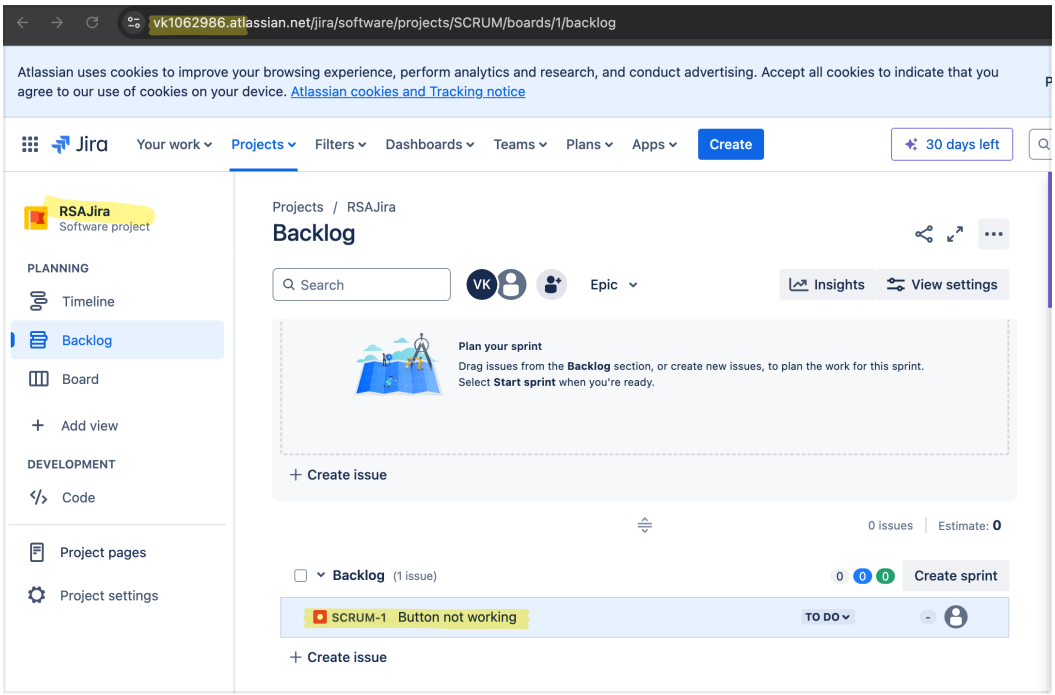
Rest Assured API advanced

Jira

- How to send File Attachments through Rest API calls
- Example used: Jira API to create Bug and attached failed screenshots through Rest APIs

https://vk1062986.atlassian.net/jira/software/projects/SCRUM/boards/1/backlog?selectedIssue=SCRUM-1
https://developer.atlassian.com/cloud/jira/platform/rest/v2/intro/#about



>Basic Auth & OAuth authentication are 2 types used to unlock APIs.

>We use Basic Auth to unlock Jira APIs
https://developer.atlassian.com/cloud/jira/platform/basic-auth-for-rest-apis/

https://id.atlassian.com/manage-profile/security/api-tokens

>Basically when we say Basic Authentication that means we will be able to login using Tokens.
Here we usually call tokens as Bearer tokens

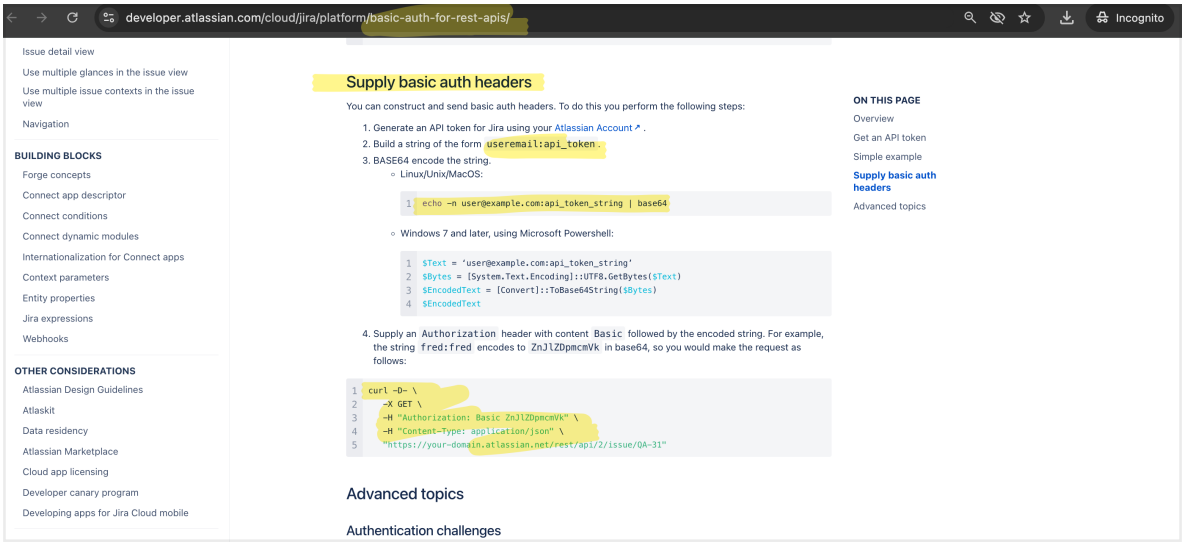
https://id.atlassian.com/manage-profile/security/api-tokens

Secret key for API token created in Jira:

```
ATATT3xFfGF0rvYTdYHyuHwizA8wWuHHUZLqoZq-
IhdZ39Xw6gUFQPnM6SR4XntU8KLLuAlvQ-
F9nsEnZI4YZT2pYCS92Ga0dSf1FyU5EuFhPvYChFp2QIFMxDByKVHxQoQxyi
FxXIlCkpDdhkGqHUmZvS1kEOgzvSjfIH0sSjAefgw5f3Yhcbo=3602714E
```

