## 目标：

开发一款较为通用的组织机构系统，为其它系统服务，避免相同的事情重复做

## 设计原则

1 用户和部门、角色或其它的实体项都使用多对多的关系，以便扩展。

2只有一切权限设置只针对角色进行，这样可以使得权限分配、控制更加清晰。

## 实体介绍

### 实体关系



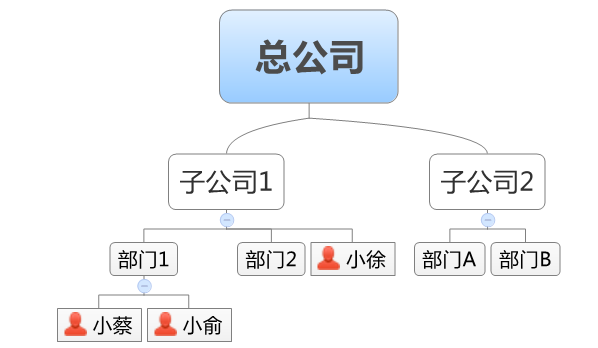
|  |  |  |  |
| --- | --- | --- | --- |
| **实体1** | **实体2** | **关系** | **备注** |
| 部门 | 用户 | 多对多 | 用户是可以同时存在与多个部门的，但需要指定一个主部门。 |
| 部门 | 角色 | 一对多 | 角色相当于组织下的一种职务 |
| 角色 | 用户 | 多对多 | 用户可以跨部门分配角色 |
| 角色 | 功能项 | 多对多 |  |

### 1部门

**说明:**

1部门为多级结构

### 2用户

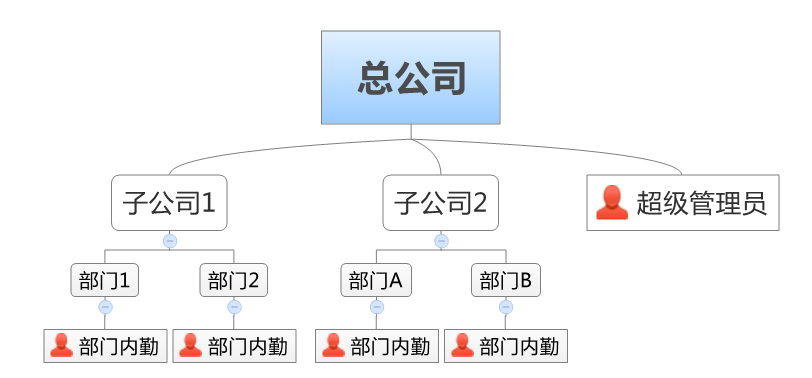


**说明:**

1 用户可添加至组织的任何一个层级中。

2 用户只属于一个组织单位（部门）

### 3角色



**说明:**

1 角色可添加至组织的任何一个层级中（和用户类似）。

### 4功能点（应用项）

**说明:**

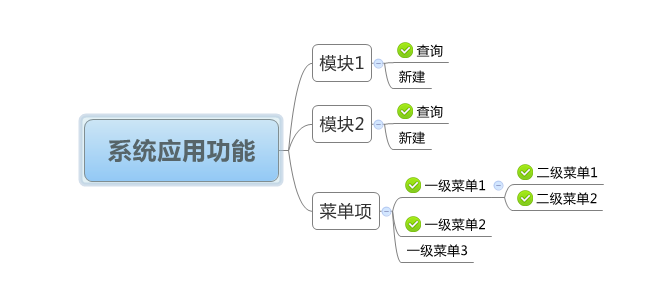
1 系统应用功能也是一个层级结构

2 针对每个功能可配置属性（名称、url、功能编号、类型 等 ）

**功能项类型**（必选项）

**1访问控制**：用来对url进行访问控制

**2可用性**： 用来控制界面功能元素的显示如（按钮、菜单等）



## 数据库设计



参考：《org.pdm》

## 模块说明

### 1部门管理

略

### 2用户管理

略

### 3角色管理

略

### 4功能项管理

略

### 5权限设置

略