

Lesson 1 - Creating an FAQ workflow tutorial

Comala Workflows can be used to define an unobtrusive workflow to let users submit FAQs with a simple approval process.

A workflow lets you define a set of states, approvals, rules and actions. The workflow is defined once only and is applied to all the pages on a space or spaces.

Requirements

- Any user can submit an FAQ
- Only the Author and FAQ Administrators can see the submitted approval
- FAQ Administrator determines if the FAQ is appropriate and assigns the appropriate Component lead
- If the FAQ is approved, then the FAQ becomes visible to the Component lead and he/she is notified
- If the FAQ is rejected, then the Author is notified
- If the FAQ is approved by the Component lead then it is re-assigned to the FAQ administrator for publishing
- If the FAQ is published then it becomes visible to everybody
- If the FAQ is rejected in this state, then the Author and Component lead are notified
- If the FAQ is updated after it has been published, then it has to be approved by the Component lead and FAQ administrator again



Using the workflow markup

See [this tutorial](#) to learn how to work with the workflow markup.

The states workflow

Let's define the states and their approvals, using the `{state}` and `{approval}` macros:

```
{workflow}
  {state:Draft|submit=Submitted}
  {state}
  {state:Submitted|approved=Approved|rejected=Rejected}
    {approval:Approve}
  {state}
  {state:Approved|approved=Reviewed|rejected=Closed}
    {approval:Review}
  {state}
  {state:Reviewed|approved=Published|rejected=Closed}
    {approval:Publish}
  {state}
  {state:Published|final=true|hideselection=true|updated=Approved}
  {state}
  {state:Closed}
  {state}
  {state:Rejected}
  {state}
{workflow}
```

Setting approval permissions

We have to update the workflow to define who can give the different approvals.

We are going to need two [workflow parameters](#) *FAQ Administrators* and *Component Leads* to handle the permissions.

For the **Review** approval, we want any of the FAQ administrators to select which of the Component lead is to review the FAQ. For that we use the [approver selection feature](#) which requires to specify the users group from which we want to select:

```

{workflow}
  {workflowparameter:FAQ Administrators|type=group}
    confluence-users
  {workflowparameter}
  {workflowparameter:Component Leads|type=group}
    confluence-users
  {workflowparameter}
  {state:Draft|submit=Submitted}
  {state}
  {state:Submitted|approved=Approved|rejected=Rejected}
    {approval:Approve|group=@FAQ Administrators@}
  {state}
  {state:Approved|approved=Reviewed|rejected=Closed}
    {approval:Review|selectedapprover=@Component Leads@}
  {state}
  {state:Reviewed|approved=Published|rejected=Closed}
    {approval:Publish|group=@FAQ Administrators@}
  {state}
  {state:Published|final=true|updated=Approved|hideselection=true}
  {state}
  {state:Closed}
  {state}
  {state:Rejected}
  {state}
{workflow}

```

You can change the [workflow parameters](#) values to set the appropriate groups:

You can override the default Workflow settings and they will apply to all the workflows in the space

Name	Value	Operations
Default view What version of pages should users view by default	Latest	Save

You can set the values of the Workflows Parameters for this space.

Name	Value	Operations
FAQ Administrators	faq-administrators	Save Delete
Component Leads	componente-leads	Save Delete

Now, the FAQ Administrator can select the component lead when giving the **Approve** approval:

Approve

 State: Submitted | Approvals: Approve (activity stream)

Team Lead:

Looking for a user? Just start typing.

- Erika CEO
- Frank CFO

comment... reject Cancel

Email notifications

We need to update the workflow to add the required triggers for sending the email notifications. We use the `{trigger}` and `{send-email}` for this.

First, let's define the email notification to be sent to the FAQ administrators when the author submits the FAQ:

```
{trigger:statechanged|state=Submitted}
  {set-metadata:FAQ Author}@user@{set-metadata}
  {send-email:user=@FAQ Administrators@|subject=FAQ Submitted}
  The FAQ @page@ has been submitting by [~@user@] and is waiting for your approval.
  {send-email}
{trigger}
```

The [trigger](#) is defined on the `statechanged` event for the `Submitted` state. The email is sent to all FAQ Administrators. The body of the email contains [references](#) to the page and the user giving the submit approval.

