

환경조절 스마트팜

안상현, 배상진, 송환석, 이정원, 이은서*
안동대학교 컴퓨터공학과

e-mail : ansanghyun20@gmail.com, bsj3232@gmail.com, aqz1178@naver.com,
dmks326@naver.com, eslee@andong.ac.kr*

Environmental Control Smart Farm

Sang-Hyun An, Sang-Jin Bae , HwanSeok Song, Jeong-Won Lee, Eun-Ser Lee*

Dept of Computer Engineering, Andong National University

요 약

라즈베리파이에 아두이노를 사용해 조도, 냉각기, 열선패드 등의 센서를 사용하여 하드웨어를 구축한다. 또한 라즈베리파이에 웹 서버(PHP)를 구축하고 데이터베이스(MySQL)에 센서 값을 담아 실시간으로 웹 서버에 출력하고 제어한다. 본 논문에서는 환경이 조절되는 스마트팜의 설계 과정(UML, Use Case, DataBase 설계, 모듈 설계 등)을 포함하고 웹 서버에서 센서를 동작할 수 있게끔 만드는 과정을 설명한다.

1. 서론

현재 폭염으로 인한 농작물뿐만 아니라 농업인에게 피해가 가는 상황이 발생하고 있다.[1] 뿐만 아니라 폭우, 폭설, 홍수 등 다양한 재해로 인하여 작물의 재배가 어려워 값이 폭등하는 경우가 많다.[2] 이로 인해 소비자가 작물을 위해 지불하는 금액이 상승하고 있다. 본 논문에서는 일반 소비자를 위해 가정에서 개인이 손을 쓰지 않고 자동으로 환경이 조절되는 스마트팜을 이용해 작물을 길러 먹을 수 있도록 하여 경제적인 이득을 취할 수 있는 것을 목적으로 한다. 또한 환경 조절 스마트팜의 설정에 따라서 작물 재배 뿐만 아니라 화훼의 목적으로 사용할 수 있다.

기존의 스마트팜은 시설 재배지의 환경이 다르고 불규칙적인 날씨의 변화를 겪는 문제가 많이 있다.[3] 오토 파밍 팀에서 개발된 스마트 팜은 기존 스마트 팜의 기능을 가진 빛 공급, 수분 공급을 자동화할 뿐만 아니라 냉각기, 열선을 이용한 환경 자동 조절기능 까지 존재하는 시스템이다. 더하여 라즈베리파이의 PI Cam을 설치해 웹 서버에 접속만 하면 언제 어디서든지 식물의 현재 상태나 빛의 양, 물이 공급되는 상황 등을 실시간으로 접속하고 확인할 수 있다.

1)

2. 관련 연구

웹 서버 구축, 데이터베이스사용을 위한 라즈베리파이 그리고 센서를 제어하는 아두이노와 같이 프로젝트에서 사

용된 장치들의 관련 연구를 나타낸다.

2.1 라즈베리파이 (Raspberry Pi)

라즈베리파이 (Raspberry Pi)는 영국 잉글랜드의 라즈베리 파이 재단이 교육용 목적으로 만든 초소형 싱글보드 컴퓨터이다. 리눅스 커널 기반 운영체제를 사용하며 현재는 Raspbian이라는 라즈베리 파이에 최적화된 데비안 계열의 운영 체제를 권장하고 있다. 소형 컴퓨터이기 때문에 CPU, 메인보드, 입출력 포트, 네트워크 장치를 내장하고 있으며 키보드, 마우스, 디스플레이를 장치에 연결하면 성능은 낮지만 데스크탑 컴퓨터와 동일한 작업을 할 수 있다. 라즈베리파이는 센서 네트워크에 아두이노 등을 상호 연결 또한 가능하다.[4]

2.2 아두이노 (Arduino)

