一.概要说明：这次主要加入了联网这一大块功能，便于实现老师的班级管理和作业功能，老师设计了布置作业和添加班级的功能，学生则必须加入班级实现联网功能，老师可以通过后台查看学生的练习情况。其他的大部分继承于第一次的版本。

二.设计规范部分:

1.是否经过模块儿化设计 是

2.是否建立了相应模型评估与测量 是

3.是否使用合适的设计模式 是

三.代码规范部分：

1.是否遵循确定的命名规则 否

2.对代码是否注释 否

3.语句格式是否规范 否

四.具体代码部分

1. 类的设计和抽象是否合适 是

2*.*是否符合面向接口编程的思想 是

3. 对象生命周期的处理，是否对象引用已失效可设置null并被回收 是

4.在对象传值和传参方面有无问题，对象的clone方法使用是否过度 否

5.是否大量经常地创建临时对象 否

6.是否尽量使用局部对象（堆栈对象) 是

7.在只需要对象引用的地方是否创建了新的对象实例 否

8.方法的参数是否都做了效验 是

9.数组类结构是否做了边界效验 是

10.变量在使用前是否做了初始化 是

11.返回堆对象的引用，不要返回栈对象的引用 是

12.数据库设计或SQL语句是否便于移植（主义与性能会存在冲突） 是

13.数据库资源是否正常关闭和释放 是

14.数据库访问模块是否正确封装，便于管理和提高性能 是

15.是否采用合适的事务隔离级别 是

16.是否采用存储过程以提高性能 否

17.是否采用PreparedStatement以提高性能 是

五.效能：

1在出现海量数据时，队列、表、文件在传输、上载等方面是否会出现问题，是否控制如分配的内存块大小、队列长度等 是

2对Hashtable、Vector等集合类数据结构的选择和设置是否合适 是

3有无滥用String对象的现象 否

4是否采用通用的线程池、对象池等高速缓存技术以提高性能 否

5类的接口是否定义良好，如参数类型等应避免内部转换 是

6是否采用内存或硬盘缓冲机制以提高效率? 是

7并发访问时的应对策略 否

8.I/O方面是否使用了合适的类或采用良好的方法以提高性能（如减少序列化、使用buffer类封装流等) 否

9.同步方法的使用是否得当，是否过度使用? 是

10.递归方法中的迭代次数是否合适(应保证在合理的栈空间范围内) 是

11.如果调用了阻塞方法，是否考虑了保证性能的措施 否

六.可读性：

缺乏大部分注释，可读性较差 否

七.可测试性：

经测试，大部分功能均无较大缺陷 是