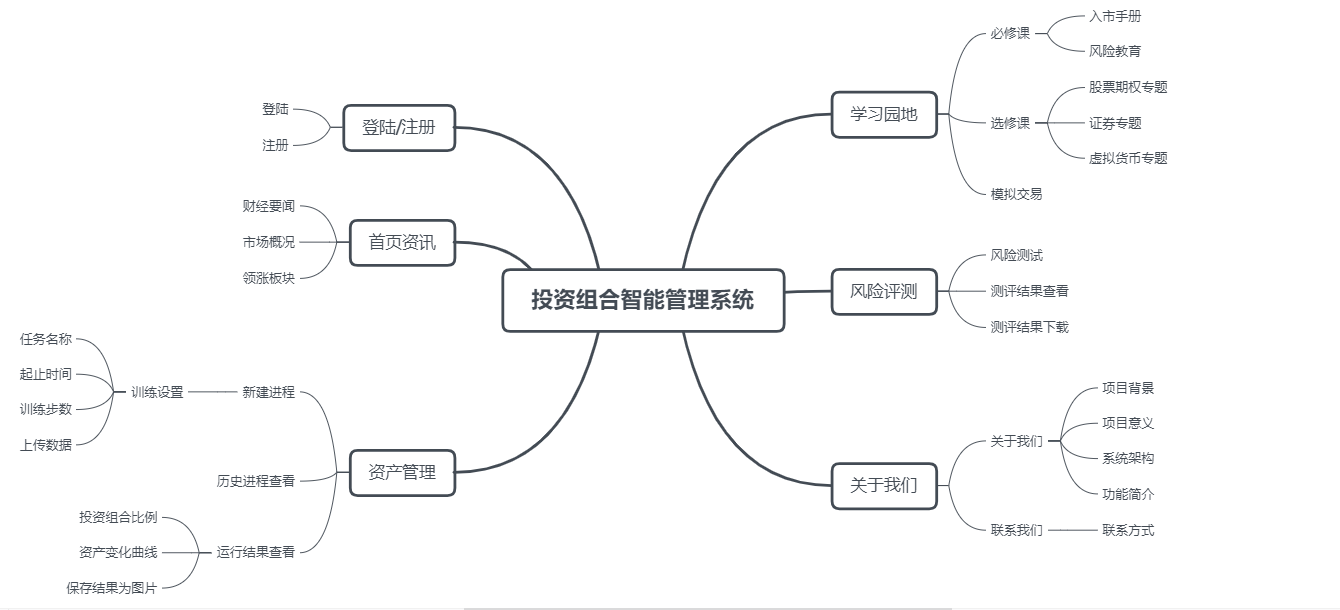
**投资组合智能管理系统测试报告**

测试人：何天舒 施沐凡 谢紫漪

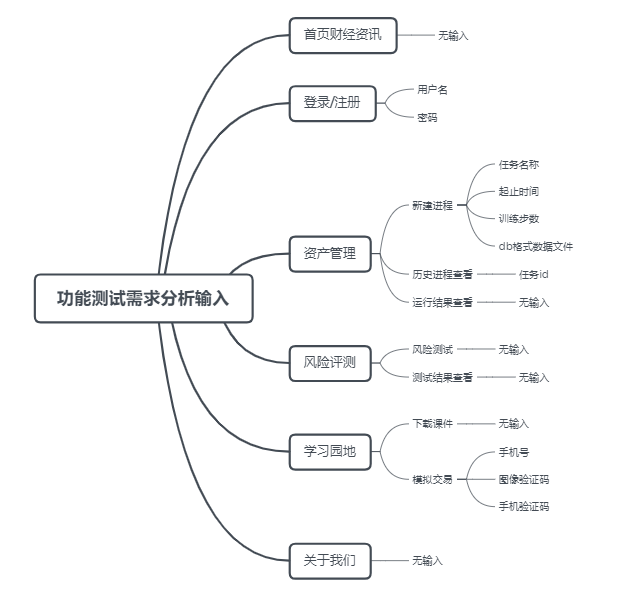
**一、功能描述**

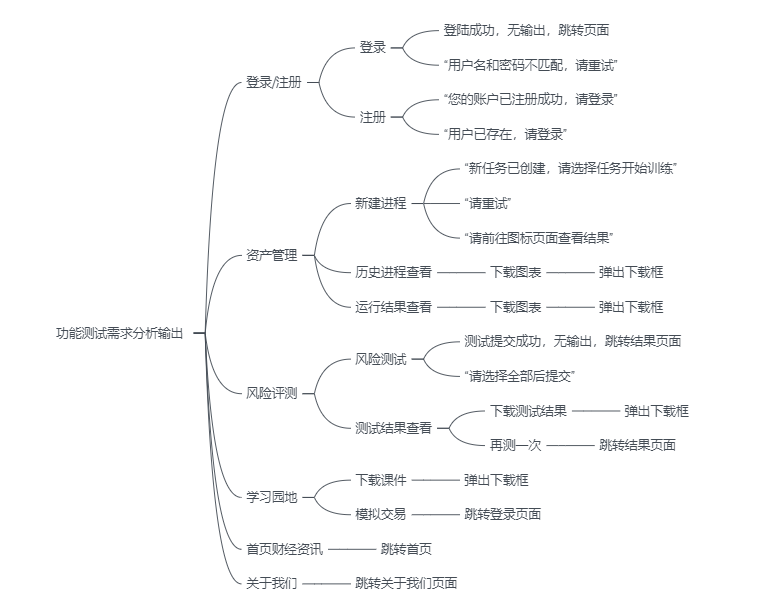
 投资组合智能管理系统主要分为四大功能模块：财经资讯、资产管理、学习园地和风险评测。

* 财经资讯功能为用户提供实时的海内外财经资讯，帮助用户及时获取海量财经信息，第一时间了解市场概况及领涨板块信息。
* 资产管理功能在结合了用户风险偏好的基础上利用深度强化学习算法为用户推荐最合适的投资组合。
* 学习园地为用户提供全面前沿的金融投资知识，风险教育、股票期权、证券、虚拟货币等相关知识一网打尽，模拟交易平台帮助用户通过实践掌握更专业的资产管理技巧。
* 风险评测通过用户的风险评估测试对其风险偏好进行准确评级，同时推荐个性化金融产品。

