

AI Story UX 사전 과제

박강근 · Senior Product Designer · 웹툰 Product Designer (AI Story본부) · 2026.10

Prototype power-xc.github.io/ai-story-ux 직접 눌러 보는 프로토타입입니다. 각 항목의 ▶ 링크는 해당 화면으로 바로 이동합니다.

요약

저는 AI가 틀리거나 막히는 순간을 설계해 온 디자이너입니다. 업무용 AI에서는 그 순간이 불편으로 끝나지만, 크랙에서는 몰입과 그동안 쌓은 관계가 한 번에 무너집니다. 그래서 두 질문 모두 "사용자가 이야기 밖으로 밀려나지 않게"를 기준으로 답했습니다.

- 1번** 크랙은 웹툰·웹소설이 주던 쉬움과 리듬, 믿고 보는 품질을 지키면서, 내가 주인공이 되고 나에게만 쌓이는 이야기를 더해야 한다고 봅니다. 가장 먼저 해 보고 싶은 것은 첫 턴(원칙 1)과 설정의 일관성(원칙 3)입니다.
- 2번** 할루시네이션을 골랐습니다. 완전히 없앨 수는 없으니, 틀릴 수 있는 답이 사용자를 속이지 않게 하는 것이 화면의 몫입니다. 크랙에서는 이를 "설정 할루시네이션"으로 보고, 이야기 안에서 바로잡게 하겠습니다.
- 근거** 모두 AI 에이전트와 AI 보안 제품에서 직접 설계하고 숫자로 확인한 흐름에서 가져왔습니다. 차단 후 이탈률 36% → 6%, 근거 확인 시간 2분 48초 → 38초, 응답 평가 360회.

1번 문항

크랙이 웹툰·웹소설 시장을 대체하기 위해 차별점을 두어야 하는 사용자 경험 원칙

출발점	지킬 것	엄지 하나로 넘기는 쉬움(원칙 1, 2), 다음 화를 기다리게 하는 리듬(원칙 4), 작가와 편집이 지켜 주는 믿음(원칙 3)
	더할 것	내가 주인공이 되는 경험, 나에게만 쌓이는 이야기, 보던 사람이 만드는 사람이 되는 순환(원칙 5). 크랙이 앱 소개에서 스스로를 "내가 주인공인 웹소설"이라고 부르는 것도 같은 방향이라고 읽었습니다.
원칙 1 참여는 읽기만큼 쉬워야 한다 ▶ 프로토타입 E · 선택지로 이어가기	왜	웹툰은 스크롤 한 번이면 시작되지만, 채팅은 빈 입력창 앞에서 "뭐라고 쓰지?"부터 고민하게 합니다. 쓰는 부담이 읽는 즐거움보다 먼저 오면, 이야기에 들어오기도 전에 떠납니다.
	어떻게	첫 장면은 캐릭터가 먼저 말을 겁니다. 상황과 관계, 지금 고를 수 있는 선택지까지 함께 건넵니다. 행동 제안을 누르거나 한 단어만 써도 장면이 이어지게 합니다.
	지표	첫 메시지를 보냈는지보다, 초반 몇 턴을 이어 갔는지를 봅니다.
	해 본 일	직업훈련 플랫폼 STEP에서 신청하고도 5명 중 1명이 첫 학습 전에 떠나는 지점을 다시 설계해, 첫 학습 시작률을 80.12%에서 85.66%로 올렸습니다. 첫 행동까지의 거리가 짧아야 그다음에 있다고 봅니다.
원칙 2 몰입의 깊이는 사용자가 고른다 ▶ 프로토타입 E · 관전/선택/직접 쓰기	왜	같은 사람도 출근길에는 그냥 보고 싶고, 자기 전에는 깊게 빠져들고 싶습니다. 늘 직접 쓰기를 요구하면 웹툰이 차지하던 "가볍게 보는 시간"은 가져올 수 없습니다.
	어떻게	관전, 선택, 직접 쓰기 중 얼마나 개입할지 고르는 다이얼을 두고, 대화 중에도 한 번에 바꿀 수 있게 합니다.
	지표	모드별 세션 길이와 재방문, 그리고 관전하던 사람이 직접 쓰기로 넘어오는 비율을 봅니다.

원칙 3
이야기의 규칙은 흔들리지 않는다

▶ 프로토타입 E · 기억 노트

왜	어제까지 다정하던 캐릭터가 갑자기 다른 사람처럼 말하면, 독자는 그 순간 작품에서 빠져나옵니다. 웹툰에서 작가와 편집이 지키던 일을, AI 스토리에서는 시스템과 화면이 대신 지켜야 합니다.
어떻게	AI가 무엇을 기억하고 이야기를 잇는지 "기억 노트"로 보여 주고, 사용자가 직접 고칠 수 있게 합니다. 틀렸을 때는 이야기 밖으로 나가지 않고 그 자리에서 바로잡습니다(2번 문항).
지표	"앞 내용과 달라요" 같은 설정 신고가 얼마나 나오는지, 신고한 뒤에도 이야기를 이어 가는지를 봅니다.
해 본 일	캐릭터에 관심이 많아, 1인으로 운영하는 서비스 MOHOW의 대표 캐릭터를 AI로 직접 만들었습니다. 캐릭터 4종을 하나로 정리하면서 원래 색만 쓰는 규칙과, 캐릭터가 나오는 자리와 나오지 않는 자리를 정했습니다. 그림 캐릭터에서 생김새가하던 일을, 크랙에서는 말투와 기억이 한다고 봅니다.

