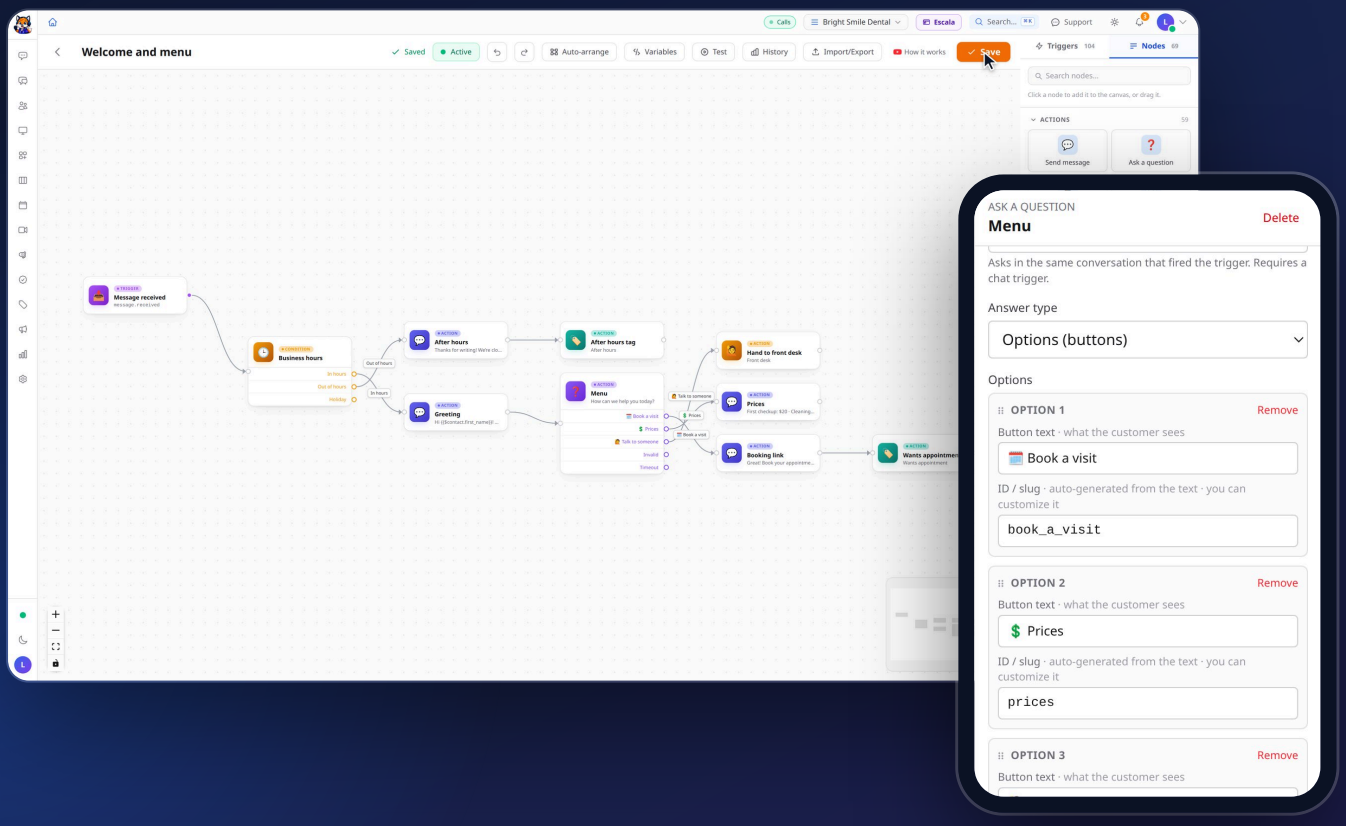
Omnifox

WORKFLOWS · COMPLETE GUIDE

Workflows

Automations that reply, tag, assign, remind people of appointments and alert your team on their own, 24 hours a day. The list, the editor, every trigger and every node, variables, testing, history, templates and how they connect with the inbox, the CRM, the calendar, AI, Projects and Team.



CONTENTS

Table of contents

How to read this manual

The screenshots are real Omnifox screens with an example business: **Bright Smile**, a fictional dental clinic that serves patients on WhatsApp and through the chat on its website. It has a human team, **Front desk**, and tags such as **Wants appointment**, **After hours**, **New patient**, **Orthodontics** and **VIP**. Across chapters 9 to 24 we build its **Welcome and menu** workflow, step by step. The patients and conversations are made up.

Button and field names are in **bold**, exactly as they appear on screen. This manual follows, chapter by chapter, the "Workflows · Complete guide" video on the Omnifox YouTube channel, and expands it with every option. Some features depend on your plan and your permissions (chapter 52).

01 What a workflow is 3
02 The workflow list 5
03 Search and filter 6
04 Turn on, duplicate, export and delete 8
05 Three ways to create a workflow 9
06 Editor tour 10
07 Moving around the canvas 12
08 The palette: triggers and nodes 13
09 Choosing the trigger 17
10 Trigger filters 23
11 Stop when an agent replies, and trigger variables 24
12 Adding nodes to the canvas 25
13 Connecting nodes and branches 26
14 Business hours 27
15 Send message 28
16 Variables 30
17 Asking a question 32
18 One branch per option 35
19 Adding and removing tags 37
20 Transferring to a person or a team 38
21 The after-hours branch 39
22 Move, rename, duplicate, copy and delete 40
23 Undo, redo and auto-arrange 40
24 Validate, save and turn on 41
25 Testing the workflow 42
26 The live run 44
27 The run history 45
28 In real life: a patient writes from the website 47
29 The workflow in the inbox 49
30 If condition: a single rule 51
31 Multiple branches 52
32 Contact has tag 53
33 Date and time 54
34 Detect language and agent in team 55
35 Wait time and wait for an event 57
36 WhatsApp templates and the 24-hour window 58
37 Emails: to the customer and to the team 59
38 Contacts and variables 61
39 Conversation actions 63
40 Respond with AI and AI classifier 64
41 AI agents and workflows, together 66
42 HTTP request 67
43 Flow control 68
44 Shortcuts for your team in the inbox 70
45 Scheduled and incoming webhook 72
46 CRM: triggers and nodes 73
47 Appointments: triggers and nodes 74
48 Projects and Team 76
49 Orders, payments, calls and more 78
50 The template gallery 79
51 Export and import 81
52 Plans, limits and permissions 82
53 Best practices 84
54 Troubleshooting 85

01

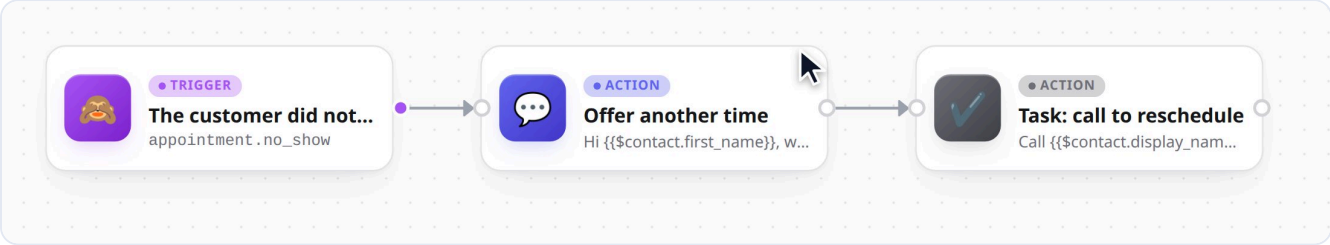
CHAPTER 1

What a workflow is

A trigger and a chain of steps that run on their own, every time.

A **workflow** is an automation that always has the same shape:

- **A trigger**: the event that sets it in motion. For example, “Message received”, “Appointment created” or “The customer did not show up (no-show)”. Each workflow has only one.
- **A chain of connected nodes**: **actions** that do something (send a message, add a tag, assign the conversation), **conditions** that choose a path (are we open? which button did they pick?) and **waits** that pause the workflow (for a day, or until the customer replies).

