

# Vacation Manager for JIRA - REST API 1.0

(For Jira Server, Jira Data Center)

By Transition Technologies PSC

Visit our website: <https://ttpsc.com/en/atlassian/ttpsc-apps/>

Marketplace: <https://marketplace.atlassian.com/vendors/37453/transition-technologies-psc>

App documentation: <https://psc-software.atlassian.net/wiki/spaces/VMSD/overview>

## Table of Content:

|   |    |
|---|----|
| 1. REST API 1.0                           | 2  |
| 1.1 REST API usage example in VacMan flow | 2  |
| 1.2 Paths for Vacation Manager REST API   | 9  |
| 1.3 Initial setup [API 1.0]               | 11 |
| 1.4 Configuration [API 1.0]               | 15 |
| 1.5 Vacation instances                    | 26 |
| 1.6 Vacation documents                    | 32 |
| 1.7 Vacation definitions                  | 43 |
| 1.8 Vacation types                        | 48 |
| 1.9 Teams                                 | 56 |
| 1.10 Holidays                             | 68 |
| 1.11 Holiday schemes                      | 72 |
| 1.12 Workweek schemes                     | 78 |
| 1.13 Statistics                           | 87 |

## REST API 1.0

- [REST API usage example in VacMan flow](#)
- [Paths for Vacation Manager REST API](#)
- [Initial setup \[API 1.0\]](#)
- [Configuration \[API 1.0\]](#)
- [Vacation instances](#)
- [Vacation documents](#)
- [Vacation definitions](#)
- [Vacation types](#)
- [Teams](#)
- [Holidays](#)
- [Holiday schemes](#)
- [Workweek schemes](#)
- [Statistics](#)

### REST API usage example in VacMan flow

This page provides example of Vacation Manager REST API usage.

- [1. Do initial setup of Vacation Manager](#)
- [2. Create first team](#)
- [3. Add supervisor to team](#)
- [4. Add users to team](#)
- [5. Add new holiday](#)
- [6. Add vacation definition to user](#)
- [7. Create vacation document](#)
- [8. Send vacation document to supervisor](#)
- [9. Approve vacation document](#)

#### 1. Do initial setup of Vacation Manager

User: [JIRA ADMINISTRATOR](#)

[Documentation](#)

#### Request

```
POST {jira_baseurl}/rest/vacation-manager/1.0/initial-setup
```

▼ [see request body](#)

### Body

```
{
  "project": {
    "name": "Vacation Manager",
    "key": "VACMAN",
    "description": "Vacation project",
    "lead": "admin",
    "groupHr": "vacman-hr"
  },
  "vacationType": {
    "name": "Leave",
    "description": "Standard leave",
    "kindType": "RECURRING",
    "periodDefinition": {
      "type": "MONTHS",
      "value": 12,
      "startDate": "2018-01-01"
    },
    "hasCarryOver": true,
    "defaultDayLimit": 20,
    "vacationDocumentPeriodType": "DAILY_OR_HALF_DAILY"
  }
}
```

## 2. Create first team

User: **HR**

[Documentation](#)

### Request

```
POST {jira_baseurl}/rest/vacation-manager/1.0/team
```

▼ [see request body](#)

### Body

```
{
  "name": "Workers",
  "description": ""
}
```

▼ see response

Response: **STATUS 201**

### Response

```
{
  "id": 1,
  "name": "Workers",
  "description": ""
}
```

### 3. Add supervisor to team

User: **HR**

[Documentation](#)

### Request

```
POST {jira_baseurl}/rest/vacation-manager/1.0/team/1/member
```

▼ see request body

### Body

```
{
  "name": "supervisor",
  "role": "SUPERVISOR",
  "type": "USER"
}
```

### 4. Add users to team

User: **HR**

[Documentation](#)

### Request

```
POST {jira_baseurl}/rest/vacation-manager/1.0/team/1/member
```

▼ see request body

### Body

```
{
  "name": "jira-software-users",
  "role": "USER",
  "type": "GROUP"
}
```

## 5. Add new holiday

User: **HR**

### 5.1. Get list of holiday schemes to obtain id

[Documentation](#)

### Request

```
GET {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme
```

▼ [see response](#)

Response: **STATUS 200**

### Response

```
[
  {
    "id": 1,
    "name": "Default holiday scheme"
  }
]
```

## 5.2. Add new holiday

[Documentation](#)

### Request

```
POST {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme
```

▼ [see request body](#)

### Body

```
{
  "date": "2018-12-25",
  "yearly": true,
  "description": "Christmas"
}
```

## 6. Add vacation definition to user

User: **HR**

[Documentation](#)

### Request

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-definition
```

▼ [see request body](#)

### Body

```
{
  "vacationTypeId": "1",
  "userName": "john",
  "startDate": "2018-01-01",
  "endDate": "2018-12-31",
  "dayLimit": 15
}
```

## 7. Create vacation document

User: john **JIRA USER**

[Documentation](#)

### Request

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-document
```

▼ [see request body](#)

### Body

```
{
  "reporterName": "john",
  "supervisorName": "supervisor",
  "vacationKindId": 1,
  "periodType": "DAILY",
  "startDate": "2018-11-02",
  "endDate": "2018-11-03"
}
```

## 8. Send vacation document to supervisor

User: john **JIRA USER**

### 8.1. Get list of actions for vacation document

[Documentation](#)

### Request

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/VACMAN-1/actions
```

▼ [see response](#)

Response: **STATUS 200**

### Response

```
[
  {
    "id": 31,
    "name": "Send to supervisor"
  },
  {
    "id": 61,
    "name": "Delete"
  }
]
```

### 8.2. Change status of vacation document

[Documentation](#)

### Request

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/status
```

▼ see request body

### Body

```
{
  "issueKey": "VACMAN-1",
  "actionId": 31
}
```

## 9. Approve vacation document

User: **SUPERVISOR**

### 9.1. Get list of actions for vacation document

[Documentation](#)

### Request

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/VACMAN-1/actions
```

▼ see response

Response: **STATUS 200**



### Response

```
[
  {
    "id": 11,
    "name": "Approve"
  },
  {
    "id": 21,
    "name": "Reject"
  },
  {
    "id": 41,
    "name": "More information required"
  }
]
```

## 9.2. Change status of vacation document

[Documentation](#)

### Request

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/status
```

▼ [see request body](#)

### Body

```
{
  "issueKey": "VACMAN-1",
  "actionId": 11
}
```

## Paths for Vacation Manager REST API

### Postman rest collection

For your convenience we prepared [Postman](#) collection, so you can easily test Vacation Manager REST API on your environment.

[Vacation Manager REST API 1.0.postman\\_collection.json](#)

*Variables:*

`{{baseurl}}` - path to your Jira,

`{{username}}` - name of Jira user that will execute REST queries,

{{userpassword}} - password for that user.

## Paths

{jira\_baseurl}/vacation-manager/1.0

| Name   | REST type | Description                          | Permissions |
|--|-----------|--------------------------------------|-------------|
| <b>Teams</b><br><a href="#">Documentation</a>                |           |                                      |             |
| /team  | POST      | create team                          | ADMIN, HR   |
| /team/{teamId}/member  | POST      | add member                           | ADMIN, HR   |
| /team  | GET       | get all teams                        | ADMIN, HR   |
| /team/{teamId}   | GET       | get team info (users)                | ADMIN, HR   |
| /team/{teamId}/member  | GET       | get team info (members)              | ADMIN, HR   |
| /team/{teamId}   | PATCH     | update team                          | ADMIN, HR   |
| /team/{teamId}   | DELETE    | delete team                          | ADMIN, HR   |
| /team/{teamId}/member  | DELETE    | delete team member                   | ADMIN, HR   |
| <b>Vacation types</b><br><a href="#">Documentation</a>       |           |                                      |             |
| /vacation-type   | POST      | create vacation type                 | ADMIN       |
| /vacation-type   | GET       | get all vacation types               | ALL         |
| /vacation-type/{vacationTypeId}                              | GET       | get vacation type info               | ALL         |
| /vacation-type/{vacationTypeId}/update-info                  | GET       | get vacation type update info        | ALL         |
| /vacation-type/{vacationTypeId}                              | PUT       | update vacation type                 | ADMIN       |
| /vacation-type/{vacationTypeId}                              | DELETE    | delete vacation type                 | ADMIN       |
| <b>Holiday schemes</b><br><a href="#">Documentation</a>      |           |                                      |             |
| /holiday-scheme  | POST      | create holiday scheme                | ADMIN, HR   |
| /holiday-scheme  | GET       | get all holiday schemes              | ADMIN, HR   |
| /holiday-scheme/{holidaySchemeId}                            | PATCH     | update holiday scheme                | ADMIN, HR   |
| /holiday-scheme/{holidaySchemeId}                            | DELETE    | delete holiday scheme                | ADMIN, HR   |
| /vacmanuser/holiday/bulk                                     | PUT       | assign holiday scheme                | ADMIN, HR   |
| <b>Holidays</b><br><a href="#">Documentation</a>             |           |                                      |             |
| /holiday-scheme/{holidayschemeId}/holiday                    | POST      | create holiday                       | ADMIN, HR   |
| /holiday-scheme/{holidayschemeId}/holiday                    | GET       | get holidays from scheme             | ADMIN, HR   |
| /holiday-scheme/{holidayschemeId}/holiday/{holidayId}        | PUT       | update holiday                       | ADMIN, HR   |
| /holiday-scheme/{holidayschemeId}/holiday/{holidayId}        | DELETE    | delete holiday                       | ADMIN, HR   |
| <b>Vacation definitions</b><br><a href="#">Documentation</a> |           |                                      |             |
| /vacation-definition   | POST      | add vacation definition              | ADMIN, HR   |
| /vacation-definition/recurring                               | PUT       | update recurring vacation definition | ADMIN, HR   |
| /vacation-definition/{userName}                              | GET       | get vacation definition for user     | ADMIN, HR   |
| /vacation-definition/{vacationDefinitionId}                  | DELETE    | delete vacation definition           | ADMIN, HR   |
| <b>Vacation documents</b><br><a href="#">Documentation</a>   |           |                                      |             |
| /vacation-document   | POST      | create new vacation document         | ALL         |