아두이노 (Arduino)는 사용하기 쉬운 하드웨어 및 소프트웨어를 기반으로 하는 오픈 소스 전자 플랫폼이다. 완성된 보드와 관련 개발 도구 및 환경 전부를 일컫는다. 아두이노는 다수의 센서로부터 값을 받아들여 LED나 모터 등과 같은 외부 전자 장치들을 통제하여 환경과 상호작용이 가능한 물건을 만들 수 있다. 또한 통합 개발 환경(IDE)를 제공하며 소프트웨어 개발과 실행코드도 제공한다. 컴파일된 펌웨어를 USB를 통해 쉽게 업로드 할 수 있으며 윈도우를 비롯해 Mac OS, 리눅스 등 여러 OS를 모두 지원한다. 보드의 회로도 또한 공개되어 있어서 누구나 직접 보드를 만들고 수정할 수 있다. 아두이노의 가장 큰 특징은 하드웨어 및 소프트웨어의 전문지식이 없어도 누구나 쉽게 배우고 사용할 수 있도록 개발되었다. [5]

* 본 논문의 교신저자임.

“본 연구는 2021년 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학사업의 연구결과로 수행되었음”(2019-0-01113)

3. 개발 방법론

소프트웨어 개발 방법론은 개발하는 방법에 대한 이론

을 말하며 개발 과정과 절차 그리고 산출물 등을 체계적으로 정리하고 표준화시킨 것을 말한다. 본 논문은 여러 개발 방법론 중 선형 순차 모델인 폭포수 모델을 사용한 설계 과정을 포함한다. 폭포수 모델은 계획, 요구 분석, 설계, 구현, 테스트 단계를 순서대로 진행하며 각 단계마다 완벽하게 진행해야한다. 이는 병행해서 진행할 수 없으며 이전 단계로 돌아갈 수 없는 개발 방법론이다.

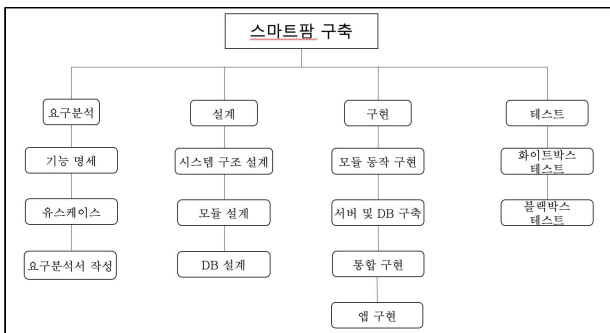
프로세스명	담당자	3	4	5	산출물
		3 4 5	1 2 3 4 5	1 2 3 4 5	
1. 계획					
목표 및 문제정의	송환석	■			하드웨어 명세서 일정 계획표 WBS
하드웨어 정의	배상진		■		
일정 예측	안상현			■	
작업 분할	이정원				
2. 요구 분석					
기능 명세	송환석,이정원		■		기능명세서 유스케이스 요구분석서 작성
유스케이스	송환석,안상현			■	
요구분석서 작성	배상진, 안상현				
3. 설계					
시스템 구조 설계	안상현,이정원			■	아키텍처 모듈 설계서 DB 설계서
모듈 설계	배상진				
DB 설계	송환석				
4. 구현					
모듈 동작 구현	배상진			■	모듈 웹 서버 모듈 웹 연동 안드로이드 앱
서버, DB 구축	송환석,이정원		■		
통합 구현	김진채			■	
애플리케이션 개발	안상현			■	
				■	
5. 테스트					
화이트박스 테스트	김진채				모듈 통합 시스템
블랙박스 테스트	김진채				

<그림-1> 일정계획표

그림 1 에서 계획, 요구 분석, 설계, 구현, 테스트 단계가 순서대로 나열 되어 있는 것을 확인할 수 있다. 또한 주차 마다 진행해야하는 과정을 검은색 박스로 나타내었으며 각 프로세스 마다 담당자와 산출물로 나타낸다.

4. 계획

계획 단계는 목표 및 문제정의를 가장 우선적으로 하며 다음으로 하드웨어를 정의하고 일정계획표와 WBS(Work Breakdown Structure)를 정의한다. WBS는 작업 분해 구조를 말하며 업무일정을 관리하고 계획하는 문서를 말한다. WBS의 산출물은 아래와 같다.



