

# 클라우드 기반 O2O 소독 서비스 시스템 설계 및 구현<sup>☆</sup>

## Design and Implementation on Cloud-based System for O2O Disinfection Services

장 예 진<sup>1</sup>                      백 종 호<sup>2\*</sup>  
Ye-jin Jang                      Jong-ho Paik

### 요 약

코로나19의 발생으로 인한 방역 및 소독업무의 중요성이 높아짐에 따라 소독원을 모집하고 싶은 소독사업자와 일하고 싶은 소독원의 수요가 증가했지만, 구인·구직은 쉽지 않은 실정이며, 소독 업무에 대한 자동화와 업무를 효율적으로 관리할 수 있는 시스템이 필요하다. 따라서 본 논문에서는 O2O 소독 서비스 시스템을 MVC 패턴을 사용하여 설계하고, MySQL, Ejs/Bootstrap 뷰, Node.js를 통해 구현한다. 또한, AWS 클라우드 서비스와 연동하여 사용자가 언제 어디서나 시스템을 사용할 수 있게 한다. 제안한 O2O 소독 서비스 시스템을 통해 국내 소독방역 시장의 인력 채용의 어려움을 해결하고, 기존의 비효율적인 소독 업무 프로세스 개선을 기대한다.

☞ 주제어 : O2O, 소독 서비스 시스템, MVC패턴, 클라우드

### ABSTRACT

Due to the outbreak of COVID-19, the domestic disinfection and quarantine market has grown, and the demand for disinfection operators who want to recruit disinfectants and disinfectants who want to work has increased, but it is not easy to find jobs. In addition, there is a need for a system that can automate or efficiently manage work by hand-recording and writing a disinfection record for disinfection work. Therefore, in this paper, an O2O disinfection service system is designed using the MVC pattern and implemented through MySQL, Ejs/Bootstrap view, and Node.js. Additionally, it connects with AWS cloud services so users can use the system anytime, anywhere. Through the proposed O2O disinfection service existing system, we hope to solve the difficulties of recruiting personnel in the domestic disinfection station market and improve the existing inefficient disinfection work process.

☞ keyword : O2O, Service System for Disinfection, MVC Pattern, Cloud

## 1. 서 론

코로나19로 인한 비대면의 일상화는 많은 영역에서의 디지털 전환을 불러왔다. 디지털 전환으로 인해 사회 전반에 여러 가지 다양한 변화가 나타났으며, 대표적으로 ‘빅블러(Big Blur)’ 현상을 들 수 있다. 빅블러는 ‘생산자-소비자, 소기업-대기업, 온·오프라인, 제품 서비스 간 경계 용화를 중심으로 산업/업종 간 경계가 급속하게 사라

지는 현상’을 의미한다[1-2]. 온·오프라인 채널 단절을 극복하고 온라인의 편리함과 오프라인의 현장성을 결합하는 O2O(Online to Offline) 서비스는 빅블러 현상의 대표적인 예로 꼽히고 있다. 사물인터넷(IoT), 빅데이터(Big Data), 인공지능 등 ICT 발전에 힘입어 성장하고 있는 O2O는 현재 음식 배달, 숙박, 교통 등 다양한 영역에서의 생활 밀착형 서비스 형태로 공급되고 있다[3-5].

특히, 코로나19의 발생으로 인한 방역 및 소독업무의 중요성이 높아짐에 따라 소독원을 모집하고 싶은 소독사업자와 일하고 싶은 소독원 각각의 수요는 넘쳐나지만, 실상은 구인·구직조차 쉽지 않은 실정이다. 현재 소독·방역 시장에서의 인력은 대부분 아르바이트 형식으로 모집되며, 소독사업자가 고용한 정직원 소독원의 비율은 상당히 낮으며, 이러한 이유는 법으로 정해진 계절별 소독 시행 횟수가 건물이나 시설에 따라 다르기 때문이다. 따라서 주로 단기 계약직이나 일반 인터넷 카페 또는 지인 소개 등을 통해 소독원을 모집하고 있다. 이와 같은 구인·

1 Dept. of Computer, Major in Software Fusion, Graduate School, Seoul Women's University, Seoul, 01797, Republic of Korea

2 Dept. of Software Convergence, Seoul Women's University, Seoul, 01797, Republic of Korea

\* Corresponding author : paikjh@swu.ac.kr

[Received 18 January 2024, Reviewed 23 January 2024, Accepted 4 February 2024]

☆ This work was supported by a research grant from Seoul Women's University (2023-0187).

☆ This manuscript is an addition based on the first author's master's thesis from Seoul Women's University.

구직 방식은 새로 교육을 이수한 소독원이 인지하고 있지 못하면 구직하기 어렵다는 문제가 있어서 결국 인력 충원이 어렵다는 문제가 발생한다.

또 다른 문제점으로, 소독 업무 처리의 어려움을 들 수 있다. 국내 소독업체들의 경우, 법정업무 소독 건물이나 시설에 소독원들을 파견할 때 대부분 수기로 소독실시대장을 기록하고 있다. 아파트 같은 공동주택의 경우 소독 실시대장을 1차 수기로 작성 후, 회사로 복귀하여 문서작성 프로그램을 이용한 별도의 소독보고서 및 소독 증명서를 2차로 작성하고 발송하고 있다. 이와 같은 업무수행 방식은 번거롭다는 단점이 있어, 업무를 자동화하고 체계적으로 관리 가능한 시스템이나 솔루션은 전무한 상황이다.

따라서 앞서 제시한 문제점을 해결하기 위해 방역·소독시장에서의 O2O 서비스 시스템의 필요성이 대두되고 있다. 먼저, 사업자등록을 완료한 소독사업자와 법정 교육 이수를 완료한 소독원만 이용이 가능한 소독시장에서의 맞춤형 인력 모집 시스템이 필요하다. 이때, 기존 일반적인 구인·구직 시스템과는 다르게 인력 충원이 더 원활하게 이루어지도록 소독원 추천 기능이 요구된다. 또한 전체 소독업무 프로세스를 관리해줄 독자적인 소독 시스템이 필요하다. 번거로웠던 기존의 업무수행 방식을 효율적으로 처리할 수 있도록 업무 체계를 자동화하여 하나의 시스템 안에서 관리가 가능하도록 하는 시스템이 요구된다.

