武汉理工大学毕业设计（论文）

基于移动终端和二维码的系统登录技术研究

学院（系）： 信息工程学院

专业班级： 电信1401班

学生姓名： 彭宇

指导教师： 龙毅宏

学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包括任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

作者签名：

年 月 日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士论文评选机构将本学位论文的全部或部分内容编入有关数据进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于1、保密囗，在 年解密后适用本授权书

2、不保密囗 。

（请在以上相应方框内打“√”）

作者签名： 年 月 日

导师签名： 年 月 日

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。

关键词：QRcode；Android；JavaEE

**Abstract**

With the popularization of network technologies, people’s lives cannot live without the Internet. In various software applications, it is indispensable to log in to personal accounts. This involves personal privacy security issues, and studies and proposes more secure and reliable. The login method became necessary. This article aims at this issue, and proposes a login method based on two-dimensional codes. The general process is that when a user wants to perform a user login operation on a website, a two-dimensional code is generated on the web page according to the information of the web server, and the user passes the mobile terminal, such as Android phone and other open the corresponding application for scan code login, so as to achieve fast and safe login results. This article will explain in detail the design and implementation of this process, and provide related code, to facilitate other software developers to improve their own software security, the two-dimensional code login technology is quickly integrated into the existing system , Reducing related investment and maximizing industrial benefits.

**Key Words：**QRcode；Android；JavaEE

目录

[第1章 绪论 1](#_Toc515374881)

[1.1 研究背景、目的和意义 1](#_Toc515374882)

[1.2 课题研究内容 1](#_Toc515374883)

[1.3 国内外研究现状 2](#_Toc515374884)

[第2章 基于移动终端登录系统的整体设计 4](#_Toc515374885)

[2.1 系统结构 4](#_Toc515374886)

[2.2 系统主要功能的实现 5](#_Toc515374887)

[2.2.1系统整体流程图 5](#_Toc515374888)

[2.2.2 登录信息的传递 5](#_Toc515374889)

[2.2.3 移动终端的登录 5](#_Toc515374890)

[2.2.4 登录状态的转变 6](#_Toc515374891)

[第3章 二维码技术分析 7](#_Toc515374892)

[3.1 QR二维码概述 7](#_Toc515374893)

[3.2 二维码开源框架——ZXing 8](#_Toc515374894)

[第4章 移动端的设计方案 10](#_Toc515374895)

[4.1 Android和IOS系统的比较 10](#_Toc515374896)

[4.2 Android系统体系结构 10](#_Toc515374897)

[4.2.1 应用程序层 11](#_Toc515374898)

[4.2.2 应用框架层 11](#_Toc515374899)

[4.2.3 系统运行库层 12](#_Toc515374900)

[4.2.4 linux核心框架 12](#_Toc515374901)

[4.3 移动端具体功能的实现 12](#_Toc515374902)

[4.3.1 登录界面 12](#_Toc515374903)

[4.3.2 注册界面 14](#_Toc515374904)

[4.3.3 扫码界面 15](#_Toc515374905)

[第5章 后台服务器的设计方案 18](#_Toc515374906)

[5.1 后台开发语言的选择 18](#_Toc515374907)

[5.2 J2EE的简介 18](#_Toc515374908)

[5.3 后台服务器的具体设计思路 19](#_Toc515374909)

[5.3.1 主页的设计 19](#_Toc515374910)

[5.3.2 登录成功后的页面设计 21](#_Toc515374911)

[5.3.3 响应Android客户端请求的Servlet 22](#_Toc515374912)

[第6章 结论 24](#_Toc515374913)

[参考文献 25](#_Toc515374914)

[致谢 26](#_Toc515374915)

第1章 绪论

1.1 研究背景、目的和意义

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，虽然将文字直接识别并跳转到该网页很难，但是将文本转化成二维码后，再去识别里面的内容就变得很简单了。可以说二维码是个很好的中间件，通过对其编码、解码能够无差别进行信息传递。

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。在复杂的网络环境中，用户自身账户的安全性是排在第一位的。尤其是在现如今的大数据时代，一个用户的账号不仅仅包含了用户的个人基本资料，更多的还关联了用户的生活习惯，以及各个其他软件平台上的用户数据。如果被别有用心的人掌握了用户的数据，通过全方位的信息检索，可以将用户的全部个人信息查出，包括用户的个人经历都能摸的一清二楚。这样的现象在网络中时有发生，如人肉事件。现在在网络上发生严重冲突，可能会导致一方去“人肉”搜索另一方，“人肉”搜索的方法就是去查看他人各个网络账号的信息，现代人或多或少都会用QQ、微信、微博等社交软件，搜索一方就会通过各种手段盗取账号并查看个人资料，并从中得出线索进而顺藤摸瓜找到更多隐藏信息。“人肉”搜索是很恐怖的事情，不仅对当事人造成威胁，甚至连身边的人也会牵连进去。如此可见信息安全的重要性，这也是本论文的研究目的，通过一种安全可靠的方法，从用户登录系统的角度去确保用户的登录信息不会发生泄露，从一定程度上保障用户信息安全。

1.2 课题研究内容

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，尤其是通过绑定手机号的登录方式最为可靠。只要不存在手机被人盗取或者是手机短信被人监控的情况，手机短信验证登录的方式几乎可以完全的确定当前进行登录的就是用户本人。除了登录验证外，许多重要的操作如修改密码、确定异常登录等都是通过短信验证完成的。这也是当前许多登录方式中最为重要的一种，但是该方法对于服务商来说存在一定的成本，项目设计者一般会综合考虑使用。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。正如条形码一样，条形码在上世纪七十年代开始得到普遍使用，在各种商品和产品包装上随处可见条形码的身影，其原因是按照条形码这种生成和识别方式可以快速的通过计算机查到产品的关键信息，方便在生产和出售过程中的使用。二维码得以广泛应用的原因也是如此，二维码的编码方式是对外开放的，需要使用到二维码的开发人员可以根据其编码协议设计出二维码编码和解码的算法，然后加以实现。为了方便也可以使用现有的二维码生成和识别的开源框架，或对其进行二次开发。

