



국방 모바일 반납

폰 좀 내이만 (Pon Johm Naimann)

병장 박찬정

병장 강동민

병장 이진휘

- 개발 동기
- 개발 목표설정
- 개발 과정
- 시행착오 및 개선
- 개발 결과
- 기대효과 및 발전방향



국방모바일보안

국방부 비즈니스

③

기기와의 호환되는 앱입니다.

★★★★★ 6,148

설치됨

이 “앱”을 아시나요?

● 개발동기

“해제했는데도 카메라가 켜지지 않아요.”

“앱을 설치하니 핸드폰이 느려졌어요.”

“유*브는 왜 강제종료시키죠?”



많은 불편과 불안

모든 데이터 삭제

초기화를 수행하여 경고 없이 휴대전화 데이터를 지웁니다.

“이 권한은 왜 요구하죠?
제 핸드폰을 어떻게 하려는 건가요?”

● 지정과제

기술분야	제안내용	제안처
모바일	생물관 병사 스마트폰 비대면 반납 앱(스마트폰 등)	사이버안전사령부
웹 서비스	비대면 상담 서비스(고민상담, 피드백 제공 등)	사이버안전사령부
	Help Desk 지원을 위한 24시간 AI 챗봇 서비스	군수 전산소

제 2의 '국방모바일보안'이 탄생하는게 아닌지
정말 앱만으로 반납을 통제해야 하는지

IoT로도... 될 것 같은데?!

● 개발 목표 설정

비대면 반납 '시스템'을 만들자!

- 💡 **비대면**으로 이루어져야 함
- 💡 앱에서 **많은 권한**을 요구하면 안됨
- 💡 비정상 반납 행위에 **충분한 방어력**을 갖춰야 함
- 💡 **기종에 무관**하게 적용할 수 있어야 함

● 개발 과정

이를 만족시킬 수 있는 시스템..?

시스템 구성

전체 시나리오

1. App에서 반납 버튼을 누르면, App에서 서버에 반납절차 시작을 알린다.
2. 서버는 App에게 반납할 RaspPi에 대한 정보(칸 번호, 위치 등)를 Response로 준다.
3. 서버는 RaspPi에게 반납 절차를 시작됨을 알린다.
4. App에는 반납 대기 화면이 뜬다.
5. RaspPi에서는 무게 센서의 값을 주시한다.
6. 병사가 스마트폰을 무게 센서에 올려두고 문을 닫으면, RaspPi는 이를 감지하고 서버에 알린다.
7. RaspPi는 QR코드 인식 상태로 대기한다.
8. 서버에서 App에게 난수를 전송한다.
9. App에서는 난수(와 추가적인 정보)를 QR코드로 변
10. RaspPi는 기기의 화면에서 송출되는 QR코드를 읽으
11. 이 때 무게 센서에서 읽은 무게 정보도 같이 전송한
12. 서버는 RaspPi에게서 받은 정보와 처음에 App에 보
13. 서버에 저장된 무게 정보와 RaspPi에게서 받은 무게
14. 문제가 없다면 서버 DB에서 사용자/핸드폰의 반납
15. 특이사항이 생긴다면, 일단 RaspPi에서 카메라로 병

기기

③ 서버에서 화면인식을 난수 생성



④ 인식되면 서버에 알림

Ras

IoT 서버 (소켓)

- 서버 → 라즈 : App에서 반납을 누른 후, 반납 과정이 시작됨을 라즈베리파이에게 알린다.
- 라즈 : 무게 센서를 주목하기 시작한다. 무게 센서에 반응이 있을 경우(핸드폰을 올려놓음) 문을 닫고 잠근다. 그리고 아래 통신!
- 라즈 → 서버 : QR코드 인식이 준비되었음을 알린다.
- 서버 : 난수를 생성하여 App에 전송한다. App은 QR코드를 띄운다.
- 서버 → 라즈 : 난수, 무게 정보를 라즈베리파이에게 전달한다.
- 라즈 : 난수에서 QR코드 암호화 Key값을 추출한다(App과 동일한 방식 사용). 카메라를 활성화하여 QR코드를 인식한다.
- 라즈 → 서버 : 인식한 난수, 무게 정보, 핸드폰의 사진을 전송한다.
- 서버 : 반납 정보를 저장한다. 라즈베리파이에서 받은 정보가 기존과 다르다 싶으면 통제간부에게 알린다. App에 알러서 App이 보관중 절전 상태로 있게 한다.
- 라즈 : 성공시 보관&도난방지 모드로 변환. 실패시 잠겨있던 문을 열어준다.

