



국민대학교  
전자정보통신대학  
컴퓨터공학부

# 캡스톤 디자인 I

## 종합설계 프로젝트

프로젝트명	야 너도 키울수 있어 스마트팟 (야스)
팀명	귀농
문서제목	결과보고서

Version	1.3
Date	2019-05-28

팀원	원정희 (조장)
	조현우
	이우재
	주연호

CONFIDENTIALITY/SECURITY WARNING

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개설 교과목 캡스톤 디자인 I 수강 학생 중 프로젝트 “XXXX XXXX” 를 수행하는 팀 “XXXXX” 의 팀원들의 자산입니다. 국민대학교 컴퓨터공학부 및 팀 “XXXXXX” 의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

## 문서 정보 / 수정 내역

<b>Filename</b>	수행결과보고서-야너도키울수있어 스마트팟.doc
<b>원안작성자</b>	원정희, 조현우, 이우재, 주연호
<b>수정작업자</b>	원정희, 조현우, 이우재, 주연호

수정날짜	대표수정 자	Revision	추가/수정 항목	내 용
2019-04-17	원정희	1.0	최초 작성	목표 및 하드웨어 부분 작성
2019-05-04	조현우	1.1	내용 추가 및 수정	수정된 연구내용 추가
2019-05-24	이우재	1.2	내용 추가 및 수정	AWS 부분 작성
2019-05-28	주연호	1.3	내용 추가 및 수정	안드로이드 부분작성

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

## 목 차

1	개요	4
1.1	프로젝트 개요	4
1.2	기존 스마트팟의 문제점	4
2	개발 내용 및 결과물	5
2.1	목표	5
2.2	연구/개발 내용 및 결과물	6
2.2.1	연구/개발 내용	6
2.2.2	시스템 기능 요구사항	8
2.2.3	시스템 비기능(품질) 요구사항	10
2.2.4	시스템 구조 및 설계도	10
2.2.5	활용/개발된 기술	11
2.2.6	현실적 제한 요소 및 그 해결방안	12
2.2.7	결과물 목록	13
2.3	기대효과	13
3	자기평가	13
4	참고 문헌	14
5	부록	15
5.1	사용자 매뉴얼	15
5.2	운영자 매뉴얼	15
5.3	테스트 케이스	16

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

# 1 개요

## 1.1 프로젝트 개요

요즘 깨끗한 채소,과일 등을 구하는 것이 점점 어렵게 되어가고 있다. 이러한 문제를 조금이나마 극복하고자 바쁜 현대인들도 집에서 신선하고 안전한 작물을 재배할 수 있는 방법으로 스마트팜을 선정하였다.

스마트팜이란 ICT 기술을 비닐하우스, 축사, 과수원 등에 접목해 원격, 자동으로 작물의 생육환경을 적절히 제어할 수 있는 농장이다. 우리는 스마트팜에 있는 기술을 접목하여 집에서도 키울 수 있는 작은 농장을 운영하려고 한다. 이 프로젝트는 작물을 키울 수 있는 스마트팜을 축소한 형태로 기존의 대규모의 스마트팜 보다 축소하여 비용적인 측면을 줄이고 접근성을 향상시킬 수 있는 스마트팟을 제작 할 예정이다.


작물이 자랄 수 있는 최적의 환경을 구축 할 수 있게 센서들을 사용하여 스마트팟을 구성할 예정이다. 또한 집에 있지 않아도 원격으로 조종하여 관리를 할 수 있다.

