

2020

# API DOCUMENT

## Wappin

PT. Solusi Pembayaran Elektronik



# Table of Contents

<b>Document Change Control</b> .....	3
<b>1. Introduction</b> .....	4
<b>2. Getting Started</b> .....	4
<b>3. API HSM/Notification</b> .....	4
3.1. About HSM Message Template .....	4
3.2. How to .....	5
3.3. API GET Token .....	7
3.4. API Send Notification .....	10
3.5. API Send Notification with Media .....	12
<b>1. API Chat Bot</b> .....	14
4.1. How to .....	14
4.2. API GET Token .....	18
4.3. API Send Message (Text) .....	20
4.4. API Send Message (Media) .....	22
<b>5. API Inquiry</b> .....	24
<b>6. API Callback</b> .....	26

## Document Change Control

Date	Author	Summary	Version
7 Mei 2019	Rizki Kurniawan	-	V 1.0
20 Mei 2019	Rizki Kurniawan	Add Sample Request and Reponse API	V 1.1
23 Juli 2019	Rizki Kurniawan	<ul style="list-style-type: none"> <li>Update Success Response Code and Add status code</li> <li>Add API Send Message (Media)</li> </ul>	V 1.2
25 Juli 2019	Rizki Kurniawan	<ul style="list-style-type: none"> <li>Update API Inquiry Status Code</li> </ul>	V 1.3
13 Agustus 2019	Rizki Kurniawan	Update Document Screenshoots	V 1.4
10 Oktober 2019	Rizki Kurniawan	Update URL API	V 1.5
28 November 2019	Rizki Kurniawan	Update API Authorization <b>API Get Token</b> -> Basic [base64_encode(username:password)] <b>Others API</b> -> Bearer [token]	V 1.6
06 Desember 2019	Rizki Kurniawan	Update Screenshoot UI	V 1.7
17 Januari 2020	Rizki Kurniawan	<ul style="list-style-type: none"> <li>Translate to English</li> </ul>	V 1.8
06 Mei 2020	Rizki Kurniawan	<ul style="list-style-type: none"> <li>Update API HSM UAT &amp; PROD (3.4 &amp; 3.5)</li> <li>Take Out API UAT</li> </ul>	V 1.9
14 Desember 2020	Adrian Punggara	<ul style="list-style-type: none"> <li>Add API Send Notification With Media</li> </ul>	V 2.0
21 Mei 2021	Idham Khalid	<ul style="list-style-type: none"> <li>Header Parameter &amp; Button API Send notification</li> <li>Button API Send Notification With Media</li> </ul>	V 2.1

## 1. Introduction

WhatsApp is a simple, safe and reliable way for businesses to reach customers all over the world. This guide will help businesses and build their first official WhatsApp messaging experience using the WhatsApp Business API. If you do not have access to the API, please contact the Business solution provider in Indonesia, one of them is SPE ([www.spesolution.com](http://www.spesolution.com)).

## 2. Getting Started

## 3. API HSM/Notification

### 3.1. About HSM Message Template

HSM (Highly Structured Message) is a message that sent by system as a form of notification without being needed to do initiation from the customer.

Notification template example:

Hello, {{1}}!  
Welcome to WAPPIN Service.

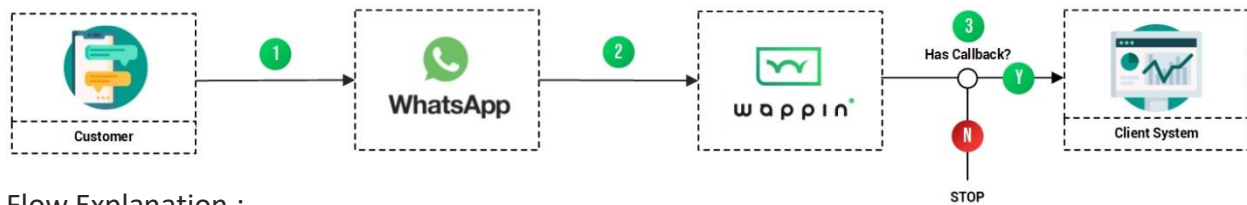
**HSM Flow :**



**Flow Explanation :**

1. Client System sends Notification to Wappin.
2. Then the notification is forwarded to WhatsApp.
3. WhatsApp sends the notification to the destination number.

## HSM Callback Flow :



### Flow Explanation :

1. WhatsApp receives a callback status from the customer message (Sent,Delivered,Read, Failed)
2. WhatsApp sends the callback status to Wappin.
3. Wappin checks if the masking number has a callback URL or not. If it does, then Wappin will send the message status to the client system.

