

基于Wifi探针的数据分析系统

测试文档



2017-6-24

南京大学软件学院

Coding Fairy

目录

[1. 概述 3](#_Toc486273865)

[2. 测试环境 3](#_Toc486273866)

[3. 界面功能测试 3](#_Toc486273867)

[3.1. 测试方法 3](#_Toc486273868)

[3.2. 测试数据 3](#_Toc486273869)

[3.2.1. TC1: 客流量分析 3](#_Toc486273870)

[- TUS1的测试用例 3](#_Toc486273871)

[- TUS1-4测试结果示例 4](#_Toc486273872)

[- TUS2的测试用例 4](#_Toc486273873)

[- TUS2-1测试结果示例 4](#_Toc486273874)

[3.2.2. TC2：顾客活跃度分析 5](#_Toc486273875)

[- TUS1的测试用例 5](#_Toc486273876)

[- TUS1-4测试结果示例 6](#_Toc486273877)

[- TUS2的测试用例 6](#_Toc486273878)

[- TUS2-1测试结果示例 6](#_Toc486273879)

[3.2.3. TC3：新老顾客分析 7](#_Toc486273880)

[- TUS1的测试用例 7](#_Toc486273881)

[- TUS1-4测试结果示例 8](#_Toc486273882)

[- TUS2的测试用例 8](#_Toc486273883)

[3.2.4. TC4：驻店时长分析 9](#_Toc486273884)

[- TUS1的测试用例 9](#_Toc486273885)

[- TUS1-4测试结果示例 9](#_Toc486273886)

[- TUS2的测试用例 10](#_Toc486273887)

[- TUS2-1测试示例 10](#_Toc486273888)

[3.2.5. TC5：来访周期分析 11](#_Toc486273889)

[- TUS1的测试用例 11](#_Toc486273890)

[- YUS1-4测试结果示例 11](#_Toc486273891)

[- TUS2的测试用例 12](#_Toc486273892)

[- TUS2-1测试结果示例 12](#_Toc486273893)

[4. 数据接收服务器测试 13](#_Toc486273894)

[4.1. 测试方法 13](#_Toc486273895)

[4.2. 测试数据 13](#_Toc486273896)

[4.3. 测试结果 15](#_Toc486273897)

[4.3.1. 以树的形式查看结果 15](#_Toc486273898)

[4.3.2. 以表格形式查看结果 16](#_Toc486273899)

1. 概述

该文档描述了wifiProbe系统的测试环境搭建，测试数据说明和测试方法使用。

**修改记录**

|  |  |  |
| --- | --- | --- |
| 修改人 | 修改时间 | 修改原因 |
| 袁阳阳 | 2017/06/24 | 初始版本 |
| 袁阳阳 | 2017/06/25 | 添加测试结果 |

