

M03-04 · GIBALJA VISUAL MANUAL

사용자 흐름과 화면 상태 설계하기

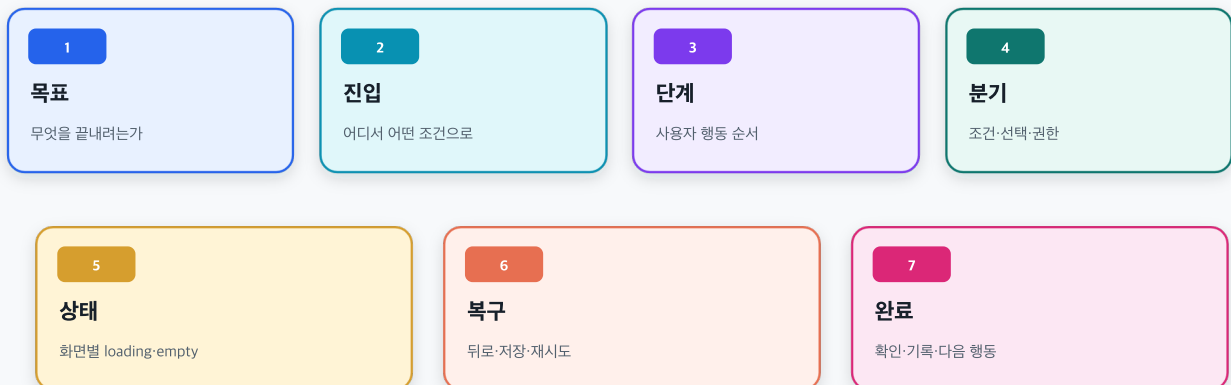
한 문장 목표: 사용자가 왜 들어와 어떤 화면·행동·결정·상태를 거쳐 목표를 끝내는지 그리고, 대상 아님·뒤로가기·저장·오류·만료·중복 제출에서도 막다른 길이 없는 화면 정의를 만듭니다.

난이도	개념	실습	셀프 테스트	최종 산출물
Level 1	60분	60분	15분	사용자 흐름도, 화면×상태 표, 화면 정의서

“신청 화면 5개가 필요합니다”는 화면 목록입니다. “준비 안내에서 시작해 자격 예는 정보 입력으로, 아니요는 대체 과정으로 이동하며, 자료가 없으면 30일 draft로 저장합니다. 최종 제출은 검토 뒤 한 번만 처리하고 접수 번호를 보여 줍니다”라고 써야 사용자 흐름이 됩니다.

사용자 흐름은 일곱 질문으로 설계한다

목표에서 시작해 진입·단계·분기·상태·복구를 거쳐 완료 증거까지 이어 붙인다



최종 흐름 = 사용자 목표 + 모든 도착 가능한 경로 + 막힘 없는 다음 행동

그림 1. 사용자 흐름은 목표에서 시작해 진입·단계·분기·상태·복구를 지나 완료 증거와 다음 행동으로 끝납니다.

1

목표

무엇을 끝내려는가

2

진입

어디서 어떤 조건으로

5

상태

화면별 loading·empty

6

복구

뒤로·저장·재시도

최종 흐름 = 사용자 목표 + 모든 도착

3

단계

사용자 행동 순서

4

분기

조건·선택·권한

7

완료

확인·기록·다음 행동

가능한 경로 + 막힘 없는 다음 행동

1. 이 PDF를 공부하는 방법

1회차 · 일곱 질문 익히기 · 15분

목표 → 진입 → 단계 → 분기 → 상태 → 복구 → 완료 를 소리 내어 읽습니다. 화면 이름부터 적지 말고 사용자가 얻을 현실 결과와 시작 신호를 먼저 말합니다.

2회차 · 지도와 화면 상태 분리하기 · 30분

사용자 여정·사용자 흐름·화면 흐름·서비스 청사진의 확대 수준을 구분합니다. 흐름도의 화면마다 loading·content·empty·error·blocked 상태 행을 붙입니다.

3회차 · 여섯 경로 실행하기 · 35분

사용자 흐름·화면 상태 스튜디오에서 정상 완료·대상 아님·저장 후 재진입·오류 후 재시도·세션 만료·중복 제출을 실행합니다.

4회차 · 실제 업무 화면 정의하기 · 40분

교육 신청·문의 접수·문서 승인처럼 시작과 끝이 분명한 업무를 고릅니다. AS-IS 증거를 붙이고 TO-BE 흐름과 화면×상태 표를 작성합니다.

5회차 · 셀프 테스트 · 15분

정답을 가리고 10문제를 푼다. 틀린 문제는 “화면만 봄”, “happy path만 봄”, “보존·복구 빠짐”, “완료 증거 빠짐” 중 원인을 표시합니다.

학습 완료 기준

“오류 화면을 추가해 주세요”를 “검토 화면에서 제출이 503으로 실패하면 입력 5개를 30일 보존하고 problem 상태로 이동합니다. 다시 시도는 보존된 검토 화면으로 돌아가며, 저장 후 나가는 draft ID·마지막 위치·만료일을 보여 줍니다”처럼 경로·상태·보존·복구로 설명할 수 있습니다.

2. 기능이 아니라 사용자의 전체 문제에서 시작합니다

GOV.UK Service Standard는 기술이나 미리 고른 해법이 아니라 사용자의 필요를 중심으로 전체 문제를 해결하고, 관련 서비스가 사용자에게 하나의 여정처럼 이어지게 하도록 안내합니다. 사용자는 조직도와 담당 팀 경계를 이해하기 위해 서비스를 쓰지 않습니다.

조직 관점: 안내팀 → 자격팀 → 접수팀 → 심사팀

사용자 관점: 지원할 수 있는지 알고 → 필요한 것을 내고 → 결과를 확인

2.1. 목표를 결과로 씁니다

약한 목표	더 좋은 목표
신청 페이지 사용	지원 신청을 접수하고 접수 증거를 얻음
대시보드 조회	미처리 문서를 찾아 승인 여부를 결정함
챗봇 대화	근거 있는 답을 얻고 필요한 후속 절차로 이동함

화면·버튼·API는 목표를 위한 수단입니다. 목표에는 사용자가 확인할 수 있는 완료 증거가 있어야 합니다.

2.2. 시작 신호와 완료 증거

사용자: 교육 지원이 필요한 직장인

시작 신호: 모집 공고를 보고 자격을 확인하려 함

목표: 신청을 접수하고 처리 일정과 접수 번호를 얻음

완료 증거: YC-260715-08, 2영업일 안내, PDF 기록

2.3. 범위를 너무 넓거나 좁게 잡지 않습니다

- 너무 좁음: “파일 업로드”만 정의해 사용자가 왜 무엇을 제출하는지 빠짐
- 너무 넓음: “취업 성공” 전체를 한 거래가 책임진다고 정의
- 적절함: “교육 지원 신청 접수와 결과 확인 시작”처럼 팀이 개선할 범위와 이어지는 다음 흐름을 구분

BIG IDEA 01

화면 흐름의 시작은 첫 화면이 아니라 사용자의 필요가 생긴 시점이고, 끝은 마지막 버튼이 아니라 사용자가 결과를 확인한 시점입니다.

30초 확인

회원 가입 완료 화면이 “완료”만 보여 줍니다. 사용자 목표의 완료 증거로 충분합니까?

정답 보기

부족할 수 있습니다. 계정 식별·확인 메일·다음 로그인 또는 시작 행동·문제 발생 시 문의처럼 사용자가 결과를 확인하고 다음 일을 할 정보가 필요합니다.

3. 지도 네 종류의 관점을 구분합니다

지도 네 종류는 확대 수준과 관점이 다르다

사용자의 전체 경험부터 화면 내부 상태와 조직의 뒷단 과정까지 목적에 맞는 지도를 고른다



그림 2. 같은 서비스를 지도화해도 시간 범위·관점·구현 상세가 다릅니다. 목적에 맞는 지도를 선택합니다.

