

# Conforms Makro für Jira Epic und Issues Erstellung

## Übersicht der Lösung

Diese Lösung erstellt ein Conforms-Formular in Confluence, das:

1. Ein Epic in Jira mit den angegebenen Informationen erstellt
2. Eine vorhandene Checkliste in das Epic einfügt
3. Optional bis zu vier Issues erstellt, die dem Epic zugeordnet sind
4. In jedes Issue eine unterschiedliche vorhandene Checkliste einfügt
5. Eine Bestätigungstabelle mit den erstellten Tickets anzeigt

## Formularstruktur

### 1. Conforms Form Definition

```
## Epic-Informationen
```

### 2. Conforms Field Definition für Epic-Informationen

Folgende Felder werden benötigt:

1. **epicName** (Text) - Name des Epics
2. **epicDescription** (Textarea) - Beschreibung des Epics
3. **priority** (Select) - Priorität
4. **assignee** (User Picker) - Zugewiesen an
5. **dueDate** (Date) - Fällig am
6. **issueType** (Select) - Issue-Typ
7. **jiraProject** (Select) - Zugehöriges Jira-Projekt
8. **jiraKey** (Text, optional) - Jira-Key, falls bereits vergeben
9. **createIssue1** (Checkbox) - Issue 1 erstellen
10. **createIssue2** (Checkbox) - Issue 2 erstellen
11. **createIssue3** (Checkbox) - Issue 3 erstellen
12. **createIssue4** (Checkbox) - Issue 4 erstellen

### **3. Conforms Registration Control**

Zur Anzeige des Formulars auf der Seite.

### **4. IFTTT Integration Rules**

Mehrere IFTTT-Regeln werden benötigt:

1. Epic erstellen
2. Issue 1 erstellen (wenn Checkbox aktiviert)
3. Issue 2 erstellen (wenn Checkbox aktiviert)
4. Issue 3 erstellen (wenn Checkbox aktiviert)
5. Issue 4 erstellen (wenn Checkbox aktiviert)
6. Bestätigungstabelle erstellen

## **Implementierungsdetails**

### **Schritt 1: Formular definieren**

Fügen Sie das "Conforms Form Definition" Makro ein und geben Sie einen Namen für das Formular an, z.B. "EpicCreationForm".

### **Schritt 2: Felder definieren**

Fügen Sie für jedes Feld ein "Conforms Field Definition" Makro ein:

#### **Epic Name**

Feld-ID: epicName  
Feldtyp: Text  
Label: Epic Name  
Erforderlich: Ja

#### **Epic Beschreibung**

Feld-ID: epicDescription  
Feldtyp: Textarea  
Label: Epic Beschreibung  
Erforderlich: Ja

## Priorität

Feld-ID: priority  
Feldtyp: Select  
Label: Priorität  
Optionen: Highest | High | Medium | Low | Lowest  
Erforderlich: Ja

## Zugewiesen an

Feld-ID: assignee  
Feldtyp: User Picker  
Label: Zugewiesen an  
Erforderlich: Ja

## Fällig am

Feld-ID: dueDate  
Feldtyp: Date  
Label: Fällig am  
Erforderlich: Ja

## Issue-Typ

Feld-ID: issueType  
Feldtyp: Select  
Label: Issue-Typ  
Optionen: Epic  
Default: Epic  
Erforderlich: Ja

## Jira-Projekt

Feld-ID: jiraProject  
Feldtyp: Select  
Label: Jira-Projekt  
Optionen: [Liste der verfügbaren Projekte]  
Erforderlich: Ja

## Jira-Key

Feld-ID: jiraKey  
Feldtyp: Text

Label: Jira-Key (falls bereits vergeben)  
Erforderlich: Nein

## Issue 1 erstellen

Feld-ID: createIssue1  
Feldtyp: Checkbox  
Label: Issue 1 erstellen

## Issue 2 erstellen

Feld-ID: createIssue2  
Feldtyp: Checkbox  
Label: Issue 2 erstellen

## Issue 3 erstellen

Feld-ID: createIssue3  
Feldtyp: Checkbox  
Label: Issue 3 erstellen

## Issue 4 erstellen

Feld-ID: createIssue4  
Feldtyp: Checkbox  
Label: Issue 4 erstellen

## Schritt 3: Formular anzeigen

Fügen Sie das "Conforms Registration Control" Makro ein, um das Formular anzuzeigen.

## Schritt 4: IFTTT-Regeln für Epic-Erstellung

Fügen Sie das "Conforms IFTTT Integration Rules" Makro ein:

Event: onCreated  
Action: Create JIRA Issue

Im Makro-Body (mit No Format Makro):

```
{
  "fields": {
    "project": {
      "key": "[entry.jiraProject]"
    },
    "summary": "[entry.epicName]",
    "description": "[entry.epicDescription]",
    "issuetype": {
      "name": "Epic"
    },
    "priority": {
      "name": "[entry.priority]"
    },
    "assignee": {
      "name": "[entry.assignee]"
    },
    "duedate": "[entry.dueDate]",
    "customfield_XXXXX": "[Template für Epic-Checkliste]"
  }
}
```

Hinweis: `customfield_XXXXX` muss durch das tatsächliche Feld für die Checkliste in Ihrem Jira-System ersetzt werden.