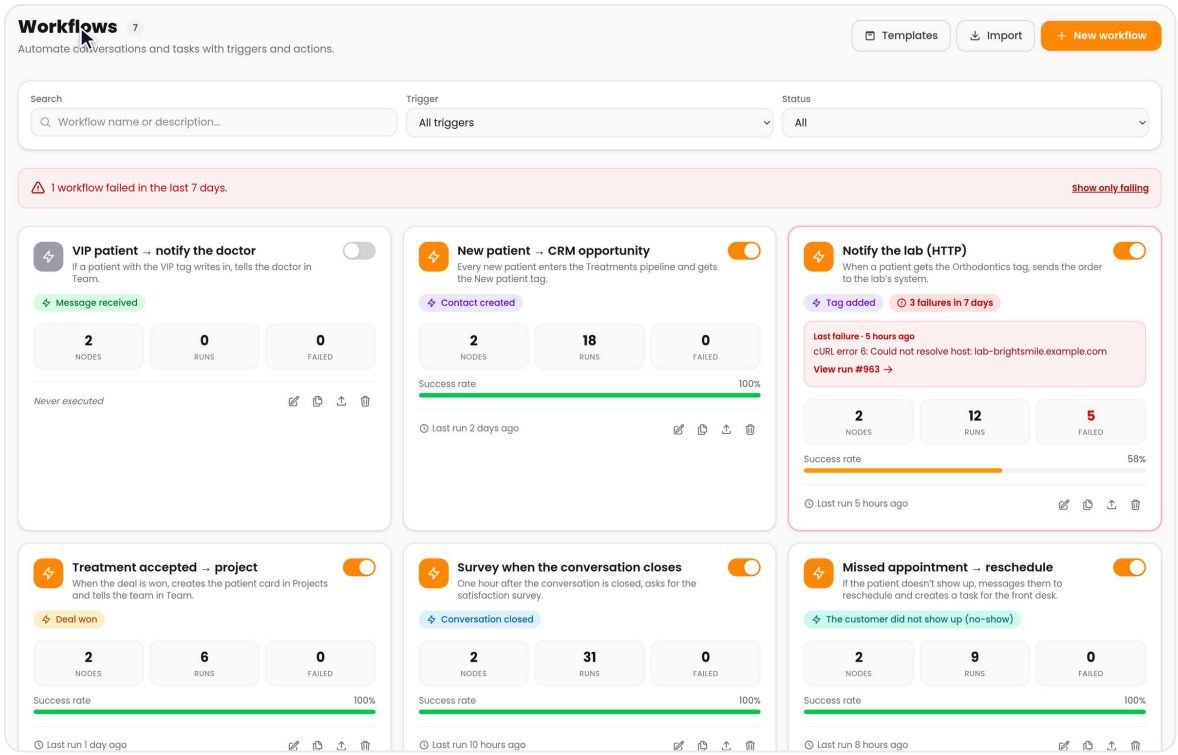


A workflow from start to finish: the “The customer did not show up (no-show)” trigger and two actions, offering another time and creating a task for the front desk.

In this example, when a patient misses their appointment, the workflow writes to them to reschedule and creates a follow-up task for the front desk. Nobody has to remember: it happens on its own, every time the event occurs.

Where they are

Workflows live in **Settings** → **Workflows**. Each workspace has its own: a workflow only sees the conversations, contacts, channels and teams of its workspace.



Settings → Workflows: Bright Smile's workflows, each with its card.

What you can automate

If you want to…	Use
Welcome first-time writers and show them a menu	**Message received** + **Ask question** (chapters 9 to 24)
Reply differently at night or on holidays	**Business hours** (chapter 14)
Remind people of appointments and ask them to confirm	**Appointment reminder** + **Send WhatsApp template** (chapters 36 and 47)
Hand the conversation to the front desk	**Transfer to an agent** (chapter 20)
Create a CRM deal for every new patient	**Create deal (CRM)** (chapter 46)
Alert your team through Team or by email	**Publish to Team**, **Notify the team by email** (chapters 37 and 48)
Connect with another system	**HTTP request** and **Incoming webhook** (chapters 42 and 45)

NOTE

Workflows are not the same as AI agents. A workflow follows exactly the steps you drew; an AI agent converses in its own words. They work very well together: a workflow can hand the conversation to an AI agent, and an agent can launch a workflow (chapter 41).

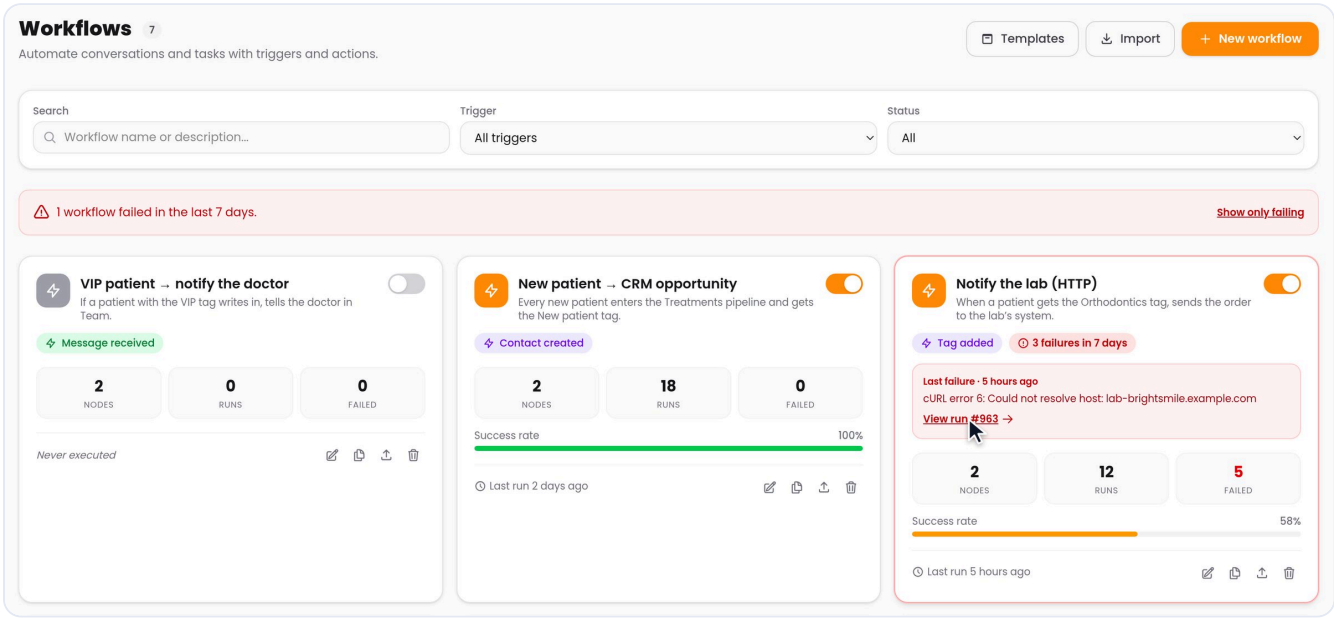
02

CHAPTER 2

The workflow list

Each card tells you what the workflow does and how it's doing.

At the top you see the **Workflows** title with the workspace's total number of workflows, the text "Automate conversations and tasks with triggers and actions." and three buttons: **Templates**, **Import** and **New workflow**. Below, the filters (chapter 3) and the cards, in a grid of up to three columns.

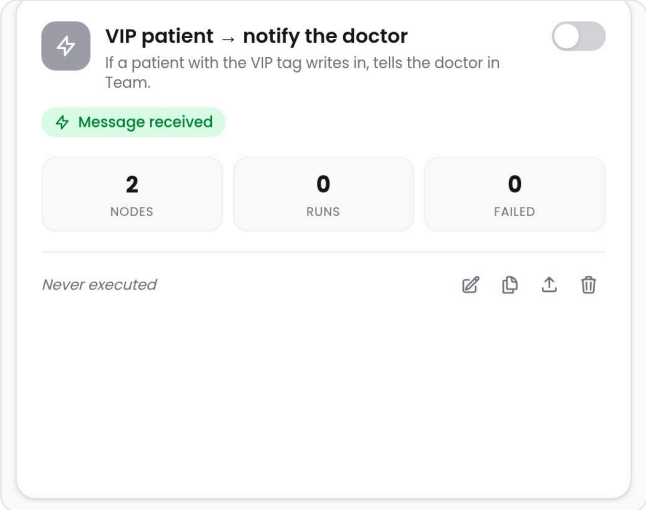


The list: the number of workflows, Templates, Import, New workflow, the filters, the failure notice and the cards.

Each workflow's card

Part	What it shows
Lightning icon	Orange if the workflow is active, gray if it's inactive.
Name and description	Clicking the name opens the editor.
Switch	Turns the workflow on or pauses it instantly (chapter 4).
Trigger pill	The event that sets it in motion, with one color per category.
Nodes	How many steps it has, not counting the trigger.
Runs	How many times it has run.
Failed	How many of those runs ended with an error.
Success rate	Percentage of runs without errors, with a green bar (80% or more), amber (50 to 79%) or red.
Footer	"Last run ... ago" or "Never executed", and the action buttons.

When a workflow fails



A card with recent failures: “3 failures in 7 days”, the last error and the View run link.

If a workflow had errors in the **last 7 days**, its card is marked in red:

- the **N failures in 7 days** badge;
- the **Last failure** box, with how long ago it happened and the error text (in the example, the lab wasn’t responding);
- the **View run #...** link, which opens the editor with that run in the history (chapter 27);
- if a later run went well, the notice **The latest run succeeded**.

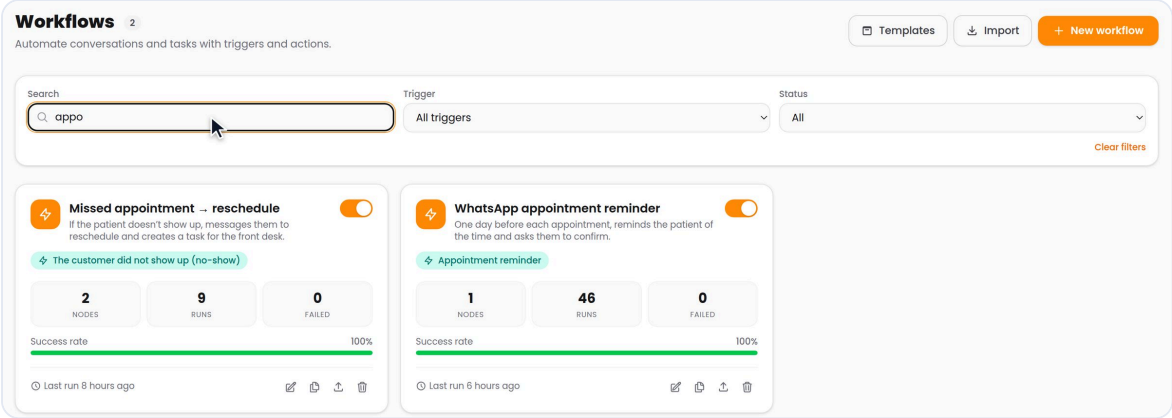
> **NOTE**
> The total runs on the card cover the workflow’s whole life; your plan’s monthly allowance is counted separately (chapter 52).

