

Spyder

사진분석을 이용한 군장병 기밀 유출 보안 솔루션



Team 스파이더

Team



조용인 17사단 507여단 2대대 연수구지역대 상병

Harvard University
하버드대학교 (재학)
Biomedical Engineering / Computer Science
생명공학 / 컴퓨터 과학
기획, Backend, Frontend, 문서, 영상 담당



김진 수도군단 100정보통신단 701 통신대대 병장

Chungbuk Semiconductor Highschool
충북반도체고등학교 (졸업)
Semiconductor
반도체과
기획, Frontend, 문서 담당
Spyder

Problem

I

2022년 3월, **KBS 뉴스의**
인터뷰로 인해 우크라이나
국제군단의 소재지가 노출되는
사건이 있었습니다.

우크라이나 군은 **러시아의**
폭격에 당할 수 있는 큰 위험에
놓였습니다.



II

우크라이나-러시아 전쟁 중
해커가 매력적인 여성 행세로
러시아 군인에게 접근하였고
사진을 전송하도록 유도했습니다.

결국 러시아군이 보낸 사진으로
러시아군 **기지의 위치가 노출**이
되었습니다.



III

휴대폰 사용으로 인해 **병영생활**
에 대한 문제들이 많이 해결된
점을 고려하였을 때, **휴대폰 사용**
의 금지는 해결방법이 아닙니다.

모바일 기기는 사용하되 **보안**
문제를 해결하는 방향으로
가야합니다.



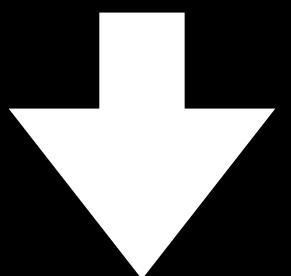
Spyder

사진과 이미지, 군 보안에 치명적인 위험 요소

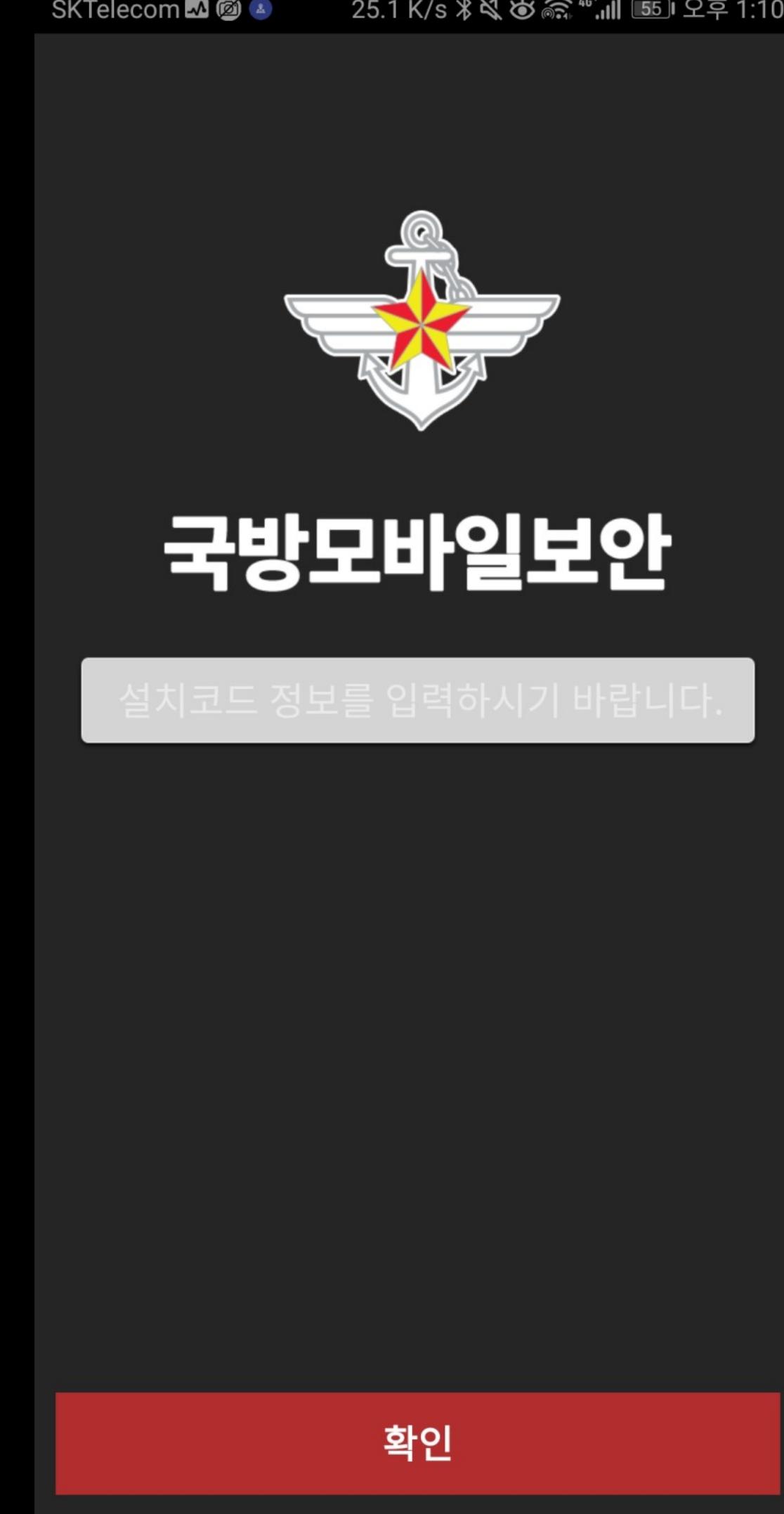
주요 방송사도 실수하는 **기밀 유출**,
군 장병이 **보안**을 지킬 수 있을까요?

