

사용자 요구사항 명세서

User Requirement Specification

사후부착용 스마트 LED 조명제어 시스템
(Smart LED lighting control system for post-adhesion)



팀명	미정
담당 교수	주홍택 교수님
팀원	권우영, 정유진, 최혜리, 이효엽

목차

1.	URS 소개 -----
1.1.	개요
1.2.	목적
1.3.	범위
1.4.	사용대상
1.5.	용어 정의
1.6.	참고 문헌
2.	시스템 개요 -----
2.1.	시스템 개요
2.2.	시스템 목적
2.3.	시스템 범위
2.4.	시스템 고려사항
2.5.	시스템 완성예상도
3.	기능 요구 사항 -----
3.1.	유스케이스 다이어그램
3.2.	유스케이스 시나리오
3.3.	기능 구조사항
3.4.	상세 기능 요구사항
4.	인터페이스 요구사항 -----
4.1.	사용자 인터페이스 요구사항
4.2.	네트워크 인터페이스 요구사항
4.3.	소프트웨어 인터페이스 요구사항
4.4.	하드웨어 인터페이스 요구사항
4.5.	보안 요구사항
5.	제약사항 -----
5.1.	일반적 제약사항
5.2.	설계 및 구조적인 제약사항
6.	가정 및 의존성 -----
7.	소프트웨어 품질 요구사항 -----

1. URS 소개

1.1 개요

* 본 문서는 사후부착용 스마트 LED 조명제어 시스템 개발에 대한 요구사항을 명세 한다.

1.2 목적

- * 본 문서의 목적은 사용자가 전등 제어를 기존보다 쉽고 싸게 할 수 있다. 그리고 본 문서에서는 탈부착 스마트 전등 제어 개발을 위한 요구사항을 명세하고 있다.
- * 본 문서는 고객, 담당자를 대상으로 한다.
- * 본 문서는 IoT 스마트 전등 개발을 내용을 통해 고객의 요구사항을 명확하게 도출 하여 향후 개발 과정에서 이를 반영하는데 그 목적이 있다. 따라서 본 문서는 고객의 정확한 요구사항을 수집하고 이를 분석하여 명세 한다.
- * 본 문서는 고객과 개발자 간에 다음과 같은 역할을 한다.
 - 1) 본 문서는 고객과 개발자간의 계약서와 동일한 효력을 갖는다
 - 2) 본 문서는 고객 요구사항을 구체적으로 명시한다.
 - 3) 개발자는 본 문서에 명세 된 고객의 요구사항에 따라 프로토타입을 개발한다.
개발된 목적물은 본 문서에 명세 된 모든 요구사항을 만족해야 한다.
 - 4) 본 문서는 향후 테스트의 베이스 라인이 된다.
 - 5) 본 문서는 명세된 모든 요구 사항을 바탕으로 설계하고, 테스트 케이스를 작성한다.

1.3 범위

- * 프로젝트 명 : 캡스톤 디자인(사후부착용 스마트 LED 조명제어 시스템 개발)
- * 이번 캡스톤 디자인 프로젝트를 통해서 IoT를 이용한 스마트 LED 조명제어 시스템을 개발하는 것을 주안점으로 두고 실시한다. 우선 WIFI와 BLUETOOTH를 통하여 여러개의 LED를 원격 조정하는 것을 우선점으로 보고 개발한 후, 추후에 다양한 기능을 넣어서 심미성과 기능성을 두루 갖춘 제품으로 만드는 것을 지향하는 방향으로 프로젝트를 진행할 것이다.
- * 프로젝트 요약
 - 스마트 디바이스를 통해서 LED 전등을 원격 조정한다
 - 최종적으로 원격 조정뿐만 아닌 색깔, 밝기조정 및 날씨확인 까지 가능한 프로토타입을 제작한다

1.4 사용대상

구분	이름	역할	연락처
팀장	권우영	프로젝트관리	010-5918-7699
팀원	정유진	안드로이드개발	010-9432-8246
팀원	최혜리	아두이노개발	010-5630-3970
팀원	이효엽	하드웨어개발	010-9409-7682
지도 교수님	주홍택	프로젝트 지도	010-5456-1890