<그림 -2> WBS

WBS를 통해 팀의 업무를 효율적으로 관리할 수 있고 시각적으로 표를 표현해 프로젝트 진행 과정 중에 무엇을 해야 하는지 명확하게 알 수 있다.

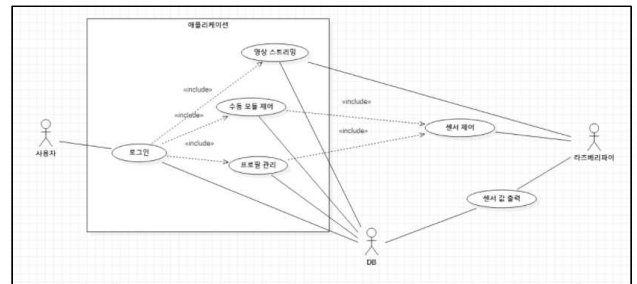
5. 요구 분석

요구 분석 단계에서는 사용자의 요구 사항을 수집, 기능 및 비기능 요구사항을 나누며 유스케이스를 작성하는 단계다. 본 논문은 기능을 명세하고 유스케이스를 작성하는 데에 초점을 두었다.

기능명	설명
1. 주소	사용자가 IP주소를 입력하는 것이다.
2. Enter	1번의 주소창에서 입력한 정보가 정확하기 확인한다. 여러 행렬하는 주소가 없습니다.
기능 명세서	
기능명	설명
1. 온도 출력	장치의 현재 온도를 출력한다.
2. 습도 출력	장치의 현재 습도를 출력한다.
3. 영상 스트리밍	라즈베리파이 카메라를 이용하여 현재 장치의 영상을 스트리밍한다.
4. 자동 전환 버튼	자동 제어 화면으로 넘어간다.
5. 수동 전환 버튼	수동 제어 화면으로 넘어간다.
기능 명세서	
기능명	설명
1. 온습도 출력	장치의 현재 온도와 습도를 출력한다.
2. 희망 온도 입력	장치의 희망 온도를 출력한다.
3. LED 시간 입력	LED등이 켜져있어야하는 시간을 입력한다.
4. 온도 제어	조도 센서 값을 입력한다.
5. 프로필	현재 사용할 수 있는 프로필을 출력한다. 터치하면 해당 프로필에 맞는
6. 프로필 저장	현재 입력되어 있는 희망 값을 저장한다. 저장버튼을 터치할 시 저장
7. 프로필 불러오기	DB에 저장되어 있는 프로필을 전부 불러온다.
기능 명세서	
기능명	설명
1. 온습도 출력	장치의 현재 온도와 습도를 출력한다.
2. LED 제어	LED를 ON/OFF한다. 팝업 메시지로 바뀐 상태를 출력한다.
3. 온도 제어	온도를 ON/OFF한다. 팝업 메시지로 바뀐 상태를 출력한다.
4. 자동기 제어	자동기를 ON/OFF한다. 팝업 메시지로 바뀐 상태를 출력한다.
5. 온도 제어(상)	희망 온도를 상승시킨다. 팝업 메시지로 바뀐 희망 온도를 출력한다.
6. 온도 제어(하)	희망 온도를 하락시킨다. 팝업 메시지로 바뀐 희망 온도를 출력한다.

<그림 - 3> 기능 명세서

기능 명세서에는 그림 3과 같이 UI 예상도, 각 기능에 대한 기능명 및 설명을 포함한다. 기능명은 버튼, 레이블 등에 해당하는 동작 이름을 결정하며 설명에는 이름에 대한 간단한 흐름을 작성하였다. 이는 최종적으로 희망하는 기능을 나타낸 명세서이며 구현 시 참고 문서가 된다.



