

<超市管理系统>

系统设计说明书

一 引言

1.1 编写目的

- 本数据库设计说明书是关于超市管理系统设计，主要描述此系统的各个功能模块，设计类图，ER 设计，表结构设计等。
- 本数据库设计说明书读者：用户，系统设计人员，系统测试人员，系统维护人员。

1.2 编写背景

- 本次待开发的系统：超市管理系统。
- 此项目的提出者：能混绝不 C
- 开发者：能混绝不 C
- 用户：XX 超市

1.3 参考资料

《超市管理系统设计说明书》

二 体系结构设计

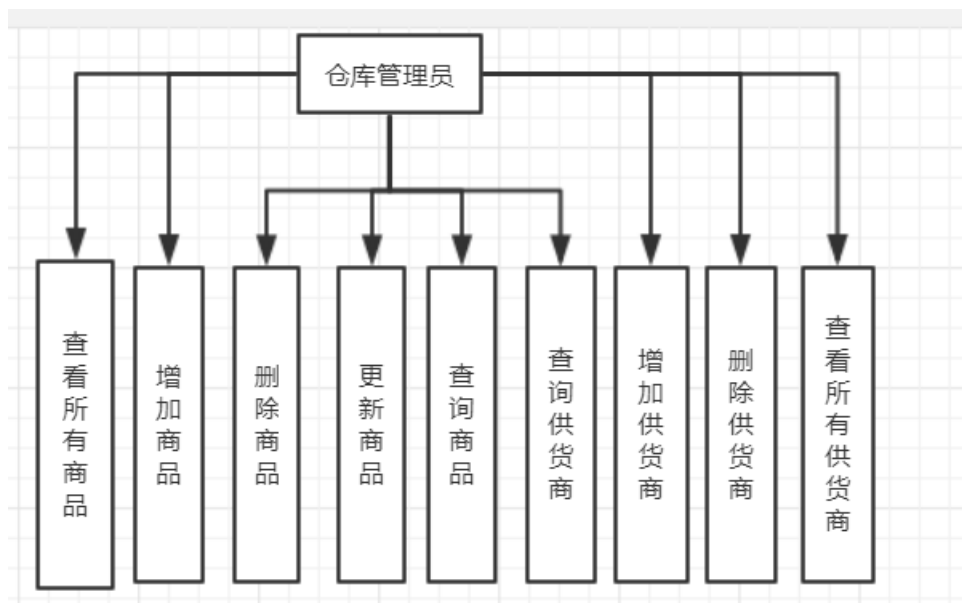
本系统的设计主要是基于 MVC 设计模式，MVC 是一个设计模式，它强制性的使应用程序的输入、处理、输出分开。使用 MVC 应用程序被分成三个核心部件：模型、视图、控制器。它们各自处理自己的任务。

- View:视图 最顶层，为用户提供使用界面，和用户直接进行交互，是程序的外壳。对老式的 Web 应用程序来说，视图就是由 HTML 元素组成的界面，在新式的 Web 应用程序中，HTML 依旧在视图中扮演着重要的角色，但一些新的技术已层出不穷，它们包括 Macromedia Flash 和象 XHTML, XML/XSL, WML 等一些标识语言和 Web services。如何处理应用程序的界面变得越来越有挑战性。MVC 一个大的好处是它能为你的应用程序处理很多不同的视图。在视图中其实没有真正的处理发生，不管这些数据是联机存储的还是一个雇员列表，作为视图来讲，它只是作为一种输出数据并允许用户操纵的方式。

