

# 软件开发设计文档

## 文档管理信息表

项目名	VR 资讯网站
版本	V1.1
创建时间	2017.05.01
创建人	陈锐彬
最新发布日期	2017.06.24

# 目 录

<b>1</b>	<b>开发规划.....</b>	<b>1</b>
1.1	开发环境和工具 .....	1
<b>2</b>	<b>总体设计.....</b>	<b>2</b>
2.1	基本设计描述 .....	2
2.1.1	SpringMVC 架构图.....	3
2.1.2	系统部署结构图.....	3
2.1.2	系统总体逻辑结构图.....	3
2.2	模块列表 .....	6
<b>3</b>	<b>数据结构.....</b>	<b>7</b>
<b>4</b>	<b>接口规范.....</b>	<b>8</b>
4.1	<模块 1 API>.....	8
4.1.1	Interface1 .....	8
4.1.2	Interface2 .....	8
4.2	<模块 2 API>.....	9
4.3	<模块 3 API>.....	9
4.4	<数据库 API> .....	9
<b>5</b>	<b>模块设计.....</b>	<b>10</b>
5.1	MODULE1 设计 .....	10
5.1.1	模块设计描述.....	10
5.1.2	模块界面描述.....	10
5.2	MODULE2 设计 .....	11
<b>6</b>	<b>附录.....</b>	<b>12</b>
6.1	第三方组件 .....	12
6.2	参考资料 .....	错误!未定义书签。
6.3	附加文档 .....	错误!未定义书签。

## 开发规划

### 开发环境和工具

#### 开发工具

工具	作用
Myeclipse	后端开发
Mysql+Navicat	数据库搭建和管理
Webstrom+Chrome	前端开发
Xshell+Tomcat	服务器搭建和部署

