



ServicePRO JIRA

*ServicePRO –JIRA Integration
User Reference guide*

Contents

1. Overview	1
1.1 ServicePRO to JIRA	2
1.2 JIRA to ServicePRO	2
2. Pre-Requisites:	3
2.1. Creation of Custom fields in JIRA: -	3
2.1.1. JIRA Custom Field - RequestID	3
2.1.2. JIRA Custom Fields - Category, Department Name and Reporter Name.....	4
2.1.3. Grant Modify Reporter Access.....	4
2.1.4. Associate EPIC Name custom field to Projects	5
3. Implementation	6
3.1 ServicePRO to JIRA	6
3.1.1 Required Permissions in JIRA	6
3.1.2 Generate API Token	7
3.1.3 Create Custom fields in ServicePRO.....	7
3.1.4 ServicePRO API Configuration.....	7
3.2 JIRA to ServicePRO	12
3.2.1 Creating 'Update Service request Memo' Automation rule in JIRA:	12
3.2.2 Create a Zapier account	15
3.2.3 Create a Zap to update status in ServicePRO.....	15
4. Notes & Limitations	16
a. Limitations.....	16
b. Recommendations: -	16
c. Points to Remember:	16
d. Out of Scope:.....	16

1. Overview

This document outlines the pre-requisites and steps that are required to configure the ServicePRO-JIRA Integration using the Zapier platform and ServicePRO JIRA Plug-in.

The following workflows can be configured through this integration:

- Creation of a new JIRA issue when a ServicePRO request is placed in a specific queue
- Two-way memo updates syncing between ServicePRO request and JIRA Issue
- Closing of the corresponding ServicePRO request when a JIRA issue is closed

ServicePRO to JIRA:

- Creation of a new JIRA issue with the comments and attachments, when a ServicePRO request (with memos and/or attachments) is placed in a specific queue.
- Memo update (along with embedded images) in a ServicePRO service request syncing to the corresponding JIRA Issue [Last Memo and / or First Memo based on the setting in the JSON mappings file].
- Attachment added in a ServicePRO Service Request syncing to the corresponding JIRA issue
- Custom fields in the ServicePRO Service Request syncing as a comment in JIRA when enabled in JSON mappings file.

JIRA to ServicePRO:

- Comments added in JIRA issue (without embedded images and attachments) syncing to the corresponding ServicePRO Service request. If the automatic email notification is enabled on the ServicePRO Service request for requester and cc users, then the automatic email update will be triggered after the memo gets added to the request.
- Closing of the corresponding ServicePRO request when a JIRA issue is closed
- Moving the issue from one project to another project in JIRA and commenting, updates the corresponding ServicePRO Service Request with the updated JIRA Issue ID and the memo.

The integration has been implemented by using the following components:

1. ServicePRO on Zapier
2. ServicePRO Custom Fields
3. JIRA Custom Fields
4. ServicePRO's JIRA Plug-in
5. ServicePRO Integration Service (Windows Service)
6. A JSON mapping file to specify the mappings between ServicePRO and JIRA fields, and to specify the filter criteria for queue folders to consider for initiating the sync operation, type of memos/traces to consider for initiating the sync operation.

This integration feature helps the user to perform the action in one application, which automatically syncs with the other application and updates the necessary field. In order to enjoy this feature, the user must setup the Zapier account and create zaps for each project that require the integration with

ServicePRO. There should be 2 zaps setup separately, one for updating memo in ServicePRO and the other for updating status in ServicePRO.

ServicePRO Administrator needs to create a custom form type and give end user access for that form in ServicePRO and match that field to JIRA using the ServicePROJIRA.json file. This will help the ServicePRO tickets create and sync the custom fields in JIRA.

ServicePRO- Search User and Search request feature is also facilitated in Zapier to check the availability of JIRA reporters in ServicePRO.

1.1 ServicePRO to JIRA

ServicePRO to JIRA integration has been configured using ServicePRO's JIRA Plug-in, ServicePRO Integration Service (Windows Service) and a JSON mapping file.

In ServicePRO, when new service requests are created and moved to a specific designated queue folder, then the ServicePRO JIRA plug-in, in combination with the ServicePRO Integration Service invokes the JIRA API to create the same Issue in JIRA.