## Schritt 5: IFTTT-Regel zum Speichern des Epic-Keys

Fügen Sie ein weiteres "Confirms IFTTT Integration Rules" Makro ein:

```
Event: onCreated
Action: Update Entry
```

Im Makro-Body:

```
epicJiraKey=[iftttResult_0]
```

## Schritt 6: IFTTT-Regeln für Issue 1

Fügen Sie ein "Confirms IFTTT Integration Rules" Makro mit Bedingung ein:

```
Event: onCreated
Action: Create JIRA Issue
Condition: [entry.createIssue1] == "true"
```

Im Makro-Body (mit No Format Makro):

```
{
  "fields": {
    "project": {
      "key": "[entry.jiraProject]"
    },
    "summary": "Issue 1 für [entry.epicName]",
    "description": "Unteraufgabe 1 für Epic [entry.epicJiraKey]",
    "issuetype": {
      "name": "Issue"
    },
    "priority": {
      "name": "[entry.priority]"
    },
    "assignee": {
      "name": "[entry.assignee]"
    },
    "customfield_XXXXX": "[entry.epicJiraKey]",
    "customfield_YYYYY": "[Template für Issue 1 Checkliste]"
  }
}
```

Hinweise: - `customfield_XXXXX` muss durch das tatsächliche Epic-Link-Feld in Ihrem Jira-System ersetzt werden - `customfield_YYYYY` muss durch das tatsächliche Feld für die Checkliste ersetzt werden

## Schritt 7: IFTTT-Regel zum Speichern des Issue 1 Keys

Event: `onCreated`  
 Action: `Update Entry`  
 Condition: `[entry.createIssue1] == "true"`

Im Makro-Body:

```
issue1JiraKey=[iftttResult_2]
```

## Schritt 8-13: Wiederholen für Issues 2-4

Wiederholen Sie die Schritte 6-7 für die Issues 2, 3 und 4, mit entsprechend angepassten Bedingungen und Feldwerten.

## Schritt 14: IFTTT-Regel für Bestätigungstabelle

Fügen Sie ein letztes "Conforms IFTTT Integration Rules" Makro ein:

Event: onCreated  
Action: Show Message

Im Makro-Body:

```
<h2>Erstellte Jira-Tickets</h2>
<table class="confluenceTable">
<tr>
<th>Typ</th>
<th>Jira-Key</th>
<th>Name</th>
<th>Ersteller</th>
</tr>
<tr>
<td>Epic</td>
<td><a href="https://[jira-url]/browse/[entry.epicJiraKey]">[entry.epicJiraKey]</a></td>
<td>[entry.epicName]</td>
<td>[entry.assignee]</td>
</tr>
#if([entry.createIssue1] == "true")
<tr>
<td>Issue</td>
<td><a href="https://[jira-url]/browse/[entry.issue1JiraKey]">[entry.issue1JiraKey]</a></td>
<td>Issue 1 für [entry.epicName]</td>
<td>[entry.assignee]</td>
</tr>
#end
#if([entry.createIssue2] == "true")
<tr>
<td>Issue</td>
<td><a href="https://[jira-url]/browse/[entry.issue2JiraKey]">[entry.issue2JiraKey]</a></td>
<td>Issue 2 für [entry.epicName]</td>
<td>[entry.assignee]</td>
</tr>
#end
#if([entry.createIssue3] == "true")
<tr>
<td>Issue</td>
<td><a href="https://[jira-url]/browse/[entry.issue3JiraKey]">[entry.issue3JiraKey]</a></td>
<td>Issue 3 für [entry.epicName]</td>
<td>[entry.assignee]</td>
</tr>
#end
#if([entry.createIssue4] == "true")
<tr>
<td>Issue</td>
```

```
<td><a href="https://[jira-url]/browse/[entry.issue4JiraKey]">[entry.issue4JiraKey]</a></td>
<td>Issue 4 für [entry.epicName]</td>
<td>[entry.assignee]</td>
</tr>
#end
</table>
```

## Anpassungen für Ihre Umgebung

Für die vollständige Implementierung müssen Sie:

1. Die tatsächlichen Jira-Projektschlüssel in den Optionen für das jiraProject-Feld angeben
2. Die tatsächlichen Feld-IDs für Epic-Link und Checklisten in Ihrem Jira-System identifizieren
3. Die Namen der Checklisten-Templates anpassen
4. Die Jira-URL in den Links der Bestätigungstabelle anpassen

## Hinweise zur Implementierung

- Stellen Sie sicher, dass der Application Link zwischen Confluence und Jira korrekt konfiguriert ist
- Testen Sie das Formular zunächst mit einem Test-Projekt in Jira
- Überprüfen Sie die Berechtigungen des Benutzers, der das Formular ausfüllt, um sicherzustellen, dass er Jira-Issues erstellen kann