**系统的功能需求**

1. 管理员用户可以在学期初选课数据导入，包括班级、教师、课程进程表。
2. 教师在上课开始前或者开始后打开自助考勤功能。
3. 系统根据设置的时间窗口（默认是100分钟）自动关闭自助考勤或请假功能，结束之后不再接收学生提交的考勤和请假请求。
4. 对于早退的情况，上课期间，教师可以随机选择部分学生或者选择全部学生，再次进行考勤，系统通过无线网络向学生手机发出一条考勤消息，收到通知的学生再次进行一次抽点考勤。未参与的学生将被认定为早退。
5. 开始考勤后，学生做上传【特征】的操作，辅助系统完成对“你是你”和“你在课堂”的认定，对认定成功的考勤予以登记，并反馈给参与考勤的学生结果信息，记录考勤发生的时间。
6. 学生可以在考勤窗口关闭之前申请请假并上传假条。
7. 教师可以在上课中，上课期间，上课之后查看最近一次上课的出勤率，请假人数，以及缺勤人员的信息。
8. 教师可以在期末统计每门课每次上课的出勤率、平均出勤率，生成每次上课的出勤情况统计表。输出表格的样式参考学生平时成绩表。
9. 若考勤窗口关闭后，教师可以为某次课程手工增加记录。
10. 系统提供维护当前教师所授课班级学生名单的功能。
11. 教师可以在课下进行对学生请假的审批。后补的假条可以冲抵学生某次的早退、迟到和缺勤记录。

**系统的非功能需求**

1. 系统保持界面简单、使用方便。

2. 简化用户的使用，并减少操作的步骤。

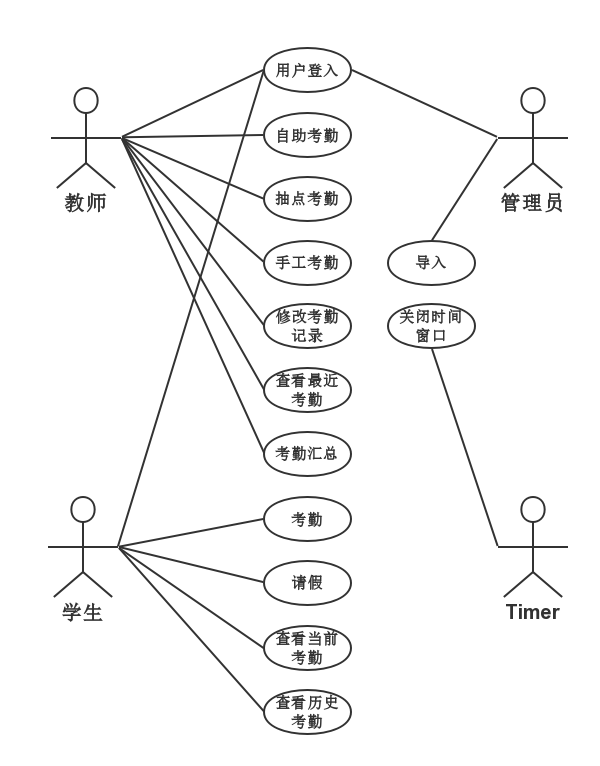
3. 系统保证7\*24小时不间断运行，提高系统的可靠性

4. 系统在使用高峰期间的响应时间少于1s，保证系统的实时性。

5. 避免数据的耦合性。

6. 可以在原来的基础上很容易地修改，增加功能，以及外部系统，以适应新 的需求，具有良好的可扩充性。

**系统用例图**

****

**一、导入教师、班级、选课信息**

**1. 目的**

本用例允许教师将学生信息、课程信息和教师信息导入系统。

**2. 事件流**

**2.1 基本流程**

当管理员需要导入选课数据时，进入系统后台，打开数据导入功能，本用例开始执行。

（1）管理员选择需要导入的文件。

（2）系统打开文件。

（3）系统检查文件的格式，字符集，每个单元格的有效性，以及完整性。

（4）系统应该检查即将导入的数据会不会造成重复，若重复以外部数据为准。

**2.2 备选流程**

（1）若系统无法打开文件，则提示用户重新选择需要被导入的文件。

（2）若系统检查到数据格式有错，则将错误的原因、文件名以及错误的具体位置写入错误日志中，并继续检查导入文件中的数据项是否存在错误，直到文件中所有的数据项都被检查完毕。

（3）对于重复的数据系统应该对其进行去重处理。

**3. 特殊需求**

使用教务系统导出的数据作为外部数据。

**4. 前提条件**

以管理员身份进入系统。

**5. 后置条件**

用例执行成功后，系统中的学生、教师、以及选课信息将会被更新，否则系统信息不变。

**二、用户登陆**

**1. 目的**

本用例允许用户通过微信服务器端直接接入本系统。

**2. 事件流**

**2.1 基本流程**

微信服务器向本系统发出登录请求时，本用例开始执行。

（1）系统根据登录的微信号信息自动匹配内部数据文件匹配登陆者的用户信息。

（2）若匹配到的用户类型是教师，则显示该用户则显示教师名下的所有的课头列表，并且系统为其提供相应操作的功能菜单，其中包括：自助考勤，手动考勤，抽点考勤、维护考勤记录、学生信息维护、随堂统计出勤状况、考勤汇总、批量增加考勤记录、请假审批、维护考勤记录、计算考勤结果。