03 CHAPTER 3 Search and filter

Find a workflow in seconds, or only the ones that are failing.

Above the cards there are three filters, which you can combine:

Filter	What it does
Search	Searches each workflow’s name and description as you type.
Trigger	Shows only the workflows with that trigger. The list has every trigger, grouped by category.
Status	**All**, **Active**, **Inactive** or **With failures (last 7 days)**.

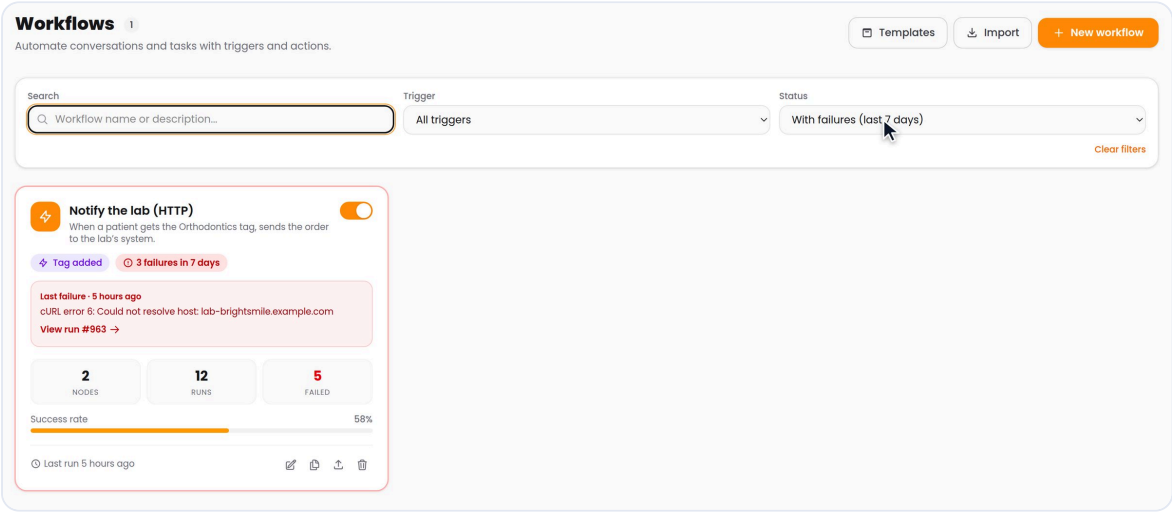


Searching for “appointment”: the two workflows whose name or description contains it remain, and Clear filters appears.

With any filter applied, **Clear filters** appears and takes you back to the full list. Filters are kept in the page address, so you can bookmark it or share it with your team.

The failure notice

If any workflow is failing, a red notice appears at the top (“1 workflow failed in the last 7 days”) with the **Show only failing** button, which sets the status to **With failures** for you.



Status “With failures (last 7 days)”: only the workflow that is failing.

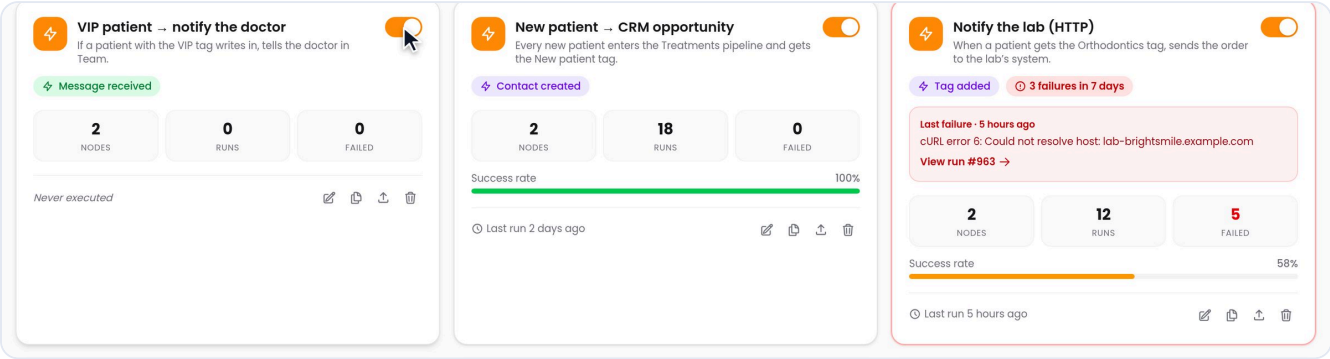
> **TIP**
>
> Check the failure notice once a week. A workflow that fails silently is usually a broken integration, a disconnected channel or a paused WhatsApp template.

04

CHAPTER 4

Turn on, duplicate, export and delete

What you can do with a workflow without opening the editor.



Each card's switch turns the workflow on or pauses it instantly; at the bottom, Edit, Duplicate, Export and Delete.

The switch

The switch on the card turns the workflow on or pauses it **instantly**. An inactive workflow never fires, but it keeps everything: its nodes, its settings and its history. If the workflow has unresolved errors (for example, a node with an empty required field), it can't be turned on: Omnifox shows you the list of what's missing.

The footer buttons

Button	What it does
Edit (pencil)	Opens the workflow in the editor.
Duplicate	Creates an **inactive** copy called "Name (copy)", with the same nodes and connections. It's the safe way to try out big changes without touching the original.
Export	Downloads the workflow as a .json file, to keep it or take it to another workspace (chapter 51).
Delete	Asks you to confirm, because it deletes the workflow with all its nodes, connections and run history.

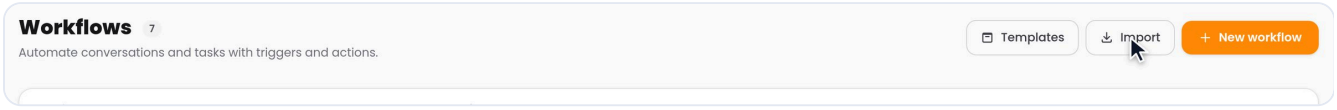
> **NOTE**
>
> Turning on, duplicating, exporting and deleting require the permission to manage workflows. People who can only view them see the list and the editor in read-only mode, without those buttons (chapter 52).

05

CHAPTER 5

Three ways to create a workflow

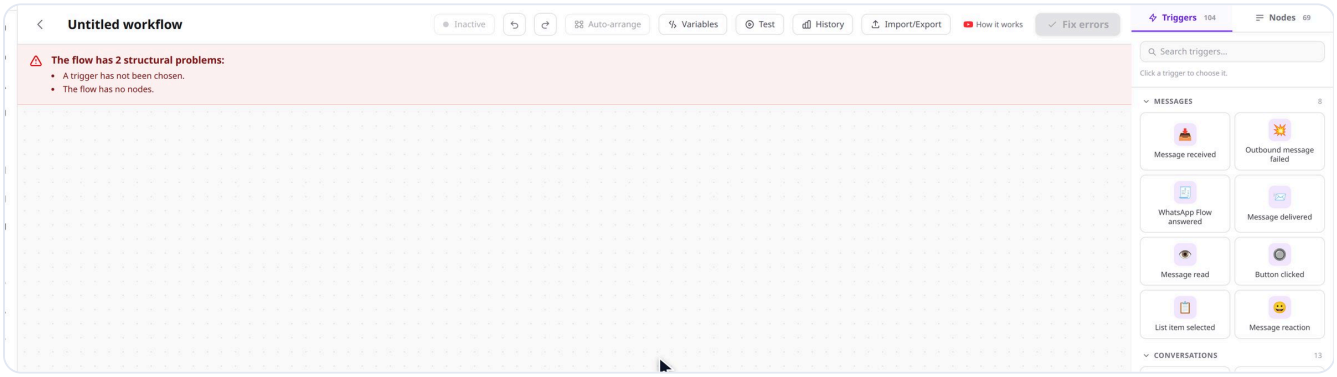
From scratch, from a template or by importing a file.



The three doors: Templates, Import and New workflow.

Path	What you get
New workflow	A blank editor, with a workflow called “Untitled workflow”, with no trigger or nodes.
Templates	A gallery of ready-made workflows for common cases: follow-ups, team alerts, appointments, CRM (chapter 50).
Import	You upload a .json file exported from another workspace or account and match its channels, teams and tags (chapter 51).

