



metamint

Helpdesk

Addon

User Documentation

Version 1.20.0

License: GPL-2.0

<https://metamintapps.com>

1. Overview.....	4
2. License & Activation.....	5
3. Addon Manager.....	7
4. Plugin Update System.....	8
5. Slack Integration.....	9
5.1 Setup & Configuration.....	11
5.2 Features.....	11
5.3 Category-Based Channel Routing.....	12
5.4 Unanswered Ticket Reminders.....	12
5.5 AJAX Endpoints.....	12
6. Discord Integration.....	13
6.1 Setup & Configuration.....	13
6.2 Features.....	15
6.3 Category-Based Channel Routing.....	15
6.4 Unanswered Ticket Reminders.....	15
6.5 AJAX Endpoints.....	15
7. WhatsApp Integration.....	16
7.1 Setup & Configuration.....	16
7.2 Features.....	17
7.3 Webhook Security — HMAC Signature Validation.....	17
7.4 Auto-Reply & Bot Settings.....	17
7.5 Incoming Message Handling.....	18
7.6 AJAX Endpoints.....	18
8. AI Bot Addon.....	19
8.1 Overview.....	19
8.2 API Key Setup.....	19
8.3 Bot Configuration.....	19
8.4 Knowledge Base / Data Sources.....	21
8.5 Custom Knowledge.....	22
8.6 AI Tools (Abilities).....	23
8.7 REST API Endpoints.....	24
8.8 Chat Streaming.....	24
8.9 Frontend Chat Widget.....	24
8.10 Ticket Conversion.....	25
9. Signature Addon.....	25
9.1 Setup.....	25
9.2 Per-User Signatures.....	26
9.3 Usage.....	26
9.4 AJAX Endpoints.....	26
10. Saved Replies (Canned Replies).....	27

10.1 Setup.....	27
10.2 Usage.....	27
10.3 AJAX Endpoints.....	27
11. WooCommerce Integration.....	28
11.1 Setup.....	28
11.2 Features.....	28
11.3 AJAX Endpoints.....	28
12. Filter Emails.....	29
12.1 Setup.....	29
12.2 Usage.....	29
12.3 Statistics.....	30
12.4 Wildcard Patterns.....	30
13. Multi-Inbox.....	30
13.1 Setup.....	30
13.2 Usage.....	31
13.3 Fetching Emails:.....	31
14. Zapier.....	32
14.1 Setup.....	32
14.2 Usage.....	34
15. Rating & Review.....	36
15.1 Setup.....	37
15.2 Choosing a Rating System.....	38
15.3 Settings.....	39
15.4 How Customers Leave a Review.....	41
15.5 How Agents Rate a Customer.....	42
15.6 Review Reports.....	43
15.7 AJAX & REST Endpoints.....	43

1. Overview

 Slack Integration Connect your Metamint Helpdesk with Slack to receive and manage support tickets directly from Slack channels. v1.0.0 Activate	 Discord Integration Integrate Discord webhooks to send ticket notifications and updates to your Discord server channels. v1.0.0 Activate	 Signature Add customizable signatures to your ticket replies for a professional touch. v1.0.0 Activate
 Canned Replies Create and manage pre-saved reply templates for faster responses to common questions. v1.0.0 Activate	 AI Bot Add a Metamint Helpdesk AI chatbot to your site. v1.0.0 Activate	 Filter Emails Configure email filtering rules to control which emails can create support tickets using blacklist/wildcard patterns. v1.0.0 Activate
 WooCommerce Show WooCommerce order insights for customers directly in conversation details. v1.0.0 Activate	 Rating & Review Let customers rate closed conversations with like/dislike, emoji, or star ratings plus a review comment, and track assignee ratings per customer review. v1.0.0 Activate	 Multi-Inbox Configure up to 3 IMAP inboxes with custom names to fetch emails and create tickets with inbox tracking. v1.0.0 Activate
 WhatsApp Integration Connect your Metamint Helpdesk with WhatsApp Business API to manage customer conversations, run automated bots, and configure settings directly from your dashboard. v1.0.0 Activate	 Zapier Integration Connect Metamint Helpdesk with 8000+ apps via Zapier. Provides webhook triggers and REST API actions for ticket management. v1.0.0 Activate	

The Metamint Helpdesk Addon plugin (v1.20.0) extends the core plugin with integrations and productivity tools. It provides:

- **Slack Integration** — ticket notifications and management via Slack
- **Discord Integration** — ticket notifications and management via Discord
- **WhatsApp Integration** — two-way WhatsApp messaging via Meta Cloud API
- **AI Bot** — AI-powered chatbot with a knowledge base and a tool called
- **Signature** — per-user agent signatures on replies
- **Saved Replies** — canned reply templates for quick responses
- **WooCommerce** — order context and customer stats within tickets
- **Filter Emails** — email filtering rules for email-to-ticket

- **Rating & Review** - Rate and review conversations
- **Multi Inbox**— email integration to support up to 3 independent IMAP inboxes
- **Zapier** – allows Metamint Helpdesk to connect with thousands of third-party applications, enabling automated workflows without writing code.

2. License & Activation

Metamint Helpdesk Addons Extend Metamint Helpdesk functionality with powerful addons

Metamint Helpdesk Addon License

Activate your paid Addons license here to unlock addon management below.

License Email

License Key

Current Status: Inactive

Activate License

Check Status

After installing and activating the addon plugin:

1. Go to Metamint Helpdesk → Addons in the admin.
2. Enter your License Email and License Key.
3. Click Activate License.

License verification flow

- License validated via remote API at <https://metamintapps.com/wp-json/mmhd-licensing/v1/> (filterable via `mmhd_addon_license_api_base`)
- Status stored in `mmhd_addon_license_state` option (inactive / active / expired / suspended / revoked)
- License metadata stored in `mmhd_addon_license_meta`: `licenseId`, `orderId`, `expiresAt`, `checkedAt`
- Status is checked daily via a cron job

Note: A valid license is required to use any addon features.

3. Addon Manager

The Addons page lists all available addon modules as cards with Activate/Deactivate toggles:



Inactive

Slack Integration

Connect your Metamint Helpdesk with Slack to receive and manage support tickets directly from Slack channels.

v1.0.0

Activate



Inactive

Discord Integration

Integrate Discord webhooks to send ticket notifications and updates to your Discord server channels.

v1.0.0

Activate



Inactive

Signature

Add customizable signatures to your ticket replies for a professional touch.

v1.0.0

Activate



Inactive

Canned Replies

Create and manage pre-saved reply templates for faster responses to common questions.

v1.0.0

Activate



Inactive

AI Bot

Add a Metamint Helpdesk AI chatbot to your site.

v1.0.0

Activate



Inactive

Filter Emails

Configure email filtering rules to control which emails can create support tickets using blacklist/wildcard patterns.

v1.0.0

Activate



Inactive

WooCommerce

Show WooCommerce order insights for customers directly in conversation details.

v1.0.0

Activate



Inactive

Rating & Review

Let customers rate closed conversations with like/dislike, emoji, or star ratings plus a review comment, and track assignee ratings per customer review.

v1.0.0

Activate



Inactive

Multi-Inbox

Configure up to 3 IMAP inboxes with custom names to fetch emails and create tickets with inbox tracking.

v1.0.0

Activate



Inactive

WhatsApp Integration

Connect your Metamint Helpdesk with WhatsApp Business API to manage customer conversations, run automated bots, and configure settings directly from your dashboard.

v1.0.0

Activate



Inactive

Zapier Integration

Connect Metamint Helpdesk with 8000+ apps via Zapier. Provides webhook triggers and REST API actions for ticket management.

v1.0.0

Activate

Addon	Default Status	Description
Slack Integration	Inactive	Slack workspace integration
Discord Integration	Inactive	Discord server integration
WhatsApp Integration	Inactive	WhatsApp Business API integration
AI Bot	Inactive	AI-powered chatbot

Addon	Default Status	Description
WooCommerce	Inactive	WooCommerce order lookup
Canned Replies	Inactive	Canned reply templates
Signature	Inactive	Agent signatures on replies
Ratings & Review	Inactive	Rate and review conversations
Multi Inbox	Inactive	Add multiple Inbox emails
Filter Emails	Inactive	Email filtering rules
Zapier Integration	Inactive	Automation

Addon toggles

- Cards gray out when the license is inactive
- Each toggle triggers an AJAX activate/deactivate call
- "Configure" button navigates to the addon's settings tab

4. Plugin Update System

The addon includes a custom plugin updater (plugin-updater.php):

- Intercepts `pre_set_site_transient_update_plugins` to inject update data
- Validates license via `upgrader_pre_download` before allowing download
- Checks remote version via `mmhd_addon_remote_version` transient

5. Slack Integration

Basic Settings

Mail Settings

Email Templates

Fetch Tickets

Advanced

Slack

Slack Integration

Connect your Slack workspace to sync conversations and collaborate with your team.

