

# 생성형 AI 기술을 이용한 S/W원격 협업개발 개선에 대한 연구

최 기 원\*, 강 동 구\*, 김 진 태\*\*, 조 주 필<sup>o</sup>

## A Study on Improving S/W Remote Collaboration Development Using Generative AI Technology

Kiwon Choi\*, Dong Koo Kang\*, Jin Tae Kim\*\*, Ju Phil Cho<sup>o</sup>

### 요 약

글로벌 IT 분야는 빠르게 발전하고 있으나, 국내 IT 시장은 고급 인력 이탈과 신규 개발 인력 부족으로 인한 인력난을 겪고 있다. 이를 해결하기 위해 생성형 AI 도구의 활용이 주목받고 있으며, 본 연구는 해당 도구를 통해 소프트웨어 개발 산출물 작성, 프로그램 개발, 및 테스트 코드 과정을 자동화하는 방안을 실험하였다. 실험 결과, 요구사항 정의서와 프로그램 사양서 작성은 규격을 준수하며 글로벌 협업 개발이 가능한 수준의 번역 결과를 확인하였으며, 프로그램 개발과 테스트 코드 생성은 로직과 문법에 오류 없는 결과를 얻을 수 있었다. 본 연구는 생성형 AI 도구가 소프트웨어(S/W) 개발 과정의 일부를 자동화하는 데 효과가 있음을 확인하였다.

**키워드** : 인공지능, 딥러닝, 생성형 AI, S/W 원격 개발, ChatGPT

**Key Words** : Artificial Intelligence, Deep Learning, Generative AI, ChatGPT, Off-Shore S/W Development

### ABSTRACT

The global IT industry is advancing at an unprecedented pace, yet the domestic IT market faces significant challenges, particularly a shortage of skilled talent due to the outflow of experienced professionals and limited inflow of new developers. To mitigate this issue, the application of generative AI tools has emerged as a promising solution. This study explores the potential of these tools to automate key aspects of the software development lifecycle, including the generation of deliverables, program development, and test code creation. The results demonstrate that the AI tools effectively adhere to industry standards in producing requirement documents and program specifications, with translation outputs suited for global collaborative development. Additionally, program development and test code generation were executed without any logical or syntactical errors. These findings suggest that generative AI tools can significantly enhance automation in software development, thereby addressing workforce shortages and improving overall process efficiency.

\* First Author : Graduate Program in IT Policy and Management, Soongsil University, dwightchoi@gmail.com

<sup>o</sup> Corresponding Author : Dept. of Integrated IT & Communication Engineering, Kunsan National University, stefano@kunsan.ac.kr, 종신회원

\* Graduate Program in IT Policy and Management, Soongsil University

\*\* (주)파이브텍, 건국대학교 정보통신대학원

논문번호 : 202404-072-E-RN, Received April 21, 2024; Revised June 26, 2024; Accepted September 4, 2024

## I. 서 론

최근 생성형 AI 기술이 등장하면서 기계가 인간의 언어를 이해하고 대량의 요구사항을 수행하는 시도가 급격히 증가하고 있다. 인공지능, 머신러닝, 딥러닝, 그리고 생성형 AI는 인공지능의 하위 기술로 발전해 왔으며, 생성형 AI는 사용자의 요구에 따라 결과를 능동적으로 생성하는 기술로 최근 각광을 받고 있다<sup>[1]</sup>.

생성형 AI의 주요 기술 요소에는 LLM (Large Language Model), GAN (Generative Adversarial Network), VAE (Variational Auto-Encoder) 등이 있다. 생성형 AI의 핵심 기능은 사용자의 질문을 이해하고 분석하여 관련 정보를 스스로 찾아 요약하고 제공하는 것이다. 질문을 구체화하거나 맥락을 추가할수록 더 정확한 답변을 얻을 수 있으며, 이는 사람과 대화하듯이 맥락을 이해하고 답변을 제공한다는 점에서 기존 기술과 차별화되며, 특히 LLM은 거대 언어 데이터를 학습하여 텍스트 기반의 결과를 생성하는 핵심 기술이다 [2-4,19,20].

생성형 AI의 자연어 처리를 위한 핵심 알고리즘은 트랜스포머(Transformer)라는 알고리즘이다. 트랜스포머 알고리즘은 어텐션(Attention)이라는 메커니즘을 사용하는데, 이는 입력 문장과 출력 문장 사이의 관계를 파악하고, 입력 데이터를 되짚어 보면서 어느 입력 데이터에 초점을 맞출지를 파악하고, 핵심이 되는 부분에 집중하여 더 빠르게 훈련하는 방식으로 최근 자연어 처리 분야에 대단한 혁신을 일으켰다<sup>[3,5]</sup>.

최근 IT 업계에서 생성형 AI 도구 활용에 주목하는 이유는 방대한 범위의 데이터 학습으로 특별한 추가 학습 없이 많은 태스크를 수준 높게 수행하는 범용성이 높은 언어모델이라는 점이다. 이로 인해 사람이 작성한 글과 구분하기 어려운 수준의 높은 완성도의 글쓰기가 가능할 뿐만 아니라 문장의 요약, 번역 및 재구성 등이 가능하다. 또한, 복잡한 프로그래밍 언어 대신 간단한 자연어 명령 몇 줄만으로 일반인들도 쉽고 애플리케이션 개발이 가능해졌다<sup>[2,21]</sup>.