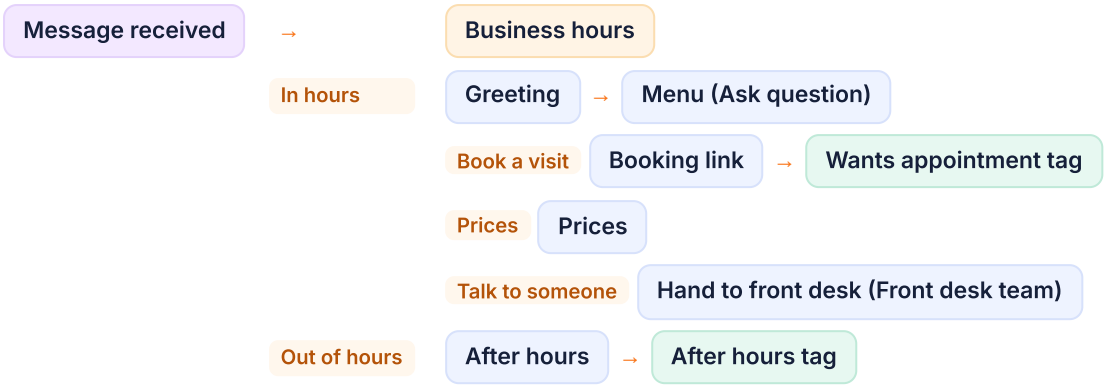
Whichever path you choose, **the workflow starts out inactive**: nothing fires until you turn it on. That way you can prepare and test it calmly.



A newly created workflow: “Untitled workflow”, inactive, with the notice of what’s missing and the trigger palette.

When you create a blank one, the editor warns you in red about the two starting problems: **A trigger has not been chosen** and **The flow has no nodes**. On the right, the palette opens on the **Triggers** tab, ready for the first step.

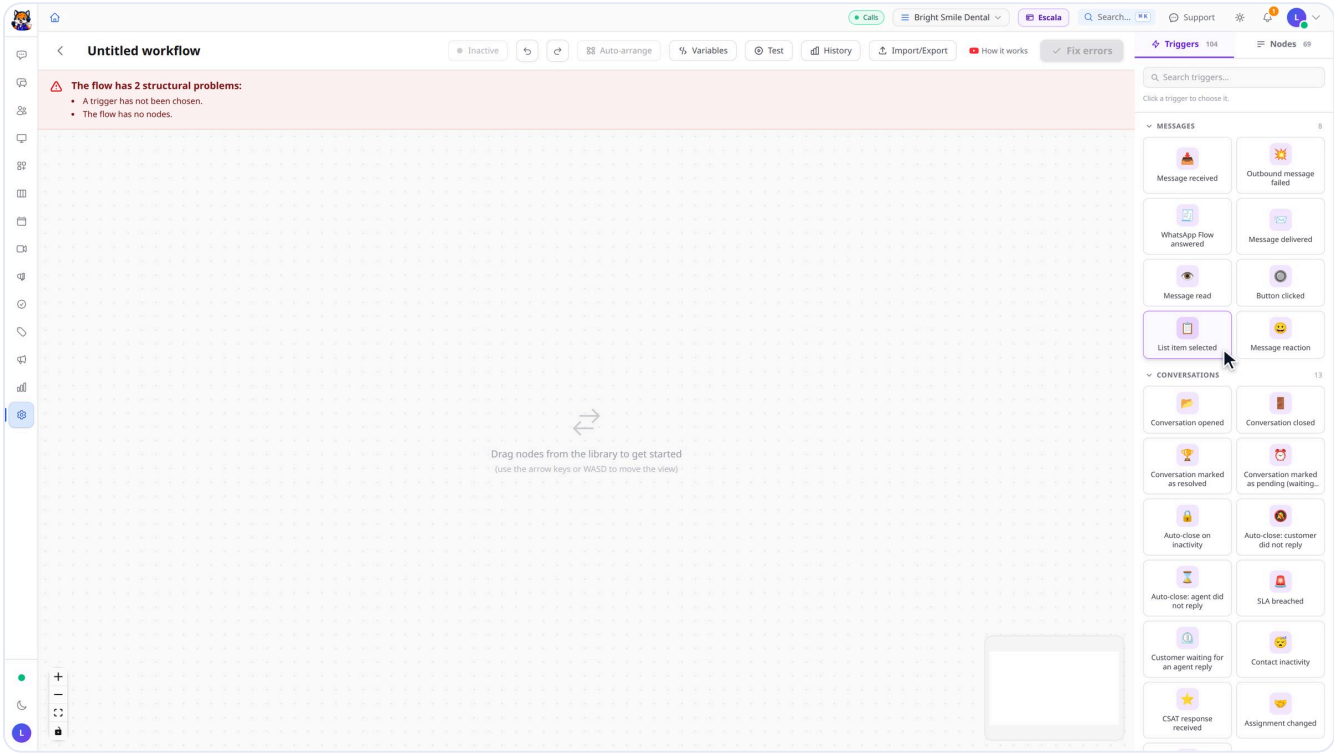
> **WHAT WE ARE GOING TO BUILD**
>
> In chapters 9 to 24 we build “Welcome and menu” from scratch: the workflow that greets every patient who writes to Bright Smile.



CHAPTER 6

06 Editor tour

The canvas, the right-hand panel and the top bar.

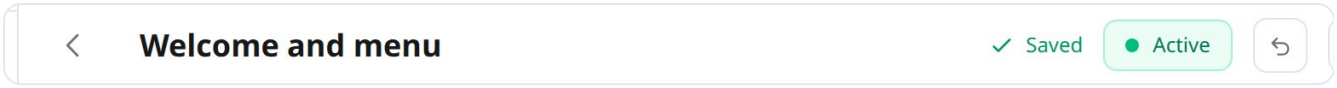


The editor: the canvas on the left, the right-hand panel and the top bar.

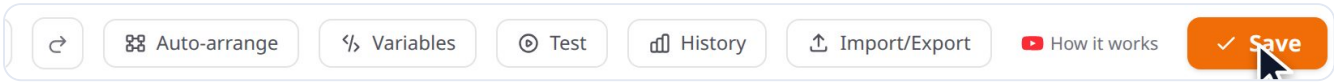
The editor has three areas:

- **The canvas**, on the left: where you draw the workflow with its nodes and connections.
- **The right-hand panel**, which changes depending on what you're doing: the **palette** (triggers and nodes), the **inspector** of the selected node, the **trigger** settings or the **Variables** panel.
- **The top bar**, with all the workflow commands.

The top bar, button by button



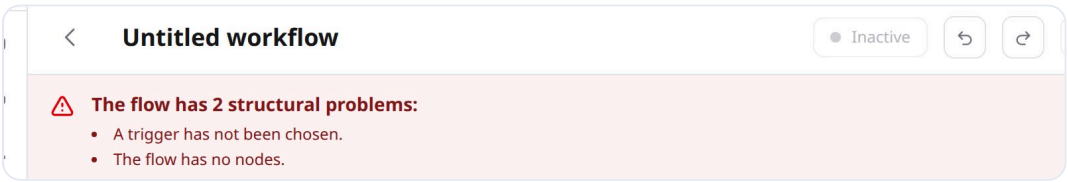
The bar, first half: Back, the name, Saved, Active, Undo, Redo, Auto-arrange and Variables.



Second half: Test, History, Import/Export, How it works and Save.

Button	What it does
Back	Returns to the workflow list. If there are unsaved changes, the editor warns you before leaving.
Workflow name	Click to change it (“Click to rename”).
Save status	**Saved**, **Unsaved** or **Save failed**.
Active / **Inactive**	The pill to switch the workflow on or off. The change applies when you save (chapter 24).
Undo / **Redo**	The arrows; also Ctrl + Z and Ctrl + Y (chapter 23).
Auto-arrange	Lays out the whole workflow from left to right.
Variables	Opens the panel with all the variables you can use (chapter 16).
Test	Starts a test run (chapter 25).
History	The workflow’s latest runs, step by step (chapter 27).
Import/Export	The workflow as JSON text to copy, download or paste (chapter 51).
How it works	Opens this video and this manual.
Save	Saves the workflow. While something is missing it says **Fix errors** and won’t let you save.

Fix errors



With unresolved problems, the main button says Fix errors and the list appears in red at the top.

As long as the workflow is missing something, the main button turns into **Fix errors** and a red notice at the top lists what’s missing: structural problems (“A trigger has not been chosen”, “The flow has no nodes”) or nodes with empty required fields (“1 node needs attention before publishing”), which are also marked in red on the canvas.

Giving it a name

1. Click “Untitled workflow”, at the top left.
2. Type the name: **Welcome and menu**.
3. Press Enter or click outside. The status changes to **Unsaved**.

> **TIP**
> Use names that say what the workflow does, such as “WhatsApp appointment reminder” or “Missed appointment → reschedule”. With twenty workflows on the list, you’ll be glad you did.

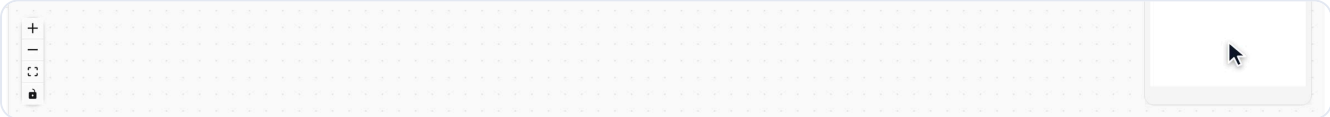
07

CHAPTER 7

Moving around the canvas

Drag, zoom in, zoom out and don’t get lost as the workflow grows.