## 1.2 기존 스마트팜의 문제점

### 1.2.1 스마트팜의 문제점

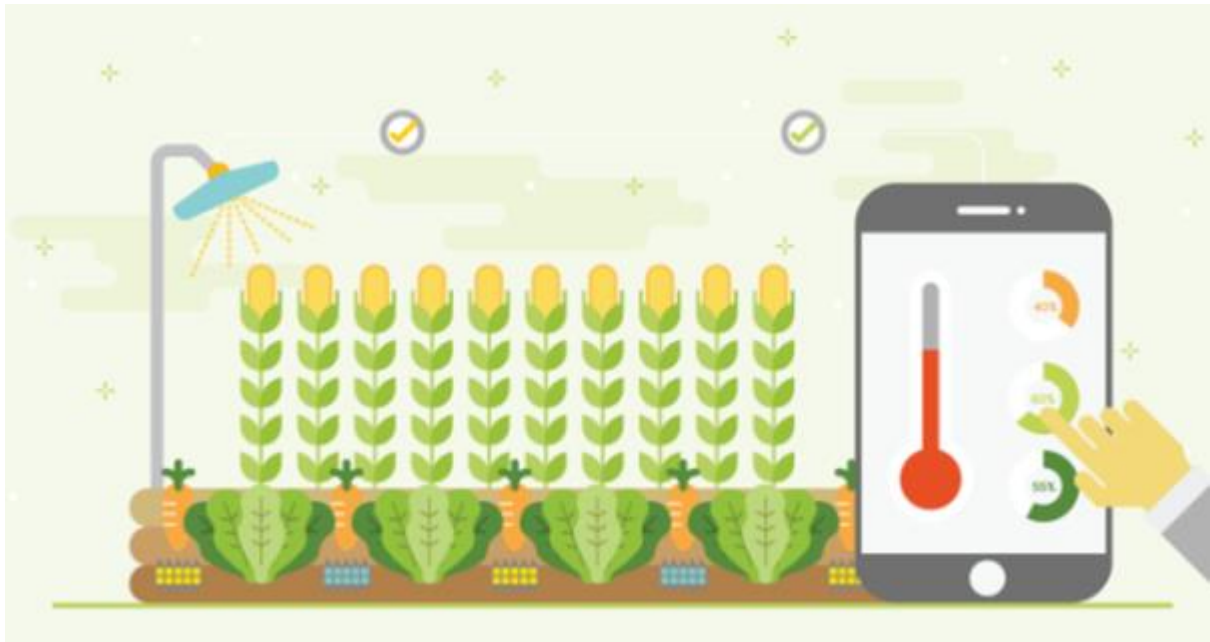
- 1) 현재 스마트팜이 빅데이터를 기반으로 운영하는데 농업관련 데이터를 모으는데 오랜 기간이 걸려 데이터를 축적하여야 해서 어려움이 많아 데이터가 부족하다.
- 2)일반 가정집에서 하기에는 공간적, 비용적 제약이 많아 접근성이 낮다.



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	야 너도 할수있어 스마트팟(야스)	
	팀 명	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

## 2 개발 내용 및 결과물

### 2.1 목표



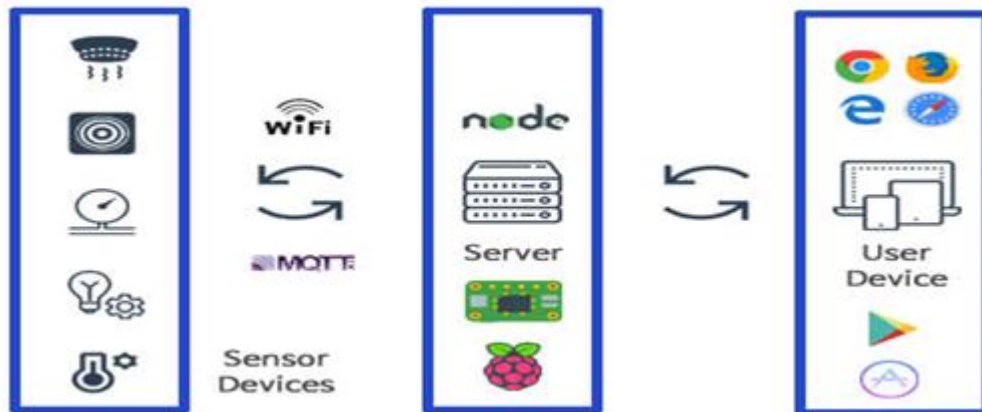
본 프로젝트는 스마트팟을 제작하여 서버를 통해 멀리 있어도 스마트폰으로 현재 상황을 체크하고 컨트롤 할 수 있게 컨트롤 하여 식물이 자랄 수 있게 최적의 상태를 만들어 좀 더 신선한 작물을 재배할 수 있다.