통신별 시나리오

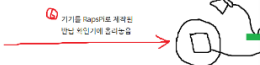
어플

- 어플 → 서버 : App에서 반납 버튼을 누르면, 서버에 해당 : 서버 : 반납키 번호, 위치 등을 지정한다. 반납 과정이 시작
- 서버 → 어플 : 반납키 번호, 위치 등의 정보를 App에게 알린다.
- 사용자가 반납기에 폰을 넣을때까지 모두 대기한다.
- 서버 : 사용자가 폰을 반납기에 넣고 라즈베리파이가 이를 감지하여 서버에 알리면, 아래 통신!
- 서버 → 어플 : 난수를 전송한다.
- 어플 : QR코드를 생성하여 화면에 띄운다.
- 라즈베리파이가 인식하고 성공 여부를 판단할때까지 모두 대기한다.
- 서버 : 라즈베리파이가 성공 여부를 알리면, 아래 통신!
- 서버 → App : 성공시 App에게 절전모드에 들어가도록 지시한다. 실패시 App에게 실패 페이지를 띄우도록 지시한다.

Server

⑤ 서버는 지정된 RaspPi만

RaspPi는 화면인식을 준비함



⑥ 기기를 RaspPi로 제재한

반납 부팅기에 들어놓음

⑦ 등록 완료

Server

⑧ 등록 완료

⑨ 등록 완료

⑩ 등록 완료

⑪ 등록 완료

⑫ 등록 완료

⑬ 등록 완료

⑭ 등록 완료

⑮ 등록 완료

⑯ 등록 완료

⑰ 인식 결과를 서버와 비교하여 반납해야 하는

핸드폰이 있는지 확인

⑱ 인식 결과

⑲ 인식 결과

⑳ 인식 결과

㉑ 인식 결과

㉒ 인식 결과

㉓ 인식 결과

① 초기 등록

App을 처음 설치하면 지정 정보를 입력하는 페이지가 나온다.

정보를 입력하고 후에 서버를 누르면 그 정보를 서버에 전송하고,

서버는 받은 정보를 저장하면서 고유 시리얼을 부여하여 지정한다. 그리고 해당되는 그 시리얼을 App에 전송한다.

App은 이 시리얼을 저장해둔다. 이 시리얼이 이후 APP 설정에 사용된다.

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

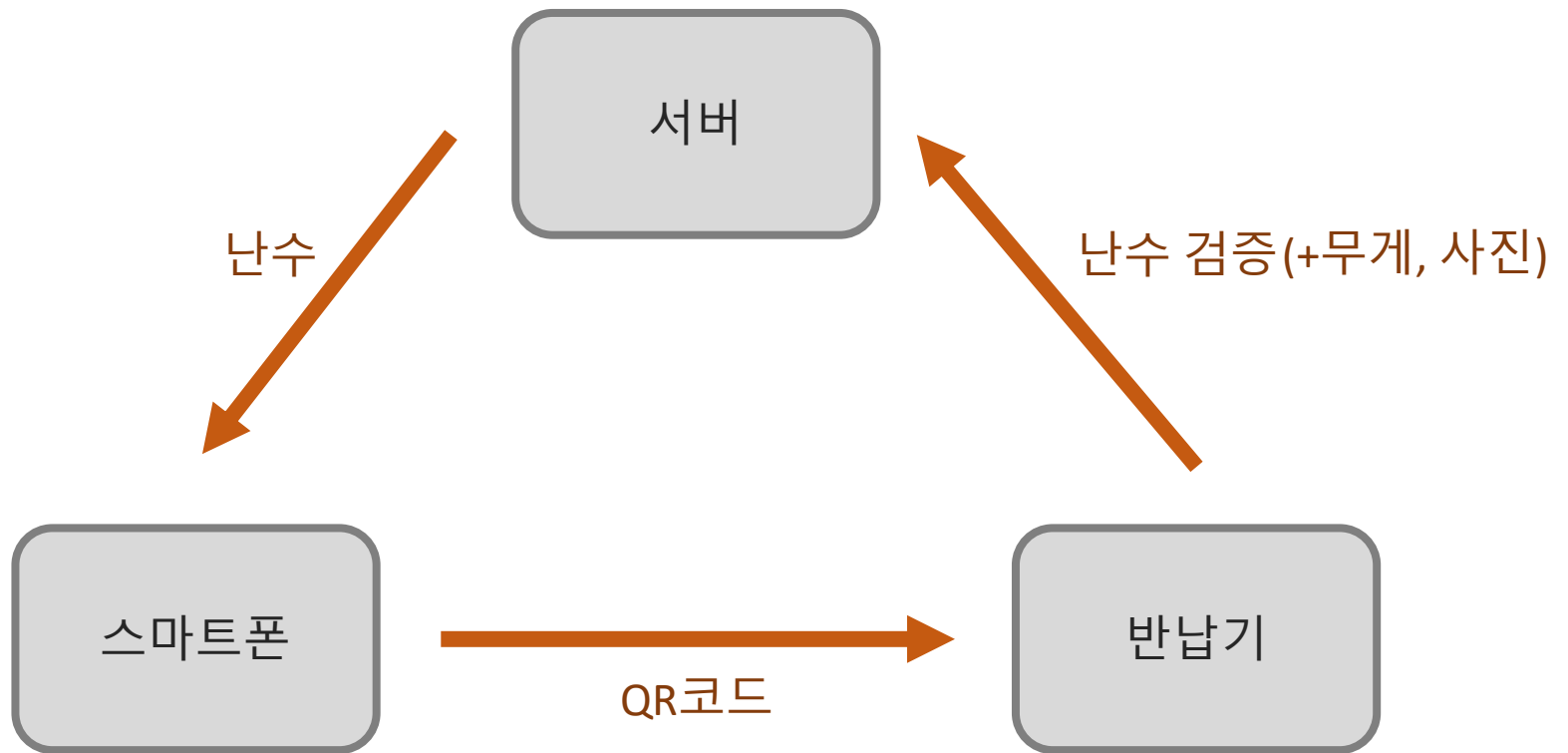
이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"

