|  | 文档编号 |  | 版本 | V0.1 | 密级 |  |
| --- | --- | --- | --- | --- | --- | --- |
| 项目名称 | 东方证券DevOps平台详细设计 | | | | |
| 项目来源 |  | | | | |

**东方证券DevOps-Jira模块详细设计说明书**

文档变更记录

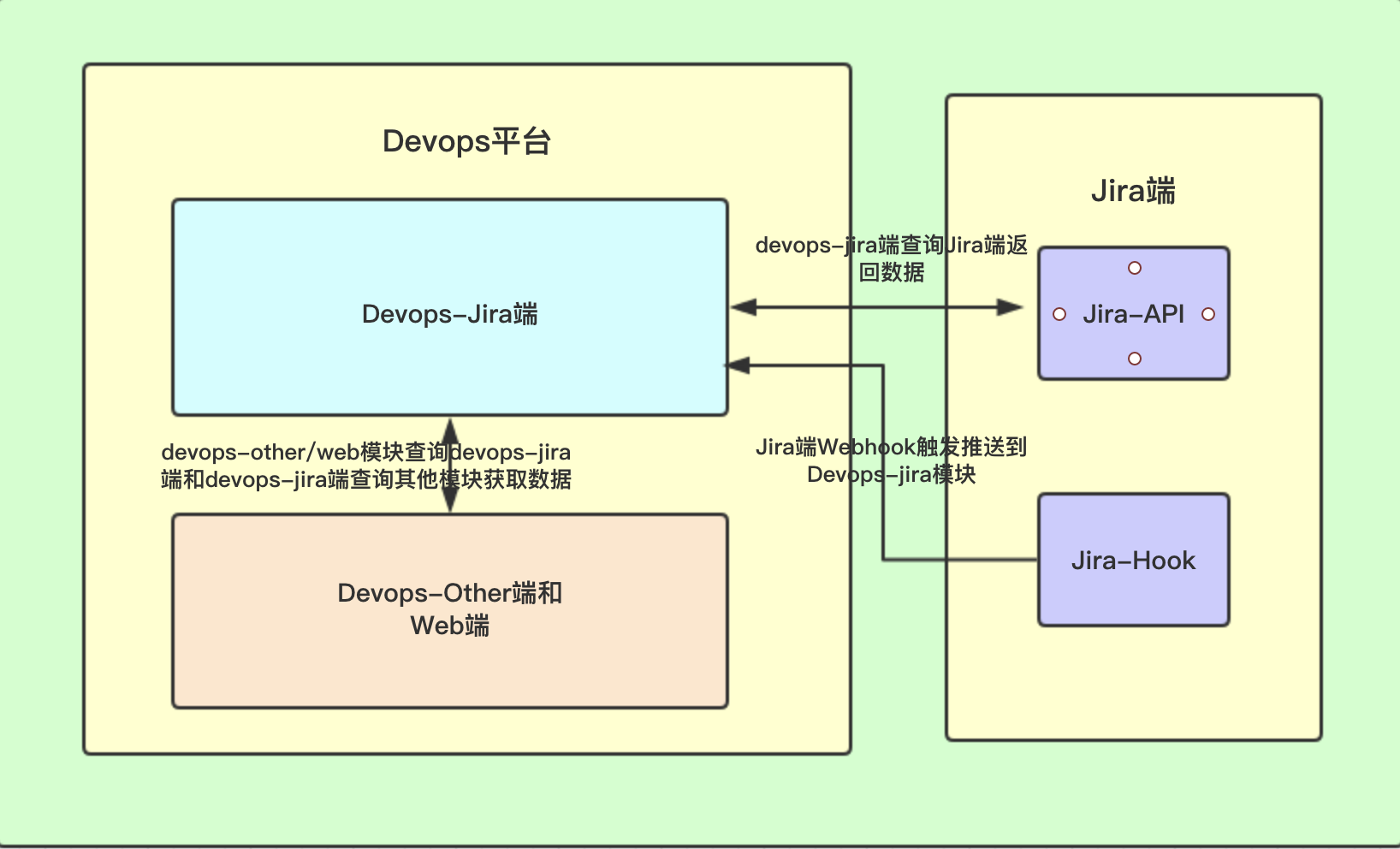
| 版本 | 日期 | 拟稿和修改 | 作者 | 说明 |
| --- | --- | --- | --- | --- |
| V0.1 | 2019.11.13 | 创建 | 丁正刚 | 创建 |

**DevOps-Jira模块详细设计**

# **DevOps-Jira**模块总览

## 功能描述

DevOps-Jira模块中工作主要涉及两部分：查询外部Jira端接口和外部Jira端反馈到Devops-Jira模块。查询外部Jira端口主要是查询Jira暴露出来的接口，这些接口有原生的和插件提供；外部Jira反馈，是Jira端配置相关Webhook，设定触发操作，达到条件即可返回到Devops-jira端。



## 功能概览

### 基本描述

Devops-jira模块功能上基本覆盖执行Jira内部指令、查询各种需求下Issue信息、查询项目信息、迭代信息、版本信息和相关用户信息。

### 表字段

**1. 数据表名：jira\_boost\_history 注释： Jira推进记录表**

**Jira端的Webhook触发操作，会推送到devops-jira 端,每个项目应用版本下的第一次推送会被记录，下次就不会推进该版本的工作流。**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 数据类型 | 注释 | 类型 |
| id | int(11) | 记录id | 主键 |
| project\_id | int(11) | jira项目id |  |
| application\_id | int(11) | jira应用标签id |  |
| version\_id | int(11) | jira版本id |  |
| create\_time | datetime | 创建推进记录时间 |  |

**2. 数据表名：jira\_version\_bug 注释： Jira个版本缺陷每日记录表**

**定时任务每天查询全部应用下全部版本的每日缺陷完成情况，添加情况和剩余量情况，该表做此记录。**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 数据类型 | 注释 | 类型 |
| id | int(11) | 记录id | 主键 |
| application\_id | int(11) | devops应用标签id |  |
| jira\_project\_id | int(11) | jira项目id |  |
| jira\_application\_id | int(11) | jira应用id |  |
| jira\_version\_id | int(11) | jira版本id |  |
| create\_num | int(11) | 创建的数量 |  |
| solve\_num | int(11) | 解决的数量 |  |
| no\_solve\_num | int(11) | 未解决的数量 |  |
| date\_sort | varchar(10) | 插入日期 |  |
| create\_time | datetime | 创建记录时间 |  |

## **接口功能**

### 项目/应用/服务相关接口

#### 应用服务标签，增删改

应用服务标签为Jira自定义字段，通过Jira界面操作绑定于项目之下，通过逻辑操作，是项目-应用-服务成为一个整体，该接口可以对应用下的服务进行增删改。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作应用服务标签，增删改 | | | | |
| 接口描述 | 操作应用服务标签，增删改 | | | |
| 接口备注 | 应用服务的实体类 | | | |
| URL | /project/applicationService/handle | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | ProjectBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"appOrPro":"应用类型 基于Project的为project 基于Project下的Application标签的为application","applicationId":"项目下应用标签Id","applicationName":"项目下应用标签名称","description":"项目描述","eventType":"操作类型 created/updated/deleted","id":"项目Id","key":"项目关键词Key","leadName":"项目负责人","name":"项目名称","projectId":"Jira项目Id","projectName":"Jira项目名称","projectTypeKey":"项目类型(business|software|service\_desk)","serviceId":"项目下应用标签下服务标签的Id","serviceName":"项目下应用标签下服务标签的名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 查询所有项目

该接口获取指定Jira端内的所有可用的项目，从而获取项目Id，为平台导入应用服务。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 查询所有项目 | | | | |
| 接口描述 | 查询所有项目 | | | |
| 接口备注 | null | | | |
| URL | /project/getAllProjects | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 操作项目的添加/修改/删除

该接口为项目的增删改，一般是不需要的。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作项目项,目添加/修改/删除 | | | | |
| 接口描述 | 操作项目项,目添加/修改/删除 | | | |
| 接口备注 | 项目的实体类 | | | |
| URL | /project/handle | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | ProjectBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"appOrPro":"应用类型 基于Project的为project 基于Project下的Application标签的为application","applicationId":"项目下应用标签Id","applicationName":"项目下应用标签名称","description":"项目描述","eventType":"操作类型 created/updated/deleted","id":"项目Id","key":"项目关键词Key","leadName":"项目负责人","name":"项目名称","projectId":"Jira项目Id","projectName":"Jira项目名称","projectTypeKey":"项目类型(business|software|service\_desk)","serviceId":"项目下应用标签下服务标签的Id","serviceName":"项目下应用标签下服务标签的名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 获取应用系统的Issue完成情况

