LOCAL AI for E-GOV FRAMEWORK

Palyx

문서에서 코드로,
코드에서 검증으로.

폐쇄망 내 전자정부프레임워크 사업을 위한 올인원 솔루션

요구사항부터 제출까지, 한 흐름에



폐쇄망에서 안전하게 사용합니다.



폐쇄망내에 구축된 로컬 모델과
임베딩 모델을 사용하여
소스·문서등 데이터의 외부 유출을
원천적으로 방지합니다.

* 실제 Palyx 화면을 캡처하여 만들었습니다.

성공적인 개발은 끊김 없는 환경이 중요합니다.

코드 작성, 형상 관리, 서버 환경, 보안 점검,
산출물 관리는 각각의 업무가 아닙니다.

**사업에 필요한 도구와 기록을 한 IDE에서
수행하고 확인하고 관리합니다.**



개발과 품질, 문서 담당자가 함께 사용합니다.



개발자

화면 이름으로 코드를 찾고
수정한 뒤 서버에서 확인합니다.

품질·보안 담당자

개발보안·웹접근성·웹표준을
수정한 코드 옆에서 확인합니다.

문서 담당자

요구와 가이드를 확정하고
그 글에서 제출본을 만듭니다.

문서를 근거로 개발합니다.

01



프로젝트 관련 문서

제안요청서, 기술협상내용 등
작업용 Markdown으로 정리

02



탐색과 수정

코드와 문서를 함께 검색

03



실행과 점검

서버, 보안, 접근성, 표준

04



산출물 제출과 재사용

확정한 산출물을 제출하고 다시
작업용 Markdown으로 정리

AI가 참조하는 문서정본과 고객에게 제출된 산출물을 동일하게 유지하고 참조합니다.

시큐어 코딩, 웹취약점, 웹표준 점검을 상시 수행합니다.

PALYX는 행정안전부, KISA, NIA 등에서 배포하는
가이드 기준을 RAG로 활용합니다.

+ 소프트웨어 개발보안 가이드
+ 한국형 웹 콘텐츠 접근성 지침
+ 전자정부 웹사이트 품질관리 가이드 등
  현재 9개 문서 참조 중



공식 가이드를 기준으로, 점검부터 수정 확인까지

전자정부 프레임워크에 최적화된 검색을 수행합니다.



구문 구조

+

의미 검색

Tree-sitter와 벡터 임베딩 인덱스로 정확한 관련 코드 구간을 연결합니다.

개발에 필요한 이미지도 작업하는 자리에서 만듭니다.

화면 시안부터 아이콘, 설명용 그림까지
로컬 이미지 생성 모델로 만들고,
프로젝트와 산출물에 바로 활용합니다.