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，再由服务器对数据库进行查询和更改事务，最后由浏览器的轮询机制捕捉到用户登录状态的改变，并进行浏览器页面的跳转。

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。

1.3 国内外研究现状

本设计的内容为基于移动终端和二维码的系统登录技术。其中二维码是整个系统的关键要素，联系Web系统和移动终端的枢纽是session id，而session id为字符串，以二维码为载体，此节着重介绍二维码在国内外的研究现状。

上世纪80年代末，美国、日本等国家的企业、研究机构开始进行二维码的研究，1989年美国国际资料公司发明了Data Matrix，原名为Data Code；PDF417码为1991年留美华人王寅敬(音)博士发明并由讯宝(Symbol)公司制定完成；1992年美国知名的UPS(United Parcel Service)快递公司推出了UPS码，即Maxicode二维码的前身；1994年，日本Denso公司发明了QR（Quick Response Code）；2009年微软推出的一种新的二维码“Microsoft Tag”，增加了色彩维度，因此又称为彩色条码。二维码标准化的国外研究机构主要有国际自动识别制造商协会(AIMI)、美国标准化协会(ANSI)以及新成立的国际标准化组织/国际电工委员会第一联合委员会的条码自动识别技术委员会(ISO/IEC/JTCI/SC31)，其中AIMI与ANSI已经完成了PDF417、QR Code、Code One、Code16K、Code49等码制的符号标准，条码自动识别技术委员会(ISO/IEC/JTCI/SC31)已经制定了包括QR码的国际标准ISO/IEC 18004:2006、PDF417码的国际标准ISO/IEC 15438:2006、Data Matrix的国际标准ISO/IEC 16022:2006等二维码的国际标准，并且不断的完善。二维码一经发布就得到了世界发达国家的重点关注，这些国家对这种新一代的存储信息的技术非常看重，并在后来对其进一步研究，将二维码技术应用在安全领域，如通过二维码完成对多种证件信息的采集；物流领域，如通过二维码对商品的信息进行识别，减少商品在过运输检查关卡的时间。

我国对二维码的研究是从20世纪90年代初开始的，最初是由中国物品编码中心对几种常用的二维码PDF417、QR Code、Code One、Code16K、Code49的技术规范进行翻译和跟踪研究。2003年上海龙贝信息科技推出了龙贝二维矩阵，2005年中国编码中心完成了汉信码的研发，深圳矽感科技公司分别于2002年和2003年研发了具有自主知识产权的CM二维码和GM二维码。国家质量监督局制定了相关的二维码国家标准，二维码网格矩阵码（SJ/T 11349-2006）和二维码紧密矩阵码（SJ/T 11350-2006）。

第2章 基于移动终端登录系统的整体设计

2.1 系统结构

二维码扫描登录系统的整体结构如图2.1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[1]。

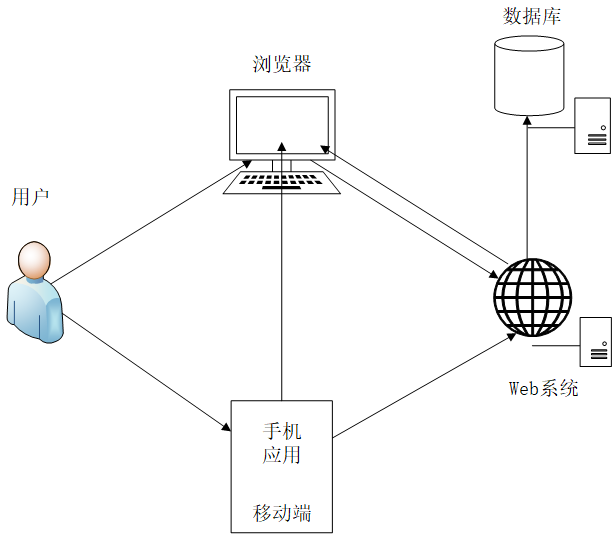


图2.1 基于移动终端登录系统的结构图

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。

2.2 系统主要功能的实现

2.2.1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2.2所示。

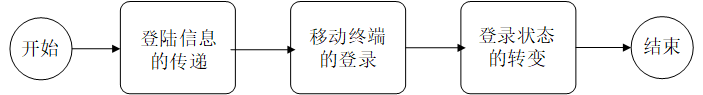


图2.2 方案整体流程图

2.2.2 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。

2.2.3 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。

2.2.4 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。

第3章 二维码技术分析

3.1 QR二维码概述

QR二维码又称快速响应矩阵码，是一种矩阵式二维码，它在二维方向上都存贮了信息[2]。QR二维码具有信息容量大、编码范围广、错误修正水平强、全方位识别、可加密等特点，这使得QR码在东亚以及欧美快速发展，直至目前QR码已经是二维码家族中最为流行的一员。QR码的基本结构如下图3.1所示：

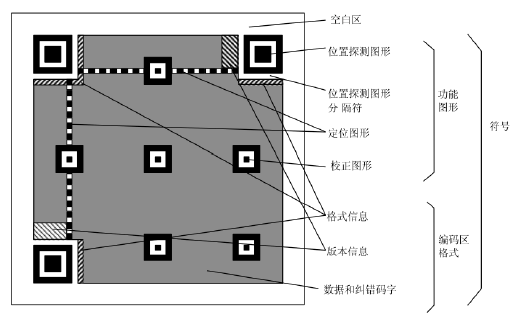


图3.1 QR码基本结构图

（1）位置探测图形

如图中所看到的，二维码图案在左上方、右上方以及左下方都绘制了一个相同的图案，这个图案呈“回”字形，该图案的作用是用来确定二维码的位置，思考可知确定一个矩形的三个角落即可确定该图案的准确位置[3]。“回”字图案中外围的黑色边框其边长为7个单位，比其小一点的白色边框边长为5个单位，位于正中心的黑色正方形其边长为3个单位。这三者组成的“回”字图案因为是用于确定二维码位置的关键图案，所以其具有唯一性，在整个二维码图形中除了这三个角落外是找不到这种图案的。

