

---

# XRPL Websocket Client

*Release 0.1.0.rc1*

**Aug 10, 2021**



---

## Contents:

---

<b>1</b>	<b>xrpl_websocket package</b>	<b>3</b>
1.1	Submodules . . . . .	3
1.2	xrpl_websocket.client module . . . . .	3
1.3	xrpl_websocket.exceptions module . . . . .	4
1.4	Module contents . . . . .	4
<b>2</b>	<b>Indices and tables</b>	<b>5</b>
	<b>Python Module Index</b>	<b>7</b>
	<b>Index</b>	<b>9</b>



Release v0.1.0.rc1



## 1.1 Submodules

### 1.2 xrpl\_websocket.client module

```
class xrpl_websocket.client.Client (server=None, timeout=None, log_level=None, *args,  
                                     **kwargs)
```

Bases: `threading.Thread`

Higher level of APIs are provided. The interface is like JavaScript WebSocket object.

```
__init__ (server=None, timeout=None, log_level=None, *args, **kwargs)
```

#### Parameters

- **server** – rippled node url.
- **timeout** – connection timeout seconds
- **log\_level** – loggin level
- **on\_open** – callable object which is called at opening websocket.
- **on\_reconnect** – callable object which is called at reconnecting
- **on\_error** – callable object which is called when we get error.
- **on\_close** – callable object which is called when closed the connection.
- **on\_transaction** – callback object which is called when we recieve transacion
- **on\_ledger** – callback object which is called when we recieve ledger close
- **on\_validation** – callback object by the validations stream when the server receives a validation message
- **on\_manifest** – callback object sent by the manifests stream when the server receives a manifest.

**connect** (*nowait=True*)

Simulate self.start(), run the main thread

**Returns**

**disconnect** ()

Disconnects from the websocket connection and joins the Thread.

**Returns**

**reconnect** ()

Issues a reconnection by setting the reconnect\_required event.

**Returns**

**run** ()

Main method of Thread.

**Returns**

**send** (*payload=None, \*\*kwargs*)

Sends the given Payload to the API via the websocket connection.

**Parameters**

- **payload** –
- **kwargs** – payload parameters as key=value pairs

**Returns**

**send\_ping** ()

Sends a ping message to the API and starts pong timers.

**Returns**

## 1.3 xrpl\_websocket.exceptions module

**exception** `xrpl_websocket.exceptions.ConnectionError` (*message, data*)

Bases: `xrpl_websocket.exceptions.Error`

**exception** `xrpl_websocket.exceptions.DisconnectedError` (*message, data*)

Bases: `xrpl_websocket.exceptions.ConnectionError`

**exception** `xrpl_websocket.exceptions.Error` (*message, data*)

Bases: `Exception`

**exception** `xrpl_websocket.exceptions.NotConnectedError` (*message, data*)

Bases: `xrpl_websocket.exceptions.ConnectionError`

**exception** `xrpl_websocket.exceptions.ResponseFormatError` (*message, data*)

Bases: `xrpl_websocket.exceptions.ConnectionError`

**exception** `xrpl_websocket.exceptions.TimeoutError` (*message, data*)

Bases: `xrpl_websocket.exceptions.ConnectionError`

**exception** `xrpl_websocket.exceptions.UnexpectedError` (*message, data*)

Bases: `xrpl_websocket.exceptions.Error`

## 1.4 Module contents



## CHAPTER 2

---

### Indices and tables

---

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



### X

`xrpl_websocket`, 4

`xrpl_websocket.client`, 3

`xrpl_websocket.exceptions`, 4



## Symbols

`__init__()` (*xrpl\_websocket.client.Client method*), 3

## C

`Client` (*class in xrpl\_websocket.client*), 3

`connect()` (*xrpl\_websocket.client.Client method*), 3

`ConnectionError`, 4

## D

`disconnect()` (*xrpl\_websocket.client.Client method*),  
4

`DisconnectedError`, 4

## E

`Error`, 4

## N

`NotConnectedError`, 4

## R

`reconnect()` (*xrpl\_websocket.client.Client method*), 4

`ResponseFormatError`, 4

`run()` (*xrpl\_websocket.client.Client method*), 4

## S

`send()` (*xrpl\_websocket.client.Client method*), 4

`send_ping()` (*xrpl\_websocket.client.Client method*), 4

## T

`TimeoutError`, 4

## U

`UnexpectedError`, 4

## X

`xrpl_websocket` (*module*), 4

`xrpl_websocket.client` (*module*), 3

`xrpl_websocket.exceptions` (*module*), 4