

## 공유 온 팩토리 서비스를 위한 수발주 시스템

# Purchase and Acquisition Order System for sharing on factory

서연경

인천재능대학교 시컴퓨터정보과

Youn-Kyoung Seo

Department of AI Computer Information, JEI University, Incheon 123-456, Korea

### [요 약]

산업통상자원부에서 추진하는 산업융합 규제 샌드박스 의결에 따라, 산학협력업체인 마이메이커(주)에서 신청한 “공장 내 공작기계 공유 서비스”를 실증 특례를 받았다.[1] 이를 실행하면서 수발주 서비스 부분을 소프트웨어로 개발해야 하는 상황으로 업체의 제조도메인 문제를 파악 분석하고 절차를 정립하고 관련 요구사항에 맞는 시스템을 설계하였다. LINC+ 산업체 애로 사항 해결 공동연구 개발 프로젝트로 공유 온 팩토리 서비스[2]를 위한 수발주시스템을 최종 개발했다. 수 발주 시스템에 필요한 데이터 구조를 설계하고 관련 데이터를 활용한 화면 설계와 사용자 관점의 기능 구현을 중심으로 하였으며 정립된 절차와 요구사항 분석에 맞추어 기획, 설계, 프로토타이핑, 구현 제작 단계로 진행했다. 마지막으로 최종개발 내용을 요구사항에 맞게 잘 구현되었는지 검토 및 기능 검증 시연과 제품 인수인계를 통해 애로 사항 해결을 확인하였다.

### [Abstract]

In accordance with the industrial convergence regulation sandbox decision promoted by the Ministry of Trade, Industry and Energy, the “machine tool sharing service in the factory” applied for by MyMaker Co., Ltd., an industry-academia cooperation company, was granted a demonstration exception. While implementing this, in a situation where the ordering service part had to be developed as software, the company's domain problems were identified and analyzed, procedures were established, and a system was designed that met the related requirements. As a joint research and development project to solve the difficulties of the LINC+ industry, the order-to-order system was finally developed for shared factory services. In accordance with the procedure and requirements analysis, the planning, design, prototyping, and implementation production stages were carried out. Finally, it was confirmed that the final development contents were well implemented according to the requirements, and the resolution of the difficulties was confirmed through functional verification demonstrations.

**Key word** : Acquisition, Development, Factory, Information, Order, Purchase, Sharing, System etc.

<http://dx.doi.org/10.12673/jant.2023.27.1.146>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 24 January 2023; Revised 31 January 2023

Accepted (Publication) 22 February 2023 (28 February 2023)

\*Corresponding Author, Youn-Kyoung, Seo

Tel: +82-32-890-7520

E-mail: seoyk@jeiu.ac.kr

## I. 서론

인천 및 경기도 남부지역의 소상공, 중소기업들을 연결하는 플랫폼 형태의 발주, 입찰, 견적, 수주 등의 진행 절차상 필요한 기능을 웹서비스로 구현하고, 수주업체들과 발주업체들이 원활히 관련 정보들을 조회, 등록, 수정, 관리할 수 있게 도와주는 서비스 개발 필요성 요구에 따라 기획 설계, 개발, 테스트 단계를 수행하였다. 마이메이커 공유온팩토리는 “공장 내 공작기계 공유 서비스”를 활용하는 고객들이 수주 서비스와 발주 서비스를 활용하고, 입찰 현황들을 살펴보고, 매칭 진행 상황을 파악할 수 있으며, 입찰 관리를 통해 최적의 견적과 신뢰도가 높고 우수한 수주업체를 선정하는 부분에 도움을 줄 수 있도록 요구하였다. 수주업체들은 공장 내 현황과 자재 현황, 납기일에 맞는 운영 가능한 입찰들을 조회하여 견적을 통해 낙찰받을 수 있어야 했다. 그러면 유휴 장비 없이 지속 생산 거래를 할 수 있는 기반을 제공할 수 있게 된다. 정량적, 정성적 선정 기준에 맞춰 매칭된 결과를 쌓아 신뢰도와 품질 관정에 기반한 매칭에 도움을 주고자 웹 기반 서비스 개발을 목표로 연구기획, 설계, 구현하였다.

지역 내 협력업체들과 영세 소상공 기업들에 필요한 물량을 공급할 수 있도록 지원하며, 부품이 필요한 장비 업체들에 빠르게 공급을 지원하여, 원활한 기업 활동이 이루어지도록 지원할 수 있게 되는 효과를 가지게 된다. 지역의 소기업을 대상으로 하는 만큼 지역경제의 활력에 도움이 될 것이다. 현재 업체가 잘 운영 중인 ‘공유 온 팩토리’ 사이트의 협력업체들을 대상으로 실증 테스트 후 신규 협력업체들과 다양한 품목별 수주 및 발주 거래 공정 과정에 확대하여 고도화 개발에 중요한 실증 및 연구 개발 토대를 마련하였다. 기능적, 비기능적 요구사항을 분석 및 데이터베이스 설계 구현을 하고 웹으로 개발하여 실증 및 효율성을 분석할 수 있었다.