|   |        |  |                 |
|---|--------|--|-----------------|
| /vacation-document  | PUT    | update vacation document                               | ALL             |
| /vacation-document/status   | PUT    | update status of vacation document                     | ALL             |
| /vacation-document  | GET    | get vacation documents list                            | ALL             |
| /vacation-document/{issueKey}                                     | GET    | get single vacation document by issue key              | ALL             |
| /vacation-document/waiting-for-approval                           | GET    | get vacation documents waiting for approval            | ALL             |
| /vacation-document/issueKey/actions                               | GET    | get actions available for vacation document            | ALL             |
| <b>Initial setup</b><br><a href="#">Documentation</a>             |        |  |                 |
| /initial-setup  | POST   | create initial setup                                   | ADMIN           |
| <b>Statistics</b><br><a href="#">Documentation</a>                |        |  |                 |
| /statistics/user  | GET    | get instances statistics by logged user                | ALL             |
| /statistics/user/{userName}                                       | GET    | get instances statistics by user                       | ADMIN, HR       |
| /statistics/vacation-type/{vacationTypeId}                        | GET    | get instances statistics by vacation type              | ADMIN, HR       |
| <b>Workweek schemes</b><br><a href="#">Documentation</a>          |        |  |                 |
| /workweek-scheme  | GET    | get workweek schemes list                              | ADMIN, HR       |
| /workweek-scheme/{schemedId}                                      | GET    | get single workweek scheme                             | ADMIN, HR       |
| /workweek-scheme?userName={userName}                              | GET    | get workweek scheme for user                           | ADMIN, HR       |
| /workweek-scheme  | POST   | create workweek scheme                                 | ADMIN, HR       |
| /workweek-scheme/{schemedId}                                      | PATCH  | update workweek scheme                                 | ADMIN, HR       |
| /workweek-scheme/{schemedId}                                      | DELETE | delete Workweek Scheme                                 | ADMIN, HR       |
| <b>Vacation instances</b><br><a href="#">Documentation</a>        |        |  |                 |
| /vacation-instance?vacationDefinitionId=ID                        | GET    | Get vacation instances for user                        | ADMIN, USER, HR |
| /vacation-instance/{userName}?startDate=ISO_DATE&endDate=ISO_DATE | POST   | Add instance to existing recurring vacation definition | ADMIN, HR       |
| /vacation-instance/{vacationInstanceId}                           | PUT    | Update existing vacation instance                      | ADMIN, HR       |
| <b>Configuration</b><br><a href="#">Documentation</a>             |        |  |                 |
| /configuration  | GET    | get whole configuration                                | ADMIN           |
| /configuration/general  | GET    | get general configuration                              | ADMIN           |
| /configuration/groups   | GET    | get groups configuration                               | ADMIN           |
| /configuration/fields   | GET    | get fields configuration                               | ADMIN           |
| /configuration/refresh-recurring-vacation-definitions             | POST   | recalculate recurring vacations                        | ADMIN           |
| /configuration/general  | PUT    | update general configuration                           | ADMIN           |
| /configuration/groups   | PUT    | update groups configuration                            | ADMIN           |
| /configuration/fields   | PUT    | update fields configuration                            | ADMIN           |
| /configuration  | DELETE | delete configuration                                   | ADMIN           |

## Initial setup [API 1.0]

### Overview

Vacation Manager's REST API provides initial setup API which allows you to:

- [Create initial setup](#)

**Methods:**

**POST**

▼ [Create initial setup](#)

Creates Vacation Manager project and first Vacation Type.

**Path**

```
POST {jira_baseurl}/rest/vacation-manager/1.0/initial-setup
```

**Permissions**

**ADMIN** **USER** **SUPERVISOR** **HR**

**JSON body fields**

| Name         | Type         | Description                  |
|--------------|--------------|------------------------------|
| project      | Project      | Project details object.      |
| vacationType | VacationType | First Vacation Type details. |

Project

| Name        | Type              | Description   |
|-------------|-------------------|---|
| name        | String            | Name of a new project.  |
| key         | String            | Key of a new project (all naming rules applies just like in other Jira projects). |
| description | String (Optional) | Description of a new project.   |
| lead        | String            | Lead user name of a new project.  |
| groupHr     | String            | Group name that will be assigned as HR group.                                     |

VacationType

| Name                       | Type                                       | Description  |
|----------------------------|--|--|
| name                       | String                                     | Name of new vacation type (this is also name for issue type associated with that new vacation type). |
| description                | String (optional)                          | Description of vacation type.  |
| kindType                   | KindType                                   | Kind of new vacation type.   |
| vacationDocumentPeriodType | VacationDocumentPeriodType                 | Type of available vacation document period.  |
| periodDefinition           | Object (required if kindType is RECURRING) | PeriodDefinition object for recurring vacation type.   |
| hasCarryOver               | bool (required if kindType is RECURRING)   | Specifies if unused days will be transferred to new vacation periods.                                |
| hasEndDate                 | bool (required if kindType is ONE_TIME)    | Specifies if assigned vacation days will be having expiration date.                                  |

|                 |   |  |
|-----------------|---|--|
| hasDayLimit     | bool (required if kindType is ONE_TIME) | Specifies if there will be limit of days that user can use in this vacation type.  |
| defaultDayLimit | int                                     | Default vacation days limit for vacation type. This value will be suggested when HR will be assigning new vacation days for users. |

#### PeriodDefinition

| Name      | Type                 | Description  |
|-----------|----------------------|--|
| type      | PeriodDefinitionType | Type of recurring period.<br>Currently only monthly period is available. |
| value     | int                  | Number of months in recurring period.<br>Valid values: 1, 2, 3, 4, 6, 12 |
| startDate | Date                 | Date of when period will start. ISO 8601 format.                         |

#### VacationDocumentPeriodType

| Name                | Type | Description                                      |
|---------------------|------|--|
| DAILY               | Enum | Vacation type for daily vacations                |
| DAILY_OR_HALF_DAILY | Enum | Vacation type for daily and half daily vacations |
| DAILY_OR_HOURLY     | Enum | Vacation type for hourly vacations               |

#### KindType

| Name      | Type | Description                                |
|-----------|------|--|
| ONE_TIME  | Enum | Vacation kind type for one time vacations  |
| RECURRING | Enum | Vacation kind type for recurring vacations |

#### PeriodDefinitionType

| Name   | Type | Description                      |
|--------|------|----------------------------------|
| MONTHS | Enum | Type of monthly recurring period |

## Response

201 400 500

## Example

Example

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/initial-setup
```

## Body

```
{
  "project":{
    "name":"Vacation Manager",
    "key":"VM",
    "description":"Vacation project",
    "lead":"admin",
    "groupHr":"vacman-hr"
  },
  "vacationType":{
    "name":"Leave",
    "description":"Standard leave",
    "kindType":"RECURRING",
    "periodDefinition": {
      "type": "MONTHS",
      "value": "12",
      "startDate": "2018-01-01"
    },
    "hasCarryOver":true,
    "defaultDayLimit":20,
    "vacationDocumentPeriodType":"DAILY_OR_HALF_DAILY"
  }
}
```

## Response

```
{
  "done": true,
  "projectId": 10001,
  "groups": {
    "HR": "vacman-hr"
  },
  "fields": {
    "SUBSTITUTE": 10115,
    "USED_VACATION_MINUTES": 10117,
    "START_DATE": 10109,
    "SUPERVISOR": 10114,
    "END_DATE": 10110,
    "CHART": 10107,
    "HALF_DAY": 10113,
    "USED_VACATION_DAYS": 10116,
    "START_DATE_TIME": 10111,
    "END_TIME_MINUTES": 10112,
    "PERIOD_TYPE": 10108
  },
}
```

```
"schemes": {
  "FIELD_LAYOUT": 10100,
  "FIELD_SCREEN_SCHEME": 10002,
  "ISSUE_TYPE_SCREEN_SCHEME": 10001,
  "FIELD_CONFIG_SCHEME": 10308,
  "FIELD_LAYOUT_SCHEME": 10000
},
"options": {
  "HALF_DAY_AM": 10006,
  "PERIOD_TYPE_HALF_DAILY": 10004,
  "PERIOD_TYPE_DAILY": 10003,
  "PERIOD_TYPE_HOURLY": 10005,
  "HALF_DAY_PM": 10007
},
"automaticSummary": true,
"automaticSummaryTexts": {
  "DAILY": "${reporter} Vacation Document (${vacationType}):
on ${startDate}, ${vacationDays} days",
  "HOURLY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, from ${vacationStartTime} to
${vacationEndTime} (${vacationHours}h ${vacationMinutes}m)",
  "HALF_DAILY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, half of the day
(${vacationHalfOfDay})"
},
"documentStatusesApprovedIds": [
  "10004"
],
"documentStatusesClosedIds": [
  "10005",
  "10007",
  "10008"
],
"jobTime": 0,
"showUserPickerWarning": true,
"daysBeforeLeaveWarning": 7,
"defaultHolidaySchemeId": 1,
"defaultWorkweekSchemeId": 1
}
```

## Configuration [API 1.0]

### Overview

Vacation Manager's REST API provides configuration API which allows you to:

- [recalculate recurring vacations](#)
- [update general configuration](#)
- [update groups configuration](#)
- [update fields configuration](#)

- [get whole configuration](#)
- [get general configuration](#)
- [get groups configuration](#)
- [get fields configuration](#)
- [delete configuration](#)

## Methods:

### POST

#### ▼ [Recalculate recurring vacations](#)

Triggers recalculation of recurring vacations.

#### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/configuration/refresh-recurring-vacation-definitions
```

#### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

#### Example

##### ▼ [Example](#)

#### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/configuration/refresh-recurring-vacation-definitions
```

### PUT

#### ▼ [Update general configuration](#)

Updates and returns general configuration.

#### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/configuration/general
```

#### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

#### JSON body fields

| Name | Type | Description |
|------|------|-------------|
|------|------|-------------|



|                        |                         |   |
|------------------------|-------------------------|---|
| automaticSummary       | Boolean                 | true - summary for new vacation documents will be filled automatically<br>false - users will have to provide summary for vacation documents           |
| automaticSummaryTexts  | Map<PeriodType, String> | Automatic summary texts that will be used to fill summary for new vacation documents. Each request period type has to have corresponding summary text |
| jobTime                | Integer                 | Number representing full hour to run job recalculating recurring vacation (job is run everyday)   |
| showUserPickerWarning  | Boolean                 | true - show warning near user picker fields when user is currently out on vacation or will be in near future<br>false - hide warning                  |
| daysBeforeLeaveWarning | Integer                 | Number of days before user leave that user picker warning about upcoming absence will be visible  |

