

스마트 도어락

김종태 / 전상규 / 조영기

목차

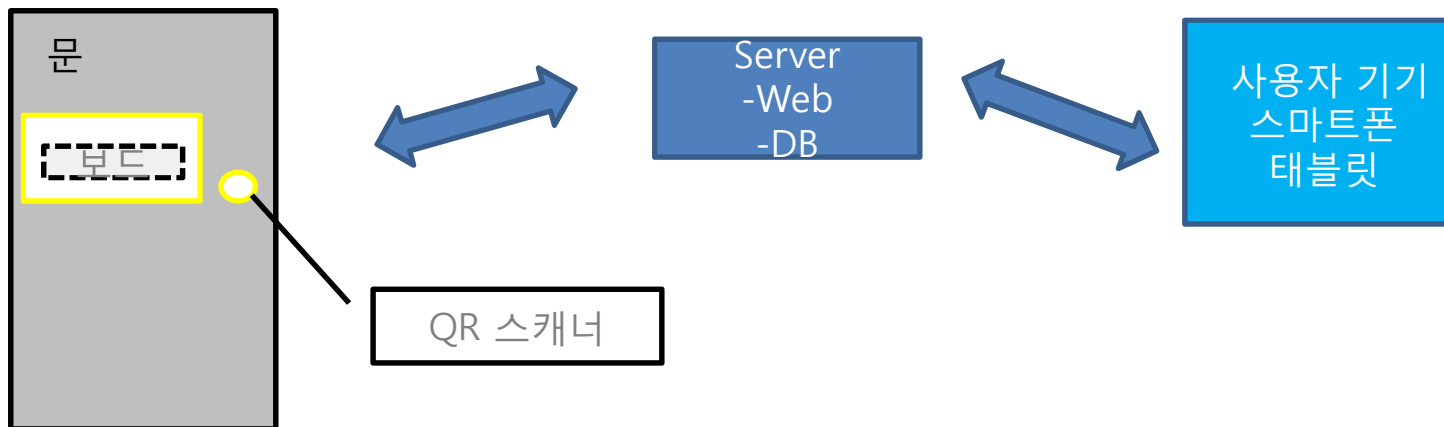
1. 아이디어 설명
2. 요구사항 분석
3. 하드웨어 설계
4. 하드웨어 테스트
5. 개발 환경
6. 화면 구성
7. 시나리오
8. 유스케이스
9. 테스트케이스

1. 아이디어 설명 – 스마트 도어락

기존 시스템상에서는 호텔 등 체크인이 필요한 장소를 이용할 때, 이용자는 객실 키를 받기 위해 반드시 관리자와 만나야 한다.

또 관리자의 입장에서는 많은 키를 관리해야 하기 때문에 관리의 어려움과 분실의 위험성을 가지고 있다.

이런 불편한 과정을 개선하기 위해 사용자가 가지고 있는 모바일 기기를 키로 사용하는 시스템을 구현해보려 한다.



2. 요구사항 분석

호텔 측

1.1 예약

- 1.1.1 이용자들의 예약현황을 웹을 통해 확인할 수 있다.
- 1.1.2 이용자들의 예약정보를 활용하여 객실을 조정할 수 있다.

1.2 상태조회

- 1.2.1 이용자의 입실 퇴실 여부를 관리자가 확인할 수 있다.
- 1.2.2 도어락의 상태를 관리자가 확인할 수 있다.
- 1.2.3 각 도어락의 인증 내역을 확인할 수 있다.

1.3 관리

- 1.3.1 객실 관리를 위해 마스터키가 제공되어야 한다.
- 1.3.2 바르게 인증된 사용자만 이용할 수 있도록 한다.
- 1.3.3 객실 보안을 위해 인증 과정에 암호화된 인증이 필요하다.

2. 요구사항 분석

이용자 측

2.1 예약

- 2.1.1 원하는 시간과 장소를 온라인이나 앱을 통해서 확인한다.
- 2.1.2 원하는 시간과 장소를 온라인이나 앱을 통해서 예약한다.
- 2.1.3 예약 후 입실시간이 되었을 때 키를 전송해준다.

2.2 예약 취소

- 2.2.1 자신이 예약한 내용을 조회할 수 있다.
- 2.2.2 조회한 예약을 취소할 수 있다.
- 2.2.3 취소한 예약에 따른 비용은 환불받는다.