1. 测试环境

//todo

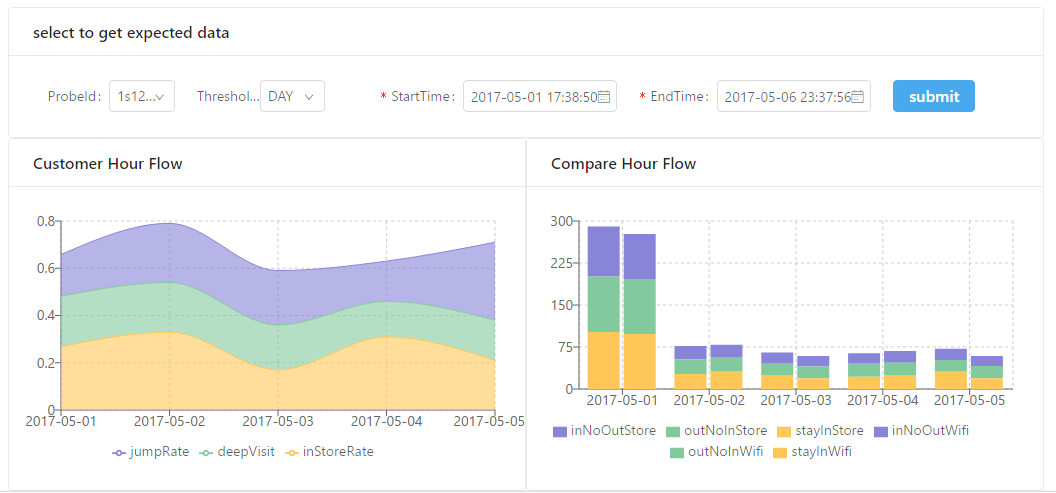
1. 界面功能测试
   1. 测试方法

使用selenium平台录制脚本，并配置参数，进行自动化测试

* 1. 测试数据
     1. TC1: 客流量分析
        + TUS1的测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **输入** | | | | **预期输出** |
| **探针ID** | **维度** | **开始时间** | **结束时间** |
| TUS1-1 | 1s12sz | 年 | 2013/1/1  20:00:00 | 2017/1/1  21:00:00 | 按年统计的2013-2017年的客流量分析数据 |
| TUS1-2 | 1s12sz | 月 | 2017/1/1  20:00:00 | 2017/6/1  21:00:00 | 按月统计的1月-6月的客流量分析数据 |
| TUS1-3 | 1s12sz | 周 | 2017/5/1  20:00:00 | 2017/6/4  21:00:00 | 按周统计的此五周的客流量分析数据 |
| TUS1-4 | 1s12sz | 日 | 2017/5/1  20:00:00 | 2017/5/6  21:00:00 | 按天统计的1-6号的客流量分析数据 |
| TUS1-5 | 1s12sz | 时 | 2017/5/1  17:00:00 | 2017/5/1  21:00:00 | 按小时统计的9点到21点的客流量分析数据 |

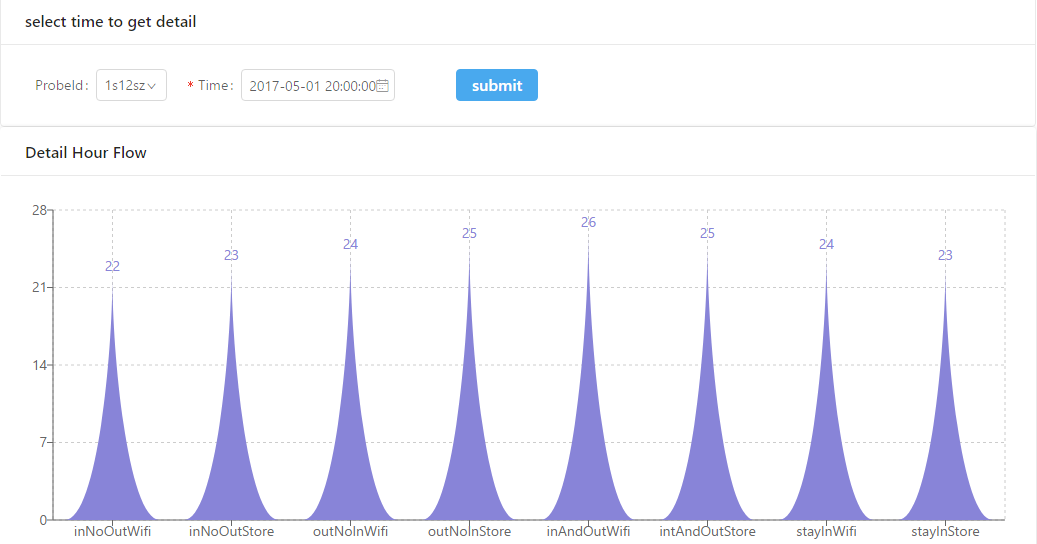
* + - * TUS1-4测试结果示例



* + - * TUS2的测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **输入** | | **预期输出** |
| **探针ID** | **时间** |
| TUS2-1 | 1s12sz | 2017/5/1 20:00:00 | 该时间的客流量各项数据 |

