

# Docker를 이용한 프로그래밍 실습 시스템 설계

채민수\*, 전은광\*,이화민\*\*

\*순천향대학교 컴퓨터학과

\*\*순천향대학교 컴퓨터소프트웨어공학과

e-mail: \*cmspr0@nate.com , \*imdae11@naver.com , \*\*leehm@sch.ac.kr

## Programming practice system using Docker<sup>1)2)</sup>

Minsu Chae\*, Jeon EunKwang\*,Hwamin Lee\*\*

\*Dept. of Computer Science, Soonchunhyang University

\*\*Dept. of Computer Software Engineering, Soonchunhyang University

### 요 약

최근 소프트웨어 교육이 중요해짐에 따라 학교 SW 교육이 강화되고 있다. 2017년부터는 초등학교, 2020년부터는 중학교에 소프트웨어교육이 필수화 될 예정이다. 그러나 현재 부족한 정보 교사의 수와, 열악한 실습실 환경을 갖는 학교가 많다. 그에 따라 본 논문에서는 학습자에 의해 시스템이 영향을 받지 않고, 다양한 언어를 제공하며, 실습 체점을 지원하는 프로그래밍 실습 시스템을 설계하였다.

### 1. 서론

최근 소프트웨어 교육이 필수화 됨에 따라 학교 SW 교육 강화하고 있다. 2015년부터, 희망학교에 한해 초등학교에서 SW교육을 실시하였고, 2017년부터 정식 교과목으로 채택되었다. 교육부가 발표한 내용에 따르면 2019년부터 초등학교는 17시간, 2018년부터 중학교는 34시간 이상 소프트웨어교육이 필수화가 되었다[1]. 교육부에서는 2018년까지 초등교사 6만명, 중등 정보 교사 전체를 대상으로 연수를 실시하여 소프트웨어 교육 역량을 강화할 예정이다[1]. 또한 교육청와 협력하여 중학교 정보 교사를 2016년까지 50명, 2017년까지 84명을 신규 채용할 계획이며, 2020년까지 신규 및 복수전공연수 등을 통해 500명 이상으로 확보할 예정이다[1]. 또한 실습실 미확보 학교 172곳중 2020년까지 69곳의 학교에 설치할 예정이며, 그 외 학교는 노트북, 태블릿 PC 등을 통하여 교육할 예정이다[1]. 현재 학생 수에 비해 전문 SW 교육 인력이 부족하고, 또한 실습 환경이 부족한 실정이다. 프로그래밍 실습의 경우 학생들이 실습 문제를 풀고, 교사는 주어진 문제에 대해 제대로 했는지 채점 및 지도를 하게 된다. 본 논문에서는 Docker를 이용한 프로그래밍 실습 시스템을 설계하고, 이를 통해 실습 시스템을 개선하고자 한다.

### 2. 관련 연구

#### 2.1 학습자와의 상호작용을 고려한 프로그래밍 학습 지원 시스템 설계 및 구현[2]

학습 지원 시스템은 CM(Computer Managed Instruction), LMS(Learning Management System), LCMS(Learning Content Management)로 나뉘게 된다[3]. 이를 통해 학교에서는 교사-학생간 학습 관리 서비스를 제공한다. 그러나 프로그래밍 학습의 경우 실습이 중요한데, 이에 대한 제공이 미흡하여 해당 논문에서는 새로운 학습 지원 시스템을 설계 및 구현하였다[2]. 그러나 학습자가 고의 혹은 실수로 시스템에 심각한 영향을 끼치는 소스가 들어갈 수 있다. 최악의 경우 프로그래밍 학습 지원 시스템 서버의 다운이 될 수 있다. 그에 대한 대비가 되어 있지 않다. 또한 다양한 언어에 대한 확장성을 고려하지 않았다.

