
Yolapy Documentation

Release 0.0.1

Yola

September 03, 2015

1	Service Class	1
1.1	Partners	1
1.2	Users	2
1.3	Sites	3
1.4	Subscriptions	5
1.5	Campaigns	6
2	Indices and tables	7
	Python Module Index	9

Service Class

The `Yola` class is a thin wrapper over the Yola API.

```
class yolapy.services.Yola (**kwargs)
    Client for Yola's API.
```

```
yola = Yola(
    url='https://wl.yola.net/',
    auth=('username', 'password'))

yola.get_user('user_id')
```

When appropriate, successful responses will return parsed json objects.

Failures will raise instances of *demands.HTTPServiceError*.

```
__init__ (**kwargs)
    Initialize with url, auth, and optional headers.
```

```
Yola(
    url='https://wl.yola.net/',
    auth=('username', 'password'),
    headers={'Header-Name': 'value'})
```

1.1 Partners

The following methods are mixed in to the `Yola` class to provide access to the Partner resource:

```
class yolapy.resources.partner.PartnerResourceMixin
    Methods for managing Partner resources.
```

```
create_partner (**attributes)
    Create a partner.
```

```
>>> partner = yola.create_partner(
    id='COMPANY_ID',
    name='Company Name',
    parent_partner_id='PARENT_ID',
    properties={'name': 'value'})
>>> partner['name']
'Company Name'
```

See <https://wl.qa.yola.net/partners/> for available properties.

delete_partner (*partner_id*)

Delete a partner.

```
>>> yola.delete_partner('PARTNER_ID')
```

get_partner (*partner_id*)

Return details for a particular partner.

```
>>> partner = yola.get_partner('PARTNER_ID')
>>> partner['name']
'Company Name'
```

list_partners (**options)

Return paginated list of partners.

```
>>> yola.list_partners()
{
  'count': 999,
  'previous': None,
  'next': 'https://wl.qa.yola.net/pr/partners/?page=2',
  'results': [{...}, {...}, ...]
}
```

You can pass `page_size` and `page` as keyword arguments:

```
>>> yola.list_partners(page_size=50, page=2)
```

update_partner (*partner_id*, **attributes)

Update a partner.

```
>>> yola.update_partner('PARTNER_ID', name='New name')
```

1.2 Users

The following methods are mixed in to the `Yola` class to provide access to the User resource:

class `yolapy.resources.user.UserResourceMixin`

Methods for managing User resources.

create_user (**attributes)

Create a user.

```
>>> user = yola.create_user(
    name='John',
    surname='Smith',
    email='johnsmith@example.com',
    partner_id='WL_PARTNER_ID',
    preferences={'preference_name': 'preference_value'})
>>> user['name']
'John'
```

delete_user (*user_id*)

Delete a user.

```
>>> yola.delete_user('user_id')
```

get_sso_create_site_url (*user_id*, *domain*)

Get SSO create site url for a particular user and domain.

```
>>> yola.get_sso_create_site_url('user_id', 'example.com')
```

get_sso_open_site_url (*user_id*)
Get SSO open site url for a particular user.

```
>>> yola.get_sso_open_site_url('user_id')
```

get_user (*user_id*)
Get details for a particular user.

```
>>> user = yola.get_user('user_id')
>>> user['name']
'John'
```

list_users (***filters*)
Return paginated list of users.

```
>>> yola.list_users()
{
  'count': 999,
  'previous': None,
  'next': 'https://wl.qa.yola.net/pr/users/?page=2',
  'results': [
    {'name': 'John', 'surname': 'Smith', ...}
  ]
}
```

If there are no users, `results` will be an empty list. No exception will be raised.

You may pass pagination options and attribute filters as keyword arguments. See <https://wl.qa.yola.net/users/> for available parameters.

