의성2077

프로젝트 개요

프로젝트명: **의성 데몬 헌터**

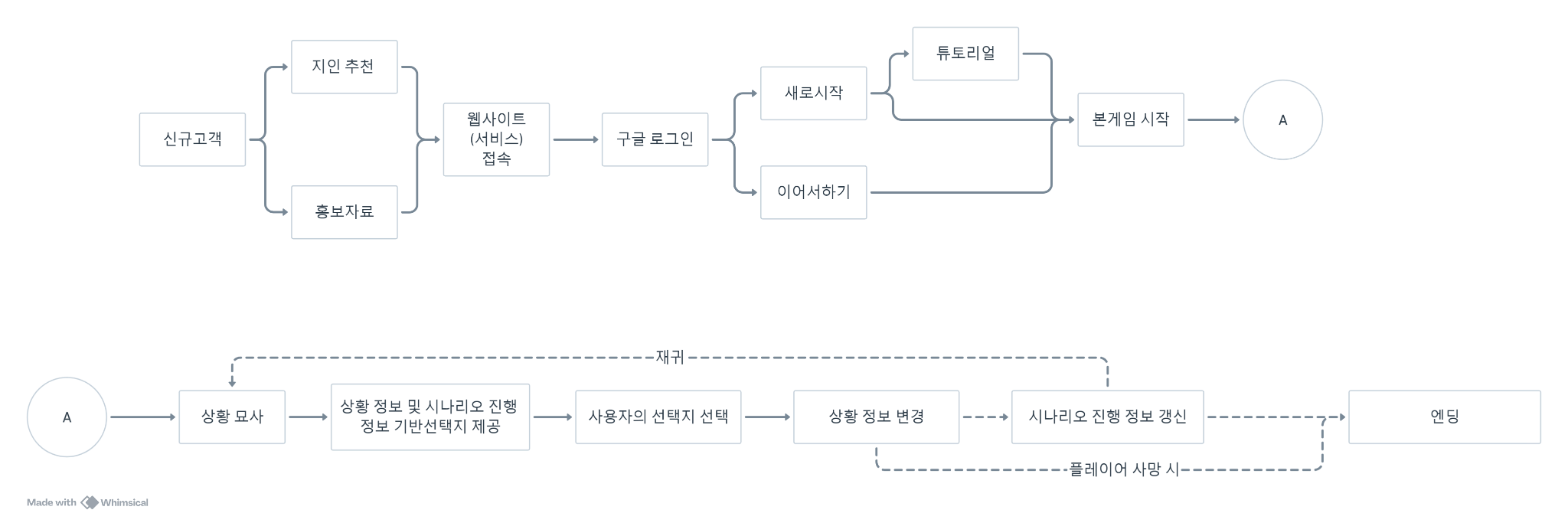