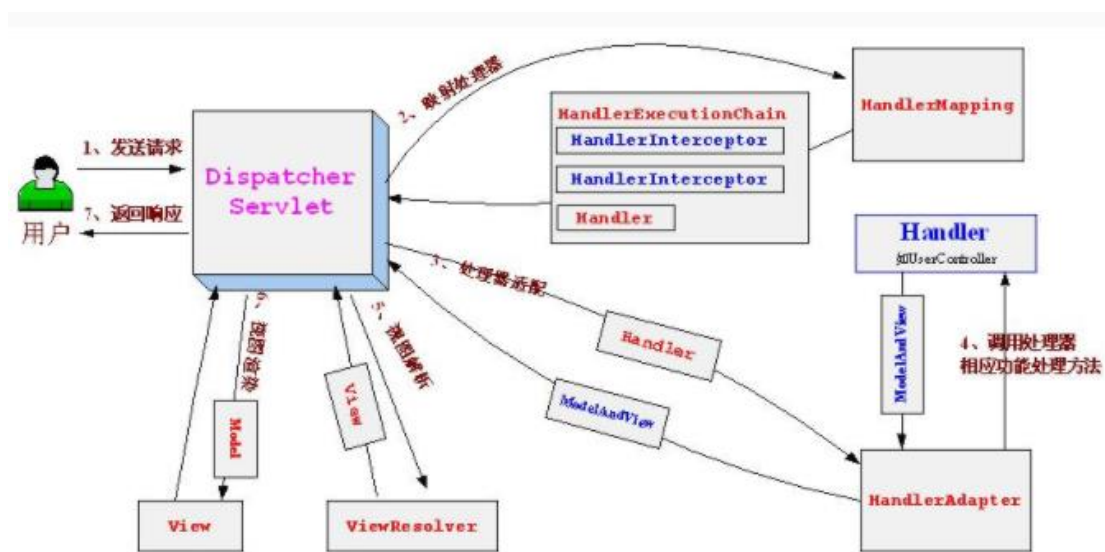
## 总体设计

### 基本设计描述

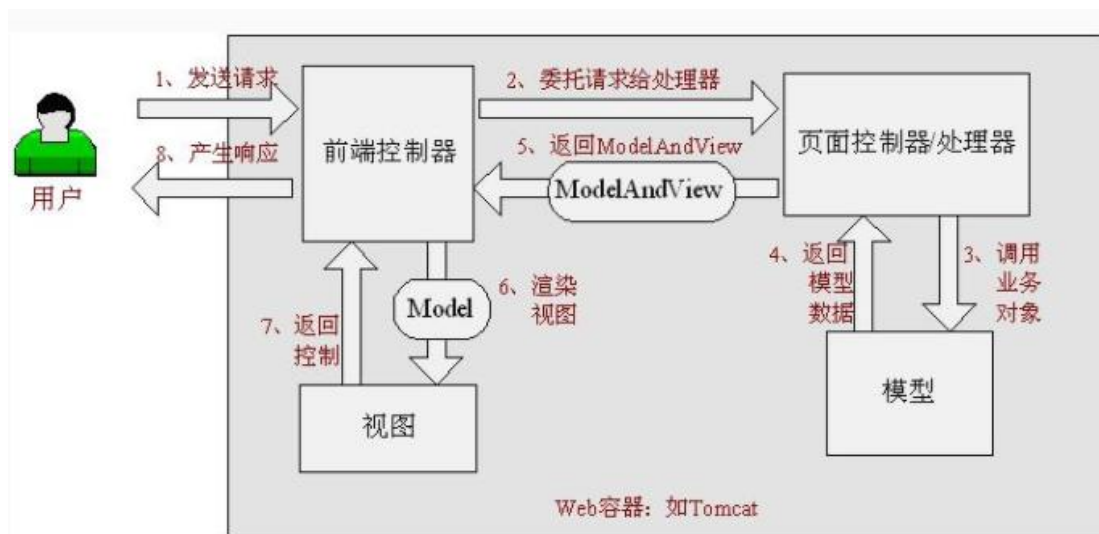
本项目采用 springmvc 作为服务端框架，采用 mybatis 开发持久层进行数据库交互，前端视图开发采用 jsp 开发，爬虫子系统采用 python 开发。

采用 springmvc 是因为该框架能够非常简单地设计出干净的 web 层，开发简单且效率高，易于敏捷开发；同时框架提供了强大的约定优于配置的契约式编程支持，以及灵活的 URL 到页面控制器的映射，耦合度高，利于开发稳定易于维护的系统。

### springmvc 框架图



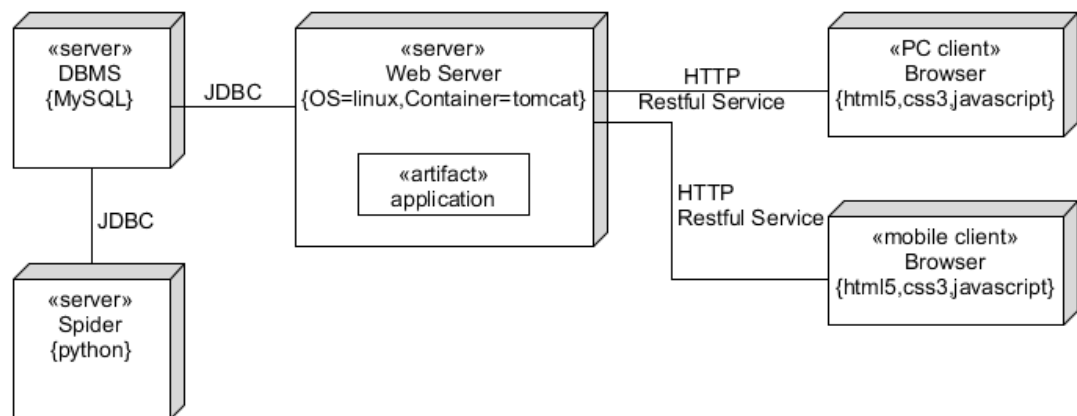
### springmvc 处理请求流程图



如图，Spring MVC 大致的执行流程如下：

- 1、首先浏览器发送请求给前端控制器 DispatcherServlet, DispatcherServlet 根据请求信息, 基于一定的原则选择合适的控制器进行处理并把 请求委托给它。
- 2、页面控制器接收到请求之后进行功能处理, 首先需要收集、绑定请求参数到一个对象(命令对象), 并进行验证, 然后将该对象委托给业务对象进行处理(service 层); 业务对象处理之后控制器将返回一个 ModelAndView(模型数据和逻辑视图名);
- 3、DispatcherServlet 根据返回的逻辑视图名, 选择合适的视图进行渲染(界面展示、资源加载), 并把模型数据传入以便视图渲染。
- 4、前端控制器将响应返回个客户端浏览器。