该接口获取应用下的issue完成情况，使用Jql语句查询应用下所有的Issue（故事，缺陷，子任务）的完成情况然后进行整合，缺陷和子任务使用的是Issue下的关联的子任务和缺陷。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 获取应用系统的Issue完成情况 | | | | |
| 接口描述 | 获取应用系统的Issue完成情况 | | | |
| 接口备注 | 项目id,应用id | | | |
| URL | /project/qualityApplication | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | ProjectBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"appOrPro":"应用类型 基于Project的为project 基于Project下的Application标签的为application","applicationId":"项目下应用标签Id","applicationName":"项目下应用标签名称","description":"项目描述","eventType":"操作类型 created/updated/deleted","id":"项目Id","key":"项目关键词Key","leadName":"项目负责人","name":"项目名称","projectId":"Jira项目Id","projectName":"Jira项目名称","projectTypeKey":"项目类型(business|software|service\_desk)","serviceId":"项目下应用标签下服务标签的Id","serviceName":"项目下应用标签下服务标签的名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 单个项目信息查询

从Jira端获取指定项目的信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 单个项目信息查询 | | | | |
| 接口描述 | 单个项目信息查询 | | | |
| 接口备注 | code:200 | | | |
| URL | /project/{projectId} | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| projectId | integer | path | Y | projectId |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"projectId":0} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

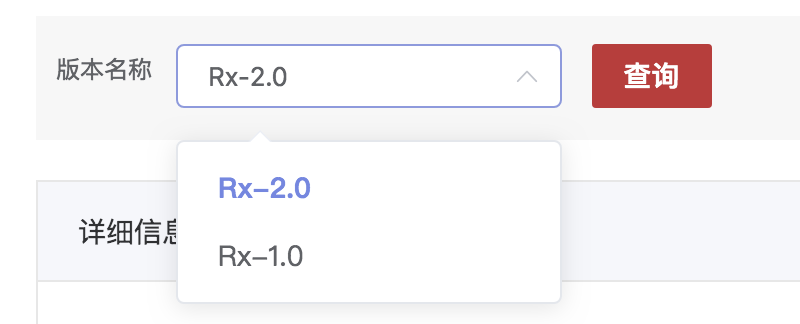
#### 单个项目应用标签查询

从Jira端中查询指定应用标签的信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 单个项目应用标签查询 | | | | |
| 接口描述 | 单个项目应用标签查询 | | | |
| 接口备注 | code:200 | | | |
| URL | /project/{projectId}/application | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| projectId | string | path | Y | projectId |
| entity | body | ProjectBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"appOrPro":"应用类型 基于Project的为project 基于Project下的Application标签的为application","applicationId":"项目下应用标签Id","applicationName":"项目下应用标签名称","description":"项目描述","eventType":"操作类型 created/updated/deleted","id":"项目Id","key":"项目关键词Key","leadName":"项目负责人","name":"项目名称","projectId":"Jira项目Id","projectName":"Jira项目名称","projectTypeKey":"项目类型(business|software|service\_desk)","serviceId":"项目下应用标签下服务标签的Id","serviceName":"项目下应用标签下服务标签的名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

### 应用版本相关接口

#### 查询指定项目下的所有版本



该接口可以查询指定项目下的版本列表信息，以供其他模块查询。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 查询指定项目下的所有版本 | | | | |
| 接口描述 | 查询指定项目下的所有版本 | | | |
| 接口备注 | 需要指定项目应用信息 | | | |
| URL | /version/getAllVersions | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | VersionBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用ID格式为(int32)","applicationName":"应用ID","archived":"是否归档","description":"版本描述","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endTime":"查询结束时间戳格式为(int64)","endTimeStr":"结束查询日期 2019-07-14","eventType":"操作类型 created/updated/deleted","id":"版本ID","jiraApplicationId":"对应的Jira应用ID格式为(int32)","name":"版本名称","overdue":"版本是否延期","projectId":"对应的Jira的版本对应的项目ID格式为(int32)","releaseDate":"计划发布时间格式为(date-time)","releaseDateStr":"计划发布时间字符串","released":"是否发布","self":"查询URL","startDate":"计划开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","startTimeStr":"开始查询日期 2019-07-13","userReleaseDate":"","userStartDate":""} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 单个版本信息查询

获取单个版本的详情信息。

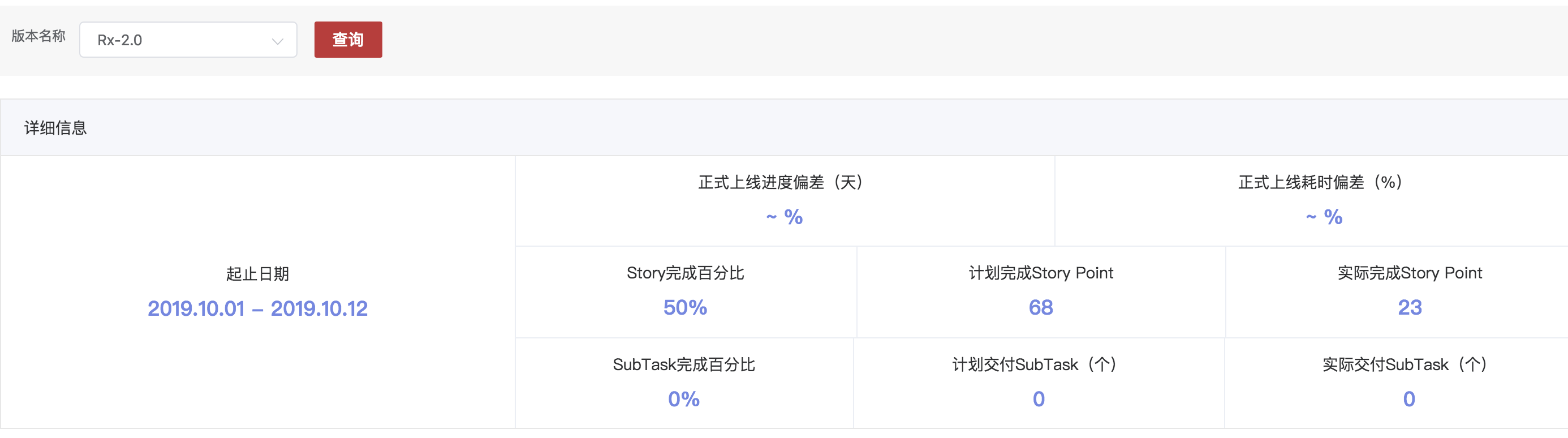
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 单个版本信息查询 | | | | |
| 接口描述 | 单个版本信息查询 | | | |
| 接口备注 | 获取单个版本的信息，传版本ID | | | |
| URL | /version/{versionId} | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| versionId | integer | path | Y | versionId |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"versionId":0} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 项目版本添加/修改/删除

该接口为操作项目下版本的创建修改和删除，因为版本不和应用挂钩，所以操作版本时，需要携带应用的项目信息，创建的版本名称会使用应用和应用Id标识。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作项目 | | | | |
| 接口描述 | 操作项目版本 | | | |
| 接口备注 | 创建/更新/删除 | | | |
| URL | /version/handle | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | VersionBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用ID格式为(int32)","applicationName":"应用ID","archived":"是否归档","description":"版本描述","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endTime":"查询结束时间戳格式为(int64)","endTimeStr":"结束查询日期 2019-07-14","eventType":"操作类型 created/updated/deleted","id":"版本ID","jiraApplicationId":"对应的Jira应用ID格式为(int32)","name":"版本名称","overdue":"版本是否延期","projectId":"对应的Jira的版本对应的项目ID格式为(int32)","releaseDate":"计划发布时间格式为(date-time)","releaseDateStr":"计划发布时间字符串","released":"是否发布","self":"查询URL","startDate":"计划开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","startTimeStr":"开始查询日期 2019-07-13","userReleaseDate":"","userStartDate":""} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 获取度量信息中质量的版本的详细信息



该接口获取指定版本的详情信息，如故事完成率完成量，子任务完成率完成量，缺陷的解决率等信息。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 获取度量信息中质量的版本的详细信息 | | | | |
| 接口描述 | 获取度量信息中质量的版本的详细信息 | | | |
| 接口备注 | 项目id,应用id,版本id | | | |
| URL | /version/qualityVersionDetail | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | VersionBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用ID格式为(int32)","applicationName":"应用ID","archived":"是否归档","description":"版本描述","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endTime":"查询结束时间戳格式为(int64)","endTimeStr":"结束查询日期 2019-07-14","eventType":"操作类型 created/updated/deleted","id":"版本ID","jiraApplicationId":"对应的Jira应用ID格式为(int32)","name":"版本名称","overdue":"版本是否延期","projectId":"对应的Jira的版本对应的项目ID格式为(int32)","releaseDate":"计划发布时间格式为(date-time)","releaseDateStr":"计划发布时间字符串","released":"是否发布","self":"查询URL","startDate":"计划开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","startTimeStr":"开始查询日期 2019-07-13","userReleaseDate":"","userStartDate":""} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### **获取度量信息中质量的迭代的详细信息**