2.3 예약 변경

- 2.3.1 자신이 예약한 내용을 조회할 수 있다.
- 2.3.2 조회한 예약을 변경 할 수 있다.
- 2.3.3 변경 시 금액 변동분은 추가로 납부 또는 환불이 가능하다.

2. 요구사항 분석

이용자 측

2.4 앱사용

- 2.4.1 오픈마켓을 통해 다운로드가 가능하다.
- 2.4.2 어플리케이션이 해킹되지 않도록 보안이 되어있다.
- 2.4.3 회원가입시 전화번호를 통한 회원가입을 할 수 있다.
- 2.4.4 핸드폰 분실 시 새로운 핸드폰으로 정보를 옮길 수 있다.
- 2.4.5 예약, 예약취소, 예약변경을 할 수 있다.
- 2.4.6 사용자가 물리적인 열쇠 없이 문을 열 수 있다.
- 2.4.7 적절한 보안을 제공해준다.
- 2.4.8 예약정보를 조회할 수 있다.

2.5 입실

- 2.5.1 도어락 카메라를 통해 키를 확인 할 수 있다.
- 2.5.2 서버와 통신을 하여 사용자가 맞는지 확인한다.
- 2.5.3 사용자가 받은 키를 통해서 문이 열린다.

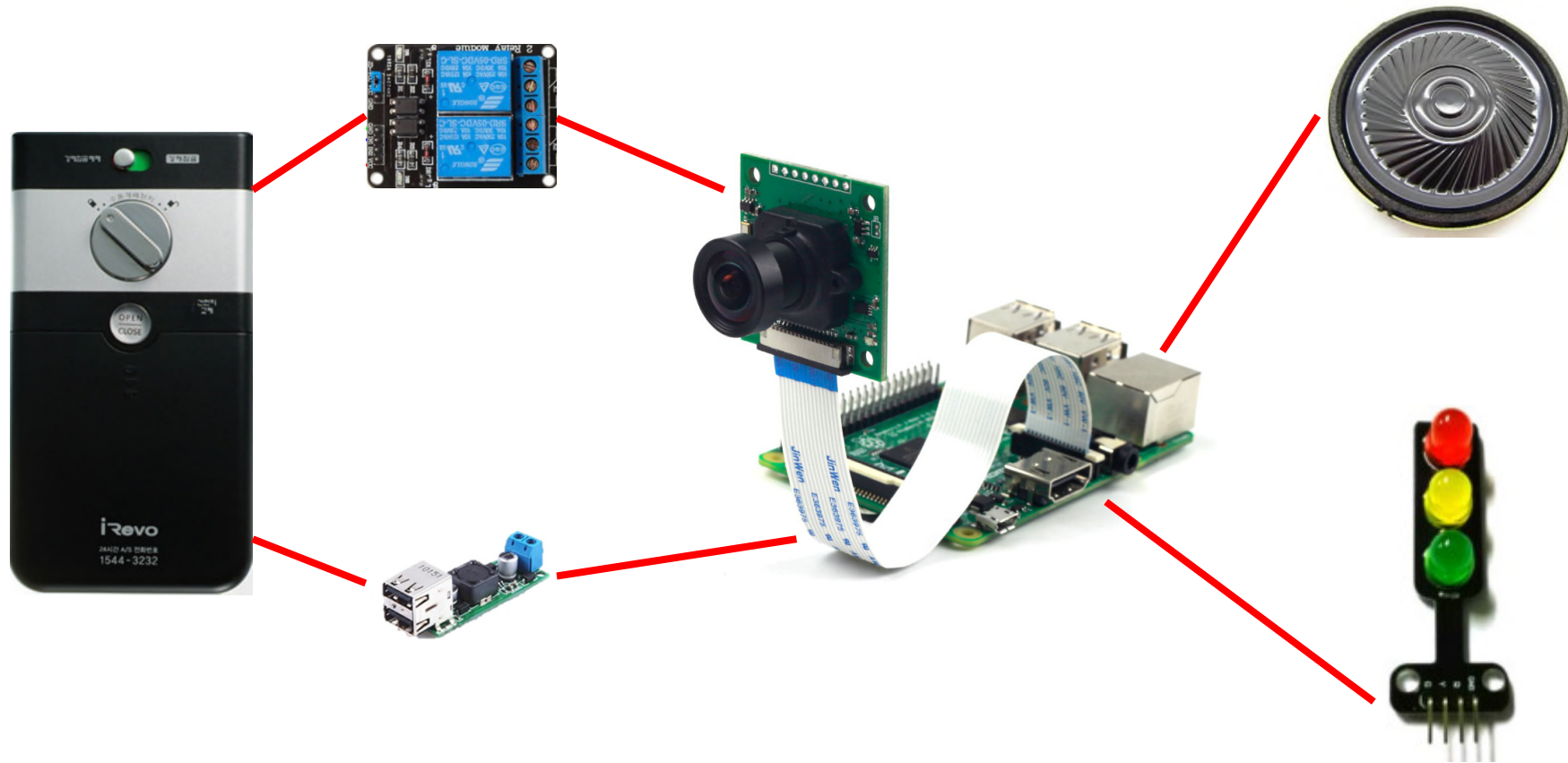
2. 요구사항 분석

이용자 측

2.6 보안

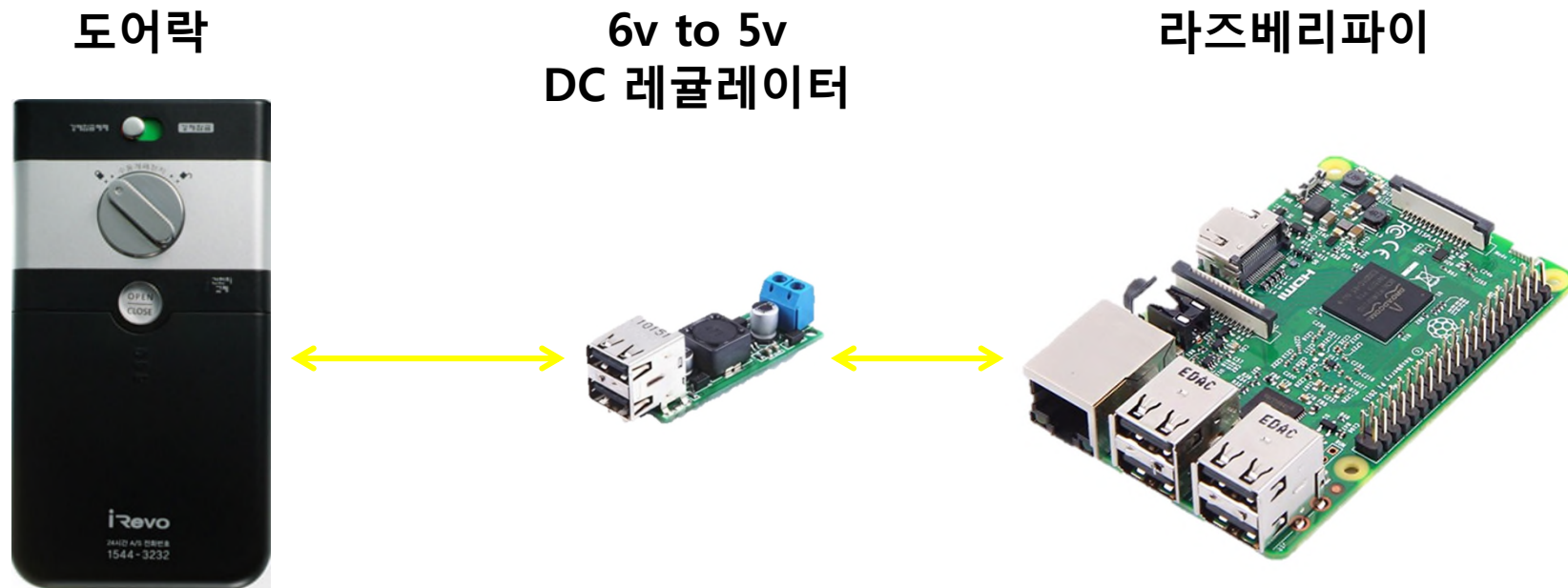
- 2.6.1 키는 예약이후까지 유효하지 않다.
- 2.6.2 어플리케이션을 해킹당하지 않는다.
- 2.6.3 서버와 도어락간의 통신을 하이제킹당하지 않는다.
- 2.6.4 잘못된 키를 통해 문을 열 수 없다.

