## 网站前台和后台

网站前台是面向网站访问用户的，通俗的说是给访问网站的人看的内容和页面；网站后台，有时也称为网站管理后台，是指用于管理网站前台内容的，是面向网站管理人员的。前台和后台都是程序人员做的开发的网站页面，通常开发带网站管理后台功能的网站空间必须支持程序语言和数据开发功能。拿一个新闻网站来说，网站前台访问可以浏览公开发布的内容，如新闻信息，网站管理员可以通过用户名和密码进到网站的后台，来发布新闻、编辑用户评论等操作。网站后台可以有效的管理网站供浏览者查阅的信息，通常需要帐号及密码等信息的登陆验证，登陆信息正确则验证而后进入网站后台的管理界面进行相关的一系列操作。

## 权限管理系统定义

**权限管理是一个几乎所有后台系统的都会涉及的一个重要组成部分**，主要目的是对整个后台管理系统进行权限的控制，而针对的对象是员工，避免因权限控制缺失或操作不当引发的风险问题，如操作错误，数据泄露等问题。其实权限管理的设计并不难(非常简单)，就目前来说最广泛的是一个用户对应多个角色，每个角色对应相应的权限集（**RBAC模型Role Based Access Control**）这种模型基本可以应对所有的问题，且通过角色可以实现灵活且多样的的权限控制需求，我们梳理一下上面主要提到的几个比较重要的名词：**用户、角色、权限**。

## 用户的定义

每个员工想要进入后台管理系统肯定都会有一个用户，而这个用户就是一把钥匙。我们通过控制用户所具备的权限，进而控制这个员工的操作范围。因此需要告诫员工，用户名密码不能轻易提供他人，不然遇到的问题由自己承担。后台管理系统的账号，通常是由公司内部人员进行创建。(企业OA办公系统)

## 角色的定义

角色是搭建在账号与权限之间的一道桥梁。系统中会有很多权限，如果每建一个账号都要配置一遍权限，将是很繁琐的一项工作。角色这一概念的引入，**极大地增加了权限管理系统配置的灵活性和便捷性**。创建账号时，可以将不同的角色配置在同一个账号上，也可以给不同的账号配置相同的角色。创建角色时，可以根据角色的差异赋予其不同的权限。

角色的命名最好按照职位而定，例如市场部普通员工，市场部主管等。因为职位在任何企业都是存在的，且是有限的，并且容易理解，市场部文员那就是市场部文员角色，方便我们配置权限时的判断，避免配置错误。

## 权限的定义

**权限可以分为三种：页面权限，操作(按钮)权限，数据权限。**

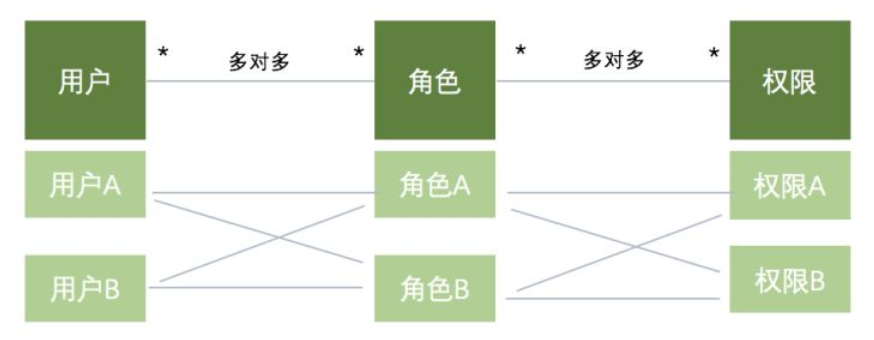
* **页面权限：**控制你可以看到哪个页面，看不到哪个页面，很多系统都只做到了控制页面这一层级，它实现起来比较简单，一些系统会这样设计，但是这种控制相对操作权限来说比较粗放，难以对权限进行精细管理。。
* **操作(按钮级别)权限：**控制你可以在页面上操作哪些按钮（当我们进入一个页面，我们的目的无非是在这个页面上进行增删改查，那在页面上对应的操作可能是：查询，删除，编辑，新增四个按钮）可能你在某个页面上，只能查询数据，而不能修改数据。
* **数据权限：**控制你可以看到哪些数据，就是用户在同一页面看到的数据是不同的,比如财务部只能看到其部门下的用户数据,采购部只看采购部的数据,在一些大型的公司,全国有很多城市和分公司,比如杭州用户登录系统只能看到杭州的数据,上海用户只能看到上海的数据（数据的控制我们一般通过部门去实现，每条记录都有一个创建人，而每一个创建人都属于某个部门，因此部门分的越细，数据的控制层级也就越精细）。

## RBAC模型

RBAC（Role-Based Access Control）意思是基于角色的权限控制，有别于传统模型中的直接把权限赋予账号，增加了“角色”的概念，把权限赋予角色，再将角色赋予账号。提高了账号管理效率，降低了出错的概率。**用户与角色是多对多关系，角色与权限是多对多关系。**

例如：当有多个账号需要配置相同的权限时，有了角色后便不需要给每个账号挨个配置权限，只需要在角色上配置权限，再把角色配置到账号上。

|  |
| --- |
| **RBAC模型解读** |
| 根据RBAC的复杂度不同，可分为RBAC0，RBAC1，RBAC2，RBAC3.最常用的为RBAC0.  RBAC0模型就是将一个或多个权限挂到角色下，在将一个或多个角色赋予用户，权限与角色的关系，角色与用户的关系，均是多对多的关系。 |



## 权限管理的设计

### 用户管理模块设计

**用户管理就是用来管理用户信息的，包括用户账号的增、删、改、查、停用等功能，这其中包括用户的个人信息，联系电话，账号密码以及角色。**

* **对于用户密码，一般不允许管理员直接修改，而是进行重置。**
* **通过对用户赋予不同的角色可以赋予相应的权限，一个用户可以赋予多个角色。**

### 角色管理模块设计

**角色管理模块就是对系统角色进行增、删、改、查、停用，包括角色名称，角色描述，角色状态，为角色分配权限等。**

* **对于删除功能，需要判断是否有用户关联了角色，如果有关联就不能直接删除。**
* **角色配置权限直接配置操作权限(按钮级别)就行**

### 权限管理模块

**权限管理模块就是对系统权限进行增删改查**

* **展示系统权限数据时以树形结构来展示**

程序设计过程中，我们常常用树形结构来表示某些数据的关联关系，如企业上下级部门、商品分类等等，通常而言，这些树状结构需要借助于数据库完成持久化。然而目前的各种基于关系的数据库，都是以表格的形式记录存储数据信息，在数据库中我们可以用自关联来保存这样的关系。



**如下图的企业组织架构就是一种典型的树形结构**

