

SaltConf21: Adding Workflow Approval to Salt

Jira Settings

Custom Fields

Minions

The ID of the Salt minions.

▼ Select List (multiple choices)

Linux Minions

The ID of the Salt minions.

▼ Select List (multiple choices)

Linux Package

The system package and version to install, upgrade or downgrade to, or remove.

▼ Select List (cascading)

Windows Minions

The ID of the Salt minions.






▼ Select List (multiple choices)

Windows Package

The system package and version to install, upgrade or downgrade to, or remove.

▼ Select List (cascading)

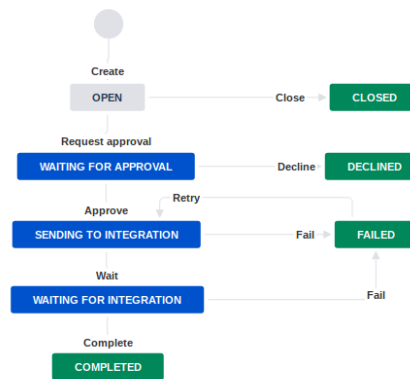
Issue Types

Name 	Type 	Related Schemes
<div> Linux Package Management Request</div> <div>Request to manage a Linux package on a Salt minion.</div>	Base	<div><ul style="list-style-type: none">Default Issue Type SchemeSC21: Jira ITSM Issue Type Scheme</div> <div>...</div>
<div> Reboot Request</div> <div>Request to reboot a Salt minion.</div>	Base	<div><ul style="list-style-type: none">Default Issue Type SchemeSC21: Jira ITSM Issue Type Scheme</div> <div>...</div>
<div> Windows Package Management Request</div> <div>Request to manage a Windows package on a Salt minion.</div>	Base	<div><ul style="list-style-type: none">Default Issue Type SchemeSC21: Jira ITSM Issue Type Scheme</div> <div>...</div>

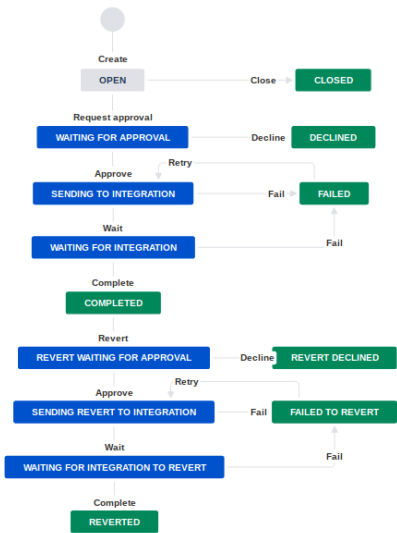
Workflows

Workflow	Issue Types	Actions
SC21: Automated ITSM Request With Revert Workflow (View as text / diagram)	<div><div></div> Linux Package Management Request</div> <div><div></div> Windows Package Management Request (Assign)</div>	<div><div></div> <div></div></div>
SC21: Automated ITSM Request Workflow (View as text / diagram)	<div><div></div> Reboot Request (Assign)</div>	<div><div></div> <div></div></div>

SC21: Automated ITSM Request Workflow



SC21: Automated ITSM Request With Revert Workflow



Automations

<div>Rules Audit log Library</div>				
<div>Filter rules</div>	Name		Owner	Projects
<div>All rules</div> <div>Global rules</div> <div>Project rules</div> <div>Add label</div>	SC21: Linux Package Management Request Automation		Automation for Jira	SaltConf21
	SC21: Package Management Revert Automation		Automation for Jira	SaltConf21
	SC21: Reboot Request Automation		Automation for Jira	SaltConf21
	SC21: Request Sync		Automation for Jira	SaltConf21
	SC21: Windows Package Management Request Automation		Automation for Jira	SaltConf21

SC21: Request Sync

SC21: Request Sync

🕒 Rule details

📄 Audit log

📅 When: Scheduled

Every day at 5:00 AM

🔗 Then: Send web request

POST http://13.89.191.219:8080/sync

⊕ Add component

Rule details

Name *

SC21: Request Sync

Description

Request a sync of custom field options from the automation.