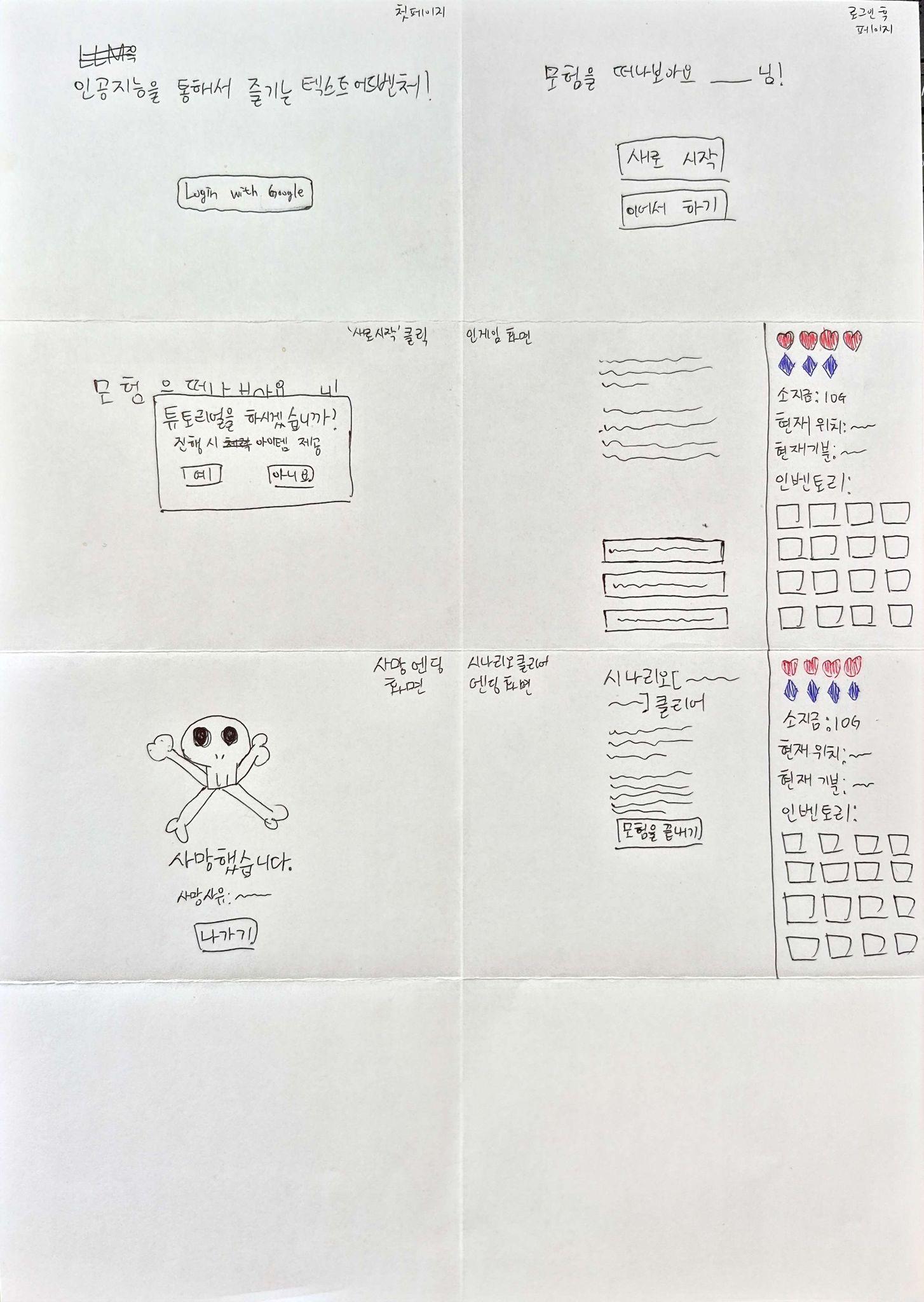
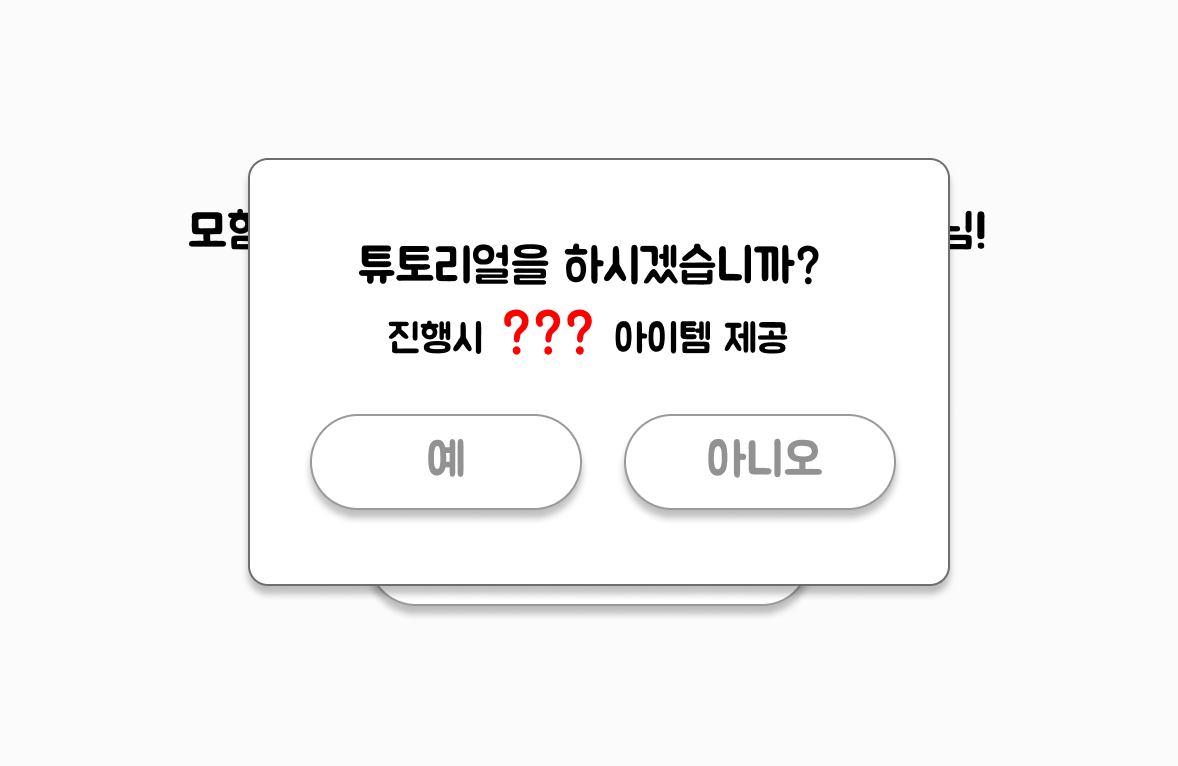