기존 솔루션
국방모바일보안 앱

1. 카메라를 원천 차단하면 발생할 수 있는 불편함
2. 비콘이 있는 위병소로 직접 가서 앱을 활성화하고 비활성화 해야 하는 비효율적인 방식
3. 앱의 정상작동 유무확인을 위해 간부들이 직접 일일이 병사들의 모바일 기기를 확인해야 하는 방식



“매우 불편한 앱”



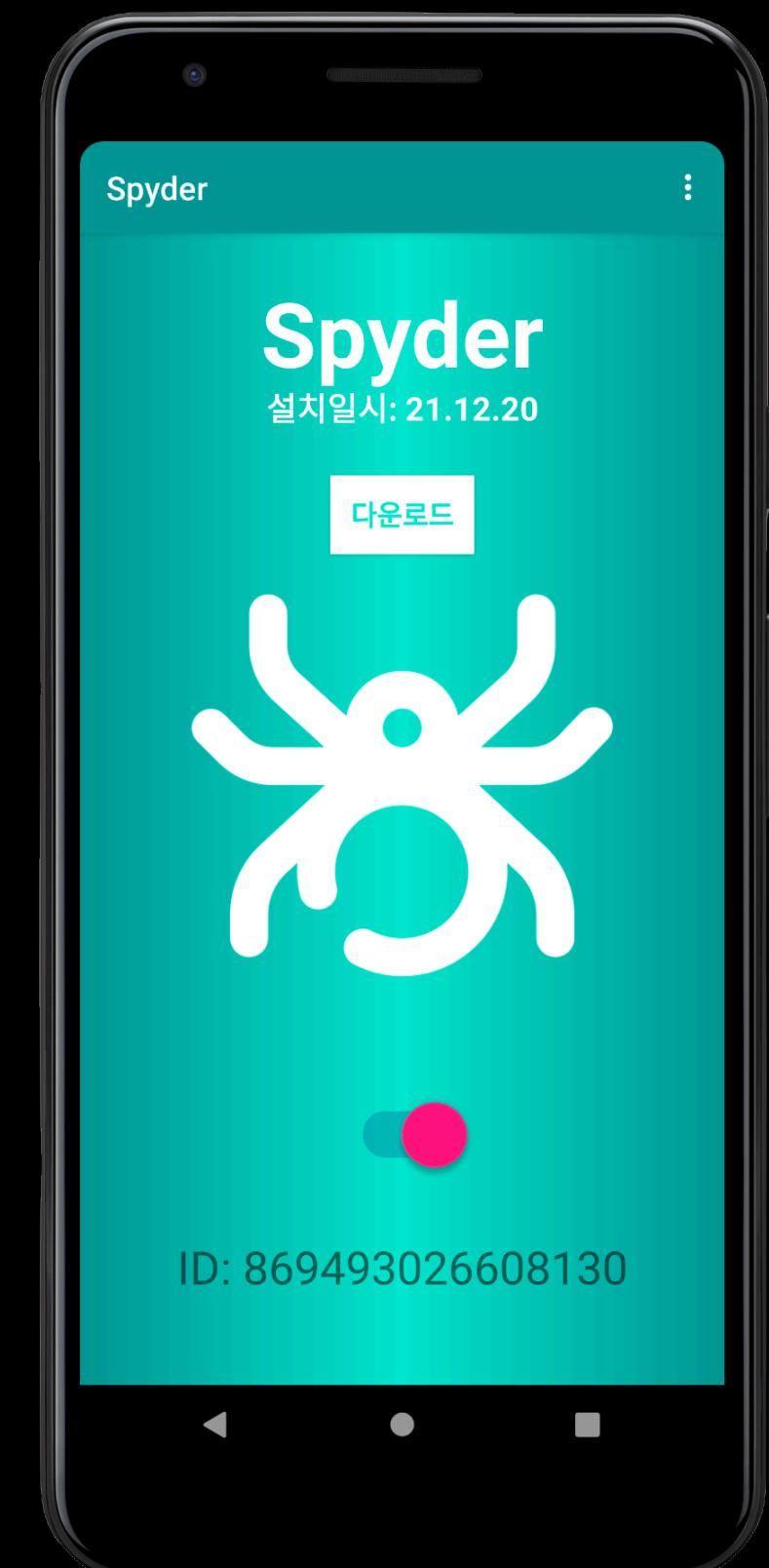
Spyder



Spyder

‘사진분석을 이용한 군장병 기밀 유출 보안 솔루션’