For example:

```
>>> yola.list_users(page=2, page_size=50, partner_id='WL_YOLA')
```

resume_user (*user_id*)
Resume a user.

```
>>> yola.resume_user('user_id')
```

suspend_user (*user_id*)
Suspend a user.

```
>>> yola.suspend_user('user_id')
```

update_user (*user_id*, ***attributes*)
Update a user.

```
>>> yola.update_user('user_id', name='New name')
```

1.3 Sites

The following methods are mixed in to the `Yola` class to provide access to the Site resource:

class `yolapy.resources.site.SiteResourceMixin`
Methods for managing Site resources.

change_site_domain (*site_id*, *new_domain*)
Change site's domain.

```
>>> yola.change_site_domain('site_id', 'newdomain.com')
```

change_site_owner (*site_id*, *new_user_id*)
Change site owner.

```
>>> yola.change_site_owner('site_id', 'new_user_id')
```

delete_site (*site_id*)
Delete a site.

```
>>> yola.get_site('site_id')['deleted_at']
None
>>> yola.delete_site('site_id')
>>> yola.get_site('site_id')['deleted_at']
'2015-08-27T10:00:07'
```

disable_site (*site_id*)
Disable a site.

```
>>> yola.disable_site('site_id')
```

enable_site (*site_id*)
Enable a site.

```
>>> yola.enable_site('site_id')
```

get_site (*site_id*)
Return details for a particular site.

```
>>> site = yola.get_site('site_id')
>>> site['name']
'My Site'
```

list_sites (**options)
Return paginated list of sites.

```
>>> yola.list_sites()
{
  'count': 999,
  'previous': None,
  'next': 'https://wl.qa.yola.net/pr/sites/?page=2',
  'results': [{...}, {...}, ...]
}
```

You can pass `page_size` and `page` as keyword arguments:

```
>>> yola.list_sites(page_size=50, page=2)
```

You can also pass filters and ordering options as keyword arguments. See <https://wl.qa.yola.net/sites/> for available options.

undelete_site (*site_id*)
Un-delete a site.

```
>>> yola.get_site('site_id')['deleted_at']
'2015-08-27T10:00:07'
>>> yola.delete_site('site_id')
```



```
>>> yola.get_site('site_id')['deleted_at']
None
```

1.4 Subscriptions

The following methods are mixed in to the `Yola` class to provide access to the Subscription resource:

class `yolapy.resources.subscription.SubscriptionResourceMixin`
Methods for managing Subscription resources.

activate_trial_subscription (*subscription_id*)
Convert trial subscription to active.

```
>>> yola.activate_trial_subscription('subscription_id')
```

cancel_subscription (*subscription_id, reason*)
Cancel active subscription.

```
>>> yola.cancel_subscription('subscription_id', 'some reason')
```

change_subscription_type (*subscription_id, new_type*)
Change subscription type.

```
>>> yola.change_subscription_type('subscription_id', 'new_type')
```

See <https://wl.qa.yola.net/subscriptions/> for available types.

create_subscription (*subscription_type, properties*)
Create a new subscription.

```
>>> properties = {...}
>>> yola.create_subscription('subscription_type', properties)
```

See <https://wl.qa.yola.net/subscriptions/> for available types and properties.

get_subscription (*subscription_id*)
Return details for a particular subscription.

```
>>> subscription = yola.get_subscription('subscription_id')
>>> subscription['name']
'My Subscription'
```

list_subscriptions (***options*)
Return paginated list of subscriptions.

```
>>> yola.list_subscriptions()
{
  'count': 999,
  'previous': None,
  'next': 'https://wl.qa.yola.net/pr/subscriptions/?page=2',
  'results': [{...}, {...}, ...]
}
```

You can pass `page_size` and `page` as keyword arguments:

```
>>> yola.list_subscriptions(page_size=50, page=2)
```

You can also pass filters and ordering options as keyword arguments. See <https://wl.qa.yola.net/subscriptions/> for available options.

reactivate_subscription (*subscription_id, reason*)

Re-activate a cancelled subscription.

```
>>> yola.reactivate_subscription('subscription_id', 'some reason')
```

1.5 Campaigns

The following methods are mixed in to the `Yola` class to provide access to the Campaign resource:

class `yolapy.resources.campaign.CampaignResourceMixin`

Methods for managing Campaign resources.

cancel_campaign_subscription (*site_id, campaign_id*)

Cancel a campaign subscription.

delete_campaign (*site_id, campaign_id*)

Delete a campaign.

get_campaign (*site_id, campaign_id*)

Return details for a particular campaign.

```
>>> campaign = yola.get_campaign('site_id', 'campaign_id')
>>> campaign['name']
'My Campaign'
```

list_campaigns (*site_id, **options*)

Return list of campaigns for a site.

subscribe_to_campaign (*site_id, campaign_id*)

Subscribe to a campaign.

```
>>> params = {...}
>>> yola.subscribe_to_campaign('site_id', 'campaign_id', params)
```

See https://wl.qa.yola.net/sites/SITE_ID/campaigns/ for expected params.