사용자 여정

시간·채널·생각·감정

서비스 전후 전체 문제

구체

화면 흐름

화면·URL·이동·상태

구현할 화면 관계

연

사용자 흐름

목표·행동·결정·경로

제품 안팎의 과업 경로

서비스 청사진

접점·직원·정책·시스템

보이는 경험과 뒷단 연결

지도	중심 질문	포함 내용	대표 산출물
사용자 여정·경험 지도	사용자는 시간에 따라 무엇을 하고 느끼는가	단계·채널·생각·감정·문제	전체 경험 지도
사용자 흐름	한 목표를 어떤 행동·결정으로 끝내는가	진입·행동·분기·끝점	목표별 흐름도
화면 흐름	어떤 화면·URL·상태가 연결되는가	화면·route·이동·상태	구현 화면 지도
서비스 청사진	보이는 경험을 조직이 어떻게 지원하는가	접점·직원·정책·백엔드·기술	서비스 운영 지도

3.1. 경험 지도

GOV.UK Service Manual은 경험 지도를 사용자가 서비스를 필요로 하기 시작한 때부터 사용을 멈출 때까지 무엇을 하고 생각하고 느끼는지 시간에 따라 시각화하는 것으로 설명합니다. 여러 장소·팀·서비스 접점이 있는 긴 경험에 유용합니다.

3.2. 사용자 흐름과 화면 흐름

사용자 흐름은 목표와 선택의 논리, 화면 흐름은 구현할 화면·URL·상태 관계에 더 가깝습니다. 하나의 사용자 흐름 노드가 여러 화면 상태로 구현될 수 있고, 하나의 화면이 여러 목표 흐름에 재사용될 수도 있습니다.

3.3. 서비스 청사진

온라인 화면에서 “심사 중”이 보일 때 뒷단에서는 담당자 배정·서류 확인·외부 기관 조회가 진행될 수 있습니다. 사용자의 대기 메시지와 실제 운영 과정·처리 시간을 연결할 때 서비스 청사진이 필요합니다.

지도 이름은 조직과 방법론마다 조금 다르게 씁니다. 이 매뉴얼의 네 종류는 학습을 위한 구분이며 보편적으로 강제되는 표준 용어 체계가 아닙니다. 문서 첫머리에 관점·범위·표기 규칙을 적습니다.

4. AS-IS는 증거로, TO-BE는 가설로 그림니다

현재 흐름은 증거로 그리고 목표 흐름은 가설로 검증한다

기억에 의존한 예쁜 화살표보다 관찰·문의·데이터가 붙은 as-is와 시험 가능한 to-be가 강하다

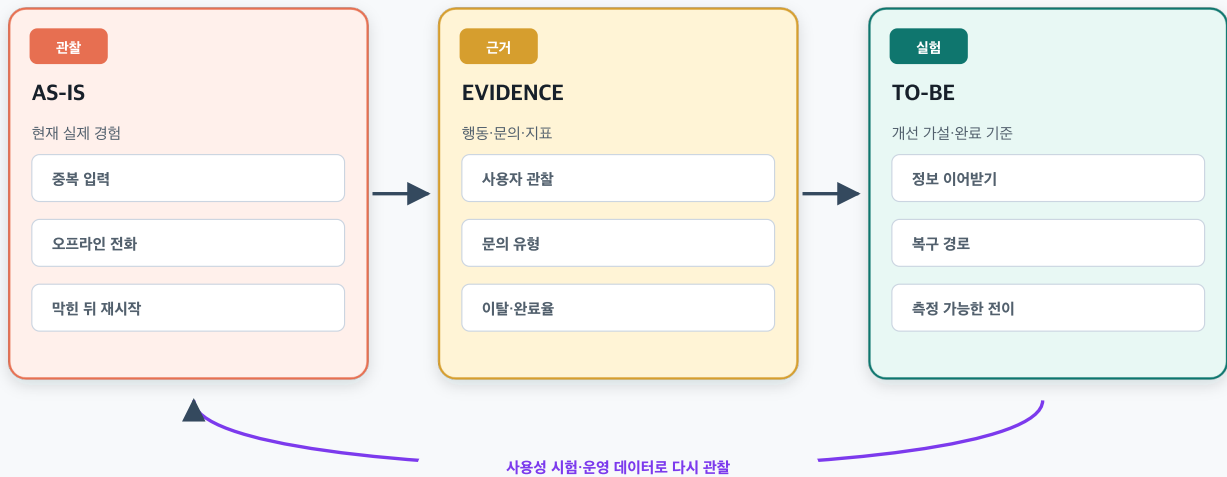


그림 3. 현재 흐름은 관찰 증거로 설명하고, 목표 흐름은 측정 가능한 전이와 완료 기준을 가진 가설로 시험합니다.





4.1. AS-IS 증거

증거	알 수 있는 것	주의
사용자 관찰·인터뷰	실제 순서·우회·생각·감정	말한 행동과 실제 행동 구분
문의·상담 기록	반복되는 막힘·문구 오해	문의하지 못한 사용자도 있음
분석 이벤트	단계별 진입·완료·이탈	사건 정의와 분모 확인
오류·운영 로그	실패 조건·처리 시간	개인·보안 정보 보호
현장 직원·업무 규칙	오프라인·뒷단 제약	조직 관점만으로 대체하지 않음

4.2. 예쁜 흐름과 사실 흐름

예쁜 흐름: 시작 → 입력 → 제출 → 완료

실제 흐름: 공고 발견 → 자격이 모호해 전화 → 서류를 찾으려 이탈
→ 모바일에서 다시 시작 → 같은 정보 재입력 → 제출 오류
→ 보존 여부를 몰라 처음부터 재작성 → 접수 번호 캡처

4.3. TO-BE는 완료 기준을 붙입니다

가설: 자료가 없을 때 저장 후 나가기를 제공하면 재시작 반복이 줄어든다.

전이: evidence → saved → re-entry → evidence

완료 기준: 30일 안 재진입 시 답변과 마지막 위치가 복원되고,
저장·완료·삭제 조건을 사용자가 확인한다.

측정: 저장 사용자 중 재진입률·완수율·중복 문의 변화

AS-IS와 TO-BE를 색만 다르게 두지 말고 증거 ID·가설·검증 방법으로 연결합니다.

5. 흐름도 문법을 일관되게 씁니다

흐름도 문법은 도형보다 읽는 규칙이 중요하다

시작·화면·행동·결정·외부 접점·종료를 일관되게 쓰고 모든 화살표에 사건이나 조건을 붙인다

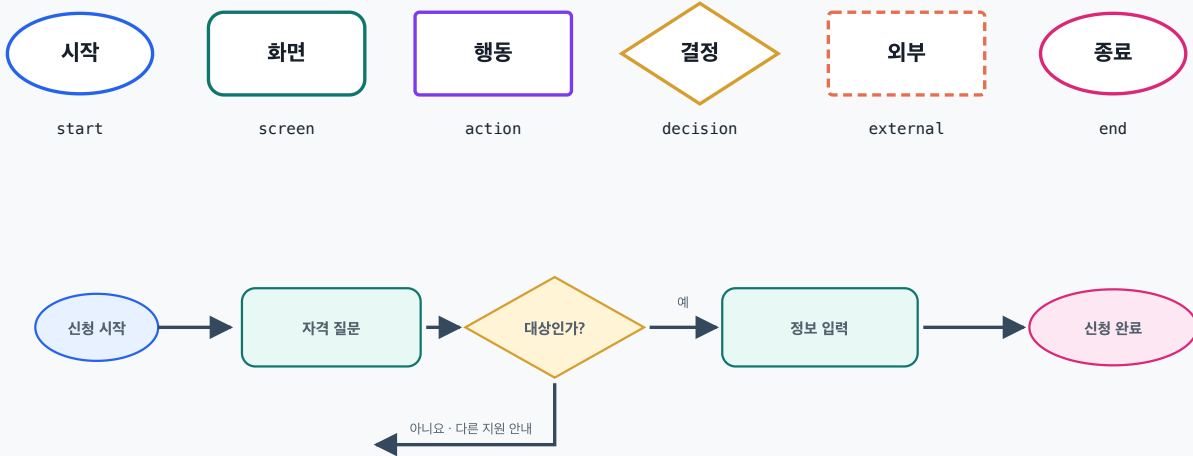


그림 4. 도형은 팀 안의 읽기 규칙입니다. 모든 화살표에 행동·사건·조건을 붙이고 결정의 결과를 빠짐없이 표시합니다.



