

빌더 계정 사용 매뉴얼

PDF 다운로드

빌더 계정 사용 매뉴얼

목차

Part 1: 시작하기

- 시작하기
- 새로운 프로젝트 개발 환경 설치하기

Part 2: 프로젝트 관리

- Order Sheet 작성하여 보내기 및 Reports 가져오기
- 10단계 진행 프로세스 관리 및 진행률 표시하기
- Project Task Plan 및 SAL Grid 작성 및 활용법
- 예시 프로젝트 모든 문서 및 소스코드 파일 복사하여 저장하기

Part 3: 학습 및 지원

7. AI Tutor 사용법
8. 다른 AI에게 질문하기 및 크레딧 충전하기
9. Sunny에게 질문하기 및 1:1 맞춤형 코칭 프로그램

Part 4: 계정 관리

10. My Page 관리

1. 시작하기

1-1. 회원가입하기

1. SSAL Works 웹사이트에 접속합니다.
2. “회원가입” 버튼을 클릭합니다.
3. 필수 정보를 입력합니다:
 - **이메일:** 고유한 이메일 주소
 - **비밀번호:** 8자 이상 (영문+숫자+특수문자)
 - **닉네임:** 2-20자
 - **실명:** 2-50자 (필수 - 개설비 입금자명 확인, 환불, 세금계산서 발행에 사용)
4. 약관에 동의합니다:
 - 서비스 이용약관 (필수)
 - 개인정보 처리방침 (필수)
 - 마케팅 정보 수신 (선택)

5. “회원가입” 버튼을 클릭합니다.
6. 회원가입 완료 후 **회원 ID**(8자리 영숫자)가 발급됩니다.

1-2. 로그인하기

1. SSAL Works 웹사이트에 접속합니다.
2. 우측 상단의 “로그인” 버튼을 클릭합니다.
3. 로그인 방법을 선택합니다:
 - **Google 계정**: Google 버튼 클릭 후 계정 선택
 - **이메일**: 회원가입 시 등록한 이메일과 비밀번호 입력

1-3. 빌더 계정 개설하기

1. 로그인 후 왼쪽 사이드바 상단 **My Page** > 서비스 이용 현황으로 이동합니다.
2. “빌더 계정 개설비 납부” 버튼을 클릭합니다.
3. 개설비 결제 안내를 확인합니다:
 - 개설비: ₩3,300,000 (얼리버드 300만원 + VAT, 1회)
 - 월 이용료: ₩50,000/월 (첫 3개월 무료)
4. 결제를 완료하면 빌더 계정이 활성화됩니다.
5. **빌더 계정 ID**(12자리)가 발급됩니다.
6. 빌더 대시보드에 접근할 수 있습니다.

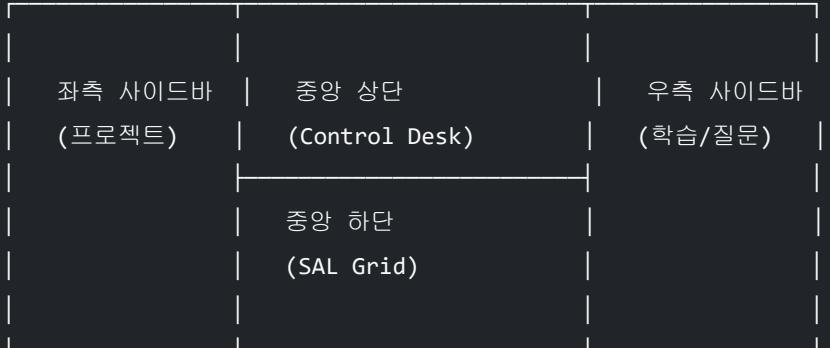
빌더 계정 ID 형식:

YYMMNNNNNNXX
||| | ← 금액 코드 (TH=300만원, FI=500만원)
||| └─ 월별 순번 (000001~)
|| └── 월 (01~12)

|————— 년도 (25=2025년)
|————— 예: 2512000005TH = 2025년 12월 5번째 등록, 300만원

1-4. 대시보드 화면 구성

대시보드는 5개 영역으로 구성되어 있습니다:



- 좌측 사이드바:** 프로젝트 목록, 진행 프로세스
- 중앙 상단:** Order Sheet 작성 공간
- 중앙 하단:** PROJECT SAL GRID
- 우측 사이드바:** Books, 실전 Tips, 외부 연동 설정 가이드, AI에게 질문하기, Sunny에게 묻기

2. 새로운 프로젝트 개발 환경 설치하기

빌더 계정을 개설하면 **프로젝트 등록**과 **개발 환경 설정**이 필요합니다.

