# 基于移动终端和二维码的系统登录技术研究

摘要：随着网络技术的普及，人们的生活已经离不开网络了，而在各种软件应用中，少不了要进行个人账户的登录，这就涉及到了个人隐私的安全问题，研究并提出更安全可靠的登录方式成为了需要。本文针对这一问题，提出了基于二维码的登录方式，大致过程就是用户要在网站上进行用户登录操作时，网页上会根据Web服务端的信息生成一个二维码，用户通过移动端，如android手机等打开相应的应用程序进行扫码登录，从而达到快速安全的登录效果。本文将对这一过程的设计和实现进行详细的讲解，并给出相关代码，方便其他软件开发者对自己的软件安全性进行完善，将二维码登录技术快速的集成到已存在的系统中，减少相关投入，实现工业效益的最大化。

关键字：QRcode Android JavaEE

一、引言

当今时代是互联网+的时代，人们生活的各个方面都实现了互联网化，各个软件的使用都离不开注册和登录，这一过程的安全性将直接影响到客户对该软件的信任程度。与此同时，不难发现二维码在我们的生活中也是无处不在的，人们现在使用最多的支付方式无疑就是扫描二维码进行支付，这个过程快捷方便。除此之外，还可以发现现在很多信息都是隐藏在二维码中的，比如街上的宣传海报上都会有一个二维码，有的是扫描加群，有的是扫描转到一个网站，这样会为用户省去很多麻烦，不用手动输入网址，只需拿出手机扫描即可。

在以前许多Web应用程序的登录都是采用输入账号和密码的方式，外加一个随机验证码来排除非人工的登录。可想而知，每次的登录都要输入账号和自己为了防止盗号而想出的各种复杂的密码，这样的登录方式给用户带来了诸多不便，而且安全性也得不到保障。现在先进的几种登录方式有手机短信或邮箱进行验证登录，用户在登录时输入账号，这个账号通常为绑定的手机号或者是电子邮箱，系统在进行账号验证，如果存在该账户则会给该账户绑定的手机号或者是邮箱发送一个随机字符的验证码，再由用户输入登录。除了邮箱短信验证，另外一种现在常用的登录方式就是本文所说的扫描二维码进行登录的方式，本文提出了通过移动终端加上二维码的方式进行登录的方案，旨在为用户带来方便安全快捷的登录体验。

二维码技术早在上个世纪九十年代就已经产生了，但是近几年随着移动终端的迅速普及才开始用到这个技术。二维码的编码方式是对外开放的，研究者可以根据相关协议对二维码进行快速的解析，得到编码前的内容。在本文的设计方案中，二维码在Web服务器中进行生成，然后用户通过android手机进行二维码的扫描和解析，并进行相关登录信息的提交到服务器，再由服务器提交到数据库。

本文将先后进行设计整体说明，对系统的结构以及验证过程进行阐述；分析二维码技术发展，以及本文用到的二维码开源框架ZXing；对比Android系统和IOS系统进行比较，对Android系统做大致的介绍，以及逐步介绍移动端功能的实现；对系统后台的功能和设计进行详细说明，介绍javaEE中servlet的用法，javaScript脚本的使用，jdbc的使用，以及数据库的设计和使用。

二、基于移动终端登录系统的整体设计

1、系统结构

二维码扫描登录系统的整体结构如图1所示，分为四个部分，即浏览器、Web系统、移动端登录软件、Web系统数据库。

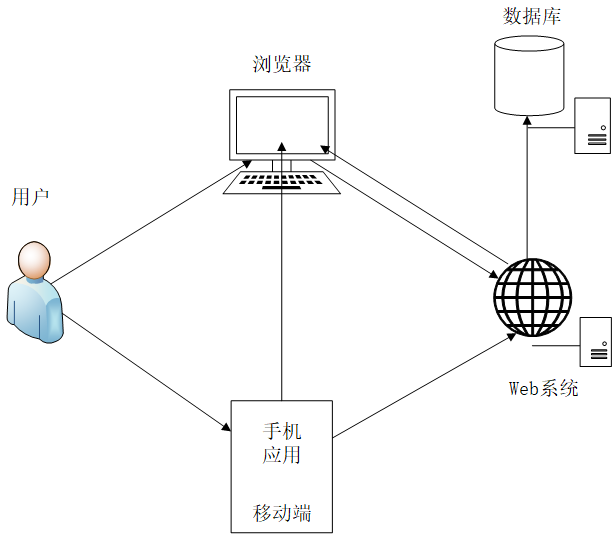


图1 基于移动终端登录系统的结构图

该图中，由于本设计针对的是Web系统，所以浏览器就是应用系统的客户端，而Web系统依托于Tomcat服务器存在，为用户提供设计好的服务，通过互联网进行信息传递，本文只针对其中的登录功能进行设计。图中的手机应用为用户登录时需要下载安装的应用，而数据库存储了用户的信息和登录相关的数据。