본 논문은 S/W 개발 과정의 문제를 해결하기 위해 생성형 AI 기술을 활용하여 기존의 개발 공정 일부를 자동화하고, AI와 협업하여 생산성을 향상시키는 방안을 제시한다. 이를 위해 현재의 오프쇼어(Off-Shore) 개발 방식에 기반하여 원격지에서 수행하는 주요 작업을 생성형 AI로 대체하고 그 결과를 분석하였다. 또한, 개발 생산성 향상을 위한 최적의 생성형 AI 기반 작업 모델을 제안하였다.

## II. 연구개발 필요성

### 2.1 소프트웨어(S/W) 개발 과정의 문제점

최근 4차 산업혁명과 디지털 트랜스포메이션의 확산으로 IT 시장은 더 많은 개발 인력과 빠른 소프트웨어 출시를 요구하고 있다. 그러나 국내 IT 업계는 개발자 인력 부족 문제에 시달리고 있으며, 이로 인해 개발 기간 단축이 어려운 상황이며, 국내 업계에서는 개발자 부족이 가장 큰 원인으로 지적되고 있다<sup>[6]</sup>.

이러한 현상에 따라 다수의 기업들은 중국, 인도, 베트남 등에 원격 S/W개발센터를 구축하고, 해외 인력채용을 통해 국내 IT개발자 인력난을 해소해나가는 오프쇼어 개발방식이 늘어나고 있다<sup>[7]</sup>.

원격지 개발은 증가하였지만 원격 현장에서는 표준화된 문서 부족으로 업무 파악에 시간이 소요되고, 경험 부족 개발자의 투입으로 개발 시간과 통합 과정에서 결함이 발생하여 테스트에 시간과 비용이 많이 드는 문제가 발생하고 있다<sup>[8]</sup>.

생성형 AI 기술 발표 이후, 많은 기업들이 이 기술을 활용하여 업무 생산성을 향상시키는 방법을 연구하고 있으며, 특히 컴퓨터 프로그래밍 언어의 이해와 생성에서 뛰어난 성능을 보여줌에 따라 소프트웨어(S/W) 개발 분야에서도 적극 활용될 것으로 예상된다<sup>[9-14,18,21]</sup>.

### 2.2 소프트웨어(S/W)개발 주요 단계

S/W 개발은 요구사항 분석, 아키텍처 설계, 프로그램 구현, 통합 및 테스트 과정을 거쳐 제품을 출시한다. 다음으로 소프트웨어 개발의 단계를 살펴보겠다.

첫 번째 단계는 요구사항 분석이다. 이 과정의 목적은 S/W 구성 요소에 대한 요구사항을 수립하는 것이다. 고객의 요구사항을 도출하고 분석하여 최종적으로 원하는 제품을 설계할 수 있도록 한다. 요구사항의 주요 특성은 완전성, 일관성, 정확성, 테스트 용이성, 명확성, 설계 독립성, 전달성, 견고성, 필요성이다<sup>[15]</sup>.

두 번째는 설계 단계로, S/W 요구사항을 구현하고 검증할 수 있는 설계를 제공하는 것이 목표이다. 이 단계에서는 S/W가 어떻게 구현될지를 결정한다.

세 번째는 S/W 통합 단계이다. 이 과정의 목적은 S/W 단위를 결합하여 설계와 일치하는 S/W 항목을 생성하고, 동일하거나 적합한 운영 플랫폼에서 기능적 및 비기능적 요구사항을 충족시키는 것이다<sup>[16,17]</sup>.

네 번째는 테스트 단계로서, 이 단계의 목표는 통합된 S/W 제품이 요구사항을 충족하는지 확인하는 것이다. 통합 프로세스가 S/W 단위 간의 상호작용을 점검하는 반면, 테스트 프로세스는 전체 S/W 제품이 기능적

표 1. 소프트웨어 요구사항 주요 특성  
Table 1. Main characteristics of SW requirements

| 주요 특성                                               | 설명                                     |
|-----------------------------------------------------|----------------------------------------|
| 완전성<br>(Completeness)                               | 사용자의 요구사항이 누락 없이 작성되어야 함               |
| 일관성<br>(Consistency)                                | 요구사항 간에 불일치 가 없어야 함                    |
| 정확성<br>(Correctness):                               | 시스템이 원래 사용자 요구를 충족시켜야 함                |
| 테스트 용이성<br>(Testability)                            | 요구사항은 테스트에 의해서 검증 가능하여야 함              |
| 명확성<br>(Unambiguity)                                | 요구사항의 기술은 애매모호함이 없어야 함                 |
| 설계와의 독립성<br>(Design freedom)                        | 요구사항 정의가 설계에 제약을 주어서는 안됨               |
| 전달성<br>(Communicability)                            | 요구사항은 명시적으로 기술되어 의사전달이 쉬워야 함           |
| 변경에 대한<br>견고성<br>(Modularity/<br>Change robustness) | 항목 추가, 수정 시 다른 부분에 영향을 미치지 않도록 구조화해야 함 |
| 필요성<br>(Necessity)                                  | 사용자 요구 목적에 기여해야 함                      |

및 비기능적 측면에서 오류와 결함을 발견하고 수정하는 과정이다<sup>[16,17]</sup>.