Likewise, when memo updates happen in service requests that have a corresponding JIRA Issue, then the ServicePRO JIRA plug-in, in combination with the ServicePRO Integration Service invokes the JIRA API to create a comment in the respective JIRA Issue.

1.2 JIRA to ServicePRO

ServicePRO to JIRA integration has been configured using Zaps on Zapier to invoke specific actions in ServicePRO based on specific triggers in JIRA.

ServicePRO application on Zapier platform facilitates the following two actions:

1. Add Memo in ServicePRO service request
2. Update Status in ServicePRO Service request

Zaps should be created in Zapier for the following scenarios:

- To track the trigger in JIRA Zapier app for addition of new comments in JIRA Issue and to invoke the 'Add Memo' action in ServicePRO Zapier app to add the memo into the respective ServicePRO service request.
- To track the trigger in JIRA Zapier app for closing of JIRA Issue and to invoke the 'Update Service Request' action in ServicePRO Zapier app to update the status to closed in the respective ServicePRO service request.

2. Pre-Requisites:

- ServicePRO's Web API is required to facilitate the integration with JIRA.
- 1 named ServicePRO license is required to access the ServicePRO Web API from Zapier.
- ServicePRO Web API URL should be public facing. It can either be on the premise or on the cloud.
- Minimum required Zapier plan for this integration is Professional
- The following custom fields, "**RequestID**", "**Category**", "**Department Name**", "**Reporter Name**" must be created in JIRA
- The JIRA URL needs to be provided in order to get the custom field key in JIRA.
- User type 'JIRA' must be created in ServicePRO with the following custom text fields:
 - **Reporter Name**
 - **Assignee Name**
- Service Request Form must be created in ServicePRO with the following custom fields:

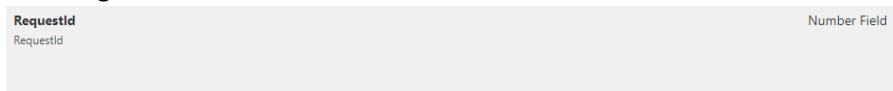
Label	Field Type	Values	Permissions
Projects	Drop down (Trace Enabled)	Enter the JIRA project key that needs to be integrated	View and Update
Issue Type	Drop down (Trace Enabled)	Enter the values that match the JIRA Issue Type (see Limitations Section)	View and Update
Issue Number	Text (Trace Enabled)		View

2.1. Creation of Custom fields in JIRA: -

The following custom fields, grant/modify report access and epic name custom field association to projects must be configured in JIRA.

2.1.1. JIRA Custom Field - RequestID

Login to the JIRA account as an administrator and create the custom field "RequestID" by selecting the number field.



To create the custom fields, Navigate to JIRA settings-> Issues-> Custom fields

Click on Create custom field button on top right

Once you created the custom field please associate the field to the respective Project screens.

The following image shows

Custom Field information for Category

Details **Screens** Contexts

This custom field is currently included in the following **17** screens.

[Add or remove associated screens](#)

Screen Name	Contexts
Default Screen	
Workflow Screen	
Resolve Issue Screen	
SP: Scrum Default Issue Screen	
SP: Scrum Bug Screen	
IT: Kanban Default Issue Screen	
IT: Kanban Bug Screen	
TJI: Scrum Default Issue Screen	
TJI: Scrum Bug Screen	
TJI: Kanban Default Issue Screen	
TJI: Kanban Bug Screen	

2.1.2. JIRA Custom Fields - Category, Department Name and Reporter Name

Login to the JIRA account as an administrator and create the custom fields “Category”, “Department Name” and “Reporter Name” by selecting the Text.

Issues

Custom fields

Name	Type
Approvers Contains users needed for approval. This custom field was created by Jira Service Desk.	User Picker (multiple users)
Category Category	Text Field (single line)

Reporter name
Reporter name

Aa Text Field (single line)