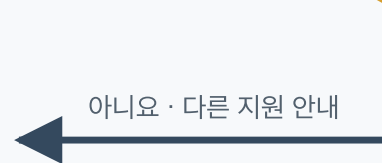
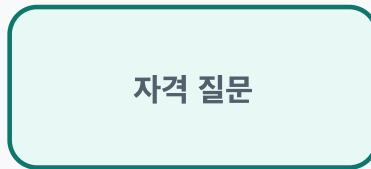
start



screen



action



아니요 · 다른 지원 안내



decision



external



end



5.1. 이 매뉴얼의 표기

모양	뜻	이름 예
타원	시작·종료	신청 시작·접수 완료
둥근 사각형	화면	자격 질문·검토 화면
사각형	행동·처리	자료 제출·답변 저장
마름모	결정	지원 대상인가?
점선 사각형	외부 접점	전화 상담·외부 결제

이 표기는 학습을 위한 팀 규칙입니다. UML이나 BPMN을 엄격히 쓰는 조직이면 해당 표준을 따릅니다.

5.2. 노드는 명사·행동은 동사

화면: 자격 확인
행동: “예”를 선택하고 계속
결정: `eligible = yes?`
전이: 예 → 신청자 정보 / 아니요 → 대체 과정 안내

5.3. 결정 노드의 결과를 닫습니다

“자격 확인” 마름모에 yes 경로만 있으면 no 사용자는 지도 밖으로 떨어집니다. yes·no·알 수 없음·조회 오류처럼 실제 가능한 결과를 확인합니다.

5.4. 화살표에 원인을 씁니다

화살표는 단순 화면 순서가 아니라 전이를 만든 사건입니다.

- 계속 클릭·`valid=true`
- `eligible=false`
- HTTP 503·답변 보존
- 30일 안 저장 링크 재진입
- 브라우저 뒤로

6. 진입·끝점·채널 경계를 먼저 정합니다

6.1. 진입점은 하나가 아닙니다

진입	필요한 확인
검색 결과·광고 링크	준비 조건과 대상이 보이는가
알림·이메일 깊은 링크	로그인·만료·권한이 맞는가
직접 URL·북마크	사전 상태가 없을 때 도움 되는가
다른 서비스	전달 정보와 책임 전환이 명확한가
전화·방문 뒤 온라인	상담 내용·참조 번호가 이어지는가

6.2. 끝점의 네 요소

1. 무슨 결과인지
2. 입력·답변·작업이 보존됐는지
3. 지금 할 수 있는 다음 행동
4. 필요하면 문의·대체 채널·다시 가능한 시점

완료·대상 아님·사용자 취소·오류·서비스 중단은 모두 끝점이 될 수 있습니다. “성공 종료”만 종료로 보지 않습니다.

6.3. 온라인과 오프라인을 연결합니다

GOV.UK 지침은 사용자 여정을 이해할 때 온라인·오프라인 접점, 뒷단 과정, 사용자가 제출해야 할 증거를 함께 보도록 안내합니다. 전화 번호만 붙이는 것으로 끝내지 않고 상담 시간·준비 정보·온라인으로 돌아오는 방법을 정의합니다.

30초 확인

세션 만료 화면에 “다시 로그인” 버튼만 있습니다. 충분합니까?

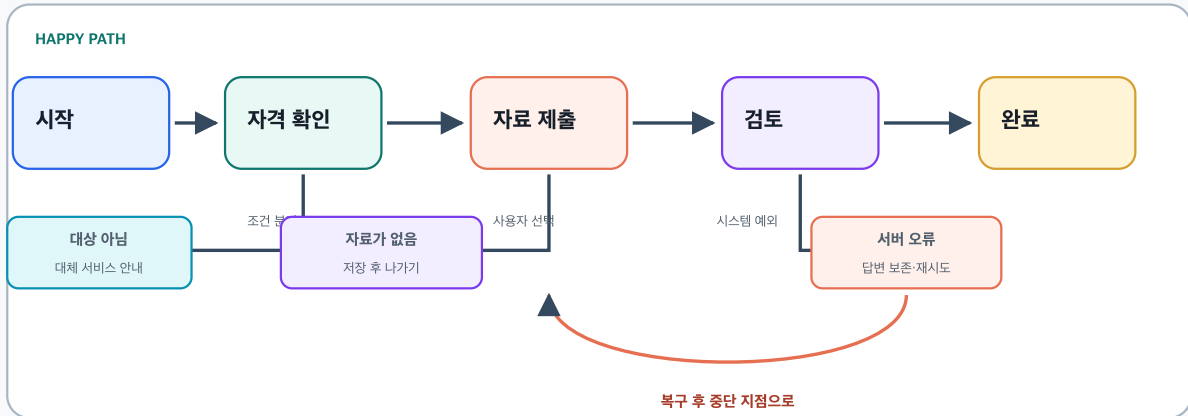
정답 보기

로그인 뒤 어디로 돌아오는지, 입력이 보존됐는지, 사라진 값이 무엇인지, 만료 원인과 재시작 방법을 알려야 합니다.