이제, 준비, "그로 시리얼"



Q

QR코드를
사용하자!

A

QR코드를 베끼면?
(스크린샷, 모눈종이)

Solution

기기가 반납기에 들어간 후에
앱에 난수를 전송해
QR코드를 생성하자!

Q

그런데 네트워크
연결이 제한될 때는?
(파견지 등)

Solution

난수 대신 **TOTP**를 사용하자!
처음 등록시에 서버와 **UUID**를
공유하고 SEED값으로 사용하자.

모든 현황 관리 가능

비
대
면

기기에 고유함

(초기등록시)
난수

서버

난수

TOTP 검증 후
기기 사진, 무게 전송

스마트폰

난수

+

현재
시간

TOTP



QR코드

반납기

요구하는 권한
일절 없음

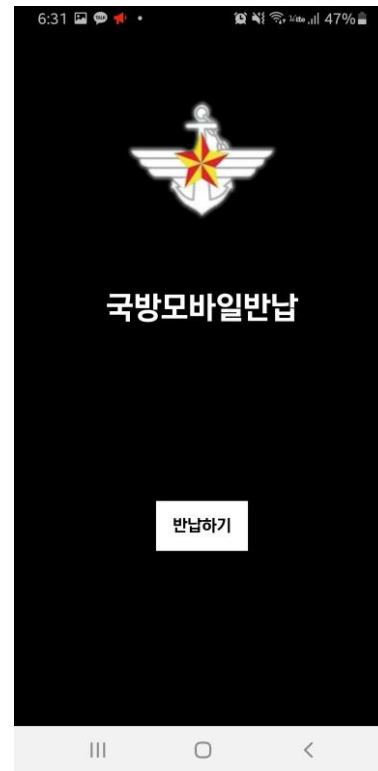
기종과 전혀
관계 없음

비대면 반납 '시스템' 의 조건

- 💡 **비대면**으로 이루어져야 함
- 💡 앱에서 **많은 권한**을 요구하면 안됨
- 💡 비정상 반납 행위에 **충분한 방어력**을 갖춰야 함
- 💡 **기종에 무관**하게 적용할 수 있어야 함

비대면 반납 '시스템' 의 조건

- ✓ 비대면으로 이루어져야 함
- ✓ 앱에서 **많은 권한**을 요구하면 안됨
- ✓ 비정상 반납 행위에 **충분한 방어력**을 갖춰야 함
- ✓ **기종에 무관**하게 적용할 수 있어야 함