Supply basic auth headers in Jira

https://developer.atlassian.com/cloud/jira/platform/basic-auth-for-rest-apis/



You can construct and send basic auth headers. To do this you perform the following steps:

1. Generate an API token for Jira using your Atlassian Account.
2. Build a string of the form useremail:api_token.
3. BASE64 encode the string.
    - Linux/Unix/MacOS:

```
echo -n user@example.com:api_token_string | base64
```

    - Windows 7 and later, using Microsoft Powershell:

```
$Text = 'user@example.com:api_token_string'
$Bytes = [System.Text.Encoding]::UTF8.GetBytes($Text)
$EncodedText = [Convert]::ToBase64String($Bytes)
$EncodedText
```

4. Supply an Authorization header with content Basic followed by the encoded string. For example, the string fred:fred encodes to ZnJlZDpmcmVk in base64, so you would make the request as follows:

```
curl -D- \
   -X GET \
   -H "Authorization: Basic ZnJlZDpmcmVk" \
   -H "Content-Type: application/json" \
```

```
   "https://your-domain.atlassian.net/rest/api/2/issue/
QA-31"
```

Actual hands on

- Syntax:

```
useremail:api_token
```

- Actual token:

```
vk1062986@gmail.com:ATATT3xFfGF0rvYTdYHyuHwizA8wWuHHUZLqoZq-
IhdZ39Xw6gUFQPnM6SR4XntU8KLLuAlvQ-
F9nsEnZI4YZT2pYCS92Ga0dSf1FyU5EuFhPvYChFp2QIFMxDByKVHxQoQxyi
FxXIlCkpDdhkGqHUmZvS1kEOgzvSjfIH0sSjAefgw5f3Yhcbo=3602714E
```

- Syntax:

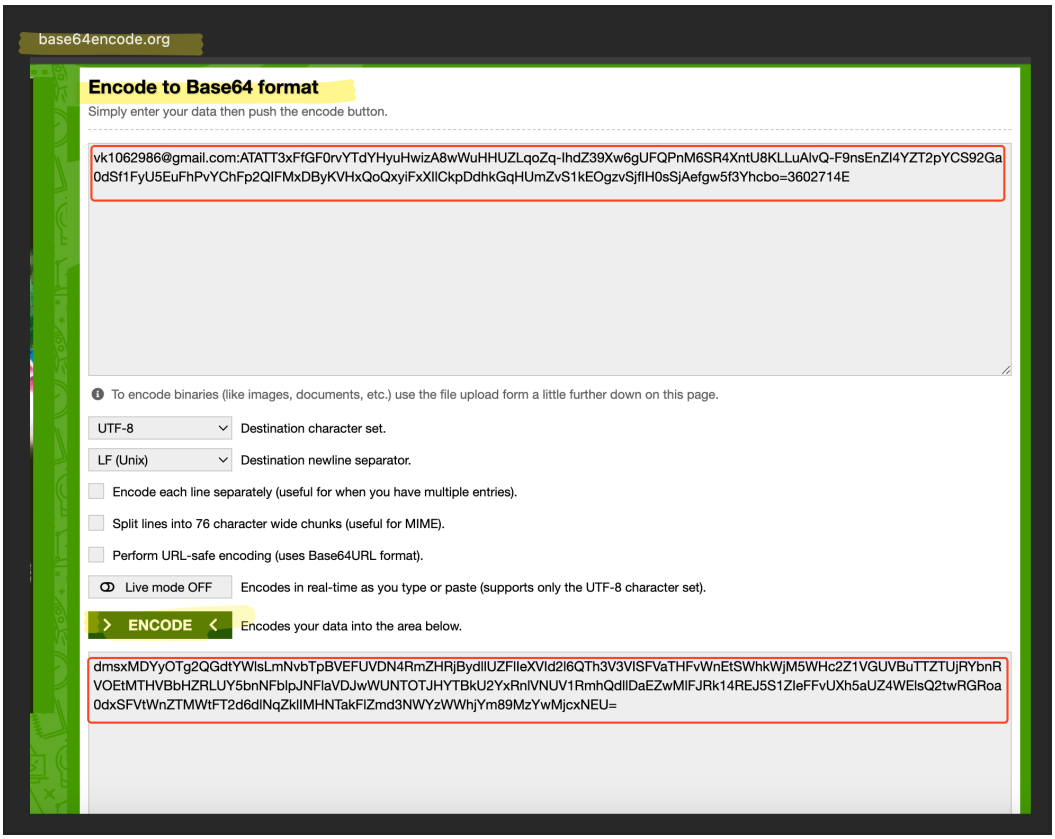
```
echo -n user@example.com:api_token_string | base64
```

- Actual command

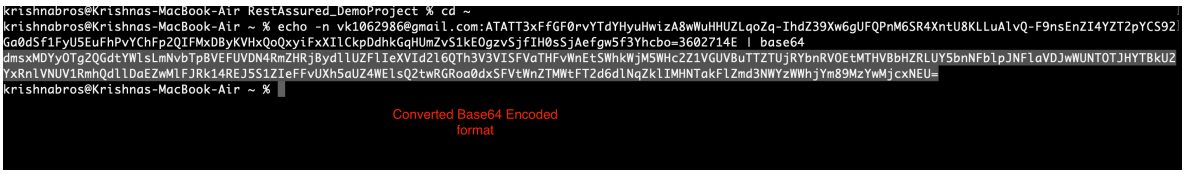
```
echo -n
vk1062986@gmail.com:ATATT3xFfGF0rvYTdYHyuHwizA8wWuHHUZLqoZq-
IhdZ39Xw6gUFQPnM6SR4XntU8KLLuAlvQ-
F9nsEnZI4YZT2pYCS92Ga0dSf1FyU5EuFhPvYChFp2QIFMxDByKVHxQoQxyi
FxXIlCkpDdhkGqHUmZvS1kEOgzvSjfIH0sSjAefgw5f3Yhcbo=3602714E |
base64
```

https://www.base64decode.org/

>We can use online base64 encode & decoder to create base64 encoded format or execute above command in terminal to get base64 encoded format



