

SeatOn

관리자 가이드

설치·설정·계정·운영·장애 대응·보안

버전 v1.4.1

작성일 2026-09-11

문서 관리자 가이드

SeatOn 관리자 가이드

SeatOn v1.4.1 기준. 화면을 쓰는 사람을 위한 조작법은 [사용자 가이드](#)에 있으며, 이 문서는 그 화면을 띄워 놓고 지키는 사람을 위한 것입니다. API 세부은 [API_AND_MCP.md](#), 내부 구조는 [ARCHITECTURE.md](#)를 봅니다.

1. 구성 요소

구성 요소	무엇	주고받는 것
seaton 컨테이너	Go 단일 바이너리. 내장 React UI, REST/OpenAPI, MCP(Streamable HTTP), 도면 분석 워커, 백그라운드 정리·인사 동기화 스케줄러를 한 프로세스에서 실행	:8080 HTTP. 인터넷 통신 없음(기본)
PostgreSQL 14+	유일한 상태 저장소. 도면 파일·좌석·직원·배정·이력·감사·세션·설정을 모두 보관. 스키마는 기동 시 자동 생성	POSTGRES_DSN
볼륨 seaton-data → /var/lib/seaton	master.key (32바이트). 설정의 비밀값 암호화(AES-256-GCM)와 세션·API 키 해시(HMAC)에 쓰임	컨테이너만 읽음
Keycloak (선택)	OIDC Discovery + Authorization Code/PKCE. 그룹으로 역할 결정	Issuer URL로 나가는 HTTPS, 콜백 /api/v1/auth/oidc/callback
인사 시스템 API (선택)	직원·조직 JSON을 내려주는 사내 엔드포인트	hr.api_url 로 GET + Bearer
사내 비전 모델 서버 (선택)	OpenAI 호출 /chat/completions (vLLM·Ollama 등)	ai.vlm_base_url 로 HTTPS/HTTP
리버스 프록시 (권장)	HTTPS 종료	X-Forwarded-Proto , X-Forwarded-Host 전달

컨테이너는 read_only 루트, cap_drop: ALL , no-new-privileges , UID 10001로 돌며 /tmp 만 tmpfs로 씁니다. 이 미지 크기는 약 186MB이고 PDF 도면 변환용 poppler-utils 가 들어 있습니다.

2. 설치

릴리즈 자산 [SeatOn-v1.4.1.tar.gz](#) (GitHub Release 첨부)와 이 저장소의 `compose.yaml` 하나면 됩니다. PostgreSQL은 외부 또는 사내 것을 준비합니다(빈 데이터베이스와 소유 계정만 있으면 스키마는 SeatOn이 만듭니다).

항목	값
포트	컨테이너 8080 → 호스트 8080 (compose.yaml)
볼륨	seaton-data:/var/lib/seaton — master.key 보관. 반드시 백업
tmpfs	/tmp 256MB — PDF 변환·업로드 임시 파일
자원	도면 분석 중 CPU 1코어를 잠시 씀. 500석 도면 기준 힙 수십 MB. 메모리 512MB면 충분
헬스체크	seaton healthcheck 30초마다(/healthz), 시작 유예 20초

2.1 처음부터 끝까지

```
# 1. 이미지 적재 — seaton:v1.4.1 태그가 생긴다
docker load < Seat0n-v1.4.1.tar.gz

# 2. 필수 환경변수 3개 + Compose 이미지 태그
export POSTGRES_DSN='postgres://seaton:change-db-password@postgres.intra:5432/seaton?sslmode=require'
export BOOTSTRAP_ADMIN='admin'
export BOOTSTRAP_ADMIN_PASSWORD='change-this-strong-password' # 12자 이상
export SEATON_IMAGE_TAG='v1.4.1'

# 3. 기동
docker compose up -d

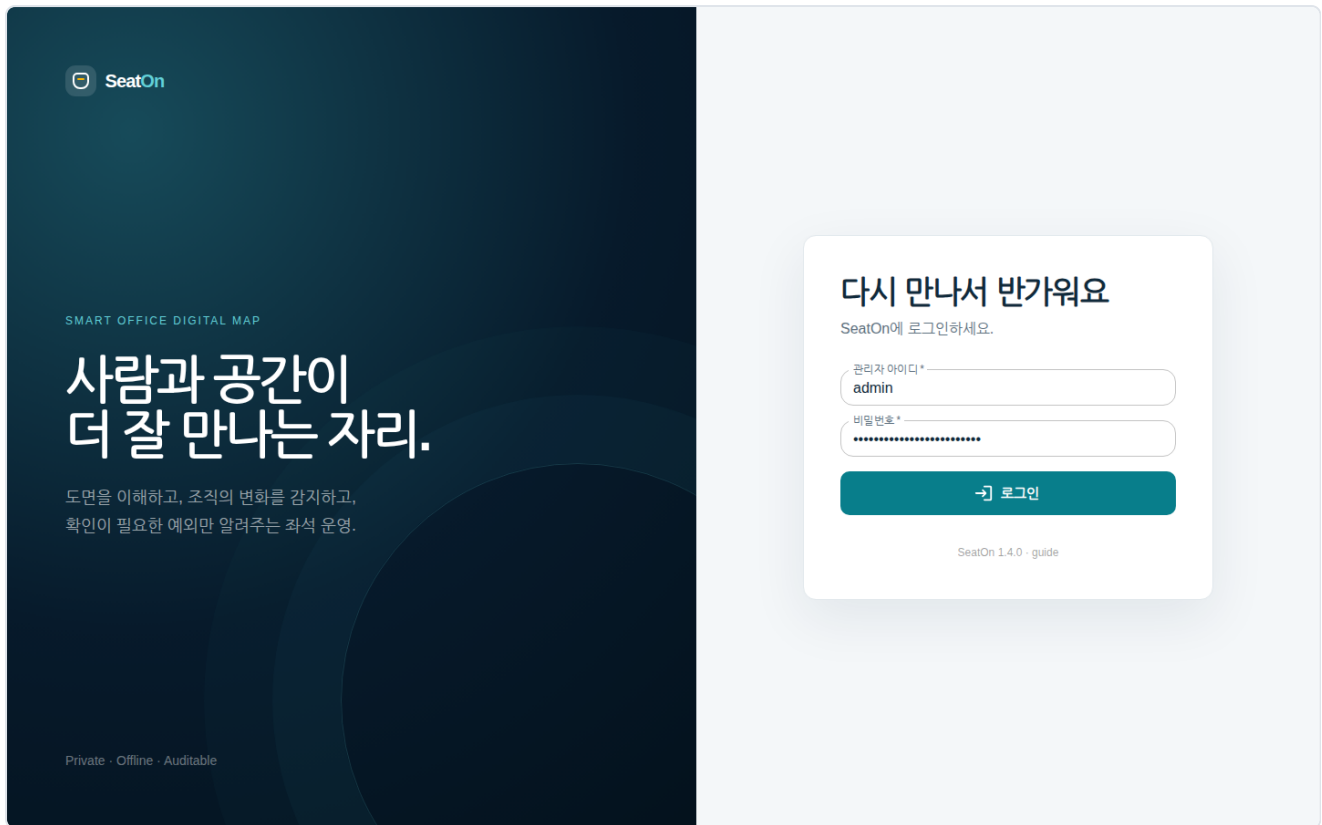
# 4. 준비 확인 — {"status":"ready"} 가 나올 때까지
curl -s http://127.0.0.1:8080/readyz
curl -s http://127.0.0.1:8080/api/v1/version
```

SEATON_IMAGE_TAG 는 Compose가 이미지를 고르는 셸 치환값이며 컨테이너 안으로 전달되지 않습니다.