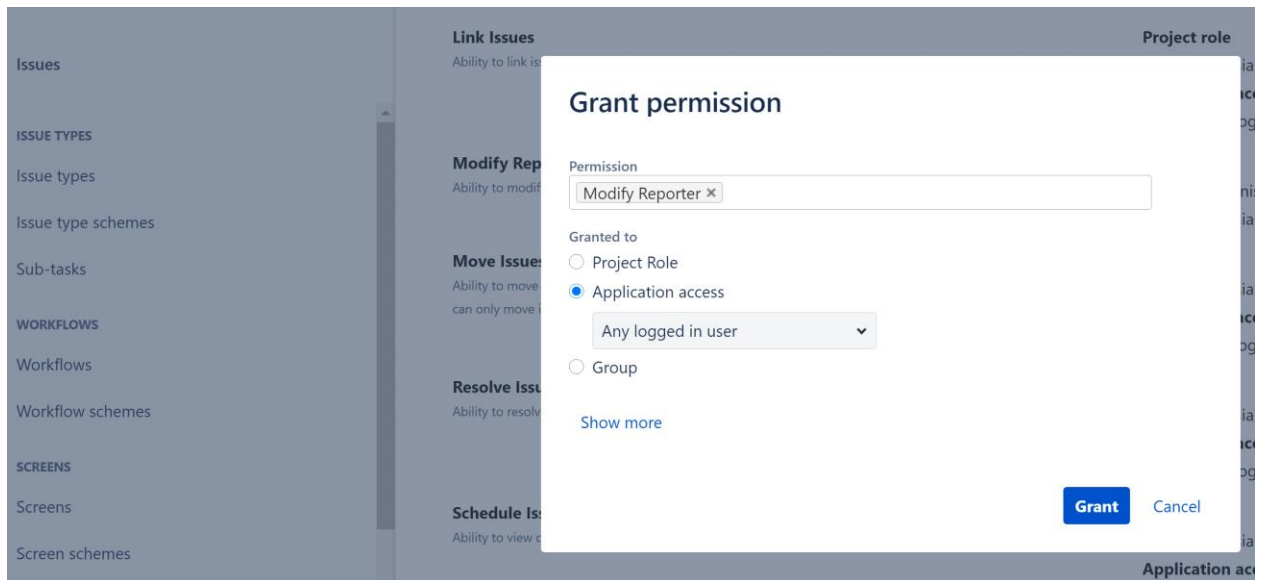
Department name
Department name

Aa Text Field (single line)

2.1.3. Grant Modify Reporter Access

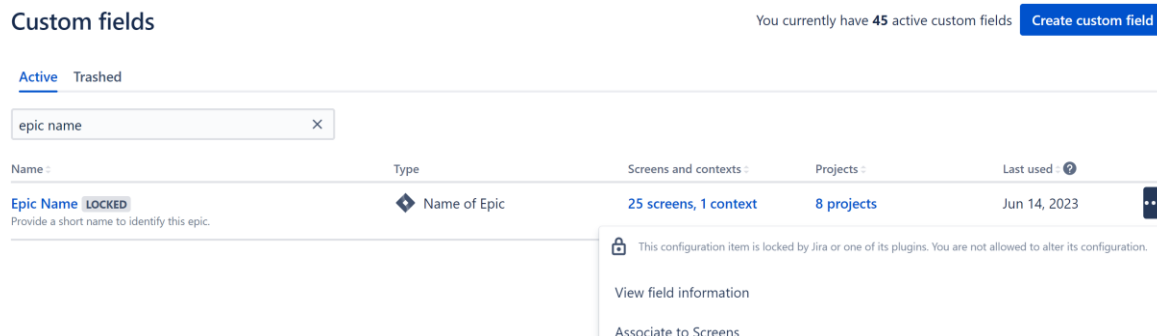
Grant Modify Reporter permission under JIRA Settings and then grant Application Access to “Any Logged in user”.

JIRA Settings-> Issues->Permission Schemes->Default Permission Scheme



2.1.4. Associate EPIC Name custom field to Projects

In the custom field UI in JIRA, search for the existing custom field “Epic Name” and Associate the same to the Projects required. This will show the added Epic Name field in the UI.



Click on Associate to Screens and check the required screens for which the ‘Epic Name’ should be displayed.

Issues

Associate field Epic Name to screens

Associate the field Epic Name to the appropriate screens. You must associate a field to a screen before it will be displayed. New fields will be added to the end of a tab.