PeriodType

| Name       | Type | Description            |
|------------|------|------------------------|
| DAILY      | Enum | Daily period type      |
| HALF_DAILY | Enum | Half daily period type |
| HOURLY     | Enum | Hourly period type     |

## Responses

200 400 500

## Example

▼ Example

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/configuration/general
```

### Body

```
{
  "automaticSummary": true,
  "automaticSummaryTexts": {
    "DAILY": "${reporter} Vacation Document (${vacationType}):
on ${startDate}, ${vacationDays} days",
    "HALF_DAILY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, half of the day
(${vacationHalfOfDay})",
    "HOURLY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, from ${vacationStartTime} to
${vacationEndTime} (${vacationHours}h ${vacationMinutes}m)"
  },
  "jobTime": 0,
  "showUserPickerWarning": true,
  "daysBeforeLeaveWarning": 7
}
```

### Response

```
{
  "id": 1,
  "done": true,
  "project": {
    "id": 10100,
    "key": "VACMAN",
    "name": "Vacation Manager"
  },
  "automaticSummary": true,
  "automaticSummaryTexts": {
    "DAILY": "${reporter} Vacation Document (${vacationType}):
on ${startDate}, ${vacationDays} days",
    "HALF_DAILY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, half of the day
(${vacationHalfOfDay})",
    "HOURLY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, from ${vacationStartTime} to
${vacationEndTime} (${vacationHours}h ${vacationMinutes}m)"
  },
  "jobTime": 0,
  "jobLastFireTime": 1539208800000,
  "showUserPickerWarning": true,
  "daysBeforeLeaveWarning": 7
}
```

- Update groups configuration  
Updates groups configuration.

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/configuration/groups
```

### Permissions

ADMIN USER SUPERVISOR HR

### JSON body fields

| Name          | Type    | Description  |
|---------------|---------|--|
| hrGroupName   | String  | Name of Jira group that will be used as HR in VacMan   |
| createSchemes | Boolean | <b>true</b> - creates and assigns new Security and Permission Schemes for Vacation Manager project. Old schemes will be left intact<br><b>false</b> - no new schemes are created |

### Responses

204 400 500

### Example

- Example

#### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/configuration/groups
```

#### Body

```
{  
  "hrGroupName": "hr-users",  
  "createSchemes": true  
}
```

- Update fields configuration  
Updates fields configuration.

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/configuration/fields
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## JSON body fields

| Name                             | Type    | Description  |
|----------------------------------|---------|--|
| chartCustomFieldId               | Long    | Id of custom field to set as chart   |
| periodTypeCustomFieldId          | Long    | Id of custom field to set as period type   |
| startDateCustomFieldId           | Long    | Id of custom field to set as start date  |
| startDateTimeCustomFieldId       | Long    | Id of custom field to set as start date time   |
| endDateCustomFieldId             | Long    | Id of custom field to set as end date  |
| endTimeMinutesCustomFieldId      | Long    | Id of custom field to set as time period picker  |
| halfDayCustomFieldId             | Long    | Id of custom field to set as half of day   |
| supervisorCustomFieldId          | Long    | Id of custom field to set as supervisor  |
| substituteCustomFieldId          | Long    | Id of custom field to set as substitute person   |
| usedVacationDaysCustomFieldId    | Long    | Id of custom field to set as used vacation days  |
| usedVacationMinutesCustomFieldId | Long    | Id of custom field to set as used vacation minutes   |
| migrateData                      | Boolean | true - copies data from old to new fields in all vacation documents<br>false - no data is copied |

## Responses

**204** **400** **500**

## Example

▼ Example

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/configuration/fields
```

### Body

```
{
  "chartCustomFieldId": 10000,
  "periodTypeCustomFieldId": 10001,
  "startDateCustomFieldId": 10002,
  "startDateTimeCustomFieldId": 10003,
  "endDateCustomFieldId": 10004,
  "endTimeMinutesCustomFieldId": 10005,
  "halfDayCustomFieldId": 10006,
  "supervisorCustomFieldId": 10007,
  "substituteCustomFieldId": 10008,
  "usedVacationDaysCustomFieldId": 10009,
  "usedVacationMinutesCustomFieldId": 10010,
  "migrateData": true
}
```

## DELETE

### ▼ Delete configuration

Deletes configuration.

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/configuration
```

### Permissions

ADMIN USER SUPERVISOR HR

### Responses

204 500

### Example

#### ▼ Example

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/configuration
```

## GET

### ▼ Get whole configuration

Returns whole configuration.

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/configuration
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## Responses

**200** **500**

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/configuration
```

### Response

```
{
  "id": 1,
  "done": true,
  "projectId": 10100,
  "groups": {
    "HR": "vacman-hr"
  },
  "fields": {
    "CHART": 10000,
    "START_DATE": 10002,
    "PERIOD_TYPE": 10001,
    "START_DATE_TIME": 10004,
    "USED_VACATION_DAYS": 10009,
    "END_DATE": 10003,
    "END_TIME_MINUTES": 10005,
    "HALF_DAY": 10006,
    "USED_VACATION_MINUTES": 10010,
    "COMMENT": null,
    "SUPERVISOR": 10007,
    "SUBSTITUTE": 10008
  },
  "schemes": {
    "ISSUE_TYPE_SCREEN_SCHEME": 10100,

```

```
        "FIELD_LAYOUT_SCHEME": 10000,
        "FIELD_SCREEN_SCHEME": 10100,
        "FIELD_LAYOUT": 10100,
        "FIELD_CONFIG_SCHEME": 10200
    },
    "options": {
        "PERIOD_TYPE_DAILY": 10000,
        "HALF_DAY_AM": 10003,
        "PERIOD_TYPE_HALF_DAILY": 10001,
        "PERIOD_TYPE_HOURLY": 10002,
        "HALF_DAY_PM": 10004
    },
    "automaticSummary": true,
    "automaticSummaryTexts": {
        "DAILY": "${reporter} Vacation Document (${vacationType}):
on ${startDate}, ${vacationDays} days",
        "HALF_DAILY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, half of the day
(${vacationHalfOfDay})",
        "HOURLY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, from ${vacationStartTime} to
${vacationEndTime} (${vacationHours}h ${vacationMinutes}m)"
    },
    "documentStatusesApprovedIds": [
        "10102"
    ],
    "documentStatusesClosedIds": [
        "10103",
        "10105",
        "10106"
    ],
    "jobTime": 0,
    "jobLastFireTime": 1539208800000,
    "showUserPickerWarning": true,
    "daysBeforeLeaveWarning": 7,
    "defaultHolidaySchemeId": 1,
    "defaultWorkweekSchemeId": 1
}
```

▼ [Get general configuration](#)

Returns general configuration.

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/configuration/general
```

### Permissions

ADMIN USER SUPERVISOR HR

## Responses

200 500

### Example

▼ Example

#### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/configuration/general
```

#### Response

```
{
  "id": 1,
  "done": true,
  "project": {
    "id": 10100,
    "key": "VACMAN",
    "name": "Vacation Manager"
  },
  "automaticSummary": true,
  "automaticSummaryTexts": {
    "DAILY": "${reporter} Vacation Document (${vacationType}):
on ${startDate}, ${vacationDays} days",
    "HALF_DAILY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, half of the day
(${vacationHalfOfDay})",
    "HOURLY": "${reporter} Vacation Document
(${vacationType}): on ${startDate}, from ${vacationStartTime} to
${vacationEndTime} (${vacationHours}h ${vacationMinutes}m)"
  },
  "jobTime": 0,
  "jobLastFireTime": 1539208800000,
  "showUserPickerWarning": true,
  "daysBeforeLeaveWarning": 7
}
```

▼ [Get groups configuration](#)

Returns groups configuration.

#### Path



```
GET {jira_baseurl}/rest/vacation-manager/1.0/configuration/groups
```

## Permissions

ADMIN USER SUPERVISOR HR

## Responses

200 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/configuration/groups
```

### Response

```
{
  "groups": {
    "HR": {
      "name": "vacman-hr",
      "type": "HR"
    }
  }
}
```

▼ Get fields configuration

Returns fields configuration.

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/configuration/fields
```

## Permissions

ADMIN USER SUPERVISOR HR

## Responses

200 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/configuration/fields
```

### Response

```
{
  "fields": {
    "CHART": {
      "id": 10000,
      "name": "Statistics",
      "nameKey": "Statistics",
      "fieldName": "Statistics",
      "type": "CHART"
    },
    "USED_VACATION_DAYS": {
      "id": 10009,
      "name": "Used vacation days",
      "nameKey": "Used vacation days",
      "fieldName": "Used vacation days",
      "type": "USED_VACATION_DAYS"
    },
    "END_DATE": {
      "id": 10003,
      "name": "End Date",
      "nameKey": "End Date",
      "fieldName": "End Date",
      "type": "END_DATE"
    }
  }
}
```

## Vacation instances

### Overview

Vacation Manager's REST API provides vacation types API which allows you to obtain:

- [create vacation instance for recurring vacation definition](#)
- [list of vacation instances for user](#)
- [edit existing vacation instance](#)

### Methods:

## POST

▼ [Add instance to existing recurring vacation definition](#)

Creates new vacation instance for given vacation definition. Currently it works only for RECURRING vacation types, as for ONE\_TIME can only have one instance that is already automatically created at vacation definition creation.

## Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-instance?
vacationDefinitionId=ID
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## Query params

| Name                 | Type    | Description                |
|----------------------|---------|----------------------------|
| vacationDefinitionId | Integer | Id of vacation definition. |

## JSON body fields

| Name         | Type   | Description  |
|--------------|--------|--|
| entitledDays | Double | Integer value of entitled days to add. If vacation type is "Daily or half-daily", then you can also add half of the day to this value. |

## Responses

**200** **404** **500**

## Example

▼ [Example](#)

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-instance?
vacationDefinitionId=3
```

### Body

```
{
  "entitledDays": "10"
}
```

## Response

```
{
  "id": 9,
  "startDate": "2020-05-01",
  "endDate": "2021-04-30",
  "availableDays": 10,
  "availableMinutes": 0,
  "entitledDays": 10,
  "carryOverDays": 0,
  "carryOverMinutes": 0,
  "usedDays": 0,
  "usedMinutes": 0,
  "vacationDocuments": []
}
```

## GET

### Get vacation instances for user

Returns all vacation instances for selected users.

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-instance/
{userName}?startDate=ISO_DATE&endDate=ISO_DATE
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

*User can get vacation instances only for himself*

## Path params:

| Name     | Type   | Description                            |
|----------|--------|--|
| userName | String | Name of user to get vacation instances |

## Query params

| Name      | Type              | Description                                |
|-----------|-------------------|--|
| startDate | String (optional) | Start date for instances. ISO 8601 format. |
| endDate   | String (optional) | End date for instances. ISO 8601 format.   |

## Responses

200 400 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-instance/admin?startDate=2018-01-01&endDate=2019-01-01
```

### Response

```
[
  {
    "id": 1,
    "startDate": "2018-11-01",
    "endDate": "2018-11-30",
    "availableDays": 4,
    "availableMinutes": 0,
    "entitledDays": 5,
    "carryOverDays": 0,
    "carryOverMinutes": 0,
    "usedDays": 1,
    "usedMinutes": 0,
    "vacationDocuments": [
      {
        "id": 1,
        "vacationInstanceId": 1,
        "issueId": 10100,
        "issueKey": "VACMAN-1",
        "reporterKey": "admin",
        "supervisorKey": "supervisor5",
        "status": "APPROVED",
        "calculatedDays": 1,
        "calculatedMinutes": 0,
        "selectedPeriod": {
          "periodType": "DAILY",
          "startDateTime": 1541545200000,
          "endDateTime": 1541631599999,
          "startDate": "2018-11-07 0:00",
          "endDate": "2018-11-07 23:59",
          "usedDays": 0,
          "usedMinutes": 0
        },
        "periodType": "DAILY"
      }
    ]
  }
]
```

```
    },
    {
      "id": 2,
      "startDate": "2018-11-01",
      "endDate": "2019-04-30",
      "availableDays": 25,
      "availableMinutes": 0,
      "entitledDays": 25,
      "carryOverDays": 0,
      "carryOverMinutes": 0,
      "usedDays": 0,
      "usedMinutes": 0,
      "vacationDocuments": []
    },
    {
      "id": 3,
      "startDate": "2019-05-01",
      "endDate": "2020-04-30",
      "availableDays": 25,
      "availableMinutes": 0,
      "entitledDays": 25,
      "carryOverDays": 0,
      "carryOverMinutes": 0,
      "usedDays": 0,
      "usedMinutes": 0,
      "vacationDocuments": []
    }
  ]
```

## PUT

### Update existing vacation instance

Updates selected vacation instance.

## Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-instance/
{vacationInstanceId}
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path params:

| Name | Type | Description |
|------|------|-------------|
|------|------|-------------|

|                    |         |                           |
|--------------------|---------|---------------------------|
| vacationInstanceid | Integer | Id of vacation definition |
|--------------------|---------|---------------------------|

### JSON body fields:

| Name             | Type    | Description   |
|------------------|---------|---|
| startDate        | String  | New start date for instance. ISO 8601 format.<br><i>Only for ONE_TIME instances.</i>  |
| endDate          | String  | New end date for instance. ISO 8601 format.<br><i>Only for ONE_TIME with hasEndDate instances.</i>  |
| availableDays    | Integer | New number of available days. If vacation type is "Daily or half-daily", then you can also add half of the day to this value.<br><i>Only for hasDayLimit instances.</i> |
| availableMinutes | Integer | New number of available minutes.<br><i>Only for "Daily or hourly" instances.</i>  |
| entitledDays     | Integer | New number of entitled days. If vacation type is "Daily or half-daily", then you can also add half of the day to this value.<br><i>Only for hasDayLimit instances.</i>  |

### Responses

200 400 500

### Example

▼ Example

#### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-instance/10
```

#### Body

```
{
  "startDate": "2018-01-01",
  "endDate": "2018-01-03",
  "availableDays": 10,
  "entitledDays": 20
}
```

## Response

```
{
  "id": 10,
  "startDate": "2018-01-01",
  "endDate": "2018-01-03",
  "availableDays": 10,
  "entitledDays": 20,
  "carryOverDays": 0,
  "carryOverMinutes": 0,
  "usedDays": 0,
  "usedMinutes": 0,
  "vacationDocuments": []
}
```

## Vacation documents

### Overview

Vacation Manager's REST API provides vacation document API which allows you to:

- [create new vacation document](#)
- [update vacation document](#)
- [update status of vacation document](#)
- [get vacation documents list](#)
- [get single vacation document](#)
- [get vacation documents waiting for approval](#)
- [get actions for vacation document](#)

### Methods:

#### GET

##### ▼ [Get vacation documents list](#)

Returns list of vacation documents available to logged user

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme?reporter=
{reporter}&supervisor={supervisor}&startdate={startdate}&enddate=
{enddate}
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### Query params



| Name       | Type   | Description  |
|------------|--------|--|
| reporter   | String | Name of reporter of vacation document                    |
| supervisor | String | Name of supervisor assigned to vacation document         |
| startdate  | String | Date that vacation have to start after. ISO 8601 format. |
| enddate    | String | Date that vacation have to end before. ISO 8601 format.  |

## Responses

200 404 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-document
```

## Response

```
[
  {
    "issueId": 10600,
    "issueKey": "VACMAN-1",
    "reporterKey": "user",
    "supervisorKey": "supervisor",
    "status": "OPEN",
    "selectedPeriod": {
      "periodType": "DAILY",
      "startDateTime": 1540159200000,
      "endDateTime": 1540331999999,
      "startDate": 1540159200000,
      "endDate": 1540331999999,
      "usedDays": 0,
      "usedMinutes": 0
    }
  },
  {
    "issueId": 10502,
    "issueKey": "VACMAN-2",
    "reporterKey": "user",
    "supervisorKey": "supervisor",
    "status": "OPEN",
    "selectedPeriod": {
      "periodType": "HOURLY",
      "startDateTime": 1540371600000,
      "endDateTime": 1540378800000,
      "startDate": 1540371600000,
      "endDate": 1540378800000,
      "usedDays": 0,
      "usedMinutes": 0
    }
  }
]
```

### ✓ [Get single vacation document](#)

Returns information about existing vacation document.

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/
{issueKey}
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path params

| Name     | Type   | Description |
|----------|--------|-------------|
| issueKey | String | Issue key   |

## Responses

200 400 404 500

## Example

▼ Example

### Request

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/  
{issueKey}
```

### Response

```
{  
  "issueId": 10600,  
  "issueKey": "VACMAN-1",  
  "reporterKey": "user",  
  "supervisorKey": "supervisor",  
  "status": "OPEN",  
  "selectedPeriod": {  
    "periodType": "DAILY",  
    "startDateTime": 1540159200000,  
    "endDateTime": 1540331999999,  
    "startDate": 1540159200000,  
    "endDate": 1540331999999,  
    "usedDays": 0,  
    "usedMinutes": 0  
  }  
}
```

▼ [Get vacation documents waiting for approval](#)

Returns vacation documents waiting for approval

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-document
/waiting-for-approval?reporter={reporter}&supervisor={supervisor}
&startdate={startdate}&enddate={enddate}
```

## Permissions

ADMIN USER SUPERVISOR HR

## Query params

| Name       | Type   | Description  |
|------------|--------|--|
| reporter   | String | Name of reporter of vacation document                    |
| supervisor | String | Name of supervisor assigned to vacation document         |
| startdate  | String | Date that vacation have to start after. ISO 8601 format. |
| enddate    | String | Date that vacation have to end before. ISO 8601 format.  |

## Responses

200 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-document
/waiting-for-approval
```

## Response

```
[
  {
    "issueId": 10600,
    "issueKey": "VACMAN-1",
    "reporterKey": "user",
    "supervisorKey": "supervisor",
    "status": "OPEN",
    "selectedPeriod": {
      "periodType": "DAILY",
      "startDateTime": 1540159200000,
      "endDateTime": 1540331999999,
      "startDate": 1540159200000,
      "endDate": 1540331999999,
      "usedDays": 0,
      "usedMinutes": 0
    }
  },
  {
    "issueId": 10502,
    "issueKey": "VACMAN-2",
    "reporterKey": "user",
    "supervisorKey": "supervisor",
    "status": "OPEN",
    "selectedPeriod": {
      "periodType": "HOURLY",
      "startDateTime": 1540371600000,
      "endDateTime": 1540378800000,
      "startDate": 1540371600000,
      "endDate": 1540378800000,
      "usedDays": 0,
      "usedMinutes": 0
    }
  }
]
```

### Get actions for vacation document

Returns available actions that could be executed upon vacation document.

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/
{issueKey}/actions
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path params

| Name     | Type   | Description |
|----------|--------|-------------|
| issueKey | String | Issue key   |

## Responses

200 404 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/VACMAN-2/actions
```

### Response

```
[
  {
    "id": 11,
    "name": "Approve"
  },
  {
    "id": 21,
    "name": "Reject"
  },
  {
    "id": 41,
    "name": "More information required"
  }
]
```

## POST

▼ Create new vacation document

Create and returns new vacation document. It creates new Jira issue.

