



# CoreX WhatsApp Automations

## Free Edition — Complete Setup & User Documentation

Version 0.9.1 | October 2026 | [corexcodes.pro](https://corexcodes.pro)

This guide documents the actual Free v0.9.1 plugin package. It does not claim that Meta delivery, security or WordPress.org review have been independently certified.

## 1. Product overview

CoreX WhatsApp Automations connects WooCommerce to the Meta WhatsApp Cloud API using credentials supplied by the site owner. The Free edition supports order notifications, abandoned-cart reminders, approved template synchronization, country-aware phone formatting, message logs and setup guidance.

| Included in Free                  | Optional Pro add-on                                 |
|-----------------------------------|-----------------------------------------------------|
| Dashboard and API Settings        | Two-way WhatsApp Inbox                              |
| Setup Guides and Country          | WhatsApp OTP & Login                                |
| Templates and Order Notifications | Additional Pro capabilities (as separately offered) |
| Abandoned Cart and Message Logs   | License activation for Pro                          |

The Inbox and OTP & Login menu pages in Free show lightweight HTML/CSS feature previews and a “Want This Add-on” link to <https://corexcodes.pro/>. The premium PHP functionality is not part of this Free package.

## 2. Requirements and preparation

- WordPress 6.3 or later and PHP 7.4 or later, as declared by the package readme; use currently supported versions in production.
- WooCommerce installed and activated.
- HTTPS website with publicly accessible WordPress REST API for webhooks.
- Meta developer app and WhatsApp Business Account, plus approved message templates.
- Meta permissions and customer consent appropriate for the messages you plan to send.
- Working WordPress cron/traffic or a reliable server cron for scheduled cart reminders.

## 3. Install the plugin

1. Back up the site and test first on staging.
2. In WordPress, go to Plugins → Add New → Upload Plugin.
3. Choose `corex-whatsapp-automations-free-v0.9.1.zip`, install and activate.
4. Open CoreX WhatsApp Automations in the left admin menu.

5. Follow Setup Guides and configure API Settings before enabling notifications.

If replacing an earlier combined commercial plugin, deactivate the old plugin first. Do not install two plugins that register the same shared hooks without migration testing. The Free and Pro packages are intended to be separate plugins.

## 4. Admin navigation

| Menu                       | Purpose                                              |
|----------------------------|------------------------------------------------------|
| <b>Dashboard</b>           | Connection and setup overview                        |
| <b>API Settings</b>        | Meta credentials, webhook, connection tests          |
| <b>Setup Guides</b>        | In-dashboard setup instructions (third menu item)    |
| <b>Country</b>             | Default dialing country and custom input placeholder |
| <b>Templates</b>           | Sync and view approved Meta templates                |
| <b>Order Notifications</b> | Configure templates per WooCommerce order event      |
| <b>Inbox</b>               | Pro feature preview or Pro inbox when available      |
| <b>OTP &amp; Login</b>     | Pro feature preview or Pro authentication settings   |
| <b>Abandoned Cart</b>      | Reminders, guest phone capture, variable mapping     |
| <b>Message Logs</b>        | Recent outbound events, errors and statuses          |

## 5. Configure Meta WhatsApp Cloud API

6. Visit Meta for Developers and open or create the relevant app with WhatsApp access.
7. Identify the App ID and App Secret under app settings; locate the WhatsApp Business Account ID (WABA ID), Phone Number ID and an appropriate System User access token.
8. Open CoreX WhatsApp Automations → API Settings.
9. Enter Graph API Version, App ID, App Secret, WABA ID, Phone Number ID and access token.
10. Click Save Credentials. Saved secrets are masked in the form; leave a masked credential blank when updating other fields.
11. Use Connection Test, then Send Test WhatsApp with a real opted-in test number and an approved template.
12. Review Message Logs if Meta rejects the request.

Never publish access tokens, App Secrets or webhook verification secrets in screenshots, support tickets, documentation or plugin ZIPs. Meta access and messaging charges are separate from CoreX plugin licensing.