（2）位置探测图形分隔符

通过图中可以看到，“回”字形图案的外围还有一圈白色的边框，这个白色边框起到了分隔作用，将位置探测图形和其他真正编码的地方进行划分。

（3）定位图形

图中有两条黑白相间组成的窄带，形成了一个类似坐标一样的横轴和纵轴。其作用就是坐标轴的作用，通过这一参考系来确定二维码图案中每个像素点的准确位置。

（4）校正图形

校正图形可看成由5x5大小的黑色模块、3x3白色模块及中间的一单位宽黑色模块组成的重叠同心正方形构成，数量随版本变化，版本越高数量越多。校正图形主要作用是对不规则的图形失真进行校正。

（5）编码区域

由于编码的过程中，存在编码格式以及编码版本等信息，此外还要对其进行编码纠错加入的纠错码字，真正编码的信息中会加入这些辅助信息，所以编码区域还由这多个部分组成，其中编码版本的选择将确定编码出来的数据大小。

（6）格式信息

格式信息的编码由两部分组成：第一部分为数据位，共有5位；第二部分为纠错位，共有10位。这两部分共同组成15位的格式信息编码。

（7）版本信息

版本信息涉及到编码出来的数据大小，是非常重要的数据，所以会出现在二维码的两个地方，以此提高冗余度。版本信息同样由数据位和纠错为两部分组成，各占6位和12位[4]。

（8）空白区域

顾名思义，整个二维码最外围的白色图案就为空白区域，其宽度有4个单位。

3.2 二维码开源框架——ZXing

ZXing是当前比较流行的一个关于一维码和二维码编码与解码的的开源框架，除了一些公司为了满足自己的产品需求会自己开发一个二维码框架外，大部分与二维码开发相关的程序都会使用ZXing框架。ZXing由谷歌公司发布，其内部的代码实现是JAVA语言，但是同时也提供了一些其他编程语言调用的接口，除此之外，针对解码操作，ZXing库中有一Android库提供了一个功能齐全的二维码生成和识别的例子。通过对该Android实例进行裁剪或修改后调用其主界面，可以调出手机相机扫描的界面并实现相机的扫码识别。使用ZXing框架时，需要导入其核心包，但是手动导包往往会出现缺少依赖包的问题，建议通过Maven来管理项目，通过配置依赖关系让Maven自动地到远程库下载。下载解压后得到ZXing的源码结构如图3.2所示：



图3.2 ZXing源码目录

在ZXing源码的目录中有一个Android目录，该目录是一个由Maven构建的完整Android项目。该项目是Android端开发的关键要素，Android项目中扫描二维码的业务将交由该ZXing的Android项目来处理。本设计中的Android项目只需要使用其回调的数据即可[5]。ZXing的Android项目的java逻辑代码中，分了许多文件夹，有camera、encode、history等，还有根目录下的各个Activity代码。其中的CaptureActivity文件和camera、decode目录是本设计中需要使用到的，CaptureActivity是扫码的界面，会打开手机的相机并在相机界面上绘制扫描的动画，使扫码过程更加人性化。关于手机相机的初始化和调用的代码都放在了camera目录中，后台会将相机中的画面进行逐帧的识别，并进行灰度化、二值处理、方向矫正等操作，最后将图片信息转化为数据。得到数据后decode目录下的相关类会对数据进行处理并提取出二维码的有关信息和内容。

第4章 移动端的设计方案

4.1 Android和IOS系统的比较

现如今主流的移动终端的操作系统有Android和IOS两种，发展到现在已经对于哪个系统更加优秀已经有很大的争议了，各自有自己的长处和优势。可以说IOS开创了智能手机的先河，由于其当时非同凡响的操作体验，和多种多样炫酷实用的应用，将传统手机市场打破。而当时在手机行业称霸的诺基亚公司现如今已经被微软公司收购，摩托罗拉公司也被谷歌公司收购了。IOS打开了智能手机的大门，而Android系统则不断的推进智能手机的发展，使其壮大。由于Android系统的开源特性，各个手机厂商都能够进行Android手机的开发，这样一来整个智能移动终端的行业也就得到了飞速发展。

在IOS系统发布的初期，其功能是不够完善的，甚至是许多在传统手机上很容易实现的功能，IOS也无法完成。如IOS在早期4.0的版本是做不到更换手机壁纸的，而这样的功能对于传统手机来说是非常容易实现的。除此之外，IOS系统还缺少查看手机来电归属地的功能，只能通过手机越狱来获得，而这样无疑会给手机带来安全性的问题。除了这些缺点，IOS的优点是非常明显的，用过IOS系统的人都知道，IOS最明显的一个有点就是流畅，IOS的流畅是因为其对用户交互处理的顺序不一样，IOS系统的构架会最优先地处理用户的交互操作，再后面才是媒体、服务以及核心架构的运行，这样的设计带给用户非常好的交互体验。除此之外，IOS系统还带来了全新的设计风格，带给用户全新的视觉体验，传统的UI设计都是依照尽量模仿实际物体设计的，而IOS采用了扁平化的设计风格，达到了简单、精美的效果。

Android系统在早期被人们认为是模仿着IOS系统来做的，这也很难否认，毕竟Android系统在早期确实许多功能和设计模式都跟IOS系统大致相同。但是经过不断的发展，现如今，Android系统已经独树一帜，有许多自己创新的技术。Android系统的发展是非常迅速的，这一直接原因就是其开源特性。Android的源代码是开放的，任何人都可以去Github上查看，并对其进行定制，加入全新的组件和布局。因此许多手机厂商都推出了自家的Android系统。虽然开源的特性让Android系统多样化，但是这是把双刃剑，开源的特性也使得各个手机厂商设计出来的手机没有统一的规格，直接导致了各种应用的兼容性问题。如屏幕尺寸不同导致应用中的图片显示存在差异，从而影响外观效果。而且手机硬件配置的不同也使得Android应用开发者在进行应用开发时要考虑那些低端配置手机是否能够流畅运行自己的应用。这也在一定程度上抑制了其发展。