3. 하드웨어 설계



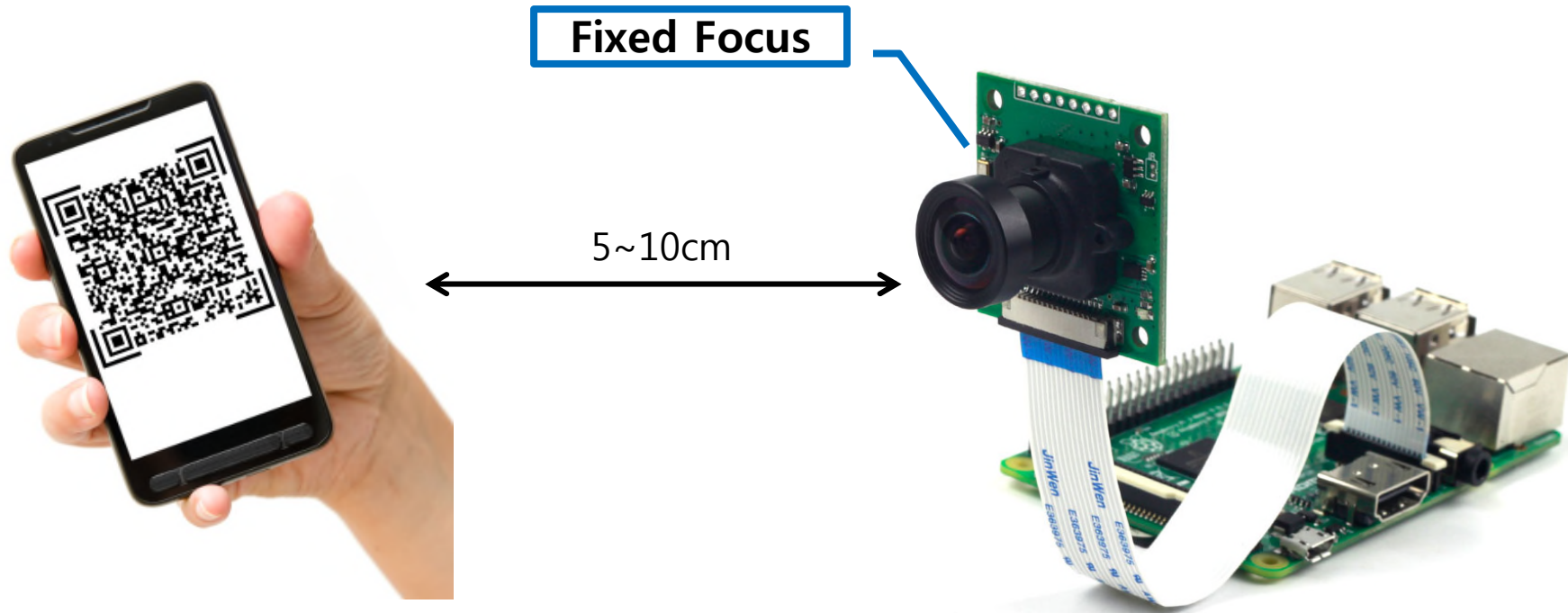
3. 하드웨어 설계 - 전원부

도어락 전원은 $1.5v * 4 = 6v$ 로 라즈베리파이의 구동전압인 5v보다 높음.
레귤레이터를 이용해 전압을 조정한 뒤 usb 케이블로 전원 공급