2.2 최초 관리자 계정

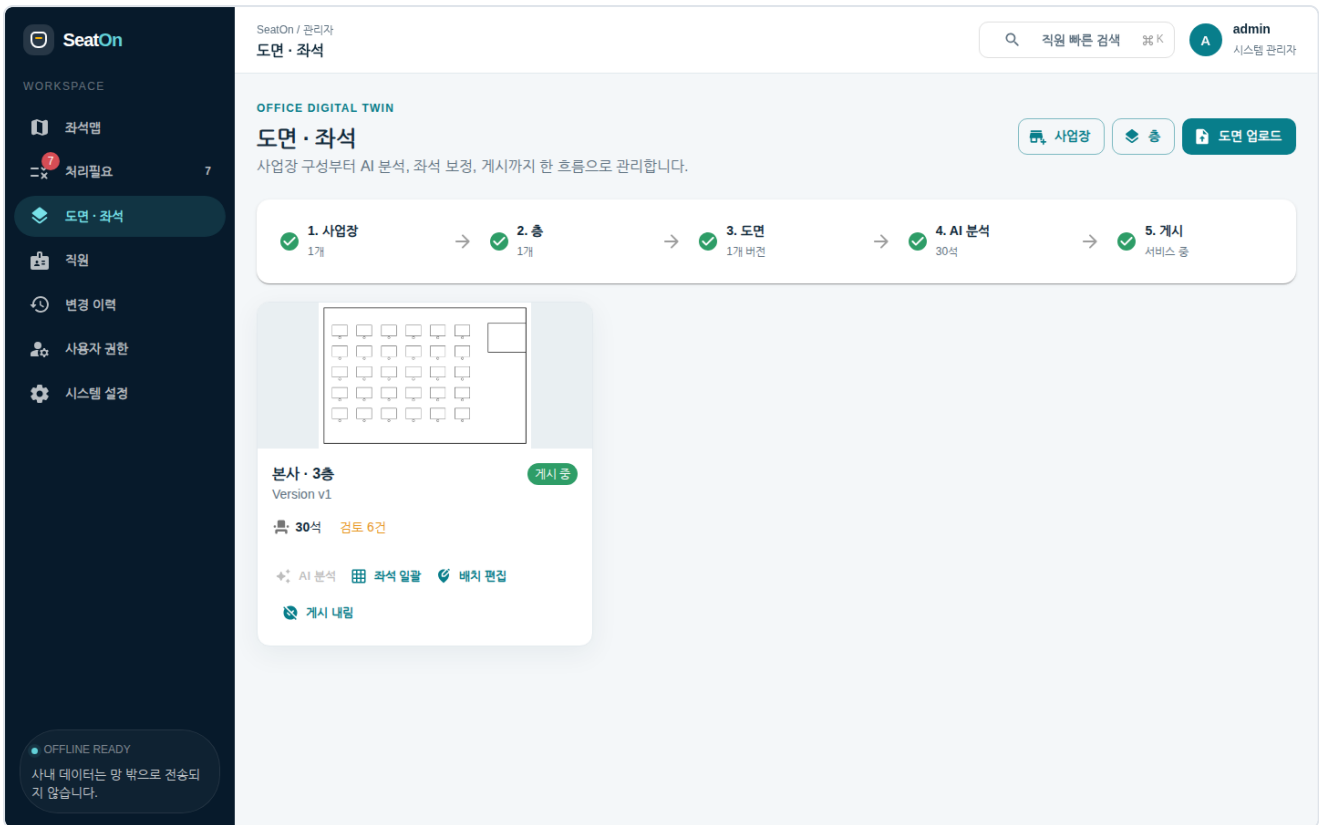
BOOTSTRAP_ADMIN / BOOTSTRAP_ADMIN_PASSWORD 로 **시스템 관리자** 계정이 첫 기동 때 만들어집니다. 같은 이름의 계정이 이미 있으면 환경변수 비밀번호로 덮어쓰지 않으므로, 화면에서 비밀번호를 바꾸거나 SSO로 전환한 뒤에도 안전합니다. 이 계정은 삭제할 수 없는 비상 복구용(break glass) 계정으로 두고, 평소 운영은 SSO 계정으로 합니다.

http://호스트:8080 에 접속해 로그인합니다. 로그인 카드 아래에 빌드 버전이 보이면 UI가 정상적으로 내장된 것입니다.



로그인 — 부트스트랩 관리자 계정으로 첫 로그인. 카드 아래에 SeatOn 1.4.0 버전이 표시된다

로그인 뒤 시스템 설정 → 보안 · 키 탭의 내 로컬 관리자 비밀번호에서 부트스트랩 비밀번호를 바로 바꿉니다. 그다음은 도면 · 좌석에서 사업장·층·도면을 등록하는 순서이며, 이 부분은 사용자 가이드의 "새 층 도면을 올려서 게시하기"를 따릅니다.



도면 · 좌석 — 사업장 → 층 → 도면 → AI 분석 → 게시 다섯 단계. 새 설치에서는 여기서 시작한다

3. 설정

3.1 환경 변수

애플리케이션이 읽는 환경 변수는 세 개뿐입니다(`internal/platform/config.go`). 나머지는 모두 DB의 설정 테이블에 있고 시스템 설정 화면에서 바꿉니다.

이름	기본 값	필 수	설명
<code>POSTGRES_DSN</code>	없음	예	PostgreSQL 접속 문자열. 앞뒤 공백은 제거됨. 기동 시 최대 2분 동안 접속을 재시도
<code>BOOTSTRAP_ADMIN</code>	없음	예	최초 시스템 관리자 아이디. @ 가 들어 있으면 이메일로도 저장
<code>BOOTSTRAP_ADMIN_PASSWORD</code>	없음	예	최초 관리자 비밀번호. 12자 이상이 아니면 기동 실패