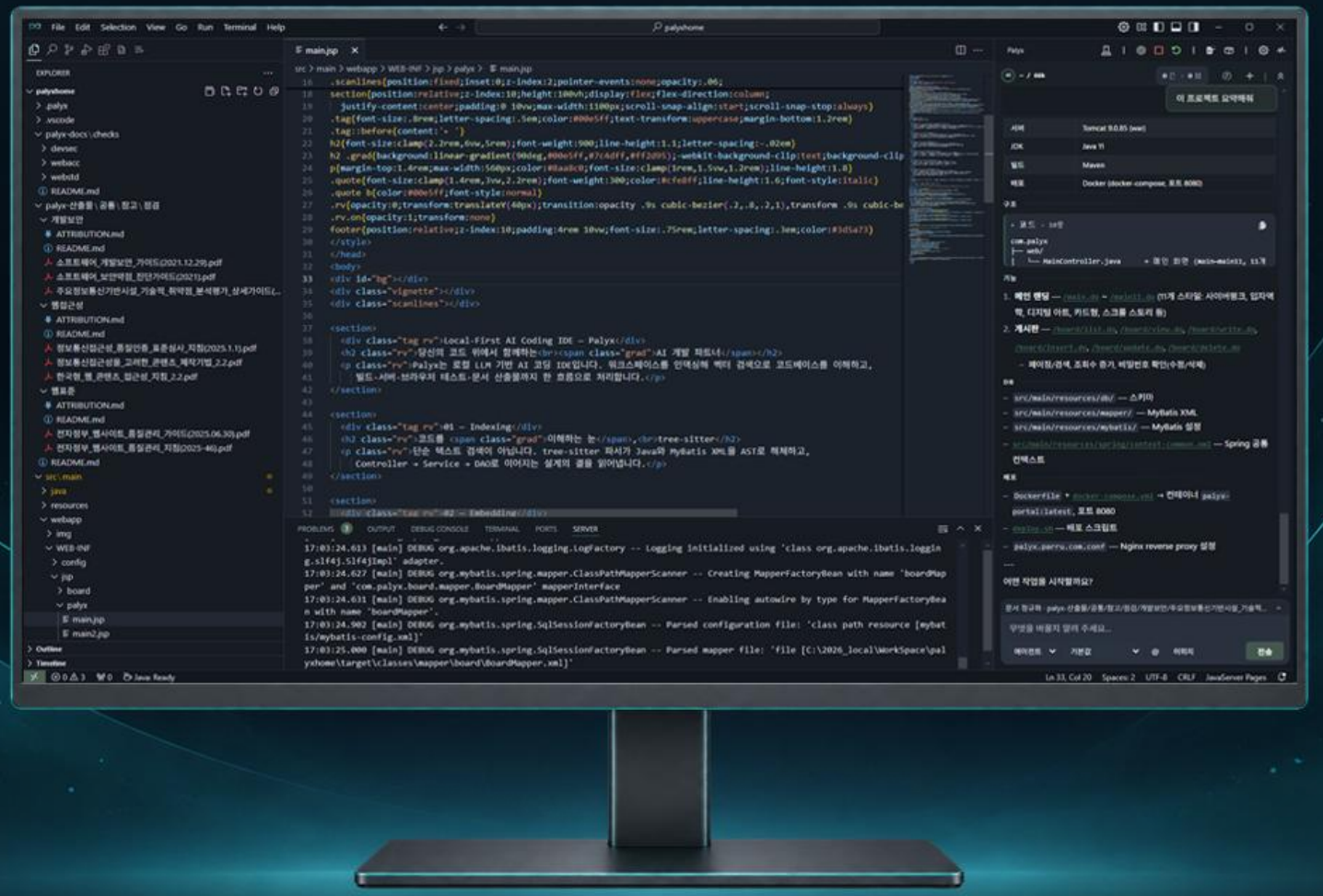


외부 연결 없이, 필요한 이미지를 내 작업 안에서.

실제 PALYX 실행 화면 입니다.

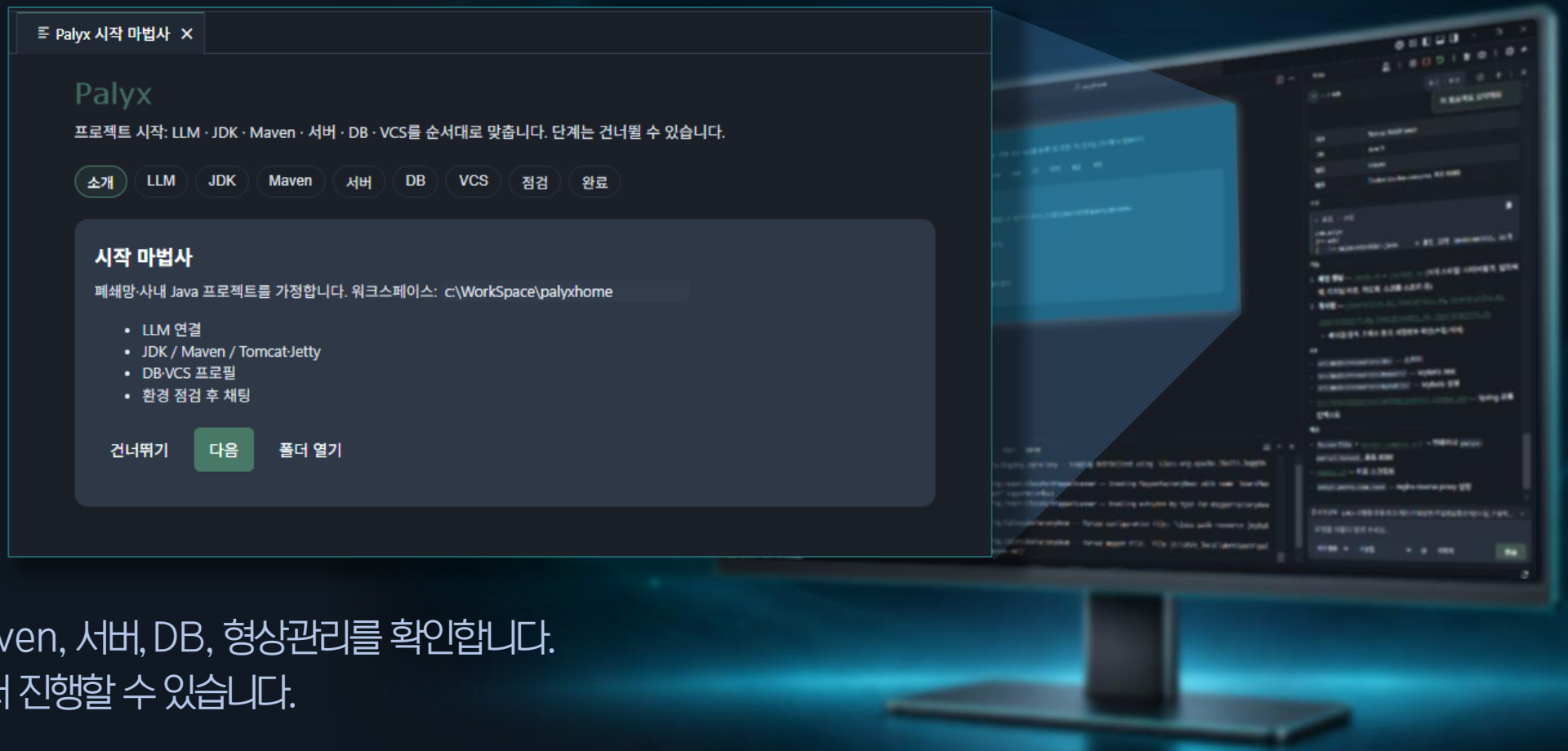
제공된 실제 화면

+ 탐색기
+ 작업영역
+ 에이전트
+ 출력패널



* 실제 Palyx 화면을 캡처하여 만들었습니다.