● Not Connected



Connect your Slack workspace to start syncing conversations

Slack API Credentials

 **IMPORTANT: Configure Redirect URI First!**

Before using "Connect to Slack" button, you MUST add this URL to your Slack app:

`http://metamint.local/wp-admin/admin.php?page=mmhd-dashboard&slack_callback=1`  Copy

How to add it:

1. Go to api.slack.com/apps → Select your app
2. Go to "OAuth & Permissions"
3. Scroll to "Redirect URLs" section
4. Click "Add New Redirect URL"
5. Paste the URL above and click "Add"
6. Click "Save URLs" at the bottom

How to get your Slack API credentials

Step-by-step guide:

1. Go to api.slack.com/apps
2. Create a new app or select an existing one
3. From "Basic Information" → copy Client ID and Client Secret
4. **Add Redirect URL:** See red box above and add the URL to "OAuth & Permissions" → "Redirect URLs"
5. **Add Bot Scopes:** From "OAuth & Permissions" → "Scopes" → add required Bot Token Scopes (see yellow box below)
6. Install/Reinstall your app to the workspace
7. From "OAuth & Permissions" → copy "Bot User OAuth Token" (starts with xoxb-)
8. (Optional) From "Basic Information" → copy Signing Secret

 **Tip:** You can use just the Bot Token for basic integration. The "Connect to Slack" button is optional for OAuth flow.

Client ID *

Client Secret *  Saved

 Leave empty to keep existing secret, or enter a new one to update

Bot User OAuth Token * (Required for channels)

 **Required Bot Token Scopes**

Signing Secret (Optional)

Used to verify requests from Slack

Webhook URL (Optional)

For incoming webhooks (simple notifications)

 Save API Credentials

 Disconnect

 Test Connection

5.1 Setup & Configuration

1. Ensure the Slack addon is Active in Metamint Helpdesk → Addons.
2. Go to Metamint Helpdesk → Settings → Slack Integration.
3. Enter your Slack Client ID, Client Secret, and OAuth Scopes.
4. Click Save Credentials, then Test Connection.
5. (Optional) Click Get OAuth URL for OAuth-based authentication.
6. Click Get Workspace Data to fetch workspace channels and users.
7. Configure notification settings.

Setting	Description
Default Channel	Target channel for notifications
Notify on New Ticket	Send a message when a ticket is created
Notify on Reply	Send a message when a reply is added
Notify Reply Mode	Immediate or delayed (configurable delay in minutes)
Notify on Close	Send a message when a ticket is closed
Enable Thread Replies	Replies appear as thread replies
Auto-Sync	Enable auto-sync of Slack data
Auto-Send New Tickets	Automatically post new tickets to Slack
Auto-Sync Replies	Automatically sync replies to Slack
Unanswered Ticket Reminders	Enable cron-based reminders
Reminder Interval	Hours/minutes between checks
Unanswered Threshold	Hours of inactivity before reminder triggers
Reminder Message Template	Custom reminder message

5.2 Features

- Ticket auto-notify — new tickets posted to a selected Slack channel
- Reply notifications — replies forwarded to Slack threads
- Ticket closure notifications — when a ticket is closed
- Send ticket to Slack — manually send a ticket from the admin
- Send reply to Slack thread — forward agent replies
- Test message — verify the integration
- Per-category Slack channels — route tickets to different channels by department

5.3 Category-Based Channel Routing

- Edit a Department (category) and select a Slack channel from the dropdown. The channel ID is stored as term meta (mmhd_slack_channel). Tickets in that department will automatically post to the assigned channel.

5.4 Unanswered Ticket Reminders

- A cron job (mmhd_slack_check_unanswered_tickets) runs at the configured interval and posts a message listing unanswered tickets to the configured channel. Custom cron schedule is dynamically registered as mmhd_reminder_interval.

5.5 AJAX Endpoints

Action	Purpose
mmhd_slack_save_credentials	Save Slack OAuth credentials
mmhd_slack_get_credentials	Get stored Slack credentials
mmhd_slack_test_connection	Test Slack connection
mmhd_slack_get_status	Get Slack integration status
mmhd_slack_disconnect	Disconnect the Slack workspace
mmhd_slack_save_settings	Save Slack notification settings
mmhd_slack_get_oauth_url	Get the OAuth URL for authentication
mmhd_slack_get_workspace_data	Fetch Slack workspace data
mmhd_slack_send_ticket	Send ticket to Slack
mmhd_slack_send_reply	Send a reply to the Slack thread
mmhd_slack_test_message	Send a test Slack message
mmhd_slack_test_reminder	Test reminder configuration
mmhd_slack_get_taxonomy_channels	Get Slack channels for categories

6. Discord Integration

6.1 Setup & Configuration

Basic Settings

Mail Settings

Email Templates

Fetch Tickets

Advanced

Discord

AI Bot

Discord Integration

Connect your Discord server to receive real-time ticket notifications.

Connection Status

Connected

Disconnect

Server: robodesk-server

Channel: #test-2

Bot Configuration

To connect Robodesk to Discord:

1. Create a Bot in the Discord Developer Portal.

2. Copy the Bot Token and paste it below.

3. Invite the Bot to your server.

Application ID (Client ID)

Invite Bot

Found in Discord Developer Portal > General Information. Required to generate an invite link.

Bot Token

Saved

Save Bot Token

Channel Selection

Notification Channel

Current Channel (1521483030022062081)

Select the channel where you want to receive notifications. Clicking "Connect Channel" will finalize the setup.

Connect Channel

Send Test Message

Legacy Webhook Configuration (Optional)

Notification Settings

Notify when new conversation is created

Send a notification to Discord when a new support conversation arrives

Notify on new reply

Send a notification when someone replies to a conversation

Immediately

After delay

Notify when conversation is closed/completed

Get notified when a conversation is marked as closed or completed

Reminder Interval

Hours ▾

How long to wait after the last customer reply before sending a reminder

💡 How Reminders Work:

- Checks run automatically at your specified interval
- Only sends reminders for conversations where the last reply was from a customer
- Excludes closed/completed/resolved conversations
- Won't spam - waits for the full interval before sending another reminder

⚠ Automatic Reminders Status:

Status: ⚠ Not Scheduled

Enable reminders and save settings to activate automatic checks.

Save Settings

🔍 Message Preview

💎 New conversation #123 - Sample Conversation Title

This is how your conversation notifications will appear in Discord.

👤 Customer

John Doe

📄 Status

Open

👤 Assignee

John Doe

📧 Via Robodesk • Today at 7:45 AM

✉ Email

john@example.com

⚡ Priority

● Medium

1. Ensure the Discord addon is Active in Metamint Helpdesk → Addons.
2. Go to Metamint Helpdesk → Settings → Discord Integration.
3. Enter your Discord Webhook URL (from Discord channel → Integrations → Webhooks).
4. Click Save Webhook, then Test Connection.
5. (Optional) Enter Bot Token and Application ID for enhanced features.
6. Click Fetch Channels to load available channels.
7. Configure notification settings.

Settings

Setting	Description
Default Channel	Target Discord channel
Notify on New Ticket	Send a message on ticket creation
Notify on Reply	Send a message in reply
Notify on Status Change	Send a message on status change
Mention Role ID	Role to @mention on new tickets
Use Thread	Create Discord threads for ticket conversations
Auto-Send New Tickets	Automatically post new tickets
Auto-Send Replies	Automatically sync replies

Setting	Description
Auto-Send Status Changes	Automatically sync status changes
Unanswered Ticket Reminders	Enable cron-based reminders
Reminder Interval	Hours/minutes between checks
Unanswered Threshold	Hours of inactivity

6.2 Features

Same as Slack: ticket auto-notify, reply notifications, closure notifications, send ticket/reply to Discord, test message. Adds role mention support and Discord thread creation.

6.3 Category-Based Channel Routing

- Same as Slack: assign Discord channels per department/category via term meta (mmhd_discord_channel_id).

