

## 요구사항 정의서

요구사항 ID	요구사항명	기능 ID	기능명	상세 설명	비고
ACC01	계정 관리	FFC-01	구글 계정 로그인	구글 계정 API를 이용하여 로그인할 수 있어야 한다.	
		FBF-01	계정 정보 도메인 생성	식별 가능한 사용자 정보를 저장하기 위한 도메인을 생성한다.	PostgreSQL
		FBF-02	가입 고객 계정 등록	Google 사용자 정보를 DB에 저장한다.	PostgreSQL
		FBF-03	가가입 여부 체크	로그인 시, DB를 확인하여 userID 확인을 통해 가입일 여부를 확인한다.	PostgreSQL
		FFC-02	로그아웃	사용자 토글을 반전하여 로그아웃할 수 있도록 한다. Chrome에서 로그아웃하면 함께 로그아웃되며, 문서 접근 권한을 잃는다.	
		FFC-03	로그인 상태 확인	최초 접속 시 로그인 상태를 확인하여 로그인인 되어 있지 않으면 DB에 등록 요청한다.	PostgreSQL
		FFC-04	사용자 계정으로 구글 Docs 접근	사용자 구글 계정으로 로그인하여 구글 Docs에 접근할 수 있도록 OAuth 기능을 구현한다.	
		FFJ-01	로그인에 따른 팝업창 변형	로그인 시 우측 상단 아이콘 클릭으로 팝업창에서 로그인 여부를 알아볼 수 있도록 구현한다. (예: 사용자 이메일 정보 및 구글 프로필 사진 표시)	
		FFC-05	로그인 상태 유지	웹페이지를 닫아도 크롬 로그인을 통해 로그인을 유지한다.	
EXT01	Log-in Page	FBF-04	회원 탈퇴	사용자의 정보를 DB에서 삭제할 수 있어야 한다.	PostgreSQL
		FFH-01	구글 Log-in 화면 구성	구글 Log-in page 화면을 구성한다.	
		FFC-01	페이지 로그 디자인	Log-in Page 화면에서 사용할 서비스 로고를 디자인 한다.	
		FFH-02	구글 로그인 버튼 구현	구글 계정 로그인인 가능한 로그인 버튼을 구성한다.	
		FFC-06	로그인에 따른 시작 페이지 접속	Log-in Page에서 구글 계정 로그인 후, start-Page로 라우팅 한다. 구글 로그인 상태가 아니라면 Landing-Page로 라우팅한다. 마찬가지로 로그인 되어 있다면 popup-Page로 라우팅 한다.	
		FFC-07	로그인 여부 확인		
		FFH-03	기본 Landing-Page 화면 구성	Landing-Page 화면을 구성한다.	Django Templates
EXT02	Landing Page	FFH-04	"Try it in Google Docs" 버튼 구성	로그인된 상태로 JJACKLETTE Assistance가 Add-in 된 구글 Docs에 접근하는 버튼을 구성한다.	
		FFH-05	런딩 페이지 설명 화면 구성	"Get Started" 버튼 하단에 해당 서비스를 설명하는 Contents 들을 구성합니다.	
		FFH-06	Assistance '사이드 패널' 구성	Chrome 브라우저 환경에서 우측에 사이드 패널 화면을 구성한다.	
		FFH-07	Assistance 'Chat-세션' 컴포넌트 구성	우측에 열린 사이드 패널에 'Chat-세션' 컴포넌트를 구성한다.	
