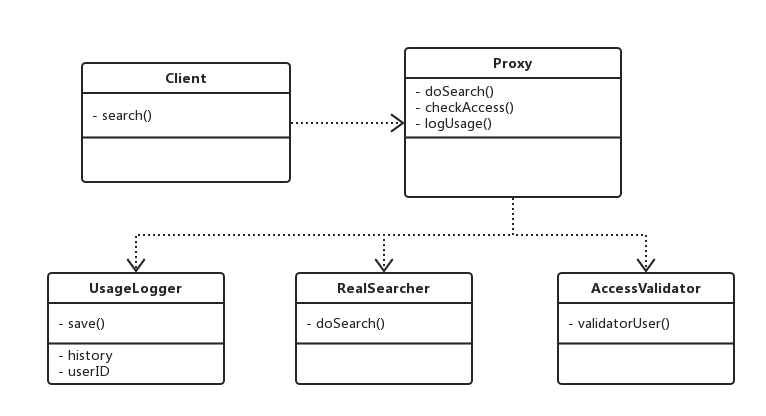
系统要求

完善基于安全的计数程序

根据用户的id,和输入的查询条目，基于安全来统计数据库中的这个条目的数目!

其他？

UML类图



类图使用PrcessOn（https://www.processon.com）在线生成工具完成。

计数程序包含五个类，Client、Proxy、UsageLogger、RealSearcher及AccessValidator。各类之间关系见上图。

Client类用于对使用数据库的客户的抽象，包含一个方法：

1. search()：实现查找功能引导；

Proxy类中为基于安全的代理模式的抽象，包含三个方法：

1. doSearch()：实现查询功能；
2. checkAccess(): 实现访问验证功能；
3. logUsage(): 实现用户使用记录功能；

UsageLogger类为记录用户访问功能模块的抽象，包含一个方法，两个属性。

两个属性为：

1. userID：客户ID号；
2. history: 记录用户访问信息，包含用户ID和访问时间；

一个方法为：

（1）save()：实现记录用户信息的功能；

AccessValidator类为验证用户身份信息功能模块的抽象，包含一个方法：

（1） validatorUser(): 用于验证身份信息；

RealSearcher类为数据库查询功能模块的抽象，包含一个方法：

1. doSearch()：实现查找数据库中某个条目的数量功能；

系统实现

该版本的系统使用C++实现。工程包含十一个文件（main.cpp client.cpp client.h proxy.cpp proxy.h realsearcher.cpp realsearcher.h usagelogger.cpp usagelogger.h accessvalidator.cpp accessvalidator.h）。

其中：

1. main.cpp: 包含简单用户引导界面实现；
2. client.cpp & client.h: 包含用户类定义的实现，代理模式的引导；
3. proxy.cpp & proxy.h：包含代理类定义的实现，实现代理功能，以及对UML图中所表示的以下三个模块的引导；
4. realsearcher.cpp & realsearcher.h：包含查找类定义的实现，实现对数据库中某一条目下数量的查询；
5. usagelogger.cpp & usagelogger.h：包含对用户信息记录类定义的实现；记录用户使用数据库的时间；
6. accessvalidator.cpp & accessvalidator.h：包含对身份信息验证类定义的实现，判断用户是否拥有使用数据库的权限；

数据库条目查询实现过程：在用户输入用户ID及要查询的条目后，判断用户是否拥有使用数据库权限，若有则读取数据库文件查询相应条目下数据的数量，并记录下该用户的访问。

需要注意的实现细节如下：

1. 默认用户ID为admin可以使用该数据库；
2. 对作业所给的数据库文件转化为文本文件进行读取；
3. 由于数据库查询操作不属于本次作业的重心且时间有限，因此对数据库查询功能部分进行了简化；

系统运行结果

