

얼굴인식 도어락과 애플리케이션



팀명	LockLock			
담당 교수	김영웅 교수님			
팀원	이름	학번	연락처	E-mail
	박병민(팀장)	1491051	010-3088-0839	bm941201@naver.com
	임영선	1871228	010-4087-4871	dudtjs1021@hansung.ac.kr
	정은주	1871246	010-9665-0339	dmswn0210@hansung.ac.kr
	정지현	1871248	010-4512-0133	809jjh@naver.com
Project URL	https://github.com/dudtjs1021ej/faceRecognition_doorlock			

목차

1. 프로젝트 수행 목적

1.1 프로젝트 정의

1.2 프로젝트 배경

1.3 프로젝트 목표

2. 프로젝트 개요

2.1 프로젝트 설명

2.2 프로젝트 구조

2.3 시나리오

2.4 기대효과

2.5 제약조건

2.6 관련기술

2.7 개발도구

3. 프로젝트 추진 체계 및 일정

3.1 역할 분담

3.2 작업 흐름도

3.3 개발 일정

4. 참고자료

1. 프로젝트 수행 목적

1.1 프로젝트 정의

Raspberry Pi를 이용하여 얼굴인식 도어락을 제작하고 도어락과 관련된 다양한 기능을 제공하는 애플리케이션 개발

1.2 프로젝트 배경

최근 현대사회의 가장 큰 화두는 '광범위한 디지털화'입니다. 우리의 일상과 산업 현장 전반에 걸쳐 주변의 모든 것이 인터넷과 결합되고 있습니다. 자동차에 인터넷이 결합되어 자율주행 자동차가 등장하였고, 스마트 미러, 스마트 워치 등 사물인터넷(IoT) 기술을 활용한 다양한 제품들이 계속해서 출시되고 있습니다. 그 중 저희는 **스마트 도어락**에 주목하였습니다.

현재 시중에서 사용하는 스마트 도어락은 스마트폰 모바일키를 이용한 편리한 개폐시스템으로 많은 사람들이 사용하고 있습니다. 그러나 사용자들이 귀가 시 양손에 짐이 많아 모바일키 이용에 불편함을 느끼는 경우가 생길 수 있으며, 비밀번호를 입력하거나 지문인식을 해야 하는 경우에도 이러한 문제가 해결되기 어려운 시스템입니다.

또한 현재 우리나라는 고령인구가 연평균 4.4%, OECD 평균에 비해 1.7배 빠른 속도로 증가하고 있습니다. 고령화와 디지털화가 함께 가속화되는 현대사회에서 우리가 고려해야하는 사용자는 바로 고령자입니다. 기억력이 감퇴하고 스마트폰 이용이 서툰 고령자에게 시중의 스마트 도어락은 사용에 어려움이 많을 것입니다.

그리하여 우리는 위 두가지 어려움을 해결함으로써, 현대사회 요구에 맞는 '**편리함, 실용성, 보안성**'을 모두 갖춘 LockLock만의 얼굴인식 스마트도어락을 만드는 프로젝트를 진행하게 되었습니다.

1.3 프로젝트 목표

현재 시중에 얼굴인식 도어락이 존재하지만, 일반적으로 상용화되기 어려울 정도의 고가로 거래되고 있습니다. 또한 사진과 사람을 구분하지 못 해서 보안성에 있어 문제의 소지가 보입니다. 그리하여 저희는 Raspberry Pi를 이용하여 기존의 도어락을 얼굴인식 도어락으로 손쉽게 바꿀 수 있는 시스템을 만듦으로써 가격 접근성을 높이고 사람과 사진을 구분할 수 있게 함으로써 보안성을 향상시킵니다.

또한 도어락을 제어할 수 있는 애플리케이션을 제작하여 사용의 편의성을 높일 것입니다. 애플리케이션의 기능으로는 지문인식을 통한 도어락 제어, 도어락 이용 시 출입한 사람의 사진 및 시간 등 출입기록을 확인할 수 있는 기능을 갖춰 보안성과 편의성을 높입니다.

얼굴등록의 경우 휴대폰에서 사진을 촬영하거나 선택하여 firebase(클라우드)로 관리할 수 있게 함으로써 관리자가 데이터를 관리하기 용이하게 하고 사용자는 얼굴등록을 쉽고 편하게 할 수 있도록 할 것입니다.

