

FinalTerm 보고서

제목: SmartLib

도서관 자동 퇴실 감지 시스템

학번:2022105745

이름: 정윤미

Mobile/WebService Project

목차

- 요구조건
- 목적
- 필요성
- 기능 계획 (신규/추가 기능 중심)
- 사용자 시나리오 (UI 구성)
- 기대효과

요구조건

요구조건 - Edge System (Python 기반)

- 1-1. YOLOv5 pretrained model 사용 - yolov5s.pt 모델 사용 (person detection)
- 1-2. MS COCO 80가지 객체 검출 - person 클래스(class 0)만 필터링하여 사용
- 1-3. Change Detection - 좌석 ROI 내 사람 감지 → 상태 변화 감지
- 1-4. HTTP RESTful API 사용 - POST /api_root/Seat/update_from_edge/ 호출
- 1-5. 신규 추가 기능
 - 좌석 ROI 설정: 화면을 4등분하여 각 좌석 영역 지정
 - 실시간 감지: 5초마다 서버에 감지 결과 전송
 - ROI 시각화: 화면에 좌석 박스 및 라벨 표시
 - Person confidence score 표시 (0.76)

요구조건

요구조건 - Service System (Django 기반)

2-1. 사용자 보안 기능

- JWT 토큰 인증 (/api-token-auth/)
- Username: jeong-yunmi

2-2. Image Blog 관리

- 기존 Post 모델 유지 (공통과제)

2-3. 게시용 RESTful API

- /api_root/Post/ 엔드포인트

2-4. 신규 추가 기능

- Seat 모델 추가 - seat_number, status, user_name, reserved_at - last_detected_at, auto_released
- 좌석 관리 API - GET /api_root/Seat/ (전체 좌석 조회) - POST /api_root/Seat/{id}/reserve/ (예약) - POST /api_root/Seat/{id}/release/ (수동 반납)
- Edge 연동 API - POST /api_root/Seat/update_from_edge/
- 자동 퇴실 API - POST /api_root/Seat/auto_release_check/ - 15분 이상 미감지 시 자동 퇴실 처리

요구조건

요구조건 - Client System
(Android Native App)

3-1. Image list view 기능 - 기존
공통과제 유지

3-2. HTTP RESTful API 사용 -
Volley 라이브러리로 GET
/api_root/Seat/ 호출

3-3. 신규 추가 기능

- 좌석 배치도 GridView - 5x6 = 30석 배치 - 색상으로
상태 구분 초록색: 사용 가능 (empty) 빨간색: 사용 중
(occupied) 주황색: 예약됨 (reserved)

- 실시간 갱신 - 5초마다 자동 새로고침

- 통계 카드 - 전체 좌석 수 - 사용 중 좌석 수 (빨강) -
사용 가능 좌석 수 (초록)

- 좌석 상세보기 - 좌석 클릭 시 상태 토스트 메시지

목적

프로젝트 목적 : 우리 학교 도서관 좌석 예약 시스템의 근본적인 문제 해결

- 현재 시스템의 문제점:
 - 퇴실 시 수동으로 퇴실 버튼을 눌러야만 좌석 반납
 - 퇴실 버튼을 누르지 않고 퇴실하는 학생들 다수 존재
 - 실제로는 빈 좌석이지만 시스템상 사용 중으로 표시 - 다른 학생들이 이용 불가능한 문제 빈번 발생
- 본 시스템의 목적:
 - 사람이 일정 시간(15분) 이상 자리를 비우면 자동 퇴실 처리
 - AI 객체 인식(YOLOv5)으로 실제 착석 여부 확인
 - "유령 좌석" 문제 해결
 - 실질적 사용 가능 좌석 정보 제공

필요성

- 현재 도서관 시스템의 문제 상황
 1. 사용자 불편 - 퇴실 버튼을 깜빡하고 나가는 학생 다수, 시스템상 만석이지만 실제로는 빈 좌석 많음, 이용 원하는 학생들의 접근성 제한
 2. 관리자 부담 - 수동으로 좌석 상태 확인 및 초기화 필요, 관리 불편 → 방치 → 이용 불가
 3. 시험기간 악화 - 시험기간에 문제 심각해짐, 대학교 커뮤니티에 불편 호소 게시글 다수, 도서관 공간 활용도 현저히 저하