### III. 연구 내용

#### 3.1 생성형 AI 기술 활용 실험 환경

생성형 AI 기술 활용 실험 환경을 구성하기 위해 현재 많은 기업들이 운영하고 있는 해외 원격지 개발방식인 오프쇼어 개발을 실험 환경을 가정하였다. 오프쇼어 개발은 한국에서 요구사항을 전달하면 해외 원격지에서 요구사항 정의서 작성, 프로그램 개발, 프로그램 테스트를 수행하고 각 공정마다 한국에서 검증을 수행하는 방식을 취하고 있다. 이에 대한 Flow는 그림 1과 같다.

실험을 위해 활용된 생성형 AI 기술을 2023년 8월 한국 AI교육협회에서 측정한 AI 기술 성능 측정 순위에 따라 1위를 기록한 Open AI의 Chat GPT 3.5버전으로 선정하였고, 사용자 클라이언트 환경은 2023년 1월 기준 전 세계 데스크톱 컴퓨터 점유율 1위인 MS Windows 10 Pro. 버전을 채택하였다<sup>[22]</sup>.

기타 실험 환경은 표 2에 정리하였다.

생성형 AI 기술을 활용하여 S/W개발의 생산성 향상을 위한 실험방법은 오프쇼어 개발방식의 원격지 작업

한국 수행 작업

해외 원격지 수행 작업

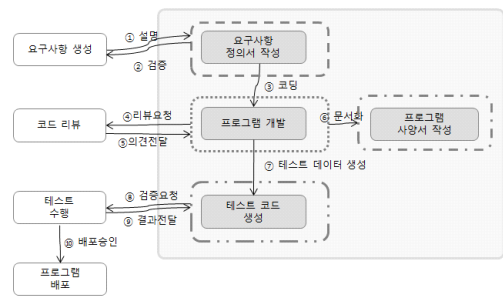


그림 1 오프쇼어 개발방식의 작업 Flow  
Fig. 1. Workflow of Off-shore Development

표 2. 생성형 AI 기술 실험환경  
Table 2. Simulation Environments of Generative AI

| 구성요소       | 실험환경                                                          |
|------------|---------------------------------------------------------------|
| 생성형 AI 모델  | OpenAI ChatGPT                                                |
| 버전         | 3.5                                                           |
| 클라이언트 환경   | Microsoft Windows 10 Professional                             |
| 요구사항 번역 언어 | 한글 → 영어 → 한글                                                  |
| 프로그램 언어    | JAVA                                                          |
| 사양서 작성언어   | 한글, 영어                                                        |
| 텍스트 비교 도구  | Text Compare (https://text-compare.com)                       |
| 번역결과 비교 도구 | Google Collab. (활용 플랫폼)<br>Python (구현 언어)<br>BLEU 계산법 (비교 기법) |

단계를 참고하여 4단계로 나누어 실험하고 결과를 평가하였다. 첫 번째 단계는 요구사항 정의서 작성, 두 번째 단계는 프로그램 작성, 세 번째는 프로그램 사양서 작성, 네 번째 단계는 테스트 코드를 작성하는 실험을 각각 수행하였다. 그리고, 문서를 생성하는 실험은 한글과 영어를 생성하고, 영어를 한글로 재 번역하여 결과를 비교하였고, 실험 방식은 그림 2의 Flow와 같다.

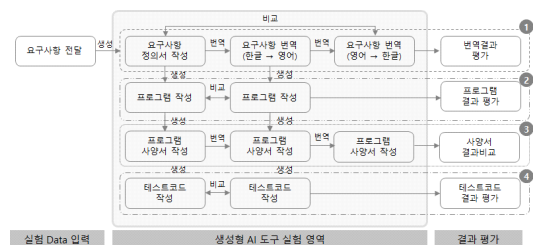


그림 2 생성형 AI 기술 실험 Flow  
Fig. 2. Workflow of Generative AI Simulation

### 3.2 요구사항 정의서 작성 실험 및 결과

S/W개발의 첫 번째 단계인 요구사항 정의서 작성 실험은 앞서 설명한 그림 2의 ①번 영역으로서, 생성형 AI 기술을 활용하여 한글로 요구사항 정의서를 작성하고, 해외 원격지 개발을 위해 영어로 번역한 후 영어로 번역된 결과의 정확도 확인을 위해 한글로 다시 재번역하여 처음 생성된 한글 요구사항 정의서와 비교하는 방식으로 실험을 수행하였고 상세한 실험 Flow는 그림 3과 같다. 요구사항 정의서 작성은 기업용 임직원 관리 프로그램 개발을 주제로 작성하도록 요청하였다.

또한, 구체적인 요구사항 범위 지정이 필요하여, 인프라 환경은 AWS 기반의 클라우드 환경, 아키텍처는 MSA(Microservice Architecture) 구조, 세부 요구사항으로 검색기능을 요청하였다. 또한, 생성형 AI 에게 문장을 생성하는 구체적인 톤(Tone)을 요청하였다. 요구사항 정의서 작성을 위해 생성형 AI 에게 입력한 명령은 4단계에 걸쳐서 한글로 표 3과 같이 전달했다.