마지막으로 어두운 환경, 사용자가 안경을 쓰고 있거나 마스크를 착용하고 있는 경우에도 얼굴인식을 하게 함으로써 사용자의 편의성과 기술의 사용성을 높일 것입니다.

2. 프로젝트 개요

2.1 프로젝트 개요

얼굴인식을 통하여 별도의 비밀번호 입력없이 얼굴 정보가 등록된 사용자라면 얼굴 인식을 통하여 출입이 가능합니다. 또한 눈의 깜박임을 통해 사진과 사람을 구분하여 사진을 통해서는 출입이 불가능하고 사용자가 안경을 쓰고 있는 상황, 사용자가 마스크를 쓰고 있는 상황, 불빛이 약해 어두운 상황에서도 얼굴 인식이 가능합니다.

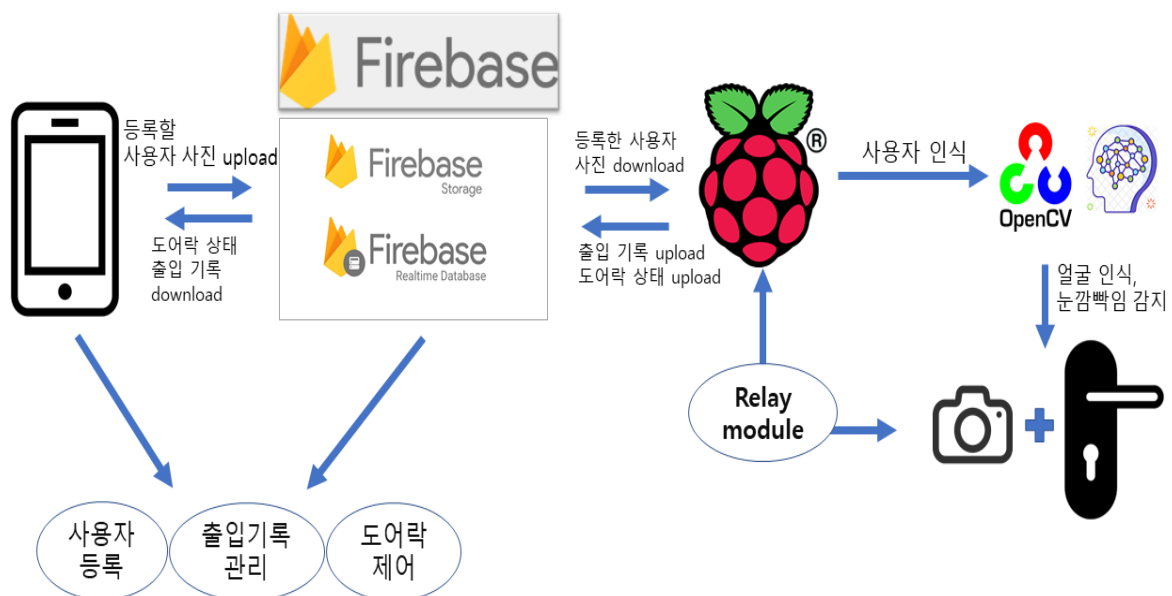
또한 도어락을 제어할 수 있는 애플리케이션이 있으며 다양한 기능을 제공합니다.

첫번째, 앱을 사용하여 인터넷을 사용할 수 환경이라면 언제 어디서든 지문인식을 통해 도어락을 제어할 수 있습니다.

두번째, 앱에서 사진을 선택하거나 촬영하여 자신의 얼굴 데이터를 저장할 수 있고 등록된 사용자가 누구인지 확인할 수 있습니다.

세번째, 도어락을 이용 시 출입한 사람의 사진, 시간 등 출입기록을 확인 가능합니다.

2.2 프로젝트 구조



2.3 시나리오

가. 사용자 정보 관리

회원가입

Username

Email

Password

가입하기

LockLock

jh@gmail.com

.....


로그인

회원가입



1. Firebase Authentication, 어플리케이션을 통해 회원가입을 한 뒤 로그인한다.
2. Firebase Realtime Database에서 가입한 사용자의 정보를 확인할 수 있다.

나. 얼굴 등록



CHOOSE

촬영

이름을 입력하세요

UPLOAD

1. 라즈베리파이의 카메라와 어플리케이션의 카메라를 사용하여 사진을 찍는다.