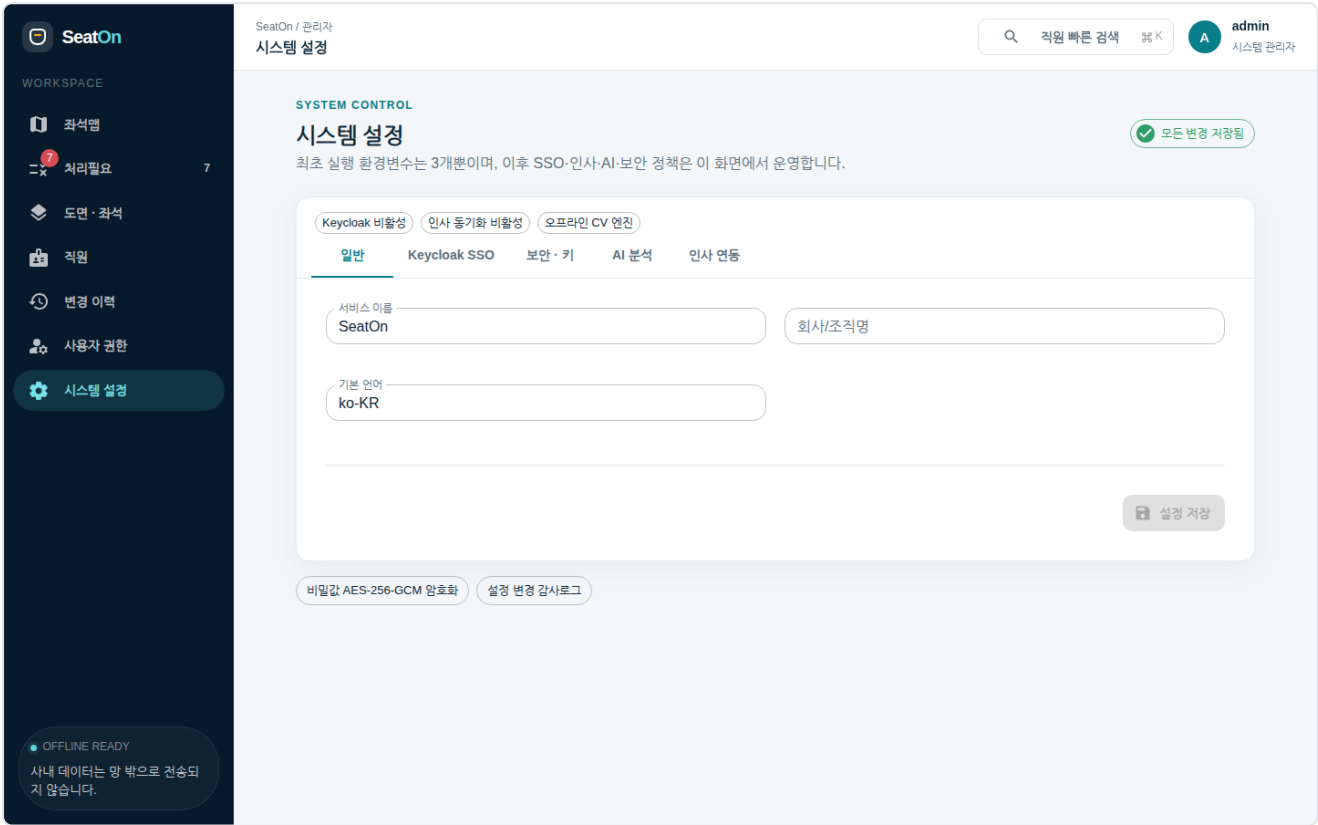
Compose 전용 (컨테이너에 전달되지 않음):

이름	기본값	설명
<code>SEATON_IMAGE_TAG</code>	<code>latest</code>	<code>seaton:<태그></code> 이미지 선택

셋 중 하나라도 비면 기동 로그에 configuration error ... missing required environment variables: ... 가 찍히고 종료합니다.

3.2 시스템 설정 화면

주소 /admin/settings . 시스템 관리자만. 상단 칩이 현재 상태(Keycloak 비활성 , 인사 동기화 비활성 , 오프라인 CV 엔진)를 요약하고, 오른쪽 위에 저장 여부(모든 변경 저장됨 / 저장하지 않은 변경)가 표시됩니다. 비밀번호는 저장 후 ***** 로만 보이고, 빈 값이나 ***** 로 다시 저장하면 기존 값을 유지합니다. 모든 저장은 감사 로그 (settings.update)에 바뀐 키 이름과 함께 남습니다.



시스템 설정 · 일반 — 서비스 이름, 회사/조직명, 기본 언어

설정 키 전수(internal/database/migrations.sql 기본값). 화면 탭과 항목 이름을 함께 적었습니다.

일반

키	화면 항목	기본값	설명
general.service_name	서비스 이름	SeatOn	화면 제목
general.company_name	회사/조직명	빈 값	표시용
general.default_locale	기본 언어	ko-KR	표시용

Keycloak SSO

키	화면 항목	기본값	설명
oidc.enabled	Keycloak SSO 사용	false	켜면 로그인 화면에 사내 SSO로 로그인 단추가 나타남
auth.local_enabled	로컬 관리자 로그인 허용	true	끄면 아이디·비밀번호 로그인이 403 local_login_disabled . SSO가 검증되기 전에는 끄지 말 것
oidc.auto_provision	SSO 사용자 자동 생성	true	첫 SSO 로그인 때 사용자 자동 생성
oidc.issuer_url	Keycloak Issuer URL	빈 값	예 https://keycloak.intra/realms/company . Discovery 문서에서 나머지 엔드포인트를 자동 구성
oidc.client_id	Client ID	빈 값	
oidc.client_secret	Client Secret	빈 값	비밀값(암호화 저장)
oidc.scopes	Scopes	openid profile email groups	
oidc.admin_group	시스템 관리자 그룹	/seaton-admins	이 그룹이면 system_admin
oidc.seat_manager_group	좌석 관리자 그룹	/seaton-seat-managers	이 그룹이면 seat_manager , 둘 다 아니면 employee

보안 · 키

키	화면 항목	기본 값	설명
security.session_hours	세션 유효시간 (시간)	8	넘기면 401 session_expired
security.api_key_days	개인 키 기본 유효기간 (일)	90	키 생성 시 만료일 계산. 요청에서 1~3650일로 지정 가능
security.rotation_grace_hours	키 회전 유예시간 (시간)	24	회전 후 옛 키가 함께 동작하는 시간

