



Client

Contents

- `WhatsApp`

The `WhatsApp` client has 3 main responsibilities:

1. Sending messages (text, media, location, contact, etc.)
2. Creating and managing templates, flows, profile and other business-related resources
3. Listening for incoming messages and events

for sending messages and the other API calls, you will need to provide an phone number id that connects to the WhatsApp Cloud API. for listening to incoming messages and events, you will need to tell WhatsApp to send the events to a webhook that you provide.

Tip

Pywa provides two clients, synchronous and asynchronous, you can choose the one that fits your needs.

```
from pywa import WhatsApp, types
wa = WhatsApp(...)

@wa.on_message()
def on_message(_: WhatsApp, msg: types.Message):
    msg.reply("Hello!")
```

```
from pywa_async import WhatsApp, types
wa = WhatsApp(...)

@wa.on_message()
async def on_message(_: WhatsApp, msg: types.Message):
    await msg.reply("Hello!")
```

`class pywa.client.WhatsApp`

```
    __init__(phone_id: str | int | None = None, token: str = None,
             base_url: str = 'https://graph.facebook.com', api_version: str | int |
```

```
float | ~typing.Literal[<Version.GRAPH_API: '19.0'>] =
Version.GRAPH_API, session: ~httpx.Client | None = None, server:
~pywa.utils.Flask | ~pywa.utils.FastAPI | None = <object object>,
webhook_endpoint: str = '/', verify_token: str | None = None,
filter_updates: bool = True, business_account_id: str | int | None =
None, callback_url: str | None = None, webhook_fields:
~typing.Iterable[str] | None = None, app_id: int | None = None,
app_secret: str | None = None, webhook_challenge_delay: int = 3,
business_private_key: str | None = None,
business_private_key_password: str | None = None,
flows_request_decryptor: ~typing.Callable[[str, str, str, str, str |
None], tuple[dict, bytes, bytes]] | None = <function
default_flow_request_decryptor>, flows_response_encryptor:
~typing.Callable[[dict, bytes, bytes], str] | None = <function
default_flow_response_encryptor>, continue_handling: bool = True,
skip_duplicate_updates: bool = True, validate_updates: bool = True,
**kwargs: ~typing.Any) → None
```

The WhatsApp client.

- Full documentation on pywa.readthedocs.io.

Example without webhook:

```
>>> from pywa import WhatsApp
>>> wa = WhatsApp(phone_id="100944", token="EAADKQ19oJxx")
```

Example with webhook (using Flask):

```
>>> from pywa import WhatsApp
>>> from flask import Flask
>>> flask_app = Flask(__name__)
>>> wa = WhatsApp(
...     phone_id="100944",
...     token="EAADKQ19oJxx",
...     server=flask_app,
...     callback_url='https://6b3e.ngrok.io',
...     verify_token="XYZ123",
...     app_id=1234567890,
...     app_secret="my_app_secret",
... )
>>> @wa.on_message()
... def message_handler(_: WhatsApp, msg: Message): print(msg)
>>> flask_app.run(port=8000)
```

Parameters:

- **phone_id** – The Phone number ID to send messages from (if you manage multiple WhatsApp business accounts (e.g. partner solutions), you can specify the phone ID when sending messages, optional).
- **token** – The token of the WhatsApp business account (In production, you should [use permanent token](#)).
- **base_url** – The base URL of the WhatsApp API (Do not change unless you know what you're doing).
- **api_version** – The API version of the WhatsApp Cloud API (default to the latest version).
- **session** – The session to use for requests (default: new `httpx.Client()`, For cases where you want to use a custom session, e.g. for proxy support. Do not use the same session across multiple WhatsApp clients!).
- **server** – The Flask or FastAPI app instance to use for the webhook. required when you want to handle incoming updates.
- **callback_url** – The server URL to register (without endpoint. optional).
- **verify_token** – The verify token of the registered `callback_url` (Required when `server` is provided. The verify token can be any string. It is used to challenge the webhook endpoint to verify that the endpoint is valid).
- **webhook_challenge_delay** – The delay (in seconds, default to 3) to wait for the verify token to be sent to the server (optional, for cases where the server/network is slow or the server is taking a long time to start).
- **webhook_fields** – The fields to register for the callback URL (optional, if not provided, all supported fields will be registered. modify this if you want to reduce the number of unused requests to your server).
- **app_id** – The ID of the app in the [App Basic Settings](#) (optional, required when registering a `callback_url`).
- **app_secret** –
The secret of the app in the [App Basic Settings](#) (optional, recommended for validating updates, required when registering a `callback_url`).
- **webhook_endpoint** – The endpoint to listen for incoming messages (if you're using the server for another purpose, or for multiple WhatsApp clients, you can change this to avoid conflicts).
- **filter_updates** – Whether to filter out user updates that are not sent to this phone_id (default: `True`, does not apply to raw updates or updates that are not user-related).
- **business_account_id** – The WhatsApp business account ID (waba id) that owns the phone ID (optional, required for some API methods).