원칙 4
이야기에는 매듭이 있고, 내 이야기는 쌓인다

왜	웹툰의 한 화는 작은 마무리와 다음 화를 기다릴 이유를 함께 줍니다. 끝없이 이어지는 채팅에는 이 리듬이 없어서, 이야기가 흐지부지 끝나기 쉽습니다.
어떻게	대화를 장면과 회차로 묶어, 장면이 끝나면 요약 카드와 다음 장면으로 이어지는 갈고리를 보여 줍니다. 내가 만든 이야기는 회차처럼 다시 읽고, 이어 하고, 친구에게 보여 줄 수 있게 합니다.
지표	장면 완주율, 다음 장면 이어 하기 비율, 다음 날 다시 오는 비율을 봅니다.

원칙 5
즐기는 사람이 만드는 사람이 된다

왜	웹툰·웹소설은 작가가 쓰는 속도보다 빨리 나올 수 없습니다. 크랙에서는 즐기던 사람이 만든 세계관과 캐릭터가 다시 누군가의 플레이로 이어집니다. 이 순환이 가장 큰 차별점이라고 봅니다.
어떻게	플레이하던 장면에서 "이 설정으로 만들기"로 바로 넘어가게 합니다. 만든 사람에게는 인기 지표와 함께, 사람들이 어디서 떠났고 어디서 다시 쓰기를 눌렀는지 같은 개선 신호와 공개 전 테스트 대화를 줍니다.
지표	플레이하다 만들기로 넘어오는 비율, 만든 사람이 고쳐서 다시 공개하는 비율, 작품별 재방문을 봅니다.

무엇부터 할까

함께하게 된다면 원칙 1과 원칙 3부터 해 보고 싶습니다. 처음에 이야기에 들어오지 못하면 나머지 원칙은 경험될 기회조차 없고, 설정이 무너지면 쌓아 온 이야기와 관계가 한 번에 무너지기 때문입니다. 첫 장면 형식처럼 작은 실험부터 빠르게 해 보고, 지표가 좋은 것을 골라 넓히겠습니다.

2번 문항

LLM의 한계 중 "할루시네이션"을 UX/UI로 해결하는 방법

왜 할루시네이션인가

느린 답변이나 앞뒤가 안 맞는 말은 사용자가 바로 알아챱니다. 할루시네이션은 그럴듯해서, 믿고 움직인 뒤에야 드러납니다. 그때 잃는 것은 답 하나가 아니라 서비스를 믿던 마음입니다. 할루시네이션을 아예 없애는 것은 모델과 검색의 몫이고, 화면이 할 일은 **틀릴 수 있는 답이 사용자를 속이지 않게 하는 것**이라고 생각합니다.

① 알아챱 수 있게 (과정)	② 확인할 수 있게 (근거)	③ 바로잡을 수 있게 (복구)	④ 다음에는 덜 틀리게 (평가)
--------------------	--------------------	---------------------	-------------------------

해결 1
답이 만들어지는 과정을 보여 준다

▶ 프로토타입 A · 과정 보여 주기

어떻게	"질문 이해 → 자료 찾기 → 답 작성"처럼 지금 어디까지 왔는지 보여 줍니다. 첫 단계에서 "이렇게 이해했어요"를 먼저 보여 주면, 답이 나오기 전에 잘못 알아들은 부분을 고칠 수 있습니다. 기다리는 시간도 "멈춘 것"이 아니라 "일하는 중"으로 느껴집니다.
해 본 일	NL2SQL 데이터 분석 에이전트(SAPIE DeepQ)에서 AI가 질문을 어떻게 해석하고, 어떤 데이터를 찾고, 어떻게 결과를 만드는지 보여 주는 패널을 설계했습니다.

해결 2

근거를 답 옆에 두고, 확인 비용을 낮춘다

▶ 프로토타입 B · 근거 연결

어떻게	사실을 말하는 문장마다 근거(참조한 자료, 대화 앞부분, 설정)를 이어, 눌러 보면 바로 확인되게 합니다. 근거를 찾지 못한 내용은 단정하지 않는 말투와 표시로 구분합니다. 출처를 답 끝에 모아 두기만 하면 잘 열어 보지 않기 때문입니다.
해 본 일	생성형 AI 보안 제품(SAPIE Guardian)의 관리자 화면을 원본과 탐지 근거, 적용 정책을 한눈에 보는 구조로 바뀌, 관리자 7명 UT에서 처리 근거를 확인하는 시간을 2분 48초에서 38초로 77% 줄였습니다. DeepQ에서는 참조 데이터와 스키마, 생성 SQL을 검토하는 화면을 설계했습니다.