<그림 - 4> 유스케이스

유스케이스는 UML의 행위자와 액터가 요구하여 시스템이 수행하는 일의 목표를 말한다.[6] 액터는 시스템을 사용하는 외부 존재로써 사용자나 시스템이 될 수 있고 유스케이스는 시스템의 기능의 단위로 시스템이 요구하는 기능을 나타낸다. 그림 4는 액터를 사용자, Database, 라즈베리파이로 두었다. 사용자가 로그인을 하게 되면 수동 모드를 제어 기능, 프로필 관리 기능, 영상 스트리밍기능을 제공한다. Database는 주로 센서와 관련된 기능과 연결이 되며 라즈베리파이에서 센서 제어와 값을 Database

로 보낸다. 이와 같이 유스케이스를 통해서 시스템이 수행하는 일을 한눈에 파악 가능하다.

6. 설계

설계 단계에서는 모듈을 설계하고 Database설계와 클래스 다이어그램 설계를 진행한다.

구분	기능
기능 명칭	이벤트 발생과 센서값을 처리하고 로그를 찍는 기능
요구 사항	<pre> enter message NULL, 사용자 ID, 온도, 습도, 조도, 시간, 위치, 상태 enter message int, 사용자 ID, 온도, 습도, 조도, 시간, 위치, 상태 enter message int, 사용자 ID, 온도, 습도, 조도, 시간, 위치, 상태 logn message 로그 기록을 위한 사용자 ID, 온도, 습도, 조도, 시간, 위치, 상태 </pre>
주요 알고리즘	<pre> 사용자 ID, 온도, 습도, 조도, 시간, 위치, 상태 사용자 ID, 온도, 습도, 조도, 시간, 위치, 상태 사용자 ID, 온도, 습도, 조도, 시간, 위치, 상태 </pre>
데이터 타입	온도, 습도, 조도, 시간, 위치, 상태
검정조건	온도, 습도, 조도, 시간, 위치, 상태
외부 인터페이스	온도, 습도, 조도, 시간, 위치, 상태
검정조건	온도, 습도, 조도, 시간, 위치, 상태
외부 인터페이스	온도, 습도, 조도, 시간, 위치, 상태
검정조건	온도, 습도, 조도, 시간, 위치, 상태
외부 인터페이스	온도, 습도, 조도, 시간, 위치, 상태
검정조건	온도, 습도, 조도, 시간, 위치, 상태

<그림 - 5> 모듈 설계서

그림 5의 모듈 설계서에는 기능에 대한 이름을 작성하고, 기능에 대한 설명, 오류 처리, 내부 자료구조, 사용하는 파일, 알고리즘의 내용을 포함한다. 이 항목들은 유스케이스를 참조하여 작성하며 유스케이스에 포함된 모든 기능을 하나씩 설명하는 모듈 설계서가 된다.

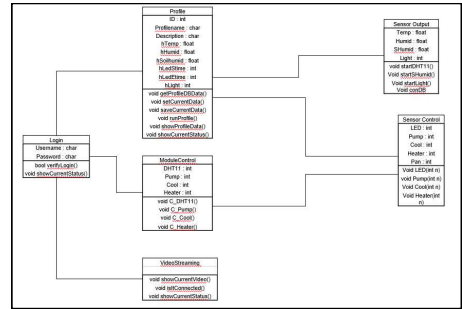
데이터베이스 설계서는 실제 구현할 테이블을 설계한다. 센서의 값을 실시간으로 불러와 저장하는 부분, 희망하는 환경을 저장하는 부분, 로그인, 프로필 부분으로 나타낸다.

