

---

# **Terralego Documentation**

***Release 0.1***

**Gagaro**

**Jul 03, 2017**



---

## Contents

---

<b>1</b>	<b>Getting started</b>	<b>1</b>
1.1	Geocoding . . . . .	1
1.2	Geodirectory . . . . .	2
<b>2</b>	<b>Indices and tables</b>	<b>5</b>
	<b>Python Module Index</b>	<b>7</b>



# CHAPTER 1

---

## Getting started

---

Install using pip:

```
pip install terralego
```

Set your credentials using the environment variables:

```
export TERRALEGO_USER="my_user"
export TERRALEGO_PASSWORD="my_password"
```

You can now use terralego:

```
from terralego import geocoding

results = geocoding.search('paris france')
```

Contents:

## Geocoding

`terralego.geocoding.autocomplete` (*text*, *params=None*)  
Autocomplete locations from a string.

### Parameters

- **text** – The string which will be used to do the autocomplete.
- **params** – A dict including other get parameters.

**Returns** A geojson including the results.

`terralego.geocoding.reverse` (*lat*, *long*, *params=None*)  
Search addresses from a location.

### Parameters

- **lat** – The latitude of the location.
- **long** – The longitude of the location.
- **params** – A dict including other get parameters.

**Returns** A geojson including the results.

`terralego.geocoding.search(text, params=None)`  
Search locations from a string.

**Parameters**

- **text** – The string which will be used to do the search.
- **params** – A dict including other get parameters.

**Returns** A geojson including the results.

## Geodirectory

`terralego.geodirectory.closest(entry_id, tags=None)`  
Get the closest entry.

**Parameters**

- **entry\_id** – The id of the entry on which to get the closest one.
- **tags** – Optional, a list of tags to filter the entry which can be the closests.

**Returns** A geojson describing the entry as a python dictionary. Raise 404 if no entry are found.

`terralego.geodirectory.closest_from(lat, long, tags=None, dist=None)`  
Get the closest entry from the point.

**Parameters**

- **lat** – The latitude of the point.
- **long** – The longitude of the point.
- **tags** – Optional, a list of tags to filter the entry which can be the closests.
- **dist** – Optional, a distance in meters.

**Returns** A geojson describing the entry as a python dictionary. Raise 404 if no entry are found.

`terralego.geodirectory.create_entry(geometry, tags=None)`  
Create a new entry.

**Parameters**

- **geometry** – A WKT string representing the geometry of the entry or a dict representing the geojson.
- **tags** – A list of string describing the entry. Can be used for filtering later on.

**Returns** A geojson describing the entry as a python dictionary.

`terralego.geodirectory.delete_entry(entry_id)`  
Delete an entry.

**Parameters** **entry\_id** – The id of the entry.

`terralego.geodirectory.get_entry(entry_id)`  
Get an entry.

**Parameters** `entry_id` – The id of the entry.

**Returns** A geojson describing the entry as a python dictionary.

`terralego.geodirectory.update_entry(entry_id, geometry, tags=None)`

Update an entry.

**Parameters**

- **entry\_id** – The id of the entry.
- **geometry** – A WKT string representing the geometry of the entry or a dict representing the geojson.
- **tags** – A list of string describing the entry. Can be used for filtering later on.

**Returns** A geojson describing the updated entry as a python dictionary.





## CHAPTER 2

---

### Indices and tables

---

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



### t

`terralego.geocoding`, [1](#)  
`terralego.geodirectory`, [2](#)



### A

`autocomplete()` (in module `terralego.geocoding`), 1

### C

`closest()` (in module `terralego.geodirectory`), 2

`closest_from()` (in module `terralego.geodirectory`), 2

`create_entry()` (in module `terralego.geodirectory`), 2

### D

`delete_entry()` (in module `terralego.geodirectory`), 2

### G

`get_entry()` (in module `terralego.geodirectory`), 2

### R

`reverse()` (in module `terralego.geocoding`), 1

### S

`search()` (in module `terralego.geocoding`), 2

### T

`terralego.geocoding` (module), 1

`terralego.geodirectory` (module), 2

### U

`update_entry()` (in module `terralego.geodirectory`), 3