본 논문에서는 소독사업자, 소독원 각각의 사용자 요구사항을 반영하여, 소독사업자들의 공고에 따라 소독원을 모집하는 기능을 제공한다. 또한 소독실시대장 및 소독 보고서 작성의 전산화를 통해 소독 업무 구조 개선을 위한 새로운 서비스 시스템 설계 및 구현 방법을 제시한다.

## 2. 클라우드 기반 O2O 소독 서비스 시스템 설계

본 논문에서 제안하는 O2O 소독 서비스 시스템은 시스템에 적용되어야 할 사용자 요구사항을 바탕으로 기능을 정의하며, 소프트웨어 디자인 패턴인 MVC(Model-View-Controller) 패턴을 사용하여 설계한다[6]. 모델(Model)에는 DB를 통한 데이터 관리, 뷰(View)에는 웹과 앱 화면 처리, 컨트롤러(Controller)에는 Node.js 기반 서버를 통해 사용자 명령을 모델과 뷰로 라우팅하는 역할을 한다. 일반적으로 Node.js 기반 서버에는 확장성과 처리 성능이 높은 NoSQL 형식의 MongoDB를 사용하지만 본 논문에서 제안한 O2O 소독 서비스 시스템은 모집 공고 데이터나 거래 주체의 개인 정보 보안을 위하여 MySQL을 사용

하여 DB를 구현한다[7].

### 2.1 사용자 및 시스템 기능 요구사항 분석

사용자 요구사항이란 사용자가 시스템에서 어떠한 작업 처리가 가능한지를 명시하는 요구사항이며, 기능 요구사항이란 사용자들이 요구하는 작업을 처리할 수 있도록 개발된 시스템에서의 기능들을 정리한 것을 의미한다. 사용자 요구사항을 정의하는 목적은 사용자를 위한 것이고, 기능 요구사항을 정의하는 목적은 시스템 개발자를 위한 것으로 볼 수 있다.

본 논문에서 제안한 O2O 소독 서비스 시스템의 주 사용자는 소독사업자와 소독원으로 나뉜다. 먼저, 소독사업자는 회원가입 시 제안한 시스템 내에서 사용할 수 있는 소독사업자의 고유번호가 자동으로 생성된다. 소독원은 소독 교육 이수 완료자만 시스템 가입이 가능하며, 회원가입 시 소독원 고유번호가 자동으로 생성된다. 소독사업자는 소독공고를 게시할 수 있으며, 소독공고에는 공고제목, 공고분류, 공고내용, 일 시작 날짜, 일 종료 날짜, 소독 장소, 모집인원, 모집지역, 모집조건이 포함된다. 공고 게시 시 공고 번호는 자동으로 부여되며, 소독사업자는 여러 개의 공고를 올릴 수 있다. 한 공고에는 여러 명의 소독원이 신청할 수 있으며, 소독사업자가 최종적으로 승인한 소독원만 실제 업무가 가능하다. 소독사업자는 소독 업무가 끝난 소독원에 한하여 근태, 친절도, 적극성, 소독 업무 관련지식의 네 가지 기준에 따라 소독원을 평가하며, 해당 평가 내용은 각 소독원의 경력 정보로 저장된다.

정의한 사용자 요구사항에 따른 기능 요구사항은 총 다섯 가지로 나눌 수 있다. 첫 번째로 O2O 소독 서비스 시스템은 회원 정보 관리 기능을 갖춰야 한다. 회원 정보 관리 기능은 소독사업자, 소독원의 주체별로 요구되는 회원 정보를 정의하고 관리한다. 시스템을 이용하기 위해서 소독사업자는 사업자 등록번호, 소독원은 소독 교육 이수 번호를 입력하여만 회원가입이 가능하다. 따라서 시스템은 해당 정보를 포함한 나머지 개인 정보들을 관리하는 기능을 제공해야 한다. 두 번째로 시스템은 공고 정보 관리 기능을 제공해야 한다. 공고 정보 관리 기능은 소독사업자가 등록하는 공고에 대한 정보를 정의하고 관리한다. 소독사업자는 소독원을 모집하는 공고를 등록하고 승인하도록 하며, 소독원이 공고를 신청한 신청 정보는 따로 관리한다. 세 번째는 일정 정보 관리 기능을 제공해야 한다. 소독원이 신청하고 업무 승인을 받은 공고에 한하여 개인별로 해당 일정을 관리할 수 있도록 기능을 제공해

야 한다. 네 번째는 경력 정보 관리 기능을 제공해야 한다. 제안한 시스템을 통하여 업무를 진행하고 완료한 경우, 소독사업자는 반드시 소독원에 대해 평가를 하도록 사용자 요구사항에서 정의되어 있다. 따라서, 개인별 경력을 관리하는 기능을 제공해야 한다. 마지막으로 시스템은 소독원 매칭 기능을 제공해야 한다. 각 공고별로 조건에 맞는 소독원을 추천하여 소독사업자와 1대 1 매칭을 시켜주는 기능을 갖추어야 한다. 표 1에서 분석한 기능 요구사항이 보여진다.