## II. 요구사항 분석 설계

주문을 통해 만들어야 하는 제작품은 고객의 요구에 맞는 품질과 납기가 우선되어야 하고, 가격 경쟁력은 선순환 구조 속에서 연속성을 유지하는 중요 요소로서 종합적으로는 수요자와 공급자 간 신뢰 형성이 중요한 핵심 요소이다. 따라서 이 시스템은 발주와 수주를 연결하는 기능이며 신뢰가 바탕이 되도록 설계해야 한다. 그리고 수주, 발주 시스템은 B2B 대상 기계 제작품 관리를 위해 플랫폼 내에서 수요자와 공급자가 원활하게 사업에 필요한 처리를 수행하도록 상세한 기능적 요구사항이 반영되어 수행되어야 한다. 이를 위해 필요한 기능과 내용을 요약하여 정리하면 아래 표 1과 같다. 이를 바탕으로 프로그램 개발이 진행되도록 분석 및 절차 정의와 개발 준비가 필요하다. 수발주 시스템은 이미 개발된 제품이 많으나 공작기계 제작 관련 공유온팩토리 시스템 내 소상공인들에게 최적화되고 해당

도메인에 맞춤형으로 기존 고객들이 활용 가능한 개발된 웹 기반의 통합 연결이 필요하다. 반도체장비업체, 반도체 제조업체, 산업용 기계 제조업체, 생활용품 업체, 해외업체 등 공유온 팩토리 회원사, 인천 소상공인 회원사 등 발주자와 수주업체가 거래 수수료 없이 사용 가능해야 한다. 표1에 있는 거래절차 상세처리 요구 사항을 반영하여 개발해야 하는 부분에서 해당 문제 도메인 맞춤형으로 요구사항 분석을 하였다.

표 1. 수발주처리를 위한 필요 기능과 상세처리 요구사항

Table 1. Necessary functions and details of processing

Title	Purpose	Details of processing
Information-registration	Identify credit information registration	Company information/ personal information/transaction agreement/company registration process
public-announcement	Receipt of required products	Manufacture name/material/ quantity/delivery/price standards/ transaction conditions/ drawing registration/processing required information/personal information/ transaction agreement/company registration process
Receipt-notification	Notify relevant manufacturers	message notification/ reception details to the target person
Quotation-registration	Propose production conditions and cost	estimate, cost, production details, delivery date process
Quotation-notification	Provide quotes to vendors	Quotes provided, private delivery required
Issuance of purchase order	Notification of company selection, start of production	notification of selection/ issuance of purchase order/specification of delivery conditions
Inspection	Delivery inspection	inspection and notify the delivery date process
Delivery	Confirmation	Delivery confirmation process
Result-Notification	quality check	according to prior regulations through pass/fail process
payment	Project finish confirmation	supplier transaction evaluation process
AS	defect repair	Defect issuance process
After-Action	Business trust index management	Handling exposure of reference data/evaluating credit rating/

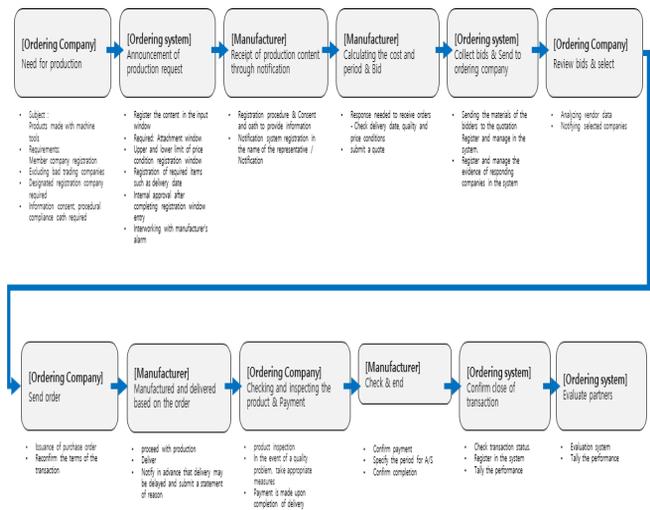


그림 1. 초기 요구 사항 분석-시스템 절차

Fig.1. Initial requirement analysis System Procedure

요구사항 파악을 위한 도메인 문제 인터뷰를 통한 거래절차를 개괄적으로 먼저 파악하였을 때 그림 1과 같은 시스템 사용자 기반의 흐름과 내용을 기반으로 표현되어 절차 정의가 다소 복잡하고 업무와 데이터기반의 분석관점이 시급해 보였다. 해당 불필요한 것을 최소화 하고 업무 중심-데이터 중심 기반으로 절차와 모델을 만들어야 할 필요성을 느끼고 7단계로 수정 보완 및 시스템설계를 진행했다.

초기 요구사항을 재분석하여 처리 절차와 단계를 간소화, 명확화 작업을 하고 사용자 중심, 업무 중심, 데이터 중심 관점의 흐름을 재정의했다. 이에 맞게 데이터 CRUD[3] 작업에 대한 등록, 조회, 수정, 삭제 기능들을 세부 서비스별 단위 기능으로 설계하였다. 해당 부분에 대한 데이터 설계는 그림 2와 같다.