APP



WEB

Spyder

FUNCTION APP



Spyder

Spyder APP 기능 설명

Spyder



GPS 강제 ON

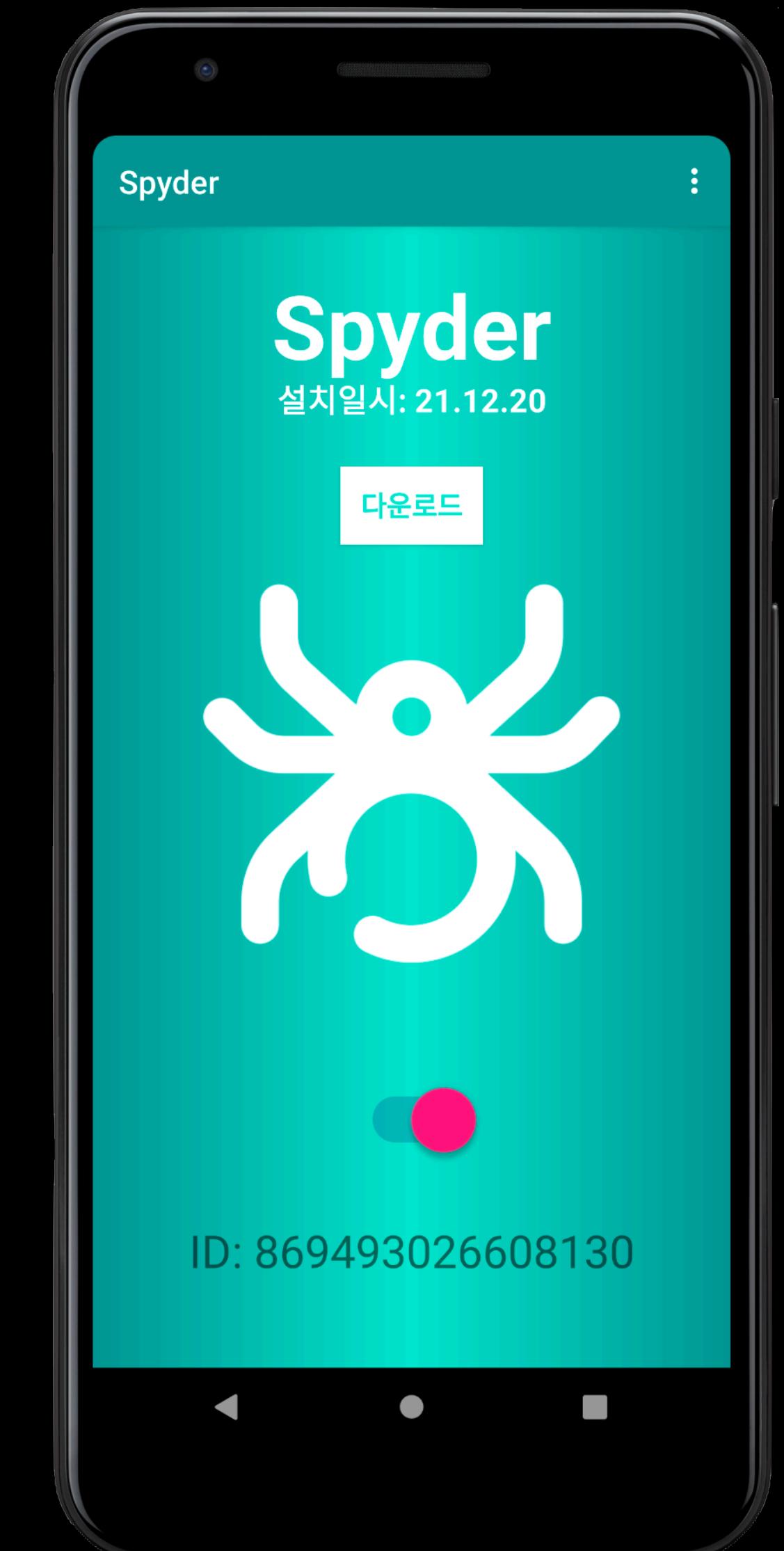
Spyder는 사진 분석을 위해 **GPS 데이터**를 이용합니다.

군 장병들이 임의로 GPS를 **비활성화** 하여 앱의 기능을 저하시키지 못하도록 **GPS 강제 ON 기능**을 넣었습니다.

부대 GPS내에서 찍은 사진 감지

Spyder는 **EXIF Analysis**의 GPS 데이터를 이용해 사진을 분석합니다.

만약 군 장병이 찍은 사진의 GPS 데이터가 **Admin(간부)**이 설정한 GPS 안이라면, 그 사진을 **감지합니다**.





서버 업로드 및 사진 삭제

EXIF Analysis로 감지된 사진은 Spyder 서버에
업로드 시킵니다.

장병들의 **모바일 기기**에서 감지된 사진은 **삭제됩니다**.

확인된 사진 서버에서 다운로드

Admin WEB에서 확인된 사진은 Spyder 서버에서
다운로드 할 수 있습니다.

이렇게 확인된 사진은 장병들의 **모바일 기기**에서
사진 원본 확인이 가능합니다.