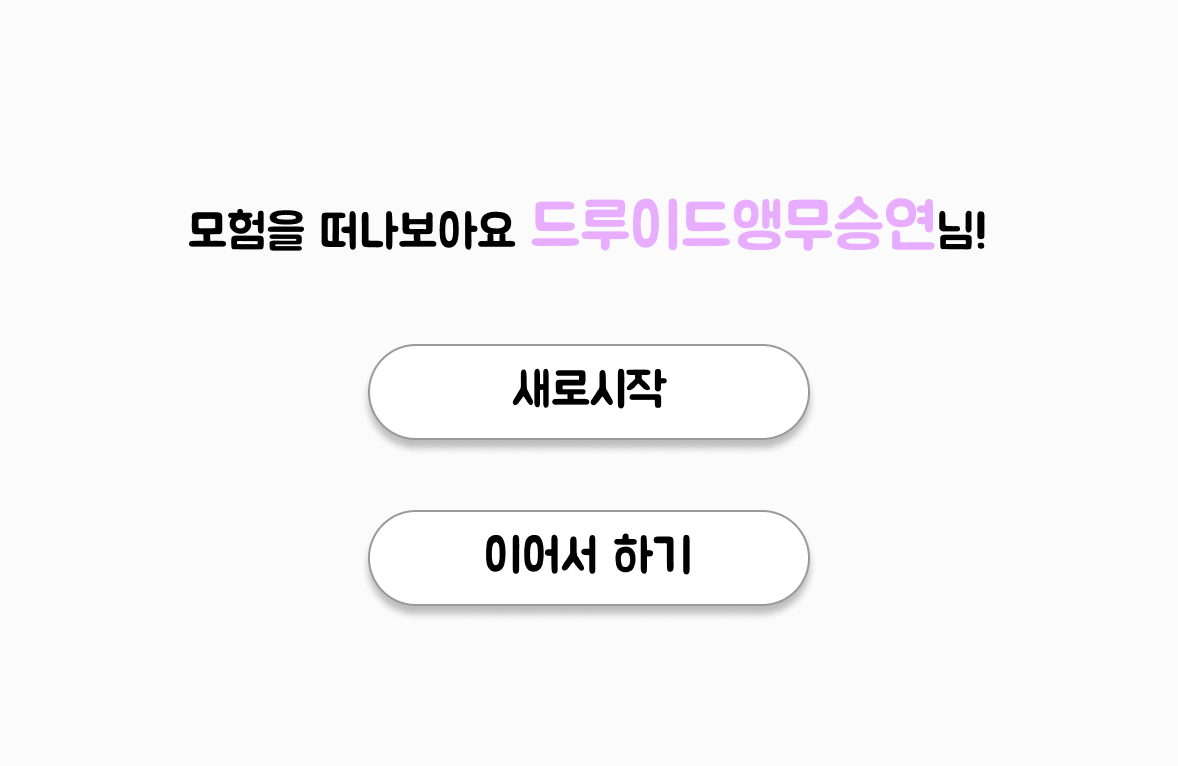
한 줄 목표: [의성 데몬 헌터]는 LLM을 활용한 의성을 배경으로 하는 텍스트 어드벤처 게임이다.

