目录

[1. 主要功能点和操作场景分析 2](#_Toc14852)

[1.1主要功能点 2](#_Toc21812)

[1.2操作场景分析 3](#_Toc14167)

[1.2.1用户管理 3](#_Toc6472)

[1.2.2学生功能 3](#_Toc12313)

[1.2.3管理员功能 5](#_Toc29867)

[2. 两种架构选择及对应的架构图设计 7](#_Toc3856)

[2.1 MVC架构 7](#_Toc15874)

[2.1.1模块视图 7](#_Toc21372)

[2.1.3组件-连接器视图 7](#_Toc5641)

[2.2 SOA架构 8](#_Toc18339)

[2.2.1模块视图 9](#_Toc135)

[2.2.2组件-连接器视图 9](#_Toc118)

[3. 非功能需求及ASR描述 10](#_Toc1477)

[3.1安全性 10](#_Toc6072)

[3.2可获得性 11](#_Toc10412)

[3.3互操作性 11](#_Toc23572)

[3.4可修改性 11](#_Toc7939)

[3.5性能 12](#_Toc20210)

[3.6可维护性 13](#_Toc25694)

[4. 系统类图设计 13](#_Toc17951)

[5. 组件/连接器到实现类的映射 13](#_Toc17851)

[6. 两种架构的比较及最终选择 13](#_Toc30589)

[7. 基于MVC架构的ADD过程 15](#_Toc13030)

[7.1第一次迭代结果 15](#_Toc10062)

[7.2第二次迭代过程 15](#_Toc19480)

[7.2.1识别所选模块的ASR 15](#_Toc13798)

[7.2.2每个ASR可选的设计决策 15](#_Toc5483)

[7.2.3设计决策的选择及分析 16](#_Toc28603)

[7.2.4第二次迭代结果 17](#_Toc18206)

[7.3第三次迭代过程 18](#_Toc23344)

[7.3.1识别所选模块的ASR 18](#_Toc10425)

[7.3.2每个ASR可选的设计决策 18](#_Toc22932)

[7.3.3设计决策的选择及分析 19](#_Toc5776)

[7.3.4第三次迭代结果 20](#_Toc23844)

[8. 基于SOA架构的ADD过程 21](#_Toc3137)

[8.1第一次迭代结果 21](#_Toc17427)

[8.2第二次迭代过程 21](#_Toc9233)

[8.2.1识别所选模块的ASR 22](#_Toc10472)

[8.2.2每个ASR可选的设计决策 22](#_Toc14944)

[8.2.3设计决策的选择及分析 23](#_Toc15024)

[8.2.4第二次迭代结果 24](#_Toc7561)

[8.3第三次迭代过程 24](#_Toc979)

[8.3.1识别所选模块的ASR 24](#_Toc13796)

[8.3.2每个ASR可选的设计决策 25](#_Toc31893)

[8.3.3设计决策的选择及分析 26](#_Toc11978)

[8.3.4第三次迭代结果 26](#_Toc28288)

[9. ATAM分析过程 27](#_Toc12492)

[9.1质量属性效用树 27](#_Toc6226)

[9.2ATAM分析 29](#_Toc32372)

[9.3敏感点和权衡点 33](#_Toc15905)

[9.4风险和非风险 34](#_Toc4175)

[10. 挑战和经验 35](#_Toc28626)

[11. 组员和分工 35](#_Toc29205)

# 主要功能点和操作场景分析

## 1.1主要功能点

* 用户管理
  + 注册
  + 登录
  + 用户权限管理
  + 个人信息修改
* 学生功能
  + 讲座推荐
  + 查看讲座信息
  + 讲座报名
  + 讲座退选
  + 讲座评价
  + 讨论讲座
  + 为收费讲座支付入场费
  + 收看允许录像的讲座的直播和回放
* 管理员功能
  + 导入、修改讲座信息
  + 强制报名退选讲座
  + 讲座反馈
  + 讨论区管理

## 1.2操作场景分析

### 1.2.1用户管理

表1 登录/注册

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 希望访问该系统的用户 |
| 刺激 | 用户希望访问该系统 |
| 制品 | 登录/注册子系统 |
| 环境 | 正常操作 |
| 响应 | 验证用户登录信息/注册用户 |
| 响应度量 | 响应时间不超过0.5秒。  用户的个人信息及密码应被有效保护，被攻击盗取的概率不大于99.99% |

表2 用户权限管理

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 管理员 |
| 刺激 | 管理员希望查看或修改用户权限 |
| 制品 | 用户权限管理子系统 |
| 环境 | 正常操作 |
| 响应 | 显示用户权限/保存用户权限修改设置 |
| 响应度量 | 响应时间不超过0.5秒。权限设置不应有误。 |

表3 个人信息修改

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 希望修改个人信息的用户 |
| 刺激 | 用户希望修改个人信息 |
| 制品 | 个人信息修改子系统 |
| 环境 | 正常操作 |
| 响应 | 修改个人信息并保存 |
| 响应度量 | 响应时间不超过0.5秒。个人信息存储不应出错。 |

### 1.2.2学生功能

表4 查看讲座信息

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 学生用户 |
| 刺激 | 学生用户希望查看当前/个人的讲座信息 |
| 制品 | 查看讲座信息子系统 |
| 环境 | 正常操作 |
| 响应 | 显示当前/个人的讲座信息 |
| 响应度量 | 响应时间不超过0.5秒。讲座信息显示不应有误。 |

表5 报名参加讲座

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 学生用户 |
| 刺激 | 学生用户希望报名参加某讲座 |
| 制品 | 报名参加讲座子系统 |
| 环境 | 正常操作 |
| 响应 | 若名额未满，则学生用户报名成功；否则显示报名失败。 |
| 响应度量 | 系统应该允许500个用户同时进行正常的访问、报名操作 |

表6 讲座签到/支付

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 学生用户 |
| 刺激 | 学生用户希望报名参加某讲座 |
| 制品 | 讲座签到/支付子系统 |
| 环境 | 正常操作 |
| 响应 | 报名参加某讲座的用户签到/支付成功 |
| 响应度量 | 响应时间不超过0.5秒。成功记录用户签到/支付信息的概率不低于99.9% |

表7 讲座退选

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 学生用户 |
| 刺激 | 学生用户希望退选已报名的某讲座 |
| 制品 | 讲座退选子系统 |
| 环境 | 正常操作 |
| 响应 | 若用户不是管理员强制设定必须参加某讲座，则用户退选讲座成功，并开放可报名参加讲座的名额；否则提示退选失败。 |
| 响应度量 | 响应时间不超过0.5秒。系统成功处理不同退选情况的概率不小于99.99% |

表8 讲座讨论/评价

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 学生用户 |
| 刺激 | 学生用户希望评价观看的某讲座并进行讨论 |
| 制品 | 讲座讨论/评价系统 |
| 环境 | 正常操作 |
| 响应 | 记录用户的讨论/评价信息 |
| 响应度量 | 响应时间不超过0.5秒。 |

表9 讲座直播

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 学生用户 |
| 刺激 | 学生用户希望观看某正在进行的讲座直播 |
| 制品 | 讲座直播子系统 |
| 环境 | 正常操作 |
| 响应 | 系统播放允许摄像的公开讲座的直播视频 |
| 响应度量 | 响应时间不超过0.5秒。  系统应该允许500个用户同时进行讲座直播视频观看。  视频系统直播延时不超过0.5s |

表10 讲座回放

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 学生用户 |
| 刺激 | 学生用户希望观看某已结束讲座的视频回放 |
| 制品 | 讲座回放子系统 |
| 环境 | 正常操作 |
| 响应 | 系统播放允许摄像的公开讲座的回放视频 |
| 响应度量 | 响应时间不超过0.5秒。  系统至少能够存储50T的视频数据 |

### 1.2.3管理员功能

表11 导入/修改讲座信息

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 管理员用户 |
| 刺激 | 管理员希望导入/修改讲座信息 |
| 制品 | 讲座信息管理子系统 |
| 环境 | 正常操作 |
| 响应 | 系统保存导入/修改的讲座信息 |
| 响应度量 | 响应时间不超过0.5秒。  系统成功保存讲座信息的概率不小于99.99% |