1.5 용어 정의

용어 이름	머리 글자	약어	정의
Power Supply	PS	-	안정적인 전원을 공급하는 전원 공급장치
Use Case	UC		사용자의 시점을 기반으로한 사용 시나리오를 서술
Internet of Things	IoT		사물 인터넷
User Requirement Specification	URS		사용자 요구 명세서
Smart LED lighting control system for post-adhesion	SLED		본 제품의 명칭

1.6 참고 문헌

순번	작성자	문서이름	문서 종류	문서 형식	버전	발행일
1	김재호외 3명	IoT 플랫폼 개발 동향 및 발전방향	논문	pdf		2013

2. 시스템 개요

2.1 시스템 개요

스마트 전등은 아두이노를 통해 전등을 쉽게 조작할수 있는 기능으로 여러 액터들(관리자,사용자)로 구성되어 있다. 사용자는 색의 변화, on/off의 조절, 예약, 밝기의 조절 등을 쉽게 할수 있다. 또한 날씨 표시, 감지 등의 기능이 있다.

2.2 시스템 목적

기존의 스위치를 통해 ON/OFF를 조절하는 전등의 방식을 편리하게 바꾸고 많은 사람들이 이용할 수 있는 보급형 LED전등의 제어장치를 제작한다. 국내 IoT 시장 규모도 2015년 4.7조원에서 41.5% 증가한 6.6조원(2017년), 2020년에는 17 조원까지 성장이 예상된다. 이러한 통계결과에서 보여주듯이 앞으로 IoT 시장은 점점 확장되고 보급될것이기 때문에 IoT기술을 이용하여 제품을 개발하게 되었다. 그리고 개발한 제품을 시장에 내어놓음으로써 기존 IoT제품들의 가격대를 낮추고 대중들의 IoT제품에 대한 접근성을 높려 보다 많은 사람들이 IoT제품들을 접할 기회를 가지게 한다. 또한 IoT제품을 이용하게 될 구매자들의 삶의 질을 향상시킴으로써 IoT 제품들을 활용한 사회를 형성할 기반을 다진다.

2.3 시스템 범위

* 개발 정의

* IoT LED 전등은 현재 시중에도 몇몇 제품들이 출시되어있다. 하지만 비싼가격과 접근성으로 인하여 아직 대중화 되지 못하고 있는 것이 현실이다. 전 연령 및 계층의

사용자들이 쉽고 편하게 IoT 기술을 적용한 LED를 사용할수 있게 하는 것이 목표다.

* 아직 IoT같은 시스템에 적응이 안된 사람들을 위해 간단하고 쉽게 사용할수 있도록

어플리케이션을 적용한다.