한글 요구사항 정의서 작성을 실험한 결과 생성형 AI 도구는 사용자의 입력 값을 받아서 약 3초 이내에 A4 1장 분량의 요구사항 정의서를 출력하였고, 지속적으로 결과물의 보완이 가능한 것을 확인하였다.

한글 요구사항 정의서 생성 실험의 요약 결과는 표 4와 같다. 해외 원격지 개발을 위해 요구사항 정의서를

표 4. 한글 요구사항 정의서 요약

Table 4. Summary of Korean requirements definition document

| 구분        | 상세 내용                |
|-----------|----------------------|
| 생성형 AI 답변 | 임직원 관리 프로그램 요구사항 정의서 |
|           | 1. 서론                |
|           | 1.1 배경               |
|           | 1.2 목적               |
|           | 1.3 범위               |
|           | 2. 기능적 요구사항          |
|           | 2.1 임직원 정보 관리        |
|           | 2.2 검색 정보 관리         |
|           | 2.3 임직원 정보 검색        |
|           | 2.4 보고서 생성           |
|           | 2.5 데이터 백업 및 복원      |
|           | 3. 비기능적 요구사항         |
|           | 3.1 보안               |
|           | 3.2 사용자 인터페이스        |
|           | 3.3 성능 및 확장성         |
|           | 4. 기술 스택             |
|           | 5. 클라우드 네이티브 및 MSA   |
|           | 6. 유지보수 고려사항         |
|           | 7. 마무리               |

영어로 번역하는 실험을 수행하였고, 영어 번역의 정확도 측정을 위해 다시 한글로 번역하여 최초 한글 문서와 비교하는 실험을 수행하였다.

영어로 번역한 요구사항 정의서를 요약한 결과는 표 5와 같다. 한글로 생성한 요구사항 정의서와 영어 생성

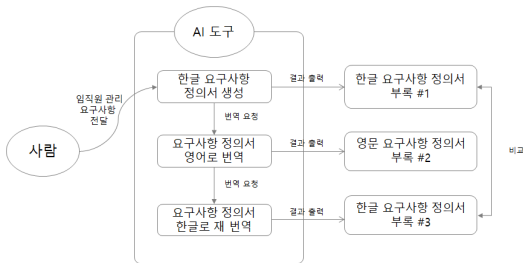


그림 3 요구사항 정의서 작성 실험 Flow

Fig. 3. Flow of Requirements Definition Document Simulation

표 3. 요구사항 정의서 작성을 위한 입력사항

Table 3. Input for requirements definition document

| 전달순서 | 상세 내용                                            |
|------|--------------------------------------------------|
| 1    | 임직원 관리 프로그램을 요구사항 정의서를 작성해줘.                     |
| 2    | 요구사항에 AWS기반의 클라우드 Native 환경, MSA구조에 대해 자세히 추가해줘. |
| 3    | 요구사항에 검색기능에 대해 추가하고 검색은 최대 20개까지 가능하도록 설명해줘.     |
| 4    | 요구사항 정의서를 IT전문가처럼 작성해줘.                          |

표 5. 임직원 관리 요구사항 정의서 영어 번역 결과

Table 5. English translation result of staff care requirements definition document

| 영어로 번역한 요구사항 정의서                                  |
|---------------------------------------------------|
| Employee Management Program Requirements Document |
| 1. Introduction                                   |
| 1.1. Background                                   |
| 1.2. Purpose                                      |
| 1.3. Scope                                        |
| 2. Functional Requirements                        |
| 2.1. Employee Information Management              |
| 2.2. Concurrent Employment Management             |
| 2.3. Employee Information Retrieval               |
| 2.4. Report Generation                            |
| 2.5. Data Backup and Restoration                  |
| 3. Non-Functional Requirements                    |
| 3.1. Security                                     |
| 3.2. User Interface                               |
| 3.3. Performance and Scalability                  |
| 3.4. Documentation                                |
| 4. Technology Stack                               |
| 5. Cloud-Native and MSA                           |
| 6. Maintenance Considerations                     |
| 7. Conclusion                                     |

한 결과를 비교한 결과 원본과 번역이 일치하는 것을 확인하였다.

마지막으로 영문 요구사항 정의서를 다시 한글로 번역하여 최초로 생성한 한글 요구사항 정의서와 일치하는지 실험을 하였다. 재번역 실험을 수행한 최종 결과 목차에 대한 요약은 표 6과 같다.

한글로 최초 작성한 요구사항 정의서와 영문으로 번역 후 한글로 재번역한 요구사항 정의서를 비교한 결과 목차는 동일하나 세부적인 항목에서 단어 선택과 내용의 일부가 달라진 것을 확인하였다. 원본과 재번역 문서의 서론을 발췌해 비교 결과는 표 7과 같다.

두 문서의 정확도 비교를 위해 기계 번역의 성능 평가지표인 BLEU(Bilingual Evaluation Understudy)와, 사람이 직접 원본과 비교하는 정성적 기법을 활용하였다. 정량적 성능 평가인 BLEU 측정 결과는 그림 4와 같으며, 결과는 높은 수준의 번역(High quality translations)으로 평가받는 53.92점이 측정되었다.

