

1、该项目的web访问

PC版本地址：http://123.57.141.84

管理员登陆用户名：Admin 密码：zabbix

注意：APP所有的访问数据都应该与web侧一一对应；

2、所有的json-api都会提供测试好的Android-JAR包以及调用方法程序，但所有JSON转换（字符串与JSON对象之间）都必须处理可能的异常；同时，开发者要能根据Android-JAR包源代码写出相应的IOS开发包（交接源代码时一并提供开发包源代码）；特别注意：对于JAR包代码，开发者觉得任何需要改动的地方都要和我方商议后进行；

3、Android/IOS APP必须都相应地支持Phone/Pad两个机型；并且都应该支持正式上线前的最新Android/IOS OS版本；同时，Android 支持最低4.0版本，IOS 支持最低IOS7版本；

4、Android/IOS版本都必须有不同语言的切换机制，具体要调用的文本翻译文件(.po/.mo)由我方提供，这些文件也是WEB所用的文件，所以具体对应及调用方式参考WEB即可；

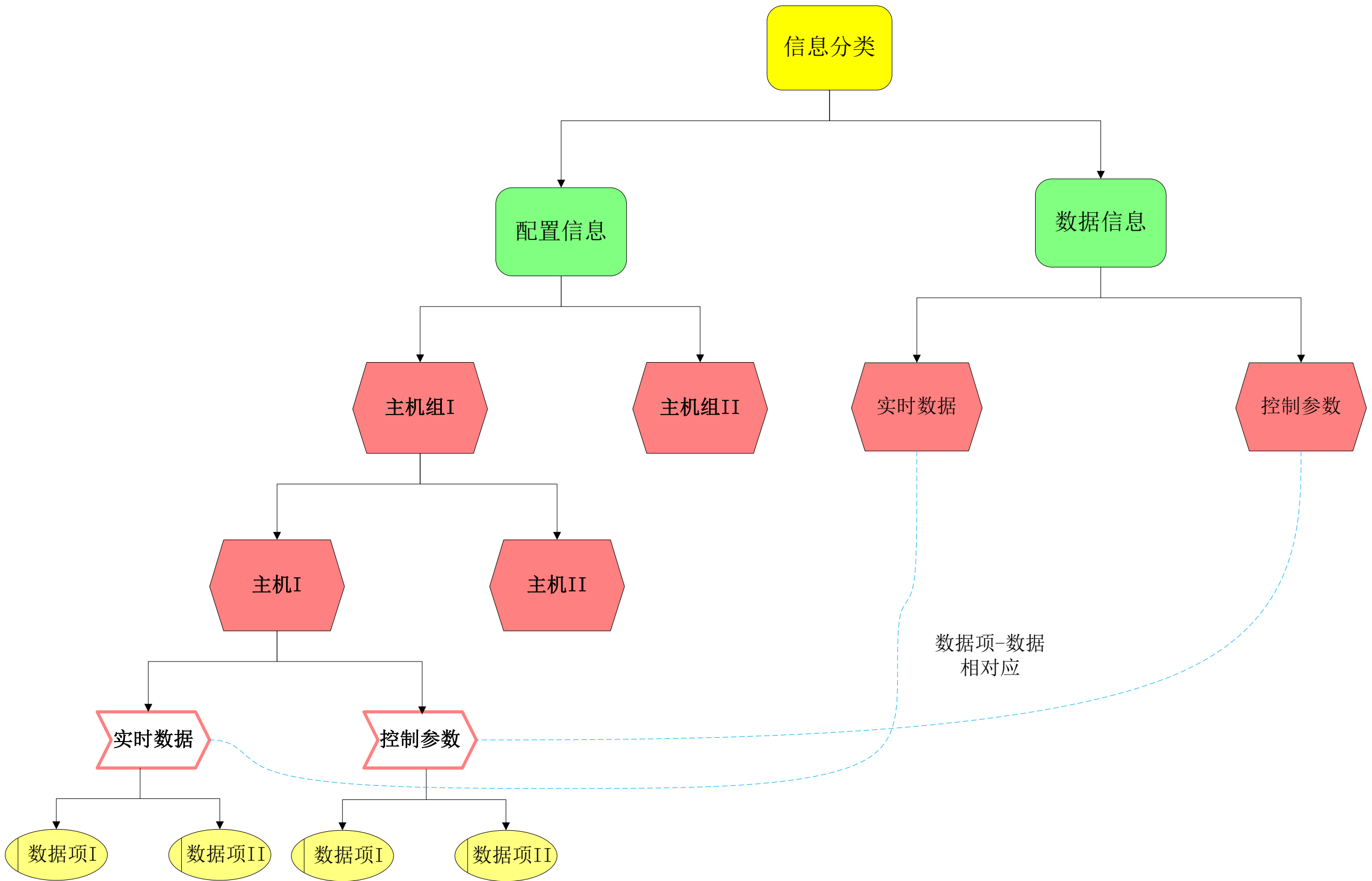
5、美工切图会在开发者主体界面完成后提供，到时贴图即可；

注意：本项目后续会分期维护、更新，这次是第一期；

无论哪一期，都必须保密，不得擅自泄露各方面的信息及文档；

开发者一旦接下开发任务，我方付了订金后，即认为开发者接受保密条件；