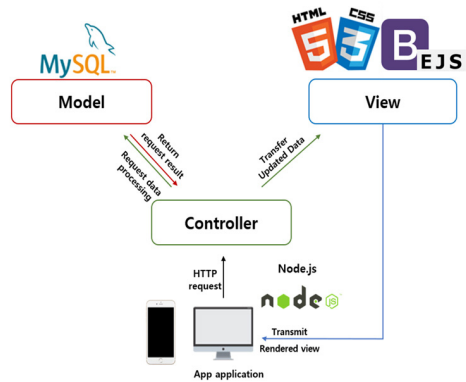
(표 1) 시스템 기능 요구사항  
(Table 1) Requirement of System Function

기능 분류	세부 기능	세부 요구사항
사용자 정보	주체별로 요구되는 사용자 정보 정의	소독사업자는 사업자 등록 번호, 소독원은 교육 이수 번호를 포함한 정보 등록 및 관리
공고 정보	소독사업자가 등록하는 공고에 대한 정보 정의	소독원이 공고 신청이 가능하도록 공고 등록 및 신청, 승인 관리
일정 정보	소독원 개인별 일정 정의	공고 승인에 따른 소독원 일정 관리 및 개인별 일정 등록, 관리
경력 정보	소독원 개인별 경력 정의	시스템을 통한 업무 완료에 따른 개인별 경력 관리
소독원 매칭	조건에 따른 소독원 추천	공고별 조건에 맞는 소독원을 추천하여 소독사업자와 매칭

## 2.2 클라우드 기반의 서버-클라이언트 MVC 패턴 설계

MVC 패턴이란 소프트웨어 공학에서 자주 사용되는 소프트웨어 디자인 패턴이다. MVC 패턴은 애플리케이션의 구성요소를 세 가지로 나누어 관리하고, 소프트웨어의 비즈니스 로직과 사용자 인터페이스를 구분하는 데에 중점을 둔다. MVC 패턴의 각각의 부분은 별도의 컴포넌트로 나뉘어 있어서 서로 영향을 받지 않고 작업을 수행한다. 일반적으로 모델(Model)은 애플리케이션의 데이터를 관리하는 역할을 한다. 뷰(View)는 모델에서 데이터를 가져온 뒤 사용자에게 보여주는 역할을 한다. 즉, 애플리케이션의 레이아웃과 화면을 처리한다. 그림 1에서 보여지는 MVC 패턴에서는 뷰를 여러 개 가질 수 있으며, 모델에서 전달받은 데이터를 별도로 저장하지 않고 화면에 띄우는 역할을 한다. 컨트롤러(Controller)는 모델과 뷰를 이어주는 매개체 역할을 한다. 컨트롤러는 클라이언트(Client)로부터 받은 입력을 처리한다.

본 논문에서 제안하는 O2O 소독 서비스 시스템은 MVC 패턴이 적용되어, 모델은 MySQL 데이터베이스, 뷰는 HTML/EJS 템플릿 및 xml 기반의 Bootstrap 뷰, 컨트롤러는 Node.js 서버가 해당된다. 설계한 MVC 패턴 구성도는 그림 1과 같다.



(그림 1) MVC 패턴 서버 구성도  
(Figure 1) Server Configuration based on MVC Pattern

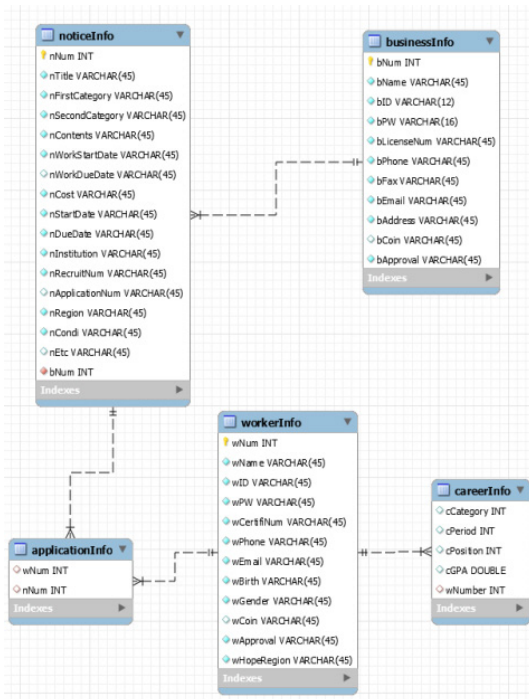
### 2.2.1 Model : MySQL 기반 데이터 모델 설계

데이터 모델은 구축할 서비스 시스템의 요구사항을 명확하게 표현할 수 있는 도구이다. 데이터 모델은 데이터베이스 설계 시 이용되며, 데이터의 구조를 논리적으로 설명하기 위해 사용된다. 본 논문에서 제안한 시스템의 데이터 모델은 논리적 데이터 모델을 의미하며, MySQL 기반 데이터베이스에 적용할 수 있도록 설계되었다. 또한, 2.1절에서 정의한 사용자 요구사항에 따라 설계하였다. 사용자 요구사항에 따른 E-R(개체-관계) 다이어그램은 그림 2와 같다. 먼저, 제안한 O2O 소독 서비스 시스템의 이용 주체 중 하나인 소독사업자에 대한 개인 정보는 businessInfo라는 이름의 테이블에 저장한다. businessInfo 테이블은 이름(bName), ID(bID), 비밀번호(bPW), e-mail 주소(bEmail), 사업장 주소(bAddress), 사업자등록번호(bLicenseNum)를 컬럼(column)으로 구성한다.

다음으로 다른 시스템 사용자인 소독원에 대한 개인 정보는 workerInfo라는 테이블에 저장한다. workerInfo 테이블은 이름(wName), ID(wID), 비밀번호(wPW), 휴대폰 번호(wPhone), e-mail 주소(wEmail), 생년월일(wBirth), 성별(wGender), 소독 희망 지역(wHopeRegion), 소독 교육 이수증 번호(wCertifiNum)를 컬럼으로 구성한다.

소독사업자는 제안한 시스템 이용 시 소독원 모집을

공고할 수 있으며, 공고 정보를 저장할 테이블이 noticeInfo 테이블이다. 공고는 공고제목(nTitle), 공고분류(nFirstCategory/nSecondCategory), 공고내용(nContents), 일 시작 날짜(nWorkStartDate), 일 종료 날짜(nWorkDueDate), 모집 시작 날짜(nStartDate), 모집 종료 날짜(nDueDate), 모집인원(nRecruitNum), 모집지역(nRegion), 모집조건(nCondi) 등의 내용을 포함하기 때문에 해당 정보를 noticeInfo 테이블의 컬럼으로 설정한다. noticeInfo 테이블의 PK는 공고고유번호(nNum)로, AI로 설정되어 자동으로 부여된다.



