



GenRoom AI

System Test Plan(STP)

과목명: 졸업 프로젝트
팀원 : 안재민, 김인교, 노을영
담당교수: 유준범
제출일: 2026.05.12

1 개요

1.1. 목적

- 이 문서는 **GenRoom AI** 시스템의 전반적인 기능이 요구사항에 따라 제대로 동작하는지 검증하기 위한 **System Test** 계획을 정의한다.
- 다음 **Layer**들의 기능이 **SRS**에 따라 정확히 작동하는지 확인한다.
 - **UI Layer**
 - **Positioning Layer**
 - **3D-Modeling Layer**
 - **Generation Layer**

1.2 범위

- 사용자의 텍스트 입력 및 후처리(**UI Layer**), 방 배치 model의 입력 및 처리(**Positioning Layer**), 3D 모델링 생성 및 검증(**3D-Modeling Layer**), 최종 검증 및 배치(**Generation Layer**) 등의 전체 기능을 테스트 범위로 한다.
- 각 **Layer**의 모든 주요 기능에 대한 단위 테스트(**Unit Testing**)와 **Layer**간의 통합 테스트(**Integration Testing**)를 수행한다.
- 성능 테스트는 각 **Layer**의 기능 수행 시간등을 측정 대상으로 한다.

1.3 참조 문서

- **Software Requirements Specification - GenRoom AI**
- **IEEE Std 829.1998**

2 테스트 전략

2.1 Unit Testing

- 각 **Layer**의 개별 기능을 독립적으로 검증한다. 외부 의존성(**API**, 모델 등)은 **Mock**을 활용하여 격리하며, 기능별 입력-출력-예외 시나리오를 모두 커버한다.

2.2 Integration Testing

- **Layer** 간 데이터 흐름과 인터페이스를 검증한다. 실제 모듈을 연결하여 데이터가 올바르게 전달되고 변환되는지 확인한다. **Iteration** 및 **Auto-Adjustment** 등 복합 흐름도 포함한다.

2.3 System Testing

- 전체 파이프라인을 **end-to-end**로 검증한다. 실제 사용자 시나리오를 기반으로 정상 및 비정상 입력에 대한 시스템 전체 거동을 확인한다.

2.4 Performance Testing

- 각 **Layer**의 기능 수행 시간을 측정하여 성능 기준 충족 여부를 검증한다. 전체 파이프라인 수행 시간 및 **Iteration** 반복에 따른 성능 영향도 측정한다.

3 테스트 환경

3.1 공통 테스트 환경

구분	항목	상세 내용
HW	개발/테스트 PC	RAM 16GB, GPU VRAM 8GB 이상

SW	운영 체제	Windows 11
	Unity	Unity LTS 6000.3.11F
	Python	python 3.9 이상
	3D Modeling Framework	python-fcl
Network	외부 API	Gemini API 접근 가능 인터넷 환경 및 API Key 보유
Tool	테스트 툴	pytest, Unity Test Framework
ETC	기타	pytorch

4 테스트 케이스
4.1 **Unit Test**
4.1.1 **UI Layer**

ID	Test Item	Procedural	Expected Output	Env need
UL-UT-001	Mock API 응답 확인 성공	.JSON 파일 입력값 전송	.JSON 형태의 출력 반환	Mock API
UL-UT-002	사용자 입력값 전처리	사용자 텍스트 입력 시 프롬프트 자동 작성	표준화된 프롬프트 생성	-
UL-UT-003	예외: 무관한 입력 처리	방 생성과 무관한 텍스트 입력 시도	적절한 오류/예외 메시지 반환	-
UL-UT-004	전처리된 입력값 전달	생성된 프롬프트를 Mock API에 전달	API 호출 성공 응답(200)	Mock API
UL-UT-005	API 응답 대기	API 호출 후 응답 대기 상태 유지	timeout 내 응답 수신 또는 timeout 예외 처리	-
UL-UT-006	API 응답 수신	API로부터 응답 메시지 수신	응답 데이터 메모리 저장 성공	Mock API
UL-UT-007	API 출력 유효성 체크 성공	응답값이 유효한 .JSON 형식 및 유효한 값인 경우	출력값 저장, 후처리 단계 진행	-
UL-UT-008	API 출력 유효성 체크 실패	응답값이 유효하지 않은 형식이거나 비정상 값인 경우	API 재호출 수행(최대 N회 재시도)	-

UL-UT-009	출력값 후처리	유효한 출력값 파싱 및 가구 수 추출	.JSON 출력 파일 생성 및 integer 형태 가구 수 변수 생성	-
-----------	---------	----------------------	--	---

