

GATEWAY Smart Door lock

개발 기간	2020.08 ~2020.10
개발 환경	Raspberry Pi, Android Studio, Django, SQLite, AWS S3
개발 인원	4명
기능	1) 스마트폰 어플리케이션을 통한 도어락 제어 <ul style="list-style-type: none"> - NFC/RFID 태그 이용 잠금 해제 - 원격 잠금 해제 - 잠금 해제 로그 조회 - 기기 추가, 삭제 등 2) 움직임 감지 시 비디오 녹화 및 저장

◆ [Appendix] Detailed Implementation Spec

A. Authorization Management Module (인증 관리 모듈)

a. Authorization (고유번호 인증)

앱에서의 최초 로그인을 위해 필요한 도어락 고유번호의 검증 및 *signed* 토큰 발급을 요청하는 API이다. HTTP GET Method를 사용하며, 도어락의 고유번호 String값을 파라미터로 전송하면 고유번호의 유효성을 검증하고, 유효할 시 *signed* 토큰을 발급한 뒤 유효성 여부(boolean)와 토큰(string)을 json 형태로 반환한다. 필요한 파라미터가 전달되지 않으면 400_BAD_REQUEST NoRequireParameter 에러 메시지를 발생시킨다.

- HTTP Method : GET
- URL : /api/auth
- Request Example : {door_id} = 'A384%SUI9P'

```
Get /api/auth/ HTTP/1.1
Content-Type : applicaiton/json
QueryString : {
    'door_id' : 'A384%SUI9P'
}
```

- Response Data (Success - Door is exist)

```
HTTP/1.1 200 OK
{
    'is_door_exist' : 'True', // 도어락 고유번호의 유효성 여부
    'Authorization': 'Token 6n54583bjcam4ced9f03s49527a3264c' // signed 토큰
}
```

- Response (Fail)

```
HTTP/1.1 400_BAD_REQUEST
{
  'error' : 'NoRequiredParameter',
  'date' : '2020-10-01 12:00:00'
}
```

B. Setting Module (자동 녹화 설정 관리 모듈)

a. Recording Setting (자동 녹화 기능 설정 변경)

비디오 자동 녹화 기능을 on/off 설정을 통해 변경을 요청하는 API이다. HTTP PUT Method를 사용하며, 변경하고자 하는 상태로 Boolean값을 파라미터로 전송하면 데이터베이스 내의 값이 변경된다. 필요한 파라미터가 전달되지 않으면 400_BAD_REQUEST NoRequireParameter 에러 메시지를 발생시킨다.

- HTTP Method : PUT
- URL : /api/setting
- Request Example : {recording} = True

```
PUT /api/setting HTTP/1.1
Content-Type : applicaiton/json
Header : {
  Authorization
}
Data : {
  'recording' : 'True'
}
```

- Response Data (Success)

```
HTTP/1.1 200 OK
{
  'recording' : 'True' // 자동 녹화 기능의 상태
}
```

- Response (Fail)

```
HTTP/1.1 400_BAD_REQUEST
{
  'error' : 'NoRequiredParameter',
  'date' : '2020-10-01 12:00:00'
}
```

b. View Recording Setting (자동 녹화 기능 설정 조회)

비디오 자동 녹화 기능의 on/off 설정 상태 조회를 요청하는 API이다. HTTP GET Method를 사용하여 요청하면 데이터베이스에 저장된 현재 상태를 JSON 형식으로 반환한다. 실패시 404_NOT_FOUND 에러 메시지를 발생시킨다.

- HTTP Method : GET
- URL : /api/setting
- Request Example

```
GET /api/setting/ HTTP/1.1
Content-Type : applicaiton/json
Header : {
  Authorization
}
```

- Response Data (Success)

```
HTTP/1.1 200 OK
{
  'recording' : 'True' // 자동 녹화 기능의 상태
}
```

- Response (Fail)

```
HTTP/1.1 404_NOT_FOUND
{
}
```

C. Video Management Module (비디오 관리 모듈)

a. Video Recording (비디오 녹화 및 저장)

카메라 센서를 통해 녹화한 영상을 데이터베이스와 클라우드 저장소(AWS S3)에 저장을 요청하는 API이다. 적외선 센서가 사람을 인식하여 자동으로 영상을 녹화하면 내부적으로 호출되며, 영상이 녹화된 시간을 전달하면 데이터베이스에 비디오 이름과 생성된 시간, 클라우드 저장소(AWS S3) 링크가 저장되며 201_CREATED 메시지를 반환한다. 필요한 파라미터가 전달되지 않으면 400_BAD_REQUEST NoRequiredParameter 에러 메시지를 발생시킨다.

