간과 가동현황을 확인 할 수 있고, 일별 및 월별 세탁기 가동횟수와 예상 수익을 볼 수 있다. 또한 점검이 필요한 세탁기를 점검상태로 설정함으로써 User mode(사용자 모드)의 사용자에게 세탁기가 점검중임을 알릴 수 있다.

3. 시스템 구성도

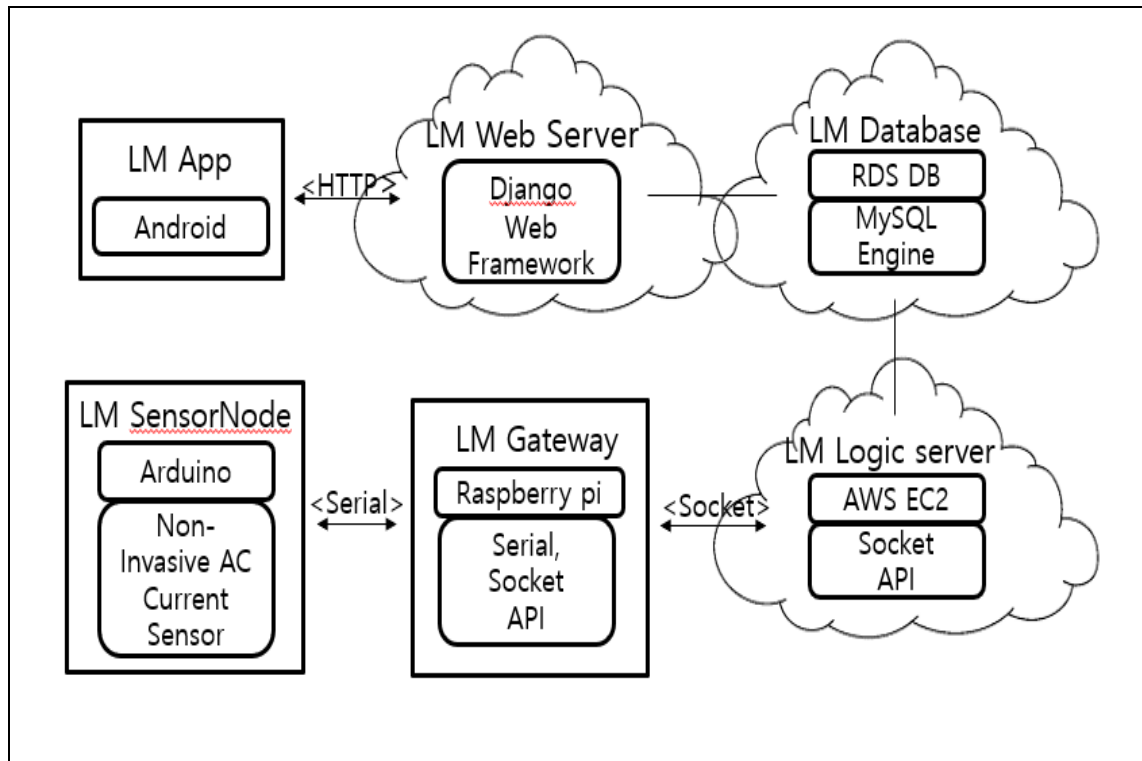
본 장에서는 전체 시스템 구성, 세부 시스템 구성 그리고 시스템을 구성하는 개별 소프트웨어, 하드웨어를 기술한다.

3.1 전체 시스템 구성도



[Figure 4] 전체 시스템 구성도

3.2 세부 시스템 구성도



[Figure 5] 세부 시스템 구성도

3.3 소프트웨어

3.3.1 Python



[Figure 6] Python 3.6

객체 지향형 스크립트 언어의 하나로서 문법 구조가 간단 명확하고, 언어 코어가 상대적으로 작은 대신 많은 시스템 호출과 풍부한 라이브러리 모듈을 제공한다. 또한 시스템 언어와 스크립트 언어의 중간 위치 성격을 띄어서 C, C++을 사용하여 새로운 모듈을 쉽게 만들 수 있는 등 다른 언어와 잘 융합되는 특징이 있다.

3.3.2 Android Studio



[Figure 7] Android Studio

Android Studio는 Android 앱 개발을 위한 공식 통합 개발 환경(IDE)이며, IntelliJ IDEA를 기반으로 합니다. IntelliJ의 강력한 코드 편집기와 개발자 도구 외에도, Android Studio는 Android 앱을 빌드할 때 생산성을 높여주는 다양한 기능을 제공한다.