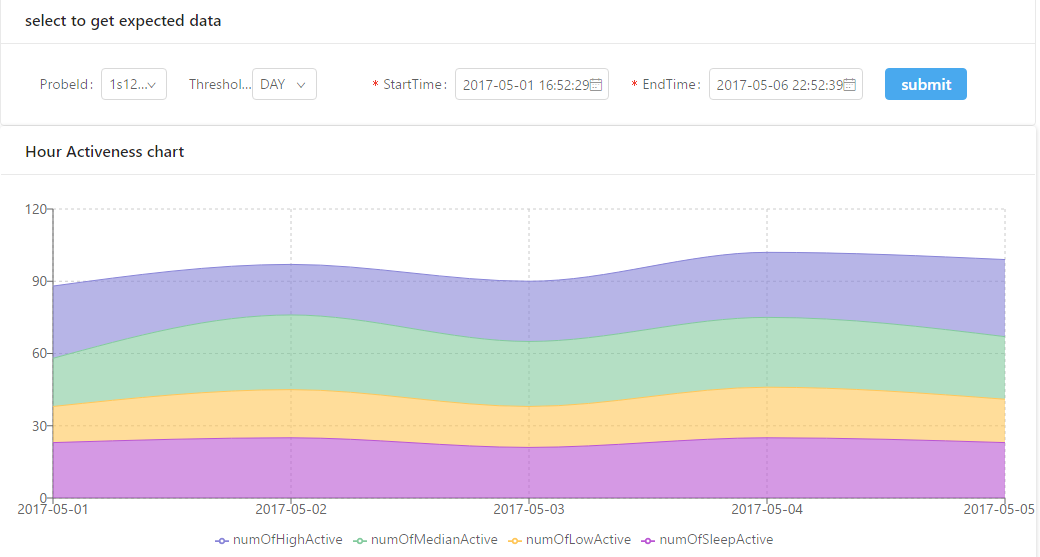
* + - * TUS2-1测试结果示例



* + 1. TC2：顾客活跃度分析
       - TUS1的测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **输入** | | | | **预期输出** |
| **探针ID** | **维度** | **开始时间** | **结束时间** |
| TUS1-1 | 1s12sz | 年 | 2013/1/1  20:00:00 | 2017/1/1  21:00:00 | 按年统计的2013-2017年的活跃度分析数据 |
| TUS1-2 | 1s12sz | 月 | 2017/1/1  20:00:00 | 2017/6/1  21:00:00 | 按月统计的1月-6月的活跃度分析数据 |
| TUS1-3 | 1s12sz | 周 | 2017/5/1  20:00:00 | 2017/6/4  21:00:00 | 按周统计的此五周的活跃度分析数据 |
| TUS1-4 | 1s12sz | 日 | 2017/5/1  20:00:00 | 2017/5/6  21:00:00 | 按天统计的1-6号的活跃度分析数据 |
| TUS1-5 | 1s12sz | 时 | 2017/5/1  17:00:00 | 2017/5/1  21:00:00 | 按小时统计的9点到21点的活跃度分析数据 |

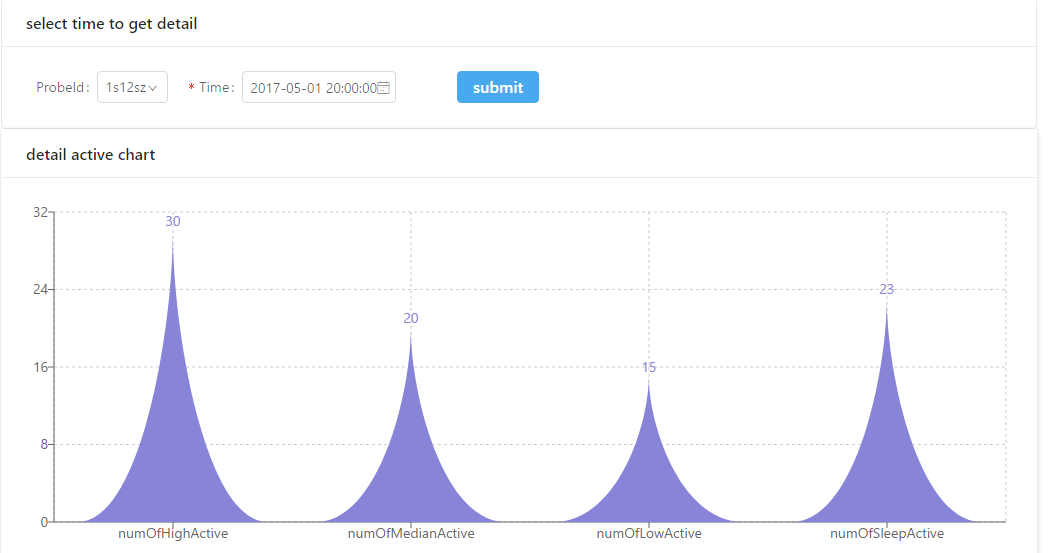
* + - * TUS1-4测试结果示例



* + - * TUS2的测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **输入** | | **预期输出** |
| **探针ID** | **时间** |
| TUS2-1 | 1s12sz | 2017/5/1 20:00:00 | 该时间的活跃度各项数据 |