Scope

SaltConf21

Scope can only be modified in the [global administration](#).

Allow rule trigger

☐ Check to allow other rule actions to trigger this rule. Only enable this if you need this rule to execute in response to another rule.

Notify on error

Don't notify

Owner *

Automation for Jira

The owner will receive emails when the rule fails.

Created

a day ago

Updated

a day ago

Actor *

Automation for Jira

Actions defined in this rule will be performed by the user selected as the actor.

Save

SC21: Request Sync

🕒 Rule details

📄 Audit log

📅 When: Scheduled

Every day at 5:00 AM

🔗 Then: Send web request

POST http://13.89.191.219:8080/sync

⊕ Add component

📅 Scheduled

✎

Executes this rule on the provided schedule. You can choose if you'd like to perform actions on a set of issues gathered with a JQL search or simply execute the rule.

Basic

Advanced (CRON expression)

Run rule every

1 Days

At

5:00 AM

America/Sao Paulo

🕒 Next run: Wednesday, November 3, 2021 5:00 AM -03

Show next 10 runs

☐ Run a JQL search and execute actions for each issue in the query.

Cancel

Save

SC21: Request Sync

① Rule details

📄 Audit log

📅 When: Scheduled
Every day at 5:00 AM

🔗 Then: Send web request
POST http://13.89.191.219:8080/sync

○ Add component

Send web request 🗑️

This action will send a HTTP request to the url specified below:

Web request URL *

http://13.89.191.219:8080/sync

Request parameters must be url encoded, smart values should use: {{value.urlEncode}}.

Headers (optional)

Name (e.g. Content-Type)

Value (e.g. application/json)

➕ Add

HTTP method *

POST

Web request body *

Empty

Wait for response

☐ Delay execution of subsequent rule actions until we've received a response for this web request

› Validate your web request configuration

Cancel

Save

› How do I access web request response values in subsequent rule actions?

SC21: Reboot Request Automation

SC21: Reboot Request Automation

Rule details

Audit log

When: Issue transitioned

FROM

Waiting for Approval, Failed

TO

Sending to Integration

Issue Type equals

Reboot Request

Then: Send web request

POST http://<address>:8080/reboot

If: Compare two values

Checks if:

`{{webResponse.status}}` does not equal 200

Then: Transition the issue to

FAILED

Add component

Rule details

Name *

SC21: Reboot Request Automation

Description

Reboot request automation for SaltConf21.

Scope

SaltConf21

Scope can only be modified in the [global administration](#).

Allow rule trigger

☐ Check to allow other rule actions to trigger this rule. Only enable this if you need this rule to execute in response to another rule.

Notify on error

Don't notify

Owner *

Automation for Jira

The owner will receive emails when the rule fails.

Created

12 days ago

Updated

a day ago

Actor *

Automation for Jira

Actions defined in this rule will be performed by the user selected as the actor.

Save

SC21: Reboot Request Automation

Rule details

Audit log

When: Issue transitioned

FROM

Waiting for Approval, Failed

TO

Sending to Integration

Issue Type equals

Reboot Request

Then: Send web request

POST http://<address>:8080/reboot

If: Compare two values

Checks if:

`{{webResponse.status}}` does not equal 200

Then: Transition the issue to

FAILED

Add component

Issue transitioned

Rule is run when an issue is transitioned through its workflow.