2-1. 프로젝트 등록 및 Dev Package 다운로드

Dev Package은 프로젝트 개발에 필요한 폴더 구조, 설정 파일, 템플릿이 포함된 패키지입니다.

1. 좌측 사이드바의 “+ 새로운 프로젝트 등록” 버튼을 클릭합니다.
2. 프로젝트 정보를 입력합니다:
 - **프로젝트명**: 개발할 웹사이트 이름
 - **프로젝트 설명**: 간단한 설명 (선택)
3. “다음” 버튼을 클릭합니다.
4. 등록번호가 부여되면 “Dev Package 다운로드” 버튼을 클릭합니다.
5. Dev Package ZIP 파일이 다운로드됩니다 (project_id가 자동 주입됨).
6. 다운로드 완료 후 “확인” 버튼을 클릭하면 등록이 완료됩니다.
7. 등록된 프로젝트가 좌측 사이드바에 표시됩니다.

참고: Dev Package는 프로젝트 등록 시에만 다운로드할 수 있습니다.
다운로드를 완료해야 등록이 완료됩니다.

2-2. Dev Package 구성

패키지 주요 내용물:

```
SSAL_Works_Dev_Package/
|
├── api/                                ← 백엔드 API (개발하면서 채움)
├── pages/                               ← 프론트엔드 페이지 (개발하면서 채움)
└── assets/                              ← 정적 자원 (개발하면서 채움)
|
├── scripts/                            ← 자동화 도구
│   ├── sync-to-root.js                ← Stage → Root 자동 복사
│   └── build-web-assets.js          ← 통합 빌드
|
└── 표준 디렉토리 구조
    ├── P0_작업_디렉토리_구조_생성/
    ├── P2_프로젝트_기획/
    ├── P3_프로토타입_제작/
    └── S1_개발_준비/ ~ S5_개발_마무리/
```

```
|  
|   └── S0_Project-SAL-Grid_생성/  
|       ├── data/sal_grid.csv      ← Task 데이터 템플릿  
|       ├── manual/              ← SAL Grid 매뉴얼  
|       └── sal-grid/            ← Task/검증 지침 템플릿  
|  
└── Human_ClaudeCode_Bridge/  
    ├── Orders/                ← 작업 지시서 폴더  
    └── Reports/               ← 작업 결과 폴더
```

2-3. AI 규칙 (자동 포함)

Dev Package에는 AI 작업 규칙이 이미 포함되어 있습니다.

```
└── .claude/  ← AI가 참고하는 규칙 폴더
```

이 폴더 안에는 파일 저장 위치, 코드 작성 방식, 검증 기준 등 AI가 따라야 할 규칙이 들어있습니다.

별도 설정 불필요! 패키지를 다운받으면 AI 규칙이 자동으로 적용됩니다.

2-4. 개발 도구 설치

패키지만으로는 개발할 수 없습니다. 개발 도구도 함께 설치해야 합니다.

용어 설명 - 터미널: 컴퓨터에 텍스트 명령어를 입력하는 프로그램.

Windows의 “명령 프롬프트”, Mac의 “터미널” 앱이 이에 해당합니다. -

CLI (Command Line Interface): 마우스로 버튼을 클릭하는 대신, 텍스트 명령어를 직접 입력해서 프로그램을 실행하는 방식. 텍스트 명령창이라고 생각하시면 됩니다.

1. Claude Code 설치 (터미널에서 실행):

```
npm install -g @anthropic-ai/claude-code
```

2. 패키지 폴더에서 Claude Code 실행:

```
cd 패키지폴더경로  
claude
```

3. 설치 요청:

"프로젝트 개발 환경 설정을 위한 필수 도구 다 설치해 줘"

4. Claude Code가 Git, Node.js, npm 패키지 등을 자동으로 설치합니다.

3. Order Sheet 작성하여 보내기 및 Reports 가져오기

3-1. Order Sheet란?

Order Sheet는 Claude Code에게 작업을 지시하는 문서입니다. SSAL Works의 핵심 작업 흐름인 **Human-ClaudeCode Bridge** 시스템의 일부입니다.