또한, 본 실험에 활용 기법이 다른 기법보다 우위에 있는지를 측정하기 위해 번역 전용 도구인 Google 번역기와 Naver Papago 번역기로 동일 실험을 수행하였고, BLEU 점수는 각각 Google 번역기는 28.56점, Naver 번역기는 28.63점을 기록하는 것을 확인하였고, 본 실험에 활용한 생성형 AI 도구가 더 우수한 품질의 결과를 생성해내는 것을 확인하였고 이에 대한 요약 결과는

표 7. 한글 원본과 한글로 재번역한 문서의 비교  
Table 7. Comparison of Korean document and Korean re-translation document

| 한글 원본 요구사항 정의서                                                                                                                                                                                                                                                                                                                                                                                                                                  | 재번역한 요구사항 정의서                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 임직원 관리 프로그램 요구사항 정의서<br>2 1. 서론<br>3 1.1. 배경<br>4 현재 회사는 임직원 관리와 관련된 업무를 효율적으로 수행하기 위해 현대적이고 강력한 소프트웨어 도구에 의존하고 있습니다. 이러한 요구를 충족하기 위해 임직원 관리 프로그램이 필요합니다.<br>5<br>6 1.2. 목적<br>7 이 요구사항 정의서의 목적은 임직원 관리 프로그램의 범위, 기능 및 품질 속성을 정의하고 이를 개발 및 배포 팀의 기준으로 제시하는 것입니다.<br>8<br>9 1.3. 범위<br>10 임직원 관리 프로그램은 임직원 정보 관리, 검색, 정보 관리, 검색 및 보고서 생성 기능을 포함하는 데, 이러한 기능은 사용자 편의성과 데이터 보안에 중점을 둘 것입니다. 또한, 클라우드 네이티브 환경과 MSA 아키텍처를 활용하여 확장성을 갖출 것입니다.<br>11 | 1 임직원 관리 프로그램 요구사항 정의서<br>2 1. 서론<br>3 1.1. 배경<br>4 현재 회사는 임직원 관리와 관련된 업무를 효율적으로 수행하기 위해 현대적이고 강력한 소프트웨어 도구에 의존하고 있습니다. 이 요구사항 정의서는 임직원 관리 프로그램의 범위, 기능 및 품질 속성을 정의하고 개발 및 배포 팀의 기준으로 제시합니다.<br>5<br>6 1.2. 목적<br>7 이 요구사항 정의서의 목적은 임직원 관리 프로그램을 개발하고 배포하는 데 필요한 요구사항을 정의하고 프로젝트 이해관계자 간에 공통된 이해를 돕기 위한 참조 자료를 제공하는 것입니다.<br>8<br>9 1.3. 범위<br>10 임직원 관리 프로그램은 임직원 정보 관리, 검색, 정보 관리, 검색 기능, 보고서 생성 기능을 포함하며, 사용자 편의성과 데이터 보안을 강조합니다. 또한, 클라우드 네이티브 환경에서 작동하고 확장성을 갖추기 위해 Microservices Architecture (MSA)를 활용합니다.<br>11 |

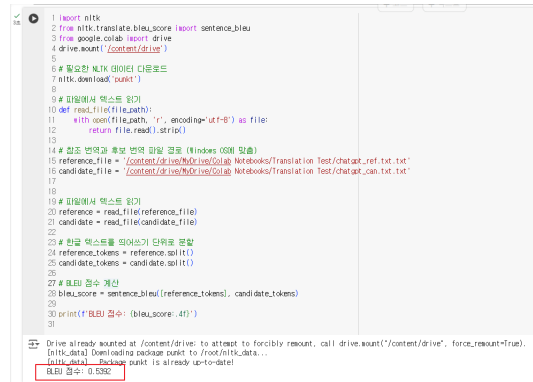


그림 4. 요구사항 정의서 번역결과 BLEU 측정 점수  
Fig. 4. BLEU Score for Requirements Definition Document Translation

표 6. 영문 요구사항 정의서를 한글로 번역한 결과  
Table 6. Korean translation result of English requirements definition document

| 한글로 번역한 요구사항 정의서                                                                                                                                                                                                                                                                                           |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 임직원 관리 프로그램 요구사항 정의서<br>1. 서론<br>1.1. 배경<br>1.2. 목적<br>1.3. 범위<br>2. 기능적 요구사항<br>2.1. 임직원 정보 관리<br>2.2. 검색 정보 관리<br>2.3. 임직원 정보 검색<br>2.4. 보고서 생성<br>2.5. 데이터 백업 및 복원<br>3. 비기능적 요구사항<br>3.1. 보안<br>3.2. 사용자 인터페이스<br>3.3. 성능 및 확장성<br>3.4. 문서화<br>4. 기술 스택<br>5. 클라우드 네이티브 및 MSA<br>6. 유지보수 고려사항<br>7. 결론 |

표 8. 번역 도구별 BLEU 측정결과 비교  
Table 8. Comparison of BLEU Scores Across Different Translation Tools

| 번역도구         | BLEU Score |
|--------------|------------|
| ChatGPT 3.5  | 53.92      |
| Google 번역기   | 28.56      |
| Naver Papago | 28.63      |

표 8에 정리하였다.