## 3.2. How to

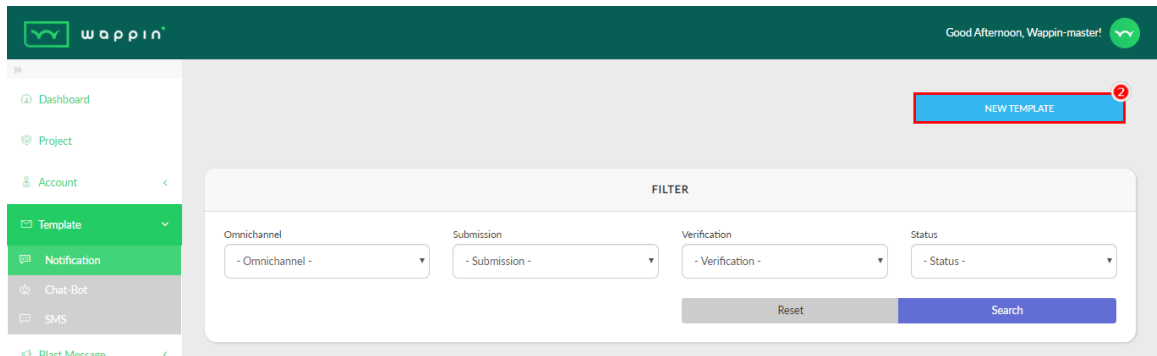
1. Add Integration, to add project integration
  - To add Notification Integration, access Wappin dashboard portal in the Integration Menu and Click Sub Menu **Notification**. Then click New Integration button.

The screenshot shows the Wappin dashboard interface. On the left is a sidebar menu with options: Dashboard, Project, Account, Template, Blast Message, Chat Panel, Contact, Number Checker, History, and Integration. The main content area is titled 'NEW NOTIFICATION INTEGRATION'. It contains three input fields: 'PROJECT' (a dropdown menu with 'Please select a project'), 'CALLBACK URL (UAT)', and 'IP WHITELIST (UAT)'. A blue 'SAVE INTEGRATION' button is at the bottom right. The top of the dashboard has a green header with the Wappin logo and a greeting 'Good Afternoon, Wappin-master!'.

### Step how to do new Integration :

- Choose **Project** that will be Integrated
- Fill in the **Callback URL** to receive the message status from WhatsApp.
- Fill in the **IP Address** field to list IP Address that can access the system.

## 2. Create Notification Template



Add New Template :

The screenshot shows the 'NEW NOTIFICATION TEMPLATE' form. It includes a 'PROJECT' dropdown menu, a 'TEMPLATE NAME' text input field, a 'MESSAGE TYPE' dropdown menu, and a 'CONTENT' text area. A note states: 'Note: You cannot edit message template once its submission is 'On Process' by Wappin'. At the bottom, there are formatting buttons (B, I, S, C) and a '+ NEW VARIABLE' button. A 'REQUEST TEMPLATE' button is at the bottom right. The character count '1024' is visible at the bottom right of the content area.

Explanation :

- **Select Project** – Project that will be used.
- **Template Name** – Name of the Template. It must consist of lowercase, number and underscore **Ex : opening\_message1**
- **Message Type** – Type of notification that will be created
- **Content** – The Notification content
- **Request Template** – Button to process the template request