该接口获取指定迭代的详情信息，如故事完成率完成量，子任务完成率完成量，缺陷的解决率等信息。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 获取度量信息中质量的迭代的详细信息 | | | | |
| 接口描述 | 获取度量信息中质量的迭代的详细信息 | | | |
| 接口备注 | 项目id,应用id,版本id | | | |
| URL | /version/qualitySprintDetail | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | SprintBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用Id","completeDate":"Sprint实际完成时间格式为(date-time)","completeDateStr":"","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endDate":"Sprint计划结束时间格式为(date-time)","endDateStr":"计划发布时间字符串","endTime":"查询结束时间戳格式为(int64)","id":"SprintId","name":"Sprint名称","projectId":"项目Id","projectIds":"Sprint体系内规划项目ID们，多个用,隔开","rapidViewId":"Sprint的面板Id","sprintSelectName":"Sprint燃耗图查询参数","startDate":"Sprint最初开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","state":"Sprint的状态 ：刚创建 future 还未开始；active:正在进行的；closed:已经关闭的","versionId":"版本Id","versionName":"版本名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

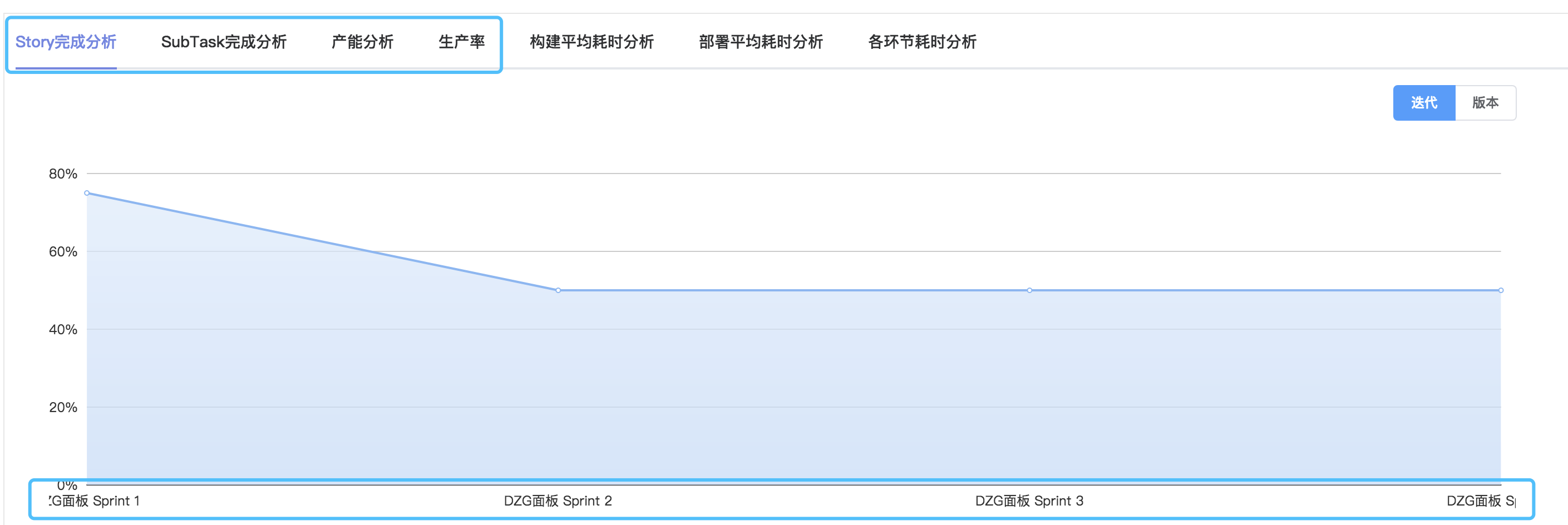
#### **根据时间范围查询应用下的版本列表信息**



**在给定的世界范围内查询该应用的版本列表信息（从Jira中查询实际版本）**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 根据时间范围查询应用下的版本列表信息 | | | | |
| 接口描述 | 根据时间范围查询应用下的版本列表信息 | | | |
| 接口备注 | 项目id,应用id,时间范围 | | | |
| URL | /version/getVersionsByAppInTime | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | VersionBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用ID格式为(int32)","applicationName":"应用ID","archived":"是否归档","description":"版本描述","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endTime":"查询结束时间戳格式为(int64)","endTimeStr":"结束查询日期 2019-07-14","eventType":"操作类型 created/updated/deleted","id":"版本ID","jiraApplicationId":"对应的Jira应用ID格式为(int32)","name":"版本名称","overdue":"版本是否延期","projectId":"对应的Jira的版本对应的项目ID格式为(int32)","releaseDate":"计划发布时间格式为(date-time)","releaseDateStr":"计划发布时间字符串","released":"是否发布","self":"查询URL","startDate":"计划开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","startTimeStr":"开始查询日期 2019-07-13","userReleaseDate":"","userStartDate":""} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

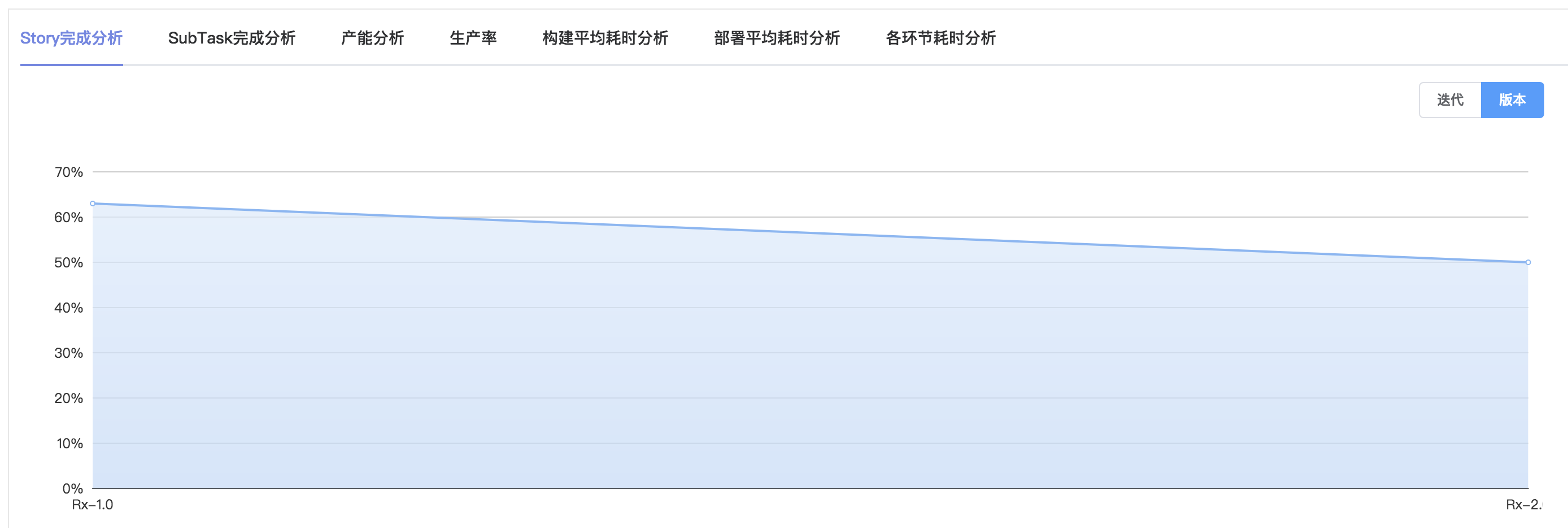
#### 获取当前应用所有迭代的效率分析图表



该接口获取的是指定应用下的所有迭代的StoryPoints的完成分析表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 获取当前应用所有迭代的效率分析图表 | | | | |
| 接口描述 | 获取当前应用所有迭代的效率分析图表 | | | |
| 接口备注 | Jira应用Id,DevOps应用Id,图表类型 | | | |
| URL | /version/getAllSprintsReportInType | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | SprintBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用Id","completeDate":"Sprint实际完成时间格式为(date-time)","completeDateStr":"","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endDate":"Sprint计划结束时间格式为(date-time)","endDateStr":"计划发布时间字符串","endTime":"查询结束时间戳格式为(int64)","id":"SprintId","name":"Sprint名称","projectId":"项目Id","projectIds":"Sprint体系内规划项目ID们，多个用,隔开","rapidViewId":"Sprint的面板Id","sprintSelectName":"Sprint燃耗图查询参数","startDate":"Sprint最初开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","state":"Sprint的状态 ：刚创建 future 还未开始；active:正在进行的；closed:已经关闭的","versionId":"版本Id","versionName":"版本名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 获取当前应用所有版本的效率分析图表