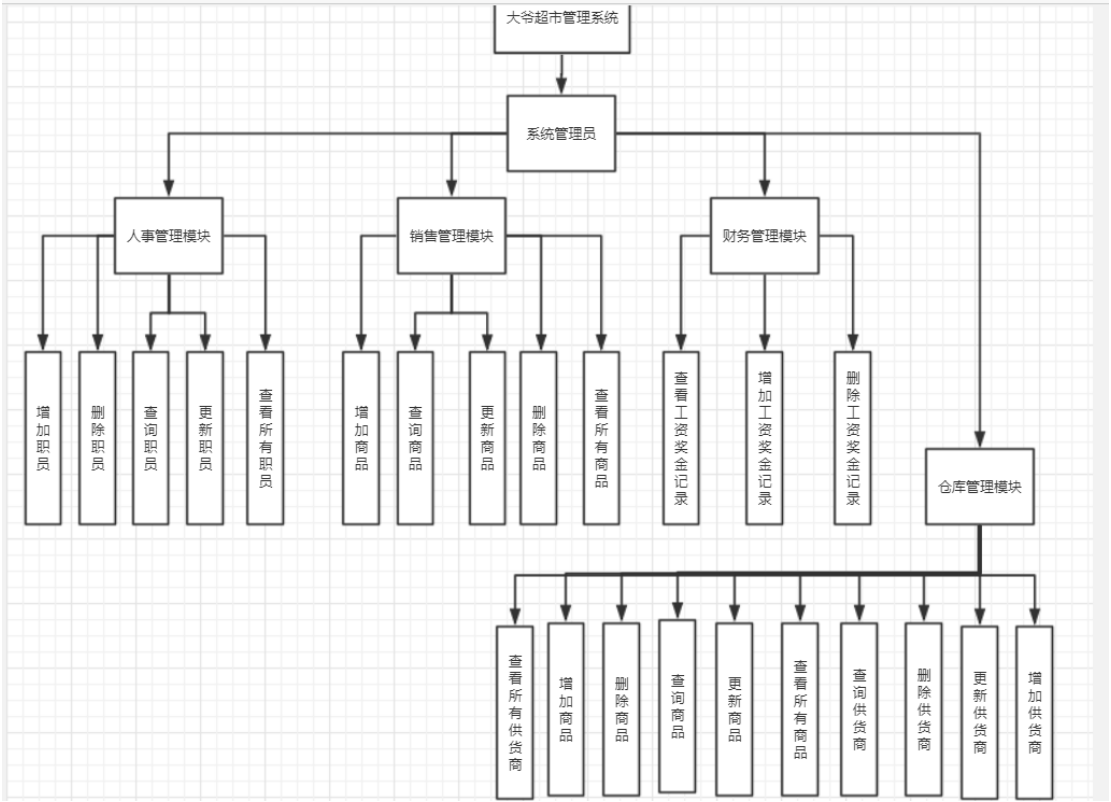
- Model:数据层 最底层，是核心的数据层，程序需要操作的数据或信息. 在 MVC 三个部件中，模型拥有最多的处理任务。被模型返回的数据时是中立的，就是说模型与数据格式无关，一个模型能为多个视图提供数据。由于应用于模型的代码只需要写一次就可以被多个视图重用，减少了代码的重复性。
- Controller: 控制层 负责根据用户从View层输入指令，选取“数据层”中的数据，然后对其进行相应的操作，产生最终的结果. 例如单击 web 页面中的超链接和发送 HTML 表单时，它只是接收请求并决定调用哪个模型构件去处理请求，然后在确定用哪个视图来显示返回的数据。
- MVC 是一种软件设计典范，用于中业务逻辑，数据, 页面实现分离的方法组织代码，将业务逻辑整合到一个部件里面，在改进和个性化定制界面以及用户交互的同时不需要重写编写业务逻辑，实现业务逻辑和前端界面的分离. MVC 被独特的发展起来用于映射传统的输入, 处理和输出功能在一个逻辑的图形化用户界面的结构。
- MVC 最开始是用于桌面程序当中，M 指的是业务模型，V 指的是用户界面，C 指的是控制器。使用 MVC 设计模式的目的是 在桌面应用程序软件将 MV 分离，之前的设计模式都是，将数据从数据库中取出之后直接开始渲染界面，如果中间加上 Controller 层，我们可以让 Model 只负责和数据库进行交互操作，Controller 负责处理数据 这样可以实现较多的代码复用，减少代码冗余和降低耦合性。