결론적으로 S/W 개발을 수행하는 과정에서 생성형 AI 기술을 활용한 번역은 생산성 향상 측면에서 도움이 되고, 번역 품질에 문제가 없음을 확인할 수 있었다. 다만, 문서의 성격에 따라 계약서나 법률정보와 같이 단어의 선택과 문장 표현의 차이가 민감한 경우에는 AI 기술을 통한 번역 결과의 활용에 주의가 필요하고 해당 분야의 전문가에 의한 추가적인 검증이 반드시 필요하다는 결론을 얻을 수 있었다.

### 3.3 프로그램 작성 실험 및 결과

생성형 AI를 활용하여 프로그램을 작성하는 실험을 수행하기 위해, 앞선 실험을 통해 생성한 요구사항 정의서를 참고하여 임직원 관리 프로그램에서 입력받은 임직원 속성 정보를 데이터베이스에 입력하는 프로그램을 생성하는 실험을 수행하였다.

오피쇼어 개발환경을 재현하기 위해 한글과 영어로 각각 임직원 관리 프로그램 생성을 요청하고 결과를 비교하였다. 프로그램은 JAVA 언어를 기반으로 프로그램의 이름은 프로젝트의 코딩 컨벤션 규칙에 따라 사전에 지정하여 해당 규칙에 맞게 프로그램이 생성되는지 실험을 수행하였다. 이에 대한 상세한 실험 Flow는 그림 5와 같다.

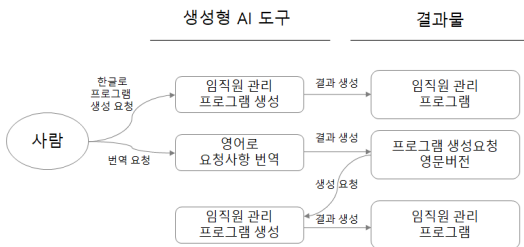


그림 5 임직원 관리 프로그램 생성 실험 flow  
Fig. 5. Flow of staff care program generation simulation

### 3.4 S/W 개발을 위한 생성형 AI 기술 활용 고찰

본 논문에서는 생성형 AI를 활용하여 S/W 개발 공정을 실험한 결과를 표 9에 제시하였다. 실험 결과를 요약하면, 첫째, 요구사항 정의서 작성과 번역에서 생성형 AI 도구는 요청된 요구사항을 누락 없이 반영하였으며, 번역 시 일부 단어를 의역하여 더 자연스러운 표현을 제공했다. 이는 S/W 개발에 실질적인 문제가 되지 않을 것으로 판단되었다.

둘째, 프로그램 작성 실험에서 생성형 AI는 다양한 프로그래밍 언어의 문법과 패턴을 학습하여 사용자 요구사항에 맞는 프로그램을 사람이 작성하는 코드와 유사하게 생성하였다. 또한, 약 50라인의 코드를 5초 이내에 자동 생성하며 개발 생산성 향상에도 기여할 것으로 예상된다.

세 번째, 프로그램 사양서 작성 실험 결과, 프로그램 문단 단위로 2-3줄의 설명이 생성되어 프로그램을 이해할 수 있는 수준으로 작성되었다. 동일 프로그램의 사양서를 한글과 영문으로 작성한 결과, 영문 사양서가 더 자세하고 많은 항목을 포함하였다. 이는 생성형 AI가 영문에 대해 더 많은 지식과 학습을 했기 때문에, 프

표 9. 생성형 AI 기술 활용 실험 결과  
Table 9. Result of generative AI application simulation

| 실험 내용            | 실험 결과                                                                                                               |
|------------------|---------------------------------------------------------------------------------------------------------------------|
| 요구사항 정의서 작성 및 번역 | (작성) 요청자의 요구사항을 반영하여 A4 분량으로 총 7개 항목으로 정리된 요구사항 정의서 작성<br>(번역) 최종 번역 언어에 맞는 더 자연스러운 문장을 생성하기 위해 의역을 하거나 내용을 추가하여 번역 |
| 프로그램 작성          | 사전에 학습된 (Pre-Trained) JAVA 프로그램 문법과 패턴에 따라 프로그램을 제작하고 배경설명을 추가됨                                                     |
| 프로그램 사양서 작성      | 한글과 영어로 사양서 작성을 요청했을 시 한글의 사양서도 실무에서 활용이 가능한 수준으로 작성이 되었으나, 영어의 결과가 더 많은 분량으로 작성됨                                   |
| 테스트 코드 작성        | 사전에 학습된 (Pre-Trained) JAVA 및 JUnit 문법에 따라 작성이 되었으며, 한글/영어 모두 동일한 결과 생성                                              |

로그램 사양서는 영문으로 작성 후 한글로 번역하는 것이 더 효과적이라는 결론을 얻었다.