Indices and tables

- `genindex`
- `modindex`
- `search`

`yolapy` provides a `Yola` service class for interacting directly with the Yola API. Future versions will provide higher-level abstractions.

y

`yolapy.resources.campaign`, 6
`yolapy.resources.partner`, 1
`yolapy.resources.site`, 3
`yolapy.resources.subscription`, 5
`yolapy.resources.user`, 2

Symbols

- `__init__()` (yolapy.services.Yola method), 1
- ### A
- `activate_trial_subscription()` (yolapy.resources.subscription.SubscriptionResourceMixin method), 5
- ### C
- `CampaignResourceMixin` (class in yolapy.resources.campaign), 6
- `cancel_campaign_subscription()` (yolapy.resources.campaign.CampaignResourceMixin method), 6
- `cancel_subscription()` (yolapy.resources.subscription.SubscriptionResourceMixin method), 5
- `change_site_domain()` (yolapy.resources.site.SiteResourceMixin method), 3
- `change_site_owner()` (yolapy.resources.site.SiteResourceMixin method), 4
- `change_subscription_type()` (yolapy.resources.subscription.SubscriptionResourceMixin method), 5
- `create_partner()` (yolapy.resources.partner.PartnerResourceMixin method), 1
- `create_subscription()` (yolapy.resources.subscription.SubscriptionResourceMixin method), 5
- `create_user()` (yolapy.resources.user.UserResourceMixin method), 2
- ### D
- `delete_campaign()` (yolapy.resources.campaign.CampaignResourceMixin method), 6
- `delete_partner()` (yolapy.resources.partner.PartnerResourceMixin method), 1
- `delete_site()` (yolapy.resources.site.SiteResourceMixin method), 4
- `delete_user()` (yolapy.resources.user.UserResourceMixin method), 2
- `disable_site()` (yolapy.resources.site.SiteResourceMixin method), 4
- ### E
- `enable_site()` (yolapy.resources.site.SiteResourceMixin method), 4
- ### G
- `get_campaign()` (yolapy.resources.campaign.CampaignResourceMixin method), 6
- `get_partner()` (yolapy.resources.partner.PartnerResourceMixin method), 2
- `get_site()` (yolapy.resources.site.SiteResourceMixin method), 4
- `get_sso_create_site_url()` (yolapy.resources.user.UserResourceMixin method), 2
- `get_sso_open_site_url()` (yolapy.resources.user.UserResourceMixin method), 3
- `get_subscription()` (yolapy.resources.subscription.SubscriptionResourceMixin method), 5
- `get_user()` (yolapy.resources.user.UserResourceMixin method), 3
- ### L
- `list_campaigns()` (yolapy.resources.campaign.CampaignResourceMixin method), 6
- `list_partners()` (yolapy.resources.partner.PartnerResourceMixin method), 2
- `list_sites()` (yolapy.resources.site.SiteResourceMixin method), 4
- `list_subscriptions()` (yolapy.resources.subscription.SubscriptionResourceMixin method), 5
- `list_users()` (yolapy.resources.user.UserResourceMixin method), 3
- ### P
- `PartnerResourceMixin` (class in yolapy.resources.partner), 1

R

reactivate_subscription() (yolapy.resources.subscription.SubscriptionResourceMixin method), 5
resume_user() (yolapy.resources.user.UserResourceMixin method), 3

S

SiteResourceMixin (class in yolapy.resources.site), 3
subscribe_to_campaign() (yolapy.resources.campaign.CampaignResourceMixin method), 6
SubscriptionResourceMixin (class in yolapy.resources.subscription), 5
suspend_user() (yolapy.resources.user.UserResourceMixin method), 3

U

undelele_site() (yolapy.resources.site.SiteResourceMixin method), 4
update_partner() (yolapy.resources.partner.PartnerResourceMixin method), 2
update_user() (yolapy.resources.user.UserResourceMixin method), 3
UserResourceMixin (class in yolapy.resources.user), 2

Y

Yola (class in yolapy.services), 1
yolapy.resources.campaign (module), 6
yolapy.resources.partner (module), 1
yolapy.resources.site (module), 3
yolapy.resources.subscription (module), 5
yolapy.resources.user (module), 2