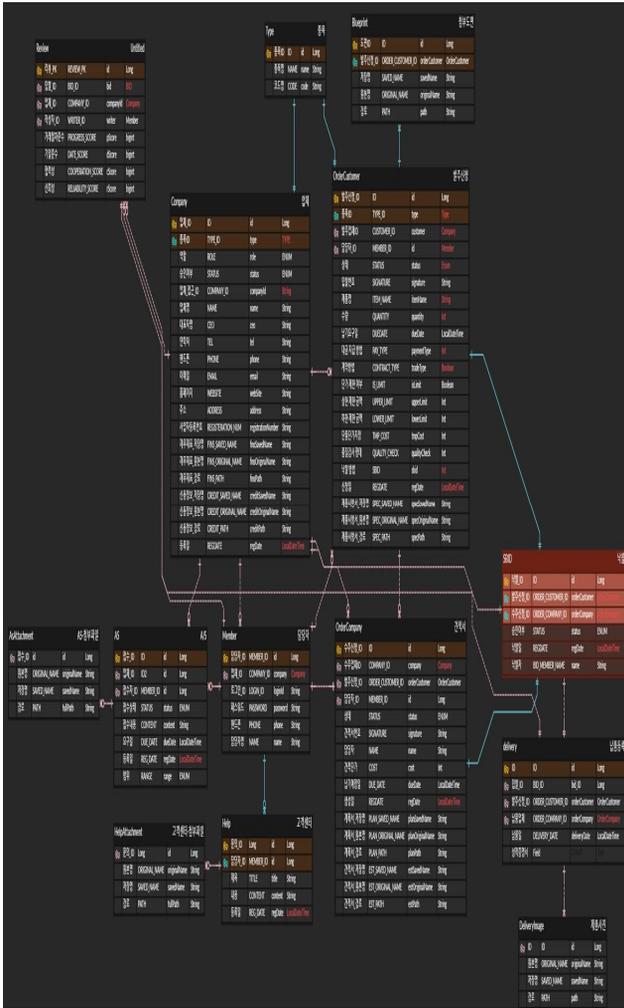


그림 2. 수발주시스템을 위한 개체 관계 다이어그램(ERD)  
Fig. 2. P-A Order System Entity Relationship Diagram

아래 총 7개 단계 서비스별 세부 기능 설계는 표 2와 같다. 기타 전체 수발주 시스템의 서비스 흐름도는 그림 3과 같이 정의하였다. 수발주시스템(P-A Order System)의 프로세스를 총 7개 단계로 정리했다. [(1)입찰처리 -(2) 견적처리 -(3) 선정

낙찰 처리 -(4) 발주 발행 확인 처리 -(5) 제품 제작 처리-(6) 납품처리 -(7) 최종 검사 및 결과 처리] 로 표현했다. 우선 사용자 관리는 발주업체와 수주업체 구분해서 업체등록을 할 수 있고, 관리자 측의 승인을 받은 뒤 업체 아이디를 이용해 업체 담당자가 회원가입 후 관리자 승인을 받아야 수발주 서비스를 이용할 수 있도록 2단계를 거치도록 한다.

표 2. 서비스별 세부 기능 단위 설계

Table 2. Service-specific functional unit design

Service Phase	using table	call function by user
(1)Bid	Order Customer Blueprint	-Ordering company : Performs bid registration & production name, material, quantity, delivery date, price, standard, transaction conditions and drawing registration
(2)Quotation	Order Company Estimate	-Order-receiving company : Inquire the bid notice and register the estimated unit price, expected delivery date, production plan, and quotation conditions for the desired bid.
(3)Selection & Matching	Matching_BID	-Ordering company : After looking up the quotation, select a successful bidder that meets the conditions and register the purchase order.
(4)Order issuance confirmation	Order Company	-Order-receiving company: Can use the successful bid notification service and search the issuer's issuance details.
(5)Product production status	Order Company Status	-Order-receiving company : Registers production status information during the product manufacturing process. -Ordering company : Production status information can be inquired.
(6)Delivery	Delevary ProductImage	-Order-receiving company : Register the final production image of the product and process delivery. Register for notification after delivery.
(7)Final inspection	Review AS	-Register product inspection results and process post-service registration.
(8)Etc.	Help Administrator Member All Tablese	-Customer FAQ , Q&A -Manager: member approval and management, order transaction approval and management, authority modification, etc.

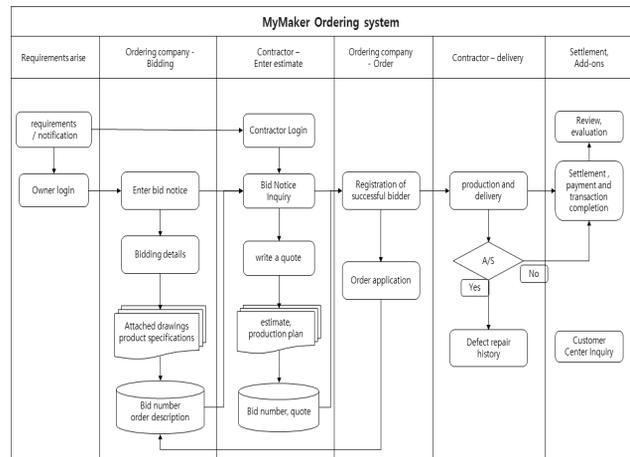


그림 3. 수발주 시스템 흐름 구현도

Fig.3. P-A Order System Flow Chart

입찰광고 등록을 통해 고유 입찰번호(등록 일자 + 발주업체 아이디 + 일련번호)가 자동으로 생성되고 제품 제작 시방서, 도면 파일을 첨부해야 한다. 입찰 진행은 고유 견적서 등록 번호(등록일 + 견적업체 아이디 + 일련번호)가 자동으로 생성이 되고 반드시 하나의 견적서는 필수로 첨부해야 한다. 제작 계획서는 필수는 아니지만 첨부한다면 반드시 하나의 파일로 만들어 올린다. 그리고 낙찰업체 선정은 발주업체가 견적서 가격, 납기 예정일, 수주업체의 신뢰 지수 등을 확인 후 비교 선정을 한다.