6.4 Unanswered Ticket Reminders

- Cron job mmhd_discord_check_unanswered_tickets with configurable interval and threshold. Custom cron schedule mmhd_discord_reminder_interval.

6.5 AJAX Endpoints

Action	Purpose
mmhd_discord_save_webhook	Save Discord webhook URL
mmhd_discord_get_webhook	Get stored Discord webhook
mmhd_discord_test_connection	Test Discord connection
mmhd_discord_disconnect	Disconnect the Discord bot
mmhd_discord_save_bot_token	Save the Discord bot token
mmhd_discord_fetch_channels	Fetch Discord channel list from API
mmhd_discord_save_channel	Save Discord channel selection
mmhd_discord_save_settings	Save Discord notification settings
mmhd_discord_send_test_message	Send a test Discord message

7. WhatsApp Integration

7.1 Setup & Configuration

WhatsApp Business Integration

Connect your WhatsApp Business account to receive and reply to conversations directly from Robodesk tickets.

API Credentials

Configure your Meta (WhatsApp Business) credentials.

 How to get your WhatsApp Business credentials

Phone Number ID *

WhatsApp Business Account ID

Access Token * 

.....



 Leave blank to keep the existing token.

App Secret (Optional – for webhook signature verification) 

.....



 Save Credentials

 Test Connection

 Disconnect

Webhook Configuration

Add these values in your Meta app under WhatsApp → Configuration → Webhook.

CALLBACK URL



VERIFY TOKEN



 Subscribe to the "messages" webhook field in your Meta app dashboard.

Notification Settings

☒ Notify when a new WhatsApp conversation is created

☒ Notify when a customer replies

☒ Notify when a conversation is closed

☒ Automatically create a ticket/conversation for every incoming WhatsApp message

 Save Settings

Send Test Message

Send a test message to verify your connection is working.

Phone (with country code, no +), e.g. 9779812345678

 Send Test

1. Ensure the WhatsApp addon is Active in Metamint Helpdesk → Addons.
2. Go to Metamint Helpdesk → Settings → WhatsApp Integration.

3. Configure Meta Cloud API credentials:

Field	Description
Access Token	From Meta Developer Console
Phone Number ID	WhatsApp Business phone number ID
Business Account ID	WhatsApp Business Account ID
App Secret	From Meta Developer Console
Verify Token	User-defined token for webhook verification

4. Click Save Credentials, then Test Connection.

5. Configure the Webhook URL in Meta Developer Console to:

<https://yoursite.com/wp-json/mmhd-whatsapp/v1/webhook>

7.2 Features

- **Receive WhatsApp messages** — incoming messages create or update tickets
- **Send replies via WhatsApp** — agent replies are forwarded to customers
- **Ticket closure notification** — customers notified when the ticket is closed
- **Test message** — send a test WhatsApp message
- **Conversation mapping** — each ticket stores the customer's WhatsApp number in the `_mmhd_whatsapp_phone` post meta, which is how incoming messages are matched to existing tickets

7.3 Webhook Security — HMAC Signature Validation

The webhook endpoint at `/mmhd-whatsapp/v1/webhook` validates incoming requests:

- GET — handles Meta's webhook verification (`hub_mode`, `hub_verify_token`, `hub_challenge`)
- POST — validates the X-Hub-Signature-256 header using HMAC-SHA256 with the configured App Secret

Invalid signatures are rejected to prevent unauthorized data injection.

7.4 Auto-Reply & Bot Settings

[WhatsApp Bot Settings Screenshot]



Setting	Description
Bot Enabled	Enable/disable auto-reply
Greeting Message	First response sent to new conversations

Setting	Description
Away Message	Message sent outside business hours
Away Hours Enabled	Toggle away hours
Away Start/End	Business hours definition (HH: MM)
Keyword Triggers	Keyword → response pairs for automated routing
Fallback Message	Default response when no keyword matches

7.5 Incoming Message Handling

1. Webhook validates signature using App Secret.
2. New sender → ticket created + message added as comment.
3. Existing open ticket → message appended as reply.
4. Auto-reply sent back (if bot is enabled).
5. Optional staff notification.

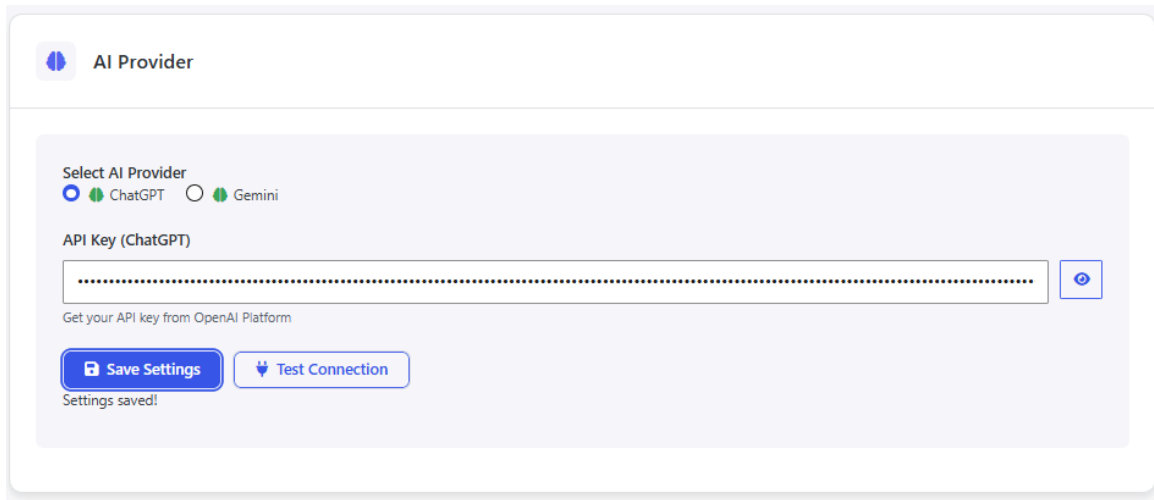
7.6 AJAX Endpoints

Action	Purpose
mmhd_whatsapp_save_credentials	Save WhatsApp API credentials
mmhd_whatsapp_get_credentials	Get WhatsApp credentials (masked)
mmhd_whatsapp_test_connection	Test Meta Cloud API connection
mmhd_whatsapp_disconnect	Disconnect WhatsApp
mmhd_whatsapp_get_status	Get WhatsApp integration status
mmhd_whatsapp_save_settings	Save WhatsApp notification settings
mmhd_whatsapp_save_bot_settings	Save auto-reply bot settings
mmhdd_whatsapp_send_test_message	Send a test WhatsApp message

8. AI Bot Addon

8.1 Overview

The AI Bot provides a frontend chat widget powered by ChatGPT or Gemini with capabilities to answer questions, perform actions, and escalate to human agents. It includes a knowledge base indexing system for site content.



The screenshot shows the 'AI Provider' configuration window. At the top, there's a title bar with a purple icon and the text 'AI Provider'. Below this, a section titled 'Select AI Provider' contains two radio buttons: 'ChatGPT' (selected) and 'Gemini'. Underneath, the 'API Key (ChatGPT)' is shown in a text field with a masked key '.....' and a toggle icon. A note below the field says 'Get your API key from OpenAI Platform'. At the bottom, there are two buttons: 'Save Settings' (with a checkmark icon) and 'Test Connection' (with a plug icon). A status message 'Settings saved!' is displayed below the 'Save Settings' button.

8.2 API Key Setup

1. Ensure the AI Bot is Active in Metamint Helpdesk → Addons.
2. Go to Metamint Helpdesk → Settings → AI Bot.
3. In the Tools tab, configure Provider (ChatGPT or Gemini), API Key, and Model.

Options stored: mmhd_bot_chatgpt_api_key, mmhd_bot_gemini_api_key,
mmhd_bot_active_model, mmhd_bot_chatgpt_filestore, mmhd_bot_gemini_filestore.

8.3 Bot Configuration

- Primary Bot selection
- Enable Talk to Human toggle
- Profile image
- Widget position (left/right)
- Accent color
- Theme (light/dark)
- Show on mobile toggle
- Display condition (all pages, home only, specific pages, exclude pages)
- Welcome/initial message

Stored in mmhd_bot_widget_config option.

