**通用业务文档**

## 组织机构

目录

[组织机构 1](#_Toc18676358)

[文件上传 3](#_Toc18676359)

[websocket 3](#_Toc18676360)

1. **首先对于一个组织机构，一般存在以下几个表结构**

**1、user表 这个一般是存储系统的登录用户的账号与密码的表。**

**2、depart表 这个一般储存系统中的公司、部门、岗位等信息，就是这个公司的层级结构**

**3、user\_org表 这个是关联user表与depart的中间表**

**4、role 这个是专门存储系统用户角色的表**

**5、role\_user 这个是role与user之间关联的中间表。一个用户一般对应一个角色**

**6、menu表 这个是存贮菜单信息的表**

**7、role\_menu表 这个是role与menu的中间表**

**8、permission 表, 权限能操作的资源以及操作方式。**

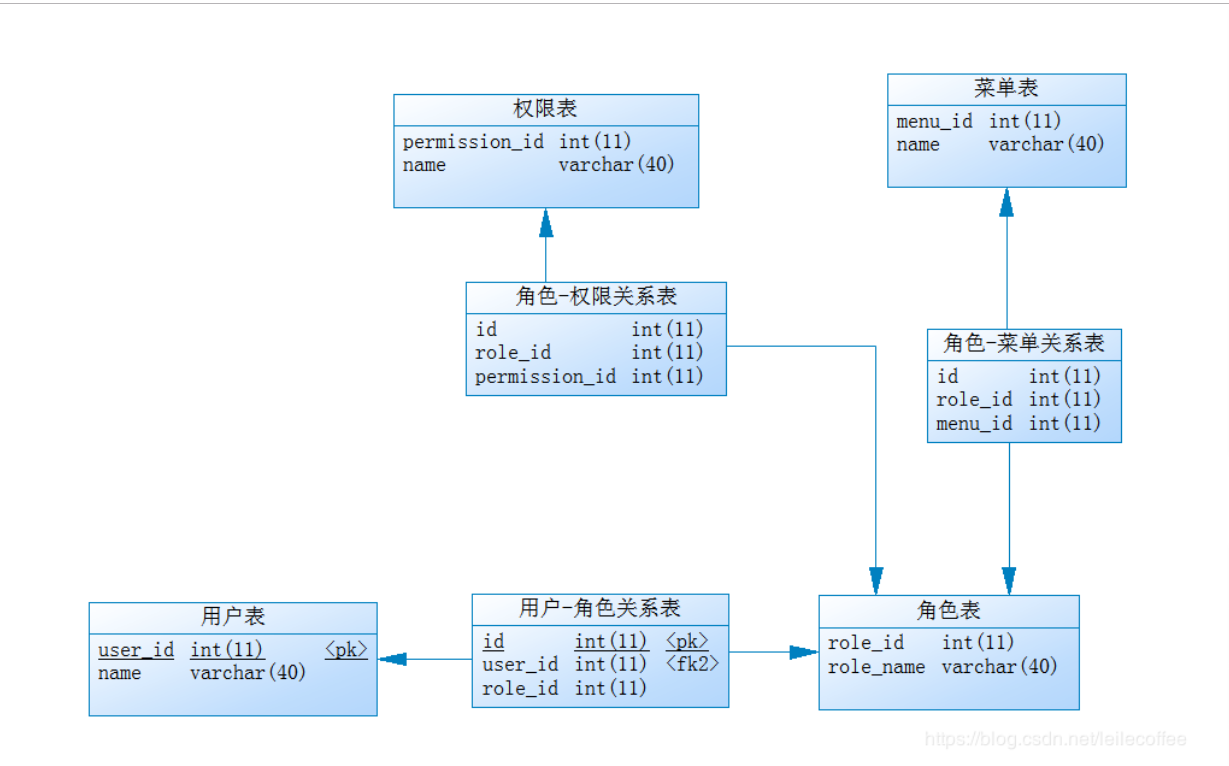
**9、role\_permission,这个是角色所能进行的资源权限**

**备注：后端代码 权限控制的代码不要浸入性太强，每个controller都要写权限代码，权限应该统一控制。**

**参考项目**

[**https://github.com/CaiBaoHong/biu**](https://github.com/CaiBaoHong/biu)

**参考图片**



## 文件上传

**现在系统里面的文件上传的实例有K文件上传、钟文件上传与自己网上的文件上传**

## websocket

**Websocket是存在于互联网交互架构应用层的协议，虽然在实际的运用场景中websocket需要http去启动链接，但该依赖关系不是必要的。Websocket连接完全可以独立于http，之所以将连接方式建立在http之上是因为http已经提供了便捷的沟通协议。利用http的协议完成wesocket握手连接可以节省开发独立握手协议的成本。**

**Websocket的优势在于两台计算机传输信息的实时性。在HTTP1.1推出长连接之后，websocket和http在tcp上的性能损耗已经没有过大的差别。在HTTP之前每次交互后都需要重新建立tcp链接，因此在一个交互频繁的引用环境下，其机制非常浪费计算机资源。不过Websocket的机制是完全针对实时通讯的，所以在设计底层存在很大的差异。HTTP针对两台计算机分配了不同的角色，客户端和服务端。所有需求会通过客户端提出，服务端会做出相应的回应。所以建立在HTTP协议的交互上，只有客户端有权发出请求。如果想实时获得信息更新，客户端需要频繁的想服务器发送请求。**

**Websocket协议并没有划分服务端和客户端角色，而将两端都视为信息传输端。因此服务器也有了可以主动推送数据的功能，有助于任何信息更新都可以第一时间通知用户。这样的链接不仅能用在聊天软件中，也可以用于在服务器设定监视日志脚本，将任何报错信息实时推送给用户。其设计降低了交互成本，因为如果建立在http协议上，客户端必须定期调用数据，而数据在何时更新是未知的。每次调用却未获得新数据的情况下，都是一次计算机资源的浪费。而websocket是非常好的通讯解决方案。**

**————————————————**

**版权声明：本文为CSDN博主「Uncle\_Joe」的原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接及本声明。**

**原文链接：https://blog.csdn.net/weixin\_40784198/article/details/81811039**

**websocket的应用场景**

[**https://blog.csdn.net/lldouble/article/details/80742082**](https://blog.csdn.net/lldouble/article/details/80742082)

**学习的项目：**[**https://github.com/yizhiwazi**](https://github.com/yizhiwazi)

[**https://www.jianshu.com/p/55cfc9fcb69e**](https://www.jianshu.com/p/55cfc9fcb69e)

**springboot解决跨域问题**

[**https://www.jianshu.com/p/477e7eaa6c2f**](https://www.jianshu.com/p/477e7eaa6c2f)

**跨域概念**

[**https://www.cnblogs.com/ysocean/p/9380551.html**](https://www.cnblogs.com/ysocean/p/9380551.html)