

Spring Boot를 이용한 간단한 블로그 만들기

권순범*, 오재용*, 조승우*, 김성진***, 이형목**, 이준동^o

*강릉원주대학교 멀티미디어공학과,

***명지전문대학, ICT융합공학과,

**명지전문대학 컴퓨터공학과,

^o강릉원주대학교 멀티미디어공학과

e-mail: dice14@naver.com*, ojy5251@naver.com*, kjsw12@gwnu.ac.kr*, ict214548@mjc.ac.kr***, lhm1022@mjc.ac.kr**, jlee@gwnu.ac.kr^o

Make Simple Blog with Spring Boot

SunBeom Kwon*, JaeYong Oh*, SeungWoo Jo*, SungJin Kim***, HyungMook Lee**, JunDong Lee^o

*Dept. of Multimedia Engineering, GangNeungWonju National University,

***Dept. of ICT Convergence Engineering, Myongji College,

**Dept. of Computer Science and Engineering, Myongji College,

^oDept. of Multimedia Engineering, GangNeungWonju National University

● 요약 ●

스프링은 자바 엔터프라이즈 애플리케이션 개발에 사용되는 프레임워크로서, 애플리케이션을 빠르고 효율적으로 개발할 수 있도록 틀과 공통 프로그래밍 모델, 기술 API 등을 지원해주며, 스프링 부트는 스프링 프레임워크를 사용하는 프로젝트를 간단하게 준비할 수 있는 스프링 프레임워크의 서브 프로젝트이다.

본 연구에서는 스프링 부트를 활용하여 간단한 블로그를 설계 및 구현한다. 간단한 블로그에서는 다른 사람과의 식별을 위한 로그인 및 회원가입, 다른 사람과의 생각을 공유하기 위해 게시물 및 댓글 읽기, 쓰기, 수정, 삭제의 기능을 설계하였다.

이러한 설계는 스프링 부트를 사용하여 모듈 간의 의존성 관리, Spring Web MVC를 사용하여 서비스 로직과 사용자 인터페이스를 분리하며, Spring Data JPA, Spring Security 등을 이용하여 회원식별과 게시물 및 댓글 쓰기, 읽기, 수정, 삭제 등을 구현하였다.

키워드: Spring Boot, Spring Web MVC, Spring Data JPA, Spring Security

I. Introduction

스프링은 자바 엔터프라이즈 애플리케이션 개발에 사용되는 프레임워크로서, 애플리케이션을 빠르고 효율적으로 개발할 수 있도록 틀과 공통 프로그래밍 모델, 기술 API 등을 지원해준다[6].

스프링 부트는 스프링 프레임워크를 사용하는 프로젝트를 간단하게 준비할 수 있는 스프링 프레임워크의 서브 프로젝트로 독립 컨테이너에서 동작할 수 있으므로 더욱 쉬운 개발이 가능하다.

기존의 웹개발은 Apache, PHP, MySQL을 조합하여 개발을 진행하였다[1]. 위와 같은 조합으로 개발하면 문법을 배우기 쉽고 직관적이므로 빠른 속도로 개발이 가능하다. 하지만 유지보수 및 확장성과 방대한 트래픽 처리에 취약한 단점을 가지고 있다. 따라서 이러한 단점을 보완한 JAVA 언어를 사용한 Spring Boot로 웹을 제작하고자

한다.

블로그는 웹(web)과 로그(log, 기록)를 합친 단어로, 자신만의 생각 또는 알리고 싶은 견해들을 게시물로 작성하고, 댓글을 통해서 다른사람과 자유롭게 의견을 주고 받을 수 있다.

본 논문에서는 스프링부트를 활용하여 블로그를 구현하였다. 블로그의 특징을 구현하기 위해 다른 사람과의 식별을 위해 로그인 및 회원가입, 다른 사람과의 생각을 공유하기 위해 게시물 및 댓글 읽기, 쓰기, 수정, 삭제의 기능을 설계하였다.

II. Preliminaries

2.1 스프링 부트(Spring Boot)

Spring Boot는 단독으로 사용가능한 수준의 Spring 기반의 어플리케이션을 쉽게 만들어주는 프레임워크이다[2].

Spring Boot의 특징은 아래와 같다.