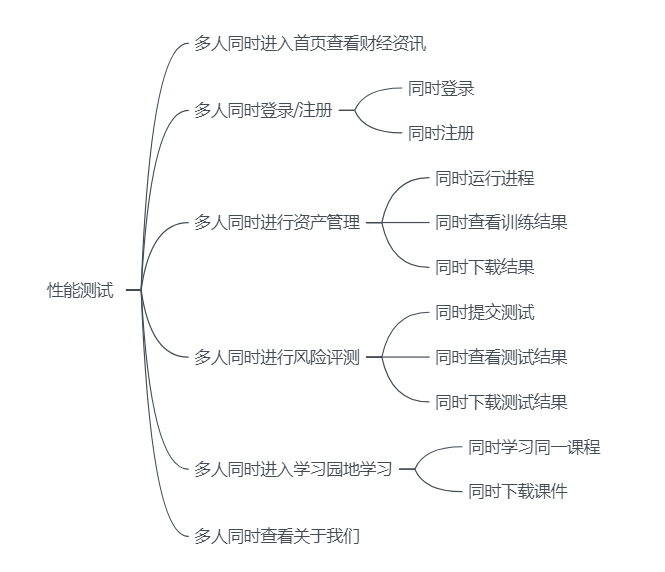


1. **测试需求分析**
2. 功能测试

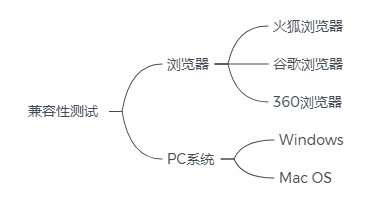




1. 其他测试
   1. 压力与性能测试：采用Jmeter及chrome下的performance进行压力测试



* 1. 安全性测试：Junit进行数据库加密的单元脚本测试
  2. 兼容性测试：不同的操作系统，不同的浏览器的兼容测试



1. **测试内容**
2. 白盒测试

1.1概述

白盒测试是全面了解程序内部逻辑结构、对所有逻辑路径进行的测试。对本项目的白盒测试从检查程序的逻辑着手，设计测试用例，并得出测试结果。

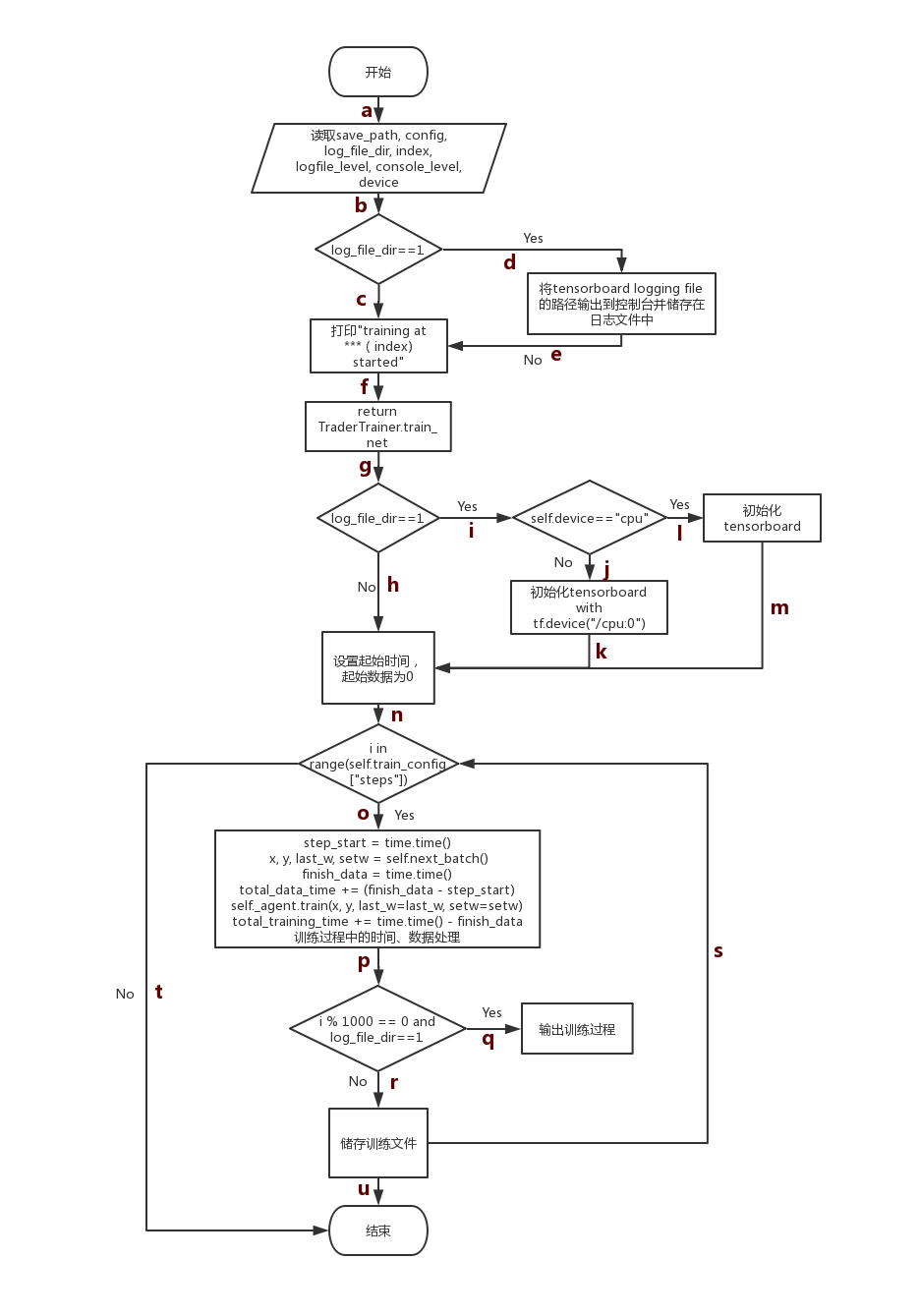
1.2 测试方法

由于项目且人手有限，故选取核心模块即资产管理模块中的数据训练过程进行测试。具体步骤如下：

* [计划](https://baike.baidu.com/item/%E6%B5%8B%E8%AF%95%E8%AE%A1%E5%88%92)阶段：根据测试需求分析，制定测试进度。
* 设计阶段：依据程序设计框架，按照规范的方法进行软件结构划分和测试用例设计。
* 执行阶段：输入测试用例，得到测试结果。
* 总结阶段：对比测试的结果和代码的预期结果，分析错误原因，找到并解决错误。

1.3白盒测试逻辑流程图

**v**



1.4测试用例列表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | save\_path | config | log\_file\_dir | index | logfile\_level | console\_level | device | 路径 |
| 1 | NULL | config.json | 0 | 0 | 5 | 0 | 0(cpu) | abcfghnopqrstuv |