### 3. 프로그래밍 실습 시스템 설계

#### 3.1 고려 사항

##### 3.1.1 시스템 보안

프로그래밍 실습 시스템을 제공함에 있어, 서버가 학습자에 의해 종료되지 않아야 한다. 그렇기 위해선, 학습자가 입력된 소스에 의해 서버가 영향을 받아서는 안된다. 그에 따라 두 가지 방법으로 해결이 가능하다. 첫 번째 방법은, 언어차원에서 보안을 위한 정책을 설정하여 시스템과 관련된 기능을 제한하는 것이다. JAVA 언어의 경우 이에 속하며, 실행시 런타임 예외를 발생시킨다[4]. 그러나 모든 언어가 제공하는 것은 아니다. 또 다른 방법은, 가상화 기술을 이용하여 시스템에 영향을 끼치지 않는 방법이다. 가상 머신의 경우 클라우드 컴퓨팅에서 자원 할당

1) 본 연구는 미래창조과학부 및 정보통신기술진흥센터의 대학ICT연구센터육성 지원사업의 연구결과로 수행되었음 (IITP-2017-2015-0-00403)

2) 이 논문은 2016년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (NRF-2014R1A1A2057878).

및 OS 설치에 대한 시간이 추가적으로 들지만, Docker의 컨테이너의 경우 이미 생성한 이미지를 통해 수 초 내에 실행가능하다[5]. 그에 따라 본 논문은 Docker를 이용하여 학습자에 의해 시스템에 영향을 받지 않도록 하고자 한다.

### 3.1.2 다양한 언어 제공 및 버전 관리

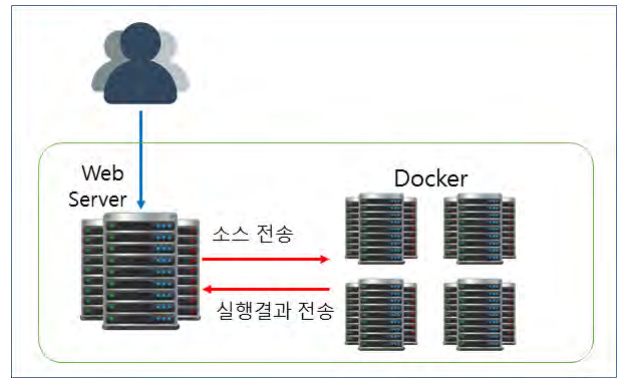
학교 수업에서 C언어 뿐만 아니라 다양한 언어를 가르치고 있다. C언어와 비교했을 시 긍정적인 새로운 동기와, 고차원적인 사고력이 증진이 되고, 학습자가 프로그래밍에 쉽게 접근할 수 있다고 하였다[6]. 그에 따라 실습 시스템에서 다양한 언어를 지원을 위한 것이 필요로 하다. 또한 실습 환경 구축을 위해 학습자가 실습실을 이용할 경우 상이한 버전에 따라 생기는 문제점을 해결하기가 어렵다. 그에 따라 실습실에서 사용되는 언어의 버전이 동일해야 한다. 몇몇 언어의 경우 서드파트 프로그램으로 언어 버전을 관리할 수 있는 툴을 제공한다. 그러나 이를 지원하지 않은 언어도 존재한다. 그러나 학습에 사용되는 모든 언어의 버전 관리 툴을 개발하는 것 현실적으로 불가능하다. 그에 따라 Docker를 이용하여, 각 언어별 실행환경을 이미지로 구축하여 사용되는 버전에 따라 이미지를 변경하여 실행하여 결과를 확인할 수 있다.

### 3.1.3 실습 결과 분석

실습 시간내에 교사가 모든 학생의 소스를 분석하고 채점하면서 지도하기는 어렵다. 프로그래밍 실습의 경우 입력을 하고, 그에 대한 처리를 진행하고, 출력을 가지고 채점을 하게 된다. 이를 이용하여 실습한 것에 채점을 하게 된다. 교사가 직접 보는 경우 학습자가 고의로 출력문을 이용하여 정해진 결과만 나오게 하는 것을 방지할 수 있다. 그러나 프로그램을 통해 처리하는 경우 문제와, 소스를 분석하지 못한 경우 처리하기가 어렵다. 이를 해결하고자 입력에 대한 출력을 미리 정의하여 정상적으로 작동을 하는지 테스트를 한다. 그러나 모든 프로그램이 정해진 입력에 대한 정해진 출력을 갖는 것은 아니다. 랜덤 함수를 이용하거나 시간에 따른 결과가 바뀌는 경우도 존재하기 때문이다. 이를 위해 교수가 수동으로 채점이 가능하도록 하게 한다.