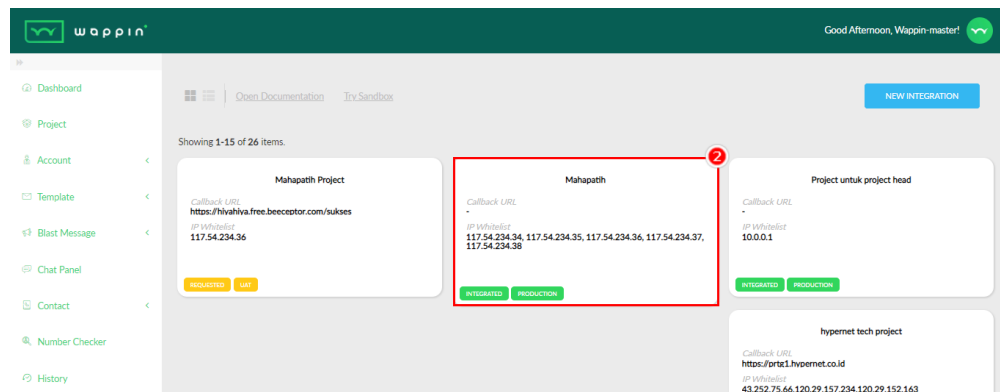
### 3.3. API GET Token

#### 1. Get token description

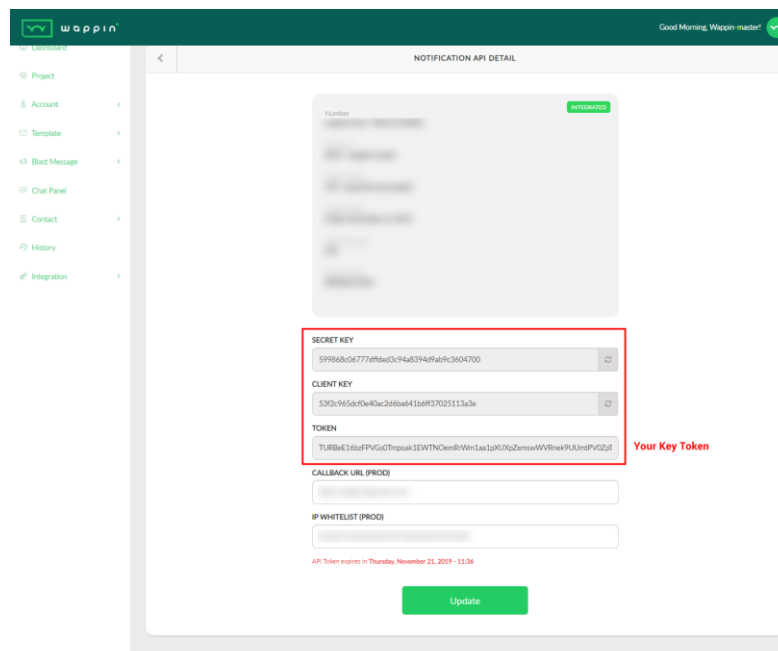
- Is used for Authentication. If you don't use the authentication token when making a request, Wappin will give an error notification.
- Key/Token have 1 week active periode.

#### 2. How to get the key/token


- Go to the Wappin dashboard portal on the Menu Integration, sub-menu Notification
- Select the project to view its Key/Token (shown in the image below)




- #### 3. And then the token will be shown with the project detail from the project that you choose :



4. If the token is Expired, click the regenerate button to get a new token.

 wappin

Good Morning, Wappin-master! 

Dashboard

Project

Account

Template

Blast Message

Chat Panel

Contact

History

Integration


NOTIFICATION API DETAIL

Number

INTEGRATED


SECRET KEY

599868c06777dffded3c94a8394d9ab9c3604700



CLIENT KEY

53f2c965dcf0e40ac2d6ba641b6ff37025113a3e



TOKEN

TURBeE16bzFPVGs0Tnpoak1EWTNOemRrWm1aa1pXUXpZemswWVRnek9UUmPV0ZpI

CALLBACK URL (PROD)

IP WHITELIST (PROD)

API Token expires in Thursday, November 21, 2019 - 11:36

Token Expired Date

Update



## 5. API Get Token

Get Token				
Hostname	<a href="https://api.wappin.id">https://api.wappin.id</a>			
HTTP Method	POST			
Path	/v1/token/get			
HTTP Headers	Name	Format	Mandatory	
	authorization	Basic [base64_encode(username:password)]	Yes	
	content-type	application/json	Yes	
Request Definition	Name	Format	Mandatory	Description
	Username	string	yes	Provided Client ID
	Password	string	yes	Provided Secret Key
Request Header	Content-Type:application/json Authorization: Basic OTo4OWNiNzgyYTBhMmZiNDEyMzgzMzA3MzhjYTgwMDJlZjZhN2YzYjJi			
Success Response Body	<pre>{   "status": "200",   "message": "Success",   "data": {     "access_token": "d49865d016a32ab31734ed6972a818c41808c9f109c3614bec5da2508ef3be20",     "expired_datetime": "2019-07-20 16:21:51",     "token_type": "access_token"   } }</pre>			

## Status Code

Status code	Message	Description
200	1. Success 2. Token not expired	Get Token Success
401	Invalid credential	Invalid credential/token
406	Auth not recognize	Invalid authorization
407	IP Address not allowed	Invalid IP address

### 3.4. API Send Notification

This is API Send Notification for the Production Environment.

Send Notification				
Hostname	<a href="https://api.wappin.id">https://api.wappin.id</a>			
HTTP Method	POST			
Path	/v1/message/do-send-hsm			
HTTP Headers	Name	Format	Mandatory	Description
	Authorization	Bearer [token]	Yes	
	Content-Type	application/json	Yes	
Request Definition	Name	Format	Mandatory	Description
	client_id	string	Yes	Client ID
	project_id	string	Yes	Project ID
	type	string	Yes	Template that has been approved by WhatsApp
	recipient_number	string	Yes	Number must be already registered on WhatsApp
	header param	String	No	Is used when the template is using header type text and variable, other than that params header not necessary to declare.
	params	string	Yes	Insert the parameters
	language_code	String	No	Language code, if its empty use the first key of the template.
Request Body	button	String	No	Parameter for buttons
	<pre>{   "client_id": "0070",   "project_id": "476",   "type": "johan_test_button",   "recipient_number": "6281327666166",   "language_code": "id",   "params": {     "1": "Name",     "2": "8000181008192601",     "3": "034567",     "4": "0345aa67"   },   "button": {     "url" : "bell.com",     "reply_payload":{       "1" : "hsdushfu",</pre>			