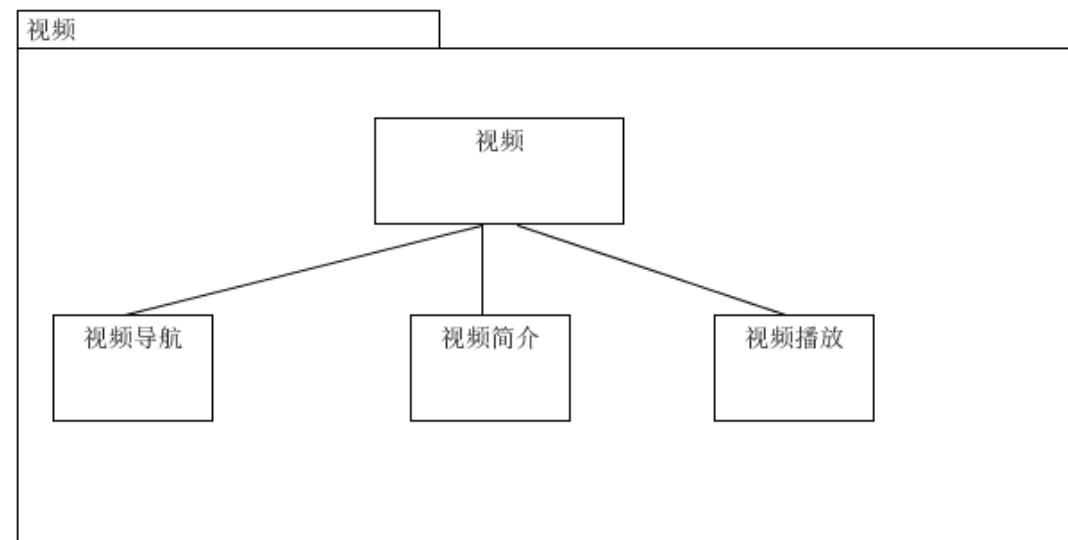
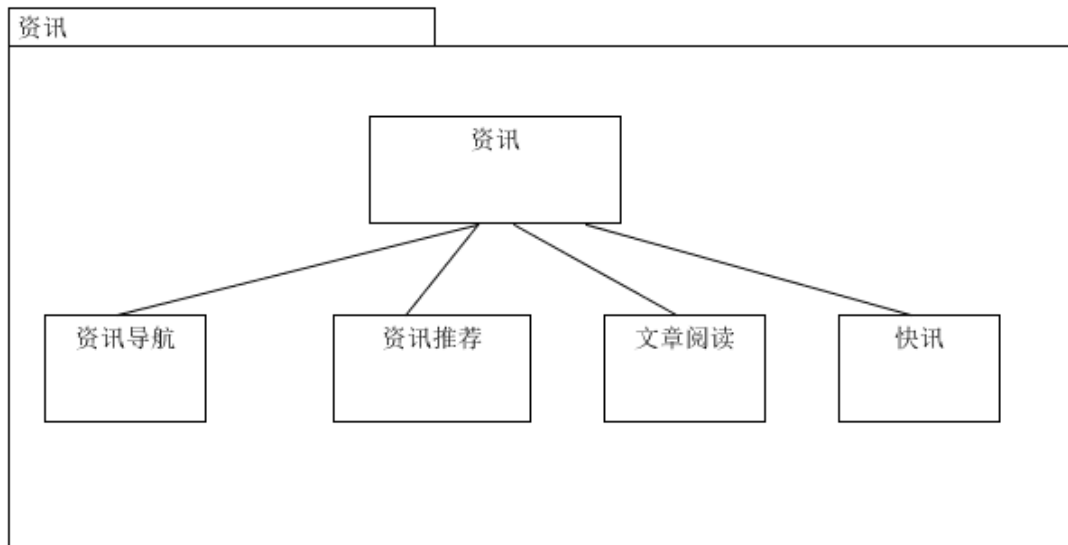
### 系统部署结构图

