**锦宏产业园管理软件实现方案**

1. **用户权限控制**

原始需求:

1. 4个角色: 总经理, 招商管理, 财务管理, 物业管理; 总经理拥有所有权限;
2. 每个部门有部长, 有本部门内部分配权限的权限;
3. 特殊岗位可以拥有进入多个模块的权限;

方案:

采用经典用户-角色-权限的实现即可:

用户(User): 包含拥有的角色(Role)的集合;

角色(Role): 包含一个权限(Privilege)的集合; 也可考虑包含子角色列表;

权限(Privilege): 描述一个对应的操作权限;

权限列表:

1. 财务模块*[访问]*
   1. 收入日报表*[访问, 操作]*
   2. 未交的房租,押金明细表*[访问, 操作]*
   3. 已交的房租,押金明细表*[访问, 操作]*
   4. 未转的物业管理费明细表*[访问, 操作]*
   5. 已转的物业管理费明细表*[访问, 操作]*
   6. 月度收支明细表*[访问, 操作]*
   7. 半年度收支明细表*[访问, 操作]*
   8. 年度收支明细表*[访问, 操作]*
   9. 月度收支汇总表*[访问, 操作]*
   10. 财务模块相关权限授予的权限
2. 物业模块*[访问]*
   1. 人事管理*[访问]*
      1. 任职全厂职工名单*[访问, 操作]*
      2. 退休职工名单*[访问, 操作]*
   2. 消防管理*[访问]*
      1. 锦宏消防资料*[访问, 操作]*
   3. 物业管理*[访问]*
      1. 租赁单位水电费一览表*[访问, 操作]*
      2. 水电费总厂开票一览表*[访问, 操作]*
      3. 停车费统计表*[访问, 操作]*
      4. 钥匙管理,空调设备*[访问, 操作]*
      5. 电讯电话,电信收费清单*[访问, 操作]*
      6. 维修记录*[访问, 操作]*
      7. 各楼层租赁单位情况表*[访问, 操作]*
      8. 收缴停车费明细表*[访问, 操作]*
      9. 锦宏平面图*[访问, 操作]*
      10. 应收停车费汇总表*[访问, 操作]*
      11. 租赁单位安全管理协议表*[访问, 操作]*
      12. 租赁户入住表*[访问, 操作]*
      13. 租赁户退组表*[访问, 操作]*
      14. 监控探头*[访问, 操作]*
      15. 车位信息管理表*[访问, 操作]*
      16. 抄电表数*[访问, 操作]*
      17. 水电费收费清单*[访问, 操作]*
   4. 物业管理相关权限授予的权限
3. 招商模块*[访问]*
   1. 楼宇区域位置图示*[访问]*
   2. 租赁户企业基本情况*[访问]*
   3. 租金,物业费缴纳情况记录管理业务*[访问, 操作]*
   4. 租赁期限变更,续租,退租一栏业务*[访问, 操作]*
   5. 退租流程展示表格*[访问, 操作]*
   6. 仓库租赁情况*[访问, 操作]*
   7. 合同管理*[访问, 操作]*
   8. 招商模块相关权限授予的权限

对于叶子功能, 至少设定访问权限, 对于可操作的则另外设定操作权限; 对于非叶子功能, 一般提供访问权限;

一个权限的定义一般包括:

名称: 用英文表示,, 比如Access\_EmployeeList

描述: 一般就是上述列出的描述, 比如 访问人事列表

角色主要就是权限的集合:

比如: 财务部长这个角色拥有的权限就是1.1-1.10, 财务部普通职工这个角色拥有的权限是1.1-1.9;

用户,角色,权限的实现:

定义用户表, 角色表, 和权限表

用户表User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 |  |  |  |  |
| Username |  |  |  |  |
| password |  |  |  |  |
| RoleIds |  |  |  |  |

角色表Role

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 |  |  |  |  |
| Id |  |  |  |  |
| Name |  |  |  |  |
| Description |  |  |  |  |
| PrivilegeIds |  |  |  |  |

权限表Privilege

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 |  |  |  |  |
| Id |  |  |  |  |
| Name |  |  |  |  |
| Description |  |  |  |  |