表12 强制报名/退选讲座

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 管理员用户 |
| 刺激 | 管理员希望强制某些学生用户报名/退选讲座 |
| 制品 | 讲座报名管理子系统 |
| 环境 | 正常操作 |
| 响应 | 系统对应更改讲座及报名信息记录 |
| 响应度量 | 响应时间不超过0.5秒。  系统成功保存信息的概率不小于99.99% |

表13 讲座反馈

|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 管理员用户 |
| 刺激 | 管理员希望查看讲座反馈信息 |
| 制品 | 讲座反馈子系统 |
| 环境 | 正常操作 |
| 响应 | 系统显示讲座反馈信息及学生参与情况 |
| 响应度量 | 响应时间不超过0.5秒。 |

表14 讨论区管理

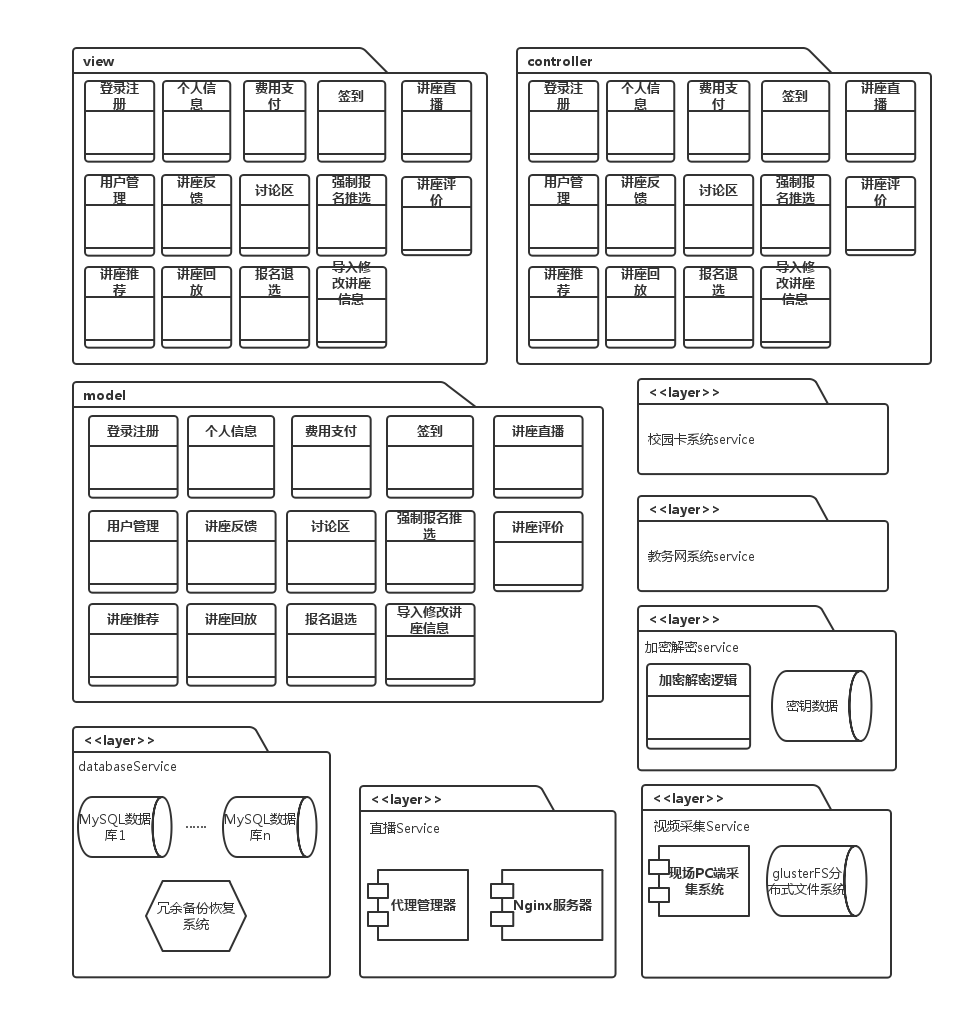
|  |  |
| --- | --- |
| **场景要素** | **可能的值** |
| 刺激源 | 管理员用户 |
| 刺激 | 管理员希望管理讨论区，进行删除/置顶讨论内容，并设置某些用户禁言 |
| 制品 | 讨论区管理子系统 |
| 环境 | 正常操作 |
| 响应 | 系统保存响应操作 |
| 响应度量 | 响应时间不超过0.5秒。 |