[Learn more about transitioning an issue with automation.](#)

From status

WAITING FOR APPROVAL x FAILED x

To status

SENDING TO INTEGRATION x

Cancel

Save

SC21: Reboot Request Automation

Rule details

Audit log

When: Issue transitioned
FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals
Reboot Request

Then: Send web request
POST http://<address>:8080/reboot

If: Compare two values
Checks if:
{webResponse.status} does not
equal 200

Then: Transition the issue to
FAILED

Add component

SC21: Reboot Request Automation

Rule details

Audit log

When: Issue transitioned
FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals
Reboot Request

Then: Send web request
POST http://<address>:8080/reboot

If: Compare two values
Checks if:
{webResponse.status} does not
equal 200

Then: Transition the issue to
FAILED

Add component

Issue fields condition

Checks whether an issue's field meets a certain criteria. [Learn more.](#)

Field *

Issue Type

Condition *

equals

Value

Field

Reboot Request

Cancel

Save

Send web request

This action will send a HTTP request to the url specified below:

Web request URL *

http://<address>:8080/reboot

Request parameters must be url encoded, smart values should use: {{value.urlEncode}}.

Headers (optional)

Content-Type

application/json

Add

HTTP method *

POST

Web request body *

Custom data

Custom data *

```
{
  "itsm_id": {{issue.key.asJsonString}},
  "minion_ids": {{issue.Minions.asJsonStringArray}}
}
```

Wait for response

☒ Delay execution of subsequent rule actions until we've received a response for this web request

☒ Continue running the rule even if the request response is not successful (i.e. non-200 response)

SC21: Reboot Request Automation

- Rule details
- Audit log

When: Issue transitioned
FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals
Reboot Request

Then: Send web request
POST http://<address>:8080/reboot

If: Compare two values
Checks if:
{{webResponse.status}} does not
equal 200

Then: Transition the issue to
FAILED

Add component

SC21: Reboot Request Automation

- Rule details
- Audit log

When: Issue transitioned
FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals
Reboot Request

Then: Send web request
POST http://<address>:8080/reboot

If: Compare two values
Checks if:
{{webResponse.status}} does not
equal 200

Then: Transition the issue to
FAILED

Add component

Advanced compare condition

Compares two values using smart values and regular expressions. [Learn more about issue properties for Advanced compare condition.](#)

First value *

{{webResponse.status}}

Condition

does not equal

Second value

200

Cancel

Save

> What values can I compare?

Transition issue

Transitions an issue from one status to another, through a workflow. [Learn more.](#)

Choose the status to transition the issue to:

Destination status

FAILED

Ensure a transition from the issue's source status to your selected destination status exists; [more info.](#)

+ add regex to distinguish between multiple transitions to the same status

Choose fields to set...

> More options

Cancel

Save

SC21: Linux Package Management Request Automation

SC21: Linux Package Management Request Automation

ⓘ Rule details

📄 Audit log

🔗 When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

⚙️ Issue Type equals

Linux Package Management Request

📡 Then: Send web request

POST http://<address>:8080/install

⚖️ If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

🔗 Then: Transition the issue to

FAILED

⊕ Add component

SC21: Linux Package Management Request Automation

ⓘ Rule details

📄 Audit log

🔗 When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

⚙️ Issue Type equals

Linux Package Management Request

📡 Then: Send web request

POST http://<address>:8080/install

⚖️ If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

🔗 Then: Transition the issue to

FAILED

⊕ Add component

Rule details

Name *

SC21: Linux Package Management Request Automation

Description

Linux package management request automation for SaltConf21.

Scope

SaltConf21

Scope can only be modified in the [global administration](#).

Allow rule trigger

☐ Check to allow other rule actions to trigger this rule. Only enable this if you need this rule to execute in response to another rule.

Notify on error

Don't notify

Owner *

Automation for Jira

The owner will receive emails when the rule fails.

Created

12 days ago

Updated

a day ago

Actor *

Automation for Jira

Actions defined in this rule will be performed by the user selected as the actor.