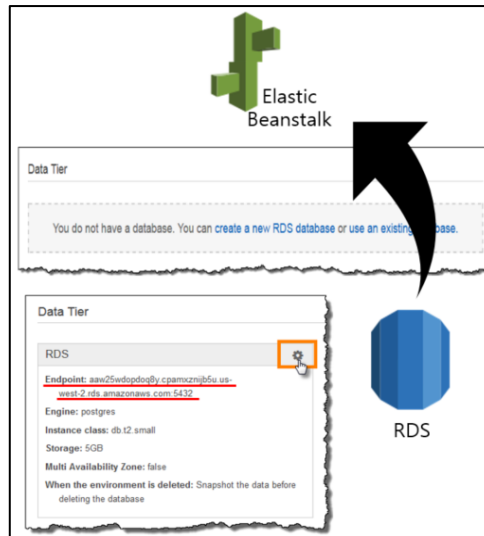
3.3.3 Amazon EC2



[Figure 8]Amazon EC2

Amazon Elastic Compute Cloud(EC2)는 클라우드에서 안전하고 크기 조정이 가능한 컴퓨팅 파워를 제공하는 웹 서비스입니다. 개발자가 더 쉽게 웹 규모의 클라우드 컴퓨팅 작업을 할 수 있도록 설계되어있습니다.

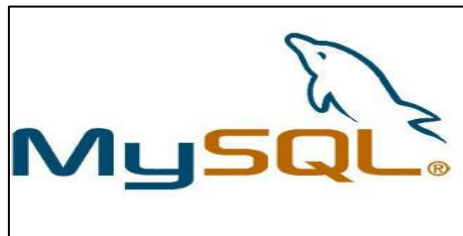
3.3.4 Amazon RDS



[Figure 9] Amazon RDS

클라우드에서 관계형 데이터베이스를 더욱 쉽게 설정, 운영 및 확장할 수 있도록 지원하는 관리형 클라우드 데이터베이스로 MySQL, MariaDB, PostgreSQL, OracleDB 등 다수의 상용 DB를 지원한다. Amazon Beanstalk에 Endpoint로 연동하여 사용할 수 있다.

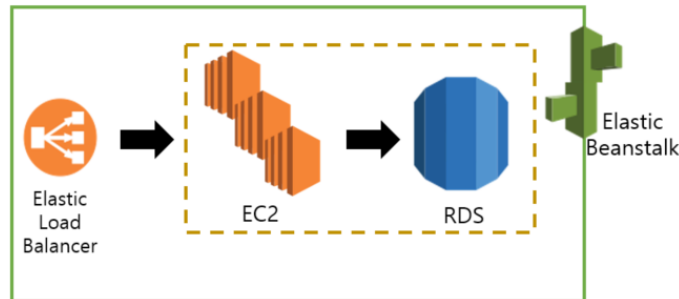
3.3.5 MySQL



[Figure 10] MySQL Database

표준 데이터베이스 질의 언어인 SQL(Structured Query Language)을 사용하는 개방 소스의 관계형 데이터베이스 관리 시스템(RDBMS). 매우 빠르고 유연하며 사용하기 쉬운 특징이 있다. 다중 사용자, 다중 스레드를 지원하고 다양한 프로그래밍 Language를 위한 API를 지원한다.

3.3.6 Amazon Elastic Beanstalk

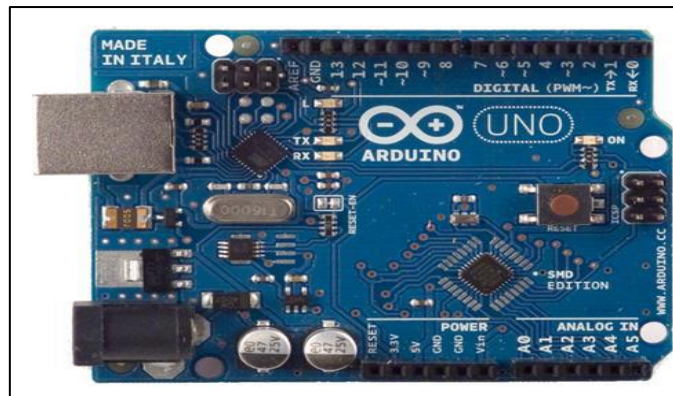


[Figure 11] Amazon Elastic Beanstalk

EC2는 아마존에서 제공하는 가장 기본적인 가상 서버 임대 서비스인데 SSH를 통해 원격으로 접속해 편의대로 조작하여 이용 할 수 있다. Elastic Beanstalk는 그 자유도를 제한하되 편리성을 증가시킨 서비스로서 손쉽게 웹 앱과 서비스를 배포, 관리 및 증감할 수 있는 서비스이다.

3.4 하드웨어

3.4.1 Arduino



[Figure 12] Arduino Uno

오픈 소스를 지향하는 마이크로 컨트롤러(micro controller)를 내장한 기기 제어용 기판. 컴퓨터 메인 보드의 단순 버전으로 이 기판에 다양한 센서나 부품 등의 장치를 연결 할 수 있다. 본 프로젝트 에서는 비접촉 전류 센서와 연결하여 1구 연장선의 연결된 세탁기의 전류를 측정한다.