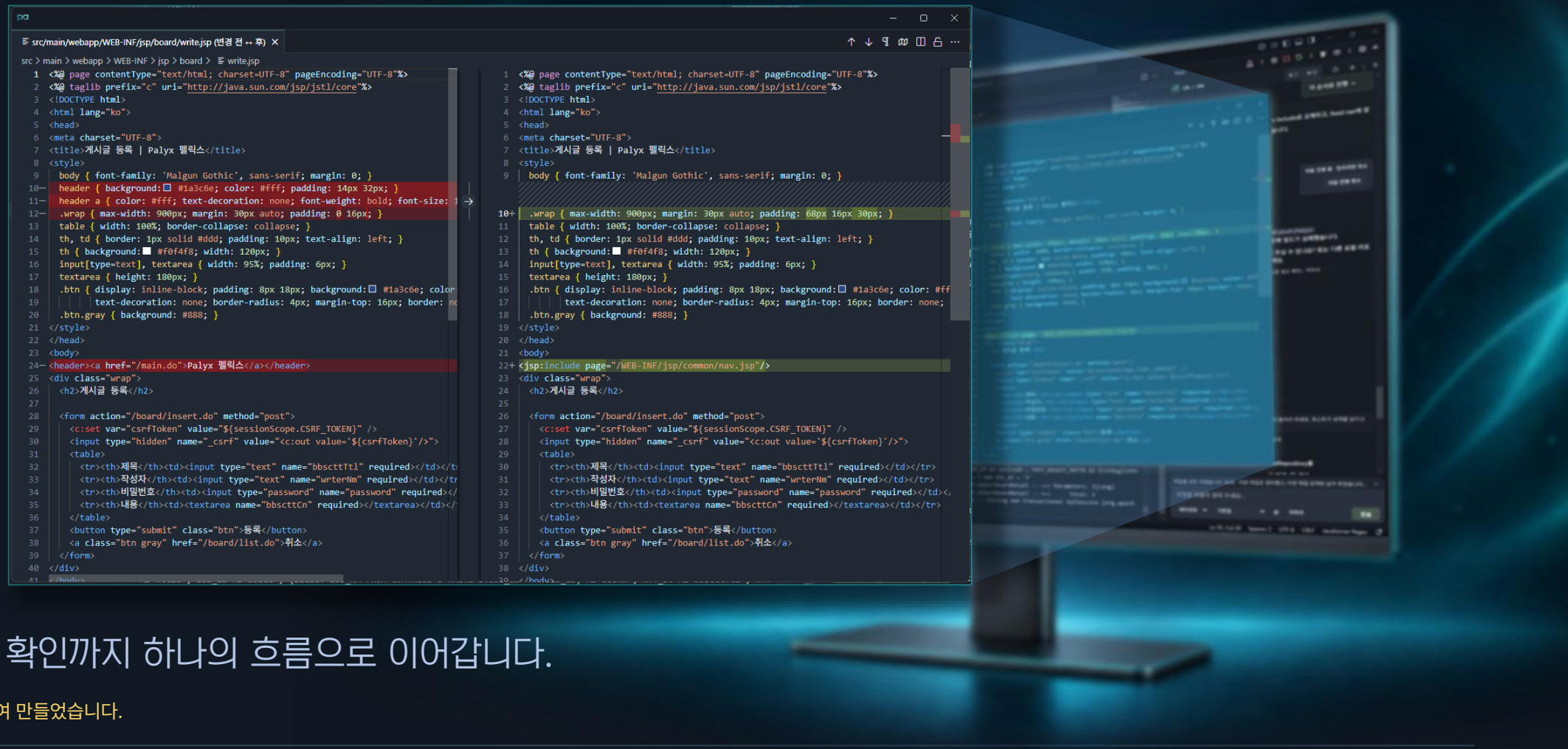
프로젝트 시작 마법사로 간단하게 환경 설정합니다.



모델, JDK, Maven, 서버, DB, 형상관리를 확인합니다.
필요한 단계부터 진행할 수 있습니다.

* 실제 Palyx 화면을 캡처하여 만들었습니다.

수정결과를 즉시 확인하고 보완합니다.