		FFJ-02	Chat-세션' 채팅 기능 구현	사용자의 프롬프트에 따라 JJACKLETTE AI 모델의 답변을 표시하는 채팅 기능을 구현한다.	sLLM
		FFJ-03	Chat-세션'에서 생성된 문장 문서에 반영	생성된 문장에 대해 삽입, 복사 2가지 기능을 각각의 버튼을 눌러 적용할 수 있도록 한다.	Google Docs API 연동
		FFJ-04	Chat-세션'에서 생성된 컨텐츠의 출처 제공	생성된 문장에 대해 참고한 출처를 사용자가 확인할 수 있도록 버튼을 눌러 출처의 목록을 반환한다.	Qdrant (RAG 검색 출처), PostgreSQL (문서 메타데이터)
		FFJ-05	업로드된 자료 목록 '미디어 섀드' 출력	업로드 된 자료들을 아이콘 및 제목으로 시각화하여 보여준다. 오른쪽에 해당 문서를 적용할지 여부를 선택할 수 있는 토글을 추가	PostgreSQL (메타데이터)
		FFJ-06	검색된 자료 목록 '미디어 섀드' 출력	검색에 사용된 자료에 대한 출처를 미디어 섀드에 출력한다.	Qdrant (RAG 검색 출처), PostgreSQL (문서 메타데이터)
EXT03	Docs Page	FFJ-07	업로드 및 변형 시 시각화	파일이 업로드되고 변환되는 동안 시각화한다.	
		FFJ-08	도움 화면 구현	JJACKLETTE AI 서버에서 생성된 컨텐츠를 반환받기까지 로딩 화면에 프로그레스 게이지와 함께 간단한 쿼츠를 출력한다.	
		FBF-05	업로드된 자료 목록 반환	사용자가 초기화하기 전 부여된 업로드 ID에 해당하는 문서를 목록을 반환한다.	PostgreSQL (메타데이터)
		FBF-06	업로드된 자료 목록 저장	업로드된 자료 목록을 사용자, 세션, 검색, 질문 블록 ID와 연결하여 저장한다.	PostgreSQL
		FBF-07	사용자 명칭 저장	사용자가 입력한 명칭을 사용자, 세션, 질문 블록 ID와 연결하여 저장한다.	PostgreSQL
		FBF-08	질문 세션 초기화	사용자가 입력한 내용을 초기화 한다.	
		FBF-09	업로드 목록 초기화	사용자가 업로드한 파일을 초기화 한다.	
		FAC-01	텍스트 작성 Agent 구현	초안 생성, 단락 생성 Process에서 활용될 문서 자료 Vector DB, 검색 자료를 기반으로 텍스트를 생성하는 Agent를 구현합니다.	SLLM, Qdrant
		FAC-02	데이터 분석 Agent 구현	업로드 기반 데이터 시각화, 웹 기반 데이터 시각화 Process에서 활용될 다양한 데이터를 기반으로 그래프 시각화 코드를 생성	
JLT01	JJACKLETTE Model	FAC-03	초안 생성 Process 구현	특정 주제로 보고서 초안을 생성하는 Application Process를 구현합니다.	
		FAC-04	단락 생성 Process 구현	업로드 자료와 검색된 자료를 기반으로 텍스트를 생성하는 RAG Pipe-line을 구현합니다.	Qdrant (RAG 파이프라인)
		FAC-05	웹 기반 데이터 시각화 Process 구현	사용자가 원하는 시각화를 위한 자료를 웹에서 수집하여 시각화 하는 Process를 구현합니다.	
		FAC-06	업로드 기반 데이터 시각화 Process 구현	사용자가 업로드한 자료를 분석하여 다양한 시각에서 시각화하는 Process를 구현합니다.	
		FAC-07	요약/확장 Process 구현	사용자가 지정할 텍스트를 중심 내용을 예치지 않고 요약하거나 확장하는 Process를 구현합니다.	
		FAC-08	JJACKLETTE AI Application 프로세스 점검/평가	JJACKLETTE AI Application 프로세스 전 과정을 점검하고, Inference Time 등 개선 사항을 도출하여 해결한다.	
		FAC-09	JJACKLETTE Server와 통합	JJACKLETTE 서버에서 AI 기능을 실행할 수 있도록 통합한다.	FAR-01 (sLLM 파인튜닝)과 연계
		FAR-01	요약/확장 sLLM 파인튜닝	주어진 데이터를 기반으로 텍스트 요약, 확장을 수행할 수 있도록 파인튜닝 합니다.	SLLM
		FAR-02	요약/확장 sLLM 성능 테스트	자료들 기반으로 텍스트 요약, 확장 과정에서 지정 도메인의 내용을 잘 포함하는 지 성능을 확인합니다.	
SVR01	File upload	FBF-10	PDF 문서 업로드	사용자가 업로드한 PDF 파일의 이름과 업로드 시간을 PostgreSQL에 저장한다.	