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-document
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## JSON body fields

| Name           | Type                   | Description   |
|----------------|------------------------|---|
| reporterName   | String                 | Reporter name   |
| supervisorName | String                 | Supervisor name   |
| substituteName | String (Optional)      | Substitute name   |
| description    | String (Optional)      | Description for created issue   |
| vacationKindId | Integer                | Vacation document is created for vacation kind with this id                 |
| periodType     | PeriodType             | Period type   |
| startDate      | String (Optional)      | Start date of the leave for daily and half-daily requests. ISO 8601 format. |
| startDateTime  | String (Optional)      | Start date time of the leave for hourly requests. ISO 8601 format.          |
| endDate        | String (Optional)      | End date of the leave for daily requests. ISO 8601 format.                  |
| halfDayType    | HalfDayType (Optional) | Required for half-daily requests.   |
| endTimeMinutes | Integer (Optional)     | Duration of hourly leave in minutes. Required for hourly request            |

PeriodType

| Name       | Type | Description            |
|------------|------|------------------------|
| DAILY      | Enum | Daily period type      |
| HALF_DAILY | Enum | Half daily period type |
| HOURLY     | Enum | Hourly period type     |

HalfDayType

| Name | Type | Description             |
|------|------|-------------------------|
| AM   | Enum | First half of work day  |
| PM   | Enum | Second half of work day |

## Responses

**201** **400** **500**

## Example

∨ [Example](#)

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-document
```

### Body

```
{
  "reporterName": "user",
  "supervisorName": "supervisor",
  "substituteName": "james",
  "vacationKindId": 1,
  "periodType": "DAILY",
  "startDate": "2018-10-10",
  "endDate": "2018-10-12"
}
```

### Response

```
{
  "issueId": 10600,
  "issueKey": "VACMAN-1",
  "reporterKey": "user",
  "supervisorKey": "supervisor",
  "status": "OPEN",
  "selectedPeriod": {
    "periodType": "DAILY",
    "startDateTime": 1540159200000,
    "endDateTime": 1540245599999,
    "startDate": 1540159200000,
    "endDate": 1540245599999,
    "usedDays": 0,
    "usedMinutes": 0
  }
}
```

## PUT

### Update vacation document

Updates existing vacation document.

### Path



```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-document
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## JSON body fields

| Name           | Type                   | Description   |
|----------------|------------------------|---|
| reporterName   | String                 | Reporter name   |
| supervisorName | String                 | Supervisor name   |
| substituteName | String (Optional)      | Substitute name   |
| description    | String (Optional)      | Description for created issue   |
| vacationKindId | Integer                | Vacation document is created for vacation kind with this id                 |
| periodType     | PeriodType             | Period type   |
| startDate      | String (Optional)      | Start date of the leave for daily and half-daily requests. ISO 8601 format. |
| startDateTime  | String (Optional)      | Start date time of the leave for hourly requests. ISO 8601 format.          |
| endDate        | String (Optional)      | End date of the leave for daily requests. ISO 8601 format.                  |
| halfDayType    | HalfDayType (Optional) | Required for half-daily requests.   |
| endTimeMinutes | Integer (Optional)     | Duration of hourly leave in minutes. Required for hourly request            |

### PeriodType

| Name       | Type | Description            |
|------------|------|------------------------|
| DAILY      | Enum | Daily period type      |
| HALF_DAILY | Enum | Half daily period type |
| HOURLY     | Enum | Hourly period type     |

### HalfDayType

| Name | Type | Description             |
|------|------|-------------------------|
| AM   | Enum | First half of work day  |
| PM   | Enum | Second half of work day |

## Responses

**201** **400** **404** **500**

## Example

▼ [Example](#)

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-document
```

### Body

```
{
  "issueKey": "VACMAN-1",
  "reporterName": "user",
  "supervisorName": "supervisor",
  "periodType": "DAILY",
  "startDate": "2018-10-22",
  "endDate": "2018-10-23"
}
```

#### Update status of vacation document

Updates status of vacation document.

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/status
```

### Permissions

ADMIN USER SUPERVISOR HR

### JSON body fields

| Name     | Type              | Description  |
|----------|-------------------|--|
| issueKey | String            | Key of issue to update status  |
| actionId | Integer           | Id of action to execute on issue in order to make transition. To retrieve available actions, see <a href="#">get actions</a> |
| comment  | String (optional) | Comment to add upon changing status. Some actions require comments   |

### Responses

200 400 404 500

### Example

Example

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-document/status
```

### Body

```
{
  "issueKey": "VACMAN-1",
  "actionId": 1,
  "comment": "Example comment"
}
```

## Vacation definitions

### Overview

Vacation Manager's REST API provides team API which allows you to obtain:

- [create vacation definition](#)
- [get vacation definition for user](#)
- [update vacation definition](#)
- [delete vacation definition](#)

### Methods

#### GET

##### Get vacation definition

Returns all vacation definitions for current user or only selected one vacation definition.

#### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-definition/{userName}?vacationTypeId={vacationTypeId}
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### Path params

| Name     | Type   | Description |
|----------|--------|-------------|
| userName | String | User name   |

## Query params

| Name           | Type | Description      |
|----------------|------|------------------|
| vacationTypeId | int  | Vacation type Id |

## Responses

200 400 500

## Example

Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-definition/username
```

### Response

```
[
  {
    "id": 1,
    "vacationTypeId": 1,
    "userKey": "userKey1",
    "userName": "userName",
    "vacationKindName": "vacation kind name 1",
    "startDate": "2018-10-18",
    "dayLimit": 0
  },
  {
    "id": 2,
    "vacationTypeId": 2,
    "userKey": "userKey1",
    "userName": "userName",
    "vacationKindName": "vacation kind name 2",
    "startDate": "2018-10-30",
    "dayLimit": 23
  }
]
```

## POST

Example

Creates and returns vacation definition.

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-definition
```

## Permissions

ADMIN USER SUPERVISOR HR

## JSON body fields

| Name           | Type    | Description  |
|----------------|---------|--|
| vacationTypeId | Integer | vacation type id   |
| userName       | String  | username to which dew vacation definition will be assigned   |
| startDate      | String  | Date when user is enabled to request vacation using this definition. ISO 8601 format.  |
| endDate        | String  | Date when one time vacation type no longer enables user to request vacation using it. Must be used only when vacation has valid through option enabled. ISO 8601 format. |
| dayLimit       | Double  | Default day limit. Should be integer when vacation is "daily only" or "daily or hourly". May contain additional half if vacation is "daily or half daily"                |

## Responses

201 400 500

## Example

Example

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-definition
```

### Body

```
{
  "vacationTypeId": "1",
  "userName": "userName",
  "startDate": "2018-09-10",
  "endDate": "2018-09-20",
  "dayLimit": 5
}
```

## Response

```
{
  "id": 8,
  "vacationTypeId": 1,
  "userKey": "userKey",
  "userName": "userName",
  "vacationKindName": "Vacation kind name",
  "startDate": "2018-09-10",
  "endDate": "2018-09-20",
  "dayLimit": 5
}
```

## PUT

### update vacation definition

Returns and updates vacation definition.

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-definition/
{vacationDefinitionId}
```

### Permissions

ADMIN USER SUPERVISOR HR

### Path params

| Name                 | Type | Description            |
|----------------------|------|------------------------|
| vacationDefinitionId | int  | Vacation definition ID |

### JSON body fields

| Name     | Type   | Description |
|----------|--------|-------------|
| dayLimit | double | Day limit   |

### Responses

200 400 500

### Example

Example

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-definition/2
```

### Body

```
{  
  "dayLimit": 2  
}
```

### Response

```
{  
  "id": 2,  
  "vacationTypeId": 2,  
  "userKey": "userKey",  
  "userName": "userName",  
  "vacationKindName": "vacation kind name 2",  
  "startDate": "2018-10-30",  
  "dayLimit": 2  
}
```

## DELETE

### ▼ Delete vacation definition

Deletes vacation definition.

### Path

```
delete {jira_baseurl}/rest/vacation-manager/1.0/vacation-definition/  
{vacationDefinitionId}
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### Path params

| Name                 | Type | Description            |
|----------------------|------|------------------------|
| vacationDefinitionId | int  | Vacation definition ID |

## Responses

204 404 500

## Example

Example

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/vacation-definition/1
```

## Vacation types

### Overview

Vacation Manager's REST API provides vacation types API which allows you to obtain:

- [Create vacation type](#)
- [Delete vacation type](#)
- [Update vacation type](#)
- [Get all vacation types](#)
- [Get information about vacation type](#)
- [Get vacation type update information](#)

### Methods

## GET

Get single vacation type

Returns information about specific vacation type.

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-type/{vacationTypeId}
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path params

| Name           | Type | Description      |
|----------------|------|------------------|
| vacationTypeId | int  | Vacation type Id |

## Responses

200 404 500



## Example

### Example

#### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-type/1
```

#### Response

```
{
  "id": 1,
  "name": "name",
  "issueTypeId": "1",
  "vacationKindType": "ONE_TIME",
  "validThrough": false,
  "hasDayLimit": false,
  "carryOver": false,
  "description": "",
  "vacationDocumentPeriodType": "DAILY"
}
```

### Get vacation types list

Returns information about all vacation type.

#### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-type
```

#### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

#### Responses

**200** **500**

#### Example

### Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-type
```

### Response

```
[
  {
    "id": 1,
    "name": "vacation type 1",
    "issueTypeId": "1",
    "vacationKindType": "ONE_TIME",
    "validThrough": false,
    "hasDayLimit": false,
    "carryOver": false,
    "description": "",
    "vacationDocumentPeriodType": "DAILY"
  },
  {
    "id": 2,
    "name": "vacation type 2",
    "issueTypeId": "2",
    "vacationKindType": "RECURRING",
    "validThrough": false,
    "hasDayLimit": true,
    "carryOver": true,
    "periodDefinition": {
      "type": "MONTHS",
      "value": 12,
      "start": "2018-10-10"
    },
    "description": "",
    "vacationDocumentPeriodType": "DAILY_OR_HALF_DAILY"
  }
]
```

#### ▼ [Get vacation type update information](#)

Returns information required for update about specific vacation type.

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-type/
{vacationTypeId}/update-info
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## Path params

| Name           | Type | Description      |
|----------------|------|------------------|
| vacationTypeId | int  | Vacation type Id |

## Responses

**200** **404** **500**

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/vacation-type/1
/update-info
```

### Response

```
{
  "anyInstanceHasAvailableHalfDay": false,
  "anyInstanceHasNonZeroAvailableMinutes": false
}
```

## POST

▼ Create vacation type

Creates and returns new vacation type.