（3）若是学生，则系统为其提供相应操作的功能菜单，其中包括：自助考勤自助请假、查看最近考勤、查看历史考勤。

（4）若是管理员，则系统为其提供导入功能。

**2.2 备选流程**

若微信号不是在系统内登记的则提示用户登入失败。

**3. 前提条件**

无

**4. 后置条件**

无

**三、自助考勤**

**1. 目的**

本用例允许教师开启自助考勤。

**2. 事件流**

**2.1 基本流程**

（1）教师可以在任意恰当的时间输入相应的课程号，然后系统为其维护一个对应时间窗口。

（2）维护考勤次序表和详细考勤表。详细考勤表中所有学生的考勤状态为空。

（3）若教师以抢占的方式进入时间窗口队列，则合并被抢占的教师抽点考勤随机表，然后其所对应的抽点考勤详细表。

**2.2 备选流程**

由于同一个教师只能在一个时间给一个课头的班级上课，同一个班级也只能在一个时间内听一个老师上课。当系统检测到有逻辑冲突发生时需要分下列情况处理。

（1）两个老师尝试开启考勤的时间在相同上课节次区间内，后来的教师尝试打开时间窗口，系统会提示失败，并提示开启失败的详细原因。

（2）两个老师尝试开启考勤的时间在不同上课节次区间内，后来的教师尝试打开时间窗口时，之前教师的时间窗口自动关闭，然后开启后来的教师的时间窗口。

**3. 前提条件**

（1）教师必须在系统登入成功且输入课程号是应对自己所教的课程。

（2）配置文件setting.ini 中上课节次和时间窗口大小设置完毕。

**4. 后置条件**

若用例执行成功，系统为教师维护一个新的时间窗口，被关闭的时间窗口不再接收任何的考勤和请假请求。

**四、抽点考勤**

**1. 目标**

本用例允许教师开启抽点考勤来完成对学生的考勤。

**2. 事件流**

**2.1 基本流程**

当教师在上课期间开启了自助考勤，一段时间后又想对学生进行随机的抽点考勤再一次了解学生考勤情况，在恰当的时机以不断重复使用此用例。

（1）教师在开启过自助考勤后，系统根据其课程号自动在时间窗口队列中找到自己的时间窗口。

（2）输入抽点人数，系统生成抽点考勤学生列表。初始化随机考勤细节表并提示被抽点到的学生考勤。

（3）维护考勤次序表以及详细考勤表。

**2.2 备选流程**

（1）教师没有开启自助考勤提示失败原因；教师的时间窗口剩余时间小于5分钟提示开启失败原因。

（2）教师输入的抽点数量大于当前课程学生数量、输入格式非法等提示开启失败信息。

（3）若教师多次开启抽点考勤，需要重新初始化随机考勤细节表。

**3. 前提条件**

（1）教师登入系统成功。

（2）当前时间窗口队列有此教师课程头的时间窗口。

**4. 后置条件**

老师开启抽点考勤功能成功之后在教师对应课程头的抽点考勤细节表文件中初始化被抽点的学生的相应记录，若开启失败其内部数据信息不变。

**五、手工考勤**

**1. 目的：**

本用例支持教师在考勤窗口关闭后为某次课程手工增加考勤记录。

**2. 事件流**

**2.1 基本流**

当教师需要手工增加考勤记录时选择系统的手工考勤本用例开始执行。

1） 教师向系统输入需要手工考勤的课程号。

2） 系统读取课程号信息并初始化对应的每一个学生的考勤记录初始化存入一个新的详细考勤表中，默认所有学生的考勤状态是缺勤。

3） 教师根据考勤实际情况输入该学生的考勤状态并提交给系统。

4） 系统检查输入信息是否合法并反馈给教师提交结果，若不通过显示错误信息，此次操作无效。

5）根据教师的需求将学生考勤信息的录入，最后生成考勤细节表并在考勤次序表中追加此课头的考勤信息。

2.2 备选流程

若当前教师存在考勤窗口系统提示发起手动考勤失败。

3. 前置条件

1）当前教师没有开启考勤窗口。

2）当前教师的课头列表信息完备。

4．后置条件

**六、请假审批**

**1. 目标**

本用例允许老师对该自己发起的历史发生考勤记录进行统计并汇总生成考勤汇总表。

**2. 事件流**

**2.1 基本流程**

教师在统计课程考勤记录时遇到学生提交的未认定的假条时本用例开始执行。

（1）系统为教师提供请假图片与请假人的信息，包括学号，考勤次序，提交假条时间。

（2）教师可以“同意请假”、“不同意请假”或者忽略。

（3）在考勤细节表中修改此学生的请假状态并记入考勤汇总表。

**2.2 备选流程**

（1）教师时间窗口开启时执行本用例提示失败信息。

（2）若教师忽略某学生的假条，系统对追加记录到汇总表文件中该学生考勤状态置为空。

（3）若没有请假记录提示没有请假人员。

**3．前置条件**

教师登入系统成功，教师时间窗口关闭，并且至少完成一次自助考勤。

**4. 后置条件**

如果用例执行成功，该次请假的学生请假状态将被更新，追加记录到汇总表文件否则记录不变。

本用例执行成功后生成此课头的考勤细节表并增加此课头的考勤次序数。

**七、教师查看最近考勤**

**1. 目的**

本用例允许教师在开启自动考勤或者完成考勤后查看最近一次的考勤结果。

**2. 事件流**

**2.1 基本流程**

教师登陆后输入课程号，教师打开查看最近考勤用例开始执行。

考勤窗口正在进行或者下课后时，教师打开查看最近考勤状态功能本用例开始执行。

（1）若教师正在考勤，则系统检测相应lea.csv、randomdetail.csv（对于本次的考勤次序有效）、detail.csv中学生的考勤状态，其考勤状态的计算优先级降序排列。

（2) 若教师未进行考勤，系统根据*seq.csv*查找该老师最近一次的考勤状况得到课程号和考勤次序号，检测相应的lea.csv、detail.csv中学生的考勤状态，其考勤状态的计算优先级降序排列。