이름	타입	키	기본값	타입	키	기본값
id	INT	NOT NULL, PRIMARY KEY		id	INT	NOT NULL, PRIMARY KEY
password	VARCHAR	NOT NULL		password	VARCHAR	NOT NULL
username	VARCHAR	NOT NULL		username	VARCHAR	NOT NULL
name	VARCHAR	NOT NULL		name	VARCHAR	NOT NULL
age	INT	NOT NULL		age	INT	NOT NULL

<그림 - 6> DB 설계서

각 테이블 항목은 다음과 같은 항목을 포함한다. 열 이름, 내용, 데이터 형식, 길이, 비고(ex. NULL, AUTO_INCREMENT)가 있다. 상태 저장 (Status)테이블에는 실시간 데이터(온도, 습도, 조도 등)가 입력이 된다. 사용자 설정 센서 (Sen_set)테이블은 사용자가 설정한 온도, 습도, 조도를 받아오고 센서를 제어하는데 사용해 제어의 중요한 부분을 나타낸다. 프로필 테이블에서는 사용자가 프로필을 저장할 수 있는 테이블이지만 프로젝트에서 실질적으로 구현되지 않은 테이블이다. 위 설계서의 데이터베이스에서

는 주로 센서들의 OUTPUT 값을 저장하는 용도로 사용하였고 센서를 동작시키는 부분은 데이터베이스를 사용하지 않는다.



<그림 - 7> 클래스 다이어그램

7. 구현

데이터를 확인하고 제어하기 위해서는 데이터가 존재해야 한다. 따라서 센서 값을 아두이노에서 가져와 데이터를 삽입하는 것을 가장 우선순위로 설명한다. 아래의 코드는 상태 테이블에 데이터를 삽입하는 간단한 코드이다.

```

val = (Temp, humid, Light, CTime)
sql = "INSERT INTO Status(Temp, Humid, Light, CTime) VALUES(%s, %s, %s, %s)"
curs.execute(sql, val)
conn.commit()
    
```

<그림 - 8> 센서 값 삽입

여기서 Temp(온도), Humid(습도), Light(빛), CTime(현재 시간)을 아두이노에서 불러와 삽입한다. 이를 위해 그림 9는 아두이노와 라즈베리파이의 파이썬과 시리얼 통신을 해 센서 값을 불러와 그림 8에 필요한 데이터를 제공해준다.

```

port = '/dev/ttyACM0'
brate = 9600
seria = serial.Serial(port, baudrate = brate, timeout=None)
    
```

<그림 - 9> 아두이노 시리얼 통신

위와 같이 값을 데이터베이스에 입력했다면 다음으로 웹서버에서 출력을 해준다. 그림 10은 PHP에 값을 출력하는 코드이다.

```

$query3 = "SELECT * FROM Status ORDER BY ID DESC LIMIT 1";
$result = $connect->query($query3); //쿼리실행
$row = mysqli_fetch_object($result); //실행된 쿼리값을 읽음

echo 'Temperature : '. $row->Temp. '<br>';
echo 'Humid : '. $row->Humid. '<br>';
echo 'Light : '. $row->Light. '<br>';
echo 'Time : '. $row->CTime. '<br>';
    
```

<그림 - 10> PHP 출력

query로 마지막줄을 가져와 항상 최신의 정보를 가져온다. 데이터는 웹페이지를 새로 고침 시켜주는 기능으로 갱신할 수 있다. 이는 javascript를 통해서 가능하다.

센서를 웹 서버에서 확인이 가능하게 해주었다면 장치

들(LED, 냉각기, 열선패드 등)을 제어해야 한다. 이를 위해서 실시간 데이터 값을 받아와 원하는 값을 비교하여 동작시키는 코드를 작성한다.

```
query2 = "SELECT * FROM Sen_set ORDER BY ID DESC LIMIT 1;"
```

<그림 - 11> 설정된 데이터 SQL

그림 11에서는 PHP에서 보낸 데이터 값을 마지막 줄만 읽어온다. 이를 통해서 기존의 Status(실시간 센서데이터)와 비교를 하며 장치를 제어할 수 있다.

```
if(int(sen[2])>sta[2] and ok_Humid ==0):
    print("Water ON")
    c = "5"
    c= c.encode('utf-8')
    arduino.write(c)
    i=i+1
    time.sleep(1)
    if(i==3):
        ok_Humid=1
        i=0
        print("break")
```

<그림 - 12> 동작 예시

그림 12의 코드는 Status테이블(sen변수) Sen_set테이블(sta변수)와 비교를 해 모듈을 동작시켜주는 간단한 예시를 나타낸다. 위의 코드를 통해서 열선패드, 물 공급, 빛 제어 등 모든 모듈을 동작한다.