## 6. Webhook configuration

13. In API Settings, copy the read-only Webhook Callback URL. Its path is `/wp-json/wce-whatsapp/v1/webhook` on your own domain.
14. Save a unique Webhook Verify Token in the plugin.

15. In Meta App Webhooks, paste the exact callback URL and identical verify token.
16. Complete verification and subscribe the WhatsApp Business Account to relevant events as applicable.
17. Use the subscription and diagnostic controls in API Settings.
18. Check server firewall, SSL and REST API accessibility if verification fails.

The Free edition exposes webhook diagnostics but does not provide a two-way inbox. The Pro add-on may consume verified webhook events when installed. Do not run competing webhook handlers.

## 7. Country settings and phone numbers

19. Open CoreX WhatsApp Automations → Country.
20. Select the store default country from the country/territory list.
21. Optionally set Custom phone placeholder (e.g. 07123456789) and save.
22. Test both a local-format number and a number explicitly beginning with +country code.

The default country is an administrator setting, not a country selector shown to website visitors. Numbers with explicit international prefixes are preserved. The custom placeholder changes display guidance only; it does not validate that a mobile number exists or receives WhatsApp messages.

Example: with Pakistan selected, 03218452718 may be normalized to the international recipient form +923218452718. Other countries have different national dialing rules; use explicit +country codes for cross-border contacts and verify delivery in Meta.

## 8. Approved WhatsApp templates

23. Create templates in WhatsApp Manager using an appropriate category: Utility for transactional order events, Marketing for cart reminders and Authentication for OTP workflows where applicable.
24. Wait until Meta shows the template as APPROVED.
25. Open CoreX WhatsApp Automations → Templates and select Sync Templates.
26. Check each template name, language, category and status.
27. Choose synced templates in Order Notifications or Abandoned Cart.

Template body placeholders such as {{1}}, {{2}} and {{3}} must match the values supplied by the plugin. Do not assume an approved template with different parameter counts will work unchanged. Dynamic URL buttons require correctly configured URL parameters.

## 9. WooCommerce Order Notifications

28. Open Order Notifications and enable order notifications.
29. Select an approved template for each desired event: New Order, Processing, Completed, Cancelled, Failed and Refunded. Leave unused events disabled.
30. Save Notification Rules.
31. Confirm checkout captures a usable customer phone number and the customer has appropriate messaging consent.
32. From API Settings, test an existing WooCommerce order where supported.
33. Review Message Logs and the actual WhatsApp delivery result.

Use transactional Utility templates for relevant order updates. Message content and variables must align with Meta-approved template parameters. Avoid duplicate notifications triggered by overlapping order status transitions.

## 10. Abandoned Cart Automation

34. Open Abandoned Cart and enable the automation.
35. For Reminder 1, 2 and 3, choose a Meta-approved Marketing template and a send delay in minutes. Unused reminders may remain disabled.
36. Configure guest phone capture and the authentication template only if the corresponding verification workflow is supported and tested in your installation.
37. Customize each reminder's template-variable mapping, then Save Abandoned Cart Automation.
38. Test with a consenting test customer: start a cart, leave checkout, wait for the scheduled reminder, and inspect Message Logs.
39. Open the recovery URL to confirm it restores the expected cart and does not disclose another customer's data.

### 10.1 Customizable variable mapping

Each reminder has independent selectors for positions `{{1}}` through `{{10}}`. The default positions preserve the original mapping for existing stores.

| Position           | Default field               |
|--------------------|-----------------------------|
| <code>{{1}}</code> | Customer name               |
| <code>{{2}}</code> | Cart items                  |
| <code>{{3}}</code> | Cart total (store currency) |
| <code>{{4}}</code> | Store name                  |
| <code>{{5}}</code> | Recovery URL                |

Other selectable fields include Customer phone, Cart quantity, Cart subtotal (store currency), and Custom text. Choose Custom text and enter a value only when that Meta template position is intended to contain fixed text. Only positions present in the approved template body should be sent.