FUNCTION
WEB



Spyder

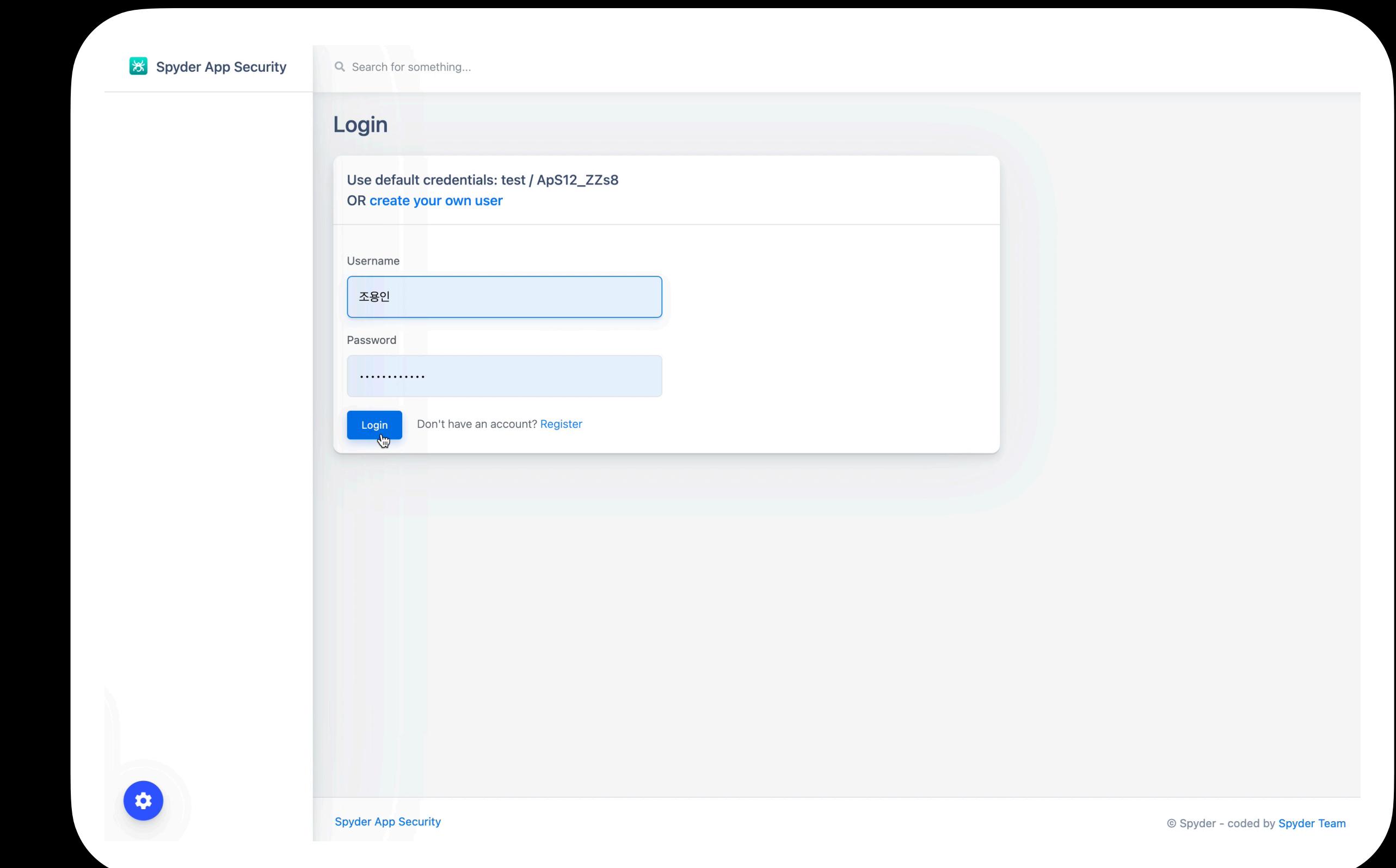
Spyder **WEB** 기능 설명

Spyder



관리자 로그인

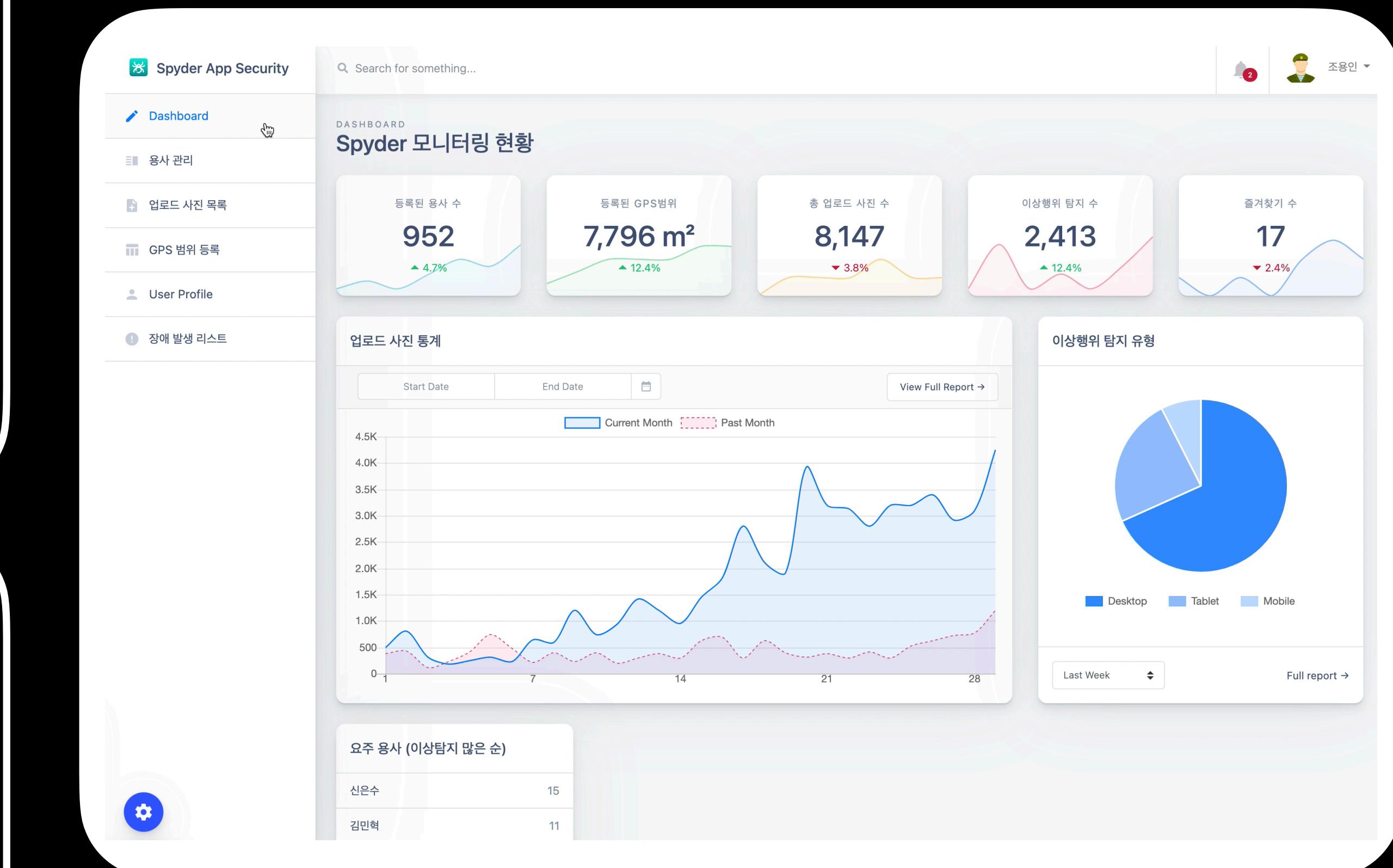
Spyder WEB은
간부와 관리자를 위해
간편한 Admin
계정 생성 및 로그인이
가능합니다.





현황 종합

등록된 **용사** 수, 등록된 **GPS** 범위, 총 **업로드 사진** 수, **이상행위** 탐지 수, **즐겨찾기** 수의 현황과 증가 감소 %를 그래프로 **시각화** 하였습니다.



업로드 사진 통계

업로드 사진 통계는 기간 조정을 통해 **그래프**로 **시각화**하여 손쉽게 **모니터링**이 가능합니다.