기능 - 조건대비표

시스템 구분	세부 기능	구현 여부	대체 여부	소스 파일명(함수 또는 class명)
1. Edge System(Python기반, 공통)	1-1. YoloV5 pretrained model 사용(대체 가능)	O	X	seat_detection.py
	1-2. Ms coco 훈련데이터 기준 검출 객체 (Classes) : 80가지 객체 검출 기능(대체 가능 함)	O	X	
	1-3. 한 종류의 객체를 동일한 객체로 가능한 Change Detection(대체가능 함)	O	X	
	1-4. 게시를 위한 HTTP Restfull API 사용(공통)	O	X	
	1-5. 기타 추가기능 (더 있을 경우 아래 표 추가) - 좌석 ROI(Region of Interest) 설정 (4개 좌석) - 실시간 Person Detection (Confidence 0.5 이상) - 5초마다 서버에 감지 결과 전송 - ROI 시각화 (초록/빨강 박스) - Frame count 표시	O	-	
2. Service System(Python, Django 기반, Pythonanywhere 클라우드상 서비스 구동, 일부 확장 기능 가능)	2-1. 사용자 보안 기능(보안키를 이용한 로그인, 공통) - JWT 토큰 인증 구현	O	X	views.py(BlogImages, SeatViewSet)
	2-2. Image Blog 및 관리 기능(공통, 일부 확장 기능 가능)	O	X	models.py (Post, Seat)
	2-3. 게시를 위한 HTTP Restfull API 제공(공통)	O	X	serializers.py (PostSerializer, SeatSerializer)
	2-4. Image 목록, 획득을 위한 HTTP Restfull API 제공(신규 추가 필요)	O	-	urls.py
	2-5. 기타 추가 기능 (더 있을 경우 아래 표 추가) - Seat 모델 추가 (30개 좌석 관리) - 자동 퇴실 로직 (15분 미감지 시) - 실시간 좌석 상태 업데이트 - last_detected_at 시간 추적	O	-	
3. Client System(Android, Native App, 개별 제안)	3.1. Image list view 기능(공통 기능, 개별 제안) - 좌석 배치도 GridView (5x6 = 30석)	O	-	MainActivity.java
	3.2. Image 목록, 획득을 위한 HTTP Restfull API 사용(신규 추가 필요)	O	X	Seat.java, SeatAdapter.java
	3.3. 공통기능 및 추가기능을 활용한 사용자 시나리오 및 UI 제공 (신규 추가 필요)	O	-	activity_main.xml, item_seat.xml
	3-4. 추가 기능	O	-	AndroidManifest.xml

추가 구현사항

- Edge System: YOLO Person Detection + ROI 기반 좌석 모니터링
- Service System: Django REST API + JWT 인증 + 자동 퇴실 로직
- Client System: Android GridView + Volley HTTP + 실시간 갱신
- 전체 시스템: Edge → Server → Client 3-Tier Architecture 구현

기능 - 1. Edge System(Python기반, 공통)

Edge System 실행 화면

- 4분할 화면으로 4개 좌석 동시 모니터링
- YOLO Person Detection (confidence: 0.76)
- Seat 3: EMPTY (초록색 박스)
- Seat 4: OCCUPIED (빨간색 박스)
- Frame: 1745 (실시간 처리 중)

기능 - 2. Service System (Python, Django 기반)