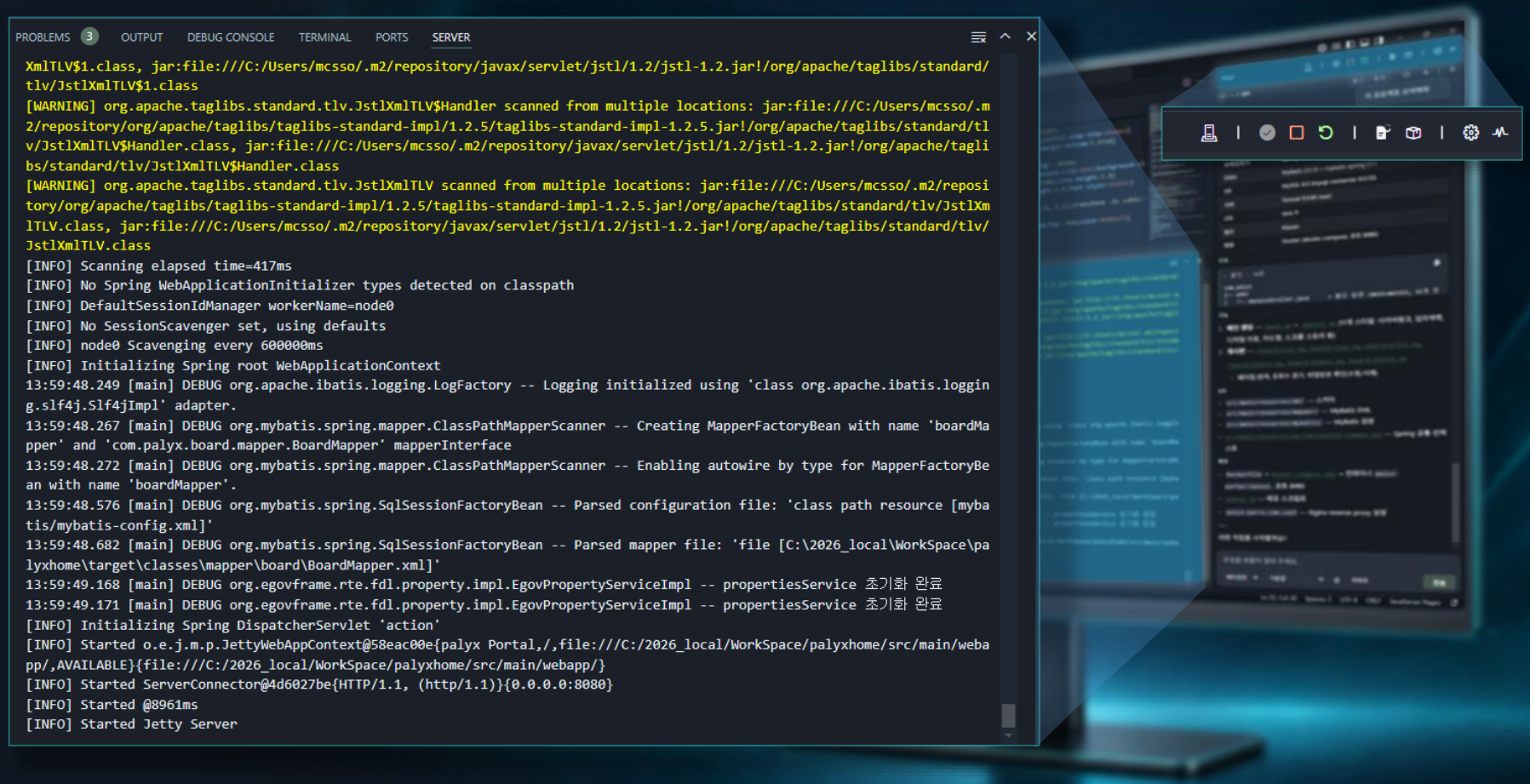
개발부터 실행·확인까지 하나의 흐름으로 이어갑니다.

* 실제 Palyx 화면을 캡처하여 만들었습니다.

작성한 코드를 실행하고 서버의 응답까지 같은 환경에서 이어갑니다.

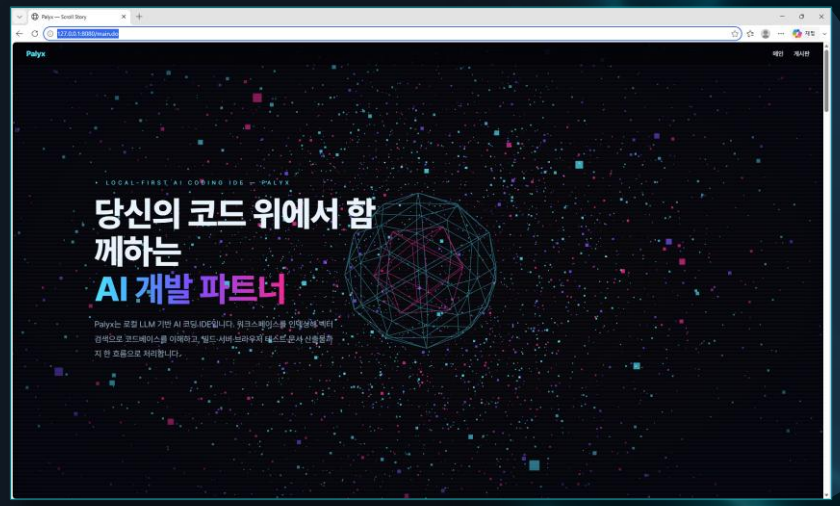
Maven 빌드부터
서버 구동, 디버그와
로그 확인까지
연결됩니다.

* Tomcat, Jetty 기본지원



* 실제 Palyx 화면을 캡처하여 만들었습니다.

수정한 화면을 스스로 브라우저에서 확인하고 점검합니다.



서버를 기동한 뒤 브라우저로 네비게이션 메뉴를 테스트하겠습니다.

1. 서버 기동
2. 메인 페이지 확인
3. 게시판 목록/상세/작성 페이지 확인

서버 start

서버가 이미 8080 포트에서 실행 중입니다. 로그를 확인하고 브라우저 테스트를 진행하겠습니다.

읽음 server.log

서버 정상 작동 중입니다. 이제 브라우저로 각 페이지의 네비게이션 메뉴를 테스트합니다.

