

使用介绍：文件上传下载服务开发

接上一篇《zPaaS低代码平台使用介绍：第一个功能开发》，本篇主要介绍一下如何使用zPaaS低代码开发平台进行文件上传下载服务的快速开发。

1. 开发的功能说明

文件的上传与下载服务涉及到服务端文件的存储，目前低代码平台的后端文件存储支持本地文件系统、FTP/SFTP以及MinIO四种文件存储类型，本次功能开发将采用MinIO作为后端文件存储服务，实现web文件的上传、下载和删除服务，并通过前端的文件组件，实现简单的文件上传、下载与删除功能，实现的示例前端功能为《我的学习记录》。

涉及的数据库表为《my_study_record》，建表语句如下：

```
CREATE TABLE `my_study_record` (  
  `id` varchar(32) NOT NULL COMMENT '主键',  
  `user_id` varchar(32) NOT NULL COMMENT '用户标识',  
  `study_date` date NOT NULL COMMENT '学习日期',  
  `class_name` varchar(3) DEFAULT NULL COMMENT '课程',  
  `study_content` varchar(100) NOT NULL COMMENT '学习内容',  
  `study_time` int NOT NULL COMMENT '学习时间',  
  `attachments` varchar(512) DEFAULT NULL COMMENT '附件',  
  `create_time` datetime DEFAULT NULL COMMENT '创建时间',  
  `update_time` varchar(100) DEFAULT NULL COMMENT '更新时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci COMMENT='我的学习记录';
```

其中：

- 用户标识user_id字段：取登录用户的用户标识
- 课程class_name字段：为下拉类型，下拉选择为

```
[  
  {  
    "label": "语文",  
    "value": "CHN"  
  },  
  {  
    "label": "数学",  
    "value": "MAT"  
  },  
  {  
    "label": "英语",  
    "value": "ENG"  
  },  
  {  
    "label": "历史",  
    "value": "HIS"  
  },  
  {  
    "label": "地理",  
    "value": "GEO"  
  }  
]
```

- ```

 },
 {
 "label": "生物",
 "value": "BIO"
 },
 {
 "label": "道法",
 "value": "PLI"
 }
]

```
- 附件attachments字段：用于存储文件的存储信息，其对应的文件上传下载组件的配置信息为
 

```

{
 "listType": "P",
 "fileLimit": 3,
 "maxFileSize": 10000000,
 "uploadTip": "最多上传3个文件，每个文件大小不超过10MB。",
 "multiple": true,
 "accept": ".jpg,.jpeg,.png",
 "uploadService": "/file/upload",
 "deleteService": "/file/delete",
 "downloadService": "/file/download"
}

```

## 2. 前提条件

目前低代码平台未直接提供对数据库以及数据库表的管理功能，因此需要使用其他工具，提前创建一个数据库，并在数据库中创建好对应的表。

前提工作：

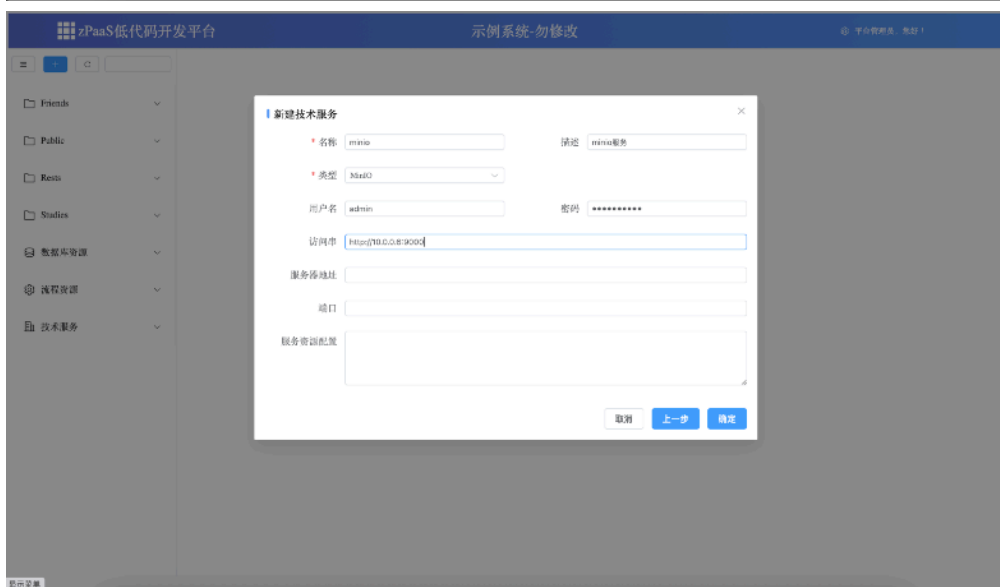
- (1) 部署mysql数据库
- (2) 创建一个database，如demo
- (3) 在demo库中，使用如上的建表语句，创建my\_study\_record表
- (4) 已经部署minio服务，访问url：http://10.0.0.6:9000，用户名密码：admin/minio@123z，已经创建bucket：demo-bucket