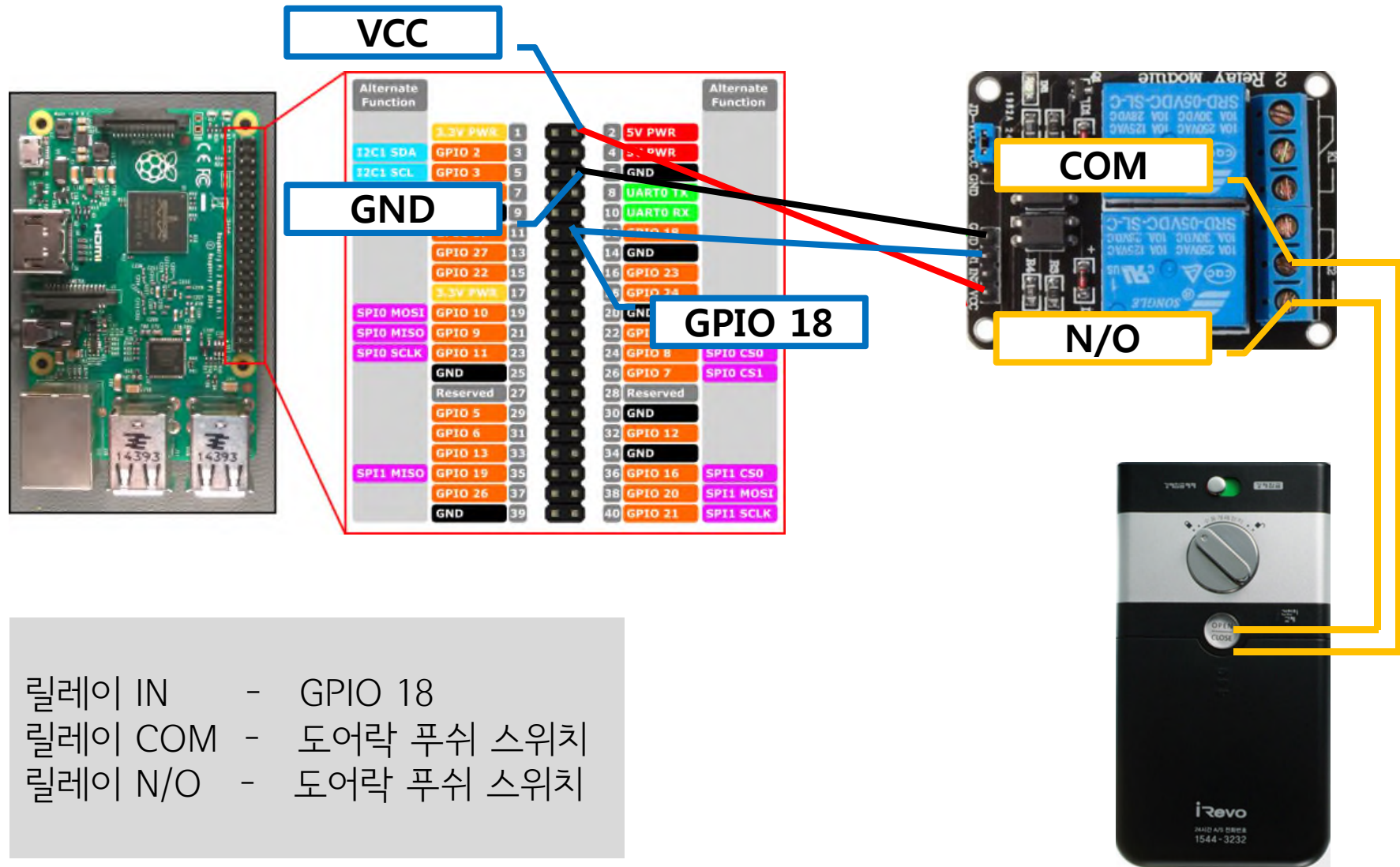


3. 하드웨어 설계 - 카메라

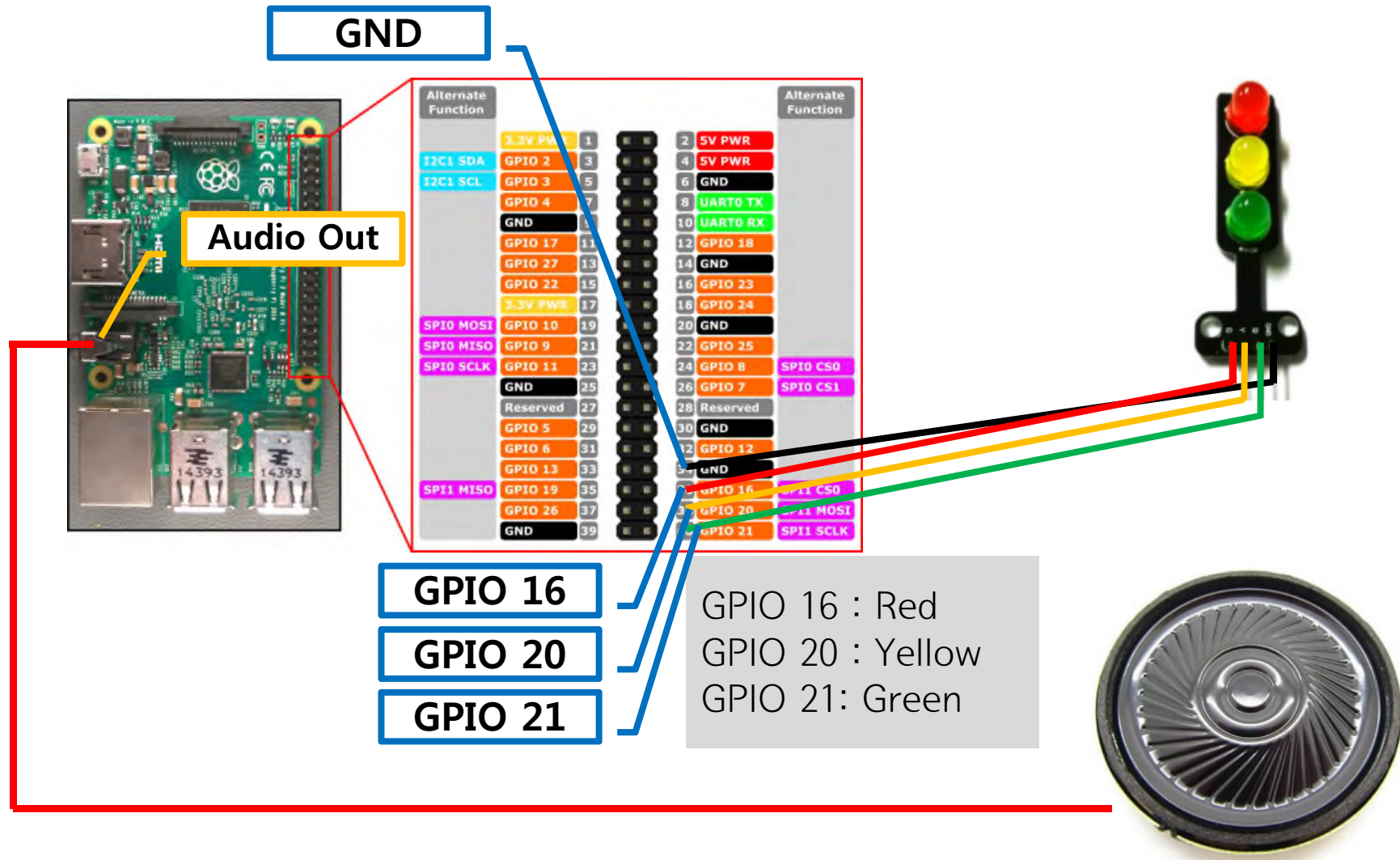
파이카메라는 초점거리가 고정이므로 5~10cm 로 초점 거리 조절 후 거치대를 이용해 거리를 유지.



