# 系统设计方案

（information collection system of 3D machine room equipment）

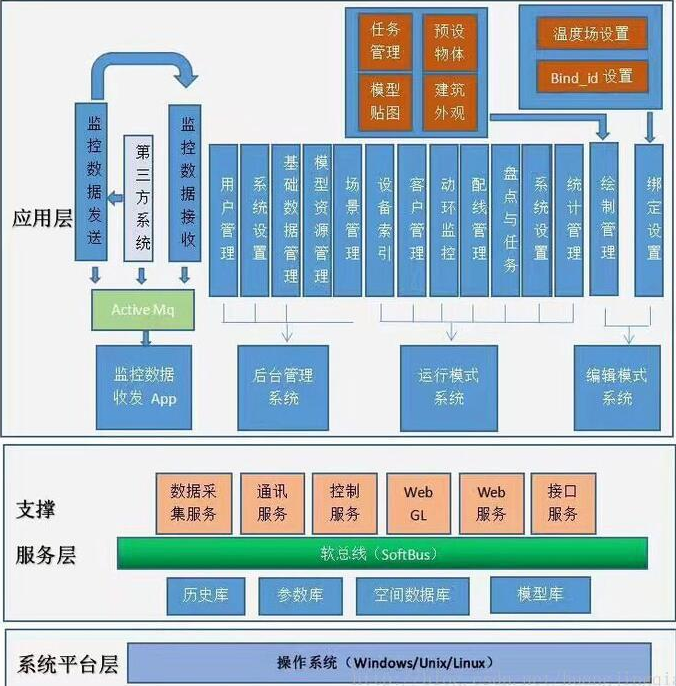
（Three-dimensional computer room equipment information acquisition system）

## 系统描述

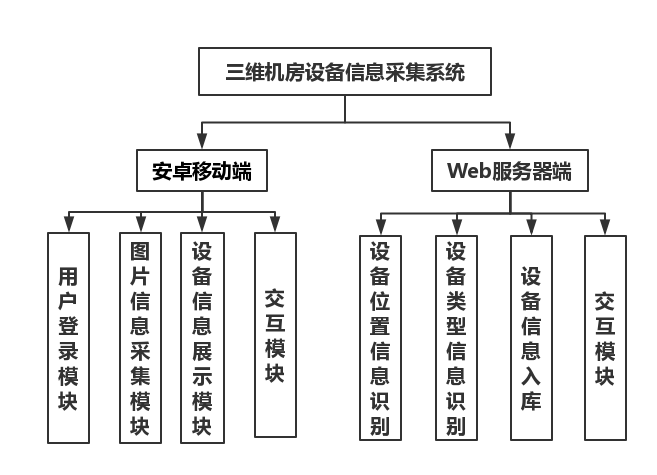
近年来随着社会信息化智能化的程度的提升，机房设备的数量与日俱增，如供配电系统、UPS电源、空调、服务器、机柜等，必须时刻更新机房内设备的状态和信息。因此，机房设备的信息管理就显得很重要。为了提高机房维护和管理的可靠性和实效性，本文将研究设计并实现一个智能化的三维机房设备信息采集系统，为机房设备信息采集管理提供更方便高效的途径。该系统由两大子系统组成：一是服务器端（Web管理端），用来管理基础数据，包括存储设备信息、更新设备信息，同时为机房管理人员提供数据展示（待选：提供数据统计分析报表的功能）；另一个是Android移动端，用于数据采集上传、数据同步、数据展示等。两大子系统之间互相通信，同步设备信息。