	<pre>                 "2" : "sdfsfd",                 "3" : "payload 3"             }         },         "header": {             "param": "johan setiawan"         }     } </pre>
<b>Success Response Body</b>	<pre> {     "message_id": "5bc83037c79f5",     "status": "200",     "message": "Success" } </pre>

### API Send Notification (PROD) Status Code List

Status code	Message	Description
<b>200</b>	Success	Message sent
<b>400</b>	Error no input received	Invalid format
	Project ID or Client ID cannot be process.	Invalid client or project
	Message param count do not match	Number of parameters does not match the expected number of params.
<b>401</b>	Invalid credential	Invalid credential/token
<b>404</b>	API Notification Template not found or inactive	Invalid template
<b>406</b>	Auth not recognize	Invalid authorization
<b>407</b>	IP Address not allowed	Invalid IP address
<b>601</b>	Contact invalid	Recipient number is not a valid WhatsApp number

### 3.5. API Send Notification with Media

This is API Send Notification for the Production Environment.

Send Notification				
Hostname	<a href="https://api.wappin.id">https://api.wappin.id</a>			
HTTP Method	POST			
Path	/v1/message/do-send-hsm-with-media			
Request Definition	Name	Format	Mandatory	Description
	media	File	Yes	Media to be used on message template
	client_id	string	Yes	Client ID
	project_id	string	Yes	Project ID
	type	string	Yes	Template that has been approved by WhatsApp
	recipient_number	string	Yes	Number must be already registered on WhatsApp
	language_code	String	No	Language code, if its empty use the first key of the template.
	header param	String	No	Is used when the template is using header type text and variable, other than that params header not necessary to declare.
	params	Json string	Yes	Insert the parameters Example: {"1":"Name","2":"8000181008192601"}
	button	Json String	No	Parameters for button Example: {"url":"bell.com","reply_payload":{"1":"hsdushfu","2":"sdfsdd","3":"payload 3"}}
Request Body	KEY	VALUE	DESCRIPTION	
	<input checked="" type="checkbox"/> media	IMG_9180.JPG	media override or not	
	<input checked="" type="checkbox"/> client_id	0070		
	<input checked="" type="checkbox"/> project_id	476		
	<input checked="" type="checkbox"/> type	template_name01	template name	
	<input checked="" type="checkbox"/> recipient_number	628989898989	with 62xxx	
	<input checked="" type="checkbox"/> language_code	id	lang code	
	<input checked="" type="checkbox"/> params	{"1":"Name","2":"8000181008192601"}	fill with json	
	<input checked="" type="checkbox"/> button	{"url":"bell.com","reply_payload":{"1":"hsdushfu","2":"sdfsdd","3":"payload 3"}}	fill with json	

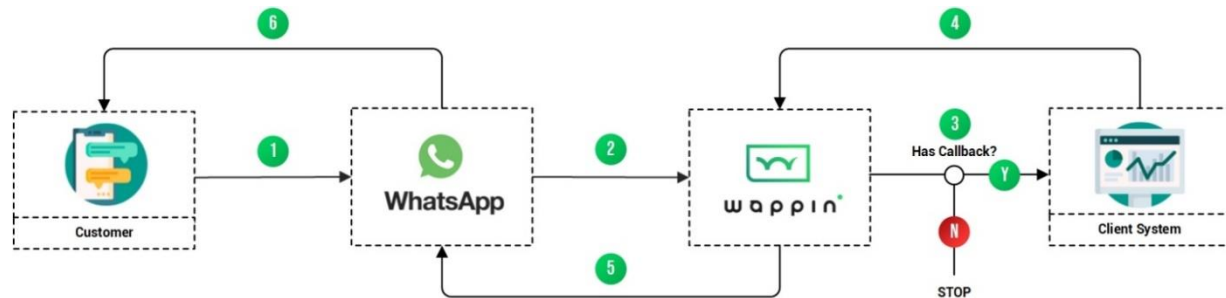
<b>Example json</b>	
<b>Success Response Body</b>	<pre>{   "message_id": "5bc83037c79f5",   "status": "200",   "message": "Success" }</pre>

#### API Send Notification (PROD) Status Code List

Status code	Message	Description
<b>200</b>	Success	Message sent
<b>400</b>	Error no input received	Invalid format
	Project ID or Client ID cannot be process.	Invalid client or project
	Message param count do not match	Number of parameters does not match the expected number of params.
<b>401</b>	Invalid credential	Invalid credential/token
<b>404</b>	API Notification Template not found or inactive	Invalid template
<b>406</b>	Auth not recognize	Invalid authorization
<b>407</b>	IP Address not allowed	Invalid IP address
<b>601</b>	Contact invalid	Recipient number is not a valid WhatsApp number

## 1. API Chat Bot

### Flow Chat Bot (Response from Client) :



### Flow Explanation :


1. Customer sends message keyword via WhatsApp.
2. WhatsApp forwards the message to Wappin.
3. If the client has the callback URL, then Wappin will send the keyword to client system.
4. Client system sends the response to Wappin by using API Send Message ( **See Point 4.34.3 API Send Message (Text)** ).
5. Wappin sends the message to WhatsApp.
6. WhatsApp sends the response to the customer.


