고흐 갤러리 — 메인기획서

> 재구성 기획서 (역설계) / 작성 기준 시점: 2021-03-15 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P039 / 근거 수준: C / 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

핵심은 "클릭 → 다음 작품 페이드인" 단일 메카닉이다. 용어: 페이드 중 플래그 `NowFadeIn`(bool), 현재 순번 `NowEvent`(int), 전시 이벤트 `ShowEvent`(글로벌 이벤트). [근거: GoghGallary2.unity, Button.prefab]

클릭 입력과 진행 허용 판정

- 의도(Why): 연타해도 전시가 꼬이지 않고 한 점씩만 진행되게 한다.
- 설계(How): 버튼 FSM이 `UiButtonOnClickEvent`로 클릭(`click`)을 받아 `CheckFadeIn` 상태에서 `NowFadeIn`을 `BoolTest`한다. `yes`(페이드 중)면 아무 일 없이 대기 상태로 복귀하고, `no`면 `SendEvent` 상태에서 `ShowEvent`를 전역 송신한 뒤 대기로 돌아간다. [근거: Button.prefab] 직접 호출 대신 이벤트를 쓴 이유는 [추정: 버튼 프리팹을 씬과 독립해 재사용하려는 목적].
- 필요(What): PlayMaker 액션 `UiButtonOnClickEvent`, `BoolTest`, `SendEvent`.
- 작업 방법: 프리팹으로 제작 후 Canvas에 배치. 검증은 연타 테스트.
- 완료 기준: 페이드 중 클릭이 무시되고, 종료 후 클릭만 다음 작품을 부른다.
- 리스크/대안: 프리팹 텍스트가 `1`로 남아 있어 버튼 표기 정리가 필요하다. [근거: Button.prefab m_Text: 1]
- 근거: [근거: Button.prefab]

작품 전환(페이드인) 시퀀스

- 의도(Why): 작품이 암전에서 서서히 드러나는 감상 호흡을 만든다.
- 설계(How): 매니저 FSM이 `ShowEvent`를 글로벌 전이로 받아 4개 상태를 거친다. `SetNowFadeInTrue`(`NowFadeIn=true`) → `Disappear`(Image 4개 색을 투명으로 초기화) → `FadeIn` → `SetNowFadeInFalse`(`NowFadeIn=false`). `FadeIn`에서는 `IntAdd`로 `NowEvent`를 +1, `ArrayGet`으로 `images[NowEvent]`를 꺼내 `targetimage`에 저장, `EaseColor`로 투명→흰색을 2초 동안 보간, `UIImageSetColor`로 매 프레임 적용한다. [근거: GoghGallary2.unity] 매 전환 시 `Disappear`를 거치므로 이전 작품은 사라지고 새 작품만 보인다. [추정: 상태 이름과 알파 0 설정]
- 필요(What): `SetBoolValue`, `UiGraphicSetColor`, `IntAdd`, `ArrayGet`, `EaseColor`, `UIImageSetColor`.
- 작업 방법: 상태별로 순서 구성 → 에디터에서 `ShowEvent` 송신 테스트 → 빌드 확인.
- 완료 기준: 클릭마다 새 작품이 2초에 걸쳐 나타난다.
- 리스크/대안: `NowEvent` 초기값이 0이면 첫 클릭에서 배열 1번 항목이 먼저 나오고 0번이 건너뛰어진다. 마지막 이후 클릭은 `indexOutOfRange` 처리가 있어 안전한지 확인 필요하다. [근거: ArrayGet 파라미터 indexOutOfRange 존재 / 초기값 판정은 추정]

- 근거: [근거: GoghGallary2.unity]

2. 구현 방법

- 클라이언트: Unity 2019.4.22f1, 씬 `GoghGallary2.unity` 1개. 객체는 Canvas, Image 4개(`Image`, `Image (1)~(3)`), 매니저 `Maneger`(PlayMaker FSM), `PlayMakerGUI`, `EventSystem`, 버튼 프리팹 인스턴스. [근거: GoghGallary2.unity]
- 데이터 구조: FSM 변수 `fadecolor`(검정 a1), `targetimage`(Image), `images`(Image 배열 4), `NowFadeIn`, `NowEvent`. 외부 데이터 파일 없음. [근거: GoghGallary2.unity]
- 상태 머신: 매니저 FSM 시작 상태 `NotUsed`(빈 상태)에서 `ShowEvent` 글로벌 전이로 진입한다. [근거: GoghGallary2.unity]
- 코드: 프로젝트 고유 C# 없음. 모든 로직이 PlayMaker. [근거: Assets 폴더 구성]

3. 시스템 간 상호작용과 밸런스 기준

- 버튼 FSM과 매니저 FSM은 `ShowEvent`로만 연결되며, 공유 변수는 `NowFadeIn`(매니저 쪽 변수)과 같은 이름을 버튼 쪽에서 참조한다. [근거: 두 FSM 변수명]
- 핵심 변수: 페이드 시간 2초(EaseColor time). 이 값이 감상 호흡의 유일한 튜닝 값이다. [근거: GoghGallary2.unity]
- 밸런스 개념 없음(점수/난이도 없음).

4. 게임의 나머지 부분의 방향

- 콘텐츠 구조: 단일 씬, 작품 4점. 확장은 03 문서에서 다룬다.
- 메타 진행, 저장, 설정: 없음(기록 없음).
- UI/UX: 버튼 1종, 풀 화면 Image. 1920x1080 기준. [근거: ProjectSettings.asset]
- 아트/사운드: 작품 이미지만 있고 사운드 기록 없음.

5. 제작 과정 설계

1. 작품 이미지 임포트 → Image 4개 구성(선행).
2. 매니저 FSM → 버튼 프리팹(병렬 가능하나 이벤트명 `ShowEvent` 선합의).
3. 에디터 재생 확인 → 빌드(exe, zip). - 담당: 1인 수작업, AI 활용 여부 기록 없음. 검증은 에디터 클릭 테스트와 빌드 실행(이번 조사에서는 미실행). - 인수인계 문서: 없음.

6. 세부기획서 목록

문서	정할 것
03_세부기획서_전시구성과UIUX	전시 순서/전환 연출, UI, 작품 데이터, 리소스, 검수 기준