To…	Do this
Move the view	Drag the canvas background with the mouse, or use the arrow keys (W A S D work too).
Zoom in or out	The mouse wheel, the **+** and **–** buttons at the bottom left, or the + and – keys.
See the whole workflow	The fit button, below + and –: centers the entire workflow on the screen.
Avoid moving things by accident	The padlock locks the canvas; press it again to unlock.
Find your way	The minimap, at the bottom right: it shows the whole workflow and a box with what you’re looking at. You can drag that box.
Move a node	Drag it; with a node selected, the arrow keys move it.



At the bottom left, zoom in, zoom out, fit and lock; at the bottom right, the minimap.

> **NOTE**
> When the canvas is empty, the center says “Drag nodes from the library to get started”. It’s just a hint: it disappears as soon as you add the trigger.

08

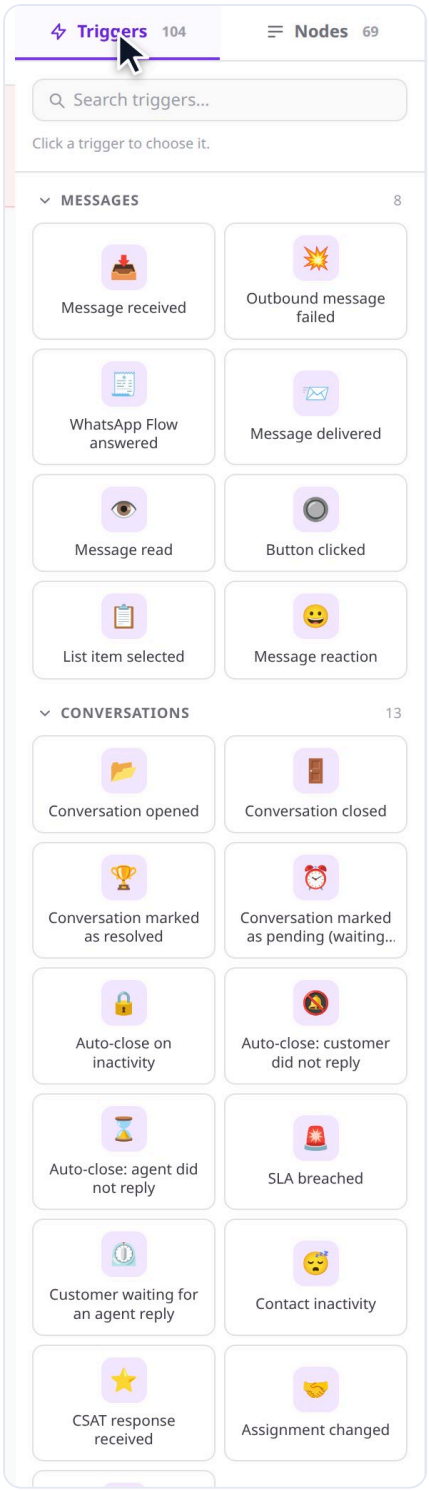
CHAPTER 8

The palette: triggers and nodes

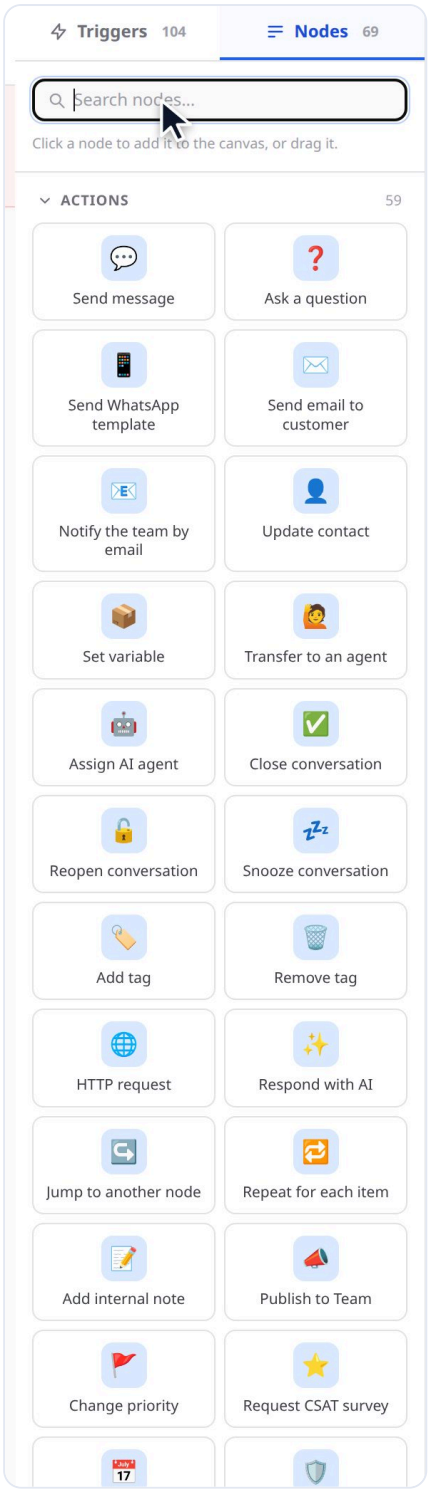
Everything you can put in a workflow, in two tabs.

The palette, on the right of the editor, has two tabs:

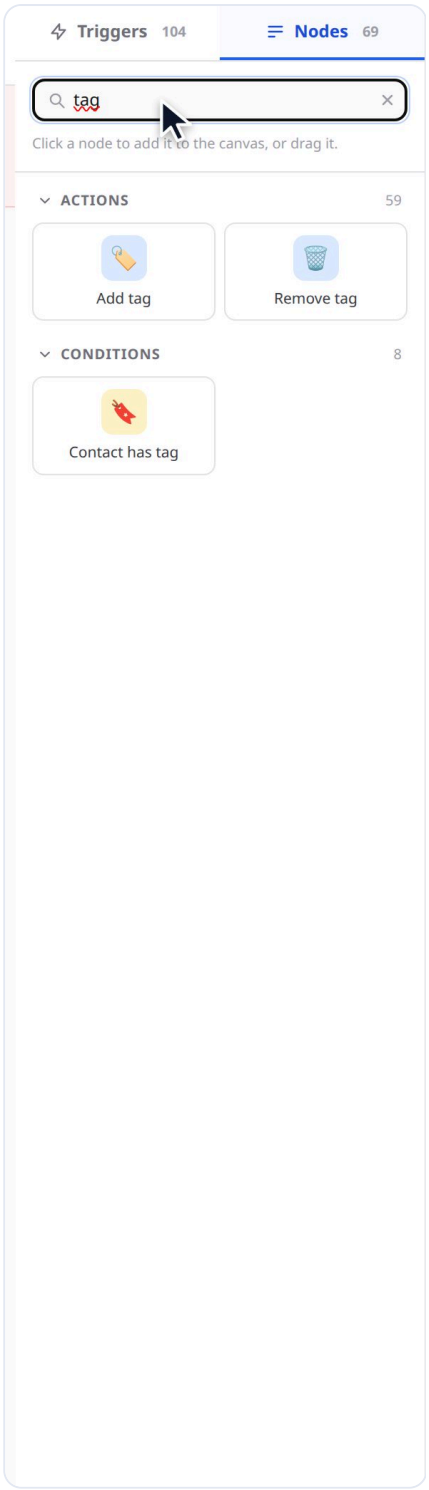
- **Triggers**: every event that can start a workflow, **grouped by category** with collapsible headers: Messages, Conversations, Contacts, System, Call triggers, CRM and records, Boards, Appointments, Internal Team and Time. Clicking a card picks it as the workflow's trigger.
- **Nodes**: actions, conditions and waits, in three collapsible sections with a counter. Clicking a card adds the node in the center of the view; you can also drag it to the exact spot on the canvas.



The Triggers tab, grouped by category.



The Nodes tab: Actions, Conditions and Waits.



The search box: “tag” leaves three nodes.

The search box at the top filters the open tab as you type part of the name (“Search triggers...” or “Search nodes...”). If a node needs a higher plan, its card appears dotted, with a padlock and the plan name (“Available on the ... plan or higher”).

All the nodes at a glance

This is the quick reference for the **Nodes** tab. Chapters 14 to 21 and 30 to 49 explain each one with its fields.