该接口获取的是指定应用下的所有版本的StoryPoints的完成分析表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 获取当前应用所有版本的效率分析图表 | | | | |
| 接口描述 | 获取当前应用所有版本的效率分析图表 | | | |
| 接口备注 | Jira应用Id，DevOps应用Id,图表类型 | | | |
| URL | /version/getAllVersionsReportInType | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | VersionBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用ID格式为(int32)","applicationName":"应用ID","archived":"是否归档","description":"版本描述","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endTime":"查询结束时间戳格式为(int64)","endTimeStr":"结束查询日期 2019-07-14","eventType":"操作类型 created/updated/deleted","id":"版本ID","jiraApplicationId":"对应的Jira应用ID格式为(int32)","name":"版本名称","overdue":"版本是否延期","projectId":"对应的Jira的版本对应的项目ID格式为(int32)","releaseDate":"计划发布时间格式为(date-time)","releaseDateStr":"计划发布时间字符串","released":"是否发布","self":"查询URL","startDate":"计划开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","startTimeStr":"开始查询日期 2019-07-13","userReleaseDate":"","userStartDate":""} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

### 迭代Sprint相关接口

#### Sprint创建/开始/结束接口

该接口可以控制Jira端的迭代Sprint创建/开始/结束。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sprint创建/开始/结束 | | | | |
| 接口描述 | Sprint创建/开始/结束 | | | |
| 接口备注 | code:200 | | | |
| URL | /sprint/handle | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | SprintBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用Id","completeDate":"Sprint实际完成时间格式为(date-time)","completeDateStr":"","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endDate":"Sprint计划结束时间格式为(date-time)","endDateStr":"计划发布时间字符串","endTime":"查询结束时间戳格式为(int64)","id":"SprintId","name":"Sprint名称","projectId":"项目Id","projectIds":"Sprint体系内规划项目ID们，多个用,隔开","rapidViewId":"Sprint的面板Id","sprintSelectName":"Sprint燃耗图查询参数","startDate":"Sprint最初开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","state":"Sprint的状态 ：刚创建 future 还未开始；active:正在进行的；closed:已经关闭的","versionId":"版本Id","versionName":"版本名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

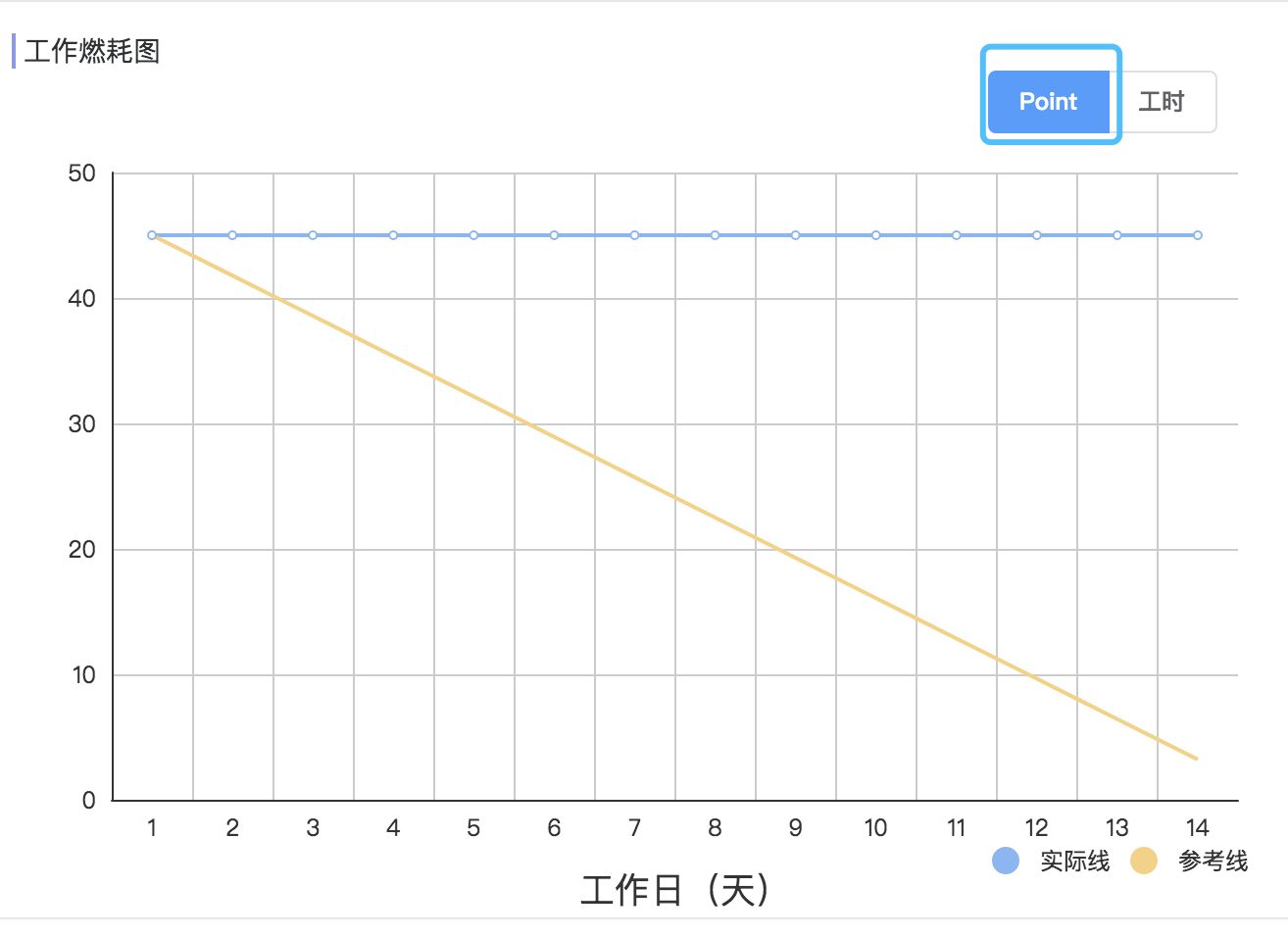
#### 基于版本查询当前应用内的Issue内覆盖的Sprint信息



根据项目和应用ID使用Jql语句查询出所有相关Issue，根据Issue中的Sprint信息进行整合出的迭代列表信息，因为Sprint是一个全局概念。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 基于版本查询当前版本内的Issue内覆盖的Sprint信息 | | | | |
| 接口描述 | 基于版本查询当前版本内的Issue内覆盖的Sprint信息 | | | |
| 接口备注 | 传Jira应用ID和项目Id | | | |
| URL | /sprint/getAllSprintsIssuesTime | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| sprintBaseEntity | body | SprintBaseEntity | Y | sprintBaseEntity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用Id","completeDate":"Sprint实际完成时间格式为(date-time)","completeDateStr":"","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endDate":"Sprint计划结束时间格式为(date-time)","endDateStr":"计划发布时间字符串","endTime":"查询结束时间戳格式为(int64)","id":"SprintId","name":"Sprint名称","projectId":"项目Id","projectIds":"Sprint体系内规划项目ID们，多个用,隔开","rapidViewId":"Sprint的面板Id","sprintSelectName":"Sprint燃耗图查询参数","startDate":"Sprint最初开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","state":"Sprint的状态 ：刚创建 future 还未开始；active:正在进行的；closed:已经关闭的","versionId":"版本Id","versionName":"版本名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

