

基于JOB作业平台实现游戏日志备份

 蓝鲸智云 *Tencent* 腾讯

目录

- 课程回顾
- 开发案例需求分析与设计
- 课程重难点分析
- 开发实战
- 实验评分标准

我们学会了什么？

- 创建、运行和部署SaaS应用
- 通过调用API Gateway实现接口服务
- 实现前后端的分离

我们遇到了什么问题？

- 数据库相关：migrate和makegrations
- 访问路径相关：404 Not Found
- 前后端分离相关：跨域问题
- 其它：环境安装、代码拼写

目录

- 课程回顾
- 开发案例需求分析与设计
- 课程重难点分析
- 开发实战
- 实验评分标准

●课题目标：

- 1、了解蓝鲸作业平台的功能与使用方法
- 2、掌握作业平台的基本概念和作业平台的基本使用
- 3、熟悉脚本的概念和语法，学习简单脚本的编写
- 4、掌握作业平台“方案执行”、“获取作业执行状态”等接口的调用
- 5、熟悉Django模型的设计和ORM操作，完成备份记录的建模和数据库读写功能
- 6、提升SaaS应用的开发技能，巩固Django基础知识

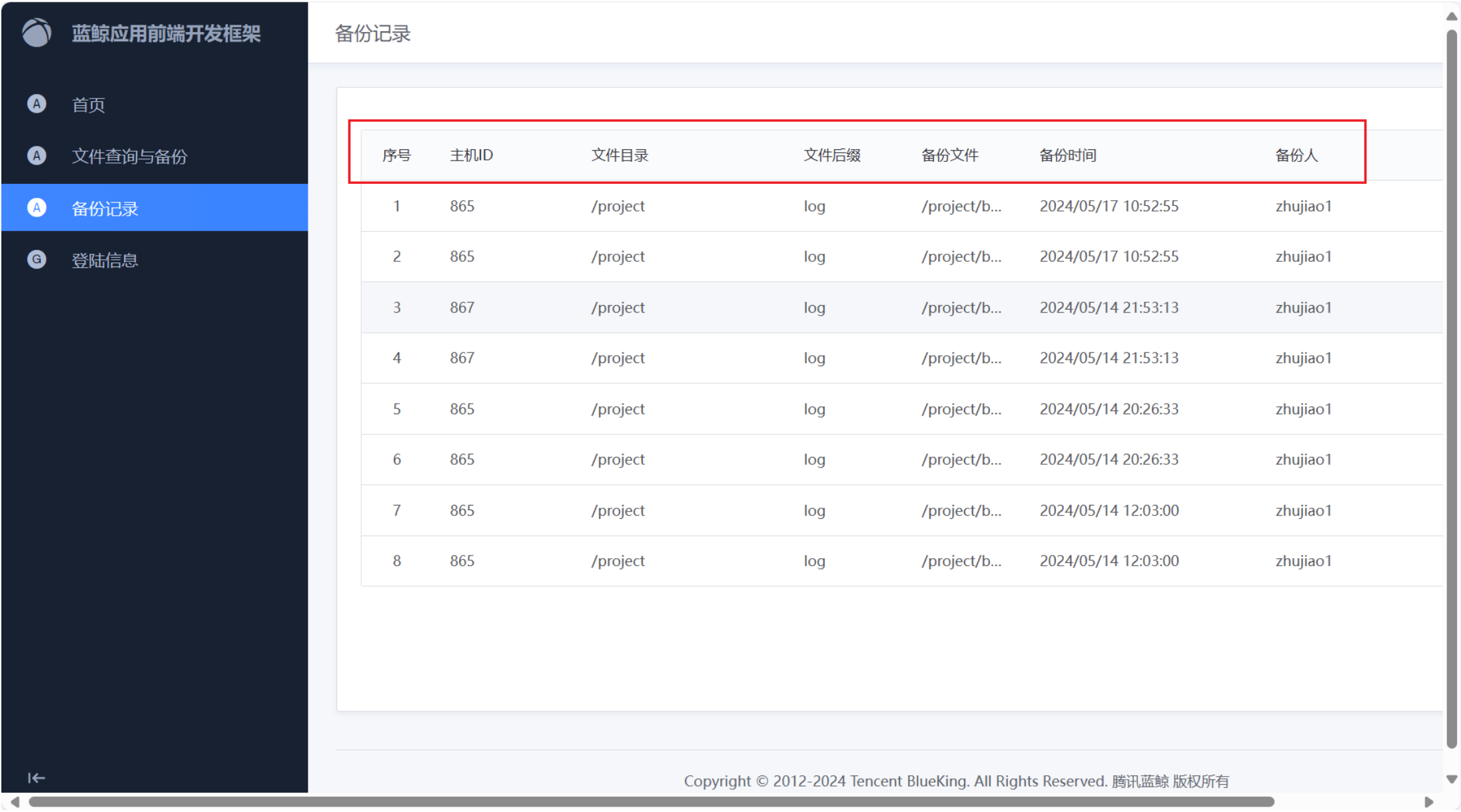
●课题内容：

在《基于CMDB配置平台管理游戏的主机》的基础上，借助蓝鲸作业平台编写脚本，通过蓝鲸网关/ESB组件API调用作业平台接口实现文件的查询和备份，并根据文件查询和备份条件，设计对应接口；同时记下备份记录，实现备份记录的查询功能。

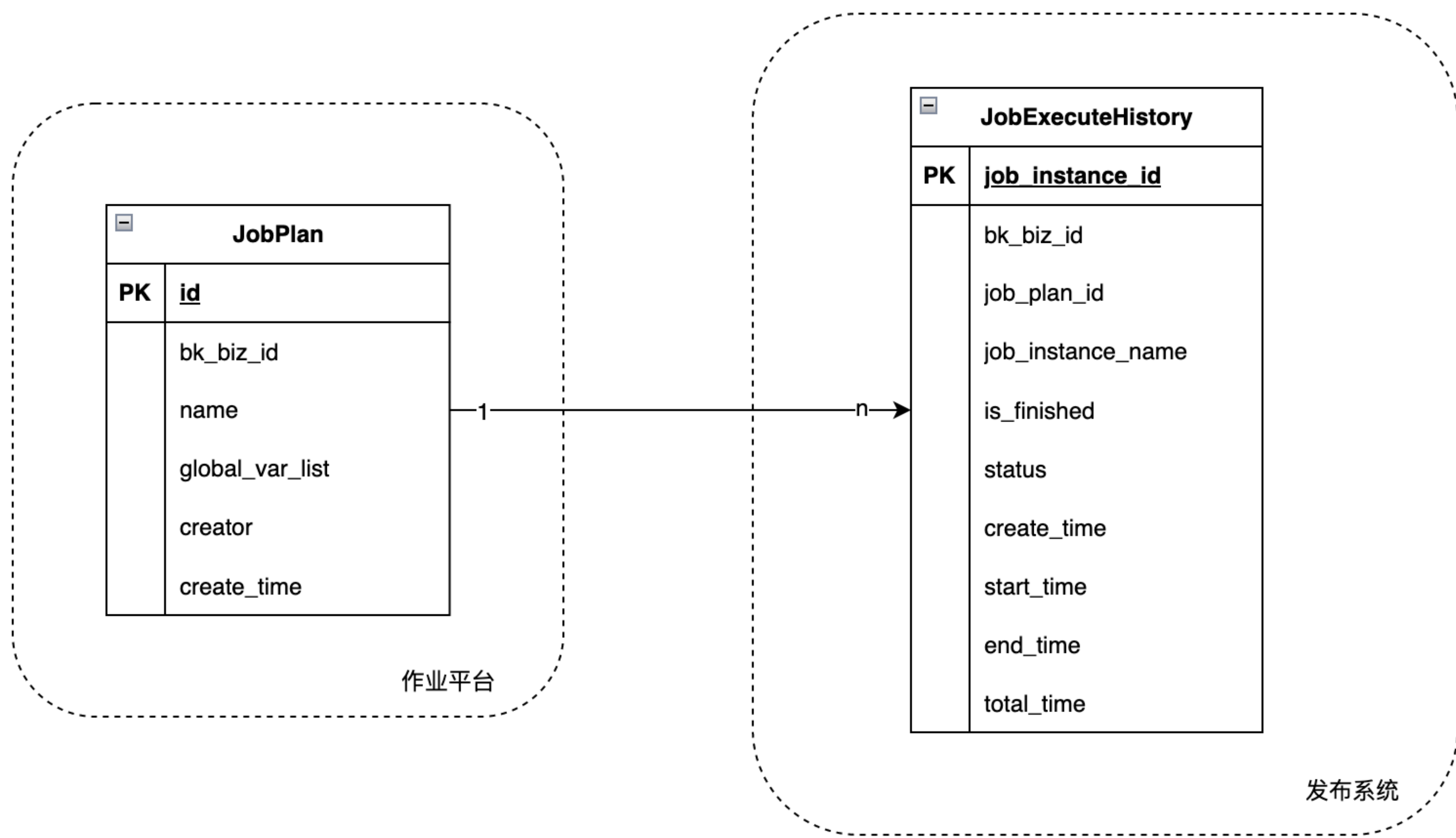
●样例展示：[基于JOB作业平台实现游戏日志备份-案例](#)