Dashboard

반납 현황

전체 반납 기록

비정상 반납 기록

관리 현황

관리자 정보

병사 기기 정보

DOCUMENTATION

Dashboard

반납 현황

전체 반납 기록

#
8
7
6
5
4
3
2
1

반납 기록 정보

이름

박찬정

군번

33-333333

기기 사진



제조사

Apple

종류

PHONE

기기 식별자

S/N33333

반납 시간

2020-11-02 23:11:38

기기 무게

150

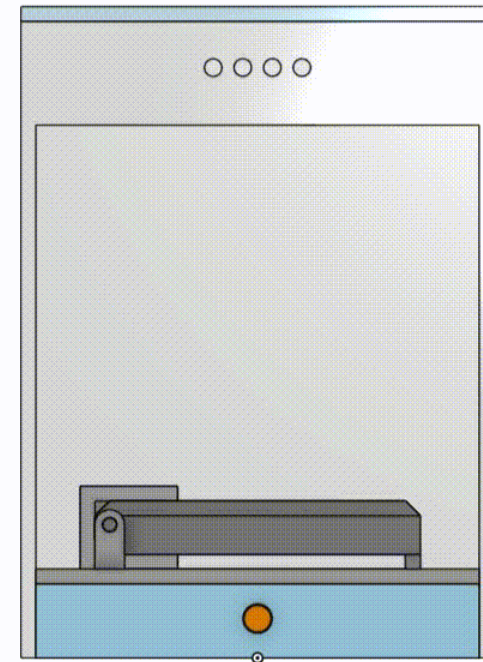
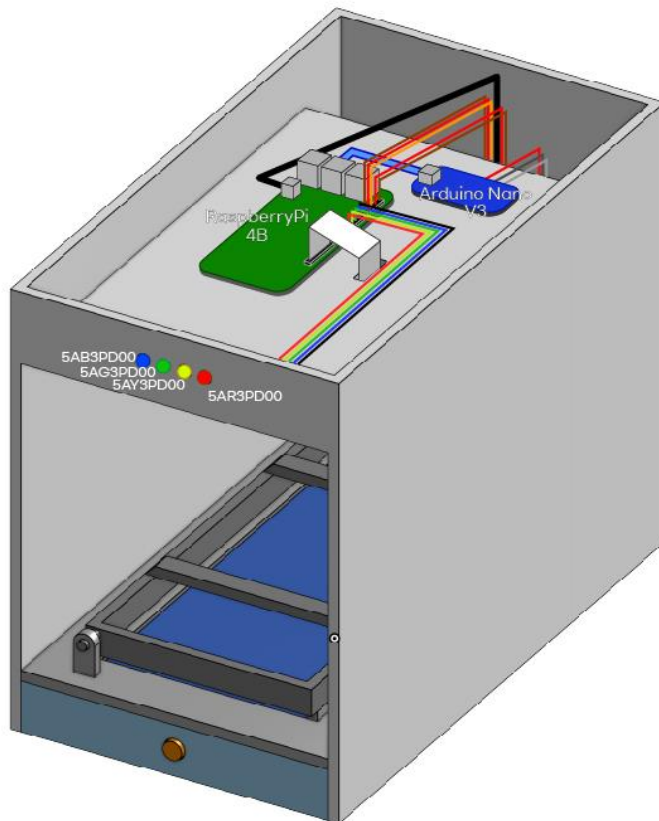
반납 상태