마지막으로, 개발된 프로그램의 테스트 코드 생성을 실험한 결과, 한글과 영문 모두 동일한 결과를 생성하였으며, 테스트 주도 개발과의 접목을 통해 개발자는 생성형 AI 도구를 사용하여 자신이 작성한 프로그램의 테스트 코드를 생성할 수 있음을 확인하였고,

본 실험의 단계별 활용 가능성에 대한 고찰은 표 10과 같이 요약할 수 있다.

표 10. 생성형 AI 기술 활용 가능성 고찰  
Table 10. Survey of generative AI application possibility

| 실험 내용            | 활용 가능성 고찰                                                                            |
|------------------|--------------------------------------------------------------------------------------|
| 요구사항 정의서 작성 및 번역 | - 비정형 데이터를 분석하여 규격에 맞는 요구사항 정의서 작성에 활용 가능<br>- 법률, 의학, 계약 문서와 같은 민감 문서는 사람의 추가 검증 필요 |
| 프로그램 작성          | - 이미 학습된 프로그램 문법과 패턴에 따라 프로그램 생성에 즉시 활용 가능<br>- 현재는 개발자를 보조하는 도구로서 활용 가능             |
| 프로그램 사양서 작성      | - 프로그램 문단 단위로 로직 설명이 가능하며, 영어로 최초 생성 후 번역하는 방법을 추천                                   |
| 테스트 코드 작성        | - 개발된 소스 코드에 대한 테스트 코드 생성에 즉시 활용 가능<br>- TDD 개발, Test Code Automation 분야 활용          |

#### IV. 결 론

이 연구에서는 S/W 개발 공정의 여러 문제점을 생성형 AI 도구를 활용하여 상당 부분 자동화하고 대체할 수 있음을 확인할 수 있었다. 현재 일부 제품에서는 이미 생성형 AI 도구를 소스 코드 생성에 부분적으로 활용하고 있지만, 본 실험 결과를 바탕으로 인간이 자연어로 요구사항을 전달하면 생성형 AI가 자동으로 요구사항 정의서를 작성하고 프로그램을 생성하며, 테스트 코드를 작성하고 검증하며 사양서까지 작성하는 높은 수준의 S/W 개발 공정 자동화(Hyper Automation for S/W Development Process)도 가능할 것으로 기대된다.

다만, 생성형 AI 도구가 학습되지 않은 분야는 기존에 학습된 데이터를 기반으로 거절된 정보를 생산할 수 있기 때문에 추가적인 학습이 필요한 분야는 보강 학습(Fine Tuning)을 수행하고, 지식시스템의 데이터를 검색을 통해서 보강하는 RAG(Retrieval Augmented Generation) 모델을 활용한다면 생성형 AI가 잘못된 답변을 제공하는 착각현상(Hallucination)을 예방하고, 더 사실기반의 자동화 작업을 수행할 것으로 생각된다.

#### References

- [1] 양지훈, 윤상혁, *ChatGPT를 넘어 생성형 (Generative) AI 시대로 : 미디어·콘텐츠 생성형 AI 서비스 사례와 경쟁력 확보 방안*, DBPia Domestic Report, pp. 3-8, 2023.
- [2] 하이투자증권, *챗 GPT, AI 시대의 게임 체인저*, DGB금융그룹 Industry Report, Jan. 2023.
- [3] 미래에셋증권 디지털리서치팀, *Everything Everywhere All at Once : AI 가 불러온 新산업 혁명*, Apr. 2023.
- [4] 국경완, *생성형AI주요 이슈 및 동향, 한인과학기술자네트워크*, May 2023, from [https://kosen.kr/info/kosen/REPORT\\_0000000002341](https://kosen.kr/info/kosen/REPORT_0000000002341)
- [5] A. Vaswani, et al., "Attention is all you need," *Advances in NIPS*, pp. 1-11, Long Beach, CA, USA, 2017.
- [6] 변희원, *의대 솔루션 후폭풍... K스타트업, 인도-베트남 찾아가 '개발자 급구'*, 『조선일보』, Mar. 2024, from [https://www.chosun.com/economy/t ech\\_it/2024/03/25/SFN7DW2BBFEVDESTPV RHXJVHKA/](https://www.chosun.com/economy/t ech_it/2024/03/25/SFN7DW2BBFEVDESTPV RHXJVHKA/)
- [7] 김도연, [기고] *베트남 IT 생태계와 아웃소싱 현*

