

Spring-Boot JPA를 활용한 소셜 온-오프 네트워크 서비스

김대철
경희대학교 컴퓨터공학과
kdc9619@khu.ac.kr

Social On-off network Service using Spring-Boot JPA

Daecheol Kim
Department of Computer Science and Engineering, KyungHee University

요 약

온라인에서만 아니라 오프라인을 통해서도 사회적 관계망을 형성할 수 있는 소셜 온-오프 네트워크 서비스를 개발하고자 한다. 이는 기존 SNS들이 가지는 장점을 흡수하고 한계점들을 보완하여 건전한 사회적 관계를 만들기 위한 어플리케이션이다.

1. 서 론

1.1. 연구배경

대부분의 사용자들은 온라인 상에서 뿐만 아니라 오프라인상에서도 다양한 사회적 관계형성에 대한 기대를 원하고 있다. 그리고 기존 SNS에서는 본인과 비슷한 의견과 생각 등을 지속적으로 수집함으로써 편향된 사고를 가질 확률이 매우 높다. 이로 인해 다양한 사람들과 갈등이 생길 확률이 매우 높다. 결과적으로 서로에게 도움을 주고 받는 것에 대해 각박한 사회가 되어가고 있다. 따라서 스타트업 드림의 일환으로 S.O.S라는 어플리케이션을 개발하여 많은 도움들을 주고 받을 수 있도록 하여 온라인 뿐만 아니라 오프라인에서도 사회적 관계를 만들어 건강한 사회를 만들고자 한다.

1.2. 연구목표

S.O.S(소셜 온-오프라인 네트워크 서비스)는 온라인상에서 외에도 오프라인에서도 다양한 사회적 관계를 형성하여 서로에게 도움을 주고 받는 것에 각박한 사회에서 벗어나고자 한다. 또한, 기존의 SNS로 인한 갈등 및 중독 등의 사회적 문제를 해결하고자 한다. 그리고 기존 SNS에서 본인과 비슷한 의견들을 수집하여 편향적인 사고를 가지는 것에서 벗어나 다양한 생각과 가치관의 콘텐츠를 공유할 수 있는 형태의 SNS를 개발하고자 한다.

2. 관련연구

2.1. Spring Boot

Spring Boot는 스프링 포트폴리오를 신속하게, 미리 정의된 방식으로, 이식성 있게, 실제 서비스 환경에 사용할 수 있도록 조합해 놓은 것이다.

2.1.1 신속성

의존관계를 포함하여 애플리케이션의 여러 요소에 기반한 의사결정을 신속하게 적용할 수 있게 하여 개발속도를 높인다.

2.1.2. 미리 정의된 방식

스프링 부트의 구성을 정하면, 그 구성에 따라 미리 정의된 방식으로 기본적인 설정을 자동으로 지정해 준다.

2.1.3. 이식성

자바의 표준도구인 스프링 프레임워크를 기반으로 만들어져, JDK가 있다면 어느 환경에서나 실행할 수 있다. 또한, 특정한 인증을 받은 애플리케이션 서버가 필요하지 않으므로 스프링 부트로 애플리케이션을 만들고 패키지를 만들면 어디에서든 배포하여 실행이 가능하다.

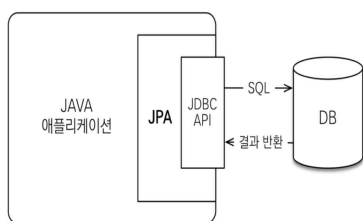
2.1.4. 실제 서비스 환경에서 사용 가능

작은 부분에서 사용해야 하는 제약이 없으며, 실제로도 광범위하게 사용되고 있다. 실제로 넷플릭스의 경우에는

클라우드 인프라의 요구사항인 안정성, 확장성, 효율성, 보안을 충족하기 위해 스프링 부트를 채택하였고, 일부는 넷플릭스 자체의 소프트웨어의 사용 및 조정을 통해 인프라를 발전시켰다.