## 3. 准备工作

同《zPaaS低代码平台使用介绍：第一个功能开发》的准备工作，其中文件的上传下载服务放到公共域Public，我的学习记录相关对象放到学习域Studies。

## 4. 开发上传下载服务

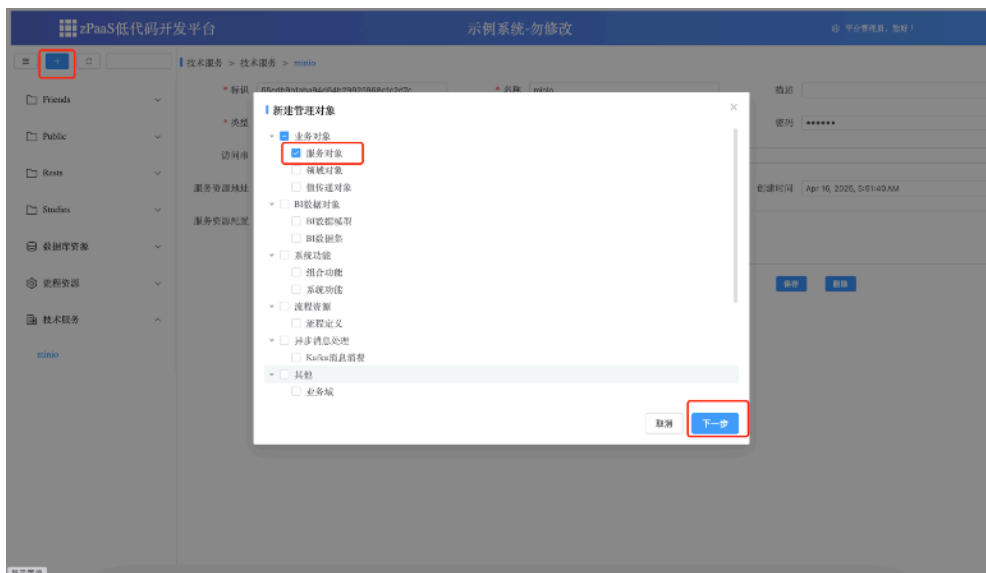
- (1) 使用工作台左上角的“+”按钮，创建一个技术服务，用于配置MinIO服务



(2) 创建好的minio服务如下：



(3) 新建文件服务对象并在该对象中创建上传、下载和删除三个方法



(4) 上传方法的出入参配置，入参包括文件名fileName和文件类型contentType，出参为Json类型。

方法编辑

\* 标识

3d960733121b4539ad2c41cc4194b4db

\* 名称

上传

\* 业务流标识

942465f8a7fd417aa814b4db581553c6

\* 编码

upload

\* 状态

有效

参数 新建

| 操作                                    | 编码          | 名称   | 是否入参 | 是否列表 | 参数类型                                 |
|---------------------------------------|-------------|------|------|------|--------------------------------------|
| <a href="#">编辑</a> <a href="#">删除</a> | fileName    | 文件名  | 是    | 否    | Java原生类型(java.lang.String)           |
| <a href="#">编辑</a> <a href="#">删除</a> | contentType | 文件类型 | 是    | 否    | Java原生类型(java.lang.String)           |
| <a href="#">编辑</a> <a href="#">删除</a> | return      | 返回值  | 否    | 否    | Java原生类型(com.google.gson.JsonObject) |