항목	설명
작업 유형	어떤 종류의 작업인지 (기능 개발, 버그 수정, 문서 작성 등)
작업 내용	구체적으로 무엇을 해야 하는지
참고 자료	참고할 파일이나 문서 경로
기대 결과물	어떤 결과물이 나와야 하는지
Task ID	SAL Grid의 Task ID (선택)

3-2. 작업 흐름 (Human-ClaudeCode Bridge)



3-3. Order Sheet 작성하기

- 중앙 상단의 **Control Desk** 영역 확인
- “**Order Sheet 템플릿**” 드롭다운에서 템플릿 선택:
 - 기능 개발: 새 기능 구현
 - 버그 수정: 오류 해결
 - 코드 리뷰: 코드 검토 요청
 - 문서 작성: 문서 생성/수정
 - 테스트: 테스트 실행
- 템플릿이 에디터에 로드됩니다
- 각 항목을 채웁니다

작성 팁: - 구체적으로 작성할수록 더 정확한 결과물을 얻을 수 있습니다 - 참고 파일 경로를 정확히 명시하세요 - 예상 결과물의 형태를 설명하세요

3-4. Order Sheet 복사하여 Claude Code에 전달하기

1. Order Sheet 작성 완료 후 “Order Sheet 복사하기” 버튼 클릭
2. 전체 내용이 클립보드에 복사됩니다
3. Claude Code 터미널 열기:

```
cd 프로젝트폴더  
claude
```

4. **Ctrl+V**로 붙여넣기

5. Enter를 눌러 작업 시작

3-5. Reports 가져오기

AI가 작업을 완료하면 [Human_ClaudeCode_Bridge/Reports/](#) 폴더에 결과가 저장됩니다.

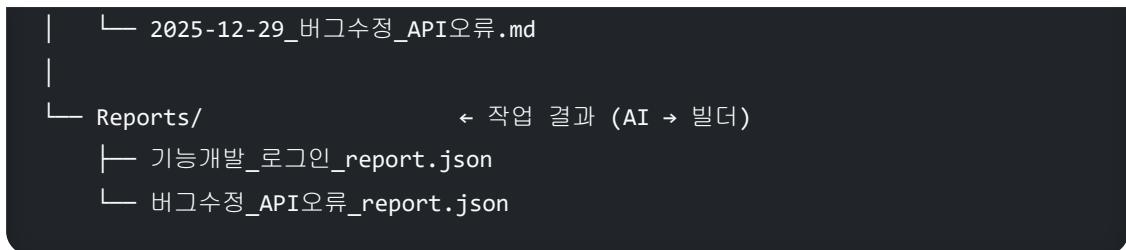
Report 저장 형식: `{작업명}_report.json`

1. “Reports 가져오기” 버튼 클릭
2. 파일 목록에서 확인할 Report 선택
3. Report 내용이 화면에 표시됩니다

Report에 포함되는 정보: | 항목 | 내용 | |————|————| | 작업 요약 | 수행한 작업 개요 | | 생성 파일 | 새로 생성된 파일 목록 | | 수정 파일 | 수정된 파일 목록 | | 테스트 결과 | 테스트 통과 여부 | | 검증 상태 | verification_status | | 주의사항 | 추가 작업/확인 필요 사항 |

3-6. Orders/Reports 폴더 구조

```
Human_ClaudeCode_Bridge/  
|—— Orders/           ← 작업 지시서 (빌더 → AI)  
|   |—— 2025-12-29_기능개발_로그인.md
```



4. 10단계 진행 프로세스 관리 및 진행률 표시하기

SSAL Works의 개발 프로세스는 **10단계**로 구성되어 있습니다.