Filter by name 	
Screen	Tab ☒
DP: Kanban Bug Screen	Field Tab ☒
DP: Kanban Default Issue Screen	Field Tab ☒
Default Screen	Field Tab ☒
IT: Kanban Bug Screen	Field Tab ☒
IT: Kanban Default Issue Screen	Field Tab ☒

[Return to search](#) 1 of 52

 2
 


Details

Reporter	 Fresh Install
Labels	None
Category	Custom Category Test
Department name	QA
Reporter name	sakthi
Requestid	430
Priority	 Medium
Epic Name	Demo for FR- JIRA Request Verification

3. Implementation

3.1 ServicePRO to JIRA

3.1.1 Required Permissions in JIRA

Modify reporter permission in JIRA for application access is required in order to map the ServicePRO reporter to JIRA reporter. If this permission is not given, then all the service requests created in ServicePRO will be mapped in JIRA to a default user specified in ServicePRO configuration (i.e. to the username specified in the JSON Mapping file).

Note: The username of the reporters and assignee custom fields in ServicePRO and the full name of the user in JIRA must match in order to successfully sync, otherwise the request in JIRA would be created under the account specified in ServicePRO.JSON mapping file.

3.1.2 Generate API Token

API token is required for ServicePRO to connect to the JIRA Application. In order to get the API token navigate to <https://id.atlassian.com/manage/api-tokens> and create an API token.

3.1.3 Create Custom fields in ServicePRO

Create a service request custom form that contains the following fields to match in JIRA.

Label	Field Type	Values	Permissions	Matching JIRA Field Name
Projects	Drop down	Enter the projects that need to be integrated	View and Update	JIRAProjectKey
Issue Type	Drop down	Enter the values that match the JIRA	View and Update	JIRAIssueType
Issue Number	Text		View	JIRAIssueNumber

Create a user custom form type “JIRAUser” that contains the following fields to match in JIRA.

Label	Field Type	Values	Permissions	Matching JIRA Field Name
Assignee	Text		View and Update	JIRAReporterName
Reporter	Text		View and Update	JIRAAssigneeName

For ServicePRO users to be integrated with JIRA, the “Assignee” and “Reporter” fields must be pre-populated with ServicePRO **Username (or Name) value**.

3.1.4 ServicePRO API Configuration

The following information is needed from the JIRA account for ServicePRO Configuration:

1. Jira account user's Email Address
2. Jira API token reference
3. default Jira issue type
4. default Jira project key where the JIRA ticket will get created

Json file is needed to connect the ServicePRO application with the JIRA application and map the fields between ServicePRO and JIRA.

Map the ServicePROJIRA.json file by using the client Jira account settings, based on the guidance given in the below table.

Fields	Sample Data	Interpretation	Comments
"Username"	ClientUserID@xxxx.com	Jira account user's Email Address	The user who has access to Modify reporters & create project roles
"API Token"	"xxxxxxxxxxxxxxxxxxxx"	Jira API token reference	
"QueueId"	[1,5,9]	ServicePRO folder location id, check the table 'tblqueue' in database	Creates a ticket in JIRA only for the service request created in specific folder <queue id>. When queue id is blank all the service request created is integrated to Jira.
"DefaultJIRAIssueType"	"Story"	default Jira issue type	This is used when the Issue type is not specified by user while creating the Service Request.
"DefaultJIRAProjectKey"	"NTP"	default Jira project key where the JIRA ticket will get created	This is used when the Project is not specified by user while creating the Service Request.
"ServicePROJIRAProject Key"	"JIRAProjectKey"	Projects drop down- Custom form field name	In the application Navigate to ServicePRO-> Design-> Service Catalog (Custom forms)
"ServicePROJIRAIssueType"	"JIRAIssueType"	Issue Type drop down- Custom form field name	In the application Navigate to ServicePRO-> Design-> Service Catalog (Custom forms)
"ServicePROJIRAIssueNumber"	"JIRAIssueNumber"	Issue Number drop down- Custom form field name	In the application Navigate to ServicePRO-> Design-> Service Catalog (Custom forms)

ServicePROUserIdentity Flag	1	Identifies user with custom key fields if not found then ServicePRO email address OR name.	0 or 1 can be entered. If 0 then the primary identification will be by email address or name
"UseFirstMemoAsJIRADescription"	true	First memo created in Service request is used as Jira description	true or false can be entered, but the value is case sensitive. When true the first memo in ServicePRO will be displayed in description section of Jira ticket
"UsePrivateMemoForVisibility"	true		true or false can be entered, but the value is case sensitive. When true the private memo is enabled and will be visible only for particular Jira user role.
"JIRACommentVisibilityForPrivateMemo"	"Administrators"	The user who is given authority to view private memo	Required when "UsePrivateMemoForVisibility" is true.
"Enable"	true	Integration feature of ServicePRO and JIRA	true for enable and false to disable. The values entered is case sensitive.
"ScheduleJIRASyncByMinutes"	1	The duration required for the service to run	Maximum minutes that can be specified is 5
"SyncFirstMemo"	false	Syncing first memo with JIRA initially during the integration	By default, the value is set as false. When it's set to true, the first memo in ServicePRO service request will be synced to JIRA ticket. When it's false, the last memo in ServicePRO service request will be synced to JIRA ticket.
"SyncLastMemo"	false	Syncing last memo with JIRA initially during the integration	By default, the value is set as false. When it's set to true, the last memo in ServicePRO service request will be