4.2 Android系统体系结构

和Linux、IOS等操作系统一样，Android系统在设计时也将系统分为了几层架构。如图4.1Android系统的架构图所示，可以将Android系统分为四个层次，其中，最高层为应用程序层，其次是框架层，再其次是系统运行库层，最后就是底层的linux核心层[6]。

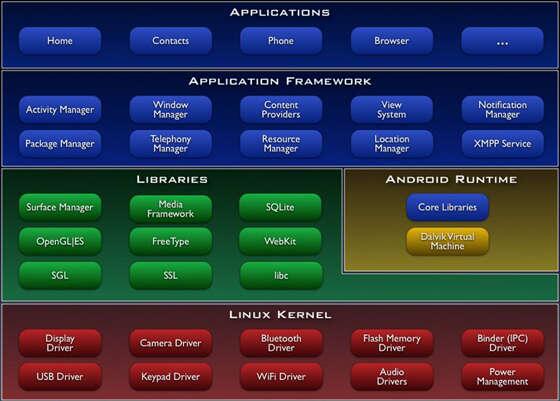


图4.1 Android系统架构图

4.2.1 应用程序层

这一层是直接和用户打交道的架构层，其包含了手机的基本功能的应用，即手机必须安装好的应用，如电话、短信等系统应用，为用户提供最基础的服务。除此之外还使得用户能够安装第三方的应用软件，对Android系统的功能进行扩展，丰富用户体验。这些应用程序早期都是由Java语言编写的，现在越来越多的开发者选用Kotlin语言来进行更快捷安全的开发。

4.2.2 应用框架层

该层提供了Android开发所需要的组件,开发者根据应用程序的需要可以随时调用其API达到快速开发的效果，并且这种架构使得Android开发可以进行划分。一个应用程序就是由开发者对其一个或多个框架进行开发得到的。其中Activity Manager负责管理Activity的生命周期，而Activity可以看做是Android应用的一个界面。View System为Android开发提供了各种控件，和开发windows程序一样，Android开发也是需要大量的控件来完成一个界面的布局，

这些控件最基本的有TextView、Button、ImageView等，其效果分别为文本框、按钮、图片框等。使得Android界面可以显示图片文字等资源，以及响应用户的点击事件，高级的控件还有如TabLayout、ListView等，可以实现选项卡、列表等高级布局，实现更人性化的操作。此外，Notification Manager框架使得应用程序能够向手机的通知栏添加自身的提示信息，Content Providers框架使得某个应用程序能够查询或使用其他应用程序的数据。

4.2.3 系统运行库层

（1）左边绿色部分为Android底层的C/C++库，封装了系统对底层资源的操作。如OpenGL|ES库能够完成对三维空间的相关计算和处理，这个功能在Android游戏开发中用的多。SQLite是一个轻型关系数据库，能够快速进行数据库的创建以及更删改查操作，这个库使得Android能够方便的进行数据存储。Medie Framework库使Android系统能够识别出大多数市面上的媒体格式的文件，从而使得Android能够播放视频查看图片。WebKit库可以让Android系统具有解析网页的功能，以及兼容各种浏览器插件的功能，并能够进行web浏览的视图资源。

（2）右边部分主要为Android系统运行在linux系统上提供了可能，linux是一个操作系统，而Android也是一个操作系统。如果要在一个系统上运行另一个系统，这在windows操作系统上的解决方案为在windows操作系统上装上如vmware之类的虚拟机[7]。所以linux操作系统上需要这样一个虚拟机软件来运行android操作系统，所以就有了Dalvik虚拟机。并将Android系统上的每一个应用都交给单独的一个虚拟机运行。

4.2.4 linux核心框架

Android系统是运行在linux 2.6内核上的，通过linux的内核来进行真正底层的操作，linux框架中有很多底层的驱动来实现网络访问、存储控制等操作，根据实际需求还会对其进行裁剪移植，在其他环境中运行[8]。

4.3 移动端具体功能的实现

根据第二章中系统结构和验证过程进行分析，此Android应用需要三个界面：用户登录界面、注册界面、个人信息及二维码扫描界面。因此创建了三个对应的Activity以及对应的layout布局：MainActivity、RegisterActivity、LoginSuccessActivity。

4.3.1 登录界面

登录界面的布局如图4.2所示。界面采用线性布局，两个EditView供用户填写用户名和密码，两个按钮监听用户点击事件。



图4.2 Android登录界面布局

在进行Android开发时，需要有一个Activity的子类，和对应的xml布局文件来完成某个界面的功能。而Activity有其自己的生命周期，在打开应用时，系统会找到该应用的MainActivity作为程序的入口，MainActivity被创建后会进行到onCreate方法中，在onCreate方法中执行布局文件的关联，以及对布局文件中的资源进行初始化。Android提供了findViewById这一方法使布局文件中的控件和java引用对象相绑定。在登录界面中，还需要给登录和注册两个按钮设置点击事件，调用button对象的setOnClickListener方法可以为其设置一个监听事件，该方法需要传入一个View.OnClickListener的实现类，并重写其中的onClick方法来响应该点击事件。