메인 페이지

브라우저 open

✅ 메인 페이지 — 네비게이션 메뉴 정상 (Palyx / 메인 / 게시판)

게시판 목록 페이지

브라우저 click

✅ 게시판 목록 — 네비게이션 메뉴 정상 (Palyx / 메인 / 게시판)

게시판 상세 페이지

브라우저 click

✅ 게시판 상세 — 네비게이션 메뉴 정상 (Palyx / 메인 / 게시판)

게시판 작성 페이지

브라우저 click

✅ 게시판 작성 — 네비게이션 메뉴 정상 (Palyx / 메인 / 게시판)

메인 페이지로 돌아가기 확인

브라우저 click

✅ 메인 페이지로 정상 이동

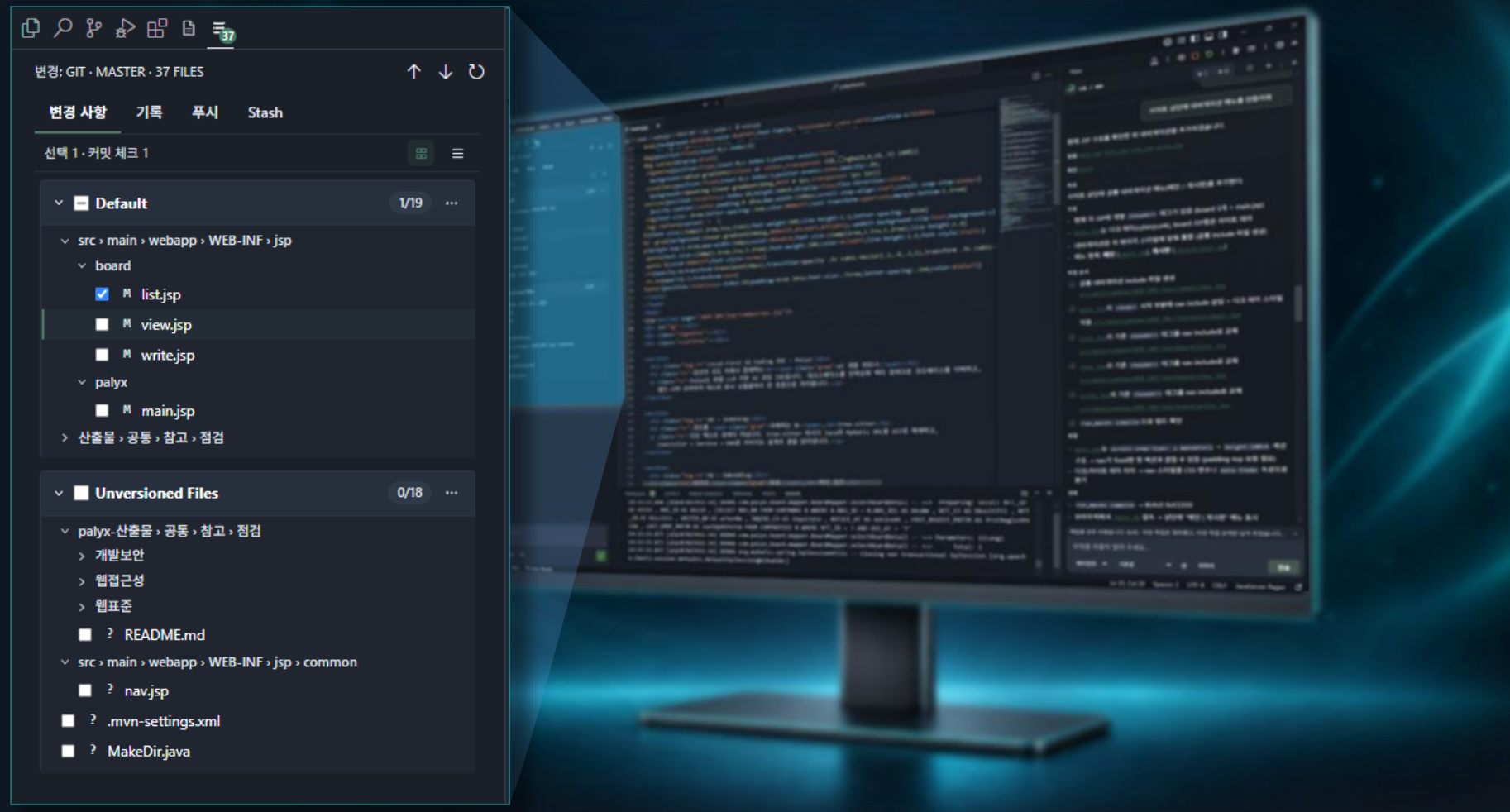
스크린샷으로 시각적으로도 확인하겠습니다.

서버에 반영한 화면을 직접 열어 기능과 동작을 확인하고, 필요한 부분을 보완합니다.

* 실제 Palyx 화면을 캡처하여 만들었습니다.

형상관리로 변경과 이력을 한곳에서 관리합니다.

자체 형상관리
GUI툴 제공으로
Git·SVN의
변경·커밋·동기화
작업을 수행합니다.



* 실제 Palyx 화면을 캡처하여 만들었습니다.

데이터베이스에 연결하고, 데이터를 확인 합니다.

DB 접속 상태를 점검하고 쿼리를
실행해 개발에 필요한 데이터를
같은 작업 환경에서 조회합니다.



* 실제 Palyx 화면을 캡처하여 만들었습니다.

산출물을 산출물 답게 활용합니다.

요구와 산출물을 Markdown으로
정리해 RAG로 구축/활용하여
AI가 코드와 함께 검색하고
참조하도록 연결 합니다.



* 실제 Palyx 화면을 캡처하여 만들었습니다.

사업별 산출물 테일러링하고, 일관된 기준으로 관리 합니다.