현재 스마트팟의 문제점인 데이터 문제를 해결하기 위하여 농가뿐만 아니라 가정집에서도 온도와 습도 등 식물이 자라는 환경을 조절하고 카메라로 촬영 하여 여러 데이터를 모아 후에 작물이 잘 자랄 수 있는 환경을 확인할 수 있다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	야 너도 할수있어 스마트팟(야스)	
	팀 명	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

## 2.2 연구/개발 내용 및 결과물

### 2.2.1 연구/개발 내용



### 2.2.2 작동 순서

1. 아두이노의 연결된 센서를 통해 얻은 데이터를 라즈베리파이로 보낸 후 AWS IOT 를 이용하여 dynamoDB 에 데이터를 전송한다.
2. AWS IOT 로 받은 데이터를 lambda 서비스를 이용해 어플리케이션으로 데이터 결과값을 보낸다. 이 때 동시에 AWS dynamoDB 에 시간 별로 온도, 습도, 토양 습도의 데이터를 보내 데이터를 관리 할 수 있다.
3. 어플리케이션으로 사용자는 원하는 조작을 하여 그 데이터에 맞게 역순으로 라즈베리파이에 입력된 알고리즘에 맞게 모터들이 가동하여 온도, 습도 ,토양습도를 사용자가 원하는 대로 유지 하게 한다

### 2.2.3 작동 순서 설명

1. 아두이노와 여러가지 센서(온,습도,모터,water pump,Fan,LED, soil moisture,카메라 등) 연결한 뒤 온도,습도,토양습도 센서들로부터 데이터를 받는다. 받은 데이터를 라즈베리파이로 전송한 후 라즈베리파이에 다시 AWS IOT 로 전송한다. 그 후 AWS IOT 에서 dynamoDB 로 보낸다. 이렇게 중간에 라즈베리파이를 쓰는 이유는 아두이노와 AWS IOT 와 바로 연결하는 것은 호환 및 보안 문제가 있어

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

라즈베리파이를 사용한다.


우선 AWS IOT 는 https 로 지원을 하는데 아두이노는 http 만 지원을 한다. 이 뿐만 아니라 인증서 관련 문제점이 있다. 따라서 아두이노와 AWS IOT 를 바로 연결하기 위해서는 중간에 API gateway 와 lambda 기능을 이용해서 인증서와 https 문제점을 해결해야 하는데 이러한 번거러움과 복잡함을 없애기 위해서 중간에 변환기 역할로 라즈베리파이를 거치게 한다. 꼭 이 때문만이 아니라 라즈베리파이와 AWS 는 호환이 잘되어 있어 AWS 의 모든 기능들을 사용하기 편하다.

AWS IOT 에 사물등록을 한 후 사물에 필요한 정책(AWS IAM 서비스)을 정한다. 그 다음 private key, public key, 인증서 이 3 가지를 자동 생성 한 후 다운로드 받고 인증서를 라즈베리파이에 저장한다.

2. 연결된 AWS IOT 를 통해 라즈베리파이에서 전송해온 데이터를 각 분야별로 이용할 수 있다. 우선 AWS lambda 기능이 있는데 lambda 기능이란, 보통의 물리적 서버 또는 가상 서버 기반의 일반적인 접근방법과 다르다. 여러 함수로 구성된 논리적인 구성과 이들 함수를 실행하는 서비스만 신경 쓸 수 있게끔 만들어놓은 서비스이다.

각 함수는 컨테이너에서 실행한다. 각 컨테이너는 운영체제의 커널이 격리된 환경을 구현하는 서버 가상화 방식이기에 이러한 인프라를 관리할 필요가 없어진다. 이러한 접근 방식을 서버리스(serverless)라 부른다. 이 lambda 서비스를 이용해서 어플리케이션에 함수를 전송하여 사용자가 어플리케이션을 통해 AWS IOT 를 관리할 수 있다.

또한, lambda 서비스에서 따로 추가하여 AWS 에서 제공하는 dynamoDB 를 동시에 관리할 수 있다. Dynamo DB 는 noSQL 기반의 AWS 에서 제공하는 DB 이다. 여기서

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

AWS 에 강력한 기능이 나오는데 한곳에서 데이터를 가지고 와서 다른 AWS 기능에서 쉽게 가져와 이용하고 관리를 할 수 있다. 이렇게 lambda 서비스로 사용자가 사용할 어플리케이션에 정보를 제공한다.

