**Prompt 모음**

**【요구 사항 명세서 추출】**

**ChatGPT**

**### 역할 ###**

너는 RFP 내역을 분석하고, RPF에서 요구 사항을 도출하는 데 있어 전문가야. 특히 Tech 영역에서 분석에 탁월해

**### Objective ###**

나는 너에게 RFP 문서를 텍스트 형태로 제공할 거야. ‘ABC Campus Agent 개발’에 관한 내용이야. 개발 센터에서 효율적인 개발을 위해 수치나 자료를 제공하는 시스템이야.

내가 덧붙여줄 RFP 내용을 읽고, 구현 필요한 요구 사항 리스트를 도출해 줘.

필요하다면 일부 요구 사항은 네가 선택적으로 제외할 수 있어. 반대로 필요하다면 기술되지 않았더라도 필수적인 요구 사항을 추가해 줘도 괜찮아.

**### 프로세스 ###**

다음 프로세스에 따라 요구 사항을 도출해 줘.

1. 고객의 요구 사항을 정확히 이해하고, 핵심 요소를 추출하기.

2. 기능적 요구 사항과 비기능적 요구 사항으로 구분하기.

3. 요구 사항을 논리적으로 분해하고 세분화하기.

4. 각 요구 사항에 대해 명확하고 측정 가능한 설명을 작성하기.

5. 정해진 포맷으로 결과를 출력하기.

**### 출력 포맷 ###**

**### 출력 결과 예시 ###**

[

{

"요구사항ID": "REQ-020101",

"유형": "기능",

"요구사항명": "수능 데이터 업로드 기능",

"분류": "원서 접수",

"설명": [

"타행 송금 내역을 실시간 확인 가능해야 함",

"거래 알림 기능이 있어야 함"

],

"중요도": "중",

"근거": "제안요청서\_108p[2]"

}

]

**### 주의 사항 ###**

* “적절하게”, “빠르게” 같은 모호한 표현은 금지
* 구현 방법이 아닌 요구사항 수준으로 기술할
* 필드 누락 없이 모든 출력 항목을 포함해야 함
* 비즈니스 용어는 고객 문서에 기반해야 함

**Claude**

### 역할 ###

당신은 RFP(요청제안서) 분석 전문가입니다. 특히 기술(Tech) 영역의 RFP에서 구체적이고 명확한 요구사항을 도출하는 데 탁월한 능력을 갖추고 있습니다.

### Objective ###

제공된 'ABC Campus Agent 개발' RFP 문서를 분석하여 개발팀이 구현해야 할 구체적인 요구사항 리스트를 도출합니다. 이 시스템은 개발 센터에서 효율적인 개발을 위해 수치나 자료를 제공하는 것이 목적입니다.

모든 중요 요구사항을 누락 없이 포함하되, 모호하거나 중복된 요구사항은 정제하십시오. RFP에 명시되지 않았더라도 시스템 구현에 필수적인 요구사항은 추가해도 좋습니다.

### 프로세스 ###

다음 5단계 프로세스를 따라 요구사항을 체계적으로 도출하십시오:

1. \*\*요구사항 파악\*\*: 고객의 명시적/암묵적 요구사항을 전체적으로 파악

2. \*\*분류\*\*: 각 요구사항을 다음 유형으로 분류 - 기능적 요구사항: 시스템이 수행해야 할 기능 - 비기능적 요구사항: 성능, 보안, 확장성 등 품질 속성 - 인터페이스 요구사항: 외부 시스템과의 연동 - 데이터 요구사항: 데이터 처리 및 저장 관련

3. \*\*세분화\*\*: 복합적인 요구사항을 논리적으로 분해하여 단일 기능 단위로 세분화

4. \*\*명세화\*\*: 각 요구사항을 측정 가능하고 검증 가능한 형태로 기술

5. \*\*문서화\*\*: 정해진 형식에 맞게 결과 정리

### 요구 사항 ID 체계 ###

요구사항 ID는 다음 체계를 따릅니다:

- REQ-[대분류][중분류][일련번호]

- 대분류: 01(사용자 인터페이스), 02(기능), 03(데이터), 04(인터페이스), 05(비기능)

- 중분류: 01~99 (기능 그룹별 번호)

- 일련번호: 01~99 (각 요구사항별 고유번호)