| 2 | NULL | config.json | 1 | 3 | 2 | -3 | 0(cpu) | abdefgilmnopqrstuv |
| 3 | NULL | config.json | 0 | 2 | 3 | 4 | 1(gpu) | abcfgijknopqrstuv |
| 4 | NULL | config.json | 1 | 1 | 0 | 2 | 1(gpu) | abdefgijknopqrstuv |
| 5 | D:\test\_sample | config.json | 0 | 4 | -4 | 2 | 0(cpu) | abcfghnopqrstuv |
| 6 | D:\test\_sample | config.json | 1 | -1 | 1 | 4 | 0(cpu) | abdefgilmnopqrstuv |
| 7 | D:\test\_sample | config.json | 0 | 2 | 3 | 1 | 1(gpu) | abcfgijknopqrstuv |
| 8 | D:\test\_sample | config.json | 1 | 1 | 2 | 2 | 1(gpu) | abdefgijknopqrstuv |
| 9 | D://test\_sample | config.json | 0 | 2 | 2 | 2 | 0(cpu) | abcfghnopqrstuv |
| 10 | D://test\_sample | config.json | 1 | 1 | -1 | 3 | 0(cpu) | abdefgilmnopqrstuv |
| 11 | D://test\_sample | config.json | 0 | 1 | 3 | 1 | 1(gpu) | abcfgijknopqrstuv |
| 12 | D://test\_sample | config.json | 1 | 3 | 2 | -1 | 1(gpu) | abdefgijknopqrstuv |