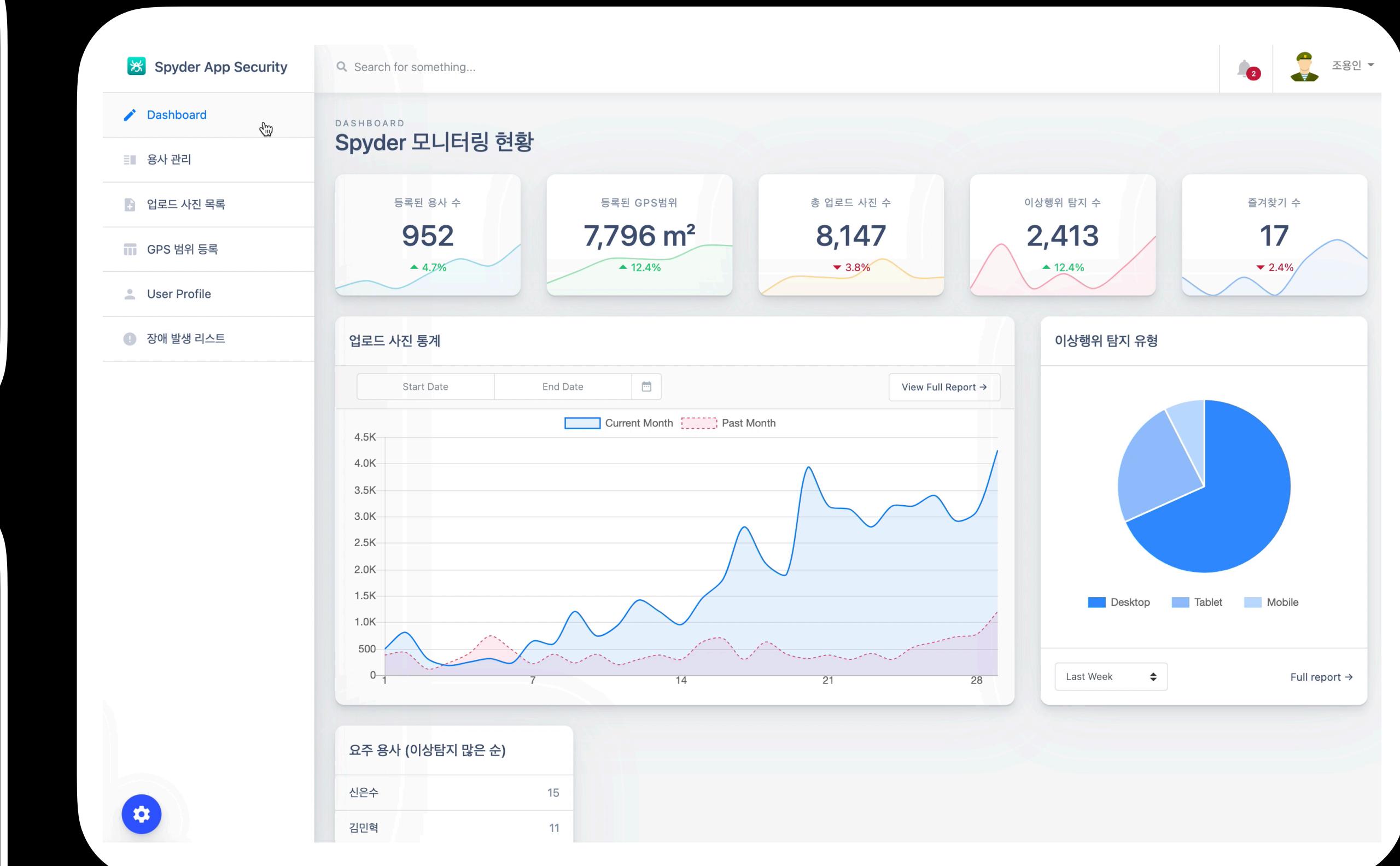


이상행위 탐지 유형

이상행위 탐지가 어떤 **모바일 기기**에서 탐지 되었는지 확인이 가능합니다.

요주의 용사

이상탐지가 많은 순으로 **요주의 용사**를 확인할 수 있습니다.





용사 관리 리스트

등록된 모든 용사 기기의 **IMEI 숫자**를 통해 리스트가 보이며, 활성화 된 유저, **24시간 내**에 앱이 비활성화 된 유저, 앱이 **비활성화** 된 유저로 구분하여 확인이 가능합니다.

The screenshot shows the 'User Management' section of the Spyder App Security web application. On the left, a sidebar menu includes 'Dashboard', 'User Management' (which is active), 'Upload Photo Log', 'GPS Location Log', 'User Profile', and 'Recent Events'. The main content area has a search bar at the top. Below it, there are three tables: 'Active Users (활성화 유지)', 'Partial Inactive Users in 24hr (앱이 꺼져있었던 유저)', and 'Inactive Users (앱이 꺼져있는 유저)'. Each table has columns for #, 이름 (Name), 소속부대 (Unit), 계급 (Rank), 직책 (Position), 군번 (Service Number), and 핸드폰 번호 (Phone Number). The 'Active Users' table lists four users: 조용인, 김민수, 박진우, and 김기범. The 'Partial Inactive Users' table lists two users: 조윤형 and 서강준. The 'Inactive Users' table lists three users: 김민혁, 신은수, and 안형준.

#	이름	소속부대	계급	직책	군번	핸드폰 번호
1	조용인	17사단 507여단 2대대 연수구지역대	상병	인사행정병	21-71904874	010-2389-2229
2	김민수	17사단 507여단 2대대 3중대	대위	3중대장	15-16571	010-4829-5811
3	박진우	17사단 507여단 2대대 8중대	일병	무기관리병	21-71902031	010-1192-5822
4	김기범	17사단 507여단 2대대 1중대	병장	초병	21-71904032	010-9957-2834

#	이름	소속부대	계급	직책	핸드폰 번호	접속 끊긴 시간	접속 복구 시간	
1	조윤형	17사단 507여단 2대대 3중대	일병	상황병	22-71909932	010-4883-2938	2022-10-27 19:00:00	2022-10-27 19:05:00
2	서강준	17사단 507여단 2대대 10중대	중사	소대장	00-529492	010-5822-5994	2022-10-28 11:34:22	2022-10-28 11:40:15

#	이름	소속부대	계급	직책	군번	핸드폰 번호	접속 끊긴 시간	
1	김민혁	17사단 507여단 2대대 1중대	상병	초병	22-71905821	010-5824-3990	2022-10-27 20:00:00	
2	신은수	17사단 507여단 2대대 8중대	중사	소대장	00-529953	010-9898-3528	2022-10-28 11:34:22	
3	안형준	17사단 507여단 2대대 5중대	병장	행정병	21-71902044	010-5733-2999	2022-10-28 03:42:19	

앱 ON/OFF 기록

유저들의 앱이 얼마나 **비활성화** 되어있었는지 확인이 가능합니다.