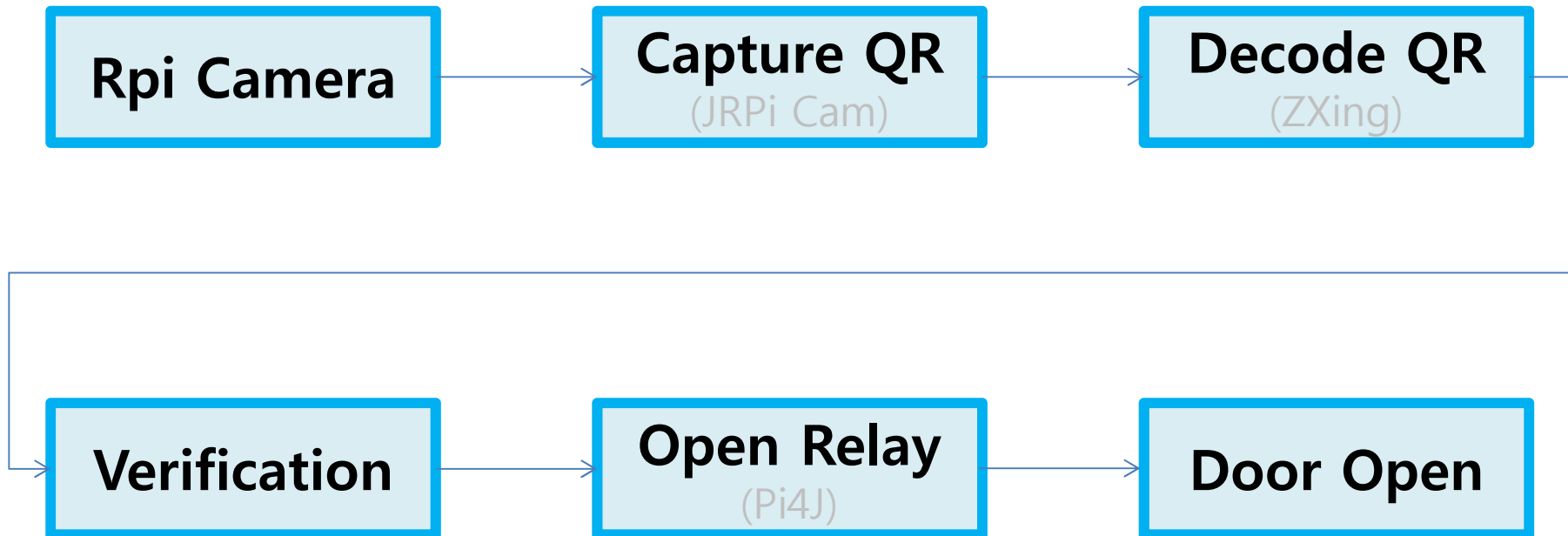
3. 하드웨어 설계 - 릴레이



3. 하드웨어 설계 - 기타



4. 하드웨어 테스트



하드웨어 동작 확인을 위해 QR 코드 인식과 릴레이 개폐 여부를 테스트 함.