### 출력 결과 예시 ###

요구사항 목록은 다음 JSON 형식으로 출력하십시오:

```json

[

{

"요구사항ID": "REQ-020101",

"유형": "기능|비기능|인터페이스|데이터",

"요구사항명": "명확하고 간결한 제목",

"분류": "원서 접수",

"설명": [

"구체적이고 측정 가능한 설명 1",

"구체적이고 측정 가능한 설명 2"

],

"우선순위": "상|중|하",

"근거": "제안요청서\_페이지번호[항목번호]"

}

]

### 품질 기준 ### 모든 요구사항은 다음 SMART 기준을 만족해야 합니다:

• Specific(구체적): 모호한 표현이 아닌 명확한 내용으로 기술

• Measurable(측정 가능): "적절하게", "빠르게" 같은 주관적 표현 대신 정량적 기준 제시

• Achievable(달성 가능): 현실적으로 구현 가능한 수준

• Relevant(관련성): 시스템 목적과 직접 연관된 내용

• Time-bound(시간 제약): 필요시 시간적 제약 명시

### 주의 사항 ###

• 구현 방법이 아닌 '무엇을' 해야 하는지 요구사항 수준으로 기술하십시오

• 모든 필드는 누락 없이 완전히 작성하십시오

• 비즈니스 용어는 고객 문서에 기반해야 합니다

• 요구사항 간 중복이나 모순이 없도록 검토하십시오

• 필수적인 요구사항이 RFP에 명시되지 않았다면 '추가 권장 요구사항' 섹션을 별도로 만들어 제안하십시오

**【프로그램 설계서 추출】**

**ChatGPT**

다음은 학습관리시스템 구축 프로젝트의 요구사항 명세 JSON 파일이다. 이 파일에는 각 기능에 대한 요구사항 ID, 기능명, 설명, 중요도, 근거 등이 포함되어 있다. 이 정보를 기반으로 다음과 같은 형식의 프로그램 설계서를 작성하라:

- 기능명 (요구사항명)

- 요구사항 ID

- 개발 목적

- 기능 설명 (간단 요약)

- 입력 데이터 및 유효성

- 출력 데이터 및 포맷

- 처리 로직 요약 (Flow 또는 Pseudocode)

- 관련 연계 시스템 또는 API

- 중요 고려사항 (보안, 성능 등)

- 예상 테이블 및 속성 (있을 경우만)

모든 요구사항을 개별적으로 분석하여 위 형식에 맞춘 기능 단위 프로그램 설계서를 작성하라.

**Claude**

I'll improve this prompt to be more effective for extracting program design documentation from JSON requirement specifications. Here's an enhanced version:

요구사항 기반 프로그램 설계서 자동 생성 프롬프트

다음의 학습관리시스템(LMS) 구축 프로젝트 요구사항 명세 JSON 파일을 분석하여, 개발자가 바로 활용할 수 있는 상세 프로그램 설계서를 작성해 주세요. 요구사항별로 독립적인 설계 문서를 생성하되, 서로 연관된 기능들 간의 관계성과 의존성을 명확히 표현해 주세요.

문서 구조

각 요구사항에 대한 프로그램 설계서는 다음 구조를 따라 작성하세요:

1. 기본 정보

기능명: 요구사항에 명시된 기능명

요구사항 ID: 요구사항 식별자

중요도: 요구사항의 우선순위 수준

개발 목적: 해당 기능이 시스템에서 담당하는 역할과 존재 이유

2. 기능 상세

기능 개요: 기능에 대한 간결한 요약 (3-5줄)

상세 설명: 기능의 세부 동작 방식

사용자 시나리오: 주요 사용 사례 및 흐름

3. 기술적 명세

입력 데이터:

필수 입력 파라미터와 타입

데이터 유효성 검증 규칙

예외 처리 방안

출력 데이터:

반환 데이터 구조 및 포맷

성공/실패 응답 패턴

처리 로직:

단계별 처리 과정 (순서도 또는 의사코드)

주요 알고리즘 설명

트랜잭션 처리 방식

4. 시스템 연계

의존 모듈: 해당 기능이 의존하는 내부 모듈

외부 연계: 외부 API, 서비스, 시스템과의 통합 방식

데이터 모델:

관련 데이터베이스 테이블 및 스키마