1.5 测试总结

该白盒测试通过检查所选模块内部的逻辑结构，根据逻辑流程图对逻辑路径完成了完整的覆盖测试。通过在程序不同地方设立检查点检查状态，经测试，实际运行状态与预期状态一致。

但测试方法本身可能存在无法查出程序违反设计规范、因遗漏路径而出错、查找不出与数据相关的错误等缺陷。

1. 黑盒测试
   1. 概述

由于时间和技术限制，且项目组人力不足（只有三个人，比其他组更少），此外考虑到web应用部分功能更新频繁，测试代价过高，故这部分测试方法采用黑盒测试。对本项目地测试方法尽量涵盖web程序的各个方面，测试技术方面在继承传统测试技术的技术上结合web应用的特点

* 1. 测试方法
     1. 界面测试：直观检验直观性、一致性、舒适性、灵活性；
     2. 功能测试：采用seleniumIDE进行页面功能测试；查阅相关资料我们发现，vue框架下只能做 shadow 测试和 e2e 测试，e2e测试的本质就是模拟人的点击；因此进行手工测试。
  2. 测试内容
     1. 界面测试：
* 直观性：各界面洁净整齐、不拥挤不唐突，并且全部出现在预期位置；表单位置和导航栏均表现良好；
* 一致性：术语一致，快捷键（如F5刷新）表现统一，可以及时获取最新信息；
* 舒适性：页面跳转点击表现良好，软件外观和感觉较好，且相应操作有提示符和反应期，可以让用户明确程序在做什么；
* 灵活性：状态跳转功能良好，用户登录注册与进行资产管理绑定跳转，具有容错和处理能力，当网页框大小改变时，页面可以进行良好的调整，不会出现拥挤和错乱，如图所示可以看出，当页面宽窄调整，信息并不会拥挤错乱，灵活性测试表现良好。



图表 1：宽页面显示示例



图表 2：窄页面显示示例

* + 1. 功能测试
* 测试目的：功能测试的目的在于覆盖全部的功能，模拟用户及场景的实际测试；
* 测试工具：seleniumIDE，人工进行点击；
* 测试方案：按照测试需求分析种的内容进行点击；
* 测试用例（各种操作）及结果分析：
  + 首页：

1. 财经资讯部分按F5进行测试，可以进行新闻的实时刷新；
2. 且点击新闻可以跳转源网站；
3. 未点击的新闻字条为蓝色，已查看变为紫色，表现良好；
4. 大盘和股票以及行情信息为实时刷新；
5. 后端爬虫抓取数据打印到控制台显示均为实时刷新；
   * 资产管理：
6. 登录注册页面：
   1. 用户未注册有提示，用户密码错误有提示，用户注册或登录成功有提示，用户名和密码可以是字母或者数字或标点单独或者组合均可 ；
   2. 右上角可以在登录/注册于用户名之间动态转换，但是似乎需要进行刷新显示。
   3. 退出登录可以有效退出用户退至首页。
7. 新建进程：
   1. 进程名称可以保存至数据库，名称数字或字母不限，不超过50字；进程的时间可以进行选择或者输入，均能有效保存至数据库；
   2. 用户新建的进程可以和用户名进行有效绑定，查看进程只限制该用户的内容；
   3. 用户上传文件暂时只支持.db文件，未上传时创建进程按钮无法点击，上传文件之后可以点击进行训练。（开发的时候其实发现一个bug,似乎时element的，即使用户上传成功后台返回也是404但是确实可以有效创建进程，因此当创建进程按钮可以点击时就认为上传成功，仍可以训练，这一Bug后期会进行修补）。
8. 历史数据：
   1. 可以有效显示只属于该用户id（按照注册顺序分配用户id）的历史数据，没有出现混乱；
9. 显示图标：
   1. 投资组合比例：选中进程之后点击刷新数据即可显示投资组合比例，点击相关的虚拟货币可以单独显示或组合显示，图像跟随鼠标动态显示，但是目前时间戳为时间步，随后会进行更好的优化
   2. 资产管理曲线：也是刷新后显示，跟随鼠标及滚轮动态显示，可以单独或组合显示,数据可以进行下载和保存。
   * 学习园地：
10. 目录：