(그림 2) 설계한 데이터 모델의 E-R 다이어그램  
(Figure 2) E-R diagram of designed Data Model

소독원의 경력정보를 저장하는 테이블은 careerInfo로, 경력 카테고리(cCategory), 경력 기간(cPeriod), 평점(cGPA) 등의 컬럼으로 구성된다. cGPA 컬럼의 경우, 소독사업자에게 평가된 근태, 친절도, 적극성, 소독업무 관련 지식의 네 가지 기준에 따라 매겨진 총점이 저장된다.

소독원이 공고를 신청했다는 정보를 담고 있는 테이블은 applicationInfo 테이블이며 신청한 공고 번호(nNum)와 소독원 고유 번호(wNum)를 FK(Foreign Key)로 가진다.

## 2.2.2 View: 웹/앱 기반 Bootstrap View 설계

본 논문에서 제안한 O2O 소독 서비스 시스템은 사용자에 따라 웹/앱으로 나누어 시스템을 사용할 수 있도록 한다.

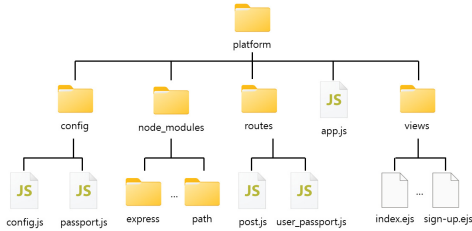
먼저, 소독사업자에 해당하는 사용자는 웹 브라우저 기반 시스템을 사용한다. 따라서 HTML 기반 Node.js 템플릿인 EJS(Embedded JavaScript Templates)를 사용하여 뷰 화면을 설계한다. EJS는 HTML 문법을 그대로 사용하여 뒤 추가로 자바스크립트 문법을 사용할 수 있다[8].

Bootstrap은 프론트엔드 개발 시 라이브러리를 이용해 개발을 간소화 할 수 있도록 만들어 놓은 오픈 소스 프레임워크이다[9]. 여러 종류의 레이아웃, 버튼, 입력폼, 이미지, 아이콘 등의 디자인과 기능을 HTML과 CSS, JavaScript 기반으로 미리 정의해두었기 때문에 간편하게 사용자 인터페이스 디자인 적용이 가능하다는 장점이 있다. 화면 해상도에 따라서 크기가 자동으로 맞춰지는 그리드 기반의 반응형 웹으로 구성되어 있기 때문에 모바일 화면도 지원할 수 있다.

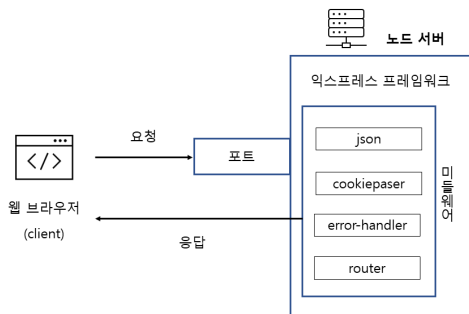
## 2.2.3 Controller: Node.js 기반 페이지 라우팅 설계

Node.js는 Chrome V8 자바스크립트(JavaScript) 엔진으로 빌드된 자바스크립트 런타임 (RunTime)이다. Node.js를 통해 다양한 자바스크립트 기반 애플리케이션을 실행할 수 있지만, Node.js는 서버 애플리케이션을 실행하는데 가장 많이 사용된다[10]. 본 논문에서 제안한 시스템은 Node.js를 사용하여 Controller를 설계한다. 앞서 설계한 O2O 소독 서비스 시스템의 사용자는 소독사업자와 소독원으로 나뉜다. 따라서 라우팅 모듈은 소독사업자와 소독원이 공통으로 사용하는 모듈과 시스템 기능 요구사항에 따른 주제별 모듈로 나누어 설계한다. 기본적으로 라우팅 모듈은 사용자의 웹 브라우저를 통한 URI 또는 HTTP 요청을 분기시켜 처리해주는 역할을 한다[10]. 이 역할은 Node.js의 express 모듈이 담당하여, 이 모듈을 사용한 서버 라우팅 구조 설계도는 그림 3에서 보여진다. 그림 3에서의 바와 같이 메인 서버 파일인 "app.js"는 express 모듈을 사용하여 구동한 서버 코드를 포함한다. app.js에서 만드는 서버는 클라이언트의 요청을 받아서 처리한 이후 클라이언트에게 요청에 따른 결과값을 응답한다. 이때, Node.js의 미들웨어를 사용하여 요청과 응답을 조작하기도 하고, 나쁜 응답을 걸러내기도 한다. 익스프레스 서버의 구조도는 그림 4에서 보여진다. 제안한 O2O 소독 서비스 시스템의 환경 설정을 담당하는 코드 파일이 담겨져 있는 [config] 폴더는 두 개의 파일로 구성된다.

config.js 파일은 MySQL DB 연동과 관련된 환경설정 코드가 포함된다. passport.js 파일은 passport 모듈을 사용하여 LocalStrategy 방식의 사용자 인증 방식을 통한 시스템 회원가입과 로그인과 관련된 코드를 구현하여 포함한다.