需求分析-功能点

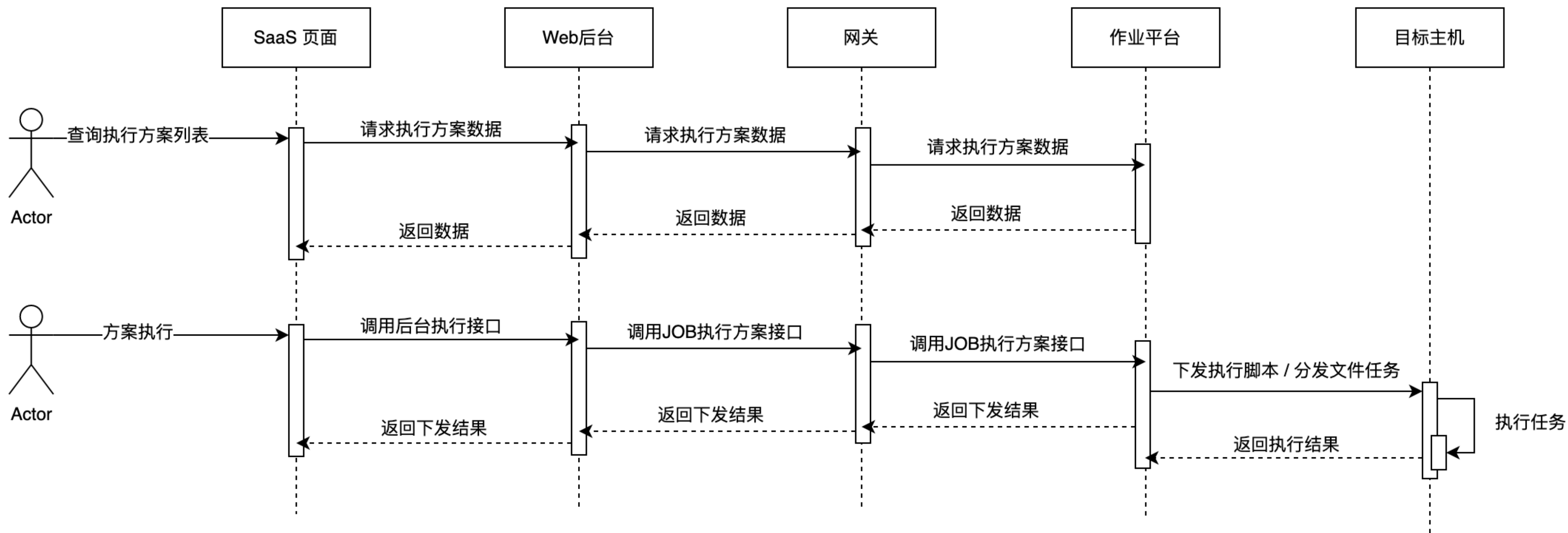
- 功能一： 主机文件查询
- 功能二： 主机文件备份
- 功能三： 备份记录展示