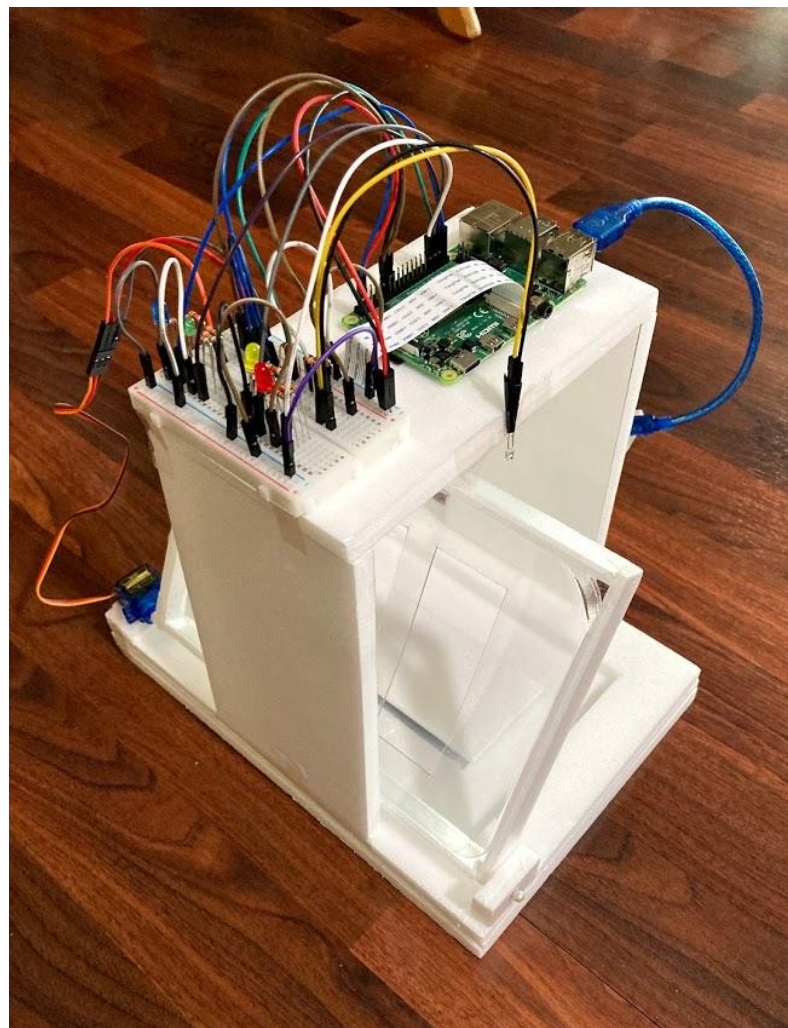
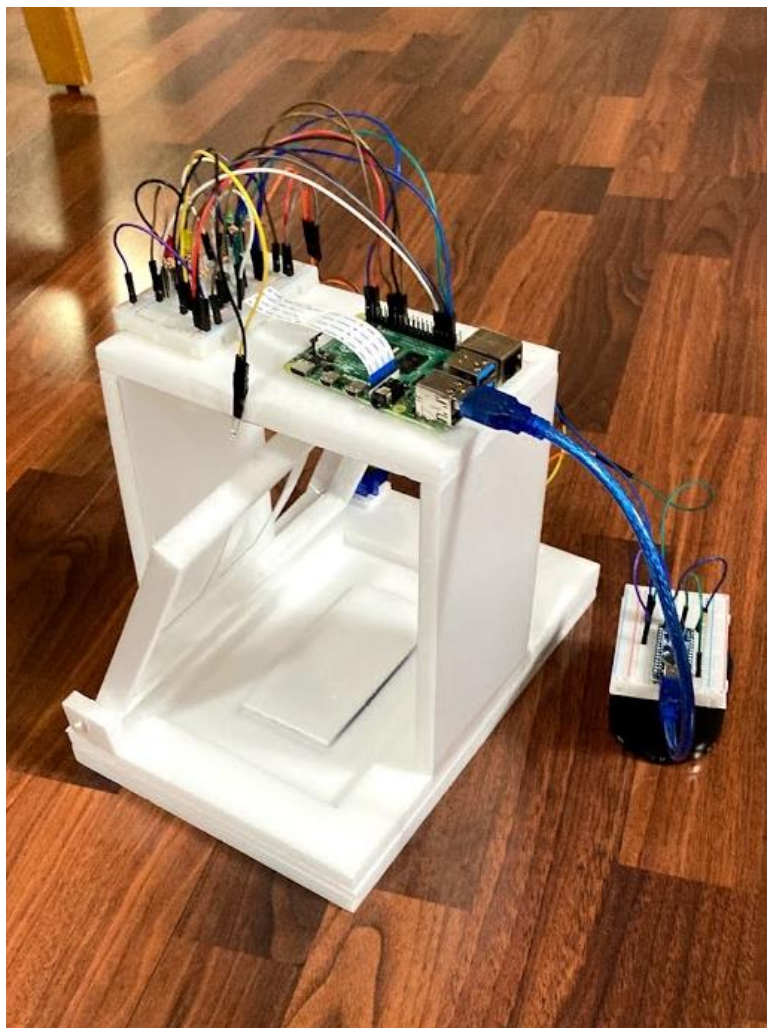
TIME_DELAY