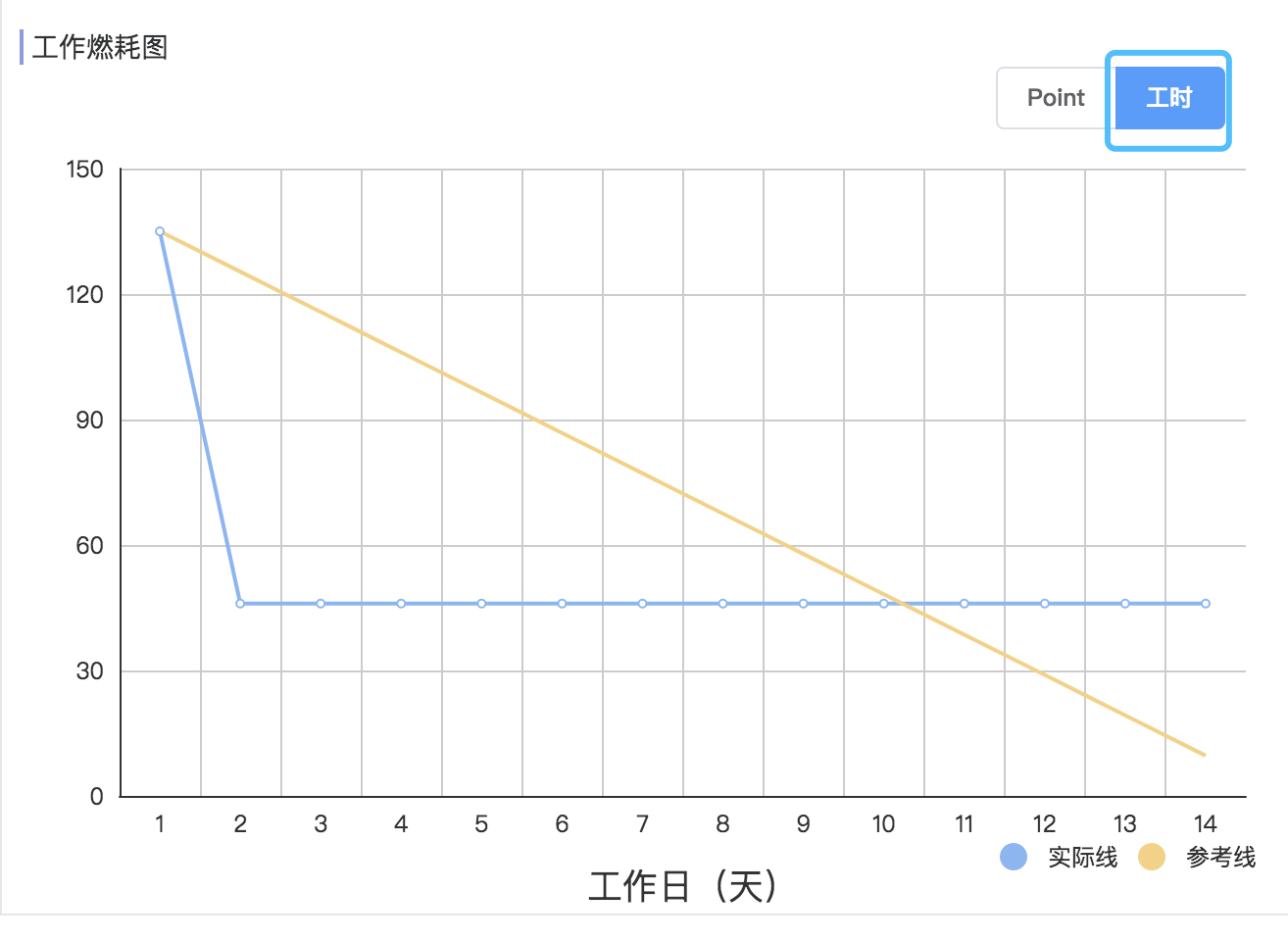
#### 查询Sprint的StoryPoint燃耗图工作量



该接口使用Jira本身的查询接口查询得出迭代在一个周期内的所有操作，然后进行整合处理出每天StoryPoints的完成情况图。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 查询Sprint的StoryPoint燃耗图工作量 | | | | |
| 接口描述 | 查询Sprint的StoryPoint燃耗图工作量 | | | |
| 接口备注 | 传Jira的迭代Id | | | |
| URL | /sprint/getSprintIssueStoryPoint/{sprintId} | | | |
| 请求方式 | get | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| sprintId | string | path | Y | sprintId |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"sprintId":"string"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

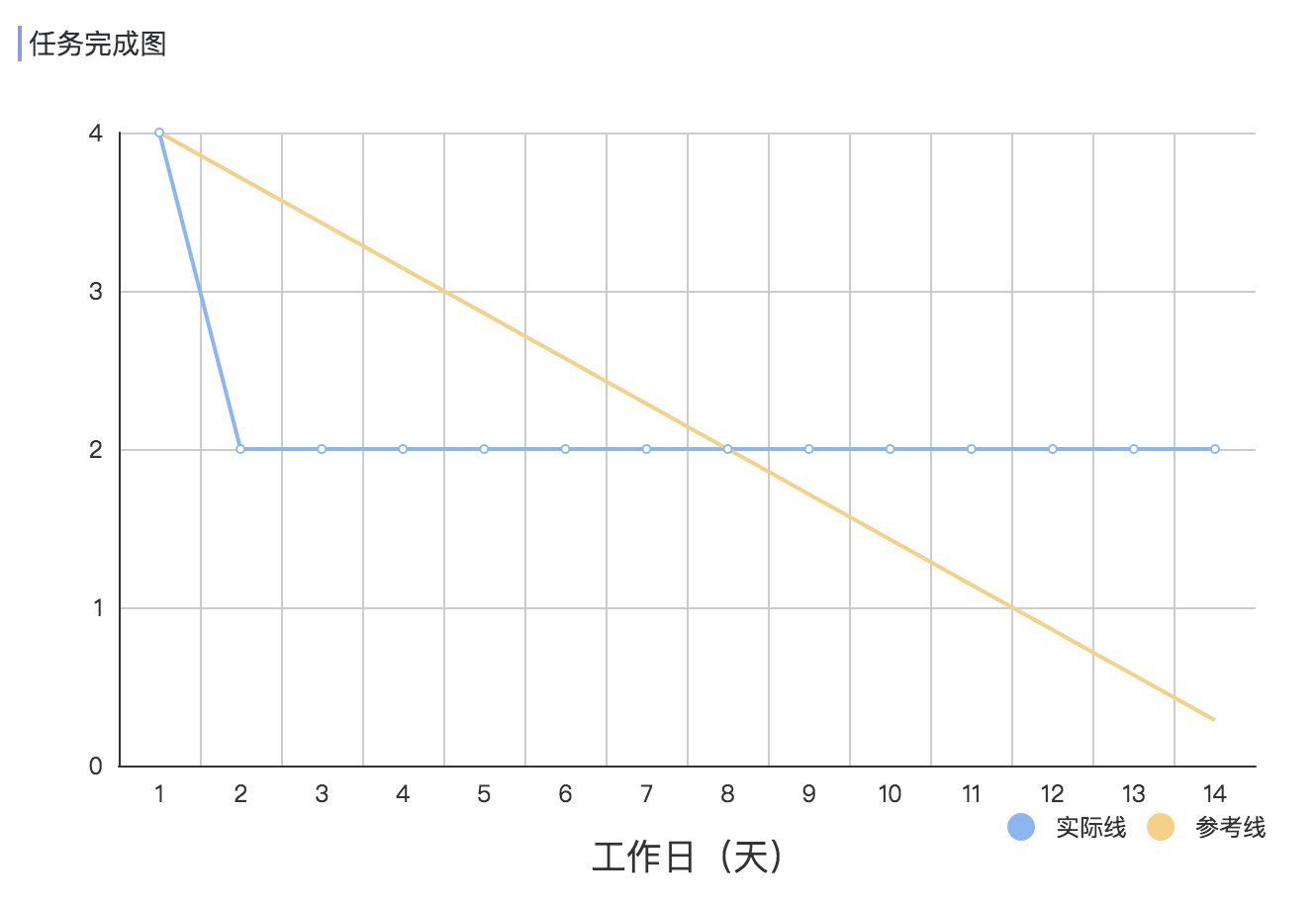
#### 查询Sprint的Issue耗时燃耗图工作量



该接口使用Jira本身的查询接口查询得出迭代在一个周期内的所有操作，然后进行整合处理出每天Issue耗时的情况图。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 查询Sprint的Issue耗时燃耗图工作量 | | | | |
| 接口描述 | 查询Sprint的Issue耗时燃耗图工作量 | | | |
| 接口备注 | 传Jira的迭代Id | | | |
| URL | /sprint/getSprintIssueTime/{sprintId} | | | |
| 请求方式 | get | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| sprintId | string | path | Y | sprintId |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"sprintId":"string"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 查询Sprint的Issue数量燃耗图工作量



该接口使用Jira本身的查询接口查询得出迭代在一个周期内的所有操作，然后进行整合处理出每天Issue完成数量的情况图。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 查询Sprint的Issue耗时燃耗图工作量 | | | | |
| 接口描述 | 查询Sprint的Issue耗时燃耗图工作量 | | | |
| 接口备注 | 传Jira的迭代Id | | | |
| URL | /sprint/getSprintIssueTime/{versionId}/{sprintId} | | | |
| 请求方式 | get | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| sprintId | string | path | Y | sprintId |
| versionId | string | path | Y | versionId |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"sprintId":"string","versionId":"string"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

### Issue相关接口

#### 查询条件下的所有Issue，根据Jql查询

JQL语句生成是查询Jira中Issue的重点问题，Jql语句类似Mysql使用各个标识来查询过滤Jira的Issue的列表值。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 查询条件下的所有Issue，根据Jql查询 | | | | |
| 接口描述 | 查询条件下的所有Issue，根据Jql查询 | | | |
| 接口备注 | 传入指定的Jql实体类 | | | |
| URL | /issue/getAllIssuesByJql | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | JqlEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用id","applicationName":"应用名称","applicationServiceId":"应用下服务id","applicationServiceName":"应用下服务名称","bugEnvId":"Issue发现环境Id","bugEnvName":"Issue发现环境名称","endTimeStr":"结束查询日期 2019-07-14","isLevel":"查询结果是否为层级结构 是 true 否 false","issueId":"Issue的Id,查询时可以多个,以,隔开","issueKey":"Issue的key","issueTypeId":"Issue类型的id","issueTypeName":"Issue类型的名称","notStatusId":"非Issue状态 0 代办 1 开发中 2 开发集成 3 待测试 4 SIT测试中 5 SIT测试完成 6 UAT测试中 7 UAT测试完成 8 待发布 9 已发布","notStatusName":"非Issue状态 0 代办 1 开发中 2 开发集成 3 待测试 4 SIT测试中 5 SIT测试完成 6 UAT测试中 7 UAT测试完成 8 待发布 9 已发布","priorityId":"Issue优先级Id","priorityName":"Issue优先级名称","projectId":"项目id","projectKey":"项目key","resolutionEndDateStr":"解决查询结束日期 2019-07-14","resolutionStartDateStr":"解决查询开始日期 2019-07-14","sprintId":"Sprint的id","sprintName":"Sprint名称","startTimeStr":"开始查询日期 2019-07-13","statusId":"Issue状态 0 代办 1 开发中 2 开发集成 3 待测试 4 SIT测试中 5 SIT测试完成 6 UAT测试中 7 UAT测试完成 8 待发布 9 已发布","statusName":"Issue状态 0 代办 1 开发中 2 开发集成 3 待测试 4 SIT测试中 5 SIT测试完成 6 UAT测试中 7 UAT测试完成 8 待发布 9 已发布","versionId":"版本id","versionName":"版本名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 获得单个Issue的详细详细

