

实战项目 – 银行账户管理系统

一、项目名称

XX 银行账户管理系统

二、项目目的

1. 检验 Module03 – C++语言 中的 语言基础 部分的掌握程度；
2. 初步锻炼项目参与者对业务需求的分析、项目架构的思考；
3. 考核项目参与者的编码习惯。

三、知识覆盖范围

1. M0301-C++语言基础；
2. C++ I/O 流；
3. C 时间函数。

四、开发方式与时间

个人，时间：36 hours。

五、建议的开发环境

Linux、Eclipse、vim、make、g++、gdb、CVS。

六、简单需求描述

（请注意：以下需求描述不是正式的项目需求文档格式）

1. 账户管理：
 - A. 账户类型：定期储蓄账户、活期储蓄账户
 - B. 开户（一个顾客可以有多个账户）、销户
 - C. 账户的冻结与解冻
2. 客户服务：
 - A. 存款、取款、转账（柜台受理、ATM 自助服务）
 - B. 查询余额（柜台受理、ATM 自助服务）

- C. 打印存取清单（柜台受理、ATM 自助服务）
- D. 更改密码（柜台受理、ATM 自助服务）
- E. 账户挂失
- 3. 账务管理：
 - A. 账户的年费扣除
 - B. 各类手续费的扣除
 - C. 储蓄利率
 - D. 账户储蓄利息
- 4. 统计：
 - A. 每月新增的账户数
 - B. 本月新增储蓄总额
 - C. 余额高于 n 元的账户数
 - D. 余额低于 m 元的账户数
 - E. ...
- 5. 查询：
 - A. 按指定姓名查找账户
 - B. 按指定电话号码查找账户
 - C. 按指定查找余额大于、小于 n 元的账户（列出所有符合条件的账户的信息）
 - D. ...
- 6. 事务日志（log）：
 - A. 所有的事务必须按照易于理解的格式记录：如对于某账户中的一次取款，必须记录如账户名、账号、取款金额、时间、柜员机或银行受理人员的 ID 等
- 7. 数据持久化（保存）
 - A. 所有运行期产生的信息（如银行雇员、柜员机、账户等等），均需保存到磁盘，下次启动程序的时候可以从保存的文件读入

七、交付

- 1. 所有的 case 描述：
 - 例：银行卡账户开户 case 描述：
 - A. 顾客 C1 在个人现金受理柜台前坐下；
 - B. 银行雇员 EC2 起立，询问顾客 C1 要办理何种业务；
 - C. 顾客 C1 告诉银行雇员 EC2，要办理银行卡开户；

- D. 银行雇员 EC2 请顾客 C1 出具身份证明、顾客 C1 出具自己的身份证明；
 - E. 银行雇员 EC2 验证顾客 C1 提供的身份证明；
 - F. 银行雇员 EC2 将顾客 C1 的身份信息录入系统、产生新用户；
 - G. 银行雇员 EC2 将顾客 C1 所要办理的业务、身份信息打印成表单，交由顾客 C1 确认；
 - H. 银行雇员 EC2 将银行卡账号与顾客 C1 在系统中的用户关联起来；
 - I. 银行雇员 EC2 将银行卡交与顾客 C1；
 - J. 顾客 C1 离柜。
- 2. 完整的项目代码，包括完整的目录结构、C++源代码、Makefile、项目可能使用的各类 Script 等等；
 - 3. 用于测试该系统各部分功能所需的数据，如至少 500 个账户的信息、100 次事务的记录等，或提供能产生等效数据的工具（如 C++程序、Shell Scripts 等）；
 - 4. 输出的数据格式描述；
 - 5. 本项目不需提供《概要设计说明书》、《详细设计说明书》等。