- HTTP Method : POST
- URL : /api/video

```
POST /api/video HTTP/1.1
Content-Type : applicaiton/json
Header : {
    Authorization
}
Data : {
    'video' : '(video data)'
}
```

- Response Data (Success)

```
HTTP/1.1 201 CREATED
{
    'vid_id' : '1', // 저장된 비디오 id
    'created' : '2020-10-01 00:00:00', // 비디오가 생성된 시간
    's3link' : 'https://amazon.s3.bucket/2020-10-01%..' // 클라우드 저장소 링크
}
```

- Response (Fail)

```
HTTP/1.1 400_BAD_REQUEST
{
    'error' : 'NoRequiredParameter',
    'date' : '2020-10-01 00:00:00'
```

```
}
```

b. Video List (비디오 목록 조회)

저장된 비디오의 목록 조회를 요청하는 API이다. HTTP GET Method를 사용하며, 데이터베이스에 저장된 비디오 목록을 json array 형태로 전달한다. 저장된 비디오가 없을 경우 에러 메시지를 전달하는 것이 아닌 빈 Data 목록을 전달한다.

- HTTP Method : GET
- URL : /api/video

```
GET /api/video HTTP/1.1
Content-Type : applicaiton/json
Header : {
    Authorization
}
```

- Response Data (Success)

```
HTTP/1.1 200 OK
{
    [
        {
            'vid_name' : '~~~~', 'created' : '2020-10-01 00:00:00',
            'vid_time': '00:21', 'thumb': 'https://~~~~',
            'vid_name' : '~~~~', 'created' : '2020-10-02 10:00:00',
            'vid_time': '00:32', 'thumb': 'https://~~~~'
        }
    ] // 비디오 목록
}
```

c. Video Download (비디오 확인 및 다운로드)

비디오가 저장된 클라우드 저장소(AWS S3)의 링크를 요청하는 API이다. HTTP GET Method를 사용하며 다운로드하고자 하는 파일의 id를 URI에 포함하여 전달하면 저장된 해당 비디오를 다운로드할 수 있는 클라우드 저장소(AWS S3)의 링크(s3link)를 받아온다.

존재하지 않는 비디오 아이디(vid_name)를 요청하면 404_NOT_FOUND 에러 메시지를 발생시킨다.

- **HTTP Method : GET**
- **URL : /api/video/{vid_name}**
- **Request Example : {vid_name} = '~~~~'**

```
GET /api/video/1 HTTP/1.1
Content-Type : applicaiton/json
Header : {
    Authorization
```

- **Response Data (Success)**

```
HTTP/1.1 200 OK
{
    's3link' : 'https://amazon.s3.bucket/2020-10-01%..' // 클라우드 저장소 링크
}
```

- **Response (Fail)**

```
HTTP/1.1 404_NOT_FOUND
{
}
```

d. Video Delete (비디오 수동 삭제)

저장된 비디오를 선택하여 영구적으로 삭제하는 API이다. HTTP DELETE Method를 사용하며, 삭제하고자 하는 파일의 이름을 URI에 포함하여 전달하면 django의 queryset 필터를 통해 해당 파일이 데이터베이스와 클라우드 저장소(AWS S3)에서 영구히 삭제된다. 존재하지 않는 비디오 아이디(vid_name)를 요청하면 404 NOT_FOUND 에러 메시지를 발생시킨다.

- **HTTP Method : DELETE**
- **URL : /api/video/{vid_name}**
- **Request Example : {vid_name} = '~~~~'**

```
DELETE /api/video/1 HTTP/1.1
Content-Type : applicaiton/json
Header : {
    Authorization
}
```

- Response Data (Success)

```
HTTP/1.1 200 OK
{
}
```

- Response (Fail)

```
HTTP/1.1 404_NOT_FOUND
{
}
```

e. Video Delete (Auto) (비디오 자동 삭제)

저장된 비디오 중 일정 시간이 지난 비디오를 자동으로 삭제하는 API이다. (crontab을 이용하여 매일 자정에 자동으로 호출된다.) HTTP DELETE Method를 사용하며, 내부적으로 비디오가 생성된 시간(created)와 현재 시간을 비교하여 7일 이상의 차이가 발생하면 해당 비디오를 데이터베이스와 저장공간에서 영구적으로 삭제한다. 삭제할 파일이 없는 경우에도 HTTP 에러 메시지를 발생시키지 않는다.