7. happy path보다 분기·예외·복구를 먼저 검수합니다

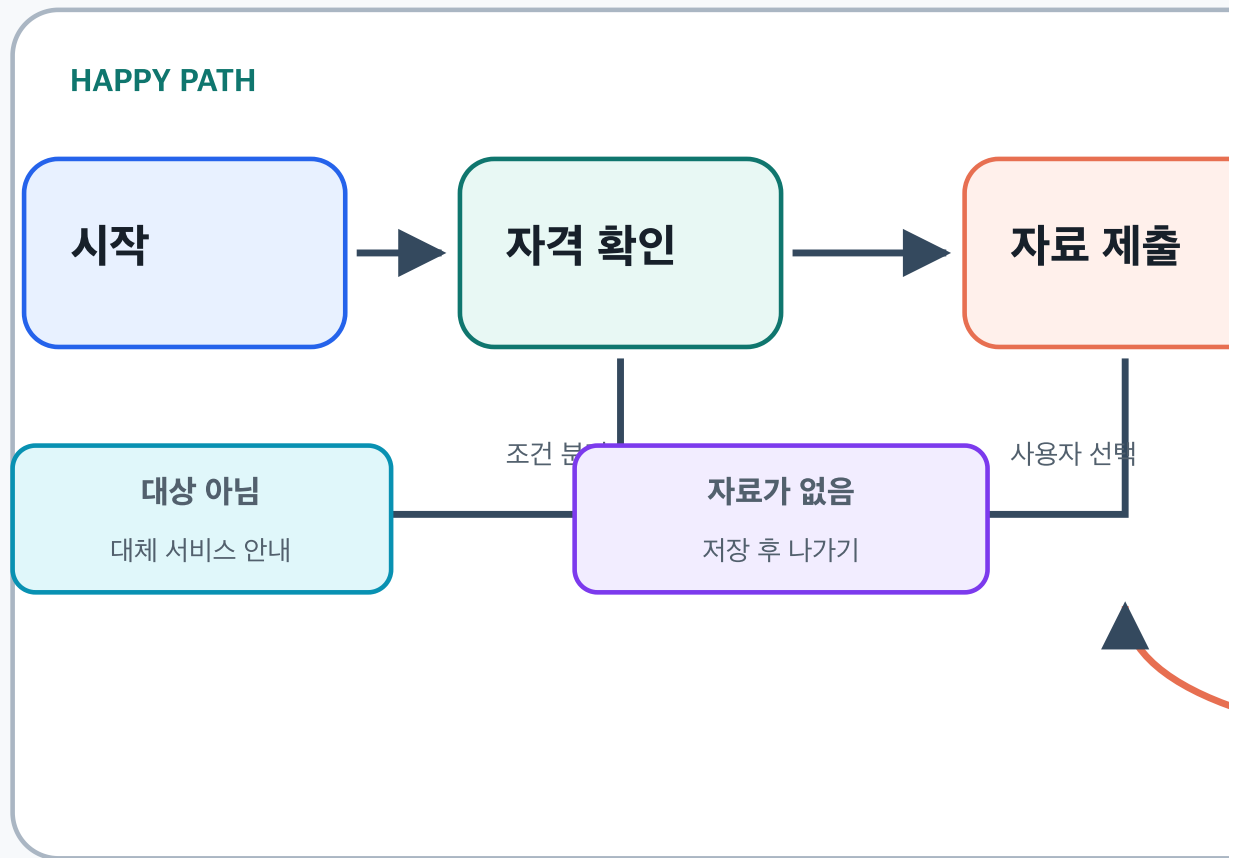
happy path보다 막힌 뒤 돌아오는 길이 더 중요하다

조건 분기·사용자 선택·시스템 오류를 구분하고 각 끝점에 다음 행동이나 복구 경로를 둔다

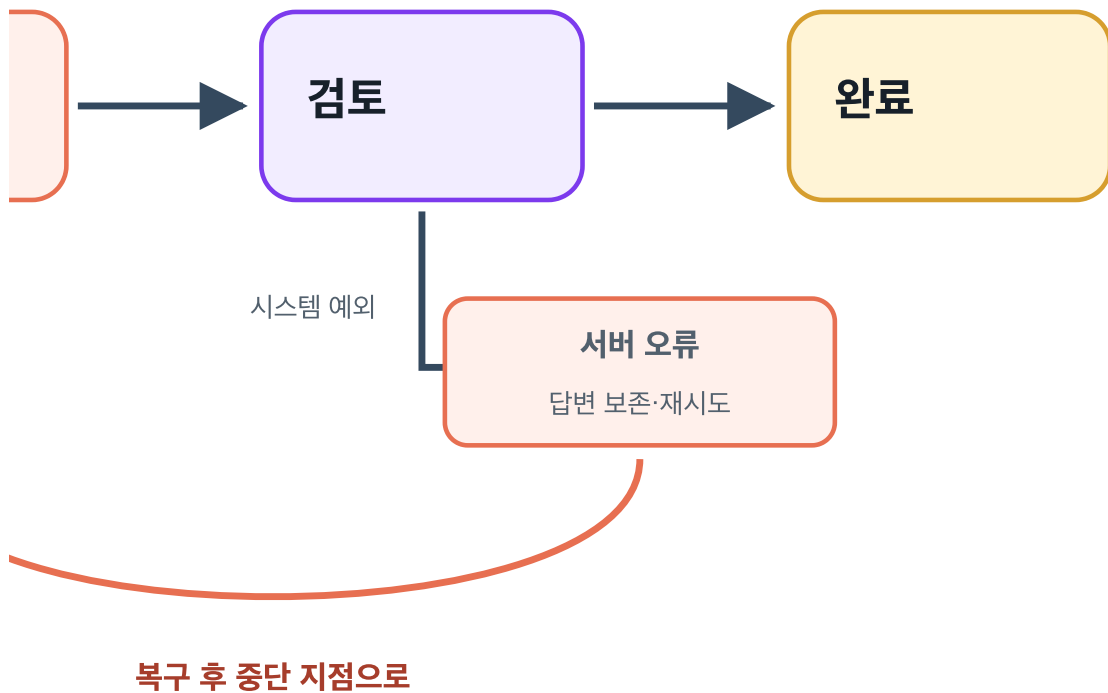


DEAD END = 다음 행동·보존 정보·문의·대체 경로가 없는 도착점

그림 5. 조건 분기·사용자 선택·시스템 예외는 다른 원인을 가지며, 각 끝점에 복구 또는 의미 있는 다음 행동이 필요합니다.



DEAD END = 다음 행동·보존 정



보·문의·대체 경로가 없는 도착점

7.1. 세 종류

종류	예	설계할 것
조건 분기	대상 아님·권한 없음	이유·대체 경로·재판정 조건
사용자 선택	자료가 없어 나중에 계속	저장 범위·재진입·만료
시스템 예외	503·네트워크 실패	답변 보존·재시도·대체 채널

7.2. dead end 검수

도착 화면: _____

결과 설명: 있음·없음

다음 행동: 있음·없음

보존 여부: 설명함·설명 안 함

복구 시점·경로: 있음·없음

문의·대체: 필요·불필요·있음·없음

7.3. 오류 문구는 사용자 과업으로 씁니다

GOV.UK의 service problem 패턴은 기술 용어 대신 답변에 무슨 일이 생겼는지, 나중에 다시 할 수 있는지, 다른 서비스·문의로 목표를 이어갈 수 있는지를 알려 주도록 안내합니다.

약함: 500 Bad Request

개선: 신청을 제출하지 못했습니다. 입력한 답변은 30일 보존됩니다.
잠시 뒤 다시 시도하거나 지원 센터에 문의하세요.

BIG IDEA 02

예외 화면의 품질은 오류를 예쁘게 설명하는 데 있지 않고, 사용자가 잃은 것과 남은 것과 다음 행동을 정확히 아는 데 있습니다.

8. 화면×상태 매트릭스로 누락을 찾습니다

화면 흐름도에 상태 행을 더하면 누락이 보인다

화면 하나당 정상 화면만 그리지 않고 초기·로딩·내용·빈 결과·오류·권한 상태를 교차 검증한다

	시작	자격 질문	자료 목록	검토	완료
초기	✓	✓			
loading		✓	✓	✓	
content		✓	✓	✓	✓
empty			✓		
error		✓	✓	✓	
권한			✓	✓	

빈 칸은 “상태가 필요 없음”인지 “정의가 빠짐”인지 근거를 남긴다

그림 6. 흐름도의 각 화면을 상태 열과 교차하면 happy path에서는 보이지 않던 empty·error·권한 상태가 드러납니다.

	시작	자격 질문	
초기			
loading			
content			
empty			
error			
권한			

자료 목록	검토	완료
		
		
		
		
		

8.1. 두 축

- 행: 시작·자격·자료·검토·완료 같은 화면·route
- 열: initial·loading·content·empty·error·blocked·saved·complete

8.2. 빈 칸을 해석합니다

완료 화면에 loading이 불필요할 수 있습니다. 자료 목록의 empty는 반드시 필요할 수 있습니다. 빈 칸마다 “해당 없음” 또는 “정의 누락”을 판정합니다.

8.3. 화면 정의의 최소 항목

항목	질문
목적	이 화면에서 사용자가 무엇을 판단·완료하는가
진입	어떤 사건·조건·이전 상태로 오는가
데이터	무엇을 보여 주고 어디서 오는가
행동	primary·secondary·back·cancel은 무엇인가
상태	loading·empty·error·blocked 문구와 행동은 무엇인가
전이	각 행동·조건이 어느 화면·상태로 가는가
접근성	제목·초점·상태 메시지·오류 연결은 무엇인가
분석	진입·완료·이탈을 어떤 사건으로 측정하는가

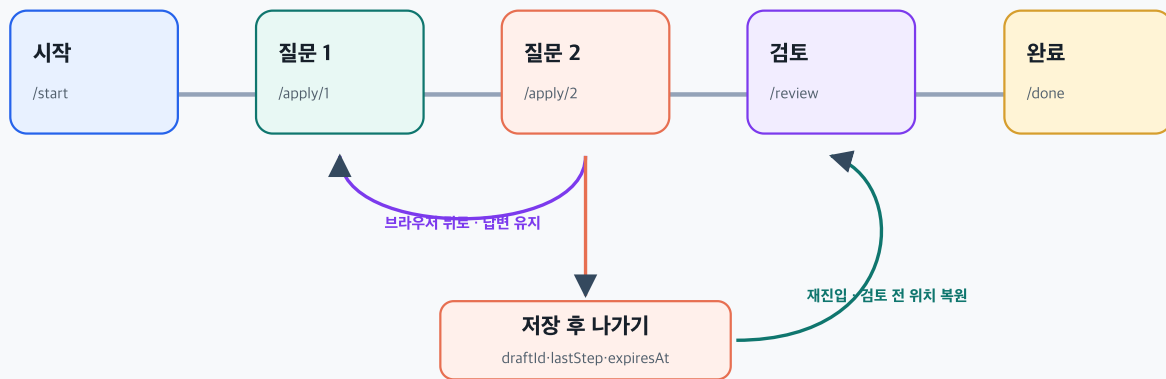
8.4. 화면과 상태를 이름에 섞지 않습니다

신청오류페이지2 보다 review 화면의 error 상태처럼 화면 정체성과 표시 상태를 분리하면 같은 기능의 상태 관계를 읽기 쉽습니다. 별도 전체 오류 페이지가 필요한 서비스 중단과 입력 오류는 구분합니다.