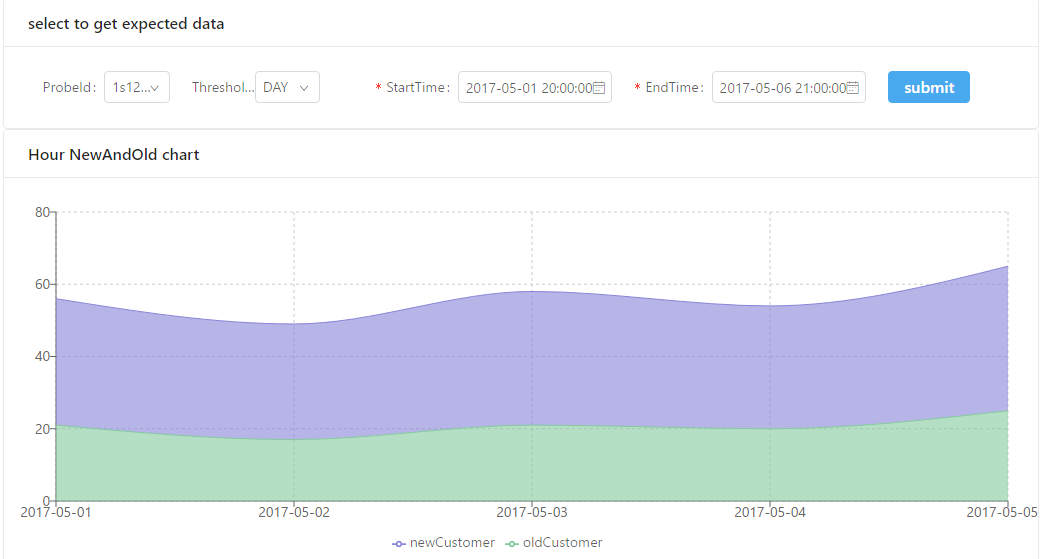
* + - * TUS2-1测试结果示例



* + 1. TC3：新老顾客分析
       - TUS1的测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **输入** | | | | **预期输出** |
| **探针ID** | **维度** | **开始时间** | **结束时间** |
| TUS1-1 | 1s12sz | 年 | 2013/1/1  20:00:00 | 2017/1/1  21:00:00 | 按年统计的2013-2017年的新老顾客分析数据 |
| TUS1-2 | 1s12sz | 月 | 2017/1/1  20:00:00 | 2017/6/1  21:00:00 | 按月统计的1月-6月的新老顾客分析数据 |
| TUS1-3 | 1s12sz | 周 | 2017/5/1  20:00:00 | 2017/6/4  21:00:00 | 按周统计的此五周的新老顾客分析数据 |
| TUS1-4 | 1s12sz | 日 | 2017/5/1  17:00:00 | 2017/5/6  21:00:00 | 按天统计的1-6号的新老顾客分析数据 |
| TUS1-5 | 1s12sz | 时 | 2017/5/1  9:00:00 | 2017/5/1  21:00:00 | 按小时统计的9点到21点的新老顾客分析数据 |