一旦泄密或者资料到处乱放，则我方保留追究承包方法律方面责任的权利！



【2-实时监控】界面

- 1、会读取所有的配置信息并放在界面上，只有在接口返回的信息中提示配置有改变时才从新读取配置信息，读取后存到本地数据库，下次登录直接调用本地数据；配置改变信息参见接口函数参数【DvsCfgVer dcf】
- 2、程序定时刷新或用户单击刷新按钮时读取并刷新各个数据项的数据；

【1-信息总览】界面

- 1、只读取所有的主机组配置信息，只有在接口返回的信息中提示配置有改变时才从新读取配置信息，读取后存到本地数据库，下次登录直接调用本地数据；
- 2、程序定时刷新或用户单击刷新按钮时读取各个主机组的告警信息，以及每个告警信息所归属的主机信息；



***平台 V1.0

用户名

密码

登录

演示

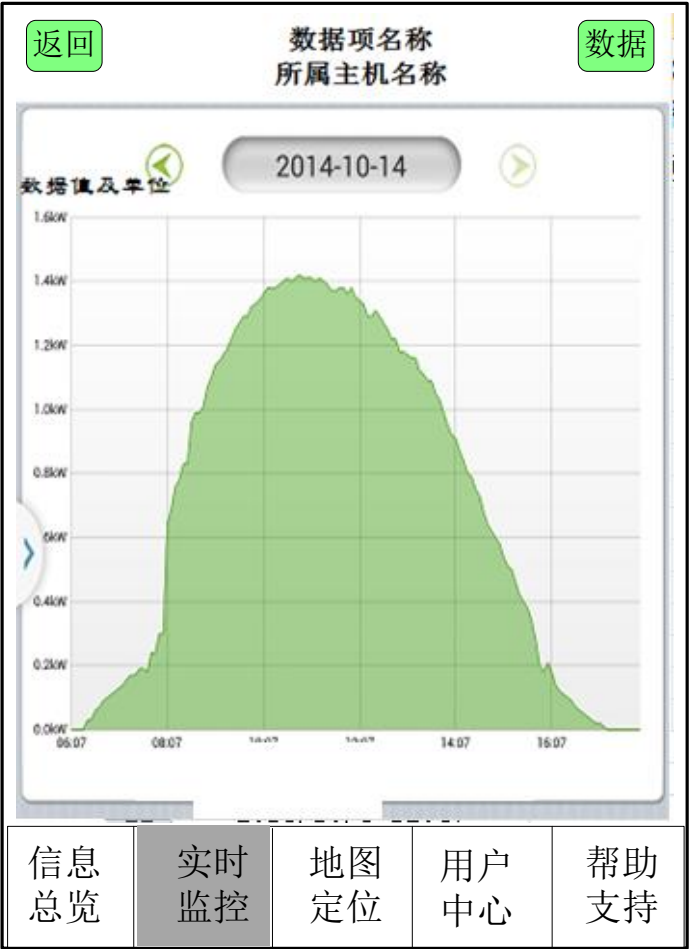
www.***.com

***公司 版权所有

- 1、这里的“演示”按钮指的是直接以账户(guest guest)登陆；
- 2、整个APP用该界面的一个用户名/密码，可以调用json函数【*userLogin*】登录；即进入后：
 - 1) 在用户中心页面，显示用户的名称；
 - 2) 该session在用户没有活动达到设定时间后失效并退出登录；
- 3、界面上其它信息则固定展示；
- 4、上部的图标会自动根据图标的大小放置在居中靠上的位置；