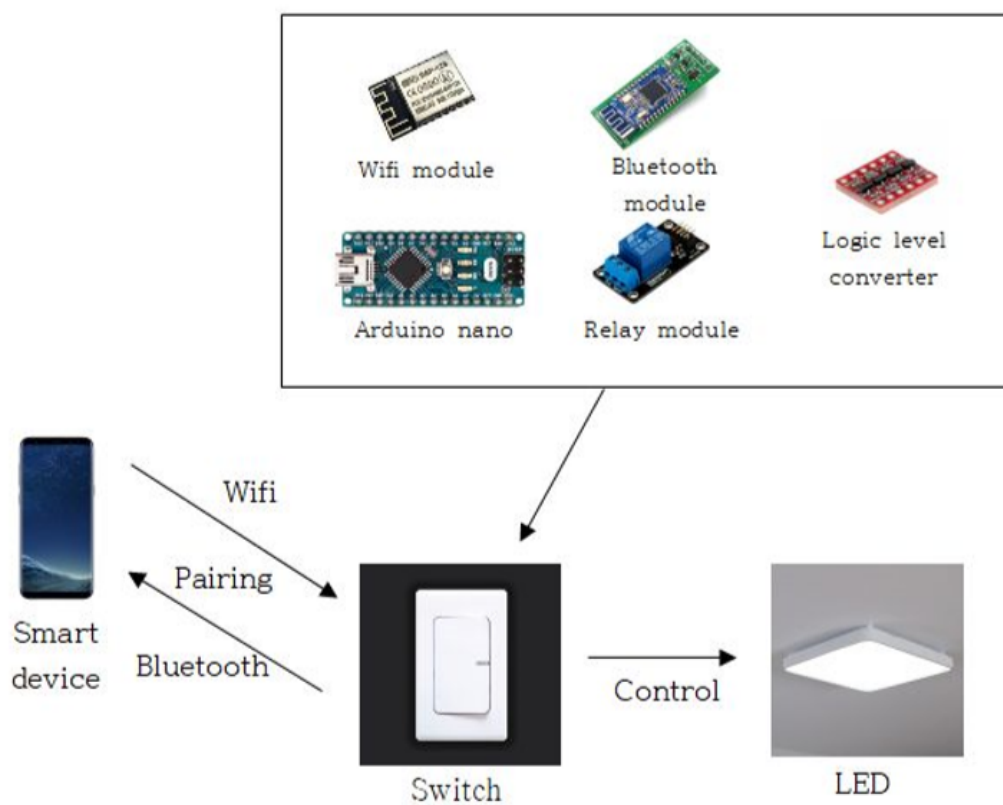
2.4 시스템 고려사항

* 원격조정을 통해서만 LED를 조절 할 때 스마트 디바이스가 없을경우 조절이 불가능하므로, 직접제어를 통해서도 LED를 조절할수 있게한다.

* 기상청 서버의 정보를 가져올경우에 기상청 서버가 다운되거나 막힐수도있으니 다른 경로도 파악해둔다.

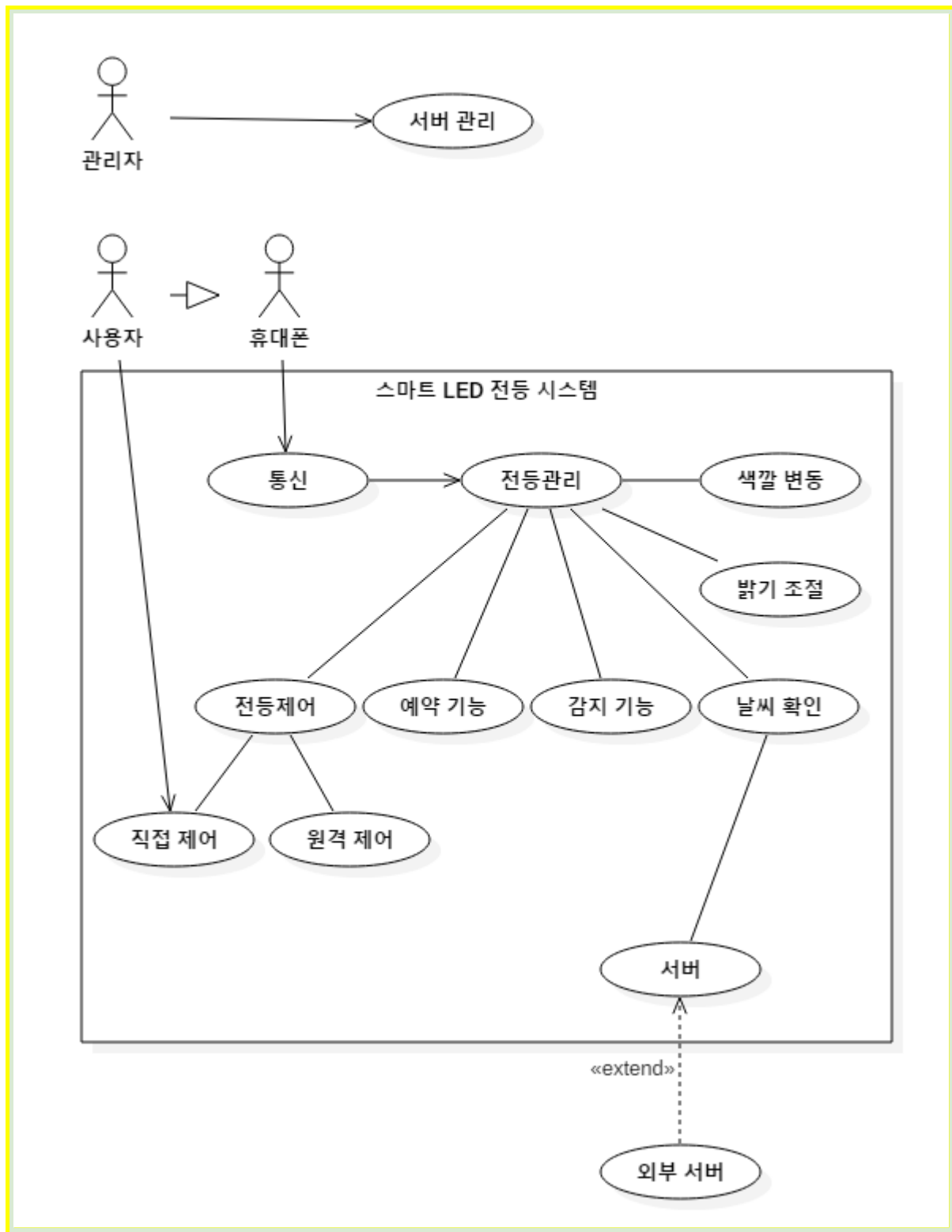
* 다양한 사용자들이 사용하는데 불편함이 없도록 개발자는 최대한 직관성있게 제작해야 한다.