Storage

Files Rules Usage

gs://fir-connjva.appspot.com > images

<input type="checkbox"/>	이름	크기	유형	최종 수정 날짜
<input type="checkbox"/>	sys.jpg	565.72 KB	image/jpeg	2021. 3. 7.
<input type="checkbox"/>	OIP.jpeg	13.39 KB	image/jpeg	2021. 2. 21.
<input type="checkbox"/>	bm1.jpg	1.1 MB	image/jpeg	2021. 3. 8.
<input type="checkbox"/>	ej1.jpg	421.78 KB	image/jpeg	2021. 3. 7.
<input type="checkbox"/>	fdyys.jpg	517.36 KB	image/jpeg	2021. 3. 7.
<input type="checkbox"/>	jh.jpg	72.86 KB	image/jpeg	2021. 3. 9.
<input type="checkbox"/>	jh1.jpg	956.3 KB	image/jpeg	2021. 3. 3.
<input type="checkbox"/>	jh2.jpg	1.14 MB	image/jpeg	2021. 3. 3.
<input type="checkbox"/>	jh3.jpg	1.28 MB	image/jpeg	2021. 3. 3.

파일 업로드



이름
[jh.jpg](#)

크기
74,608바이트

유형
image/jpeg

생성 날짜
2021. 3. 9. 오후 3:33:34

업데이트 날짜
2021. 3. 9. 오후 3:33:34

파일 위치

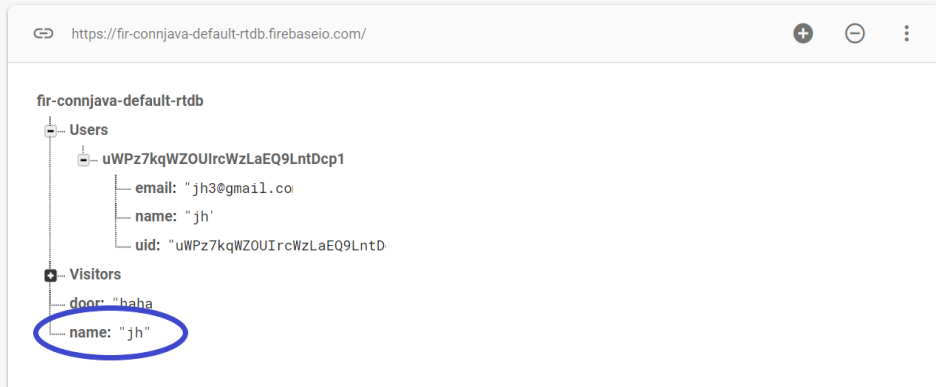
기타 메타데이터

2. 찍힌 사진을 Firebase Storage를 이용하여 사용자의 얼굴을 등록한다.

