**工资系统设计问题及建议**

**一：关于“五、系统详细设计：**

你写的有点乱，也不够详细，我不能很好地理解你的设计思想。建议你将每一个模块的界面用QT画出来，并放在设计文档里面。这个过程你会发现问题。

**二：关于“六、系统数据结构设计”：**

数据库结构不合理，不符合数据库设计的三个范式要求。以下是部分建议，你参考完善。

1、雇员数据库(Clerk.db)：

（1）没有索引字段；两个员工重名怎么办？所以需要有“员工编号”；

（2）缺少字段：登录账号、密码、所属部门编码、权限编码等

2、雇员账户数据库(Acount.db)：

数据有重复冗余：和Clerk.db两个数据表不能同时都有Name…….这个表应该和Clerk.db合并

3、应该有一个“部门信息表”：

这个表存储本公司的部门设置信息，如：部门编码、部门名称等，Clerk.db表中每个雇员通过部门编码和本表关联

4、工资数据表(Salary.db)：

这个是最核心的数据表。这个表要看你的系统怎么设计了。理论上说，每一张工资表都是一张独立的表，比如工资表1和工资表2是没有任何关系的，因为两张表的发放人员可能不同、工资项目也可能不同；或者，一份表是基本工资表，另一份表是补贴和奖金发放表等情况。

所以：（1）每一份表的数据结构都应独立存储。（2）表的数据列不能固定，是用户动态定义的。

**三、你需要实现的功能是设计最基本的结构，比如：你可能需要设计下面几个表：**

1. 工资表总表：

工资表编码、工资表名称、制表人、审核人、批准人、发放日期、是否已归档（归档以后这个表的结构和数据不能再修改了，除非解除归档）等…….

1. 工资明细表：

工资表编码、员工编码、应发项目（基本工资、职务工资、各种补贴等这些也是要用户自己定义的）、应扣项目（公积金、个人所得税等也是由用户自己定义的）、小计项、小计项的计算公式、每个列的显示和打印顺序等；

这个表通过“工资表编码”和（1）表关联。

上面的表是我粗略的建议，根据数据库设计范式可能还要优化拆分出更多的子表来（比如上面红色字体部分），这个看你怎么设计了，你参考这个思路设计数据库吧。

**四、其他**

所有的管理系统基本功能模块无非就是“增删改查”。你要想一下，你的系统交付给用户后，用户首先需要做的工作是系统初始化工作：用默认的管理员账户定义公司信息、部门信息、注册用户、分配用户权限等，然后普通操作员用户才能开始定义工资表结构、制作一个具体的工资表等工作；设计得完善的话，你的总菜单及菜单项也是应该根据用户的操作权限动态开关的（这些是锦上添花的功能，可以在以后实现）…….

这个系统并不简单，很多功能都需要动态实现。另外：如果用户用了几年，有上百个历史工资表数据，需要统计的话可以用多线程实现；可以用Matplotlib做工资结构分析图等；你不是想玩网络编程吗，可以添加生成工资发放数据表，通过邮件或远程接口的形式向银行的系统接口发送数据等高大上的功能……可扩展的功能多着呢……

慢慢玩吧…..