

도커 컨테이너 기반 데이터베이스 플랫폼

허태성*, 홍지후^o

*인하공업전문대학 컴퓨터정보공학과,

^o인하공업전문대학 컴퓨터정보공학과

e-mail: tshur@inhac.ac.kr^o, gmactol23@gmail.com*

Docker Container Based Database Platform

Tai-Sung Hur*, Ji-Hoo Hong^o

*Dept. of Computer Information, Inha Technical College,

^oDept. of Computer Information, Inha Technical College

● 요약 ●

도커 컨테이너 기반 데이터베이스 플랫폼은 IT 개발자들을 위한 데이터베이스 플랫폼으로써 개발용 데이터베이스를 웹에서 생성할 수 있으며, 개발자는 개발 데이터베이스 세팅을 위한 시간을 줄일 수 있다. 데이터베이스 생성 후, 웹에서 데이터베이스 접속정보를 확인할 수 있으며, 서버상태를 분단위로 확인할 수 있다.

키워드: 도커(Docker), 데이터베이스(Database)

I. 서론

대부분의 IT 관련 프로젝트에서 데이터베이스관리시스템을 사용하고 있다. 새로운 프로젝트를 진행 할 때 해당 프로젝트에 개발 환경을 맞추기 위해 개발용 데이터베이스관리시스템을 설치해야 하며, MySQL MariaDB, Oracle 등 각 프로젝트마다 사용하는 데이터베이스관리시스템도 다양하고, 버전도 다양하다. 이런 문제점을 해결하기 위해 본 제안 시스템은 도커 컨테이너를 기반으로 하여 웹에서 MariaDB 데이터베이스 자동 생성 및 관리를 할 수 있도록 한다.

II. 프로그램 설계 및 구현

2.1 시스템 설계

2.1.1 시스템 구성도

본 개발 시스템은 그림 1에서 보느냐와 같이 사용자는 로그인 및 회원가입, 데이터베이스 컨테이너 생성 및 삭제 조회 및 백업을 진행할 수 있으며, 관리자는 그림 2와 같이 사용자 관리, 서버관리, 이미지 관리를 진행할 수 있다.

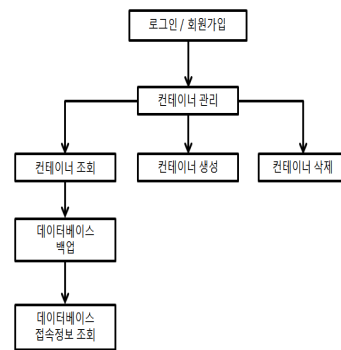


Fig. 1. 사용자 웹 구성도

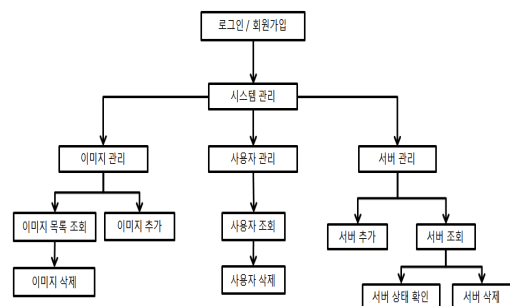


Fig. 2. 관리자 웹 구성도

또한 내부 컨테이너 생성 절차는 그림 3과 같다. 사용자가 컨테이너 생성을 요청할 경우 사용자마다 부여되어있는 컨테이너 생성 가능 개수 초과여부를 확인한다. 이 후, 등록되어있는 도커 서버들 중 컨테이너 개수가 가장 적은 도커서버를 조회하여 해당 서버에 컨테이너 생성을 진행한다.

컨테이너가 생성된 후, 사용자는 컨테이너 조회 화면에서 생성된 컨테이너의 데이터베이스 접속정보를 확인할 수 있다.

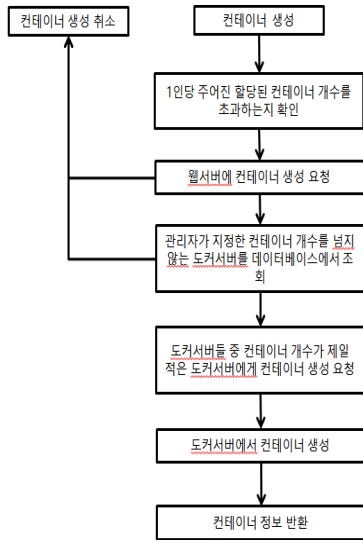


Fig. 3. 컨테이너 생성 절차

2.1.2 ERD

그림 4에서 보는 바와 같이 본 시스템은 총 5개의 테이블로 구성되어 있다.



Fig. 4. ERD(논리)

2.2 구현

본 시스템은 크게 웹서버와 도커서버로 나뉘며, 웹서버에서 웹컨테

이너로 톰캣8을 사용하여 데이터베이스를 실행하며, 도커서버에서는 서버 모니터링을 위한 파이썬 프로그램, 도커 제어를 위한 노드 프로그램을 실행한다.

2.3 개발 환경

Table 1. 시스템 개발환경

시스템 개발 환경	
OS	Window 10, Linux(Docker)
사용언어	Java (jdk1.8)
웹 프레임워크	SpringBoot
데이터베이스	MariaDB 10.3.13
웹컨테이너	Tomcat8

2.4 실행화면

그림 5는 시스템의 실행 화면이다. 그림 5에서는 도커서버를 관리할 수 있으며, 서버들의 CPU 사용률, 메모리 사용량을 그래프로 확인할 수 있다.



Fig. 5. 실행화면 - 서버관리

III. 결론

본 논문에서 구현한 시스템은 프로젝트 개발 시, 개발 데이터베이스 생성 및 관리를 수월하게 할 수 있도록 하는데 목적이 있다. 현재 MariaDB만 생성 가능하지만, 추후 MySQL, Oracle, postgres 등 더 많은 데이터베이스 및 버전을 지원할 수 있도록 추가할 계획이다.

REFERENCES

[1] Docker-API, <https://github.com/AgustinCB/docker-api>
 [2] 스프링부트, <https://spring.io/projects/spring-boot>
 [3] 도커 위키, <https://openwiki.kr/tech/docker>
 [4] Express 프레임워크, <https://expressjs.com/ko/>
 [5] 도커 허브, <https://hub.docker.com/>