* 장기 목표: [의성 데몬 헌터]는 KPOP에 관심많은 10~30대의 외국인 여행자 대상의 최신 유행하는 KPOP주제의 “케이팝 데몬 헌터스”컨셉의 의성 배경의 텍스트 어드벤처 게임을 제공한다.

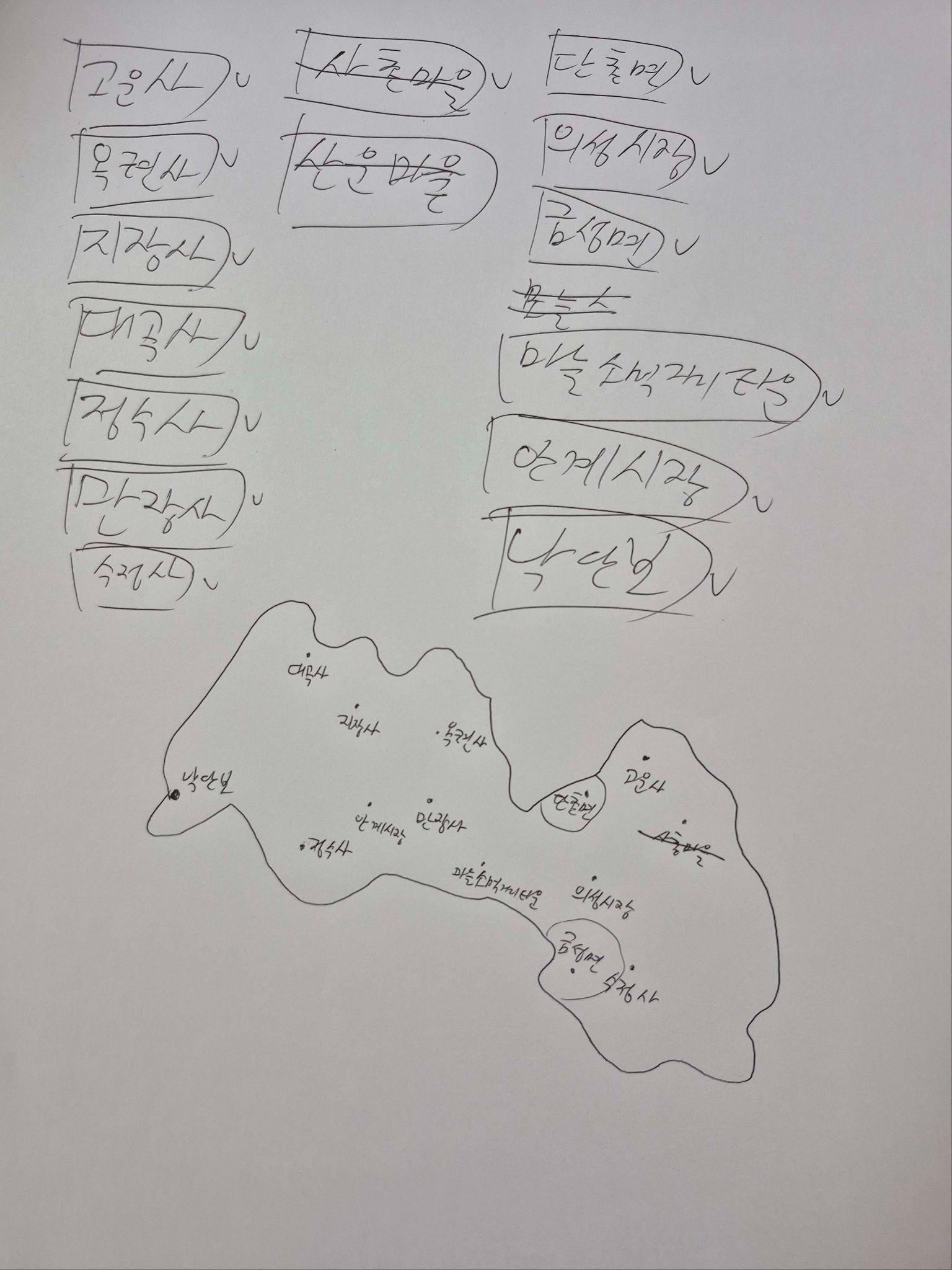
스마트워크시스템

* 프로젝트 관리: 깃허브
* 파일 관리: 구글 드라이브
* 사내문서 시스템: 디스코드
* 메신저: 디스코드
* 라이트닝 데모
  + 서울2033
  + 모험가 이야기
  + 언더월드 오피스

프로젝트 설계

* 서비스 플로우 
* 스토리보드
* UI 설계

아이디어 도출 자료



게임 구조

* 상태(Statement)
  + 체력
  + 정신력
  + 정화도
  + 현재 위치
  + 현재 기분
  + 아이템 목록
  + 직업(난이도)
  + 선택지
  + 히스토리
  + 맵
  + 요괴
  + 플레이어 기억
  + 마스터 기억
* 프론트로 보이는 정보
  + 체력 - health: int
  + 정신력 - sanity: int
  + 정화도 - purification: int
  + 현재 위치 - current\_location: str
  + 현재 기분 - current\_mood: str