系统架构



## 系统模块结构：



## 详细功能模块介绍

（1）Android移动端的图片信息采集模块

该模块负责扫描设备位置信息图片和设备图片，将位置信息图片信息输入空间信息识别模块处理；将设备图片输入设备信息识别处理模块处理。

（2）Android移动端的设备信息展示模块

通过接受用户的search输入，查询用户所需要的信息，展示在移动端页面上。

（3）Android移动端的交互模块

该模块主要负责给服务器端发送图片信息、搜索信息、设备状态信息等，同步更新数据库信息。

（4）Android移动端的客户登录模块

客户打开该应用时，登录个人账号查看该账号下管理的机房信息。

（5）Web服务器端的设备位置信息识别

该模块负责接收Android移动端扫描的设备位置信息图片，利用目标检测算法:RCNN、Fast RCNN 、Faster RCNN等对图片上的数字进行检测，然后通过LeNET等神经网络模型对图片上的数字进行识别采集，最后获得该设备的机柜编号和U位数。

（6）Web服务器端的设备类型信息识别模块

该模块负责接收客户端扫描得到的设备图片，将图片输入设备类型识别模型如VGG-16等神经网络处理，采集设备的类型信息。

（7）Web服务器端的信息入库

Web服务器端通过监听Android移动端的数据请求，并将其存储到对应的数据库表格中。

## 系统工作流程

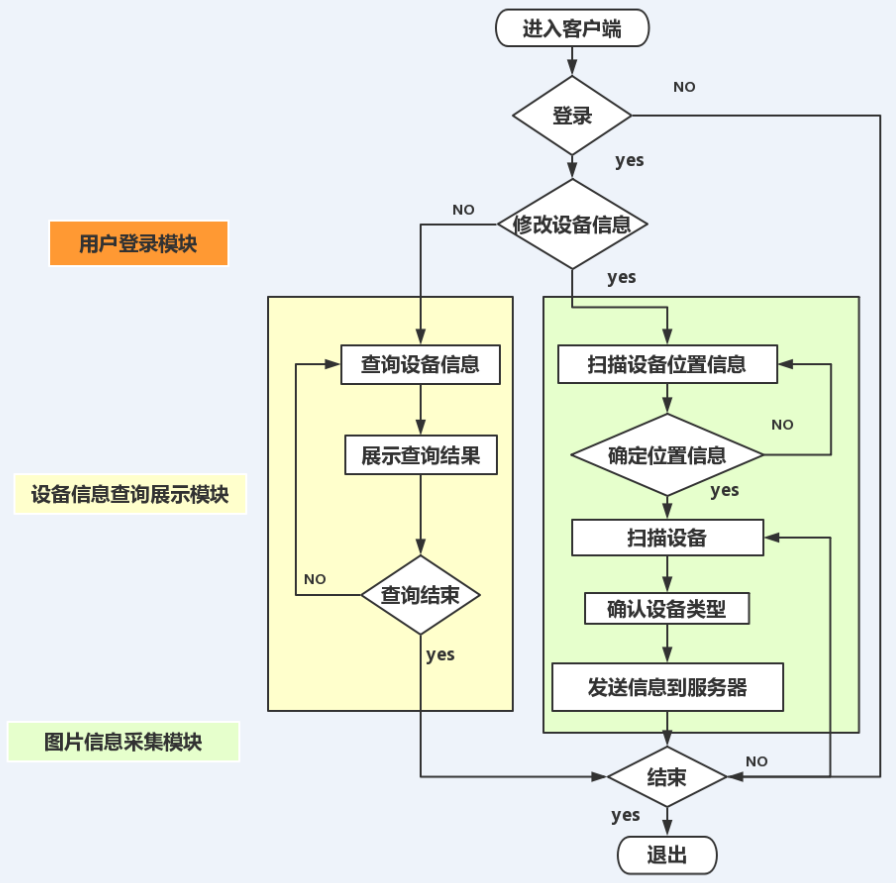


图3：客户端流程图

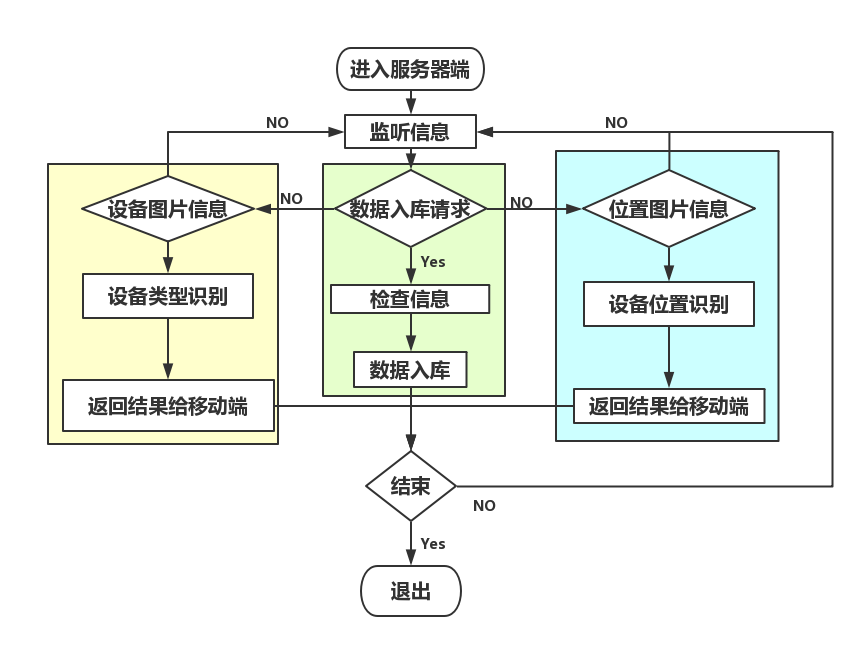


图4：服务器端流程图

## Android移动端开发（Hydrid开发模式）

### 混合开发模式的优点

hybrid开发有native的性能，也有web app开发的速度。

<https://blog.csdn.net/jingwen3699/article/details/68922945>

Hybrid App主要以JS+Native两者相互调用为主，从开发层面实现“一次开发，多处运行”的机制，成为真正适合跨平台的开发。Hybrid App兼具了Native App良好用户体验的优势，也兼具了Web App使用HTML5跨平台开发低成本的优势。目前已经有众多Hybrid App开发成功应用，比如淘宝、美团、滴滴、爱奇艺、大众点评、微信等知名移动应用，都是采用Hybrid App开发模式。