# 两种架构选择及对应的架构图设计

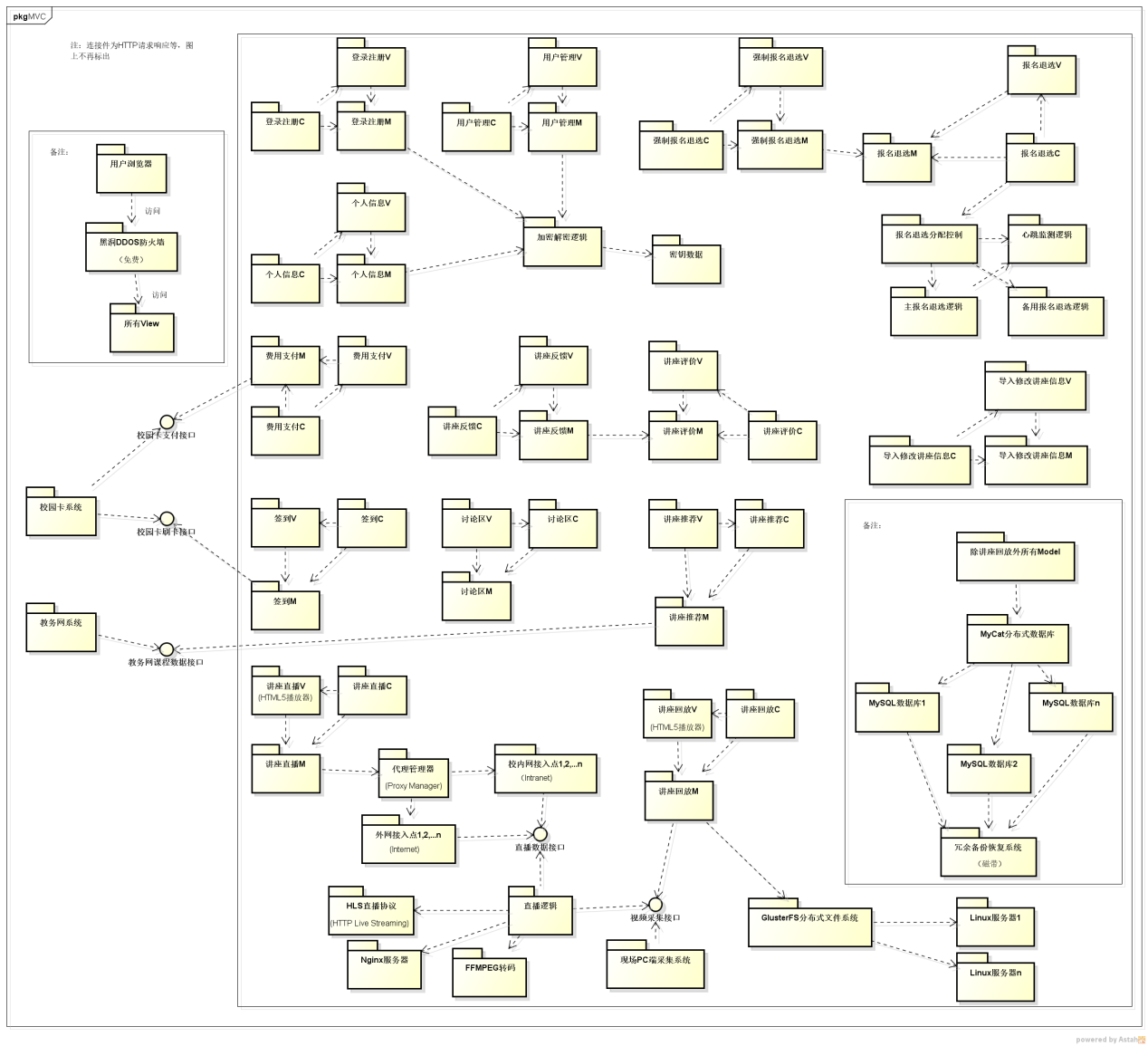
注：清晰的架构图设计详见附属的架构图设计文件夹。

## 2.1 MVC架构

### 2.1.1模块视图

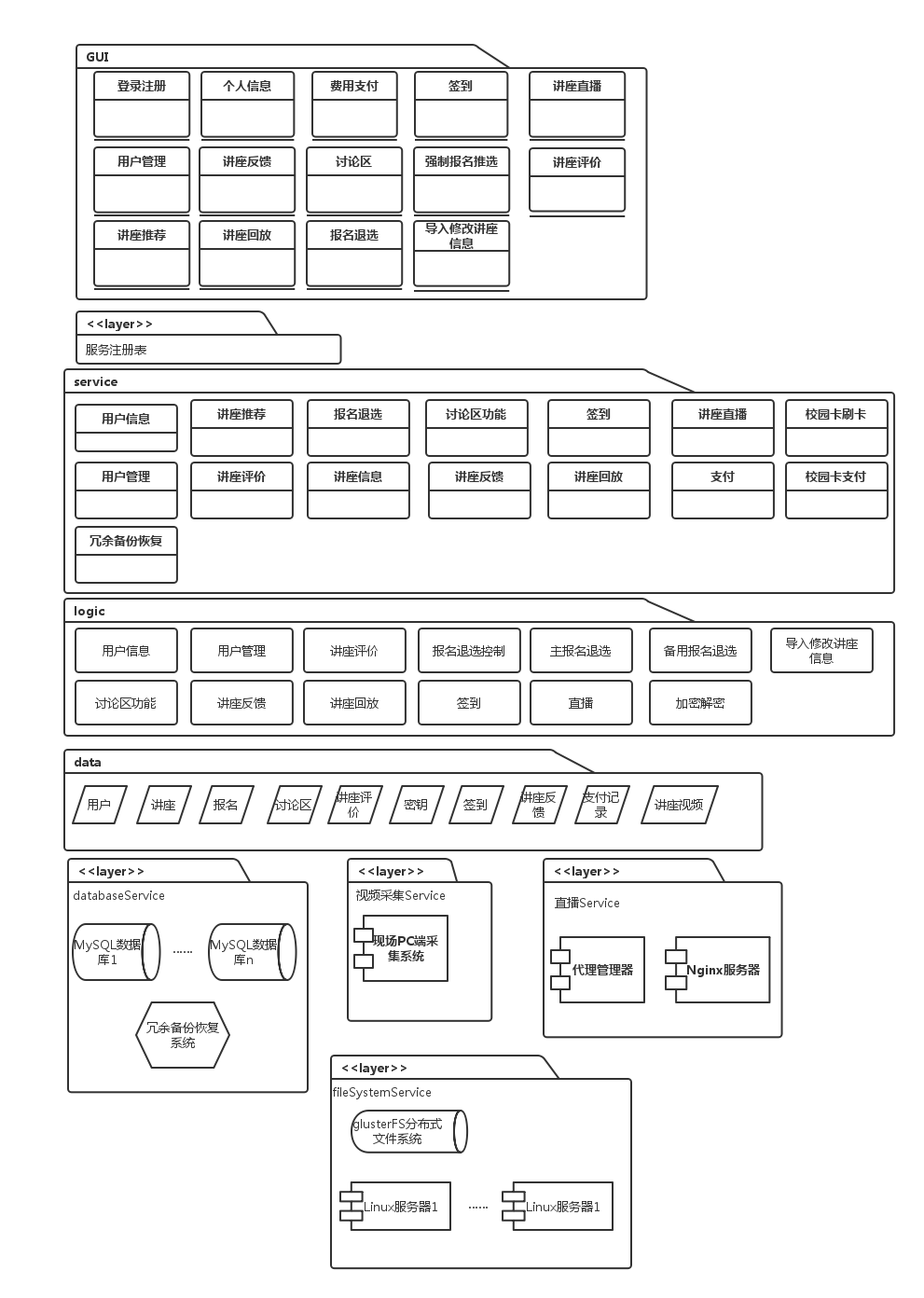


### 2.1.3组件-连接器视图

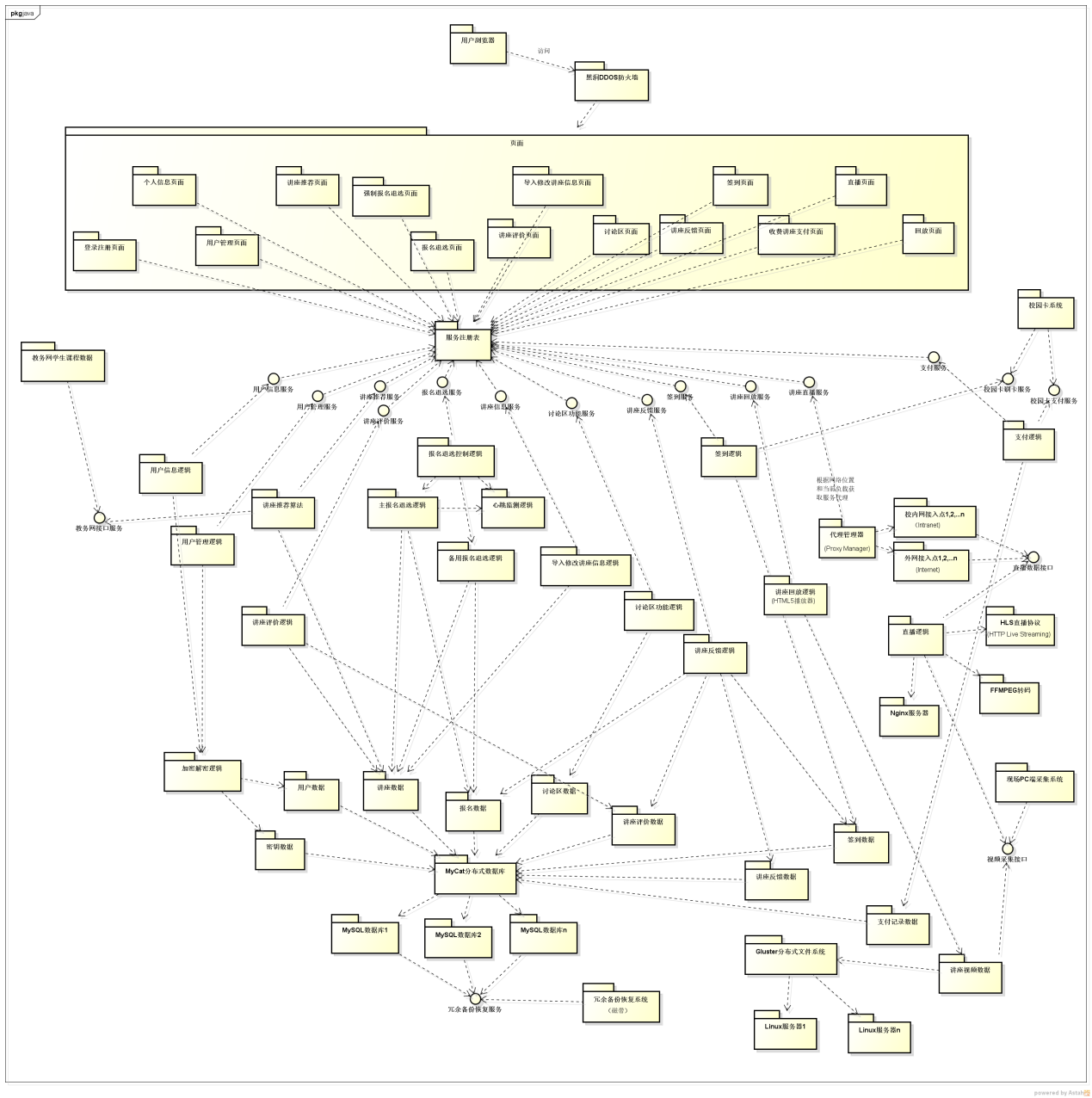


## 2.2 SOA架构

### 2.2.1模块视图



### 2.2.2组件-连接器视图



# 非功能需求及ASR描述

## 3.1安全性

Scenario 1

|  |  |
| --- | --- |
| **场景的部分** | **可能的值** |
| **源** | 来自内部/外部的经过了授权/未经过授权的个人或系统 |
| **刺激** | 试图修改/删除数据，访问系统服务，降低系统服务的可用性 |
| **制品** | 系统服务、系统中的数据 |
| **环境** | 在线/离线，联网/断网，连接有防火墙或直接连接到了网络上 |
| **响应** | 对用户进行验证；加密用户的账户信息；阻止未授权用户访问；自动侦测攻击，受到攻击后通知管理员并锁死数据访问 |
| **响应度量** | 未认证用户无法访问数据和发布控制指令  受到攻击后1min内将信息发送给管理员并锁死数据访问  数据被恶意修改/删除后可以在10min内进行恢复 |

