网页数学竞赛测试文档

1. 概述
   1. 产品简介

本次开发是基于web的单页面应用，目的是实现在不同Client端同步回答问题、对答案进行自动判别、同步刷新成绩，并实现历史数据的持久化存储。

* 1. 测试范围

本次测试计划是针对需求分析中规定的内容进行测试，包括

1. GUI测试
2. 用户登录模块的功能实现
3. 用户答题模块的功能实现
4. 用户历史记录的功能实现
   1. 测试目的

通过测试，达到以下目的：

* 测试已实现的产品是否达到设计的要求，包括：各个功能点是否已实现，业务流程是否正确
* 产品规定的操作和运行稳定
* 检测出可能存在的Bug

1. 测试策略
   1. 测试种类

计划完成以下类型的测试：

* 单元测试（功能测试）
* 集成测试（业务逻辑测试）
* 用户界面测试
* 压力测试
  1. 测试方法及标准

1. 功能测试

这里仅为测试重点的描述

1. 登录功能

* 登录&未登录的检测功能是否实现
* 登录界面的输入过滤功能是否实现
* 登录成功后界面更改是否正确

1. 出题功能

* 出题界面是否符合要求
* 题目获取是否成功

1. 答案提交和判断功能
2. 答案提交功能是否实现
3. 答案判断功能是否实现
4. 答题超时功能

* 超时后是否有提醒
* 超时提醒后是否更新下一道题

1. 实时更新功能

* 下一题更新后，房间内成绩排名是否实时更新

1. 查看历史成绩功能

* 显示历史成绩功能是否实现

1. 集成测试

功能测试完成后进行集成测试，即对应用的业务流程的测试。测试数据从一个模块流到另一个模块的过程中的正确性。关注于系统的流程实现，模拟用户操作测试应用的业务流程是否能符合逻辑地执行完毕。

1. 用户界面测试

用户界面测试主要验证以下要求是否满足：

* 页面设计人性化便于操作，布局合理
* 页面间跳转正确

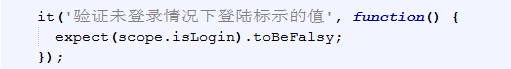
1. 压力测试

本次压力测试根据实际情况包含性能测试，重点模拟多用户进行测试。

压力测试要求多用户同时进行游戏时，测试页面更新是否流畅，排名等部分是否实时性更新。

1. 测试重点
   1. 功能测试
2. 测试工具
3. karma 自动化单元测试 Jasmine
4. Google Chrome 浏览器
5. 登录功能
6. 未登录检测功能是否实现

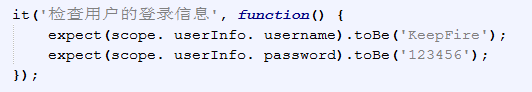
在用户未登录的情况下：



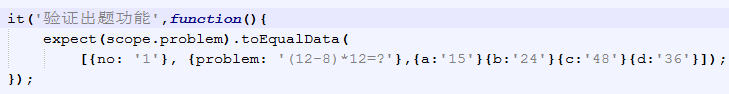
验证isLogin的值是否为false，如果是false则表示未登录；

1. 已登录界面的输入过滤功能是否实现

已在已登录情况下，获取登录的用户名&密码进行验证：



1. 出题功能
2. 题目获取是否成功



用来验证problem对象的6个元素是否为所出题目的内容

1. 答案提交和判断功能

选择答案，提交后

1. 若答案正确，界面应跳转为：right.html



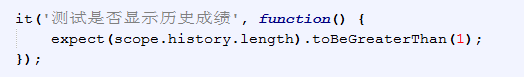
1. 若答案错误，界面应跳转为：wrong.html



1. 答题超时功能
2. 超时后是否有提醒
3. 超时提醒后是否更新下一道题

测试方法：出题后不进行任何操作，等待超时后，是否有提示&若干秒后更新下一题

1. 实时更新功能
2. 更新下一题后，房间内成绩排名是否实时更新
3. 查看历史成绩功能
4. 显示历史成绩功能是否实现



通过判断历史记录的数组长度来确定是否显示了历史记录。

* 1. 集成测试

1. 系统流程图

主页

页面跳转

登录

历史成绩

出题

加分

锁屏

sum += 1;