In the Settings tab:

Basic SettingsMail SettingsEmail TemplatesFetch TicketsAdvancedAI BotWhatsApp

>
⚙️
📁
📁
📁
📁

AI Bot Settings

Configure your AI provider and chatbot settings

🤖 Chatbot Status

☒ Enable Chatbot

Display the AI chatbot widget on your website

☒ Set bot as primary

Set the AI chatbot as the primary support channel

☒ Talk to human

Set the AI chatbot to talk to human

🟢 Chatbot is active

👤 Bot Identity

Bot Name

BRO

The name displayed in the chat widget header.

Bot Avatar



[Change Image](#)

×

Shown as the bot avatar in the chat widget.

🧠 AI Provider

Select AI Provider

☒ ChatGPT ☐ Gemini

API Key (ChatGPT)

🔒

[Change Key](#)

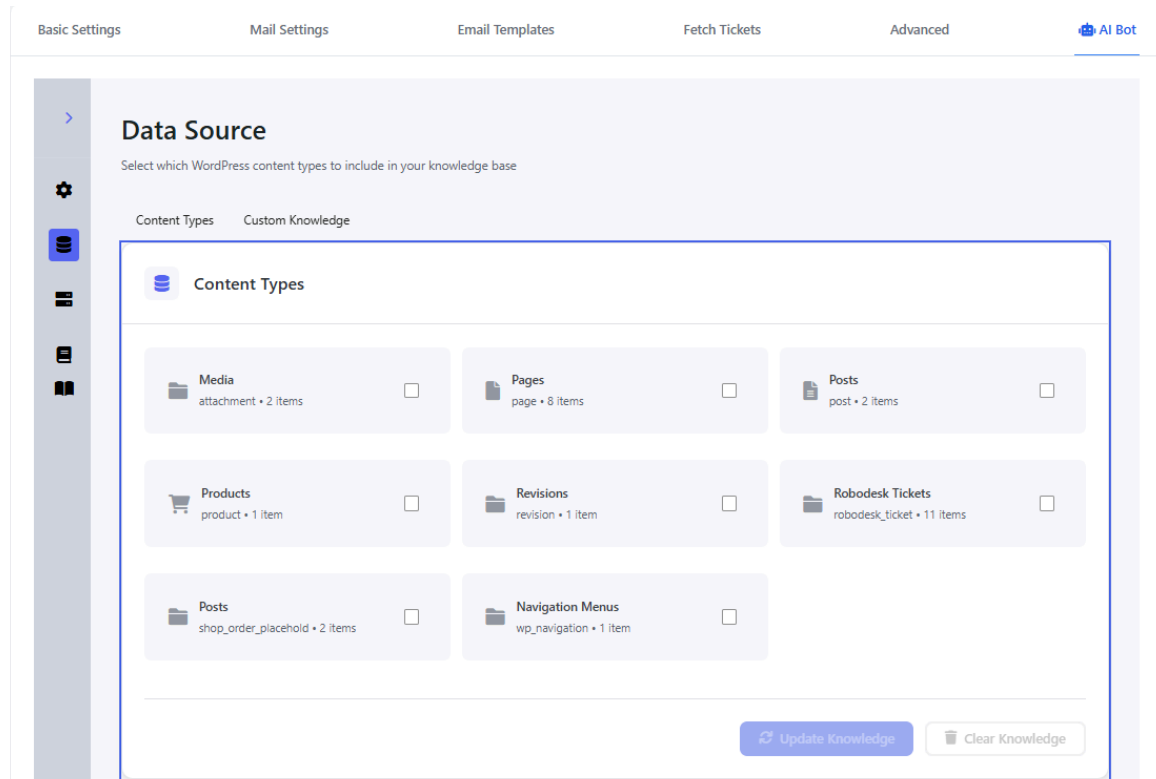
Get your API key from OpenAI Platform

💾 Save Settings

🧪 Test Connection

🔔 Click [Change Key](#) and enter a new key to test the connection.

8.4 Knowledge Base / Data Sources



The knowledge base uses a custom MySQL table `wp_mmhd_knowledge_chunks` with FULLTEXT index on `chunk_text`.

Table structure

Column	Type	Description
id	BIGINT UNSIGNED AUTO_INCREMENT PK	
source_type	VARCHAR(50)	Source type (post_type, custom, file)
source_key	VARCHAR(255)	Source identifier
url	VARCHAR(500)	Source URL
chunk_text	MEDIUMTEXT	Indexed content (FULLTEXT)
updated_at	DATETIME	Last updated

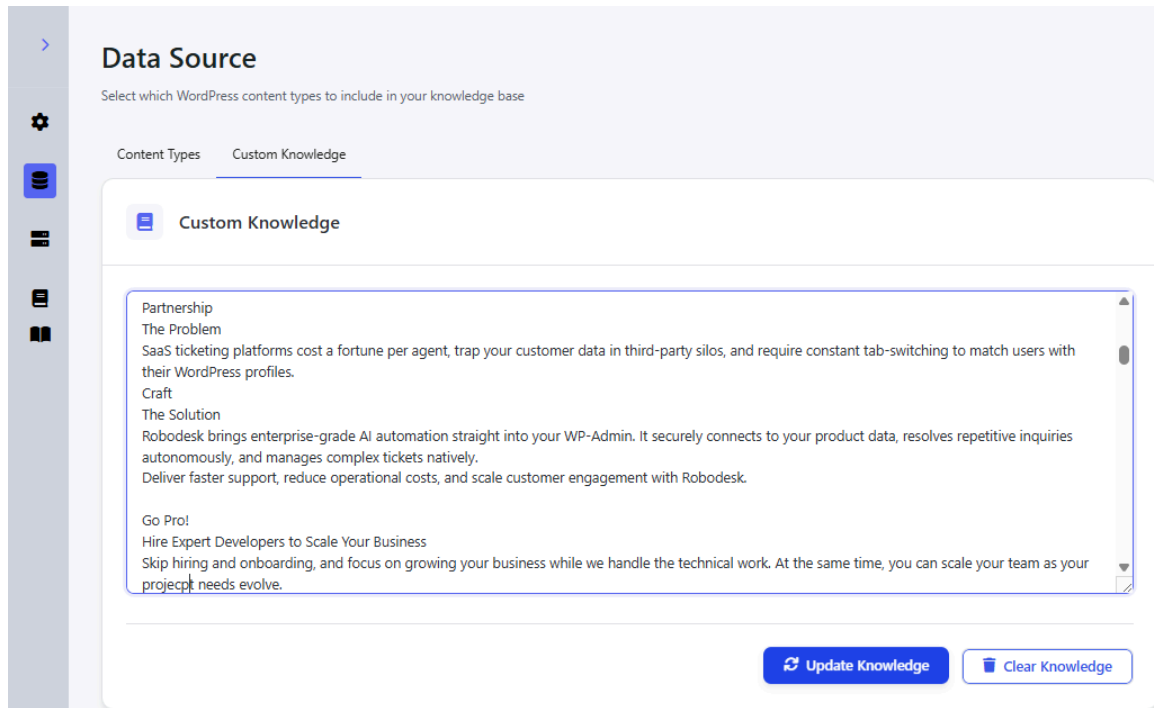
Content types that can be indexed: Posts, Pages, FAQs, any registered custom post type.

