

공 컨테이너 리포지셔닝 사이트 개발에 대한 연구

배 석 태[†]

A Study on Website Development for Empty Container Repositioning

Suk Tae Bae[†]

ABSTRACT

As the container market is expected to continue to increase, the volume of containers will increase, and the volume of empty containers will be proportional to this. Imbalances between imports and exports in each country have emerged, and the issue of collection and storage of empty containers is becoming more and more important, and it is necessary to appropriately deal with container demand for container presents. Therefore, in this paper, we are trying to develop a website that provides various empty container information and provides empty container leasing services based on this information. Through the development of these empty container websites, we would like to make empty container leasing and operations more efficient, improve empty container-related services and reduce costs.

Key words: Empty Container, Repositioning, Index Page, Query Code

1. 서 론

최근 컨테이너 화물 운송시장의 규모가 커짐에 따라 20,000TEU 이상을 운송할 수 있는 초대형 컨테이너 전용 선박이 출현하면서 컨테이너 선박의 대형화가 더욱 진전되고 있다[1]. 이와 같은 국제 해상운송 환경의 변화로 인해 컨테이너 선사가 화물 운송을 위해 보유해야 하는 공 컨테이너의 보유량이 급격하게 증가하게 되었다. 이에 따라 컨테이너 선사는 공 컨테이너 용기 및 관련 기기의 구입과 임대에 막대한 자본을 투입하고 이러한 관련 장비의 사용 및 관리에 많은 부대비용을 부담하고 있다. 이를 해소하기 위해 컨테이너의 화물적재효율의 정량화를 위한 방안이 고려되고 있다[2]. 그리고 시간이 지날수록 수출입 국가들 간의 수출입 불균형이 부각되어지면서 자연

스럽게 공 컨테이너가 남아도는 상황이 급격하게 증가하게 되었다. 공 컨테이너는 전략적으로 배치되는 경우 컨테이너 정기선사의 중요한 자산이 될 수 있지만, 공 컨테이너를 운송·회수·보관할 때 소요되는 대부분의 시간은 선사의 수익에 상당한 부담으로 작용하고 있다[3].

이러한 컨테이너의 보편화로 인해 공 컨테이너에 관련한 문제가 대두됨에 따라 이를 해결하기 위해서 공 컨테이너 임대를 효율적으로 할 수 있도록 IT기술을 활용한 임대 사이트의 필요성과 중요성이 부각되었다. 미국의 시장조사기업인 451리서치에서 발표한 컨테이너 시장 규모에 관한 전망에 따르면 2016년의 전 세계 컨테이너 시장의 규모를 7억 6,200만 달러로 집계했으며 앞으로 2020년까지 매년 40%씩 늘어나 2020년에는 26억 달러에 이를 것으로 전망했다[4].

※ Corresponding Author : Suk Tae Bae, Address: (48520) Sinseon-ro 428, Nam-gu, Busan, Korea, TEL : +82-51-629-1433, FAX : +82-51-629-1419, E-mail : st1281@naver.com

Receipt date : Aug. 16, 2019, Revision date : Aug. 29, 2019

Approval date : Aug. 30, 2019

[†] Dept. of Port Logistics systems, Tongmyong University

※ This Research was supported by the Tongmyong University Research Grants 2017(2017A022)

이처럼 컨테이너 시장이 앞으로도 더욱 더 증가할 것으로 예측됨에 따라 자연스레 컨테이너의 량이 늘어나게 될 것이고, 그에 따라 공 컨테이너의 량 또한 늘어나게 될 것이다. 이에 더해, 각 나라마다의 수출입 불균형이 발생하면서 공 컨테이너의 회수 및 보관 문제가 더욱 부각되어지고 있어 컨테이너 선사들의 컨테이너 수요에 적절한 대처가 필요하다[5].

따라서 본 논문에서는 공 컨테이너에 대한 각종 정보를 제공해주고 이를 토대로 공 컨테이너 임대 서비스를 해주는 웹 사이트를 개발 하고자 한다. 이러한 공 컨테이너에 관한 웹 사이트 개발을 통하여 공 컨테이너 임대 및 운영을 더욱 효율적으로 할 수 있도록 하여 공 컨테이너 관련 서비스를 향상시키고 비용도 줄이고자 한다.

2. 본 론

2.1 컨테이너의 유통과정

보통 컨테이너는 크게 LCL(Less Than Container Load) 컨테이너, FCL(Full Container Load) 컨테이너, 공 컨테이너의 3가지로 분류되어 유통 및 관리되고 있으며, 그 중 가장 먼저 LCL 컨테이너의 유통 과정에 대해 설명하고자 한다.

수입 컨테이너 중 LCL화물은 현재는 터미널 CFS(Container Freight Station) 또는 ODCY(OnDock Container Yard) CFS로 우선적으로 배정되고 있다. 터미널에서 반출된 컨테이너는 CFS로 이송되어 컨테이너로부터 화물을 인출한다. 세관으로의 업무보고는 화물이 CFS 창고에 입고된 후 이루어지고 있다. 통상 화물의 물품은 수입신고, 통관 작업이 이루어진 다음 인출할 수 있게 되며, CFS에서의 적출이 완료되어진 컨테이너는 공 컨테이너화가 되어 수출을 위해 바로 송화주의 문전으로 가게 된다.

다음으로는 FCL컨테이너의 유통 과정에 대해 설명하고자 한다. FCL컨테이너는 LCL 컨테이너와는 다르게 따로 CFS적출이 필요 없기 때문에 바로 수화주의 문전으로 이송되어 지거나 아니면 일시적으로 CY(Container Yard)에 보관했다가 바로 수화주의 문전으로 이송된다. 그 후 화물이 반출되어져 공 컨테이너화가 되고 그 수화주가 바로 송화주가 되거나 인접지역의 수화주가 송화주가 되어 재수출을 준비하게 된다. 또는 수화주의 문전에서 화물 반출이 완

료된 공 컨테이너가 다시 CY에 회수되어 보관되어 지다가 필요에 따라 재수출을 위해 또 다른 송화주의 문전으로 가게 된다. 이러한 공 컨테이너 운영 관리와 관련하여 발생하는 주된 비용 요소를 살펴보면 다음과 같이 크게 3가지로 구분할 수 있다.

첫째, 지역 간 공 컨테이너의 수요와 공급의 불균형으로 인해 공 컨테이너 배송 시스템에는 필연적으로 균형 물동량(Balancing Flow)이 발생하게 되는데 이러한 균형 물동량을 재배치하기 위한 재배치 비용이 발생한다.

둘째, 각 지역 또는 데포별 공 컨테이너 수요 및 공급의 변화와 불확실성에 신속적으로 대응하기 위해서는 데포에 적정한 수준의 재고를 유지해야 하는 데 이를 위한 재고유지 비용이 발생한다.

셋째, 갑작스러운 선박 운항 스케줄 변경이나 재배치 계획 시의 지역별 수요 및 공급 예상 또는 예측의 부정확으로 인해 필요한 시기에 공 컨테이너가 공급되지 못할 때, 단기 임대 할 경우의 단기 임대비용이 발생한다.