  + 아이템들 - items: str
  + 직업 - job: str
  + 선택지 - choices: str[]
  + 히스토리 - {“type”: “narration/dialogue”, “text”: str}[]
* 프론트로 보이지 않는 정보
  + 맵: str
  + 요괴: str
  + 플레이어 기억: player\_lore:str
  + 마스터 기억: master\_lore: str
* 구현 방식: 의성의 여러 절에 요괴들을 배치시키고, 사용자는 그 요괴들을 퇴치한다. 이 때, 할 수 있는 행동과 전체적인 진행들을 LLM을 에이전틱하게 사용해서 구현한다.
* 전체적인 스토리: 불타버린 고운사를 요괴들이 장악해버렸다. 플레이어는 각 장소의 요괴들을 퇴지하며 고운사에 자리잡은 **업화귀**를 퇴치해야 한다.
* 진행 방식: 고운사 내의 여러 장소들을 이동해가며 요괴들을 아이템을 모으고 요괴를 퇴치하며 업화귀를 약화시켜서 퇴치하여 고운사를 해방한다.
* 난이도
  + 일반인: 어려움 - 아무런 능력이 없기에 요괴를 상대하기 어렵다.
  + 스님: 일반 - 스님
  + 무당: 쉬움 - 신내림을 받았기에 신이 도움을 준다.
* 요괴