**2.2 备选流程**

若无考勤记录则提示失败信息。

**3. 前置条件**

课程头有考勤记录。

**4. 后置条件**

显示相应学生的考勤状态信息。

**八、教师考勤总汇**

**1. 目的**

在本用例教师可以在期末对本学期统计汇总表并对每个学生的考勤状态和出勤率进行显示。

**2. 事件流**

**2.1 基本流**

当老师打开考勤状态汇总统计功能时本用例开始执行。

（1）教师输入相应的课程号，系统计算后详细数据后，获取的考勤汇总表信息，并对其中每个学生的考勤状态进行计算。

（2）计算完毕后按课头列表顺序显示此课头下每个学生的考勤信息。并统计每次上课的出勤率、平均出勤率。

**2.2 备选流程**

（1）老师没有完成一次考勤时执行本用例系统将提示错误信息。

（2）若之前存在老师没有统计的考勤细节表系统将会显示错误信息并显示考勤次序号和课程号。

**3. 前置条件**

（1）用户登录的身份为教师

（2）当前存在至少一次完整的考勤细节表。

（3）假条已审批

**4. 后置条件**

系统自动对所有的考勤汇总表进行统计，显示，并对没有汇总表的考勤细节表生成汇总表。

**九、学生考勤**

**1. 目的**

本用例允许学生自助考勤、抽点考勤。

**2. 事件流**

**2.1 基本流**

教师开启考勤后，学生选择考勤来执行此用例，此用例自动实现自助考勤和抽点考勤的识别并记录详细的学生考勤结果。

（1）系统根据学号在教室课程头时间窗口“过一遍”，检查时间窗口内的课程头号是否有自己学号所对应的全部课程号的一个。若系统没找到属于自己的课程头，将提示学生不在考勤时间。

（2）根据对应课程的抽点考勤详细表文件是否需要抽点考勤，若需要则最后的考勤记录信息将保存在抽点考勤详细表文件中。否则保持在对应课程的详细表中。

（3）学生提交自己相应的特征信息给系统。

（4）系统认定“你是你”和“你在这”，并记录认定的结果（包含成功和失败的情况）及发生时间。并写入详细考勤数据表中。

**2.2 备选流程**

（1）若时间窗口不存在学生对应的课程的课程头号，则无法参与考勤或者抽点考勤，并提示学生未到考勤时间。

（2）若学生的考勤证据路径被系统确定为失败则减少一次此学生上传信息的次数，并提示失败信息和剩余上传次数。

（3）若学生用完了上传次数则无法提交考勤信息。

（4）若学生已经完成了考勤再次向系统提交考勤证据路径，系统将忽略他的考勤证据路径并提示对应的错误信息。

**3.前提条件**

（1）微信号登入系统，且系统识别的信息是学生。

（2）时间窗口有学生对应的课程号存在。

**4. 后置条件**

用例执行成功后，系统中的考勤信息将会被更新。

**十、学生请假**

**1. 目的**

本用例允许学生将请假条上传到系统。

**2. 事件流**

**2.1 基本流程**

当学生选择系统的请假功能时本用例开始执行。

（1) 系统根据学生的个人信息判断当前是否存在满足要求的课头列表的考勤窗口。

（2) 若存在则提示学生提交请假证据。

（3) 学生向系统提交请假证明并记录上传时间，向请假表文件中追加请假信息，考勤状态置为请假。

**2.2 备选流程**

（1）当前没有满足要求的考勤窗口或者说没有任何一个老师发起与他有关的考勤窗口则提示错误信息并忽略请假要求。

（2）若当前此学生已经提交过一次假条则提示是否继续，此时学生可以帮助同学提交假条。

**3. 前置条件**

时间窗口中有对应的课程号存在。

**4. 后置条件**

若学生请假成功后，将相应的数据追加至请假数据表。

**十一、学生查看当前考勤**

**1. 目的**

本用例允许学生查看正在参与考勤的实时考勤结果。

**2. 事件流**

**2.1 基本流**

学生完成自助考勤后打开查看当前考勤状态功能本用例开始执行。

（1）系统根据学生查看考勤状态的时间从考勤次序表中获取与该学生有关的最近一次的教师号，课程号和考勤次序号。

（2）若系统检测相应请假表文件中出现了此学生的请假状态，则直接显示请假。若系统检测到随机考勤详细表中有相应的有效数据，则显示对应的考勤结果状态。最后再检测相应的详细表文件中的考勤状态。