Save

🔗 Issue transitioned

Rule is run when an issue is transitioned through its workflow.
[Learn more about transitioning an issue with automation.](#)

From status

WAITING FOR APPROVAL x FAILED x

To status

SENDING TO INTEGRATION x

Cancel Save

SC21: Linux Package Management Request Automation

① Rule details

📄 Audit log

🔗 When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

⚙️ Issue Type equals

Linux Package Management Request

🔗 Then: Send web request

POST http://<address>:8080/install

⚙️ If: Compare two values

Checks if:
{{webResponse.status}} does not
equal 200

🔗 Then: Transition the issue to

FAILED

○ Add component

SC21: Linux Package Management Request Automation

① Rule details

📄 Audit log

🔗 When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

⚙️ Issue Type equals

Linux Package Management Request

🔗 Then: Send web request

POST http://<address>:8080/install

⚙️ If: Compare two values

Checks if:
{{webResponse.status}} does not
equal 200

🔗 Then: Transition the issue to

FAILED

... ..

⚙️ Issue fields condition 🗑

Checks whether an issue's field meets a certain criteria. [Learn more.](#)

Field *

Issue Type

Condition *

equals

Value

Field

Linux Package Management Request

Cancel

Save

🔗 Send web request 🗑

This action will send a HTTP request to the url specified below:

Web request URL *

http://<address>:8080/install

Request parameters must be url encoded, smart values should use: {{value.urlEncode}}.

Headers (optional)

Content-Type

application/json

+ Add

HTTP method *

POST

Web request body *

Custom data

Custom data *

```
{
  "itsm_id": {{issue.key.asJsonString}},
  "minion_ids": {{issue.Linux Minions.asJsonStringArray}},
  "package_name": {{issue.Linux Package.value.asJsonString}},
  "package_version": {{issue.Linux Package.child.value.asJsonString}},
  "after": "{{issue.After.jiraDateTime}}"
}
```

Wait for response

☒ Delay execution of subsequent rule actions until we've received a response for this web request

☒ Continue running the rule even if the request response is not successful (i.e. non-200 response)

SC21: Linux Package Management Request Automation

Rule details

Audit log

When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals

Linux Package Management Request

Then: Send web request

POST http://<address>:8080/install

If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to

FAILED

Add component

SC21: Linux Package Management Request Automation

Rule details

Audit log

When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals

Linux Package Management Request

Then: Send web request

POST http://<address>:8080/install

If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to

FAILED

Add component

Advanced compare condition

Compares two values using smart values and regular expressions. [Learn more about issue properties for Advanced compare condition.](#)

First value *

{{webResponse.status}}

Condition

does not equal

Second value

200

Cancel

Save

> What values can I compare?

Transition issue

Transitions an issue from one status to another, through a workflow. [Learn more.](#)

Choose the status to transition the issue to:

Destination status

FAILED

Ensure a transition from the issue's source status to your selected destination status exists; [more info.](#)

+ add regex to distinguish between multiple transitions to the same status

Choose fields to set...

> More options

Cancel

Save

SC21: Windows Package Management Request Automation

SC21: Windows Package Management Request Automation

Rule details

Audit log

When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals

Windows Package Management Request

Then: Send web request

POST http://<address>:8080/install

If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to

FAILED

Add component

SC21: Windows Package Management Request Automation

Rule details

Audit log

When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals

Windows Package Management Request

Then: Send web request

POST http://<address>:8080/install

If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to

FAILED

Add component

Rule details

Name*

SC21: Windows Package Management Request Automation

Description

Windows package management request automation for SaltConf21.

Scope

SaltConf21

Scope can only be modified in the [global administration](#).

Allow rule trigger

☐ Check to allow other rule actions to trigger this rule. Only enable this if you need this rule to execute in response to another rule.

Notify on error

Don't notify

Owner*

Automation for Jira

The owner will receive emails when the rule fails.

Created

7 days ago

Updated

a day ago

Actor*

Automation for Jira

Actions defined in this rule will be performed by the user selected as the actor.