2.2. JPA

JPA는 JAVA의 ORM 기술 표준으로, 인터페이스의 모음을 의미한다. 현재 Hibernate, EclipseLink, DataNucleus의 JPA 2.1 표준 명세를 구현한 3가지 구현체가 존재한다. JPA는 애플리케이션과 JDBC 사이에서 동작하며, 개발자가 JPA를 사용하면 내부에서 JDBC API로 SQL을 호출하여 DB와 통신하도록 한다.



[그림 1] JPA의 동작과정

JPA는 코드의 무한반복, 객체와 관계 DBMS의 차이 등 SQL 중심적인 개발에서 발생할 수 있는 문제점을 해결하고자 한다. 또한, 간단한 CRUD 구문을 통해 생산성을 높일 수 있고, 그로 인해 유지보수도 용이하게 할 수 있다. 그리고 Object와 RDB간의 패러다임의 불일치를 해결하고, 성능을 최적화 하기 위해 JPA를 사용한다.

2.3. Kotlin

2.3.1. JAVA와 Kotlin의 비교

Java와 Kotlin은 모두 정적타입으로 컴파일러에서 Type-Error를 찾을 수 있다. 또한, 멀티 플랫폼 개발을 할 수 있는 장점을 보유하고 있다. 그리고 모두 객체지향 프로그래밍을 지원하여 class 키워드를 사용할 수 있는 공통점을 가지고 있다.

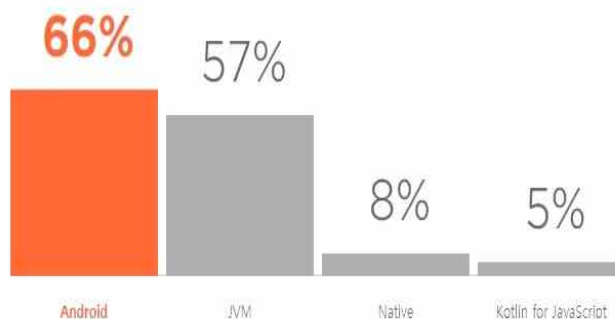
하지만, Java는 class가 기본단위로 class 안에서 로직을 만들어야 하는 반면, 코틀린은 class가 선택사항이다. 따라서 Top-Level에서 변수, 함수의 선언과 구현이 가능하다. 그리고 자바 9이하의 버전에서는 타입 추론이 불가능하지만, 코틀린은 타입추론이 가능하다. 마지막으로 Null Point의 예외처리에서 자바의 경우에는 런타임에 Null 참조시 예외를 발생시킬 수 있다. 그에 반해 코틀린은 Null을 확인하는 연산자가 존재하여 이를 컴파일 시점에서 방지할 수 있다.

	Type system	Multi-Platform	OOP	FP	Type Inference	Nullable
자바 (Java)	정적 타입	0	0	X	X (9 이하)	X
코틀린 (Kotlin)	정적 타입	0	0	0	0	0

[그림 2] JAVA와 Kotlin의 비교

2.3.2. Kotlin의 장점

코틀린은 다른 사람이 이해하기 쉽도록 코드를 작성할 때 시간이 단축된다. 이로 인해 코드가 짧으며 가독성이 매우 좋다. 이로 인해 에러발생률을 낮출수 있다. 또한, coroutines, extension functions, lambdas 등의 다른 라이브러리와 Android Jetpack 을 지원한다. 그리고 Java와 호환이 가능하여 기존의 어플리케이션을 모두 Kotlin으로 손쉽게 바꿀수 있다. 이로 인해 Kotlin은 2019년 기준으로 안드로이드나 JVM에서 대부분 사용하고 있다.



[그림 3] 2019년 기준 개발자들의 Kotlin을 언어로 하는 플랫폼 비중

3. 어플리케이션에서의 데이터 설계 및 통신

S.O.S 어플리케이션에서 필요한 데이터 테이블은 회원, 프로필, SOS(도움 요청), 채팅방, 채팅이다.

회원 테이블에는 사용자의 모든 정보를 저장하는 테이블로 로그인에 필요한 ID, 이메일, 비밀번호와 함께, 사용자의 기본정보인 이름, 닉네임, 생년월일 등이 들어가도록 한다.

