

Final Project 接口文档

Last update @ 2020/5/17 14:02

修改记录

2020/5/15 00:20 最初版本

2020/5/15 00:47 加入「获取文件列表」

2020/5/17 14:02 修改「获取文件列表」，返回 `result` 中加入 key: `files`

概述

服务端返回为 JSON，返回编码 UTF-8。

以下为一成功返回示例。

```
{
  "code": 0,
  "message": "",
  "result": {
    "simple_similarity": 0.925,
    "levenshtein_distance": 18
  }
}
```

- `code` 异常代码，正常时为0。完整错误代码见后文。
- `message` 异常信息，可用于显示给用户
- `result` 正常结束时存放结果。如果没有结果，为空 dict。

规定的异常代码

见 `FailureCause` 枚举。

1. 文件不存在
2. 文件内容与 hash 值不匹配
3. 文件已存在

存在确认

请求地址: `/files/:md5/exists` (注: 所有 `:xxx` 都代表名称为 `xxx` 的 URL 参数)

请求方法: GET

URL 参数: `md5`: 要检查的文件的 MD5 hash。(注: MD5 大小写不敏感)

请求结果: `exists`: true/false, 文件是否存在

可能异常: 无。

示例:

```
{
  "code": 0,
  "message": "",
  "result": {
    "exists": true
  }
}
```

上传

请求地址: /files/:md5

请求方法: POST

URL 参数: `md5` : 要上传的文件的 MD5 hash

POST Body: 要上传的文件的二进制数据 (以 UTF-8 编码)

请求结果: 成功时, `result` 中 `success` 为 `true`, 失败时 `result` 中 `success` 为 `false`, `code` 和 `message` 指明失败原因。

可能异常: 文件已存在、hash 与文件内容不符合

注

- 上传前需要将文件在客户端转换为 UTF-8, 并在 UTF-8 编码下计算 md5 值
- 一次调用只能上传一个文件

成功返回示例

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

失败返回示例

```
{
  "code": 3,
  "message": "File with the same MD5 already exists",
  "result": {
    "success": false
  }
}
```

比较

请求地址: /files/:md51/compare/:md52

请求方法: GET

URL 参数: `md51`, `md52` 第二个比较文件的 MD5 hash

请求结果

- `simple_similarity`: 简单相似度, 两个文件相同位置上字符相同的位置个数 / 两个文件中最长文件的长度, **精确至 3 位小数**
例: `str1 = aaa, str2 = aba, simple_similarity = 2/3 = 0.666`
- `levenshtein_distance`: 编辑距离, 见 [wiki](#), 应该为整数值

可能异常: 文件不存在

成功返回示例

```
{
  "code": 0,
  "message": "",
  "result": {
    "simple_similarity": 0.925,
    "levenshtein_distance": 18
  }
}
```

下载

请求地址: `/files/:md5`

请求方法: GET

URL 参数: `md5`: 要下载的文件 MD5 hash

请求结果:

- 文件存在时, `result` 中 `content` 为文件内容
- 文件不存在时, `code`, `message` 为异常信息

可能异常: 文件不存在

注: 下载时直接将 `content` 内字符串以 UTF-8 编码写入文件

成功返回示例

```
{
  "code": 0,
  "message": "",
  "result": {
    "content": "这是一段测试文本: 我能吞下玻璃而不伤身体"
  }
}
```

获取文件列表

请求地址: `/files`

请求方法: GET

URL 参数: 无

请求结果: `result` 中 `files` 内为文件摘要列表, 每个文件摘要包括 `md5` (文件的md5值), `length` (文件长度, 按照字符数计算) 及 `preview` (前100个字符, 如不足100字符长则为完整内容)

注：可以使用以下方法放入 `List` 至 `ObjectNode` 中

```
response.getResult().set("files", new ObjectMapper().valueToTree(list));
```

成功返回示例

```
{  
    "code": 0,  
    "message": "",  
    "result": {  
        "files": [  
            {  
                "md5": "fac27083ec676a0999e6d22e4c40f31a",  
                "length": 20,  
                "preview": "这是一段测试文本：我能吞下玻璃而不伤身体"  
            },  
            {  
                "md5": "aa1b5fccb80521924433a5cea41dce3b",  
                "length": 1000,  
                "preview": "这是一段超长文本！！这是一段超长文本！！这是一段超长文本！！这是一段超长文本！！这是一段超长文本！！这是一段超长文本！！这是一段超长文本！！这是一段超长文本！！这是一段超长文本！！"  
            }  
        ]  
    }  
}
```