
Django Google Map using Fusion Tables Documentation

Release 1.0.2

Tonye Jack

Dec 13, 2019

Contents:

1	map_app	1
1.1	map_app module	1
2	Indices and tables	5
	Python Module Index	7
	Index	9

1.1 map_app module

1.1.1 Submodules

map_app.decorators

class map_app.decorators.OAuth2Decorator (*client_id, secret, scope*)

class OAuth2Decorator

oauth_required (*method*)

Decorator that starts the OAuth 2.0 dance.

Starts the OAuth dance for the logged in user if they haven't already granted access for this application.

Args:

method: callable, to be decorated method of a webapp.RequestHandler instance.

map_app.forms module

class map_app.forms.AddressForm (*data=None, files=None, auto_id='id_%s', pre-fix=None, initial=None, error_class=<class 'django.forms.utils.ErrorList'>, label_suffix=None, empty_permitted=False, instance=None, use_required_attribute=None*)

Address form validator

Validate the address is unique and it's geocode.

save (*commit=True, request=None*)

Save this form's self.instance object if commit=True. Otherwise, add a save_m2m() method to the form which can be called after the instance is saved manually at a later time. Return the model instance.

map_app.models module

class map_app.models.CredentialsModel (*id, credential*)

exception DoesNotExist

exception MultipleObjectsReturned

class map_app.models.UserTokens (**args, **kwargs*)

User token model use for google OAuth authentication.

exception DoesNotExist

exception MultipleObjectsReturned

map_app.utils module

map_app.utils.messages_to_dict (*request, to_str=False*)

:py:function *message_to_dict*

Returns an iterator that yields a JSON formatted string or dictionary.

Retrieve the _messages from an instance of django.http.request.HttpRequest. :param request: An instance of django.http.request.HttpRequest, :type request: django.http.request.HttpRequest :param to_str: Boolean indication to serialize message object to string. :type to_str: bool :rtype: strdict

map_app.utils.set_fusion_table_cookie (*response*)

Set the fusion_table_id cookie using the settings.FUSION_TABLE_ID.

Parameters response (django.http.HttpResponse) – An instance of response object.

Returns An instance of the HttpResponse object.

Return type django.http.HttpResponse

map_app.utils.verify_table_id_cookie_set (*request, response*)

Verify the google fusion table_id is set on the request.

Parameters

- **request** (django.http.request.HttpRequest) – HttpRequest object
- **response** (django.http.response.HttpResponse) – HttpResponse object

map_app.views module

class map_app.views.FusionTableHandler (***kwargs*)

Bases: django.views.generic.base.TemplateView, [map_app.lib.FusionTableMixin](#)

map_app.views.address_view (*request*)

Render a table of valid searched/clicked addresses.

map_app.lib module

```
class map_app.lib.FlowClient (request, client_secret_json='/home/docs/checkouts/readthedocs.org/user_builds/django-  
google-app/checkouts/latest/client_id.json',  
scope='https://www.googleapis.com/auth/fusiontables',      redi-  
rect_url='http://localhost:8000/oauth2callback')
```

FlowClient to manage credentials.

```
class map_app.lib.FusionTableMixin
```

Mixin to manage Interactions with google fusion table.

```
classmethod address_model_to_dict (address)
```

Convert the map_app.models.Address instance to a dict.

Parameters **address** (map_app.models.Address) – The address model instance.

Return type dict

```
map_app.lib.verify_client_id_json (filename)
```

Verify the required client_id.json values are set.

A context manager to parse the client_id.json and set the missing values with the env vars. :param filename:

Client id json filename. :type filename: str

1.1.2 Module Contents

CHAPTER 2

Indices and tables

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

m

- `map_app`, 3
- `map_app.decorators`, 1
- `map_app.forms`, 1
- `map_app.lib`, 3
- `map_app.models`, 2
- `map_app.utils`, 2
- `map_app.views`, 2

A

`address_model_to_dict()`
 (*map_app.lib.FusionTableMixin* class method), 3
`address_view()` (*in module map_app.views*), 2
`AddressForm` (*class in map_app.forms*), 1

C

`CredentialsModel` (*class in map_app.models*), 2
`CredentialsModel.DoesNotExist`, 2
`CredentialsModel.MultipleObjectsReturned`, 2

F

`FlowClient` (*class in map_app.lib*), 3
`FusionTableHandler` (*class in map_app.views*), 2
`FusionTableMixin` (*class in map_app.lib*), 3

M

`map_app` (*module*), 3
`map_app.decorators` (*module*), 1
`map_app.forms` (*module*), 1
`map_app.lib` (*module*), 3
`map_app.models` (*module*), 2
`map_app.utils` (*module*), 2
`map_app.views` (*module*), 2
`messages_to_dict()` (*in module map_app.utils*), 2

O

`OAuth2Decorator` (*class in map_app.decorators*), 1
`OAuth2Decorator.OAuth2Decorator` (*class in map_app.decorators*), 1
`oauth_required()` (*map_app.decorators.OAuth2Decorator* method), 1

S

`save()` (*map_app.forms.AddressForm* method), 1
`set_fusion_table_cookie()` (*in module map_app.utils*), 2

U

`UserTokens` (*class in map_app.models*), 2
`UserTokens.DoesNotExist`, 2
`UserTokens.MultipleObjectsReturned`, 2

V

`verify_client_id_json()` (*in module map_app.lib*), 3
`verify_table_id_cookie_set()` (*in module map_app.utils*), 2