4.1.2 Positioning Layer

ID	Test Item	Procedural	Expected Output	Env need
PL-UT-001	배치 모델 입력 성공	유효한 카테고리 종류/개수를 모델에 입력	모델 정상 수신 로그 반환	Mock Model
PL-UT-002	배치 모델 입력 실패	비정상 카테고리 값 입력 시 모델에서 오류 발생	오류 코드 반환 및 예외 처리 수행	Mock Model
PL-UT-003	배치 모델 출력	모델 추론 완료 후 Bounding Box(위치/회전) 값 저장	각 가구별 BB 좌표 및 회전값 반환	Mock Model
PL-UT-004	출력값 후보정/후처리	가구별 출력값에 대해 바닥·벽 관통 방지 위치 보정	보정된 위치/회전 값 반환	-
PL-UT-005	유효성 검증 성공	collision=False 반환	해당 가구별 위치 값 다음 레이어로 전송	-
PL-UT-006	유효성 검증 실패	collision=True 반환 시	iteration 수행, 배치 모델 재입력	-
PL-UT-007	단일 가구 Bounding Box 생성 검증	단일 가구의 위치/회전 값 입력하여 BB 생성	올바른 3D BB 좌표(min/max) 반환	-
PL-UT-008	두 가구 간 충돌 없음 판정	충분한 거리를 둔 두 가구 BB 입력	collision=False 반환	-
PL-UT-009	두 가구 간 충돌 발생 판정	동일 좌표에 겹치는 두 가구 BB 입력	collision=True 반환, 충돌 가구 ID 포함	-
PL-UT-010	BB 경계 면 접촉(edge) 판정	두 가구 BB가 정확히 면 접촉(겹침=0)하는 값 입력	collision=False 반환 (접촉은 충돌로 간주하지 않음)	-
PL-UT-011	바닥 관통 판정	가구 BB 하단 y좌표가 바닥(y=0) 미만인 값 입력	바닥 관통 오류 반환, 해당 가구 ID 포함	-
PL-UT-012	벽 관통 판정	가구 BB가 방 경계(벽) 외부로 벗어나는 값 입력	벽 관통 오류 반환, 해당 축/방향 정보 포함	-
PL-UT-013	회전 적용 BB 충돌 판정 (OBB)	회전각이 적용된 가구 BB 충돌 판정 수행	회전 고려 시 충돌 여부 정확히 판정	-

PL-UT-014	빈 입력 예외 처리	가구 목록이 비어 있는 상태로 충돌 판정 호출	빈 결과 또는 적절한 예외 반환, crash 없음	-
-----------	------------	---------------------------	-----------------------------	---

4.1.3 3D-Modeling Layer

ID	Test Item	Procedural	Expected Output	Env need
ML-UT-001	프롬프트 전처리	UI Layer 카테고리 + Positioning Layer 가구 수신, 3D 모델 생성용 프롬프트 생성	각 가구별 프롬프트 생성 완료	-
ML-UT-002	3D 모델 생성 요청	생성된 프롬프트를 Mock 3D 모델에 입력	모델 생성 작업 시작 응답	Mock Model
ML-UT-003	3D 모델 생성 성공	모델링 모델이 유효한 3D 메시 파일 반환	유효한 3D 모델 파일(.glb/.obj 등) 저장	Mock Model
ML-UT-004	3D 모델 생성 실패	모델링 모델이 오류 반환, 타임아웃 발생, 또는 미지원 카테고리 요청	오류 처리 후 재시도 또는 기본(Default) 모델 대체 로직 수행	Mock Model
ML-UT-005	Rotation/Orientation 검증	생성된 모델의 회전·방향 값이 허용 범위 내인지 검증	유효 범위 내 시 검증 통과, 범위 초과 시 보정값 반환	-
ML-UT-006	Scale 검증	생성된 모델의 스케일이 Bounding Box 크기와 일치하는지 검증	일치 시 통과, 불일치 시 스케일 조정값 반환	-

4.1.4 Generation Layer

ID	Test Item	Procedural	Expected Output	Env need
GL-UT-001	Bounding Box 정보 수신	Positioning Layer로부터 각 가구의 BB 위치/회전 정보 수신	BB 정보 정상 파싱 및 저장 후 대기	-
GL-UT-002	비정상 BB 정보 수신	비정상 값 수신	비정상 값에 대해 배치 스킵	-
GL-UT-003	3D 모델 수신	3D-Modeling Layer로부터 가구 3D 모델 파일 수신	모델 파일 저장 후 대기	-
GL-UT-004	비정상 모델 파일 수신	비정상 값 수신	예외처리 후 기본 모델 배치(정육면체 모양의	-

			default 모델)	
GL-UT-005	BB 개수 - 3D 모델 개수 불일치	BB의 개수와 3D모델 개수의 불일치	매칭되는 BB와 3D-Model끼리의 배치 진행	-