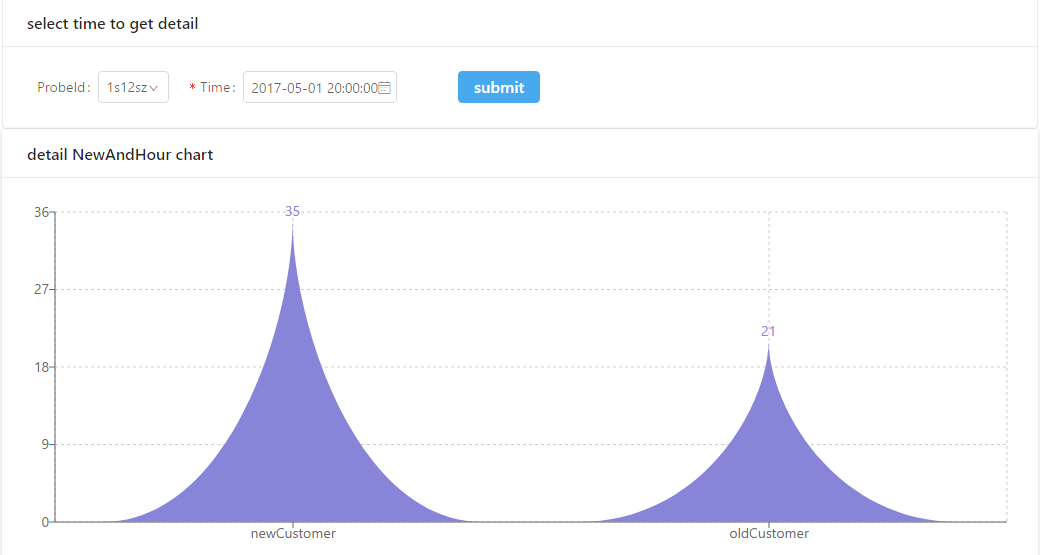
* + - * TUS1-4测试结果示例



* + - * TUS2的测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **输入** | | **预期输出** |
| **探针ID** | **时间** |
| TUS2-1 | 1s12sz | 2017/5/1 20:00:00 | 该时间的新老顾客各项数据 |

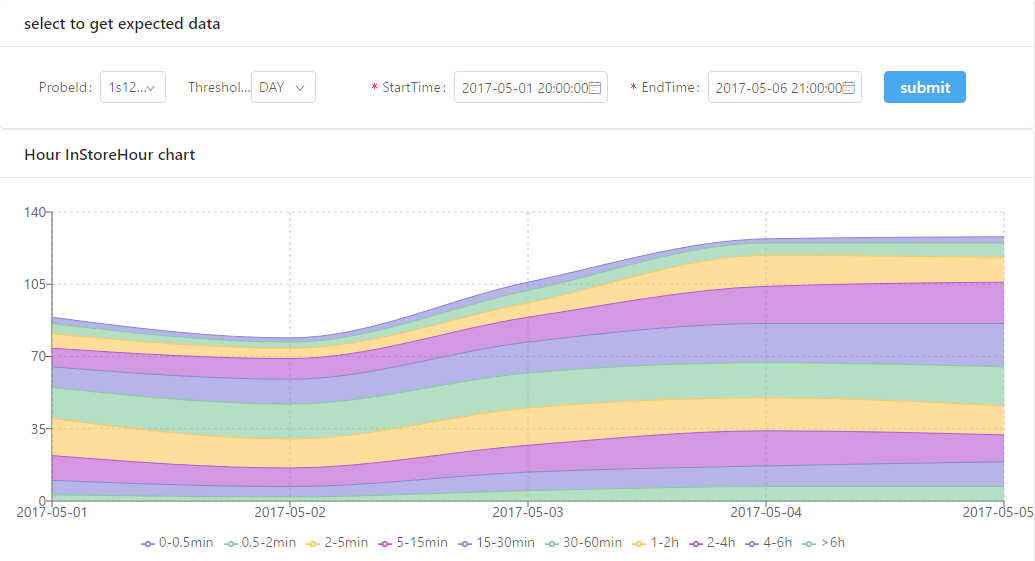
* + - * TUS2-1测试结果实例



* + 1. TC4：驻店时长分析
       - TUS1的测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **输入** | | | | **预期输出** |
| **探针ID** | **维度** | **开始时间** | **结束时间** |
| TUS1-1 | 1s12sz | 年 | 2013/1/1  20:00:00 | 2017/1/1  21:00:00 | 按年统计的2013-2017年的驻店时长分析数据 |
| TUS1-2 | 1s12sz | 月 | 2017/1/1  20:00:00 | 2017/6/1  21:00:00 | 按月统计的1月-6月的驻店时长分析数据 |
| TUS1-3 | 1s12sz | 周 | 2017/5/1  20:00:00 | 2017/6/4  21:00:00 | 按周统计的此五周的驻店时长分析数据 |
| TUS1-4 | 1s12sz | 日 | 2017/5/1  20:00:00 | 2017/5/6  21:00:00 | 按天统计的1-6号的驻店时长分析数据 |
| TUS1-5 | 1s12sz | 时 | 2017/5/1  17:00:00 | 2017/5/1  21:00:00 | 按小时统计的9点到21点的驻店时长分析数据 |

