Notification APIs

Notification [notifications.ait.al NotificationGroup [notificationg Notification

Notification [notifications.ait.alauda.io/v1beta1]

Description

Notification is the Schema for the notifications API

Type

`object`

Required

`spec`

Specification

Property	Type	Description
`apiVersion`	`string`	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources ↗

Property	Type	Description
`kind`	`string`	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	NotificationSpec defines the desired state of Notification

.spec

Description

NotificationSpec defines the desired state of Notification

Type

`object`

Required

`default`

Property	Type	Description
`default`	`object`	Default subscriptions for this notification

.spec.default

Description

Default subscriptions for this notification

Type

`object`

Required

`methods` `template`

Property	Type	Description
`groups`	`array`	Groups for this subscription
`methods`	`array`	Methods for this subscription
`template`	`string`	Template to render notification message
`templateNamespace`	`string`	Namespace of the template to render notification message. If empty, the notification namespace is used.
`users`	`array`	Users for this subscription

.spec.default.groups

Description

Groups for this subscription

Type

`array`

.spec.default.groups[]

Type

`string`

.spec.default.methods

Description

Methods for this subscription

Type

`array`

.spec.default.methods[]

Type

`string`

.spec.default.users

Description

Users for this subscription

Type

`array`

.spec.default.users[]

Type

`string`

API Endpoints

The following API endpoints are available:

- `/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notifications`
  - `DELETE`: delete collection of Notification
  - `GET`: list objects of kind Notification
  - `POST`: create a new Notification
- `/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notifications/{name}`
  - `DELETE`: delete the specified Notification
  - `GET`: read the specified Notification
  - `PATCH`: partially update the specified Notification
  - `PUT`: replace the specified Notification

/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notifications

HTTP method

`DELETE`

Description

delete collection of Notification

HTTP responses

HTTP code	Response body
200 - OK	`Status` schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

list objects of kind Notification

HTTP responses

HTTP code	Response body
200 - OK	`NotificationList` schema
401 - Unauthorized	Empty

HTTP method

`POST`

Description

create a new Notification

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Parameter	Type	Description
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
`body`	`Notification` schema	`application/json` formatted

HTTP responses

HTTP code	Response body
200 - OK	`Notification` schema
201 - Created	`Notification` schema
202 - Accepted	`Notification` schema
401 - Unauthorized	Empty

/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notifications/{name}

HTTP method

`DELETE`

Description

delete the specified Notification

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	`Status` schema
202 - Accepted	`Status` schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

read the specified Notification

HTTP responses

HTTP code	Response body
200 - OK	`Notification` schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update the specified Notification

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	`Notification` schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace the specified Notification

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Parameter	Type	Description
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
`body`	`Notification` schema	`application/json` formatted

HTTP responses

HTTP code	Response body
200 - OK	`Notification` schema
201 - Created	`Notification` schema
401 - Unauthorized	Empty

NotificationGroup [notificationgroups.ait.alauda.io/v1beta1]

Description

NoticationGroup is the Schema for the templates API

Type

`object`

Required

`spec`

Specification

Property	Type	Description
`apiVersion`	`string`	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources ↗

Property	Type	Description
`kind`	`string`	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	NotificationGroupSpec defines the desired state of NotificationGroup

.spec

Description

NotificationGroupSpec defines the desired state of NotificationGroup

Type

`object`

Property	Type	Description
`addresses`	`array`	Addresses for notification group
`mail`	`string`	Mail for notification group

Property	Type	Description
`users`	`array`	Users for notification group
`webhookType`	`string`	WebhookType for notification group
`webhookUrl`	`string`	WebhookUrl for notification group

.spec.addresses

Description

Addresses for notification group

Type

`array`

.spec.addresses[]

Description

NotificationGroupAddress defines the desired state of NotificationGroup

Type

`object`

Required

`type`

Property	Type	Description
`type`	`string`	AddressType defines template type

Property	Type	Description
`url`	`string`	Url for notification group

.spec.users

Description

Users for notification group

Type

`array`

.spec.users[]

Type

`string`

API Endpoints

The following API endpoints are available:

- `/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notificationgroups`
  - `DELETE`: delete collection of NotificationGroup
  - `GET`: list objects of kind NotificationGroup
  - `POST`: create a new NotificationGroup
- `/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notificationgroups/{name}`
  - `DELETE`: delete the specified NotificationGroup
  - `GET`: read the specified NotificationGroup
  - `PATCH`: partially update the specified NotificationGroup

- `PUT`: replace the specified NotificationGroup

/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notificationgroups

HTTP method

`DELETE`

Description

delete collection of NotificationGroup

HTTP responses

HTTP code	Response body
200 - OK	`Status` schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

list objects of kind NotificationGroup

HTTP responses

HTTP code	Response body
200 - OK	`NotificationGroupList` schema
401 - Unauthorized	Empty