엔티티 관계도(ERD) 관점에서의 위치

5. 비기능적 요구사항

보안 요구사항: 인증, 권한, 데이터 보호 방안

성능 요구사항: 응답시간, 처리량, 확장성 고려사항

품질 요구사항: 가용성, 안정성, 내결함성 전략

6. 구현 및 테스트 가이드

개발 참고사항: 구현 시 특별히 고려해야 할 사항

테스트 시나리오: 주요 테스트 케이스 및 검증 포인트

예상 이슈: 발생 가능한 문제점과 대응 방안

출력 형식

각 설계 문서는 마크다운 형식으로 작성하여 가독성을 높여주세요

복잡한 로직은 다이어그램이나 표를 활용하여 시각화해주세요

연관된 요구사항이 있는 경우 상호 참조할 수 있도록 링크를 제공해주세요

분석 및 개선 제안

요구사항에서 발견되는 모호성, 불일치, 또는 개선이 필요한 부분이 있다면 별도로 표시하고 개선 방향을 제안해주세요.

결과물은 실제 개발팀이 즉시 참고하여 개발을 시작할 수 있을 정도로 구체적이고 명확해야 합니다.

**【ERD 추출】**

**ChatGPT**

다음은 학습관리시스템 구축 프로젝트의 요구사항 명세이다. 이 요구사항들을 분석해서 전체 시스템에서 사용될 수 있는 데이터베이스 테이블을 최대한 도출해줘.

- 테이블명은 명확하고 의미 있게 작성해줘 (예: users, courses, enrollments, certificates 등)

- 각 테이블은 다음 형식으로 작성해줘:

Table 테이블명 {

필드명 자료형 [속성]

...

}

- 각 필드는 가능한 다음 속성을 가질 수 있음: [primary key], [foreign key], [unique], [not null]

- 데이터 타입은 일반적인 SQL 기준 (integer, varchar, text, boolean, timestamp 등)으로 해줘

- 가능한 경우 테이블 간 관계를 반영해줘 (예: user\_id는 다른 테이블의 foreign key로 표시)

최대한 많은 테이블을 도출하고, 중복되거나 불필요한 테이블은 제외해줘. 테이블 수에 제한 없이 모두 도출해줘.

✅ 출력 예시 (AI가 생성해야 할 형태)

Table users {

id integer [primary key]

username varchar

password varchar

role varchar

created\_at timestamp

}

Table courses {

id integer [primary key]

title varchar

category varchar

start\_date date

end\_date date

}

Table enrollments {

id integer [primary key]

user\_id integer [foreign key]

course\_id integer [foreign key]

status varchar

enrolled\_at timestamp

}

✨ 팁: 이런 요구도 추가 가능해

"테이블 간 관계도 함께 설명해줘." → ERD 다이어그램에 도움이 됨

"외래키 관계는 필드 옆에 명시해줘 (ex. user\_id → users.id)"

"필요 시 중간 테이블(join table)도 생성해줘 (예: course\_instructors)"

**Claude**

이 ERD 추출 프롬프트를 더 효과적으로 개선해 보겠습니다. 여기 향상된 버전입니다:

요구사항 기반 ERD 최적화 프롬프트

다음은 학습관리시스템(LMS) 구축 프로젝트의 요구사항 명세서입니다. 이 요구사항을 철저히 분석하여 데이터베이스 설계에 필요한 모든 테이블과 관계를 도출해주세요. 최종 산출물은 데이터베이스 설계자와 개발자가 즉시 구현에 활용할 수 있는 완전한 ERD(Entity-Relationship Diagram) 명세가 되어야 합니다.

출력 요구사항:

1. 테이블 정의

테이블명은 영문 소문자로 작성하고 복수형 명사 사용 (예: users, courses, assessments)

일관된 네이밍 컨벤션 적용 (snake\_case 권장)

각 테이블은 다음 형식으로 정의:

Table 테이블명 {

필드명 데이터타입 [제약조건]

...

}

2. 필드 정의

모든 테이블에 id integer [primary key, increment] 포함

생성/수정 시간 추적을 위한 created\_at, updated\_at timestamp 필드 포함

삭제 플래그(is\_deleted 등) 또는 deleted\_at 필드 추가 (소프트 삭제 지원)

적절한 인덱스 필드 제안 (검색 최적화를 위해)

3. 데이터 타입 표준화

문자열: varchar(길이), text

숫자: integer, decimal(전체자릿수,소수점자릿수), bigint

날짜/시간: date, timestamp, time

논리값: boolean

열거형: enum('값1', '값2', ...)

JSON 데이터: jsonb 또는 json

4. 제약조건 명시

[foreign key]: 외래키 참조 (예: user\_id integer [ref: > users.id])

[not null]: 필수 값

[unique]: 고유 값

[increment]: 자동 증가

[default: 값]: 기본값 설정

[note: '설명']: 필드에 대한 설명

5. 관계 정의

관계 방향 표시:

일대다(1): [ref: > 참조테이블.필드]

다대일(N:1): [ref: < 참조테이블.필드]

일대일(1:1): [ref: - 참조테이블.필드]

다대다(N): 중간 테이블 생성 필수

6. 데이터 무결성 고려사항

CASCADE, SET NULL, SET DEFAULT 등의 참조 무결성 액션 지정

복합 유니크 키 적용 (필요한 경우)

CHECK 제약조건 추가 (값의 범위나 형식 제한)

7. 부가 설명

각 테이블의 목적과 사용 사례 간략히 설명

테이블 간 중요 관계에 대한 설명 추가

특별한 비즈니스 룰이나 제약사항 명시

완성도 확인 체크리스트:

모든 핵심 엔티티가 테이블로 표현되었는가?

다대다(N) 관계가 적절한 중간 테이블로 분해되었는가?

테이블 간 관계가 명확히 정의되었는가?

정규화 원칙이 준수되었는가? (3NF 수준 권장)

확장성을 고려한 설계인가?

데이터 조회 성능을 위한 인덱스가 고려되었는가?

권장 추가 설명:

정규화/비정규화 결정에 대한 근거

필요한 인덱스 전략

성능 최적화를 위한 추가 고려사항

대용량 데이터 처리를 위한 파티셔닝 전략 (필요시)

시각적인 ERD 다이어그램을 제공할 수 없다면, 테이블 간 관계를 설명하는 섹션을 별도로 추가해주세요.

**【AI를 위한 UI Prompt 도출】**

**ChatGPT**

다음은 시스템 기능 요구사항이다. 이 기능을 사용자에게 제공할 수 있도록, 직관적이고 반응형 웹 기반 UI를 디자인하기 위한 AI Prompt를 생성해줘.

Prompt는 다음 조건을 충족해야 함:

- \*\*화면 목적\*\*을 명확하게 설명 (ex. 로그인 화면, 교육 과정 신청 화면 등)

- \*\*필수 UI 요소\*\*를 포함 (버튼, 입력창, 테이블, 알림 등)

- \*\*사용자의 흐름과 행동\*\*을 중심으로 UI 구조를 설계

- \*\*모바일 및 데스크탑 반응형 디자인 고려\*\*

- \*\*가능하다면 Tailwind CSS 또는 Figma에서 활용 가능한 요소 중심으로 설명\*\*

출력은 다음과 같은 형식을 따르도록 해줘:

🔹 기능명: 교육과정 신청

🔹 화면 목적: 사용자가 교육과정을 신청하고 신청 상태를 확인할 수 있는 화면

🔹 AI Prompt:

디자인해줘: "교육과정 신청 화면" 기능: 사용자는 교육과정을 선택하고 신청 버튼을 클릭할 수 있다. 신청 후 상태 확인 및 취소가 가능하다. 요소:

교육과정 목록 카드 (과정명, 기간, 신청상태, 신청 버튼 포함)

신청 완료 알림

마이페이지로 이동 버튼 스타일: Tailwind CSS 기반, 모바일 우선 디자인, 직관적인 레이아웃

✅ 한 줄 프롬프트 버전 (간결 요약형)

