



AI Software Engineer, 정예울입니다.

다양한 개발 경험

LINEBank Japan 코어뱅킹 개발, AI 애플리케이션 개발, DevOps 경험 보유

자동화된 프로세스 구축

지속 가능한 API 문서 자동화, 테스트 데이터 생성 최적화 경험

협업 및 커뮤니케이션

이슈 관리 능숙, 사내 포럼 52개 글 작성, 개인 기술블로그 운영을 통한 지식 공유

Mobile	010-3595-9818
E-mail	papooo.dev@gmail.com
Github	github.com/papooo-dev
LinkedIn	https://www.linkedin.com/in/yewool-jeong-091719342
Blog	now : https://papooo-dev.github.io/ previous : https://creamilk88.tistory.com/

Skills

구분	Skill
AI	Langchain, Prompt Engineering, LLM, Memory, RAG
Programming Languages	Java, Python, JavaScript
Framework / Library	SpringBoot, JPA, React, FastAPI, Flask
Tools & DevOps	Docker, AWS, Git, Gitlab, CI/CD
Databases	Oracle, MySQL, PostgreSQL

Work Activities

Bankware Global (총 3년 10개월)

2023.07 ~ 현재 SW 연구소 주임 (1년 8개월)

- AI 기반 어플리케이션 개발

- AI를 활용하여 BDD 개발 방식의 자동화 도구와 대량의 테스트 데이터 생성 프로그램을 성공적으로 개발하였습니다. 이를 통해 개발 효율성과 데이터 품질을 크게 향상시켰습니다.

- **지속적이고 자동화된 프로세스 구축**

- Java POJO 기반의 API 명세서 자동화 프로세스를 구축하여 문서의 지속 가능성과 최신화를 보장하였습니다.
- 파이프라인 구조 개선을 통해 배포 속도를 2.5배 향상시켰습니다.

- **지식 공유 활동**

- 10회 차로 구성된 LangChain 사내 강의를 통해 기본 이론부터 실습까지 진행하며, 사내 AI 제품 개발 역량을 강화하였습니다.
- 회사 포럼에 AI 트렌드, 사용법, 개발 관련 포스팅을 한 해 동안 51개 이상 작성하여 사내 지식 공유에 활발히 기여하였습니다.

Impactful Experiences & Insights

- **LLM을 활용한 테스트 데이터 생성 최적화 경험**

- 초기 문제점: 필드별 LLM 호출로 인해 비용 과다와 속도 저하 발생.
- 해결 방안: LLM으로 데이터 구조 분석 후 데이터 생성이 가능한 Python 코드를 생성, 초기 1회 LLM 호출 뒤 이후 LLM 호출 없이 데이터를 무한 확장 가능하도록 설계. faker 모듈을 활용해 실제와 유사한 고품질 데이터 생성.
- 성과: 1,000,000건의 mock 데이터를 72.38초 만에 생성, 데이터 생성 비용과 속도를 획기적으로 최적화.

2021.03 ~ 2022.05 Software Engineer (2년 2개월)

- **LINE Bank Japan 코어뱅킹 시스템 개발**

- 고객업무를 담당하여 비대면 고객가입 및 고객 관리 시스템을 개발하였습니다.

- **테스트 자동화 프로그램 운영**

- 고객업무와 타 팀 업무를 연계하여, 테스트 자동화 프로그램을 설계 및 운영하여 시스템 안정성을 확보하였습니다.

Projects

API Documentation Automation: 지속 가능한 API 문서 자동화 프로세스 구축

- 기간: 2025/01 (1M)
- 프로그램 개요: Java 기반 프로젝트에서 코드 내 Javadoc을 사용하여 Docusaurus 기반 Markdown 문서를 자동 생성하고, CI/CD 파이프라인을 통해 문서를 자동 배포하는 프로세스 구축
- 기술 스택: `Java`, `Javadoc`, `Docusaurus`, `JavaParser`, `CI/CD`
- 인력구성 및 기여도: SWE(1명) / 기여도 100%
- 주요 업무 및 상세 역할:
 - JavaParser 라이브러리를 사용해 Javadoc 주석을 분석하여 원하는 문서 포맷의 Markdown로 변환하는 프로그램 개발
 - CI/CD 파이프라인을 통해 Javadoc 변경 시 문서 자동 생성 및 웹사이트 환경으로 자동 배포환경 구축