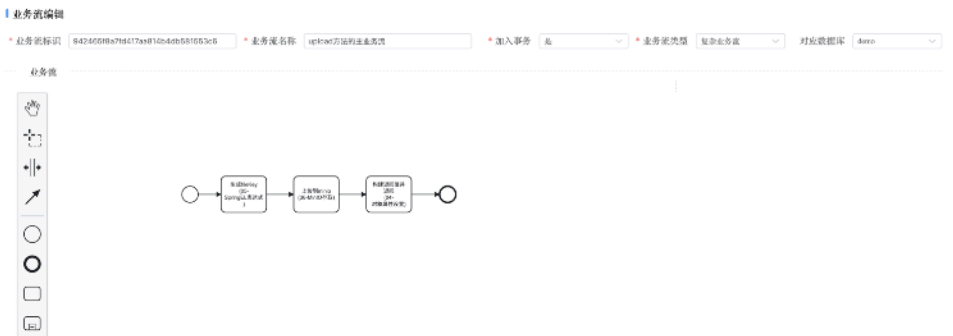
由入参校验规则设置

入参校验规则

选择 清空

取消

保存



节点说明

生成fileKey

节点类型

05-SpringEL表达式

查看可用数据

节点处理

SpEL表达式

\*{1}+'\_'+\*{2}

设置取值

值对象来源

输入参数

选择

是否列表

值对象Key

属性

取值方式

原值

上下文Key

添加

删除

来源

列表

Key

X

输入参数

否

fileName

X

固定值

否

UIDO

节点后处理

☒ 作为过程数据输出

存储关键字

fileKey

转换规则

设置 清空

☐ 作为领域对象输出

☐ 作为业务流程结果输出

节点说明

上传者minio

节点类型

36-MinIO存取

查看可用数据

节点处理

Minio管理

minio

Minio操作

上传

源文件来源

上传文件

源文件名来源

输入参数

选择

源文件Key

fileName

源文件名称属性

桶名称来源

固定值

选择

桶名称Key

demo-bucket

桶名称属性

内容类型来源

输入参数

选择

内容类型Key

contentType

内容类型属性

对象Key来源

过程数据

选择

对象KeyKey

fileKey

对象Key属性

节点后处理

☒ 作为过程数据输出

存储关键字

fileId

转换规则

设置 清空

☐ 作为领域对象输出

☐ 作为业务流程结果输出

节点说明

构建源路径并返回

节点类型

04-对象属性设置

查看可用数据

节点处理

目标对象来源

Select

选择

是否列表

目标对象Key

目标对象属性

目标对象类型

Java源生类型

目标对象

com.google.gson.JsonObject

设置规则

源对象类型

Select

选择

是否列表

源对象Key

属性映射规则

[{"fromAttrPath":"","toObjectAttr":""}]

添加

删除

来源

是否列表

Key

X

输入参数

否

fileName

X

过程数据

否

fileKey

节点后处理

☐ 作为过程数据输出

☐ 作为领域对象输出

☒ 作为业务流程结果输出

转换规则

设置 清空

(6) 下载方法的出入参配置，入参为文件的fileKey，出参的类型为java.io.File

方法编辑

标识: 93d1a4aab6174f2a8f3262e3f8288ff4 编码: download 名称: 下载 状态: 有效 业务流标识: c3f44eab6bbf457198569eac0aab38c2

参数

| 操作    | 编码      | 名称   | 是否入参 | 是否列表 | 参数类型                       |
|-------|---------|------|------|------|----------------------------|
| 编辑 删除 | fileKey | 文件id | 是    | 否    | Java原生类型(java.lang.String) |
| 编辑 删除 | return  | 返回值  | 否    | 否    | Java原生类型(java.io.File)     |

入参校验规则设置

入参校验规则: 选择 清空

取消 保存

(7) 下载方法的业务逻辑配置，只包含一个节点，根据fileKey从mioio中下载文件并将该文件作为业务流结果返回。

业务流编辑

业务流标识: c3f44eab6bbf457198569eac0aab38c2 业务流名称: download方法的业务流 加入事务: 是 业务流类型: 复杂业务流 输出数据串: demo

业务流

节点说明: 从minio下载

节点类型: 36-MinIO存取 查看可用数据

节点处理

Minio资源: minio Minio操作: 下载

源文件来源: Select

源文件名来源: Select 选择

源文件名Key: 源文件名属性:

桶名称来源: 固定值 选择

桶名称Key: demo-bucket 桶名称属性:

内容类型来源: Select 选择

内容类型Key: 内容类型属性:

对象Key来源: 输入参数 选择

对象KeyKey: fileKey 对象Key属性:

节点后处理

☐ 作为过程数据输出

☐ 作为领域对象输出

☒ 作为业务流结果输出

转换规则: 设置 清空

(8) 删除方法的出入参配置，该方法只包含fileKey作为输入参数，无返回参数

方法编辑

\*

标识

99d5f092f84641b0b310b8f410680992

\*

名称

删除

\*

业务流标识

4b4f8e9d23964ad6b4521fd8fb98ee12

\*

编码

delete

\*

状态

有效

参数

新建

| 操作 | 编码 | 名称      | 是否入参 | 是否列表 | 参数类型 |                            |
|----|----|---------|------|------|------|----------------------------|
| 编辑 | 删除 | fileKey | 文件id | 是    | 否    | Java原生类型(java.lang.String) |

入参校验规则设置

入参校验规则

选择 清空

取消

保存

## (9) 删除方法的业务逻辑配置

业务流编辑

\*

业务流标识

4b4f8e9d23964ad6b4521fd8fb98ee12

\*

业务流名称

fileKey方法的业务流

\*

加入事件

是

\*

业务流类型

复杂业务流

对应数据库

demo

业务流

从minio删除

36-Minio存取

BPMN ID

取消

保存

节点说明

从minio删除

节点类型

36-Minio存取

查看可用数据

节点处理

Minio资源

minio

Minio操作

删除文件

源文件来源

Select

源文件名来源

Select

选择

源文件名Key

源文件名属性

桶名称来源

固定值

选择

桶名称Key

demo-bucket

桶名称属性

内容类型来源

Select

选择

内容类型Key

内容类型属性

对象Key来源

输入参数

选择

对象KeyKey

fileKey

对象Key属性

节点后处理

☐

作为过程数据输出

☐

作为领域对象输出

☐

作为业务流结果输出

(10) 三个方法的对外发布信息如下：

取消发布服务方法

发布标识

ccf4290fe6ab4cf68b80727ae366d9ba

发布状态

有效

服务编码

FileService

方法编码

upload

HttpMethod

POST

请求URL

/file/upload

创建时间

Apr 16, 2025, 6:02:05 AM

更新时间