9. 뒤로가기·저장·재진입은 별도 흐름입니다

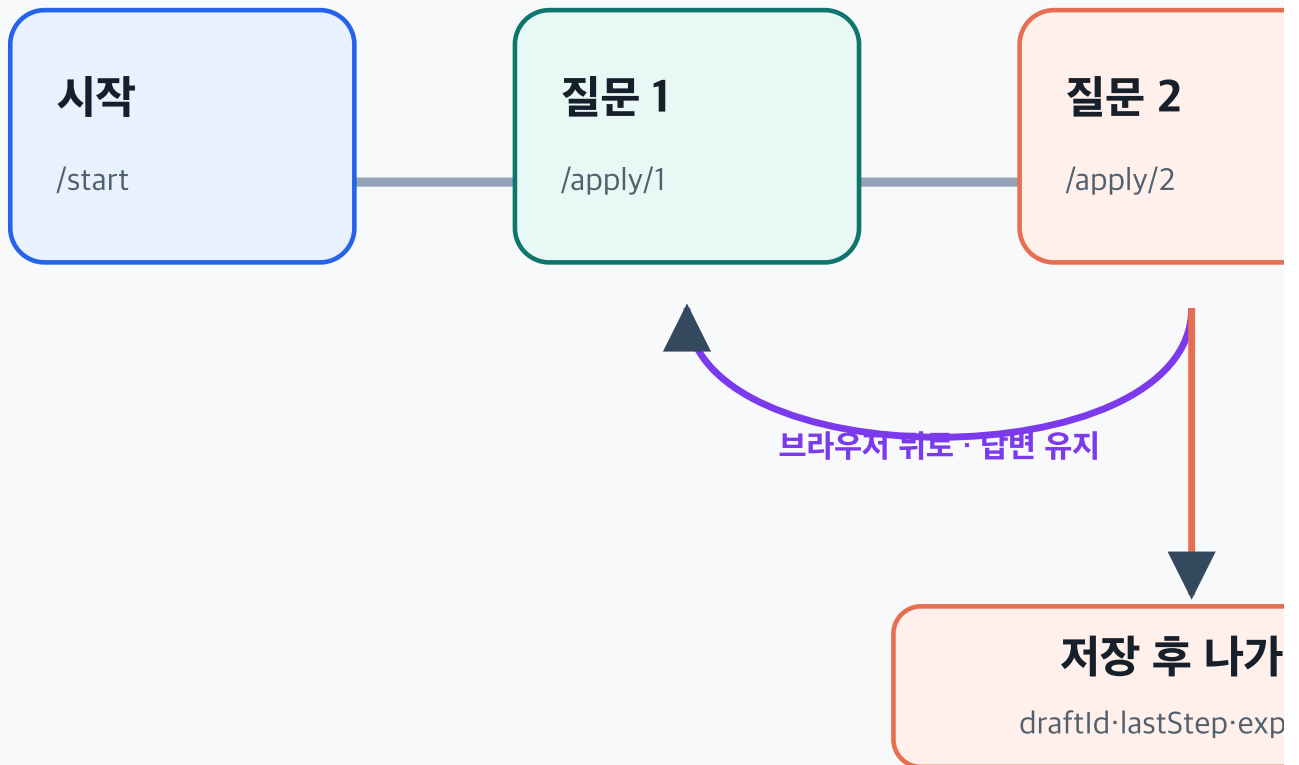
뒤로가기·저장·재진입은 별도의 사용자 흐름이다

브라우저 이력과 앱 상태를 함께 설계해 이전 답변·중단 위치·만료 조건을 예측 가능하게 만든다

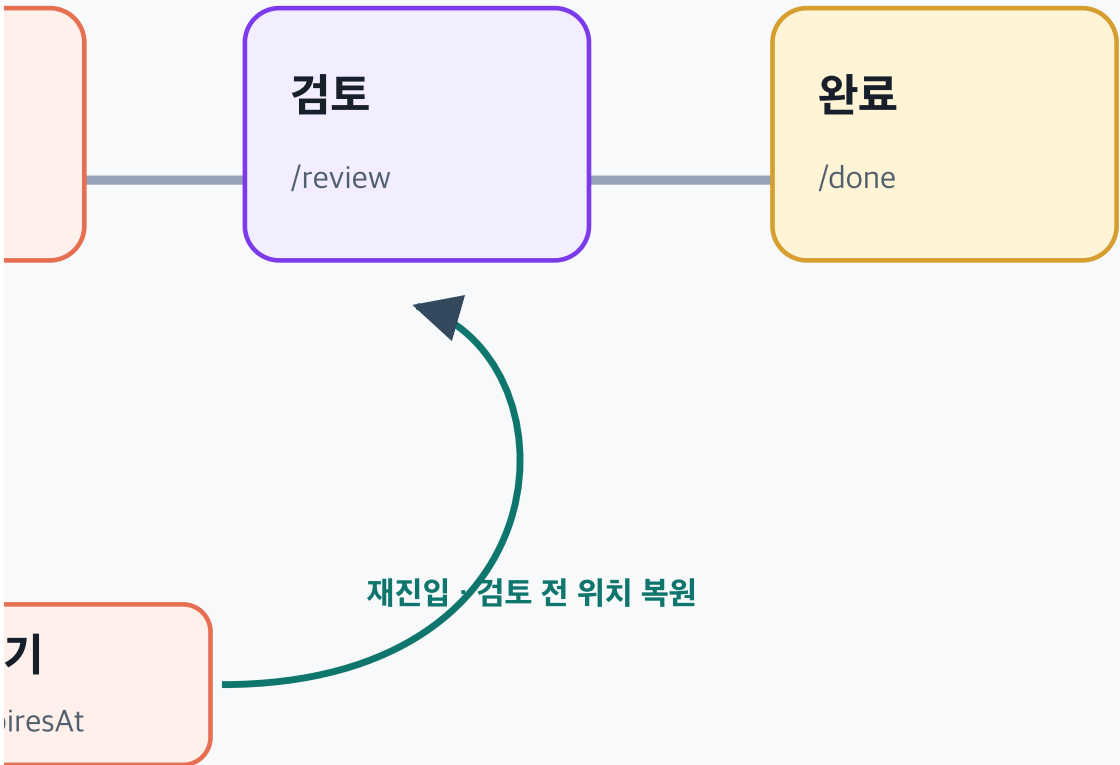


기록할 것 = URL·이력 동작 + 보존 필드 + 저장 위치 + 만료 시 메시지 + 재시작 경로

그림 7. URL·세션 이력과 앱 데이터 보존은 다른 층입니다. 둘을 함께 정의해야 이전 화면이 같은 상태로 복원됩니다.



기록할 것 = URL·이력 동작 + 보존 필드 +



저장 위치 + 만료 시 메시지 + 재시작 경로

9.1. 브라우저 이력

WHATWG HTML은 navigation과 session history를 통해 현재 문서·이력 항목·뒤로 이동을 정의합니다. 실제 구현 세부는 복잡하지만 화면 설계자는 다음을 확인해야 합니다.

행동	기대
앱 뒤로 링크	이전 질문과 답변 상태로 이동
브라우저 뒤로	사용자가 마지막으로 본 이전 상태 복원
앞으로	이미 방문한 다음 이력 항목 복원
새로고침	현재 URL에서 의미 있는 화면 복원 또는 안내
깊은 링크	사전 조건이 없으면 로그인·시작·만료 안내

GOV.UK question page 지침도 back link를 제공하되 브라우저 뒤로 버튼을 깨뜨리지 말고, 이전 페이지를 마지막으로 본 상태로 보여 주도록 권합니다.