Realtime Database

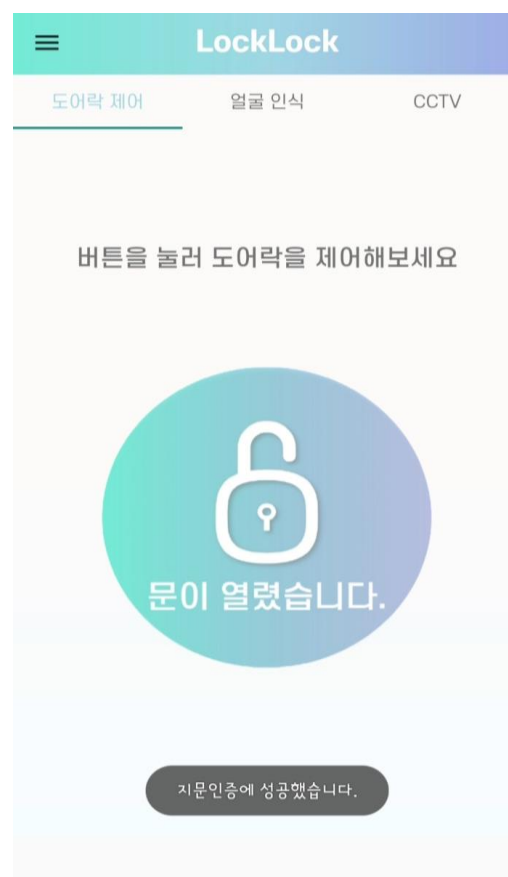
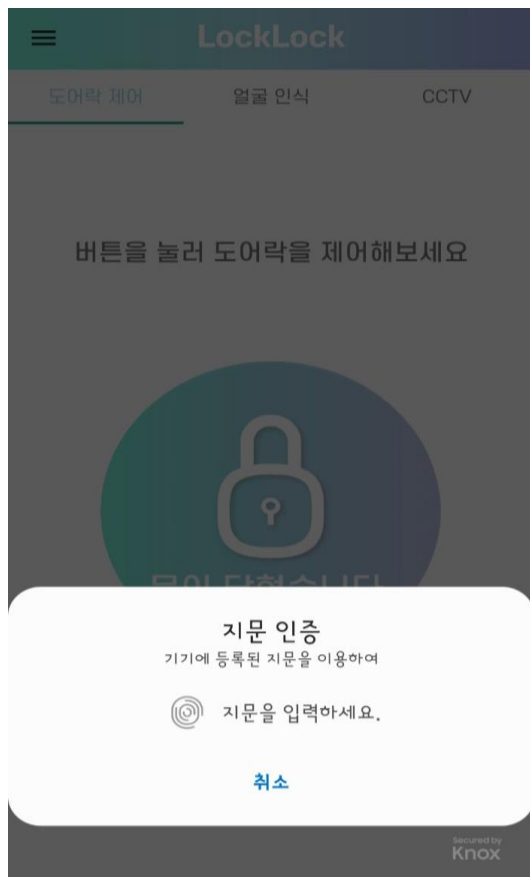
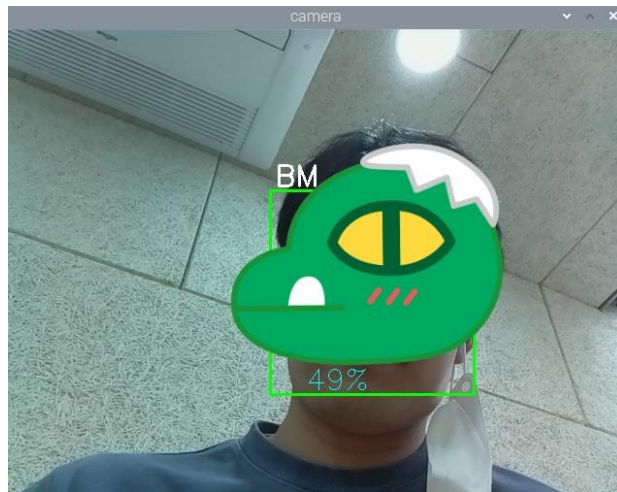
[데이터](#) [규칙](#) [백업](#) [사용량](#)

 로컬 에뮬레이터 도구 모음을 사용한 프로토타입 제작 및 엔드 투 엔드 테스트에 **Firebase** 인증 지원 도입 [시작하기](#)