Actions · messages and conversation

Node	What it does	Outputs
Send message	Sends text, or an image or file by URL with caption text.	1 output
Ask question	Asks and pauses until the answer arrives; saves it in a variable.	One per option (or next) + Invalid + Timeout
Send WhatsApp template	Sends a Meta-approved template, with its parameters.	1 output
Send email to customer	An email to the contact from your Email channel.	1 output
Notify the team by email	An internal email notice to workspace members.	1 output
Transfer to an agent	Assigns to a person, to a person in a team, to a team, or removes the assignment.	1 output
Assign AI agent	Hands the conversation to an AI agent.	1 output
Close conversation	Closes it or marks it as resolved.	1 output
Reopen conversation	Opens it again.	1 output
Snooze conversation	Takes it out of the inbox for a while.	1 output
Add internal note	A note only your team can see.	1 output
Set priority	No priority, low, medium, high or urgent.	1 output
Request CSAT survey	Sends the workspace's satisfaction survey.	1 output
Schedule message	Leaves a message ready to go out later.	1 output
Look up prior conversation	Finds the contact's latest conversation.	Found, with agent · Found, no agent · Not found

Actions · contact and data

Node	What it does	Outputs
Update contact	Writes profile fields (custom ones too).	1 output
Set variable	Stores values for the next steps.	1 output
Add tag · Remove tag	Adds tags to or removes them from the contact.	1 output
Find or create contact	Identifies them by phone or email and creates them if they don't exist.	Created · Found · Not identified

Actions · AI, integrations and control

Node	What it does	Outputs
AI respond	An AI agent replies at that point (uses AI credit).	1 output
HTTP request	Calls another system's API.	Success · Error
Jump to another node	Continues at another node of the workflow.	1 output

Node	What it does	Outputs
Repeat for each item	Goes through a list, one pass per item.	One pass per item · When done
If it fails... (handle errors)	Checks the previous step.	Success · On error
Run in parallel	Opens 1 to 5 branches.	One per branch
Merge branches	Waits for all branches to arrive.	1 output
Call another workflow	Runs another of your workflows with a Manual trigger.	1 output
Publish to Team	Posts in a Team space.	1 output

Actions · CRM, orders and appointments

Node	What it does	Outputs
Create deal (CRM) · Update deal (CRM) · Assign deal owner (CRM)	Creates or changes pipeline deals.	1 output
Log activity / note (CRM) · Create follow-up task (CRM)	Leaves a record on the profile or creates a task with a due date.	1 output
Change order status · Send cart link · Empty the cart · Turn the cart into an order	Catalog orders and cart.	1 output
Schedule appointment (Calendar) · Reschedule appointment · Cancel appointment · Confirm appointment · Mark as no-show · Add to waitlist · Check availability · Block time in the calendar	The Calendar schedule.	1 output

Actions · Projects and Team

Node	What it does	Outputs
Create board item · Update task cell (board) · Complete task (board) · Move task to another group (board)	Cards on Projects boards.	1 output
Create project from deal · Create board card (from the conversation) · Create board card (from the appointment) · Link record (item ↔ deal)	Cards created from other records.	1 output
Create Team reminder · Schedule Team message · Send Team DM	Internal communication.	1 output

Conditions

Node	What it does	Outputs
If (a single rule)	Compares a field with a value.	Yes · No
Multiple branches (switch)	Several branches with rules; the first one that matches wins.	One per branch + Default
AI classifier (route by intent)	The AI picks one of your categories (uses credit).	One per category + No match
Contact has tag	Any, all or none of the chosen tags.	Yes · No
Detect language	Works out the contact's language.	One per language + Other
Agent belongs to team	Checks the agent's team.	Yes · No
Business hours	Compares the time with the workspace's hours.	In hours · Out of hours · Holiday
Date / time	Compares the current date and time.	Yes · No

Waits

Node	What it does	Outputs
Wait time	Pauses for minutes, hours or days.	1 output
Wait until an event happens	Pauses until an event or until the time limit.	Event · Timeout

> **NOTE**
> The appointment, CRM, order, Projects and Team nodes work with those modules: if your workspace doesn't use them, you simply don't need them. AI nodes (AI respond, AI classifier, and Detect language when it uses AI) use AI credit.

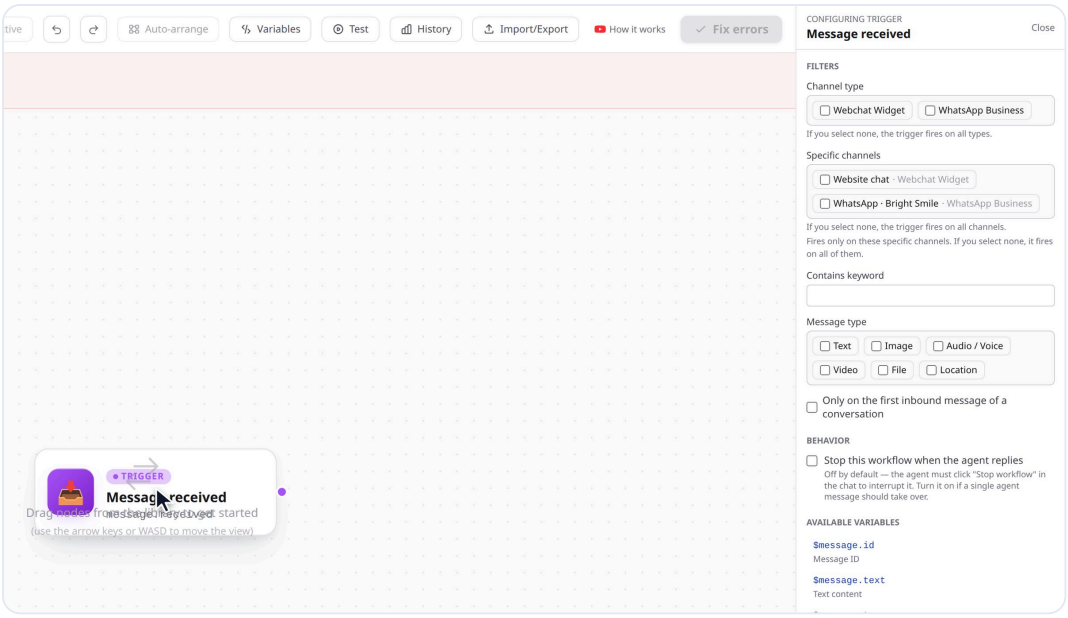
09

CHAPTER 9

Choosing the trigger

Everything starts with the event that sets the workflow in motion.

1. On the **Triggers** tab, type "message" in the search box, or open the **Messages** category.
2. Click **Message received**.
3. The **TRIGGER** card appears on the canvas and, on the right, its settings: **Configuring trigger · Message received**.



Message received chosen: its card on the canvas and its settings on the right.

A workflow has **only one trigger**. You can change it from its settings (the selector at the top, with a search box and groups by category); Omnifox warns you that **changing the trigger may invalidate the previous trigger's filters**, so check them afterwards.

Depending on the trigger, its settings include: a notice of its own (if it needs one), the **Filters** (chapter 10), the **Run only once per contact** checkbox (on the ones that support it), the **Behavior** section and the **Available variables** (chapter 11). If a trigger has no filters, it says so: it fires on every matching event.

All the triggers

They're listed in the order and with the names you see in the palette. Call triggers only appear on plans with calls, and the Azur ones only with that integration connected.

Messages

Trigger	When it fires
Message received	A message arrives from the customer. Filters: channel type, specific channels, keyword, message type and "Only on the first inbound message of a conversation".
Outbound message failed	A message you sent could not be delivered (for example, no WhatsApp balance). Filter by error code.
WhatsApp Flow answered	The customer completed a WhatsApp Flows form. Filter by Flow ID.
Message delivered	One of your messages reached the customer's phone.
Message read	The customer read your message.
Button clicked	The customer tapped a reply button (from a template or an interactive message). Filter by button ID.
List item selected	The customer picked a row from a list message. Filter by row ID.
Message reaction	The customer reacted to a message with an emoji. Filter by emoji.

Conversations

Trigger	When it fires
Conversation opened	A new conversation opens or a closed one is reopened. Filters: source, channel, team and previous status. Supports “Run only once per contact”.
Conversation closed	Someone (or the system) closed the conversation. Filters: source, channel, team that closed it and resolution time.
Conversation marked as resolved	The conversation was marked as resolved.
Conversation marked as pending (waiting on the customer)	The agent left it waiting for the customer’s reply.
Auto-close on inactivity	The workspace’s auto-close closed the conversation. Filter by case (customer or agent did not reply).
Auto-close: customer did not reply	It was closed or resolved automatically because the customer stopped replying.
Auto-close: agent did not reply	The agent inactivity rule kicked in: resolved, closed, reassigned to someone on shift, or tagged as unanswered.
SLA breached	The first-response or resolution time ran out. Filters: channel, team and metric.
Customer waiting for an agent reply	The customer wrote and nobody on the team is answering.
Contact inactivity	The contact hasn’t replied for N minutes (required field). “Maximum age” keeps it from firing on your whole history when you turn it on.
CSAT response received	A reply to the satisfaction survey arrived. Filters: minimum and maximum score.
Assignment changed	The conversation was assigned or reassigned to another person or team.
The AI handed off the conversation	An AI agent passed the case to a person. Filter by handoff reason.

Contacts

Trigger	When it fires
Contact created	A new contact was created (by hand, from a message, from a form or through the API).
Contact field updated	A field on the profile changed. Filters: field (required), old value, new value, only if it was empty before, and source.
Tag added	The contact got a tag. Filter by name; empty = any tag.
Tag removed	A tag was removed from the contact.

System

Trigger	When it fires
Incoming webhook	Another system sends data to the workflow’s own URL (chapter 45).
Scheduled (cron)	Runs on a schedule: every X minutes, hourly, daily, weekly or monthly (chapter 45).

Trigger	When it fires
Shortcut (manual from the inbox)	A button your team presses from a conversation (chapter 44).
Manual (called by another workflow or the AI)	It doesn't start on its own: another workflow runs it with "Call another workflow", or an AI agent with its tool (chapters 41 and 43).
Template approved	Meta approved a WhatsApp template.
Template rejected	Meta rejected a WhatsApp template.
Template paused	Meta paused a template for low quality.
WABA quality changed	The quality rating of your WhatsApp Business account changed.
Channel disconnected	A channel of any type (WhatsApp, Instagram, Messenger and the rest) lost its connection. It doesn't repeat for the same channel within 1 hour.