Save

Issue transitioned

Rule is run when an issue is transitioned through its workflow.
[Learn more about transitioning an issue with automation.](#)

From status

WAITING FOR APPROVAL x FAILED x

To status

SENDING TO INTEGRATION x

Cancel

Save

SC21: Windows Package Management Request Automation

Rule details

Audit log

When: Issue transitioned
FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals
Windows Package Management Request

Then: Send web request
POST http://<address>:8080/install

If: Compare two values
Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to
FAILED

Add component

SC21: Windows Package Management Request Automation

Rule details

Audit log

When: Issue transitioned
FROM
Waiting for Approval, Failed
TO
Sending to Integration

Issue Type equals
Windows Package Management Request

Then: Send web request
POST http://<address>:8080/install

If: Compare two values
Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to
FAILED

Issue fields condition

Checks whether an issue's field meets a certain criteria. [Learn more.](#)

Field *

Issue Type

Condition *

equals

Value

Field

Windows Package Management Request

Cancel

Save

Send web request

This action will send a HTTP request to the url specified below:

Web request URL *

http://<address>:8080/install

Request parameters must be url encoded, smart values should use: {{value.urlEncode}}.

Headers (optional)

Content-Type

application/json

Add

HTTP method *

POST

Web request body *

Custom data

Custom data *

```
{
  "itsm_id": {{issue.key.asJsonString}},
  "minion_ids": {{issue.Windows Minions.asJsonStringArray}},
  "package_name": {{issue.Windows Package.value.asJsonString}},
  "package_version": {{issue.Windows Package.child.value.asJsonString}},
  "after": "{{issue.After.jiraDateTime}}"
}
```

Wait for response

☒ Delay execution of subsequent rule actions until we've received a response for this web request

☒ Continue running the rule even if the request response is not successful (i.e. non-200 response)

SC21: Windows Package Management Request Automation

🕒 Rule details

📄 Audit log

🔗 When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

⚙️ Issue Type equals

Windows Package Management Request

🔗 Then: Send web request

POST http://<address>:8080/install

⚙️ If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

🔗 Then: Transition the issue to

FAILED

○ Add component

SC21: Windows Package Management Request Automation

🕒 Rule details

📄 Audit log

🔗 When: Issue transitioned

FROM
Waiting for Approval, Failed
TO
Sending to Integration

⚙️ Issue Type equals

Windows Package Management Request

🔗 Then: Send web request

POST http://<address>:8080/install

⚙️ If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

🔗 Then: Transition the issue to

FAILED

○ Add component

⚙️ Advanced compare condition 🗑️

Compares two values using smart values and regular expressions. [Learn more about issue properties for Advanced compare condition.](#)

First value *

{{webResponse.status}}

Condition

does not equal

Second value

200

Cancel

Save

> What values can I compare?

🔗 Transition issue 🗑️

Transitions an issue from one status to another, through a workflow. [Learn more.](#)

Choose the status to transition the issue to:

Destination status

FAILED

Ensure a transition from the issue's source status to your selected destination status exists; [more info.](#)

+ add regex to distinguish between multiple transitions to the same status

⚙️ Choose fields to set... ▾

> More options

Cancel

Save

SC21: Package Management Revert Automation

SC21: Package Management Revert Automation

🕒 Rule details

📄 Audit log

🔗 When: Issue transitioned

FROM
Revert Waiting for Approval, Failed to Revert
TO
Sending Revert to Integration

⚙️ Issue Type is one of

Linux Package Management Request, Windows Package Management Request

📡 Then: Send web request

POST http://<address>:8080/revert

🔗 If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

🔗 Then: Transition the issue to

FAILED TO REVERT

SC21: Package Management Revert Automation

🕒 Rule details

📄 Audit log

🔗 When: Issue transitioned

FROM
Revert Waiting for Approval, Failed to Revert
TO
Sending Revert to Integration

⚙️ Issue Type is one of

Linux Package Management Request, Windows Package Management Request

📡 Then: Send web request

POST http://<address>:8080/revert

🔗 If: Compare two values

Checks if:
{{webResponse.status}} does not equal 200

🔗 Then: Transition the issue to

FAILED TO REVERT

Rule details

Name *

SC21: Package Management Revert Automation

Description

Package management revert automation for SaltConf21.