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-type
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## JSON body fields

| Name | Type | Description |
|------|------|-------------|
|------|------|-------------|

|              |              |                              |
|--------------|--------------|------------------------------|
| project      | Project      | Project details object.      |
| vacationType | VacationType | First Vacation Type details. |

Project

| Name        | Type              | Description   |
|-------------|-------------------|---|
| name        | String            | Name of a new project.  |
| key         | String            | Key of a new project (all naming rules applies just like in other Jira projects). |
| description | String (Optional) | Description of a new project.   |
| lead        | String            | Lead user name of a new project.  |
| groupHr     | String            | Group name that will be assigned as HR group.                                     |

VacationType

| Name                       | Type                                       | Description  |
|----------------------------|--|--|
| name                       | String                                     | Name of new vacation type (this is also name for issue type associated with that new vacation type).                               |
| description                | String (optional)                          | Description of vacation type.  |
| kindType                   | KindType                                   | Kind of new vacation type.   |
| vacationDocumentPeriodType | VacationDocumentPeriodType                 | Type of available vacation document period.  |
| periodDefinition           | Object (required if kindType is RECURRING) | PeriodDefinition object for recurring vacation type.   |
| hasCarryOver               | bool (required if kindType is RECURRING)   | Specifies if unused days will be transferred to new vacation periods.  |
| hasEndDate                 | bool (required if kindType is ONE_TIME)    | Specifies if assigned vacation days will be having expiration date.  |
| hasDayLimit                | bool (required if kindType is ONE_TIME)    | Specifies if there will be limit of days that user can use in this vacation type.  |
| defaultDayLimit            | int  | Default vacation days limit for vacation type. This value will be suggested when HR will be assigning new vacation days for users. |

PeriodDefinition

| Name      | Type                 | Description  |
|-----------|----------------------|--|
| type      | PeriodDefinitionType | Type of recurring period.<br>Currently only monthly period is available. |
| value     | int                  | Number of months in recurring period.<br>Valid values: 1, 2, 3, 4, 6, 12 |
| startDate | Date                 | Date of when period will start. ISO 8601 format.                         |

VacationDocumentPeriodType

| Name  | Type | Description                       |
|-------|------|-----------------------------------|
| DAILY | Enum | Vacation type for daily vacations |

|                     |      |  |
|---------------------|------|--|
| DAILY_OR_HALF_DAILY | Enum | Vacation type for daily and half daily vacations |
| DAILY_OR_HOURLY     | Enum | Vacation type for hourly vacations               |

#### KindType

| Name      | Type | Description                                |
|-----------|------|--|
| ONE_TIME  | Enum | Vacation kind type for one time vacations  |
| RECURRING | Enum | Vacation kind type for recurring vacations |

#### PeriodDefinitionType

| Name   | Type | Description                      |
|--------|------|----------------------------------|
| MONTHS | Enum | Type of monthly recurring period |

## Response

201 400 500

## Example

▼ Example

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/vacation-type
```

### Body

```
{
  "name": "Vacation type name",
  "kindType": "ONE_TIME",
  "hasEndDate": false,
  "hasDayLimit": false,
  "vacationDocumentPeriodType": "DAILY"
}
```

## Response

```
{
  "id": 2,
  "name": "Vacation type name",
  "issueTypeId": "10101",
  "kindType": "ONE_TIME",
  "periodDefinition": {},
  "hasEndDate": false,
  "hasDayLimit": false,
  "hasCarryOver": false,
  "vacationDocumentPeriodType": "DAILY"
}
```

## PUT

### Update vacation type

Updates and returns vacation type.

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-type/
{vacationTypeId}
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### Path params

| Name           | Type | Description      |
|----------------|------|------------------|
| vacationTypeId | int  | Vacation type Id |

### JSON body fields

| Name                           | Type                       | Description   |
|--------------------------------|----------------------------|---|
| vacationDocumentPeriodType     | VacationDocumentPeriodType | Type of vacation requests period.   |
| changeHalfDaysToFullDays       | Boolean (optional)         | When changing vacation type from "daily or half-daily" to "daily only", this option allows to convert half day left over to full day  |
| minutesCountToChangeToFullDays | Integer (optional)         | When changing vacation type from "daily or hourly" to "daily only", this option allows to specify how many minutes is required to grant full day off when somebody has spare minutes or hours |

|                                |                    |  |
|--------------------------------|--------------------|--|
| minutesCountToChangeToHalfDays | Integer (optional) | When changing vacation type from "daily or hourly" to "daily or half-daily", this option allows to specify how many minutes is required to grant half day off when somebody has spare minutes or hours |
| minutesCountToGetFromHalfDays  | Integer (optional) | When changing vacation type from "daily or half-daily" to "daily or hourly", this option allows to specify how many minutes is granted when someone has available half day                             |

#### VacationDocumentPeriodType

| Name                | Type | Description                                      |
|---------------------|------|--|
| DAILY               | Enum | Vacation type for daily vacations                |
| DAILY_OR_HALF_DAILY | Enum | Vacation type for daily and half daily vacations |
| DAILY_OR_HOURLY     | Enum | Vacation type for hourly vacations               |

## Responses

200 400 404 500

### Example

▼ Example

#### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacation-type/1
```

#### Body

```
{
  "vacationDocumentPeriodType": "DAILY"
}
```

#### Response

```
{
  "id": 1,
  "name": "vacation type 1",
  "issueTypeId": "1",
  "vacationKindType": "ONE_TIME",
  "validThrough": false,
  "hasDayLimit": false,
  "carryOver": false,
  "description": "",
  "vacationDocumentPeriodType": "DAILY"
}
```

## DELETE

### ▼ Delete vacation type

Deletes vacation type.

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/vacation-type/  
{vacationTypeId}
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### Path params

| Name           | Type | Description      |
|----------------|------|------------------|
| vacationTypeId | int  | Vacation type Id |

### Responses

**204** **404** **500**

#### ▼ Example

#### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/vacation-type/1
```

## Teams

### Overview

Vacation Manager's REST API provides team API which allows you to obtain:

- [create new team](#)
- [add member to team](#)
- [update team](#)
- [delete team](#)
- [remove member from team](#)
- [get teams list](#)
- [get single team](#)
- [get team members](#)

### Methods:

## GET

### ▼ Get teams list

Get list of all teams.



## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/team
```

## Permissions

ADMIN USER SUPERVISOR HR

## Responses

200 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/team
```

### Request

```
[
  {
    "id": 1,
    "name": "team 1",
    "description": "description 1"
  },
  {
    "id": 2,
    "name": "team 2",
    "description": "description 2"
  },
  {
    "id": 3,
    "name": "team 3",
    "description": "description 3"
  }
]
```

▼ Get single team

Get information about existing team members.

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/team/{teamId}
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path params

| Name   | Type | Description |
|--------|------|-------------|
| teamId | int  | Team ID     |

## Responses

200 404 500

Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/team/1
```

## Response

```
{
  "id": 1,
  "name": "teamName",
  "description": "description",
  "users": [
    {
      "id": 1,
      "key": "user1",
      "name": "user 1",
      "displayName": "User 1",
      "email": "user1@mail.com",
      "active": true
    },
    {
      "id": 2,
      "key": "user2",
      "name": "user 2",
      "displayName": "User 2",
      "email": "user2@mail.com",
      "active": true
    },
    {
      "id": 3,
      "key": "user3",
      "name": "user 3",
      "displayName": "User 3",
      "email": "user3@mail.com",
      "active": true
    }
  ],
  "supervisors": [
    {
      "id": 4,
      "key": "supervisor1",
      "name": "supervisor 1",
      "displayName": "Supervisor 1",
      "email": "supervisor1@mail.com",
      "active": true
    }
  ]
}
```

### ▼ [Get team members](#)

Get information about existing team.

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/team/{teamId}/member
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## Path params

| Name   | Type | Description |
|--------|------|-------------|
| teamId | int  | Team ID     |

## Responses

**200** **400** **500**

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/team/1/member
```

### Response

```
{
  "id": 1,
  "name": "teamName",
  "description": "description",
  "user": {
    "users": [
      {
        "id": 1,
        "userId": 10,
        "role": "USER",
        "teamId": 1
      },
      {
        "id": 2,
        "userId": 20,
        "role": "USER",
        "teamId": 1
      }
    ]
  }
}
```

```
        "id": 3,
        "userId": 30,
        "role": "USER",
        "teamId": 1
      }
    ],
    "groups": [
      {
        "id": 7,
        "groupName": "users",
        "role": "USER",
        "teamId": 1
      }
    ]
  },
  "supervisor": {
    "users": [
      {
        "id": 4,
        "userId": 40,
        "role": "SUPERVISOR",
        "teamId": 1
      },
      {
        "id": 5,
        "userId": 50,
        "role": "SUPERVISOR",
        "teamId": 1
      },
      {
        "id": 6,
        "userId": 60,
        "role": "SUPERVISOR",
        "teamId": 1
      }
    ],
    "groups": [
      {
        "id": 8,
        "groupName": "supervisors",
        "role": "SUPERVISOR",
        "teamId": 1
      }
    ]
  }
}
```

**POST**

▼ Create team

Creates new team.

## Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/team
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## JSON body fields

| Name        | Type              | Description                             |
|-------------|-------------------|---|
| name        | String            | Team name, must be unique for each team |
| description | String (Optional) | Team description                        |

## Responses

**201** **400** **500**

## Example

▼ Example

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/team
```

### Body

```
{
  "name": "Team name",
  "description": "Team Description"
}
```

### Response

```
{
  "id":
  "name": "Team name",
  "description": "Team Description"
}
```

▼ [Add member to team](#)

Add member to team.

## Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/team/{teamId}/member
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## Path params

| Name   | Type | Description |
|--------|------|-------------|
| teamId | int  | Team id     |

## JSON body fields

| Name | Type       | Description                                    |
|------|------------|--|
| type | MemberType | Type of member                                 |
| name | String     | Name of the member you want to add to the team |
| role | MemberRole | Role for member                                |

MemberType

| Name  | Type | Description |
|-------|------|-------------|
| USER  | Enum | User type   |
| GROUP | Enum | Group type  |

MemberRole

| Name       | Type | Description                |
|------------|------|----------------------------|
| USER       | Enum | User role for member       |
| SUPERVISOR | Enum | Supervisor role for member |

## Responses

**201** **400** **404** **500**

## Example

▼ [Example](#)

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/team/1/member
```

### Add user to team

#### Body

```
{
  "type": "USER",
  "name": "user 1",
  "role": "USER"
}
```

#### Response

```
{
  "id": 1,
  "userId": 10000,
  "role": "USER",
  "teamId": 1
}
```

### Add group to team

#### Body

```
{
  "type": "GROUP",
  "name": "users",
  "role": "USER"
}
```



## Response

```
{
  "id": 6,
  "groupName": "users",
  "role": "USER",
  "teamId": 1
}
```

## PATCH

### Update team

Update existing team.

### Path

```
PATCH {jira_baseurl}/rest/vacation-manager/1.0/team/{teamId}
```

### Permissions

ADMIN USER SUPERVISOR HR

### Path params

| Name   | Type | Description |
|--------|------|-------------|
| teamId | int  | Team ID     |

### JSON body fields

| Name        | Type              | Description               |
|-------------|-------------------|---------------------------|
| name        | String            | Team name, must be unique |
| description | String (Optional) | Team description          |

### Responses

200 400 404 500

### Example

Example

### Path

```
PATCH {jira_baseurl}/rest/vacation-manager/1.0/team/1
```

### Body

```
{  
  "name": "new team name"  
}
```

### Request

```
{  
  "id": 1,  
  "name": "new team name",  
  "description": "Team Description"  
}
```

## DELETE

### ▼ Delete team

Delete existing team.

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/team/{teamId}
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### Path params

| Name   | Type | Description |
|--------|------|-------------|
| teamId | int  | Team ID     |

### Responses

**204** **404** **500**

### Example

#### ▼ Example

### Request

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/team/1
```

#### Remove member from team

Remove member from team.