2.4 시스템 완성 예상도



3. 기능요구사항

3.1 유스케이스 다이어그램



3.2 유스케이스 시나리오

Use Case	UC1
----------	-----

이름	페어링
개요	사용자가 본 제품에 연결을 한다.
액터	사용자
사전조건	사용자는 안드로이드 어플리케이션을 다운받아 설치한 상태이다.
기본 흐름	Bluetooth나 Wifi를 이용하여 본제품과 페어링을 한다.

Use Case	UC2
이름	페어링 실패
개요	사용자가 본 제품에 연결을 실패한다.
액터	사용자
사전조건	사용자는 안드로이드 어플리케이션을 다운받아 설치한 상태이다.
기본 흐름	1. Bluetooth나 Wifi를 이용하여 본제품과 페어링을 한다. 2. 페어링을 실패한다. 3. Bluetooth나 Wifi를 이용하여 본제품과 페어링을 다시 시도 한다.

Use Case	UC3
이름	원격 조종
개요	원격으로 전등의 스위치를 ON/OFF 시킨다.
액터	사용자
사전조건	안드로이드 어플리케이션을 설치한 상황이고 페어링을 성공한 상태이다.
흐름	1. 사용자는 메인 화면의 ON/OFF 버튼을 누른다. 2. ON/OFF 화면으로 넘어간다. 3. ON 과 OFF중 원하는 버튼을 눌러 전등의 상태를 변경한다. 4. 시스템에서 LED 현재 상태에 정보를 전달한다. 5. LED 현재 상태가 LED에게 전달한다. 6. LED가 꺼지거나 켜진다.

Use Case	UC4
-----------------	-----

이름	밝기 제어
개요	전등의 밝기를 제어한다.
액터	사용자
사전조건	안드로이드 어플리케이션을 설치한 상황이고 페어링을 성공한 상태이다.
기본 흐름	<ol style="list-style-type: none"> 1. 사용자는 메인 화면의 밝기 조절 버튼을 누른다. 2. 밝기 조절 화면으로 넘어간다. 3. 원하는 밝기로 전등을 조절한다. 4. 시스템에서 LED 현재 상태에 정보를 전달한다. 5. LED 현재 상태가 LED에게 전달한다. 6. LED가 전등의 밝기를 조절한다.

