

안드로이드 애플리케이션 로그인 세션관리

이태권*, 권두위*, 박수현*

*동서대학교 컴퓨터정보공학부

e-mail: dlxornjs33@gmail.com*, kdoowy@hanmail.net*, subak63@gmail.com*

Management the log-in session Using Android Application

Tae-Kwon Lee*, DooWy Kwon*, SuHyun Park*

*Dept of Computer and Information, Dongseo University

요 약

본 논문에서는 안드로이드 스마트폰을 이용하여 회원가입, 로그인, 로그아웃 및 세션유지를 할 수 있는 방법을 제안했고, Android, php와 MySQL을 사용하여 개발했다. 로그인관리를 효율적으로 하기 위한 알고리즘을 설계했고 이 알고리즘을 이용하여 회원가입, 로그인, 로그아웃 및 세션유지를 할 수 있는 구현화면을 제작했다. 처음 회원가입 기능을 통해 아이디와 비밀번호를 등록 후 로그인을 시도하면 로그인인증 과정을 거치게 되며 인증에 성공하면 다음 화면으로 이동된다. 로그인인증 후 사용자의 조작에 의하여 안드로이드 Activity간 이동이 있더라도 php에서 제공하는 세션유지 기능을 사용하여 로그인상태를 유지시키며, 로그아웃을 하게 되면 초기화면으로 이동하고 세션이 해제됨으로써 로그인유지 세션이 종료된다.

1. 서론

현재 구글 플레이 스토어에서 사용 가능한 애플리케이션 수가 70만개를 돌파했고 한 달에 약 2만5천개 이상의 애플리케이션이 새롭게 추가되고 있다[1].

또한 현재 우리나라 국민 10명 중 7명 이상인 3721만 명이 스마트폰을 사용하고 있다[2].

이처럼 스마트폰은 이제 우리 실생활에서 빼놓을 수 없는 필수품이 되었다. 스마트폰 애플리케이션 사용자가 증가함에 따라 애플리케이션 개발자는 내가 출시한 애플리케이션을 누가 사용하고 있는지, 여러사항은 없는지 확인해야 할 필요가 있다.

위의 문제를 해결하기 위해서는 애플리케이션에 로그인 기능을 구축하여 사용자들의 정보를 데이터베이스를 사용하여 관리해야 하고 보안에도 신경 써야한다. 하지만 로그인 기능을 제공하는 애플리케이션을 보면 대부분 안드로이드 내에서 간단한 로그인 인증과정만 거친 후 다음 화면으로 이동하거나 웹으로 구성되어진 하이브리드 애플리케이션을 이용하여 로그인 처리를 하는 것이 대부분이다. 본 논문에서는 안드로이드 내에서 제공하는 HTTP객체를 통해 외부에서 만들어놓은 웹 페이지와 서로 연동하여 세션을 유지하고 로그인관리를 효율적으로 하는 방법을 알아보겠다.

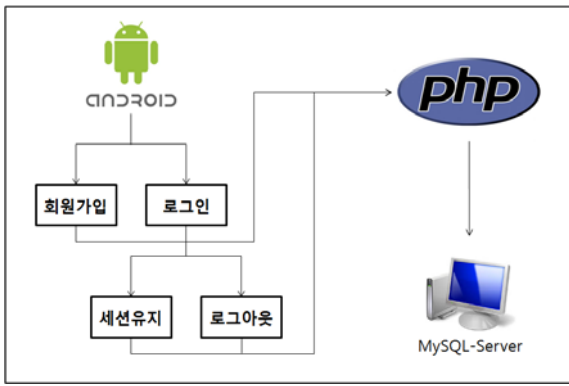
2. 본론

1) 시스템 구성

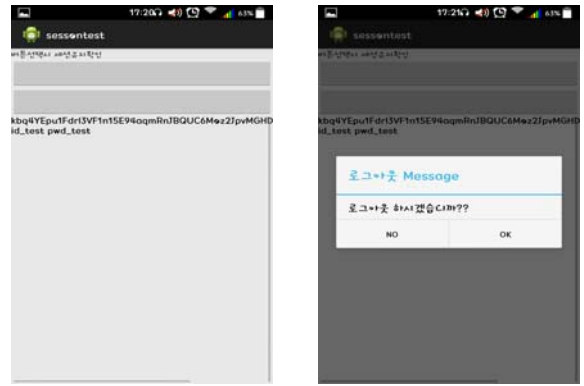
안드로이드 스마트폰을 이용한 세션유지는 개발자가 안드로이드를 기반으로 회원가입, 로그인, 로그아웃 및 세션유지 기능을 구축하고자 할 때 많은 도움을 준다. 본 애플리케이션의 시스템 내용 및 기능 구성은 다음과 같다.

