

hotSpot 体系结构

- hotSpot 体系结构
- 1. 引言
 - 1.1. 编制目的
 - 1.2. 词汇表
 - 1.3. 参考资料
- 2. 产品概述
- 3. 逻辑视图
- 4. 组合视角
 - 4.1.1 开发包
 - 4.1.2 开发包图
 - 4.2. 运行时进程
 - 4.3. 物理部署
- 5. 架构设计
 - 5.1. 模块职责
 - 5.2. 用户界面层分解
 - 5.2.1. 职责
 - 5.2.2. 接口规范
 - 5.3. 业务逻辑层分解
 - 5.3.1. 职责
 - 5.3.2. 接口规范
 - 5.4. 数据层分解
 - 5.4.1. 职责
 - 5.4.2. 接口规范
 - 5.5. 信息视角
 - 5.5.1. User类
 - 5.5.2. news类
 - 5.5.3. weather类
 - 5.5.4. video类

1. 引言

1.1. 编制目的

- 1. 本文档详细完成了对hotSpot的概要设计，达到指导详细设计和开发的目的，同时实现和测试人员及用户的沟通
- 2. 本报告面向开发人员、测试人员及最终用户而编写，是了解系统的导航。

1.2. 词汇表

词汇名称	词汇含义	备注
hotSpot	热点	软件名称

1.3. 参考资料

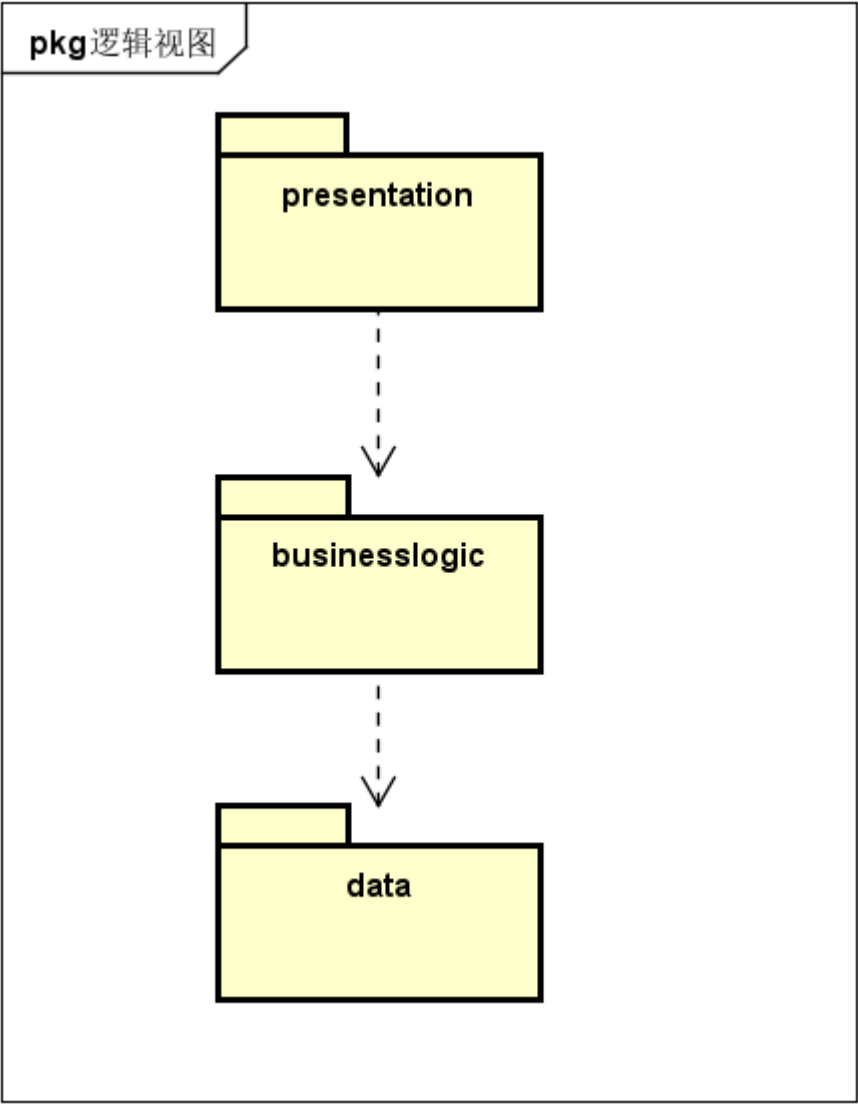
- 1. IEEE标准
- 2. 需求分析文档

2. 产品概述

- 1. 参考需求分析文档中对产品的概括描述。

3. 逻辑视图

- 在开发过程中，选择了分层体系结构风格，将系统分为3层(展示层、业务逻辑层、数据层)能够很好地示意整个高层抽象。展示层包含前端页面的实现，业务逻辑层包含业务逻辑处理的实现，数据层负责数据的持久化和访问。分层体系结构的了逻辑视图和逻辑设计方案如下图所示。



- 软件体系结构逻辑设计方案

详情见[移动端和服务端的开发包图](#)