4. 하드웨어 테스트



파일(F) 편집(E) 탭(T) 도움말(H)

```
Started taking the snap: 1
About to scan the snap file: snap_20171117172129.png for the QR code..
Scan Decode is successful!!! The decoded QR Code text is:
```

```
http://en.m.wikipedia.org
```

```
=====End of processing the snap 1=====
```

```
[ INFO] .....
[ INFO] BUILD SUCCESS
[ INFO] .....
[ INFO] Total time: 01:49 min
[ INFO] Finished at: 2017- 11- 17T17: 21: 36+09: 00
[ INFO] Final Memory: 9M/121M
[ INFO] .....
pi@inu_doorlock: ~/QRScantest $ mvn exec:java -Dexec.mainClass="camera.QRScanner"
-Dexec.classpathScope=runtime
[ INFO] Scanning for projects...
[ INFO] .....
[ INFO] Building iot-pi4j-raspicamera-QRScanner 0.0.1- SNAPSHOT
[ INFO] .....
Downloading from oss-snapshots-repo: https://oss.sonatype.org/content/groups/public/com/pi4j/pi4j-core/1.2- SNAPSHOT/maven-metadata.xml
```

5. 개발 환경



The Pi4J Project
Java I/O library for the Raspberry Pi



JRPiCam

Maven™



Raspberrypi 의 GPIO를 사용하기 위한 라이브러리
GPIO의 입출력을 지원한다.

Google 오픈 소스 라이브러리로 QR Code
디코딩 기능을 지원한다.

Raspicamera의 이용을 위해 python 드라이버를
JAVA API로 바인딩해주는 라이브러리.

빌드 의존성 관리를 위해 사용한다.

6. 화면 구성 – 모바일 어플리케이션

유스케이스 4 - 회원인증

휴대폰 번호를 입력하세요.

확인

기기분실 / 재인증

This is a wireframe of a mobile application screen for mobile number verification. It features a light gray background. At the top, there is a large empty rectangular area. Below this, a gray rectangular box contains the text "휴대폰 번호를 입력하세요." (Enter your mobile number.). Underneath the text box is a white rectangular input field with a blue border. Below the input field is a blue rectangular button with the white text "확인" (Confirm). At the bottom of the screen, there is another gray rectangular box containing the text "기기분실 / 재인증" (Device lost / Re-authentication).

인증번호를 입력하세요.

확인

This is a wireframe of a mobile application screen for authentication code verification. It features a light gray background. At the top, there is a large empty rectangular area. Below this, a gray rectangular box contains the text "인증번호를 입력하세요." (Enter your authentication code.). Underneath the text box is a white rectangular input field with a blue border. Below the input field is a blue rectangular button with the white text "확인" (Confirm).

6. 화면 구성 – 모바일 어플리케이션

유스케이스 5 – 초기상태

예약 내역

예약 1 1.1 ~ 1.3
...
...
...
예약 2 1.3 ~ 1.5

문 열기

비밀번호 변경

유스케이스 6 – 예약 상세내역 조회

예약 상세 내역 조회

예약 1
o 호텔 o 호실
시작 - 1.1 08:00
종료 - 1.3 20:00

상태 - 결제완료

홈

6. 화면 구성 – 모바일 어플리케이션

유스케이스 7 – 비밀번호 변경

비밀번호 변경

새 비밀번호를 입력하세요.