- 주소*, kotra, Apr. 2023, from [https://dream.kotra.or.kr/kotranews/cms/news/actionKotraBoardDetail.do?SITE\\_NO=3&MENU\\_ID=130&CONTENTS\\_NO=1&bbsGbn=246&bbsSn=246&pNttSn=202078](https://dream.kotra.or.kr/kotranews/cms/news/actionKotraBoardDetail.do?SITE_NO=3&MENU_ID=130&CONTENTS_NO=1&bbsGbn=246&bbsSn=246&pNttSn=202078)
- [8] 김진환, *베트남 현지 IT 업계 이슈 및 ERP 현황*, DevMento, Nov. 2023, from <https://www.devmento.co.kr/%EB%B2%A0%ED%8A%B8%EB%82%A8-%ED%98%84%EC%A7%80-it-%EC%97%85%EA%B3%84-%EC%9D%B4%EC%8A%88-%EB%B0%8F-erp-%ED%98%84%ED%99%A9>
- [9] B. Violino, *AI 기반의 똑똑한 자동화... '하이/퍼오토메이션' 시대가 온다*, CIO Korea, Mar. 2021, from <https://www.ciokorea.com/news/185098>
- [10] 김성수, *'하이/퍼오토메이션', 업무 혁신의 해결책 될 수 있다*, 산업일보, Jun. 2023, from <http://kidd.co.kr/news/232964>
- [11] E. Schuman, 칼럼 | *'거부하기 힘든 독 사과' 생성형 AI로 코딩하기*, CIO Korea, Apr. 2023, from <https://www.ciokorea.com/news/288631>
- [12] I. Sacolick, 칼럼 | *'코드 생성은 시작일 뿐'... 생성형 AI가 이끌 SW 개발의 변화 10가지*, CIO Korea, Feb. 2024, from <https://www.ciokorea.com/news/325534>
- [13] J. Rose, 칼럼 | *생성형 AI가 대체하는 개발자... 필요한 건 '주니어 개발자 역할 재정의'*, CIO Korea, Dec. 2023, from <https://www.ciokorea.com/news/316446>
- [14] J. Fruhlinger, *AI 코딩 도구 이렇게 쓰면 효과 ↑... 개발자 9명이 전하는 AI 프로그래밍 활용법*, CIO Korea, Jul. 2023, from <https://www.ciokorea.com/news/299191>
- [15] H.-G. Kim, *소프트웨어 요구사항 분석 및 정의, 2014 Softw. Eng.*, pp. 1-56, 2014, from <http://contents.kocw.or.kr/KOCW/document/2014/deagucatholic/kimhangkon1/2.pdf>
- [16] H. Kong, K.-Y. Eom, and M.-H. Kim, "A study of development process model appropriate for Korean middle-sized software development projects," in *Proc. KIPS Conf.*, pp. 343-346, 2008.
- [17] 정호원, 김길조, 김도관, *"ISO/IEC15504(SPICE) 기반의 중소기업을 위한 소프트웨어 표준 프로*

세스” 산업자원부 기술표준원, 2007.

- [18] S. Kim, “Developing code generation prompts for programming education with generative AI,” *The J. Korean Assoc. Comput. Edu.*, vol. 26, no. 5, pp. 111-115, 2023.
- [19] Y. Lee, “A study on the antecedents and the outcome of generative AI use : Focusing on the moderating effect of metacognition,” Ph.D. dissertation, Kookmin University The Graduate School of Business IT, pp. 8-16, 2023.
- [20] Y. Kim, “특허 데이터를 이용한 인공지능 분야 주요 기술 동향 분석과 시사점 : 생성형 AI 기반 챗봇 기술을 중심으로,” M.S. Thesis, Korea University Graduate School of Management of Technology, pp. 9-13, 2024.
- [21] S. Yoon, “생성형AI(GenAI) 서비스 사용의도에 미치는 영향에 관한 연구 : 확장된 통합기술 수용이론(UTAUT2) 중심으로,” Ph.D. dissertation, Dongguk University Graduate School, pp. 38-51, 2024.
- [22] 김재민, 한국AI교육협회, 최초로 ‘챗GPT·바드·빙·하이퍼클로바X’ 성능 비교 평가, 블록체인투데이, Sep. 2023, from <https://www.blockchaintoday.co.kr/news/articleView.html?idxno=36928>

#### 최 기 원 (Kiwon Choi)



2004년 2월 : 성균관대학교 정보통신공학부 졸업  
2024년 2월 : 건국대학교 정보통신대학원 석사  
2024년 3월~현재 : 숭실대학교 IT정책경영학과 박사과정

<관심분야> 엔터프라이즈 아키텍처, 인공지능, 그룹웨어, 엔터프라이즈 포탈

[ORCID:0009-0004-0525-7566]

#### 강 동 구 (Dong Koo Kang)



2006년 2월 : 삼육대학교 컴퓨터정보과 졸업  
2012년 2월 : 성균관대학교 정보통신대학원 석사  
2022년 3월~현재 : 숭실대학교 IT정책경영학과 박사과정

<관심분야> IT교육 공학, 프로그래밍언어, AI 및 데이터분석

[ORCID:0009-0007-2133-3620]

#### 김 진 태 (Jin Tae Kim)



1991년 2월 : 건국대학교 전자공학과 졸업  
1993년 8월 : 건국대학교 전자공학과 석사  
1999년 2월 : 건국대학교 전자공학과 박사  
(주)파이브텍 대표이사, 건국대학교 정보통신대학원 겸임교수

<관심분야> 실시간시스템, 스마트ICT응용

[ORCID:0009-0002-0623-7001]

#### 조 주 필 (Ju Phil Cho)



2001년 2월 : 전북대학교 전자공학과 공학박사  
2000년~2005년 : ETRI 이동통신연구단 선임연구원  
2006년~2007년 : ETRI 이동통신연구단 초빙연구원  
2011년~2012년 : 미국 USF, 교환교수

2005년~현재 : 국립군산대학교 IT융합통신공학과 교수

<관심분야> 5G, 6G, Cognitive Radio, LED-ID, IoT 융합기술

[ORCID:0009-0003-1041-2538]