4.2 Integration Test

ID	Test Item	Procedural	Expected Output	Env need	Dependencies
IT-001	UI → Positioning 연동	사용자 입력 텍스트 → Gemini API 처리 → 카테고리/개수 Positioning Layer 전달	Positioning Layer가 유효한 카테고리 종류/개수 수신 확인	Gemini API, 방 배치 모델	UL-UT-008, PL-UT-001
IT-002	Positioning → 3D-Modeling 연동	Positioning Layer 가구 수 + UI Layer 카테고리 → 3D-Modeling Layer 프롬프트 생성	3D-Modeling Layer가 올바른 프롬프트 수신 및 모델 생성 시작	3D 모델링 모델	PL-UT-005, ML-UT-001
IT-003	3D-Modeling → Generation 연동	3D 모델 파일 → Generation Layer 전달 및 Unity 씬 배치	Unity 씬에 모든 가구 모델 정상 배치 확인	Unity 런타임	ML-UT-006, GL-UT-003
IT-004	Positioning → Generation BB 연동	Positioning Layer BB 정보 → Generation Layer 수신 및 모델 배치에 활용	BB 좌표에 따른 정확한 모델 배치 확인	Unity 런타임	PL-UT-005, GL-UT-003
IT-005	전체 파이프라인 통합	사용자 텍스트 입력 → UI → Positioning → 3D-Modeling → Generation → Unity 출력 전 과정 수행	최종 3D 방 배치 결과가 Unity에서 정상 렌더링	전체 환경	IT-001 ~ IT-004
IT-006	Collision 발생 시 iteration 재처리 연동	Positioning Layer에서 collision 발생 → iteration → 재입력 → 유효값 전달까지 전 과정	최종 유효한 BB 값이 Generation Layer에 전달됨	방 배치 모델, Unity	PL-UT-006, IT-004
IT-007	BB 내 모델 배치	각 3D 모델을 해당 BB 좌표/회전에 따라 Unity 씬에 배치	모든 가구 모델이 지정 BB 내 배치 완료	Unity 런타임	GL-UT-001, GL-UT-002
IT-008	배치 유효성 검증 성공	배치 후 각 모델이 BB 내 정상 위치 여부 판정 → 유효	최종 배치 Unity 씬 출력	Unity 런타임	GL-UT-003
IT-009	배치 유효성 검증 실패	배치 후 모델이 BB를 벗어나거나	예외 반환 및 Auto-Adjustment 발동	Unity 런타임	GL-UT-003

		충돌 발생 → 유효하지 않음			
IT-010	Unity 출력 렌더링	최종 배치 완료된 씬을 Unity에서 렌더링	사용자에게 3D 방 배치 결과 시각화 출력	Unity 런타임	GL-UT-004
IT-012	스케일 불일치 시 런타임 보정 연동	에셋 크기가 BB보다 클 경우 Unity 레이어 스케일 다운	모델이 BB 경계를 벗어나지 않게 강제 조정	MOD, Unity	GL-UT-006
IT-013	비동기 에셋 로딩 상태 동기화	대용량 모델 로딩 중 GEN 레이어의 배치 트리거 대기	로딩 완료 전까지 렌더링 스레드 안전 대기	MOD, GEN	ML-UT-003
IT-014	최종 씬 하이라키 및 태그 정합성	생성 객체의 부모-자식 관계 및 태그 설정 확인	Unity Hierarchy 창에 구조화되어 자동 정돈됨	Unity	GL-UT-007
IT-015	API Rate Limit 발생 시 UI 지연 처리	짧은 시간 반복 생성 시 API 제한 발생 상황 연출	UI 대기 시간 안내 및 자동 재개 로직 확인	Network, UI	UL-UT-004