### 4.1. How to

Following are the steps to create a Chat bot with a callback URL:

1. First is to do Add Integration  
These are needed to use API Send Message :
  - Callback URL
  - IP to Whitelist

Register through the **Integration** menu, then choose submenu **Chatbot**.



Good Afternoon, Wappin master! 

Dashboard

Project

Account

Template

Blast Message

Chat Panel

Contact

Number Checker

History

Integration

NEW CHATBOT INTEGRATION

PROJECT

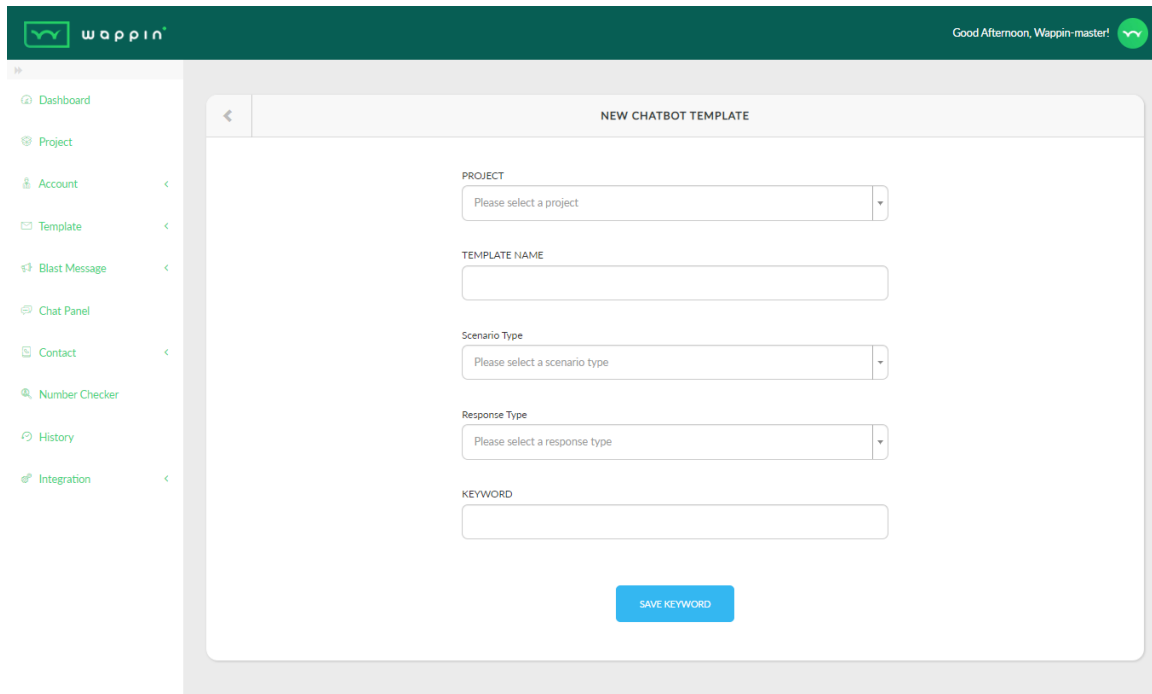
Please select a project

CALLBACK URL (UAT)

IP WHITELIST (UAT)

SAVE INTEGRATION

## 2. Create the Chatbot Template



The screenshot shows the Wappin dashboard with a sidebar menu on the left containing: Dashboard, Project, Account, Template, Blast Message, Chat Panel, Contact, Number Checker, History, and Integration. The main content area is titled 'NEW CHATBOT TEMPLATE' and contains the following fields:

- PROJECT**: A dropdown menu with the placeholder text 'Please select a project'.
- TEMPLATE NAME**: A text input field.
- Scenario Type**: A dropdown menu with the placeholder text 'Please select a scenario type'.
- Response Type**: A dropdown menu with the placeholder text 'Please select a response type'.
- KEYWORD**: A text input field.

A blue button labeled 'SAVE KEYWORD' is located at the bottom right of the form.