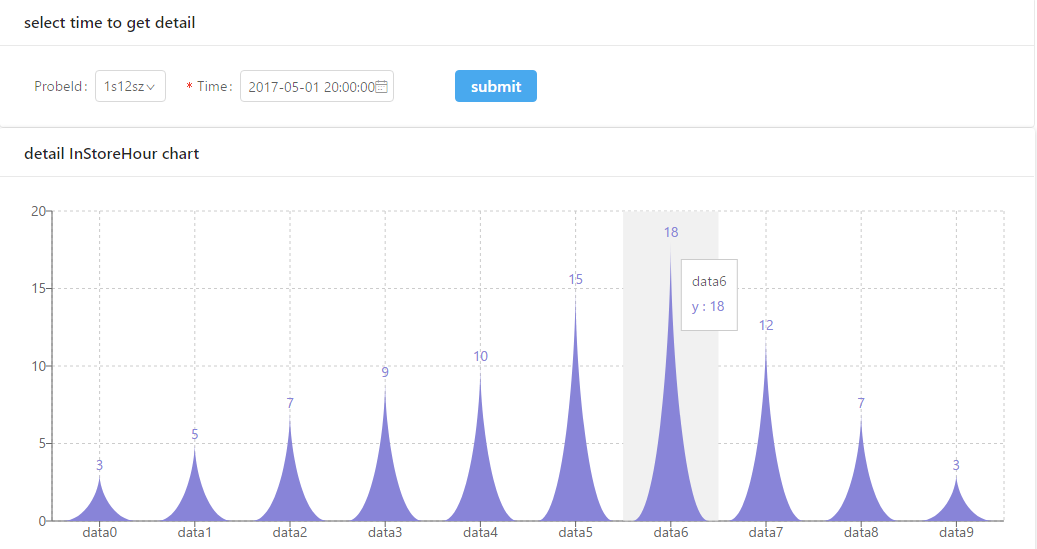
* + - * TUS1-4测试结果示例



* + - * TUS2的测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **输入** | | **预期输出** |
| **探针ID** | **时间** |
| TUS2-1 | 1s12sz | 2017/5/1 20:00:00 | 该时间的驻店时长各项数据 |

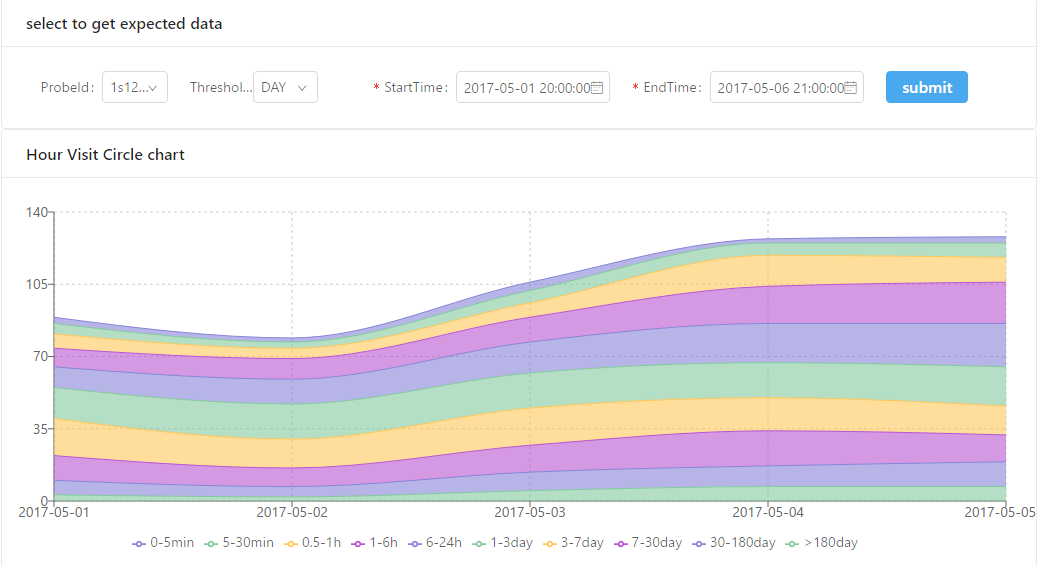
* + - * TUS2-1测试示例



* + 1. TC5：来访周期分析
       - TUS1的测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **输入** | | | | **预期输出** |
| **探针ID** | **维度** | **开始时间** | **结束时间** |
| TUS1-1 | 1s12sz | 年 | 2013/1/1  20:00:00 | 2017/1/1  21:00:00 | 按年统计的2013-2017年的来访周期分析数据 |
| TUS1-2 | 1s12sz | 月 | 2017/1/1  20:00:00 | 2017/6/1  21:00:00 | 按月统计的1月-6月的来访周期分析数据 |
| TUS1-3 | 1s12sz | 周 | 2017/5/1  20:00:00 | 2017/6/4  21:00:00 | 按周统计的此五周的来访周期分析数据 |
| TUS1-4 | 1s12sz | 日 | 2017/5/1  20:00:00 | 2017/5/6  21:00:00 | 按天统计的1-6号的来访周期分析数据 |
| TUS1-5 | 1s12sz | 时 | 2017/5/1  17:00:00 | 2017/5/1  21:00:00 | 按小时统计的9点到21点的来访周期分析数据 |