	A	B	C	D	E	
1	ID	단계	표준 산출물명	사업 산출물명	작성 여부	
60	CBD-D4	설계	인터페이스 설계서		N	
61	CBD-D7	설계	시스템시험 시나리오		N	
62	CBD-D8	설계	엔티티 관계 모형 기술서		N	
63	CBD-D9	설계	데이터베이스 설계서	테이블 정의서	Y	RFP 제
64	CBD-D12	설계	데이터 전환 및 조기데이터 설계서		Y	「데이
65	IM-UT	구현	단위테스트 계획/결과서	단위 시험 결과서	Y	RFP 제
66	IM-IT	구현	통합테스트 계획/결과서	통합 시험 결과서	Y	RFP 제
67	IM-AT	구현	인수테스트 계획/결과서		N	
68	IM-A11Y	구현	웹접근성점검보고서	웹접근성 점검 결과서	Y	RFP 제
69	IM-WEBSTD	구현	웹표준점검보고서	웹표준 점검 결과서	Y	RFP 제
70	IM-COMPAT	구현	웹호환성점검보고서		N	
71	IM-SRCQ	구현	소스품질검사보고서		N	
72	IM-SEC	구현	보안약점진단결과서		Y	RFP 제
73	IM-SRC	구현	프로그램 소스(이미지 포함)		N	
74	IM-PERF	구현	성능시험 결과서		Y	RFP 제
75	IM-EGOV	구현	표준프레임워크 적용 점검 결과서		Y	RFP 제
76	CBD-I3	구현	데이터베이스 테이블		N	
77	CBD-T2	시험	시스템시험 결과서		N	
78	TR-USER	이행	업무 사용자 매뉴얼		N	
79	TR-ADMIN	이행	업무 관리자 매뉴얼		N	
80	TR-OPS	이행	시스템 운영자 매뉴얼		N	
81	CBD-T5	시험	시스템 설치 결과서		N	
82	DT-ASIS	이행	데이터 현황 분석서		N	
83	DT-STD-P	이행	목표시스템 데이터 표준화 방안서		N	
84	DT-DICT	이행	데이터 표준 사전/정의서		N	
85	DT-STD-M	이행	데이터 표준 관리 방안서		N	
86	DT-STR	이행	데이터 구조 설계 방안서		N	
87	TR-MIG-P	이행	데이터 이행 계획서		Y	「데이
88	TR-MIG-R	이행	데이터 이행 결과서	데이터 이관 결과서	Y	RFP 제

이번 사업 테일러링

마스터 목록은 `tailoring-catalog.md` 를 보십시오.
고칠 때는 `palyx-산출물/공통/산출물테일러링결과서.xlsx` 의 작성 여부·사업 산출물명·비고를 고쳐 저장합니다(이 파일에 자동 반영).
작성 여부가 N인 산출물은 「산출물 작업」 메뉴에 나오지 않습니다. N으로 둔 산출물은 비고에 미적용 사유를 적습니다.

사업 정보

항목	내용
사업명	가상시 민원 게시판 시스템 구축 사업
작성일	2026-10-28
근거	`palyx-docs/sources/rfp.md`, `palyx-docs/sources/tech-consult.md`

산출물 테일러링 결과

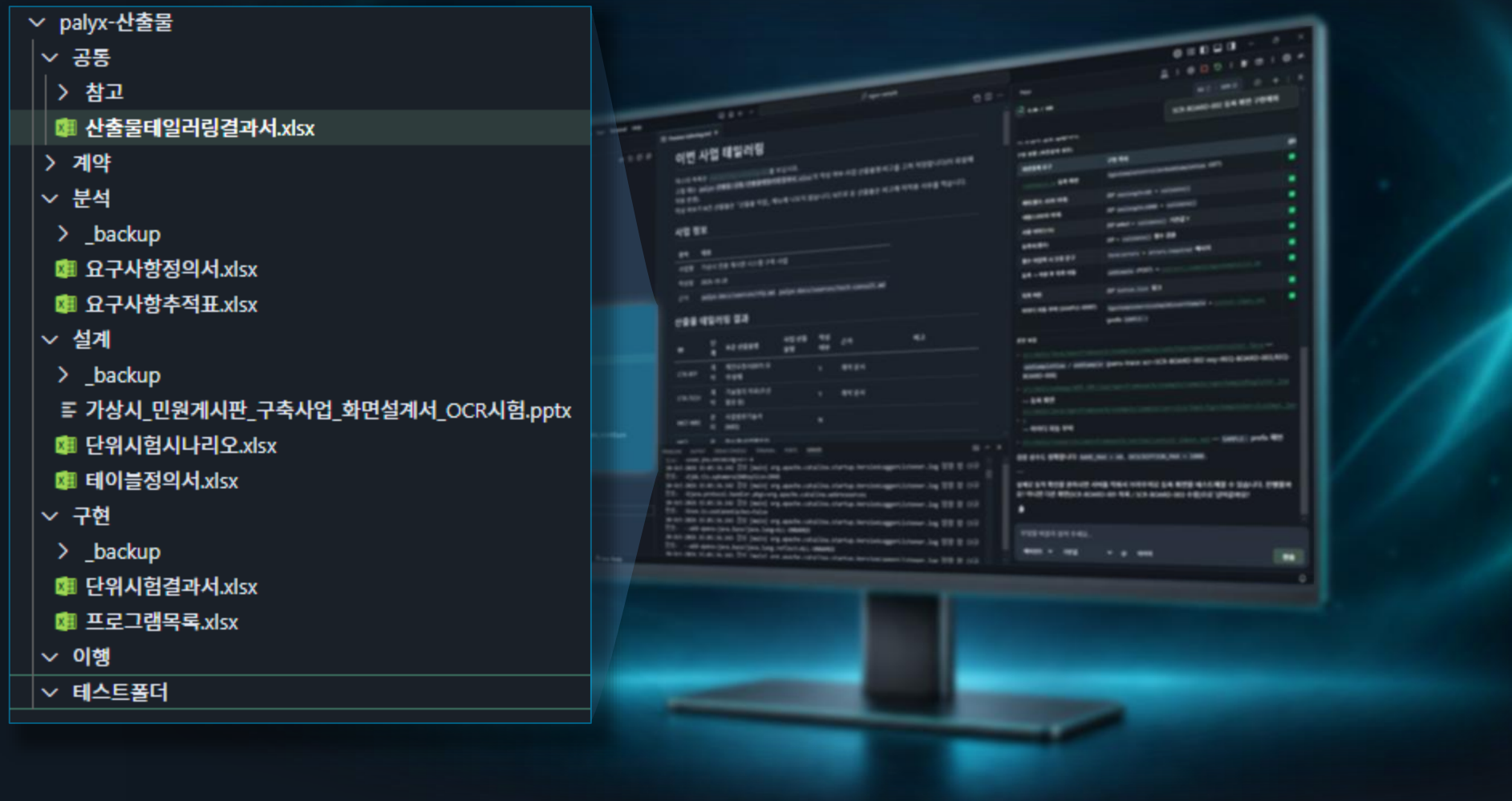
ID	단계	표준 산출물명	사업 산출물명	작성 여부	근거	비고
CTR-RFP	계약	제안요청서(RFP) 요약·발췌		Y	계약 문서	
CTR-TECH	계약	기술협의 자료(우선협상 등)		Y	계약 문서	
MGT-WBS	관리	사업범위기술서(WBS)		N		