数据刷新时间 2015/8/20 09:07:01 				
主机组I 严重(1) 警告(3) 其它(5) 				
信息*** [严重级别] [发生时间] [归属主机] 描述				
信息*** [严重级别] [发生时间] [归属主机] 描述				
主机组II 严重(1) 警告(3) 其它(5) 				
主机组III 严重(1) 警告(3) 其它(5) 				
主机组IV 严重(1) 警告(3) 其它(5) 				
主机组V 严重(1) 警告(3) 其它(5) 				
主机组VI 严重(1) 警告(3) 其它(5) 				
信息总览	实时监控	地图定位	用户中心	帮助支持

- 这里显示主机状态（以下信息全部通过通过JSON-API获取并显示；）：
- 1、app在每次刷新时获取有问题的主机组，以分主机组（包括其上层目录）的方式列出所有trigger问题（问题名称、严重级别、问题描述、所属主机等信息），并且没个问题都有相关联的主机，单击该主机即可进入实时监控的界面（类似于在此界面单击某个主机后进入的界面）；
 - 2、在主机组的标题栏上要有三类数字显示，表明各类问题总数，分三类并用三类颜色表示（严重以上用红色、警告级别用黄色、其它用绿色）；
 - 3、要求能够采用个推推送服务向客户手机推送警告以上的信息，单击告警信息后进入相应的告警栏目；
 - 4、要求开发者写出各种情况下调用API接口的具体思路，以便确保最优；参考思路如下：
 - 1）进入该页面时，默认为读取本地保存的配置，但如果需要刷新配置（本地为空或者在【登录】时服务器返回的信息中提示需要刷新），那么就从新读取本页的所有配置项；
 - 2）用户可以直接单击顶部右侧刷新按钮刷新数据，也有定时机制在定时刷新本页数据，并且顶部显示最近一次数据的更新时间（而非刷新操作时间）；
 - 3）读取本页配置信息时：可以通过jason函数【**userGroupsGet**】获取所有主机组信息；由此信息可以列出本页的第一级目录所有配置信息；例如：`DvsHostGroup[] groupList = userGroupsGet(sessionID);`
 - 4）进入某个主机组时，通过jason函数【**groupTriggersGet**】获取所有告警信息，并显示在第二级目录；同时针对每条告警信息，可以通过及告警ID通过JSON函数【**groupHostsGet**】获取到对应的HOSTID信息；并将HOST做成一个超链接，以便可以进入实时监控的相应主机项；
 - 5）要求每次只能打开一个主机组，但每次刷新要刷新所有条目上的告警信息数量（并且严重级别以上的则条目背景显示红色，警告级别的显示背景黄色，其他级别则显示背景绿色；按以上优先级选择颜色）；主机组上的告警信息数量由专门接口返回, 参见接口【**groupTriggersSumGet**】；



时间	数据 (单位名称)
2015/9/14 12:47	1
2015/9/15 12:47	1
2015/9/16 12:47	1
2015/9/17 12:47	1
2015/9/18 12:47	1
2015/9/19 12:47	1
2015/9/20 12:47	1
2015/9/21 12:47	1
2015/9/22 12:47	1
2015/9/23 12:47	1
2015/9/24 12:47	1
2015/9/25 12:47	1
2015/9/26 12:47	1
2015/9/27 12:47	1
2015/9/28 12:47	1
2015/9/29 12:47	1
2015/9/30 12:47	1
2015/10/1 12:47	1
2015/10/2 12:47	1
2015/10/3 12:47	1
2015/10/4 12:47	1
2015/10/5 12:47	1
2015/10/6 12:47	1