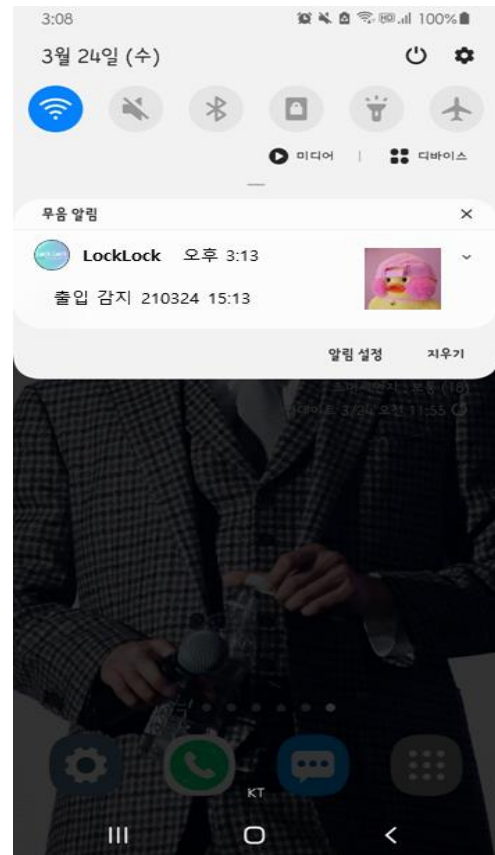
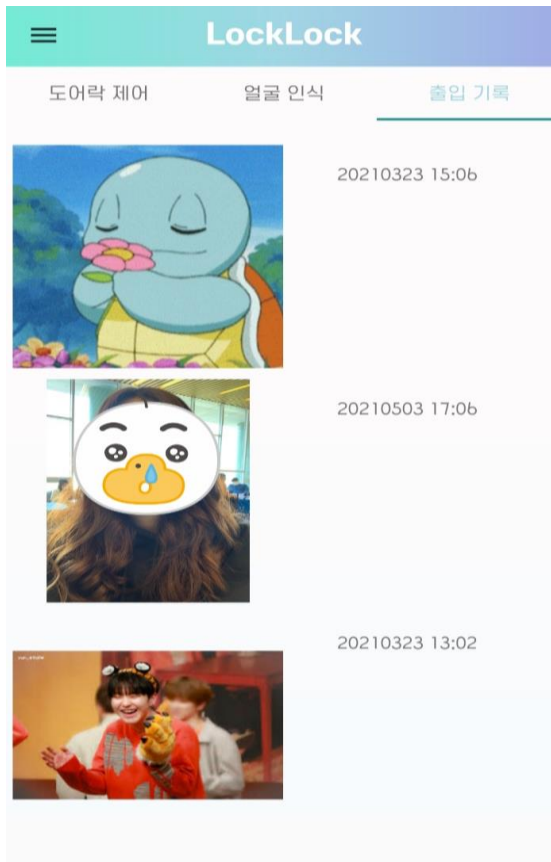
3. 입력한 이름을 Firebase Realtime Database를 이용하여 저장한다.

다. 도어락 제어



1. 사용자의 얼굴을 라즈베리파이 카메라로 실시간 비교하여 등록된 사용자의 경우 Firebase Realtime database를 이용하여 도어락이 해제된다.
2. 어플리케이션에서 로그인한 뒤 도어락 제어 부문에서 지문인식을 한 뒤에 Firebase Realtime database를 이용하여 도어락을 열거나 닫을 수 있다.

라. 출입기록 관리



1. 얼굴인식을 통해 도어락 해제를 시도한 경우, 인식 유무에 관계없이 출입을 시도한 시간은 firebase realtime database의 Visitors에 저장하고 출입을 시도한 사람의 사진은 firebase storage의 Vistors폴더에 저장하여 관리한다.
2. 도어락 해제를 시도할 때마다 앱을 통해 누군가 출입을 시도한다는 사실을 그 사람의 사진, 시간을 Notification을 통해 사용자에게 알린다.
3. 출입을 시도한 사람의 사진과 시간을 앱을 통해 확인할 수 있다.

2.4 기대 효과

- 현재 시중에서 판매되고 있는 40~60만원 가량의 고가의 얼굴인식 도어락을 구매하지 않고도 현재 사용하는 도어락을 13만원 가량의 금액만 지불하고 얼굴인식 기능이 탑재된 도어락으로 업그레이드하여 사용할 수 있기 때문에 가격 접근성을 높일 수 있습니다.
- 애플리케이션을 통해 언제 어디서든 필요에 의해 도어락을 제어할 수 있고 손이 자유롭지 못한 상황에서 비밀번호를 입력해야 하는 번거로움없이 도어락을 열 수 있으므로 사용자의 편리성이 높아질 수 있습니다.
- 비밀번호를 직접 입력하지 않고 생체인증을 통해 도어락을 제어함으로써 비밀번호 유출의 가능성을 줄이고 사진과 사람을 구별함으로써 보안성을 높일 수 있습니다.
- 사용자가 안경이나 마스크를 쓴 상태나 불빛이 약해 어두운 환경에서도 얼굴을 인식하여 구별함으로 사용에 있어 편리성을 높일 수 있습니다.
- 출입을 시도한 사람의 사진과 시간을 앱을 통해 확인할 수 있고 출입기록을 데이터로 저장함으로써 사용자의 편리성과 보안성을 높일 수 있습니다.