4-1. 프로세스 개요

단계	코드	명칭	주요 내용	산출물
1	P0	디렉토리 구조 생성	폴더 구조 자동 생성	표준 폴더 구조
2	P1	사업계획	시장조사, 경쟁분석	사업계획서
3	P2	프로젝트 기획	Requirements, Architecture	기획 문서 세트
4	P3	프로토타입	UI/UX 목업, DB 설계	프로토타입
5	S0	SAL Grid 생성	Grid 템플릿 생성	sal_grid.csv, Task 템플릿
6	S1	개발 준비	환경설정, DB 스키마, Auth	개발 환경
7	S2	개발 1차	OAuth, 회원가입, 핵심 API	인증 시스템

단계	코드	명칭	주요 내용	산출물
8	S3	개발 2차	AI 연동, 구독 권한	AI 기능
9	S4	개발 3차	결제, 관리자, 크레딧	결제 시스템
10	S5	개발 마무리	배포, QA, 안정화	프로덕션 배포

4-2. 단계별 상세 내용

[P0] 디렉토리 구조 생성 - 표준 폴더 구조 자동 생성 - .claude 규칙 파일 배치
- Human_ClaudeCode_Bridge 폴더 생성

[P1] 사업계획 - 시장조사 보고서 - 경쟁분석 - 사업계획서

[P2] 프로젝트 기획 - Project Plan - Requirements Document - Architecture Design - UI/UX Design - DB Design - API Design

[P3] 프로토타입 - UI/UX 목업 - DB 스키마 구현 - 기본 페이지 구조

[S0] SAL Grid 생성 - Task 계획서 작성 - 5x11 매트릭스 구성 - Task/Verification Instruction 템플릿

[S1] 개발 준비 - Vercel 프로젝트 설정 - 환경변수 설정 - DB 스키마 생성 - Auth 설정 (Supabase Auth)

[S2] 개발 1차 (인증) - Google OAuth - 이메일 로그인 - 회원가입/로그인 UI - 세션 관리

[S3] 개발 2차 (AI) - AI API 연동 - 구독 권한 체계 - AI Q&A UI

[S4] 개발 3차 (결제) - 결제 연동 (TossPayments) - 관리자 대시보드 - 크레딧 시스템 - E2E 테스트

[S5] 개발 마무리 - 프로덕션 배포 - QA 테스트 - 버그 수정 - 안정화

4-3. 프로세스 확인하기

1. 좌측 사이드바의 “진행 프로세스” 섹션을 확인합니다.

2. 각 단계를 클릭하면:

- 하위 작업 목록이 펼쳐집니다
- 상태 아이콘으로 진행 상황 확인:
 - 대기
 - 진행 중
 - 완료

4-4. 안내문(Briefing) 열람하기

1. 진행 프로세스에서 단계 옆의 아이콘을 클릭합니다.

2. 해당 단계의 안내문이 팝업으로 표시됩니다.

3. 안내문에는 다음 내용이 포함됩니다:

- 단계 목표
- 수행 작업 목록
- 참고 자료
- 예상 산출물

4-5. 진행률 표시하기

대시보드에서 프로젝트 진행률을 실시간으로 확인할 수 있습니다.

진행률 표시 위치: - 좌측 사이드바: 단계별 진행률 바 - 중앙 상단: 전체 프로젝트 진행률 - SAL Grid: Task별 개별 진행률

진행률 계산 방식:

$$\text{전체 진행률} = (\text{Completed Task 수} / \text{전체 Task 수}) \times 100\%$$

$$\text{Stage별 진행률} = (\text{Stage 내 Completed Task} / \text{Stage 전체 Task}) \times 100\%$$

진행률 색상: | 진행률 | 색상 | 상태 | |————|————|————| | 0% | 회색 | 미시작 ||
1-49% | 파란색 | 진행 중 || 50-99% | 주황색 | 중간 단계 || 100% | 녹색 | 완료 |

4-6. Stage Gate 검증

Stage Gate = 다음 단계로 넘어가기 전 검문소

건설 현장에서 1층 공사가 끝나야 2층 공사를 시작하듯이, 웹사이트 개발도 각 단계(Stage)가 완료되어야 다음 단계로 넘어갑니다.

STAGE	이름	GATE 통과 조건
S1	개발 준비	환경설정, DB 스키마 완료
S2	개발 1차	로그인, 회원가입 기능 완료
S3	개발 2차	AI 연동 기능 완료
S4	개발 3차	결제, 관리자 기능 완료
S5	개발 마무리	전체 테스트, 배포 완료

Gate 통과 = 빌더(PO)의 승인

AI가 "S2 Stage Gate 검증 완료" 리포트를 제출하면, 빌더가 직접 테스트 후 승인해야 S3로 넘어갑니다.

검증 결과: - **Pass:** 다음 Stage로 진행 가능 - **Fail:** 문제 해결 후 재검증 필요

5. Project Task Plan 및 SAL Grid 작성 및 활용

법

5-1. SAL Grid란?