			synced to JIRA ticket. When it's false, the first memo in ServicePRO service request will be synced to JIRA ticket.
When both SyncFirstMemo and SyncLastMemo are set to false, only the last memo will be synced from ServicePRO to JIRA			
SyncCustomField	false	Syncing the ServicePRO custom fields in JIRA as part of the comment when the first time the request is synced to JIRA.	By default, the value is set as false. If set to false, custom field values in the service request will not sync with JIRA. If set to true, the custom field values in ServicePRO service request will sync with JIRA ticket as part of the comment.
SendAutoEmailNotification	false	Syncing the ServicePRO auto email notification trace to JIRA ticket.	By default, the value is false. If set to true, the trace that is added in ServicePRO service request for auto email notification will be synced to JIRA ticket as a comment.
"mappings"	<pre>{ "ServicePROIdentifier": "memo", "JIRAIdentifier": "body", "Conditions": [{ "FieldName": "memo", "Operator": "not empty" }]}, { "ServicePROIdentifier": "trace", "JIRAIdentifier": "body",</pre>	The conditions work like an "OR" operator.	At least one condition is required. But a maximum of six conditions present in json file are supported. The action is performed when the given condition is satisfied.

	<pre> "Conditions": [{ "FieldName": "trace", "Operator": "contains", "Value": "queue" }] </pre>		
All the ServicePRO Identifier fields and a few Jira Identifiers like title, JiraIssueType and JiraProject, RequestId are mandatory non-customizable.			
"JIRAIdentifier"	"reporter"	Maps ServicePRO reporter to JIRA if present in both the applications	Optional. If the user wanted to have the default reporter (the username present in json file), the reporter can be blank just with ""
	"assignee"	Maps ServicePRO assignee to JIRA if present in both the applications	Optional. If assignee is not required, the assignee can be blank just with ""
	"priority"		Optional. If priority is not needed remove priority and let it display with ""
	"customfield_xxxx"	Custom field Id for Category	Optional. If Category is not needed remove category and let it display with ""
	"customfield_xxxx"	Custom field Id for epicname	Required if Epic needed to be created in JIRA.
	"customfield_xxxx"	Custom Field Name for ServicePRO RequestId in JIRA	Required
	"customfield_xxxx"	Custom field name for Reporter Name in JIRA	Required
	"customfield_xxxx"	Custom field name for Department	Required
For "ServicePROIdentifier" - "RequestId", "Reporter name", "Department name", "category", and epicname the appropriate "JIRAIdentifier" custom field needed to be given. In order to get the custom field key in JIRA, go to the link (<a href="http://<Client Jira URL >/rest/api/2/field">Client Jira URL > /rest/api/2/field) and search for the custom field name <example: RequestId>.			