GPS 범위 등록

Admin(간부)은 **보안 구역**을
GPS 범위로 설정할 수 있고,
이 구역 내에서 찍힌 모든
사진은 Spyder WEB으로
업로드 됩니다.

The screenshot shows the 'GPS 범위 등록' (GPS Range Registration) section of the Spyder App Security web interface. On the left, a sidebar lists 'Dashboard', '용사 관리', '업로드 사진 목록', 'GPS 범위 등록' (with a cursor pointing at it), 'User Profile', and '장애 발생 리스트'. The main area displays a table titled 'GPS 범위 등록' with one entry:

#	이름	위도	경도	등록자	등록 시간
1	17사단 507여단 2대대 지역	37.563216232190754 ~ 37.566819703831484	126.97058884948852 ~ 126.97868621727491	조용인 인 사행정관	2022-10-24 20:13:52

To the right, there's a 'Categories' section with checkboxes for '1급 보안 구역' (checked), '2급 보안 구역' (checked), and '3급 보안 구역' (unchecked). A 'New category' button is also present. The footer of the page reads 'Spyder App Security' and '© Spyder - coded by Spyder Team'.



실시간 사진 업로드

보안구역 내에서 찍힌 사진은
실시간으로 업로드 됩니다.

ML 1차적 검증

업로드 된 사진은 1차적으로
OpenCV 사진분석을 통해
무기가 감지됩니다.

관리자 2차적 검증

간부들의 2차적 **수동적인 확인**을 통해 업로드 된 사진을
수락/거절을 할 수 있습니다.

The screenshot shows the 'SPYDER APP SECURITY' dashboard. On the left, a sidebar includes 'Dashboard', '용사 관리', '업로드 사진 목록' (selected), 'GPS 범위 등록', 'User Profile', and '장애 발생 리스트'. The main area displays a grid of uploaded photos with AI detection results:

- AI DETECTED**: A photo of a person holding a rifle with a green bounding box and red text 'AI DETECTED'.
- AI DETECTED**: A photo of many birds in flight with a blue bounding box and red text 'AI DETECTED'.
- CLEAN**: A photo of the Great Pyramids with a green bounding box and red text 'CLEAN'.
- AI DETECTED**: A photo of a person holding a rifle with a red bounding box and red text 'GUN'.
- AI DETECTED**: A photo of a hand holding a gun with a green bounding box and red text 'AI DETECTED'.
- CLEAN**: A photo of a colorful pixelated image with a green bounding box and red text 'CLEAN'.

Each card includes user information (e.g., '용사 : 김민수'), a timestamp ('2022.10.29'), and two buttons: '확인' (Confirm) and '거절' (Reject).

**DEVELOP
APP/WEB**



Spyder

Spyder **APP/WEB** 개발 과정

Spyder

DEVELOP APP

APP DEVELOP PROCESS



Spyder

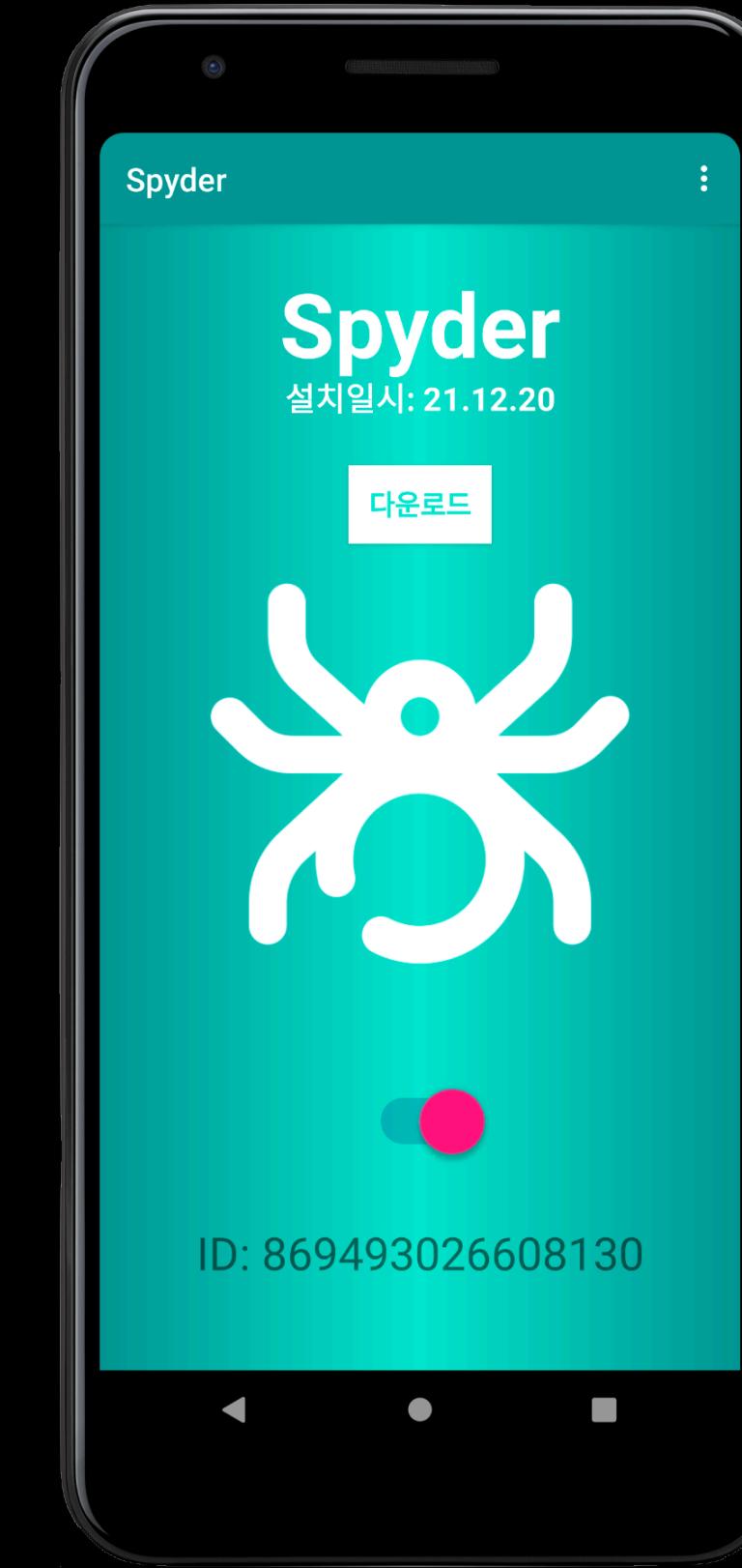
```
private void onClick(View view) {
    @Override
    public void onClick(View view) {
        System.out.println("download");
        // DOWNLOAD ALGORITHMS!!!!!!
        Toast.makeText(getApplicationContext(), text: "All downloaded", Toast.LENGTH_SHORT).show();
    }
}

private void initView() {
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    TelephonyManager telephonyManager = (TelephonyManager) getSystemService(Context.TELEPHONY_SERVICE);
    @SuppressLint("MissingPermission") String deviceId = telephonyManager.getDeviceId();
    TextView textView = findViewById(R.id.editTextPersonName2);
    textView.setText("ID: " + deviceId);
    System.out.println(deviceId);
}

private GoogleApiClient getApiClientInstance() {
    GoogleApiClient mGoogleApiClient = new GoogleApiClient.Builder(this)
        .addApi(LocationServices.API).build();
    return mGoogleApiClient;
}

private void requestGPSSettings() {
    LocationRequest locationRequest = LocationRequest.create();
    locationRequest.setPriority(LocationRequest.PRIORITY_BALANCED_POWER_ACCURACY);
    locationRequest.setInterval(2000);
    locationRequest.setFastestInterval(500);
    LocationSettingsRequest.Builder builder = new LocationSettingsRequest.Builder().addLocationRequest(locationRequest);
}
```