// 需求分析 – 业务系统资源样例



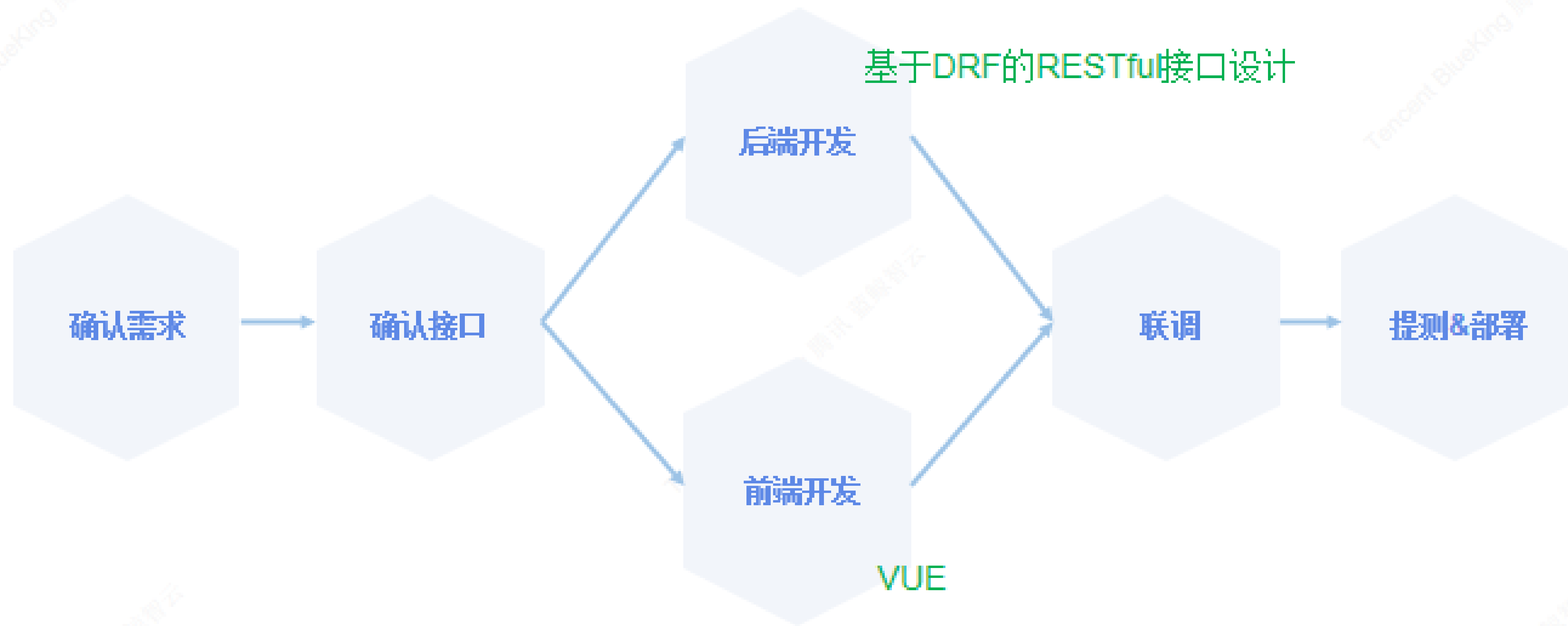
系统时序流程图样例



// 为什么需要前后端分离

- 后端接口的开发语言与模板解耦
- 后端接口可以在不同的终端复用
- 明确前后端开发职责
- 前后端可以并行开发

基于DRF的RESTfu接口设计



目录

- 课程回顾
- 开发案例需求分析与设计
- 重难点分析
- 开发实战
- 实验评分标准

// 重难点分析

- 业务逻辑比较复杂
- JOB接口输入输出参数比较复杂（存在嵌套结构）
- 接口的实现需要调用多个JOB接口并进行数据处理

课程讲解：

- 如何理清作业平台流程与其接口的对应关系
- 如何使用接口调用工具（以apifox为例）
- 如何在开发过程中使用断点来阶段性验证代码的正确性

目录

- 课程回顾
- 开发案例需求分析与设计
- 课程重难点分析
- 开发实战
- 实验评分标准



功能一：主机文件查询

// 接口设计: 主机文件查询

文件查询

GET /search-file • 开发中 ▾

创建时间 2024年5月14日 修改时间

目录 本地SaaS应用

Mock >

请求参数

Query 参数

host_id_list string 必需

主机列表, 用英文逗号进行分隔

示例值: 3

search_path string 必需

示例值: /project

suffix string 必需

示例值: log

示例

异常示例

成功示例

```
{
  "result": false,
  "code": 0,
  "message": "search failed"
}
```

示例

异常示例

成功示例

```
{
  "result": true,
  "code": 0,
  "data": [
    {
      "bk_file_cnt": 2,
      "bk_file_list": "test0.log;test1.log",
      "bk_file_total_size": 982,
      "bk_host_id": 865
    }
  ]
}
```

业务 demo体验业务

查询目录

/project

查找后缀

log

主机

865-10.0.48.45

备份目录

/project/backup

查询

主机ID

文件列表

文件数量

文件总大小(字节)