下拉框有响应；点击屏幕左侧目录，可跳转到对应专题，没有出现混乱。

1. 必修课/选修课：
2. 点击上/下一篇文章的标题能够实现同模块资料的切换折叠；
3. 阅读中pdf格式文件阅读页面上方有搜索按钮，点击可进行文档关键字查找，并跳转到上/下一个关键词位置，选择是否将关键词高亮显示；
4. pdf格式文件阅读可调整大小，并对页面进行顺时针或逆时针旋转；
5. 点击打印/下载按钮，可对pdf文档进行打印或下载；
6. 输入页数或点击转到第一页/最后一页，可直接跳转到指定位置，反应灵敏准确；
7. 文本选择工具能够准确选择文本并复制，手型工具能任意拖动文档调整位置；
8. 可以根据习惯选择垂直/水平/平铺滚动方式，pdf能够对对应方式进行调整，有序显示；
9. 可根据习惯选择视图模式（单页/双页/书籍）。
10. 模拟交易
11. 点击模拟交易后，按提示输入手机号码及验证码，可点击发送短信验证码，若前两者有缺失则会提示不能为空，符合要求则手机能够顺利收到验证码，输入验证码后可进入模拟交易界面；
12. 模拟交易见面内可以1000000元的模拟总资产进行投资，页面能够实时显示持股的市值，总盈利率，日盈利率，我的个股及历史交易记录；
13. 点击新增个股输入股票代码及相关参数进行股票虚拟交易，反应灵敏，计算准确；
14. 点击注销可顺利退出模拟交易页面，返回登录界面。
    * 风险测评：
15. 未测评完整会有提示；
16. 测评完整后进行页面跳转，有效给出得分（已随机测试十种正常选择，并进行极端选择使得分数最低甚至为负或者全部选得分最高的选项均可以正常显示）
17. 可以进行测试结果图下载；
18. 根据用户的不通过风险偏好，给出有效建议均符合预期结果；
    * 关于我们：有效显示信息和链接，下拉框有响应；

3.其他测试

* 1. 测试方法
     1. 压力测试与性能测试：采用Jmeter及chrome下的performance进行压力测试；（附脚本代码）
     2. 安全性测试：Junit进行数据库加密的单元脚本测试；（附测试代码）
     3. 兼容性测试：不同的操作系统，不同的浏览器的兼容测试；
  2. 测试内容