해운선사의 공 컨테이너 관리 업무를 살펴보면, 담당자는 모든 지역의 영업부서로부터 화물 선적 예상량(Booking Prospect, B/P)을 접수하고, 이를 바탕으로 장래 특정 시점의 화물수송수요를 충족시키기 위해 보유하고 있어야 하는 컨테이너 보유수량(Standard GroundPool, S.G.P)을 사정한다. 그리고 각 지점별 재고 현황을 파악하고 예상되는 공 컨테이너의 유출과 유입을 고려하여 지점별 공 컨테이너 재고를 예상한다. 이와 같은 지점별 보유수량, 예상 재고, 그리고 이송시간을 고려하여 공 컨테이너 수급 계획을 수립하고 선박 스케줄과 여유 선복을 확인하여 재배치 계획을 세운다. 만일 자사 소유의 공 컨테이너를 사용하여 수요를 충족할 수 없을 경우에는 단기 임대계획을 수립하게 되고, 이 때 담당자는 임대회사가 공급할 수 있는 컨테이너 수량, 계약 기간, 계약 조건 등을 고려하여 필요한 수량을 단기 임대하여 재배치한다. 한편, 각 지역의 영업 관련 부서에서 공 컨테이너 재배치 및 임대를 요청할 경우 담당자는 이를 반영하여 재배치 및 임대 계획을 수립한다. 담당자가 공 컨테이너 수급 계획을 수립하여 실제 재배치되기까지는 상당한 시간이 소요되기 때문에 통상 2-4주 전에 수급계획이 수립되고 늦어도 1주 전에는 모든 재배치 및 임대계획이 수립되어 시행되게 된다.

보통 해운선사의 공 컨테이너 운영 관리 관점에서 가장 바람직한 화물 운송 전략은 각 지역에서 화물 운송의 결과로 발생하는 공 컨테이너의 공급량과 해당 지역의 화물 운송 요청으로 인한 공 컨테이너 수요량이 일치하면 된다. 또한 화주의 공 컨테이너 요청시기와 화물 운송 완료로 공 컨테이너가 발생하는 시기가 일치하여 수입업체에서 회수되는 공 컨테이너가 바로 수출업체로 이동하여 완벽하게 수요와 공급이 균형을 이루어 추가 임대나 데포에서의 재고를 최소화하는 것이 가장 이상적인 경우이다. 그러나 실제 화물 운송 과정에서는 필연적으로 지역 간의 공 컨테이너 수급 불균형 현상이 발생하고, 지역 내에서도 수요와 공급이 매 시간마다 변화함으로 현실적으로는 어렵기 때문에 본 논문에서 언급했던 것과 같이 자사 컨테이너와 임대 컨테이너를 적절히 사용하여 수급 조절을 효율적으로 관리하는 것이 중요하다[6].

2.2 공 컨테이너 리포지셔닝

앞서 언급했던 공 컨테이너의 운영에 관련된 비용을 줄이고 공 컨테이너를 효율적으로 관리하기 위한 방법의 하나로 나타난 것이 바로 공 컨테이너 리포지셔닝이며, 본 논문에서는 이러한 공 컨테이너 리포지셔닝에 대해 알아보고 공 컨테이너의 재배치 방법 및 기존의 리포지셔닝의 문제점과 해결 방안에 대해 설명하고자 한다.

우선, 공 컨테이너 리포지셔닝(Container Repositioning)은 공 컨테이너(Empty Container)를 필요한 곳에 효율적으로 배치하는 것을 말한다. 일반적으로 컨테이너는 선사의 자산이며 비용이기도 하다. 앞에서 언급했듯이 대개 각 해운선사들은 일정량의 자가(선사 소유) 컨테이너와 리스(돈 내고 빌림) 컨테이너를 운용한다. 하지만 그 컨테이너들을 제대로 쓰지 못하고 방치하고 있다면 그것은 자산이 아닌 비용이 된다. 또한, 보통 각 나라마다 수입과 수출이 일정하지 않다.

예를 들어, A국의 주 수출품은 광물인데 주 수입품이 일반화물이라면 A국으로 들어오는 컨테이너는 많으나, 나가는 컨테이너는 적을 것이다. 이렇게 되면 흔히 말하는 공 컨테이너가 쌓이게 되며, 그것은 곧 관리비용으로 직결된다. 하지만 B국에서의 상황은 이와 반대의 경우라면, 컨테이너가 항상 부족하게 된다. 이 때, 선사들은 컨테이너가 부족할 때 마다

공 컨테이너를 매번 구입하거나 리스하기에는 무리가 있음은 분명하다. 따라서 A국에 쌓인 공 컨테이너를 B국으로 이동시키거나 혹은 그 거리가 너무 멀어 효율이 떨어진다면 인근의 공 컨테이너를 해당 지역으로 보내는 절차가 필요하게 된다.

물론, 이해하기 쉽게 단순히 A국과 B국을 비교하여 설명했지만 의심할 여지없이 이 과정은 세계 모든 나라에서 일어나는 일이다. 이럴 경우에 선사들은 가장 적은 비용으로 빠르게 공 컨테이너를 공급할 방법을 찾아야 하는데 그 방법이 바로 '공 컨테이너 리포지셔닝' 이라고 한다.

2.3 기존의 리포지셔닝의 문제점 및 해결방안

이러한 공 컨테이너 리포지셔닝(Container Repositioning)은 각 나라마다 무역의 불균형으로 인하여 발생하는 컨테이너의 과부족 현상을 해결하기 위해 제안된 아이디어이다. 하지만 기존의 공 컨테이너 리포지셔닝 관련 연구에서는 그저 공 컨테이너 이송을 최소화하여 공 컨테이너 재배치 관련 비용을 최소화하는 것에 초점을 맞추었다. 하지만 이러한 방법으로는 공 컨테이너 재배치에 관련된 비용을 줄이는데 한계가 있었다. 또한 기존의 공 컨테이너는 화물이 없는 상황에서도 많은 공간을 차지하여 공간 및 재배치 관련 비용 및 운영에 낮은 효율성을 보이며, 불필요한 재배치 과정으로 인해 재배치 비용이 과도하게 들어가는 등의 한계에 부딪히게 된다. 따라서 최근에는 이러한 한계를 극복하고자 공 컨테이너 재배치에 관련된 비용을 최소화하고자 하는 새로운 방안으로 기존의 공 컨테이너와 규격은 동일하지만 형태에 변화를 준 접이식 공 컨테이너에 대한 개발 및 연구가 진행되고 있다[7,8].

접이식 컨테이너는 컨테이너 자체를 기계 또는 인력을 이용하여 접고 펼 수 있는 컨테이너를 말하며, 컨테이너 제조사에 따라 다양한 작업방법 및 형태를 가지고 있다. 이러한 접이식 공 컨테이너는 기존의 컨테이너에 비해 양·적화 작업의 간편화를 통한 비용 절감과 동시에 컨테이너 이용률을 향상시킬 것이다. 또한, 공 컨테이너 수송을 위한 컨테이너 선박 항해도 줄일 수 있을 것이며, 최근 큰 이슈로 부각되고 있는 선박의 탄소 배출량 감축에도 일조할 것으로 기대하고 있다. 이러한 장점에도 불구하고, 접이식 공 컨테이너의 초기 상용화가 실패했던 이유는 작업

자가 직접 컨테이너의 연결부분을 떼어내는 작업을 함으로써 작업의 위험성, 분리된 부분의 도난 위험성, 작업시간 증가 및 일반 공 컨테이너 대비 높은 컨테이너 생산 비용 등의 문제점이 있었기 때문이다. 다만, 현재는 컨테이너 제조사들의 끊임없는 기술 개발로 인해 위의 문제들을 보완하여 상용화가 가능할 수 있도록 개발을 마친 상태이다. 예를 들어, 로테르담의 Cargoshell 사가 제안한 접이식 컨테이너는 대량 생산을 위한 표준인증을 추진하고 있으며, 이는 섬유강화 플라스틱으로 만들어져 기존 철재 컨테이너에 비해 25% 경량화 되었으며, 기계로 하나의 접이식 컨테이너를 접는데 30초 밖에 걸리지 않는다고 한다.