该系统的验证过程可以分为以下几步：

（1）用户打开计算器上的浏览器访问该Web系统，如果是第一次访问，Web系统还没有记下登录状态，则需要用户进行登录操作。Web系统会在网页上输出一个二维码图片供用户扫描。而二维码的内容则是本次会话的标识数据，在本设计中就是一个该Web系统的网址链接上本次访问的session id。

（2）当网站上出现二维码后，用户可以打开手机上进行登录的应用进行登录操作。该应用需要用户采用传统的账号密码进行登录，登录成功后才能使用扫描二维码的功能。该应用扫描二维码后会根据和Web系统协商好的数据格式对二维码中的数据进行解析，验证该二维码是否为本Web系统登录使用的二维码，若是，则将二维码中的session id包含在Web系统的登录请求中发送出去。

（3）Web系统中扫描二维码登录的网页会不断重复的给Web系统发送查询登录状态的请求，若移动端已经将登录请求发送成功，则当Web系统查询到该用户已经跳转都已登录状态，则页面会跳转为登录成功后的页面。

2、系统主要功能的实现

2.1系统整体流程图

分析上文系统进行的登录验证的过程，可以将整个系统的功能大致分为以下三个部分，即登录信息的传递、移动终端的登录、登录状态的转变。方案流程图如图2所示。

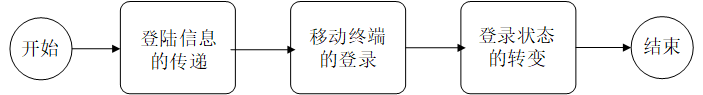


图2 方案整体流程图

2.2 登录信息的传递

在整个登录过程中，二维码的功能尤为重要，它使得登录信息能够存储其中并不能被其他系统所用。整个系统把二维码的生成和识别分为两个模块，二维码的生成在Web系统中完成，二维码的扫描在手机应用上完成。生成和扫描的二维码中的数据格式需要协商统一，才能正确的识别信息。

二维码作为信息传递的关键要素，其中存储了两种数据，一个则是登录所需的地址，用来验证是否为此系统生成的二维码，用以确定后续是否向系统发送登录请求。另外一个则是会话标识符，即Web系统中的session id，这一数据是整个系统中最为重要的数据，其能够让Web系统知道是哪个浏览器进行了登录操作，是将使用手机的用户和浏览器用户联系起来的关键数据。以上两种数据采用“，”进行链接。后续移动终端扫描到信息后也将按照此规则解析出登录地址和会话标识符，并验证登录地址后再向系统提交会话标识符，否则就终止登录过程。

2.3 移动终端的登录

用户在使用移动端的应用时，同样需要进行登录操作，可以看做是把Web端的登录转移到移动端的登录上，简化Web端的登录操作。用户在移动端登录成功后，来到个人信息的界面，并可以进行二维码扫描，扫描成功后根据信息将账户号以及会话标识符再提交到Web服务器上。

2.4 登录状态的转变

Web服务器接收到移动端发送来的账户号和会话标识符将查询数据库将会话标识符即session id保存到指定账户号对应的用户表记录中，并将该条记录中的登录状态转变为已登录状态。此时进行登录的网页将在后台不断地根据本地的session id查询数据库中是否有该session id，并判断其登录状态，若都符合则登录成功，页面进行跳转。除此之外，登录成功后还要设计一个用户注销的按钮，来注销数据库中的用户登录状态。

三、二维码技术分析

1、二维码技术的发展

2、二维码开源框架——ZXing

四、移动端的设计方案

1、Android和IOS系统的比较

2、Android系统的介绍

3、移动端具体功能的实现

五、后台服务器的设计方案

1、后台开发语言的选择

2、java EE的简介

3、servlet的功能和用法

4、java script的功能和用法

5、后台服务器的具体设计思路