Converted base64 Encoded format from string format of Api token for MAC:



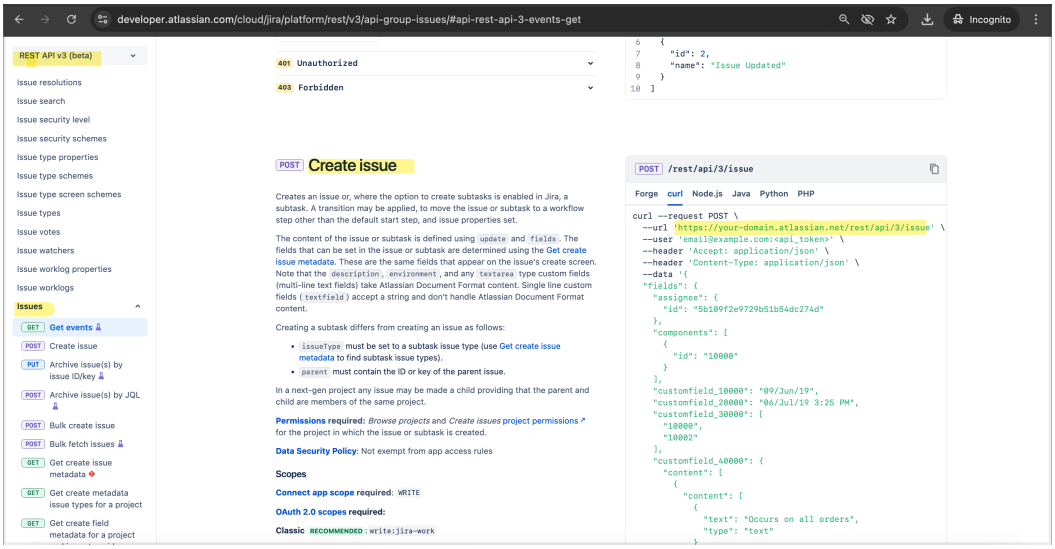

```
dmsxMDYyOTg2QGdtYWlsLmNvbTpBVEFUVDN4RmZHRjBydllUZFlIeXVId2l6
QTh3V3VISFVaTHFvWnEtSWhkWjM5WHc2Z1VGUVBuTTZTUjRYbnRVOEtMTHVB
bHZRLUY5bnNFblpJNFlaVDJwWUNTOTJHYTBkU2YxRnlVNUV1RmhQdllDaEZw
MlFJRk14REJ5S1ZleFFvUXh5aUZ4WElsQ2twRGRoa0dxSFVtWnZTMWtFT2d6
dlNqZklIMHNTakFlZmd3NWYzWWhjYm89MzYwMjcxNEU=
```

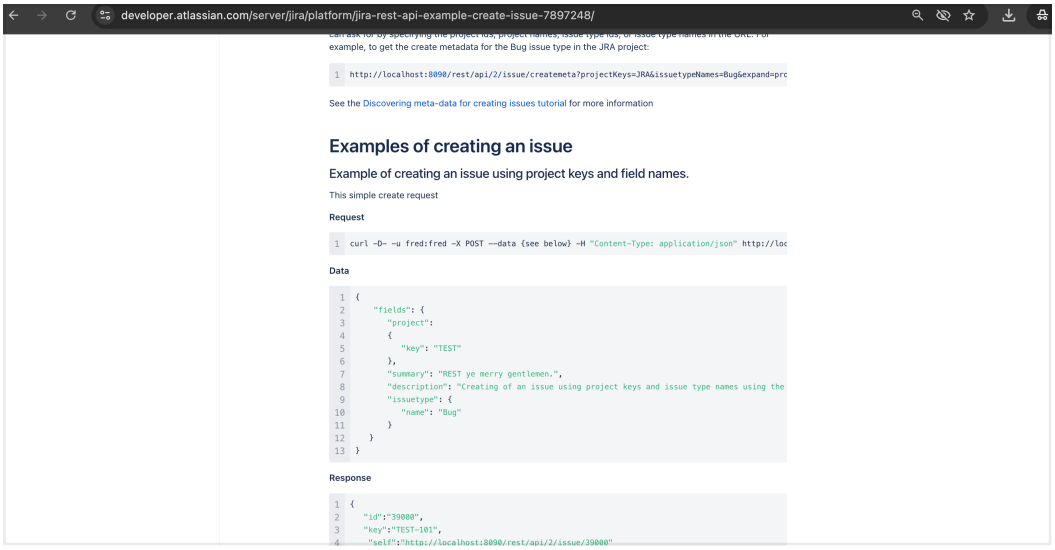
> Above is header we need to pass to Authentication header value of API.