첫째, 회원가입 기능으로 사용자가 등록할 아이디와 비밀번호를 입력 한 후 회원가입을 하게 되면 회원정보가 등록 된다. 둘째, 로그인 기능으로 회원가입 후 등록 되어있는 아이디와 비밀번호를 입력하고 로그인을 시도하게 되면 서버와의 통신을 통해 로그인 인증과정을 거친 후 로그인이 된다. 셋째 세션유지 기능으로 로그인 후 사용자의 조작에 의해 화면이 다른 화면으로 변경되면 비정상적으로 접근한 사용자는 없는지 세션 검사를 통해 확인과정을 거친 후 이상이 없으면 로그인을 계속 유지하고, 그렇지 않으면 강제로 로그아웃 기능을 수행한다. 넷째 로그아웃 기능으로 사용자가 로그아웃을 시도할 경우 서버와의 통신을 통해 로그아웃 기능이 호출되며, 현재 유지하고 있는 세션기능을 해제한 후 로그아웃을 수행한다.

앞서 시스템 내용 및 기능에 대하여 설명하였고 이제부터 시스템을 구현함에 있어 어떤 언어를 사용하였는지, 어떤 알고리즘을 적용하여 세션을 유지하며 로그인관리를 하고 있는지의 방법을 설명하겠다.

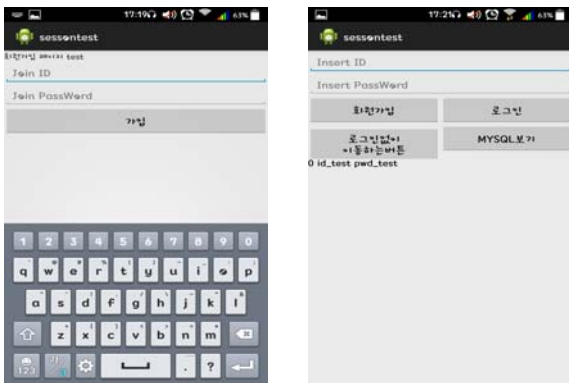


(그림 1) 시스템구조



(그림 3) 로그인성공 & 로그아웃

2) 시스템 구현



(그림 2) 회원가입 & 로그인

첫째, 회원가입 기능은 사용자가 가입할 정보를 입력한 후 그 정보를 보안이 강한 POST방식으로 서버에 있는 php파일에 전송한다. php는 전송된 데이터를 받아와 MySQL 데이터베이스에 연결한 후 받아온 데이터를 입력한다. 데이터베이스 테이블의 속성은 로그인 세션유지 검사를 위한 Token속성, 사용자의 ID를 저장하는 ID속성, Password를 저장하는 PWD속성으로 만들었다.

둘째, 로그인 기능은 ID와 Password를 입력하면 앞서 회원가입 기능에서 설명했던 POST방식으로 데이터를 전송 후, php에서 데이터를 전송받아 MySQL 데이터베이스 테이블에 저장된 Id와 그에 해당하는 Password가 일치하는지 검사한다. 로그인 인증에 성공하면 앞서 설명한 Token 속성에 64개의 랜덤 난수를 생성하여 데이터베이스에 입력한 후, 서버에 XML형태의 문서를 생성하여 로그인이 성공했다는 문자를 입력한다. 로그인 인증 실패 시 XML 문서에 로그인이 실패했다는 문자를 입력한다.

그 후 안드로이드 XML파싱 기능을 통해 앞서 php가 생성한 XML파일을 읽은 후, 로그인이 성공하였다는 메시지가 출력된다면 로그인을 허용하고 다음 화면으로 이동한다. 반대로 로그인이 실패했다는 메시지가 출력된다면 로그인을 허용하지 않는다.

셋째, 세션유지 기능은 로그인을 성공한 후 사용자가 조작을 통해 다른 화면으로 이동하는 버튼을 클릭하게 되면 앞서 생성한 Token과 현재 로그인되어 있는 Id를 php에서 제공하는 세션함수를 이용하여 저장하게 된다. 다음 화면으로 넘어가게 되면 이전 화면에서 저장한 Token과 Id를 세션함수를 통해 불러오게 되며 이전 화면에서 불러온 Token과 현재 데이터베이스에 입력되어 있는 Id에 해당하는 Token을 비교함으로써 로그인 세션을 유지한다. 넷째, 로그아웃 기능은 사용자가 로그아웃을 시도할 경우 현재 데이터베이스에 입력되어 있는 Id에 해당하는 Token을 0으로 초기화 시키고 php의 세션을 해제함으로써 로그아웃 기능이 수행된다.

3. 결론

본 논문에서는 Android, php, MySQL을 이용하여 회원가입, 로그인, 세션유지 및 로그아웃에 대한 기능구현에 대하여 소개하였다. 안드로이드의 HttpClient, HttpPost 두 객체를 사용하여 서버에 있는 php파일과 연결하였고 데이터를 전송하였다. 또한 각 사용자의 세션을 유지함에 있어 임의의 Token을 발급하여 비교함으로써 비정상적으로 접근하는 사용자를 차단할 수 있다. 추후 세션뿐만 아니라 쿠키를 함께 사용하여 응용하고, 개발함으로써 체계적으로 로그인 관리가 되게 할 것이다.

Acknowledgment

“이 논문은 2014년도 Brain Busan 21사업에 의하여 지원되었음“

참고문헌

- [1] i news 24, http://news.inews24.com/php/news_view.php?g_serial=700916&g_menu=020600&rrf=nv
- [2] iusm, <http://www.iusm.co.kr/news/articleView.html?idxno=447983>