발주서 전달을 통해 최종 선정 후 관리자 측에서 승인 허락을 해주어야 진행을 할 수 있다. 발주업체는 발주서를 첨부하여 견적업체가 확인, 인쇄할 수 있도록 하고 수주업체 담당자에게 카카오 문자 전송을 통해 알려주도록 한다. 수주업체에서는 발주서 근거로 제품을 제작을 시작하고 납품하며 납품에 지연이 생길 시 사전 통보 및 사유서를 제출해야 한다. 최종적으로 발주업체에서는 납품 후 검사를 하며 품질에 문제가 있을 시 A/S신청을 할 수 있다. 제작 완료, 제품 납기 거래 종료 시 수발주시스템은 거래 상황 파악과 실적 집계를 통해 추후 다른 업체에서 참고 할 만한 지표가 필요하다. 수주업체에서 제품 제작 완료 시, 발주업체에서 확인 할 수 있도록 성적증명서와 제품 사진을 첨부해야 하고 납품 예정일과 입찰등록자에게 카카오톡 자동 알림 서비스가 가도록 한다. 발주업체에서 납품을 받고 제품의 검사 결과를 입력하며 불량품이 있을 시에는 하자보수(A/S)를 신청한다. 검사 결과 양품이면 거래한 수주업체에 대해 평가를 작성할 수 있도록 시스템을 구현하고자 한다.

### III. 시스템 구현 개발

소프트웨어 개발 전 수발주시스템의 요구사항과 지속적인 변경으로 인해 일정 지연이 있었다. 초기부터 명확한 요구사항 정의를 기반으로 시스템 개발이 이루어질 수 있도록 초기 요구사항의 불명확성을 최대한 명확하게 하려고 노력하였다. 재정의 분석을 통해 요구 공학 프로세스와 방법론을 적용하였다. 프로젝트 초기부터 명확한 개발 목표와 범위를 근간으로 요구사항을 정의하고 시스템 개발을 하는 것이 힘들었다. 그 이유는 IT 기업이 아닌 제조업으로서 요구사항이 공학적 기술에 대한 이해를 기반으로 하고 있지 않으며 요구사항이 추상적 수준으로 정의되는 경우가 많았기 때문이다. 이를 위해 개발 단계에서 수정 보완하며 반영해 나갔다. 지속적인 요구사항 변경 문제는 명세 단계에서 시스템 관점으로부터 요구사항 변경 최소화를 위해 애자일(Agile) 기반 요구사항 정의를 하여 개발 내용을 증분적으로 제공함으로써 개발 후반에 대규모 변경 위험을 최소화 하였다. [4] 기타 해당 수발주 시스템을 구현하기 위해 프론트엔드와 백엔드 개발에 사용된 기술들을 설명하고자 한다.

수발주시스템의 프론트엔드 개발 구현에서는 기본적인

HTML, CSS, JavaScript, phpGrid 를 사용하여 기본적인 UI와 테이블 처리용 CRUD작업들을 페이지 단위로 각 각 구현하였다. Data Grid의 기능들을 편리하게 사용할 수 있고 사용자의 가시성과 편의성을 위해 phpGrid 라는 PHP 라이브러리를 사용하였다. 세부 동적 기능들은 JavaScript와 JQuery를 사용하고 백엔드와의 비동기 통신을 위하여 ajax를 사용했다. FORM안에 데이터를 입력하고 통신, 파일도 넘겨주는 부분들이 많아서 주로 multipart-form data 형식으로 데이터와 파일을 동시에 보내는 작업을 수행하고 텍스트 데이터만 보내도 되는 경우에는 단순 POST 혹은 GET 방식을 활용했다. 각 phpGrid 테이블들의 세부 화면들은 팝업을 이용하는 경우가 많아서 이는 GET방식으로 특정 데이터만 선별해서 보여줄 수 있게끔 하였다. 보내어진 파일을 다시 불러와서 팝업을 띄우는 것도 병행했다.

수발주시스템의 백엔드는 스프링부트를 활용하여 편리한 MVC 구조로 적용하고 유지보수와 확장성이 용이하도록 개발 하였다. DAO 역할은 JPA 구현체인 Hibernate ORM을 사용해 구현하였다. 로그인과 회원가입 등 API의 권한과 인가 관리를 위해 JWT 토큰 인증 방식을 사용하였다. 복잡한 커넥션 풀 관리와, CRUD처리들을 빠르도록 Hibernate JPA를 사용하여 HikariCP를 통한 커넥션 관리와 손쉽게 JPAREpository<T, K>를 상속함으로써 CRUD를 구현하였다. JPA는 테이블과 객체 간 패러다임 불일치를 극복시켜주는 자바 진영의 ORM 라이브러리로서 러닝 커브가 높으나 기술이해를 하여 사용할 경우 자바로 DB 작업을 할 때 큰 도움이 된다. 수발주 시스템의 백엔드의 DB 처리는 모두 JPA 이용해 처리하였다. JPA는 스프링 기본 설정 파일인 application을 참조하기 때문에 JPA 관련 설정은 application.yml에 명시하였다. 요구사항에 어떤 데이터베이스를 사용할지, 파일 데이터를 어디에 어떤 형식으로 저장할지 관련해 구체적인 명시가 없기에 언제든지 변하는 요구사항에 유연한 대처가 가능하도록 서비스, 레포지토리 계층을 추상화하였다. 그림4는 서버 구현 프로젝트 구성도이며 클라이언트로부터 요청은 RestController에서 받고 요청에 관련한 비즈니스 로직은 서비스계층에서 수행할 수 있도록 구현하고 서비스 계층에서는 파일 시스템관리 및 Repository 계층을 사용해 데이터를 저장 및 관리하도록 구현하였다.