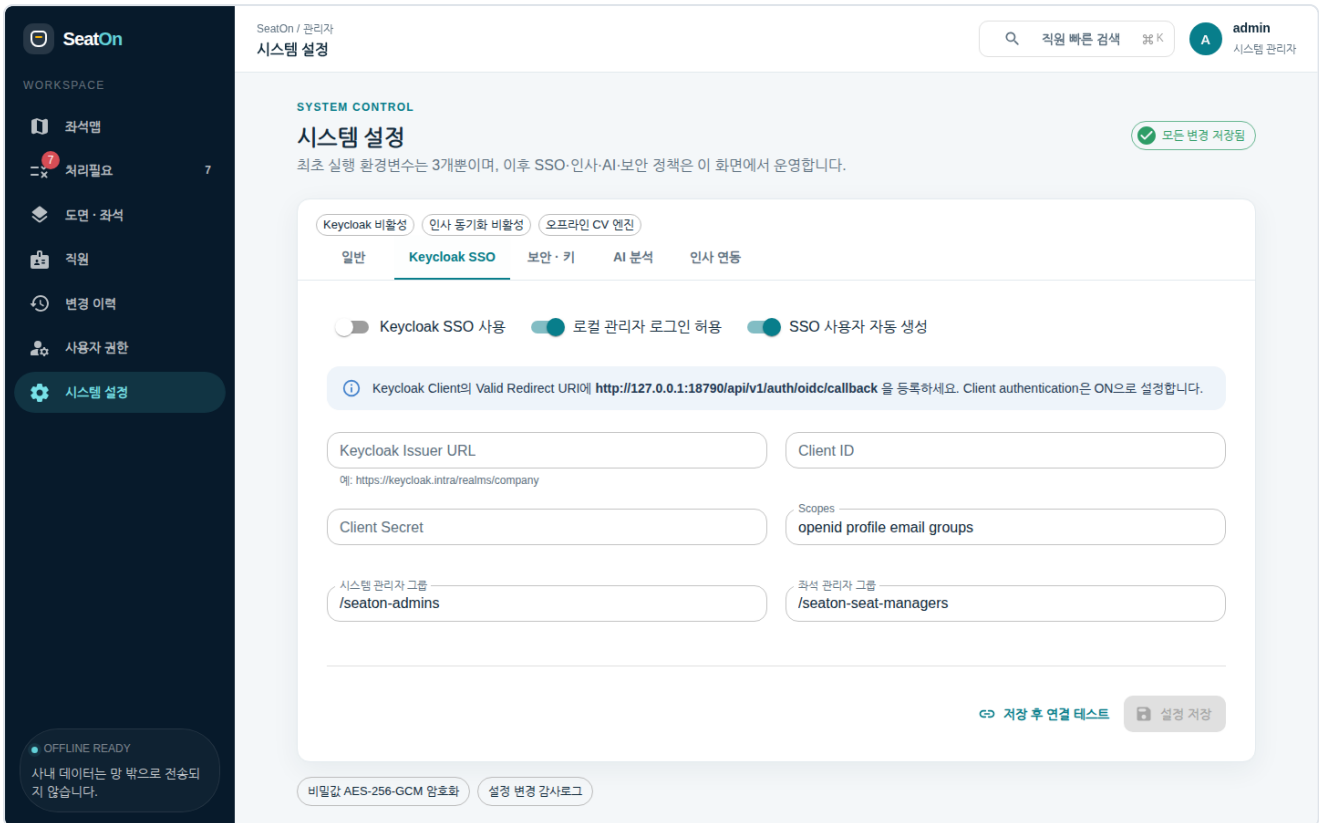
API 분석

키	화면 항목	기본값	설명
ai.engine	좌석 인식 엔진	cv	cv · vlm · hybrid
ai.confidence_threshold	좌석 후보 최소 신뢰도	0.80	이 아래 후보는 버림
ai.auto_approve_threshold	자동 승인 신뢰도	0.95	이 아래면 검토 필요로 표시
ai.vlm_base_url	VLM 엔드포인트 (OpenAI 호환)	빈 값	예 http://vlm.intra:8000/v1. /chat/completions 는 자동으로 붙음
ai.vlm_model	VLM 모델 이름	qwen2.5-vl-7b-instruct	
ai.vlm_api_key	VLM API 키	빈 값	비밀값(암호화 저장)
ai.vlm_timeout_seconds	VLM 응답 제한 (초)	120	10~600
ai.vlm_max_image_side	VLM 전송 이미지 최대 변 (px)	1600	512~4096
ai.vlm_max_seats	VLM 좌석 상한	400	1~500
ai.vlm_tiles	VLM 타일 분할 (한 번)	1	1~4. 1 권장 (§8.2)
ai.vlm_json_mode	VLM JSON 강제 모드	true	response_format: json_object 사용
ai.fusion_iou	하이브리드 일치 판정 IoU	0.35	0.1~0.9

인사 연동

키	화면 항목	기본값	설명
hr.sync_enabled	자동 동기화 사용	false	켜면 스케줄러가 매분 일정을 확인
hr.api_url	인사 시스템 API URL	빈 값	GET 으로 호출. 응답은 {"organizations": [...], "employees": [...]} JSON
hr.api_token	API Bearer Token	빈 값	비밀값(암호화 저장)
hr.schedule	동기화 일정 (Cron)	0 2 * * *	표준 5필드 cron. 잘못된 식은 로그 invalid HR sync schedule

3.3 Keycloak 연결

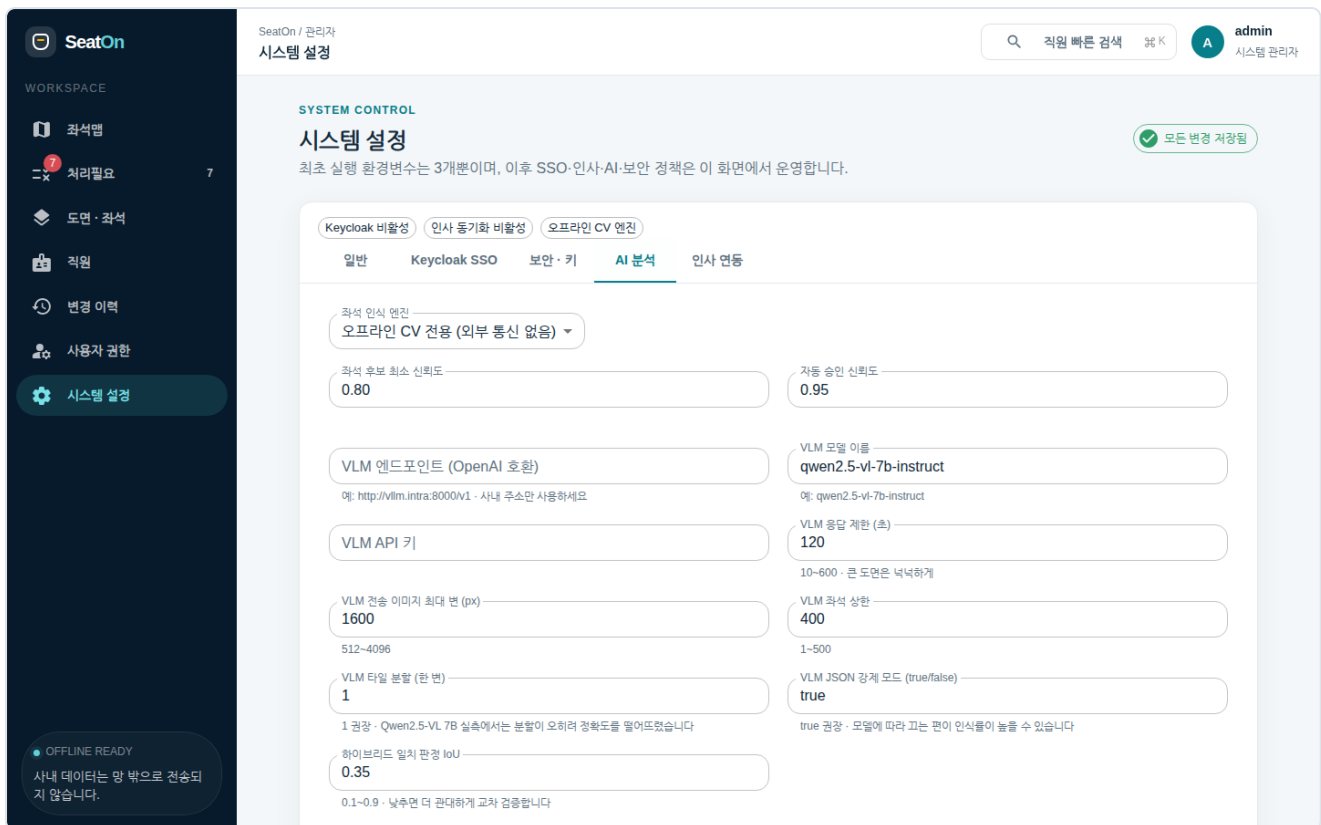


시스템 설정 · Keycloak SSO — 사용 스위치, 콜백 URL 안내, Issuer URL·Client ID·Client Secret·Scopes·그룹 이름, 저장 후 연결 테스트

1. Keycloak에 Client를 만들고 **Client authentication**을 켭니다. Valid Redirect URI에 화면 안내대로 `https://<서비스 주소>/api/v1/auth/oidc/callback` 을 등록합니다. 안내 주소는 리버스 프록시가 넘겨준 `X-Forwarded-Host` · `X-Forwarded-Proto` 로 만들어지므로 프록시 설정을 먼저 맞춥니다.
2. **Group Membership mapper**로 `groups claim`을 ID Token에 넣고, `/seaton-admins` · `/seaton-seat-managers` 그룹을 만듭니다(이름을 바꾸면 설정의 그룹 이름도 맞춥니다).
3. 시스템 설정 → **Keycloak SSO**에 Issuer URL·Client ID·Client Secret을 넣고 **설정 저장**, 이어서 **저장 후 연결 테스트** (`POST /api/v1/settings/oidc/test`)로 Discovery 문서와 콜백 주소를 확인합니다.
4. **Keycloak SSO 사용**을 켜고 저장합니다. 다른 브라우저에서 SSO 로그인인 되는 것을 확인한 뒤에야 **로컬 관리자 로그인 허용**을 끌지 결정합니다.