和界面的绑定

用户成功登陆后, 系统会记录当前登录用户的实例, 会定义一个权限管理器PrivilegeManager类, 专门针对用户计算是否拥有某项权限(如果是内置测试系统用户所有权限全开); 对于WPF, 定义两个转换器, HasPrivilegeToVisibilityConverter和HasPrivilegeToIsEnabledConvert; 相关模块对应控件直接绑定登录用户进行转换即可;

1. **全局变量**

系统配置: 需要时添加具体的项

应用程序信息: 当前登录用户, 登录时间等等

1. **界面相关**

原本不准备采用UI框架, 但是经理发过来了, 暂时没有下载下来; 不过套用不麻烦; 对于这类项目, 仅仅是一些表格的增删改查的操作, 其实用不用UI框架并不大重要. 以前很多界面很复杂的项目都不怎么用UI框架. TODO. 到时候会套用;

项目采用M-V-VM Pattern. 每个模块的对应界面制作对应的View类, 同事定义和这个View类对应的ViewModel类, ViewModel是对M的数据封装和逻辑封装, 满足View的直接绑定的需求; 本项目的ViewModel会比较简单, 对于表格操作界面, 一般主要包括对应的DataTable, 和界面上对应的控制变量等等;

1. **表格的实现**

不管是需要进行增删改查的可操作的表格还是输入参数输出只读报表的表格, 都是在每个对应模块的View上使用简单的ListView或DataGrid控件实现; 根据需要提供排序, 过滤筛选, 样式统一定义尽量美观;

有一点要说明, 表格的操作界面不会模仿纸质表格的样式, 还是注重操作性能为主, 导出后不保证可见即所得. 注重数据本身, 样式次之.

每个模块根据需要提供导出为Excel和Pdf的功能, 具体实现有可参考的, 不用多说;

1. **数据访问**

客户端不直接访问数据库, 而是通过服务端的WebService进行操作, 服务端的WebService服务提供基本的数据库操作支持; 对于复杂的数据业务, 不在WebService中提供过多面向具体业务的操作接口, 而是仅仅提供基本的增删改查接口, 具体业务由客户端对这些接口封装实现;

兼容性问题:

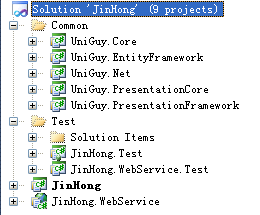
公司希望兼容其他客户端的访问, 个人不大确定某种形式的Web service能如何做到这点, 不过如果服务端的服务接口比较简单, 提供基本数据库访问操作而适当和具体业务解偶联, 那么提供多套服务也是一样的, 比如同时提供WebService, WCF, ASP.NET for MONO啥的.

安全性问题:

如果服务端的服务提供基本的数据库访问接口, 可能会有安全问题, 比如某个用户直接访问该服务Delete所有记录等等; 这个时候可以采用WebService服务本身的安全机制, 可能通过SoapHeader添加访问密码等, 只有通过验证的客户端可以访问此服务;

数据库采用MySql, TODO, 目前MySql没有安装成功, 临时使用Sqlite, 可在之后替换为MySql; 不会有影响;

1. **项目层次**



采用垂直架构, 便于项目的复用;

1) Common(目的是为了复用)

UniGuy.Core: 业务不相关, 界面不相关

UniGuy.EntityFramework: 业务不相关, 包含一些基本实体的定义

UniGuy.Net: 和通信有关, 这里没用到

UniGuy.PresentationCore, UniGuy.PresentationFramework: 业务不相关, 界面相关WPF

2) Test 测试项目, 需要进行测试的时候添加相关方法进行测试

3) JinHong 主项目, 客户端

4) JinHong.WebService 服务端

注: Common中的某些基本类库,如UniGuy.Core, UniGuy.EntityFramework可能通过被JinHong客户端和服务端引用;

1. **说明**

**本文档会根据具体进展不断组织完善和优化, 暂时只写这么多, 先进行代码处理, 文档和代码会同步更新;**

**// TODO 整理到合适位置**

关于查找, 一般是直接查询返回DataTable, 添加和修改记录会根据DataTable中的一行创建对应数据模型的实例进行操作并保存到数据库, 最后刷新; 还有一种模式是直接实例化每个行为数据模型实例, 性能损耗大, 可异步进行动态绑定, 但增加行和修改行可直接绑定不用刷新;