Indexing process

1. Choose post types in the Content Types tab

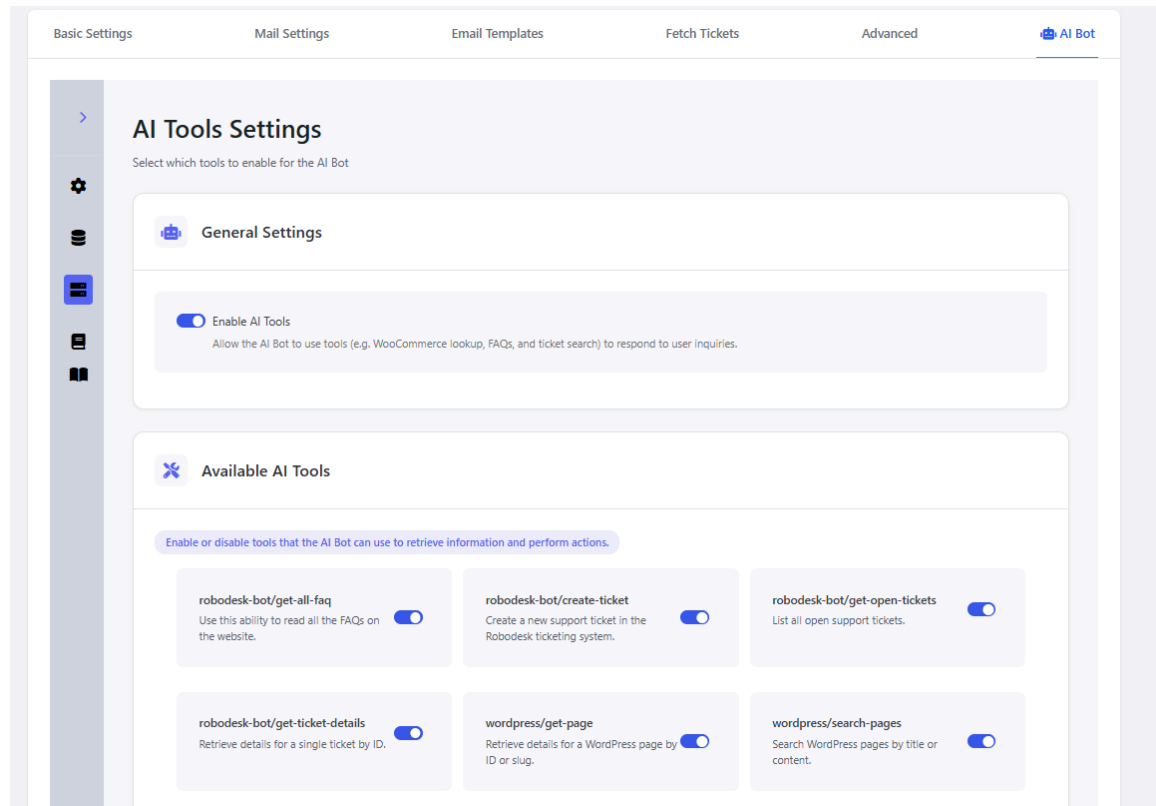
2. Click Start Crawl — batch processing with pagination
3. Each post is split into chunks and stored in the knowledge table
4. FULLTEXT MATCH queries for relevance-ranked search
5. Crawl progress bar shows status (idle / processing / complete / error)

8.5 Custom Knowledge



Upload or paste custom content the bot can reference (product docs, policies, etc.). Stored in mmhd_bot_custom_knowledge option.

8.6 AI Tools (Abilities)



In the Tools tab, enable/disable specific abilities.

Active Ticket Abilities

Name	Description
mmhd-bot/create-ticket	Create a support ticket
mmhd-bot/get-open-tickets	Get open tickets for the current user
mmhd-bot/get-ticket-details	Get ticket details by ID

Active FAQ Abilities

Name	Description
mmhd-bot/get-all-faq	Get all FAQs

Knowledge Base Search (Internal)

- `site__search__knowledge` — searches `mmhd_knowledge_chunks` table using `FULLTEXT MATCH` (used internally in chat stream, not a configurable ability)

8.7 REST API Endpoints

Namespace: `mmhd-bot/v1`

Method	Route	Permission	Purpose
GET	<code>/post-types</code>	Public	List available post types for indexing
GET	<code>/api-key</code>	Admin	Get API key status
POST	<code>/api-key</code>	Admin	Save API key
GET	<code>/widget-config</code>	Public	Get widget configuration
POST	<code>/chat-stream</code>	Public	Chat streaming endpoint (SSE)
POST	<code>/fetch-posts-init</code>	Admin	Initialize post crawler
POST	<code>/fetch-posts-batch</code>	Admin	Process batch of posts
POST	<code>/fetch-posts-finalize</code>	Admin	Finalize crawl
POST	<code>/custom-knowledge-update</code>	Public	Update custom knowledge
GET	<code>/custom-knowledge-fetch</code>	Public	Fetch custom knowledge
POST	<code>/delete-knowledge</code>	Public	Delete knowledge entry
GET	<code>/get-filestore</code>	Admin	Get file store status
POST	<code>/abilities-status</code>	Admin	Update the ability toggle status
POST	<code>/save-mcp-config</code>	Admin	Save the MCP configuration
GET	<code>/get-mcp-config</code>	Admin	Get the MCP configuration

8.8 Chat Streaming

Chat streaming uses Server-Sent Events (SSE) via `/mmhd-bot/v1/chat-stream` for real-time AI responses with local function/tool calling capabilities.

8.9 Frontend Chat Widget

See the core plugin documentation for the frontend chat widget base. When the AI Bot addon is active, conversations are powered by an LLM (ChatGPT or Gemini). Chat sessions are persisted in

sessionStorage and can be converted to formal support tickets when escalated. Messages include ISO timestamps that are preserved during ticket conversion.

8.10 Ticket Conversion

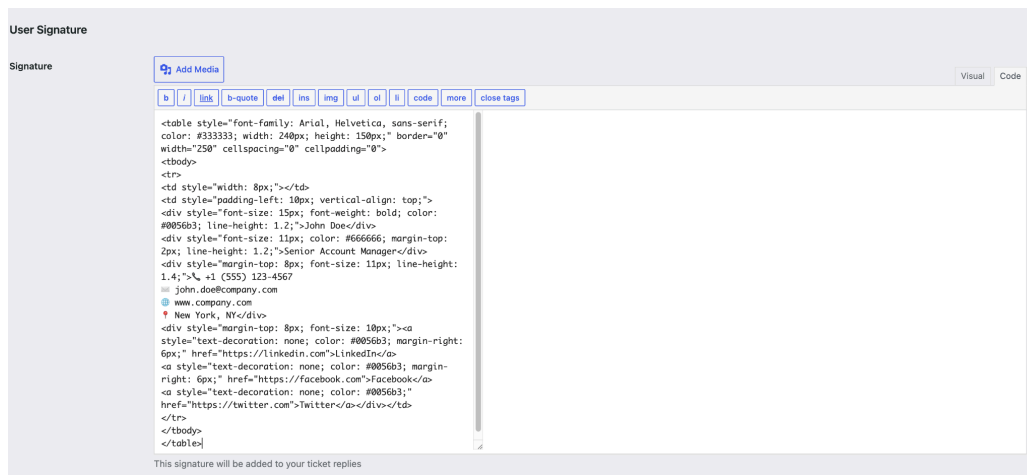
[Ticket Conversion Screenshot]

When a user clicks "Talk to Human":

- The AI conversation is converted into a support ticket
- All messages (user + assistant) are migrated as ticket comments with their original timestamps
- A transition message is appended: "Transitioned conversations from AI to human agent."
- The ticket is created with "open" status and assigned to support staff

9. Signature Addon

9.1 Setup



User Signature

Signature

Add Media

B

I

U

A

John Doe

Senior Account Manager

+1 (555) 123-4567

john.doe@company.com

www.company.com

New York, NY

LinkedIn

Facebook

Twitter

This signature will be added to your ticket replies

1. Activate Signature in Metamint Helpdesk → Addons.
 2. Go to Metamint Helpdesk → Settings → Signature.
 3. Configure: enable toggle, default signature content (rich text), position (above/below reply).
- Options stored: mmhd_signature_enabled, mmhd_signature_content, mmhd_signature_position.

9.2 Per-User Signatures

Each agent can set a personal signature from their WordPress Profile page (/wp-admin/profile.php). The signature field is a rich text editor. Stored as user meta (mmhd_user_signature).