SSO 사용자는 첫 로그인 때 자동 생성되고 그룹으로 역할이 정해집니다. 이미 `system_admin` 인 사용자는 그룹이 바뀌어도 강등되지 않습니다.

3.4 좌석 인식 엔진과 사내 비전 모델



기본값 `cv` 는 외부 통신이 전혀 없습니다. `vlm · hybrid` 를 쓰려면 사내 추론 서버 주소를 넣고 **설정 저장** 뒤 **저장 후 VLM 연결 시험** (`POST /api/v1/settings/ai/vlm/test`)으로 주소·인증·응답 형식·좌표계를 한 번에 확인합니다. 엔진 별 동작·실효·실패 처리는 §8에 있습니다.

hr.api_url 을 GET 으로 호출하고 hr.api_token 이 있으면 Authorization: Bearer 를 붙입니다. 응답 JSON의 organizations[] (externalId, name, parentExternalId, color)와 employees[] (employeeNo, name, email, organizationExternalId, title, position, workplace, status)를 upsert합니다. 화면의 저장 후 지금 동기화 (POST /api/v1/settings/hr/sync)로 즉시 돌려 볼 수 있고, 결과는 employee_sync_runs 테이블과 감사 로그 (hr.sync)에 남습니다. status 가 retired 인 직원이 좌석을 갖고 있으면 처리필요의 퇴직자 항목이 됩니다.

4. 계정과 권한

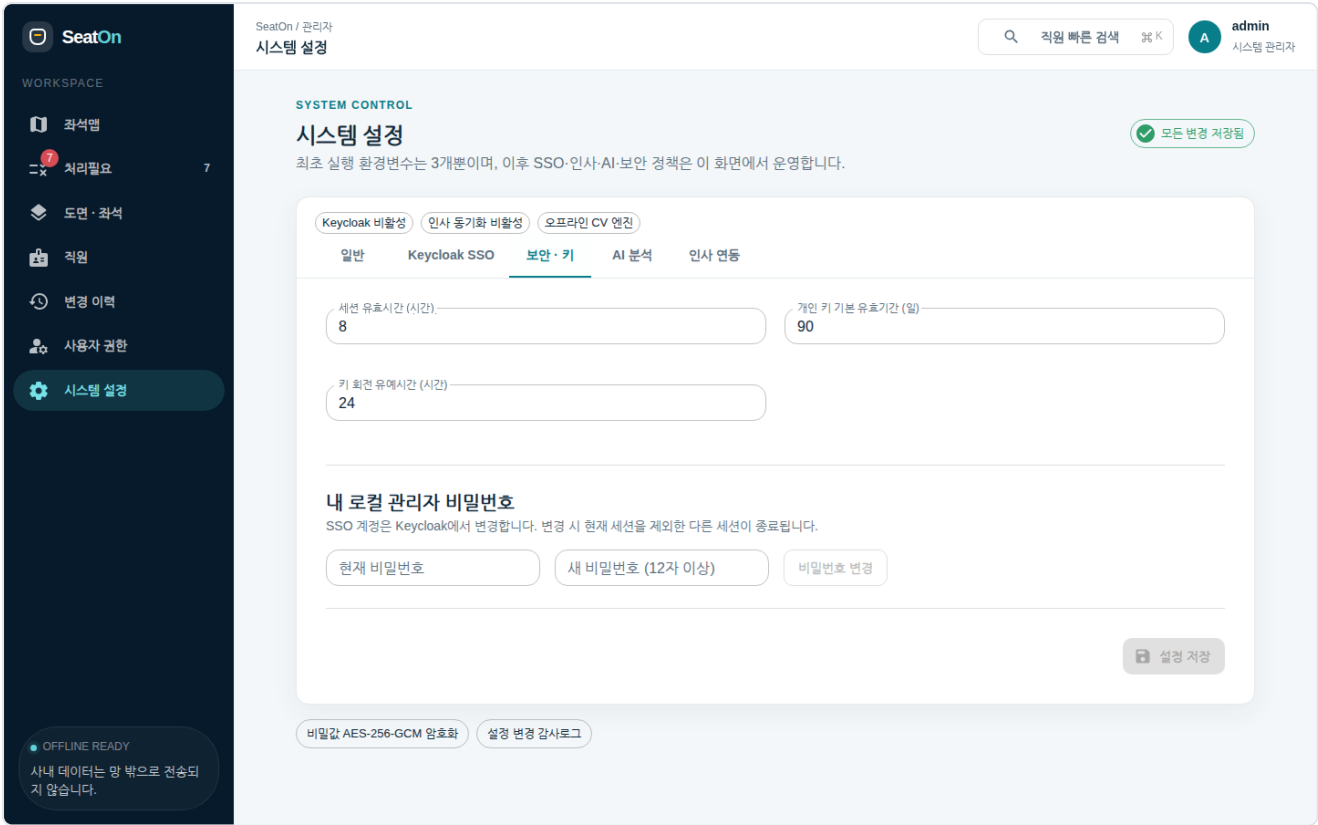
역할	값	할 수 있는 일
직원	employee	좌석맵 조회·검색, 내 API 키
부서 관리자	department_manager	현재 버전에서는 직원과 같음(역할 값만 예약)
좌석 관리자	seat_manager	+ 처리필요, 도면 · 좌석(업로드·분석·편집·게시·삭제), 직원(가져오기·일괄 변경), 변경 이력, 좌석 배정
시스템 관리자	system_admin	+ 사용자 권한, 시스템 설정, 연결 시험, 인사 동기화 실행

권한 검사는 서버에서 합니다(`requireSeatManager` → 403 `seat_manager_required` , `requireAdmin` → 403 `admin_required`). API 키로 부르는 요청은 키 소유자의 역할에 키의 범위(`read` · `write` · `mcp`)가 교집합으로 걸립니다. 예를 들어 `assign_seat` MCP 도구는 소유자가 좌석 관리자 이상이고 키에 `write` · `mcp` 가 모두 있어야 합니다.

사용자 권한 — 사용자, 로그인 방식(Local/SSO), 최근 로그인, 권한 선택. 새 설치에는 부트스트랩 관리자만 있다

사용자 권한(`/admin/users`)에서 역할을 바로 바꿉니다. 화면에는 계정 만들기가 없습니다 — 로컬 계정은 부트스트랩 관리자 하나뿐이고, 나머지 사용자는 SSO 첫 로그인 때 생깁니다. 계정을 막으려면 `PATCH /api/v1/users/{id}` 에 `{"active": false}` 를 보냅니다(비활성 사용자는 로그인·세션·API 키가 모두 거부됩니다).

로컬 관리자 비밀번호는 시스템 설정 → 보안 · 키에서 바꿉니다(12자 이상, POST /api/v1/auth/password , 브라우저 세션에서만 가능). 바꾸면 현재 세션을 제외한 다른 세션이 종료됩니다.