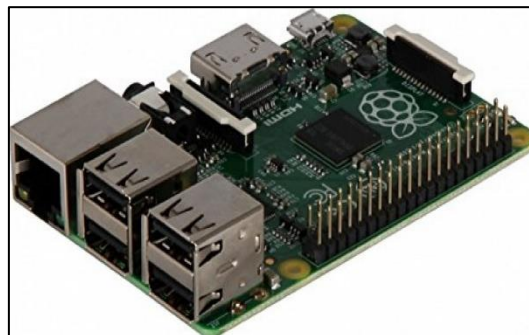
3.4.2 Non-Invasive AC Current Sensor – 100A(비접촉 전류 센서)



[Figure 13] 비접촉 AC 전류센서

비접촉 AC 비접촉 전류 센서로써 측정하고자 하는 라인에 후크를 체결하여 전선에 흐르는 전류량을 측정할 수 있는 센서이다.

3.4.3 Raspberry pi



[Figure 14] Raspberry pi3.0

Raspberry pi는 한마디로 말하면 '초소형 컴퓨터' 이다. 일반적으로 Raspberry pi는 리눅스 기반 라즈비안OS를 SD카드에 저장하여 구동한다. 본 프로젝트에서는 다음 세 가지의 역할을 한다.

- Arduino와 시리얼통신을 이용하여 각 상황에 맞는 데이터를 수신.
- EC2 서버와 소켓통신하여 Arduino로부터의 데이터를 송신.

3.4.4 Smart Phone

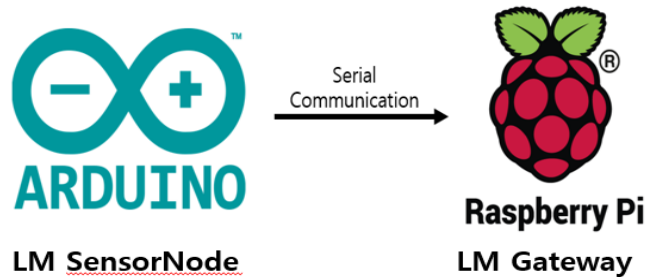


[Figure 15] Galaxy S7 & iPhone 6s

무인 공용 세탁기 관리 서비스인 LAMA서비스를 이용하기 위해서는 어플리케이션을 설치해야 한다. 단순히 사용자의 경우 별도의 회원가입 절차없이 원하는 세탁소의 세탁기 가동 현황을 살펴볼 수 있으며, 가동종료 알림을 받아볼 수 있다. 관리자의 경우, 회원가입을 통해 자신의 세탁소를 원격으로 관리할 수 있다. 관리는 고장여부 설정, 누적 가동횟수 살펴보기, 예상수익 계산하기 등의 기능으로 가능하다. 관리자가 본 어플리케이션을 이용하기 위해서는 먼저 자신의 세탁소 등록을 마쳐야 한다.

4. 네트워크 구성과 통신 방식

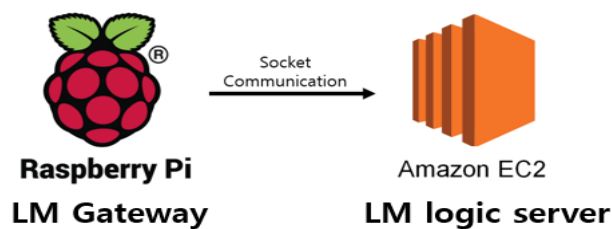
4.1 LM SensorNode – LM Gateway



[Figure 16] LM SensorNode – LM Gateway

LM SensorNode와 LM Gateway간 USB Cable을 이용하여 시리얼 통신을 한다. LM Gateway에서는 연결된 케이블의 포트환경을 설정한다. 본 프로젝트에서 LM SensorNode는 세탁기 전원 On/Off에 해당하는 문자열 데이터를 전송한다. LM Gateway에서는 수신된 문자열 데이터를 LM logic Server에 소켓통신 할 준비를 한다.

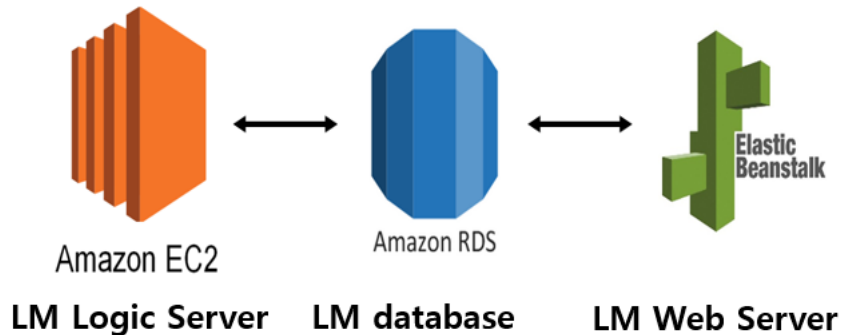
4.2 LM Gateway – LM logic Server



[Figure 17] LM Gateway – LM logic Server

LM Gateway는 내장된 무선랜을 이용하여 wifi에 접속한다. 무선 네트워크 환경이 조성된 후, 아마존에서 제공하는 가상 컴퓨팅 인스턴스인 EC2(= LM logic server)의 퍼블릭 IP를 목적지로 하여 소켓통신 환경을 구축한다. 소켓통신 환경 하에 문자열 데이터를 주고 받는다.