Call triggers (plans with calls)

Trigger	When it fires
Call received	A call comes in.
Call answered	Someone on the team answered.
Missed call	Nobody answered. Perfect for writing to the customer on WhatsApp.
Call rejected by agent	The agent rejected the call.
Call ended	The call ended (with its duration).
Call duration exceeds threshold	The call lasted longer than the number of seconds you set.
Contact's first call	It's the first time that contact calls.
Repeat caller (recurring contact)	The contact called several times in a short period.
Out-of-hours call	A call came in outside business hours.
Recording ready	The call recording is available.
Recording transcribed	The recording now has its transcript.
Voicemail received	Someone left a voicemail.
Voicemail transcribed	The voicemail now has its transcript.
IVR: option selected	The caller picked an option from the phone menu. Filter by digit.
IVR: abandoned before selection	They hung up in the phone menu without choosing anything.

CRM and records

Trigger	When it fires
Record created · Record updated · Stage / status changed · Record completed · Record overdue	Generic events for CRM records (deals, tasks and other objects), with a filter by record type.
Deal created	A deal was created in a pipeline.
Deal changed stage	The deal moved to another stage of the pipeline.

Trigger	When it fires
Deal won · Deal lost	It was marked as won or as lost.
Deal with no activity (cold)	An open deal has gone days without any activity. Filters: pipeline and minimum days.
Deal stuck in a stage	It has gone too many days without changing stage. Filters: days, stage and pipeline.
Deal priority changed	The priority went up or down.
Deal customer not replying	The contact of an open deal isn't answering.
Contact not responding (N days)	You wrote to the contact and they haven't replied for N days.
Overdue activity	A CRM task or activity passed its date without being completed.
Invoice authorized in Azur · Invoice voided in Azur	Only with the Azur integration (electronic invoicing) connected.
Order created · confirmed · status changed · paid · canceled	The life cycle of a catalog order. "Order created" filters by source and minimum total.
Cart created · updated · abandoned · turned into an order · expired without purchase	The customer's cart. "Abandoned cart" filters by minimum total, source and minutes of silence.
Payment request created · Payment receipt received · Payment request partially paid · Payment request paid · Payment request expired unpaid · Payment request refunded	The payment requests you send from the conversation.

Boards

Trigger	When it fires
Board column changed	The value of a column on a card changed. Filters: board and column.
Board item created	A card was created on a board.
Item entered a group	A card was moved into a group.
Item assigned to a person	A card got an owner.
Task status changed	The status of a task changed.
Subtask created	A subtask was created.
Task due soon (≤24h)	A task has 24 hours or less left.
Task overdue	A task passed its date without being finished.

Appointments

Trigger	When it fires
Appointment created	An appointment was booked (from the booking page, the AI or your team). Filter by calendar.
Appointment cancelled	An appointment was cancelled.

Trigger	When it fires
Appointment reminder	It's time for the appointment reminder (sent once per appointment). With **Hours before the appointment**, each workflow gets its own window (chapter 47).
Appointment rescheduled	The appointment was moved to another time.
Appointment confirmed by the customer	The customer confirmed they will attend.
The customer did not show up (no-show)	The customer was marked as a no-show.
Appointment completed	The appointment was marked as completed.
A spot opened up	A cancellation freed up a time slot.
Someone from the waitlist got a spot	Someone on the waitlist got the slot.

Internal Team

Trigger	When it fires
You were mentioned in Team	Someone mentioned you in a Team space or direct message.
Message posted in a Team Space	A message was posted in a space (filter by word).
Team reminder is due	It's time for a Team reminder.
Team poll closed	An internal poll was closed.

Time

Trigger	When it fires
Relative date (e.g. 3 days before it is due)	A few days before or after a record's date. Negative offset days = before, positive = after.

Workflow identifier

At the end of the trigger panel is the **Workflow identifier**: "The AI and other integrations can call this workflow by this identifier. Lowercase letters, numbers and hyphens only; unique in the workspace.". Type a clear one, for example "welcome-vip", and click **Save** (or press Enter): it saves right away, without saving the workflow ("Identifier saved."). As you type, it shows how it will end up ("Will be saved as: ..."). If another workflow in the workspace already uses it, it warns you and suggests one ending in "-2".

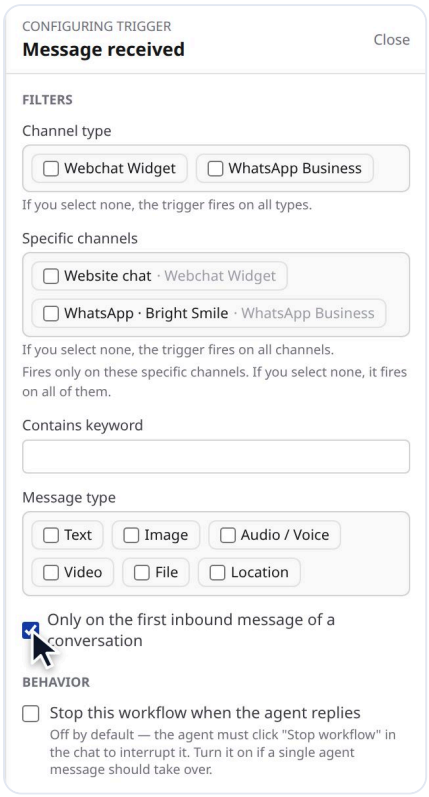
> **NOTE**
>
> The trigger card on the canvas shows only its name, for example "Message received"; the technical key no longer appears under it.

10

CHAPTER 10

Trigger filters

Filters decide when it fires and when it doesn't.



The Message received filters, with "Only on the first inbound message of a conversation" checked.

Filter	What for
Channel type	Only web chat, only WhatsApp Business, etc. If you don't check any, it fires on all of them.
Specific channels	Only on those specific channels (for example, "WhatsApp · Bright Smile").
Contains keyword	Only if the message contains that text (not case-sensitive).
Message type	Text, Image, Audio / Voice, Video, File or Location.
Only on the first inbound message of a conversation	The key to a welcome: it fires on the first message, not on every message.

For "Welcome and menu" we leave all channels and check **Only on the first inbound message of a conversation**: that way the patient gets the menu once when they start writing, and not every time they send something.

How they combine

- An empty filter doesn't restrict anything.
- If you fill in several filters, the event has to meet **all** of them.
- Within a filter with several options checked (for example, two channel types), **one** is enough.
- Text filters check that the value **contains** what you typed; number filters, the exact value or the minimum/maximum they state.

Run only once per contact

Some triggers (for example, **Conversation opened**) have this checkbox: after the first run for a contact, later events from that same contact are skipped. Handy for a once-in-a-lifetime welcome.

WATCH OUT

If a message doesn't meet the filters, the workflow simply doesn't fire, **without leaving an error** or a run in the history. When a workflow "doesn't start", the first thing to check is its filters.

11

CHAPTER 11

Stop when an agent replies, and trigger variables

Have the workflow step back on its own when a person joins, and the data the event brings.

Behavior: Stop this workflow when the agent replies

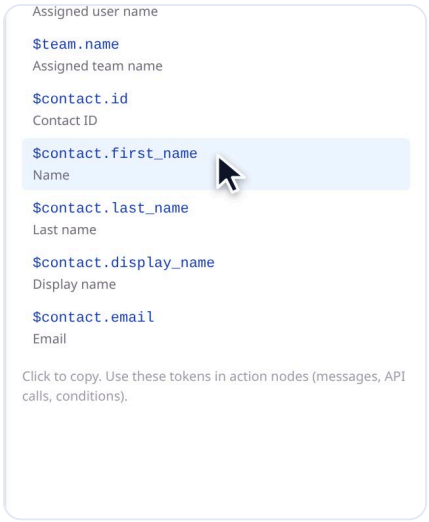


Behavior: Stop this workflow when the agent replies.

It's **off** by default. When it's off, the workflow keeps going even if someone on your team writes in the conversation; the agent can stop it by hand with **Stop workflow** from the chat (chapter 29).

Turn it on if you want the workflow to step back on its own as soon as someone on your team **writes**, **takes the conversation** or **answers a call**. It's the recommended setting for follow-ups and sequences with long waits, so the customer doesn't get an automatic message in the middle of a chat with a person.

Available variables



The trigger's available variables: one click copies them.

Below are the data this trigger brings: for Message received, the message text (`{{$message.text}}`), its type, the attachment, the channel (`{{$conversation.channel_name}}`), the assignee and the contact's data (`{{$contact.first_name}}` , `{{$contact.phone}}` ...).

Each variable has its description. **One click copies it**, ready to paste into a message or a field. Each trigger brings its own: an appointment brings the date and the service; a deal, its stage and its amount.