요구사항을 기반으로 각 기능 화면을 디자인할 수 있도록, 기능 목적과 필수 요소를 포함한 Tailwind/Figma용 AI 디자인 프롬프트를 만들어줘.

🧠 AI Prompt 생성 예시

요구사항: "REQ-030006 교육과정 신청 관리 기능"

➡️ AI Prompt 예시:

디자인해줘: "교육과정 신청 관리 화면"

기능: 사용자는 마이페이지를 통해 신청한 교육과정을 조회하고, 신청 상태를 확인하거나 취소할 수 있다.

요소:

- 교육 신청 내역 테이블 (과정명, 신청일, 상태, 취소 버튼)

- 신청 상태 필터 (전체/신청 완료/대기/취소)

- 상단 대시보드 위젯 (총 신청 건수, 진행 중 과정 등)

스타일: Tailwind CSS 기반, 데스크탑/모바일 반응형, 깔끔하고 행정 포털 스타일

**Claude**

템플릿 구조

🔹 기능명: [기능 이름]

🔹 화면 목적: [화면의 주요 목적 및 해결하려는 사용자 문제]

🔹 사용자 페르소나: [주요 대상 사용자 유형과 특성]

🔹 AI Prompt:

디자인해줘: "[화면 이름]"

목적: [화면이 해결하려는 핵심 사용자 문제]

기능:

- [주요 기능 1]

- [주요 기능 2]

- [주요 기능 3]

필수 UI 요소:

- [요소 1]: [세부 설명 및 포함 정보]

- [요소 2]: [세부 설명 및 포함 정보]

- [요소 3]: [세부 설명 및 포함 정보]

사용자 흐름:

1. [사용자 행동 1]

2. [사용자 행동 2]

3. [사용자 행동 3]

접근성 고려사항:

- [접근성 요구사항 1]

- [접근성 요구사항 2]

반응형 디자인:

- 모바일: [모바일 화면에서의 레이아웃 특징]

- 태블릿: [태블릿 화면에서의 레이아웃 특징]

- 데스크탑: [데스크탑 화면에서의 레이아웃 특징]

디자인 스타일:

- 프레임워크: [Tailwind CSS, Bootstrap 등]

- 색상 테마: [브랜드 색상 또는 테마 설명]

- 아이콘/이미지: [사용할 아이콘 세트 또는 이미지 스타일]

예시: 교육과정 신청 화면

🔹 기능명: 교육과정 신청 및 관리

🔹 화면 목적: 사용자가 교육과정을 탐색하고, 신청하며, 신청한 과정을 관리할 수 있는 통합 화면

🔹 사용자 페르소나: 조직 내 교육을 신청하려는 직원, 30-45세, 기본적인 디지털 리터러시 보유

🔹 AI Prompt:

디자인해줘: "교육과정 신청 및 관리 화면"

목적: 사용자가 교육과정을 쉽게 찾고, 신청하며, 신청 내역을 효율적으로 관리할 수 있는 직관적인 인터페이스 제공

기능:

교육과정 카테고리별 탐색 및 검색

교육과정 상세 정보 확인 및 신청

신청한 교육과정 조회, 상태 확인 및 취소 관리

필수 UI 요소:

상단 검색바: 키워드, 카테고리, 기간별 필터링 기능 포함

교육과정 카드 그리드: 각 카드는 과정명, 기간, 장소, 신청 가능 인원, 태그(필수/선택 교육), 난이도 표시, 신청버튼 포함

신청현황 탭: 신청 상태별(대기중/승인/거절/취소) 필터링 가능한 테이블 형식

상태 표시 배지: 각 과정의 상태를 직관적으로 보여주는 색상 코드 배지(승인-녹색, 대기-노랑, 거절-빨강)

액션 버튼: 신청, 취소, 상세보기 등의 명확한 버튼

알림 토스트: 신청 완료, 취소 확인 등의 사용자 액션 피드백

사용자 흐름:

검색/필터를 통해 원하는 교육과정 탐색

과정 카드 클릭으로 상세 정보 확인

신청 버튼 클릭 및 확인 모달 응답

신청현황 탭에서 신청 상태 확인

필요시 신청 취소 또는 상세 정보 재확인

접근성 고려사항:

충분한 색상 대비로 모든 텍스트 가독성 확보

스크린 리더 호환을 위한 적절한 ARIA 라벨 포함

키보드 네비게이션 지원으로 마우스 없이도 모든 기능 사용 가능

반응형 디자인:

모바일: 단일 컬럼 카드 레이아웃, 하단 네비게이션 바로 탭 전환, 축소된 테이블 대신 카드 리스트 형식

태블릿: 2열 그리드 레이아웃, 측면 탭 네비게이션으로 화면 전환

데스크탑: 3-4열 그리드 카드 레이아웃, 동시에 볼 수 있는 신청현황 섹션, 확장된 필터 옵션

디자인 스타일:

프레임워크: Tailwind CSS 기반의 확장 가능한 컴포넌트

색상 테마: 주 색상은 브랜드 블루(#2563eb), 보조 색상으로 중립 그레이(#f3f4f6), 알림 색상으로 그린(#10b981), 옐로우(#f59e0b), 레드(#ef4444)

아이콘: 심플한 라인 아이콘 세트(Heroicons 또는 Lucide), 직관적인 의미 전달에 중점

한 줄 프롬프트 버전 (간결 요약형)

사용자가 교육과정을 탐색, 신청, 관리할 수 있는 반응형 화면을 Tailwind CSS로 디자인해줘. 과정 카드 그리드, 상태별 필터링, 신청/취소 액션 버튼, 모바일/데스크탑 최적화를 포함해야 함.

개선된 이유:

사용자 페르소나 추가: 타겟 사용자에 대한 정보를 포함해 AI가 더 맞춤형 디자인을 생성하도록 유도

사용자 흐름의 확장: 단계별 사용자 여정을 구체화하여 화면 간 이동과 상호작용을 명확히 함

접근성 고려사항 섹션: 포용적 디자인을 위한 접근성 요소를 명시적으로 포함

디자인 스타일 세분화: 색상 코드, 아이콘 스타일 등 구체적인 디자인 요소를 명시하여 일관된 UI 생성

반응형 디자인 상세화: 각 화면 크기별 구체적인 레이아웃 변화를 설명하여 모든 기기에서의 사용성 보장

이 템플릿을 활용하면 AI가 더 완성도 높고 실제 사용자 경험에 중점을 둔 UI 디자인을 생성할 수 있을 것입니다.

**【구현 계획】**

요구 사항 명세를 텍스트로 공유해 줄게. 프로젝트 요건 문서를 작성해 줘. 작성 양식과 샘플도 텍스트로 공유해 줄게.

**【테스트 시나리오, 케이스 추출】**

**ChatGPT**

### 역할 ###

너는 요구 사항 정의서를 기반으로 테스트 시점에 반드시 준비해야 할 테스트 시나리오와 테스트 케이스를 도출하는 테스트 분석 전문가야.

### 목적 ###

내가 제공할 요구사항 정의는 ‘ABC Campus Agent 개발’ 프로젝트의 일부이며, 수치, 통계, 데이터 분석 정보를 제공하는 웹 기반 시스템이다.

너는 요구사항 정의를 분석하여, 명확하고 현실적인 테스트 시나리오 및 테스트 케이스를 도출해야 한다.

### 절차 ###

다음 순서에 따라 작업을 수행해 줘:

1. 요구사항 텍스트를 정확히 분석하고, \*\*핵심 기능 요소를 추출\*\*한다.

2. 각 기능에 대해 \*\*테스트 시나리오를 도출\*\*한다.

3. 테스트 시나리오별로 \*\*1개 이상 테스트 케이스를 도출\*\*한다.

4. 테스트 시나리오와 테스트 케이스는 \*\*정해진 출력 포맷\*\*에 맞춰 JSON 배열로 출력한다.

### 출력 규칙 ###

- 모든 항목은 누락 없이 채워야 한다.

- “빠르게”, “적절하게”, “잘 동작해야 한다” 등 \*\*모호한 표현 금지\*\*

- 테스트 케이스는 \*\*현실적인 데이터/조건을 기반으로 구성\*\*

- \*\*구현 방법이 아닌, 요구사항 중심 기술\*\*

---

### 출력 포맷 예시 ###

[

{

"요구사항ID": "REQ-030003",

"요구사항명": "교육 기본정보 관리",

"테스트 시나리오 ID": "TS-001",

"테스트 시나리오명": "교육 유형 등록 기능 테스트",

"테스트 케이스 ID": "TC-001-01",

"테스트 케이스명": "교육 유형을 정상적으로 등록",

"사전 조건": "관리자 계정으로 로그인",

"테스트 데이터": {

"교육유형명": "전문강사양성과정",

"카테고리": "전문과정"

},

"예상 결과": "해당 교육유형이 시스템에 정상적으로 저장되고 목록에 표시된다."

}

]

---

### 참고사항 ###

- 출력은 최대한 \*\*기능 단위로 세분화\*\*

- 시스템의 \*\*사용자 역할(관리자/사용자)\*\* 구분이 있다면 테스트 조건에 반영

- 복합 시나리오는 단순 테스트 케이스로 분해하여 작성

- 케이스별로 입력값 / 사전 조건 / 결과를 명확하게 정의

---