Use Case	UC5
이름	전등 색 변경
개요	전등의 색깔을 변경한다.
액터	사용자
사전조건	안드로이드 어플리케이션을 설치한 상황이고 페어링을 성공한 상태이다.
기본 흐름	<ol style="list-style-type: none"> 1. 색깔 변경 화면으로 넘어간다. 2. 사용자는 바꾸고 싶은 색을 결정한다. 3. 시스템에서 LED 현재 상태에 정보를 전달한다. 4. LED 현재 상태가 LED에게 전달한다. 5. LED가 전등의 색을 변경한다.

Use Case	UC6
이름	날씨 표시
개요	그날의 날씨를 표현한다.

액터	사용자
사전조건	안드로이드 어플리케이션을 설치한 상황이고 페어링을 성공한 상태이다.
기본 흐름	<p>기능의 사용</p> <ol style="list-style-type: none"> 1. 사용자는 메인 화면의 날씨 버튼을 누른다. 2. 날씨 화면으로 넘어간다. 3. ON/OFF 버튼을 눌러 기능을 사용할지를 정한다. <p>사용 중</p> <ol style="list-style-type: none"> 1. 서버에 저장된 정보를 확인한다. 2. 확인한 정보를 시스템에 전달한다. 3. 시스템에서 LED 현재 상태에 정보를 전달한다. 4. LED 현재 상태가 LED에게 전달한다. 5. LED가 날씨를 표시한다.

Use Case	UC7
이름	감지
개요	일정 시간 동안 사람이 없을 경우 전등이 자동으로 꺼진다.
액터	사용자
사전조건	안드로이드 어플리케이션을 설치한 상황이고 페어링을 성공한 상태이다.
기본 흐름	<p>기능의 사용</p> <ol style="list-style-type: none"> 1. 사용자는 메인 화면의 감지 버튼을 누른다. 2. 감지 화면으로 넘어간다. 3. ON/OFF 버튼을 눌러 기능을 사용할지를 정한다. <p>사용 중</p> <ol style="list-style-type: none"> 1. 전등이 켜져 있는 상태에서 사용자가 멀리 떠나간다. 2. 전등과 연결되어 있던 페어링이 끊긴다. 3. 시스템에서 LED 현재 상태에 정보를 전달한다. 4. LED 현재 상태가 LED에게 전달한다. 5. LED 전등이 꺼진다.

Use Case	UC8
이름	ON/OFF 예약
개요	전등의 ON/OFF를 예약한다.

액터	사용자
사전조건	안드로이드 어플리케이션을 설치한 상황이고 페어링을 성공한 상태이다.
기본 흐름	<ol style="list-style-type: none"> 1. 사용자는 메인 화면의 예약 버튼을 누른다. 2. 예약 화면으로 넘어간다. 3. 켜거나 끄고 싶은 시간을 설정한 후 서버에 저장을 한다. 4. 저장한 정보를 시스템에게 보낸다. 5. 시스템은 LED 현재 상태에 정보를 전달한다. 6. LED 현재 상태가 LED에게 전달한다. 7. 예약된 정보에 따라 전등이 켜지거나 꺼진다.

Use Case	UC9
이름	서버 관리
개요	서버를 관리한다.
액터	관리자
사전조건	관리자는 서버에 접근 할 수 있다.
기본 흐름	<ol style="list-style-type: none"> 1. 데이터를 서버에 저장한다. 2. 데이터를 보관한다. 3. 쓸모 없는 데이터를 삭제한다.

Use Case	UC10
이름	LED 현재 정보 전달 실패
개요	기능의 정보를 전달하는 것을 실패한다.
액터	LED
사전조건	안드로이드 어플리케이션을 설치한 상황이고 페어링을 성공한 상태이다.
기본 흐름	<ol style="list-style-type: none"> 1. 전등 기능에 관련된 정보를 LED에게 전달한다. 2. 전달을 실패한다. 3. 전등 기능에 관련된 정보를 LED에게 다시 전달한다.

Use Case	UC11
이름	날씨 정보
개요	날씨 정보를 서버에 저장한다.
액터	관리자
사전조건	서버에 정보를 저장 할 수있고 서버를 관리 할 수있다.
기본 흐름	<ol style="list-style-type: none"> 1. 외부서버에서 날씨 정보를 받아온다. 2. 서버에 날씨 정보를 저장한다. 3. 저장에 실패한다면 다시 받아온 후 저장한다. 4. 저장 한 정보를 시스템에 전달한다.