| 이름 | 티어 | 필요 아이템 | 스님의 대안책 | 무당의 대안책 | 대안책 가중치 |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 빛, 모래 | 염불 | 신탁 | 1.2 |
| 어둑시니 | 1 | 거울, 쑥 | 염불 | 신탁 | 1.0 |
| 달걀귀신 | 1 | 위패 | 반야 | 신탁 | 0.9 |
| 수귀 | 1 | 소금, 바가지 | 지장 | 신탁 | 1.2 |
| 목도깨비 | 1 | 쇠못 | 계율 | 신탁 | 0.9 |
|  | 2 | 밭, 운판 | 염불 | 신탁 | 1.1 |
| 가위눌림 압귀 | 2 | 방울, 북, 소금 | 염불 | 신탁 | 1.0 |
|  | 2 | 쑥, 팥, 소금 | 염불 | 신탁 | 0.8 |
| 성주신 그림자 | 2 | 금줄, 쌀, 소금 | 염불 | 신탁 | 0.9 |
|  | 2 | 칠성, 성좌 | 칠성주 | 신탁 | 0.8 |
|  | 2 | 참회문, 물 | 참회 | 신탁 | 0.9 |
| 장산범 | 3 | 금줄, 징 | 염불 | 신탁 | 0.9 |
| 구미호 | 3 | 호랑이 | 염불 | 신탁 | 0.7 |
| 산신 분노 | 3 | 공양, 향, 종 | 염불 | 신탁 | 0.9 |
|  | 3 | 종353 | 염불 | 신탁 | 0.8 |
|  | 4 | 사물4, 방위4, 경전 | 염불 | 신탁 | 0.8 |

* 요괴 사진

| 이름 | 사진 |
| --- | --- |
| 캣고스트(성불못함?) |  |
| 어둑시니 |  |
| 달걀귀신 |  |
| 수귀 |  |
| 목도깨비 |  |
|  |  |
| 가위눌림 압귀 |  |
| 처녀귀신 |  |
|  |  |
|  |  |
|  |  |
| 장산범 |  |
| 구미호 |  |
|  |  |
|  |  |
|  |  |

* 아이템

| 이름 | 태그 |
| --- | --- |
| 청사초롱 | 빛 |
| 모래/재 | 모래 |
| 소금 | 소금 |
| 나무바가지 | 바가지 |
| 팥 | 팥 |
| 쑥 | 쑥 |
| 손거울 | 거울 |
| 방울 | 방울 |
| 작은북 | 북 |
| 징 | 징 |
| 붉은끈 | 금줄 |
| 짚 | 금줄 |
| 호랑이부적 | 호랑이 |
| 위패세트 | 위패 |
| 촛불7개 | 칠성 |
| 상좌도표 | 성좌 |
| 향 | 향 |
| 공양꾸러미 | 공양 |
| 물바가지 | 물 |
| 솥뚜껑 | 솥 |
| 경전/다라니 | 경전 |
| 사물접근 | 법고, 운판, 목어, 범종 |
| 방위부적4 | 방위4 |
| 종열쇠 | 종, 종353 |

* 요괴 퇴치 스타일: 각 요괴마다 퇴치하는데에 필요한 고유의 아이템들이 있으며, 가지고 있는 아이템 중에서 필요한 태그를 가지고 있다면 활성화된다. 그리고 유저는 각종 요괴의 퇴치법을 알고있어야 퇴치를 시도할 수 있으며, 스님과 무당은 대체 행동이 선택지에 제공될 수 있다. 각 요괴에 대한 지식 수준이 있으며, 지식 수준이 요괴의 티어와 같을 때부터 퇴치 선택지가 제공된다. 요괴가 없는 장소에서의 조사나 npc, 요괴와의 대화를 통해서 지식을 올릴 수도 있다. 또는 가지고 있는 아이템을 재활용하는 식으로 요괴 퇴치 시도를 할 수 있다.