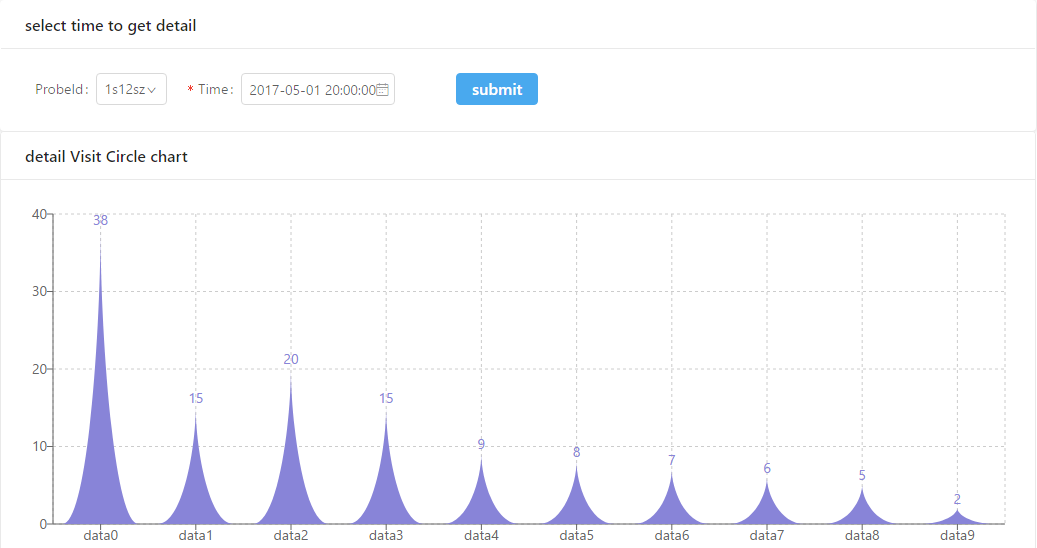
* + - * YUS1-4测试结果示例



* + - * TUS2的测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **输入** | | **预期输出** |
| **探针ID** | **时间** |
| TUS2-1 | 1s12sz | 2017/5/1 20:00:00 | 该时间的来访周期各项数据 |

* + - * TUS2-1测试结果示例



1. 数据接收服务器测试
   1. 测试方法

使用Jmeter进行并发测试，通过Jmeter客户端发送Http request，设置同步计时器（Synchronizing Timer），以模仿1000+的Wifi探针同时向数据接收服务器发送数据，具体步骤描述如下：

1. 建立test plan
2. 添加线程组，设置线程数为1000
3. 建立同步计时器，设置累计线程数（用户数）为1000，确保1000条数据同时发往接收服务器
4. 添加Http request, 并配置服务器地址，访问路径，POST方式，携带的json数据
5. 发送请求并观察测试结果
   1. 测试数据

Http request中携带的json数据如下：

{

"id": "0010f377",

"mmac": "5e:cf:7f:10:f3:77",

"rate": "1",

"wssid": "kaituo",

"wmac": "a8:57:4e:c0:d4:8c",

"time": "Sat Jun 04 20:45:28 2016",

"lat": "30.748093",

"lon": "103.973083",

"addr": "江苏省南京市玄武大道699-22号",

"data": [{

"mac": "9a:21:6a:7b:62:6a",

"rssi": "-30",

"range": "1.0",

"ts": "hello",

"tmc": "00:01:02:03:04:05",

"tc": "Y",

"ds": "N",

"essid0":"七天连锁\_wifi",

"essid1":"工商银行",

"essid2":"东方明珠",

"essid3":"home",

"essid4":"abcd",

"essid5":"xiong",

"essid6":"XX会馆"