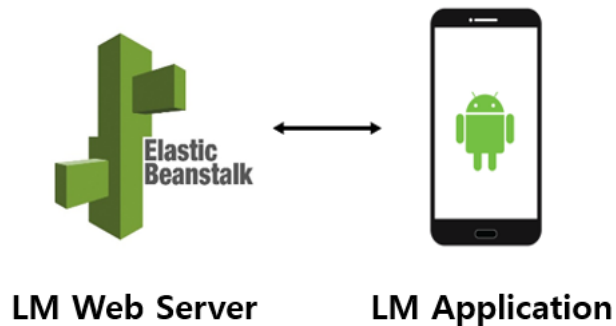
4.3 LM Logic Server – LM database – LM Web Server



[Figure 18] LM Logic Server – LM database – LM Web Server

가상 컴퓨팅 서버인 LM Logic Server과 데이터베이스 서버인 LM database간의 연동을 통해 데이터를 축적한다. 사용자에게 서비스를 제공하기 위해 데이터가 요청될 경우 LM Web Server를 경유하여 데이터베이스로부터 정보를 가져온다.

4.4 LM Application – LM Web Server

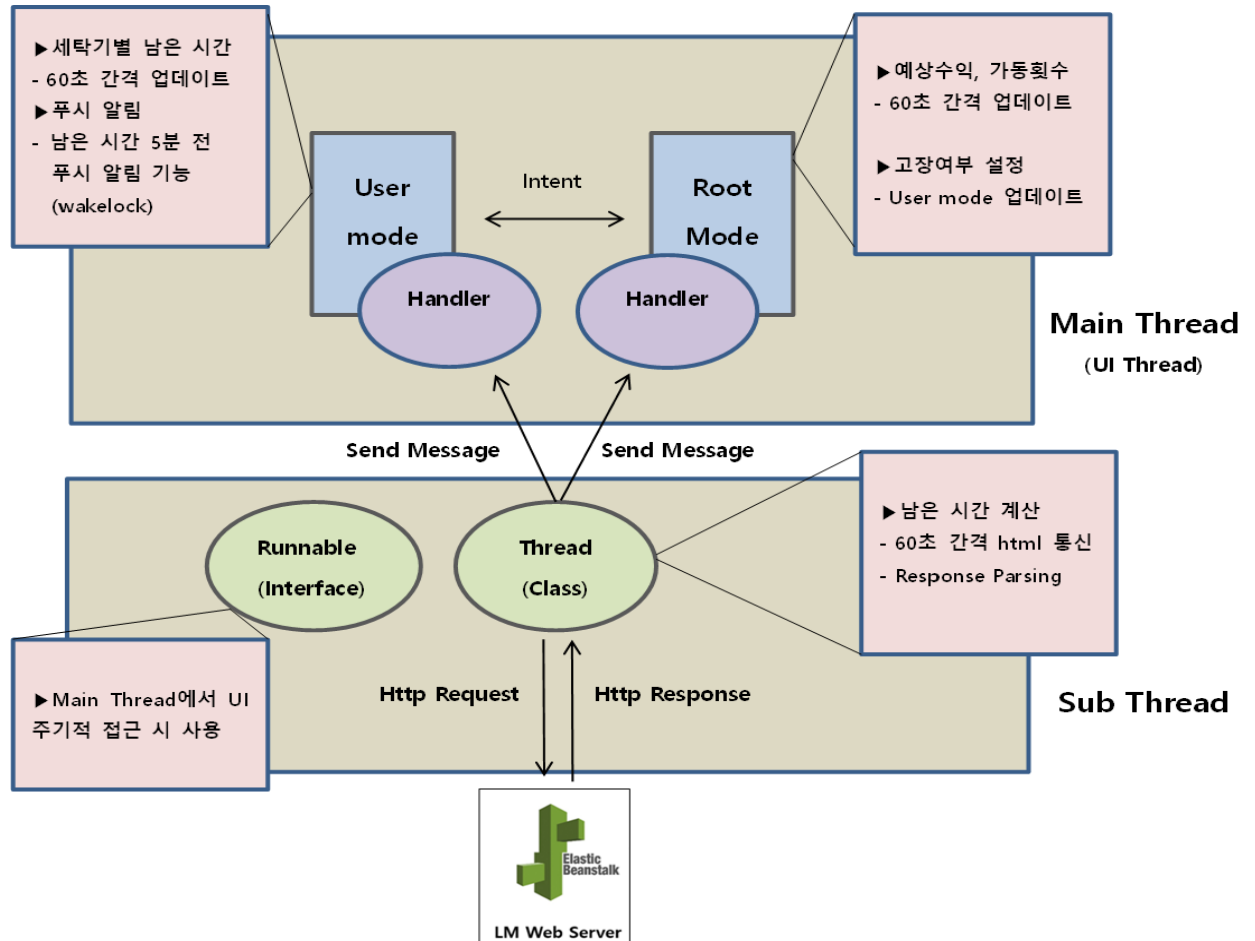


[Figure 19] LM Application – LM Web Server

LM Application은 LTE 또는 Wifi를 통해 네트워크로 연결되며 URL을 사용하여 서버와 HTTP프로토콜을 이용해 통신할 수 있도록 해준다.