从Jira中获取单个Issue的详情信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 单个Issue的详细信息 | | | | |
| 接口描述 | 单个Issue的详细信息 | | | |
| 接口备注 | 直接传指定的IssueId | | | |
| URL | /issue/{issueId} | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| issueId | string | path | Y | issueId |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"issueId":"string"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 按照缺陷等级查询缺陷列表



Jql查询出相关的Issue列表，根据缺陷的等级分类。

缺陷等级分布图

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 缺陷等级分布图 | | | | |
| 接口描述 | 缺陷等级分布图 | | | |
| 接口备注 | 需要版本Id | | | |
| URL | /issue/getBugByLevel | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | BugBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"action":"故障的状态名称","bugFrom":"故障的来源，1.SIT环境 SIT 2.UAT环境 UAT 3.生产环境 PRO 4.其他环境 OTHER格式为(int32)","bugFromType":"故障的来源，SIT UAT PRO OTHER","bugId":"故障的Id值","key":"故障的key值","name":"故障的名称","priority":"故障的缺陷等级 1.Trivial 2.Minor 3.Major 4.Critical 5.Blocker","reOpenStatus":"故障的重新打开状态，1.未被重新打开 2.被重新打开过格式为(int32)","solveStatus":"故障的解决状态，1.解决 2.未解决格式为(int32)","versionId":"故障查询的版本Id"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

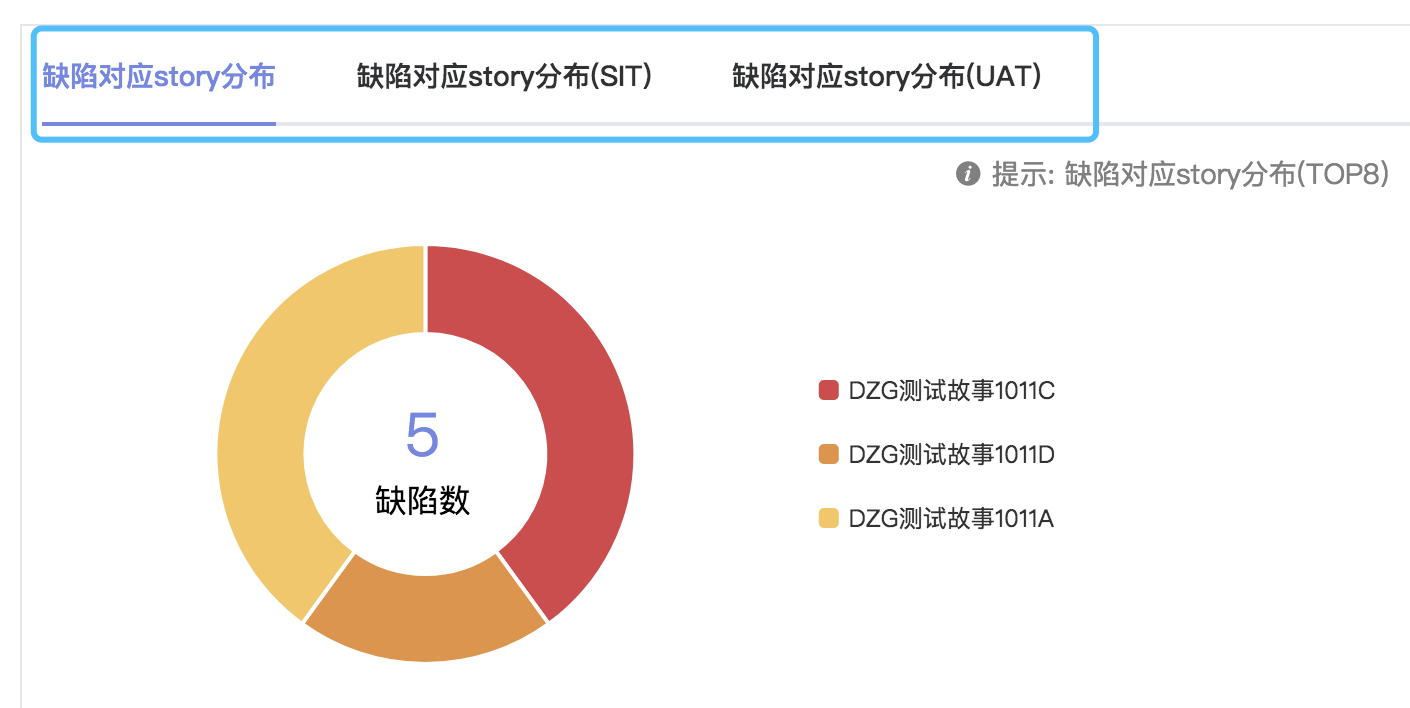
#### 根据缺陷重新打开与否查询缺陷列表



Jql查询出相关的Issue列表，根据缺陷是否重新打开进行分类。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 获取解决的缺陷中重新打开的 | | | | |
| 接口描述 | 获取解决的缺陷中重新打开的 | | | |
| 接口备注 | 需要版本Id | | | |
| URL | /issue/getBugByReOpen | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | BugBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"action":"故障的状态名称","bugFrom":"故障的来源，1.SIT环境 SIT 2.UAT环境 UAT 3.生产环境 PRO 4.其他环境 OTHER格式为(int32)","bugFromType":"故障的来源，SIT UAT PRO OTHER","bugId":"故障的Id值","key":"故障的key值","name":"故障的名称","priority":"故障的缺陷等级 1.Trivial 2.Minor 3.Major 4.Critical 5.Blocker","reOpenStatus":"故障的重新打开状态，1.未被重新打开 2.被重新打开过格式为(int32)","solveStatus":"故障的解决状态，1.解决 2.未解决格式为(int32)","versionId":"故障查询的版本Id"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

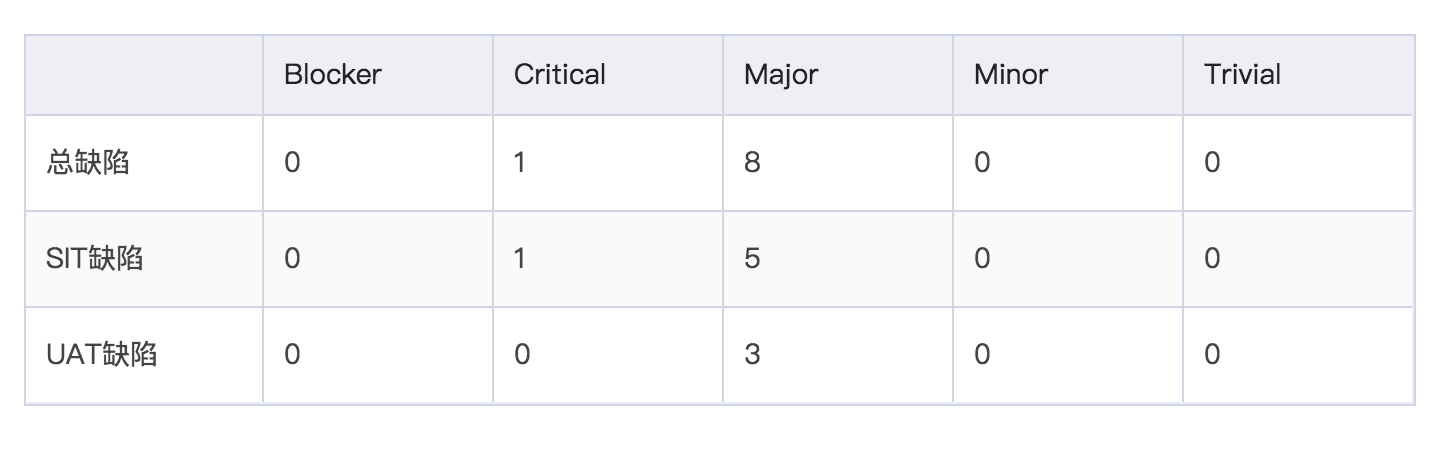
#### 查询缺陷含量最大的8个Issue图表



Jql查询出相关的Issue列表，根据缺陷数量从而对Issue进行分类的功能。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 查询本版本排名前八的 Issue含缺量 | | | | |
| 接口描述 | 查询本版本排名前八的 Issue含缺量 | | | |
| 接口备注 | 需要版本Id和缺陷类型 | | | |
| URL | /issue/getBugInStoryTop | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | BugBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"action":"故障的状态名称","bugFrom":"故障的来源，1.SIT环境 SIT 2.UAT环境 UAT 3.生产环境 PRO 4.其他环境 OTHER格式为(int32)","bugFromType":"故障的来源，SIT UAT PRO OTHER","bugId":"故障的Id值","key":"故障的key值","name":"故障的名称","priority":"故障的缺陷等级 1.Trivial 2.Minor 3.Major 4.Critical 5.Blocker","reOpenStatus":"故障的重新打开状态，1.未被重新打开 2.被重新打开过格式为(int32)","solveStatus":"故障的解决状态，1.解决 2.未解决格式为(int32)","versionId":"故障查询的版本Id"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 缺陷分类查询表一