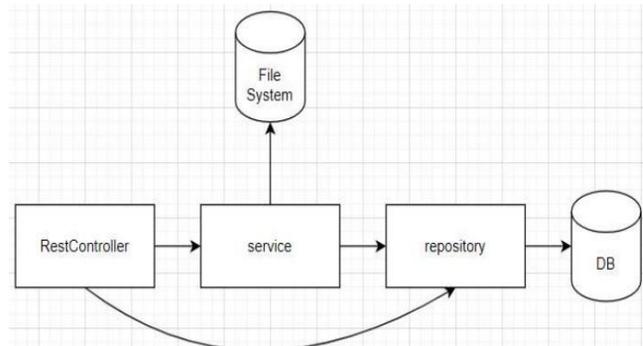
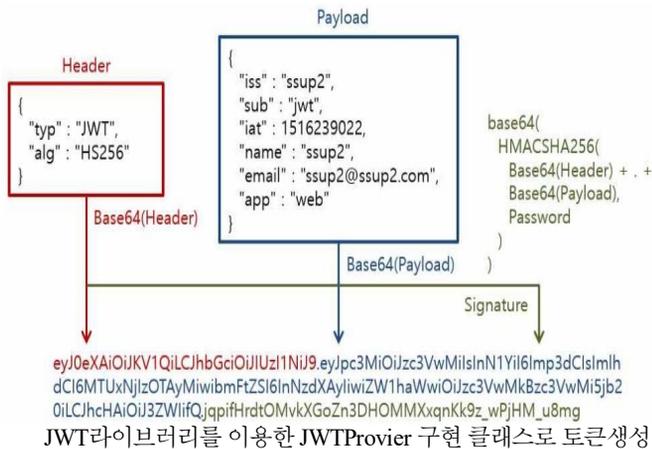


그림 4. 수발주 시스템 백엔드 프로젝트 구성도  
 Fig. 4. System backend project structure diagram

회원관리를 위해서 API 인증과 인가 관리는 회사의 요구사항이 하나의 인증 정보가 여러 곳에서 동기화가 되어야 해서 쿠키-세션을 이용해 요구사항을 구현할 수 있겠지만 이 방식대로 한다면 세션을 동기화할 DB를 별도로 구축해야 하고 이를 관리할 인증 서버도 구축해야 하며 프로젝트의 아키텍처의 복잡도, 리소스, 비용을 고려하여 JWT를 사용하기로 결정하였다. JWT는 쿠키 세션 방식과 다르게 서버에 별도의 세션 테이블을 생성하지 않아 리소스 면에서 절약되며 확장이 용이하다.

다음은 사용자 인증 관련 부분이다. 수발주 시스템에서는 일반적인 세션 방식이 아닌 JWT 토큰 방식으로 사용자의 인증과 인가를 처리한다. 그림5는 JWT Token Structure 이며 그림 6은



JWT라이브러리를 이용한 JWTProvider 구현 클래스로 토큰생성과 JWT 토큰의 유효성 체크 메소드이다.

그림 5. JWT 토큰 구조도

Fig. 5. JWT token structure diagram

```

@Component
public class JwtProvider {

    @Value("${jwt.secret}") private String secretKey;

    // 토큰 생성
    public String createToken(String subject) {
        Date now = new Date();
        Date expiration = new Date(now.getTime() + Duration.ofDays(1).toMillis());

        return Jwts.builder()
            .setHeaderParam(Header.TYPE, Header.JWT_TYPE)
            .setIssuer("JO_YOUNG_JUN")
            .setIssuedAt(now)
            .setExpiration(expiration)
            .setSubject(subject)
            .signWith(SignatureAlgorithm.HS256,
                Base64.getEncoder().encodeToString(secretKey.getBytes()))
            .compact();
    }

    // Jwt 토큰의 유효성 체크 메소드
    public Claims parseJwtToken(String token) {
        token = BearerRemove(token);
        return Jwts.parser()
            .setSigningKey(Base64.getEncoder().encodeToString(secretKey.getBytes()))
            .parseClaimsJws(token)
            .getBody();
    }

    private String BearerRemove(String token) {
        return token.substring("Bearer ".length());
    }
}
    
```