操作

865

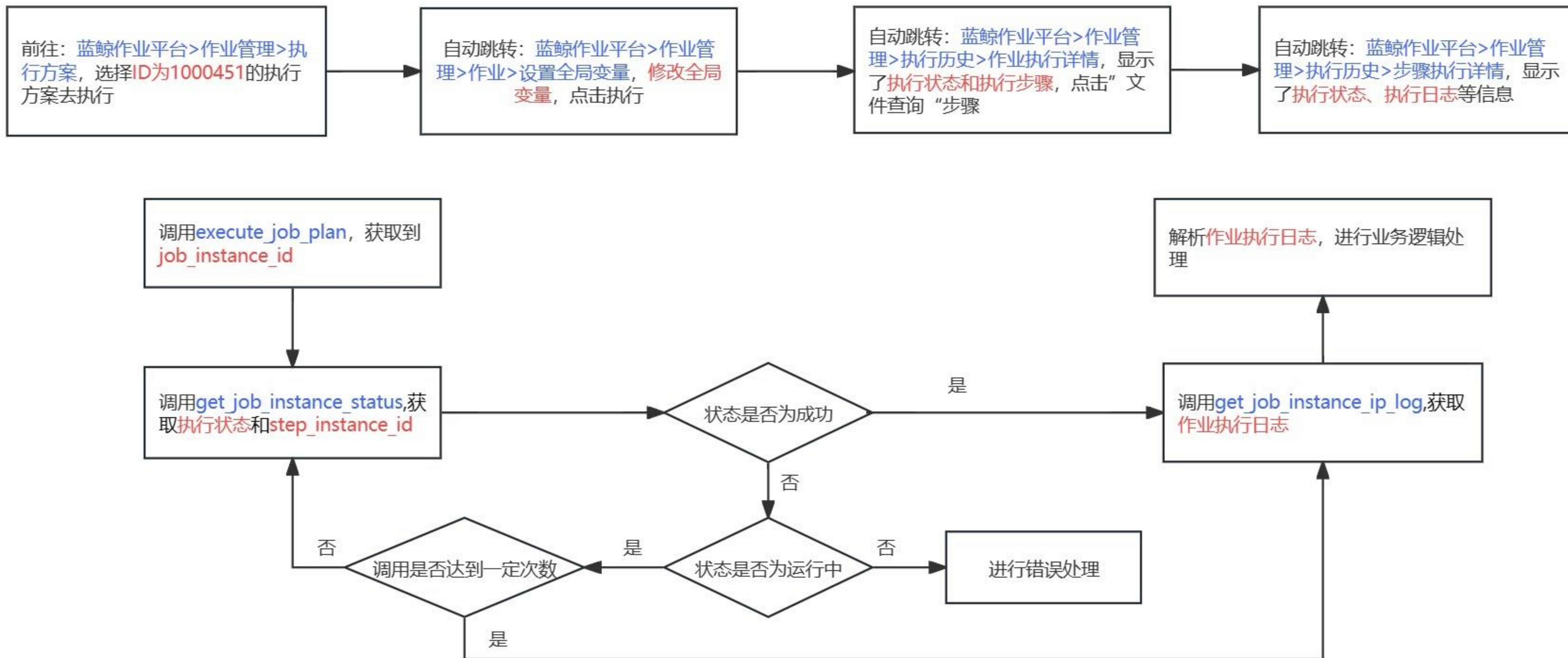
test0.log;test1.log

2

904

立即备份

执行作业方案的流程



// 作业平台: 执行作业执行方案 API

接口文档链接: https://apigw.ce.bktencent.com/docs/component-api/default/JOBV3/execute_job_plan/doc

返回结果示例

```
{
  "result": true,
  "code": 0,
  "message": "success",
  "data": {
    "job_instance_name": "Test",
    "job_instance_id": 10000
  }
}
```

接口参数

字段	类型	必选	描述
bk_biz_id	long	是	业务ID
job_plan_id	long	是	作业执行方案ID
全局变量。对于作业执行方案中的全局变量值，如果请求			

global_var_list	array	否
callback_url	string	否

global_var

字段	类型	必选
id	long	否
name	string	否
value	string	否
server	object	否

【华工课程】文件查询demo

全局变量

 **host_list**
5 台主机

 **search_path**
/project

 **suffix**
log

执行步骤

[1] 文件查询

去执行

编辑 (1/1)

定时执行

去同步

删除

成功与否。true:请求成功; false请求失败
编码。0表示success, >0表示失败错误
失败返回的错误信息
返回的数据
信息
连id

// 作业平台: 根据作业实例 ID 查询作业执行状态 API

接口文档链接: https://apigw.ce.bktencent.com/docs/component-api/default/JOBV3/get_job_instance_status/doc