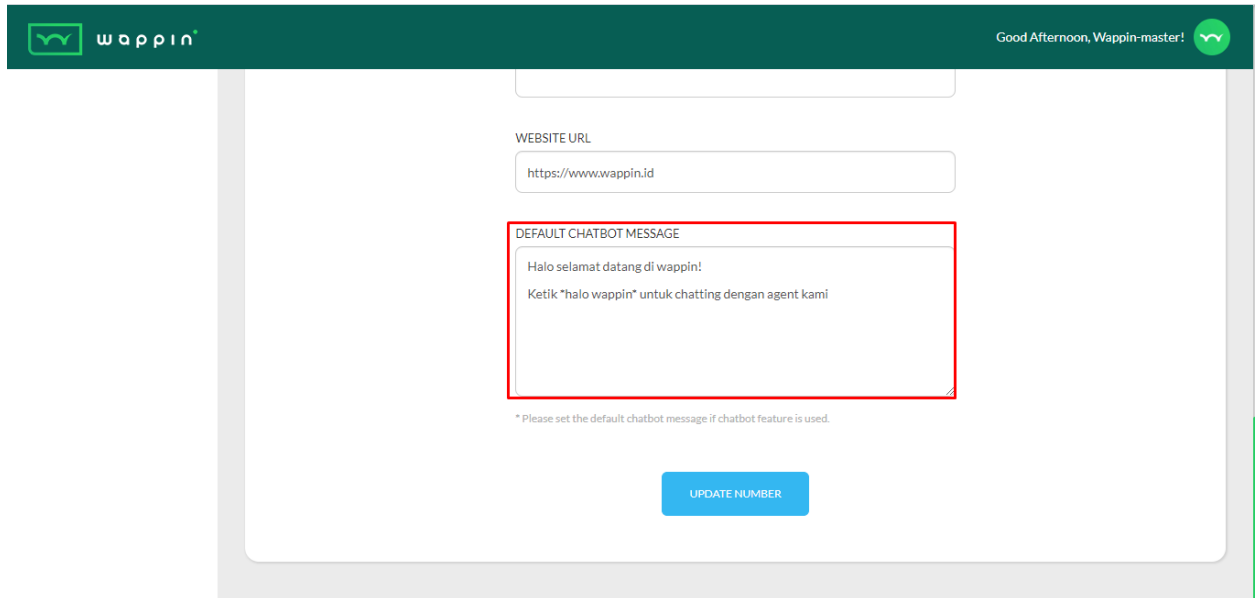
Explanation :

- **Select Project** – Project that will be used
- **Template Name** – Name of template
- **Scenario Type** – Type of the Scenario
  - **Normal** : if the client only have the chatbot feature
  - **Chat Panel** : Template for chat panel feature
- **Response Types** – Response for the Message
  - There are only 4 media formats WhatsApp can receive : Audio, Image, Document, and Text
- **Keyword** – Keyword to trigger the Chatbot message



### 3. Default Chatbot Message

If the customer types a keyword that is not registered, then the customer will receive the default response message. Your default response message can be edited in the **Account Menu**, choose submenu **Official Account Number**, then click the Edit button.



The screenshot shows the Wappin account settings interface. At the top, there is a dark green header with the Wappin logo on the left and the text "Good Afternoon, Wappin-master!" with a green circular icon on the right. Below the header, there is a form with several input fields. The first field is empty. The second field is labeled "WEBSITE URL" and contains the text "https://www.wappin.id". The third field is labeled "DEFAULT CHATBOT MESSAGE" and is highlighted with a red border. It contains the text "Halo selamat datang di wappin!" and "Ketik \*halo wappin\* untuk chatting dengan agent kami". Below this field, there is a small note: "\* Please set the default chatbot message if chatbot feature is used." At the bottom of the form, there is a blue button labeled "UPDATE NUMBER".

Flow Explanation :

- **Business Number**, Client's business number
- **Business Name**, Client's business name
- **Default Chat Bot Message**, the default response that will be sent when keywords are not listed