Close



open source
hardware





시연 영상



● 기대효과

 스마트폰 사용 시간 준수

 도난 방지

 비정상 반납 방지, 미등록 기기 탐지

 격리자 운용에 활용 가능

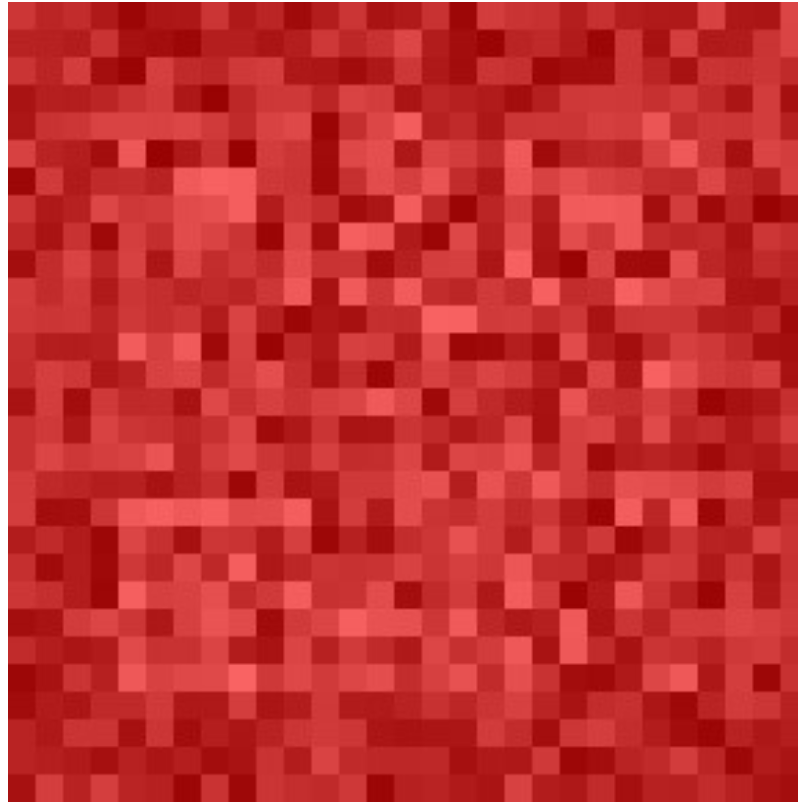
이 모든 효과를 실제 **사용자의 부담 없이!**

고급 기능

QR코드 색상암호화

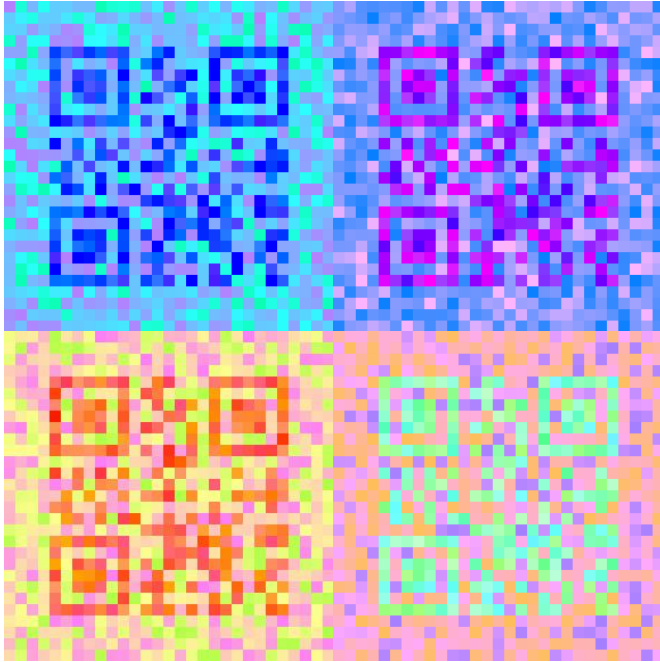


아무나 해독 가능

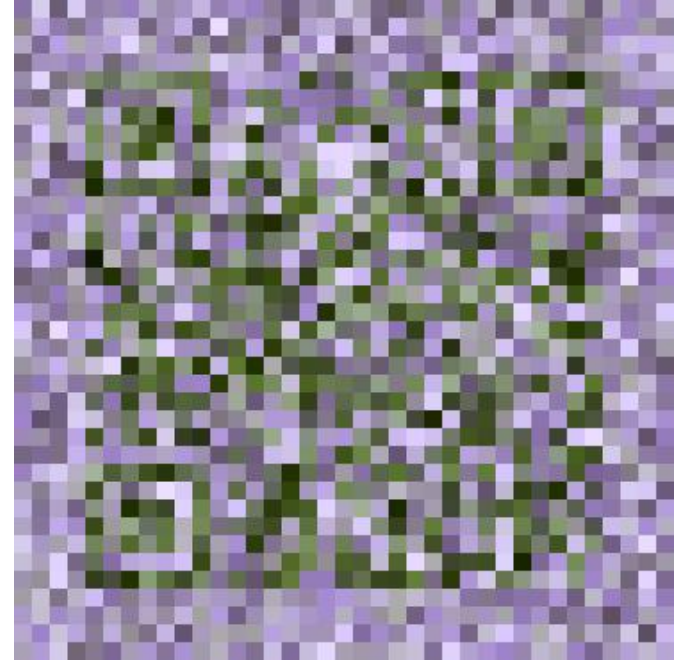


붉은색 선글라스를 끼면 QR코드의 형태가 **명확**해진다!