**2.2 备选流程**

若学生没有完成考勤认证则显示考勤状态为未参与考勤，并提示该学生尽快向系统提交自己的考勤特征信息。

**3. 前置条件**

时间窗口中有对应的课程号存在。

**4. 后置条件**

无

**十二、学生查看历史考勤**

**1. 目的**

本用例允许学生查看所有课程下该学生的出勤状态。

**2. 事件流**

**2.1 基本流**

当学生打开历史考勤状态汇总统计功能时本用例开始执行。

（1）系统获取学生课程内所有的课头列表的信息。

（2）系统挨个获取每个课头列表的考勤汇总表信息，并对每个课头中考勤状态进计算。

（3）计算完毕后按课头列表顺序显示课程下每一次考勤该学生的考勤结果，最后显示课程下该学生的出勤率。

**2.2 备选流程**

（1）学生没有完成一次考勤时执行本用例系统将提示错误信息。

（2）若之前存在老师没有统计的考勤细节表系统，则向学生显示此教师没有对考勤记录进行统计，无法显示。

**3. 前置条件**

已有详细考勤记录

**4. 后置条件**

无

**十三、计算考勤结果**

**1. 目的**

对出勤情况进行统计。

**2. 事件流**

**2.1 基本流**

当教师点击查看考勤汇总时本用例开始执行

（1）为每次考勤生成相应的汇总数据列

（2）对于没有出现请假的某次考勤和教师全部确认过的考勤，系统自动进行计算。

（3）对于出现请假的考勤，但教师未全部确认的考勤，系统帮助生成sum表的占位列（全空）。不填写考勤状态。直至教师确认假条。

**2.2 备选流程**

（1）学生没有完成一次考勤时执行本用例系统将提示错误信息。

（2）若之前存在老师没有统计的考勤细节表系统，则向学生显示此教师没有对考勤记录进行统计，无法显示。

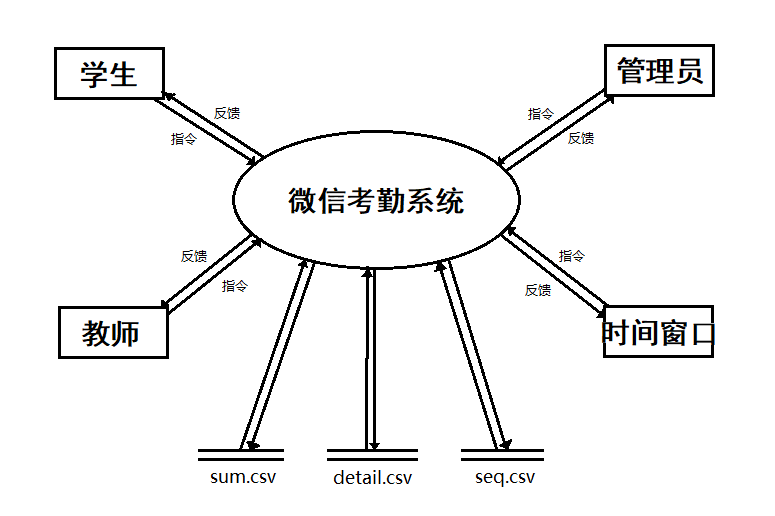
**3. 前置条件**

已有详细考勤记录

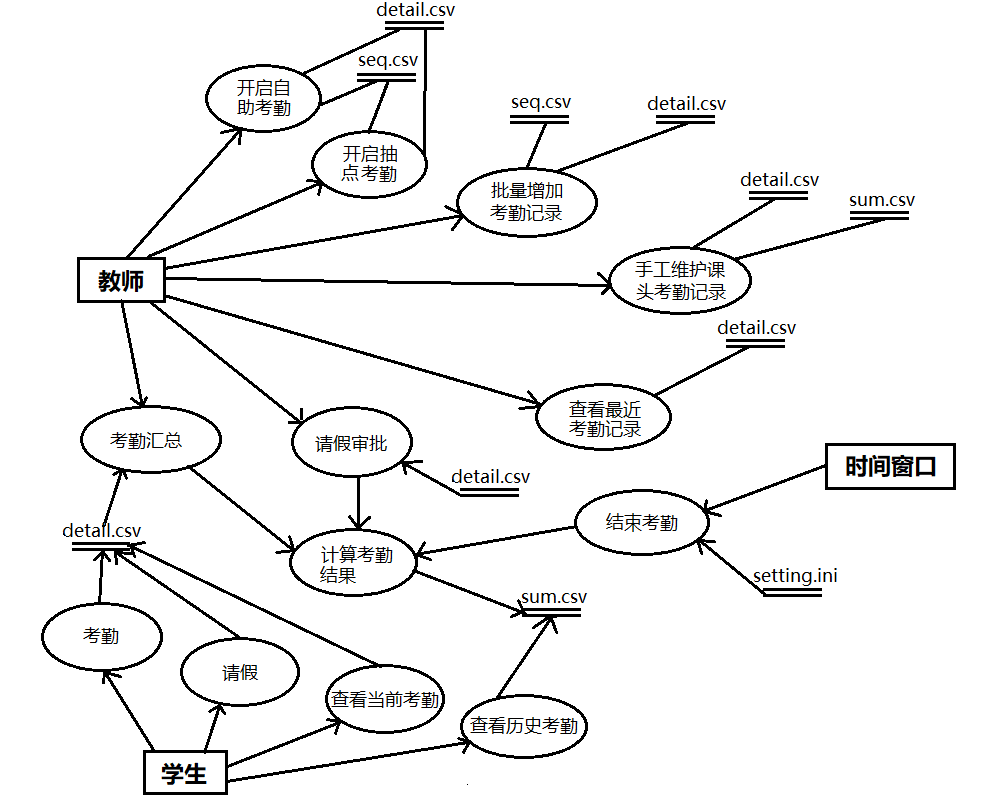
**4. 后置条件**

无

**系统数据流图**

****

顶层数据流图

****

一层数据流图

**4.1 系统架构设计**

本系统的架构分为三个层次。

1. 数据层

提供对原始文件的增删查改、创建以及删除文件，并对业务逻辑层提供接口，负责对数据的基础操作，不负责数据的逻辑操作。每个模块只负责完整的把数据读取或者写入到文本而不管文本信息是否合法，业务逻辑层通过调用该接口请求本层服务。

主要是对非原始数据（数据库或者文本文件等存放数据的形式）的操作层，而不是指原始数据，也就是说，是对数据库的操作，而不是数据，具体为业务逻辑层或表示层提供数据服务。