Use Case	UC12
이름	예약 정보
개요	예약 정보를 서버에 저장한다.
액터	관리자
사전조건	서버에 정보를 저장 할 수있고 서버를 관리 할 수있다.
기본 흐름	<ol style="list-style-type: none"> 1. 사용자가 지정한 예약 정보를 받아온다. 2. 서버에 예약 정보를 저장한다. 3. 저장에 실패한다면 다시 받아온 후 저장한다. 4. 저장 한 정보를 시스템에 전달한다.

3.3 기능 구조사항

비즈니스 요구사항	목적	우선도
ON/OFF 예약	예약한 시간에 맞춰서 전등 ON/OFF	2
밝기 조절	LED 내부의 파워서플라이의 전압을 조정하여 밝기 조절	3
색깔 변경	LED의 색깔을 사용자의 기분이나 음악 장르에 따라 조절	4
날씨표시	전등의 색깔이나 외부의 디스플레이를 이용해 그날의 날씨를 표시	5
감지	일정 시간 동안 사람이 없을 경우 전등이 자동 OFF	6
원격 조정	Bluetooth 나 WI FI를 이용하여 원격으로 전등의 스위치를 ON/OFF	1

3.4 상세 기능 요구사항

요구사항 번호	SL-001	요구사항 이름	ON/OFF 예약	요구사항 유형	기능
작성자	정유진	작성일	2018.11.9	버전	V1.0
내용	1. 사용자로부터 예약을 받아 시간이 되면 ON/OFF를 처리한다.				
입력데이터	사용자의 예약 시간		출력데이터	전등의 ON/OFF	
입출력유형	외부 입력		파일유형	DB 저장	
평가방법	1. 예약한 시간에 맞춰서 전등이 꺼지거나 켜지는지 확인한다.				
평가기준	정확성, 신뢰성				
우선순위	상	중요도	상	위험	하

요구사항 번호	SL-002	요구사항 이름	밝기 조절	요구사항 유형	기능
작성자	권우영	작성일	2018.11.9	버전	V1.0
내용	1. LED의 파워 서플라이를 조절하여 밝기를 조절한다.				
입력데이터	20~100 사이의 정수값		출력데이터	전등의 밝기	
입출력유형	int		파일유형	-	
평가방법	1. 값을 주었을때 밝기가 변화하는지 확인한다.				
평가기준	정확성, 신속성				
우선순위	중	중요도	상	위험	하

요구사항 번호	SL-003	요구사항 이름	색깔 변경	요구사항 유형	기능
작성자	최혜리	작성일	2018.11.9	버전	V1.0
내용	1. RGB 색상 값을 입력받아 전등의 색을 변경한다.				
입력데이터	색		출력데이터	전등 색	
입출력유형	setColor()		파일유형	-	
평가방법	1. 원하는 색을 지정한 후 색을 확인한다.				
평가기준	정확성, 신속성				
우선순위	중	중요도	중	위험	중

요구사항 번호	SL-004	요구사항 이름	날씨 표시	요구사항 유형	기능
작성자	이효엽	작성일	2018.11.10	버전	V1.0
내용	1. 기상청으로부터 날씨 정보를 가져온다. 2. 날씨정보를 LED의 색깔및 디스플레이로 나타낸다.				
입력데이터	기상청(www.kma.go.kr)		출력데이터	날씨	
입출력유형	외부기기		파일유형	-	
평가방법	1. 서버의 정보를 알맞게 가져오는지 확인한다.				
평가기준	정확성				
우선순위	하	중요도	하	위험	상