9.3 Usage

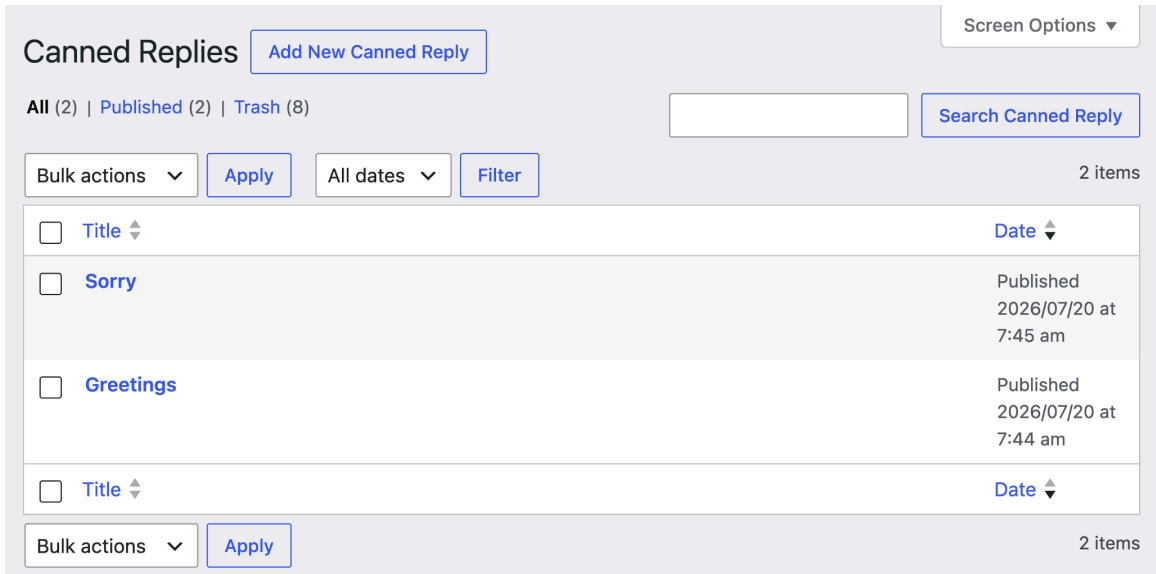
A checkbox "Append Signature" appears below the reply editor in the ticket detail view. When checked, the agent's personal signature (or the default if none is set) is appended to the reply.

9.4 AJAX Endpoints

Action	Purpose
mmhd_signature_save_settings	Save signature settings
mmhd_signature_get_settings	Get signature settings
mmhd_get_user_signature	Get per-user signature

10. Saved Replies (Canned Replies)

10.1 Setup



- 1. Activate Canned Replies in Metamint Helpdesk → Addons.
- 2. A Canned Replies menu item appears under the Metamint Helpdesk admin menu.
- 3. Create, update, or delete reply templates.

10.2 Usage

An "Insert Saved Reply" searchable dropdown appears above the reply editor in the ticket detail view. Select a reply to insert its content into the editor.

10.3 AJAX Endpoints

Action

Action	Purpose
mmhd_saved_replies_get_all	Get all canned replies
mmhd_saved_replies_create	Create a new canned reply
mmhd_saved_replies_update	Update a canned reply
mmhd_saved_replies_delete	Delete a canned reply

11. WooCommerce Integration

11.1 Setup

1. Activate WooCommerce in Metamint Helpdesk → Addons.
2. WooCommerce must be installed and activated on the site.

11.2 Features

When viewing a ticket from a WooCommerce customer, a sidebar panel displays:

- Order history — a list of recent orders from that customer
- Order stats — total count, average value, total spent (returned inline with order data)

Data is cached via transients with a short TTL.

11.3 AJAX Endpoints

Action	Purpose
mmhd_woocommerce_get_customer_orders	Get customers' WooCommerce orders (includes stats)

12. Filter Emails

12.1 Setup

▼ Filter Emails

Configure email filtering rules to control which emails can create support tickets.

☐ Filter Inactive

⚙ Filter Configuration

☒ **Enable Email Filtering**
When enabled, incoming emails will be checked against your filter rules

Excluded Email Addresses

noreply@example.com

Enter email addresses to exclude from filtering (comma-separated or one per line)

💾 Save Filter Settings

1. Activate Filter Emails in Metamint Helpdesk → Addons.
2. Go to Metamint Helpdesk → Settings → Filter Emails.
3. Toggle the filter on/off.
4. Add email addresses or domains to filter (one per line).

Settings stored in `mmhd_filter_emails_settings`:

Key	Description
enabled	Toggle on/off
excluded_emails	Newline-separated email addresses/patterns to block

Emails are filtered using blacklist mode — matching addresses are blocked from creating tickets via the email-to-ticket pipeline.

12.2 Usage

Prevents specified email addresses or domains from creating tickets via the email-to-ticket pipeline.

Hooks into the `mmhd_before_create_ticket_from_email` filter.

12.3 Statistics

Statistics stored in `mmhd_filter_emails_statistics`: `total_filtered`, `allowed`, `blocked`, and `last_updated`.

A separate log is maintained in `mmhd_filter_emails_log`.

12.4 Wildcard Patterns

Pattern	Matches
*@spam.com	Any email from spam.com
noreply@*	Any noreply email
@	All emails
user@example.com	Specific address

13. Multi-Inbox

13.1 Setup

Multi-Inbox IMAP Settings

Configure up to 3 IMAP inboxes. Each inbox will fetch emails and create tickets with the assigned inbox name.

1 Inbox #1

Inbox Name

example inbox 1

This name will appear in the dashboard & conversations to identify the source inbox.

IMAP Host

email.example.com

Port

993

Username

example user 1

Password

.....

Encryption

SSL

Test Connection

Save Inbox

2 Inbox #2

3 Inbox #3

1. Activate Multi Inbox in Metamint Helpdesk → Addons.
2. Go to Metamint Helpdesk → Settings → Multi Inbox
3. Configure Inbox 1,2,3 IMAP settings
4. Test and save connections.

13.2 Usage

The Multi-Inbox Addon extends Metamint Helpdesk's single-IMAP email integration to support up to 3 independent IMAP inboxes, each with a custom label (e.g., "Support", "Sales", "Billing"). It registers through the addon-manager. (defaults inactive) and lives under addons/multi-inbox/. Core files:

- multi-inbox-addon.php — Entry point: enqueues assets, renders settings, overrides hooks
- multi-inbox-handlers.php — All business logic: IMAP fetching, email→ticket creation, reply threading, deduplication, cron override
- multi-inbox-settings-ui.php — Accordion settings form (host/port/username/password per inbox)
- multi-inbox-admin.js — Injects an "Inbox" column into dashboard & conversations tables; fetches inbox names via AJAX

On activation, it migrates any legacy single IMAP settings into inbox slot #1. When active, it overrides the default email fetch hooks so all fetching goes through the multi-inbox engine. Each ticket stores its origin inbox in inbox_name post meta, displayed in ticket lists and detailed views.

13.3 Fetching Emails:

[Basic Settings](#) [Multi-Inbox](#) [Email Templates](#) [Fetch Tickets](#) [Advanced](#) [AI Bot](#) [WhatsApp](#) [Zapier](#)

Fetch Conversations

Use this feature to fetch new emails from your configured IMAP account and convert them into conversations and replies.

Email Fetching Settings

Fetch Interval (minutes)

Auto-assign to

Least loaded agent (auto)

▼

Leave this on "Least loaded agent (auto)" to assign fetched email tickets to the least-loaded agent.

⚙ Save Auto-Fetch

🔄 Fetch Emails Now

Fetch Conversations

The Fetch Conversations feature automatically retrieves new emails from your configured IMAP account and converts them into support conversations and replies.

Configure Auto-Fetch

1. Go to Settings → Fetch Tickets.
2. Enter the desired Fetch Interval (minutes).
3. Choose how fetched emails should be assigned using the Auto-assign to dropdown.
4. Click Save Auto-Fetch.

Auto-Fetch

After saving, Metamint Helpdesk will automatically check your configured email inbox at the specified interval. Any new emails found will be fetched and converted into support conversations without requiring manual intervention.

Fetch Emails Manually

To retrieve new emails immediately instead of waiting for the next scheduled fetch:

1. Go to Settings → Fetch Tickets.
2. Click Fetch Emails Now.

Metamint Helpdesk will instantly check the configured mailbox and import any new emails into the system.

14. Zapier

14.1 Setup

Basic Settings

Multi-Inbox

Email Templates

Fetch Tickets

Advanced

AI Bot

Zapier

Zapier Integration

Connect Robodesk with 8000+ external apps via the official Robodesk Zapier app.

Setup Instructions

Install the Robodesk app on Zapier: <https://zapier.com/apps/robodesk> (Search "Robodesk" in the Zapier app directory.)

Connect your site: Use your Site URL and the API Key below.

Build Zaps using the triggers and actions listed below — no paid Zapier plan required.

API Key

Used to authenticate the Robodesk Zapier app. Enter this in Zapier when connecting your account.

API Key

rbz_07fe168063560bb36fb5efdc10d0567f87169:

Regenerate

Site URL

https://robo-desk.codepixelz.tech

Enter this URL in Zapier when connecting your Robodesk account.

The Zapier Integration allows Metamint Helpdesk to connect with thousands of third-party applications, enabling automated workflows without writing code.

Connect Metamint Helpdesk to Zapier

1. Go to Settings → Zapier.
2. Copy the Site URL.
3. Copy the API Key.
4. In Zapier, add the Metamint Helpdesk app and enter the copied Site URL and API Key when prompted.
5. Complete the connection and start creating automated workflows (Zaps).