Apr 16, 2025, 6:02:05 AM

取消

确定

取消发布服务方法

发布标识

b5849c2fcc1a4a28b775cb360f0fc2a

发布状态

有效

服务编码

FileService

方法编码

download

HttpMethod

GET

请求URL

/file/download

创建时间

Apr 16, 2025, 6:05:27 AM

更新时间

Apr 16, 2025, 6:05:27 AM

取消

确定

取消发布服务方法

发布标识

a49e1f1623484f738f6f6244c952625d

发布状态

有效

服务编码

FileService

方法编码

delete

HttpMethod

POST

请求URL

/file/delete

创建时间

Apr 16, 2025, 6:13:50 AM

更新时间

Apr 16, 2025, 6:13:50 AM

取消

确定

(11) 至此文件上传、下载和删除方法已经开发完成，刷新缓存后，前端就可以直接进行调用。

## 5. 开发系统功能用于验证

(1) 开发《我的学习记录》功能，同《zPaaS低代码平台使用介绍：第一个功能开发》。

zPaaS低代码开发平台

示例系统-勿修改

平台管理、帮助

系统功能 > 系统功能 > 我的学习记录

标识

8de02ca797478bcb848b0c3738ace

名称

我的学习记录

描述

归属业务域

Studies

功能模板

简单管理功能模板

预览

保存

删除

绑定对象

新建

显示

对象绑定预览

绑定对象类型

绑定对象

绑定类型

关联属性

关联属性

主对象ID

新增

删除

3c0d86c37c449b991335bca033abd

领域对象

MyStudyRecord

主对象

是

功能区域

显示

区域标识

区域名称

区域模板

创建时间

更新时间

编辑

可视化设计

c46c70603047438883681350604282

新增区域模板

5新增区域模板

Apr 12, 2025, 6:51:13 AM

Apr 12, 2025, 6:51:13 AM

编辑

可视化设计

d446299527b4d13d79720360979c3

新增区域模板

6新增区域模板

Apr 12, 2025, 6:51:13 AM

Apr 12, 2025, 6:51:13 AM

编辑

可视化设计

30911446d128404b370b3c77b6d70d9

查询区域模板

1查询区域模板

Apr 12, 2025, 6:51:13 AM

Apr 12, 2025, 6:51:13 AM

编辑

可视化设计

6d0481a1169a48baf92a3075375c3d4

结果区域模板