Known limitation for v0.9.1: the "Cart subtotal" selector currently uses the stored cart total rather than a separately calculated subtotal. Do not advertise it as a distinct subtotal until this is corrected. Test dynamic URL buttons and recovery tokens independently.

### 10.2 Guest phone capture

The Abandoned Cart settings include guest phone capture and an OTP Authentication template selector. This is distinct from the separately sold Pro OTP & Login account feature. Because live guest capture/verification behavior has not been verified in this documentation review, test its complete add-to-cart and OTP path on staging before relying on it in production.

## 11. Message Logs and diagnostics

Open Message Logs to inspect the most recent outbound records. Columns include time, event, order, template, status and message/result. A successful API request is not always the same as confirmed recipient delivery; check the Meta delivery statuses where available.

## 12. Optional Pro add-on

Inbox and OTP & Login pages display HTML/CSS previews in the Free edition. The “Want This Add-on” button opens <https://corexcodes.pro/>. Customers purchase and install the separately distributed Pro add-on to use premium features. The Free edition remains usable without it.

The CoreX Codes commercial policy discussed for Pro is lifetime license usage with 12 months of update eligibility; it is not a restriction on Free plugin updates from WordPress.org. Actual license activation and update delivery depend on the separate Pro/server implementation.

## 13. Troubleshooting

| Symptom                           | Check                                                                  |
|-----------------------------------|------------------------------------------------------------------------|
| Meta connection fails             | Token permissions, IDs, API version, network/SSL, credentials          |
| No templates in list              | Meta approval status, WABA ID, Sync Templates                          |
| Template error #131008            | Missing required body/button parameter; check template definition      |
| Template error #132018            | Parameter format/count and button type mismatch                        |
| Webhook verification fails        | Exact callback URL, verify token, public HTTPS REST route              |
| No abandoned-cart reminder        | Cron execution, captured phone, enabled rule, delay, template, consent |
| Wrong cart values                 | Variable positions, source cart snapshot, template placeholders        |
| Local phone formatted incorrectly | Country default, explicit +prefix, stored billing phone                |
| Inbox/OTP shows preview           | Install/activate compatible separately purchased Pro add-on            |
| Order messages duplicate          | WooCommerce event mapping and status transitions                       |

## 14. Privacy, consent and security

- Messages are sent via Meta WhatsApp Cloud API; customer phone numbers and relevant order/cart/template data are transmitted to Meta when enabled.
- Obtain appropriate WhatsApp messaging consent and provide any required privacy notice and opt-out process.
- Restrict WordPress admin access, protect tokens, use HTTPS, and verify webhook signatures where implemented.
- Avoid placing secrets or sensitive personal data into template variables or public recovery URLs.
- Review data retention for cart records and message logs against applicable privacy requirements.
- Check current Meta messaging policy, pricing and country-specific requirements before deployment.

## 15. Pre-launch checklist

- ☐ WooCommerce active and site backed up
- ☐ Meta credentials saved and connection test passes
- ☐ Default country and placeholder checked
- ☐ Required templates approved and synced
- ☐ Order notification test sent and delivered
- ☐ Cart reminder variable positions match approved templates
- ☐ Cart recovery and cron tested
- ☐ Webhook signature and REST access reviewed
- ☐ No sensitive credentials exposed
- ☐ Privacy notice and messaging consent in place
- ☐ WordPress.org Plugin Check and security review completed before submission

## 16. Support and references

CoreX Codes website: <https://corexcodes.pro/>

Meta Cloud API: <https://developers.facebook.com/docs/whatsapp/cloud-api/>

WooCommerce documentation: <https://woocommerce.com/documentation/>

WordPress.org Plugin Guidelines: <https://developer.wordpress.org/plugins/wordpress-org/detailed-plugin-guidelines/>

This documentation is based on inspection of CoreX WhatsApp Automations Free v0.9.1 source and UI. It is not a substitute for live functional tests or WordPress.org review approval.