사업별 산출물 양식을 학습하고, 항목 ID와 파일명 규칙은 같은 목록을 따릅니다.

* 실제 Palyx 화면을 캡처하여 만들었습니다.

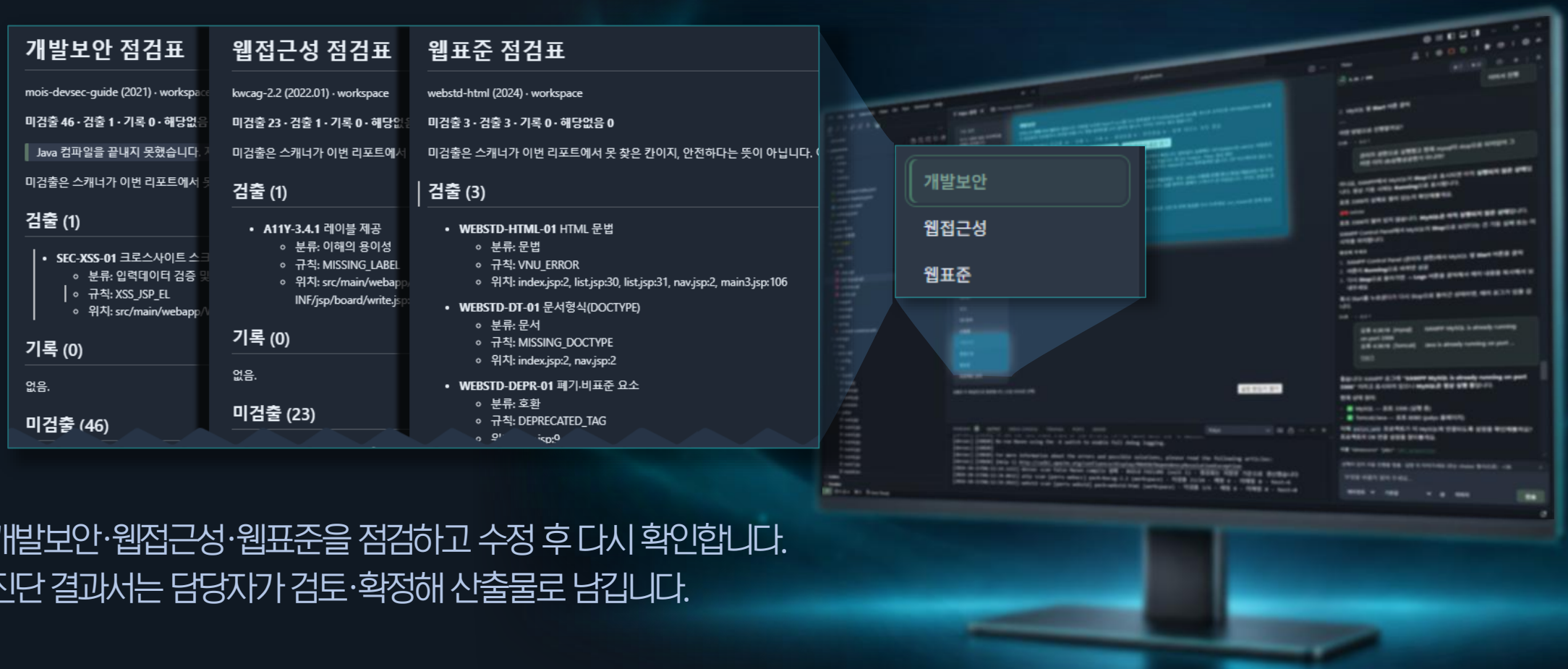
확정된 요구사항 부터 테스트 문서까지 이어갑니다.