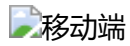
4. 组合视角

4.1.1 开发包

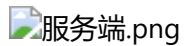
开发包	依赖的其他开发包
mainUi	userUi, weatherUi, newsUi, videoUi, util
newsUi	util, newsAPI,vo
userUi	util, userAPI,vo
weatherUi	util, weatherAPI,vo
videoUi	util, videoAPI,vo
util	
userAPI	HTTP
newsAPI	HTTP
weatherAPI	HTTP
videoAPI	HTTP
userController	HTTP,userService
newsController	HTTP,newsService
videoController	HTTP,videoService
weatherController	HTTP,weatherService
userService	po
newsService	po
videoService	po
weatherService	po
userData	po
newsData	po
videoData	po
weatherData	po
videoData	po
databaseutility	
vo	
po	

4.1.2 开发包图

- 移动端开发包图

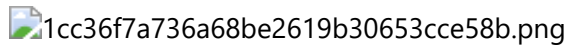


- 服务器端开发包图



4.2. 运行时进程

- APP有多个移动端进程和一个服务器端进程，其进程图如下图。结合部署图，移动端进程在移动设备上运行，服务器端进程在服务器端机器上运行。



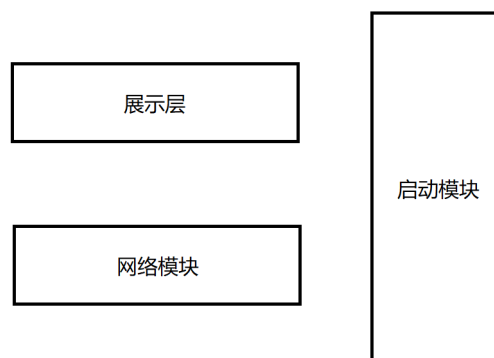
4.3. 物理部署

- 移动端组件放在移动设备上，服务器端组件都是放在服务器端机器上。

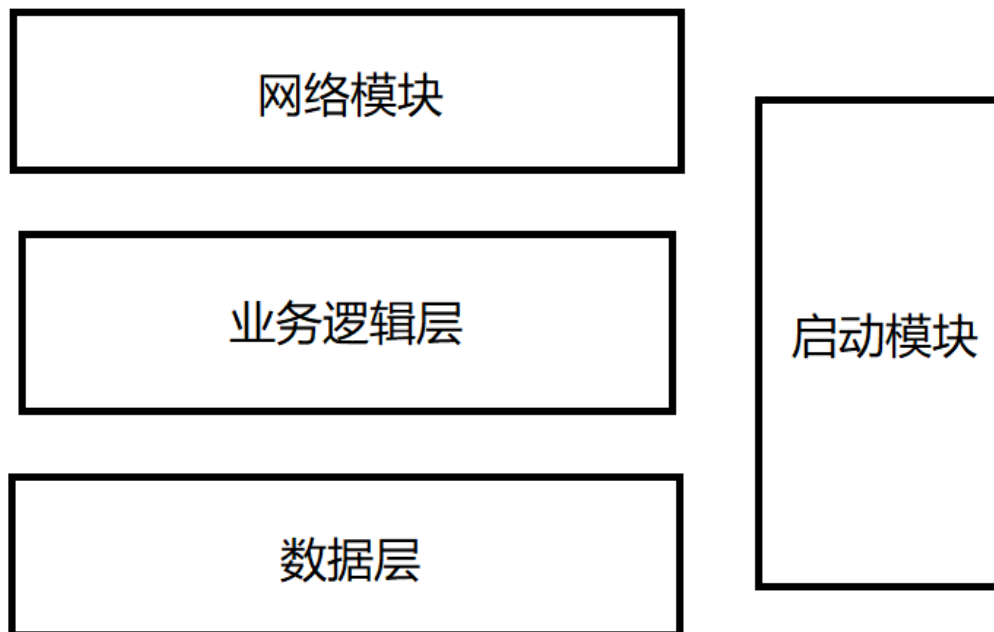
5. 架构设计

5.1. 模块职责

- 移动端模块和服务器端模块视图分别如下两张图所示（依次对应上下两张）。
 - 移动端模块视图



- 服务器端模块视图



• 移动端各层职责

层	职责
启动模块	负责初始化网络通信机制，启动用户界面
用户界面层	hotSpot的界面
移动端网络模块	利用 Rest API 机制发起HTTP请求

• 服务器端各层职责

层	职责
启动模块	负责初始化网络通信机制，启动用户界面
业务逻辑层	对于用户界面输入的响应并进行业务处理逻辑
数据层	负责数据的持久化和数据访问接口
服务端网络模块	接受移动端的HTTP请求

• 层之间调用接口

接口	服务调用方	服务提供方
newsController	移动端展示层	移动端业务逻辑层
UserController		
videoController		
weatherController		

接口	服务调用方	服务提供方
newsService	移动端业务逻辑层	服务端数据层
userService		
videoService		
weatherService		