- **business_private_key** – The global private key to use in the `flows_request_decryptor`
- **business_private_key_password** – The global private key password (if needed) to use in the `flows_request_decryptor`
- **flows_request_decryptor** – The global flows requests decryptor implementation to use to decrypt Flows requests.
- **flows_response_encryptor** – The global flows response encryptor implementation to use to encrypt Flows responses.
- **continue_handling** – Whether to continue handling updates after a handler has been found (default: `True`).
- **skip_duplicate_updates** – Whether to skip duplicate updates (default: `True`).
- **validate_updates** – Whether to validate updates payloads (default: `True`, `app_secret` required).

The available methods are:

Category	Methods
Sending messages	<code>send_message()</code> , <code>send_image()</code> , <code>send_video()</code> , <code>send_audio()</code> , <code>send_document()</code> , <code>send_location()</code> , <code>send_contact()</code> , <code>send_sticker()</code> , <code>send_template()</code> , <code>send_catalog()</code> , <code>send_product()</code> , <code>send_products()</code> , <code>send_reaction()</code> , <code>remove_reaction()</code> , <code>mark_message_as_read()</code>
Media	<code>upload_media()</code> , <code>download_media()</code> , <code>get_media_url()</code>
Templates	<code>create_template()</code>
Flows	<code>create_flow()</code> , <code>update_flow_metadata()</code> , <code>update_flow_json()</code> , <code>publish_flow()</code> , <code>delete_flow()</code> , <code>deprecate_flow()</code> , <code>get_flow()</code> , <code>get_flows()</code> , <code>get_flow_metrics()</code> , <code>get_flow_assets()</code>
Business profile	<code>get_business_profile()</code> , <code>get_business_phone_number()</code> , <code>update_business_profile()</code> , <code>update_conversational_automation()</code> , <code>set_business_public_key()</code> , <code>register_phone_number()</code>
QR Codes	<code>create_qr_code()</code> , <code>get_qr_code()</code> , <code>get_qr_codes()</code> , <code>update_qr_code()</code> , <code>delete_qr_code()</code>
Commerce	<code>get_commerce_settings()</code> , <code>update_commerce_settings()</code>
Server	<code>webhook_update_handler()</code> , <code>webhook_challenge_handler()</code> , <code>get_flow_request_handler()</code>

Client Reference

`WhatsApp.send_message()`

[`WhatsApp.send_image\(\)`](#)[`WhatsApp.send_video\(\)`](#)[`WhatsApp.send_audio\(\)`](#)[`WhatsApp.send_document\(\)`](#)[`WhatsApp.send_location\(\)`](#)[`WhatsApp.send_contact\(\)`](#)[`WhatsApp.send_sticker\(\)`](#)[`WhatsApp.send_catalog\(\)`](#)[`WhatsApp.send_template\(\)`](#)[`WhatsApp.send_product\(\)`](#)[`WhatsApp.send_products\(\)`](#)[`WhatsApp.send_reaction\(\)`](#)[`WhatsApp.remove_reaction\(\)`](#)[`WhatsApp.mark_message_as_read\(\)`](#)[`WhatsApp.upload_media\(\)`](#)[`WhatsApp.download_media\(\)`](#)[`WhatsApp.get_media_url\(\)`](#)[`WhatsApp.get_business_profile\(\)`](#)[`WhatsApp.get_business_phone_number\(\)`](#)[`WhatsApp.update_business_profile\(\)`](#)[`WhatsApp.update_conversational_automation\(\)`](#)[`WhatsApp.set_business_public_key\(\)`](#)[`WhatsApp.get_commerce_settings\(\)`](#)[`WhatsApp.update_commerce_settings\(\)`](#)[`WhatsApp.create_template\(\)`](#)[`WhatsApp.create_flow\(\)`](#)[`WhatsApp.update_flow_metadata\(\)`](#)[`WhatsApp.update_flow_json\(\)`](#)[`WhatsApp.publish_flow\(\)`](#)[`WhatsApp.delete_flow\(\)`](#)[`WhatsApp.deprecate_flow\(\)`](#)

[`WhatsApp.get_flow\(\)`](#)[`WhatsApp.get_flows\(\)`](#)[`WhatsApp.get_flow_metrics\(\)`](#)[`WhatsApp.get_flow_assets\(\)`](#)[`WhatsApp.register_phone_number\(\)`](#)[`WhatsApp.create_qr_code\(\)`](#)[`WhatsApp.get_qr_code\(\)`](#)[`WhatsApp.get_qr_codes\(\)`](#)[`WhatsApp.update_qr_code\(\)`](#)[`WhatsApp.delete_qr_code\(\)`](#)[`WhatsApp.webhook_update_handler\(\)`](#)[`WhatsApp.webhook_challenge_handler\(\)`](#)[`WhatsApp.get_flow_request_handler\(\)`](#)

API Reference

[`WhatsAppCloudApi.send_raw_request\(\)`](#)