}, {

"mac": "1c:31:72:5c:83:6b",

"rssi": "-69",

"range": "14.0",

"ts": "world",

"tmc": "00:01:02:03:04:06",

"tc": "Y",

"ds": "Y",

"essid0": "七天连锁\_wifi",

"essid1":"工商银行",

"essid2":"东方明珠",

"essid3":"home",

"essid4":"abcd",

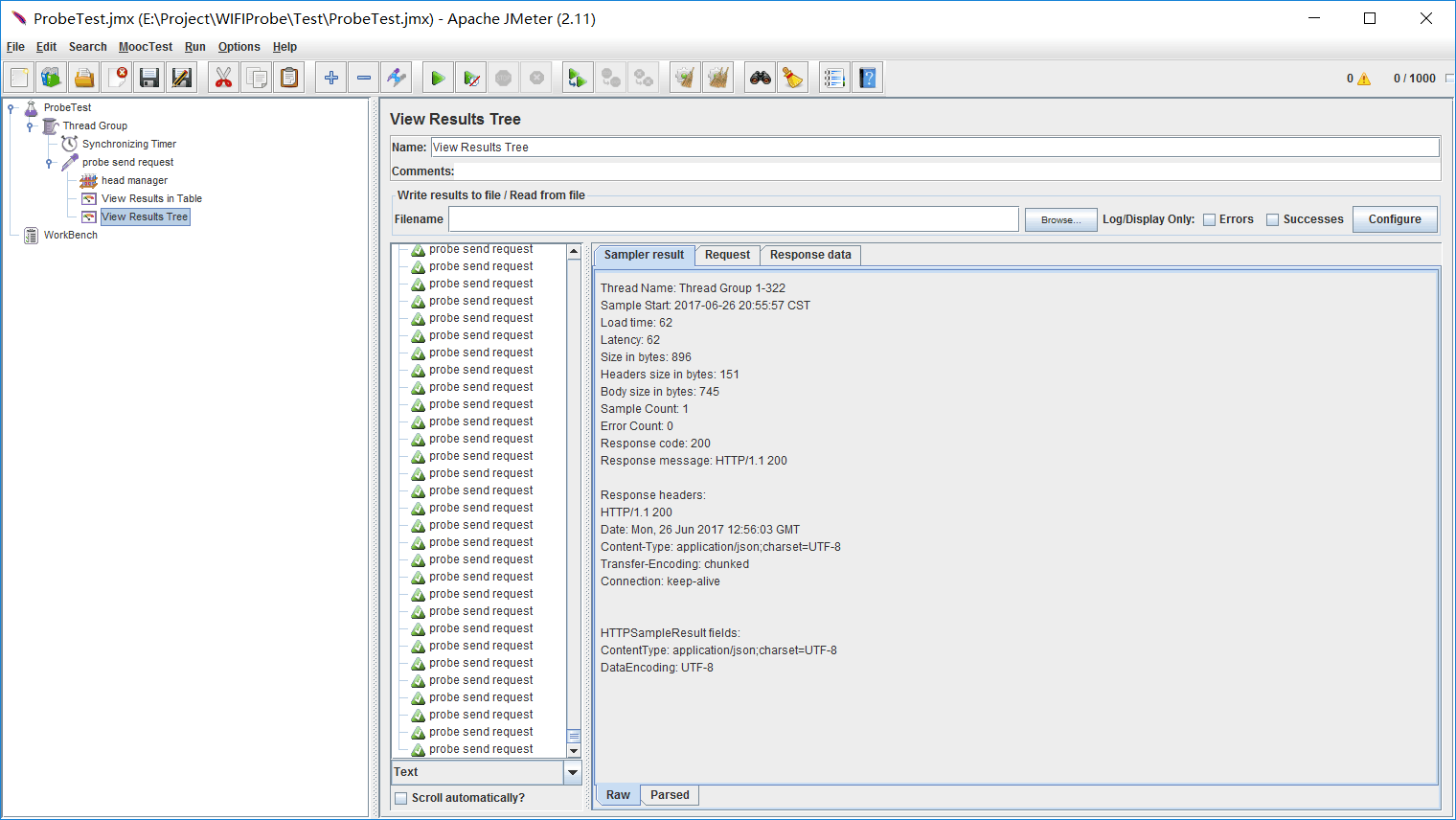
"essid5":"xiong",

"essid6":"XX会馆"

}]

}

* 1. 测试结果
     1. 以树的形式查看结果



* + 1. 以表格形式查看结果