三 功能模块层次图

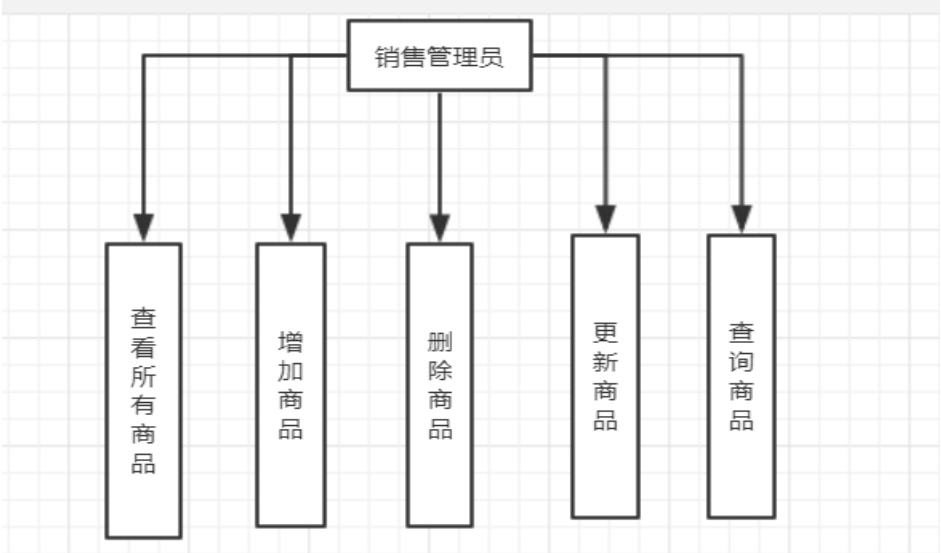
1. 仓库管理员



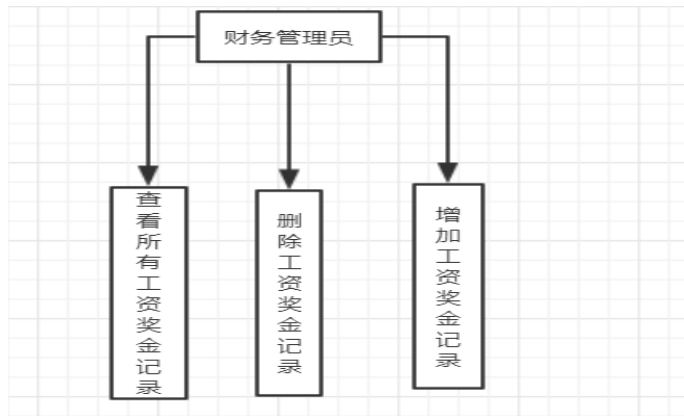
2. 系统管理员



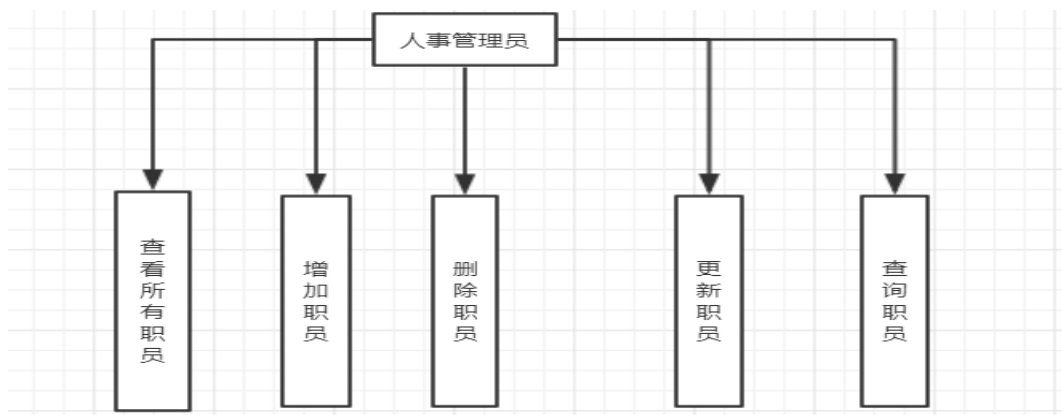
2. 销售管理员



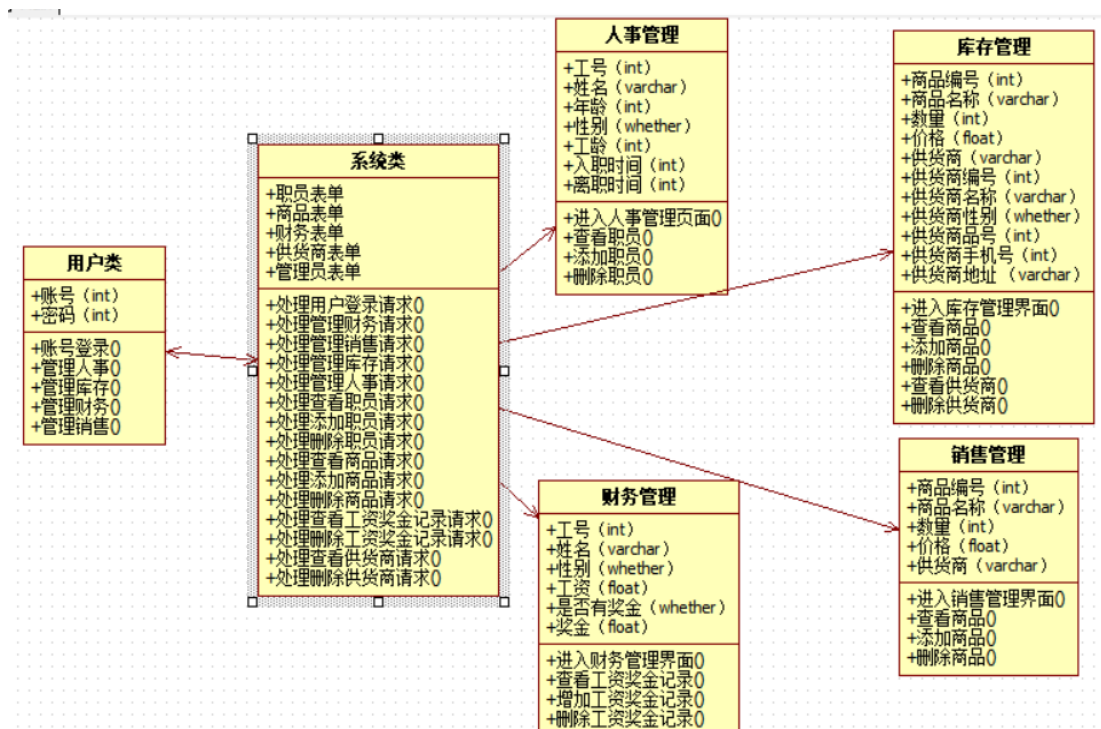
3. 财务管理员



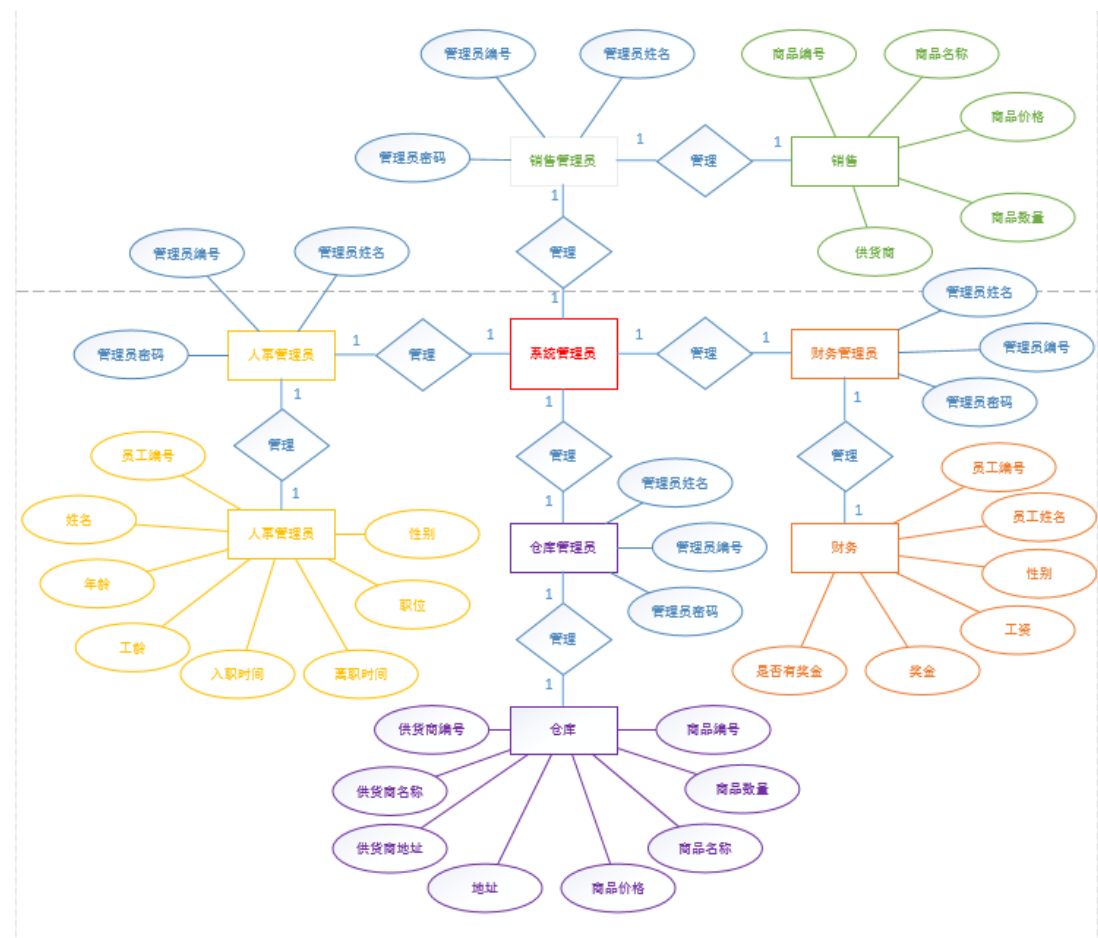
4. 人事管理员



四 设计类图



五 ER 设计



六 表结构设计

1. 职员 (user) 表:

字段名称	数据类型	说明
Id (员工编号)	Int(10)	Primary key
Username (员工姓名)	Varchar(20)	Not null
Age (员工年龄)	Int(10)	Not null
Sex (性别)	Varchar(5)	Not null
Workyear (工作年份)	Int(10)	Not null
Hiredate (入职时间)	Varchar(20)	Not null
Leavedate (离职时间)	Varchar(20)	Not null
Job (职位)	Varchar(20)	Not null

2. 管理员（ad）表：

字段名称	数据类型	说明
Id（管理员编号）	Varchar(10)	Primary key
Username（管理员姓名）	Varchar(10)	Not null
Password（管理员密码）	Varchar(10)	Not null

3. 商品（goods）表：

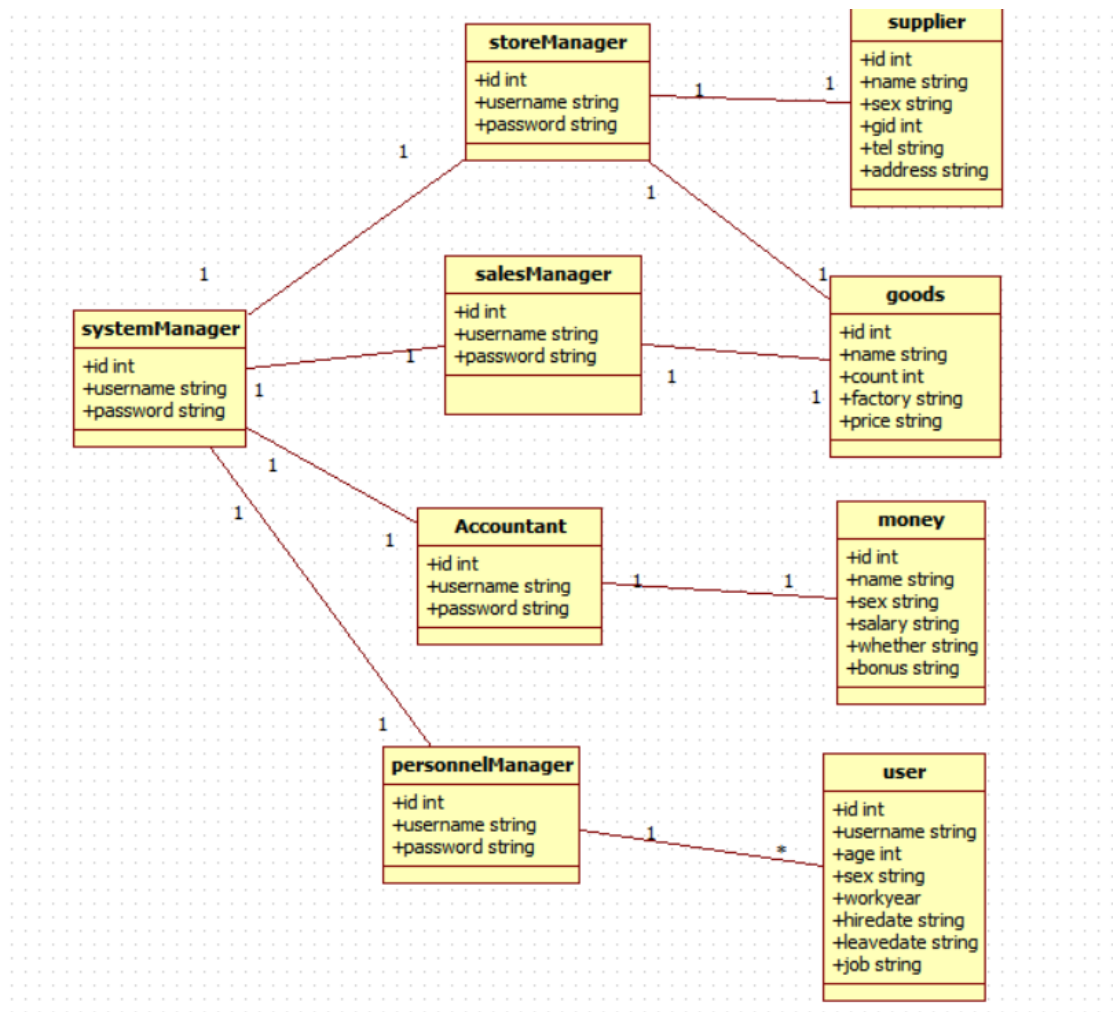
字段名称	数据类型	说明
Id（商品编号）	Int(10)	Primary key
Name（商品名称）	Varchar(20)	Not null
Count（商品数量）	Int(10)	Not null
Price（商品价格）	Varchar(20)	Not null
Factory（供货商）	Varchar(40)	Not null