모든 색으로
암호화 가능



'색'이 아닌
임의의 세 벡터로도 가능

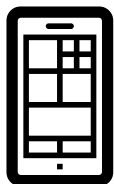
관리자용 앱



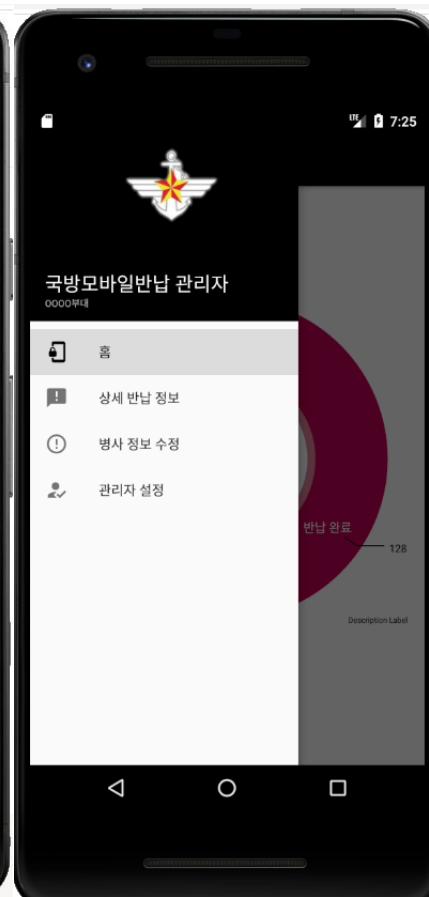
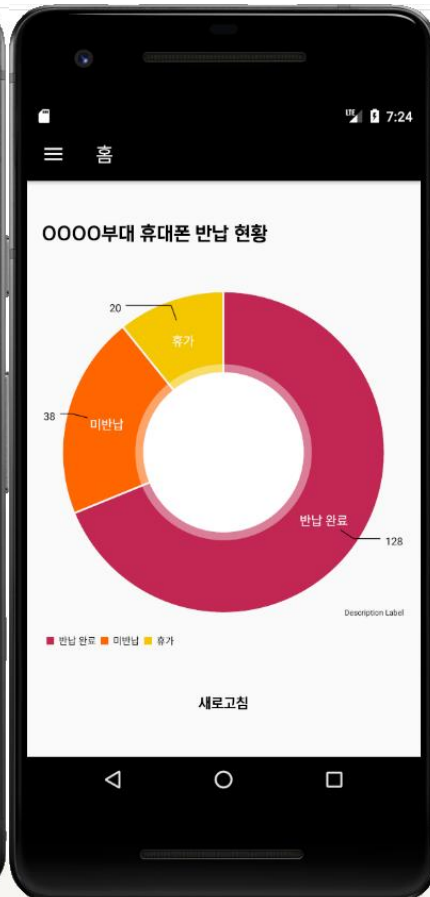
서버는 반드시 국방망 외부에 있어야 함.
(병사 기기와 통신 불가)