- 주요 성과:
 - Javadoc 작성만으로 API 명세서를 자동화하여 최신 상태 유지 및 문서 관리 효율성 증대
 - Docusaurus Markdown 문서 생성 프로그램 개발로 반복 작업 최소화
 - CI/CD 파이프라인 구축으로 문서 최신화 및 배포 자동화
 - 코드와 문서의 불일치를 최소화하여 신뢰도 높은 문서 제공

OiBDD : AI 기반 BDD 자동화 솔루션

- 기간: 2024/08 - 2024/12 (4M)
- 프로그램 개요: BDD의 디스커버리 워크숍 과정을 AI로 자동화하여 테스트 자동화 프로세스를 구축하는 프로그램
- 기술 스택: LangChain, RAG, Python, Flask, FastAPI, PostgreSQL, React, CICD, Docker, AWS E2C
- 인력구성 및 기여도: SWE(2명), PO(1명) / 기여도 50%
- 주요 업무 및 상세 역할:
 - Langchain을 활용하여 LLM 연동을 통한 Example Mapping 생성과 Feature파일 생성 상세화 단계 구축
 - RAG 파이프라인을 구축하여 정제된 Feature 파일을 생성할 수 있도록 적용
 - RESTful API 설계 및 개발 및 DB 모델 설계
 - React 화면 설계 및 개발
- 주요 성과:
 - API 코어 서버와 클라이언트 서버를 분리: 다양한 클라이언트(웹, 플러그인) 지원 가능한 확장성 증대, 코어 기능(프롬프트, 토큰)을 보호
 - AI 응답의 품질을 높이기 위해 Memory와 RAG 기술을 적용하여 보다 정확하고 일관된 결과를 제공
 - FastAPI의 Background Task 기능을 활용하여 LLM의 응답 대기 중에도 실시간으로 진행 상태를 조회
 - 버전 및 용도에 따른 프롬프트 관리 설계를 통해 UI에서의 유연한 프롬프트 수정 및 적용이 가능하도록 구현
 - 파일 변경 시 자동으로 임베딩 시스템을 업데이트하여 최신 상태를 유지
 - Docker 기반의 Gitlab CICD 배포 시스템을 구축하여 AWS E2C 인스턴스에 서버를 안정적으로 구성

Test Data Generator : AI를 활용하여 고품질의 테스트 데이터를 신속하게 생성하는 프로그램

- 기간: 2024/07 - 2024/08 (1.5M)
- 프로그램 개요: AI를 활용하여 다양한 형식의 테스트 데이터를 자동으로 생성하는 솔루션입니다. 데이터 구조와 관계를 인식하여 사용자가 정의한 자연어 규칙에 따라 mock 데이터를 생성하며, 다양한 출력 형식으로 제공하여 테스트 데이터 생성의 효율성을 극대화합니다.
- 기술 스택: LangChain, Prompt Engineering, Python, Flask, jquery
- 인력구성 및 기여도: SWE(2명) / 기여도 60%
- 주요 업무 및 상세 역할:
 - 최적화된 테스트데이터 생성 코드 템플릿을 작성하여, 다양한 데이터 생성 규칙에 따른 정확한 테스트데이터를 무한 생성 가능한 기반 코드 템플릿을 작성

- LLM과 연동하여 테스트데이터 생성 코드 생성 과정의 프롬프트 작성, Langchain을 통한 LLM 연동 과정 개발
- RESTful API 설계 및 개발
- jQuery를 이용한 화면 개발
- 주요 성과
 - **빠르고 저렴한 테스트 데이터 생성:** 데이터 생성 규칙을 기반으로 무한의 데이터를 생성할 수 있는 Python 코드를 AI를 통해 자동 생성하여, AI 호출 비용을 최소화하고 빠른 속도로 대량의 데이터를 생성합니다. LLM 사용 비용이 최초 1번만 소요되고, AI보다 프로그램 수행 속도가 훨씬 빠르기 때문에 매우 빠르게 대량의 데이터를 생성할 수 있습니다.
 - **코드 생성 템플릿 최적화:** LLM을 활용한 Python 코드 생성 시, 사전에 준비된 모듈과 테스트 데이터 규칙을 적용하여 코드 템플릿을 최적화함으로써 LLM의 할루시네이션으로 인한 오류 발생을 최소화하고 코드의 안정성을 강화하였습니다.
 - **고성능 데이터 처리:** 테이블 1건(컬럼 수:22개)에 대해 1,000,000건의 mock data를 72.38초만에 생성할 수 있는 성능을 제공합니다.