5. 시스템 상세 설계

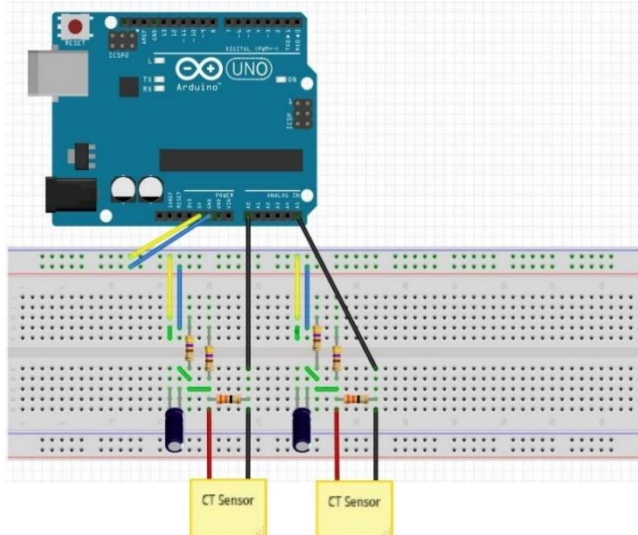
5.1 LM Application



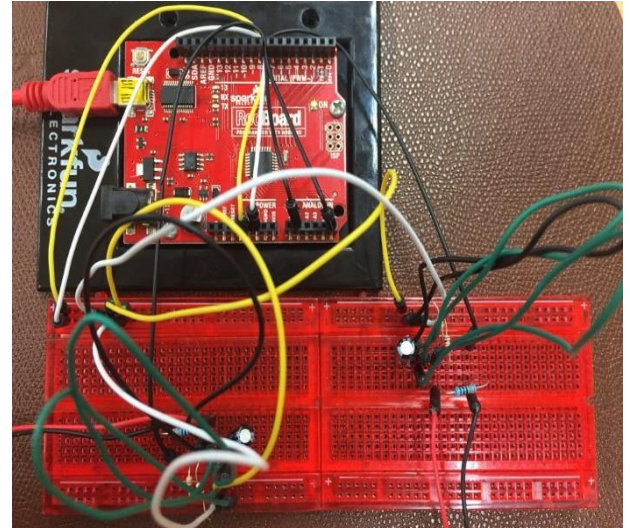
[Figure 20] LM Application

5.2 LM SensorNode

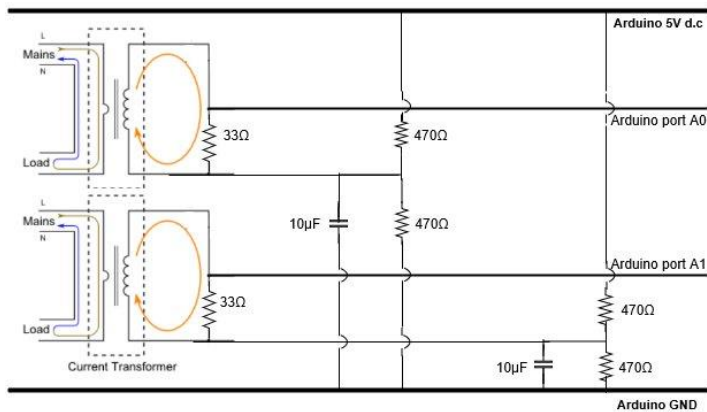
LM SensorNode는 비접촉 전류센서와 Arduino, 그리고 EnergyMonitor library를 이용해 측정하였다. 비접촉 전류센서 측정을 할 시에 측정하고자 하는 전선의 피복을 벗겨, 그라운드에 해당하는 전선을 제외하고 측정하였다