2结果区域模板

Apr 12, 2025, 6:51:13 AM

Apr 12, 2025, 6:51:13 AM

编辑

可视化设计

09a0e0b064d709c6d179f4a6303a

编辑区域模板

3编辑区域模板

Apr 12, 2025, 6:51:13 AM

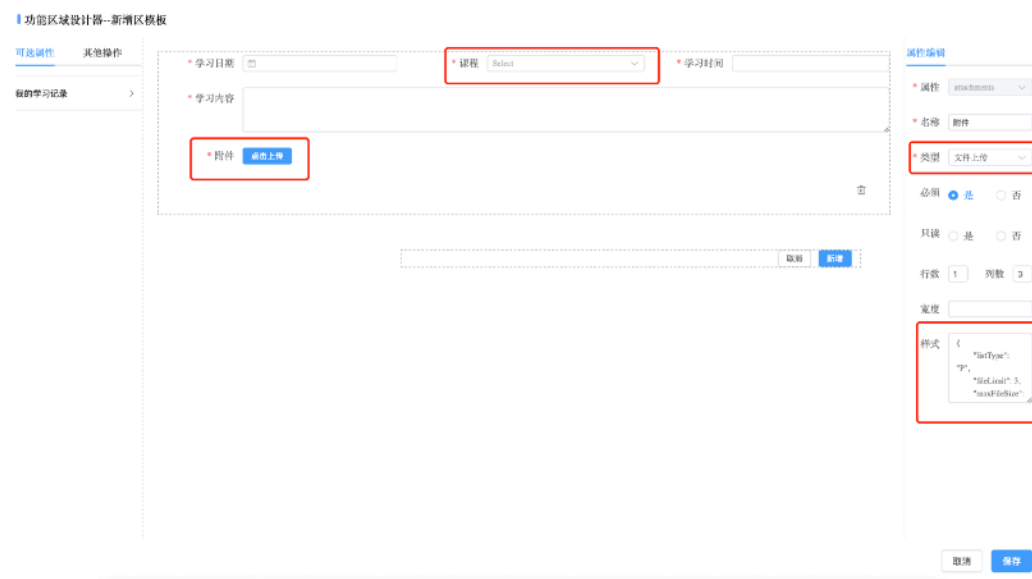
Apr 12, 2025, 6:51:13 AM

(2) 绑定对象属性选项的特殊配置：课程字段的下拉选项配置





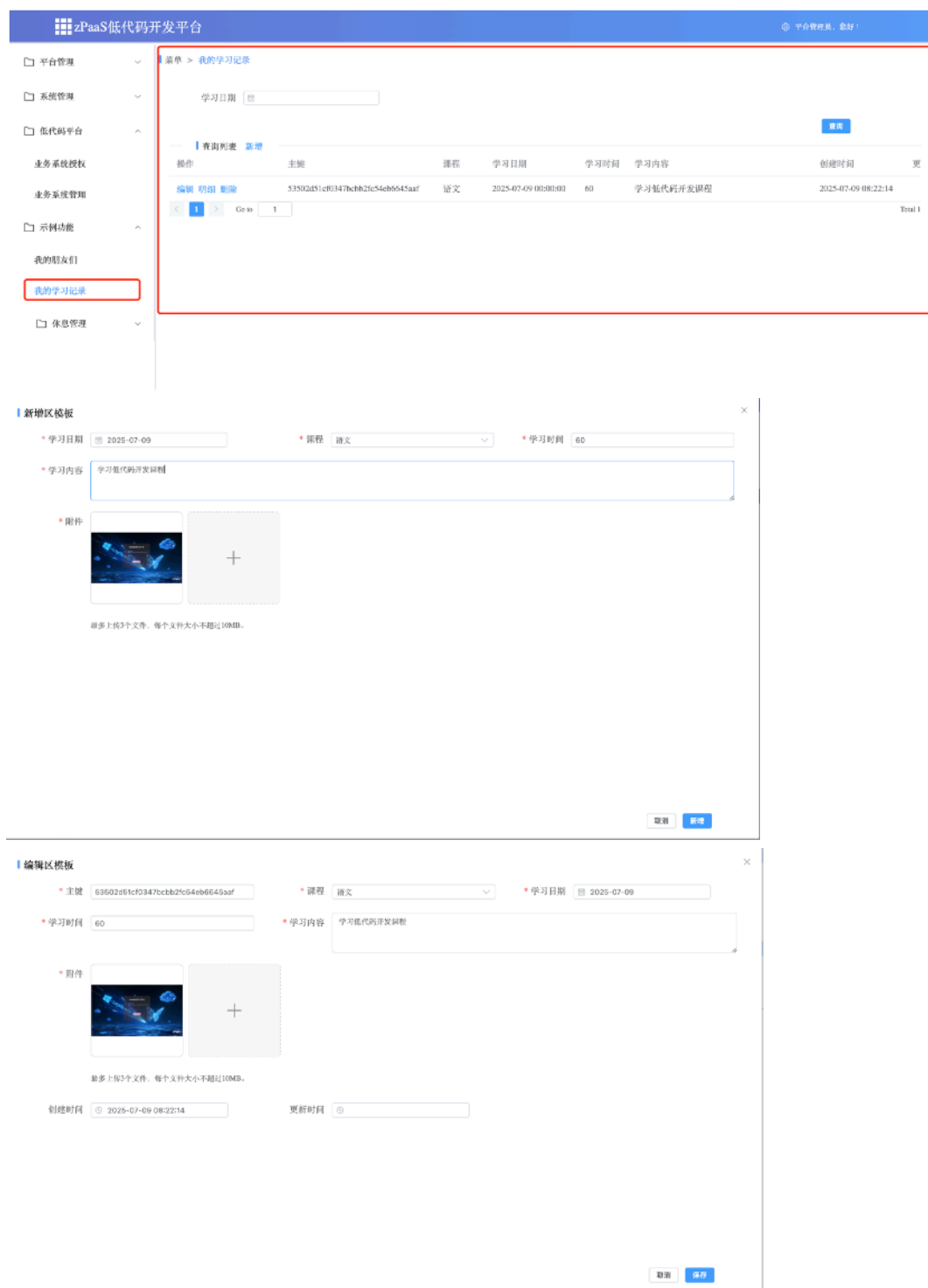
### (3) 文件上传组件的配置



样式字段的配置信息为，其中三个服务对应前一章节配置的三个文件服务的对外暴露的地址：

```
{
 "listType": "P",
 "fileLimit": 3,
 "maxFileSize": 10000000,
 "uploadTip": "最多上传3个文件，每个文件大小不超过10MB。",
 "multiple": true,
 "accept": ".jpg,.jpeg,.png",
 "uploadService": "/file/upload",
 "deleteService": "/file/delete",
 "downloadService": "/file/download"
}
```

### (4) 最终的效果见示例功能《我的学习记录》



## 6. 其他

低代码平台的源码已经提交到GitHub：<https://github.com/zjyzju/zPaaS-lowcode>，后续将通过更多的文章，逐步对低代码平台的设计理念/思想、具体的设计、部署以及操作说明进行细化，敬请关注。

另外在一台云服务器上部署了一个演示环境，有兴趣的小伙伴可以通过演示环境进行试用：

访问地址：<https://129.153.118.144:11443>

试用账户：`demo/adl28m2@ma(a!*D2`

`demo1/adl28m2@ma(a!*D2`

注：（1）该演示环境部署在一台免费的云服务器上，且位于国外，在国内访问响应会比较慢

（2）系统管理子系统的菜单框架引用的@micro-zoe/micro-app@1.0.0-rc.24组件，该组件在不同浏览器中存在兼容性问题，苹果的safari浏览器访问一切正常，但是windows的Edge和谷歌的Chrome会存在页面刷不出来的情况，需要多等一会儿（有解决方案的可以邮件发给我，多谢！<https://github.com/jd-opensource/micro-app/issues/1559#issuecomment-2784997902>）。

如果有低代码平台以及AI相关的讨论，可以发送邮件到我的邮箱：[zjyzju@163.com](mailto:zjyzju@163.com)。