프로필 테이블에는 사용자의 프로필 정보를 간략하게 요약하는 정보가 들어가고 이와 관련한 추가 이미지에 대한 URL이나 메시지 정보 등을 저장할 수 있다. 이때 사용자 한명은 하나의 프로필을 가질 수 있다.

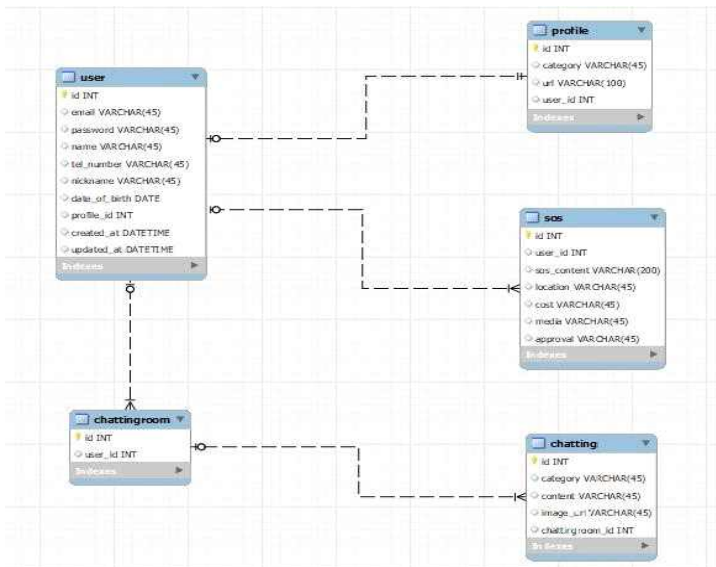
SOS 테이블은 사용자가 요청한 도움 목록들이 저장되는 테이블이다. 한 사람은 여러 개의 SOS를 등록할 수 있어 회원과 1대N의 관계를 가지도록 한다.

채팅방은 SOS에 개설된 채팅방들을 저장하는 테이블로

개설자와 채팅방 제목 등이 저장된다. 이 때, 사용자는 여러 개의 채팅방을 가질 수 있다.

채팅은 채팅방에서 오고 간 메시지들을 저장하는 테이블로 하나의 채팅방에는 여러 개의 채팅 내용을 가지게 된다.

채팅 구현 시 Socket 방식으로 서버에 보내고 이를 다시 클라이언트에 다시 돌려주는 방식이 아닌 채팅 입력 시 DB에 저장되고 이를 설계한 API를 통한 AJAX 통신으로 구현하도록 한다.



[그림 4] S.O.S 어플리케이션 데이터베이스 ER Diagram

5. 결론 및 향후연구

S.O.S는 서로 돕고 다양한 가치관이 공존하는 사회를 추구하고, 온-오프라인의 복잡한 사회적 관계망 서비스를 지향하고 있다. 또한, 도움 플랫폼을 통해 팀원 모집, 중고 거래 등의 다양한 서비스들을 중개함으로써 행복한 사회를 꿈꿀 수 있을 것으로 기대된다. 그리고 차후에는 온라인 의류 쇼핑몰이나 맛집 등과 협업을 통해 서비스를 확장하여 다양한 반응형, 노출형 광고를 통한 수익을 얻을 수 있을 것으로 예상된다.

향후 연구로는 데이터베이스에 대한 API 설계를 진행하고, 이에 대한 응답시간이나 실행시간 등을 분석하고, 인덱스 등을 활용하여 더 효율적인 쿼리를 위한 방법론들을 연구할 계획이다.

참고문헌

- [1] 스프링 부트와 AWS로 혼자 구현하는 웹 서비스
- [2] 스프링 부트 실전 활용 마스터
- [3] [JPA] JPA란
<https://gmlwjd9405.github.io/2019/08/04/what-is-jpa.html>
- [4] Kotlin vs JAVA <https://bbaktaeho-95.tistory.com/50>
- [5] Kotlin 2019 - 2019년 개발자 에코시스템의 현황 인포그래픽
<https://www.jetbrains.com/ko-kr/lp/devecosystem-2019/kotlin/>
- [6] Kotlin 공식 문서 <https://kotlinlang.org/docs>