**【这是一个技术文档】**

较之前的版本，前端文件里几个重大bug已修复，删减了大量没有必要的函数和变量，规范了文件和变量名称，增加了几个接口，同时利用cookies现在无论从哪个页面开始访问，浏览器都将正确地跳转到我们所希望的页面上（除了未答题的学生直接访问score页面会显示得分为-1这种情况外），目前所有功能都在**win7 +** **Dreamweaver CS6本地站点 + 360se chrome + xampp并脱离数据库（用php数组来模拟）**的情况下测试，基本达到预期效果，现对各个页面做简单说明，红色部分表示未完成的功能，所有页面的css、js、php均为与其同名的相应文件。注意，**为保证前端能正确解析后端发来的json对象，不管后端怎样处理数据，最终的返回形式请尽量按照现有的php中的返回格式来，否则前端可能需要修改大量代码。**

①Index.html 登陆界面

功能：-通过cookies验证是否已经登录，若已经登录则部分更新至相应界面

流程：

登录按钮初步检测一卡通号位数的合法性、密码是否填写

向服务器post字段id、pw，服务器返回一个json对象，格式见.php，**各键值类型均为文本型**

返回值的情况说明：（1）right：-1 未找到该id或者密码错误，弹出对话框

（2）right：0 score：-1 未答题的学生，局部更新至注意事项

（3）right：0 score：（非-1） 已答题的学生，载入score.html

（4）right：1 管理员，局部更新至管理页面

[注：以上只要不是情况（1）本地即会记录用户名、账号、密码、权限、分数cookies；文件夹内新建文本文档提供了三个用作测试的账号，分别对应以上2-4的情况]

局部更新页面说明：

1. 注意事项：包含退出按钮，登出；按钮开始答题，载入answersheet.html；按钮修改密码，载入register.html

（2）管理页面：包含退出按钮，登出；按钮注册，载入register.html；按钮下载学生成绩，建议服务器能导出一份以管理员id为名称的excel表格以供下载（仅作建议，因为直接在页面上显示学生成绩意义不大操作起来也麻烦，此处最终需要后端来完成）

②answersheet.html 答题界面

功能：-载入时清空所有本地已存在的答案cookies

-通过cookies检验当前答题用户是否合法，不合法（非学生或已经答过题）的登出

-倒计时并确保了在弹出对话框时计时实际仍在进行

-刷新，中途退出，前进后退都将退回到登录界面并重新答题，重新答题后试卷会被更换，时间会被重置

-倒计时结束或者按提交按钮后前端判分（提交分数部分目前通过修改本地php文件的数组键值来模拟用户信息更改，得分页面的分数目前为读取cookies获得，这个最后最好从服务端获取数据）

流程：

向服务器post请求，随机获取一套预先组好的卷子并在前端打乱题目顺序并保存每题的答案密文cookies，服务端需要返回一个json对象**数组**来传送试卷，格式见.php（其中的json生成函数在低版本php中不可用，后端视需求修改，正确答案需要加密），各键值类型均为文本型。

提交后，前端根据cookies和解密函数进行判分，score连同cookies中的id,pw一同传给服务器，后端进行身份验证后修改数据库中对应学生的分数，完成后**需返回一个json对象，除了基本身份信息（如果成功的话）外需要额外包含success：true/false 信息表示是否成功**

前端得到true后更新cookies并载入score.html（score的分数部分最好改为从服务端获取数据再更新cookies），若获得false则报错登出（目前假设情况是cookies被篡改，身份验证失败，也有可能有其他原因，需要异常处理）

③score.html 得分界面

功能：-倒计时结束或者按确定后回到登录界面

-载入该界面时其实已视作登出（cookies被清除），刷新将回到登录界面

-通过cookies检验当前用户是否合法，不合法（非学生或者未答题的学生（需要在完善分数提交功能后再来写））的登出

流程：

载入时根据cookies，**只发送id给后端**，后端返回一个json对象，格式见.php，各键值类型均为文本型，文本局部更新为得分。

④register.html 注册界面

功能：-根据cookies中的right值决定载入注册界面还是密码修改界面

-通过cookies检验当前用户是否合法，不合法（没有cookies而直接访问该界面）的登出

-确认按钮初步判定信息是否填写以及合法性，取消按钮返回到登录界面

-信息填写基本正确后，将表单信息提交给服务端，服务端成功处理后返回至之前的界面并更新cookies（建议注册学生信息时，若数据库中未存在该学生则创建一个right：0 score：-1的成员，若存在则删去后重建，以应对考试时有学生忘记了自己密码的情况）

⑤可能的缺陷：

1. ③中流程若cookies被意外篡改则可能返回其他人的信息，安全起见后端可以在读取分数时再进行一次身份验证（我觉得可能并不是特别有必要）
2. 修改密码时无法对新密码作出很好的条件约束（需要用到正则表达式？不是很懂）
3. 无法验证管理员将注册信息发给后台时格式是否正确，后台能否正确解析未知
4. 答题页面在弹出对话框但没有按确定或者取消期间如果答题时间已结束，页面仍会停留在答题界面（对话框弹出时所有函数都会被中止，这个好像是js的先天特性）
5. 界面外观在不同浏览器和不同分辨率下的表现未知，有待测试
6. 答题页面的布局写得比较怂（手动很多div标签）但是没有想到更好更方便的写法，应该没有太大关系
7. 一些提示（注意事项等）还没有写上，写上后页面可能需要重新排版（我不知道要写什么）
8. 分数提交时可能的错误情况未知（问题应该只可能出在服务端），如果报错合适的处理方式无法确定
9. 向后端发送数据的所有接口都只考虑了成功返回的情况，安全性和稳定性没有保证
10. 已经完成答题的同学不能修改密码
11. 注册页面和密码修改页面表单栏没有相应的美术资源可以用，所以显得有点丑
12. 写前端的那个家伙觉得差不多了就不想干了，后端辛苦你了，蛤蛤蛤（还有其他事情要忙，拜托你了）

—— 来自你的前端

2016.7.30