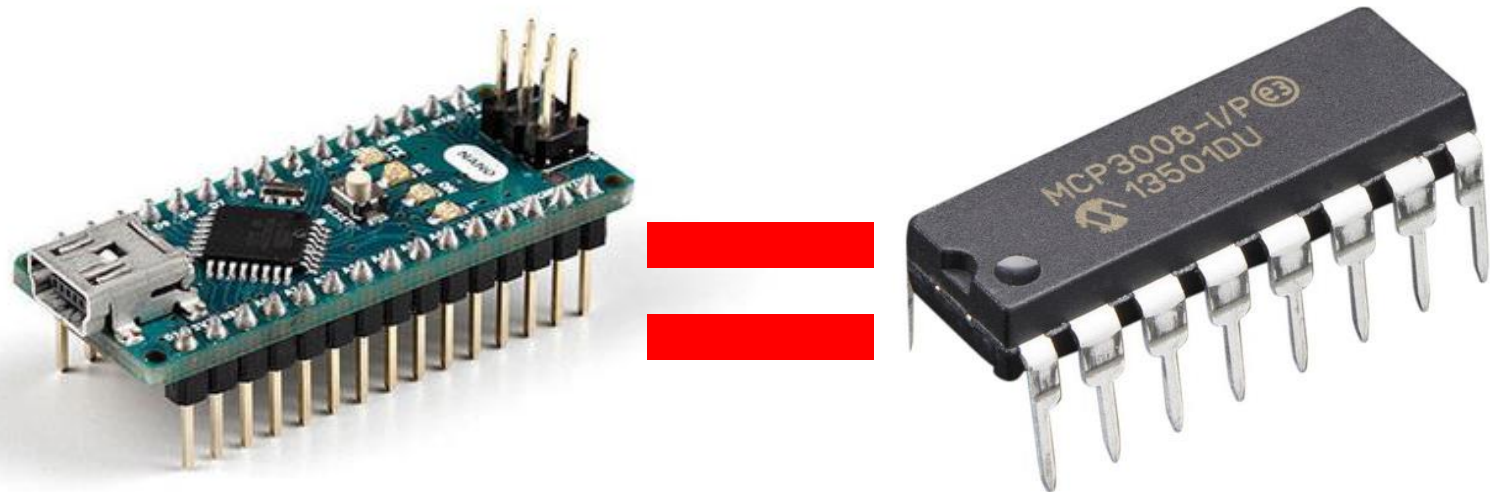
부대 내 일반PC는 사이버지식방 밖에 없음.
(사용성이 매우 떨어짐)



따라서 관리 간부의 **스마트폰**을 통해
서버에 접속하고 현황을 관리하여야 함.



아두이노 미사용



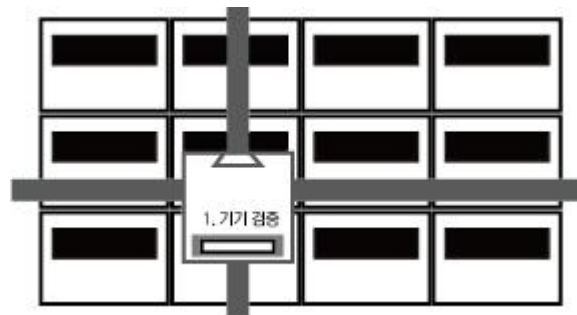
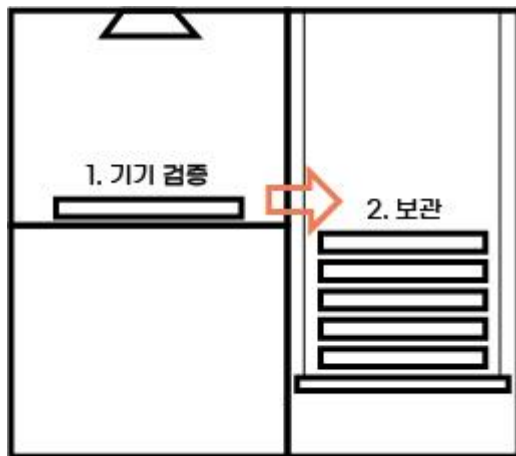
아두이노의 GPL 라이선스로부터도

자유로워 질 수 있음

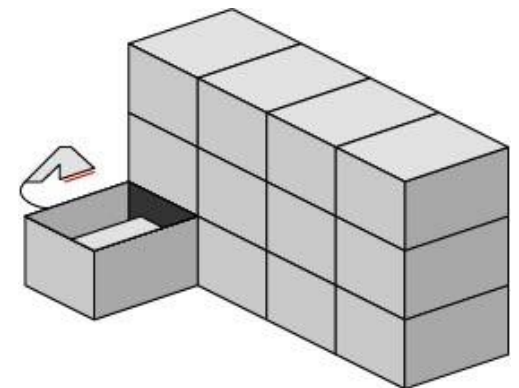
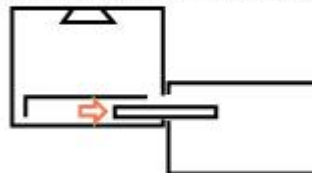
1 : N 반납기

기기 8대 : 반납기 1대

8만원 X 36만명 = 288억
8만원 X 36만명 / 8 = 36억 +@



2. 검증기에서 보관함으로 밀어넣는다.



이외에도...





자세한 내용은 [깃북 참조](#) 부탁드립니다!

감사합니다.



국방 모바일 반납