AR虚拟教学

移动互联解决方案

健康管理-技术部

**一、基本架构和思路**

服务器



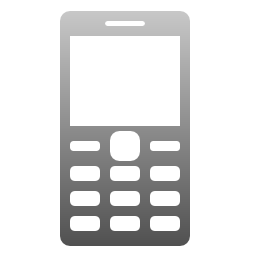
上传个人信息、机器编号、AR操作指令

下发使用者信息和操作指令

上传报告结果

查询报告结果





APP

AR显示屏

扫码绑定仪器

**AR接入信息流模型**

此方案省略了手机与AR（以下简称设备）之间的连接，只要设备与服务器建立稳定的连接，即可从手机推送个人信息、操作指令到仪器。这需要设备具备寻址的可能，需要为每台设备配置一个唯一的编号，生成一个唯一的二维码。通过用户扫码的机制，将具体的用户与具体的某一台机器进行绑定起来。

通讯采用WebSocket方式连接（消息交互都用字符串格式），设备通过wifi上网，上网后自动连接远程服务器。通过ping-pong维持长连接，每隔10秒发一次。之所以采用WebSocket连接而不是传统的HTTP连接，是因为WebSocket连接是双向的持久化连接，服务器可以直接向客户端（设备）推送数据。

在连接建立时，设备上传设备编号，服务器将此设备编号与WebSocket实例绑定，建立对应关系。以后根据设备编号选定不同的WebSocket实现与终端进行通讯。

服务器端采用RESTful架构，设备、手机与服务端的交互都是纯粹的HTTP GET和POST两种请求。

目前根据AR这边的应用场景，初步想到以下几个操作指令场景：**登陆服务器、下发用户信息**（在AR中显示该用户相关信息）**、解锁设备**（解锁二维码）**、加锁设备**（加锁二维码）、**启动设备、暂停设备、停止设备、上传报告。**下面列出这几种指令场景的通讯协议，仅供参考。

**二、登录服务器**

设备通电后连接服务器，然后发送设备登录验证消息{"message":"register","deviceID":"机器id"}，如{"message":"register","deviceID":"AR000001"}，服务器处理成功返回{"message":"register success"}，失败返回{"message":"register failed"}。若出现网络中断或其他因素导致设备与服务器之间的websocket长连接断开，需要设备自行与服务器建立连接，并发送登录消息。

**三、服务器下发消息相关操作**

**3.1 解锁设备**

解锁二维码页面，并携带用户数据给设备，服务器下发如下格式：

{

"message" : "unlock", //固定值，表示用户数据

"memberId" : "13902345678", //用户id

"memberName" : "用户1", //用户名称

"memHeadImg" : "http://img.com/1.jpg", //用户头像

}

设备处理成功{"message":"unlock ok"},失败{"message":"unlock failed"}

**3.2 加锁设备**

服务器下发如下格式：

{"message":"lock"}

设备处理成功{"message":"lock ok"},失败{"message":"lock failed"}

**3.3 启动设备**

服务器下发如下格式(courseId为课程编号，与视频名称对应)：

{"message":"startup","courseId":"12153"}

设备处理成功{"message":"startup ok"},失败{"message":"startup failed"}

**3.3.1 启动一级视频**

服务器下发如下格式(sectionList为播放视频序列，courseName为视频名称)：

{"message":"executeList","sectionList":[2,5,8],"courseName":"太极"}

设备处理成功{"message":"executeList ok"},失败{"message":"executeList failed"}

**3.4 结束课程**

服务器下发如下格式：

{"message":"closure"}

设备接收指令处理成功返回（该数据同样作为AR设备自行上传评分数据）

{

"message" : "closure ok",

"memberId" : "13902345678", //用户id，该值获取由服务器下发的用户数据

"score" : "7.6", //分数

"calorie" : "3", //卡路里，单位Kcal

"times" : "10", //锻炼时间，单位min

"isOver" : false //是否播放完，boolean类型，true是，false否

}

,处理失败{"message":"closure failed"}；

当服务器收到{"message" : "closure ok",...}时，会对该数据进行处理,相应的也会返回响应结果，

服务器处理成功返回{"message":"process successful"}，失败返回{"message":"process failed"}

**\*3.5 TV显示课程选择提示页**

{"message":"choose course"}

**四、设备上传消息相关操作**

4.1 设备上传分数、卡路里，格式如下

{

"message" : "closure ok",

"memId" : "13902345678", //用户id，该值获取由服务器下发的用户数据

"score" : "7.6", //分数

"calorie" : "3", //卡路里，单位Kcal

"times" : "10", //锻炼时间，单位min

"isOver" : true //是否播放完，boolean类型，true是，false否

}

服务器收到后处理成功返回{"message":"process successful"}，

失败返回{"message":"process failed"}

4.2 设备自行加锁消息

{"message":"isLock"},服务器收到成功返回{"message":"receive success"}，失败返回{"message":"receive failed"}

**五、AR更新**

当AR设备有更新版本时，通过服务器下发更新指令到AR设备，AR设备收到指令后进行一系列更新操作。发送指令格式为{"message":"ProgramUpdate"}，更新成功返回{"message":"update ok"}，更新失败返回{"message":"update failed"}

**六、视频更新**

6.1

后台提供一个http接口，返回视频列表，ar客户端根据version数值比较是否进行视频更新

请求url: http://dev-apisail.healthmall.cn/api/arvideo/getCourseList

HTTP请求方式：POST

请求头：*application/json*

发送数据格式：

{

"**deviceIdentity**": "**AR000000**", //设备号

"**all**": false //全量更新标记，true时获取全部视频

}

接口调用成功返回格式：

1.有视频：

{

"code": 2000,

"msg": "OK",

"data": [

{

"subsectionId": 29,

"subsectionName": "开合跳",

"videoUrl": "http://video.com/1.mp4",

"lastEditTime": 1513668083000

},

{

"subsectionId": 31,

"subsectionName": "yt",

"videoUrl": "http://video.com/2.mp4",

"lastEditTime": 1513670056000

}

],

"succeed": true

}

2.无视频

{

"code": 2000,

"msg": "OK",

"data": [],

"succeed": true

}

失败返回值：

|  |  |
| --- | --- |
| code | 描述 |
| 3406 | 值不能为空 |
| 70010 | 设备号不存在 |

6.2视频更新成功回调接口

请求url: http://dev-apisail.healthmall.cn/api/arvideo/videoUpdateCallback

HTTP请求方式：POST

请求头：*application/json*

发送数据格式：

{

"**deviceIdentity**": "**AR000000**", //设备号

"**subsectionId**": 41, //视频序号

"**lastEditTime**": 1513676585000 //取回视频列表的lastEditTime值

}

接口调用成功返回格式：

{

"code": 2000,

"msg": "OK",

"data":null,

"succeed": true

}

成功状态码：

|  |  |
| --- | --- |
| code | 描述 |
| 2000 | 表示成功 |

失败返回值：

|  |  |
| --- | --- |
| code | 描述 |
| 3406 | 值不能为空 |
| 70010 | 设备号不存在 |
| 70033 | 二级视频编号不存在 |