**Claude**

기존 템플릿이 이미 잘 구성되어 있지만, 몇 가지 중요한 측면을 개선하여 더 효과적인 테스트 시나리오와 케이스 도출 프롬프트를 만들어 보겠습니다.

개선된 테스트 시나리오 및 케이스 추출 프롬프트

역할

너는 요구사항 정의서를 분석하여 체계적이고 포괄적인 테스트 시나리오와 테스트 케이스를 도출하는 소프트웨어 QA 및 테스트 전문가야. 너의 경험은 기능 테스트, 사용자 인터페이스 테스트, 성능 테스트, 보안 테스트 등 다양한 테스트 영역을 포함한다.

목적

내가 제공할 요구사항 정의는 'ABC Campus Agent 개발' 프로젝트의 일부로, 수치, 통계, 데이터 분석 정보를 제공하는 웹 기반 시스템이다.

너는 이 요구사항을 분석하여 실제 현장에서 적용 가능한 테스트 시나리오와 테스트 케이스를 도출해야 한다. 도출된 테스트 케이스는 개발 완료 후 품질 검증 과정에서 활용될 예정이다.

테스트 유형 분류

다음 테스트 유형을 고려하여 종합적인 테스트 계획을 수립한다:

기능 테스트 - 요구사항에 명시된 기능이 의도대로 작동하는지 검증

경계값 테스트 - 입력 범위의 경계 조건에서 시스템이 올바르게 작동하는지 검증

오류 처리 테스트 - 잘못된 입력이나 예외 상황에서 시스템이 적절히 대응하는지 검증

데이터 무결성 테스트 - 데이터 저장, 수정, 삭제 시 일관성과 정확성 검증

권한/보안 테스트 - 사용자 권한에 따른 기능 접근 제한이 올바르게 작동하는지 검증

UI/UX 테스트 - 사용자 인터페이스가 요구사항에 맞게 구현되었는지 검증

절차

다음 단계에 따라 체계적으로 테스트 케이스를 도출한다:

요구사항을 철저히 분석하여 테스트 가능한 기능 단위로 분류한다.

각 기능에 대해 정상 흐름(Happy Path) 시나리오를 먼저 도출한다.

이어서 예외 흐름(Exception Path) 시나리오를 도출한다.

각 시나리오에 대해 구체적이고 검증 가능한 테스트 케이스를 도출한다.

테스트 케이스에는 명확한 사전 조건, 테스트 단계, 기대 결과를 포함한다.

테스트 케이스는 \*\*추적 가능성(Traceability)\*\*을 갖도록 요구사항 ID와 연결한다.

출력 규칙

모든 필드는 누락 없이 상세하게 채워야 한다.

"빠르게", "원활하게", "적절하게" 등 모호한 표현은 사용하지 않는다.

테스트 케이스의 기대 결과는 구체적이고 측정 가능한 형태로 작성한다.

테스트 데이터는 실제와 유사한 예시값을 사용한다.

\*\*테스트 단계(Steps)\*\*는 순차적이고 구체적으로 작성한다.

각 테스트 케이스는 단일 검증 포인트에 집중한다.

출력 포맷

json[

{

"요구사항ID": "REQ-XXXXX",

"요구사항명": "기능명",

"테스트 시나리오 ID": "TS-XXX",

"테스트 시나리오명": "시나리오 설명",

"테스트 케이스 ID": "TC-XXX-XX",

"테스트 케이스명": "케이스 설명",

"테스트 유형": "기능/경계값/오류처리/성능/보안 중 선택",

"우선순위": "상/중/하",

"사전 조건": "테스트 실행 전 필요한 조건",

"테스트 데이터": {

"필드1": "값1",

"필드2": "값2"

},

"테스트 단계": [

"1. 첫 번째 단계",

"2. 두 번째 단계",

"3. 세 번째 단계"

],

"예상 결과": "구체적이고 검증 가능한 결과",

"실제 결과": "",

"상태": "미실행"

}

]

출력 포맷 예시

json[

{

"요구사항ID": "REQ-030003",

"요구사항명": "교육 기본정보 관리",

"테스트 시나리오 ID": "TS-001",

"테스트 시나리오명": "교육 유형 등록 기능 테스트",

"테스트 케이스 ID": "TC-001-01",

"테스트 케이스명": "교육 유형을 정상적으로 등록",

"테스트 유형": "기능",

"우선순위": "상",

"사전 조건": "1. 관리자 계정으로 로그인되어 있음\n2. 교육관리 > 기본정보 관리 메뉴에 접근함",

"테스트 데이터": {

"교육유형명": "전문강사양성과정",