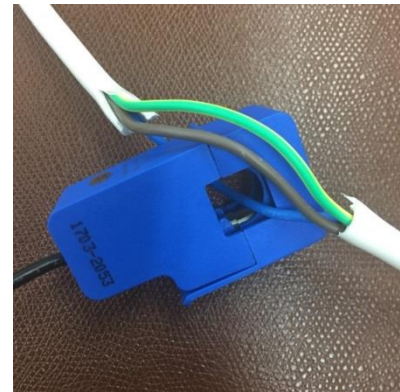
[Figure 21] 회로 모식도



[Figure 22] 회로 구성



[Figure 23] 비접촉 전류센서를 연결한 아두이노 회로도

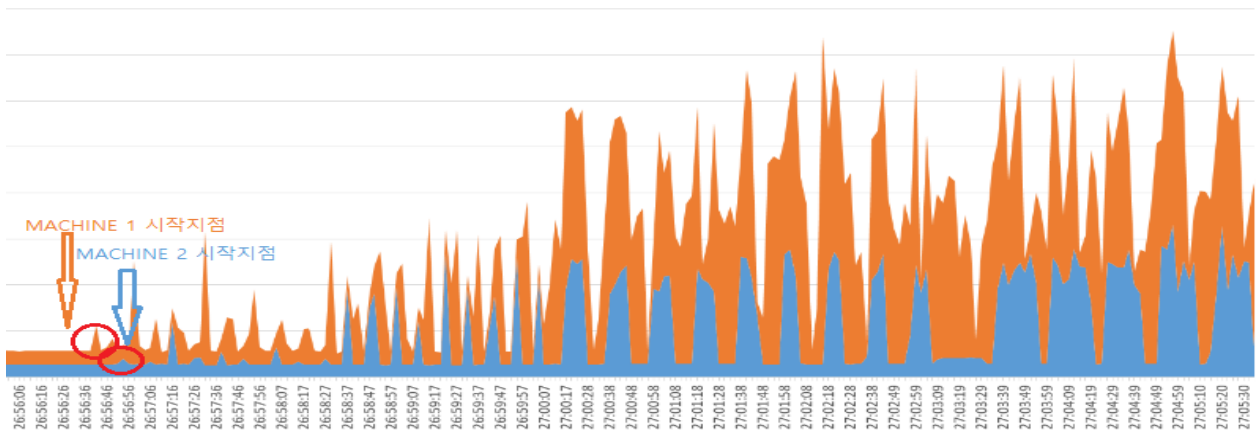


[Figure 24] 비접촉 전류센서 실제 측정 사진

LM SensorNode로 피상전류를 측정한 결과, 시작점은 표본 1개로 정확히 판단할 수 있지만, 종료시점은 표본 1개로 판단하기에 어려움이 있다. (값의 차이는 세탁물량의 차이에 비례한다.)