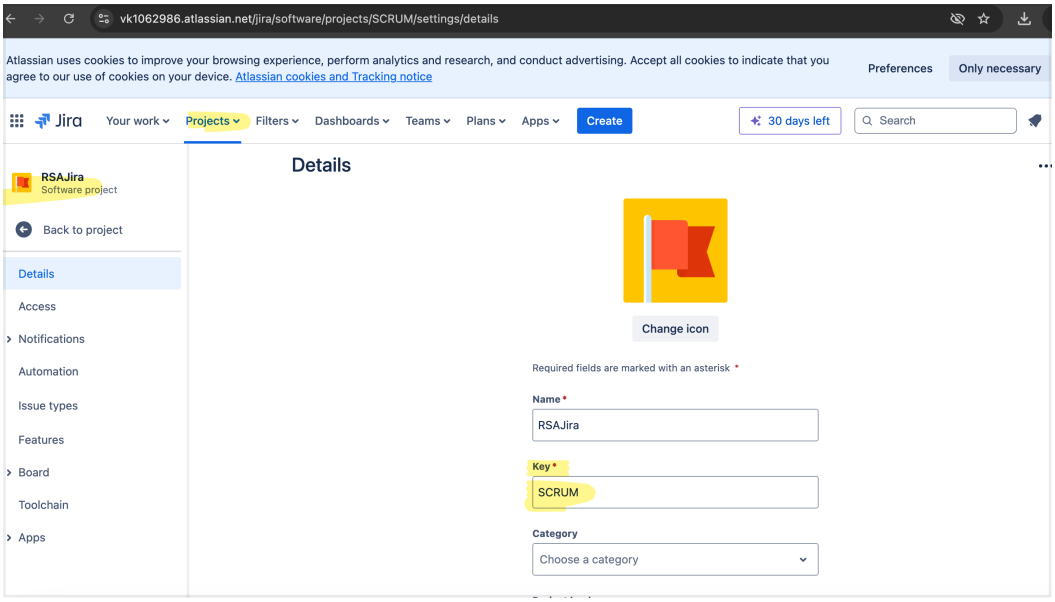
Go to Issues documentation from below link:
https://developer.atlassian.com/cloud/jira/platform/rest/v3/api-group-issues/#api-rest-api-3-events-get

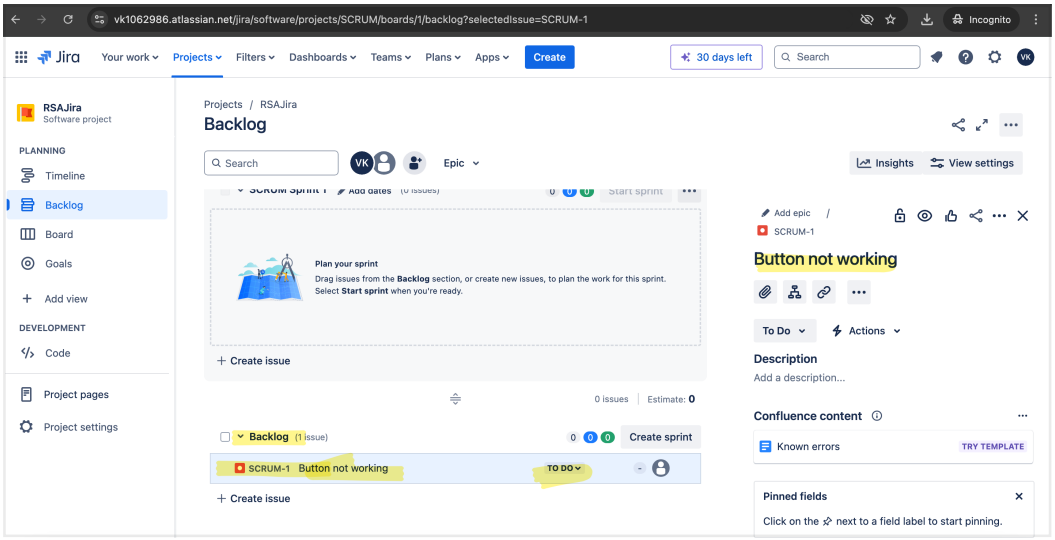
Test:

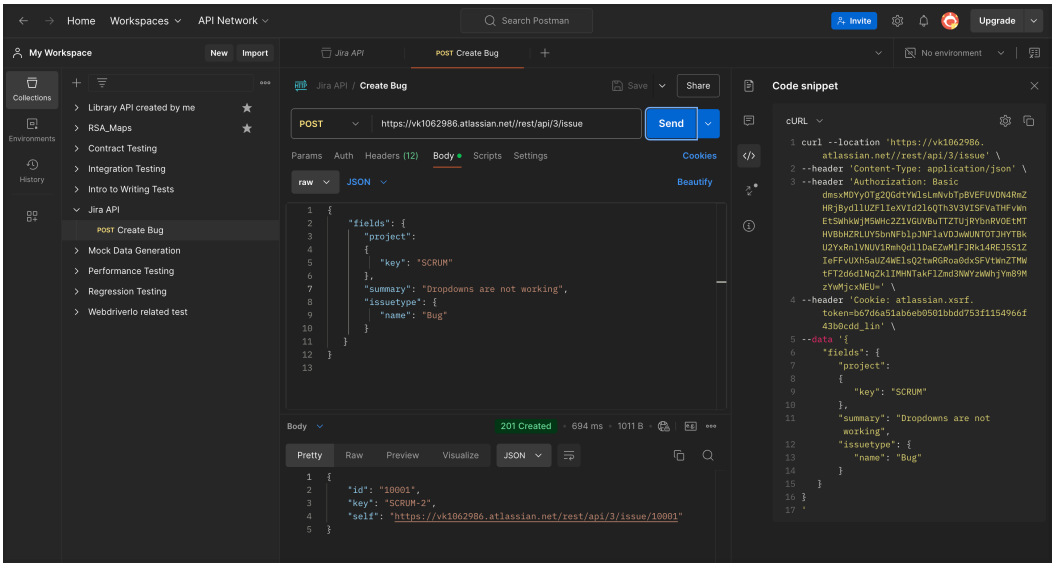
- In postman create issue post request: **POST**