Android Studio와 Java를 사용해 앱을 개발하였고,
Figma로 프론트엔드 디자인 후 적용하였습니다.
안드로이드 시뮬레이터로 테스트를 진행하였습니다.

Spyder

DEVELOP
WEB

WEB DEVELOP PROCESS



The screenshot shows the AWS EC2 Instances page. The instance summary for 'i-00fc0881237c1d85 (spyder)' is displayed. Key details include:

- Instance ID: i-00fc0881237c1d85 (spyder)
- Public IPv4 address: 13.125.225.22
- Private IPv4 addresses: 172.31.33.132
- Instance state: Running
- Public IPv4 DNS: ec2-13-125-225-22.ap-northeast-2.compute.amazonaws.com
- Private IP DNS name (IPv4 only): ip-172-31-33-132.ap-northeast-2.compute.internal
- Instance type: t2.micro
- VPC ID: vpc-dfff5cb4
- Subnet ID: subnet-840809c8
- IAM Role: -
- Auto Scaling Group name: -

The screenshot shows the pgAdmin interface. On the left, an ER diagram is displayed with tables like 'spyder_admin_tbl', 'spyder_user_tbl', 'spyder_setting_tbl', and 'spyder_photo_tbl'. On the right, the PostgreSQL monitoring interface shows various metrics and session details.

ER Diagram Tables:

- spyder_admin_tbl
- spyder_user_tbl
- spyder_setting_tbl
- spyder_photo_tbl

pgAdmin Monitoring Metrics:

- Transactions per second: 10
- Block I/O: 1200
- Server activity sessions: 7
- Sessions table (partial):

ID	PID	Database	User	Application	Client	Backend start	Transaction start	State	Wait event	Blocking PIDs
1	15462	postgres	postgres	pgAdmin 4 - DB postg...	172.31.33.132	2022-10-16 07:38:52 ...		idle	Activity: Checkpointer...	
2	15463	postgres	postgres	pgAdmin 4 - DB postg...	172.31.33.132	2022-10-16 07:38:52 ...		Activity: BgWriterHib...		
3	15465	postgres	postgres	pgAdmin 4 - DB postg...	172.31.33.132	2022-10-16 07:38:52 ...		Activity: WalWriterMain		
4	15466	postgres	postgres	pgAdmin 4 - DB postg...	172.31.33.132	2022-10-16 07:38:52 ...		Activity: AutoVacuum...		
5	15467	postgres	postgres	pgAdmin 4 - DB postg...	172.31.33.132	2022-10-16 07:38:52 ...		Activity: LogicalAunc...		
6	15479	postgres	postgres	pgAdmin 4 - DB postg...	172.31.33.132	2022-10-16 07:39:02 ...		idle	Activity: ClientRead	
7	15551	postgres	postgres	pgAdmin 4 - DB postg...	172.31.33.132	2022-10-16 07:45:53 ...		Activity: ClientRead		

APP과 WEB의 연결을 위해 **AWS EC2 서버**를 사용했습니다.
ERD Cloud로 데이터베이스를 디자인하고, **PostgreSQL** 데이터베이스를 활용했습니다.