1. Spring의 경우, 모듈의 의존성에 대하여 직접 설정해야 하지만, Spring Boot는 "spring-boot-starter"를 지원하여 자주 사용되는 모듈 간의 의존성 버전을 권장버전으로 자동 설정 해준다.
2. Spring은 모듈설정을 하기 위해 Configure 클래스 및 xml 파일을 직접 작성하여야 했지만, Spring Boot 는 yml 파일 및 properties 파일을 통해 설정을 간편하게 작성할 수 있다.
3. Spring의 경우 war 파일로 패키징한 후 외부의 WAS 설치 후 직접 배포해야 하지만, Spring Boot는 내장 WAS(Tomcat, Jetty)을 지원하여 내부 WAS에 jar파일로 간편히 배포가 가능하다. 또한 war 파일로 패키징이 가능하여 외부 WAS를 사용가능하다.

2.2 스프링 의존성 라이브러리

1) Spring Web MVC

MVC(ModelViewController)란 소프트웨어 공학에서 사용되는 아키텍처 패턴으로 비즈니스 로직과 사용자의 인터페이스를 분리함으로써 로직 간의 서로 영향 없이 쉽게 고칠 수 있도록 하는 방법론을 의미한다[3].

Spring Web MVC는 MVC Model 2 아키텍처를 사용하며, Spring을 기반으로 하고 있어서 트랜잭션 처리, DI(의존성 주입), AOP(관점 지향 프로그래밍) 등을 손쉽게 사용할 수 있다.

2) Spring Data JPA

JPA는 Java Persistence API의 약자로, 자바 어플리케이션에서 관계형 데이터베이스를 사용하는 방식을 정의한 인터페이스이다. JPA의 경우 기존의 반복적인 코드와 객체지향 프로그래밍과 관계형 데이터베이스 사이에 존재하는 패러다임의 불일치를 해결하기 위해 나온 기술이다.

Spring Data JPA의 경우, JPA를 Spring 프레임워크에서 간편하게 사용할 수 있도록 만든 프로젝트이다. CRUD처리를 위한 공통 인터페이스를 제공하며, repository 개발시 인터페이스만 간단하게 작성하면 동적으로 구현객체를 주입해준다[4].

3) Spring Security

Spring Security는 Spring 기반의 어플리케이션의 보안(인증과 권한, 인가 등)을 담당하는 프레임워크이다. Spring Security는 인증과 권한에 대한 부분을 Filter 흐름에 따라 처리하고 있다[5].

Spring Security는 보안과 관련하여 체계적으로 많은 옵션을 제공해주기 때문에 개발자 입장에서는 일일이 보안관련 로직을 작성하지 않아도 된다는 장점이 있다.

III. The Proposed Scheme

3.1 블로그 설계

블로그를 위한 클래스 파일 계층도는 그림 1과 같다.

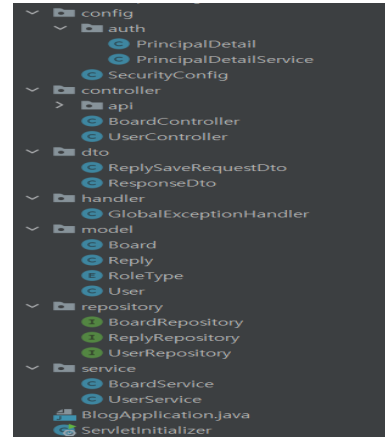


Fig. 1. 클래스 파일 계층도

1) Config

- Spring Security를 사용하여 로그인 및 회원가입, 접근제어 로직을 담당

2) Controller

- Spring Web MVC가 적용
- Controller는 사용자가 요청 할 경우, 해당 요청을 처리하는 서비스를 호출하여 처리한 후, 사용자에게 반환하는 역할

3) DTO

- 서로 다른 계층의 데이터를 처리하기 위해 사용
- 요청한 서비스의 응답코드 표시하기 위해 사용

4) Handler

- 예외가 발생하였을 경우, 사용자에게 오류페이지로 유도

5) Model

- 게시물, 댓글, 사용자, 역할 등에 대하여 클래스로 정의
- Spring Data JPA의 @Entity를 사용하여, 객체와 데이터베이스의 테이블에 매핑

6) Repository

- 작성했던 Model을 기반으로 데이터베이스에 접근해서 읽기, 쓰기, 수정, 삭제 작업을 수행
- Spring Data JPA의 JpaRepository 인터페이스를 상속하여서 사용

7) Service

- Model 과 Repository를 이용하여, 사용자가 요청한 서비스를 처리.
- 게시물 및 댓글의 읽기, 쓰기, 수정, 삭제 등을 담당

8) BlogApplication.java

- 내장 WAS에서 실행시킬 때 필요한 파일

9) ServletInitialzier.java

- 외장 WAS에서 실행시킬 때 필요한 파일

블로그 제작에 사용된 클래스들에 대한 의존성을 다이어그램으로 나타내면 그림 2와 같다.