3.2.1 压力测试与性能测试

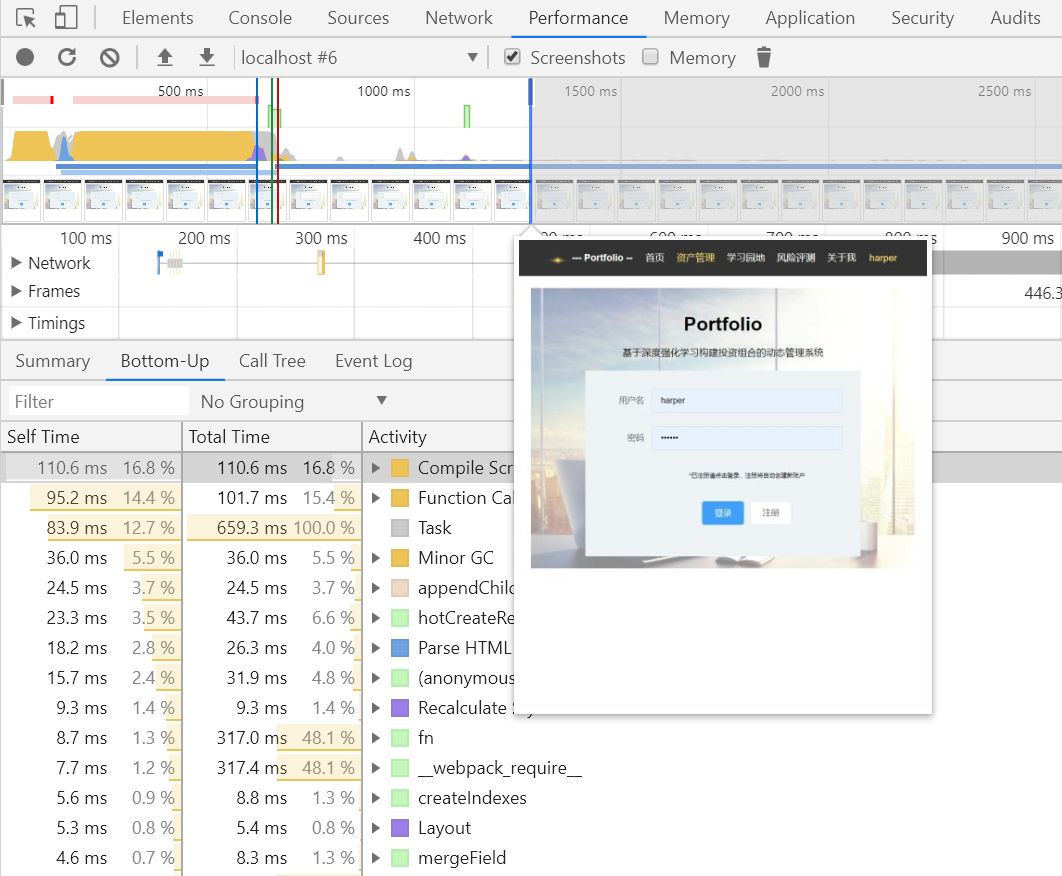
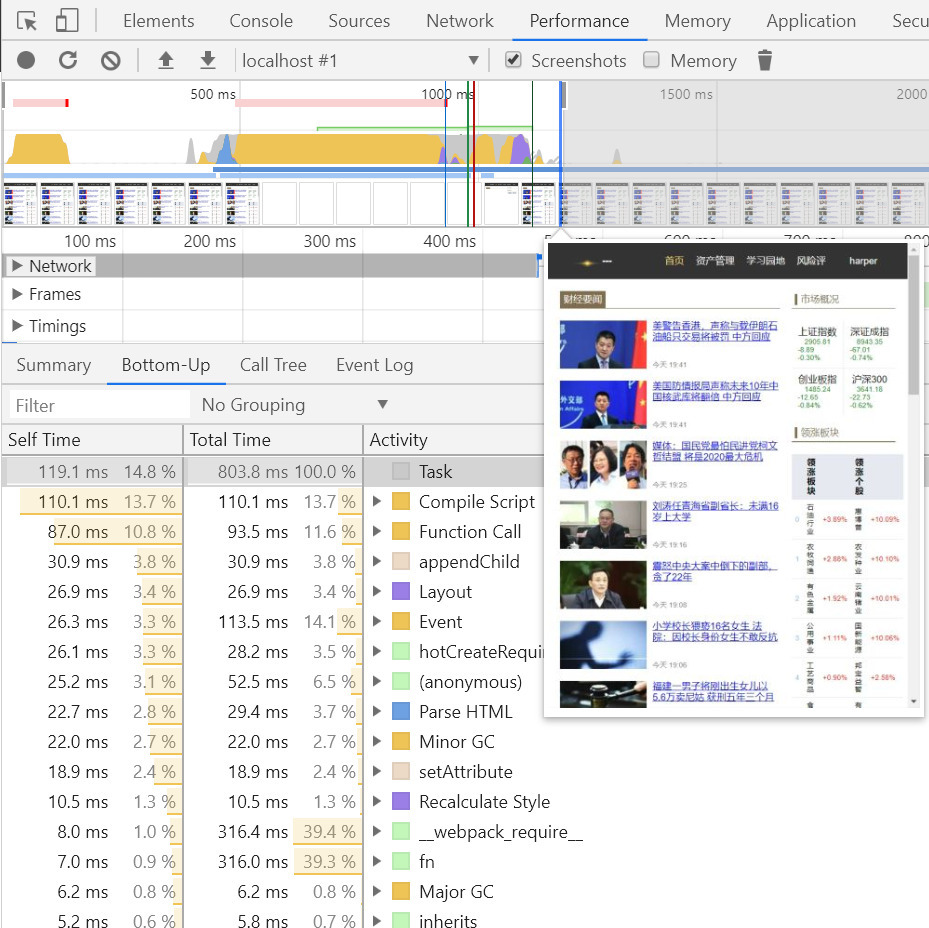
将压力测试与性能测试结合，由于条件不足，暂无法进行硬盘影响等硬件测试；