(그림 3) Express 모듈 기반의 라우팅 구조 설계도  
(Figure 3) Routing Structure Blueprint based on Express Module

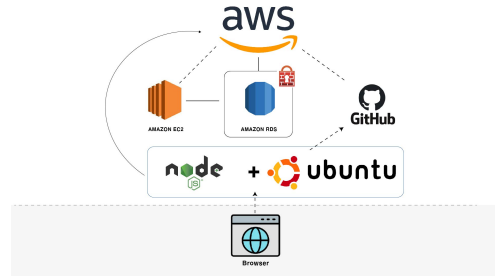


(그림 4) 익스프레스 서버 구성도  
(Figure 4) Server Configuration of Express Server

### 2.2.4 AWS 기반 아키텍처 설계

전 세계 클라우드 서비스 시장은 매년 성장세를 기록 중이다. 클라우드는 하드웨어와 소프트웨어를 포함한 각종 IT 자원을 인터넷에 접속하여 시간과 장소에 구애받지 않고 필요한 만큼 자원을 활용할 수 있게 하는 기술 및 서비스를 말한다. 최근 빅데이터와 인공지능의 중요성이 커지면서 4차 산업혁명의 기본 인프라로 클라우드가 주목받았다[11-13].

클라우드는 시간과 장소, 이용 기기 등에 따른 사용 제약이 없고 이용량이 급격하게 증가해도 유연하게 대처할 수 있다. 또한 서비스 인프라를 증축시켜야 하는 상황에도 신속한 인프라 도입이 가능하며 전 세계 어디에서나 서비스를 제공하는 것이 가능하다는 장점이 있다[14].



(그림 5) AWS 연동 서버-클라이언트 구성도 (web 기준)  
(Figure 5) Server-Client Configuration linked AWS

다시 말해, 온프레미스 환경에서 클라우드 서비스로의 전환은 비용 절감뿐만 아니라 미리 구축되어있는 전문기업의 서비스 또는 서비스 컴포넌트를 이용해 기업에 필요한 정보서비스를 신속하게 만들 수 있어 기업의 민첩성을 증가시키기도 한다[15].

클라우드 서비스를 사용하여 시스템을 개발할 경우, 인터넷 연결이 되는 모바일 기기를 사용하면, 언제 어디서나 해당 시스템에 접속 가능하다. 본 논문에서 제안한 O2O 소독 서비스 시스템은 시스템의 접근성을 높이기 위하여 AWS 클라우드 서비스와 연동이 가능하다.

AWS 클라우드 서비스를 이용한 서버-클라이언트 (server-client) 구성도는 그림 5에 보여진다. 앞서 3.2.3절에서 설계한 Node.js 서버는 AWS EC2에 배포하고, MySQL DB는 AWS RDS와 연동하여 사용한다. 또한, 외부 저장소로 Github를 사용하여 코드를 관리한다.

## 3. 클라우드 기반 O2O 소독 서비스 시스템 구현

### 3.1 MySQL 기반 데이터 모델 구현

2장에서 설계한 데이터 모델을 기반으로 구현한 MySQL 기반 데이터베이스의 테이블은 표 2에서 보여진다. 데이터베이스에 정의된 테이블은 표 2에서 보는 바와 같이 총 5개이다.

각 테이블의 이름은 businessInfo, workerInfo, noticeInfo, careerInfo, applicationInfo이다. 앞서 언급한 바와 같이 businessInfo, workInfo, noticeInfo에는 각각 소독사업자, 소독원, 공고정보를 식별할 수 있는 고유 번호인 bNum, wNum, nNum이 기본키(PK, Primary Key)로 지정되어 있다. 공고는 어떤 소독사업자가 올렸는지를 참조하기 위해 bNum을 외래키(FK, Foreign Key)로 지정하며, 어떤 공고를 어떤 소독원이 신청했는지를 참조하기 위해 applicationInfo

의 외래키는 wNumber과 nNum로 지정되어 있다. 경력 정보 또한 어떤 소독원의 경력정보인지를 참조하기 위해 wNumber가 외래키로 지정되어 있다.

소독사업자 정보를 관리하는 테이블인 businessInfo의 PK(Primary Key)인 소독사업자 고유번호(bNum)는 데이터 타입이 AI(Auto Increment)로 설정되어 자동으로 부여된다. 회원가입 시 사업자 등록증이 있는 소독사업자만 회원가입이 가능하도록 설계하여 시스템 관리자의 승인이 필요하다. 소독사업자가 회원가입 신청 폼을 제출하면 승인을 의미하는 bApproval 컬럼의 기본값이 0으로 설정되어 저장된다. 시스템 관리자는 사업자등록번호를 확인한 이후 bApproval의 값을 1로 수정해주도록 설계하였고, 해당 값이 1인 사용자만 시스템 이용이 가능하다.

workerInfo 테이블의 PK인 소독원 고유번호(wNum)도 데이터 타입을 AI로 설정하여 자동으로 부여되게 한다. businessInfo 테이블과 마찬가지로 회원가입 시 소독 교육 이수를 완료한 소독원만 회원가입이 가능하도록 설계하여 시스템 관리자의 승인이 필요하다. 소독원이 회원가입 신청 폼을 제출하면 승인을 의미하는 wApproval 컬럼의 기본값이 0으로 설정되어 저장된다. 시스템 관리자는 소독 교육 이수증 번호를 확인한 이후 wApproval의 값을 1로 수정하고, 해당 값이 1인 사용자만 시스템 이용이 가능하도록 설계하였다. 소독 희망 지역은 소독원이 일하기를 원하는 지역을 사전에 입력받아 DB에 저장하고 향후 소독 사업자에게 추천해줄 때의 기준 중 하나로 정할 수 있다.

Node.js에서의 MySQL 연동에는 mysql이라는 모듈을 사용한다. db\_info라는 객체 안에 데이터베이스 연동에 필요한 host 주소, user 및 database 이름, DB password, port 번호를 입력하여 설정한 후 mysql.createConnect 쿼리를 통해 데이터베이스 연결을 시도한다.