3. 공 컨테이너 임대사이트 설계

3.1 국내·외 임대 사이트와 개발의 필요성

해외에서 운영 중인 컨테이너 임대 사이트 대부분은 컨테이너 매매, 컨테이너 임대(장기 또는 단기), 컨테이너 개조 등의 서비스들을 제공하고 있다. 그리고 컨테이너 임대 및 판매, 철도(레일) 임대 및 판매, 컨테이너 장비 및 각종 장비 판매 등의 서비스 등도 제공하고 있다. 이들 회사들이 운영하는 사이트들은 이러한 절차들을 순수하게 온라인상에서 해결이 가능하도록 서비스들을 제공하고 있다.

다음으로는 국내에서 운영하고 있는 컨테이너 임대 사이트와 그 사이트의 제공 기능 등에서 제공하는 서비스로는 일반 및 특수 컨테이너 제작 및 판매, 컨테이너 임대(장기 또는 단기), 중고 컨테이너 매매, 스마트 컨테이너 서비스(위치 서비스), 컨테이너 장치 판매 서비스 등이 있다. 그러나 대부분의 회사들이 운영하는 사이트는 이러한 서비스들을 순수하게 온라인으로 서비스들을 제공하지 못하고 있으며, 전화상이나 메일로 오프라인 상에서 서비스들을 제공받기 위하여 제공되고 있는 서비스들이다.

따라서 해외 사이트들과 국내 사이트들의 차별되는 특징으로는 해외 사이트들은 컨테이너 임대 서비스 절차를 모두 온라인상에서 진행이 가능하지만 국내 사이트의 경우에는 그렇지 못하고 있는 것을 볼 수 있다.

본 논문에서 개발하고자 하는 사이트의 주요기능에 대해 기술하였으며, 그 기능으로는 크게 다음과

같다. 먼저 간단한 사이트 소개 및 제공 서비스 소개, 그 이후로 각종 임대 선사의 보유 컨테이너 관련 정보 제공, 그리고 컨테이너 임대 서비스를 제공 받기 위한 회원가입 및 로그인 기능, 그에 따른 임대 컨테이너의 정보 제공 서비스, 온라인으로 컨테이너 주문 기능, 해당 기능을 이용하기 위한 주문 조건 제공(컨테이너 종류, 수량, 비용, 대여 조건, 대여일, 반납일, 담당자 정보 등을 제공), 그리고 해당 회원의 현재 주문 현황 제공(주문처리 상태, 주문 정보 등)을 제공하고자 한다.

3.2 개발 환경 구축 및 운영 환경 구축

본 논문에서는 실제로 임대 사이트를 구동하기 위한 개발 프로세스는 다음과 같다.

본격적으로 개발에 들어가기에 앞서 개발환경과 운영환경에 대한 계획이 필요하다. 우선 개발환경은 OSX 운영체제와 window 7 운영체제 위에서 구축하고 개발 및 테스트를 진행하였다. 자바 가상머신(Java Virtual Machine) 상에서 동작하는 자바 웹 프로젝트(Web-Project)이기에 어떤 OS위에서도 동작하기 때문에 개발환경으로써 문제가 없다. 대표적인 자바 개발도구인 이클립스(Eclipse)를 이용하여 개발하였으며, 웹의 디자인 등을 담당하는 프론트엔드(Front-End) 라이브러리(Library)로는 bootstrap을 사용하였고, 서버 처리를 담당하는 백엔드(Back-End)의 프레임워크(Framework)로는 자바 Spring 프레임워크를 사용하였으며, 실제 항만의 컨테이너 수를 받아오기 위한 크롤링(Crawling) 라이브러리로는 jsoup를 사용하였다, 그리고 데이터베이스 처리를 원활히 하기 위해서는 MyBatis 라이브러리를 사용하였다. 웹을 동작시킬 웹 서버 프로그램으로는 톰캣(Tomcat) 서버를 사용하였다. 프론트엔드 기술로는 HTML5, CSS3, JavaScript(JQuery)을 사용하였으며 백엔드 기술로는 Java를 사용하였다. 데이터를 저장할 저장소인 데이터베이스로는 MySQL를 사용하여 QUERY를 통해 웹 서버와 통신하며 데이터를 저장하고 꺼내오는데 사용하였다.

운영 환경은 PC환경에서 포트포워딩(Port Forwarding)을 하여 직접 웹 서버로 가동시키고 24시간 운영하기엔 트래픽에 의한 부하의 위험이 따르므로 24시간 동작시키면서도 부하를 줄일 웹 서버로써 아마존 퍼블릭 클라우드(AWS)를 사용하였다. 퍼블릭

클라우드는 원격으로 해당 서버를 임대할 수 있는 서비스이며 일정용량 이하는 무료로 제공 받을 수 있으며 포트포워딩과 IP, 도메인 서비스도 무료로 제공한다. 운영서버의 운영체제로 Ubuntu 리눅스 서버를 사용하였으며 putty프로그램을 이용하여 원격으로 할당받은 서버를 제어하였다.

추가적인 작업으로 임대 서버를 개발환경과 같이 만들어 줄 수 있게 apt-get 패키지 관리자(package-manager)를 이용하여 MySQL과 Tomcat 그리고 개발환경의 이클립스 프로그램에서 개발을 완료하고, 그렇게 하여 war 프로젝트 파일이 나오게 되고, 이렇게 나온 war파일을 해석할 수 있게 jdk (Java-development-kit)을 설치하였다. 그리고 개발환경에서 데이터베이스 백업파일과 war프로젝트를 임대서버로 옮기기 위해 파일질라 클라이언트(FileZilla Client) 프로그램을 사용하여 각각 임대서버의 Tomcat서버

와 MySQL에 복원시켰다. 이렇게 복원 시킨 상태에서 Tomcat 웹 서버를 가동시키게 되면 외부의 사용자들이 아마존이 제공하는 도메인으로 접속했을 때 웹 사이트로 접속하고 서비스를 제공받을 수 있도록 구현하였다.

3.3 데이터베이스 구조 설계

데이터베이스의 구조는 Fig. 1과 같이 구성되며, 대략 모든 회원의 개인정보를 저장하는 회원정보 테이블과 회원이 주문한 주문내역에 대한 정보를 저장하는 주문정보테이블, 날짜별로 컨테이너 보유수에 대한 정보를 저장하는 컨테이너 정보 테이블로 구성되어있다. 각 테이블의 구성요소로는 회원정보 테이블에는 고객이 입력한 문자열 타입 varchar type)의 이름, 아이디, 비밀번호, 전화번호를 저장한다. 주문 정보 테이블에는 주문을 신청한 사용자의 아이디, 컨

Table 1. Development Environment

Operation System	Windows 7, OS X
CODE	Java, JavaScript
Development Environment	Eclipse
Framework/Library	
- Application Framework	Spring
- FrontEnd Framework	bootstrap
- Persistence Framework	MyBatis
- Web Crawling Library	jsoup
Database system	MySQL
HTTP Web server	Apache Tomcat
Virtual server	Amazon Web Service (AWS), Amazon Elastic Compute Cloud (EC2)

Information of Member		Profile	
Name	field	type	Remark
ID	id	varchar(25)	Pk
Name	name	varchar(50)	
password	pw	varchar(25)	
TEL number	tel	varchar(25)	

Information of Container		Container	
Number of Container	Field	Type	Remark
Number of Container	con_id	varchar(12)	PK
Type of Container	con_type	varchar(5)	Size & Type
Status of Container	con_stat	varchar(25)	default 'Ready'
Retention period	day	date	

Status of order		Order	
Name	Field	Type	Remark
Order ID	id	varchar(25)	
Order number	order_id	int	PK, increase in order
Order type	type	varchar(4)	
Amount	quantity	int	default 0
Rental date	s_date	date	
Return date	l_date	date	
Receiver address	r_address	varchar(255)	
Requests of ETC	etc	text	
Contactor's name	rman_name	varchar(25)	
Contactor's TEL	rman_tel	varchar(25)	
Contactor's company	rman_company	varchar(50)	
Contactor's E-mail	rman_mail	varchar(50)	
Status of order	order_stat	varchar(25)	default 'Receive'

Fig. 1. Design about Structure of Database.

