后台课文打分模块

基本功能：

1、按照课文获取对应录音功能

2、基于1考虑，在课文列表的课文后追加查看课文学生录音的入口

3、进入入口后，展现已经有过在该课文中上传录音记录的学生列表。

4、学生列表分待评分列表及已评分列表两部分

两部分列表均包括以下字段

-———学生名-班级-组别-上传时间 -评分（已评分列表有 ） （按完成时间升序进行排序）

5、点击学生进入评分页面

6、评分页面

学生姓名、班级、组别、完成排名、录音、录音打分板、评语

提供删除学生录音的功能，以便不合要求的学生再次提交

页面逻辑描述：

1. 进入课文列表
2. 进入某课文下未评分录音表
3. 进入录音详情页，对录音进行评分
4. 点击保存
5. 保存后可点击下一个继续对下一个录音进行评分，或者返回录音表重新选择录音进行评分
6. 进入已评分录音表可进行编辑

编辑内容包括

修改评分、评语；删除录音记录

~~后台的打分模块你倾向于哪一种打分机制呀？~~

1. ~~给朗读、背诵分别打分，并由系统生成平均分~~
2. ~~直接对两种录音同时打分（需要学生完成朗读和背诵后才能进行）~~

后台打分分数描述倾向

1. 中文描述。例如。还需努力。一般般。中等偏上等等。（这种情况下，在存储过程中以五分制的形式存储）
2. 十分制 0~10整型参数
3. 百分制 0-100整型参数

附加评语（非必输项）：

1. 长度限定200
2. 可选取模板（模板直接从数据库中获取）

**微信端录音打分模块**

1. 基础功能
2. 个人评分记录列表（依据课程名称分行）

列表包含内容： 课程名称、完成时间、评分、评语

1. 名次列表页面：

包含内容排名：前十名学生的名字、分数、完成时间、前**五**名学生公开录音，可供学习

排名规则：依据评分及完成时间进行排序

1. 页面逻辑
2. 查看课程列表时，右划进入个人评分记录列表。
3. 点击个人评分记录，可查看该课程下的名次排行
4. 点击前三名学生右侧的播放按钮，可对对应学生的音频进行播放（支持拖曳进度条）

太棒啦，10分都不足以表达我的敬佩之情！;语音语调棒棒哒，但断句的节奏可以再练习一下哈！;每个单词都读得比较准确，但语调和节奏再模仿练习一下哈！;个别单词发音不是很准确哦，多听多读你一定会变得更棒！;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课文名 | 班级 | 下发状态 | 完成人数/总人数 | 待批改人数 | 分数分布 (A:8-10 P:4-7 E:0-3 ) | 操作 |
| lession name | 101 | 已下发 | 30/40 | 24 | A-1；P-3；E-2 | 批改 |
| 102 | 待下发 | 0 | 0 | 无 | 已全部批改完成 |
| 103 | 已下发 | 41人全部完成 | 0 | A-1；P-37；E-3 | 批改 |

**查询所有班级列表包含信息（班级名 班级id） 根据班级id查询lession\_class\_bind 表下发时间，及课文id。根据班级id查询user表班级总人数** ，**根据课文id 用户id列表（通过班级id查询user表获得用户id）查询存在于studio\_info 中录音的记录条数获得完成人数**。根据课文id 用户id列表（通过班级id查询user表获得用户id）查询studio\_info 中pointer is null 的记录条数，获得代批改人数。根据课文id 用户id列表（通过班级id查询user表获得用户id）查询studio\_info 中pointer is not null 并且point 在8-10之间的记录条数，获得A分布人数。查询studio\_info 中pointer is not null 并且point 在4~7之间的记录条数，获得P分布人数。查询studio\_info 中pointer is not null 并且point 在0~3之间的记录条数，获得E分布人数

SELECT

ci.NAME as class\_name,

ci.id as class\_id,

lcb.create\_time,

lcb.lession\_id as lession\_id,

(select count(1) from user where class\_id=lcb.class\_id and del\_flag='0') as total\_num,

(select count(1) from studio\_info where lession\_id=lcb.lession\_id and del\_flag='0' and user\_id in (select id from user where class\_id=lcb.class\_id and del\_flag='0')) as complete\_num,

(select count(1) from studio\_info where lession\_id=lcb.lession\_id and del\_flag='0' and pointer is not null and user\_id in (select id from user where class\_id=lcb.class\_id and del\_flag='0')) as pointed\_num,

(select count(1) from studio\_info where lession\_id=lcb.lession\_id and del\_flag='0' and pointer is not null and point >=8 and point <= 10 and user\_id in (select id from user where class\_id=lcb.class\_id and del\_flag='0')) as num\_a,

(select count(1) from studio\_info where lession\_id=lcb.lession\_id and del\_flag='0' and pointer is not null and point >=4 and point <= 7 and user\_id in (select id from user where class\_id=lcb.class\_id and del\_flag='0')) as num\_p,

(select count(1) from studio\_info where lession\_id=lcb.lession\_id and del\_flag='0' and pointer is not null and point >=0 and point <= 3 and user\_id in (select id from user where class\_id=lcb.class\_id and del\_flag='0')) as num\_e

FROM

lession\_class\_bind lcb

JOIN class\_info ci ON lcb.class\_id = ci.id and lcb.del\_flag='0' order by lession\_id