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/team/{teamId}/member
```

### Permissions

ADMIN USER SUPERVISOR HR

### Path params

| Name   | Type | Description |
|--------|------|-------------|
| teamId | int  | Team ID     |

### JSON body fields

| Name | Type   | Description   |
|------|--------|---|
| type | String | Type of member to remove <ul style="list-style-type: none"><li>USER</li><li>GROUP</li></ul>       |
| name | String | Name of the member you want to remove from the team   |
| role | String | Role for member to remove <ul style="list-style-type: none"><li>USER</li><li>SUPERVISOR</li></ul> |

### Responses

204 404 500

### Example

#### Example

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/team/1/member
```

### Body

```
{
  "type": "GROUP",
  "name": "users",
  "role": "USER"
}
```

## Holidays

### Overview

Vacation Manager's REST API provides holiday API which allows you to:

- [Add holiday](#)
- [Update holiday](#)
- [Delete holiday from scheme](#)
- [Get holidays list from scheme](#)

### Methods:

## GET

### ▼ [Get holidays list from scheme](#)

Returns list of holidays assigned to given holiday scheme.

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/{schemeId}
/holiday
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### Path params

| Name     | Type | Description |
|----------|------|-------------|
| schemeld | int  | scheme ID   |

### Response

**200** **404** **500**

### Example

▼ [Example](#)

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/1/holiday
```

### Response

```
[
  {
    "id": 1,
    "date": "1970-12-25",
    "yearly": true,
    "description": "Christmas",
    "holidaySchemeId": 1
  },
  {
    "id": 2,
    "date": "1970-01-01",
    "yearly": true,
    "description": "New year",
    "holidaySchemeId": 1
  }
]
```

## POST

### ▼ Add holiday

Returns and creates new holiday in holiday scheme.

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/{schemeId}/holiday
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### Path params

| Name     | Type | Description |
|----------|------|-------------|
| schemeld | int  | scheme ID   |

### JSON body fields

| Name        | Type    | Description   |
|-------------|---------|---|
| date        | String  | Holiday date. ISO 8601 format.  |
| yearly      | Boolean | true - holiday is repeated yearly,<br>false - holiday occurs one time, on selected date |
| description | String  | Holiday description   |

## Responses

200 400 404 500

## Example

▼ Example

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/1/holiday
```

### Body

```
{
  "date": "2018-12-25",
  "yearly": true,
  "description": "Christmas"
}
```

### Response

```
{
  "id": 1,
  "date": "2018-12-25",
  "yearly": true,
  "description": "Christmas",
  "holidaySchemeId": 1
}
```

## PATCH

▼ Update holiday

Returns and Update existing holiday.

### Path

```
PATCH {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/  
{schemeId}/holiday/{holidayId}
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path params

| Name      | Type | Description |
|-----------|------|-------------|
| schemeld  | int  | scheme ID   |
| holidayId | int  | holiday ID  |

## JSON body fields

| Name        | Type    | Description   |
|-------------|---------|---|
| date        | String  | Holiday date. ISO 8601 format.  |
| yearly      | Boolean | true - holiday is repeated yearly,<br>false - holiday occurs one time, on selected date |
| description | String  | Holiday description   |

## Responses

200 400 404 500

## Example

▼ Example

### Path

```
PATCH {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/1  
/holiday/1
```

### Body

```
{  
  "date": "2019-01-01",  
  "yearly": true,  
  "description": "New year"  
}
```

### Response

```
{
  "id": 1,
  "date": "2019-01-01",
  "yearly": true,
  "description": "New year",
  "holidaySchemeId": 1
}
```

## DELETE

### ▼ Delete holiday from scheme

Deletes existing holiday.

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/
{schemeId}/holiday/{holidayId}
```

### Permissions

ADMIN USER SUPERVISOR HR

### Path params

| Name      | Type | Description |
|-----------|------|-------------|
| schemeld  | int  | scheme ID   |
| holidayId | int  | holiday ID  |

### Responses

204 404 500

### Example

#### ▼ Example

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/1
/holiday/1
```

## Holiday schemes



## Overview

Vacation Manager's REST API provides holiday scheme API which allows you to:

- [Create holiday scheme](#)
- [Update holiday scheme](#)
- [Assign workweek scheme to users](#)
- [Delete holiday scheme](#)
- [Get holiday schemes list](#)

## Methods:

### GET

#### ✓ [Get holiday schemes list](#)

Returns list of holiday schemes.

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### Responses

**200** **500**

### Example

#### ✓ [Example](#)

#### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme
```

## Response

```
[
  {
    "id": 1,
    "name": "Default holiday scheme"
  },
  {
    "id": 2,
    "name": "Holiday scheme 1"
  },
  {
    "id": 3,
    "name": "Holiday scheme 2"
  }
]
```

## POST

### ▼ Create holiday scheme

Creates and return new holiday scheme.

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### JSON body fields

| Name | Type   | Description   |
|------|--------|---|
| name | String | Holiday scheme name, must be unique for each scheme |

### Responses

**201** **400** **500**

### Example

▼ Example

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme
```

### Body

```
{
  "name": "Holiday scheme name"
}
```

### Response

```
{
  "id": 2
  "name": "Holiday scheme name"
}
```

## PATCH

### Update holiday scheme

Updates and return existing holiday scheme.

### Path

```
PATCH {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/
{schemeId}
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path params

| Name     | Type | Description       |
|----------|------|-------------------|
| schemeld | int  | Holiday scheme ID |

## JSON body fields

| Name | Type   | Description             |
|------|--------|-------------------------|
| name | String | Holiday scheme new name |

## Responses

200 400 404 500

### Example

▼ Example

#### Path

```
PATCH {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/2
```

#### Body

```
{  
  "name": "Holiday scheme name"  
}
```

#### Response

```
{  
  "id": 2,  
  "name": "Holiday scheme name"  
}
```

## PUT

▼ [Assign holiday scheme to users](#)

Assigns holiday scheme to users.

### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacmanuser/holidays/bulk
```

## Permissions

ADMIN USER SUPERVISOR HR

### JSON body fields

| Name     | Type    | Description                                      |
|----------|---------|--|
| schemeld | Integer | Id of holiday scheme you want to assign to users |

|           |               |   |
|-----------|---------------|---|
| userNames | Array<String> | Names of users you want to assign holiday scheme to |
|-----------|---------------|---|

## Responses

200 400 404 500

### Example

▼ Example

#### Path

```
PUT {jira_baseurl}/rest/vacation-manager/1.0/vacmanuser/holidays/bulk
```

#### Body

```
{
  "schemeId": 1,
  "userNames": ["john", "jack"]
}
```

#### Response

```
{
  "userNames": [
    "john",
    "jack"
  ],
  "schemeId": 1
}
```

## DELETE

▼ Delete holiday scheme

Deletes existing holiday scheme.

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/{schemeId}
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path params

| Name     | Type    | Description                                      |
|----------|---------|--|
| schemeld | Integer | Id of holiday scheme you want to assign to users |

## Responses

204 404 500

## Example

▼ Example

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/holiday-scheme/1
```

## Workweek schemes

### Overview

Vacation Manager's REST API provides workweek scheme API which allows you to obtain:

- [create workweek scheme](#)
- [update workweek scheme](#)
- [assign workweek scheme to users](#)
- [delete Workweek Scheme](#)
- [get workweek schemes list](#)
- [get single workweek scheme](#)
- [get workweek scheme for user](#)

### Methods:

## GET

▼ [Get workweek schemes list](#)

Returns list of workweek schemes.

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme
```

## Permissions

ADMIN USER SUPERVISOR HR

## Responses

200 400 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme
```

## Response

```
[
  {
    "id": 1,
    "name": "Default workweek scheme",
    "workweekMinutes": {
      "TUESDAY": 480,
      "SUNDAY": 0,
      "MONDAY": 480,
      "WEDNESDAY": 480,
      "FRIDAY": 480,
      "THURSDAY": 480,
      "SATURDAY": 0
    },
    "type": "SEPARATE_WORK_MINUTES_FOR_EACH_DAY"
  },
  {
    "id": 2,
    "name": "Workweek scheme 2",
    "workweekMinutes": {
      "TUESDAY": 480,
      "SUNDAY": 0,
      "MONDAY": 480,
      "WEDNESDAY": 480,
      "FRIDAY": 480,
      "THURSDAY": 480,
      "SATURDAY": 0
    },
    "type": "SAME_WORK_MINUTES_FOR_ALL_DAYS"
  },
  {
    "id": 3,
    "name": "Workweek scheme 3",
    "workweekMinutes": {
      "TUESDAY": 480,
      "SUNDAY": 0,
      "MONDAY": 480,
      "WEDNESDAY": 400,
      "FRIDAY": 480,
      "THURSDAY": 400,
      "SATURDAY": 0
    },
    "type": "SEPARATE_WORK_MINUTES_FOR_EACH_DAY"
  }
]
```



▼ [Get single workweek scheme](#)

Returns information about existing workweek scheme.