테이너 종류, 수량, 대여 날짜, 반납 날짜, 받을 주소, 기타 요청사항, 담당자 이름, 담당자 전화번호, 담당자 회사, 담당자 이메일과, 주문 상태에 대한 정보를 관리한다. 수량과 주문번호는 숫자타입(Integer Type), 빌릴 날짜와 반납 날짜는 날짜 타입(Date Type)으로 저장되며, 아이디와 컨테이너 종류, 받을 주소, 기타 요청사항, 담당자 이름, 담당자 전화번호, 담당자 회사명, 담당자 이메일, 주문 상태는 문자열 타입으로 데이터베이스에 저장된다. 컨테이너 테이블은 컨테이너 사이즈 및 종류를 식별하는 인자를 이용하여 20FT 일반컨테이너, 40FT 일반컨테이너, 20FT 접이식컨테이너, 40FT 접이식컨테이너, 냉동 컨테이너, 탱크 컨테이너로 종류별로 날짜별 회사의 보유수를 관리하는데 사용되며 사용자가 컨테이너 임대를 신청하면 신청한 수만큼 테이블의 보유수에서 감소된다.

다음으로 해당 테이블들의 MySQL 쿼리문과 구조는 Fig. 2와 같다.

다음으로 자바 데이터 전송 객체인 DTO(Data Transfer-Object) 객체들은 아래의 구조와 같으며 Fig. 3과 같이 구성된다.

Client Table	Container Table	Order Table
<pre> Create table profile(id varchar(25), Name varchar(50), pw varchar(25), Tel varchar(25), primary key(id); </pre>	<pre> create table container(con_id varchar(12), con_type varchar(5), con_stat varchar(25), default 'ready ' Day date primary key(con_id); </pre>	<pre> create table order(id varchar(25), order_id int auto_increment, con_type varchar(4), quantity int, s_date date, i_date date, r_address varchar(255), etc text, r_man_name varchar(25), r_man_tel varchar(25), r_man_company varchar(50), r_man_mail varchar(50), order_stat varchar(25) default 'Receipt ' primary key(order_id)); </pre>

Fig. 2. Structure of Database.

Client DTO	Container DTO	Order DTO
<pre> ProfileDto String id String name String Pw String Tel </pre>	<pre> ContainerDto String Con_id String Con_type String Con_stat </pre>	<pre> OrderDto String id int Order_id String Con_type int Quantity Date S_date Date L_date String r_address String Etc String rman_name String rman_tel String rman_company String rman_mail String rman_stat </pre>

Fig. 3. Object of Java DTO.

여기서 정의된 객체는 3개로 ProfileDto, Order Dto, ContainerDto가 있으며, 각각 객체별로 살펴보면 다음과 같다. ProfileDto객체는 회원의 가입 정보를 입력받거나 가입한 정보를 수정하거나 중복된 아이디를 검증하기 위하는데 사용된다. OrderDto객체는 회원의 주문정보를 받거나 주문취소, 남은 수량을 파악하는데 필요한 주문검증 등에 사용된다. ContainerDto는 현재 회사의 컨테이너 수량과 날짜별 컨테이너 수량 확인에 사용된다.

4. 컨테이너 임대 사이트 개발

총 10개 이상의 페이지로 구성되며 기본 index 페이지, 회원가입페이지, 로그인페이지, 임대주문페이지, 주문확인페이지, 관리자 페이지 등으로 구성된다. Fig. 4는 페이지의 구성을 보여주기 위해 페이지 파일구조와 url을 나타낸 것이다.

4.1 고객 프로세스

고객은 홈페이지를 방문하여 회사의 소개와 회사에서 제공하는 공 컨테이너 임대 서비스의 종류(드라이 컨테이너, 냉동 컨테이너, 탱크 컨테이너 등)를 확인 할 수 있다. 임대 서비스를 이용하기 위해서는 회원가입을 하여야한다. 이를 위해 회원가입 페이지를 방문하여 개인정보를 입력하여 아이디를 만들고 로그인 페이지를 방문하여 생성한 아이디를 이용해 로그인하면 임대 주문이 가능하다. 임대주문을 하려면 원하는 컨테이너의 수, 임대일 등을 기입하여야하여 임대 주문 접수를 할 수 있다. 주문 내역은 mylist 페이지에서 확인 할 수 있으며, 임대 주문 전에 mylist페이지에서 항만의 컨테이너 현황을 확인하여 임대 주문 결정에 도움을 받을 수도 있다. 그리고 접수를 하더라도 임대 주문을 완료한 것이 아니며 관리자가 접수를 허가하여 “접수완료” 상태가 되어야 정상적인 임대 서비스가 제공되며, 만약 접수를 취소하고

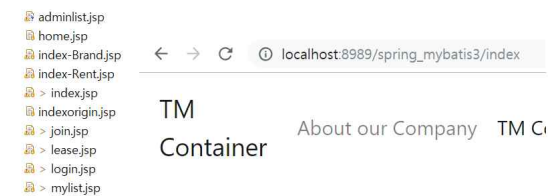


Fig. 4. Page on Web site of Architecture.

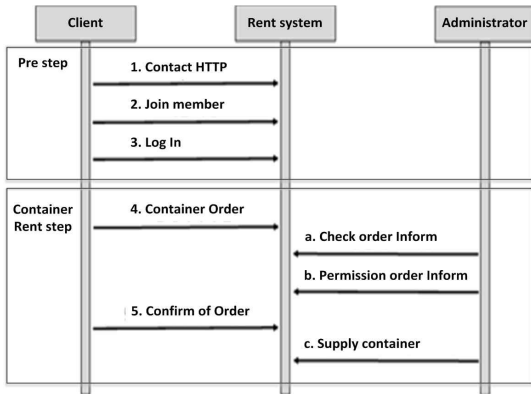


Fig. 5. Processor about Client & administrator.

싶을 시에는 임대하고자 하는 시작 날짜가 지났거나 “접수완료“ 상태가 아니라면 언제든지 접수 취소가 가능하다. Fig. 5는 고객과 관리자의 관리 프로세스를 나타낸 것이다.

4.2 관리자 프로세스

Fig. 5는 관리자의 주문허가 과정이다. 관리자는 임대 주문이나 관리자 자신의 임대 주문 목록을 확인할 수 없다. 대신에 관리자는 관리자만의 주된 업무로서 홈페이지에 가입한 모든 회원의 정보와 모든 고객의 주문 내역을 확인할 수 있으며 관리자는 고객이 임대 접수를 하면 고객이 신청한 컨테이너 임대 주문사항을 확인하고 접수 허가하여 주문 상태를 “접수완료” 상태로 만들어 정상적인 임대 서비스를

받게 할 수 있다. 또한 고객이 아이디 또는 비밀번호를 잊어버릴 시에 고객 정보를 확인하여 아이디와 비밀번호를 찾아줄 수 있으며 고객의 주문이 “접수완료” 상태일 지라도 고객과의 협의를 마친 후 관리자 페이지에서 해당 주문을 강제로 취소할 수 있다.