2.5 제약 조건

- 라즈베리파이의 성능이 아직까지 기존의 컴퓨터보다는 떨어지기 때문에 저희가 목표한 기능을 전부 넣어서 얼굴인식 프로그램을 만들어 실행시켰을 경우 시간이 오래 걸릴 수 있습니다.
- 현재 코로나19로 인하여 전세계적으로 마스크 착용이 일상화되고 있습니다. 그로 인해 사용자의 편의성을 위해서 마스크를 착용한 상태의 얼굴을 인식하게 한다면 마스크가 없는 얼굴을 인식하는 것에 비하여 인식률과 정확도가 떨어질 수 있습니다.
- 이번 프로젝트를 진행하면서 테스트를 할 수 있는 사람에 한계가 있기 때문에 정확도에 있어서 신뢰성이 떨어질 수 있습니다.

2.6 관련기술

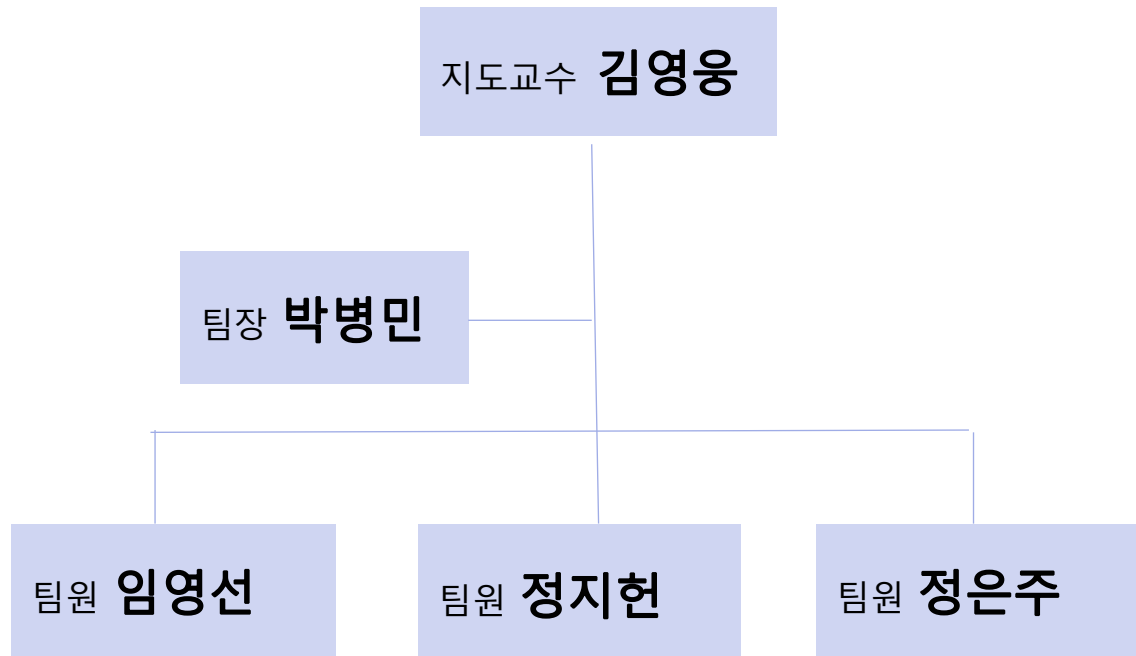
구분	설명
OpenCV	OpenCv는 실시간 컴퓨터 비전을 목적으로 한 프로그래밍 라이브러리이다. 실시간 이미지 프로세싱에 중점을 둔 라이브러리이다.
Deep Learning	머신 러닝의 한 분야로써 여러 비선형 변환기법의 조합을 통해 높은 수준의 추상화(abstractions, 다량의 데이터나 복잡한 자료들 속에서 핵심적인 내용 또는 기능을 요약하는 작업)를 시도하는 기계학습(machine learning) 알고리즘의 집합으로 정의되며, 큰 틀에서 사람의 사고방식을 컴퓨터에게 가르치는 기계학습의 한 분야
Database	여러 사람이 공유하고 사용할 목적으로 통합 관리되는 정보의 집합이다. 논리적으로 연관된 하나 이상의 자료의 모음으로 그 내용을 고도로 구조화함으로써 검색과 갱신의 효율화를 꾀한 것이다. 즉, 몇 개의 자료 파일을 조직적으로 통합하여 자료 항목의 중복을 없애고 자료를 구조화하여 기억시켜 놓은 자료의 집합체라고 할 수 있다.