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme/  
{schemeId}
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## Path param

| Name     | Type | Description |
|----------|------|-------------|
| schemeld | int  | Scheme ID   |

## Responses

**200** **400** **500**

## Example

▼ [Example](#)

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme/1
```

### Response

```
{  
  "id": 1,  
  "name": "Default workweek scheme",  
  "workweekMinutes": {  
    "TUESDAY": 480,  
    "SUNDAY": 0,  
    "MONDAY": 480,  
    "WEDNESDAY": 480,  
    "FRIDAY": 480,  
    "THURSDAY": 480,  
    "SATURDAY": 0  
  },  
  "type": "SEPARATE_WORK_MINUTES_FOR_EACH_DAY"  
}
```

▼ [Get workweek scheme for user](#)

Returns workweek scheme used by given user.

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme?userName={userName}
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## Query params

| Name     | Type   | Description |
|----------|--------|-------------|
| userName | String | User Name   |

## Responses

**200** **400** **500**

## Example

▼ [Example](#)

### Path

```
PATH{jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme?userName=user1
```

### Response

```
{
  "id": 1,
  "name": "Default workweek scheme",
  "workweekMinutes": {
    "TUESDAY": 480,
    "SUNDAY": 0,
    "MONDAY": 480,
    "WEDNESDAY": 480,
    "FRIDAY": 480,
    "THURSDAY": 480,
    "SATURDAY": 0
  },
  "type": "SEPARATE_WORK_MINUTES_FOR_EACH_DAY"
}
```

## POST

### ▼ Create new workweek scheme

Creates and returns new workweek scheme.

### Path

**Request /workweek-scheme**

```
POST {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme
```

### Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

### JSON body fields

| Name            | Type                    | Description                           |
|-----------------|-------------------------|---------------------------------------|
| name            | String                  | Workweek scheme name                  |
| workweekMinutes | Map<DayOfWeek, Integer> | Day of week : working time in minutes |
| type            | WorkweekSchemeType      | Type of work week scheme              |

DayOfWeek

| Name      | Type | Description |
|-----------|------|-------------|
| MONDAY    | Enum | Monday      |
| TUESDAY   | Enum | Tuesday     |
| WEDNESDAY | Enum | Wednesday   |
| THURSDAY  | Enum | Thursday    |
| FRIDAY    | Enum | Friday      |
| SATURDAY  | Enum | Saturday    |
| SUNDAY    | Enum | Sunday      |

WorkweekSchemeType

| Name                               | Type | Description                                |
|------------------------------------|------|--|
| SAME_WORK_MINUTES_FOR_ALL_DAYS     | Enum | Type of the same work minutes for all days |
| SEPARATE_WORK_MINUTES_FOR_EACH_DAY | Enum | Type of separate work minutes for each day |

### Responses

**201** **400** **500**

### Example

▼ Example

### Path

```
POST {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme
```

### Body

```
{
  "name": "Workweek scheme name",
  "workweekMinutes": {
    "FRIDAY": 480,
    "MONDAY": 400,
    "WEDNESDAY": 800,
    "SUNDAY": 0,
    "TUESDAY": 480,
    "SATURDAY": 0,
    "THURSDAY": 400
  },
  "type": "SEPARATE_WORK_MINUTES_FOR_EACH_DAY"
}
```

### Response

```
{
  "id": 101,
  "name": "Workweek scheme name",
  "workweekMinutes": {
    "TUESDAY": 480,
    "SUNDAY": 0,
    "MONDAY": 400,
    "WEDNESDAY": 800,
    "FRIDAY": 480,
    "THURSDAY": 400,
    "SATURDAY": 0
  },
  "type": "SEPARATE_WORK_MINUTES_FOR_EACH_DAY"
}
```

## PATCH

### Update workweek scheme

Update and return existing workweek scheme.

### Path

```
PATCH {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme/  
{schemeId}
```

## Permissions

**ADMIN** **USER** **SUPERVISOR** **HR**

## Path params

| Name     | Type | Description |
|----------|------|-------------|
| schemeld | int  | Scheme ID   |

## JSON body fields

| Name            | Type                    | Description                           |
|-----------------|-------------------------|---------------------------------------|
| name            | String                  | Workweek scheme name                  |
| workweekMinutes | Map<DayOfWeek, Integer> | Day of week : working time in minutes |
| type            | WorkweekSchemeType      | Type of work week scheme              |

### DayOfWeek

| Name      | Type | Description |
|-----------|------|-------------|
| MONDAY    | Enum | Monday      |
| TUESDAY   | Enum | Tuesday     |
| WEDNESDAY | Enum | Wednesday   |
| THURSDAY  | Enum | Thursday    |
| FRIDAY    | Enum | Friday      |
| SATURDAY  | Enum | Saturday    |
| SUNDAY    | Enum | Sunday      |

### WorkweekSchemeType

| Name                               | Type | Description                                |
|------------------------------------|------|--|
| SAME_WORK_MINUTES_FOR_ALL_DAYS     | Enum | Type of the same work minutes for all days |
| SEPARATE_WORK_MINUTES_FOR_EACH_DAY | Enum | Type of separate work minutes for each day |

## Responses

**200** **400** **404** **500**

### Example

▼ Example

### Path

```
PATCH {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme/2
```

### Body

```
{
  "name": "Workweek scheme new name",
  "workweekMinutes": {
    "FRIDAY": 480,
    "MONDAY": 400,
    "WEDNESDAY": 800,
    "SUNDAY": 0,
    "TUESDAY": 480,
    "SATURDAY": 0,
    "THURSDAY": 400
  },
  "type": "SEPARATE_WORK_MINUTES_FOR_EACH_DAY"
}
```

### Response

```
{
  "id": 2,
  "name": "Workweek scheme new name",
  "workweekMinutes": {
    "FRIDAY": 480,
    "MONDAY": 400,
    "WEDNESDAY": 800,
    "SUNDAY": 0,
    "TUESDAY": 480,
    "SATURDAY": 0,
    "THURSDAY": 400
  },
  "type": "SEPARATE_WORK_MINUTES_FOR_EACH_DAY"
}
```

## DELETE

### ▼ [Delete workweek scheme](#)

Deletes existing workweek scheme.

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme/  
{schemeId}
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path params

| Name     | Type | Description |
|----------|------|-------------|
| schemeld | int  | Scheme ID   |

## Responses

204 404 500

## Example

▼ Example

### Path

```
DELETE {jira_baseurl}/rest/vacation-manager/1.0/workweek-scheme/1
```

## Statistics

### Overview

Vacation Manager's REST API provides Statistics API which allows you to obtain:

- [get instances statistics by logged user](#)
- [get instances statistics by user](#)
- [get instances statistics by vacation type](#)

### Methods:

## GET

▼ [get instances statistics by logged user](#)

Returns instances statistics for logged user

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/statistics/user?  
startDate=ISO_DATE&endDate=ISO_DATE
```

## Permissions

ADMIN USER SUPERVISOR HR

## Query params

| Name      | Type   | Description                 |
|-----------|--------|-----------------------------|
| startDate | String | Start date. ISO 8601 format |
| endDate   | String | End date. ISO 8601 format   |

## Responses

200 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/statistics/user?  
startDate=2011-10-11&endDate=2011-10-21
```



## Response

```
{
  "user": {
    "id": 10000,
    "key": "userKey",
    "name": "userName",
    "displayName": "user",
    "email": "user@mail.com",
    "active": true
  },
  "vacationTypeInstances": {
    "1": {
      "vacationType": {
        "id": 1,
        "name": "daily vacation type",
        "description": "",
        "issueTypeId": "10100",
        "kindType": "ONE_TIME",
        "periodDefinition": {},
        "hasEndDate": true,
        "hasDayLimit": true,
        "hasCarryOver": false,
        "vacationDocumentPeriodType": "DAILY"
      },
      "instances": []
    }
  },
  "freeDays": [
    "2011-10-11",
    "2011-10-12",
    "2011-10-18",
    "2011-10-19"
  ]
}
```

### ▼ [get instances statistics by user](#)

Returns instances statistics for a given user.

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/statistics/user/
{userName}?startDate=ISO_DATE&endDate=ISO_DATE
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path param

| Name     | Type   | Description |
|----------|--------|-------------|
| userName | String | User Name   |

## Query params

| Name      | Type   | Description                 |
|-----------|--------|-----------------------------|
| startDate | String | Start date. ISO 8601 format |
| endDate   | String | End date. ISO 8601 format   |

## Responses

200 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/statistics/user/Jack?
startDate=2011-10-11&endDate=2011-10-21
```

## Response

```
{
  "user": {
    "id": 10000,
    "key": "JackK",
    "name": "Jack",
    "displayName": "user",
    "email": "user@mail.com",
    "active": true
  },
  "vacationTypeInstances": {
    "1": {
      "vacationType": {
        "id": 1,
        "name": "daily vacation type",
        "description": "",
        "issueTypeId": "10100",
        "kindType": "ONE_TIME",
        "periodDefinition": {},
        "hasEndDate": true,
        "hasDayLimit": true,
        "hasCarryOver": false,
        "vacationDocumentPeriodType": "DAILY"
      },
      "instances": []
    }
  },
  "freeDays": [
    "2011-10-11",
    "2011-10-12",
    "2011-10-18",
    "2011-10-19"
  ]
}
```

### ▼ [get instances statistics by vacation type](#)

Returns instances statistics for a given vacation type.

## Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/statistics/vacation-type/
{vacationTypeId}?startDate=ISO_DATE&endDate=ISO_DATE
```

## Permissions

ADMIN USER SUPERVISOR HR

## Path param

| Name           | Type | Description      |
|----------------|------|------------------|
| vacationTypeId | int  | Vacation type ID |

## Query params

| Name      | Type   | Description                 |
|-----------|--------|-----------------------------|
| startDate | String | Start date. ISO 8601 format |
| endDate   | String | End date. ISO 8601 format   |

## Responses

200 500

## Example

▼ Example

### Path

```
GET {jira_baseurl}/rest/vacation-manager/1.0/statistics/vacation-type/1?startDate=2011-10-11&endDate=2011-10-21
```

### Response

```
{
  "vacationType": {
    "id": 1,
    "name": "daily vacation type",
    "description": "",
    "issueTypeId": "10100",
    "kindType": "ONE_TIME",
    "periodDefinition": {},
    "hasEndDate": true,
    "hasDayLimit": true,
    "hasCarryOver": false,
    "vacationDocumentPeriodType": "DAILY"
  },
  "userInstances": {}
}
```