웹사이트 개발에는 수십~수백 개의 작업이 필요합니다. 이 작업들을 체계적으로 관리하지 않으면 “뭘 해야 하는지”, “어디까지 했는지” 파악하기 어렵습니다.

SAL Grid는 모든 작업을 **3차원 좌표**로 정리하여 한눈에 볼 수 있게 해주는 도구입니다.

축	의미	내용
S (Stage)	언제 하는가	5단계 개발 순서 (S1~S5)
A (Area)	무슨 영역인가	11개 개발 영역 (프론트엔드, 백엔드 등)
L (Level)	몇 번째 작업인가	해당 Stage+Area 내 작업 순서

예: **S2F1** = 2단계(개발1차) + 프론트엔드 + 1번째 작업

5-2. Grid 22개 속성

각 Task는 22개 속성으로 구성되며, **4개 그룹**으로 나뉩니다:

▣ 기본 정보 (1-5)

#	속성	설명
1	task_id	Task 고유 ID (예: S2F1)
2	task_name	Task 이름
3	stage	Stage 코드 (1~5)
4	area	Area 코드 (F, BA, D 등)
5	level	난이도 (1~3)

 **작업 관리 (6-13)**

#	속성	설명
6	task_status	작업 상태
7	task_progress	진행률 (0~100)
8	dependencies	선행 Task
9	task_instruction	Task 수행 지침
10	task_agent	담당 AI Agent
11	generated_files	생성된 파일 목록
12	duration	소요 시간
13	build_result	빌드 결과

 **검증 관리 (14-21)**

#	속성	설명
14	verification_instruction	검증 지침
15	verification_agent	검증 Agent
16	test_result	테스트 결과
17	build_verification	빌드 검증
18	integration_verification	통합 검증
19	blockers	차단 요소
20	comprehensive_verification	종합 검증
21	ai_verification_note	AI 검증 의견

■ Stage Gate (22)

#	속성	설명
22	stage_gate_status	Stage Gate 상태

5-3. Grid 보기 모드

1. 중앙 하단의 SAL Grid 영역 확인
2. 우측 상단 보기 모드 버튼:
 - **2D Card View:** 카드 형태
 - **3D Block View:** 3차원 블록
3. 풀스크린 버튼으로 전체 화면 전환

5-4. Task 상태 확인

상태별 색상: | 상태 | 색상 | 의미 | |——|——|——| Pending | 회색 | 대기 중 | | In Progress | 파란색 | 진행 중 | | Executed | 주황색 | 실행 완료 (검증 전) | | Completed | 녹색 | 검증까지 완료 | | Needs Fix | 빨간색 | 수정 필요 |

5-5. Task 상세 정보 보기

1. Grid에서 Task 카드 클릭
2. 상세 정보 패널에서 22개 속성 확인
3. Task Instruction, 생성 파일, 검증 상태 등 확인

5-6. Task 상태 전이 규칙 (중요!)

Task는 정해진 순서대로만 상태가 변경됩니다:

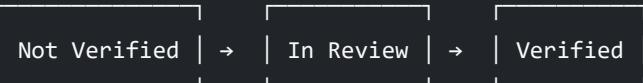
task_status 전이:





대기 중 작업 중 작업 완료 검증까지 완료

`verification_status` 전이:



검증 전 검증 중 검증 통과



Needs Fix (수정 필요)

핵심 규칙: - **Executed** = 파일 생성/수정 완료, 검증 전 - **Completed** = 검증 (Verified)까지 완료 - **Completed**는 반드시 **Verified** 후에만 가능!

5-7. Task 실행 6단계 프로세스

SAL Grid Task는 다음 6단계 프로세스로 실행됩니다:

STEP 1: Task Instruction 읽기
→ `sal-grid/task-instructions/{TaskID}_instruction.md` 확인



STEP 2: 규칙 파일 확인
→ `.claude/rules/` 폴더의 관련 규칙 읽기



STEP 3: Grid 상태 업데이트
→ `task_status: 'Pending' → 'In Progress'`



STEP 4: Task 작업 수행
→ Task Instruction에 따라 코드 작성/수정
→ 완료 시: `task_status: 'In Progress' → 'Executed'`





6단계 프로세스와 빌더의 역할:

단계	내용	빌더(PO) 역할
1단계	AI가 Task 실행	대기 - AI 작업 진행
2단계	AI가 도움 요청	응답 - 외부 설정, 의사결정
3단계	AI가 검증 수행	대기 - AI 검증 진행
4단계	Stage Gate 검증	대기 - AI 종합 검증
5단계	테스트 가이드 제공	확인 - 가이드 읽기
6단계	PO 최종 승인	테스트 & 승인 - 직접 확인 후 승인/거부

빌더가 직접 개입하는 시점: - 2단계: AI가 "Google OAuth 설정이 필요합니다" 같은 요청을 할 때 - 6단계: AI가 완성한 기능을 직접 테스트하고 승인할 때

나머지 단계: AI가 자동으로 진행하므로 기다리시면 됩니다.

5-8. Project Task Plan 만들기

Task Plan은 프로젝트의 모든 Task를 정의하는 마스터 문서입니다.

Task Plan 구성요소: | 항목 | 내용 | ——— | ——— | | Task 목록 | 전체 Task ID, 이름, 설명 | | 5x11 매트릭스 | Stage × Area 분포도 | | 의존성 다이어그램 | Task

간 선후 관계 | | Stage별 상세 | 각 Stage의 Task 목록 |

Task Plan 작성 순서: 1. 프로젝트 요구사항 분석 2. 필요한 기능 목록 도출 3. 각 기능을 Task로 분해 4. Stage/Area/Level 할당 5. 의존성 정의 6. Task Instruction 파일 작성

저장 위치:

```
S0_Project-SAL-Grid_생성/sal-grid/
├── TASK_PLAN.md           ← 마스터 문서
├── 5x11_MATRIX.md         ← 매트릭스
├── TASK_DEPENDENCY_DIAGRAM.md ← 의존성
└── task-instructions/     ← Task별 지침
```

5-9. Grid 작성하기

새 Task 추가: 1. “+ Task 추가” 버튼 클릭 2. Task ID 형식: **S[Stage][Area]**
[번호] - 예: S2F1 = Stage2 + Frontend + 1번째 - 예: S3BA2 = Stage3 +
Backend APIs + 2번째 3. 필수 정보 입력: - Task Name - Stage, Area, Level -
Task Instruction - Dependencies - Task Agent (담당 AI) - Verification Agent
(검증 AI)

Task 수정: 1. Task 카드 클릭 2. “수정” 버튼 클릭 3. 정보 수정 후 저장

Task ID 규칙: | Stage | Area 코드 | 예시 | |——|——|——| | S1 (개발 준비)
| F, BA, D, S, O | S1F1, S1D1 | | S2 (개발 1차) | F, BA, S | S2F1, S2BA1 | | S3
(개발 2차) | F, BA, E | S3E1 | | S4 (개발 3차) | F, BA, T | S4F1, S4T1 | | S5 (개발
마무리) | F, O, T | S5O1 |

6. 예시 프로젝트 모든 문서 및 소스코드 파일 복 사하여 저장하기

SSAL Works는 실제 개발 사례인 예시 프로젝트의 모든 문서와 소스코드를 제공합니다.

6-1. 예시 프로젝트란?

SSAL Works 플랫폼 자체가 예시 프로젝트입니다.

항목	내용
프로젝트명	SSAL Works
개발 방식	Claude Code + SAL Grid 기반 풀스택 개발
기술 스택	Vanilla JS, Supabase, Vercel, Resend, TossPayments

6-2. 복사 가능한 문서

1. 사업계획 문서 (P1 폴더)

- 시장조사 보고서
- 경쟁분석
- 사업계획서

2. 프로젝트 기획 문서 (P2 폴더)

- Project Plan
- Requirements Document
- Architecture Design
- UI/UX Design
- DB Design
- API Design

3. SAL Grid 관련 문서 (S0 폴더)

- Task 계획서 (SSALWORKS_TASK_PLAN.md)
- 5x11 매트릭스 (SSALWORKS_5x11_MATRIX.md)
- 의존성 다이어그램
- 모든 Task Instruction 파일
- 모든 Verification Instruction 파일

6-3. 복사 가능한 소스코드

영역	폴더	내용
프론트엔드	pages/ , assets/	HTML, CSS, JS 파일
백엔드 API	api/Backend_APIs/	비즈니스 로직 API
인증/보안	api/Security/	OAuth, 세션 관리
외부 연동	api/External/	AI, 결제 연동
데이터베이스	Database/	SQL 스키마, RLS 정책