接口参数

字段	类型	必选	描述
bk_scope_type	string	是	资源范围类型。可选值: biz - 业务, biz_set - 业务集
bk_scope_id	string	是	资源范围ID, 与bk_scope_type对应, 表示业务ID或者业务集ID
job_instance_id	long	是	作业实例ID
return_ip_result	boolean	否	是否返回每个主机上的任务详情, 对应返回结果中的step_ip_result_list。默认值为false。

job_instance

字段	类型	描述
name	string	作业实例名称
status	int	作业状态码: 1.未执行; 2.正在执行; 3.执行成功; 4.执行失败; 5.跳过; 6.忽略错误; 7.等待用户; 8.手动结束; 9.状态异常; 10.步骤强制终止中; 11.步骤强制终止成功
create_time	long	作业创建时间, Unix时间戳, 单位毫秒
start_time	long	开始执行时间, Unix时间戳, 单位毫秒
end_time	long	执行结束时间, Unix时间戳, 单位毫秒
total_time	int	总耗时, 单位毫秒
bk_scope_type	string	资源范围类型。可选值: biz - 业务, biz_set - 业务集
bk_scope_id	string	资源范围ID, 与bk_scope_type对应, 表示业务ID或者业务集ID
job_instance_id	long	作业实例ID

返回结果参数说明

response

字段	类型	描述
result	bool	请求成功与否。true:请求成功; false请求失败
code	int	错误编码。0表示success, >0表示失败错误
message	string	请求失败返回的错误信息
data	object	请求返回的数据
permission	object	权限信息

data

字段	类型	描述
finished	bool	作业是否结束
job_instance	object	作业实例基本信息。见job_instance定义
step_instance_list	array	作业步骤列表。见step_instance定义

// 作业平台: 根据作业实例ID查询作业执行日志 API

接口文档链接: https://apigw.ce.bktencent.com/docs/component-api/default/JOBV3/get_job_instance_ip_log/doc

接口参数

字段	类型	必选	描述
bk_scope_type	string	是	资源范围类型。可选值: biz - 业务, biz_set - 业务集
bk_scope_id	string	是	资源范围ID, 与bk_scope_type对应, 表示业务ID或者业务集ID
job_instance_id	long	是	作业实例ID
step_instance_id	long	是	步骤实例ID
bk_host_id	long	否	目标主机host_id
bk_cloud_id	int	否	不推荐使用, 建议使用host_id参数, 如果存在bk_host_id将忽略该参数。目标服务器管控区域ID
ip	string	否	不推荐使用, 建议使用host_id参数, 如果存在bk_host_id将忽略该参数。目标主机IP

日志查看: [蓝鲸作业平台>作业管理>执行历史>查看详情>点击相应的步骤](#)

0:10.0.48.45

执行日志

变量明细

```
1 {
2   "bk_file_total_size": 904,
3   "bk_file_list": "test0.log;test1.log",
4   "bk_file_cnt": 2
5 }
```

response

字段	类型	描述
result	bool	请求成功与否。true:请求成功; false请求失败
code	int	错误编码。0表示success, >0表示失败错误
message	string	请求失败返回的错误信息
data	object	请求返回的数据
permission	object	权限信息

data

字段	类型	描述
bk_host_id	long	主机ID
bk_cloud_id	int	目标服务器管控区域ID
ip	string	目标服务器IP地址
log_type	int	日志类型。1-脚本执行任务日志;2-文件分发任务日志
log_content	string	作业脚本输出的日志内容
file_logs	array	文件分发任务日志。定义见file_log

// 接口实现：主机文件查询

第一步： urls.py文件路由配置

- 定义对外接口的路径和views对应的实现映射

第二步： views.py 逻辑实现

- 定义接口输入参数
- 组装请求参数，调用作业平台接口
- 处理作业平台接口返回的数据
- 返回结果参数

第三步： 启动应用， 接口测试

The screenshot displays the Apifox interface for managing and testing APIs. The left sidebar contains navigation options: 接口管理 (Interface Management), 自动化测试 (Automated Testing), 分享文档 (Share Documents), 请求历史 (Request History), 项目设置 (Project Settings), and 邀请成员 (Invite Members).

The main area shows a list of interfaces under the '接口管理' tab. The selected interface is 'GET 文件查询' (GET File Query). The configuration details for this interface are shown on the right:

- 请求 (Request):** Method is GET, URL is /search-file. A red box highlights the URL field with the label '请求的url' (Request URL) and '设置url前缀' (Set URL prefix).
- Params (Parameters):** A table lists query parameters: host_id_list (value: 865), search_path (value: /project), and suffix (value: log). A red box highlights this table with the label '填写请求参数' (Fill in request parameters).
- Body (Body):** The response body is shown in JSON format. A red box highlights the JSON content with the label '请求结果' (Request result).
- Response (Response):** The status is 200, time is 2.43 s, and size is 142 B. The response is labeled '成功 (200)' (Success (200)).