We also use the `{set-metadata}` macro to *store* the author into the `FAQ Submitted` parameter. We will use it later to send notifications to the author. The [trigger](#) is defined on the `statechanged` event for the `_Submitted_` state. The email is sent to all FAQ Administrators. The body of the email contains [references](#) to the page and the user giving the submit approval.

If the administrator approves the FAQ, then an email is sent to the Component lead, which is the selected approver for the **Review** approval:

```
{trigger:pageapproved|approval=Approve}
  {send-email:user=@Review>selected@|subject=FAQ Accepted}
  The FAQ @page@ has been approved by [~@user@] and is waiting for your review.
  {send-email}
{trigger}
```

Define the trigger for sending the email to the FAQ author if it is rejected by the administrator:

```
{trigger:pagerejected|approval=Approve}
  {send-email:user=@FAQ Author@|subject=FAQ Rejected}
  Your FAQ @page@ submission has been rejected.
  {send-email}
{trigger}
```

Now define the triggers for sending the emails when the FAQs are approved or rejected by the Component lead:

```
{trigger:pageapproved|approval=Review}
  {send-email:user=@Approve>approver@|subject=FAQ approved}
  The FAQ @page@ has been approved by [~@user@] and is waiting for your approval.
  {send-email}
{trigger}
{trigger:pagerejected|approval=Review}
  {send-email:user=@Approve>approver@, @FAQ Author@|subject=FAQ rejected}
  The FAQ @page@ has been rejected by [~@user@].
  {send-email}
{trigger}
```

And we define the triggers for sending the emails when the FAQ is finally approved by the administrator:

```
{trigger:pageapproved|approval=Review}
  {send-email:user=@Approve>approver@|subject=FAQ approved}
  The FAQ @page@ has been approved by [~@user@] and is waiting for your approval.
  {send-email}
{trigger}

{trigger:pagerejected|approval=Review}
  {send-email:user=@Approve>approver@, @FAQ Author@|subject=FAQ rejected}
  The FAQ @page@ has been rejected by [~@user@].
  {send-email}
{trigger}
```

```

{workflow}
  {workflowparameter:FAQ Administrators|type=group}
  confluence-users
  {workflowparameter}
  {workflowparameter:Component Leads|type=group}
  confluence-users
  {workflowparameter}
  {state:Draft|submit=Submitted}
  {state}
  {state:Submitted|approved=Approved|rejected=Rejected}
  {approval:Approve|group=@FAQ Administrators@}
  {state}
  {state:Approved|approved=Reviewed|rejected=Closed}
  {approval:Review|selectedapprover=@Component Leads@}
  {state}
  {state:Reviewed|approved=Published|rejected=Closed}
  {approval:Publish|group=@FAQ Administrators@}
  {state}
  {state:Published|final=true|updated=Approved|hideselection=true}
  {state}
  {state:Closed}
  {state}
  {state:Rejected}
  {state}
  {trigger:statechanged|state=Submitted}
  {set-metadata:FAQ Author}@user@{set-metadata}
  {send-email:user=@FAQ Administrators@|subject=FAQ Submitted}
  The FAQ @page@ has been submitting by [~@user@] and is waiting for your approval.
  {send-email}
  {trigger}
  {trigger:pageapproved|approval=Approve}
  {send-email:user=@Review>selected@|subject=FAQ Accepted}
  The FAQ @page@ has been approved by [~@user@] and is waiting for your review.
  {send-email}
  {trigger}
  {trigger:pagerejected|approval=Approve}
  {send-email:user=@FAQ Author@|subject=FAQ Rejected}
  Your FAQ @page@ submission has been rejected.
  {send-email}
  {trigger}
  {trigger:pageapproved|approval=Review}
  {send-email:user=@Approve>approver@|subject=FAQ approved}
  The FAQ @page@ has been approved by [~@user@] and is waiting for your approval.
  {send-email}
  {trigger}
  {trigger:pagerejected|approval=Review}
  {send-email:user=@Approve>approver@, @FAQ Author@|subject=FAQ rejected}
  The FAQ @page@ has been rejected by [~@user@].
  {send-email}
  {trigger}
{workflow}

```

Page access permissions

We need to set the FAQ access permissions.

- When an FAQ is **submitted**, we want only the Author and the FAQ administrators to view it.
- When the administrator **approves**, we want the Component lead to view it.
- When it is **published**, we want everybody to view it.

We use the `{add-restriction}` and `{remove-restriction}` macros for this.