CS Portal : 유지보수 고객사를 위한 이슈 트래킹 웹 사이트

- 기간: 2024/01 - 2024/04 (4M)
- 프로그램 개요:

CS Portal은 MA(유지보수 계약) 고객사를 위한 이슈 트래킹 웹 애플리케이션으로, GitLab Service Desk의 한계점을 개선하여 프로젝트별 이슈를 중앙 집중적으로 관리할 수 있도록 지원합니다. 고객사와 기술 담당자 모두에게 효율적인 이슈 관리와 히스토리 추적이 가능한 통합 플랫폼을 제공합니다.
- 기술 스택: `Java17`, `Spring boot3`, `Spring Security`, `Spring WebFlux`, `JPA`, `Postgresql`, `React(Material-UI)`, `GitLab Service Desk API`, `Multi Module Architecture`, `Reactive Programming`, `CICD`, `AWS E2C`
- 인력구성 및 기여도: SWE(1명) / 기여도 100%
- 주요 업무 및 상세 역할:
 - RESTful API 설계 및 개발 및 DB 모델 설계
 - Java 리액티브 프로그래밍을 통한 백엔드 개발: Spring WebFlux를 활용하여, 데이터 흐름 관리 및 비동기 작업 최적화를 통해 성능 개선.
 - Gitlab API 연동: Gitlab의 Service Desk 및 Issue 연동 작업
 - React 화면 설계 및 개발
- 주요 성과:
 - **효율적인 이슈 관리 제공:** 고객사가 기존 이메일을 통한 이슈 관리의 불편을 해소. CS Portal을 통해 이슈 등록, 조회할 수 있도록 개선.
 - **프로젝트 관리 효율화:** CS Portal에서 프로젝트별 이슈를 중앙 집중식으로 관리 가능하게 하여 이슈 히스토리 및 현재 현황 공유가 가능하여 팀 협업 효율성 증대.
 - **멀티 모듈 설계:** 백엔드 모듈을 분리하여 유지보수성과 확장성을 강화.

Simulator Version 2 : 대외 연계 시뮬레이터 솔루션 버전 업 진행

- 기간: 2023/08 - 2023/12 (5M)

- 프로그램 개요: 기존 대외 연계 인터페이스 시뮬레이션 솔루션에 응답 다양화, REST 시뮬레이터 등 추가 기능 개발
- 기술 스택: `Java8`, `Spring Boot2`, `React`, `REST 통신(HTTP)`, `TCP 통신`, `MySQL`
- 인력구성 및 기여도: SWE(2명) / 기여도 30%
- 주요 업무 및 상세 역할:
 - REST API 시뮬레이터 기능 개발: OpenAPI 명세서를 파싱하여, REST API 서버 및 클라이언트 시뮬레이터 제공, 대상 기관 정보와 연동하여 인터페이스 거래 테스트 기능
 - 배포 설치 패키징 및 가이드 작성 작업: docker 이미지, jar를 통한 설치 패키징 작업
 - 기존 Angular 기반 소스 코드를 React로 전환: 기존의 Angular 기반 소스 코드를 React로 전환하여 유지보수성과 성능을 개선.
- 주요 성과:
 1. **REST 시뮬레이터 기능 개발:** REST API 거래의 시뮬레이션 기능 추가, 대상 기관 및 인터페이스 정보의 생성, 변경, 삭제 기능 구현.
 2. **대외기관 시뮬레이션 응답의 다양화:** 대외기관 시뮬레이터가 테스트 시 다양한 요청에 맞춰 다채로운 응답을 제공할 수 있도록 기능을 확장하여, 보다 현실적인 테스트 환경을 구현
 3. **FEP 시뮬레이터 확장:** Core System에서 대외 인터페이스 시뮬레이션 거래를 직접 진행(전문 변환, HTTP 엔드포인트 제공)
 - HTTP 엔드포인트 제공 및 통신 전문 포맷 정의.
 - 입력/출력 전문 변환 및 응답 생성 로직 구현.
 - 동기/비동기 응답 전송 기능 추가.
- 배운 점:
 1. 기존의 복잡하고 대규모의 소스 코드를 분석하며 코드 이해 및 유지보수 능력 향상.
 2. TCP 및 REST API 거래 처리에 대한 심층적인 이해와 개발 경험 축적.
 3. REST API HTTP 및 HTTPS 통신과 전문 변환에 대한 기술적 역량 강화.