|  |  |  |  |
| --- | --- | --- | --- |
| 特性 | Native APP | Hibrid APP | Web APP |
| 开发语言 | 只能用native | **Native或者web开发语言或者只用web开发语言** | 只用web开发语言 |
| 代码移植性与优化 | 无 | **高** | 高 |
| 访问针对特定设备的特性 | 高 | **中** | 低 |
| 高级图形 | 高 | **中** | 中 |
| **升级灵活性** | **低，通过应用商店升级** | **中，部分可不通过应用商店升级** | **高** |
| **跨平台、多种设备兼容** | **否** | **是** | **是** |
| **设备本地功能：通讯录、摄像头、语音、本地文件** | **可以** | **可以** | **不可以** |
| **离线** | **可以访问** | **不可以** | **不可以** |
| 安装包 | 大 | **中** | 小 |
| 安装体验 | 高，从应用商店安装 | **高，从应用商店安装** | 中，通过移动浏览器安装 |

开发工具:IonicLab、Sublimetext、AndroidStudio;

服务器端:PHP+MySQL

Jquery mobile、Vue、React

### hybrid app中的原生页面和h5页面是如何交互的

## Android移动端UI设计

## Web服务器端数据库设计（mysql数据库）

用户信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型 | 说明 | 能否为空 |
| 管理员id（u\_id） | 自动编号 | 常整型，递增，主键 |  |
| 管理员名称（u\_name） | 文本 | 字段大小20 |  |
| 账号密码(u\_password) | 文本 | 字段大小20 |  |
| 管理员权限（u\_auth） | 整数 | 常整型 |  |
| 上次登陆时间 | 日期 | 字长大小8 |  |
| 用户登陆历史 | 文本 | Text |  |