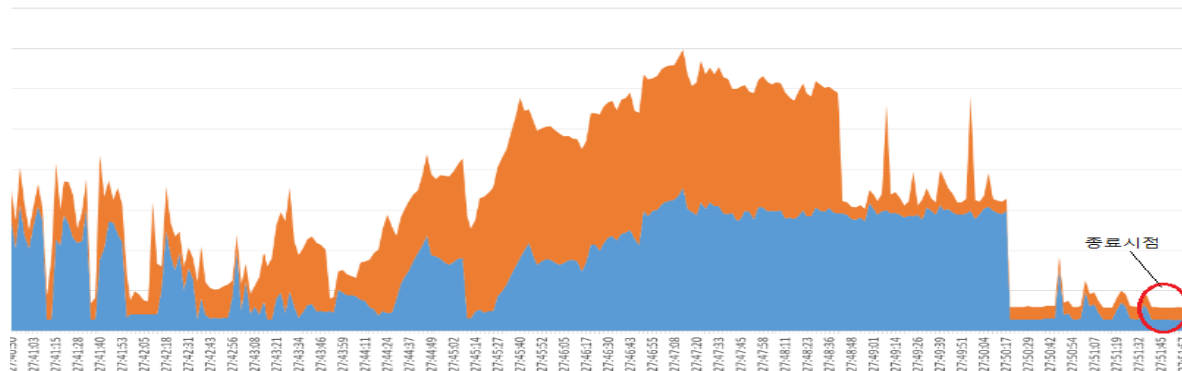
최종보고서: 비접촉 전류 센서를 이용한 무인 공용 세탁기 관리 시스템

시작 후 10분



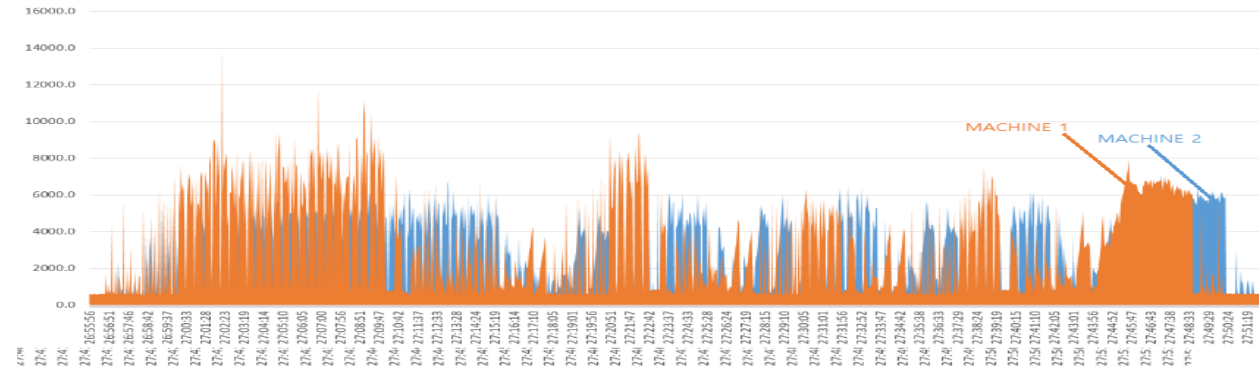
[Figure 25] 시작 후 10분 세탁기 전류 측정 그래프

종료 10분 전



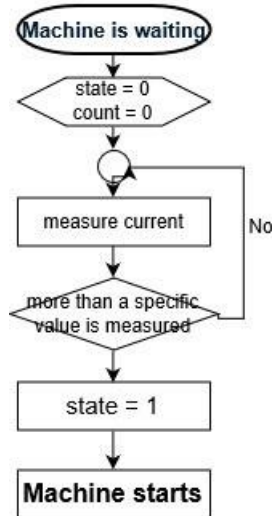
[Figure 26] 종료 10분 전 세탁기 전류 측정 그래프

시간별 세탁기 피상전류 값

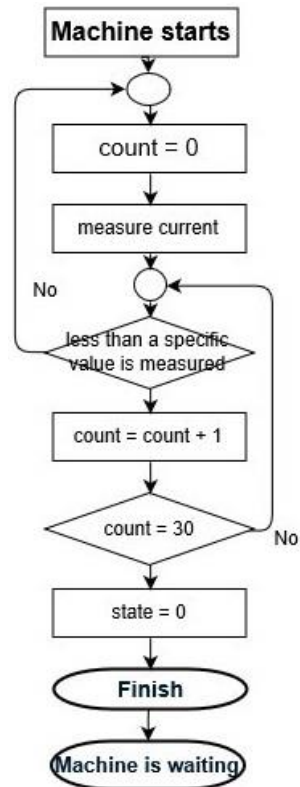


[Figure 27] 세탁기 전류 측정 그래프

결과적으로, LM SensorNode에서 센서데이터를 1.5초 단위로 수집한다. 세탁기의 종료시를 결정하기 위해서는 세탁기 코드에 흐르는 전류를 측정하는데, 확실한 종료를 결정하기 위해서는 실험 측정 결과 최소 45초의 시간을 필요로 하므로, 연속적인 30개의 표본으로 판단하였다.



[Figure 28] 시작 알고리즘

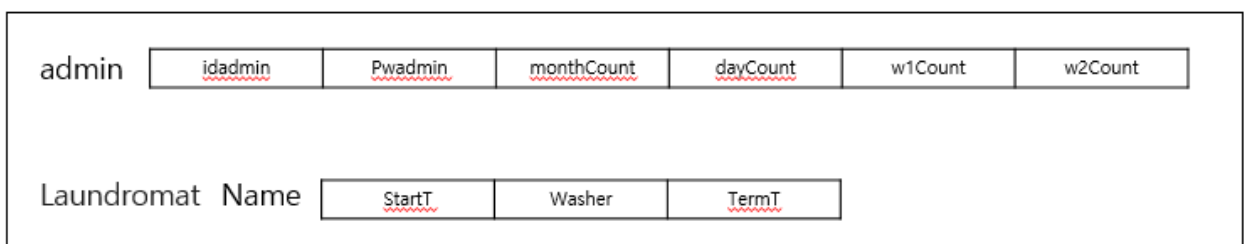


[Figure 29] 종료 알고리즘

LM SensorNode에서 센서데이터를 1.5초 단위로 수집한 후, Arduino내의 알고리즘을 거친 값들을 CF Gateway로 Serial 통신을 통해 전송하여 준다.

5.3 LM Server(Web & Logic) + LM Database

LM Logic Server는 LM SensorNode에서 알고리즘에 의해 얻어진 세탁기 별 On/Off 상태를 판별한 후 LM Database에 시작 시간과 세탁기 명을 기입한다. LM Web Server에서는 LM Database에 기입된 정보를 토대로 LAMA App로부터 들어오는 다양한 요청을 처리하기 위한 여러 Method들이 정의 되어 있다. LMAM App은 LM Web Server로 URL 요청을 하여 HTTP통신으로 데이터를 주고 받는다.



[Figure 30] LM Server(Web & Logic) + LM Database



최종보고서: 비접촉 전류 센서를 이용한 무인 공용 세탁기 관리 시스템

LM Logic Server에서 LM Database는 시작시간과 종료시간을 측정해 정보를 insert 하는 insert_StartT, insert_TermT 메소드가 있고 LAMA App의 요청에 응답하는 LM Web Server 메소드로는 User Mode에서 세탁기별 가동 현황을 알릴 수 있는 fetch 메소드, Root Mode에서 세탁기별 일, 월 별 가동 횟수를 알 수 있는 login 함수가 있다.