## 3.2可获得性

Scenario 2

|  |  |
| --- | --- |
| **场景的部分** | **可能的值** |
| **源** | 来自授权的个人用户 |
| **刺激** | 希望进入系统报名参与讲座时系统无法提供正确的反馈 |
| **制品** | 系统报名参加讲座子系统 |
| **环境** | 联网状态 |
| **响应** | 记录错误日志并从错误中恢复 |
| **响应度量** | 出现此种情况的概率小于1%  记录错误日志并从错误中恢复的时间不多于5s |

## 3.3互操作性

Scenario 3

|  |  |
| --- | --- |
| **场景的部分** | **可能的值** |
| **源** | 系统发起/收到与校园卡系统/教务网系统/现场PC端采集系统的请求/响应 |
| **刺激** | 希望与上述的系统进行请求/响应 数据交换 |
| **制品** | 希望进行互操作的上述系统 |
| **环境** | 希望进行互操作的上述系统正在运行 |
| **响应** | 请求被正确获取并成功交换数据 |
| **响应度量** | 交换信息数据成功的概率不低于99% |

## 3.4可修改性

Scenario 4

|  |  |
| --- | --- |
| **场景的部分** | **可能的值** |
| **源** | 开发者 |
| **刺激** | 开发者希望修改系统用户界面、数据标准、控制逻辑等 |
| **制品** | 代码 |
| **环境** | 设计、开发系统时 |
| **响应** | 需要修改的模块被正确的修改，并不影响其他功能的实现 |
| **响应度量** | 每个模块的修改可以在2人月内完成  修改预算不超过总预算的10%  不影响无关的系统功能 |

## 3.5性能

性能场景1：负载

Scenario 5

|  |  |
| --- | --- |
| **场景的部分** | **可能的值** |
| **源** | 访问系统的学生个人用户 |
| **刺激** | 学生用户希望在系统上进行讲座报名操作/查看讲座直播 |
| **制品** | 系统报名参加讲座子系统/讲座直播子系统 |
| **环境** | 联网状态 |
| **响应** | 系统能够正常工作，为每一个用户提供响应 |
| **响应度量** | 系统应该允许500个用户同时进行正常的访问、报名操作/观看讲座直播  用户访问所需页面的时间不超过0.1s |

性能场景2：容量

Scenario 6

|  |  |
| --- | --- |
| **场景的部分** | **可能的值** |
| **源** | 系统 |
| **刺激** | 系统希望进行讲座视频的存储 |
| **制品** | 存储视频数据的系统数据库 |
| **环境** | 运行时 |
| **响应** | 系统正确、完整、及时的存储大量讲座视频数据 |
| **响应度量** | 系统至少能够存储50T的视频数据 |

性能场景3：实时性

Scenario 7

|  |  |
| --- | --- |
| **场景的部分** | **可能的值** |
| **源** | 访问系统的学生个人用户 |
| **刺激** | 学生用户希望在系统上观看讲座直播 |
| **制品** | 系统讲座直播子系统 |
| **环境** | 联网状态 |
| **响应** | 系统直播视频清晰流畅，延时短 |
| **响应度量** | 系统直播延时不超过0.5s |

## 3.6可维护性

Scenario 8

|  |  |
| --- | --- |
| **场景的部分** | **可能的值** |
| **源** | 系统开发者 |
| **刺激** | 系统开发者希望修改加密密钥 |
| **制品** | 系统用户个人信息模块 |
| **环境** | 系统维护时 |
| **响应** | 及时、方便的进行修改，并对其他模块不造成影响 |
| **响应度量** | 修改时间不超过1人/天 |

# 系统类图设计

# 组件/连接器到实现类的映射

# 两种架构的比较及最终选择

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **SOA** | | **MVC** | |
|  | **pros** | **cons** | **pros** | **cons** |
| 安全性 |  | SOA架构的松耦合性和开放性带来安全性的问题：当信任体系存在漏洞或不统一时，可能存在非授权组织非法获取信息的情况。本系统涉及到安全性攸关的校园卡服务，使用SOA会增加安全方案的复杂度。 | 针对用户管理，个人信息和登录注册Model访问数据时加入加密解密逻辑，保证数据安全性。 |  |
| 可获得性 | 在可获得性方面，两种架构均对于报名退选功能设置备份，设置心跳机制监测主模块，心跳中断则切换至备用模块。另一方面，所有数据有冗余备份，以防意外，也便于崩溃发生后的恢复。在可获得性的方面，两种架构没有特别大的差别 | | | |
| 互操作性 | 本系统涉及到即将建设的系统和已存在的校园卡系统和教务系统，可能存在平台和异构问题。选择接口而不是语言具体的类以及基于消息交互，极大的提高了系统的互操作性。 |  |  | 在MVC架构中，费用支付Model，签到Model，讲座推荐Model均需要直接使用到外部系统提供的接口。可能存在接口难以使用的情况。 |
| 可修改性 | SOA不同服务之间保持了一种无依赖的低耦合关系；服务本身是通过统一的接口定义语言来描述具体的服务内容，并且很好地封装了底层的具体实现。所以，当发生修改时，服务之间不会影响，并且低层实现不会影响调用。 |  | MVC架构耦合性低，MVC的三个部件是相互独立，改变其中一个不会影响其它两个，这样具有良好的可修改性。 | 本系统具有与教务系统和校园卡交互的接口，当外部接口定义发生变化时，需要修改本系统的Model的实现逻辑。 |
| 负载 | 两种架构均采用相同的负载策略：对于报名退选功能设置备份，双线网络接入等。两种架构没有特别大的差别。 | | | |
| 容量 | 两种架构均采用分布式数据库来保证容量，没有特别大的差别。 | | | |
| 实时性 |  | SOA架构的性能稍低，主要是因为SOA的分布性质和web服务协议的开销。 | MVC架构在实时性上没有特别的表现。 |  |
| 可维护性 | 服务带来低耦合性和很好的模块性。 |  | 分离视图层和业务逻辑层也使得WEB应用更易于维护和修改。 |  |

【最终选择及原理】

# 基于MVC架构的ADD过程

## 7.1第一次迭代结果

采用MVC的架构，参考所需的功能需求，形成第一次迭代后的整体设计图。由于该系统的功能性需求整体较为清晰，因此在第一次迭代结束后，已经将大部分功能性需求纳入架构设计考虑。

## 7.2第二次迭代过程

选取直播模块作为系统元素进行分解。在质量属性方面，该模块与互操作性、可修改性、负载性能、实时性有密切关系。

### 7.2.1识别所选模块的ASR

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Architectural Drivers** | **Importance** | **Difficulty** |
| **1** | Scenario 3：互操作性：系统发起现场PC端采集系统的请求被正确获取并成功交换数据 | high | medium |
| **2** | Scenario 4：可修改性 | medium | low |
| **3** | Scenario 5：负载性能：系统能够允许500个用户同时观看讲座直播 | high | high |
| **4** | Scenario 7：实时性 | high | high |

### 7.2.2每个ASR可选的设计决策

* 互操作性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 在提供的服务列表中定位所需要的服务 | 主要针对所需的服务较多且有层次性的情况 |
| 使用控制机制来管理、协调服务的调用 | 针对所要服务的访问需求量较多的情况 |
| 抽取并剪裁接口 | 针对请求响应交换的数据不一致情况 |

* 可修改性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 划分模块 | 可以有效缩小要修改的范围 |
| 抽象通用服务 | 将多个模块都需要的服务提取出来，形成可供多个模块共同使用的通用服务 |
| 添加接口 | 将功能和实现分离，隔离需要更改的模块 |
| 运行时注册 | 支持即插即用操作，但需要管理注册的额外开销 |