3. 이렇게 받은 데이터를 AWS IOT 를 통해 dynamoDB 로 저장을 한 뒤 그 데이터를 어플리케이션을 통해 사용자에게 필요한 정보로 실시간으로 사용자에게 볼 수 있게 한다. 이 것은 네트워크를 통해 정보를 전달하기 때문에 장소에 구애 받지 않고 정보를 이용할 수 있다.

어플리케이션에 보여질 정보들은 센서를 통해 받은 실시간 온도,습도 토양습도의 데이터를 볼 수 있다. 카메라를 이용하여 실시간으로 확인 할 수 있다.


어플리케이션의 개발 환경은 Android Studio 이고 개발언어는 Java 다.

만들어진 어플리케이션을 통하여 사용자가 접속 한 후, 자신이 키우고 있는 스마트팟들을 선택하게 하고 위에서 언급한 농장의 정보들을 보여준다. 또 자동으로 환경들이 조절이 되는 스마트팟이지만, 특수상황을 대비해 원격으로 제어 할 수 있는 버튼들을 제시해 자신이 직접 통제할 수 있게 한다.

## 2.2.2 시스템 기능 요구사항

기능	완료여부
아두이노가 스마트팟 주변 환경에 대한 데이터들을 얻고 라즈베리파이로 전송한다.	완료
변경 전: 라즈베리파이에서 AWS IOT 서비스로 데이터를 전송한 후 lambda 서비스를 이용하여 사용자가 사용하는 어플리케이션으로 함수를 보낸다. 변경 후: 라즈베리파이에서 AWS IOT 서비스로 데이터를 전송 후 이를	완료



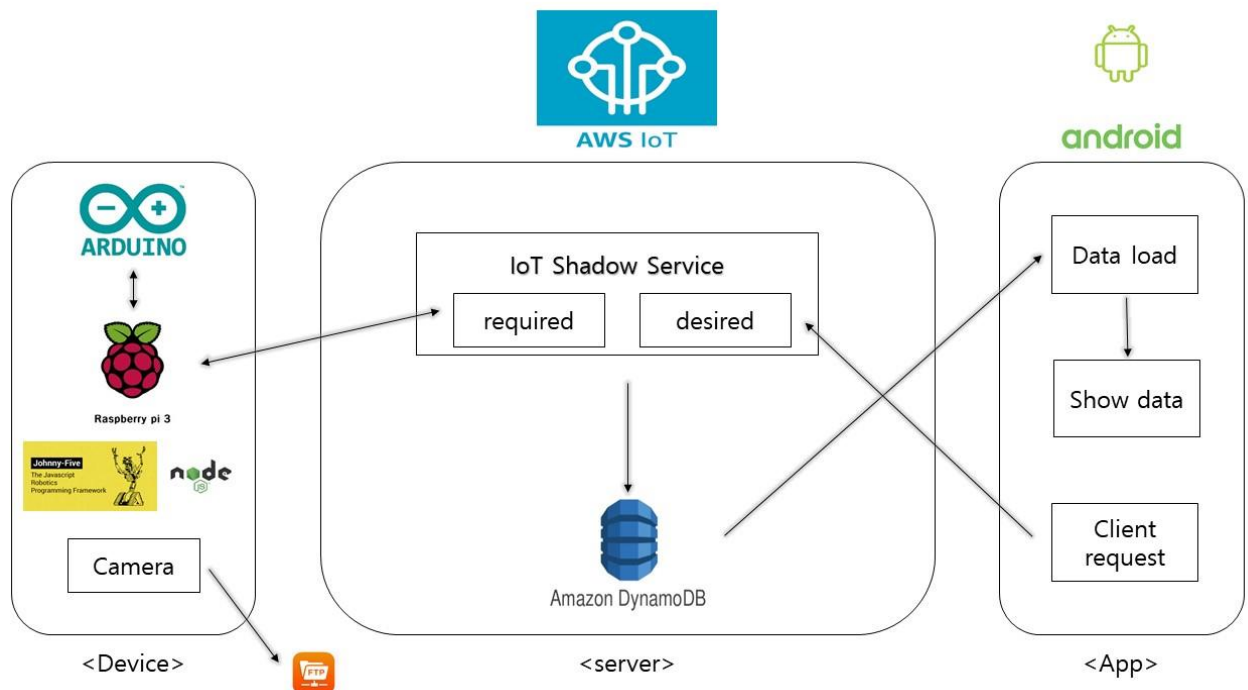
 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