그림 6. JWT 라이브러리를 이용한 구현

Fig. 6. Implementation using JWT library

세션은 사용자 정보를 서버에 저장하는 반면 토큰은 각 클라이언트의 쿠키 혹은 로컬 스토리지에 저장된다. 트래픽 관점으로 보면 토큰의 크기는 세션보다 훨씬 크기 때문에 세션이 더욱더 효율적이며, 보안 쪽으로 세션은 데이터를 서버가 저장하고 있어 토큰 방식보다 더 안전하다. 그래도 토큰 방식을 채택한 이유는 확장성을 고려하였기 때문이다. 서버를 확장할 때 보통 scale-out 방식으로 진행하여 여러 대의 서버를 두고 로드 밸런서를 배치해 트래픽을 분산시킨다. http는 stateless, connectionless 한 성질을 가지고 있어서 이 경우에 세션을 사용할 경우 각 서버에 세션 정보를 동기화시켜야 하는 문제점이 발생한다. 그리고 토큰은 클라이언트가 저장하고 있고 서버는 단지 클라이언트로부터 받은 토큰을 검증하면 되기에 세션 방식에 비해 확장성이 뛰어나다.

기타 모든 API는 /scm/api/\* 하위에 정의되어 있으며 서비스 사용에 있어 반드시 인증이 필요하다. 인증 체크는 위의 경로에 위치한 모든 컨트롤러가 해야하며 각각의 컨트롤러에 검증하는 코드를 추가하는건 상당히 반복적이고 비효율적이다, 그래서 인터셉터와 위에 첨부한 코드를 이용해 /scm/api/\* 위치한 컨트롤러 요청을 제한하는 공통 관심사 처리를 하였다. 아래 그림 7과 그림8에 관련한 구현을 살펴볼 수 있다. 그리고 아래 그림 9와 그림10은 위의 구현 흐름을 표현한 것으로 인터셉터 적용 후 흐름과 인증의 요청이 들어온 경우 인터셉터의 흐름을 확인할 수 있다.

```

@Slf4j
@RequiredArgsConstructor
public class AuthorizationInterceptor implements HandlerInterceptor {

    private final JwtProvider jwtProvider;

    @Override
    public boolean preHandle(HttpServletRequest request, HttpServletResponse response, Object handler)
        throws Exception {
        String token = request.getHeader("Authorization") != null ? request.getHeader("Authorization") : null;
        if (token == null) { throw new NeedTokenException("NEED_AUTHORIZATION"); }
        Claims claims = jwtProvider.parseJwtToken(token);
        return true;
    }
}
    
```

그림 7. 인터셉터를 이용한 접근 제어로직 구현

Fig. 7. Implementation of access control logic using interceptor

```

private final JwtProvider jwtProvider;

@Override
public void addInterceptors(InterceptorRegistry registry) {
    registry.addInterceptor(new AuthorizationInterceptor(jwtProvider))
        .order(1)
        .addPathPatterns("/scm/**");
}
    
```

그림 8. 구현된 인터셉터를 적용할 경로를 등록

Fig. 8. Register a route to apply the implemented interceptor

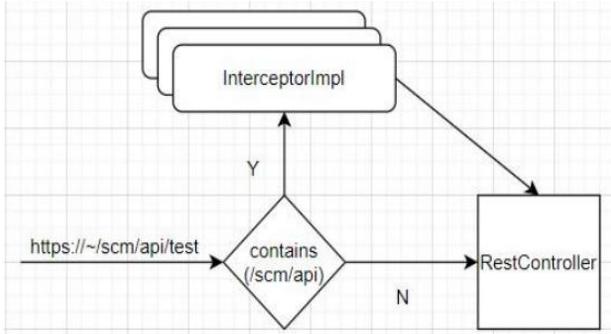


그림 9. 인터셉터 적용 후 흐름  
Fig. 9. Flow after applying the interceptor

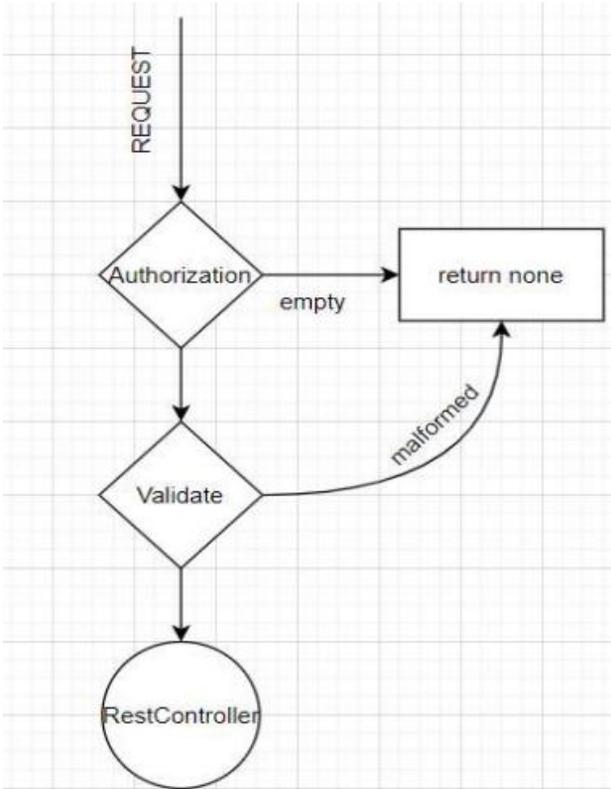
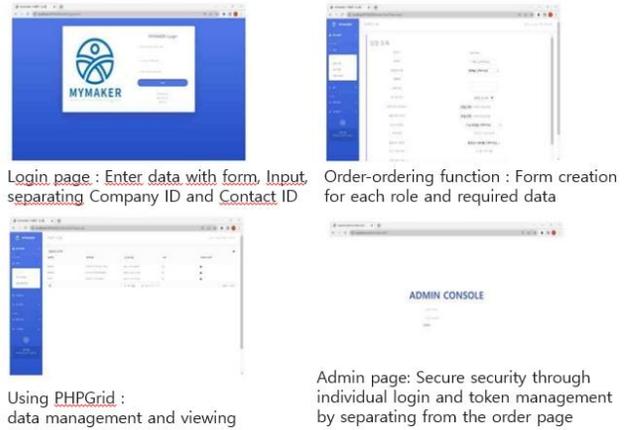