* 负载性能

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 维持计算的多个副本 | 需要考虑如何使副本保持一致和同步 |
| 增加可用资源 | 使用速度更快的处理器、额外的处理资源、额外的内存、速度更快的网络。但是会增加成本 |
| 区分校内校外网络接入 | 可以充分利用高速廉价的内网资源，但是增加了系统的复杂性 |

* 实时性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 使用RTMP(Real Time Messaging Protocol)协议 | 适合长时间播放，延时较低，但是又累积延时；是Adobe开发的协议，无法再iPhone中兼容 |
| 使用HLS（ HTTP Live Streaming）协议 | 兼容性比RTMP好，但是延时比其高 |

### 7.2.3设计决策的选择及分析

* 互操作性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 在提供的服务列表中定位所需要的服务 | 不采用。仅是关于直播的服务。 |
| 使用控制机制来管理、协调服务的调用 | 不采用。仅是关于直播的服务。 |
| 抽取并剪裁接口 | 采用。用于统一不同的数据格式。 |

* 可修改性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 划分模块 | 采用。是一种常用且有效的策略。 |
| 抽象通用服务 | 不采用。并没有通用服务需要被抽象。 |
| 添加接口 | 采用。增加可修改性常用的策略。 |
| 运行时注册 | 不采用。会带来额外的管理开销。 |

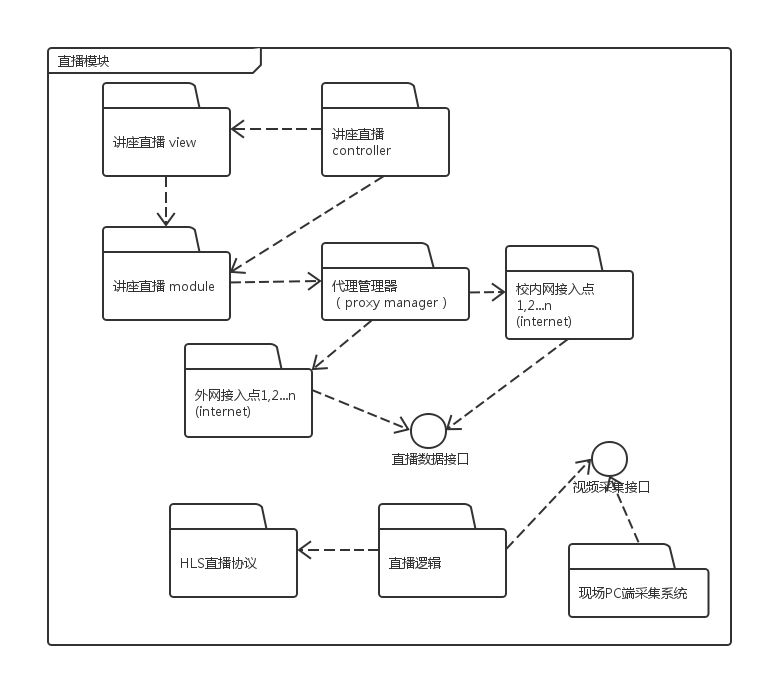
* 负载性能

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 维持计算的多个副本 | 不采用。考虑同步需要的代价太大。 |
| 增加可用资源 | 不采用。会增加较多成本。 |
| 区分校内校外网络接入 | 采用。适用于此类开发成本不应太大的系统。 |

* 实时性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 使用RTMP(Real Time Messaging Protocol)协议 | 不采用。并没有很好的兼容性。 |
| 使用HLS（ HTTP Live Streaming）协议 | 采用。该技术成熟且容易使用。此外，开源有益于控制开发系统的成本。 |

### 7.2.4第二次迭代结果



## 7.3第三次迭代过程

选取讲座报名模块作为系统元素进行分解。在质量属性方面，该模块与可获得性、可修改性、负载性能有密切关系。

### 7.3.1识别所选模块的ASR

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Architectural Drivers** | **Importance** | **Difficulty** |
| **1** | Scenario 2：可获得性 | high | medium |
| **2** | Scenario 4：可修改性 | medium | low |
| **3** | Scenario 5：负载性能：系统能够允许500个用户同时进行正常的访问、报名操作 | high | high |

### 7.3.2每个ASR可选的设计决策

* 可获得性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 命令/响应 | 一个组件发出一个命令，并希望在预定义的时间内收到一个来自审查组件的响应。与对所有进程发出命令的远程错误探测器相比，这种策略所使用的通信带宽更少 |
| 心跳 | 一个组件定期发出一个心跳信息，另一个组件收听该信息。可以在起到心跳的同时传递数据。 |
| 主动冗余（热启动） | 所有的冗余组件都以并行的方式对事件作出响应。错误发生时，使用该战术的系统的停机时间通常是几毫秒，因为备份是最新的，所以恢复所需的时间就是切换时间。 |
| 被动冗余 | 一个组件（主要的）对事件作出响应，并通知其他组件（备用的）必须进行的状态更新。该战术依赖于能够可靠地接管工作的备用组件。停机时间通常为几秒钟。 |
| 备件 | 备用件是计算平台配置用于更换各种不同的故障组件。当出现故障时，必须将其重新启动为适当的软件配置，并对其状态进行初始化。该战术的停机时间通常为几分钟。 |

* 可修改性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 划分模块 | 可以有效缩小要修改的范围 |
| 抽象通用服务 | 将多个模块都需要的服务提取出来，形成可供多个模块共同使用的通用服务 |
| 添加接口 | 将功能和实现分离，隔离需要更改的模块 |
| 运行时注册 | 支持即插即用操作，但需要管理注册的额外开销 |

* 负载性能

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 维持计算的多个副本 | 需要考虑如何使副本保持一致和同步 |
| 增加可用资源 | 使用速度更快的处理器、额外的处理资源、额外的内存、速度更快的网络。但是会增加成本 |

### 7.3.3设计决策的选择及分析

* 可获得性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 命令/响应 | 不采用。 |
| 心跳 | 采用。常用的方式，且使系统的可获得性更高。 |
| 主动冗余（热启动） | 采用。该系统对于选课报名的可获得性要求很高，采用热备份可以获得更高的可获得性。 |
| 被动冗余 | 不采用。停机时间过长，对于讲座报名这种短时间内对可获得性要求极高的子系统而言并不够好。 |
| 备件 | 不采用。理由同上。 |

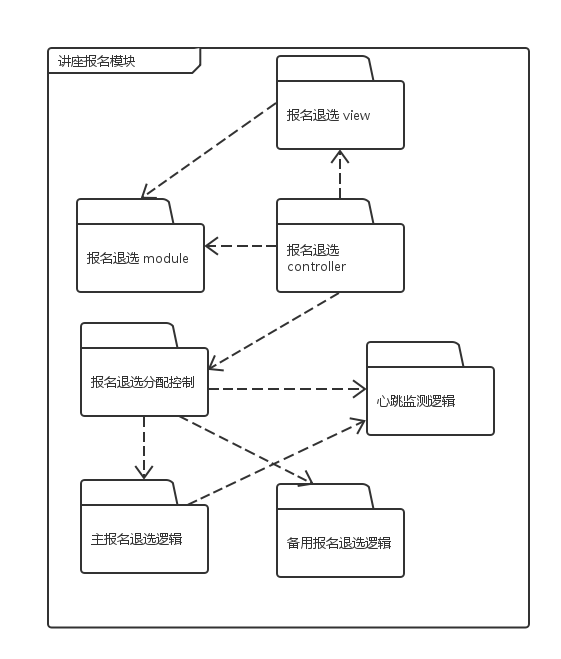
* 可修改性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 划分模块 | 采用。是一种常用且有效的策略。 |
| 抽象通用服务 | 不采用。并没有通用服务需要被抽象。 |
| 添加接口 | 不采用。不适用于该子系统。 |
| 运行时注册 | 不采用。会带来额外的管理开销。 |

* 负载性能

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 维持计算的多个副本 | 采用。保证在一个计算逻辑失效后可以在较短的时间内启用备用逻辑。 |
| 增加可用资源 | 不采用。会增加较多成本。 |

### 7.3.4第三次迭代结果



# 基于SOA架构的ADD过程

## 8.1第一次迭代结果

采用SOA的架构，参考所需的功能需求，形成第一次迭代后的整体设计图。由于该系统的功能性需求整体较为清晰，因此在第一次迭代结束后，已经将大部分功能性需求纳入架构设计考虑。