1. **分模块**
   1. **招商模块**
      1. **合同管理**

服务端有一个文件系统, 合同文件放在服务端文件系统下Contracts目录, 其下存放所有合同, 以分级目录区分层次分类;

WebService实现文件系统基本方法, 比如查找子目录, 查找目录下文件, 下载文件和上传文件的方法;

客户端左侧显示文件系统目录, LazyLoading, (其他系统的文件提供动态刷新功能), 可以点击刷新按钮刷新目录, 右侧显示下载的合同内容, 采用合适的DataTemplate呈现, 没有匹配的则直接用默认程序打开合同文件;

合同管理员提供添加合同文件的菜单; 不提供删除菜单;

* 1. **物业模块**
  2. **财务模块**

1. **数据库表设计**
   1. **楼宇信息(BuildingInfo)**

包含所有楼宇的信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **含义** | **类型** | **约束** | **备注** |
| Id |  | varchar | PK | 直接用栋名表示, 如A, B等 |
| Name | 名称 | varchar |  |  |
| Notes | 备注 | varchar |  |  |
| … |  |  |  |  |

* 1. **房室信息(RoomInfo)**

表示楼宇中的每个可独立租赁的房间的信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **含义** | **类型** | **约束** | **备注** |
| Id |  | varchar | PK | 一般栋ID+房间号, 如A101等 |
| Name | 名称 | varchar |  |  |
| Notes | 备注 | varchar |  |  |
| BuildingId | 所属楼宇Id | varchar | FK |  |
| Area | 面积 | double |  |  |
| … |  |  |  |  |

* 1. **仓库信息(WareHouseInfo)**

表示仓库

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **含义** | **类型** | **约束** | **备注** |
| Id |  | varchar | PK |  |
| Name | 名称 | varchar |  |  |
| Notes | 备注 | varchar |  |  |
| … |  |  |  |  |

* 1. **员工信息(EmployeeInfo)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **含义** | **类型** | **约束** | **备注** |
| Id |  | varchar | PK | 工号 |
| Name | 名字 | varchar |  |  |
| Description | 描述 | varchar |  |  |
| IdCardNo | 身份证号 |  |  |  |
| BirthDate | 出生日期 |  |  |  |
| BirthPlace | 出生地址 |  |  |  |
| GenderCode | 性别代码 |  |  |  |
| EnterDate | 入职日期 |  |  |  |
| LeaveDate | 离职日期 |  |  |  |
| LeaveReasonCode | 离职原因代码 |  |  |  |
| District | 家庭住址区 |  |  |  |
| Street | 街道 |  |  |  |
| Address | 详细地址 |  |  |  |
| PhoneNo | 电话 |  |  |  |
| ZipCode | 邮编 |  |  |  |
| Notes | 备注 |  |  |  |
| \… |  |  |  |  |

* 1. **社会成员信息(SocialUnitInfo)**

表示独立行为的社会单位, 可以包括组织(公司, 机构等等), 以及个人; 这里一般表示前来租赁的公司或者个人等

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **含义** | **类型** | **约束** | **备注** |
| Id |  | varchar | PK | 用企业编号等表示, 没有可直接用Guid |
| Name | 名称 | varchar |  |  |
| Notes | 备注 | varchar |  |  |
| … |  |  |  |  |

* 1. **收支信息(IncomeAndExpenditure)**

表示收支行为项和相关数据; 登记所有相关费用项目时, 除了记录相关活动数据到对应的表, 费用信息统一记录到该表;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **含义** | **类型** | **约束** | **备注** |
| Id |  | varchar | PK | 流水号, 或者Guid等唯一表示 |
| Date | 收支日期 | date |  |  |
| Category | 分类 |  |  | 比如 房租, 押金, 停车费, 物管费, 水电费等等 |
| Amount | 发生金额 |  |  | 支出用负数表示 |
| SocialUnitId | 关联收支单位Id |  |  |  |
| SocialUnitName | 关联单位名称 |  |  | 这个值是冗余的, 可以根据SocialUnitId从SocialUnitInfo表中查询得到, 容易的目的是便于查询出报表等 |
| Notes | 备注 | varchar |  |  |
| … |  |  |  |  |

* 1. **TODO…**