맵 구조

{

"nodes": [

{

"id": 1,

"name": "일주문"

},

{

"id": 2,

"name": "천왕문"

},

{

"id": 3,

"name": "고불전"

},

{

"id": 4,

"name": "가운루"

},

{

"id": 5,

"name": "범종각"

},

{

"id": 6,

"name": "대웅보전"

},

{

"id": 7,

"name": "약사전"

},

{

"id": 8,

"name": "아거각"

},

{

"id": 9,

"name": "적묵당"

},

{

"id": 10,

"name": "명부전"

},

{

"id": 11,

"name": "삼성각"

},

{

"id": 12,

"name": "연지암"

},

{

"id": 13,

"name": "연수전"

},

{

"id": 14,

"name": "고운대암"

},

{

"id": 15,

"name": "종무소"

},

{

"id": 16,

"name": "우화루"

},

{

"id": 17,

"name": "극락전"

},

{

"id": 18,

"name": "만덕당"

},

{

"id": 19,

"name": "후원"

},

{

"id": 20,

"name": "무설전"

},

{

"id": 21,

"name": "열반당"

},

{

"id": 22,

"name": "화장실"

},

{

"id": 23,

"name": "공양간"

},

{

"id": 24,

"name": "화엄문화템플관"

},

{

"id": 25,

"name": "화엄승가대학원"

},

{

"id": 26,

"name": "나한전"

},

{

"id": 27,

"name": "요사"

},

{

"id": 28,

"name": "선원"

},

{

"id": 29,

"name": "고운사삼층석탑"

},

{

"id": 30,

"name": "용왕당"