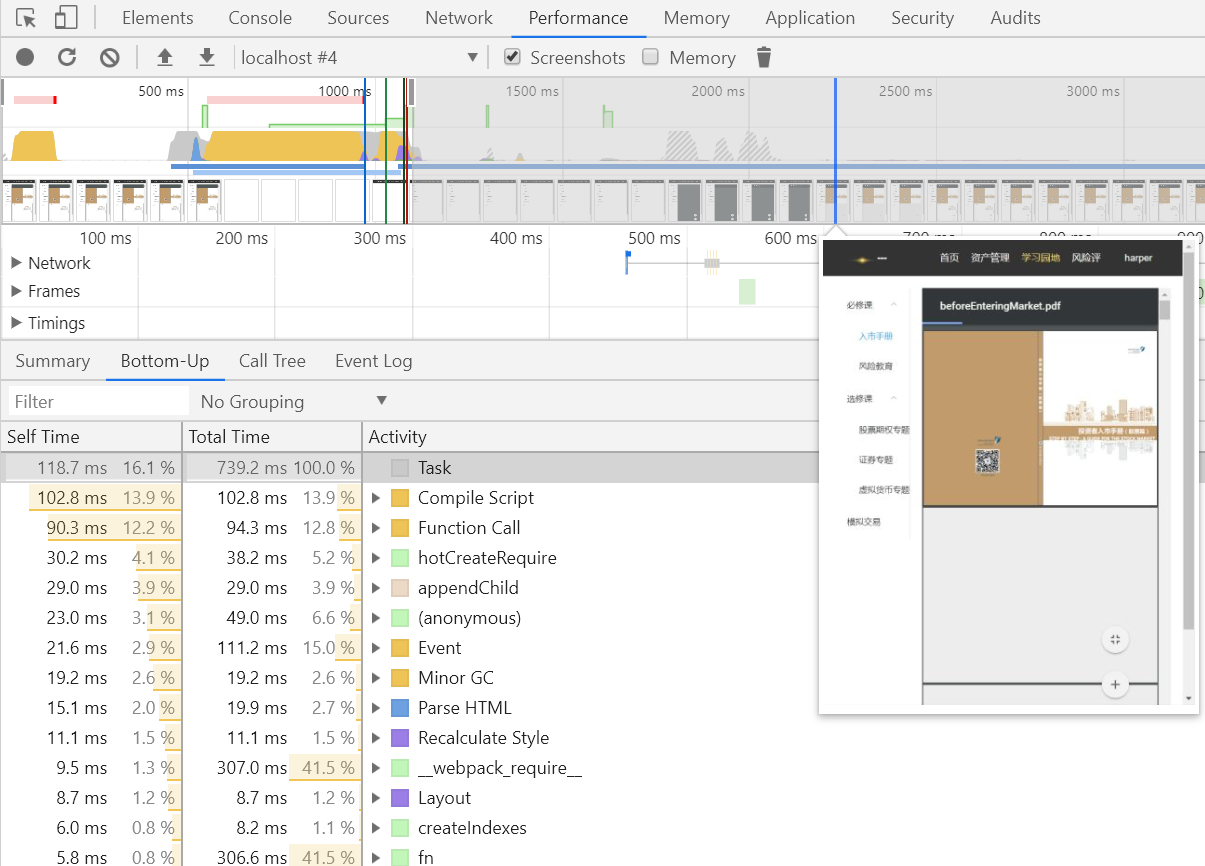
* 基本功能Jmeter聚合报告

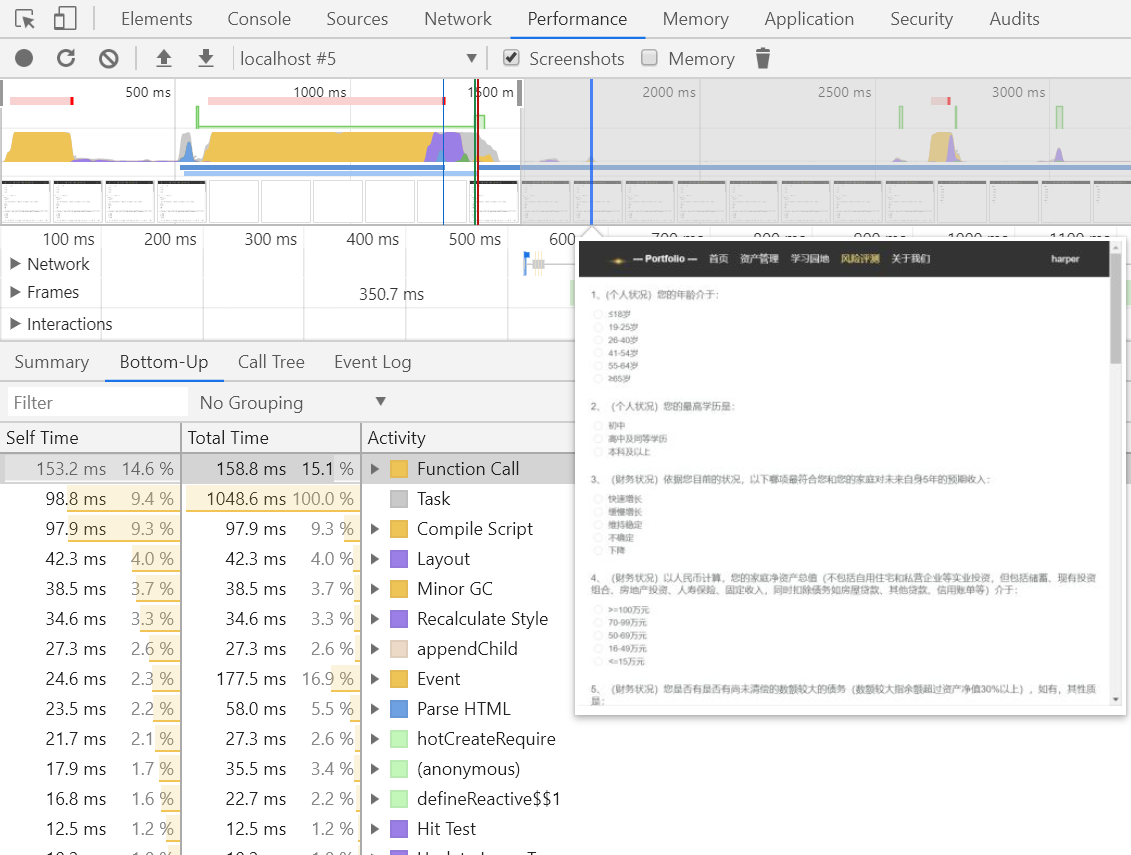


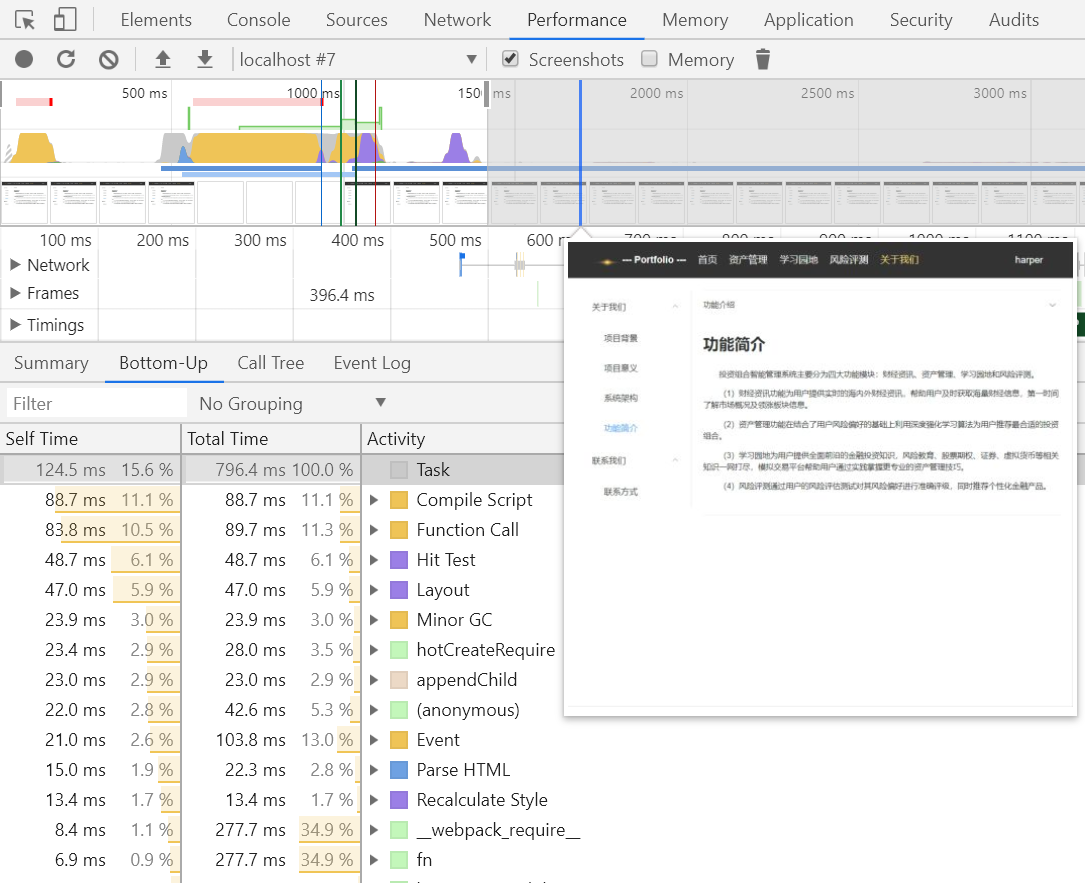
* 页面加载表现

基于chrome浏览器下的检测工具，查看各页面初始化及加载表现









3.2.2 安全性测试

为了防止数据的窃取，我们对databases进行了基于springboot搭载jasypt的加密措施，该部分的单元测试如下：

* 配置文件：

datasource.host=DES@e3zcSlYS29N0Y3i+mVdkgQ==

datasource.port=DES@S6mBLsaSBEw=

datasource.database=test

datasource.url=jdbc:mysql://${datasource.host}:${datasource.port}/${datasource.database}?useUnicode=true&amp;characterEncoding=utf8

* 测试注入

@RunWith(SpringRunner.class)

@SpringBootTest(classes = Application.class)

public class ApplicationTests {

@Value("${datasource.url}")

private String url;

@Test

public void testJasypt() {

System.out.println(url);

}

}

* 输出：

jdbc:mysql://127.0.0.1:3306/test?useUnicode=true&amp;characterEncoding=utf8

3.3.3 兼容性测试

* 平台测试：项目已在MacOS及Windows10、Windows7平台下进行测试，表现一致，暂未进行Linux操作系统下的测试；
* 浏览器软件和版本测试：我们在360浏览器、Firefox、Chrome下进行了测试，并且对页面进行滚动及缩放，各功能均有良好表现，且发现Firefox下ppt performance表现最好，Chrome和360表现相当。