Django REST API 동작 확인

- 좌석 목록 조회

GET `http://127.0.0.1:8000/api_root/Seat/`

→ 30개 좌석 JSON 반환

- 좌석 예약

POST `/api_root/Seat/1/reserve/`

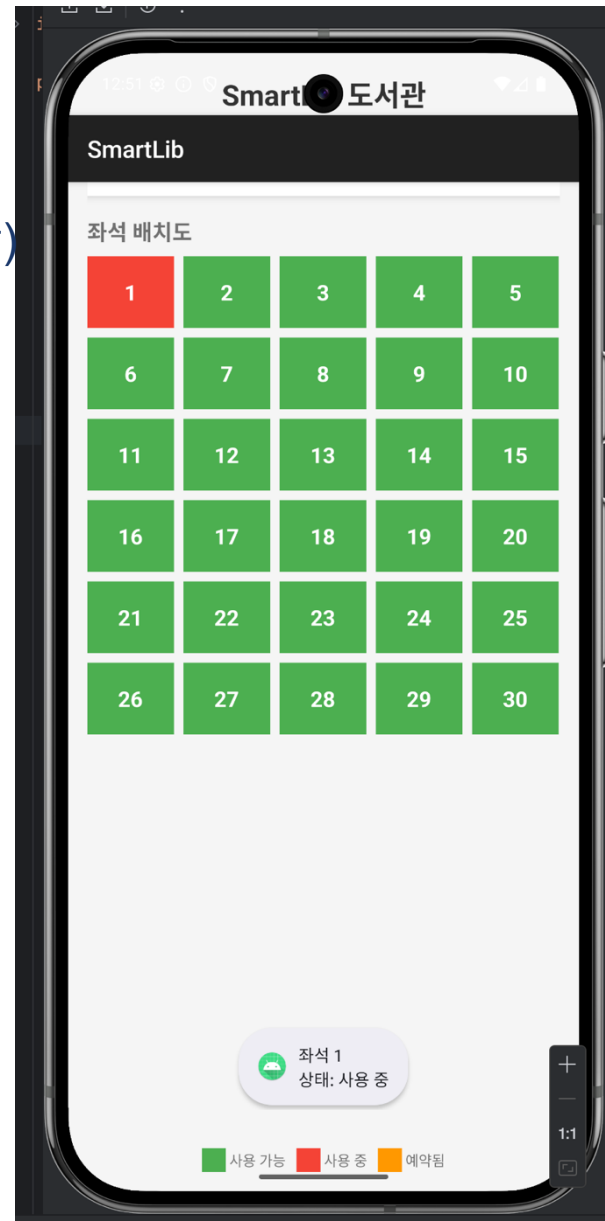
Body: `{"user_name": "정윤미"}`

→ `{"message": "Seat 1 reserved", "seat": {...}}`

- 서버 로그

`[19/Dec/2025 11:12:02] "POST /api_root/Seat/1/reserve/" 200`

기능 - 3. Client System
(Android, Java기반, 개별 제안)



사용자 시나리오 (Ui 구성)

1. 시스템 시작

- 웹캠 실행
- YOLOv5 모델 로드
- JWT 토큰 획득
- 4개 좌석 ROI 설정

2. 실시간 감지

- 5초마다 각 좌석 Person 감지
- 감지 결과를 서버에 전송
- 화면에 ROI 박스 표시
 - 초록색: 빈 좌석
 - 빨간색: 사용 중

3. 서버 연동

- POST /api_root/Seat/update_from_edge/
- last_detected_at 시간 업데이트
- 15분 미감지 시 자동 퇴실

기대효과

기대효과 - 도서관 이용자 측면

✓ 편의성 증대

- 앱으로 실질적으로 빈 좌석 확인 가능
- 헛걸음 방지 (시스템상 만석이지만 실제 빈 좌석 확인)
- 언제 어디서나 좌석 현황 조회

✓ 예약 성공률 향상

- 유령 좌석 제거로 실제 예약 가능 좌석 증가
- 자동 퇴실로 좌석 회전율 상승

✓ 사용자 경험 개선

- 수동 퇴실 버튼 누를 필요 없음 (자동 처리)
- 커뮤니티 불만 감소

기대효과 - 도서관 운영자 측면

✓ 관리 효율성 증대

- 자동화로 관리 인력 절감
- 수동 좌석 초기화 작업 불필요
- 실시간 데이터 기반 운영 가능

✓ 좌석 회전율 향상

- 실제 빈 좌석 즉시 반납 → 공간 활용도 극대화
- 시험기간 도서관 수용 인원 증가

✓ 데이터 수집 및 활용

- 시간대별, 요일별, 학기별 이용 패턴 파악
- 좌석 배치 최적화에 활용 가능
- 자동 퇴실 통계로 정책 개선

✓ 민원 감소

- 시스템 자동화로 관리 문제 해결
- 이용자 만족도 상승

결론

구현 완료 사항

- Edge System: YOLOv5 기반 실시간 좌석 감지
- - Service System: Django REST API 서버
- - Client System: Android 좌석 배치도 앱

핵심 기능 구현

- 4개 좌석 동시 모니터링
- Person 감지 (confidence 0.76)
- 15분 자동 퇴실 로직
- 실시간 좌석 상태 업데이트

실제 문제 해결

- 도서관 유령 좌석 문제 해결
- AI 자동화로 관리 효율성 향상
- 실시간 좌석 정보 제공

향후 개선 방향

- 더 많은 좌석으로 확장 (30석 → 100석+)
- 얼굴 인식 추가 (본인 확인)
- 예약 시스템 고도화
- 통계 대시보드 추가

결과물 목록

- 서비스 URL:
https://jeongyunmi2002.pythonanywhere.com/api_root/
- 소스코드 git 주소: <https://github.com/yunmi-dev/SmartLib-seat-management>