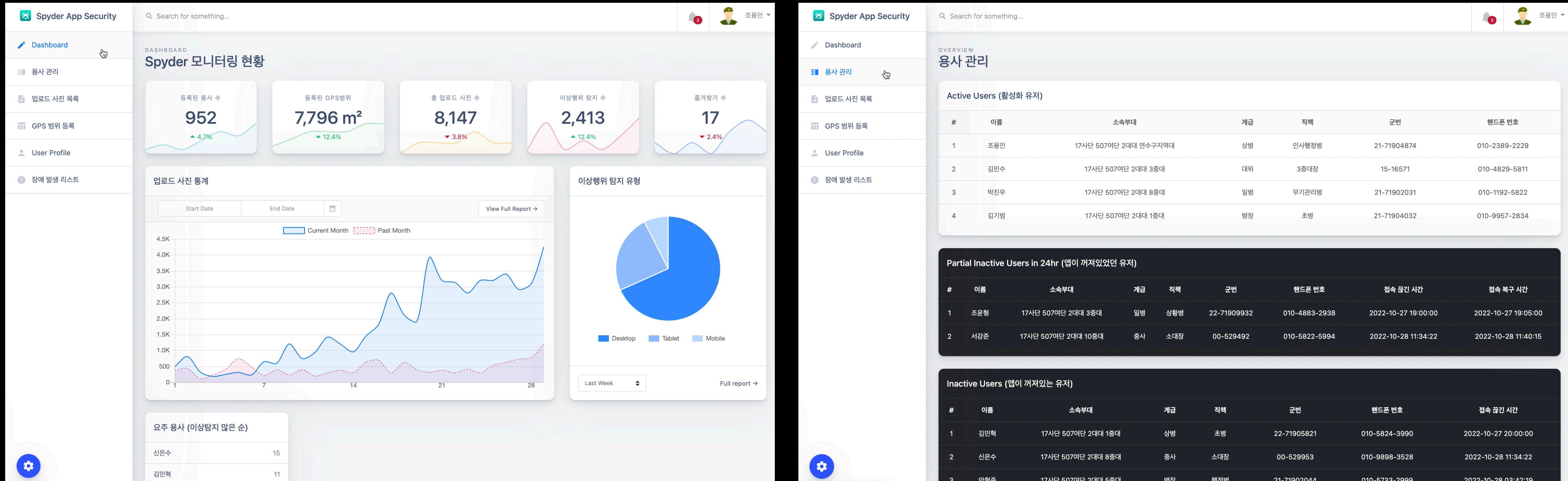
Spyder

DEVELOP
WEB



Spyder

WEB DEVELOP PROCESS



Python과 Django Framework를 활용하여 WEB 백엔드를
구성하였고, HTML/CSS/JS와 Bootstrap을 이용하여
WEB 프론트엔드를 만들었습니다.

Spyder

기대 효과

I



업무 효율성 증가

카메라의 안전한 사용을 통해
장병들의 업무 효율성과 속도가
증가할 수 있습니다.

II



강력한 보안

간부의 실시간 WEB 모니터링을
통해 보안 유출을 방지하고 더욱
강력한 보안성을 자랑합니다.

III



신속한 상황 보고

군 부대 내 병영생활 문제에 대한
신속한 상황보고가 가능합니다.

Spyder

확장성

IV



데이터 수집 ML 강화

지속된 정보 및 데이터 수집으로
스파이더의 사진 분석 머신러닝을
강화할 수 있습니다.

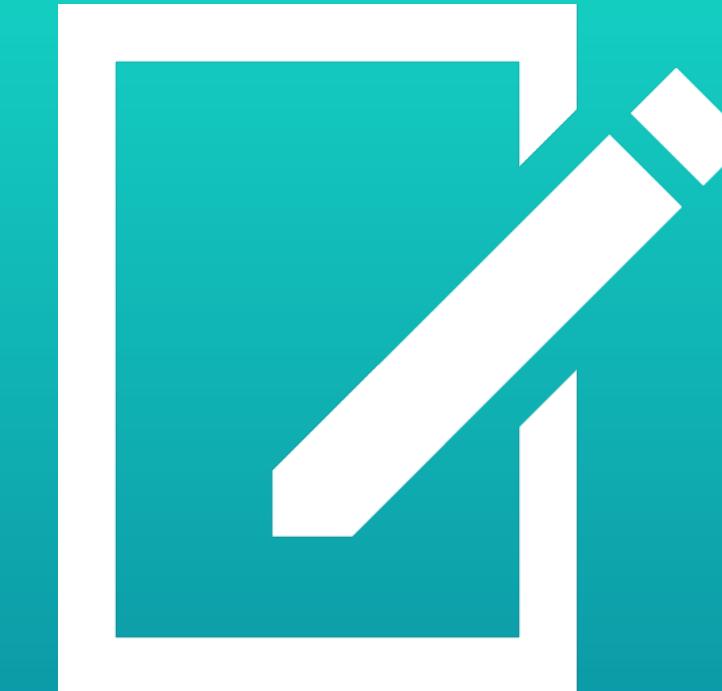
V



AI 기밀유출 탐지

축적된 데이터로 학습 시킨 AI를
이용해 기밀유출 사진을 탐지하여
국방 보안을 지킬 수 있습니다.

VI



보안 사례 통계분석

지속된 정보와 데이터 수집을 활용한
보안 사고 및 사례의 통계 분석으로
보안 사고를 예방할 수 있습니다.

Spyder

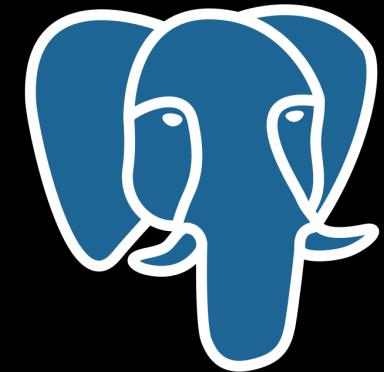
사용 기술 및 오픈소스

Infrastructure



AWS (Amazon Web Service)
| EC2, S3 (Cloud Storage)

Database



PostgreSQL

Language



Java (App)
Python 3.8 (Web BE)
HTML, JavaScript, CSS (Web FE)

Framework



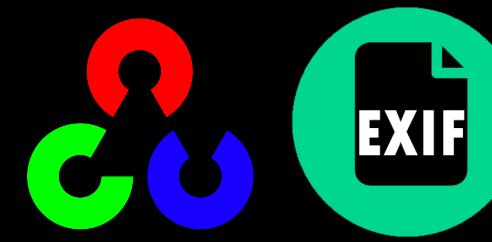
Django (BE)
Bootstrap (FE)

IDE



Android Studio (App)
Visual Studio Code (Web)

Open-Source Libraries



OpenCV (Image processing)
EXIF Analysis (Image analysis)

Spyder



Spyder