- HTTP Method : DELETE
- URL : /auto/video
- Request Example

```
DELETE /auto/video HTTP/1.1
Content-Type : applicaiton/json
Header : {
    Authorization
}
```

- Response Data (Success)

```
HTTP/1.1 200 OK
```

```
{  
}
```

D. Device Management Module (기기 관리 모듈)

a. Add Device (기기 추가 요청)

앱에서 기기 추가 버튼을 눌렀을 때 호출되는 API이다. 이 API가 호출된 후에 도어락 해제 프로세스에서 다음에 태그된 RFID 장치를 새로운 기기로 등록한다. (POST /api/device API를 호출한다.)

- HTTP Method : PUT
- URL : /api/device/request
- Request Example

```
POST api/device/ HTTP/1.1  
Content-Type : applicaiton/json  
Header : {  
    Authorization  
}
```

- Response Data (Success)

```
HTTP/1.1 200 OK  
{  
}
```

- Response (Fail)

```
HTTP/1.1 400_BAD_REQUEST  
{  
    'error' : 'NoRequiredParameter',  
    'date' : '2020-10-01 12:00:00'  
}
```


b. Add Device (기기 추가)

도어락을 관리할 수 있는 RFID 기기 추가를 요청하는 API이다. 기기 추가 상태에서 RFID 태그가 발생하였을 때 내부적으로 호출된다. HTTP POST Method를 사용하며, RFID 태그 번호를 받아온다. 자원 생성이 정상적으로 이루어지면 201_CREATED 메시지와 저장된 정보를 json형식으로 성공적으로 수행하였음을 알린다. 리소스 생성에 필요한 정보가 누락된 경우에는 400_BAD_REQUEST NoRequireParameter 에러 메시지를 발생시킨다.

- HTTP Method : POST
- URL : /api/device/
- Request Example

```
POST api/device/ HTTP/1.1
Content-Type : applicaiton/json
Header : {
    Authorization
}
Data : {
    'rfid_id' : 'RFID tag ID'
}
```

- Response Data (Success)

```
HTTP/1.1 201 CREATED
{
    'device_id' : '1', // 기기 고유 번호
    'rfid_id' : 'RFID tag ID', // RFID 태그 아이디
    'created' : '2020-10-01 12:00:00', // 생성 시각
}
```

- Response (Fail)

```
HTTP/1.1 400_BAD_REQUEST
{
    'error' : 'NoRequiredParameter',
    'date' : '2020-10-01 12:00:00'
```

```
}
```

c. Delete Device (기기 삭제)

데이터베이스에 저장된 RFID 기기 삭제를 요청하는 API이다. HTTP DELETE Method를 사용하며, 삭제하고자 하는 디바이스의 고유 ID(device_id)를 URL에 포함하여 전달하면 해당 기기를 데이터베이스에서 삭제한다. 존재하지 않는 device_id를 전달하면 400_BAD_REQUEST InvalidParameter 에러 메시지를 발생시킨다.

- **HTTP Method : DELETE**
- **URL : /api/device/{device_id}**
- **Request Example : {device_id} = '1'**

```
DELETE /api/device/1 HTTP/1.1
Content-Type : applicaiton/json
Header : {
    Authorization
}
```

- **Response Data (Success)**

```
HTTP/1.1 200 OK
{
}
```

- **Response (Fail)**

```
HTTP/1.1 400_BAD_REQUEST
{
    'error' : 'InvalidParameter',
    'date' : '2020-10-01 12:00:00'
}
```

d. View Device (기기 목록 조회)

장치를 관리할 수 있는 RFID 기기 목록 조회를 요청하는 API이다. HTTP GET Method를 사용하며, 데이터베이스에 저장된 기기 목록을 json array형태로 전달한다. 등록된 기기가 없을 경우 에러 메시지를 발생시키는 것이 아닌 빈 Data 목록을 전달한다.