해결 3

확신은 정직하게, 경고는 위험 수준에 맞게

▶ 프로토타입 C · 위험 수준별 표시

어떻게	모든 답에 경고를 붙이면 곧 아무도 읽지 않고, 위험해 보이는 것을 다 막으면 사용자는 떠납니다. 창작과 일상 대화에는 아무 표시도 하지 않고, 사실 정보에는 근거를 잇고, 의료·법률·금융처럼 틀리면 다치는 주제는 확인할 곳과 함께 분명히 알립니다. 근거가 없으면 지어내지 않고 "잘 모르겠어요"라고 말한 뒤 다음에 할 일을 제안합니다.
해 본 일	Guardian에서는 강한 경고와 일괄 차단이 사용자에게 다크패턴에 가까운 강제 중단으로 다가갔습니다. 내부 사용자 약 30명의 차단 세션 약 800건을 분석해 위험 수준별 마스킹과 차단으로 바꿨고, 차단 후 이탈률을 36%에서 6%로 낮추고 작업 복귀율을 41%에서 86%로 높였습니다. 안전장치도 사용자가 다음 걸음을 뚫 수 있게 만들어야 한다는 것을 이때 배웠습니다.

해결 4

틀렸을 때 한 번에 바로잡는다

▶ 프로토타입 D · 사유 있는 복구

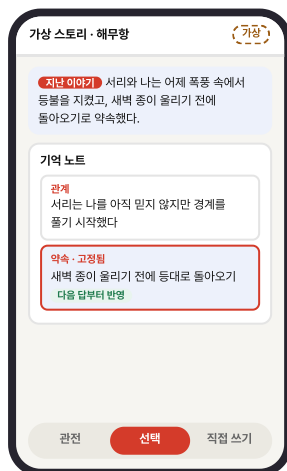
어떻게	"사실과 달라요 / 질문을 잘못 이해했어요 / 앞 내용과 달라요"처럼 이유를 고르면, 그 이유를 반영해 다시 답합니다. 그냥 다시 생성하면 왜 틀렸는지 모른 채 같은 실수를 반복하기 쉽고, 고른 이유는 그대로 개선 데이터가 됩니다. 되돌리기 어려운 행동은 AI가 제안하고 사람이 확정합니다.
해 본 일	DeepQ에서 실패 원인 안내와 재질의 복구 흐름을 설계했습니다. 대형 F&B 프랜차이즈의 AI 발주 추천 POS POC에서는 AI가 제안하면 점주가 근거와 손익을 보고 고쳐서 최종 승인하는 Human-in-the-Loop 흐름을 설계했습니다.

해결 5

출시 전후로 측정한다

어떻게	출시 전에는 질문 유형 × 검토 방향 × 반복 횟수로 평가 세트를 만들어, 모델이나 프롬프트, 화면이 바뀔 때마다 같은 세트로 비교합니다. 출시 후에는 근거를 열어 본 비율, 신고 이유의 분포, 다시 묻는 비율, 다음 날 재방문을 봅니다.
해 본 일	DeepQ에서 질의 유형 24개 × 검토 방향 3개 × 5회, 총 360회 시뮬레이션으로 응답 평가 체계를 세웠습니다. 합성 페르소나로 다크패턴과 UX 리스크를 먼저 찾아보는 리서치 도구(Nemotron Persona Lab)도 직접 만들었습니다.

크랙에 옮기면: "설정 할루시네이션"



설명용 가상 화면 · 실제 크랙 화면이 아닙니다

이야기 속 "사실"은 현실이 아니라 캐릭터 설정과 세계관, 그리고 지금까지 나눈 대화입니다. 새 전개를 지어내는 것은 창작이고, 크랙의 가치입니다. 문제는 어제 한 약속을 다르게 기억하거나, 없던 능력을 쓰거나, 관계가 이유 없이 바뀌는 순간입니다.

일반 채팅 서비스	크랙
과정 보여 주기	장면이 시작될 때 "지난 이야기" 한 줄로 무엇을 이어 가는지 보여 줍니다
근거 연결	기억 노트: 캐릭터가 기억하는 관계와 사건, 약속을 보여 주고, 사용자가 고정하거나 고치거나 지웁니다
위험 수준별 표시	창작은 막지 않습니다. 설정이나 기억과 부딪칠 때만, 몰입을 깨는 경고 대신 이야기 안에서 바로잡습니다
사유 있는 복구	"캐릭터답지 않아요 / 앞 내용과 달라요 / 설정과 맞지 않아요"를 고르면 그 이유로 다시 씁니다
출시 전후 측정	크리에이터 테스트: 공개 전에 대표 상황 × 설정 항목 × 반복 대화로 캐릭터가 설정에서 벗어나는 지점을 미리 보여 줍니다

업무용 AI를 만들며 배운 가장 큰 것은, 사람들은 AI가 완벽하기를 바라기보다 AI가 지금 무엇을 하는지 알고, 틀렸을 때 바로잡을 수 있기를 바란다는 점입니다. 크랙에서는 이 마음으로, AI의 실수 때문에 누군가 아끼던 이야기에서 밀려나지 않게 만들고 싶습니다.