LINE Bank Japan 코어뱅킹 프로젝트

- 기간: 2021/04 - 2022/04 (13M)
- 프로그램 개요:
 - LINE Bank Japan의 코어뱅킹 시스템 개발 프로젝트
- 기술 스택: `Java`, `Node.js`, `Oracle`, `Git`, `GitLab`, `AngularJS`, `BXM Framework(SpringBoot 기반)`
- 고객사: LINE Bank Japan 및 Korea (운용인원: 약 100여명)
- 인력구성 및 기여도: BwG(자사) 인력(80명), 고객팀 인력(8명) / 팀 내 기여도 30%
- 주요 업무 및 상세 역할:
 - 고객 업무계좌 조회, 비대면 법인고객 가입, 채널 연동 개인 고객 정보 검증 등 주요 고객 업무를 구현
 - 자동화 통합 테스트 프로그램을 설계 및 운영.
- 주요 성과
 - 고객 코어 뱅킹 업무 개발:

- 가장 많이 사용되어 성능이 중요한 고객 전체계좌 조회 및 상태 조회 API 개발 시 성능 향상을 위해 SQL 튜닝을 통해 안정적 운영 기여
- 자동화 통합 테스트 프로그램 관리:
 - 테스트 시나리오를 프로그램으로 구현하여 테스트 효율성 및 정확도 향상.
 - 매일 자동으로 통합 테스트를 진행하고 결과를 레포트로 출력, 수정된 프로그램의 영향도와 버그를 신속히 식별.
 - 테스트 프로그램 모듈화 및 공통화 작업을 통해 유지보수성 및 확장성 강화.
- 프로젝트 안정화 기여:
 - 자동화 테스트 도입을 통해 주요 릴리즈 및 기능 추가 시 품질 보증을 강화.
 - 팀 간 협업 효율성 증가로 개발 속도 향상.

Education

경희대학교

국어국문학과 학사, 문화관광콘텐츠학과 복수전공

2013.03 ~ 2019.02

Personal Activities

AWS GameDay 참여

2024.12.20

- 개요: AWS 클라우드 서비스를 활용한 문제 해결 실습 이벤트.
- 주요 성과:
 - Storage 부문 1등 달성.
 - AWS 서비스 활용 능력 향상 및 협업과 AI 활용 문제 해결 능력 강화.

사내 LangChain 강의 진행

2024/06 - 2024/07 (1.5M, 10회)

- 개요: LangChain 기초 및 응용 스터디 자료를 준비하고 강의를 진행하여, 사내 AI 활용 역량을 강화
- 주요 성과
 - 직원들이 AI를 회사 사업에 창의적으로 적용할 수 있는 아이디어를 발굴
 - 강의 내용을 바탕으로 실제 개발 프로젝트(상품 팩토리 AI, OiBDD)로 이어짐

Recent Interests and Studies

- **AI Trends:** 최신 AI 트렌드를 Medium으로 학습하고, 블로그에 정리해서 지식을 나누는 걸 좋아해요.

- **Generative AI Applications:** AI 어플리케이션 개발자로 성장하기 위해 AI 엔지니어 로드맵을 따라 체계적으로 공부하고 있어요.
- **System Design:** AI 시대에서 가장 중요한 역량이라고 생각해요. 시스템 설계와 다양한 설계 패턴에 대해 배우고 있어요.
- **Deployment:** AWS를 활용한 배포 및 관리 역량을 키우는 데 관심이 있어요.
- **Side Projects:** 제가 실생활에서 필요했던 어플리케이션을 사이드 프로젝트로 개발하면서 실무 경험을 쌓고 있어요.

References

Available upon request (요청 시 제공 가능)

마지막 업데이트 : 2025/02/08