6-4. 복사 방법

방법 1: Dashboard에서 다운로드 1. 우측 사이드바에서 “📁 예시 프로젝트” 클릭 2. 원하는 문서/폴더 선택 3. “다운로드” 버튼 클릭

방법 2: Claude Code에게 요청

“SSAL Works 예시 프로젝트의 [문서명/폴더명]을 내 프로젝트에 복사해 줘”

6-5. 활용 팁

- **그대로 사용 금지:** 복사한 후 자신의 프로젝트에 맞게 수정하세요
- **구조 참고:** 폴더 구조, 파일 명명 규칙, 코드 패턴 참고

- **문서 템플릿**: 기획 문서의 형식과 구성 참고

7. AI Tutor 사용법

7-1. AI Tutor란?

AI Tutor는 SSAL Works 지식베이스(학습용 Books, 실전 Tips, 외부 연동 설정 Guide, 안내문, Order Sheet, 매뉴얼, 서비스 소개, 기술 스택, AI 준수사항, 작업 규칙, 작업 방법 등)를 참고하여 맞춤 답변을 제공하는 AI 기반 튜터입니다.

학습 주제	내용
플랫폼 사용법	Dashboard, Order Sheet, SAL Grid 등
Claude Code 사용법	명령어, 슬래시 명령, 효율적 활용법
웹 개발 기초	HTML, CSS, JavaScript, API 등
프로젝트 관리	SAL Grid, Stage Gate, 의존성 관리

7-2. AI Tutor 사용하기

1. 우측 사이드바에서 “🎓 AI Tutor” 클릭
2. 학습 주제 선택 또는 질문 입력
3. AI Tutor가 답변과 학습 자료 제공
4. 추가 질문으로 심화 학습

7-3. 학습 콘텐츠



권	제목	편 수
제1권	Claude/Claude Code 사용법	34편
제2권	풀스택 웹사이트 개발 기초지식	34편
제3권	프로젝트 관리 방법	12편

💡 **실전 Tips (18개 카테고리)** - 개발 팁 - 에러 해결 - Git 사용법 - 배포 관련
- 등등

🔗 **외부 연동 설정 가이드 (5개)** - Google OAuth 설정 - Supabase 설정 -
Vercel 배포 - Resend 이메일 - TossPayments 결제

7-4. Books 열람하기

1. 우측 사이드바에서 “📚 학습용 Books” 클릭
2. 원하는 Book 선택
3. 편(Chapter) 목록에서 읽고 싶은 편 클릭
4. 내용이 팝업으로 표시

7-5. 실전 Tips 열람하기

1. 우측 사이드바에서 “💡 실전 Tips” 클릭
2. 카테고리 목록에서 원하는 주제 선택
3. 팁 내용이 팝업으로 표시

7-6. 외부 연동 설정 가이드 열람하기

1. 우측 사이드바에서 “🔗 외부 연동 설정 가이드” 클릭
2. 설정할 서비스 선택 (Google, Supabase, Vercel 등)
3. 단계별 설정 가이드 확인

8. 다른 AI에게 질문하기 및 크레딧 충전하기

8-1. 지원되는 AI 서비스

AI 서비스	추천 용도
ChatGPT	일반적인 개발 질문
Gemini	코드 리뷰, 설계 질문
Perplexity	최신 기술 정보 검색

8-2. AI에게 질문하기

1. 우측 사이드바에서 “🤖 다른 AI에게 물기” 클릭
2. AI 서비스 선택
3. 질문 입력
4. “질문하기” 버튼 클릭
5. AI 답변 확인

8-3. 크레딧 사용

- AI 질문 시 크레딧이 차감됩니다
- 대시보드 상단에서 잔액 확인
- 부족 시 “크레딧 충전” 클릭

8-4. 질문/답변 이력 보기

1. “질문/답변 이력” 버튼 클릭
2. 과거 질문과 답변 목록 표시

- 원하는 항목 클릭하여 상세 확인

9. Sunny에게 질문하기 및 1:1 맞춤형 코칭 프로그램

9-1. 코칭 서비스 개요

Sunny는 SSAL Works의 1:1 맞춤형 코칭 서비스입니다.