### 3.2 웹/앱 기반 Bootstrap View 구현

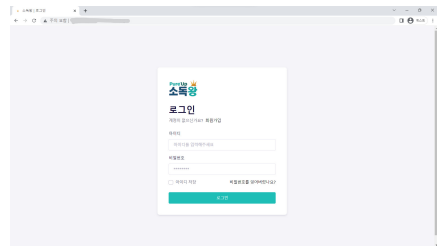
앞서 2장에서 설계한 뷰를 바탕으로 구현한 부트스트랩 뷰는 그림 6, 그림 7과 같다. 그림 6에서 보는 바와 같이 소독사업자가 웹 브라우저를 이용해 시스템에 접속한 화면이며, 그림 7에서는 소독원이 안드로이드(Android) 애플리케이션을 사용하여 시스템에 접속한 화면이다.

그림 6과 7에 보는 바와 같이 제한한 O2O 소독 서비스 시스템은 사용자별로 시스템 접속 기기를 웹과 앱으로 나누어 다르게 사용하며, 이에 따른 사용자별 뷰도 다르게 나타난다. 소독사업자 전용 뷰는 Node.js 템플릿인 ejs로 구현하며, 소독원 전용 뷰는 xml로 구현한다.

(표 2) DB 테이블 목록 및 기본 정보

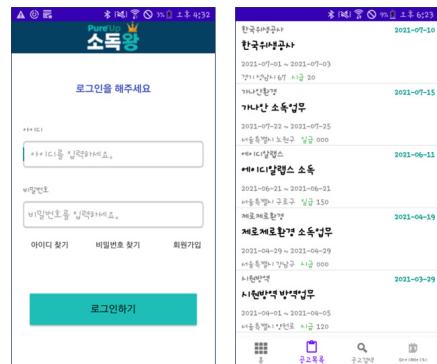
(Table 2) List and Basic Information of DB Tables

테이블명	설명	기본키	외래키
business-Info	소독사업자의 회원 정보 명시 테이블	bNum	-
worker-Info	소독원의 회원 정보 명시 테이블	wNum	-
notice-Info	소독사업자가 등록한 공고 정보 명시 테이블	nNum	bNum
career-Info	소독원의 경력 정보 명시 테이블	-	wNumber
application-Info	소독원이 신청한 내용과 공고 정보가 명시된 테이블	aNum	wNumber, nNum



(그림 6) 웹 브라우저에서의 소독사업자 전용 뷰 화면 (Figure 6) Disinfection Operator's Dedicated View Screen in Web Browser

안드로이드 앱은 사용자 요구사항에 따라 주로 ListView를 사용하여 구현하였다. 각 리스트의 컴포넌트는 서버에서 공고정보를 json 형태로 가져온 후 텍스트로 변환하여 사용자에게 보여지도록 구현되었다.



(그림 7) 안드로이드 애플리케이션에서의 소독원 전용 뷰 화면 (Figure 7) Disinfection Worker's Dedicated View Screen in Android Application

### 3.3 Node.js 기반 페이지 라우팅 구현

본 논문에서 제안한 O2O 소독 서비스 시스템은 Node.js의 여러 모듈을 통해 서버를 구현한다. 먼저, 웹 서버 생성과 관련된 기능을 담당하는 express 모듈을 사용한다. 모듈을 사용할 때는 반드시 require 함수를 통해 모듈을 불러온 뒤 사용해야 한다. 또한, 시스템 뷰 엔진은 ejs를 사용하여 구현하기 때문에 기본 셋팅을 ejs로 해준 뒤, ejs 파일을 렌더링 해올 때 html처럼 사용하도록 설정한다. 기본 환경 설정이 끝난 뒤 서버는 listen 메소드를 이용하여 실행한다.

사용자 로그인 및 회원가입 기능은 일반적으로 세션과 쿠키 처리 등 복잡한 작업이 많으므로 검증된 모듈을 사용하는 것이 바람직하다[10]. 따라서 Node.js의 Passport 모듈을 이용한다. 표 3에서는 app.js 파일에 Passport 모듈을 연동시키는 코드를 제시한다. 표 3에서 보여지는 passport.initialize() 미들웨어는 req 객체에 passport 설정을 포함하고, passport.session() 미들웨어는 req.session 객체에 passport 정보를 저장한다. req.isAuthenticated 메소드는 회원가입보다는 로그인을 구현할 때 중요한 역할을 한다. req.isAuthenticated는 passport 모듈에서 제공하는 함수이며, res.locals에 인증 정보와 현재 사용자 정보를 포함하여 세션이 끊기지 않는다면 해당 정보를 계속 사용할 수 있도록 도와준다. 시스템 사용자가 로그인 중이면 req.isAuthenticated는 true, 아닌 경우 false를 반환한다.

제안한 시스템 기능 중 공고와 관련된 기능은 크게 공고목록 조회, 공고 등록, 공고 상세내용 조회로 분류할 수 있다. 이 각각의 세부 기능을 listpost, addpost, showpost라는 함수로 기능을 분리한 뒤 post.js라는 하나의 모듈에서 통합하여 관리한다. 그림 8에서는 그 중 listpost 기능으로, 소독사업자의 공고목록 조회 화면을 보여준다. noticeInfo 테이블에서 전체 공고 정보를 모두 가져오는데 모집 마감이 임박한 공고를 상단에 배치하기 위해 nDueDate 컬럼을 기준으로 ORDER BY를 먼저 실행시킨 뒤 데이터를 조회한다. 그림 9에서는 소독원 모바일 어플리케이션을 통해 소독공고 조회 및 신청하는 화면을 보여준다.

제안한 O2O 소독 서비스 시스템의 핵심 기능 중 하나는 소독사업자에게 소독원을 추천해주는 1:1로 매칭해주는 것이다. 그와 더불어 전체 소독원 중 소독사업자가 원하는 조건에 맞는 소독원을 검색하는 기능도 제공한다. 소독원 추천 서비스는 소독원 활동 또는 업무 희망 지역, 성별, 나이 등의 소독원 평가모델을 기반으로 추천된다.