9.2. 저장할 정보

```
draftId: DRAFT-08
owner: 현재 로그인 사용자
lastStep: evidence
fields: eligible·name·phone·org
expiresAt: 30일 뒤
resumeUrl: /apply/evidence?draft=DRAFT-08
```

보존 기간과 소유 조건은 보안·개인정보 정책을 따라야 합니다. 무조건 오래 저장하는 것이 학습 목표가 아닙니다.

9.3. 같은 과정에서 반복 입력을 줄입니다

WCAG 2.2 Redundant Entry는 같은 과정에서 이미 입력하거나 제공한 정보를 다시 요구하면 자동 채우거나 선택할 수 있게 하되, 필수·보안·유효성 만료 같은 예외를 둡니다. 뒤로가기·오류·재진입에서 답변을 지우지 않는 것은 흐름 품질이자 접근성 문제입니다.

9.4. 일회성 행동 뒤로가기

결제·최종 제출 뒤 브라우저 뒤로가기가 같은 행동을 다시 실행하게 해서는 안 됩니다. 뒤로 버튼은 작동하되 “이미 접수된 신청입니다”처럼 현재 결과를 보여 주고 중복 효과를 막습니다.

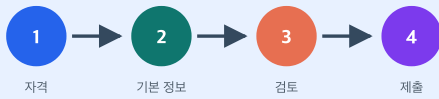
10. 실제 순서에 맞는 진행 모델을 고릅니다

순서가 필수일 때만 선형 진행 표시를 쓴다

한 번에 이어지는 질문 흐름과 여러 순서로 끝낼 수 있는 과업 목록은 다른 탐색 모델이다

선형 질문 흐름

앞 답변이 다음 질문을 정하는 경우



- 현재 위치와 뒤로가기를 제공
- 조건에 따라 총 단계가 달라질 수 있음

유연한 과업 목록

사용자가 편한 순서로 여러 일을 끝내는 경우

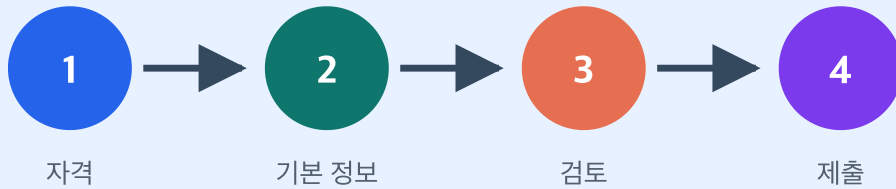
연락처	완료
증빙 업로드	진행 중
교육 선택	시작 가능
동의 확인	잠김

진행 표시의 모양은 멋보다 실제 제약·선택권·남은 일과 일치해야 한다

그림 8. 앞 답변이 다음 질문을 정하면 선형 흐름이, 독립 과업을 원하는 순서로 끝낼 수 있으면 과업 목록이 더 정직합니다.

선형 질문 흐름

앞 답변이 다음 질문을 정하는 경우



- 현재 위치와 뒤로가기를 제공
- 조건에 따라 총 단계가 달라질 수 있음

유연한 과업 목록

사용자가 편한 순서로 여러 일을 끝내는 경우

연락처

완료

증빙 업로드

진행 중

교육 선택

시작 가능

동의 확인

잠김

10.1. 선형 질문 흐름

GOV.UK question page 패턴은 한 페이지에 한 질문부터 시작하면 사용자가 특정 질문에 집중하기 쉽다고 설명합니다. back link·page heading·continue button을 기본으로 하고 같은 여정에서 같은 정보를 다시 묻지 않도록 안내합니다.

10.2. 과업 목록

여러 독립 과업을 원하는 순서로 완료할 수 있다면 task list가 어울립니다. 각 과업은 짧은 이름과 시작 가능·진행 중·완료·잠김 같은 상태를 가집니다.

10.3. 진행 표시가 거짓말하지 않게 합니다

- 분기에 따라 단계 수가 달라지는데 “2/5”를 고정하지 않음
- 선택 가능한 순서를 선형으로 강제하지 않음
- 완료한 과업·남은 과업을 색만 아니라 텍스트로 표시
- 실제 사용자 필요에 따른 순서로 나열

GOV.UK의 step-by-step navigation은 시작·끝이 분명하고 여러 안내·거래를 특정 순서로 완료하는 전체 여정에 사용하며, 거래 내부에는 별도의 multiple tasks 패턴을 권합니다. 패턴의 모양을 복사하기보다 적용 범위를 읽습니다.

30초 확인

네 과업을 어떤 순서로 해도 되는데 1→2→3→4 stepper를 만들었습니다. 위험은 무엇입니까?

정답 보기

실제 선택권과 다른 순서를 강제하거나 사용자가 뒤 과업을 할 수 없다고 오해하게 합니다. 과업 목록과 상태가 더 적합한지 검토합니다.

11. 검토·확정·완료를 분리합니다

중요한 제출은 검토·확정·완료를 분리한다

사용자가 고치고 확인할 기회를 주고 완료 뒤에는 참조 번호·다음 단계·기록 방법을 남긴다



완료 화면은 “끝”이 아니라 사용자가 결과를 증명하고 다음 일을 시작하는 출발점이다

그림 9. 중요한 제출은 입력 확인·최종 의도·완료 결과를 분리해 실수를 줄이고 결과를 증명하게 합니다.



완료 화면은 “끝”이 아니라 사용자가 결과를

3

완료

참조 번호·다음 행동

접수 YC-2407

처리 예정일

PDF 저장·문의

를 증명하고 다음 일을 시작하는 출발점이다

11.1. 검토 화면

GOV.UK check answers 패턴은 작은·중간 거래에서 완료 화면 바로 전에 한 번 검토하게 하고, 이전 답을 고치러 가도 이미 입력한 정보가 채워져 있도록 안내합니다.

이름	김기발	변경
연락처	010-1234-5678	변경
교육 과정	기발자 입문	변경
증빙	employment.pdf	변경

11.2. 중요한 결과 확인

WCAG 2.2 Error Prevention은 법적 의무·금전 거래·사용자 데이터 수정·삭제·시험 응답 같은 경우 제출을 되돌릴 수 있거나, 오류를 검사하고 고칠 기회를 주거나, 최종 전 검토·확인할 수 있어야 한다고 규정합니다.

11.3. 처리 중과 중복 보호

- 제출 후 버튼을 진행 상태로 바꾸고 반복 행동을 줄임
- 같은 요청의 서버 멍등성·업무 중복 판정 정의
- 새로고침·뒤로가기·재전송에서도 결과 조회 가능
- 성공 응답을 잃어도 접수 상태를 다시 확인할 수 있음

11.4. 완료 화면

GOV.UK confirmation page 패턴은 완료 사실, 참조 번호, 다음에 무엇이 언제 일어나는지, 문의·관련 서비스, 거래 기록을 저장할 방법을 제공하도록 안내합니다. 북마크로 돌아오는 사용자에게도 도움 되는 응답을 고려합니다.

BIG IDEA 03

완료 화면은 거래의 모비가 아니라 사용자가 결과를 증명하고 다음 현실 행동을 시작하는 출발점입니다.

12. 접근 가능한 흐름은 예측 가능합니다

접근 가능한 흐름은 순서·이름·복구가 예측 가능하다

키보드·보조 기술·인지 부담을 고려해 여러 화면에서도 같은 행동과 상태를 일관되게 제공한다

초점 순서

의미와 조작 순서 보존

일관된 탐색

반복 메뉴의 상대 순서 유지

같은 행동 이름

반복 기능의 이름 일치

반복 입력 줄이기

이전 값을 자동 채움·선택

오류 복구

문제·수정 방법을 텍스트로

중요 행동 확인

되돌림·검사·최종 확인

상태 변화·새 화면·오류가 생겨도 현재 위치와 다음 행동을 잃지 않게 한다

그림 10. 화면이 바뀌어도 순서·이름·복구 규칙이 유지되면 키보드·보조 기술·인지 부담 측면에서 흐름을 예측할 수 있습니다.