显示排名

进入房间

? 登录

? 正确

? 超时

? 他人答对

验证

sum=10

Y

Y

Y

Y

Y

N

N

Y

N

N

1. 测试用例

根据系统流程图，设置路径和判定覆盖的测试用例：

* 登录模块

1. 直接登录——进行下一步操作
2. 没有登录——登录成功——继续下一步操作
3. 没有登录——登录失败——重新登录——继续下一步操作

根据以上3条独立路径可设计2个测试用例：

* 用户进入主页，随即进行登录&&登录成功；
* 用户进入主页，紧接着进行没有登录状态下的操作，系统提示登录，第一次登录失败，第二次登录成功。
* 答题模块

1. 进入房间——答题——回答正确——加分——进行下一题——（循环累加）——显示排名——结束
2. 进入房间——答题——回答错误——锁屏——进行下一题——（循环累加）——显示排名——结束
3. 进入房间——答题——对手抢答&&答对——进行下一题——（循环累加）——显示排名——结束
4. 进入房间——答题——倒计时结束&&没有作答——进行下一题——（循环累加）——显示排名——结束

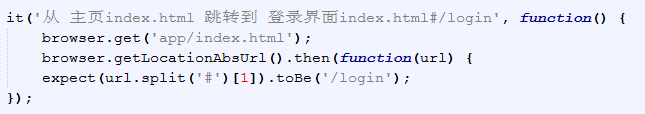
以上4条独立路径可在一次答题过程中涉及4个测试用例：

* 开始答题，结果回答错误了，然后被锁屏等待回答下一题；
* 开始答题，很快回答正确，得分并开始下一题；
* 开始答题，但是还没有回答就有被对手抢答了，然后等待进行下一题；
* 开始答题，所有选手都没有做出正确答案，时间到了，所有人没有得分，进行下一题。
  1. 界面测试

1. 界面间的正确跳转：
2. 主要的页面跳转有以下几项：

* index.html 跳转至 login.html
* index.html 跳转至 room.html
* room.html 跳转至 right.html 或 wrong.html
* index.html 跳转至 rules.html
* index.html 跳转至 history.html

1. 可以用通过捕获URL最后的字段校验是否实现跳转



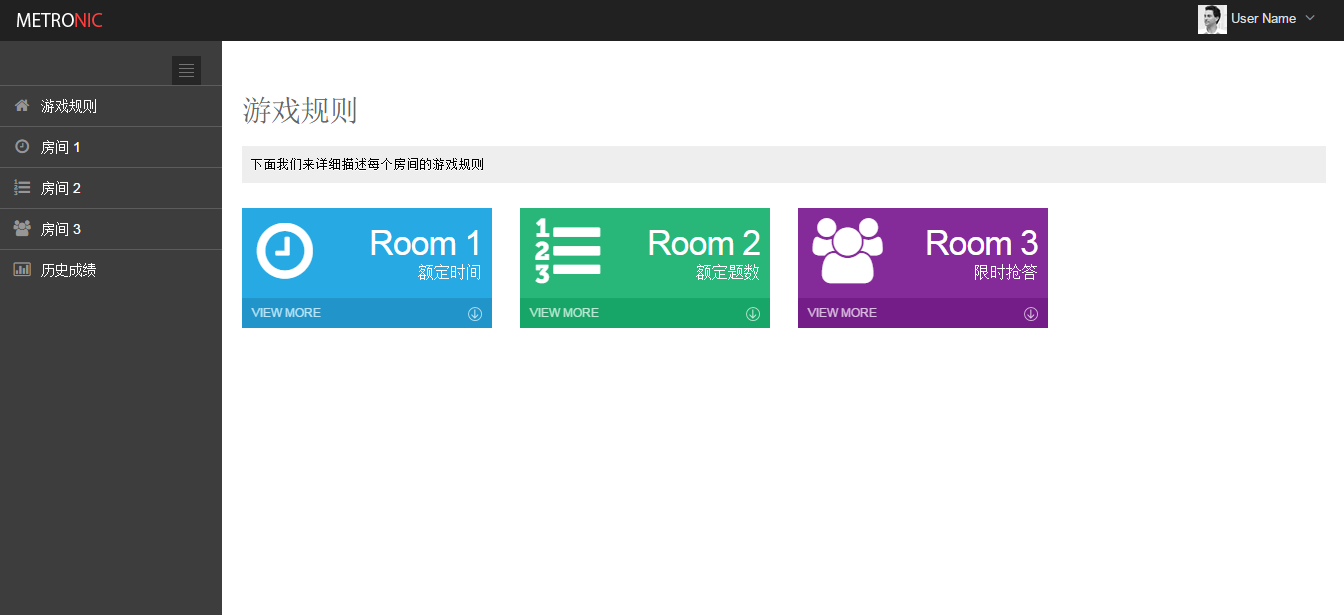
1. 响应式网页设计的测试
2. 测试工具:

* Viewport Resizer是一款基于浏览器的测试工具，它能够测试任何网站的响应能力。
* Respondr是一款轻量级、非常方便小巧的工具，用户只需输入网页的URL，然后选择你所要测试的设备，如iPhone、iPad、桌面浏览器等，即可看到网页在不同平台上的显示效果。

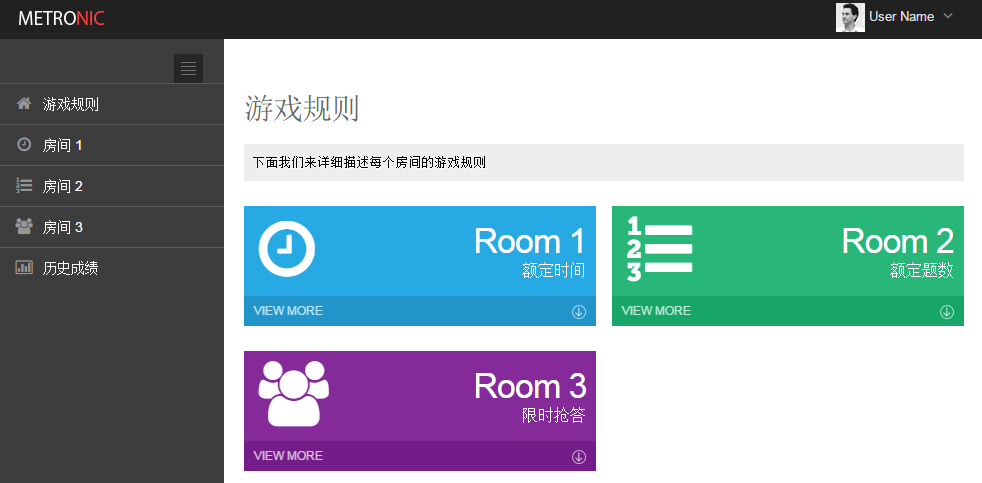
1. 测试方法：

用户只需把网页拖入书签，然后点击需要测试的网页即可检查页面的屏幕分辨率。此外，Viewport Resizer自带的分辨率尺寸不一定能够满足你的需求，所以，用户还可以自定义添加其它类型的屏幕尺寸。

* 电脑浏览器



* pad浏览器



* 手机浏览器



1. 测试计划
2. 测试任务和进度

|  |  |  |  |
| --- | --- | --- | --- |
| 测试阶段 | 测试任务 | 人员分工 | 起止时间 |
| 单元测试 | 登录模块功能测试 | 叶儿法特 | 1.11 |
| 历史成绩功能测试 | 叶儿法特 | 1.11 |
| 答题模块功能测试 | 马俊杰、刘宇飞 | 1.12-1.13 |
| 集成测试 | 测试应用业务流程 | 刘宇飞 | 1.13-1.14 |
| 界面测试 | 界面跳转&响应式设计 | 亢震 |  |

1. 测试完成后报告

* 测试用例列表
* 测试Bug列表
* 测试总结分析报告