在登录界面中，登录按钮被点击后，需要判断用户名和密码两个EditView中的值是否为空。如果任意一个为空则通过Toast的形式提示用户输入账号和密码。若不为空则进行登录操作。实现登录操作需要Android网络编程相关的知识。本Android应用通过http协议与Web服务端进行通信，再由Web服务器操作数据库并返回查询结果给Android应用，整个过程Android应用不直接操作远程数据库，从而减小了Android应用的资源开销。Android中进行http通讯可以使用HttpURLConnection类，在实现登录逻辑时，需要获得账户名和密码中的内容，并将其添加到GET请求中，所以需要创建一个URL类，在进行实例化时需要传入一个网址作为参数。该网址即是Web系统的网址，在网址后面加上“?account=用户名&password=密码”就将用户名和密码封装到了GET请求中，通过URL的openConnection方法可以获得一个HttpURLConnection的实例，通过HttpURLConnection的setRequestMethod(“GET”)方法可以将此次请求设置为GET方式。将参数设置好后，调用HttpURLConnection的getInputStream方法能够获得Web系统返回数据的字节输入流，并将请求发送给Web服务器。移动端发送请求后，就等待接收Web系统返回的数据，所以Web端和移动端需要协商返回的数据格式。本设计中，Web端在接收到Android应用的登录请求后，会返回一个结果码以及用户信息，两者用“&”连接，其中登录成功会返回字符串“100”，登录失败会返回“110”，若登录失败则用户信息为空。

为了方便数据的传输和解析，本设计将传输的信息封装为Json格式，通过Gson框架来进行实际操作。Json是一种轻量级的数据交换格式，它不是一种程序语言，只是一种数据的约定，表现为字符串格式，在Json的语法中，对象以键值对的形式出现，数据之间通过逗号来进行分割，一对花括号表示一个对象，方括号中存放的是数组。Gosn为谷歌公司出品的一个json解析工具，可以将json和POJO类互相转换，将字符串变为Model更有利于代码的重构。本设计中将返回码ResCode和用户信息User都封装成类，ResCode类中只有一个属性即返回码，而User类包含了account,password,nickname,age等属性，除了属性外还要添加各个属性的Getter和Setter方法，使其成为一个POJO类。在接收到Web系统返回的json字符串后只需转化成相应的类，然后调用其get方法就能够获得需要的数据。

首先将解析出来的返回码resCode进行比对，如果是登录失败，则通过Toast的形式打印出登录失败的字样。如果登录成功，则需要实例化一个意图类，来完成页面的跳转，同时还需要将用户信息添加到意图类对象中传到下一个页面中。至此Android应用的登录功能已经完成。

4.3.2 注册界面

当用户还没注册账号在登录界面点击注册按钮后，就跳转到RegisterActivity界面中，注册界面的布局如图4.3所示。布局中有输入用户名、密码和个人信息等编辑框，以及两个按钮[9]。布局文件为activity\_register.xml，Java逻辑为RegisterActivity.java。



图4.3 Android应用注册界面

在注册界面中初始化视图资源，为按钮添加监听事件后，当用户点击注册按钮后，先判断信息是否填写完全，如果编辑框中有任何一栏没填写信息，则通过Toast通知用户将信息填写完整，如果信息填写完整，则检查两次输入的密码是否一致，如果不一致则通知用户重新填写。当以上都正确时，将用户信息封装成json字符串，通过POST请求将json字符串发送到Web系统中，具体方法是通过HttpURLConnection的对象获得其字节输出流，并将其包装成DataOutputStream对象，通过writeBytes()方法将json字符串发送出去，当然此时还只是存在缓冲区中，当调用HttpURLConnection的getInputStream方法后，数据就真正发出去了，并等待服务器返回执行结果。若返回结果为“210”即静态类Constant中的Constant.REGISTER\_FAIL常量，则表示注册失败，原因可能是Web端不能正常对数据库进行操作；若返回结果为“200”即Constant.EXIST\_THE\_ACCOUNT常量，则表示用户名已被注册过；若返回结果为“300”即Constant.REGISTER\_SUCCESS常量，则表示注册成功，并弹出对话框提示用户已经注册成功，用户点击确定按钮后调用Activity的finish()方法结束当前Activity返回到登录界面中。

4.3.3 扫码界面

当在登录界面进行登录并成功后，会跳转到扫码界面，扫码界面的布局如图4.4所示，该布局中有一个文本框用来显示用户的信息，也确保用户登录信息的获取是完整的。除了显示用户信息的文本框，还有两个按钮，一个用于调出实际用于扫描二维码的界面，另外一个用来退出当前登录。扫码界面的布局文件为activity\_login\_success.xml，java逻辑文件为LoginSuccessActivity.java。



图4.4 Android应用扫描界面

在LoginSuccessActivity中，调用Activity的getIntent()方法可以得到在登录界面传来的用户信息的数据。由于封装数据用intent.putExtra(String name, String value)方法存储的键值对，所以在取值的时候需要调用intent.getStringExtra(String name)，通过传入键名来获得对应的值，获取到账户名、昵称、年龄等用户信息后再文本框中显示出来。当用户点击扫码按钮后，通过显性意图的方式来启动一个Activity，与上文中的startActivity方式不同，这里需要调用intent的startActivityForResult方法可以获得下一个页面回传过来的数据。当下一个Activity调用finish方法来结束自身的时候，会执行onActivityResult方法将二维码扫描的结果通过Intent对象回传给上一个Activity。当LoginSuccessActivity得到正确的二维码扫描结果后，要首先对其内容进行判断，看是否是Web系统产生的二维码。在框架设计中提到过，Web系统产生的二维码是由Web系统的网址和session id通过“，”组成的。所以需要调用String方法的split(String regex)方法来进行字符串分割，这里将“，”作为regex的实参，讲二维码结果分割成两部分result[0]和result[1][10]。首先比较result[0]中的内容是否是Web系统的网址，如果不是，则将二维码的内容通过Toast打印输出，若是，则result[1]中存放的就是session id，此时需要将当前用户的用户名和session id通过GET请求的方式发送到Web系统进行处理。当Web系统将session id保存到数据库中该用户名的一条记录中后，整个Android端的任务就完成了。