요구사항 번호	SL-005	요구사항 이름	감지	요구사항 유형	기능
작성자	정유진	작성일	2018.11.10	버전	V1.0
내용	블루투스와의 통신이 끊기고 일정시간이 흘렀을때 전등을 OFF 한다.				
입력데이터	Bluetooth 신호		출력데이터	수신X시 일정시간후 OFF	
입출력유형	Bluetooth		파일유형	-	
평가방법	1. 기기에 블루투스의 등록이 되는지 확인한다. 2. 블루투스와의 통신 거리를 확인한다.				
평가기준	정확성, 신뢰성, 보안성				
우선순위	중	중요도	중	위험	중

요구사항 번호	SL-006	요구사항 이름	원격조정	요구사항 유형	기능
작성자	정유진	작성일	2018.11.11	버전	V1.0
내용	1. 사용자에게서 ON/OFF를 입력받으면 전등을 ON/OFF 한다.				
입력데이터	WIFI 수신 정보		출력데이터	ON/OFF	
입출력유형	boolean		파일유형	-	
평가방법	1. ON/OFF를 조절할 때 마다 제대로 데이터가 수신되는지 확인한다.				
평가기준	정확성, 신뢰성, 사용성				
우선순위	상	중요도	상	위험	상

4. 인터페이스 요구사항

4.1 사용자 인터페이스 요구사항

사용자 인터페이스 이름 : 똑똑이전등

인터페이스목적 : 모바일 어플리케이션을 통해 사용자가 전등을 편리하게 사용할 수 있게 하기 위함

4.2 네트워크 인터페이스 요구사항

블루투스나 기기간의 안정적인 통신을 요구한다.

웹서버와 안드로이드간에 안정적인 통신이 가능해야 한다.

4.3 소프트웨어 인터페이스 요구사항

Java 언어를 이용하여 안드로이드 스튜디오를 통한 어플리케이션을 제작한다.

SQLite를 통하여 DB 서버를 구축한다.

4.4 하드웨어 인터페이스 요구사항

저렴한 가격의 기기를 사용하여 제작한다.

낮은 전력으로 전등을 제어한다.

Module	Model	Specification
Arduino	NANO	Microcontroller ATmega328
Wifi	ESP8266 ESP-12F	802.11b/g/n Wi-Fi SOC
Bluetooth	HM-10 BLE	2.4GHz ISM band
Relay	robotdyn 1 relay, 5v	10A (250VAC/60VDC)
converter	NER-15966	LV(3.3v) HV(5.5v)

4.5 보안 요구 사항

기기에 등록된 블루투스만 인식 및 연결함으로써 사용자를 구별

5. 제약사항

5.1 일반적 제약 사항

- LED전등이 아닌 백열전구에서는 사용이 불가능하다.
- 전등을 직접제어하는 스위치의 모양에 따라 사용이 불가능 할 수 있다.

5.2 설계 및 구조적인 제약사항

- DB서버 구성에 SQLite를 이용한다.
- 어플리케이션 구현에는 안드로이드 스튜디오와 Java를 사용한다.
- 제품의 단가를 낮추기위해 저렴한 기기를 사용한다.

6. 가정 및 의존성

본 제품은 일반 가정집이나 원룸에서의 사용을 생각하고 제작하였지만 차후 공공장소에서도 쓰일수 있음을 고려한다.

7. 소프트웨어 품질 요구사항

- 1) 사용성 : 남녀노소 상관없이 쉽게 사용가능 해야한다.
- 2) 가용성 : 시스템에 장애가 발생할 경우 유지보수자가 시스템 장애의 원인을 10분 이내에 찾을 수 있어야 한다.
- 3) 가독성 : 프로그램의 사용방법을 쉽게 알아볼 수 있게 해야한다.
- 4) 신뢰성 : LED를 제어하는 버튼이 확실하게 작동하여야 한다
- 5) 보안성 : 블루투스 또는 WiFi 연결시 사용자가 아닌 사람의 접근을 금지한다
- 6) 이식성 : 다른 전등에서도 쉽게 사용가능해야한다.
- 7) 유지보수성 : 소프트웨어 시스템에 오류가 발생할시 해결 가능해야하고 기상청의 날씨 정보 시스템이 업그레이드 될시 버전업을 하여서 날씨 정보를 불러와야한다.