시스템 설정 · 보안 · 키 — 세션 유효시간, 개인 키 기본 유효기간, 키 회전 유예시간과 로컬 관리자 비밀번호 변경

5. 운영

5.1 상태 점검

경로	메서드	인증	응답
/healthz	GET	없음	프로세스 살아 있음 {"status":"ok"} . 컨테이너 헬스체크가 이걸 봄
/readyz	GET	없음	DB Ping 성공 시 {"status":"ready"} , 실패 시 503 database_unavailable
/api/v1/version	GET	없음	{"name":"SeatOn","version":"1.4.1","commit":"...", "builtAt":"..."}
/api/v1/dashboard	GET	좌석 관리자	운영 준비도·연동 상태·처리 필요 건수

처리필요 화면의 운영 준비도와 연동 상태가 같은 정보를 사람이 보기 좋게 보여 줍니다.

The screenshot displays the SeatOn management interface. On the left is a dark sidebar with navigation options: Workspace, 좌석맵 (Seat Map), 처리필요 (Tasks to be processed) - highlighted with a red badge '7', 도면·좌석 (Diagram/Seat), 직원 (Staff), 변경 이력 (Change History), 사용자 권한 (User Permissions), and 시스템 설정 (System Settings). The main area is titled 'SeatOn / 관리자 처리필요' (SeatOn / Admin Tasks to be processed). It features a tabbed interface with '전체' (All), '미배정' (Unassigned), '퇴직자' (Retired), '조직 불일치' (Organization mismatch), and 'AI 확인' (AI Check). The '전체' tab is active, showing a list of tasks. Each task row includes a status icon (yellow for '확인' - Check, blue for '검토' - Review), a title, a description, a timestamp, and an 'AI 신뢰도' (AI Confidence) percentage. To the right of each task is a button labeled '영역 맞춤' (Adjust Area) or '확인 완료' (Check Complete). On the far right, there are two panels. The top panel, '연동 상태' (Integration Status), shows a progress bar and a list of integration items: '사업장 등록' (Registered), '층 구성' (Floor Structure), '좌석맵 게시' (Seat Map Published), '직원 데이터' (Staff Data), 'Keycloak SSO', and '인사 자동 동기화' (Automatic HR Synchronization). The bottom panel, '연동 상태' (Integration Status), shows the status of '인사 동기화' (HR Synchronization), 'Keycloak SSO', and '게시 좌석맵' (Published Seat Map). The '최근 좌석 변경' (Recent Seat Changes) section lists recent changes with details like '남디자 · HQ-3F-010' and '초기 배치 · 2026. 9. 11.'.

처리필요 — 운영 준비도(사업장·층·게시·직원·SSO·인사 동기화)와 연동 상태

5.2 로그

표준 출력에 JSON 한 줄씩(`log/slog`) 찍힙니다. `docker compose logs -f seaton` 으로 봅니다. 요청마다 "msg":"request" 에 메서드·경로·소요 시간· request_id 가 남습니다.

```
{ "time": "2026-09-11T11:41:57Z", "level": "INFO", "msg": "SeatOn started", "address": ":8080", "version": "1.4.1", "c
{ "time": "...", "level": "INFO", "msg": "request", "method": "GET", "path": "/readyz", "duration_ms": 0, "request_id": "...
{ "time": "...", "level": "INFO", "msg": "도면 분석 완료", "jobId": "...", "floorMapId": "...", "engine": "cv", "detected": ...
```

업무 감사 로그는 별도로 DB `audit_logs` 테이블에 사용자·IP·대상과 함께 쌓입니다(`auth.login` , `settings.update` , `floor_map.publish` , `assignment.create` , `api_key.revoke` 등). 좌석 이동 이력은 변경 이력 화면에서 CSV로 내보낼 수 있습니다.

5.3 백업

한 세트로 같이 받습니다.

```
# PostgreSQL
pg_dump "$POSTGRES_DSN" -Fc -f seaton-$(date +%F).dump

# master.key (볼륨)
docker run --rm -v seaton-data:/data -v "$PWD":/backup alpine \
  tar czf /backup/seaton-data-$(date +%F).tgz -C /data .
```

master.key 를 잃으면 DB의 암호화된 Client Secret·VLM 키·인사 토큰을 복호화할 수 없고, 세션과 개인 API 키의 해시도 맞지 않게 되어 모두 새로 발급해야 합니다. 개인 API 키 원문은 어떤 경우에도 복구되지 않습니다.

5.4 복구

```
docker compose down
pg_restore -d "$POSTGRES_DSN" --clean --if-exists seaton-YYYY-MM-DD.dump
docker run --rm -v seaton-data:/data -v "$PWD":/backup alpine \
  sh -c 'rm -rf /data/* && tar xzf /backup/seaton-data-YYYY-MM-DD.tgz -C /data && chown -R 10001:10001 /data'
docker compose up -d && curl -s http://127.0.0.1:8080/readyz
```

5.5 백그라운드 작업

작업	주기	하는 일
정리	30분	만료된 세션·OIDC state 삭제, 재시작으로 끊긴 분석 잡 정리
스케줄러	1분	hr.sync_enabled 가 true 이고 hr.schedule 이 되면 인사 동기화 실행
기동 직후	1회	analyzing 상태에 갇힌 도면 복구(로그 중단된 분석 잡을 정리했습니다)

5.6 업그레이드와 되돌리기

스키마는 기동 때 자동으로 맞춰지므로(migrations.sql , 명등) 별도 마이그레이션 명령이 없습니다. 대신 올리기 전에 §5.3 백업을 먼저 받습니다.

```
pg_dump "$POSTGRES_DSN" -Fc -f before-v1.5.0.dump
docker load < Seaton-v1.5.0.tar.gz
export SEATON_IMAGE_TAG='v1.5.0'
docker compose up -d
curl -s http://127.0.0.1:8080/api/v1/version # "version":"1.5.0"
```

되돌릴 때는 태그를 이전 값으로 바꿔 다시 올립니다. 새 버전이 스키마를 바꾼 뒤라면 이전 바이너리가 그 스키마를 이해한다는 보장이 없으므로, 백업한 덤프를 먼저 복원합니다.

```
export SEATON_IMAGE_TAG='v1.4.1'
docker compose up -d
```

분석이 진행 중일 때 재시작하면 그 잡은 실패로 정리되고 도면은 다시 분석할 수 있는 상태로 돌아옵니다. 릴리즈 자산은 `Seat0n-v<버전>.tar.gz` → `seaton:v<버전>` 이름 규칙을 따르고, 애플리케이션이 알리는 버전 문자열은 `v` 없는 `1.4.1` 입니다.