Scope

SaltConf21

Scope can only be modified in the [global administration](#).

Allow rule trigger

☐ Check to allow other rule actions to trigger this rule. Only enable this if you need this rule to execute in response to another rule.

Notify on error

Don't notify

Owner *

Automation for Jira

The owner will receive emails when the rule fails.

Created

12 days ago

Updated

a day ago

Actor *

Automation for Jira

Actions defined in this rule will be performed by the user selected as the actor.

Save

🔗 Issue transitioned

Rule is run when an issue is transitioned through its workflow.
[Learn more about transitioning an issue with automation.](#)

From status

REVERT WAITING FOR APPROVAL x FAILED TO REVERT x

To status

SENDING REVERT TO INTEGRATION x

Cancel Save

SC21: Package Management Revert Automation

- Rule details
- Audit log

When: Issue transitioned
FROM
Revert Waiting for Approval, Failed to Revert
TO
Sending Revert to Integration

Issue Type is one of
Linux Package Management Request, Windows Package Management Request

Then: Send web request
POST http://<address>:8080/revert

If: Compare two values
Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to
FAILED TO REVERT

SC21: Package Management Revert Automation

- Rule details
- Audit log

When: Issue transitioned
FROM
Revert Waiting for Approval, Failed to Revert
TO
Sending Revert to Integration

Issue Type is one of
Linux Package Management Request, Windows Package Management Request

Then: Send web request
POST http://<address>:8080/revert

If: Compare two values
Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to
FAILED TO REVERT

Issue fields condition

Checks whether an issue's field meets a certain criteria. [Learn more.](#)

Field *

Issue Type

Condition *

is one of

Value Field

Linux Package Management Request
Windows Package Management Request

Cancel

Save

Send web request

This action will send a HTTP request to the url specified below:

Web request URL *

http://<address>:8080/revert

Request parameters must be url encoded, smart values should use: {{value.urlEncode}}.

Headers (optional)

Content-Type

application/json

+ Add

HTTP method *

POST

Web request body *

Custom data

Custom data *

```
{
  "itsm_id": {{issue.key.asJsonString}}
}
```

Wait for response

- ☒ Delay execution of subsequent rule actions until we've received a response for this web request
- ☒ Continue running the rule even if the request response is not successful (i.e. non-200 response)

SC21: Package Management Revert Automation

- Rule details
- Audit log

When: Issue transitioned
FROM
Revert Waiting for Approval, Failed to Revert
TO
Sending Revert to Integration

Issue Type is one of
Linux Package Management Request, Windows Package Management Request

Then: Send web request
POST http://<address>:8080/revert

If: Compare two values
Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to
FAILED TO REVERT

SC21: Package Management Revert Automation

- Rule details
- Audit log

When: Issue transitioned
FROM
Revert Waiting for Approval, Failed to Revert
TO
Sending Revert to Integration

Issue Type is one of
Linux Package Management Request, Windows Package Management Request

Then: Send web request
POST http://<address>:8080/revert

If: Compare two values
Checks if:
{{webResponse.status}} does not equal 200

Then: Transition the issue to
FAILED TO REVERT

Advanced compare condition

Compares two values using smart values and regular expressions. [Learn more about issue properties for Advanced compare condition.](#)

First value *

{{webResponse.status}}

Condition

does not equal

Second value

200

Cancel Save

> What values can I compare?

Transition issue

Transitions an issue from one status to another, through a workflow. [Learn more.](#)

Choose the status to transition the issue to:

Destination status

FAILED TO REVERT

Ensure a transition from the issue's source status to your selected destination status exists; [more info.](#)

+ add regex to distinguish between multiple transitions to the same status

Choose fields to set...

> More options

Cancel Save