서비스	내용
플랫폼 사용 문의	사용법, 문제 해결
기술 멘토링	개발 방향, 아키텍처 조언
비즈니스 멘토링	사업계획서, 전략 자문
회계/세무 자문	세무, 회계 처리 안내

9-2. 플랫폼 사용 문의하기

- 우측 사이드바에서 “ Sunny에게 물기” 클릭
- 문의 유형 선택:
 - 사용법 문의
 - 문제 해결
 - 버그 신고
 - 기능 제안
- 질문 내용 작성
- 필요시 파일 첨부

5. “문의하기” 버튼 클릭

9-3. 비즈니스 멘토링 받기

1. 문의 유형에서 “비즈니스 멘토링” 선택

2. 멘토링 주제 선택:

- 사업계획서 검토
- 사업 전략 자문
- 비즈니스 모델 검토

3. 상세 내용과 질문 작성

4. 관련 자료 첨부

5. “문의하기” 버튼 클릭

9-4. 회계/세무 자문 받기

1. 문의 유형에서 “회계/세무 자문” 선택

2. 자문 주제 선택:

- 세무 관련 질문
- 회계 처리 안내

3. 질문 내용 작성

4. “문의하기” 버튼 클릭

10. My Page 관리

10-1. 프로필 수정하기

- 좌측 사이드바에서 “My Page” 클릭
- “내 프로필” 탭 선택
- 이름, 연락처 등 수정
- “저장” 버튼 클릭

10-2. 플랫폼 이용 현황 확인 및 관리하기

- “서비스 이용 현황” 탭 선택
- 현재 이용 상태 확인:
 - 이용 시작일
 - 다음 결제일
 - 월 이용료: ₩50,000/월 (VAT 별도, 첫 3개월 무료)

플랫폼 이용료 무료 기간:

항목	내용
무료 기간	개설비 입금일로부터 3개월
적용 대상	플랫폼 이용료 (₩50,000/월, 부가세 별도)
시작 시점	개설비 입금 확인일
종료 후	4개월차부터 토스페이먼츠 카드 자동결제

예시: 2025년 1월 15일 입금 → 2025년 4월 14일까지 무료 → 4월 15일부터 월 ₩50,000 자동결제

결제 방식 안내:

항목	결제 방식
빌더 계정 개설비	무통장 입금

항목	결제 방식
크레딧 충전	무통장 입금
플랫폼 이용료	토스페이먼츠 카드 자동결제

플랫폼 이용료 결제 수단 등록: 1. 첫 3개월 무료 기간 종료 전에 카드 등록 2. 토스페이먼츠 결제창에서 카드 정보 입력

자동결제: - 첫 3개월 무료 이후 매월 동일 날짜 자동결제 - 결제 3일 전 알림 발송 - 결제 실패 시 2회 재시도 후 일시정지

이용 일시정지: 1. “이용 일시정지” 버튼 클릭 2. 일시정지 기간 동안 이용료 미 청구 3. 제한 사항: - ❌ 새 프로젝트 등록 불가 - ❌ Order Sheet 작성 불가 - 기존 프로젝트 조회 가능 - Books, FAQ 이용 가능

이용 해지: 1. “이용 해지” 버튼 클릭 2. 해지 시: - 개설비는 환불 불가 - 이번 달 이용료 일할 계산 환불 - ⚠️ 재가입 시 개설비 재납부 필요

10-3. 크레딧 관리하기

1. “크레딧 관리” 탭 선택
2. 현재 잔액 확인
3. “사용 내역” 클릭하여 상세 확인

충전 금액: - ₩10,000 - ₩30,000 - ₩50,000 - ₩100,000 - ₩200,000 - ₩500,000

충전 방법: 1. “충전하기” 버튼 클릭 2. 충전 금액 선택 3. 토스페이먼츠 결제창에서 결제 4. 결제 완료 후 즉시 충전

10-4. 결제 수단 관리하기

1. “결제 수단 관리” 탭 선택
2. 등록된 카드 목록 확인

3. “**카드 추가**”로 새 카드 등록
4. 기존 카드 삭제 시 해당 카드의 “**삭제**” 버튼 클릭

이 매뉴얼은 SSAL Works 빌더 계정의 10가지 핵심 기능을 안내합니다. 추가 질문은 “*Sunny에게 묻기*” 기능을 이용해주세요.

최종 업데이트: 2026-01-06 (섹션 순서 재배치, 금액 코드 수정)