dynamoDB 로 전송하여 데이터를 축적하고 이를 어플리케이션과 연결하여 핸드폰에 띄운다.	
어플리케이션은 사용자에게 AWS IOT 에서 받은 데이터를 UI 를 통해 제공한다.	완료
<p>변경 전: 사용자는 제공받은 UI 를 통해 원하는 작업을 한 후 다시 어플리케이션을 통해 실행 함수를 lambda 서비스로 보낸다.</p> <p>변경 후: 어플리케이션에 AWS IOT 전용 SDK 를 설치하여 AWS IOT 에 신호를 보내 명령을 처리한다.</p>	완료
<p>변경 전: 이를 받은 lambda 서비스는 AWS IOT 서비스를 이용해 다시 라즈베리파이로 보내고 라즈베리파이는 아두이노로 보내 최종적으로 스마트팟에 설치되어 있는 각종 sensor 들을 작동 시킨다.</p> <p>변경 후: AWS IOT 가 어플리케이션에서 신호를 받아 라즈베리파이에 신호를 넘기고, 이를 다시 아두이노로 보내 sensor 들을 작동시킨다.</p>	완료

### 2.2.3 시스템 비기능(품질) 요구사항

기능	달성여부	원인
AWS 서비스를 이용하기 때문에 고성능의 컴퓨터가 필요하진 않지만 AWS 를 이용하기 위한 별도의 요금이 필요하다.	달성	
AWS IOT 로 데이터를 보낼 때 사용하는 하드웨어 장비들이 AWS 와 연동이 가능한지 확인이 필요하며, 어떠한 설정이 필요한지 확인해야 한다.	달성	
이 프로젝트는 하드웨어적인 부분 보다는 기능적인 면이 더 중요시 여겨진다. AWS 에서 어떠한 정보를 받고 어떠한 정보를 저장할 것이며, 어떠한 정보를 제공할 것인지 확인 해야 한다.	달성	

### 2.2.4 시스템 구조 및 설계도



 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

## 2.2.5 활용/개발된 기술

### 1. Johnny-five

- 1) Node.js-Based Open Source IoT/Robotics Programming Framework 이다.
- 2) Event-Driven, 비동기식 프로그래밍을 지원한다.
- 3) 다양 Arduino 호환 보드에서 테스트 되어, 동일한 API 를 제공하여 하나의 플랫폼에서 작성된 코드를 다른 플랫폼에서 수정 없이 사용할 수 있다.
- 4) 지원하는 않는 API 에 대하여 사용자가 Java script 라이브러리를 제작, 확장 및 실험 할 수 있다.
- 5) 다양한 센서/액추에이터의 API 를 제공한다.
- 6) 하드웨어 개발과 동시에 서버 사이드 작업이 쉽게 가능하다.
- 7) REPL 을 통해 JS 스크립트가 실행되고 있을 때, 접근하고자 하는 컴포넌트들의 상태를 쉽게 변경할 수 있다.

### 2. Node.js

- 1) 라즈베리파이에서 node JS 를 이용하여 AWS IOT 서버를 연결하여 Johnny-five 의 이용한 하드웨어 개발을 통한 센서 값을 Json 파일에 저장 dynamoDB 에 전송한다.
- 2) 센서 값에 따른 하드웨어의 모터를 실행하여 센서 값을 유지할 수 있게 만든다.
- 3) 라즈베리파이에 카메라 모듈을 장착하여 하루에 한번씩 jpg 타입의 이미지 파일로 날짜 별 이름으 FTP 서버에 데이터를 축적한다. dynamoDB 에 이미지 데이터를 넣기에는 데이터 량이 크기 때문에 FTP 서버에 축적하기로 하였다.