- HTTP Method : GET
- URL : /api/device

```
GET /api/device/ HTTP/1.1
Content-Type : applicaiton/json
Header : {
    Authorization
```

- Response Data (Success)

```
HTTP/1.1 200 OK
{
    [
        {'device_id' : '1', 'rfid_id' : 'RFID tag ID', 'created' : '2020-10-01 00:00:00'},
        {'device_id' : '2', 'rfid_id' : 'RFID tag ID', 'created' : '2020-10-01 00:00:00'},
    ]
}
```

E. Remote Unlock Module (원격 잠금 해제 관리 모듈)

a. Remote Unlock (원격 잠금 해제)

원격으로 잠금 장치 상태의 변경을 요청하는 API이다. HTTP POST Method를 사용하며, 기본적으로는 잠금(state = True)상태이고 데이터 전달을 통해 상태를 수정한다. 이 때 안드로이드 디바이스의 정보(device_name)를 포함하여 변경하고자 하는 상태를 state를 통해 전달하여야 하며, 전달되지 않는다면 400_BAD_REQUEST NoRequireParameter 에러 메시지를 전달한다.

- HTTP Method : POST
- URL : /api/remote

- Request Example

```
POST /api/remote/ HTTP/1.1
Content-Type : applicaiton/json
Header : {
    Authorization
}
Data : {
    'device_name' : 'Galaxy S10'
}
```

- Response Data (Success)

```
HTTP/1.1 200 OK
{
}
```

- Response (Fail)

```
HTTP/1.1 400_BAD_REQUEST
{
    'error' : 'NoRequiredParameter',
    'date' : '2020-10-01 12:00:00'
}
```

b. View Unlock History (원격 잠금 해제 기록 조회)

원격으로 잠금장치의 상태를 변화시킨 기록이 있는 장치와 해당 시간 목록을 요청하는 API이다. HTTP GET Method를 사용하며, 데이터베이스에 저장된 기기 목록을 json array 형태로 전달한다. 해당하는 기록이 없을 경우 에러 메시지를 발생시키는 것이 아닌 빈 Data 목록을 전달한다.

- HTTP Method : GET

- URL : /api/remote

```
GET /api/remote/ HTTP/1.1
Content-Type : applicaiton/json
Header : {
```

Authorization

```
}
```

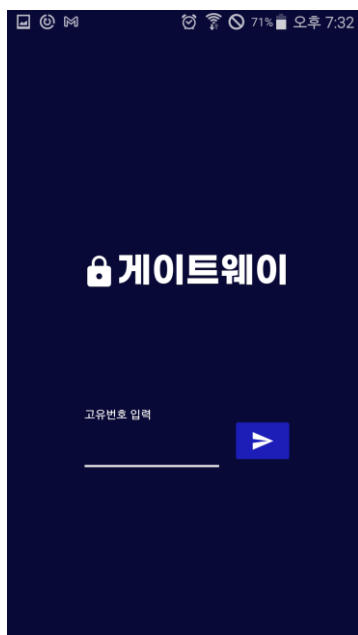
- Response Data (Success)

HTTP/1.1 200 OK

```
{  
  [  
    {'device_name' : 'Galaxy S10', 'created' : '2020-10-01 00:00:00'},  
    {'device_name' : 'Galaxy S7', 'created' : '2020-10-03 00:00:00'}  
  ] // 원격 잠금 해제한 적 있는 장치, 시간 목록  
}
```

◆ [Appendix] User Manual

1. 로그인



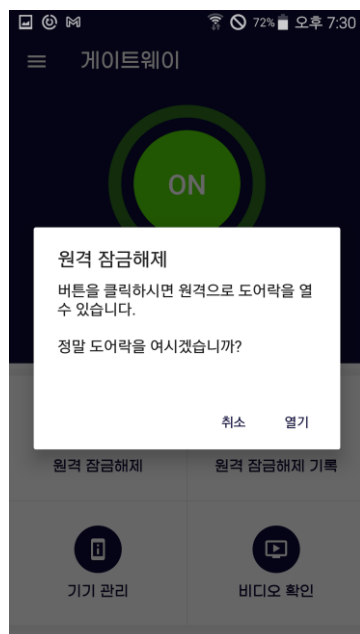
서비스에 등록된 도어락의 고유 번호를 입력하여 앱에 로그인합니다.

2. NFC/RFID 태그를 이용한 도어락 잠금 해제



[ON] 버튼을 터치하면 태그를 이용해 도어락을 잠금 해제할 수 있는 상태가 됩니다. 5초 후 자동으로 비활성화가 이루어집니다. 만약, [OFF] 버튼을 터치하면 5초 전에 비활성화할 수 있습니다.