The bottom of the interface shows the '文档模式' (Document Mode) and '调试模式' (Debug Mode) tabs, along with status indicators for '在线' (Online), 'Cookie 管理' (Cookie Management), and other settings.



功能二：主机文件备份

// 接口设计: 主机文件备份

文件备份

GET /backup-file 开发中

创建时间 2024年5月14日 修改时间

Mock >

请求参数

Query 参数

host_id_list string 必需

示例值: 865

search_path string 必需

示例值: /project

suffix string 必需

示例值: log

backup_path string 必需

示例值: /project/backup

示例 异常示例 成功示例

```
{  "result": true,  "data": "success",  "code": 0}
```

示例 异常示例 成功示例

```
{  "result": false,  "code": 0,  "message": "backup failed"}
```

业务 demo体验业务

文件备份成功!

查询目录 /project 查找后缀 log 主机 865-10.0.48.45

备份目录 /project/backup 查询

主机ID	文件列表	文件数量	文件总大小(字节)	操作
865	test0.log;test1.log	2	904	立即备份

// 接口实现：主机文件备份

第一步.: urls.py文件路由配置

- 定义对外接口的路径和views对应的实现映射

第二步: views.py 逻辑实现

- 定义接口输入参数
- 组装请求参数，调用作业平台接口
- 处理作业平台接口返回的数据
- 返回结果参数

```
for step_res in json_step_res:
    # 创建备份记录
    step_res["bk_host_id"] = bk_host_id
    step_res["bk_file_dir"] = search_path
    step_res["bk_file_suffix"] = suffix
    step_res["bk_file_operator"] = request.user.username
    step_res["bk_job_link"] = "{} /biz/{}/execute/task/{}".format(
        *args: BK_JOB_HOST,
        JOB_BK_BIZ_ID,
        job_instance_id,
    )
    BackupRecord.objects.create(**step_res)
```

第三步: 启动应用，接口测试



功能三：备份记录展示

// 接口设计: 备份记录展示

备份记录查询

GET

/backup-record

● 开发中

▼

创建时间 2024年5月14日 修改时间

Mock >

请求参数

Cookie 参数

bk_token

string

可选

```
{
  "result": true,
  "data": [
    {
      "id": 2,
      "bk_host_id": 865,
      "bk_file_dir": "/project",
      "bk_file_suffix": "log",
      "bk_backup_name": "/project/backup/backup_20240514_120300/test1.log",
      "bk_file_create_time": "2024/05/14 12:03:00",
      "bk_file_operator": "zhujiao1",
      "bk_job_link": "https://job.ce.bktencent.com/biz/3/execute/task/20002152124"
    },
    {
      "id": 1,
      "bk_host_id": 865,
      "bk_file_dir": "/project",
      "bk_file_suffix": "log",
      "bk_backup_name": "/project/backup/backup_20240514_120300/test0.log",
      "bk_file_create_time": "2024/05/14 12:03:00",
      "bk_file_operator": "zhujiao1",
      "bk_job_link": "https://job.ce.bktencent.com/biz/3/execute/task/20002152124"
    }
  ],
  "code": 0
}
```