## 8.2第二次迭代过程

选取直播模块作为系统元素进行分解。在质量属性方面，该模块与互操作性、可修改性、负载性能、实时性有密切关系。

### 8.2.1识别所选模块的ASR

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Architectural Drivers** | **Importance** | **Difficulty** |
| **1** | Scenario 3：互操作性：系统发起现场PC端采集系统的请求被正确获取并成功交换数据 | high | medium |
| **2** | Scenario 4：可修改性 | medium | low |
| **3** | Scenario 5：负载性能：系统能够允许500个用户同时观看讲座直播 | high | high |
| **4** | Scenario 7：实时性 | high | high |

### 8.2.2每个ASR可选的设计决策

* 互操作性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 在提供的服务列表中定位所需要的服务 | 主要针对所需的服务较多且有层次性的情况 |
| 使用控制机制来管理、协调服务的调用 | 针对所要服务的访问需求量较多的情况 |
| 抽取并剪裁接口 | 针对请求响应交换的数据不一致情况 |

* 可修改性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 划分模块 | 可以有效缩小要修改的范围 |
| 抽象通用服务 | 将多个模块都需要的服务提取出来，形成可供多个模块共同使用的通用服务 |
| 添加接口 | 将功能和实现分离，隔离需要更改的模块 |
| 运行时注册 | 支持即插即用操作，但需要管理注册的额外开销 |

* 负载性能

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 维持计算的多个副本 | 需要考虑如何使副本保持一致和同步 |
| 增加可用资源 | 使用速度更快的处理器、额外的处理资源、额外的内存、速度更快的网络。但是会增加成本 |
| 区分校内校外网络接入 | 可以充分利用高速廉价的内网资源，但是增加了系统的复杂性 |

* 实时性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 使用RTMP(Real Time Messaging Protocol)协议 | 适合长时间播放，延时较低，但是又累积延时；是Adobe开发的协议，无法再iPhone中兼容 |
| 使用HLS（ HTTP Live Streaming）协议 | 兼容性比RTMP好，但是延时比其高 |

### 8.2.3设计决策的选择及分析

* 互操作性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 在提供的服务列表中定位所需要的服务 | 不采用。仅是关于直播的服务。 |
| 使用控制机制来管理、协调服务的调用 | 不采用。仅是关于直播的服务。 |
| 抽取并剪裁接口 | 采用。用于统一不同的数据格式。 |

* 可修改性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 划分模块 | 采用。是一种常用且有效的策略。 |
| 抽象通用服务 | 不采用。并没有通用服务需要被抽象。 |
| 添加接口 | 采用。增加可修改性常用的策略。 |
| 运行时注册 | 不采用。会带来额外的管理开销。 |

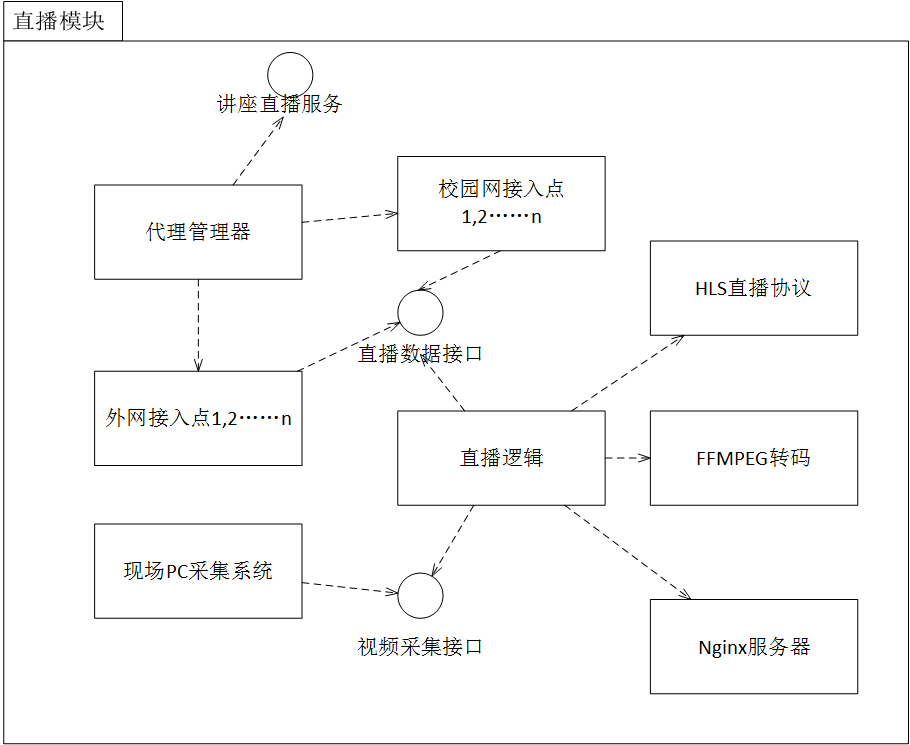
* 负载性能

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 维持计算的多个副本 | 不采用。考虑同步需要的代价太大。 |
| 增加可用资源 | 不采用。会增加较多成本。 |
| 区分校内校外网络接入 | 采用。适用于此类开发成本不应太大的系统。 |

* 实时性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 使用RTMP(Real Time Messaging Protocol)协议 | 不采用。并没有很好的兼容性。 |
| 使用HLS（ HTTP Live Streaming）协议 | 采用。该技术成熟且容易使用。此外，开源有益于控制开发系统的成本。 |

### 8.2.4第二次迭代结果



## 8.3第三次迭代过程

选取讲座报名模块作为系统元素进行分解。在质量属性方面，该模块与可获得性、可修改性、负载性能有密切关系。

### 8.3.1识别所选模块的ASR

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Architectural Drivers** | **Importance** | **Difficulty** |
| **1** | Scenario 2：可获得性 | high | medium |
| **2** | Scenario 4：可修改性 | medium | low |
| **3** | Scenario 5：负载性能：系统能够允许500个用户同时进行正常的访问、报名操作 | high | high |

### 8.3.2每个ASR可选的设计决策

* 可获得性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 命令/响应 | 一个组件发出一个命令，并希望在预定义的时间内收到一个来自审查组件的响应。与对所有进程发出命令的远程错误探测器相比，这种策略所使用的通信带宽更少 |
| 心跳 | 一个组件定期发出一个心跳信息，另一个组件收听该信息。可以在起到心跳的同时传递数据。 |
| 主动冗余（热启动） | 所有的冗余组件都以并行的方式对事件作出响应。错误发生时，使用该战术的系统的停机时间通常是几毫秒，因为备份是最新的，所以恢复所需的时间就是切换时间。 |
| 被动冗余 | 一个组件（主要的）对事件作出响应，并通知其他组件（备用的）必须进行的状态更新。该战术依赖于能够可靠地接管工作的备用组件。停机时间通常为几秒钟。 |
| 备件 | 备用件是计算平台配置用于更换各种不同的故障组件。当出现故障时，必须将其重新启动为适当的软件配置，并对其状态进行初始化。该战术的停机时间通常为几分钟。 |

* 可修改性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 划分模块 | 可以有效缩小要修改的范围 |
| 抽象通用服务 | 将多个模块都需要的服务提取出来，形成可供多个模块共同使用的通用服务 |
| 添加接口 | 将功能和实现分离，隔离需要更改的模块 |
| 运行时注册 | 支持即插即用操作，但需要管理注册的额外开销 |

* 负载性能

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **分析** |
| 维持计算的多个副本 | 需要考虑如何使副本保持一致和同步 |
| 增加可用资源 | 使用速度更快的处理器、额外的处理资源、额外的内存、速度更快的网络。但是会增加成本 |

### 8.3.3设计决策的选择及分析

* 可获得性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 命令/响应 | 不采用。 |
| 心跳 | 采用。常用的方式，且使系统的可获得性更高。 |
| 主动冗余（热启动） | 采用。该系统对于选课报名的可获得性要求很高，采用热备份可以获得更高的可获得性。 |
| 被动冗余 | 不采用。停机时间过长，对于讲座报名这种短时间内对可获得性要求极高的子系统而言并不够好。 |
| 备件 | 不采用。理由同上。 |