机房信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型 | 说明 |  |
| 机房编号（r\_id） | 数字 | 常整型，递增，主键 |  |
| 管理员id(u\_id) | 数字 | 常整型 |  |
| 机房名称(r\_name) | 文本 | 字段大小20 |  |
| 所属单位（r\_owner） | 文本 | 字段大小20 |  |
| 机房位置(r\_pos) | 文本 | 字段大小20 |  |
| 机房长度(r\_length) | 整数 | 常整型 |  |
| 机房宽度(r\_width) | 整数 | 常整型 |  |
| 机房高度(r\_height) | 整数 | 常整型 |  |
| 机房全图（r\_pic） | 整数 | 常整型 |  |
| 备注 | 文本 | 字段大小80 |  |

机柜信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型 | 说明 |  |
| 机房编号(r\_id) | 数字 | 常整型，递增，主键 |  |
| 机柜编号(c\_id) | 数字 | 常整型，递增，主键 |  |
| 机柜名称(c\_name) | 文本 | 字段大小20 |  |
| 机柜总U数(c\_un) | 整数 | 常整型 |  |
| 上次更新时间(last\_modified) | 日期 | 字段大小8 |  |
| 机柜状态(c\_status) | 整数 | 常整型 |  |
| 备注 | 文本 | 字段大小80 |  |

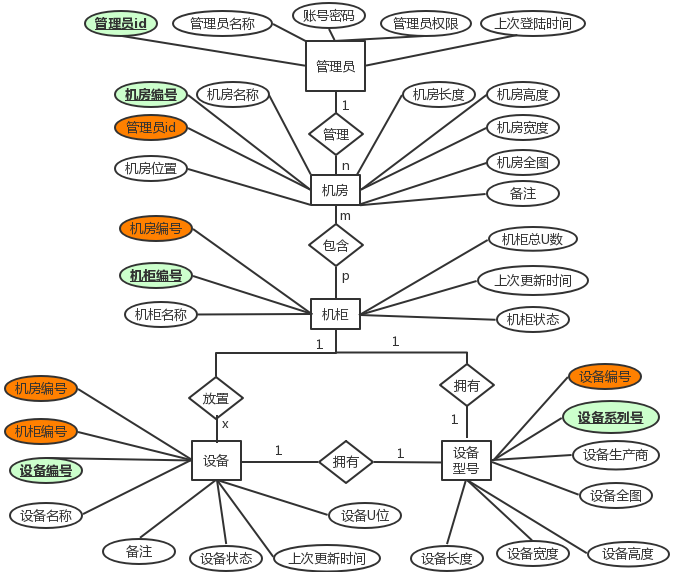
设备基础信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型 | 说明 |  |
| 机房编号(r\_id) | 数字 | 常整型 |  |
| 机柜编号(c\_id) | 数字 | 常整型 |  |
| 设备编号(d\_id) | 数字 | 常整型，递增，主键 |  |
| 设备名称(d\_name) | 文本 | 字段大小20 |  |
| 设备所在U位(d\_u) | 数字 | 常整型 |  |
| 上次更新时间(last\_modified) | 日期 | 字段大小8 |  |
| 设备状态（故障、维修、正常）(d\_status) | 整数 | 常整型 |  |
| 备注 | 文本 | 字段大小80 |  |

设备型号信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型 | 说明 |  |
| 设备编号(d\_id) | 数字 | 常整型，递增，主键 |  |
| 设备长度(d\_length) | 数字 | 常整型 |  |
| 设备宽度(d\_width) | 数字 | 常整型 |  |
| 设备高度(d\_height) | 数字 | 常整型 |  |
| 设备全图(c\_pic) | 文本 | 字段大小20 |  |
| 设备位置图片（p\_pic） | 文本 | 字段大小20 |  |
| 制造商(c\_producer) | 文本 | 字段大小20 |  |
| 设备系列号(c\_seq) | 文本 | 字段大小20 |  |

E-R图



## Android移动端和Web服务器端数据交互

客户端给服务器发送三种请求：整理后的信息入库请求、上传位置图片信息请求、上传设备图片信息的请求、设备信息查询请求

服务器给客户端发送三种请求：发送生成的位置图片信息、发送生成的设备类型信息、发送查询结果、发送信息入库后的ACK