}

],

"edges": [

{

"source": 1,

"target": 2

},

{

"source": 1,

"target": 25

},

{

"source": 1,

"target": 24

},

{

"source": 1,

"target": 22

},

{

"source": 2,

"target": 3

},

{

"source": 3,

"target": 4

},

{

"source": 4,

"target": 5

},

{

"source": 4,

"target": 16

},

{

"source": 5,

"target": 6

},

{

"source": 5,

"target": 15

},

{

"source": 5,

"target": 16

},

{

"source": 6,

"target": 7

},

{

"source": 6,

"target": 29

},

{

"source": 7,

"target": 8

},

{

"source": 8,

"target": 9

},

{

"source": 9,

"target": 10

},

{

"source": 10,

"target": 11

},

{

"source": 11,

"target": 12

},

{

"source": 12,

"target": 13

},

{

"source": 13,

"target": 14

},

{

"source": 14,

"target": 15

},

{

"source": 14,

"target": 17

},

{

"source": 15,

"target": 16

},

{

"source": 15,

"target": 17

},

{

"source": 15,

"target": 18

},

{

"source": 16,

"target": 18

},

{

"source": 17,

"target": 18

},

{

"source": 18,

"target": 19

},

{

"source": 19,

"target": 20

},

{

"source": 20,

"target": 21

},

{

"source": 21,

"target": 22

},

{

"source": 22,

"target": 23

},

{

"source": 22,

"target": 30

},

{

"source": 23,

"target": 30

},

{

"source": 24,

"target": 25

},

{

"source": 26,

"target": 27

},

{

"source": 26,

"target": 29

},

{

"source": 27,

"target": 28

},

{

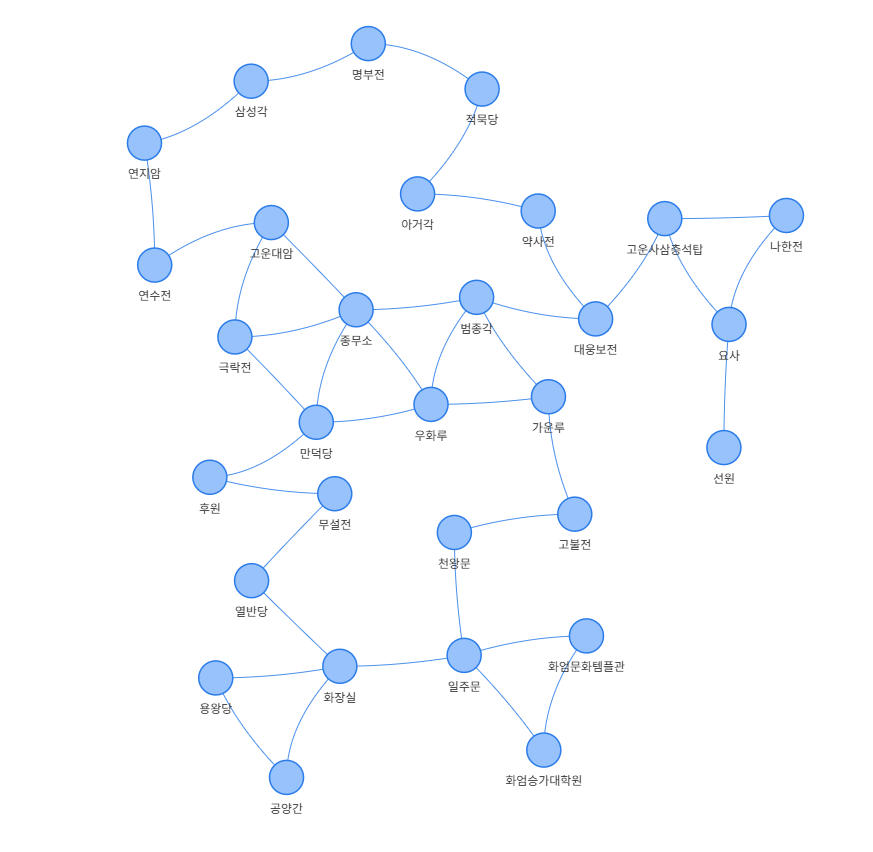
"source": 27,

"target": 29

}

]

}



JSON

* 새로시작 -> GET / newgame?tutorial=true/false
* 이어하기 -> GET / loadgame
* 정보불러오기(선택지를 반영한 모든정보) -> GET / loadinfo
* 선택지 선택 -> POST / selectchoice
* \*리다이렉션없이

| API (엔드포인트) | 설명 | 요청 (Request) JSON | 응답 (Response) JSON |
| --- | --- | --- | --- |
| **GET /newgame** | 튜토리얼 진행 여부에 따라 새로운 게임을 시작합니다. | 없음 | gameSessionId, currentState, storyText, availableChoices를 포함한 객체 |
| **GET /loadgame** | 저장된 게임 세션을 불러옵니다. | 없음 | gameSessionId, currentState, storyText, availableChoices를 포함한 객체 |
| **GET /loadinfo** | 현재 게임 세션의 모든 정보를 조회합니다. | 없음 | gameSessionId, currentState, storyText, availableChoices를 포함한 객체 |
| **POST /selectchoice** | 플레이어가 선택지를 고르면 게임 상태를 업데이트합니다. | gameSessionId와 choice를 포함한 객체 | gameSessionId, currentState, storyText, availableChoices를 포함한 객체 |

개발 마일스톤

1. FastAPI기본 구동 확인
2. 기본 구동 FastAPI gcloud app engine에 배포
3. 배포된 서비스로 외부 접속 가능 확인
4. 프론트 google 로그인 연동 확인
5. 프론트 백엔드 인터페이스 정의
   1. json형식
6. 로그인한 정보로 레디스 데이터베이스 연결
7. 기본적인 에이전트 구조 제작
8. FastAPI에 에이전트 연결

인트로 레퍼런스

* <https://www.youtube.com/watch?v=nVlG1_Q-ouI> (트랄라레오 트랄랄라 영화 인트로)