### 3. dynamoDB

dynamoDB 는 종합 관리형 NoSQL 데이터베이스 서비스로서 원활한 확장성과 함께 빠르고 예측 가능한 성능을 제공한다. dynamoDB 테이블에서 항목및 속성은 작업 필요한 핵심구성 요소이다. 테이블은 항목 집합이고 각 항목은 속성의 집합이다. dynamoDB 테이블의 각

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

항목을 고유하게 식별하고 보조 인덱스를 사용하여 보다 유연하게 쿼리를 작성하도록 해 준다. Dynamo DB 의 기본구성으로는 테이블,항목 ,속성이 있다.


테이블:데이터의 집합이다. 예를 들어, 친구, 가족 또는 기타 관심 있는 사람에 대한 정보를 저장하는 데 사용할 수 있는 people 이라는 예제 테이블을 살펴본다. 또한 cars 테이블에 사람들이 운전하는 차량에 대한 정보를 저장할 수도 있다.

항목: 테이블에는 0 개 이상의 항목이 있다. 항목은 모든 기타 항목 중에서 고유하게 식별할 수 있는 속성들의 집합이다. people 테이블에서 각 항목은 한 사람을 나타낸다. dynamoDB 의 항목은 여러 가지 면에서 다른 데이터베이스 시스템의 행, 레코드 또는 튜플과 유사하다. dynamoDB 에서는 테이블에 저장할 수 있는 항목의 수에 제한이 없다.

속성: 각 항목은 하나 이상의 속성으로 구성됩니다. 속성은 기본적인 데이터 요소로서 더 이상 나뉘어 질 필요가 없는 것이다. dynamoDB 의 속성은 여러 가지 면에서 다른 데이터베이스 시스템의 필드 또는 열과 유사하다.

## 2.2.6 현실적 제한 요소 및 그 해결방안

1. 카메라를 이용하여 식물들 성장을 실 시간으로 녹화하여 확인하려고 했으나, 방대한 데이터를 저장할 공간에 제약이 있다. 보통 바쁜 일상 속에 식물이 자라는 모습을 실시간으로 확인 할 수 없기에 매시간마다 사진 한 장 찍어 FTP 를 통해 데이터를 저장한다.
2. 모든 센서들을 작동 시키기 위해 전기를 24 시간 공급해야 된다.  
실외에 설치 할 수 없어서 실내에 설치하고, 환경을 일정하게 유지하기 위해 아크릴로 된 조그만 케이스에 화분을 설치한다.

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

## 2.2.7 결과물 목록

이름	기능	형식
AWS IOT	아두이노 센서에서 라즈베리파이를 통하여 AWS IOT 로 데이터 전송	
AWS dynamoDB	AWS IOT 에서 AWS dynamoDB 로 전송	
라즈베리파이	아두이노에서 라즈베리파이로 센서값 전송	
아두이노	온도,습도,토양 습도,팬 작동	
온열 매트	전력부족으로 인하여 작동불가	
Stepper Motor	커튼 열고 닫기 가능	

## 2.3 기대효과

### 1. 기대효과

#### 1) 원격 조종 가능

핸드폰으로 현재상황을 체크하고 컨트롤 할 수 있어 부재 시에도 작물을 관리 할 수 있다

#### 2) 내가 키우는 작물


남이 심고 키운 것이 아니라 자신이 직접 키운 것을 섭취할 수 있다.

#### 3) 작물 데이터 수집

가정에서 소규모로 채소나 과일 등을 각각 케이스 별로 데이터를 수집할 수 있다.

## 3 자기평가

우리는 바쁜 현대인들이 집에서 식물이나 작물을 기르며 삶의 여유를 찾고 믿을만한 과일 및 채소를 얻기 위한 스마트팟을 주제로 프로젝트를 시작하였다. 요즘 바쁜 도시 사람들은 작물을 기르거나 키우는 것이 어렵다고 느끼고 거리감을 느끼기 마련이다. 그래서 현대의 맞는 스마트폰으로 작물을 쉽게 관리 할 수 있게 꿈 하여 거리감을 좁히고 편하게 다가가게 하는 게 목표였다.