4.3 컨테이너 임대 사이트 각 기능

컨테이너 임대사이트는 회사에 대한 간단한 소개와 회사가 어떤 종류의 컨테이너를 제공해 주는지 알려주며, 주된 목적은 온라인으로 컨테이너 임대를 할 수 있는 서비스를 제공한다. 이를 위해 회원가입 기능과 로그인 기능을 제공하고 고객이 주문내역을 확인하고 취소하는 기능 또한 제공하며 임대하기 전에 항만의 컨테이너 현황을 확인할 수 있게 포로 띄워주는 역할도 한다. 마지막으로 고객이 주문 접수를 하여 관리자가 그 내역을 판단하고 허가하여 실제로 임대일이 다가오면 해당 컨테이너를 대여해줄 수 있게 해주는 기능을 가지고 있다.

4.3.1 회사 소개

임대 서비스를 제공하는 회사에 대한 기본사항을 확인할 수 있다. Fig. 6은 회사의 지속가능성에 대한 설명을 하고 있으며 TM Container 단어를 마우스로 클릭하여 index-brand페이지로 넘어가면 Fig. 7이 화면에 표시되며 다른 컨테이너 임대 회사에 비해 자사의 장점을 확인할 수 있다. 마찬가지로 Rent Container 단어를 클릭하여 Fig. 8은 컨테이너 임대

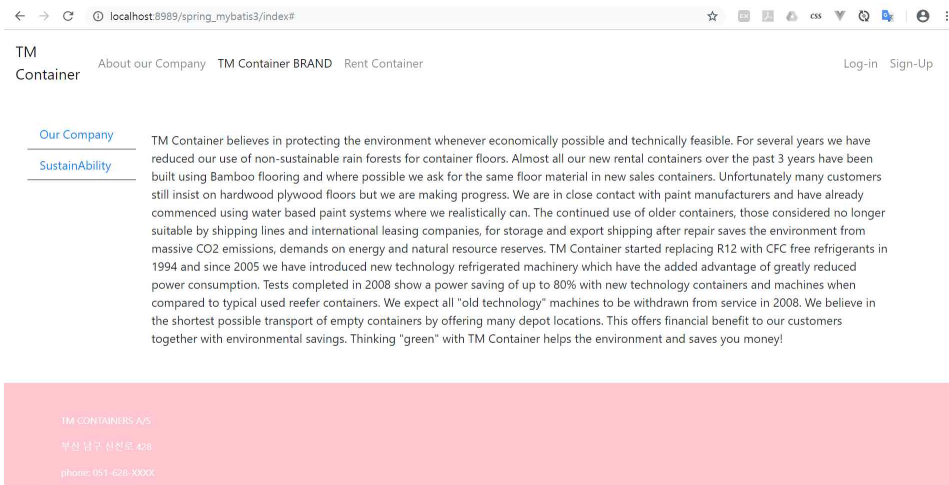


Fig. 6. Page of index.

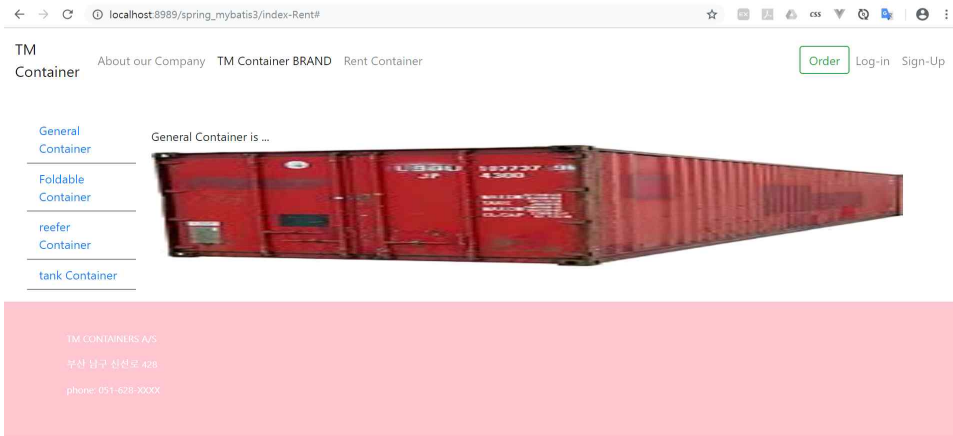


Fig. 7. Page of index-Brand.

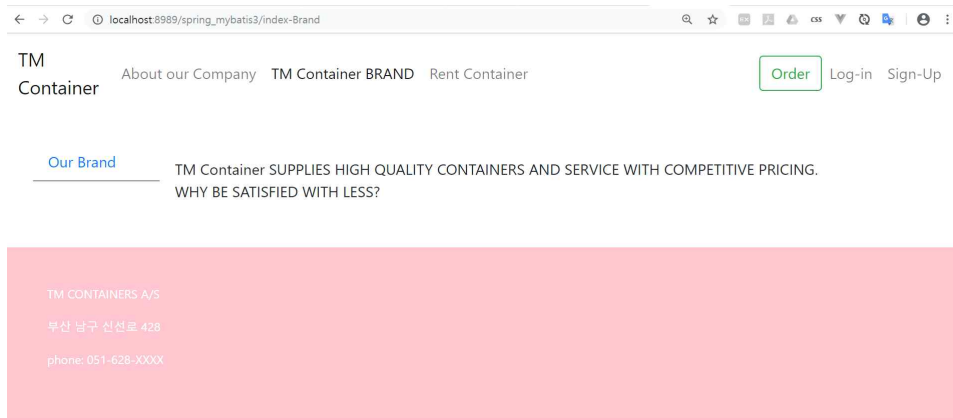


Fig. 8. Page of index-Rent.

서비스를 제공하는 컨테이너의 종류에 대해 확인할 수 있다. 또한 각 페이지 하단에는 전화번호와 주소 정보가 표시되며, 이를 통해 고객이 문제가 생기면 회사로 문의 할 수 있다. 이처럼 고객이 컨테이너 주문 전에 확인하기 위한 기본사항에 대한 설명을 인터넷으로 제공한다.

4.3.2 컨테이너 임대기능

로그인 상태에서 order버튼을 클릭하면 Fig. 9와 같이 임대페이지에 진입하게 되며 수량, 대여일, 반납일, 받을 주소, 담당자이름, 담당자전화번호, 담당자회사, 담당자이메일, 기타사항을 입력하여 submit 버튼을 누르면 주문이 접수된다. 그러나 해당 날짜에 회사에서 해당타입의 컨테이너가 보유수가 부족할 경우 Fig. 10과 같이 에러메시지를 띄워주며 다시 입

력받게 된다.

4.3.3 주문확인, 취소 및 항만의 컨테이너 현황 확인

Fig. 11와 같은 로그인 상태에서 mylist 버튼을 클릭하면 Fig. 11과 같이 나의 주문정보를 확인할 수 있는 페이지로 이동된다. 이곳에서 고객 자신이 여태 주문한 모든 사항을 표로 확인할 수 있다. 만약 고객이 주문을 했지만 일정이 변경되어 해당 주문을 취소하고 싶다면 임대일이 지났거나 주문 상태가 접수완료 된 상태가 아니라면, 표 내부의 취소버튼을 이용하여 해당 주문을 취소 할 수 있다. 또한 자신이 가입한 아이디의 비밀번호 또는 전화번호 변경하고 싶으면 Fig. 12의 해당 항목의 입력 값을 원하는 값으로 변경하고 “회원정보 바꾸기“ 버튼을 클릭하여 자신의 회원정보를 변경할 수 있다.

The form contains the following fields: ID (dnjsumf), Type (foldable20ft), amount, Rental date (YY-MM-DD), Return date (YY-MM-DD), Receive address, Contactor's name, Contactor's TEL number, Contactor's company, Contactor's E-mail, ETC, and a SUBMIT button.

Fig. 9. Rental page of container.

There is not enough containers.
The minimum number of containers for a scheduled period is 3.

Fig. 10. Insufficient Container Error Message.