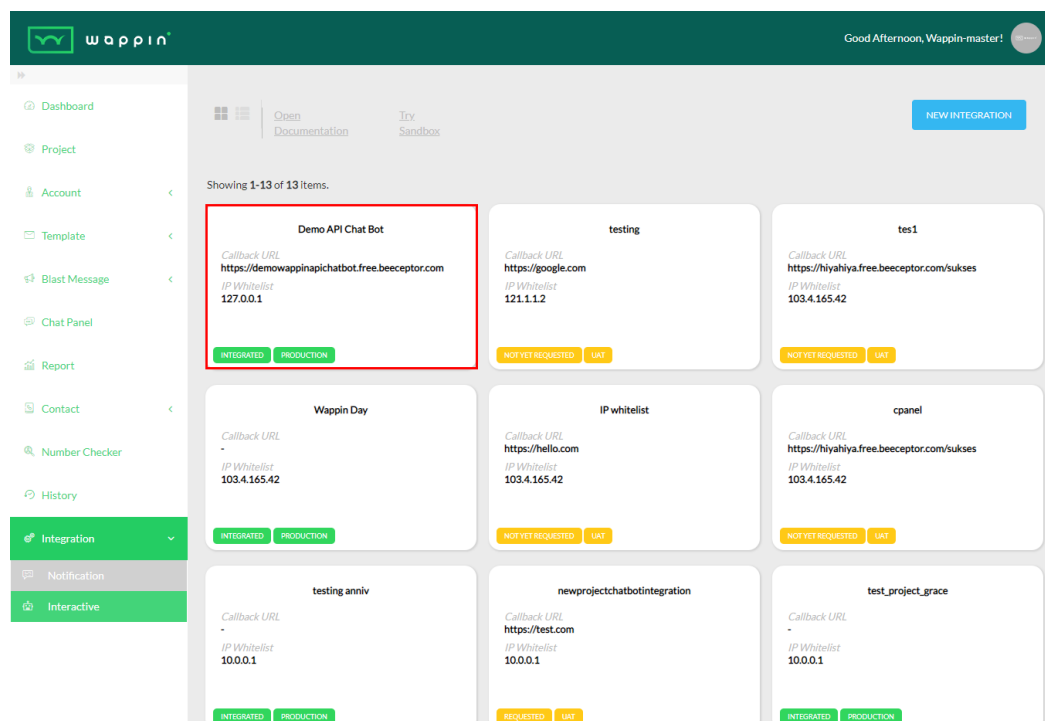
## 4.2. API GET Token

### 1. Get token description

- Is used for Authentication. If you don't use the authentication token when making a request, Wappin will give an error notification.
- Key/Token have 1 week active periode.

### 2. How to Get Token/Key

- Go to the Wappin dashboard portal on the Chatbot / List Integration menu.
- Select the Project to view its Key / Token.



- ### 3. If the token is expired, click the regenerate button to get a new token.

#### 4. API Get Token

Get Token				
Hostname	<a href="https://api.wappin.id">https://api.wappin.id</a>			
HTTP Method	POST			
Path	/v1/token/get			
HTTP Headers	Name	Format	Mandatory	
	authorization	Basic [base64_encode(username:password)]	Yes	
	content-type	application/json	Yes	
Request Definition	Name	Format	Mandatory	Description
	Username	string	yes	Provided client ID
	Password	string	yes	Provided secret key
Request Header	Content-Type:application/json Authorization: Basic OTo4OWNiNzgyYTBhMmZiNDEyMzgzMzA3MzhjYTgwMDJlZjZhN2YzYjJi			
Success Response Body	<pre>{   "status": "200",   "message": "Success",   "data": {     "access_token":     "d49865d016a32ab31734ed6972a818c41808c9f109c3614bec5da2508ef3be20",     "expired_datetime": "2019-07-20 16:21:51",     "token_type": "access_token"   } }</pre>			

#### Daftar Status Code

Status code	Message	Description
200	1. Success 2. Token not expired	Get token Success
401	Invalid credential	Invalid credential/token
406	Auth not recognize	Invalid authorization
407	IP Address not allowed	Invalid IP address

### 4.3. API Send Message (Text)

API Send Message (Text) is an API that is used to send message to WhatsApp User (with chatbot condition). Send Message API can't send parameters, it only can send text.

An example of a use case for using this API is, if you (the client) have a chatbot system, your system will consume this API to send messages to the WhatsApp user number.

Send Message (Text)				
Hostname	<a href="https://api.wappin.id">https://api.wappin.id</a>			
HTTP Method	POST			
Path	/v1/message/do-send			
HTTP Headers	Name	Format	Mandatory	Description
	authorization	Bearer [token]	Yes	
	content-Type	application/json	Yes	
Request Definition	Name	Format	Mandatory	Description
	client_id	string	yes	Client ID
	project_id	string	yes	Project ID
	message_content	string	yes	Message content sent to customer
	recipient_number	string	yes	Number registered on WhatsApp
Request Body	<pre>{   "client_id": "0003",   "project_id": "14",   "message_content": "haihello",   "recipient_number": "082315658823" }</pre>			
Success Response Body	<pre>{   "message_id": "5d31a231ab0ea",   "status": "200",   "message": "Success",   "data": null }</pre>			

### Status Code List API Send Message (Text)

Status code	Message	Description
200	Success	Message sent
400	Error no input received	Invalid format
	Project ID or Client ID cannot be process.	Invalid client or project
401	Invalid credential	Invalid credential/token
406	Auth not recognize	Invalid authorization
407	IP Address not allowed	Invalid IP address
601	Contact invalid	Recipient number is not a valid WhatsApp number

## 4.4. API Send Message (Media)

An API that used to send Message to the WhatsApp user as a Media such as :