First, we need to add restrict the page to be visible only to the Author and the `faq-administrators` when the page is created:

```
{trigger:pagecreated}
  {add-restriction:type=View|user=@user@}
  {add-restriction:type=View|group=@FAQ Administrators@}
{trigger}
```

Then, when the FAQ is approved by the administrator, we need to allow the selected Component lead to view the page. Add an extra action to the existing **Approve** trigger:

```
{trigger:pageapproved|approval=Approve}
  {send-email:user=@Review>selected@|subject=FAQ Accepted}
    The FAQ @page@ submitted by [~@FAQ Author@] has been approved by [~@user@]
    and is waiting for your review.
  {send-email}
  {add-restriction:type=View|user=@Review>selected@}
{trigger}
```

Then, when the FAQ is published by the administrator, we need to remove the view restrictions:

```
{trigger:pageapproved|approval=Publish}
  {send-email:user=@Approve>approver@|subject=FAQ published}
    The FAQ @page@ has been published
  {send-email}
  {remove-restriction:type=View}
{trigger}
```

Updating a published FAQ

Finally, we just need to add a trigger to loop the workflow to require the Component lead approval:

```
{trigger:pageupdated|state=Published}
  {send-email:user=@Review>selected@|subject=FAQ updated, requires approval}
    The FAQ @page@ has been updated by [~@user@] and requires your approval
  {send-email}
{trigger}
```

Whenever a published FAQ is updated, the approval process is re-applied:

The final workflow

This is what the final workflow looks like:

```
{workflow}
  {workflowparameter:FAQ Administrators|type=group}
    confluence-users
  {workflowparameter}
  {workflowparameter:Component Leads|type=group}
    confluence-users
  {workflowparameter}
  {state:Draft|submit=Submitted}
  {state}
  {state:Submitted|approved=Approved|rejected=Rejected}
    {approval:Approve|group=@FAQ Administrators@}
  {state}
  {state:Approved|approved=Reviewed|rejected=Closed}
    {approval:Review|selectedapprover=@Component Leads@}
  {state}
  {state:Reviewed|approved=Published|rejected=Closed}
    {approval:Publish|group=@FAQ Administrators@}
  {state}
  {state:Published|final=true|updated=Approved|hideselection=true}
```

```

{state}
{state:Closed}
{state}
{state:Rejected}
{state}
{trigger:statechanged|state=Submitted}
  {set-metadata:FAQ Author}@user@{set-metadata}
  {send-email:user=@FAQ Administrators@|subject=FAQ Submitted}
  The FAQ @page@ has been submitting by [~@user@] and is waiting for your approval.
  {send-email}
{trigger}
{trigger:pageapproved|approval=Approve}
  {send-email:user=@Review>selected@|subject=FAQ Accepted}
  The FAQ @page@ has been approved by [~@user@] and is waiting for your review.
  {send-email}
{trigger}
{trigger:pagerejected|approval=Approve}
  {send-email:user=@FAQ Author@|subject=FAQ Rejected}
  Your FAQ @page@ submission has been rejected.
  {send-email}
{trigger}
{trigger:pageapproved|approval=Review}
  {send-email:user=@Approve>approver@|subject=FAQ approved}
  The FAQ @page@ has been approved by [~@user@] and is waiting for your approval.
  {send-email}
{trigger}
{trigger:pagerejected|approval=Review}
  {send-email:user=@Approve>approver@, @FAQ Author@|subject=FAQ rejected}
  The FAQ @page@ has been rejected by [~@user@].
  {send-email}
{trigger}
{trigger:pagecreated}
  {add-restriction:type=View|user=@user@}
  {add-restriction:type=View|group=@FAQ Administrators@}
{trigger}
{trigger:pageapproved|approval=Approve}
  {send-email:user=@Review>selected@|subject=FAQ Accepted}
  The FAQ @page@ submitted by [~@FAQ Author@] has been approved by [~@user@]
  and is waiting for your review.
  {send-email}
  {add-restriction:type=View|user=@Review>selected@}
{trigger}
{trigger:pageapproved|approval=Publish}
  {send-email:user=@Approve>approver@|subject=FAQ published}
  The FAQ @page@ has been published
  {send-email}
  {remove-restriction:type=View}
{trigger}
{trigger:pageupdated|state=Published}
  {send-email:user=@Review>selected@|subject=FAQ updated, requires approval}
  The FAQ @page@ has been updated by [~@user@] and requires your approval
  {send-email}
{trigger}
{workflow}

```