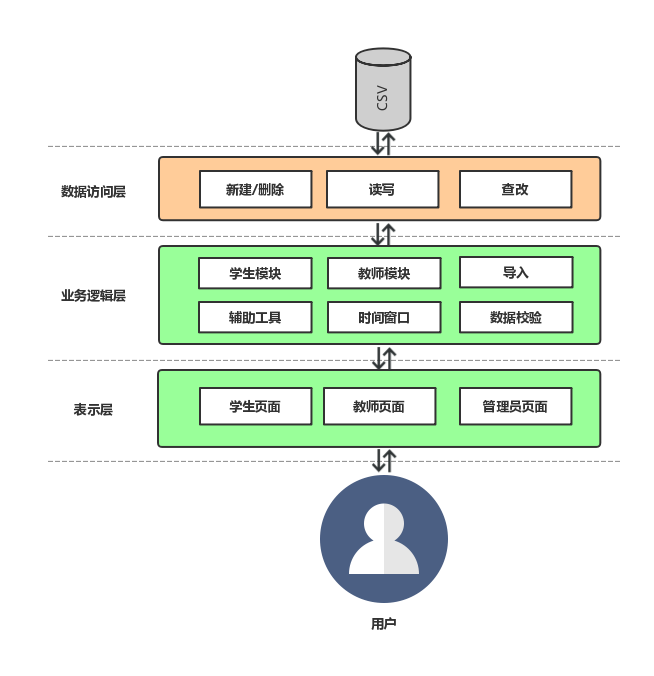
2. 业务逻辑层

主要是针对具体的问题的操作，也可以理解成对数据层的操作，对数据业务逻辑处理，如果说数据层是积木，那逻辑层就是对这些积木的搭建。

业务逻辑层向上层提供业务服务和业务逻辑服务。此层负责提供对数据的校验，更新，显示。按功能分为以下模块：学生模块、教师模块、辅助、时间窗口、导入、去重以及数据校验模块。上层的用户界面层提供输入参数调用本层接口，本层调用下层的服务接口对信息进行查询或者核对满足要求后提供对应的服务，并更新内部文件。

3. 用户界面层

为用户直接提供可视的功能，可向系统提供输入数据，调用下层的业务层提供功能支持，本层不负责对数据的验证和查询而是传递参数调用下层的接口，并显示返回的信息。



**4.2 总体实现方案**

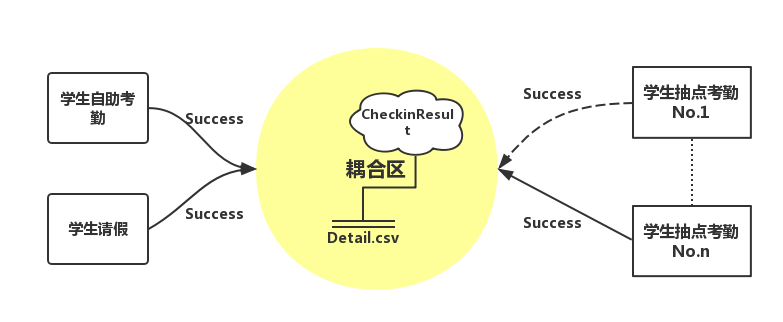
**4.2.1 学生考勤状态判定以及考勤状态耦合的解决方案**

**1. 考勤状态耦合**

由于本系统为采用数据库而是csv文件，因此详细说明本系统可能产生的数据耦合并且需要手动解决，给出一种合理或折中的解决方案有助于更好地提高整个系统的稳定性与简便性。

本系统数据耦合的产生原因：

由于内部数据文件中仅仅存在Detail.csv，那么学生在抽点考勤、自助考勤、多次的抽点考勤以及请假的时候，由于这些操作对detail.csv文件是共享的，这些操作混在一起避免不了会产生对考勤结果的耦合性，造成数据的错误，并且如果解决类似这样的错误需要付出很大的代价。



解决方案：

额外添加两个数据文件。

（1）lea.csv

学生使用请假功能，并提交自己或同学的假条，当系统识别成功后，会反馈给学生请假信息，并将识别后的请假信息以及相关信息写入lea.csv中。

学生在查看当前考勤时，不管detail.csv文件是什么状态，只要lea.csv有此学生的请假记录且处于当前开启的时间窗口内，则显示状态为请假。

（2）randomdetail.csv

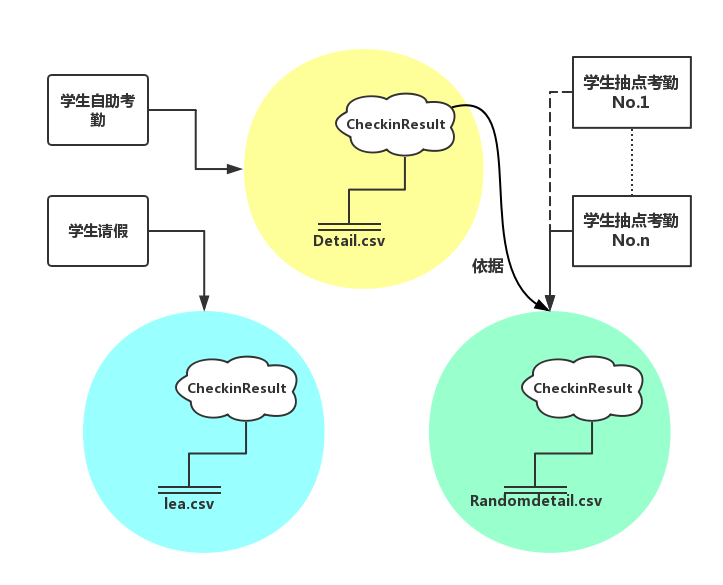
Randomdetail.csv的格式为teacherid\_courseid\_randomdetail.csv，由其命名的规范可知，此抽点记录是一次性的：