계약 단계의 문서에서
요구사항정의서를 만들고,
담당자가 확정한 요구를
바탕으로 설계·시험계획·
결과서를 작성합니다.



* 실제 Palyx 화면을 캡처하여 만들었습니다.

코드를 점검하고 보완하며,확인한 결과를 기록합니다.

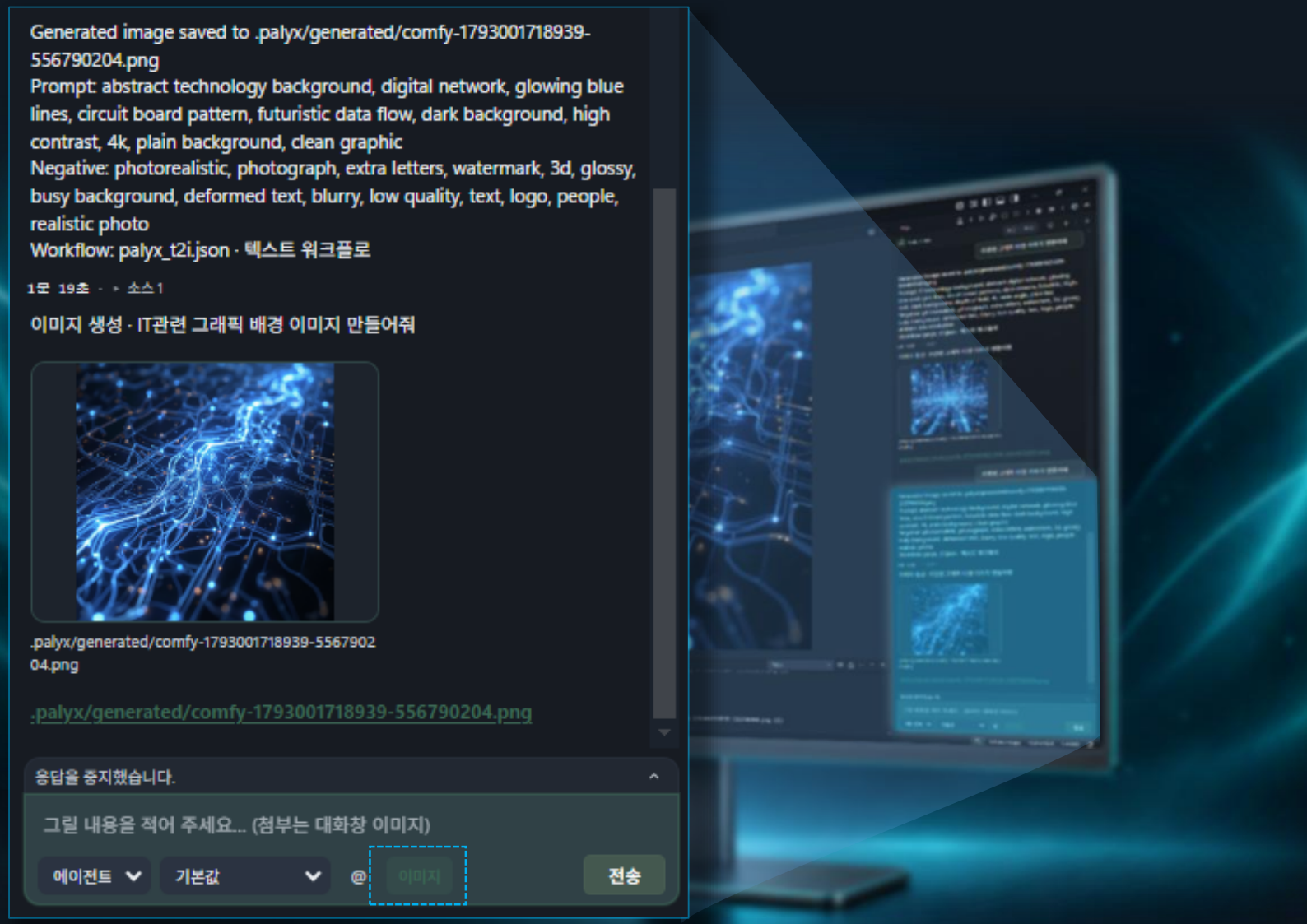


개발보안·웹접근성·웹표준을 점검하고 수정 후 다시 확인합니다.
진단 결과서는 담당자가 검토·확정해 산출물로 남깁니다.

* 실제 Palyx 화면을 캡처하여 만들었습니다.

프로젝트에 필요한 이미지를 직접 만들고 활용합니다.

화면 시안부터 아이콘, 설명용 그림까지 로컬에서 생성하고,프로젝트에 저장해 개발과 문서 작업에 사용합니다.



* 실제 Palyx 화면을 캡처하여 만들었습니다.

PALYX.PARRU.COM

프로젝트 수행의 새로운 표준을 제시합니다.

요구사항부터 구현, 점검, 제출까지 프로젝트 수행의 모든 과정을 Palyx와 함께하세요