第5章 后台服务器的设计方案

5.1 后台开发语言的选择

进行Web开发前需要了解集中Web后台的开发语言，现在比较热门的后台开发语言有PHP、RUBY以及JAVA等。RUBY语言在网上的评价很高，其语言优雅、开发便捷，但并没有成为主流的后台开发语言。现在比较主流的后台开发语言还是PHP和JAVA，而PHP和JAVA在设计理念上存在着很大的差异，这导致两种语言在进行开发时会得到不同的结果[11]。JAVA是一个非常完善的程序设计语言，它在访问数据库以及进行数学计算方面有先天的优势，JAVA WEB的开发有自己的一套流程，在开发时会用到SSH框架，SSH框架指的是Struts框架、Spring框架、Hibernate框架。其中Struts框架用来管理项目用以实现MVC的设计模式，将视图、逻辑实现分离，降低代码耦合度。Spring框架用于业务逻辑层。Hibernate框架能够使得开发者方便快捷的进行数据持久化操作，即将内存中的数据存储到数据库中[12]。反观PHP语言，其设计理念为面向过程的，并以客户作为开发的核心，能够进行快速的进行Web开发，且现在市面上的许多虚拟主机都是以支持PHP为主，这就使得PHP的开发要容易的多。但是PHP在进行大型的网站开发时，面对极其复杂的业务逻辑，非常容易出错。所以在进行大的项目开发时还是会使用到JAVA语言。由于Android开发中也使用的JAVA语言，所以本设计也将采用JAVA语言进行Web后台的设计。

5.2 J2EE的简介

J2EE是JAVA的企业版开发平台，也可以称之为JAVAEE，J2EE主要是用于开发Web网页，J2EE为网页开发者提供了一系列的工具和组件，如Servlet、JSP等。这两者也将在本设计中重复用到，通过JSP来实现网页的显示和布局，并能够在其中加入java代码扩展其功能。Servlet用来处理客户端的Http请求，是一种服务器组件，通过Servlet可以进行动态网页的开发，提高网页的交互能力。

J2EE技术一般分为四个层级：客户层、WEB表示层、业务逻辑层、数据层（EIS）。如下图5.1所示。其中客户层完成了与用户的交互，视图以及交互结果需要在客户层中呈现出来。Web层中包含了上个段落中提到的JSP、Servlet等组件，其功能就是处理用户的交互，是一个中间件。数据层用来实现数据持久化，即将需要的数据通过数据库的方式保存下来，以及供其他组件来访问。

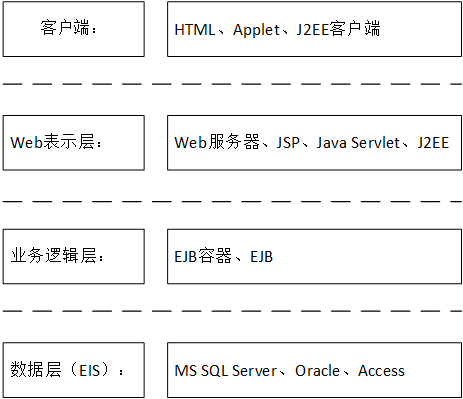


图5.1 J2EE层次结构图

5.3 后台服务器的具体设计思路

本Web后台开发用的是jsp+servlet的模式。由于本设计中对网页布局要求不高，所以不需要用到CSS对界面进行修饰，只需要在JSP页面中加入JavaScript脚本就能完成网页部分的功能。Servlet是Web后台设计的核心内容，Servlet不仅要处理前台的请求，还需要处理来着Android端的请求[13]。

网页在早期是以静态的方式呈现给用户的，只需要将设计好的页面放到Web服务器中，浏览器就能通过Http协议来访问Web服务器中的资源。随着网络技术的发展以及用户对网页有了更多的需求，开发者就开始对Web服务器和浏览器的功能进行扩展，实现浏览器与Web服务器之间的交互功能。实现这一功能就用到了servlet，servlet可以看作是一个用java实现的Web组件，通过servlet就可以完成交互功能。Servlet由servlet容器管理，本设计中用到的Tomcat服务器就是一个servlet容器，当用户打开浏览器访问Web服务器后，Web服务器会接收到来着客户端的Http请求，Web服务器会将该请求发送给servlet容器，servlet容器会加载相应的servlet，产生servlet的实例，将请求和响应的对象传给这个实例[14]。Servlet对请求进行处理后，将结果返回给客户端。在本设计中进行servlet程序的编写时，只需要对servlet的doGet和doPost两个方法进行重写即可。

5.3.1 主页的设计

主页的页面效果如图5.2所示，需要在网页上显示一个二维码图片供用户扫描。所以需要在html页面中加入一个img标签，将img标签的src属性设置为Web系统目录下的HomeServlet来处理，如上面所说，Web系统会将请求传给Servlet容器，然后Servlet容器再将HomeServlet实例化，并处理请求。由于src是由客户端即浏览器发送的请求，所以HomeServlet可以通过request对象获得浏览器的session id，这样生成二维码的工作就交由Web端来完成了。在HomeServlet中，由于生成二维码是由QRCodeWriter对象的encode(String contents, BarcodeFormat format, int width, int height, Map<EncodeHintType, ?> hints)来完成的，所以需要先对该方法的各个形参进行实现[15]。

Contents参数指的是需要进行编码的内容，在本设计中，二维码的内容为Web系统所在的网址和session id，网址由系统所在的IP地址确定，可能会经常发生变化，为了防止将程序写死不便修改，网址有常量Constant.SERVICE\_URL来代替，如果网址发生变化，只需要到Constant类里面做修改即可。Session id可以由request.getSession().getId()来获得，返回的结果为字符串。将两者用“，”进行连接组成二维码的内容。

BarCodeFormat format参数表示二维码编码的格式，BarCodeFormat是一个枚举类，由于本设计使用QRcode编码，所以传入BarCodeFormat.QR\_CODE即可。