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

아두이노와 센서를 연결하여 센서를 통해 온도,습도,토양습도 등 데이터를 받아서 아두이노를 통해 라즈베리파이로 데이터를 보낸다. 이때 아두이노에서 johnny-five 라는 API 를 이용하였다. 아두이노에서 바로 AWS IOT 로 보낼 수가 없어서 라즈베리파이를 거쳐 보냈는데 이 과정이 조금 복잡하였다. AWS IOT 를 통해 데이터를 AWS 에서 제공하는 dynamoDB 에 저장하였는데 이때 방식이 일반적으로 알고 있는 mySQL 방식이 아닌 noSQL 방식이었다. 처음 접해 보는 DB 방식이라 처음 데이터를 저장하는데 힘들었고 많은 공부가 되었다.

프로젝트를 진행하면서 하드웨어부분, AWS 부분,어플리케이션 부분으로 파트를 나눠서 진행하였는데 취업하기 전 프로젝트는 많이 진행해보는 것이 좋을 것 같다는 생각이 들었다.

## 4 참고 문헌

번호	종류	제목	출처	발행년도	저자	기타
1	서적	AWS Lambda 인 액션	제이펍	2017	다닐로 포치아	
4	서적	아마존 웹 서비스 AWS Discovery Book	정보문화사	2019	권영환	
3	서적	(Do it!) 안드로이드 앱 프로그래밍 :	이지퍼블리싱	2015	정재곤	
4	서적	아마존 웹 서비스 A to Z : start-up 기업 및 글로벌 서비스를 위한 AWS 입문서	북랩	2014	이근우	

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	야 너도 할수있어 스마트팟(야스)	
	팀 명	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

## 5 부록

### 5.1 사용자 매뉴얼

- 1) 스마트팟 설치
  1. 스마트팟 케이스를 집안에 설치한다.
  2. 각종 센서들이 동작할수있게끔 배터리를 넣는다.
  3. 라즈베리파이에서 'node my\_farm.js' 명령어를 입력한다.
  4. cmd 창에 'connect'가 뜨는지 확인한후 led 조명과 fan 이 작동하는지 확인한다.
- 2) 동작확인 및 데이터 수집
  1. cmd 창에 온도와 습도, 토양습도까지 잘 나타는지 확인한다.
  2. 어플리케이션을 통해 현재 온도,습도 등이 잘 나타나는지 확인한다.

### 5.2 운영자 매뉴얼

- 1) AWS dynamoDB 데이터 및 사진 데이터 관리

스마트팟을 설치한 각 가정마다 수집된 DB 데이터와 카메라를 통해 찍힌 작물 사진 데이터를 관리한다. 클라이언트가 필요할 때 데이터를 제공할수 있도록 관리한다.

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	야 너도 할수있어 스마트팟(야스)	
	<b>팀 명</b>	귀농	
	Confidential Restricted	Version 1.2	2019-05-27

## 5.3 테스트 케이스

대분류	소분류	기능	테스트 방법	기대 결과	테스트 결과
Hardware	라즈베리파이	아두이노를 통해 들어온 데이터를 AWS IOT 로 전송한다.	라즈베리파이에서 센서들의 값을 받아오는 node.js 파일을 작성하고 작성된 파일내부에 연결이 성공하면 'connect'라는 문구가 뜨게끔 설정한다..	센서들을 작동시키는 js 파일을 실행시켜 'connect'라는 문구가 나온다.	성공
AWS	dynamoDB	센서들의 데이터값을 시간대별로 저장한다.	라즈베리파이에서 실행하는 node.js 파일에는 AWS IoT 로 데이터를 보내는 코드가 같이 작성되어 있어 js 파일을 실행 시킨후 AWS dynamoDB 를 새로고침을 통해 확인한다.	dynamoDB 테이블에 partitionkey 에는 년,월,일이 들어가고 sort key 에는 시간대가 들어간다.payload 에는 각 센서들의 데이터값이 저장된다.	성공
어플리케이션	안드로이드 스튜디오	Aws iot 를 중간매개체로 이용하여 데이터를 아두이노로 전송하고 읽어온다.	앱을 실행해서 설정데이터(온도, 토양습도)를 주면 Aws IOT 에 데이터가 기록되고 현재데이터(현재온도, 현재토양습도)를 받아와 실시간으로 띄운다.	Aws iot 로부터 실시간 데이터 읽기 AWS IOT 에 설정 데이터 기록	성공