확인

유스케이스 8 – 문 열기

문 열기


05:00

홈

6. 화면 구성 - 웹

유스케이스 10 - 로그인

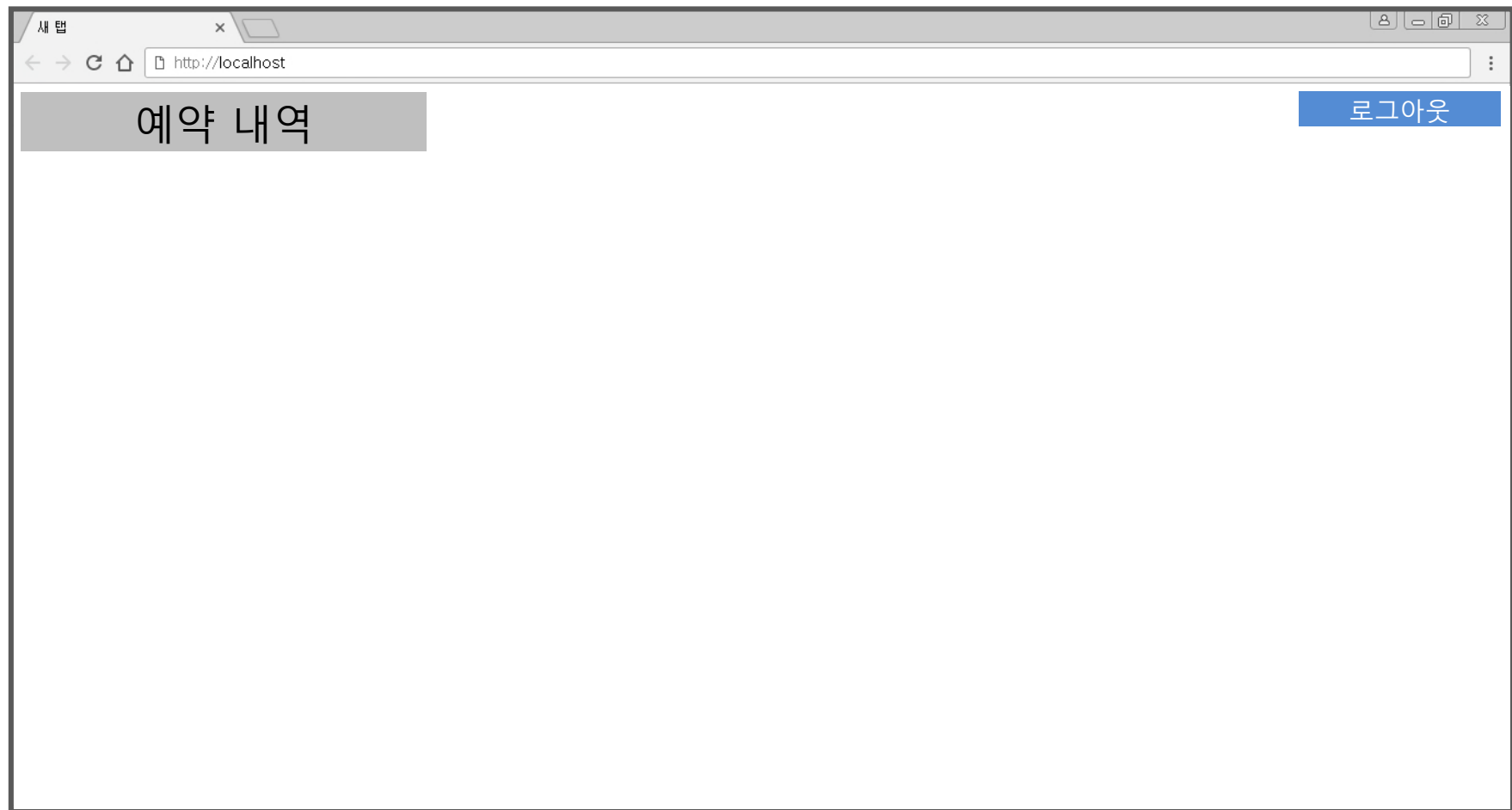


A web browser window mockup showing a login form. The browser's address bar displays 'http://localhost'. The form consists of two input fields: one for 'ID' and one for 'PW' (Password). Below these fields is a blue button labeled '로그인' (Login).

ID	<input type="text"/>
PW	<input type="password"/>
<input type="button" value="로그인"/>	

6. 화면 구성 - 웹

유스케이스 11 - 사용자 초기화면



6. 화면 구성 - 웹

유스케이스 12 - 관리자 초기화면

새 탭 x

← → ↻ ⌂ http://localhost

도어락 상태 및 제어

마스터키 발급 로그아웃

도어락 ID : 0001
도어락 위치 : 인천대학교
도어락 상태 : 정상

문열기

로그 메세지

[서버] : ~~~~
[도어락 001] : ~~~~~~
[서버] : ~~~~~~`

7. 시나리오

8. 유스케이스

9. 테스트케이스