MCP/이미지 도구 모음 — 메인기획서

> 재구성 기획서 (역설계) / 작성 기준 시점: 2026-08-11 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P050 / 근거 수준: B / 작성: Claude (재구성) 이 문서는 결과물과 제작 기록으로부터 기획 의도를 역추정해 당시 작성한 것처럼 재구성한 것이다. 확인된 사실과 추정을 [근거]/[추정] 태그로 구분한다.

1. 핵심 메카닉스

생성 호출(입력→상태→피드백)

- 의도(Why): 프롬프트 한 번으로 완성 파일 경로를 돌려받는다.
- 설계(How): `comfy_generate_image`가 프롬프트/체크포인트(접두어 매칭), 크기, steps, cfg, sampler, seed, count를 받아 기본 txt2img 그래프(체크포인트 로드, 긍정/부정 인코딩, 빈 latent, KSampler, VAE 디코드, 저장)를 만들어 큐에 등록하고, 완료까지 폴링해 PNG를 `out_dir`에 저장한다. 기본값은 1024x1024, 25 steps, cfg 6.0, euler/normal, 대기 600초. `return_image`가 참이면 4MB 이하 PNG를 인라인으로 돌려준다. [근거: `mcp_server.py`, claude_bbeaf4a4]
- 필요(What): ComfyUI 서버, 체크포인트 목록 조회.
- 작업 방법: 상태 확인 → 체크포인트 조회 → 생성 → 파일 열람 검수.
- 완료 기준: 반환 경로에 이미지가 실재한다.
- 리스크/대안: 긴 작업은 `comfy_queue_prompt+comfy_job_result`로 비동기 처리.
- 근거: [근거: `mcp_server.py`]

커스텀 워크플로

- 의도(Why): img2img, ControlNet, 업스케일처럼 기본 그래프로 표현되지 않는 작업을 허용한다.
- 설계(How): API 포맷 워크플로 JSON을 인라인 또는 경로로 받아 실행하고, `prompt`가 있으면 첫 CLIPTextEncode 텍스트를 덮어쓴다. Codex용 `comfy-bridge`는 `submit`이 API 포맷 여부(각 노드에 class_type/inputs)를 검증해 편집기용 JSON을 거부한다. [근거: `mcp_server.py`, `comfy_api.py`]
- 필요(What): "Save (API Format)" 워크플로, 필요 노드/모델.
- 작업 방법: 노드 조회 → JSON 구성 → 제출 → history 확인 → fetch.
- 완료 기준: history에 출력 이미지가 나온다.
- 리스크/대안: ControlNet 모델 미설치 시 모델을 추가한 사례가 있다. [근거: codex_14fa5]
- 근거: [근거: `comfy_api.py`, README]

편집기 MCP 연결 운용

- 의도(Why): 영상/씬 편집을 에이전트에 위임한다.
- 설계(How): Premiere는 `LLM ⇔ MCP(Node) ⇔ WebSocket(8097) ⇔ CEP 패널 ⇔ ExtendScript`, AE는 호출마다 `.jsx`를 생성해 `AfterFX.exe -r`로 실행하고 결과 JSON 파일을 폴링, Unity는 HTTP(127.0.0.1:8080/mcp) 서버로 멀티 클라이언트를 지원한다. [근거: 각 README, claude_17e1d6b7]

- 필요(What): 편집기 실행, 패널 또는 브리지 시작, 설정 허용.
- 작업 방법: 등록 → 연결 상태 조회(connected) → 실제 응답(버전/프로젝트) 확인 → 작업 → 원본 보존 검증.
- 완료 기준: 실제 응답 확인. 패널 표시 "Connected"만으로는 완료로 보지 않는다. [근거: codex_043c1]
- 리스크/대안: 중복 서버가 포트를 점유하면 정리.
- 근거: [근거: codex_043c1, claude_17e1d6b7]

2. 구현 방법

- 구조: `comfy_client.py`(HTTP, stdlib) ← `mcp_server.py`(JSON-RPC 2.0, stdin/stdout 줄 단위). `initialize`, `ping`, `tools/list`, `tools/call`만 처리하고 도구 오류는 `isError` 결과로 반환(프로토콜 오류와 분리). [근거: `mcp_server.py`]
- 등록: Claude Code는 `C:\claude\.mcp.json`(프로젝트 스코프), Codex는 사용자 설정에 `comfyui` 이름으로 전역 등록. [근거: codex_a55f, comfy-bridge README]
- 보안: `comfy_api.py`는 http와 로컬호스트(localhost/127.0.0.1/::1)만 허용하고 공개 리스너/터널을 추가하지 않는다. [근거: `comfy_api.py`, comfy-bridge README]
- 환경변수: `COMFY_SERVER`, `COMFY_OUT_DIR`. [근거: `mcp_server.py`]
- comfy-bridge: 격리 `.venv`에 mcp 1.30.0, 도구 8종 (status/models/nodes/queue/submit/history/upload/fetch). [근거: comfy-bridge README]

3. 시스템 간 상호작용과 밸런스 기준

- 비용 판단 기준: "만족한 한 장을 얻을 때까지의 총 사용량"이며, 개별 생성/검수 비율은 확인된 수치가 없다. [근거: codex_14fa5]
- 시험 결과(1회, 단일 장면): Comfy 단독은 6회 불승인으로 종료, Codex 밑그림+Comfy는 승인되나 색감/디테일 저하, Codex 단독은 승인. 사용자는 Comfy 투입 전 밑그림 품질을 가장 선호. [근거: codex_14fa5]
- 판단: 대량 생성/참조 고정 환경에서 Comfy 가치가 있고, 소량 탐색은 대화형 생성이 낫다는 가정. [추정: 에이전트 답변을 사용자 반박 후 수정한 결과]

4. 게임의 나머지 부분의 방향

- 콘텐츠 구조, UI, 저장: 해당 없음(도구).
- 방향: 생성 모델 계열을 프로젝트 톤에 맞춰 지정, 후보 일괄 생성 뒤 선별 검수. [근거: codex_14fa5]
- 사용 사례: 박쥐/우파루파 수인 캐릭터 시안, 메이플월드/Unity 프로젝트 제작 지원, 하이넘 PV 편집. [근거: codex_019ffafe 계열, codex_e4b6, codex_043c1]

5. 제작 과정 설계

- 순서: 클라이언트 → MCP 래핑 → 등록 → 재시작 후 도구 노출 확인 → 사용 사례 검증.
- 병렬: 편집기별 MCP는 서로 독립.
- 분담: 에이전트가 제작/검증, 사용자가 앱 실행/패널 버튼/권한 승인 담당.
- 검증: 프로토콜 직접 호출(`tools/list`, `comfy_status`), 단색 64x64 연결 테스트, 실제 이미지 생성.

- 인수인계: 도구별 README와 에이전트 메모리, 재시작 필요 안내. [근거: codex_a55f, comfy-bridge README]

6. 세부기획서 목록

- `03_세부기획서_도구연동운용.md`: 도구 명세, 연결 절차, 비교 시험, 검수 기준.