Jql查询出相关的Issue列表，根据缺陷来源进行分类的查询表。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 根据版本查询故障报表一 | | | | |
| 接口描述 | 根据版本查询故障报表一 | | | |
| 接口备注 | 需要版本Id和故障来源类型 | | | |
| URL | /issue/getBugInfoReportA | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | BugBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"action":"故障的状态名称","bugFrom":"故障的来源，1.SIT环境 SIT 2.UAT环境 UAT 3.生产环境 PRO 4.其他环境 OTHER格式为(int32)","bugFromType":"故障的来源，SIT UAT PRO OTHER","bugId":"故障的Id值","key":"故障的key值","name":"故障的名称","priority":"故障的缺陷等级 1.Trivial 2.Minor 3.Major 4.Critical 5.Blocker","reOpenStatus":"故障的重新打开状态，1.未被重新打开 2.被重新打开过格式为(int32)","solveStatus":"故障的解决状态，1.解决 2.未解决格式为(int32)","versionId":"故障查询的版本Id"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 缺陷分类查询表二



Jql查询出相关的Issue列表，根据缺陷等级进行分类的查询表。

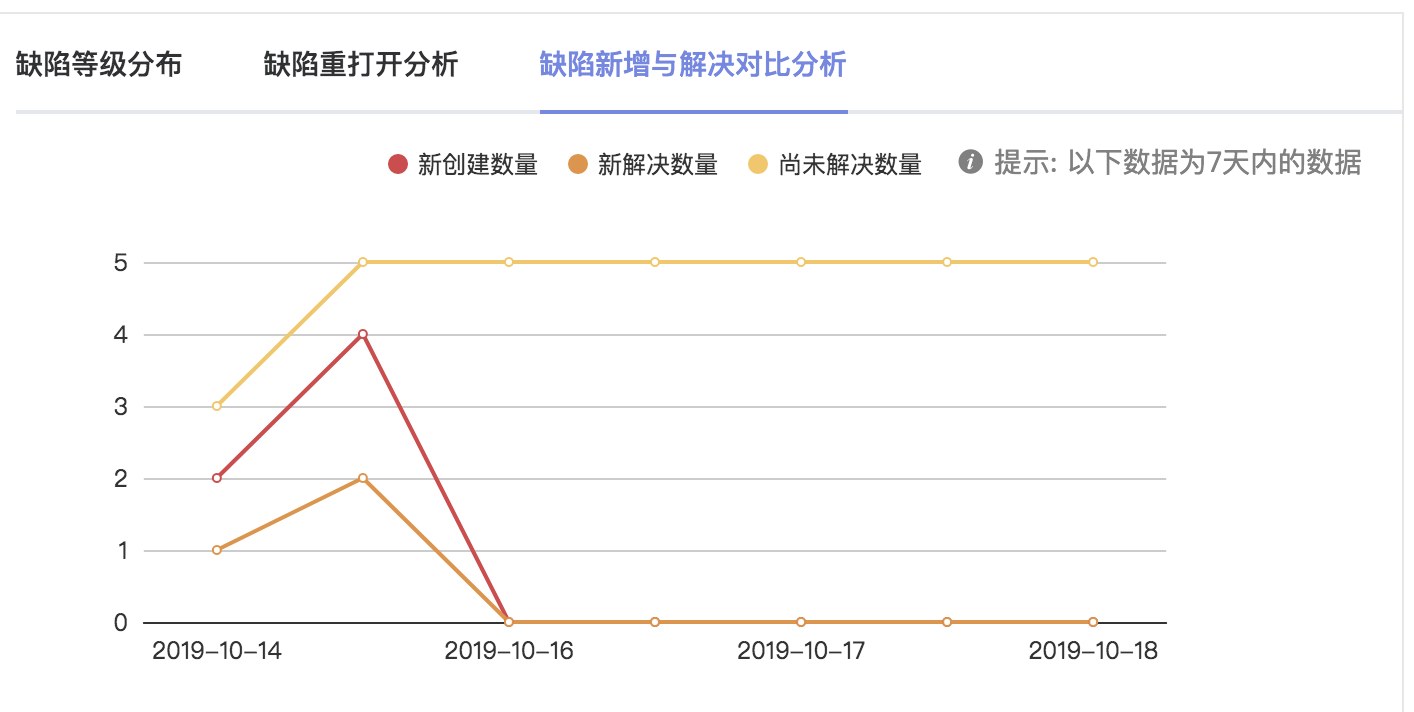
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 根据版本查询故障报表二 | | | | |
| 接口描述 | 根据版本查询故障报表二 | | | |
| 接口备注 | 需要版本Id和故障等级类型 | | | |
| URL | /issue/getBugInfoReportB | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | BugBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"action":"故障的状态名称","bugFrom":"故障的来源，1.SIT环境 SIT 2.UAT环境 UAT 3.生产环境 PRO 4.其他环境 OTHER格式为(int32)","bugFromType":"故障的来源，SIT UAT PRO OTHER","bugId":"故障的Id值","key":"故障的key值","name":"故障的名称","priority":"故障的缺陷等级 1.Trivial 2.Minor 3.Major 4.Critical 5.Blocker","reOpenStatus":"故障的重新打开状态，1.未被重新打开 2.被重新打开过格式为(int32)","solveStatus":"故障的解决状态，1.解决 2.未解决格式为(int32)","versionId":"故障查询的版本Id"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 查询指定的故事列表关联的缺陷列表信息

根据Jql查询出的Issue列表，根据这些Issue列表查询出的缺陷列表信息。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 查询指定的故事列表关联的故障列表信息 | | | | |
| 接口描述 | 查询指定的故事列表关联的故障列表信息 | | | |
| 接口备注 | 需要指定项目应用和版本 | | | |
| URL | /issue/getBugsByIssues | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | JqlEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用id","applicationName":"应用名称","applicationServiceId":"应用下服务id","applicationServiceName":"应用下服务名称","bugEnvId":"Issue发现环境Id","bugEnvName":"Issue发现环境名称","endTimeStr":"结束查询日期 2019-07-14","isLevel":"查询结果是否为层级结构 是 true 否 false","issueId":"Issue的Id,查询时可以多个,以,隔开","issueKey":"Issue的key","issueTypeId":"Issue类型的id","issueTypeName":"Issue类型的名称","notStatusId":"非Issue状态 0 代办 1 开发中 2 开发集成 3 待测试 4 SIT测试中 5 SIT测试完成 6 UAT测试中 7 UAT测试完成 8 待发布 9 已发布","notStatusName":"非Issue状态 0 代办 1 开发中 2 开发集成 3 待测试 4 SIT测试中 5 SIT测试完成 6 UAT测试中 7 UAT测试完成 8 待发布 9 已发布","priorityId":"Issue优先级Id","priorityName":"Issue优先级名称","projectId":"项目id","projectKey":"项目key","resolutionEndDateStr":"解决查询结束日期 2019-07-14","resolutionStartDateStr":"解决查询开始日期 2019-07-14","sprintId":"Sprint的id","sprintName":"Sprint名称","startTimeStr":"开始查询日期 2019-07-13","statusId":"Issue状态 0 代办 1 开发中 2 开发集成 3 待测试 4 SIT测试中 5 SIT测试完成 6 UAT测试中 7 UAT测试完成 8 待发布 9 已发布","statusName":"Issue状态 0 代办 1 开发中 2 开发集成 3 待测试 4 SIT测试中 5 SIT测试完成 6 UAT测试中 7 UAT测试完成 8 待发布 9 已发布","versionId":"版本id","versionName":"版本名称"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 获取缺陷解决和未解决的趋势图



定时任务每天查询全部应用下全部版本的每日缺陷完成情况，添加情况和剩余量情况，

该接口会查询该记录表中的记录进行展示，默认7天内。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 获取缺陷解决和未解决的趋势图 | | | | |
| 接口描述 | 获取缺陷解决和未解决的趋势图 | | | |
| 接口备注 | 需要Jira项目应用版本Id | | | |
| URL | /issue/getBugsTrendInTime | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | VersionBug对象 | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"devops应用标签id格式为(int32)","createNum":"创建的数量格式为(int32)","createTime":"创建记录时间格式为(date-time)","dateSort":"插入日期","id":"id格式为(int32)","jiraApplicationId":"jira应用id格式为(int32)","jiraProjectId":"jira项目id格式为(int32)","jiraVersionId":"jira版本id格式为(int32)","noSolveNum":"未解决的数量格式为(int32)","searchNum":"中间查询字段 7，15，30天格式为(int32)","solveNum":"解决的数量格式为(int32)"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