6. 적용방안 및 기대효과 (SWOT 분석)

- 24시가 모자란 바쁜 현대인들이 직접 무인 공용 세탁시설에 가지 않아도, 어플리케이션을 통해 세탁기 가용 유무를 알 수 있다.(편의성)
- 각각의 공용 세탁시설 관리자들의 어플리케이션 사용량이 증가하면, 어플리케이션 사용자들은 다른 세탁시설의 상황까지 알 수 있어 자신의 스케줄에 맞춰 효율적으로 세탁시설을 이용할 수 있다.(효율성)
- 어플리케이션 사용자는, 별도의 회원가입이 필요 없고, 가동 종료 알림을 알려주는 Push알림 기능을 통해, 세탁기 종료 시간 내에 빨래를 찾으러 갈 수 있는데, 이것은 세탁기 사용 순환율을 높여주며 사용자들이 세탁기 가동 시간 내에 다른 일정을 처리할 수 있도록 돕는다. (접근성, 효율성)
- 무인 공용 세탁시설 관리자는 어플리케이션을 이용하여 세탁시설 관리에 도움이 되는 정보들, 예를 들면, 사용 현황, 누적 사용 횟수 등을 알거나 예상되는 수익을 알 수 있으며, 세탁기 이용률이 많은 시간대 등을 알 수 있어 세탁시설 환경을 능동적으로 개선할 수 있다.(실용성)
- 무인 공용 세탁시설 관리자는 어플리케이션을 통해, 고장으로 인해 작동하지 않는 세탁기의 정보를 입력할 수 있다. 이것은 세탁소를 이용하는 사람들의 혼란을 막고, 점검 후 다시 그 정보를 입력하는 등 세탁기 정보에 대한 빠른 업데이트가 가능하다.(실용성)

SWOT 분석에 의거한 경영 전략

Strength

: 무인 공용 세탁시설의 빠른 순환, 세탁시설 사용자의 시간 절약, 세탁시설 관리자의 빠른 세탁시설 이용 현황 판단

Weakness

: 사람들의 이용률이 적은 세탁시설에서는 어플리케이션 사용량이 적을 수 있다.

Opportunity

: 주거지 주위의 공용 세탁시설 뿐만 아니라, 기숙사나 군대 등 공용으로 사용하는 모든 세탁시설, 나아가 다른 유형의 무인 점포로 해당 서비스를 확장할 수 있다.

Threat

: 기계가 전원에 연결되었을 때, 그 작동시간이 길 경우에만 한정적으로 사용할 수 있는 기술이다.

최종보고서: 비접촉 전류 센서를 이용한 무인 공용 세탁기 관리 시스템

[SO(강점-기회) 전략]

- 이용자가 많은 모든 공용 세탁 시설에 서비스를 도입하여, 보다 효율적인 공용 세탁 체계를 갖춘다.
- 관리자가 어플리케이션에 누적된 정보를 바탕으로, 세탁시설의 환경을 빠르게 개선하여 단골 이용자를 유지하거나 보유할 수 있다.

[ST(강점-위협) 전략]

- 해당 서비스를 적용할 수 있는 경우가 작동시간이 긴 특정 기계들에 한정적이지만, 서비스의 효율을 높여 고정 사용자들을 확보한다.

[WO(약점-기회) 전략]

- 이용자가 적은 공용 세탁시설은 어플리케이션 사용을 통한 마일리지 적립 등의 추가적인 혜택을 제공하여 이용자를 확보할 수 있다.

7. 향후 계획

세탁기를 이용하여 실험하면서 우리는 한계점에 부딪혔다. 바로 실제 세탁기의 종료 시간과 본 어플리케이션에서 제공하는 남은 시간에는 약간의 오차가 존재한다는 점이었다.

오차는 다음과 같은 이유 때문에 존재했다. 세탁기 자체적으로 모니터를 통해 제공하는 남은 시간과 실제 남은 시간에는 1분에서 3분 정도의 차이를 보였다. 또한, 세탁기의 종료 상태를 알기 위한 알고리즘에 최소 40초가 필요했다. 그리고, 어플리케이션과 서버와의 부하를 줄이기 위해서 1분마다 HTTP 통신을 한 것이 오차를 발생시키는 이유가 되었다.

이를 개선하기 위해서 우리는 몇 가지 방안을 생각하고 있다. 먼저, 세탁물 무게에 의한 세탁 시간 자체 계산 시스템을 통하여 LAMA 알고리즘의 정확도를 개선이 필요하고, 서버의 부하를 고려한 요청 횟수를 코드 상에서 개선하여 오차의 범위를 줄이는 것이 개선 방안이라고 생각한다.



8. 프로젝트 세부추진계획 및 세부일정

단계	개발계획	1주차	2주차	3주차	4주차	5주차	6주차
기획	주제선정						
	프로젝트 계획수립						
	시나리오 구상						
개발	관련지식학습						
	아두이노 라즈베리파이 개발						
	안드로이드 앱개발						
	웹 서버 구축						
	DB구축 & 연동						
최종	최적화						
	보완 및 최종발표						

[Table 3] 프로젝트 추진 일정

9. 팀원 담당업무

이름	구분	분야
신교선	팀장	Raspberry pi & Server(socket)
김진우	팀원	Arduino & Raspberry pi
김유진	팀원	Arduino & Raspberry pi
김세용	팀원	Web Server & Database
김준영	팀원	Android App & App UI
유정현	팀원	Android App & App UI

[Table 4] 팀원 담당 업무

10. Source Code

별도 파일 첨부