2.7 개발도구

구분	설명
Android Studio	안드로이드 스튜디오는 안드로이드 및 안드로이드 전용 어플 제작을 위한 공식 통합 개발 환경이다. 프로그래밍 언어로는 자바랑 코틀린을 지원한다. 유연성이 좋은 Gradle 기반의 프로젝트 빌드 시스템을 사용한다
Firebase	파이어베이스는 2011년 파이어베이스사가 개발하고 2014년 구글에 인수된 모바일 및 웹 애플리케이션 개발 플랫폼이다. 파이어베이스는 실시간 데이터베이스, Cloud Firebase, Cloud Storage 기능들을 제공하고 있다.
Raspberry pi	영국의 라즈베리 파이 재단이 학교에서 기초 컴퓨터 과학 교육을 증진시키기 위해 만든 싱글 보드 컴퓨터이다. 이번에 새로 출시한 모델은 1.2GHz Quad core, Cortex A53, 64 bit CPU와 자체적으로 802.11n WIFI, Bluetooth 제공한다.
Python	파이썬은 고급 프로그래밍 언어로, 플랫폼에 독립적이며 인터프리터식, 객체지향적, 동적 타이핑(dynamically typed) 대화형 언어이다. 다양한 플랫폼에서 쓸 수 있고, 라이브러리(모듈)가 풍부하여, 대학을 비롯한 여러 교육 기관, 연구 기관 및 산업계에서 이용이 증가하고 있다.