API Key

The API Key is used to authenticate the Metamint Helpdesk account with Zapier. Copy this key and paste it into Zapier during the connection process.

If the API key is compromised or needs to be replaced, click Regenerate to create a new API key.

Note: Regenerating the API key invalidates the previous key. Any existing Zapier connections using the old key must be updated with the new one.

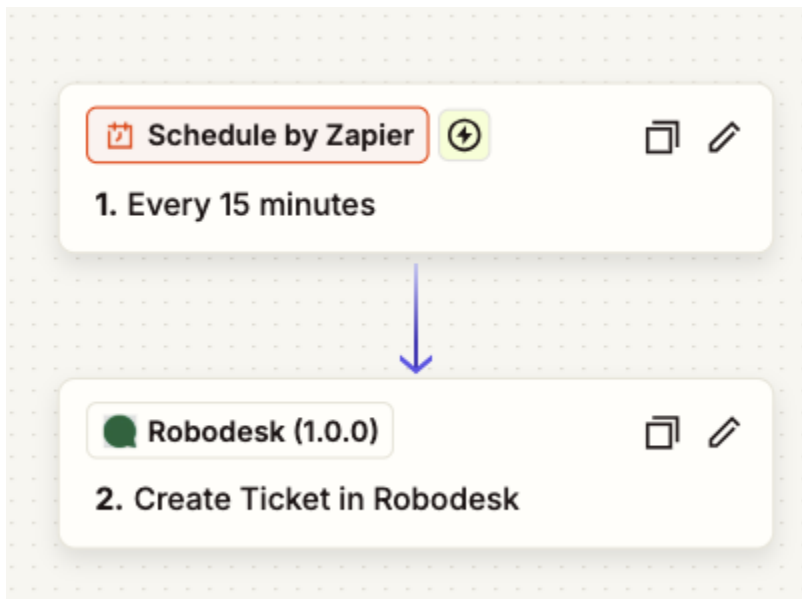
Site URL

The Site URL identifies your Metamint Helpdesk instance. Enter this URL in Zapier when connecting your Metamint Helpdesk account.

Both the Site URL and API Key are required to establish a successful connection between Metamint Helpdesk and Zapier.

14.2 Usage

Action



This example demonstrates how to automatically create a new Metamint Helpdesk ticket at regular intervals using Zapier.

Workflow

1. Schedule by Zapier triggers the workflow every 15 minutes.
2. Zapier sends the configured data to Metamint Helpdesk.
3. Metamint Helpdesk automatically creates a new support ticket using the received information.

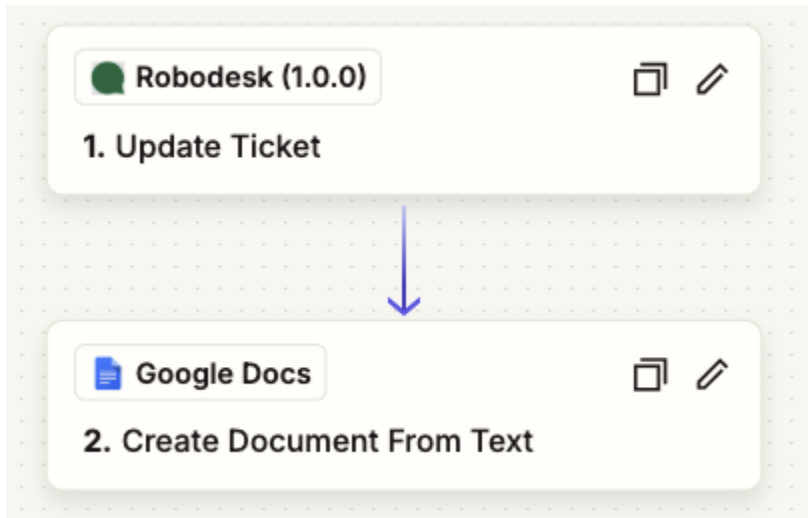
Example Use Case

A company receives support requests from another internal system every few minutes. Instead of manually creating tickets, Zapier automatically runs every 15 minutes and creates new tickets in Metamint Helpdesk.

Workflow:

- Trigger: Every 15 minutes (Schedule by Zapier)
- Action: Create Ticket in Metamint Helpdesk

Trigger



This example demonstrates how to automatically create a Google Docs document whenever a ticket is updated in Metamint Helpdesk.

Workflow

1. A ticket is updated in Metamint Helpdesk.
2. Zapier detects the ticket update.
3. Zapier creates a new document in Google Docs using the ticket information.

Example Use Case

A support team wants to maintain documentation for ticket updates. Whenever an agent updates a ticket, Zapier automatically creates a Google Docs document containing the latest ticket details for reporting or record-keeping.

Workflow:

- Trigger: Ticket Updated (Metamint Helpdesk)
- Action: Create Document From Text (Google Docs)

Available Triggers and Actions:

Triggers (Robodesk → Zapier)

The Robodesk Zapier app polls the API for new data. Available triggers:

Trigger	Description
New Ticket	Fires when a new ticket is created in Robodesk.
New Reply	Fires when a new reply is added to a ticket.
Status Changed	Fires when a ticket status is updated.
New User	Fires when a new WordPress user is created.
Deleted Ticket	Fires when a ticket is permanently deleted.
Deleted User	Fires when a user is deleted.

Actions (Zapier → Robodesk)

Available actions in the Robodesk Zapier app:

Action	Description	Required Fields
Create Ticket	Creates a new support ticket.	title , customer_email
Add Reply	Adds a reply to an existing ticket.	ticket_id , message
Update Ticket	Changes ticket status (open, in-progress, closed) and optionally adds a note.	ticket_id , status
Create User	Creates a new WordPress user with a specified role.	email
Delete User	Permanently deletes a WordPress user.	user_id
Delete Ticket	Permanently deletes a support ticket.	ticket_id

15. Rating & Review

The Rating & Review add-on turns closed conversations into customer feedback. Customers are invited to rate the support they received and can leave an optional comment. The rating is posted into the ticket thread, credited to the agent who handled it, and rolled up per agent. Staff can rate the customer in return from any ticket. Reviews work for every closed ticket, no matter which channel it came from — chat widget, email, WhatsApp, or any other integration.

Features:

- Customer reviews — a rating prompt for closed conversations with an optional written comment
- One review per conversation — a customer can review a given ticket only once
- Thread record — every review is posted into the conversation so agents see it in context
- Assignee roll-up — an average score and review count per agent on the Users list and user profile
- Agent-side customer ratings — staff rate the customer from a single ticket, and the rating follows the customer across all of their conversations
- Review link by email — a "review this conversation" button added to the closing ticket email
- Three rating systems — Like & Dislike, Emoji, or Star

15.1 Setup

★ Rating & Review

Let customers rate their closed conversations and review the service they received. Customers can review every closed ticket, regardless of where it came from (chat, email, WhatsApp, etc.).

Enable Rating & Review 

Show the review prompt for closed conversations in the chat widget and add a review link to the closing ticket email.

Review System

Star 

★ 1 → 1/5, ★ 2 → 2/5, ★ 3 → 3/5, ★ 4 → 4/5, ★ 5 → 5/5

Prompt Title

How was your experience?

Review Comment Placeholder

Tell us more about your experience (optional)

Submit Button Label

Submit Review

Success Message

Thank you for your feedback!

Already Rated Message

You have already rated this conversation.

Review link in the closing email



Add a "review this conversation" button to the closing ticket email sent to the customer.

Button Label

Review this conversation



1. Activate Rating & Review in Metamint Helpdesk → Addons.
2. Go to Metamint Helpdesk → Settings → Rating & Review.
3. Leave Enable Rating & Review switched on (it is on by default).
4. Choose a Review System
5. Click Save Settings.

15.2 Choosing a Rating System

The review system controls both what customers see and how their choice is scored. Every option is normalised to a score out of 5, so ratings stay comparable no matter which system is active.

Option	Scores
Like & Dislike	 Dislike-2/5  Like-5/5
Emoji	 Very bad-1/5 ,  Bad-2/5 ,  Neutral-3/5,  Good-4/5,  Excellent-5/5
Star	 Very bad-1/5,  Bad-2/5 ,  Neutral- 3/5,  Good- 4/5 ,  Excellent- 5/5

Note: Changing the review system never deletes existing reviews. Each stored review keeps its own score and label, so historical reports do not silently change meaning. Reviews given under a different system are still shown correctly — they simply display the label from the system they were given under.