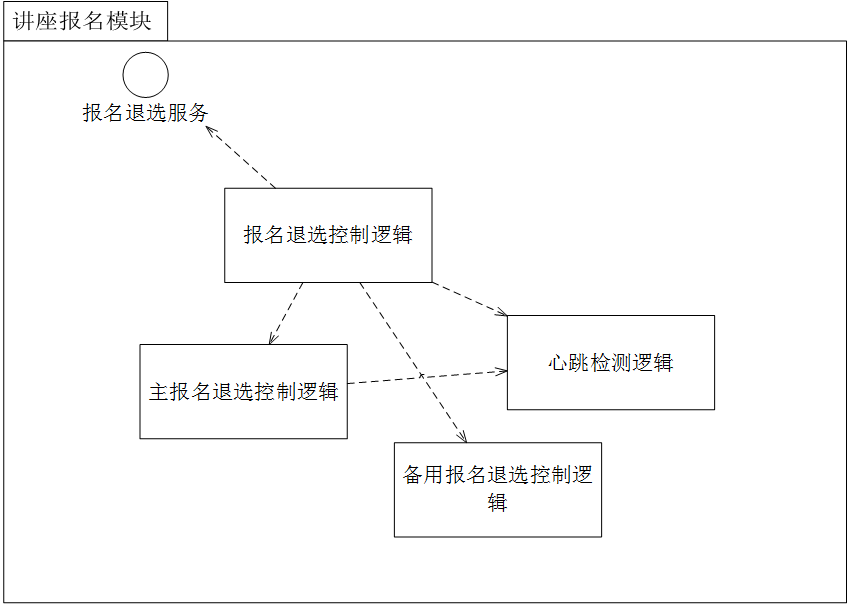
* 可修改性

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 划分模块 | 采用。是一种常用且有效的策略。 |
| 抽象通用服务 | 不采用。并没有通用服务需要被抽象。 |
| 添加接口 | 不采用。不适用于该子系统。 |
| 运行时注册 | 不采用。会带来额外的管理开销。 |

* 负载性能

|  |  |
| --- | --- |
| **可选的设计决策或模式** | **决策理由** |
| 维持计算的多个副本 | 采用。保证在一个计算逻辑失效后可以在较短的时间内启用备用逻辑。 |
| 增加可用资源 | 不采用。会增加较多成本。 |

### 8.3.4第三次迭代结果



# ATAM分析过程

## 9.1质量属性效用树

|  |  |  |
| --- | --- | --- |
| 质量属性 | 属性求精 | 场景 |
| 安全性 | 数据加密 | A1:用户数据被窃取，窃取者无法破译加密后的用户密码等敏感信息(H,H) |
|  |  | A2:定期更换加密密钥，当窃取者破解之前的加密密钥时，密钥已更换，窃取者仍然无法破译加密后的用户密码等敏感信息(M,M) |
|  | 身份验证 | A3:某用户未通过身份验证试图向探测器发送控制指令和访问图像数据，被系统拒绝访问(H,M) |
|  | 攻击侦测和防御 | A4:系统受到恶意攻击，系统侦测到攻击后，记录攻击日志，提醒管理员并锁住数据，直到确认安全(M,L) |
| 可用性 | 故障监测 | A5:采用命令/响应，一个组件发出一个命令，并在0.5s内收到一个来自审查组件的响应(M,H) |
|  |  | A6:采用心跳机制，系统某部分发生故障失去心跳后，系统在5min内检测到故障源(H,H) |
|  | 功能备份 | A7:当检测到故障后，系统在1s内切换至备用控制逻辑，恢复正常功能(H,H) |
|  | 故障重启 | A8:其他部分发生故障，若不能恢复，在5min内通过重启恢复正常功能(H,M) |
| 互操作性 |  | A9:当系统发起/收到与校园卡系统/教务网系统/现场PC端视频采集系统的请求/响应时，不需要经过数据格式转换，交换信息数据成功的概率不低于99%(H,H) |
| 可修改性 |  | A10:当与系统交互的教务系统，校园卡系统提供的接口发生修改时，与之相关的模块修改在1人月内完成(M,M) |
|  |  | A11:开发者更新某些系统模块的功能，修改预算未超过总预算的10%，其他无关的系统功能未受影响(H,L) |
| 性能 | 负载 | A12:系统至少允许500个用户同时进行正常的访问(H,H) |
|  |  | A13:报名操作/观看讲座直播用户访问所需页面的时间不超过0.1s(H,M) |
|  | 容量 | A14:系统至少能够存储50T的视频数据(H,L) |
|  | 实时性 | A15:当有大量用户同时观看直播时，系统直播延时不超过5s(H,H) |
| 可维护性 |  | A16:系统开发者希望修改加密密钥时，修改时间不超过1人日(M,L) |

## 9.2ATAM分析

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **场景：A1** | 经过授权的个人用户希望进入系统报名参与讲座，但系统无法提供正确的反馈，系统记录错误日志并在5s之内从错误中恢复 | | | |
| **质量属性** | 可用性 | | | |
| **环境** | 联网状态 | | | |
| **刺激** | 希望进入系统报名参与讲座时系统无法提供正确的反馈 | | | |
| **响应** | 系统记录错误日志并从错误中恢复 | | | |
| **架构决策** | **敏感点** | **权衡点** | **风险** | **非风险** |
| 心跳 | S4 | T6 | R2 |  |
| 主动冗余 | S5 | T2 | R3 |  |
| **理由说明** | 系统出错后恢复的速度会极大影响用户体验，心跳可在2s内检测到故障，主动冗余可以保证在出现故障后，系统的停机时间只有几毫秒。虽然会降低部分性能，但提高了可用性，也是值得的。 | | | |
| **相关架构图** |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **场景：A2** | 系统受到恶意攻击，系统侦测到攻击后，防火墙会识别攻击并阻断攻击 | | | |
| **质量属性** | 安全性 | | | |
| **环境** | 运行时 | | | |
| **刺激** | 外部攻击 | | | |
| **响应** | 系统侦测到攻击后，防火墙识别攻击并阻断攻击 | | | |
| **架构决策** | **敏感点** | **权衡点** | **风险** | **非风险** |
| 在系统服务收到访问时验证用户身份，授权给用户应有的权限 | S2 |  |  | **N2** |
| 自动攻击侦测 | S3 | T1 | R1 |  |
| 使用DDOS分防火墙 | S7 |  |  | N3 |
| **理由说明** | 本项目与学生的校园卡相关，并于教务网相关联，所以安全性尤为重要，所以对用户进行权限设置。  攻击者若不能通过用户认证，可能使用各种方式对系统发动攻击。为安全起见，自动侦测攻击并使用防火墙阻止攻击。虽然可能降低部分性能，但是这种风险是可以接受的，而泄密的风险是不可接受的。 | | | |
| **相关架构图** |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **场景：A3** | 系统发起/收到与校园卡系统/教务网系统/现场PC端采集系统的请求/响应，希望与其进行请求/响应数据交换 | | | |
| **质量属性** | 互操作性 | | | |
| **环境** | 运行时 | | | |
| **刺激** | 系统希望与外界系统进行请求/响应数据交换 | | | |
| **响应** | 请求被成功获取并成功交换数据 | | | |
| **架构决策** | **敏感点** | **权衡点** | **风险** | **非风险** |
| 抽取并裁剪接口 | S13 |  |  | N6 |
| **理由说明** | 本项目拥有校园卡支付，校园卡签到，教务网信息导入等功能，所以互操作性是很重要的质量属性。采用合适的接口，可以方便与外界进行数据交换，提高互操作性。 | | | |
| **相关架构图** |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **场景：A4** | 开发者修改系统用户界面、数据标准、控制逻辑等 | | | |
| **质量属性** | 可修改性 | | | |
| **环境** | 设计、开发系统时 | | | |
| **刺激** | 开发者希望修改系统用户界面、数据标准、控制逻辑等 | | | |
| **响应** | 需要修改的模块被正确修改，并不影响其他功能的实现 | | | |
| **架构决策** | **敏感点** | **权衡点** | **风险** | **非风险** |
| 划分模块 | S7 |  |  | N4 |
| 独立存储密钥数据的模块 | S11 |  |  | N5 |
| **理由说明** | 对于一个复杂的系统来说，系统的可修改性和可维护性都是非常重要的。采用模块划分的决策，增强内聚，可以有效缩小要修改的范围，是成熟的设计原则，可增强可修改性。并独立存储密钥数据的模块，可提高可维护性 | | | |
| **相关架构图** |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **场景：A5** | 有500名用户同时在系统上进行讲座报名操作/查看讲座直播 | | | |
| **质量属性** | 性能（负载） | | | |
| **环境** | 联网状态 | | | |
| **刺激** | 学生用户希望在系统上进行讲座报名操作/查看讲座直播 | | | |
| **响应** | 系统能够正常工作，为每一个用户提供相应 | | | |
| **架构决策** | **敏感点** | **权衡点** | **风险** | **非风险** |
| 增加可用资源 | S8 |  | R4 |  |
| 使用分布式数据库服务器 | S9 | T3 | R5 |  |
| **理由说明** | 使用速度更快的处理器、额外的处理资源、额外的内存、更快的网络，可以保证在负载方面提高系统性能。但这会使得成本偏高，造成风险。因此需要在可以负担的范围内尽可能增加可用资源。分布式数据服务器的性价比显著优于单台商业中型机和大型机，能够提供较为优质的数据服务，提高性能。 | | | |
| **相关架构图** |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **场景：A6** | 用户在系统上观看讲座直播 | | | |
| **质量属性** | 性能 | | | |
| **环境** | 运行时 | | | |
| **刺激** | 学生用户希望在系统上观看讲座直播 | | | |
| **响应** | 系统直播视频清晰流畅，延时短 | | | |
| **架构决策** | **敏感点** | **权衡点** | **风险** | **非风险** |
| 使用HLS（ HTTP Live Streaming）协议 | S10 | T4 | R6 |  |
| 使用FFMPEG转码 | S14 |  |  | N7 |
| **理由说明** | 使用HLS(HTTP LIVE STREAMING)协议，延时虽然较高，但仍在可接受的范围内。且兼容性比较好，提高了可移植性。在性能和可移植性兼备的情况下，选择了HLS新协议。使用FFMPEG转码，可以保证视频质量，同时提高了可移植性 | | | |
| **相关架构图** |  | | | |