// 接口实现：备份记录展示

第一步.: urls.py文件路由配置

- 定义对外接口的路径和views对应的实现映射

第二步： views.py 逻辑实现

- 定义接口输入参数
- 调取数据库数据
- 对数据库数据进行处理
- 返回结果参数

第三步： 启动应用，接口测试

目录

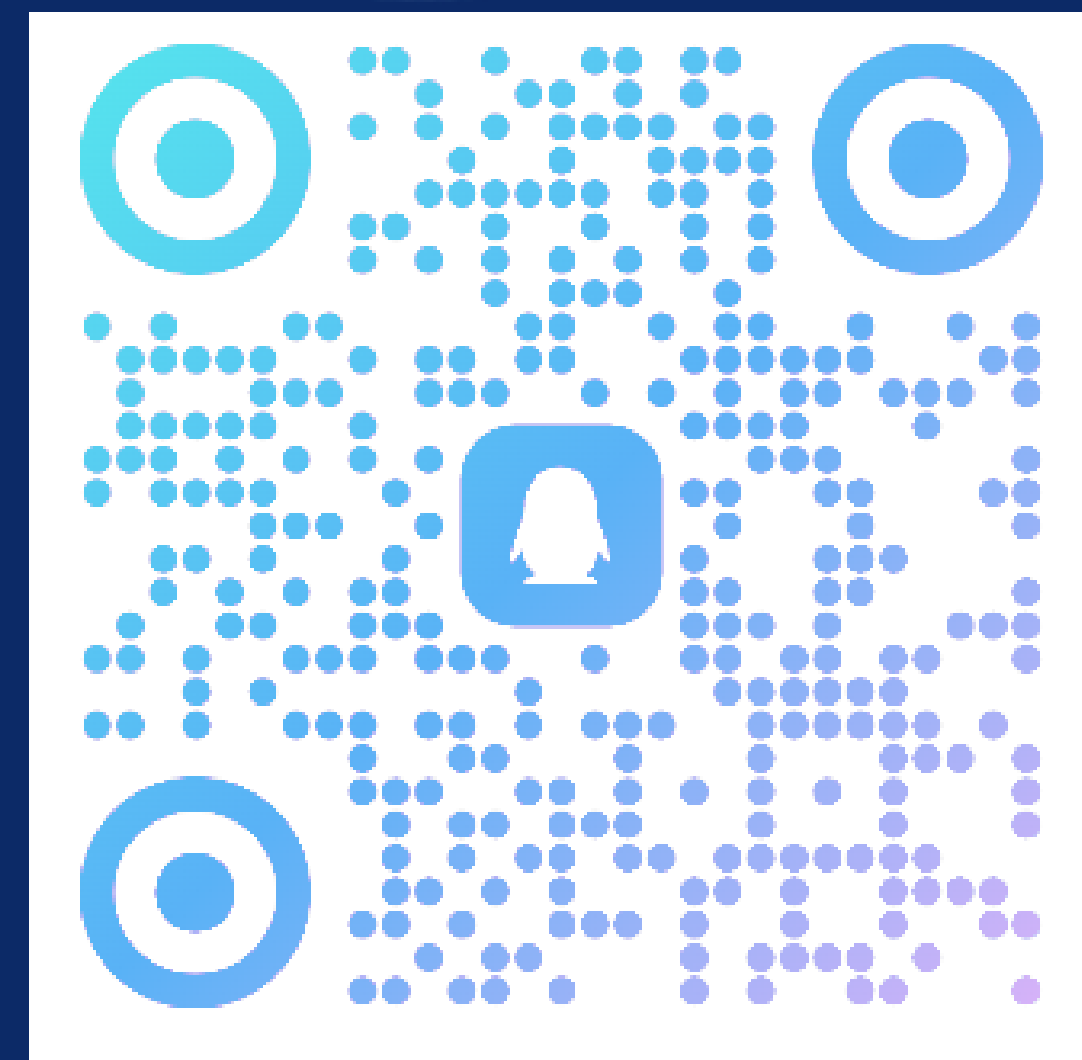
- 课程回顾
- 开发案例需求分析与设计
- 课程重难点分析
- 开发实战
- 实验评分标准

// 实验评分标准

整体要求	采用 迭代方式 进行需求分析、面向对象设计和编程实现，实训课报告中需包含相应的 需求规约 、 设计规约 ， 项目开发说明
考点一	创建SaaS应用，通过 蓝鲸ESB组件API 联通 CMDB配置平台 ，实现 业务、集群、模块级联拉取接口 ，并在前端进行下拉框组件展示与数据渲染
考点二	添加 根据 蓝鲸CMDB配置平台 的主机数据结构 设计查询条件 （包括但不限于主机名称、主机维护人、主机备份人等字段）， 实现主机查询接口 （ 模糊查询 可加分）
考点三	设计前端界面（可参考课程前端样例代码与 MagicBox 组件库），进行 前后端联调 ，实现 前端主机列表数据渲染
考点四	实现 主机详情展示接口&界面 ，要求点击主机后能够查看主机详情信息并通过前端界面进行数据展示
考点五	将实现的后端&前端代码上传至 Git 代码托管平台，并 部署到PaaS平台 ，数据交互展示无误， 不存在CORS、CSRF等问题
其他评分项	1. Python代码符合 PEP8规范 ，可酌情加分
	2. 系统边界 考虑完善， 系统性能 优良，可酌情加分
	3. 前端 界面 优美， 用户交互 体验良好，可酌情加分
	4. 实现 主机状态拉取 、 主机批量选择 等功能，可酌情加分
	5. 使用 异步任务 进行文件备份，可酌情加分
	6. 后端代码能够实现 单元测试 以及 日志 、 异常处理 等，可酌情加分



官方微信公众号



蓝鲸高校培训群