**Charity V1.0系统架构设计说明书**

# 概述

本文主要讲述Charity V1.0系统框架设计，包括了架构分析、设计思想、架构体系、模块划分等。

# 目的

本文针对Charity V1.0系统框架设计做详细的说明。

# 架构设计

## 架构分析

系统设计为BS架构，系统可以分成客户端、WebAPI应用服务器、数据库服务器，采用三层架构设计及WebAPI技术，主要为满足PC网站及移动设备访问需求，尽可能做到任何设备均可通过应用服务器访问该系统，完成捐款等事宜。

## 设计思想

客户端与服务端分别开发，并不受开发语言限制，客户端、服务端按业务类型进行分类处理。尽量减少模块间耦合，以使应用服务器更适合客户端的功能需求。

## 架构体系

 CharityV1.0按架构层次分为：UI、Service、DAL等三层，业务实体和日志为公共组件。

UI层是位于整体框架最顶端的层，该层提供用户界面。UI分为2块，前端业务实现和后台管理。其中前端业务实现包含基金会相关信息、项目捐款和支付三大功能模块；后台管理主要包含用户（普通用户、合作伙伴、操作员）管理、项目审核、合作伙伴管理、权限管理等。将UI层分离开可使整套系统在B/S,移动客户端等不同的系统体系架构进行呈现。

Service层位于UI层下方，采用WebAPI实现应用服务，主要用于UI层与DAL层之间的通信。Service层将UI层中三大功能模块细分为对用户、项目、捐款和支付等的相关操作。

DAL层位于整个框架最底部，提供操作数据库CRUD的基础类库。

## 模块划分

### 模块描述

**客户端：**

基金会相关信息：包含基金会发布的新闻动态、合作伙伴信息、项目信息、款项使用情况等。

项目捐款：用户可在网站浏览项目并对感兴趣的项目进行捐款操作。

支付: 用户选择需要捐款的项目，浏览项目详情信息并通过支付宝完成捐款操作

后台管理: 主要包含角色管理、权限管理、用户管理、项目审核等功能

**服务端包含：**

1. 用户管理：

提供对普通用户、合作伙伴及基金会操作员的管理，基金会操作人员可通过该服务对合作伙伴进行审核及相关用户操作等

1. 系统管理

用户可以通过系统管理服务对基本资料及资金情况进行管理，操作员可进行角色、权限等功能管理

1. 项目管理

项目相关人员可以对项目申请、审核、发布等进行管理。

1. 捐款管理

通过该服务处理用户的捐款情况，处理月捐、一次性捐款及项目捐款等

1. 支付服务

该服务提供用户款项的支付功能，初步实现支持支付宝支付方式