Width,height为生成的二维码的宽和高，这里都设置为250。

Hints参数为Map集合，里面的每对键值对都是对二维码编码的一个配置，或者说是限定，如EncodeHintType.CHARACTER\_SET规定了二维码内容的字符编码方式，这里设置为”utf-8”编码格式。EncodeHintType.ERROR\_CORRECTION规定了二维码编码的错误率，这里选择一个中等的错误率，设置为ErrorCorrectionLevel.M，约为15%。错误率并不是越低越好，越低的错误率在进行编码和解码时会消耗更多的时间，进而影响效率。

当所有参数都确定后，encode方法会编码返回一个BitMatrix类型对象，BitMatrix可以理解为一个位的二位矩阵，其中x代表列，y代表行，原点位于左上角，位的内部由32位整数的一维数组表示，这就是二维码的数据形态，其中BitMatrix与BitArray有良好的兼容性。

ZXing为java提供了一个额外的jar包，在java代码中使用该jar包中的方法可以将BitMatrix保存为图片文件或者通过IO流的形式输出。这里可以通过response对象获得其输出流，将二维码以流的形式返回给主页，这样主页上就会显示出二维码。



图5.2 Web系统的主页页面

Web系统的登录主页除了需要显示二维码之外，还要通过Ajax轮询的方式 查看用户的登录状态是否改变，实现页面跳转的功能。Ajax的意思是异步的 JavaScript 和 XML。通过这种方式可以实现异步操作，不需要对整个页面进行重新加载即可实现局部内容的更新。实现Ajax最重要的一点就是在javascrpt中使用XMLHttpRequest和后台进行数据交流。在本设计中，需要通过XMLHttpRequest不断的对Web系统的QueryServlet发送GET请求。若从QueryServlet中返回内容为“1”，则表示登录成功，页面跳转到登录成功的页面。

在QueryServlet中重写其doGet方法，获取到本次会话的session id，通过JDBC来进行数据库业务，使用JDBC需要先通过java反射机制，即Class类的相关方法来加载MYSQL的JDBC包，再通过DriverManager类的相关方法来获得一个connection对象，最后通过connection来开启一个事务，在QueryServlet中需要执行数据库的查询操作，查询语句为"select islogined from " + DBUtil.TABLE\_USER + " where sessionid='" + sessionID + "'"，意思是通过session id来查询该session id对应的islogined值。并将其从int类型转换为string类型通过response的字符输出流输出。

5.3.2 登录成功后的页面设计

登录成功后会从登录页面跳转到提示登录成功的页面，该页面如图5.3所示。在该页面会显示出登录成功的字样，并将用户的昵称显示出来，还有一个按钮用来注销登录。

用户昵称可以用<a id=”nickname”></a>标签来表示，在javascript里面进行一个XMLHttpRequest的GET请求发送到Web系统的GetNicknameServlet处理。GetNicknameServlet会根据传入的request对象获得此次会话的session id。然后通过数据库的查询语句根据session id来找到用户获取其nickname，再将nickname通过字符输出流传回给XMLHttpRequest.responseText。通过javascript的内置对象document的getElementById("nickname")方法来得到上述的a标签。a.innerHTML = xhr.responseText将a标签的内容赋值为nickname即可。

注销按钮<input type="button" value="注销登录" onclick="logout()"/>，设置其onclick属性为logout()函数，然后再在javascript脚本中写一个logout()方法，在该方法中同样会通过XMLHttpRequest来给Web系统的LogoutServlet发送GET请求，LogoutServle会根据request对象获得session id，然后执行数据库的更改语句，"update " + DBUtil.TABLE\_USER + " set islogined=0, sessionid='" + null + "' where sessionid='" + sessionID + "'"，根据session id找到当前用户的记录，并更改这条记录的islogined字段字段为0，sessionid字段为null。执行成功就通过response返回字符串”success”。XMLHttpRequest在接收到字符串”success”后，会调用alert("注销成功")提示用户注销成功，并通过window内置对象将网页重定向为登录页面。

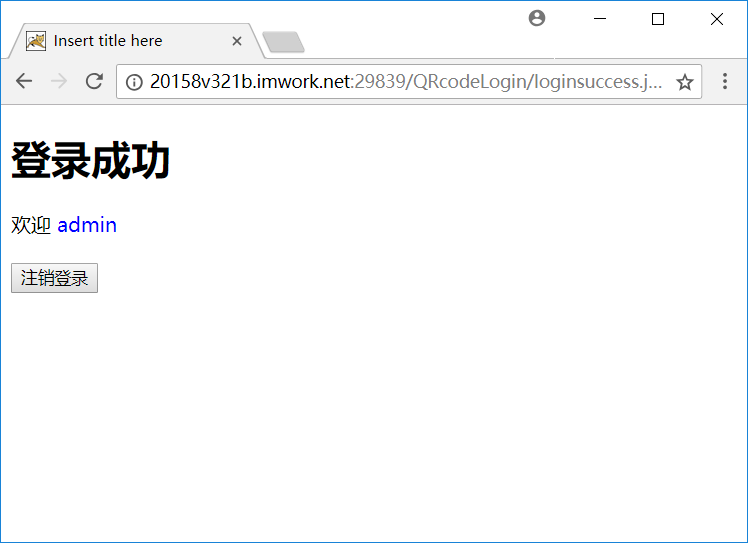


图5.3 登录成功后的页面

5.3.3 响应Android客户端请求的Servlet

（1）LoginServlet

在Android客户端进行登录操作时，会向该Servlet发送一个GET请求，在LoginServlet的doGet(HttpServletRequest request, HttpServletResponse response)方法中，可以获得此次请求的request对象和response对象，通过request对象的getParameter方法能够得到GET请求传过来的账户名和密码，然后执行查询语句"select account,nickname,age from " + DBUtil.TABLE\_USER + " where account='" + account + "' and password='" + password + "'"，该条查询语句会根据用户名和密码的判定条件进行查询，执行查询语句返回的是一个结果集result，通过result.next()方法一定指针到下一条结果记录，如果没有next方法返回为false，则说明根据传进来的账号和密码没有查到该用户，即账号或密码是错误的，将resCode设置为Constant.LOGIN\_FAIL；如果next方法返回结果为true，则说明通过账号和密码成功找到了该用户，通过result.getString()可以获得结果中的昵称和年龄，将resCode设置为Constant.LOGIN\_FAIL。得到结果码和用户信息后，通过gson工具将这些信息封装成对象并转为json格式，最后通过response的字符输出流将json字符串输出。