4.3.4 관리자 권한으로 전체회원 파악 및 주문정보 확인

로그인 페이지에서 특정 아이디와 특정 비밀번호를 입력하면 관리자만의 페이지로 접근이 가능해지

며, 일반 고객과는 달리 Fig. 13과 같이 ‘adminlist버튼’이 표시되며 관리자 페이지로 접근할 수 있다. 이곳에서 관리자는 Fig. 14와 같이 모든 회원의 정보를 확인할 수 있으며, Fig. 15와 같이 고객들의 모든 주문정보를 확인할 수도 있다. 또한 이곳에서 관리자가 주문확인 버튼을 클릭하여 고객의 주문 상태를 접수에서 접수완료 상태로 변경할 수 있으며, 이로써 실질적인 임대 서비스를 진행할 수 있다. 또한 고객의 문의가 들어올 시에 고객이 취소할 수 없는 주문에 대해 강제로 취소할 수 있는 권한도 있다.

4.3.5 항만의 컨테이너 현황 확인

Fig. 16은 mylist페이지에서 하단에 부산 신항만의 각 선사별 컨테이너 보유 개수를 실시간으로 출력하여 보여주는 것이다. 고객이 현재 항만의 컨테이너 부족사항을 확인하여 컨테이너 부족사항을 예측하여 임대할건지 아니면 컨테이너가 과다하게 많이 남은 선사의 컨테이너를 임대할 건지 선택하는데 도움을 준다. 만약 대다수의 선사들이 컨테이너가 부족한데 특정 1,2개 선사의 컨테이너 수가 과다하게 남은다면 항만 밖의 컨테이너를 무리하게 임대해 올 필요 없이 해당 선사의 안전재고 대역안의 남은 컨테이너를 임대하면 된다. 이렇게 되면 육상의 컨테이너의 이동이 줄어 교통체증을 완화할 뿐만 아니라 궁극적으로 해당 항만의 전체 컨테이너(총 선사들의 컨테이너 수)가 충분한데도 불구하고 임대를 하거나 수출을 위해 빈 컨테이너를 다른 곳에서 수입해오는 상황을 줄이게 되며, 전체 컨테이너 리포지셔닝 비용이 줄어들게 된다. 이로써 항만 전체의 컨테이너를

Order cancellation is possible one day before the scheduled date of rental.

Order Number	Type of container	Amount	Rental date	Return date	Address	Order cancellation	Order status
47	foldable40ft	3	2019-08-10	2019-10-10	Busan	Irrevocable	Reception completed
48	reefer	3	2019-08-11	2019-10-11	Seoul	Cancel	Receipt
50	foldable20ft	1	2019-08-12	2019-10-12	Busan	Irrevocable	Receipt

Fig. 11. History of member's order.

The form shows fields for ID (dnjsumf), password (masked with ****), and Tel (010-xxxx-xxxx). There is a 'Show password' button and a 'Change member Information' button.

Fig. 12. Change of member's information.

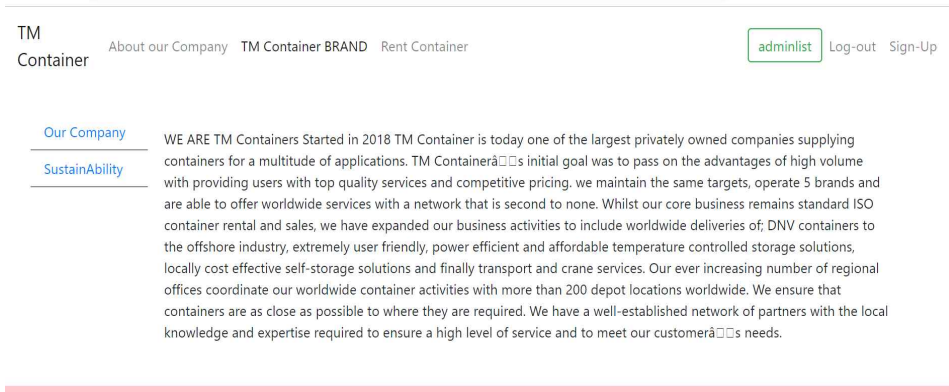


Fig. 13. index page of admin.

Member's Information

Name	ID	Password	TEL number
admin	admin	1234	010-XXXX-XXXX
Bae	sdhflsfhjh	1234	010-XXXX-XXXX
John	Dshlfsdlhfs	1234	010-XXXX-XXXX
Kim	dgfgdg	1234	010-XXXX-XXXX
Lee	trtegebb	1224	010-XXXX-XXXX

Fig. 14. Member's information.

Member's Order list

ID	No.	Type of container	Amount	Rental date	Return date	Address	Order cancellation	Order status	Receipt
dnjsxkf	44	normal40ft	1	2019-08-08	2019-10-08	Daegu, KP univ.	Cancel	Reception completed	Confirm
rudkogus	45	reefer	2	2019-08-09	2019-10-09	Busan	Cancel	Receipt	Confirm
rudkogusfsdf	46	tank	3	2019-08-10	2019-10-10	Busan	Cancel	Receipt	Confirm
dnjsxmf	47	foldable40ft	4	2019-08-11	2019-10-11	Busan, TM univ.	Cancel	Reception completed	Confirm
dnjsxmf	48	reefer	5	2019-08-12	2019-10-12	Busan, TM univ	Cancel	Receipt	Confirm
dnjsxmf	50	foldable20ft	6	2019-08-13	2019-10-13	Busan	Cancel	Receipt	Confirm

Fig. 15. Order Information.

수출현황에 맞게 완만하게 보유하는 것이다. 이는 개인 선사만의 이익만이 아니라 항만 공동체의 공익이라 할 수 있다.

4.4 그 외 동작하는데 도와주는 필요 코드

Fig. 17은 로그인 및 로그아웃을 위해 필요한 자바 컨트롤러 코드이며 사용자가 로그아웃 및 로그인 버

튼을 눌렀을 때 작동하며 로그인시 사용자가 입력한 id와 password를 데이터베이스의 회원정보 테이블에 있는 id와 password와 비교하여 일치하는 게 있으면 로그인되었음을 인증해주는 세션(Session)을 발급하고 일치하는 것이 없으면 다시 입력받게 한다. 그리고 로그아웃 버튼을 누르면 발급받았던 세션이 만료된다.

Size	HLC	ONE	YML	MAE	MSC	HMM	ZIM	CMA	COS	OOL	ESL	RCL	SML	KMD	TSL	PIL	WHL	NSL	KFC
20DC	17	34	60	3	74	0	104	1	0	0	68	58	167	0	0	0	3	7	13
40DC	43	6	236	0	21	0	71	0	0	0	0	0	35	0	0	0	8	0	0
40HC	43	26	18	6	61	19	0	7	0	0	46	11	55	0	0	0	7	1	65

Return index

Fig. 16. Container Retention Status at Port.

```

@RequestMapping("/logout")
public String logout(HttpServletRequest request) {
    HttpSession session = request.getSession();
    session.invalidate();

    return "/index";
}

@RequestMapping("/loginOK")
public String loginOK(Model model,HttpServletRequest request) {
    String id=request.getParameter("id");
    String pw=request.getParameter("pw");

    IDao dao = sqlSession.getMapper(IDao.class);
    ArrayList<ProfileDto> check = dao.loginDao(id,pw);

    if(check.isEmpty()) {
        model.addAttribute("error", "아이디 혹은 비밀번호가 일치하지 않습니다.");
        return "login";
    }

    HttpSession session = request.getSession();
    session.setAttribute("id", id);

    return "index";
}
    
```

Fig. 17. Controller code of Log in and out.