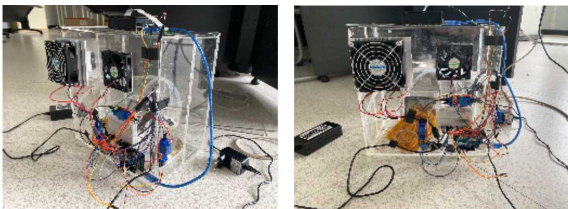


그림 13 최종 완성 시스템

그림 13은 최종 완성 시스템이다. 아두이노, 라즈베리파이를 사용한 모습을 확인할 수 있다. 또한 열선패드, 냉각기 등을 포함한다.



그림 14 스마트팜 앱

그림 14는 최종적으로 완성된 애플리케이션의 화면을 나타낸다. 상단에는 웹뷰로 구성되어있고 하단에는 버튼들로 구성이 되어있다. SENSOR 버튼은 센서의 값을 실시간 확인 가능한 웹 뷰로 연결이 된다. INSERT DATA버튼은

사용자가 설정할 수 있는 밝기, 온도, 습도를 통해서 자동으로 환경 조절을 할 수 있는 웹 페이지로 연결된다. 마지막으로 CAM버튼은 실시간으로 내부를 확인 가능한 웹 페이지로 연결된다.

8. 결론

결론적으로 어느 공간에서나 사용자가 눈으로 직접 확인하며 식물을 키울 수 있는 시스템을 구성할 수 있다. 식물을 키우는 모든 분야에 접목이 가능하며 동물을 키우는 것에도 접목할 수 있다. 기본적으로 생물에겐 필요한 물, 온도, 빛을 제공하기 때문이다. 주요 분야로는 꽃이나 작물을 재배할 때 물, 온도, 빛 등 식물이 자라는 최적의 환경을 맞출 수 있기 때문에 환경에 구애받지 않는 이점이 있어 농업, 화훼 등 식물을 키우는 분야가 적합하다. 이 제품을 사용함으로써 사용자는 식물을 키우는 데 시간적인 부담을 덜 수 있으며 좁은 공간에서도 식물을 키워 수확을 할 수 있게 된다. 또한 개인적으로 식물을 키워 심리적인 위안을 받을 수도 있다. 하지만 식물의 발아환경은 모든 식물이 각각 다르므로 사용자가 직접 설정하기에는 부담을 느낄 수 있다. 이런 점을 감안하여 DB에 각 식물의 발아환경 등을 넣어 개선할 여지가 있다.

참고문헌

[1] 농수축산신문, '전국이 불가마...농업인·농작물 고온피해 주의' 2021/07/20
<http://www.aflnews.co.kr/news/articleView.html?idxno=213822>

[2] 충남일보, '한파·폭설에 설 차례상 물가 들쭉... 과일·계란값 '경충' 2021/01/12
<http://www.chungnamilbo.co.kr/news/articleView.html?idxno=580312>

[3] 길민식, 광동걸, 최신행, 신종근. (2019). IOT 기반의 스마트 팜 시스템 구조설계에 관한 연구. 전력전자학회 학술대회 논문집, (), 543-545.

[4] Kim, Jeong-Won, "라즈베리파이를 이용한 스마트 홈 프로토타입 구현," 한국전자통신학회논문지, vol. 10, no. 10, pp. 1139 - 1144, Oct. 2015.

[5] Lee, In-Gu and Cho, Myeon-Gyun, "아두이노 기반의 독거노인을 위한 스마트홈 시스템," 대한임베디드공학회는 논문지, vol. 10, no. 5, pp. 307 - 315, Oct. 2015.

[6] 위키피디아, '유스케이스'
https://ko.wikipedia.org/wiki/%EC%9C%A0%EC%8A%A4_%EC%BC%80%EC%9D%B4%EC%8A%A4