		FBF-11	CSV 문서 업로드	사용자가 업로드한 파일을 저장하고 남기고 이름과 시간을 PostgreSQL에 저장한다.	
		FBF-12	PDF 문서 벡터 변환	서버에 업로드된 사용자의 파일을 벡터로 변환하고 Qdrant에 저장하여 RAG에 사용할 수 있는 형태로 저장한다.	
		FBF-13	파일 업로드 내구권	파일 업로드 아이콘을 클릭하여 파일을 업로드 합니다.	
SVR03	Database	FBF-14	벡터 DB 구현	AWS에 Qdrant를 구현하여 벡터 데이터베이스 저장소로 사용한다.	
		FAD-01	비정형 파일 저장	벡체인 서버인 EC2 인스턴스에 PDF, CSV 등의 임시 저장을 위한 비정형 파일 저장소로 사용한다.	
		FBF-16	메인 DB 및 대화내용 저장	AWS RDS에 PostgreSQL DB를 구현하여 사용자 계정 정보 및 구글 문서에 대한 검색 기록 DB로 사용한다.	
NET01	Network	FBN-01	HTTPS 통신	SSL인증서를 통해 통신 내용을 암호화하여 보안을 강화합니다.	
		FBN-02	요청 대기 제한	서버의 요청 부하를 방지하기 위해 요청이 1000건이 넘어가지 않고, 모델의 '간' 답변 시간에 대응, 버퍼가 차면 대대	
		FBN-03	서버 과부하 방지	서버가 과부하 되는 것을 방지하기 위해 Gunicorn과 uvicorn을 이용해 로드밸런싱을 합니다.	AWS ELB (Application Load Balancer)
JLT03	JJACKLETTE Function	FFH-08	JJACKLETTE 비전 및 핵심 기능 확인	웹사이트 내 'About Us' 또는 유사한 섹션을 통해 JJACKLETTE의 비전, 목표, 핵심 기능 및 가치를 사용자에게 명확히 전달한다.	
		FFJ-09	Asset Library를 통한 디자인 요소 브라우징	사용자는 'Asset Library' 창을 통해 JJACKLETTE가 제공하거나 내부적으로 학습된 다양한 디자인 요소(예: 자동차 부품, 스타일	EC2 로컬 스토리지 (해설 파일), PostgreSQL (메타데이터 및 분류)
		FFJ-10	생성 이미지 기반 영상/3D 시각화 (Car Dashboard)	Insight & Trends' 창에서 생성된 2D 이미지를 활용하여 짧은 영상 클립을 생성하고, 3D 생성 기능을 통해 360도 회전되는 자동차	SV-4D 2.0, Stable-Point-Aware-3D 모델 활용
		FFJ-11	시각/리뷰 기반 트렌드 분석 확인	Insight & Trends' 창을 통해 시장 데이터 및 사용자 리뷰를 기반으로 한 자동차 디자인 트렌드 분석 정보를 제공한다.	Gemma SLLM (파인튜닝) 및 데이터 분석 Agent (FAC-02) 활용
		FFJ-12	Prototype Lab 활용	Prototype Lab'은 사용자가 첫발을 활용하여 디자인 아이디어를 구체화하고, 3D 이미지 및 영상을 직접 생성하거나 수정하는 핵심	Gemma SLLM, Stable Diffusion 3.5, Stable-Point-Aware-3D, SV-4D 2.0 통합 활용. (REQ-018, REQ-019, REQ-010, REQ-011 기능 통합)
		FFJ-13	My Work를 통한 세션별(Chat History) 저장	My Work' 창을 통해 사용자의 모든 작업 세션(첫발 대화 기록, 생성/수정된 디자인 시안 등)이 개별적으로 저장되고, 사용자는 이	PostgreSQL (세션 및 기록), EC2 로컬 스토리지 (관련 파일)
		FFH-09	프롬프트 작성 가이드 및 사용법 제공 (Tutorials)	Tutorials' 창을 통해 JJACKLETTE 사용법, 프롬프트 작성 가이드, 각 기능에 대한 상세 설명(How it works) 등 사용자가 플랫폼을	FAR-01 (sLLM 파인튜닝)과 연계