## 9.3敏感点和权衡点

#### 9.3.1敏感点

|  |  |  |
| --- | --- | --- |
| S1 | 对用户信息进行加密 | 保护用户信息，影响了安全性(正面)，是**安全性的敏感点** |
| S2 | 在系统服务收到访问时验证用户身份，授权给用户应有的权限 | 有利于保证系统服务仅受到合法访问，是**安全性的敏感点** |
| S3 | 自动攻击侦测 | 出现攻击时可以自动侦测到，提高了安全性，是**安全性的敏感点**，同时会增加服务器负担，降低性能，是**性能的敏感点** |
| S4 | 心跳 | 一个组件定期发出一个心跳信息，另一个组件收听该信息。可以用来识别错误，是**可用性的敏感点**。同时可能占用资源，导致性能的下降，是**性能的敏感点** |
| S5 | 主动冗余 | 所有的冗余组件都以并行的方式对事件作出响应，错误发生时，使用该战术的系统停机时间通常是几毫秒，是**可用性的敏感点**。但备份占用了系统资源，影响了性能，是**性能的敏感点** |
| S6 | 使用DDOS分防火墙 | 对各种常见的攻击行为进行识别，并通过集成的机制实时对这些攻击流量进行处理及阻断，是**安全性的敏感点**。 |
| S7 | 划分模块 | 将系统模块化，在修改时可以有效缩小要修改的范围，是**可修改性的敏感点** |
| S8 | 增加可用资源 | 使用速度更快的处理器、额外的处理资源、额外的内存、速度更快的网络，可提高性能，是**性能的敏感点**。但会影响成本 |
| S9 | 采用分布式数据服务器 | 分布式数据服务器的性价比显著优于单台商业中型机和大型机，能够提供较为优质的数据服务，是**性能的敏感点（正面）**。但是由于数据分散存储，技术门槛较高，对维护工作，特别是数据完整性、一致性的维护提出了较高的要求，是**可维护性的敏感点（负面）** |
| S10 | 使用HLS(HTTP Live Streaming)协议 | 兼容性较好，提高了可移植性，是**可移植性的敏感点**。但延时比较高，影响了性能，是**性能的敏感点**。 |
| S11 | 独立存储密钥数据的模块 | 将密钥数据独立存储，可以提高可修改性，是**可修改性的敏感点** |
| S12 | 采用GlusterFS分布式文件系统 | 具有线性横向拓展能力，提高了可拓展性，是**可拓展性的敏感点**。增加了客户端的负载，占用了相当的CPU和内存，降低了性能，是**性能的敏感点**。 |
| S13 | 抽取并裁剪接口 | 针对请求响应交换的数据不一致情况，分别通过接口与外部系统相交互，是**互操作性的敏感点** |
| S14 | 使用FFMPEG转码 | FFMPEG转码提供了录制、转换以及流化音视频的完整解决方案，有助于播放清晰的视频，是**性能的敏感点**(正面),同时它包含了非常先进的音频/视频编解码库libavcodec，保证了高可移植性，是**可移植性的敏感点** |

#### 9.3.2权衡点

|  |  |  |
| --- | --- | --- |
| T1 | 自动攻击侦测 | 提高了安全性，但会增加服务器负担，降低性能，是**安全性和性能的权衡点** |
| T2 | 主动冗余 | 错误发生后恢复时间短，提高了可用性，但备份占用了系统资源，影响了性能，是**可用性和性能的权衡点** |
| T3 | 分布式数据服务器 | 性价比高，以相同成本提供更佳的容量和读写速率，但是维护工作技术门槛高，是**性能和可维护性的权衡点** |
| T4 | 使用gHLS(HTTP Live Streaming)协议 | 兼容性较好，提高了可移植性，但延时比较高，影响了性能，是性能和可移植性的权衡点 |
| T5 | 采用GlusterFS分布式文件系统 | 提高了可拓展性，同时降低了性能，是**可拓展性和性能的权衡点** |
| T6 | 心跳 | 提高了可用性，但降低了性能，是**可用性和性能的权衡点** |

## 9.4风险和非风险

#### 9.4.1风险

|  |  |  |
| --- | --- | --- |
| R1 | 自动攻击侦测 | 攻击不经常发生，系统一直自动侦测攻击，可能导致对资源的很大浪费，影响性能 |
| R2 | 心跳 | 可能造成性能负担，影响系统性能 |
| R3 | 主动冗余 | 备份占用了系统资源，可能会影响系统性能 |
| R4 | 增加可用资源 | 可提高性能，但成本较高 |
| R5 | 分布式数据服务器 | 分布式数据服务器技术门槛高，提高人力成本。如果维护不当，数据的完整性和一致性可能出现问题。 |
| R6 | 使用gHLS(HTTP Live Streaming)协议 | 延时比较高，可能影响系统性能 |
| R7 | 采用GlusterFS分布式文件系统 | 可能会增加客户端的负载，占用相当的CPU和内存，降低性能 |

#### 9.4.2非风险

|  |  |  |
| --- | --- | --- |
| N1 | 对用户信息进行加密 | 多用户系统和涉密系统的标准配置 |
| N2 | 在系统服务收到访问时验证用户身份，授权给用户应有的权限 | 多用户系统和涉密系统的标准配置 |
| N3 | 使用DDOS分防火墙 | 可以有效识别并阻断攻击，有效提高安全性 |
| N4 | 划分模块 | 成熟的设计原则，显著利于增强可修改性、可移植性 |
| N5 | 独立存储密钥数据的模块 | 同上 |
| N6 | 抽取并裁剪接口 | 通过接口与外部系统交互，提高了互操作性 |
| N7 | 使用FFMPEG转码 | 提高了性能和可移植性 |

# 挑战和经验

# 组员和分工

|  |  |  |
| --- | --- | --- |
| **组员** | **学号** | **分工** |
| 梁思宇 | 131250129 |  |
| 邹卓晋 | 131250158 |  |
| 曾婧 | 131250159 |  |
| 吴超月 | 131250168 |  |
| 罗瑶 | 131250177 |  |
| 陈云龙 | 131250181 |  |
| 倪小凡 | 131250185 |  |
| 丁霄汉 | 131250207 |  |