- 1、登陆后根据返回的登陆信息发现用户配置信息有变更，
则开始导入并显示等待旋转图标；
- 2、具体的一个数据单元格局见页面所示；
- 3、逻辑分层如下：

1) 一个用户可能包含多个主机组，则列出所有主机组；通过JSON-API获取并显示；

2) 一个主机组可能包含多个主机，则列出所有主机（可查看每个主机的地图位置，也可以上传地图位置信息）；通过JSON-API获取并显示；

3) 一个主机包含两大项，一个是实时数据项、一个是控制参数项；
 - 实时数据项：通过JSON-API显示最新数据（可能无单位），以及历史数据列表和曲线图；
 - 控制参数：通过JSON-API控制列表（三种控制类型以内则直接平铺显示，三个以上则弹出选择窗口），通过SOCKET获取并显示最新数据（可能无单位）；

4、要求开发者写出各种情况下调用API接口的具体思路，以便确保最优；

1) 进入该页面时，默认为读取本地保存的配置，但如果需要刷新配置（本地为空或者在【登录】时服务器返回的信息中提示需要刷新），那么就从新读取本页的所有配置项；

2) 用户可以直接单击顶部右侧刷新按钮刷新数据，也有定时机制在定时刷新本页数据，并且顶部显示最近一次数据的更新时间（而非刷新操作时间）；

3) 读取本页配置信息时：可以通过jason函数【**userGroupsGet**】获取所有主机组信息；再通过jason函数【**groupHostsGet**】获取某主机组下所有主机信息；再通过jason函数获取该主机下的所有ITEM信息；其中ITEM信息分两类，“Application_RealtimeData”和“Application_CtrlParams”，可以分两次调用【**hostItemsGet**】函数（传递不同的application参数）获取类型ITEMS;刷新完毕后，主机组栏目要能够显示主机个数，主机栏目要能够显示ITEM总数，实时数据和控制参数栏目则相应显示其下ITEM的个数；

4) 针对每个ITEM信息，要展示是ITEMNAME，以及其它属性（比如实时数据项的单位、控制参数项的各类控制）等；
《注意：一个控制参数ITEM对应的控制项属性以及相应的操作，比如ITEM项名称是“电源”，那么该项对应的控制项操作可能是“开、关”，那么这些项应该平铺（radio控件）或以下拉列表（大于三个）方式排列在“电源名称”后面，其后还要有一个提交按钮以便通过TCP-SOCKET提交用户所选择的操作；>>>>>目前控制项的各项接口参数正在开发中，开发者先做数据项，稍后给出》

5) 刷新实时数据或控制参数数据项时（定时或者用户单击刷新按钮）获取当前展开的主机下各个ITEM数据项，其中包括实时数据项和控制参数数据项，采用jason函数【**historyDataGetLast**】；注意：获取实时数据时该函数的时间区间要填写最近三天的时间（即从当前时间起往后推三天）、控制参数则需要填写从1970年到当前的时间区间、数据类型参数则要根据数据项类型来依次获取（即将数据项根据数据类型分成不同集合，然后再输入不同集合及相应的数据类型依次调用该函数）；

6) 每个数据项后还有个历史数据/图示按钮（采用数据列表图标和曲线类图标代替，如果是STRING类型，则只有历史数据，如果INT或者FLOAT则既有数据列表又有图示）；单击进入历史数据或图形列表时，如上图所示，按天展示数据，左右选择日期可以自动更新数据或图形页面；如果一页数据列表当前界面放不下，则向下滑动时再读取并显示下页数据；无论是图示页面还是数据页面，单击返回按钮都表示回到实时数据的页面；再者，如果同时有历史数据表和图示，则在两者的界面上可以互相切换；历史数据采用jason函数【**historyDataGetPage**】，指定主机、某个项目、数据类型、时间区间、PageIndex、PageSize等信息（APP不需要这里返回PageNum）；