Case1： 当教师开启抽点考勤，先清空文件里的内容，然后再根据生成被抽点的学生名单初始化至此文件中。

Case2： 当教师多次开启抽点考勤时，先将此时的Randomdetail.csv的所有内容直接覆盖至detail.csv文件中，然后再根据生成被抽点的学生名单初始化至此文件中。

Case3： 当教师时间窗口正常结束出队时，那么系统会先将此时的Randomdetail.csv文件中的所有内容直接覆盖至detail.csv文件中。

Case4： 当教师时间窗口被抢占而出队的时候，也执行Case3后面的步骤。



依据：当学生进行考勤时，系统的认证模块，需要依据detail.csv的考勤状态计算考勤结果，在下文有更详细说明。

**2. 考勤有效时间界定机制**

有效时间是：教师成功开启自助考勤或抽点考勤后，若学生签到的时间是在其有效时间范围内，学生成功考勤认证后考勤的状态可由缺勤变为出勤，但不在其有效时间范围内，则变为迟到。

计时时机：开启自助考勤，开启抽点考勤，重复开启抽点考勤

本系统把有效时间：60s

**3. 考勤结果算法**

一旦学生提交了考勤特征信息且系统反馈结果是认证成功，学生不可重复参与此次的考勤，除非有新的一轮考勤开启（新的自助考勤或新的抽点考勤）。

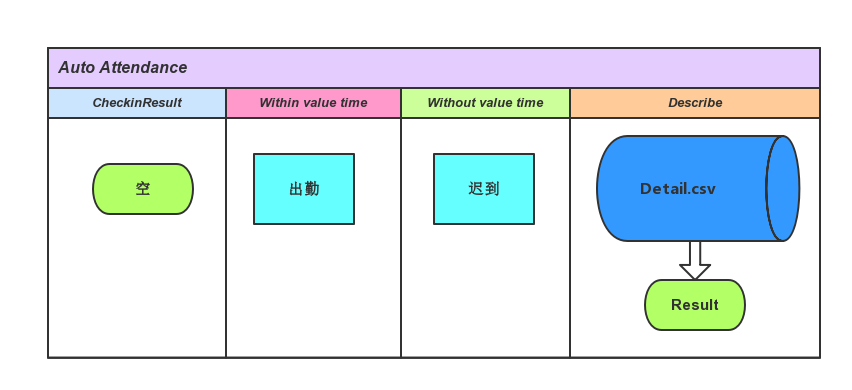
因此，只有表中的（抽点考勤详细表或考勤详细表）考勤结果状态为空时，才可以提交考勤特征信息，这其中包括了学生提交了考勤特征信息但是系统反馈的结果是认证失败不会影响考勤结果的状态。

**3.1 自助考勤**

需要注意的是，Detail.csv与randomdetail.csv的考勤状态数据项的默认值是空，而不是缺勤。这样做的效果是利大于弊的，“空”考勤状态意味着对应学生未参与或考勤结果认证失败，缺勤就是缺勤状态且不可混为一谈。

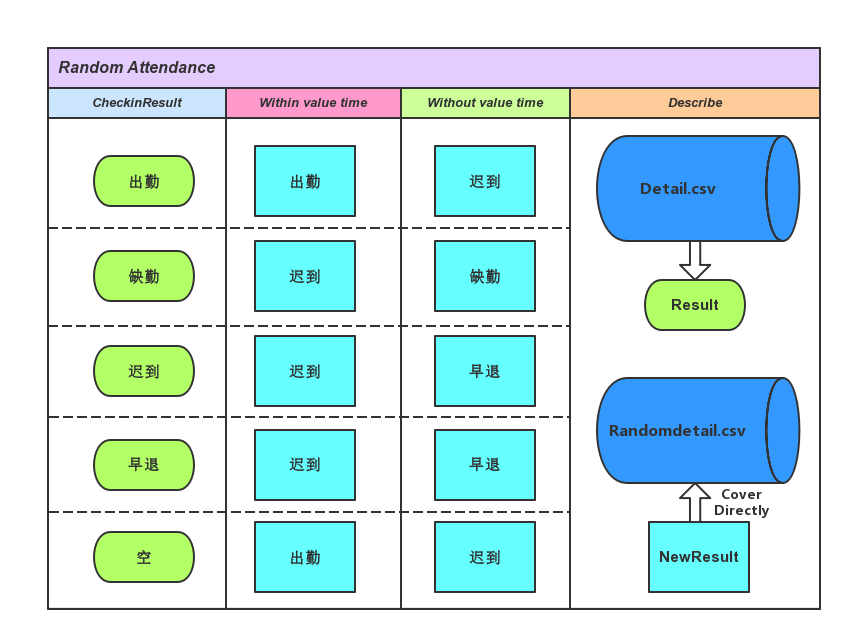
在自助考勤开启后，一旦学生提交并获得系统的成功认证反馈，参与考勤成功，即detail文件中的考勤状态不为空，则不允许学生再次提交考勤信息，但允许学生请假。

特别地，学生在自助考勤期间进行请假且认证成功之后，系统不允许该学生参与考勤，不允许该学生再次请假，并且且此学生不会出现在以后的抽点考勤生成的被抽点到的名单之中。



**3.2 抽点考勤**

若学生提交的信息特征认证成功，首先根据detail.csv相应的学生考勤结果的状态，再判断是否在有效时间范围内，则产生本次考勤的考勤状态并写入Randomdetail.csv相应的位置中。具体解决方案如下：



额外需要注意的是，由于学生参与考勤并且成功认证后，不允许学生重复参与考勤。例如当前的考勤状态以及是出勤，系统不允许学生再次提交，但是当教师开启抽点考勤后，为了避免因抽点考勤造成的数据的耦合性，系统会此次操作而初始化ramdomdetail.csv文件，当学生。