### Issue的相关操作接口

#### 获取指定Issue可执行的工作流信息

Jira端提供根据IssueId即可查询出该Issue可以执行的工作流接口，该接口调用此接口。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 单个Issue的可执行Action | | | | |
| 接口描述 | 单个Issue的可执行Action | | | |
| 接口备注 | 直接传指定的IssueId | | | |
| URL | /action/{issueId} | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| issueId | string | path | Y | issueId |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"issueId":"string"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### Issue执行指定的工作流接口

查询出Issue可执行的工作流之后，该接口用于将指定的Issue执行到指定的工作流状态

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Issue执行指定工作流 | | | | |
| 接口描述 | Issue执行指定工作流 | | | |
| 接口备注 | 变更操作 | | | |
| URL | /action/startAction | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | ActionUpdateEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"action":"Action的状态名称","actionId":"Action的状态ID格式为(int32)","key":"Issue的key值","status":"Issue的当前状态值","statusId":"Issue的状态id格式为(int32)"} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 接收Jira执行Issue的工作流信息的接口

该接口用于接收Jira端Webhook推送的接口，Issue工作流变更会推送到该接口，然后devops-jira将推送的Issue推送到devops-portal端，修改Issue工作流状态，一般portal端采用同步功能。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Issue工作流变更 | | | | |
| 接口描述 | Issue工作流变更 | | | |
| 接口备注 | 记录操作 | | | |
| URL | /action/handle | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| object | body | {} | Y | object |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

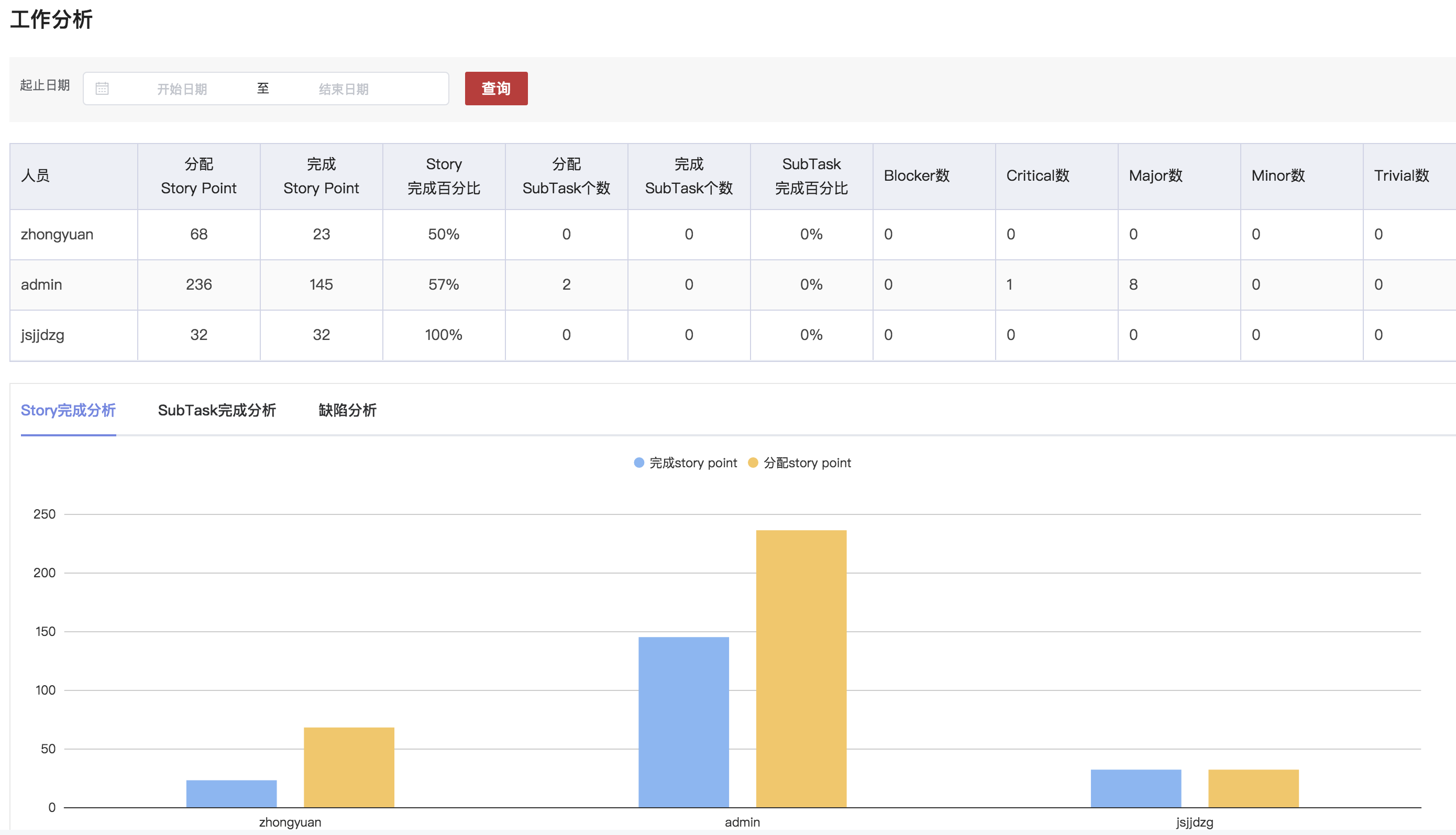
#### 接收Jira删除Issue的接口

该接口用于接收Jira端Webhook推送的接口，Issue删除会推送到该接口，然后devops-jira将推送的Issue推送到devops-portal端，修改Issue状态。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Jira删除Issue触发 | | | | |
| 接口描述 | Jira删除Issue触发 | | | |
| 接口备注 | 记录删除操作 | | | |
| URL | /action/issueDelete | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| object | body | {} | Y | object |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

### 用户相关信息的查询接口

#### 根据时间范围工作人员的Issue工作信息



根据时间范围，项目和应用查询出在该应用下，每个开发人员完成的任务、缺陷、子任务量和比例。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 根据时间范围工作人员的Issue工作学习 | | | | |
| 接口描述 | 根据时间范围工作人员的Issue工作学习 | | | |
| 接口备注 | 项目id,应用id,时间范围 | | | |
| URL | /userInfo/getMembersWorkInfoInTime | | | |
| 请求方式 | post | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| entity | body | VersionBaseEntity | Y | entity |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 201 | Created | Created | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"applicationId":"应用ID格式为(int32)","applicationName":"应用ID","archived":"是否归档","description":"版本描述","devOpsApplicationId":"DevOps平台应用Id","efficiencyType":"效率类型格式为(int32)","endTime":"查询结束时间戳格式为(int64)","endTimeStr":"结束查询日期 2019-07-14","eventType":"操作类型 created/updated/deleted","id":"版本ID","jiraApplicationId":"对应的Jira应用ID格式为(int32)","name":"版本名称","overdue":"版本是否延期","projectId":"对应的Jira的版本对应的项目ID格式为(int32)","releaseDate":"计划发布时间格式为(date-time)","releaseDateStr":"计划发布时间字符串","released":"是否发布","self":"查询URL","startDate":"计划开始时间格式为(date-time)","startDateStr":"计划开始时间字符串","startTime":"查询开始时间戳格式为(int64)","startTimeStr":"开始查询日期 2019-07-13","userReleaseDate":"","userStartDate":""} | | | |
| 返回值 | {"code":"格式为(int32)","data":"","message":"","success":"","timestamp":"格式为(int64)"} | | | |

#### 查询用户头像的Url接口

可以直接获取指定用户头像字节流，没有该用户信息有默认头像信息。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 查询用户头像的Url接口 | | | | |
| 接口描述 | 查询用户头像的Url接口 | | | |
| 接口备注 | 需要用户名称和头像类型 | | | |
| URL | /userInfo/getUserAvatar | | | |
| 请求方式 | get | | | |
| 请求类型 | application/json | | | |
| 返回类型 | \*/\* | | | |
| 参数名 | 数据类型 | 参数类型 | 是否必填 | 说明 |
| username | string | query | N | 用户名称 |
| avatarType | string | query | N | 头像类型 xsmall/small/medium/large |
| 状态码 | 描述 | 说明 | | |
| 200 | OK | OK | | |
| 401 | Unauthorized | Unauthorized | | |
| 403 | Forbidden | Forbidden | | |
| 404 | Not Found | Not Found | | |
| 示例 | | | | |
| 请求参数 | {"avatarType":"string","username":"string"} | | | |
| 返回值 | 字节流 | | | |