- Document: PDF, DOC(X), PPT(X), XLS(X)s
- Image: JPG, JPEG, PNG
- Audio: AAC, M4A, MP3, OGG, OPUS

Send Message (Media)				
Hostname	<a href="https://api.wappin.id">https://api.wappin.id</a>			
HTTP Method	POST			
Path	/v1/message/do-send-media			
HTTP Headers	Name	Format	Mandatory	Description
	authorization	Bearer [token]	Yes	
	content-Type	application/json	Yes	
Request Definition	Name	Format	Mandatory	Description
	client_id	string	yes	Client ID
	project_id	string	yes	Project ID
	recipient_number	string	yes	Number registered on WhatsApp
	media_type	String	Yes	Valid input: 1. image 2. document 3. audio
	media	File	Yes	Upload media file
	caption	String	No	For media type image and document only
Request Body	<pre>{   "client_id": "0015",   "project_id": "38",   "recipient_number": "082315658823",   "media_type": "image",   "media": "7c87d81ca...jpg",   "caption": "test", }</pre>			
Success Response Body	<pre>{   "message_id": "5d3558b4ab2a9",   "status": "200",   "message": "Success",   "data": null }</pre>			

## API Send Media Status Code List

Status code	Message	Description
200	Success	Message sent
400	Error no input received	Invalid format
	Project ID or Client ID cannot be process.	Invalid client or project
401	Invalid credential	Invalid credential/token
406	Auth not recognize	Invalid authorization
407	IP Address not allowed	Invalid IP address
605	Upload Media Failed	Invalid media_type

## 5. API Inquiry

Message inquiry API is used to check the status of messages that have been sent.

Inquiry Message				
Hostname	<a href="https://api.wappin.id">https://api.wappin.id</a>			
HTTP Method	POST			
Path	v1/message/inquiry			
HTTP Headers	Name	Format	Mandatory	Description
	Authorization	Bearer [token]	Yes	
	Content-Type	application/json	Yes	
Request Definition	Name	Format	Mandatory	Description
	message_id	String	yes	Message ID
Request Body	<pre>{   "message_id": "5c33139c82f94" }</pre>			
Success Response Body	<pre>{   "message_id": "5cd3f65880d96",   "status": "200",   "message": "Success",   "data": [     {       "status": "sent",       "timestamp": "1557395033"     },     {       "status": "delivered",       "timestamp": "1557395037"     },     {       "status": "read",       "timestamp": "1557395044"     }   ] }</pre>			



### API Inquiry Status Code list

Status code	Message	Description
<b>200</b>	Success	Inquiry Success
<b>401</b>	Invalid credential	Invalid credential/token
<b>404</b>	Data not Found	Invalid Message ID
<b>406</b>	Auth not recognize	Invalid authorization
<b>407</b>	IP Address not allowed	Invalid IP address

## 6. API Callback

Callback is a Notification initiated by Wappin system to give the client information about the message delivery. 000 Status is a successful response. If the response is not 000, then the callback will be sent again and retried 3 times. From the callback function, Wappin needs to know the URL callback that client have.

There are 2 types of callback Message :

- Message : Notification result from the user chat to WhatsApp
- Status : Contain notification status from previously sent Message (sent, delivered, read)

Callback				
Hostname	End point URL			
HTTP Method	-			
Path	-			
HTTP Headers				
	Name	Format	Mandatory	Description
	Authorization	Bearer [token]	Yes	
	Content-Type	application/json	Yes	
Request Definition				
	Name	Format	Mandatory	Description
	message_id	String	Yes	Message ID
	client_id	String	Yes	Client ID
	client_name	String	Yes	Name of the client
	project_id	String		Project ID
	project_name	String		Name of the Project
	status_message	String		Message status
	message_content	String		Content of the message
	enviromtent	String		Project environment
	timestamp	Timestamp		Time message sent
	sender_number	String		Sender Number
	callback_type	String		Callback type
Request Body (Type Status)	{ "message_id": "5cdbe7774a059", "client_id": "0009", "client_name": "wappin-info", "project id": "16",			

	<pre> "project_name": "WAPPIN-INFO", "sender_number": "628989229471", "status_messages": "delivered", "message_content": "-", "environment": "PRODUCTION", "timestamp": "1557915512", "callback_type": "status" } </pre>
<b>Request Body (Type Message)</b>	<pre> {   "message_id": "5cdbe77d09ee7",   "client_id": "0009",   "client_name": "wappin-info",   "project_id": "16",   "project_name": "WAPPIN-INFO",   "sender_number": "628989229471",   "status_messages": "-",   "message_content": "tesss",   "environment": "PRODUCTION",   "timestamp": "1557915516",   "callback_type": "message" } </pre>
<b>Success Response Body</b>	<pre> {   "status": "000", } </pre>