📄 통합형 화면 설계서

시스템명: QGIS 기반 지도 자동 갱신 시스템  
작성자: 이공

# 🔹 1. 시스템 흐름 요약

[웹사이트] (Flask/HTML)  
 └─ 사용자가 접속 → "지도 갱신하기" 클릭  
 ↓  
[QGIS 플러그인] (PyQGIS)  
 └─ MapUpdater 플러그인 자동 실행  
 ↓  
 내부 UI 버튼 클릭으로 정사영상 + 수치지도 로딩, YOLO/SAM 실행, 갱신

# 🔹 2. 주요 화면 설계 목록

|  |  |  |  |
| --- | --- | --- | --- |
| 화면 ID | 구분 | 화면명 | 설명 |
| WEB-001 | 웹페이지 | 메인 포털 페이지 | 사용자 초기 접근 페이지 |
| WEB-002 | 웹페이지 | 지도 갱신 실행 버튼 | QGIS와 연동되어 플러그인을 실행 |
| QGIS-001 | QGIS 플러그인 | 지도 자동 갱신 플러그인 UI | 6개 버튼으로 단계별 실행 가능 |

# 🔹 3. 화면 상세 설계

## 📌 WEB-001: 메인 포털 페이지

- 화면 위치: / (루트 경로)

- 주요 구성:  
 • 상단 로고 및 네비게이션: (주)이공, 회사소개/프로젝트 소개/로그인 등  
 • 중앙 텍스트: "QGIS 기반 자동 지도 갱신 시스템"  
 • 기능 설명 배너 (YOLOv8, SAM, 좌표기반 비교 등)  
 • 중심 버튼: [지도 갱신하기] (btnStartUpdate)  
 • 버튼 클릭 시: Flask → Python subprocess 또는 API 통해 QGIS 플러그인 트리거

■ 웹 페이지 텍스트/스타일 요약

|  |  |  |
| --- | --- | --- |
| 요소 | 설명 | CSS/속성 |
| 제목(h1) | QGIS 기반 자동 지도 갱신 시스템 | fw-bold, font-size: 2.5rem |
| 부제목(h2) | Smart GIS Platform for the Future | fw-light, font-size: 1.5rem |
| 본문(p.lead) | 딥러닝 기반 건물 탐지 설명 | font-size: 1.25rem |
| 버튼(btn-lg) | 🗺 지도 자동 갱신 | min-width: 200px, btn-success, padding: 1rem |
| 기술 설명 div | YOLOv8, SAM 등 요약 | small, 반투명 배경, padding: 0.25rem |

## 📌 WEB-002: 지도 갱신 실행 버튼

- 버튼명: btnStartUpdate  
- 텍스트: 🗺 지도 갱신하기  
- 기능:  
 • QGIS 자동 실행 스크립트 또는 QGIS가 실행 중일 경우 플러그인 호출  
- 시나리오 흐름:  
 1. 사용자가 클릭  
 2. 서버단에서 QGIS 플러그인 MapUpdater 실행  
 3. UI 창이 실행되어 지도 갱신 시작

## 📌 QGIS-001: 플러그인 UI (MapUpdater)

|  |  |  |
| --- | --- | --- |
| 구성요소 | 위젯 ID | 기능 설명 |
| 정사영상 불러오기 | btnLoadOrthophoto | 래스터(.tif 등) 이미지 레이어 로딩 |
| 수치지도 불러오기 | btnLoadCadastral | 벡터(.shp 등) 수치지도 로딩 |
| 건물 탐지 결과 연동 | btnLoadYOLO | YOLOv8 모델 실행 또는 결과 로드 |
| 신규 건물 분류 | btnClassifyNewBuildings | 기존 지도와 비교해 신규/철거 건물 구분 |
| SAM 결과 연동 및 후처리 | btnLoadSAM | SAM 결과 병합 및 좌표 보정 |
| 수치지도 자동 갱신 | btnUpdateMap | 갱신된 수치지도를 저장 및 적용 |
| 로그창 | txtLog | 처리 과정 로그 출력 |

■ 플러그인 UI 텍스트/스타일 요약

|  |  |
| --- | --- |
| 요소 | 설명 |
| QDialog | 단일 창(QVBoxLayout) 구성 |
| 버튼 텍스트 | 기본 font-size 12~14pt, 한글 텍스트 |
| 로그창(txtLog) | QTextEdit, 회색 배경, monospace 폰트 |

# 🔹 4. 사용자 흐름 시나리오

1. 사용자가 웹페이지 접속 → "지도 갱신하기" 클릭  
2. QGIS 실행 또는 이미 실행된 상태에서 플러그인 MapUpdater 로드  
3. 순차적으로 버튼 클릭하며 정사영상 → 수치지도 → 탐지 실행  
4. 신규 건물 분류 및 수치지도 자동 갱신 실행  
5. 결과 저장 및 로그 확인

# 🔹 5. 시각적 연계 예시

웹 → QGIS 흐름

# 🔹 6. 비고 및 확장 가능 요소

• 추후 로그인 인증 기반 플러그인 실행  
• 실시간 처리 상태 모니터링  
• 갱신 결과를 다시 웹으로 전송 및 시각화 등 기능 확장 가능  
• 웹 → QGIS 연동은 로컬 서버(localhost) 또는 QGIS Python API 바인딩을 통해 구현