抽点考勤详细表（Randomdetail.csv）清空时机：

由于此抽点考勤详细表中的内容是一次性的，因此使用之前需要对此表进行清空使内容无效才能传入新的抽点名单来进行初始化。

（1）合并数据时。

发生合并数据的情况有：教师的TCB正常出队，教师的TCB被强占而出队。

每当教师开启一次自助考勤时，就清空抽点考勤详细表中的内容。即使出现了被抢占的教师不管开启过多少此抽点考勤，都能保证新来的教师能够正常使用当前的抽点考勤详细表。

教师开启自助考勤以及教师正常或者被抢占而退出时间窗口队列时，进行清空抽点考勤详细表中的数据之前需要合并（merge）数据。因为考虑到教师被抢占时，需要将被抢占的教师当前的抽点考勤详细表（Randomdetail.csv）中的数据合并至与教师相应的考勤详细表中才算完成此教师被强占而退出时间窗口队列，然后新的教师入队即可，同时清空此表。

（2）教师开启抽点考勤时。

当教师第一次开启考勤或重复开启抽点考勤时，新的抽点名单向此表中初始化覆盖（cover）来抽点考勤，相当于一次清空（clear）操作。

**4. 优先级**

作用：

（1）制约学生重复或无效的操作

Case1： 当学生提交假条至系统中，则向lea.csv文件添加相应的数据。这时学生不能参与自助考勤，同时在教师开启抽点考勤时，此学生在其随机产生被抽点学生的名单中不会出现，因此也不会参与教师发起的抽点考勤。允许学生假条的重复上传（考虑到帮同学提交假条）。

Case2： 当学生参与抽点考勤并且已经有相应的考勤数据写入抽点详细表中，那么不允许该学生再次参与此次的抽点考勤，但是允许学生参与请假（只要你拥有有效的假条，同时可以帮别人请假）。

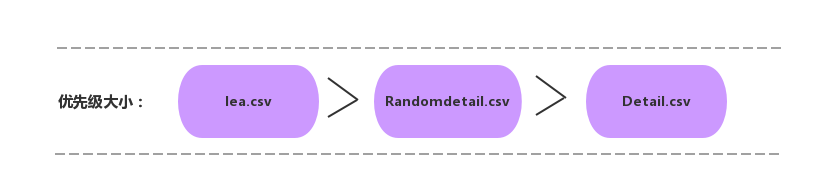
Case3： 当学生参与自助考勤并且已经有相应的考勤数据写入详细表中，那么不允许该学生再次进行自助考勤。

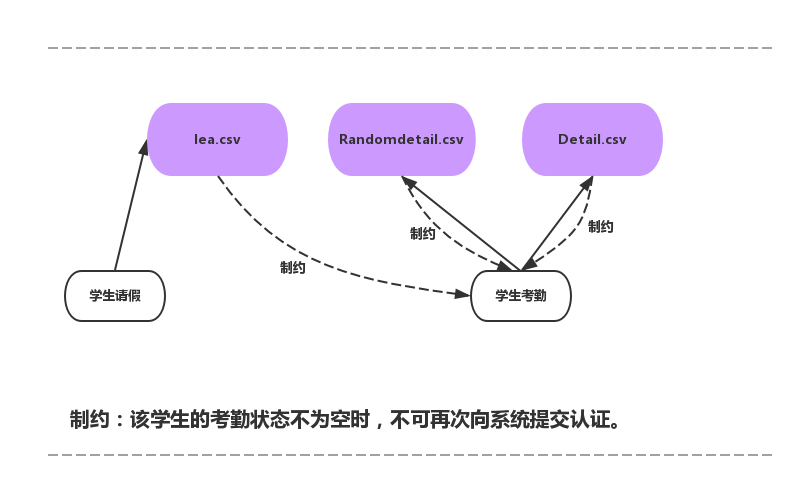
（2）优先提取考勤状态

当教师、学生查询最近的考勤状况时，系统根据优先级大小即从lea.csv到detail.csv文件依次查询考勤状态信息提取出相应的考勤状态。

（3）合并考勤结果

当教师重复开启抽点考勤、考勤正常结束或被抢占并且此时的随机考勤详细表中有有效的数据的时候，根据随机考勤详细表中所有学生的考勤信息将考勤详细表（detail.csv)中对应学生的数据全部覆盖。最后清空抽点考勤详细表中的内容。





关于请假：考虑到学生可以帮助其他同学请假，因此系统不制约学生请假，只要在对应的课头依然存在于时间窗口队列中。

1. 格式化检验（format check）

在此模块中，本系统基于正则表达式的格式化检验，其主要的特点是灵活性强、简单，可以迅速地用极简单的方式实现数据项字符串的复杂的控制与约束。正则表达式在本系统主要的应用对象是数据项文本内容，包括微信号、姓名、课程名、教工号、课程号以及文件名等。

需要对这些数据项内容设定相应的正则表达式来实现对其约束来防止各种的格式错误。

先从文件中提取出未经任何处理的数据并且将其存在一个专门的临时列表中，通过以“住行跨列”的遍历方式。当系统遍历到某行时，需要对该行的各类型的数据项进行正则匹配。

若匹配的结果不满足正则表达式，则将错误原因、位置以及出现错误的文件位置写入错误日志列表中。重复执行上述操作，直到遍历完整个列表。最后只需要将临时列表与错误列表中的数据分别存入相应的输出文件中。

2. 去重

2.1 set

set是python中一个无序且不重复的元素集合。其集合对象是一组无序排列的可哈希的值，具有不可为集合创建索引或执行切片（slice)的操作，其主要特性是不重复以及元素为不可变对象。