5.2. 用户界面层分解

5.2.1. 职责

- 用户界面跳转图

模块	职责
news	负责展示新闻界面
user	负责展示登陆、注册、个人信息界面
weather	负责展示天气界面
video	负责视频界面

5.2.2. 接口规范

- user 模块

接口名称	内容
LoginFormAPI	用户登入
RegisterFormAPI	用户注册
logoutFormAPI	用户登出
setFormAPI	编辑个人信息
submitUserInfoFormAPI	提交个人信息

- video 模块

接口名称	内容
getVideoListAPI	随机获得一组视频
getVideoAPI	获取新闻

- news 模块

接口名称	内容
addNewsAPI	添加新闻
delNewsAPI	删除新闻

接口名称	内容
changeNewsAPI	修改新闻
getNewsAPI	获取新闻
getNewsListAPI	随即得到一组新闻

- weather 模块

接口名称	内容
getWeathersAPI	获取未来七天天气
getDetailWeatherAPI	获取当日详细天气信息
getAdviceAPI	根据当天天气得到建议

5.3. 业务逻辑层分解

5.3.1. 职责

模块	职责
weather	负责从获取实时天气数据并传入前端
user	负责登陆、注册、管理个人信息
video	负责视频管理
news	负责新闻管理

5.3.2. 接口规范

- news 模块

接口	类型	参数	内容
newsController.getNewsList	CommonResponse<List<news>>		随机获取一组新闻
newsController.getnews	CommonResponse<newsDetail>	ListnewsRequest request	获到新闻信息
newsController.addnews	CommonResponse<?>	AddnewsRequest request	添加新闻
newsController.changenews	CommonResponse<?>	Long newsId, AddnewsRequest request	更改新闻

接口	类型	参数	内容
newsController.deletenews	CommonResponse<?>	Long newsId	删除新闻
newsService.getnews	newsDetailVO	Long newsId	得到目标新闻
newsService.listNews	List< news>		随机获取一组新闻
newsService.addnews	void	String name, Long routeId, newsType type, String date, List< Date> arrivalTimes, List departureTimes	添加新闻
newsService.changenews	void	Long id, String name, Long routeId, newsType type, String date, List arrivalTimes, List departureTimes	更改新闻
newsService.deletenews	void	Long id	删除新闻

• user 模块

接口	类型	参数	内容
UserController.login	CommonResponse<?>	LoginRequest request	用户登入
UserController.register	CommonResponse<?>	RegisterRequest request	用户注册
UserController.logout	CommonResponse<?>		用户登出
UserController.userInfo	CommonResponse< User>		获取用户信息，转换为视图，并返回成功响应
UserController.editInfo	CommonResponse<?>	EditUserInfoRequest request	修改个人信息
UserService.login	void	String username, String password	用户登入

接口	类型	参数	内容
UserService.editInfo	void	String username, String name, String idn, String phone, String type, String identity, double mileagePoints, double aliBalance, double wechatBalance	用户修改信息

UserService.findByUserName	User	username	通过用户名查找用户
----------------------------	------	----------	-----------

• weather 模块

接口	类型	参数	内容
weatherController.listweathers	CommonResponse<List<weather>>	ListWeatherRequest request	获取未来七天天气信息
weatherController.getweather	CommonResponse	DetailWeatherRequest request	获取当日实时天气信息
weatherController.getAdvice	CommonResponse	Weather	根据当天天气提供建议
weatherService.getWeathers	List		获取未来七天天气信息
weatherService.getDetailWeather	weather		获取当日实时天气信息
weatherService.getAdvice	String	weather	根据当天天气获得建议

• video 模块

接口	类型	参数	内容
VideoController.listVideos	CommonResponse<List<Video>>	ListVideoRequest request	列出所有视频
VideoController.getVideo	CommonResponse<Video>	ListVideoRequest request	获取某个视频
VideoService.getVideo	Video	Long VideoId	得到目标视频
VideoService.listVideos	List< Video>	Long startweatherId, Long endweatherId, String date	列出所有视频信息

5.4. 数据层分解

5.4.1. 职责

模块	职责
weather	负责管理天气信息
user	负责管理用户信息
video	负责管理视频
news	负责管理新闻

5.4.2. 接口规范

接口名称	内容
CrudRepository.save	保存实体对象
CrudRepository.saveAll	保存多个实体对象
CrudRepository.findById	根据id查找实体对象
CrudRepository.existsById	检查具有给定id的实体对象是否存在
CrudRepository.findAll	返回所有类型为T的实体对象
CrudRepository.findAllById	根据给定的id集合返回类型为T的实体对象集合
CrudRepository.count	返回可用的实体对象数量
CrudRepository.deleteById	根据给定的id删除实体对象
CrudRepository.delete	删除给定的实体对象
CrudRepository.deleteAllById	根据给定的id集合删除类型为T的实体对象
CrudRepository.deleteAll	除给定的实体对象集合

5.5. 信息视角

- 系统的PO类包含用户的对应的相关的实体类

5.5.1. User类

- 包含用户的ID、用户名、密码、手机号

5.5.2. news类

- 包含新闻编号、标题、标识（置顶&热搜）、概述、点击量、链接

5.5.3. weather类

- 包含当日最低&最高温、天气（晴雨雪阴等）、24小时的温度

5.5.4. video类

- 包含视频编号、播放量、封面、视频url