6. 장애 대응

증상	확인할 곳	조치
컨테이너가 바로 종료	<code>docker compose logs seaton</code> 에 <code>configuration error ... missing required environment variables</code> 또는 <code>BOOTSTRAP_ADMIN_PASSWORD must be at least 12 characters</code>	환경변수 3개를 채우고 비밀번호를 12 자 이상으로
기동 후 2분 뒤 종료, 로그 <code>database startup failed</code>	DSN·방화벽· <code>sslmode</code>	DB 접속을 고치고 재기동. 재시도는 2 분 동안만
로그 <code>keyring startup failed</code>	볼륨 권한, <code>master.key must be 32 bytes</code>	볼륨이 UID 10001 소유인지, 파일이 손상되지 않았는지 확인. 백업에서 복 원
<code>/readyz</code> 가 503 <code>database_unavailable</code>	DB 상태	DB 복구. <code>/healthz</code> 는 이때도 200이 므로 준비 확인은 <code>/readyz</code> 로
로그인 화면에 서버에 연결하지 못했습니다	프록시·포트	브라우저 → 프록시 → <code>:8080</code> 경로 확 인
SSO 로그인 실패 SSO 요청이 만료되었거나 유효하지 않습 니다	프록시가 <code>X-Forwarded-Host</code> · <code>X- Forwarded-Proto</code> 를 넘기는지, Keycloak Redirect URI	콜백 주소를 화면 안내와 같게. state는 만료되면 정리됨
Keycloak 연결을 확인하세요 (502 <code>oidc_discovery_failed</code>)	컨테이너에서 Issuer URL로 나가는 HTTPS, 사내 CA	Issuer URL 확인. Discovery 문서 <code>/.well-known/openid- configuration</code> 이 열리는지
Keycloak ID 토큰 검증에 실 패했습니다 / Keycloak 토큰 요청값이 일치하지 않습니다	시계 동기, Client Secret, nonce	서버 시각 동기화, Client Secret 재입 력
로컬 로그인이 로컬 로그인이 비활성화되어 있습니다 인데 SSO도 안 됨	<code>auth.local_enabled</code>	DB에서 되돌립니다: <code>UPDATE settings SET value='true' WHERE key='auth.local_enabled';</code>
세션이 자주 끊김	프록시가 <code>X-Forwarded-Proto: https</code> 를 안 넘김 → 쿠키 Secure 판단 오류, 또는 <code>security.session_hours</code>	프록시 헤더 전달, 세션 시간 조정
요청이 403 <code>csrf_failed</code>	브라우저 세션에서 <code>X-CSRF-Token</code> 없이 변 경 요청	스크립트라면 API 키(Bearer)를 쓰거 나 <code>/api/v1/auth/me</code> 의 <code>csrfToken</code> 을 헤더로

증상	확인할 곳	조치
도면이 analyzing 에 갇혀 재분석 이 409 analysis_running	재시작 여부	서비스 재시작 시 자동 정리(로그 중 단된 분석 잡을 정리했습니다). 30분마다도 정리
분석 실패, 카드에 오류	로그 분석 잡에서 패닉 , VLM 타일 분석 실패	VLM 서버 상태·제한 시간. VLM 실패 는 CV로 대체되어 완료됨 (§8.5)
PDF 도면에 좌석이 안 겹침, PDF 미리보기 없음	로그 도면 미리보기 생성 실패 / 도면 미리보기 저장 실패 , /tmp 공간	tmpfs 256MB 초과 여부, PDF 첫 페 이지 크기. PNG로 변환해 재업로드
인사 동기화가 안 됨	로그 invalid HR sync schedule , scheduled HR sync failed	cron 식(5필드) 수정, hr.api_url 응답 형식 확인
처리필요 화면이 느림, 로그 집계 쿼리 실패	DB 부하	상단 배지는 /dashboard/action- count 만 쓰므로 화면 이동은 영향 없 음. DB 점검
로그 audit log failed	DB 쓰기	감사 기록 실패는 요청을 막지 않음. DB 디스크·권한 확인
로그 panic + 스택	애플리케이션 결함	request_id 와 함께 이슈로 보고, 프 로세스는 계속 동작

7. 보안

- **바꿔야 하는 기본값:** BOOTSTRAP_ADMIN_PASSWORD 는 설치 직후 화면에서 변경. POSTGRES_DSN 은 sslmode=require 이상. 비밀번호 예시(change-this-strong-password)를 그대로 두지 않습니다.
- **밖에 열면 안 되는 것:** :8080 은 리버스 프록시 뒤에만 둡니다. PostgreSQL 포트는 SeatOn 컨테이너에서만 달게 합니다. /mcp 와 /api/v1/* 은 같은 포트이므로 프록시에서 따로 막을 필요는 없지만, 사내망 밖으로 열지 않습니다.
- **프록시에서 HTTPS 종료:** X-Forwarded-Proto: https 와 X-Forwarded-Host 를 반드시 전달합니다. 이 값으로 세션 쿠키의 Secure 플래그와 OIDC 콜백 주소가 결정됩니다.
- **컨테이너 강화:** compose.yaml 의 read_only , cap_drop: ALL , no-new-privileges , 비루트 UID 10001을 그대로 둡니다. 응답에는 CSP default-src 'self' , X-Frame-Options: DENY , nosniff 가 항상 붙습니다.
- **비밀값 저장:** 설정의 비밀값은 master.key 로 AES-256-GCM 암호화, 세션 토큰과 API 키는 HMAC 해시만 저장. master.key 는 백업 대상이자 유출 금지 대상입니다.
- **인증 연동:** Keycloak Authorization Code + PKCE(S256) + nonce 검증. SSO를 검증한 뒤 auth.local_enabled 를 끄면 부트스트랩 계정도 화면에서는 못 들어오므로, 비상시 되돌리는 SQL (§6)을 운영 문서에 적어 둡니다.
- **API 키 정책:** 기본 유효기간 90일, 회전 유예 24시간. 유출이 의심되면 소유자가 폐기하거나, 관리자가 그 사용자를 비활성화합니다(비활성 사용자의 키는 즉시 거부).
- **인터넷 통신:** 기본 설정(ai.engine=cv , SSO·인사 연동 꺼짐)에서는 컨테이너가 밖으로 나가는 연결이 없습니다. 연결이 생기는 곳은 Keycloak Issuer, hr.api_url , ai.vlm_base_url 세 군데뿐이며 모두 사내 주소를 씁니다.

8. 좌석 인식 엔진 (CV · VLM · 하이브리드)

시스템 설정 → AI 분석 에서 도면 판독 엔진을 선택합니다. 세 방식 모두 결과를 동일한 비율 좌표 좌석으로 저장하므로 이후 편집·배정 흐름은 같습니다.

엔진	동작	외부 통신	신뢰도 상한
cv (기본)	오프라인 CV 파이프라인(offline-cv-v2)	없음	격자 정합 시 0.99
vlm	사내 비전 모델 단독 판독	설정된 사내 엔드포인트	0.94 (자동 승인 불가)
hybrid	CV와 VLM을 IoU로 교차 검증	설정된 사내 엔드포인트	합의 시 0.99

기본값은 cv 입니다. hybrid 가 항상 더 낮지는 않으므로 8.2의 실측 비교를 먼저 확인하십시오.

8.1 VLM 엔드포인트 설정

OpenAI 호환 /chat/completions 인터페이스를 제공하는 사내 추론 서버를 지정합니다. vLLM, SGLang, Ollama 등으로 Qwen2.5-VL 계열을 서빙한 주소를 쓰며, 인터넷 구간으로 나가는 주소를 넣지 마십시오. 설정을 저장한 뒤 저장 후 VLM 연결 시험 버튼을 누르면 합성 도면 한 장을 왕복시켜 주소·인증·응답 형식·좌표계를 한 번에 검증합니다.

설정	설명
ai.vlm_base_url	예: http://vllm.intra:8000/v1 (/chat/completions 는 자동으로 덧붙입니다)
ai.vlm_model	서버에 로드된 모델 이름
ai.vlm_api_key	Bearer 토큰. AES-256-GCM으로 암호화 저장
ai.vlm_timeout_seconds	요청당 제한 시간(10~600)
ai.vlm_max_image_side	전송 이미지 최대 변(512~4096)
ai.vlm_max_seats	한 번에 받아들일 좌석 수 상한(1~500)
ai.vlm_tiles	한 변의 타일 수(1~4). 기본 1 권장 — 아래 실측 참고
ai.vlm_json_mode	response_format: json_object 사용 여부. 기본 true
ai.fusion_iou	하이브리드 일치 판정 기준(0.1~0.9)

8.2 실측 결과 (Qwen2.5-VL 7B, Ollama, RTX 5090)

합성 사무실 도면(책상 30개, 1600×1100, 외벽·복도벽·회의실·의자·치수선 포함)으로 엔진별 정확도를 측정한 값입니다. 정답과 IoU 0.5 이상 겹치면 일치로 계산했습니다.