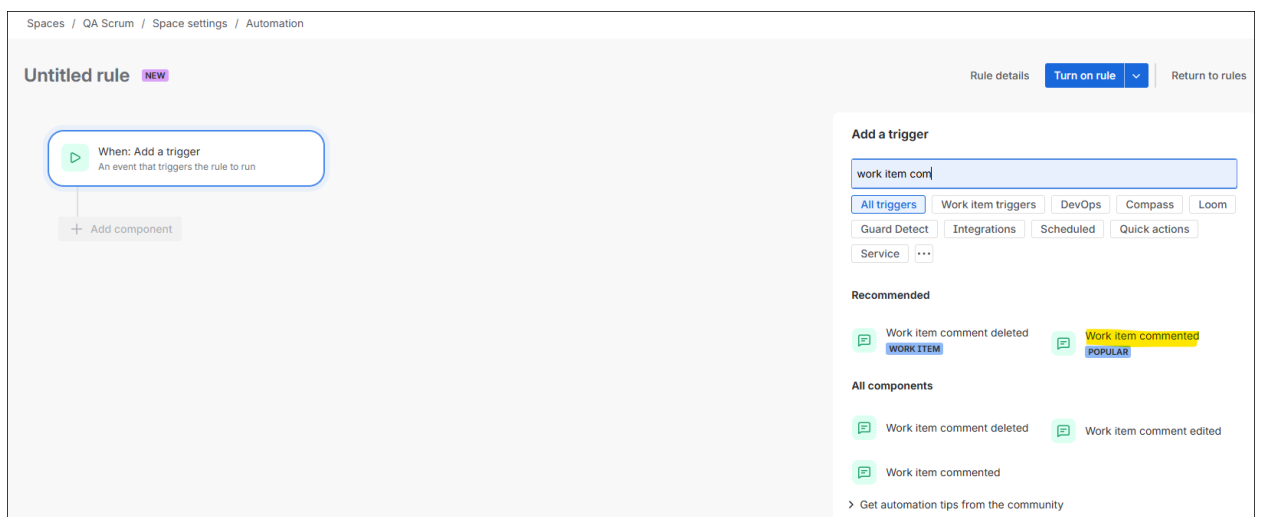
3.2 JIRA to ServicePRO

3.2.1 Creating 'Update Service request Memo' Automation rule in JIRA:

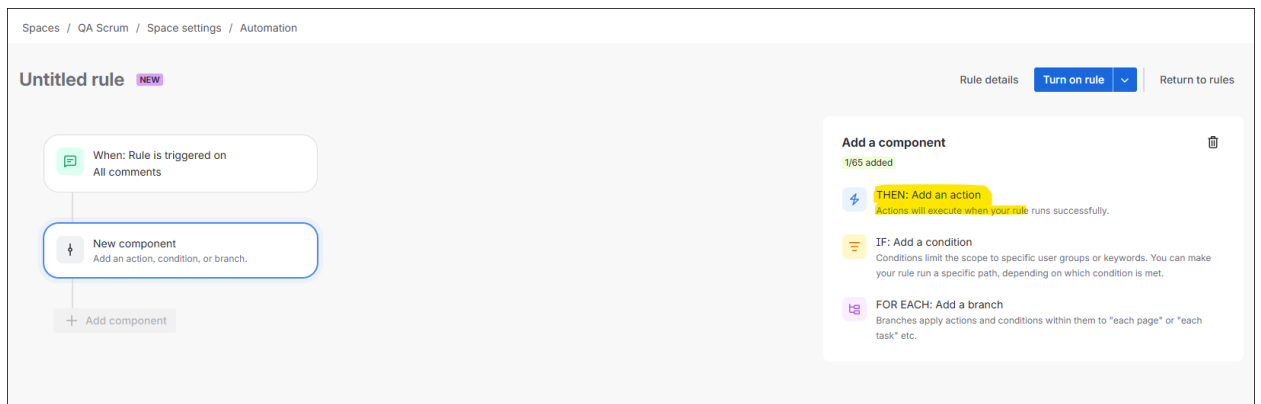
- Navigate to JIRA as an administrator and Go to Automation under Space Settings
- Click on Create Rule and select Create from scratch



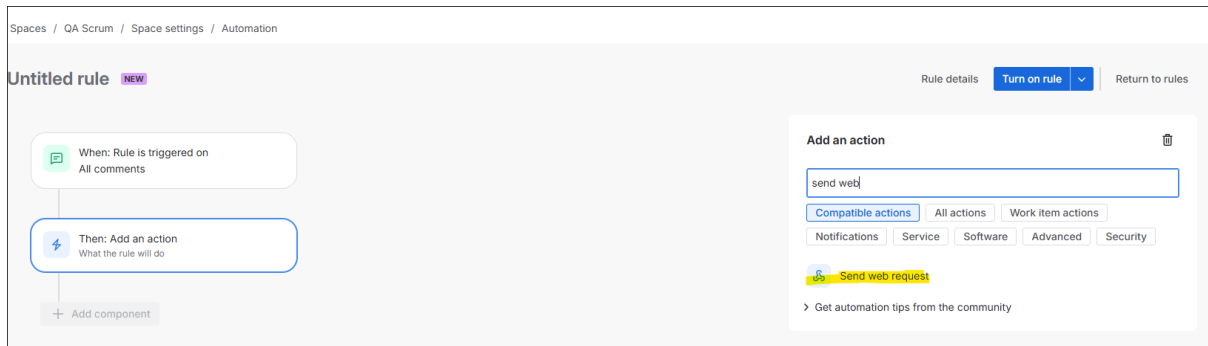
- Select Work Item commented option for trigger
- Click Add Component