"카테고리": "전문과정",

"설명": "전문 강사를 양성하기 위한 교육과정입니다",

"활성화 여부": "활성"

},

"테스트 단계": [

"1. 교육 유형 관리 화면에서 '신규 등록' 버튼을 클릭한다",

"2. 교육 유형명에 '전문강사양성과정'을 입력한다",

"3. 카테고리 드롭다운에서 '전문과정'을 선택한다",

"4. 설명 필드에 설명 텍스트를 입력한다",

"5. 활성화 여부를 '활성'으로 설정한다",

"6. '저장' 버튼을 클릭한다"

],

"예상 결과": "1. '교육 유형이 성공적으로 등록되었습니다' 메시지가 표시된다\n2. 교육 유형 목록에 새로 등록된 '전문강사양성과정'이 표시된다\n3. 등록된 데이터가 데이터베이스에 정확히 저장된다",

"실제 결과": "",

"상태": "미실행"

},

{

"요구사항ID": "REQ-030003",

"요구사항명": "교육 기본정보 관리",

"테스트 시나리오 ID": "TS-001",

"테스트 시나리오명": "교육 유형 등록 기능 테스트",

"테스트 케이스 ID": "TC-001-02",

"테스트 케이스명": "중복된 교육 유형명으로 등록 시도",

"테스트 유형": "오류처리",

"우선순위": "중",

"사전 조건": "1. 관리자 계정으로 로그인되어 있음\n2. 교육관리 > 기본정보 관리 메뉴에 접근함\n3. '전문강사양성과정' 교육 유형이 이미 등록되어 있음",

"테스트 데이터": {

"교육유형명": "전문강사양성과정",

"카테고리": "기본과정",

"설명": "새로운 설명",

"활성화 여부": "활성"

},

"테스트 단계": [

"1. 교육 유형 관리 화면에서 '신규 등록' 버튼을 클릭한다",

"2. 이미 존재하는 교육 유형명 '전문강사양성과정'을 입력한다",

"3. 카테고리 드롭다운에서 '기본과정'을 선택한다",

"4. 설명 필드에 새로운 설명을 입력한다",

"5. '저장' 버튼을 클릭한다"

],

"예상 결과": "1. '이미 존재하는 교육 유형명입니다' 오류 메시지가 표시된다\n2. 중복된 교육 유형은 저장되지 않는다\n3. 사용자는 등록 화면에 머무른다",

"실제 결과": "",

"상태": "미실행"

}

]

테스트 케이스 작성 가이드라인

완전성 - 요구사항의 모든 측면을 테스트해야 한다

독립성 - 각 테스트는 다른 테스트의 결과에 영향을 받지 않아야 한다

반복 가능성 - 동일한 조건에서 항상 동일한 결과가 나와야 한다

추적 가능성 - 각 테스트 케이스는 요구사항과 연결되어야 한다

명확성 - 테스트 내용과 예상 결과가 모호하지 않게 작성되어야 한다

효율성 - 최소한의 테스트로 최대한의 기능 검증을 해야 한다

우선순위 결정 기준

상: 핵심 기능, 데이터 무결성, 보안 관련 테스트

중: 일반 기능, UI/UX, 비즈니스 로직 테스트

하: 경미한 UI 요소, 선택적 기능 테스트

마지막 검토 사항

테스트 케이스 도출 완료 후 다음을 확인한다:

모든 요구사항이 하나 이상의 테스트 케이스로 검증되는가?

정상 흐름과 예외 흐름이 모두 포함되었는가?

테스트 데이터가 현실적이고 다양한 상황을 검증하는가?

테스트 단계와 예상 결과가 명확하게 작성되었는가?

테스트 간 중복이나 누락된 부분은 없는가?

이 개선된 프롬프트의 주요 장점은 다음과 같습니다:

테스트 유형 분류 추가: 다양한 테스트 유형을 고려하여 더 포괄적인 테스트 범위 확보

우선순위 필드 추가: 리소스 제약 시 중요 기능부터 테스트할 수 있도록 지원

단계별 테스트 절차 명시: 테스트 수행자가 정확히 따라할 수 있는 구체적인 단계 제공

테스트 결과 필드 추가: 실제 테스트 실행 결과를 기록할 수 있는 구조 제공

가이드라인과 검토 사항 제공: 테스트 케이스 품질 향상을 위한 체크리스트 제공

예외 흐름 강조: 정상 케이스뿐만 아니라 예외 상황에 대한 테스트도 중요하게 다룸

이 개선된 프롬프트를 사용하면 더 체계적이고 포괄적인 테스트 시나리오와 케이스를 도출할 수 있을 것입니다.