

# 基于云平台的老年公寓服务系统设计

□李霖 李庆达 区善宁 北方民族大学

【摘要】 我国已经步入人口老龄化阶段，然而我国的老年公寓建设依旧处于传统，落后的模式。随着物联网技术的逐渐兴起，本项目设计了一款拥有智能化，高效化的老年公寓服务系统，本文详细介绍了老年公寓系统。

【关键字】 JAVAWEB 云平台 STM32 单片机

## 一、引言

中国的物联网正处于逐渐兴起状态，很多设计方案仅仅只是解决了用户的基本功能，却远远没有达到智能化高，成本低，服务性强的智能家居产品。本项目的特点是通过大数据分析出老年人平时的生活习性，例如采集用户每天第一次打开窗帘的数据，将数据进行整合通过算法分析出精确的时间，当用户开启智能模式后在睡眠时，当达到该时间点系统将会自动打开窗帘，让用户起床的是被射进室内的第一缕阳光，用户可以通过响应式布局的 WEBAPP 页面，实时性的监视室内安全和了解室内的环境数据，通过这些数据用户可以第一时间采取相应的措施去高效的解决问题。由此可见本项目通过云存储和云计算，将物联网中的数据整合并精准计算，体现出物联网方向数据的应用价值，并给老年人呈现出一种科技感极强，幸福感高的智能家居应用。

## 二、研究意义

本项目旨在设计一个低成本，高效率，体验度高基于云平台的智能服务性系统，推动老年公寓科技化建设的脚步，相比于普通的智能家居方案设计其优点有三个。

1、对象存储 OSS 是云平台功能某部分，可以理解为一个灵活性高，存储量大的存储集群，数据可以自动备份多重冗余防止出现意外的数据丢失，提供多重的安全防护避免用户日常数据被非法盗用，且成本低无需物业聘用服务器运维人员与托管费用，从而保证企业只需考虑业务需求中采集数据，处理数据部分，而数据的持久化阶段中的安全性，稳定性全部交由云平台去处理，实现了项目的快速开发，降低企业成本，且极大的提高数据安全性。

2、WEBAPP 实现的响应式布局，可以自适应于各种移动端与电脑屏幕，不需要团队花费大量的时间成本与技术资源去即开发移动 APP 又开发和网页的局面。

3、通过传感器采集用户习性数据，并通过算法精准计算数据因此来衡量用户习性，从而自动向用户推送服务给用户一种被关怀的体验。

## 三、项目架构

### 3.1 项目总体架构：STM32、云平台、JAWAWEB

STM32 是一款微处理器可以内部控制各种外设传感器，由于 STM32F 的功耗远低于 CPU，所以能够在一定程度上降低产品的功耗，这也正是其为什么能被称为传感器中央处理器（Sensor hub）的原因所在，因此利用 STM32 的可扩展性，低功耗做为硬件设计中的微处理器，本套系统通过 STM32 外接摄像头，环境数据传感器，继电器，GPRS 模块，首

先 STM32 将传感器采集的数据进行内部处理，用 TCP/IP 协议将硬件采集数据通过 GPRS 模块传到 JAWAWEB 后台，JAWAWEB 后台将接受到的数据依据项目中所需求的业务逻辑二次处理，再通过 JAWAWEB 前端实现了公寓内实时的数据可视化和设备远程控制功能，而云平台相当于一个容纳 JAWAEE 的服务器，去保证项目的安全性，稳定性。

### 3.2 云平台产品选择

每个公司下的云平台都提供多种产品服务，本项目选择的是阿里旗下的产品服务，其中主要选择了 ECS、CDN、OSS、SLB，将传统项目中单台服务器跑整个应用程序，分离成不同功能在不同服务器中。

ECS：本质就是一个服务器，运行在云上的一个服务，主要作用做大数据处理，跑本项目程序。

CDN：分布式的缓存服务器，可以加速项目中静态文件加载，达到了缓解网站高并发下带宽瓶颈的尴尬，缓解服务器的 IO 压力。

OSS：文件存储服务器，将采集的数据持久化便于数据的存取。

SLB：均衡负载。

## 四、在 JAWAWEB 上进行核心设计的设计模式

Model（模型）：属于 MVC 设计模式的 M，在 WEB 服务器中，用 JAVA 语言编写项目中相关的数据类，将项目中复杂的数据拆分成独立的模块，把相似的数据属性封装在 JAVA 类中，利用了 JAVA 中面向对象的设计思想，使项目中数据传输逻辑更清晰且简化开发难度，在本项目中模型数据对象负责在云平台下的 OSS 进行存取数据。

View（视图）：属于 MVC 设计模式中的 V，依据创建模型数据，处理数据显示的部分负责在 WEB 前端进行数据可视化，在本项目中采用了 WEBAPP 的响应式布局，使 WEB 前端不仅能在电脑显示器上进行数据展示，还可以在 Ipad，手机移动端等设备中根据设备的分辨率进行自适应，降低了前端设计的复杂度。

Controller（控制器）：属于 MVC 设计模式中的 C，是处理业务逻辑的主要部分，用户通过编写自己定义的 JAVA 类向模型数据对象发送数据，在本项目中我们定义与业务逻辑相关的 JAVA 类，接收硬件通过 TCP/IP 协议发送的数据并进行相应的业务处理，把数据发送到相匹配的模型数据中，然后通过自己封装的阿里云开放的 API（接口）工具类，利用工具类将数据存储到 OSS 中，实现了数据的持久化，从而通过工具类进行数据的存取。