2.2 主键

主键是表中的一个或多个字段，它的值用于唯一地标识表中的某一条记录，并且它可以唯一地确定表中的一行数据，或者可以唯一确定一个实体。由于本系统采用的是否csv文件，需要手动确定主键。

2.3 确定主键

学生信息文件：

主键：StuID与WechatID

分析：学号与微信号都具有唯一性并且应“分别决定”行信息。例如，当多次出现相同的学号时，然而其微信号各不相同。或者出现多次相同的微信号然而学号都各不相同。因此学生信息文件中共有两个逻辑上独立的主键，既要去除抽重复的学号又要去除重复的微信号，因为一个微信号只能对应一个学号。

教师信息文件：

主键：TeacherID与WechatID

分析：同上，也是以“分布决定”的方式去重。

课程信息文件：

主键：CourseID与ClassNums

分析：与上述两信息文件不同的是，此处课程信息文件的去重方式采用了“共同决定”的方式处理行信息，即允许同时出现多个课程号或班级名，但是决不允许出现相同的课程号合班级区间组合，若出现即重复。

2.4 逻辑

（1）“分别决定”算法

先从文件中提取出未经任何处理的数据并且将其存在一个专门的临时列表中，通过使用python中的set集合，将临时列表中的“StuID”合“TeacherID”列放入set集合中，

记作set{“StuID”}和set{“TeacherID”}，把set{“StuID”}和set{“TeacherID”}存放至一个temp列表中，通过临时列表中的数据将temp剩余项填满。再次使用set集合，set{“WechatID”}，将其内容存入一个temp2列表中，再次通过临时列表中的数据将temp2剩余数据项填满。

（2）“共同决定”算法

先从文件中提取出未经任何处理的数据并且将其存在一个专门的临时列表中，再逐行地将每行中的课程号与班级名组合成列表并添加至set集合中，重复上述操作，直到执行到最后一行。对set集合中的数据设法存入一个temp列表中，通过临时列表中的数据将temp剩余数据项填满。

（3）日志

在进行“分别决定”或“共同决定”算法时，当临时列表中的数据为temp剩余项一次补充完成填满操作时，若存在某行数据未进行补充（或出现了重复），则写入日志，内容包括文件路径，重复错误以及位置。说明：日志写入之前，可先定义一个log临时列表，来暂时存放相应数据，最后去重结束时，再一并将其内容添加至系统内部的日志文件中，减少了I/O操作。

3. 提取班级区间

逻辑：

将外部数据的课程安排中的班级区间数据项提取出来后存入至一个临时列表中，并且需要以逗号为分割符。例如，当此数据是“软件工程1601-软件工程1603，计科1601”时，

存入至列表后的数据是[[软件工程1601-软件工程1603]，[计科1601]]，并且对列表中的数据项逐个处理，即当列表的列表中的数据出现“-”时，则提取“-”前的后4位数字以及“-”所在

列表中最后4位数字位作差得到对应的课程数目，依次以该课程数目位依据扩展班级区间，将内容存在新的列表中，若列表中的数据没有出现“-”时，直接向其内容存入新的列表中。

重复上述过程，直到遍历完整个列表。需要注意的是，分割符可以根据实际情况来更改（默认是逗号）

需要补充的是，列表中的列表对应着唯一的索引号，每个索引号都对应着外源数据的行号。当在写入新列表的过程中（隐藏班级名称），将以内部课程文件的格式规范来补充剩余的行数据。

时间窗口模块

1. 数据结构：

1.1 TCB（Timer Control Block）

教师开启考勤时，本系统就新建一个TCB结果，之后他就常驻时间窗口队列中，当此教师的考勤正常结束或者被抢占时删除（退出时间窗口队列）。系统通过其TCB了解教师考勤的现行状态信息，以便对其进行控制和管理。

PCB包含的内容：

教师微信号，课程号，开启考勤时所在的上课区间，班级列表，修改位。

1）开启考勤时所在的上课区间

当教师成功开启自助考勤时，根据系统当前时间以及配置文件（setting.ini）中的上课区间，匹配对应的上课区间（本节课开始时间，本节课结束时间）。

2）班级列表

根据传入参数的课程号以及内部文件，获取此课程号对应的班级列表。

3）修改位

考虑到特殊情况，当教师已经存在正在队列时，自己将其强占，使其出队，此事操作系统的挂起队列存在两个与此教室相关的timer计时器，第一个时间到的计时器会让后来的时间窗口提前出队。

为力避免这种情况，本系统在TCB中引入了一个修改位（默认是0），即当新来的TCB是以抢占的方式进入时间窗口队列，则将新TCB中的修改位置为1。  
因此在出队函数添加一个判断，首先根据wechatid获取要当前时间窗口的对应TCB的修改位，若位1，则发生抢占（被抢着的TCB已经出队），此函数不再执行任何操作。

当发生新来的教师又被抢占的情况，新来的TCB的持续时间总是大于任何一个旧的TCB的持续时间，只要新来的TCB是以抢占的方式入队，修改位置1。旧的TCB维护的timer计时器不会进行任何操作。除非修改位为0，出现这中情况，说明旧的TCB维护的时间窗口已经正常出队了。  
 1.2 Timer\_queque

用来存放TCB的时间窗口队列。

2. 出队函数（仅出队）  
timer计时器，在规定的时间（默认是45\*2+10）执行出队来结束此教室的考勤时间窗口。  
好处：  
（1）操作系统通过维护挂起队列帮助系统维护了时间窗口，而不用自己去实现况且效率不如操作系统。  
（2）即使再多的timer在挂起队列，也不占用cpu。