4.3 System Test

ID	Test Item	Input	Expected Output	Env need	Dependencies
ST-001	정상 시나리오 전체 기능 검증	일반적인 방 구성 요청 텍스트 (예: '소파, 침대, 책상이 있는 거실')	올바른 가구 목록 생성, 배치 완료, Unity 3D 씬 렌더링 성공	전체 환경	오류 없이 최종 씬 출력
ST-002	비정상 입력 처리	공백, 특수문자, 의미 없는 텍스트 입력	적절한 오류 메시지 반환 또는 재입력 유도, 시스템 crash 없음	전체 환경	예외 처리 정상 동작
ST-003	최대 가구 수 처리	가구 수가 최대 허용치에 해당하는 입력	모든 가구가 collision 없이 배치 완료	전체 환경	배치 성공 및 Unity 출력
ST-004	반복 실패(iteration limit) 처리	collision 반복 발생이 예상되는 극단적 가구 조합 입력	최대 iteration 초과 시 사용자 알림 또는 부분 결과 반환	전체 환경	무한 루프 없이 종료
ST-005	사용자 시나리오 E2E 검증	실제 사용자 관점의 방 구성 요청 시나리오 3가지 이상 수행	각 시나리오에서 합리적인 3D 방 배치 결과 출력	전체 환경	시나리오별 출력 확인
ST-006	초소형 공간(1x1m) 극한 배치	1x1m 방에 대형 소파 배치 요청	배치 불가 알림 또는 공간에 맞는 축소 배치	전체 환경	공간 제약 인식
ST-007	사용자 중도 취소 및 리소스 해제	생성 중(N%) 취소 버튼 클릭	백그라운드 스레드 중단 및 메모리 반환	UI, System	리소스 점유 초기화

ST-008	최종 결과물 저장 및 재로드 무결성	생성된 씬 JSON 저장 후 재로드	저장 전후 가구 좌표 및 회전값 100% 일치	Unity	데이터 무결성
--------	---------------------	---------------------	---------------------------	-------	---------

4.4 Performance Test

ID	Test Item	Input	Success Criteria	Dependencies
PT-001	배치 모델 입력 -> 출력까지의 수행 시간	Json 형식의 모델 Input Data	5분 이하의 시간	
PT-002	3D Mesh 생성 시간	1개의 Mesh 프롬프트	1분 이하의 시간	
PT-003	다중 3D Mesh 생성 시간	5개 이상의 Mesh 프롬프트	5분 이하의 시간	
PT-004	Unity World 생성 시간	Mesh 파일 및 방 공간 정보에 해당하는 Json 파일	1분 이하의 시간	
PT-005	3D Mesh 데이터 형식	fbx, glb 파일 / 기타 txt 형식 파일	FBX, GLB 형식	
PT-006	전체 시스템 수행 시간	예시 사용자 입력	15분 이하의 시간	

5. 테스트 진행 절차

5.1. 테스트 준비 단계

5.1.1. 테스트 환경 초기화

- 각 모듈 환경 구성 및 부팅 확인
- Unity 등 외부 의존 모듈 연결 확인

5.1.2. 테스트 데이터 세팅

- 로그 수집용 가상 로그 파일 및 샘플 로그 생성 스크립트 준비
- 가상 Input 파일 세팅
- Mock API 구성 및 세팅

5.1.3. 테스트 툴 준비

- Pytest, Unity Test Framework 등 툴의 사전 설치 및 구성

5.2. 테스트 실행 단계

5.2.1. Unit Test

- 각 기능별 모듈 단위에서 독립적인 테스트 수행

5.2.2. Integration Test

- 각 모듈 정상 동작 후 서로 호출 및 반응이 예상대로 이루어지는지 확인

5.2.3. System Test

- 최종 배포와 유사한 환경에서 사용자의 전체 사용 흐름 시나리오를 수행

5.2.4. Performance Test

- 전체 구성 요소의 부하 테스트 수행

5.3. 테스트 결과 기록 방법

5.3.1. 결과 기록 방식

- 각 테스트 케이스별 결과를 Pass/Fail/Skipped 상태로 기록
- 테스트 일지 형식으로 Notion 페이지에 작성

5.3.2. 스크린샷/캡처

- Web UI 기반 오류나 알림 표시 관련 기능은 스크린샷 첨부

5.3.3. 로그 저장

- 각 모듈의 콘솔 로그와 API 응답 로그는 파일로 저장

5.4. 결함 보고 및 수정 절차

5.4.1. 결함 보고서 포맷

항목	내용
오류 아이디	BUG-UL-UT-001
오류 내용	Json 파일 반환 실패
발견 모듈	UI Layer
발생 조건	Json 파일이 존재하지 않을 경우
예상 동작	Json 파일 반환
실제 동작	반환 실패 후 error 코드 반환
심각도	High
로그	Ulerror.log 첨부
테스트자	노을영
상태	Open/Fixed/Closed

5.4.2. 수정 및 재테스트 절차

- 개발자가 수정 후 hotfix 브랜치에 푸시
- 문제가 해결되었을 경우 Closed 아니라면 Reopen 후 반복

6. 테스트 결과 산출물

- 테스트 케이스 문서
- 실행 로그
- 화면 캡처본 및 API 응답
- 테스트 요약 리포트

7. 테스트 승인 기준

- 주요 기능 단위, 통합 테스트 모두 통과
- 성능 기준 만족
- 시스템 테스트 모두 통과