- Select "THEN: Add an action"



- Search for Send Web request and select it



- Setup a Web request as follows:

1. Web request URL : `https://<URL WHERE SP API IS RUNNING>/api/requests/jiramemo/zapier/<CUSTOM FIELD IN JIRA for ServicePRO RequestId>`

(This custom field that was created by following the instructions given in Section 2.1.1 and also present in the ServicePROJIRA.json file.)

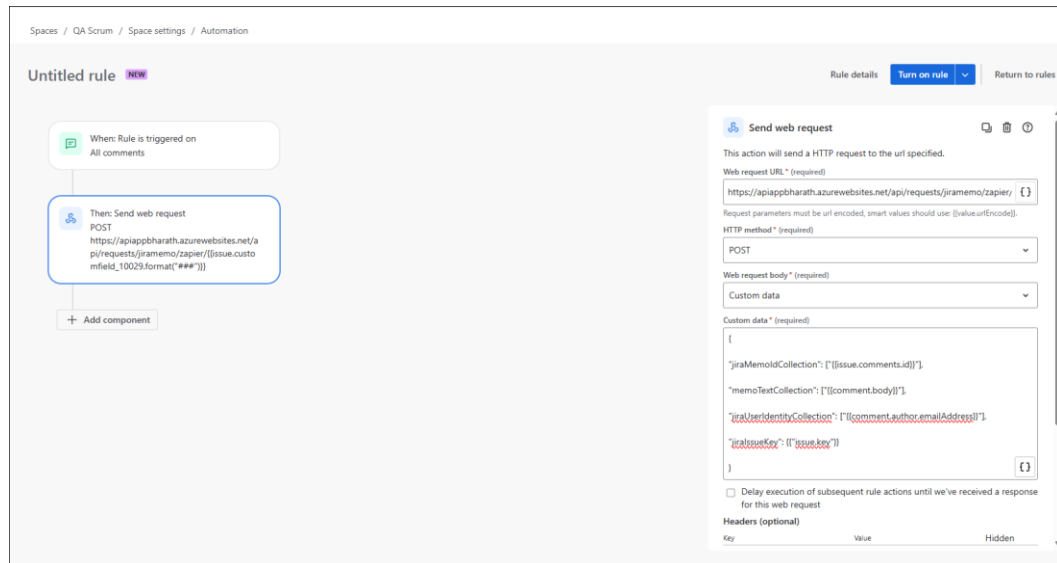
Example:

[https://apiappbharath.azurewebsites.net/api/requests/jiramemo/zapier/{{issue.customfield_10029.format\('###'\)}}](https://apiappbharath.azurewebsites.net/api/requests/jiramemo/zapier/{{issue.customfield_10029.format('###')}})

```
{
  "ServicePROIdentifier": "epicname",
  "JIRAIdentifier": "customfield_10011"
},
{
  "ServicePROIdentifier": "CF_RequestId",
  "JIRAIdentifier": "customfield_10029"
},
{
  "ServicePROIdentifier": "CF_Reporter_Name",
  "JIRAIdentifier": "customfield_10031"
},
{
  "ServicePROIdentifier": "CF_Department",
  "JIRAIdentifier": "customfield_10032"
}
```