4. 供货商（supplier）表：

Id（供货商编号）	Int(10)	Primary key
Name（供货商名称）	Varchar(20)	Not null
Sex（性别）	Varchar(5)	Not null
Gid（商品编号）	Int(5)	Not null
Tel（供货商电话）	Varchar(40)	Not null
Address（地址）	Varchar(80)	Not null
Id（供货商编号）	Int(10)	Primary key

5. 财务（money）表：

字段名称	数据类型	说明
Id（职员编号）	Int(10)	Primary key
Name（职员姓名）	Varchar(20)	Not null
Sex（性别）	Varchar(4)	Not null
Salary（工资）	Varchar(10)	Not null
Whether（是否奖励）	Varchar(5)	Not null
Bonus（奖金）	Vachar(45)	



说明：系统管理员（systemManager）能实操作所有数据库：

- ① 管理管理员（包括仓库管理员（storeManager）、销售管理员（salesManager）、财务管理员(Accountant)、人事管理员(personnelManager)）。
- ② 管理管理员管理的各部分数据库（商品（Goods）、供货商（supplier）、职员（user）、财务（money））。
- ③ 仓库管理员（storeManager）：
 - 操作商品（goods）数据库：可对商品进行增加、删除、修改、查询。
 - 操作供货商（supplier）数据库：可对供货商进行增加、删除、修改、查询
- ④ 销售管理员（salesManager）：操作商品（goods）数据库：可对商品进行增加、删除、修改、查询。
- ⑤ 财务管理员（Accountant）：操作奖金（money）数据库：可对工资奖金记录进行增加、删除、修改、查询。
- ⑥ 人事管理员（personnelManager）：操作职员（user）数据库：可对职员进行增加、删除、修改、查询

七 系统安全和权限设计

1. 用户标识与鉴别：当用户在登录界面输入登录信息并确认登录时，系统将核对其信息是否存在且正确，通过鉴定后才提供系统的使用权。
2. 存取控制：通过用户权限定义和合法权检查确保只有合法权限的用户访问数据库，未被授权的用户无法存取数据。如：财务管理员无法对商品数据库进行数据存取。
3. 视图机制：为不同的用户定义视图，通过视图机制把要保密的数据对无权的用户隐藏起来，从而自动地数据提供一定程度的安全保护。