HTTP method

`POST`

Description

create a new NotificationGroup

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
`body`	`NotificationGroup` schema	`application/json` formatted

HTTP responses

HTTP code	Response body
200 - OK	`NotificationGroup` schema

HTTP code	Response body
201 - Created	`NotificationGroup` schema
202 - Accepted	`NotificationGroup` schema
401 - Unauthorized	Empty

/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notificationgroups/{name}

HTTP method

`DELETE`

Description

delete the specified NotificationGroup

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	`Status` schema
202 - Accepted	`Status` schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

read the specified NotificationGroup

HTTP responses

HTTP code	Response body
200 - OK	`NotificationGroup` schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update the specified NotificationGroup

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the

Parameter	Type	Description
		request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	`NotificationGroup` schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace the specified NotificationGroup

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The

Parameter	Type	Description
		request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
`body`	`NotificationGroup` schema	`application/json` formatted

HTTP responses

HTTP code	Response body
200 - OK	`NotificationGroup` schema
201 - Created	`NotificationGroup` schema
401 - Unauthorized	Empty

NotificationTemplate [notificationtemplates.ait.alauda.io/v1beta1]

Description

NotificationTemplate is the Schema for the templates API

Type

`object`

Required

`spec`

Specification

Property	Type	Description
`apiVersion`	`string`	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources ↗

Property	Type	Description
`kind`	`string`	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	NotificationTemplateSpec defines the desired state of NotificationTemplate

.spec

Description

NotificationTemplateSpec defines the desired state of NotificationTemplate

Type

`object`

Property	Type	Description
`email`	`object`	NotificationTemplateEmail defines the desired state of email NotificationTemplate

Property	Type	Description
`sms`	`object`	NotificationTemplateCommon defines the desired state of common NotificationTemplate
`templates`	`array`	
`webhook`	`object`	NotificationTemplateCommon defines the desired state of common NotificationTemplate

.spec.email

Description

NotificationTemplateEmail defines the desired state of email NotificationTemplate

Type

`object`

Required

`content`

Property	Type	Description
`content`	`string`	Content template to render notification message
`subject`	`string`	Subject template to render notification message

.spec.sms

Description

NotificationTemplateCommon defines the desired state of common NotificationTemplate

Type

`object`

Required

`content`

Property	Type	Description
`content`	`string`	Content template to render notification message

.spec.templates

Type

`array`

.spec.templates[]

Description

NotificationTemplateDetail defines the desired state of common NotificationTemplateDetail

Type

`object`

Required

`content` `type`

Property	Type	Description
`content`	`string`	Content template to render notification message
`subject`	`string`	

Property	Type	Description
`type`	`string`	TemplateType defines template type

.spec.webhook

Description

NotificationTemplateCommon defines the desired state of common NotificationTemplate

Type

`object`

Required

`content`

Property	Type	Description
`content`	`string`	Content template to render notification message

API Endpoints

The following API endpoints are available:

- `/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notificationtemplates`
  - `DELETE`: delete collection of NotificationTemplate
  - `GET`: list objects of kind NotificationTemplate
  - `POST`: create a new NotificationTemplate
- `/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notificationtemplates/{name}`
  - `DELETE`: delete the specified NotificationTemplate

- `GET` : read the specified NotificationTemplate
- `PATCH` : partially update the specified NotificationTemplate
- `PUT` : replace the specified NotificationTemplate

/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notificationtemplates

HTTP method

`DELETE`

Description

delete collection of NotificationTemplate

HTTP responses

HTTP code	Response body
200 - OK	`Status` schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

list objects of kind NotificationTemplate

HTTP responses

HTTP code	Response body
200 - OK	`NotificationTemplateList` schema
401 - Unauthorized	Empty

HTTP method

`POST`

Description

create a new NotificationTemplate

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
`body`	`NotificationTemplate` schema	`application/json` formatted

HTTP responses

HTTP code	Response body
200 - OK	`NotificationTemplate` schema
201 - Created	`NotificationTemplate` schema
202 - Accepted	`NotificationTemplate` schema
401 - Unauthorized	Empty

/apis/ait.alauda.io/v1beta1/namespaces/{namespace}/notificationtemplates/{name}

HTTP method

`DELETE`

Description

delete the specified NotificationTemplate

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	`Status` schema
202 - Accepted	`Status` schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

read the specified NotificationTemplate

HTTP responses

HTTP code	Response body
200 - OK	`NotificationTemplate` schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update the specified NotificationTemplate

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields.

Parameter	Type	Description
		This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	`NotificationTemplate` schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace the specified NotificationTemplate

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object,

Parameter	Type	Description
		and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
`body`	`NotificationTemplate` schema	`application/json` formatted

HTTP responses

HTTP code	Response body
200 - OK	`NotificationTemplate` schema
201 - Created	`NotificationTemplate` schema
401 - Unauthorized	Empty