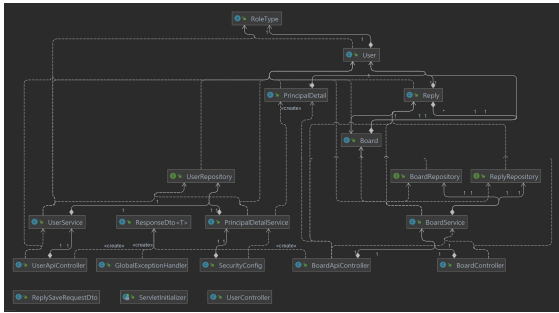


Fig. 2. 클래스 다이어그램

3.2 구현 결과

그림 3은 최초 접속시 사용자에게 보여지는 페이지이다.

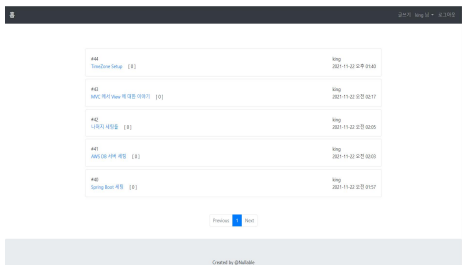


Fig. 3. 메인 페이지

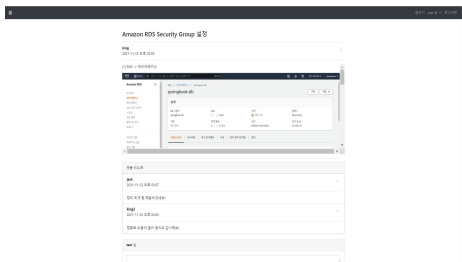


Fig. 4. 게시물 페이지

그림 4는 사용자가 작성한 게시물을 확인하고 댓글을 작성할 수 있는 페이지이며, 그림 3는 사용자가 글을 작성할 수 있는 페이지의 구현 결과이다.

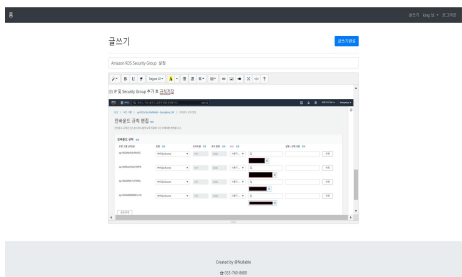


Fig. 5. 글쓰기 페이지

IV. Conclusions

본 연구에서는 스프링부트를 활용하여 간단한 블로그를 설계 및 구현하였다.

스프링부트를 활용하여 웹을 개발하면, 기존의 웹개발 방식인 APM을 활용한 것 보다 유지보수 및 확장성 방대한 트래픽 처리에 장점이 있다. 또한, 모듈 간의 의존성과 설정을 간편하게 작성할 수 있으므로, 개발환경을 보다 쉽게 만들 수 있다.

이와 더불어 Spring Web MVC를 사용하여, 서비스 계층과 사용자 인터페이스의 계층을 분리하여, 변경에 유연한 소프트웨어를 제작할 수 있으며, Spring Data JPA를 게시물 및 댓글 구현에 사용하여 반복적인 쿼리 작성을 줄여주고, 변경에 유연하여 생산성이 높은 웹의 구현이 가능하다.

마지막으로 Spring Security를 로그인 및 회원가입을 구현하기 위해 사용하여, 편리하게 보안에 대한 로직을 작성하였다.

ACKNOWLEDGEMENT

이 논문은 2021년도 정부(산업통상자원부)의 재원으로 한국 산업 기술진흥원의 지원을 받아 수행된 연구임(과제번호: P0011930, 2021년 산학융합지구조성사업).

REFERENCES

- [1] 황재호, 2019, PHP 프로그래밍 입문
- [2] Spring Boot, Spring.io
- [3] 위키피디아, 모델-뷰-컨트롤러
- [4] 김영한, 2015, 자바 ORM 표준 JPA 프로그래밍
- [5] 구명가게 코딩단, 2019, 코드로 배우는 스프링부트 웹 프로젝트
- [6] 이일민, 토비의 스프링 3.1, 에이콘출판주식회사, 2012