Fig. 18은 신항만 사이트로부터 실제 공 컨테이너 수에 대한 데이터를 받아와 배열리스트의 형태로 지

```

@RequestMapping("/mylist")
public String mylist(HttpServletRequest request,Model model) throws UnsupportedEncodingException {
    HttpSession session = request.getSession();
    String id=(String)session.getAttribute("id");
    IDao dao = sqlSession.getMapper(IDao.class);
    String title=null;
    String url = "https://svc.pncport.com/info/Main.do";
    Document doc = null;
    ArrayList<String> thead = new ArrayList<String>();
    ArrayList<String> DC20 = new ArrayList<String>();
    ArrayList<String> DC40 = new ArrayList<String>();
    ArrayList<String> HC40 = new ArrayList<String>();
    try {
        doc = Jsoup.connect(url).get();
        Elements theadE = doc.select("#mcontainer-wrap>div.econtain-sec>div>table>thead>tr>th");
        Elements tbody1 = doc.select("#mcontainer-wrap>div.econtain-sec>div>table>tbody>tr:nth-child(1)>td");
        Elements tbody2 = doc.select("#mcontainer-wrap>div.econtain-sec>div>table>tbody>tr:nth-child(2)>td");
        Elements tbody3 = doc.select("#mcontainer-wrap>div.econtain-sec>div>table>tbody>tr:nth-child(3)>td");
        for(Element th : theadE){thead.add(th.text());}
        for(Element td:tbody1) {DC20.add(td.text());}
        for(Element td:tbody2) {DC40.add(td.text());}
        for(Element td:tbody3) {
            HC40.add(td.text());
        }
    }
    catch(Exception e) {
        e.printStackTrace();
    }
    model.addAttribute("thead",thead);
    model.addAttribute("DC20", DC20);
    model.addAttribute("DC40", DC40);
    model.addAttribute("HC40", HC40);
    model.addAttribute("orderid", dao.myDao(id));
    model.addAttribute("data","mylist");
    return "/index";
}
    
```

Fig. 18. Number of possession of Empty container crawling and ordering information code at New port.

장하고 사이트에서 볼 수 있도록 도와주는 코드이다. 또한 사용자의 주문정보를 보여주는데 도움을 준다.

Fig. 19는 사용자가 입력한 주문정보를 토대로 컨테이너를 임대받을 수 있는지 확인해주고 가능하다면 주문정보를 데이터베이스의 주문정보 테이블에 저장하고 컨테이너 테이블의 컨테이너 보유수를 업데이트 해주도록 하는 코드이다.

Fig. 20과 Fig. 21은 데이터베이스에 쿼리를 전달 하는데 도움을 주며 회원가입, 로그인, 수량확인, 주문서작성, 컨테이너 수 업데이트, 주문정보 확인, 수정페이지 확인, 회원정보 수정, 주문취소, 모든 회원 정보 확인, 모든 주문 정보 확인, 고객주문 승인들의 기능에 도움을 준다.

5. 임대 사이트 파급 효과

본 논문의 공 컨테이너 임대 사이트의 웹 사이트 개발을 통하여 얻을 수 있는 효과로는 공 컨테이너를

```

@RequestMapping("/leaseOK")
public String leaseOK(HttpServletRequest request, Model model) {
    String con_type = request.getParameter("con_type");
    int quantity = Integer.parseInt(request.getParameter("quantity"));

    String s_date = request.getParameter("s_date");
    String l_date = request.getParameter("l_date");

    //System.out.println(request.getParameter("con_type"));

    IDao dao = sqlSession.getMapper(IDao.class);
    int num = dao.ordercheckDao(con_type, s_date, l_date);

    model.addAttribute("number", num + "개입니다.");

    if(quantity > num) {
        model.addAttribute("error", "컨테이너수가 부족합니다. \n\n 예약된 기간의 저최 최소 컨테이너수는 " + num + "개입니다.");
        return "lease";
    }

    dao.orderDao(request.getParameter("id"), con_type, quantity, s_date, l_date,
        request.getParameter("r_address"), request.getParameter("etc"),
        request.getParameter("rman_name"), request.getParameter("rman_tel"),
        request.getParameter("rman_company"), request.getParameter("rman_mail"));

    dao.conupdateDao(con_type, s_date, l_date); //컨테이너 임대료

    return "index";
}

```

Fig. 19. Controller code of rental container.

구하고자 하는 고객의 입장에서는 해당 웹 사이트를 통하여 보다 빠르게 공 컨테이너 정보를 구할 수 있으며, 공 컨테이너의 종류와 주문수량 등의 정보를 얻을 수 있다. 이렇게 필요한 공 컨테이너에 관한 정보를 제공받아서 해당 공 컨테이너를 구하는 입장에서 보다 정확한 정보를 제공 받고, 공 컨테이너 임대 에 관한 계획을 보다 정확하게 수립할 수 있다. 또한 필요한 수량의 공 컨테이너를 온라인으로 주문하여 해당 주문을 되도록 즉각적으로 결제하고 바로 어디 에서 어떻게 받아서 임대를 할 수 있다면, 여러 가지 주문을 통하여 어렵게 임대하던 방식이 아닌 보다 편한 방식으로 주문이 가능하게 되어 시간절약의 효

과를 기대할 수 있다. 그리고 주문한 정보들을 확인 이 가능하도록 하여 어떤 공 컨테이너를 주문하였고, 어떤 공 컨테이너를 제공 받는지 로그인만 하게 된다 면 사이트 내에서 모두 확인이 가능하여 여러 가지 서류들을 보고 확인하는 절차도 생략이 가능하다. 또 한 기존의 계약을 통하여 임대를 할 때, 임대하는 날 짜 이전이라면 중간에 여러 가지 상황의 변화로 인하여 취소하거나 변경을 할 때 많은 어려움이 있다. 하지만 본 논문의 임대 사이트 개발을 통해 주문을 취소하고 다시 재주문을 가능하게 되고 주문 변경을 보다 쉽게 하여 여러 가지 상황 변화에도 대처를 할 수 있는 유동성을 가질 수 있게 된다.

또한, 공 컨테이너를 제공하는 입장에서는 해당 공 컨테이너의 정보를 제공함으로써 해당 정보를 통해 공 컨테이너를 구매할 수 있는 고객층들을 보다 넓게 확보를 할 수 있다. 또한 해당 정보의 공 컨테이너와 관련한 업무 부분을 본 논문의 임대 사이트를 통해서 공 컨테이너 업무와 관련된 비용을 줄일 수 있는 효과를 얻을 수 있다. 그리하여 공 컨테이너의 관리를 보다 효율적이고 간편하게 할 수 있고 여러 가지 상황들로 인하여 발생할 수 있는 수출입 불균형 으로 발생하는 공 컨테이너 관련된 비용들을 절약할 수 있는 효과를 기대할 수 있다.