15.3 Settings

Administrators can configure the Rating & Review feature using the available settings.

★ Rating & Review

Let customers rate their closed conversations and review the service they received. Customers can review every closed ticket, regardless of where it came from (chat, email, WhatsApp, etc.).

Enable Rating & Review
Show the review prompt for closed conversations in the chat widget and add a review link to the closing ticket email.

Like & Dislike
Emoji
✓ Star

★ 1 → 1/5, ★ 2 → 2/5, ★ 3 → 3/5, ★ 4 → 4/5, ★ 5 → 5/5

Prompt Title
How was your experience?

Submit Button Label
Submit Review

Already Rated Message
You have already rated this conversation.

Review Comment Placeholder
Tell us more about your experience (optional)

Success Message
Thank you for your feedback!

Review link in the closing email
Add a "review this conversation" button to the closing ticket email sent to the customer.

Button Label
Review this conversation

Save Settings

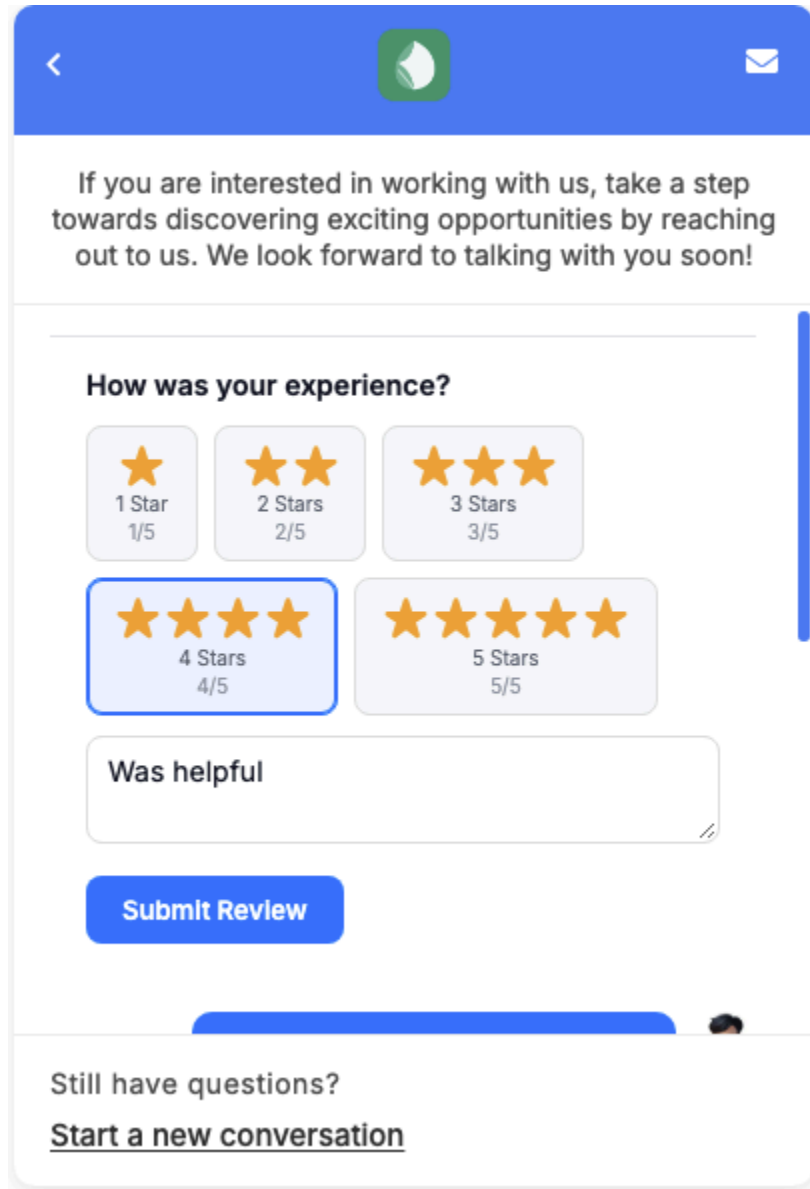
Setting	Description
Enable Rating & Review	Turns the whole add-on on or off. When off, no review prompt or email button is shown.
Review System	Selects Like & Dislike, Emoji, or Star.
Prompt Title	Heading shown above the rating control.

Review Comment Placeholder	Placeholder text inside the optional comment box.
Submit Button Label	Text displayed on the submit button.
Success Message	Message shown after a review is successfully submitted.
Already Rated Message	Message shown when a customer returns to a conversation they have already rated.
Review Link in the Closing Email	Adds a “Review this conversation” button to the closing ticket email.
Button Label	Text displayed on the review button in the email.

All settings are stored in the `mmhd_rating_review_settings` option. Deactivating the addon preserves both the settings and every review.

15.4 How Customers Leave a Review

Customers can submit a rating and review through the Rating & Review widget or the review button provided in an email.



The screenshot shows a mobile app interface for submitting a review. At the top is a blue header bar with a back arrow, a circular profile icon, and an envelope icon. Below the header is a white box containing the text: "If you are interested in working with us, take a step towards discovering exciting opportunities by reaching out to us. We look forward to talking with you soon!". This is followed by a section titled "How was your experience?". It features five star rating options: "1 Star 1/5", "2 Stars 2/5", "3 Stars 3/5", "4 Stars 4/5" (which is highlighted with a blue border), and "5 Stars 5/5". Below the ratings is a text input field with the placeholder "Was helpful". At the bottom of this section is a blue button labeled "Submit Review". Below the button is a blue bar with a small profile icon. At the very bottom is a white box with the text "Still have questions?" and a link "Start a new conversation".

Submit flow:

1. Open the closed conversation.
2. Select a rating.
3. Enter a review.
4. Submit the rating and review.

Eligibility: Only closed conversations can submit a review for a conversation.

15.5 How Agents Rate a Customer

Agents can rate customers from the conversation interface.

CONVERSATION DETAILS
**Jordan Alex** 
shakyanikhil2003@gmail.com 

Origin:

CHAT

IP Address:

103.10.28.173

Browser:

Chrome 152.0.0.0

Rate this customer
Your rating of the customer is shared across all their conversations and can be edited anytime.


1 Star


2 Stars


3 Stars


4 Stars


5 Stars

Communicated Nicely

Save Rating

1. Open a ticket.
2. In the right-hand sidebar, use the "Rate this customer" card.
3. Pick a rating and optionally add a note.
4. Click Save Rating.

The rating belongs to the customer, not to the ticket: it is shown on every conversation from that customer and can be edited at any time. Each rating is also copied into the ticket's internal Notes tab, so other agents can see the history.

15.6 Review Reports

Administrators can view customer rating and review information through the reporting interface.

The **Users** column provides access to individual customer information. A user's profile can include a **Rated Conversations** table showing conversations that have received ratings.

Rated Conversations				
Conversation	Customer	Rating	Comment	Reviewed
#1203 — test2 closed	Jordan Alex <shakyanikhil2003@gmail.com>	★★★★ 4 Stars	Was helpful	September 28, 2026 7:53 am

Open an agent's profile (Users → All Users → click a user) to see a "Metamint Helpdesk" section with:

- The agent's average customer rating and total review count
- A "Rated Conversations" table listing every conversation assigned to that agent that a customer has rated, with the conversation, customer, rating, comment and date — each conversation links straight back to its ticket

The profile section only appears for users who can access the helpdesk dashboard. Because the scores live in the standard WordPress users table, the averages are also available to any plugin that reads WordPress user meta, and the Users list can be exported to CSV as usual.

15.7 AJAX & REST Endpoints

AJAX Endpoint

Action	Purpose
mmhd_rating_review_save_settings	Saves the addon settings from the Settings tab.

REST Endpoints

REST Namespace: `mmhd-rating/v1`

Method	Route	Permission	Purpose
GET	<code>/review-status</code>	Public	Checks whether a review is possible for a ticket and returns the active rating options.
POST	<code>/review</code>	Logged-in customer or valid helpdesk token	Submits a customer review for a closed ticket.
GET	<code>/customer-rating</code>	Helpdesk staff	Retrieves the current rating for a ticket's customer.
POST	<code>/customer-rating</code>	Helpdesk staff	Saves or updates the customer's rating.