### 3.2 설계한 시스템 구성도

(그림 1)은 시스템 구성도를 보여준다. 사용자는 웹 서버를 통해 채점 요청을 하게 된다. 웹서버는 Docker 서버에 미리 생성한 이미지를 통하여 소스 파일을 컴파일하고, 컴파일 및 실행한 결과를 웹서버에게 전달하고, 웹서버가 사용자에게 실행 결과를 알 수 있게 한다. 채점을 제외한 나머지는 웹 서버를 통해 처리한다.



(그림 1) 시스템 구성도

### 3.3 설계한 시스템의 진행 절차

<표1> 시스템의 진행 절차

Designed system process
1. professor opens a class
2. professor chooses compiler and compiler version
3. while(!classClosed)
4. {
5.     professor input practice examples
6.     professor chooses whether to check the answer manually or automatically
7.     if(professor chooses automatically check)
8.         professor input test cases
9.     students coding practice examples
10.    students submit codes
11.    try compile submit codes
12.    if(hasCompileErrors)
13.    {
14.         system notice the student
15.         the student retry examples
16.    }
17.    if(professor chooses automatically check)
18.         system compare executes result and test cases
19.    else
20.         professor check submit code and executes result
21.         if(hasSubmitCodeWrong)
22.             the student retry examples
23.    else
24.         save test result
25.    }

<표1>은 제안한 시스템의 진행 절차를 알고리즘으로 표현한 것이다. 이를 통해 교사가 설정한 컴파일러와 버전을 통해 자동으로 채점이 가능하다. 또한 자동 채점뿐만 아니라 수동으로 채점할 수 있도록 제공한다. 이를 통해 교사의 부담을 줄이고, 또한 학습자 스스로 실습 문제에 대해 제대로 코딩했는지 확인할 수 있다.

#### 4. 결론

본 논문에서는 Docker를 이용한 프로그래밍 실습 시스템을 설계하였다. 설계한 시스템은 다음의 효과를 갖는다. 첫째, 제안한 시스템은 학습자의 실습 채점을 자동으로 수행함으로써 교사의 부담을 줄여준다. 둘째, 실습실이 부족한 학교에서도 스마트폰이나 태블릿을 통해 실습 환경을 제공해 준다. 이를 통해, SW 교육이 중요해짐에 따라 부족한 정보 교사 및 실습 환경을 개선할 것으로 보인다. 향후 설계한 프로그래밍 실습 시스템을 구현할 계획이다.

#### References

- [1] 교육부, 2016, 소프트웨어 교육 활성화 기본계획
- [2] 최지예, “Design and Implementation of a Programming Learning Support System Considering Interaction with a Learner”, 2014
- [3] 박덕훈. “e-Learning 의 국내외 표준화 동향.” TTA Journal 103 (2006): 140-144.  
[ 4 ]  
<https://docs.oracle.com/javase/8/docs/technotes/guides/security/PolicyFiles.html> 17.03.20
- [5] Asa Shiho 저, 신은화 역, 2016, 완벽한 IT 인프라 구축을 위한 Docker, 정보문화사
- [6] 유진아m “공개소프트웨어 Python을 이용한 프로그래밍 교육에 관한 연구”, 2008
- [7] 이재선, and 창병모. “인터넷 상에서 BASIC 프로그래밍 실습 환경 연구.” 프로그래밍언어논문지 14.1 (2000): 22-26.
- [8] 이운재, “웹 기반 프로그래밍 교육 서비스의 학습자 경험 설계 방안 연구”, 2017
- [9] 김영지, et al. “웹 기반 프로그래밍 교육 시스템의 설계 및 구현.” 한국컴퓨터종합학술대회 논문집 32.1 (2005): 67-69.
- [10] 장혜선, et al. “컴퓨터교과교육: 에러 피드백 기반의 초보자를 위한 프로그래밍 학습 지원 시스템.” 컴퓨터교육학회논문지 10.6 (2007): 1-10.
- [11] 정찬선, 정광식, and 손진곤. “프로그래밍 언어 실습을 위한 웹기반 학습시스템의 설계 및 구현.” 한국정보과학회 학술발표논문집 (2006): 247-249.
- [12] 최지예, “학습자와의 상호작용을 고려한 프로그래밍 학습 지원 시스템 설계 및 구현”, 2014
- [13] 이재홍 , 2014, 가장 빠르고 편리 만나는 Docker, 길벗