3. 원격 제어 기능을 이용한 도어락 잠금 해제



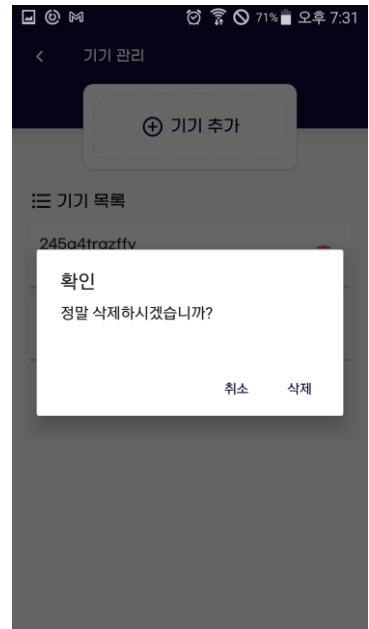
[원격 잠금해제] 메뉴를 선택하면 다음과 같은 팝업이 나타납니다. 이때, [열기]를 선택하면 원격으로 도어락을 잠금 해제할 수 있습니다.

4. 도어락 잠금 해제 기록 확인



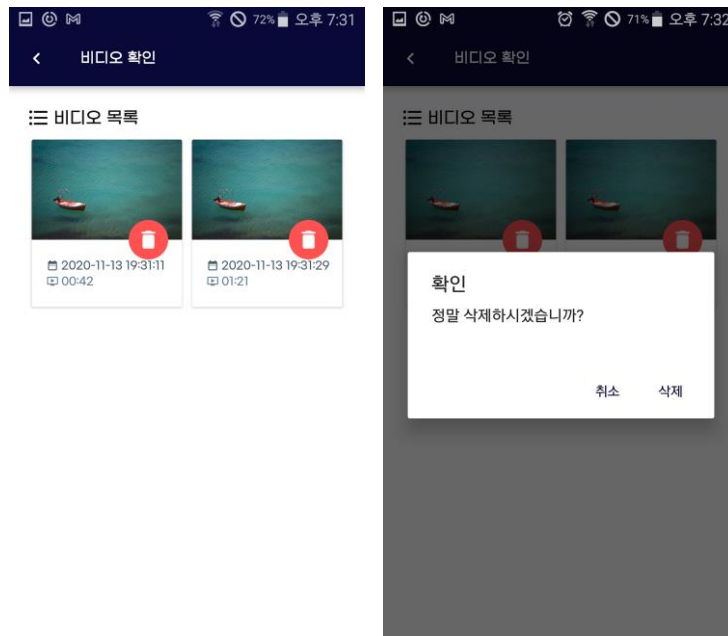
[원격 잠금해제 기록] 메뉴를 선택하여 원격으로 잠금 해제를 시도한 기기와 날짜 목록을 조회합니다.

5. 스마트폰(기기) 관리



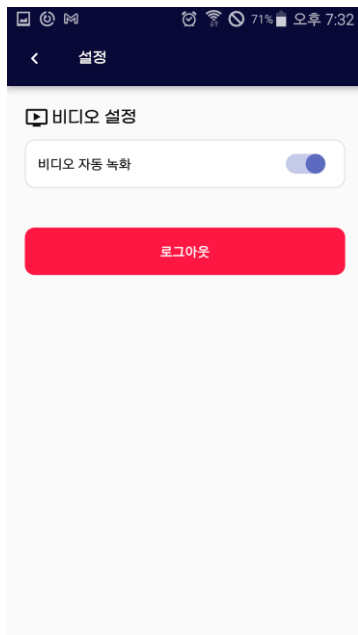
[기기 관리] 메뉴를 선택하면 해당 도어락에 접근할 수 있는 기기(스마트폰)의 목록이 나타납니다. [기기 추가] 버튼을 누른 후 추가할 기기를 도어락에 태그하면 추가가 완료됩니다. [휴지통] 아이콘을 눌러 기기의 권한을 삭제할 수 있습니다.

6. 비디오 확인 및 관리



[비디오 확인] 메뉴에서 녹화된 비디오를 재생하고 삭제할 수 있습니다. 오래된 비디오는 특정 기간 이후 자동으로 삭제됩니다.

7. 로그아웃



메인 화면 좌측 상단의 햄버거 메뉴를 누르면 [설정] 페이지에 접근할 수 있습니다. 이곳에서 비디오 자동 녹화 기능을 off할 수 있으며, 로그아웃을 할 수 있습니다.