2. HTTP Method: POST
3. Web request body: Custom data
4. Custom data:

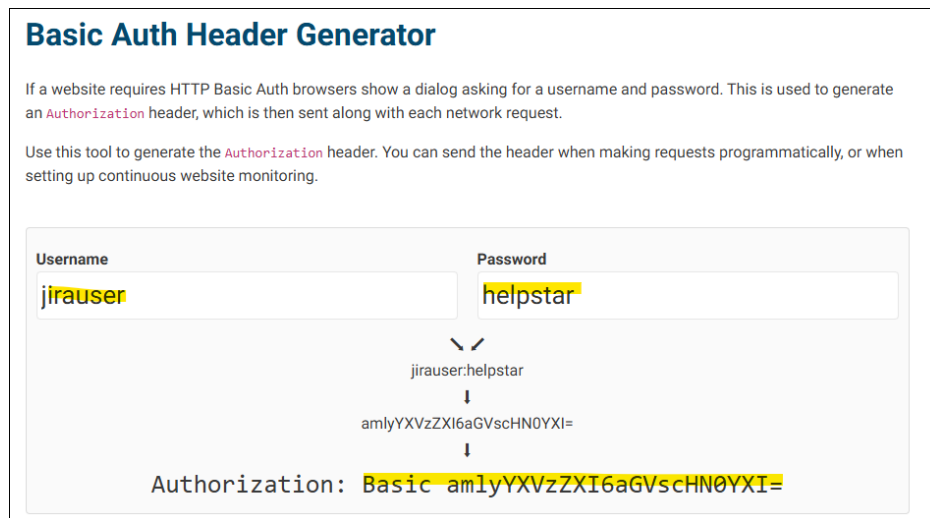
```
{
  "jiraMemoldCollection": ["{{issue.comments.id}}"],
  "memoTextCollection": ["{{comment.body}}"],
  "jiraUserIdentityCollection": ["{{comment.author.emailAddress}}"],
  "jiraIssueKey": "{{issue.key}}"
}
```



- For setting the Authorization Key in the Header for this web request, Generate HTTP Basic Authentication Header using the ServicePRO user credentials (Named License user that is reserved for this integration as stated in the pre-requisites section – same user account that will be used in Zapier connection as well) -

Launch the URL in browser <https://www.debugbear.com/basic-auth-header-generator>

Enter the ServicePRO user credentials mentioned above and generate the authorization key value.



- Set following Headers in Headers section:
 Key: Content-Type, Value: application/json
 Key: Authorization, Value: **Basic + Authorization Key VALUE generated in previous step FROM <https://www.debugbear.com/basic-auth-header-generator>**
- Expand Validate your web request configuration section

Enter any Jira Issue number - the JIRA Issue that is created using customer's project (for example, QS-123) and click on Validate button. You should receive response 200 OK.

Now click on Turn on Rule button (This will save the rule and enable it)

3.2.2 Create a Zapier account

- Navigate to <https://zapier.com> and create an account. Click on Make a Zap! button.

3.2.3 Create a Zap to update status in ServicePRO

Trigger:

- Select the Trigger App <JIRA Software Cloud> and Software Triggers <Updated Issue> and continue to the next step.

Action:

- Create a filter to continue if the field status name is “Done” or “Closed”.
 - Tip: Navigate to JIRA’s board setting to verify the status.
- Setup formatter by Zapier Utilities.

- *Lookup Table Value for ServicePRO's request status of "Closed" is "6".*
- Select <ServicePRO> as the Action app and <Update Request Status> as the action and click 'Save + Continue'.

4. Notes & Limitations

a. Limitations

- When the comment format in JIRA is given as 'Textcode' with space then the user may not receive further comments in ServicePRO.
- When the project is moved from one to another without any comments in the JIRA ticket, the trace on moving the project will not be reflected in ServicePRO
- Encrypted attachments in the ServicePRO are exported to JIRA but the attachments cannot be opened because of the security purposes.
- When a user closes the ticket in JIRA, the appropriate service request in ServicePRO gets closed. The memo author in ServicePRO displays as the default user specified in the Zapier integration.
- When a user in ServicePRO creates a service request, appropriate ticket is created in Jira. The comment author in JIRA will be set with the memo author from ServicePRO if the user is present in JIRA - If the user is not present in JIRA, the default user specified in the JSON Mapping file will be set as the comment author.

b. Recommendations: -

- Kindly refrain commenting the Jira using the 'Textcode' format
- Search User and Search Request of ServicePRO in Zapier is recommended to be used with Email address and Search Request Id respectively.
- While moving the project from one to another, comment on JIRA issues soon after moving.

c. Points to Remember:

- For any changes except the queue id and Integration Enable, work on the json file that is present in ServicePRO.integration folder. If the change is on the queue id then work on the json file present in the appropriate Web, Service or StarWatch folder.
- Every time when the integration "Enable" is changed from true to false or vice-versa, Recycle the application pool.
- If Private memo visibility is set to true – make sure People tab in the Jira project settings has people with the appropriate role set for the project

d. Out of Scope:

- Creating the Issue Type "Sub-task" is out of scope for this release.