그림 10. 인증 요청 인터셉터 적용 흐름  
Fig. 10. Authentication request interceptor application flow

#### IV. 최종 시연 테스트 및 결과

로그인 페이지는 회사의 요구 사항에 따라 회사 ID와 담당자 ID로 분리 설계 구현되어 해당 부분을 테스트 하였다. 회원가입 방식을 가이드하여 테스트 시연을 완료하였다. 역할과 필요한 데이터에 맞는 Form을 생성하여 각 메뉴에 대한 접근성을 높이기 위해 iframe을 활용하여 바로 바로 redirect가 되는지 테스트하였다. 입력된 데이터관리 및 열람은 phpGrid를 통해 데이터를 조회 및 개발하였으며 관련 CRUD 기능들이 요구사항

에 맞게 구동되는지 테스트를 하였다.



Login page : Enter data with form. Input separating Company ID and Contact ID  
Order-ordering function : Form creation for each role and required data  
ADMIN CONSOLE  
Admin page: Secure security through individual login and token management by separating from the order page  
Using PHPGrid : data management and viewing

그림 11. 구현된 화면과 기능 테스트  
Fig. 11. Implemented Screens and Functional Test

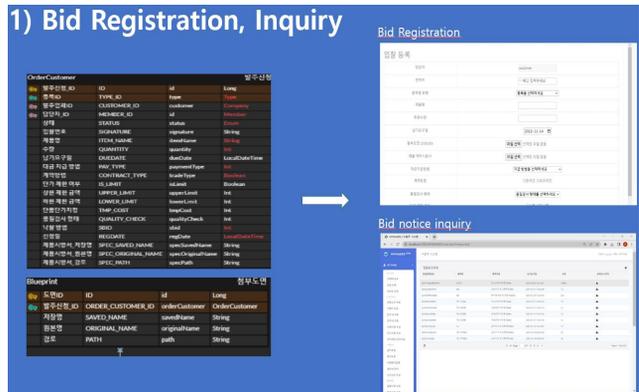
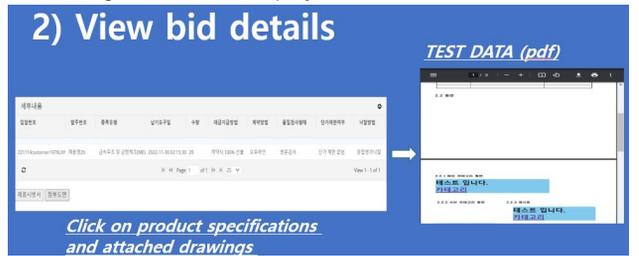


그림 12. 발주 업체 일차 공고 등록 등록 및 조회 테스트  
Fig. 12. Ordering company's temporary notice, registration and inquiry test



3) Estimate input and inquiry

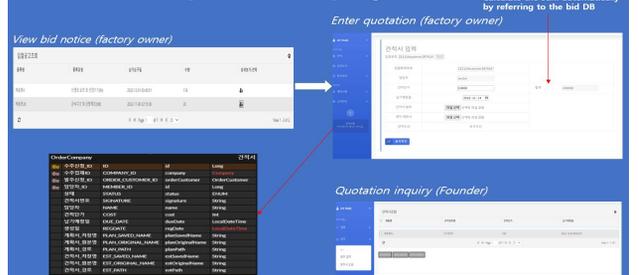


그림 13. 제조 업체 입찰 상세보기 후 견적서 등록 테스트  
Fig. 13. After viewing the bid details, the manufacturer's quotation registration test



그림 14. 낙찰 선정 후 발주 발행 및 계약 처리  
Fig. 14. Order processing contract after selection



그림 15. 제작 납품 후 제품 만족도 평가 처리  
Fig. 15. Product satisfaction evaluation processing after production and delivery

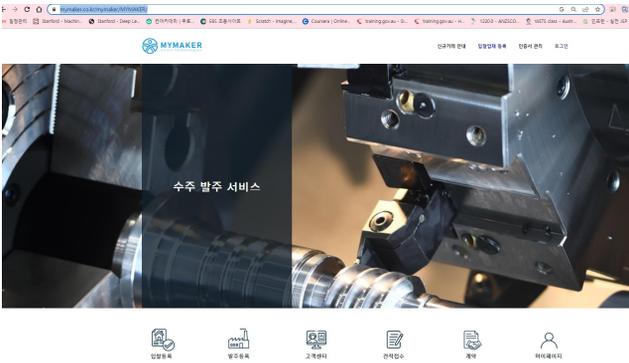


그림 16. 제작 시스템 메인화면  
Fig. 16. Manufacturing system main page

기타 관리자 페이지를 원래 있던 기존 메뉴들과 분리하여 관리자만 접근 가능하도록 구현 개발하여 관리 테스트 시연을 마무리하였다. 테스트 시나리오에 따른 시연을 무리 없이 성공 완수하였다. 해당 요청 산업체 대표가 테스트 참여하여 관련 결과 시연을 살펴보고 만족하였으며 운영 서버에 Deploy는 해당 산업체에서 전문 SI개발업체 서버 엔지니어가 담당할 수 있도록 관련 부분 인수인계 시간을 가지고 시연 테스트 후 최종 소스 설명 과 기타 deploy 과정을 보여주고 산업체로 인수인계하였

다.