（2）RegisterServlet

Android端应用在进行注册时，会将用户信息，即用户名、密码、昵称以及年龄等，通过POST请求发送给Web系统的RegisterServlet处理。RegisterServlet得到账号后先执行数据库查询操作，如果查询有结果，则这个账号已经被注册过，将resCode设置为Constant.EXIST\_THE\_ACCOUNT。如果账号没被注册，则向数据库中添加该条用户信息的记录。ExecuteUpdate函数会返回影响的记录数，如果大于0，则添加用户成功，将resCode设置为Constant.REGISTER\_SUCCESS，如果小于等于0，则添加记录失败，将resCode设置为Constant.REGISTER\_FAIL。

（3）LoginStateChangeServlet

当用户在Android端扫描二维码并解析出二维码中的信息后，会向Web端的LoginStateChangeServlet发送包含用户名和session id的GET请求，LoginStateChangeServlet通过request对象取得这两个信息后，通过用户名找到该条记录，并将islogined字段设置为1，将sessionid字段设置为传入的session id。

第6章 结论

本文提出了二维码扫描登录的方案，为传统登录方式进行了补充，使用户的登录更加快捷、安全。整个登录过程的关键要素为session id，扫码和解码的工作都是围绕着session id展开的，若有多个用户同时进行登录，后台需要确保指定的浏览器进行页面跳转，这时session id就起到了作用，这也是本设计的重点和难点。除此之外，在开发的过程中，Android端和Web端需要进行协同开发，Android端的数据都是通过http协议向Web端发出的，需要做好两端数据格式的协商。

本设计虽完成了二维码扫描登录的核心功能，但是在实现方法上还是有所欠缺，如浏览器的页面跳转实现是通过Ajax轮询通过Web系统监听数据库中用户的登录状态。轮询需要在浏览器每隔一个周期执行一次XMLHttpRequest的GET请求，是非常消耗资源的一种做法。如果能够改为用WebSocket实现浏览器和Web系统的双向通信，当用户状态发生跳转，Web系统能够向指定浏览器发送页面跳转的指令，这样的设计可以极大的减小资源开销。除此之外，在软件的设计上没有采用MVC、MVP、MVVP等设计模式，代码存在很大的冗余度以及代码耦合，在一处代码有改动时，需要改动多处其他代码。

最后，希望本设计能够给其他想要加入二维码扫描登录功能的程序开发者提供思路，共同打造网络环境的安全，保障用户信息的安全。

参考文献

[1]江超,龙毅宏.基于移动终端的系统登录方法的研究与开发[J].软件,2017,38(03):12-15.

[2]LIN, ZHI-WEI. Two-dimensional code recognition method of object effectively improves shortcoming of difficulty in decoding[P]. ：TW201729143,2017-08-16.

[3]AIISO MASASHI. MARKER, TWO-DIMENSIONAL CODE, RECOGNITION METHOD FOR MARKER, AND RECOGNITION METHOD FOR TWO-DIMENSIONAL CODE[P]. ：JP2013084031,2013-05-09.

[4]SATO, KOTARO. Two-dimensional code generation method, two-dimensional code, two-dimensional code recognition method, and image recognition dev[P]. ：EP2246805,2013-09-11.

[5]陈君. 二维码技术在移动终端的安全应用研究[D].广东工业大学,2015.

[6]祝忠方. 基于Android的移动互联终端的设计和实现[D].北方工业大学,2014.

[7]陈春林. 基于Android移动终端的二维码识别技术实现与应用研究[D].广东工业大学,2015.

[8]张晶晶.二维码识别系统在手机端的研究与实现[J].化学工程与装备,2013(02):164-166.

[9]孙旭明. 智能移动个性化信息推荐系统的设计与实现[D].南京邮电大学,2014.

[10]钟京馗.JAVA中的正则表达式及其应用[J].电脑编程技巧与维护,2005(06):50-57+69.

[11]杜闯.PHP在动态网站开发中的优势[J].电脑知识与技术,2010,6(13):3342-3344.

[12]于湛麟,李仲秋,任永昌.SSH框架实现MVC架构的电子商务软件平台[J].计算机技术与发展,2012,22(10):169-172.

[13]裴生雷.Ajax+JSP开发模式的研究及应用[J].计算机技术与发展,2013,23(01):242-245+248.

[14]刘晓峥.浅析基于JSP技术的JavaBean和Servlet[J].科技视界,2015(34):241+269.

[15]方诗虹,张一鸣.基于Zxing下移动端与PC端交互的新型购物系统开发[J].西南民族大学学报(自然科学版),2016,42(06):677-681.

致谢

不知不觉中论文和设计已经都完成了，在开始做设计的初期，我对于这个设计如何实现还是一无所知的，同时也遇到了很多困难和险阻。但是通过龙毅宏老师的悉心指导下，我查看了大量的文献，克服了重重阻碍。龙毅宏老师帮助我从论文选题到完成设计一直都给予了很大的帮助，每一步都倾注了老师的心血，在这里我要向龙毅宏老师致以诚挚的谢意和崇高的敬意。同时还要感谢其他同学和朋友的关心与帮助！在做设计的这段时间中，我学到了很多网页开发和移动开发的知识，逐步的学习让我慢慢掌握了自己的设计，打开了我的思路，同时还要感谢论坛上积极回答我问题的前辈，让我结交了朋友。