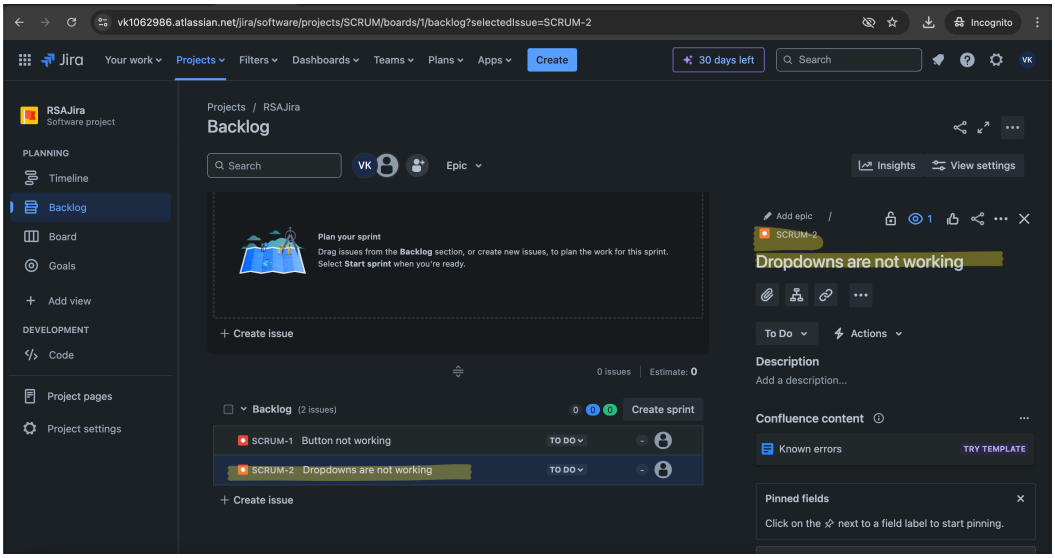

developer.atlassian.com/cloud/jira/platform/rest/v3/api-group-issues/#api-rest-api-3-events-get
Incognito
REST API v3 (beta)
Issue resolutions
Issue search
Issue security level
Issue security schemes
Issue type properties
Issue type schemes
Issue type screen schemes
Issue types
Issue votes
Issue watchers
Issue worklog properties
Issue worklogs
Issues
GET Get events
POST Create issue
PUT Archive issue(s) by issue ID/key
POST Archive issue(s) by JQL
POST Bulk create issue
POST Bulk fetch issues
GET Get create issue metadata
GET Get create metadata issue types for a project
GET Get create field metadata for a project
401 Unauthorized
403 Forbidden
POST Create issue
Creates an issue or, where the option to create subtasks is enabled in Jira, a subtask. A transition may be applied, to move the issue or subtask to a workflow step other than the default start step, and issue properties set.
The content of the issue or subtask is defined using update and fields. The fields that can be set in the issue or subtask are determined using the Get create issue metadata. These are the same fields that appear on the issue's create screen. Note that the description, environment, and any textarea type custom fields (multi-line text fields) take Atlassian Document Format content. Single line custom fields (textfield) accept a string and don't handle Atlassian Document Format content.
Creating a subtask differs from creating an issue as follows:
issueType must be set to a subtask issue type (use Get create issue metadata to find subtask issue types).
parent must contain the ID or key of the parent issue.
In a next-gen project any issue may be made a child providing that the parent and child are members of the same project.
Permissions required: Browse projects and Create issues project permissions for the project in which the issue or subtask is created.
Data Security Policy: Not exempt from app access rules
Scopes
Connect app scope required: WRITE
OAuth 2.0 scopes required:
Classic RECOMMENDED : write:jira-work
POST /rest/api/3/issue
Forge curl Node.js Java Python PHP
curl --request POST \
--url 'https://your-domain.atlassian.net/rest/api/3/issue' \
--user 'email@example.com:<api_token>' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json' \
--data '{
"fields": {
"assignee": {
"id": "5b109f2e9729b51b54dc274d"
},
"components": [
{
"id": "10000"
}
],
"customfield_10000": "09/Jun/19",
"customfield_20000": "06/Jul/19 3:25 PM",
"customfield_30000": [
"10000",
"10002"
],
"customfield_40000": {
"content": [
{
"content": [
{
"text": "Occurs on all orders",
"type": "text"


developer.atlassian.com/server/jira/platform/jira-rest-api-example-create-issue-7897248/
example, to get the create metadata for the Bug issue type in the JRA project:
http://localhost:8090/rest/api/2/issue/createmeta?projectKeys=JRA&issuetypeNames=Bug&expand=prc
See the Discovering meta-data for creating issues tutorial for more information
Examples of creating an issue
Example of creating an issue using project keys and field names.
This simple create request
Request
curl -D- -u fred:fred -X POST --data {see below} -H "Content-Type: application/json" http://loc
Data
{
"fields": {
"project":
{
"key": "TEST"
},
"summary": "REST ye merry gentlemen.",
"description": "Creating of an issue using project keys and issue type names using the
"issuetype": {
"name": "Bug"
}
}
}
Response
{
"id":"39000",
"key":"TEST-101",
"self":"http://localhost:8090/rest/api/2/issue/39000"


vk1062986.atlassian.net/jira/software/projects/SCRUM/settings/details
Atlassian uses cookies to improve your browsing experience, perform analytics and research, and conduct advertising. Accept all cookies to indicate that you agree to our use of cookies on your device. Atlassian cookies and Tracking notice
Preferences
Only necessary
Jira
Your work
Projects
Filters
Dashboards
Teams
Plans
Apps
Create
30 days left
Search
RSAJira
Software project
Back to project
Details
Access
Notifications
Automation
Issue types
Features
Board
Toolchain
Apps
Details
Change icon
Required fields are marked with an asterisk *
Name *
RSAJira
Key *
SCRUM
Category
Choose a category