지역 내 협력업체들과 영세 소 상공 기업들에 필요한 물량을 공급할 수 있도록 지원하며, 부품이 필요한 장비 업체들에 빠르게 공급을 지원하여, 원활한 기업 활동이 이루어지도록 지원할 수 있도록 향후 실제 운영 프로젝트 개발 전에 사전에 충분한 프로토타이핑과 시연을 하는 좋은 성과를 거두었다. 향후 해당 프로토타이핑 결과물을 기반으로 실질 운영 가능한 프로젝트가 개발된다면 지역의 소기업을 대상으로 하는 만큼 지역경제의 활력에 도움이 될 것으로 판단된다. 현재 업체가 잘 운영 중인 ‘공유 온 팩토리’ 사이트의 협력업체들을 대상으로 실증 테스트 후 신규 협력업체들과 다양한 품목별 수주 및 발주 거래 공정 과정으로 확대하여 고도화 개발을 할 때, 중요한 실증 및 연구 개발에 토대가 되는 실증적 구현 연구결과이다. 기능적, 비기능적 요구사항을 분석 및 데이터베이스 설계 구현을 하고 웹으로 개발하여 실증 및 효율성을 분석할 수 있는 좋은 결과로 판단된다.

## V. 결론

본 논문에서는 실증적인 연구의 결과로 다음과 같은 결론을 얻을 수 있었다. 공장 공유 서비스를 위해 세부 서비스 아이템인 수주발주 시스템을 제작하여 연동하면 고객의 확보를 쉽고 빠르게 가져갈 수 있다. 4차산업 및 디지털전환을 위한 근본적인 과제로 제조 기업의 유연성으로서 공유공장은 기존 제조 기업의 유휴 설비, 장비, 공간, 인력 등을 활용할 수 있는 계기를 제공하고 제조 분야 창업을 하고자 하는 수요자에게는 공장을 건설하지 않고 아이디어나 노하우 만으로도 신속하게 사업을 시작할 수 있는 기회를 제공하여 유연성과 성장을 촉진하는 사업 모델이라 할 수 있다. 서비스업과 같이 시장의 변화에 적극적이고 빠르게 대응할 수 있는 제조업 플랫폼 서비스이며, 제조업의 생산량 증가, 감소, 업종 전환 등의 유연성을 확보할 수 있는 것이다. 높은 비용과 구축 기간이 긴 제조 설비 자원을 전국적으로 세계적으로 신속하게 균등하게 배분할 수 있는 제조 공유 플랫폼 사업 모델임이 틀림없다. 탈세계화 및 국익우선주의로 빠르게 세계적인 추세에 따라 국가적으로 국내 제조업의 현황과 모든 역할을 빠르게 찾고, 공유하여 국가적인 제조업 전반의 경쟁력을 확보하는 기회이다. 협력 산업체 (주)마이메이커의 공유 온 팩토리 서비스와 연동되어 고객사들의 수발주 서비스를 제공한다면 그 파급 효과는 더욱 클 것이다.

## Acknowledgments

본 연구는 2022년도 인천재능대학교 LINC3.0사업 산업체 애로 기술 해결 공동 프로젝트 연구 결과로 수행하였습니다. 인천재능대 산학협력단 관계부처의 지원에 감사드립니다.

## References

- [1] Segye Ilbo, a Korean language newspaper. Available: <https://www.segye.com/newsView/20220428516583>.
- [2] MYMAKER(Korean:마이메이커), a Koran company site. Available: <https://www.mymaker.co.kr/index-mymaker>
- [3] M. Sulemani (7 April 2021). "CRUD operations explained: Create, read, update, delete". Retrieved 14 December 2021. Available: <https://www.educative.io/blog/crud-operations>
- [4] A. Rasheed, B. Zafar, T. Shehryar, N.A. Aslam, M. Sajid, N. Ali, S.H. Dar, S. Khalid "Requirement Engineering Challenges in Agile Software Development", *Mathematical Problems in Engineering*, Vol. 2021, Article ID 6696695, 18 pages, 2021. Available: <https://doi.org/10.1155/2021/6696695>



### 서연경 (Youn-Kyoung Seo)

(現)인천재능대학교 시컴퓨터정보과 교수  
 인하대학교 컴퓨터공학(인공지능연구실) 박사修了  
 경북대학교 컴퓨터공학(자연어처리연구실)석사卒業  
 경북대학교 컴퓨터공학 학사卒業  
 서울대학교 의생명지식공학연구소 선임연구원  
 (주)LG전자 MC연구소 선임연구원

※관심분야 : 인공지능활용, 웹앱, 시스템개발, 모바일, 데이터베이스 등,