6. 결 론

본 논문은 공 컨테이너의 불균형 및 컨테이너 리

```

<insert id="joinDao">
INSERT INTO profile (name,id,pw,tel) VALUES ({param1},#{param2},#{param3},#{param4})
</insert>
<!-- 회원가입 -->

<select id="loginDao" resultType="com.javalec.spring_mybatis.dto.ProfileDto">
SELECT id,pw FROM profile where id=#{param1} AND pw=#{param2}
</select>
<!-- 로그인 -->
<select id="ordercheckDao" resultType="int">
select min({param1}) from container where day between #{param2} and #{param3}
</select>
<!-- 수량확인 -->
<insert id="orderDao">
INSERT INTO `order` (id,con_type,quantity,s_date,l_date,r_address,etc,rman_name,rman_tel,rman_company,rman_mail,order_stat)
values ({param1},#{param2},#{param3},#{param4},#{param5},#{param6},#{param7},#{param8},#{param9},
#{param10},#{param11},default)
</insert>

<!-- 주문서완료 -->
<update id="conupdateDao">
update container set ${param1}=${param1}-(select quantity from `order` order by order_id desc limit 1) where day between #{param2} and #{param3}
</update>
<!-- 컨테이너 임대료 -->

<select id="myDao" resultType="com.javalec.spring_mybatis.dto.OrderDto">
SELECT * FROM `order` where id=#{param1}
</select>
<!-- 마이리스트 주문정보 -->
<select id="profileDao" resultType="com.javalec.spring_mybatis.dto.ProfileDto">
SELECT * FROM profile where id=#{param1}
</select>
<!-- 수정페이지 보기 -->

```

Fig. 20. Query code No.1 of MyBatis.

```

<update id="modifyProfile">
update profile set name=#{param1},pw=#{param3},tel=#{param4} where id=#{param2}
</update>

<!-- 회원정보 수정 -->

<delete id="cancelOrder">
delete from `order` where order_id=#{param1};
</delete>

<update id="cancelContainer">
update container set ${param1}=${param1}+#{param2} where day between #{param3} and #{param4}
</update>
<!-- 주문취소 및 컨테이너 정보수정 자카 나뉨 -->
<delete id="testCancel">
delete from `order` where order_id=#{param1};
update container set ${param2}=${param2}+#{param3} where day between #{param4} and #{param5}
</delete>
<!-- 주문취소 및 컨테이너정보수정 -->
<select id="allProfile" resultType="com.javalec.spring_mybatis.dto.ProfileDto">
select * from profile
</select>
<!-- 모든회원정보보기 -->
<select id="allOrder" resultType="com.javalec.spring_mybatis.dto.OrderDto">
select * from `order`
</select>
<!-- 모든주문정보보기 -->
<update id="checkDao">
update `order` set order_stat='접수완료' where order_id=#{param1}
</update>
<!-- 고객의 주문승인 -->

```

Fig. 21. Query code No.2 of MyBatis.

포지셔닝 과정에서 여러 가지 문제를 야기한다는 사실을 확인하였다. 공 컨테이너의 사용 주체인 기업의 경우 공 컨테이너의 불균형으로 인해 공 컨테이너가 남을 경우, 컨테이너 수리비용 등의 유지비로 인한 비용이 발생한다. 그리고 부족할 경우엔 화물 수출을 위한 공 컨테이너 임대로 인해 높은 일일 임대료를 물어야하며, 타 지역 또는 타국에 있는 공 컨테이너를 들여오는 과정에서 비용이 따라오게 된다. 이 경우 해운회사의 재무상에만 타격을 입힐 뿐만 아니라 육상으로 지역 간 혹은 지역 내에서 컨테이너 재배치를 할 경우 해당 지역의 교통 혼잡을 유발하고 CO₂ 배출로 인한 환경오염까지 유발하게 된다. 더욱 심각한 것은 공 컨테이너 리포지셔닝의 규모가 해가 지날수록 컨테이너 물동량이 많아지면서 점점 늘어나고 있다는 사실을 통계를 통해 확인하였기 때문에 이러한 혼란의 규모는 더욱 더 증가될 것이기에 기업 개인뿐만이 아니라 항만공동체 모두가 경각심을 가지고 심각한 주의가 필요하며, 이러한 공 컨테이너의 불균형을 초래하는 원인을 해결해야 할 의무가 있다. 따라서 본 논문의 공 컨테이너 임대 사이트로 개발로 다음과 같은 효과를 얻을 수 있다.

첫째, 공 컨테이너를 임대하고자 하는 고객들에게 공 컨테이너 관련한 정보를 제공하여 고객에 있어서 필요한 공 컨테이너를 보다 정확하게 제공할 수 있다. 공 컨테이너를 임대하고자 하는 고객들은 불확실한 정보들은 최소한으로 하고, 최대한 확실하고 정확한 정보들을 가지고 공 컨테이너를 임대하여 사용하

고 싶어 한다. 해당 사이트를 통해 이러한 과정들을 비교적 빠르게 진행을 하여 계획을 수립하는데 도움을 줄 수 있어 화물 수출에 있어서 비교적 빠르게 진행이 가능하다.

둘째, 필요한 공 컨테이너를 온라인으로 바로 주문이 가능하게 하여 필요할 때 바로바로 받을 수 있으며, 그로 인하여 시간 절약과 비용도 절약이 가능하게 된다. 이를 통하여 업무의 진행에 있어 빠르게 진행이 가능하며 여러 복잡한 과정들의 생략을 통하여 작업의 속도를 또한 빠르게 진행을 할 수 있도록 도와준다.

셋째, 공 컨테이너의 관리적인 측면에서 관리에 관한 여러 가지 비용들에 있어서 절약이 가능하다. 특히 본 논문에서 개발한 사이트를 통해 공 컨테이너를 임대하는 부분에 있어서 여러 가지 정보들을 한꺼번에 관리가 가능하여 여러 가지 문제점을 해결할 수 있으며, 이로 인해 업무의 진행에 있어서 지연을 최소화하여 더 많은 일처리를 함으로서 많은 이윤을 추구하며, 고객의 입장에선 보다 좋은 서비스를 제공 받을 수 있다.

넷째, 이러한 효과들을 통하여 공 컨테이너 불균형에 관한 문제를 해결하여 추가적으로 발생할 수 있는 환경문제를 해결할 수 있다. 또한 여러 가지 상황들의 변화로 인해 빠르게 따라가지 못하여 생기는 배치 문제를 해결하여 불균형을 최소화하고, 중간에 필요 없는 작업들이 간소화되어 중간 과정에서 발생할 수 있는 환경 문제를 해결할 수 있다.

REFERENCE

- [1] S.Y. Jung, *A Study on the Efficient Management of Empty Containers for Liner Shipping Company*, Master's Thesis of Chung-Ang University of Human Resource Development, 2014.
- [2] K.S. Ryu and J.S. Park, "Quantification of Loading Efficiency of Various Type Loads in a 20 FT Container with Post Selecting Process after Applying Conventional Loading Algorithms," *Journal of Korea Multimedia Society*, Vol. 21, No. 4, pp. 513-526, 2018.
- [3] J.S. Song, *A Study on the Empty Container Supply Management for the Short Sea Liner*

- Shipping Company*, Master's Thesis of Chung-Ang University of Human Resource Development, 2008.
- [4] J.P. Rodrigue, *Claude Comtois, and Brian Slack The Geography of Transport Systems*, 4th ed, London, Routledge, 2017.
- [5] J.H. Kwak, *Redistribution and Replenishment Policy of Empty Container on Ocean Transportation Network*, Master's Thesis of Dong-A University of Industrial Engineering, 2000.
- [6] S.C. Kim, *Management of Standard and Foldable Container Operations through the Empty Container Repositioning and the Contracted Lease Pricing*, Master's Thesis of Pusan National University of Industrial Engineering, 2013.
- [7] Significant Reduction in Shipping Costs for Shipping Containers, <http://www.ksg.co>.

[kr/news/main_newsView.jsp?bbsID=news&bbsCategory=KSG&categoryCode=all&backUrl=main_news&pNum=108927](http://www.ksg.co.kr/news/main_newsView.jsp?bbsID=news&bbsCategory=KSG&categoryCode=all&backUrl=main_news&pNum=108927) (accessed May 20, 2019).

- [8] S.H. Kang, *A Study on Foldable Container Selection Factors using a Conjoint Analysis*, Master's Thesis of Inha University of Logistics, 2017.



배 석 태

1996년 동아대학교 기계공학과 대학원 졸업(공학박사)
2013년~현재 해양수산부 설계자문위원회 위원
1995년~동명대학교 항만물류시스템학과 교수

관심분야 : 항만물류자동화 장비, 컨테이너 터미널 운영