엔진	검출	F1	평균 IoU	검토 필요	소요
cv	30	1.000	0.915	6건	2초
vlm	30	0.900	0.827	30건 (전부)	8초
hybrid	30	1.000	0.915	0건	9초

가로 도면에서 하이브리드가 유리한 이유는 정확도가 아니라 검토 부담입니다. CV 단독은 6건이 자동 승인선 아래로 남지만, VLM이 같은 자리를 지목해 교차 검증되면 신뢰도가 0.95 이상으로 올라가 관리자가 확인할 항목이 사라집니다.

도면별 결과는 다음과 같습니다. PDF는 A3 150 DPI 래스터(2481×1754) 기준입니다.

도면	cv	hybrid	비고
가로 1600×1100, 책상 30	F1 1.000 · 검토 6건	F1 1.000 · 검토 0건	30건 전부 교차 검증
PDF A3 가로, 책상 30	F1 1.000 · 검토 0건	F1 1.000 · 검토 0건	30건 전부 교차 검증
세로 1000×1400, 책상 28	F1 1.000 · 검토 0건	F1 0.933 · 검토 28건	VLM 합의 4건뿐, 오검출 유입

세로 도면에서는 하이브리드가 CV보다 나뉘었습니다. VLM이 책상 28개 중 4개만 CV와 합의하고 14개를 엉뚱한 곳에 보고해, 그 오검출이 좌석으로 추가되면서 정밀도가 1.000에서 0.667로 떨어졌습니다. 그래서 CV가 책상 격자를 복원한 도면에서는 격자를 벗어난 VLM 단독 상자를 제외하도록 했고(10건 제외), 정밀도가 0.875, F1 0.933으로 회복됐습니다. 남은 오검출 4건은 우연히 격자 근처에 놓인 것으로, 모두 검토 대상으로 표시되며 분석 완료 후 경고에 "VLM만 찾은 좌석이 교차 검증된 건수보다 많습니다"가 함께 뜹니다.

정리하면 도면 종류에 따라 하이브리드가 CV보다 나뉠 수 있습니다. 도입 시 대표 도면 몇 장으로 cv 와 hybrid 를 모두 돌려 비교하고, VLM 합의 건수 (details.agreed)가 CV 검출 수에 근접하는지 확인하십시오. 합의 비율이 낮으면 cv 를 쓰는 편이 낫습니다.

프롬프트 설계가 정확도를 좌우합니다. 같은 모델·같은 도면에서 프롬프트 형식만 바꿔 5회씩 측정한 결과입니다.

요청 형식	F1
JSON 스키마에 예시 좌표를 넣어 요청	0.00 — 모델이 예시 숫자를 그대로 복사
예시 없이 정규화 좌표 스키마만 서술	0.00 — 균일 격자를 발명
네이티브 grounding · 배열 반환	0.30
네이티브 grounding · 객체 래퍼 (현재 구현)	0.37~0.41
위에 "최대 N개" 상한 문구 추가	0.22

SeatOn은 마지막에서 두 번째 형식을 사용하며, 여기에 시스템 메시지를 더한 실제 구현이 위 표의 0.900입니다. 좌석 수 상한은 프롬프트가 아니라 서버에서 잘라냅니다.

타일 분할은 이 모델에서 역효과였습니다. 1×1 F1 0.30 $\rightarrow$ 2×2 0.04 $\rightarrow$ 3×3 0.00. 큰 도면에서 작은 객체를 키워 보려던 의도였지만, Qwen 계열의 grounding은 장면 전체를 볼 때 더 정확했습니다. 기본값 1을 유지하고, 다른 모델로 교체할 때만 실제 도면으로 재측정해 조정하십시오.

좌표계는 같은 모델도 입력에 따라 바뀝니다. 실측에서 Qwen2.5-VL은 1600×1100 도면에는 절대 픽셀 좌표로, 640×480 시험 도면에는 0~1000 좌표로 답했습니다. SeatOn은 후보 좌표계를 모두 적용해 자동 판별하며, 판별 결과를 분석 잡의 `details.vlmCoordinates` 에 기록합니다. 연결 시험 결과에도 표시되므로 모델 교체 시 먼저 확인하십시오.

8.3 신뢰도 보정 원칙

비전 모델이 스스로 보고하는 확신도는 보정된 확률이 아닙니다. 그래서 VLM이 단독으로 찾은 좌석은 신뢰도를 0.88~0.94로 제한해 자동 승인선(기본 0.95)을 절대 넘지 못하게 하고 항상 관리자 검토 큐에 남깁니다. 하이브리드에서 CV와 VLM이 같은 자리를 지목한 좌석만 0.90 이상으로 올라가 자동 승인 대상이 됩니다. CV가 단독으로 찾은 좌석은 소폭 감점합니다. 좌석의 `metadata.source` 값(`cv`, `vlm`, `cv+vlm`, `grid-fill`)으로 근거를 추적할 수 있습니다.

8.4 분석은 비동기 잡입니다

VLM 판독은 도면 크기와 타일 수에 따라 수십 초에서 수 분이 걸립니다. `POST /floor-maps/{id}/analyze` 는 즉시 202와 `jobId` 를 돌려주고 실제 작업은 백그라운드에서 진행됩니다. 관리자 화면은 완료까지 진행 상태를 표시합니다. 같은 도면을 동시에 두 번 분석할 수 없으며(409 `analysis_running`), 서비스가 재시작되어 중단된 잡은 다음 기동 시 실패로 정리되고 도면 상태도 복구됩니다.

8.5 예상되는 실패와 동작

상황	SeatOn 동작
VLM 서버 연결 불가 · 타임아웃 · 5xx · 429	지수 백오프로 최대 3회 재시도
인증 실패(401/403)	재시도 없이 즉시 중단, 키 확인 안내
경로 오류(404)	/v1 포함 여부 확인 안내
response_format 미지원 서버	해당 옵션을 빼고 자동 재시도
모델이 JSON 대신 설명문을 반환	코드펜스·설명 제거 후 재파싱, 실패 시 1회 교정 재요청
좌표를 픽셀이나 0~1000으로 반환	후보 좌표계를 모두 적용해 가장 정합한 해석을 자동 선택
[x1,y1,x2,y2] 와 [x,y,w,h] 혼용	필드 이름과 정합도로 형식 판별, 역순 좌표는 교정
같은 좌석 중복·반복 출력	IoU 기반 중복 제거
회의실·구역 등 과대 상자, 글자 등 과소 상자	크기·중형비 검증으로 제거
응답이 최대 길이에서 잘림	건진 좌석은 사용하고 경고 표시, 좌석 상한/타일 수 조정 안내
타일 일부만 실패	성공한 영역 결과를 사용하고 누락 가능성을 경고
VLM이 책상 격자를 벗어난 곳에 보고	CV가 격자를 세운 도면이면 해당 상자 제외 후 경고
VLM 단독 좌석이 교차 검증 좌석보다 많음	오검출 혼입 가능성 경고
VLM이 좌석을 하나도 못 찾음 · 그 밖의 실패	CV 결과로 자동 대체 하고 경고 표시 (분석 자체는 완료)

모든 경고는 분석 완료 후 도면 화면에 목록으로 표시되며 `analysis_jobs.warnings` 에 함께 기록됩니다.