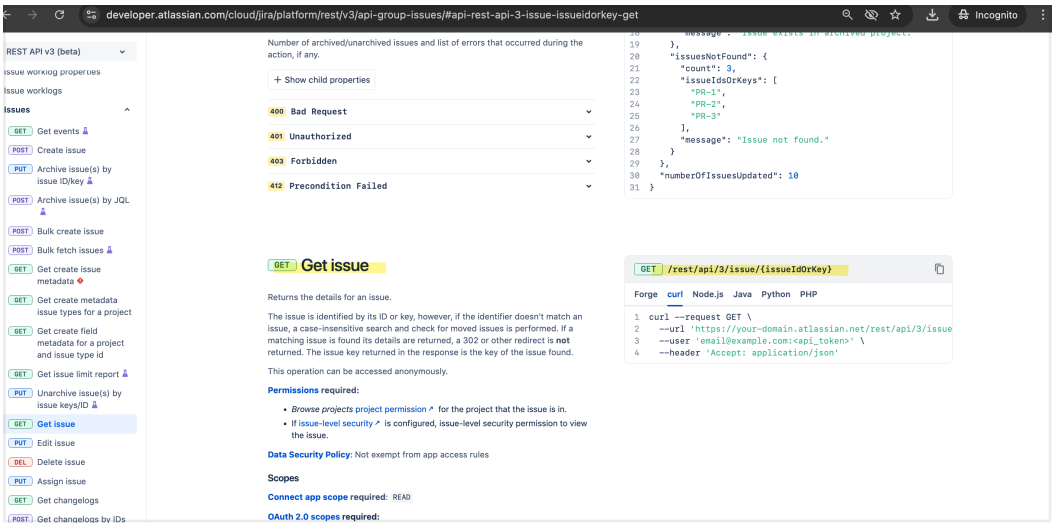
Below is the cURL command to Create Issue or Bug in Jira:

```
curl --location 'https://vk1062986.atlassian.net/rest/api/3/issue' \
```

```
--header 'Content-Type: application/json' \
--header 'Authorization: Basic
dmsxMDYyOTg2QGdtYWlsLmNvbTpBVEFUVDN4RmZHRjBydllUZFlIeXVId2l6
QTh3V3VISFVaTHFvWnEtSWhkWjM5WHc2Z1VGUVBuTTZTUjRYbnRVOEtMTHVB
bHZRLUY5bnNFblpJNFlaVDJwWUNTOTJHYTBkU2YxRnlVNUV1RmhQdllDaEZw
MlFJRk14REJ5S1ZIeFFvUXh5aUZ4WElsQ2twRGRoa0dxSFVtWnZTMWtFT2d6
dlNqZklIMHNTakFlZmd3NWYzWWhjYm89MzYwMjcxNEU=' \
--header 'Cookie:
atlassian.xsrf.token=b67d6a51ab6eb0501bbdd753f1154966f43b0cd
d_lin' \
--data '{
    "fields": {
       "project":
       {
          "key": "SCRUM"
       },
       "summary": "Dropdowns are not working",
       "issuetype": {
          "name": "Bug"
       }
   }
}
'
```

✅ In postman, get the issue : **GET**

```
curl --request GET \
  --url 'https://your-domain.atlassian.net/rest/api/3/issue/{issueIdOrKey}' \
  --user 'email@example.com:<api_token>' \
  --header 'Accept: application/json'
```



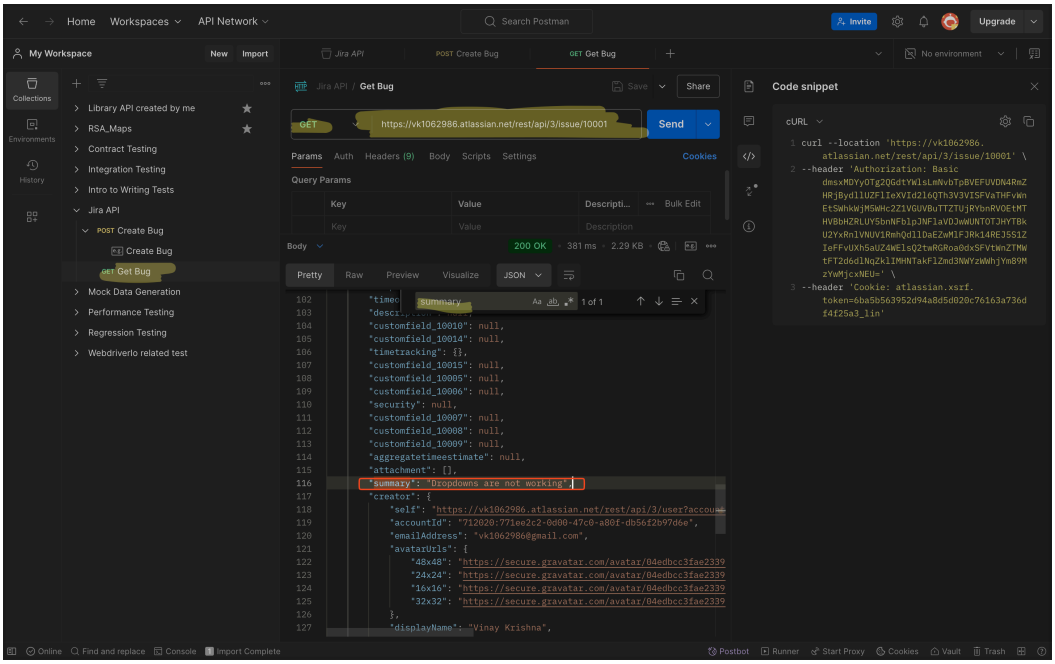
Below is the cURL command to Get issue using id from Jira:

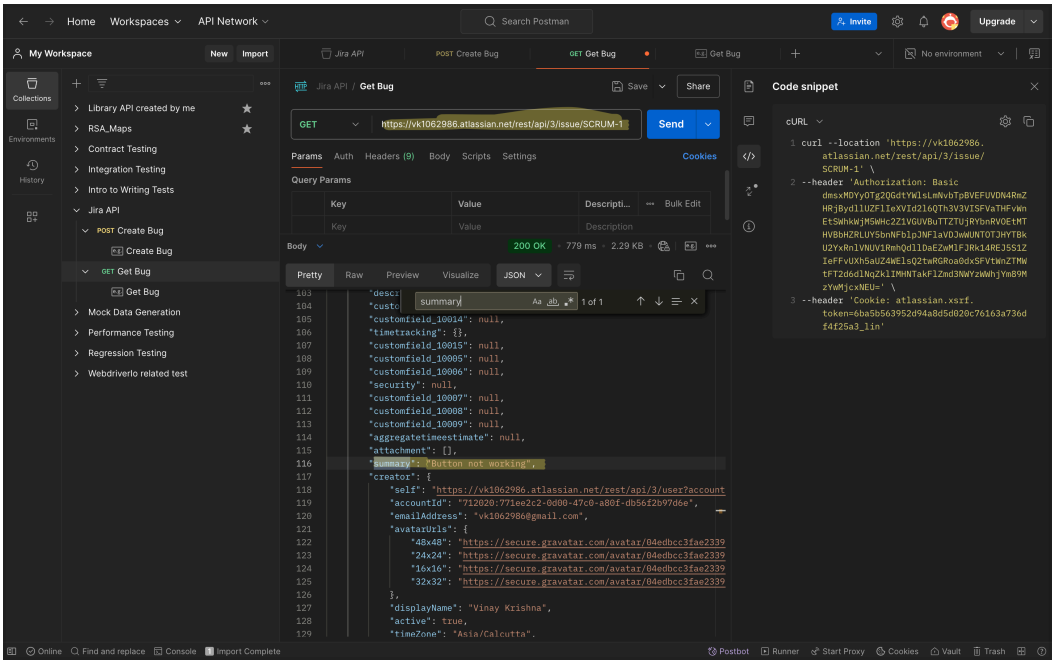
```
curl --location 'https://vk1062986.atlassian.net/rest/api/3/
issue/10001' \
```

```
--header 'Authorization: Basic
dmsxMDYyOTg2QGdtYWlsLmNvbTpBVEFUVDN4RmZHRjBydllUZFlIeXVId2l6
QTh3V3VISFVaTHFvWnEtSWhkWjM5WHc2Z1VGUVBuTTZTUjRYbnRVOEtMTHVB
bHZRLUY5bnNFblpJNFlaVDJwWUNTOTJHYTBkU2YxRnlVNUV1RmhQdllDaEZw
MlFJRk14REJ5S1ZIeFFvUXh5aUZ4WElsQ2twRGRoa0dxSFVtWnZTMWtFT2d6
dlNqZklIMHNTakFlZmd3NWYzWWhjYm89MzYwMjcxNEU=' \
--header 'Cookie:
atlassian.xsrf.token=6ba5b563952d94a8d5d020c76163a736df4f25a
3_lin'
```

Below is the cURL command to Get issue using key from Jira:

```
curl --location 'https://vk1062986.atlassian.net/rest/api/3/
issue/SCRUM-1' \
--header 'Authorization: Basic
dmsxMDYyOTg2QGdtYWlsLmNvbTpBVEFUVDN4RmZHRjBydllUZFlIeXVId2l6
QTh3V3VISFVaTHFvWnEtSWhkWjM5WHc2Z1VGUVBuTTZTUjRYbnRVOEtMTHVB
bHZRLUY5bnNFblpJNFlaVDJwWUNTOTJHYTBkU2YxRnlVNUV1RmhQdllDaEZw
MlFJRk14REJ5S1ZIeFFvUXh5aUZ4WElsQ2twRGRoa0dxSFVtWnZTMWtFT2d6
dlNqZklIMHNTakFlZmd3NWYzWWhjYm89MzYwMjcxNEU=' \
--header 'Cookie:
atlassian.xsrf.token=6ba5b563952d94a8d5d020c76163a736df4f25a
3_lin'
```