3. 프로젝트 추진 체계 및 일정

3.1 역할 분담

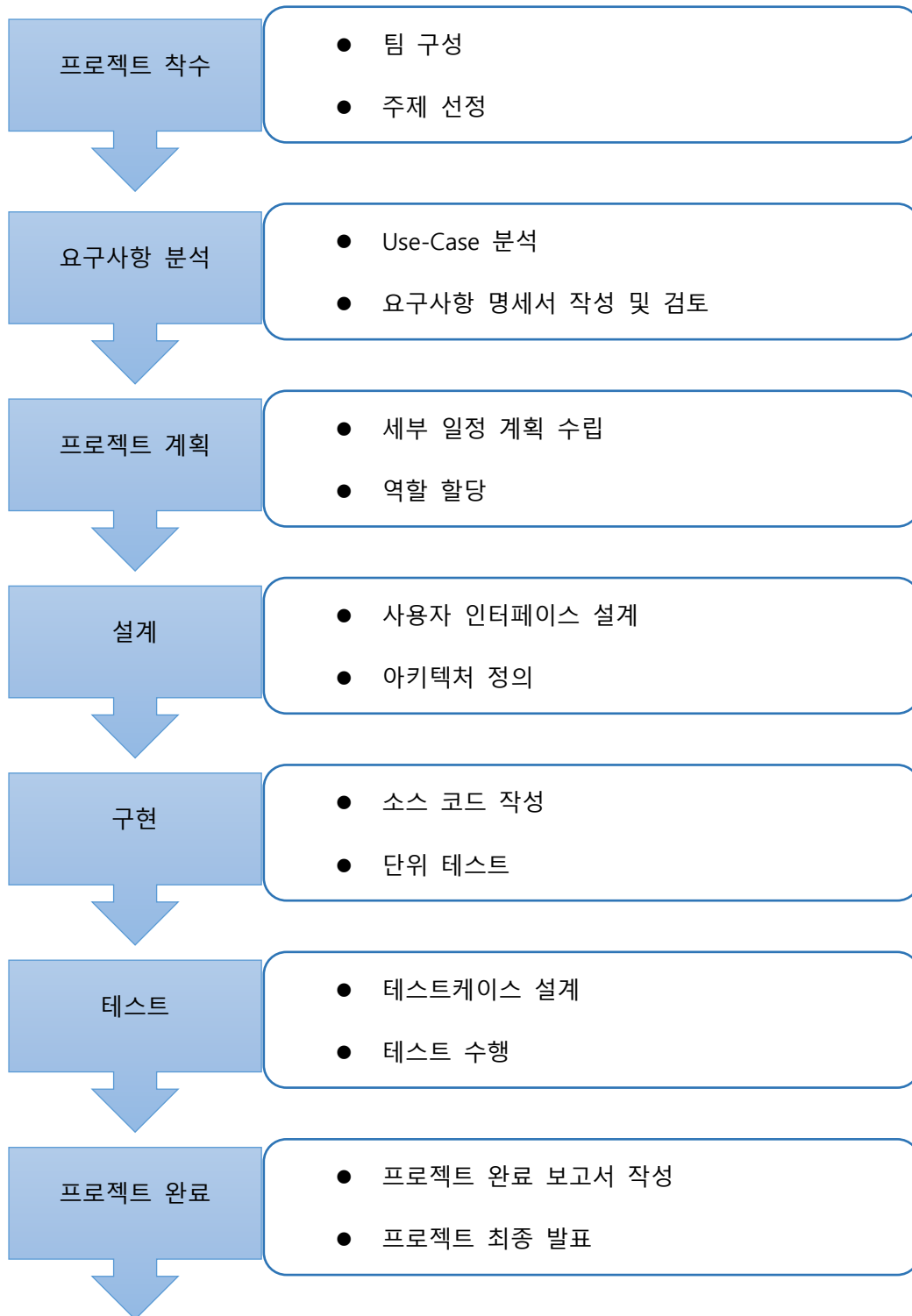


박병민	얼굴인식코드 코딩, 사진과 사람 구분하는 코드코딩, 각종 발표준비 및 발표
임영선	앱 개발, 도어락 제어코드 코딩, 파이어베이스 관련 기능코딩 주간보고서 작성
정지현	앱 개발, 마스크인식코드 코딩, 파이어베이스 관련 기능코딩 주간보고서 작성
정은주	앱개발 , 하드웨어 환경 구축, 파이어베이스 서버구축 및 연결 주간보고서 작성

3.2 개발 일정

구분	3월				4월				5월				6월			
주제선정	■	■	□	□	□	□	□	□	□	□	□	□	□	□	□	
자료수집	■	■	■	■	□	□	□	□	□	□	□	□	□	□	□	
장비 구매	□	■	■	■	□	□	□	□	□	□	□	□	□	□	□	
관련 기술 학습	□	□	■	■	■	□	□	□	□	□	□	□	□	□	□	
환경 구축	□	□	■	■	■	■	□	□	□	□	□	□	□	□	□	
얼굴인식코드	□	□	■	■	■	■	■	■	■	■	■	■	■	□	□	
앱 개발	□	□	■	■	■	■	■	■	■	■	■	■	■	□	□	
추가기능 구현	□	□	□	□	□	□	□	■	■	■	■	■	■	□	□	
통합및테스트	□	□	□	□	□	□	□	□	□	□	■	■	■	■	■	

3.3 작업 흐름도



4. 참고자료(추후 추가)

firebase 참고자료

<https://firebase.google.com/docs/storage>

<https://firebase.google.com/docs/database>

<https://firebase.google.com/docs/auth>

얼굴인식코드 참고자료

<https://www.pyimagesearch.com/2018/06/25/raspberry-pi-face-recognition/>

<https://www.pyimagesearch.com/2018/06/25/raspberry-pi-face-recognition/>

<https://ukayzm.github.io/python-face-recognition/>

<https://rethinkdb.com/>

<https://blog.naver.com/ljy9378/221438192568>

<http://blog.naver.com/PostView.nhn?blogId=chandong83&logNo=221436424539&>

<https://blog.naver.com/chandong83/221695462391>

<https://ukayzm.github.io/python-face-recognition/>

<https://webnautes.tistory.com/1278>

<https://studiou.tistory.com/3>

<http://www.gisdeveloper.co.kr/?p=7208>

https://github.com/EulerCauchyEquation/2018.11.28_FaceRecognition-Doorlock

<https://m.blog.naver.com/PostView.nhn?blogId=icbanq&logNo=221739906347&proxyReferer=https:%2F%2Fwww.google.com%2F>

<https://github.com/younggh4842/Controller-with-Face-Recognition>

<https://github.com/jaehee22/faceLock>

<https://happy-sunggeun.tistory.com/6>