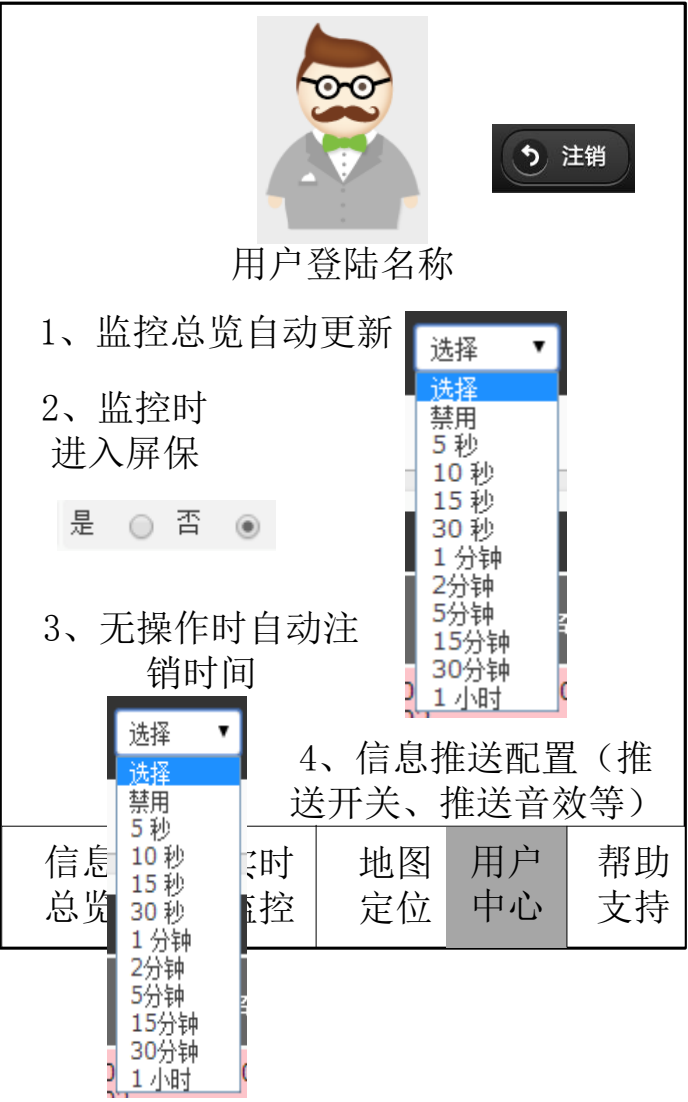
7) 每个主机后面有地图按钮（用地图图标代替），单击地图按钮，如果有位置信息（参见接口【**groupHostsGet**】）则调用地图界面显示在地图上；否则调用地图界面并定位到当前点（可以直接切换到定位地图TAB，当前位置出现一个地图大头针，界面上其它的HSOT同样显示，大头针上面为“为<主机名>添加地图定位”），单击该大头针后（弹出菜单，一个是添加定位，一个是取消）并确认后即可上传位置坐标（参见接口【**hostLocation**】）；

8) 支持循环搜索配置项（主机组名或主机名或数据项目名称）的功能，因为配置项是存储在本地的，所以在本地存储信息中搜索到某个项后，则自动展开并定位到该项（不必刷新数据）；如果继续搜索则从该项开始往后，搜索完毕再从头开始；

9) 特别注意：以上所有配置项或者数据项，比如HOST, GROUP, ITEM等等都需要以返回的信息中的ID作为唯一标记，不能使用其它信息，其它信息都是可能重复的！！



1、参见2-实时监控，7）；



一、监控数据自动更新

- 1、这里以APP的方式显示一个可以自动刷新数据页面的列表，第一期能够刷新的页面就是“1-信息总览”和“2-实时监控”页面,同时该数值将可以通过后天控制提交到服务器；
- 2、一旦APP进入后台，则停止自动刷新，进入前台则继续；
- 3、本页面的配置可以保存在本地，下次登录进入时仍然继续；

二、监控时进入屏保

- 1、在监控数据自动更新时间有效地情况下，启用该项选择；相应地，无效时禁用该项选择；
- 2、选择方式可以按照APP的方格展现；

三、无操作时自动注销时间

即在用户没有任何操作时，用户的自动注销时间

四、推送配置：比如是否打开推送，是否要打开音效等；



介绍信息内容

QQ: ***

邮箱: ***@qq.com

您的称呼

联系方式

具体内容

提交

信息
总览

实时
监控

地图
定位

用户
中心

帮助
支持

该页面

- 1、最上面是logo图标；
- 2、接下来是介绍信息（简介、联系方式、网址等），这些信息都固定显示即可；
- 3、再下来是联系我们，即可以提交信息到服务器，参见接口【***userMessage***】；