초점 순서

의미와 조작 순서 보존

일관된 탐색

반복 메뉴의 상대 순서 유지

반복 입력 줄이기

이전 값을 자동 채움·선택

오류 복구

문제·수정 방법을 텍스트로

상태 변화·새 화면·오류가 생겨도 현재

같은 행동 이름

반복 기능의 이름 일치

중요 행동 확인

되돌림·검사·최종 확인

해 위치와 다음 행동을 잃지 않게 한다

12.1. 초점 순서와 화면 전환

WCAG Focus Order는 순차 탐색의 초점 순서가 의미와 조작 가능성을 보존하도록 요구합니다. 새 화면 뒤 제목으로 초점을 옮길지, 오류 요약으로 옮길지, 현재 컨트롤을 유지할지 상황별로 정의합니다.

고정 헤더·쿠키 배너·채팅 창이 키보드 초점 요소를 완전히 가리지 않도록 검수합니다.

12.2. 일관된 탐색과 이름

- 반복 탐색은 여러 화면에서 같은 상대 순서 유지
- 같은 기능은 같은 이름과 접근 가능한 이름 사용
- `계속`, `저장 후 나가기`, `신청 확정`의 의미를 화면마다 바꾸지 않음
- 초점을 받았다는 이유만으로 자동 제출·새 창·큰 문맥 변화를 만들지 않음

12.3. 오류 식별과 수정 제안

WCAG Error Identification은 자동 감지한 입력 오류의 항목과 문제를 텍스트로 설명하도록 요구합니다. Error Suggestion은 알려진 수정 방법이 있으면 보안·목적에 해치지 않는 범위에서 제안하도록 합니다.

약함: 입력 오류
개선: 연락처는 숫자 10~11자리로 입력하세요. 예: 01012345678

12.4. 상태 메시지

초점을 옮기지 않는 저장 완료·검증 중·오류 수·검색 결과 변화는 `role="status"` 등으로 보조 기술에 전달할 수 있습니다. 새 화면으로 실제 이동한 경우에는 제목·초점·route 변화를 함께 검수합니다.

13. 흐름을 시나리오와 지표로 검증합니다

13.1. 최소 시나리오

시나리오	시작 조건	기대 끝점
happy path	자격·자료·서비스 정상	완료·참조 번호
대상 아님	자격 false	대체 경로·blocked
저장·재진입	자료 미준비	draft 복원·중단 위치

시나리오	시작 조건	기대 끝점
오류·재시도	첫 제출 503	답변 보존·재시도 완료
세션 만료	정보 입력 뒤 만료	보존 결과·재시작
중복 제출	빠른 두 번 확정	한 번 처리·결과 조회

13.2. 경로·상태 커버리지

방문한 화면 수 / 정의한 화면 수 만으로 충분하지 않습니다. 화면별 loading·empty·error·blocked 상태를 실제 데이터와 조건으로 실행합니다.

13.3. 전이 지표

자격 화면 진입 → 자격 답변 완료
 자료 화면 진입 → 저장 후 나가기
 저장 사용자 → 30일 안 재진입
 검토 진입 → 제출 성공
 제출 오류 → 재시도 성공

분모·분자·기간·중복 사건 제거를 정의합니다. 이탈이 항상 실패는 아닙니다. 저장 후 나가는 의도한 흐름일 수 있습니다.

13.4. 사용성 시험

사용자에게 화살표를 설명하지 말고 실제 목표를 줍니다. 어느 진입점을 선택하고, 어디서 멈추고, 무엇을 다시 읽고, 브라우저 뒤로를 신뢰하는지 관찰합니다. TO-BE는 이 증거로 다시 수정합니다.

14. 실습 · 여섯 경로와 화면 상태를 실행합니다

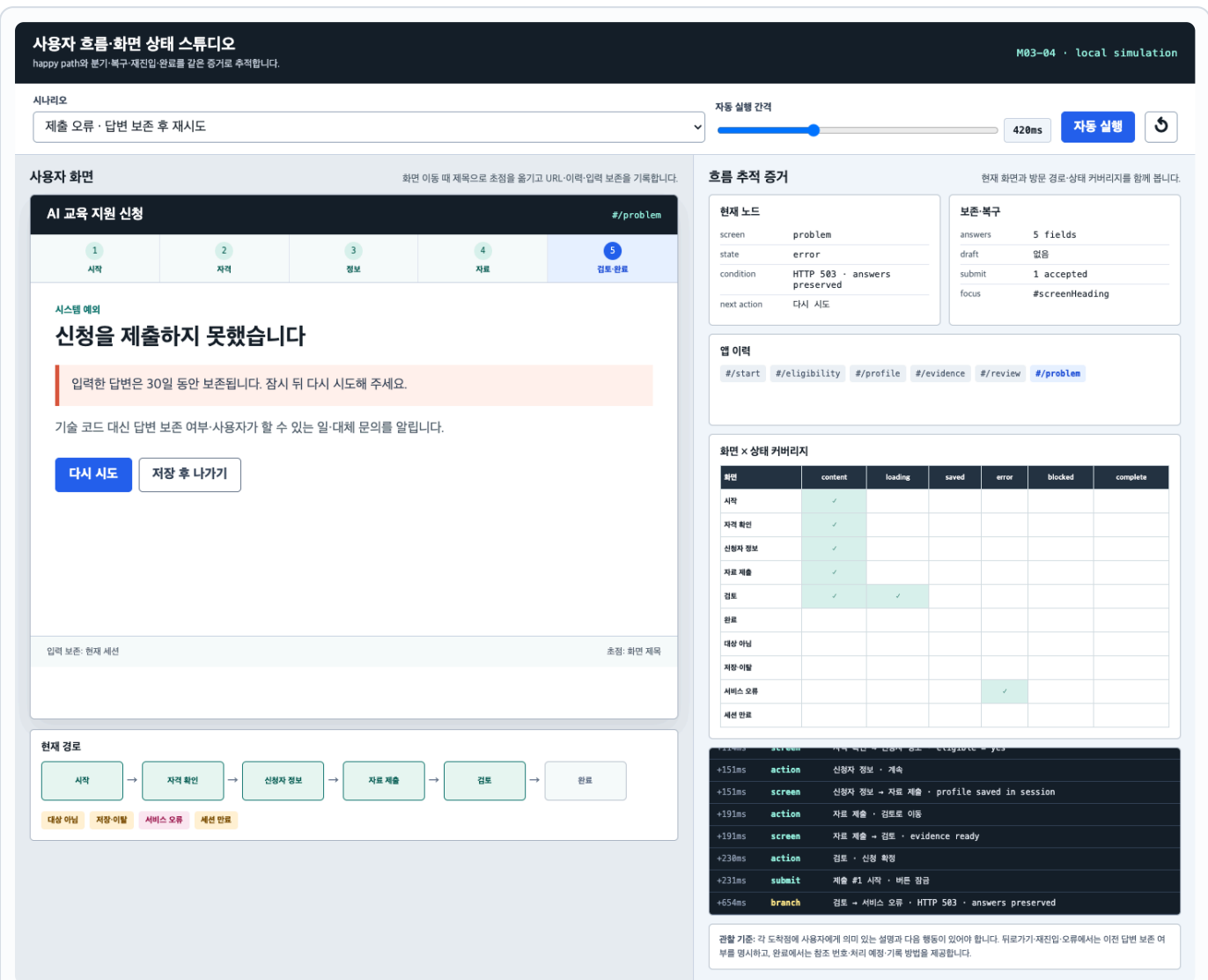


그림 11. 스튜디오는 현재 화면·URL·이력·입력 보존·화면×상태 커버리지·분기 시간선을 한 화면에 보여 줍니다.