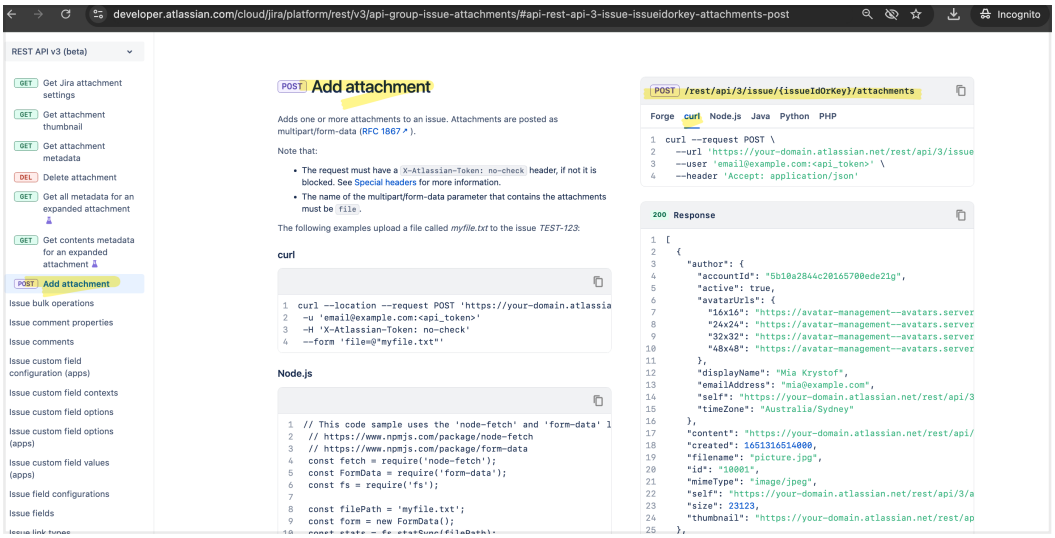


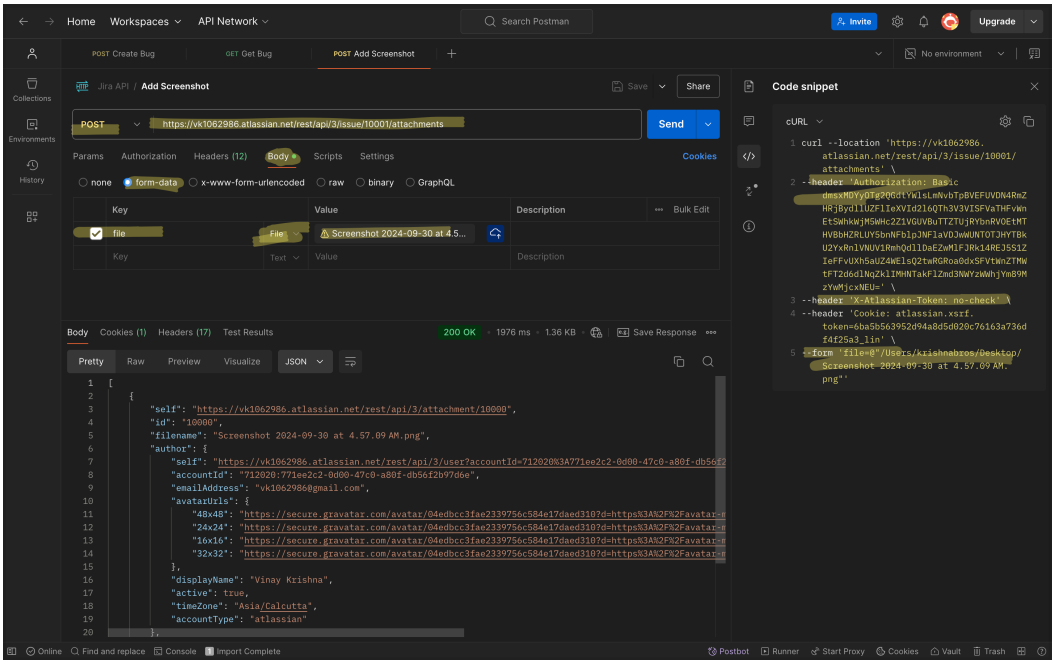


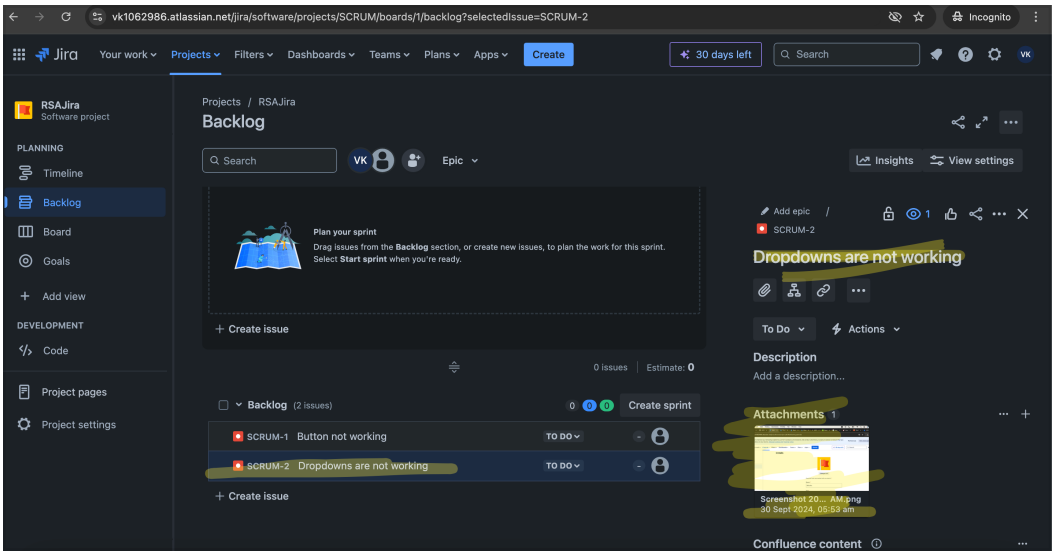
🟡 In postman, Add Screenshot : POST

```
curl --request POST \
  --url 'https://your-domain.atlassian.net/rest/api/3/issue/{issueIdOrKey}/attachments' \
  --user 'email@example.com:<api_token>' \
  --header 'Accept: application/json'
```

```
curl --location --request POST 'https://your-domain.atlassian.net/rest/api/3/issue/TEST-123/attachments'
 -u 'email@example.com:<api_token>'
 -H 'X-Atlassian-Token: no-check'
 --form 'file=@"myfile.txt"'
```







Below is the cURL command to Add screenshot in Jira:

```
curl --location 'https://vk1062986.atlassian.net/rest/api/3/issue/10001/attachments' \
--header 'Authorization: Basic dmsxMDYyOTg2QGdtYWlsLmNvbTpBVEFUVDN4RmZHRjBydllUZFlIeXVId2l6QTh3V3VISFVaTHFvWnEtSWhkWjM5WHc2Z1VGUVBuTTZTUjRYbnRVOEtMTHVBbHZRLUY5bnNFblpJNFlaVDJwWUNTOTJHYTBkU2YxRnlVNUV1RmhQdllDaEZwM1FJRk14REJ5S1ZIeFFvUXh5aUZ4WElsQ2twRGRoa0dxSFVtWnZTMWtFT2d6dlNqZklIMHNTakFlZmd3NWYzWWhjYm89MzYwMjcxNEU=' \
--header 'X-Atlassian-Token: no-check' \
--header 'Cookie: atlassian.xsrf.token=6ba5b563952d94a8d5d020c76163a736df4f25a3_lin' \
--form 'file=@"/Users/krishnabros/Desktop/Screenshot 2024-09-30 at 4.57.09 AM.png"'
```

Interview Question:

1.When we want to send any file or any attachments through REST API, we can do it through form parameters
i.e use ‘form-data’ under request body section of Postman
2.Sending parameters using form parameters is always secure as this will not concatenate parameters in the request URL
3.Sending parameters using query parameters is not secure as this will concatenate parameters in the request URL

Note:
Choose the data type you need for your request body: form data, URL-encoded, raw, binary, or GraphQL.
https://learning.postman.com/docs/sending-requests/create-requests/parameters/
Form-data or multipart