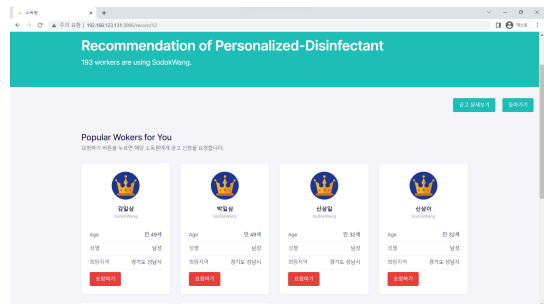
(표 3) Passport 모듈 활용 코드 (app.js)  
(Table 3) Application Code of Passport Module

```
var passport = require('passport');
...
//====패스포트 사용 설정====//
app.use(passport.initialize()); //패스포트 초기화
app.use(passport.session()); //로그인 세션 유지

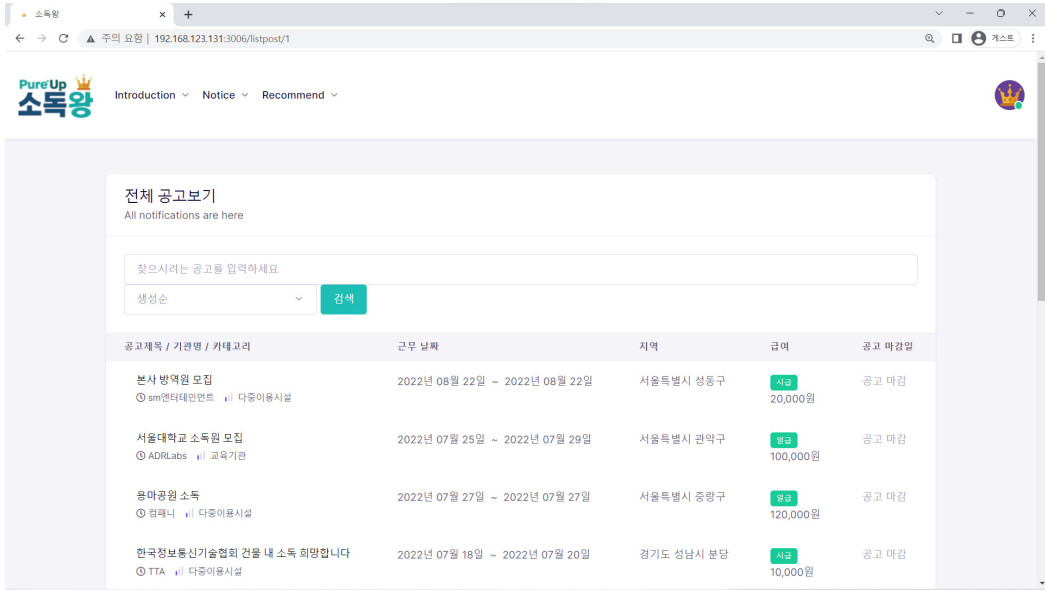
app.use(function(req,res,next){
  res.locals.isAuthenticated = req.isAuthenticated();
  res.locals.currentUser = req.user;
  next(); });
```

소독원은 앞서 제시한 세 가지 조건을 회원가입 시 기재하며, 소독사업자는 소독공고 게시 시 우대조건으로 성별, 나이 등을 기재한다. 소독사업자가 공고를 올리는 순간 시스템 내에서 공고별 우대조건에 해당하는 소독원을 찾아 그림 10와 같이 보여준다. 결과적으로 소독사업자는 공고를 직접 신청한 소독원 목록을 확인하는 것과 별개로 소독원을 추천받을 수 있으며, 공고 모집인원 미달 시 추천 기능을 통해 추천 받은 소독원을 채용할 수 있다.

모집기간이 종료되고 소독사업자가 소독원 채용하기 버튼을 누른 뒤 소독원을 채용하면, 서버에서는 소독원 채용 이벤트가 발생된다. 채용 이벤트는 Firebase 클라우드 메시징(FCM, Firebase Cloud Messaging)을 사용하여 구현한다. FCM은 iOS, Android, 웹에서 메시지를 무료로 주고받을 수 있도록 하는 서비스이며, 메시지를 쉽게 타겟팅하여 특정 주제를 구독하는 기기에 타겟팅한 메시지를 전송하여 팝업으로 보여주는 형식으로 구현된다.



(그림 10) 소독원 실시간 매칭 화면  
(Figure 10) Disinfection Worker's Realtime Matching View



(그림 8) 소독사업자 공고목록 조회 화면  
(Figure 8) Check View of Disinfection Operator's Announcement List



(그림 9) 소독원 모바일 앱의 소독공고 조회 및 신청 화면  
(Figure 9) Disinfection Announcement Check View and Application View of Disinfection Worker's Mobile App



소독원이 채용되어 오프라인에서 소독작업을 시행하면, 소독사업자는 시스템에서 소독 작업물을 실시간으로 모니터링 할 수 있다. 어느 곳의 소독이 완료되었는지, 어느 곳이 진행 중인지 한 번에 확인할 수 있으며, 전체 소독 작업장 목록 중 소독이 완료된 목록의 퍼센트를 구하여 작업률을 표시한다.

#### 4. 결 론

본 논문에서는 온라인의 편리함과 오프라인의 현장성을 결합한 O2O 서비스 시장의 발전에 따라, 서비스 분야 중 하나인 소독·방역 시장에서의 활용을 위한 O2O 소독 서비스 시스템을 설계 및 구현하였다. 코로나19의 발생으로 국내 소독·방역 시장이 성장하여 소독원을 모집하고 싶은 소독사업자와 일하고 싶은 소독원의 수요가 증가했지만 구인·구직이 쉽지 않은 상황이다. 소독 업무 처리 또한 소독실시대장을 수기로 기록하고 작성하여 업무의 자동화 또는 효율적으로 업무를 관리해줄 시스템이 필요한 상황이다.