14.1. 준비 파일

- L03-04 사용자 흐름·화면 상태 스튜디오
- L03-04 실습 안내
- T03-04 사용자 흐름·화면 정의서
- 사용자 흐름·화면 상태 용어집

14.2. 반드시 비교할 결과

실행	볼 것
정상 완료	start→confirmation·참조 번호·다음 행동
대상 아님	eligibility→ineligible·blocked·대체 경로
저장·재진입	evidence→saved→evidence·draft·30일
오류·재시도	review loading→problem error→review→complete
세션 만료	profile→expired·사라진 정보·재시작
중복 제출	submit accepted 1회·guard 로그

14.3. 산출물

- 1. 사용자 목표·진입·완료 범위 1개
- 2. happy path와 분기·예외 3개 이상
- 3. 화면×상태 표
- 4. 뒤로·저장·재진입·만료 정의
- 5. 화면 정의 3개 이상
- 6. 개선 요청 3건

실제 서비스 흐름을 캡처할 때 계정·개인정보·내부 URL·접수 번호를 제거합니다. 운영 지표는 개인 경로가 아니라 집계된 전이로 공유합니다.

15. 인쇄용 흐름·화면 워크시트

15.1. 일곱 질문

질문	기록
목표	
진입·준비	

질문	기록
단계	
분기	
화면 상태	
복구	
완료 증거·다음 행동	

15.2. 노드와 전이

ID	종류	이름	사건·조건	다음	상태·보존
1	시작				
2	화면				
3	결정				
4	화면				
5	종료				

15.3. 끝점 검수

끝점: _____ 종류: 완료·차단·취소·오류·만료

결과 설명: _____

보존 여부·기간: _____

다음 행동: _____

복구·대체·문의: _____

완료 증거: _____

16. 셀프 테스트 · 정답을 가리고 풀니다

문제 1

사용자 여정과 화면 흐름의 관점 차이를 쓰십시오.

문제 2

AS-IS와 TO-BE를 각각 무엇으로 뒷받침해야 합니까?

문제 3

흐름도의 결정 노드에 “예” 경로만 있습니다. 무엇을 추가 확인해야 합니까?

문제 4

조건 분기·사용자 선택·시스템 예외의 예를 하나씩 쓰십시오.

문제 5

화면×상태 매트릭스의 빈 칸을 어떻게 판정합니까?

문제 6

브라우저 뒤로가기에서 이전 페이지 URL만 맞으면 충분합니까?

문제 7

같은 과정에서 이전에 입력한 정보를 다시 요구할 때 접근성 관점의 기본 대책은 무엇입니까?

문제 8

독립 과업을 원하는 순서로 완료할 수 있을 때 stepper와 task list 중 무엇이 더 적합할 수 있습니까?

문제 9

중요한 제출에서 WCAG 2.2가 제시하는 오류 예방 방식 세 가지 중 하나 이상을 쓰십시오.

문제 10

완료 화면에 있어야 할 정보 네 가지를 쓰십시오.

17. 셀프 테스트 정답과 해설

정답 1

사용자 여정은 필요 발생부터 종료까지 시간·채널·생각·감정과 전체 접점을 보고, 화면 흐름은 구현할 화면·URL·상태 사이 이동을 봅니다.

정답 2

AS-IS는 사용자 관찰·문의·분석·운영 로그 같은 현재 증거로, TO-BE는 검증할 가설·측정 가능한 전이·완료 기준으로 뒷받침합니다.

정답 3

“아니요”, 알 수 없음, 권한 없음, 조회 오류 등 실제 가능한 모든 결과와 각 도착점의 설명·다음 행동을 확인합니다.

정답 4

조건 분기는 자격 미충족, 사용자 선택은 자료가 없어 저장 후 나가기, 시스템 예외는 제출 API 503 등이 될 수 있습니다.

정답 5

그 상태가 정말 필요 없는지, 정의가 빠진 것인지 근거를 남겨 판정합니다. 빈 칸을 자동 통과로 보지 않습니다.

정답 6

아닙니다. 이전 답변·오류·스크롤·초점 등 사용자가 마지막으로 본 의미 있는 상태가 복원되는지 확인합니다.

정답 7

같은 과정에서 이미 입력한 정보를 다시 묻지 않거나, 자동 채우거나, 사용자가 이전 값을 선택할 수 있게 합니다. 필수·보안·만료 예외는 별도 설명합니다.

정답 8

task list가 더 적합할 수 있습니다. stepper는 실제로 순서가 필요한 선형 흐름에 사용합니다.

정답 9

제출을 되돌릴 수 있게 하거나, 입력 오류를 검사하고 수정 기회를 주거나, 최종 전에 검토·확정하게 하는 방식입니다.

정답 10

완료 사실·참조 번호, 다음 처리와 시점, 문의·관련 다음 서비스, 거래 기록을 저장할 방법 등이 필요합니다.

오답 진단

문제	다시 볼 것	그림
1·2	지도 관점·근거	그림 2·3
3·4	문법·분기·복구	그림 4·5
5	화면×상태	그림 6
6·7	이력·보존·반복 입력	그림 7
8	진행 모델	그림 8
9·10	중요 제출·완료	그림 9

18. 한 장 요약과 다음 단계

한 장으로 복습하는 사용자 흐름·화면 상태 설계

목표에서 완료까지 일곱 칸과 화면×상태 표를 채우면 happy path 밖의 누락도 보인다



그림 12. 일곱 칸과 화면×상태 표를 채우면 happy path 밖의 막힘·정보 손실·복구 누락을 구현 전에 찾을 수 있습니다.

1

목표

사용자가 얻을 결과

2

진입

경로·자격·준비

5

상태

loading·empty·error

6

복구

뒤로·저장·재시도

완료 기준 = 모든 도착점에 의미 있는 메시지

3

단계

행동·화면 순서

4

분기

조건·선택·권한

7

완료

확인·기록·다음 일

지·다음 행동·보존 여부·측정 증거가 있다

18.1. 60초 설계 순서

1. 사용자의 목표·시작 신호·완료 증거를 쓴다.
2. AS-IS 행동·문의·지표 증거를 붙인다.
3. 시작·화면·행동·결정·외부·종료 노드를 연결한다.
4. 조건 분기·사용자 선택·시스템 예외의 모든 끝점을 닫는다.
5. 화면×상태 매트릭스로 누락을 찾는다.
6. 뒤로·저장·재진입·만료·중복 제출을 별도 실행한다.
7. TO-BE 완료 기준과 측정 전이를 쓴다.

18.2. 품질 체크

- ☐ 기능이 아니라 사용자 결과를 목표로 썼습니다.
- ☐ 범위 안·밖과 다른 서비스·채널 연결이 보입니다.
- ☐ 모든 결정 결과와 끝점이 있습니다.
- ☐ dead end마다 설명·보존·다음 행동·복구가 있습니다.
- ☐ 화면마다 필요한 상태와 문구·행동을 정의했습니다.
- ☐ 브라우저 뒤로·새로고침·깊은 링크를 검수했습니다.
- ☐ 저장본의 소유자·마지막 위치·보존 기간을 정의했습니다.
- ☐ 중요 제출의 검토·확정·처리 중·완료를 구분했습니다.
- ☐ 초점·일관된 이름·반복 입력·오류 복구를 확인했습니다.
- ☐ 경로·상태 시나리오와 완료 기준이 있습니다.

18.3. 다음 매뉴얼

다음은 **M04-01 백엔드가 처리하는 일 구분하기**입니다. 화면 흐름에서 보이지 않는 검증·권한·업무 규칙·저장·외부 연계·비동기 처리를 백엔드 책임으로 구분하고 처리 흐름도를 만듭니다.

공식 참고 자료

- GOV.UK Service Standard · Solve a whole problem for users
- GOV.UK Service Manual · Map and understand a user's whole problem
- GOV.UK Service Manual · Creating an experience map
- GOV.UK Design System · Question pages
- GOV.UK Design System · Check answers
- GOV.UK Design System · Confirmation pages
- GOV.UK Design System · There is a problem with the service

- WHATWG HTML · Navigation and session history
- W3C WCAG 2.2 · Focus Order
- W3C WCAG 2.2 · Consistent Navigation
- W3C WCAG 2.2 · Redundant Entry
- W3C WCAG 2.2 · Error Identification
- W3C WCAG 2.2 · Error Suggestion
- W3C WCAG 2.2 · Error Prevention

이 파일은 2026-07-15에 위 공식 자료와 Chrome 150 환경으로 검토했습니다. 사용자 흐름 도형과 분류는 학습을 위한 YEONCORE 표기 규칙입니다.