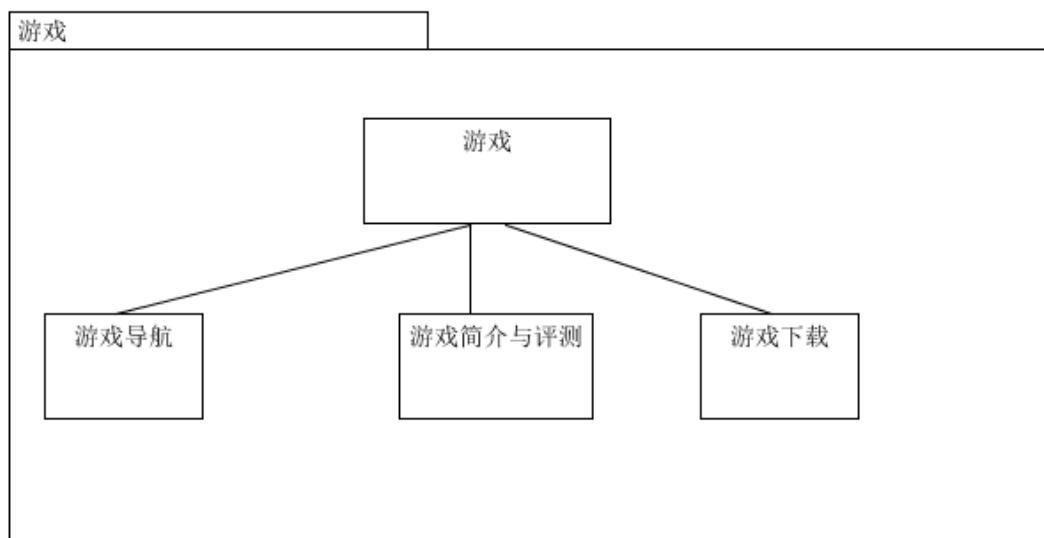


爬虫子系统定时爬取数据, 写入数据库。Web 服务器支持 PC 端和移动端, 通过 http 协议进行交互。

### 系统总体逻辑结构图

该网站下分为资讯、视频、游戏三大模块。







模块列表

模块名称	功能	备注
资讯模块	分类展示 VR 相关的资讯，推荐 VR 相关的资讯，资讯阅读与分享。	
视频模块	分类展示 VR 视频，视频介绍与播放。	
游戏模块	分类展示 VR 游戏，提供游戏简介与测评，提供游戏下载。	
爬虫模块	定时抓取最新资讯内容，对抓取内容进行处理，定时存入数据库。	

## 数据结构

数据库表列表

表名	作用
article	存储资讯标题，导览，内容，阅读量等相关信息
game	存储游戏名，简介，测评信息，以及下载链接等相关信息。
Kuaixun	存储 7x24 实时咨询简介。
video	存储视频名，类别，作者，播放地址等相关信息。
label	存储标签，用于标签管理。
Label_map	进行标签与对应内容的映射，用于标签存取。

## 接口规范

### <模块 1 API>

- 描述

<模块功能和设计描述(简要描述)>

- 集成和部署

<描述模块如何同系统集成，实现形式是 DLL, EXE 或是嵌入式代码>

#### Interface1

<interface1 description>

- Function1

**string function1(int param1, string param2, ...)**

<function1 description>

param1 - <param1 description>

param2 - <param2 description>

- Function2

**string function2(int param1, string param2, ...)**

<function2 description>

param1 - <param1 description>

param2 - <param2 description>

#### Interface2

**<模块 2 API>**

**<模块 3 API>**

**<数据库 API>**

定义数据库存储过程接口

## 模块设计

### Module1 设计

<更具具体情况细化模块设计>

#### 模块设计描述

#### 模块界面描述

## Module2 设计

## 附录

### 第三方组件

#### Delphi 组件

组件名称	版本	用途

#### .NET 组件

组件名称	版本	用途