본 논문에서 제안한 클라우드 기반 O2O 소독 서비스 시스템은 소프트웨어 설계에 사용되는 패턴 중 하나인 MVC (Model-View-Controller) 패턴을 적용하여 설계 및 구현하였다. 각 사용자 요구사항을 분석하고 기능 요구사항을 정의하여 각 사용자의 주체에 따라 웹/앱으로 시스템 사용 가능 기능을 나누어 구현하였다. Model로는 MySQL 데이터베이스, View로는 웹/앱 기반 Bootstrap 뷰, Controller에는 Node.js 기반 페이지 라우팅 방법을 적용하여 설명하였으며, 해당 설계를 바탕으로 구현한 Node.js 기반 소독 서비스 시스템의 구조 및 화면을 구현 결과로 제시하였다[16]. 또한, 사용자가 언제 어디서든 서비스를 이용할 수 있도록 제시한 시스템은 AWS 클라우드 서비스와 연동하였다.

제안한 시스템을 통해, 국내 소독·방역 시장에서의 인력 모집에 대한 어려움을 해결하여 인력 공급 문제를 해결하고, 비효율적으로 진행되는 기존의 소독 업무 처리 방식 개선이 가능하다. 번거로웠던 기존의 업무 체계를 제안한 시스템 하나로 처리할 수 있게 하여 업무를 효율적으로 관리하도록 서비스를 제공하며, 소독사업자와 소독원 모두 불필요한 업무 과정 없이 소독업무에만 온전히 집중하도록 하여 더욱 강화된 방역체계를 기대할 수 있다.

#### 참고문헌(Reference)

- [1] 강신형, “개방형 혁신, ‘빅블러’ 시대의 생존 전략,” DBR Column, 324호, 2021.  
[https://dbr.donga.com/article/view/1203/article\\_no/10093/ac/magazine](https://dbr.donga.com/article/view/1203/article_no/10093/ac/magazine)
- [2] 박정완, “빅블러, 산업 간 경계가 없어지는 시대 이제는 융합이다,” 융합경영리뷰, VOL 22, pp.2-5, 2021.  
<https://db.koreascholar.com/Article/Detail/406168>
- [3] 과학기술정보통신부, 정보통신산업진흥원, 2021 O2O 서비스산업시장조사결과보고서, 2022.  
<https://www.msit.go.kr/bbs/view.do?mId=99&bbsSeqNo=79&nttSeqNo=3173465>
- [4] Eun Jung Choi, “Current situation of food delivery O2O platform industry and policy suggestions,” J. Channel Retail, VOL 26, NO. 1, pp.171-197, 2021.  
<http://dx.doi.org/10.17657/jcr.2021.01.31.8>
- [5] Dongsoo Kim, Kwanho Kim, Donguk Choe, Jae-Yoon Jung, “Service Issues and Policy Directions for Promoting the O2O Industry in Korea,” The Journal of Society for e-Business Studies, Vol. 21, No. 4, pp.137-150, 2016.  
<http://dx.doi.org/10.7838/jsebs.2016.21.4.137>
- [6] <https://developer.mozilla.org/ko/docs/Glossary/MVC>
- [7] 박우창, 남송휘, 이현룡, MySQL로 배우는 데이터베이스 개론과 실습, 한빛아카데미, 2019년.
- [8] Sammie Smith, FULL STACK WEB DEVELOPMENT GUIDE, kindleunlimited, 2022.
- [9] 이성용, 김태근, Do it! 장고+부트스트랩 파이썬 웹 개발의 정석, 이지스퍼블리싱 2021년.
- [10] 조현영, Node.js 교과서(개정3판), 길벗, 2022년.
- [11] 조윤정, “디지털뉴딜의 핵심인프라, 클라우드 산업 생태계 동향,” KDB미래전략연구소 산은조사월보, 제782호, pp. 21-40, 2021.  
[https://rd.kdb.co.kr/FLSRIA02N01.act?\\_mnuId=FYERE R0017#\\_init\\_](https://rd.kdb.co.kr/FLSRIA02N01.act?_mnuId=FYERE R0017#_init_)
- [12] Azure, “퍼블릭, 프라이빗 및 하이브리드 클라우드란?,” <https://azure.microsoft.com/ko-kr/resources/cloud-computing-dictionary/what-are-private-public-hybrid-clouds/#overview>
- [13] 가비아 라이브러리, “클라우드란 무엇인가,” <https://library.gabia.com/contents/infrahosting/9114/>
- [14] 이영호, 한동수, “당신이 지금 알아야 할 AWS,” 비제

이퍼블릭, 2019.

- [15] S. Lee and J. Paik, "An Asynchronous-Driven Node.js Based Intermediary-free Direct Deal Distribution Platform Converged with Cloud Service," KSII Transactions on Internet and Information Systems, vol. 13, no. 8, pp. 4212-4226, 2019.  
<http://doi.org/10.3837/tiis.2019.08.022>

- [16] 장예진, 클라우드 기반 O2O 소독 서비스 시스템 설계 및 구현, 서울여자대학교 일반대학원, 석사학위논문 2023.

## ❶ 저 자 소 개 ❶



### 장 예 진(Ye-jin Jang)

2021년 서울여자대학교 소프트웨어융합학과(공학사)  
2023년 서울여자대학교 일반대학원 컴퓨터학과 소프트웨어융합전공(공학석사)  
2023년~현재 클로잇 Solution사업팀 선임 컨설턴트  
관심분야 : 서비스 개발, 데이터베이스, 클라우드, 로우코드  
E-mail : yjjang@cloit.com



### 백 종 호(Jong-ho Paik)

1994년 중앙대학교 전기공학과(공학사)  
1997년 중앙대학교 일반대학원 전기공학과(공학석사)  
2007년 중앙대학교 일반대학원 전자전기공학부(공학박사)  
1997년~2011년: 한국전자기술연구원 모바일단말융합센터 센터장  
2011년~현재: 서울여자대학교 소프트웨어융합학과 교수  
관심분야 : 서버/클라이언트 지능형 플랫폼, 빅데이터 전처리 및 기계 학습, 소프트웨어 테스트, 유무선 방송통신 시스템 설계, 영상통신 시스템 설계 및 시스템 아키텍처  
E-mail : paikjh@swu.ac.kr