12

CHAPTER 12

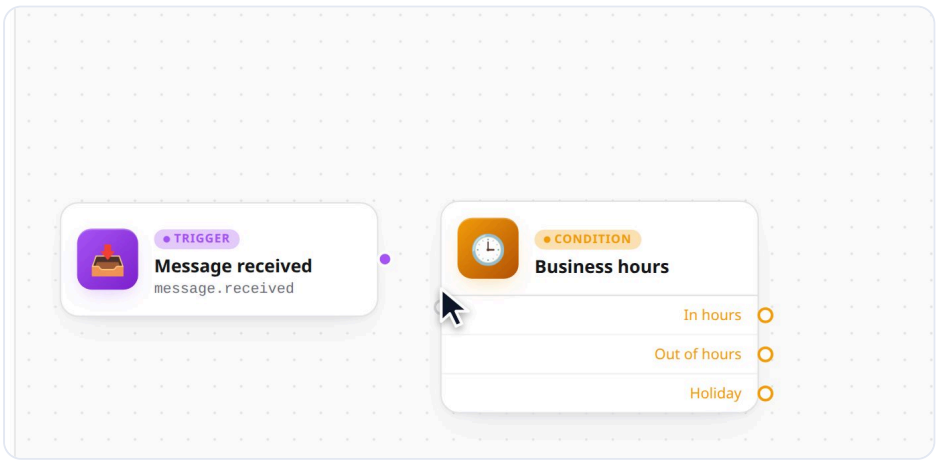
Adding nodes to the canvas

Drag, click or use the canvas menu.

First it's worth making room: we zoom out a little with the **–** button and move the trigger card to the left, so the whole workflow fits on the screen.

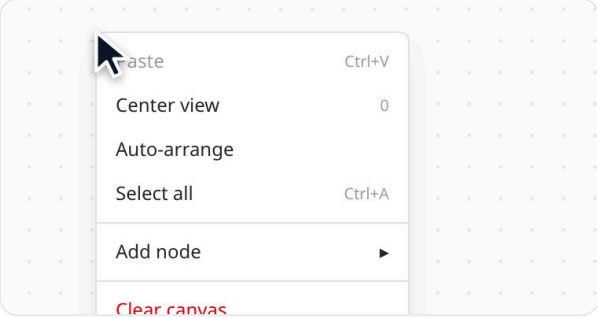
Three ways to add a node

1. **Drag**: on the **Nodes** tab, drag the card to the spot on the canvas where you want it.
2. **Click**: clicking the card adds the node in the center of the view (if there's already one there, it shifts down).
3. **Right-click on the canvas → Add node**: a search box opens with all the nodes.



Business hours already on the canvas, next to the trigger.

We drag **Business hours** (Conditions section) next to the trigger.



Right-click on the canvas: Paste, Center view, Auto-arrange, Select all, Add node...

The right-click menu on the canvas background also has **Paste**, **Center view**, **Auto-arrange**, **Select all**, **Undo**, **Redo** and **Clear canvas**.

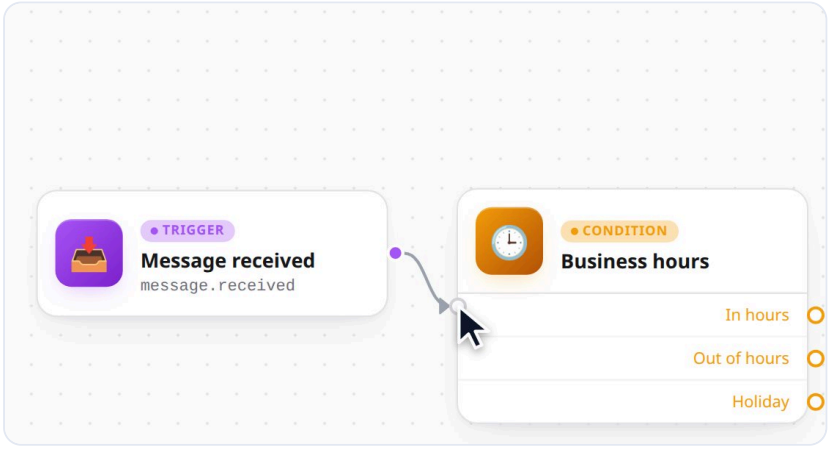
> **NOTE**
> Each node starts with sensible default values (for example, Wait time = 5 minutes). What you do have to fill in are the required fields, marked with an asterisk: until then, the node shows in red.

CHAPTER 13

13 Connecting nodes and branches

Drag from an output to an input; each branch has its name.

1. Find the node's **output** point, on its right.
2. Drag it to the **input** point of the next node, on its left.
3. Let go: the connection is made. That's how we connect the trigger to **Business hours**.



The trigger connected to Business hours, with its In hours, Out of hours and Holiday outputs.

Condition nodes have **one output per branch**, each with its name: Business hours has **In hours**, **Out of hours** and **Holiday**. The connection keeps the name of the branch it comes from, and the workflow follows the right one in each case. An unconnected output is a path where the workflow ends.

Deleting or renaming a connection

- Select it with a click and press Del, or use its menu (**Delete connection**).
- Double-click the connection's label to edit it.

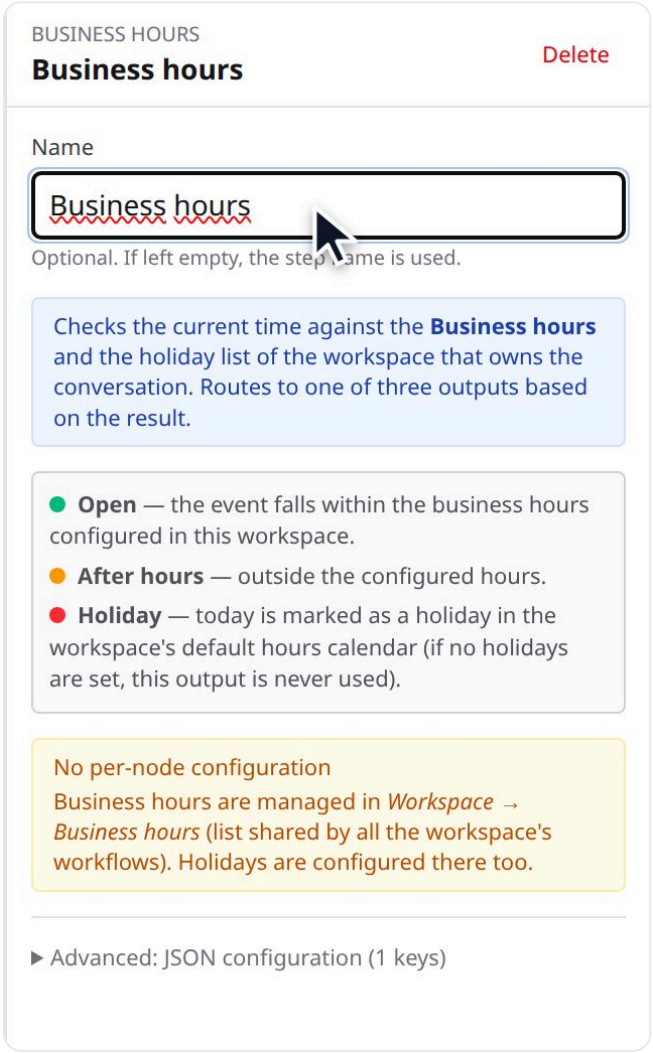
> **NOTE**
> Nothing can connect into the trigger: it's always the first step. And one output can go to several nodes at once if you need it to.

14

CHAPTER 14

Business hours

Reply differently depending on the time, using your workspace's hours.

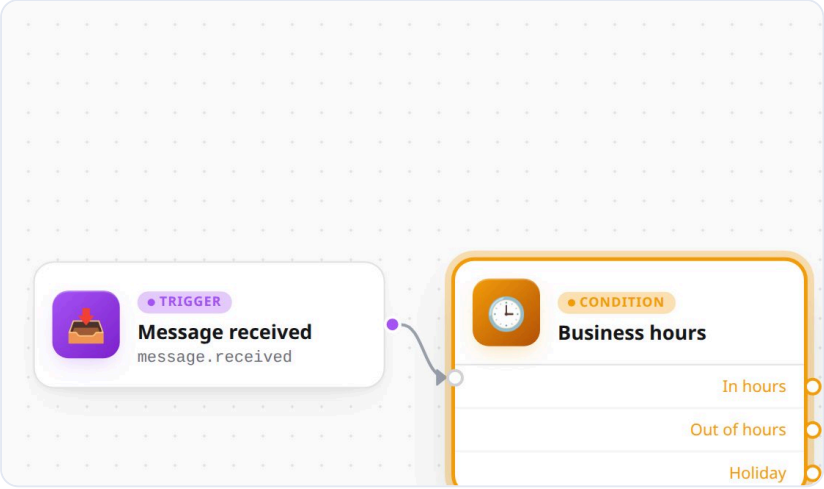


The Business hours inspector: it has no fields, it uses the workspace's hours.

Business hours compares the current time with **your workspace's business hours**, in its time zone, and continues along one of three outputs:

Output	When
In hours	The time is within the configured hours.
Out of hours	It's outside the hours.
Holiday	The date is a holiday in the workspace's default hours calendar. If you don't connect this output, on that day the workflow ends there.

The node **has no fields**: there's a single schedule for the whole workspace, in the workspace settings (Business hours), and every workflow shares it. If you change the clinic's hours tomorrow, every workflow follows them without you touching anything.



The selected node and its three outputs.

At Bright Smile the hours are Monday to Friday from 9 AM to 7 PM, and Saturdays until 2 PM.

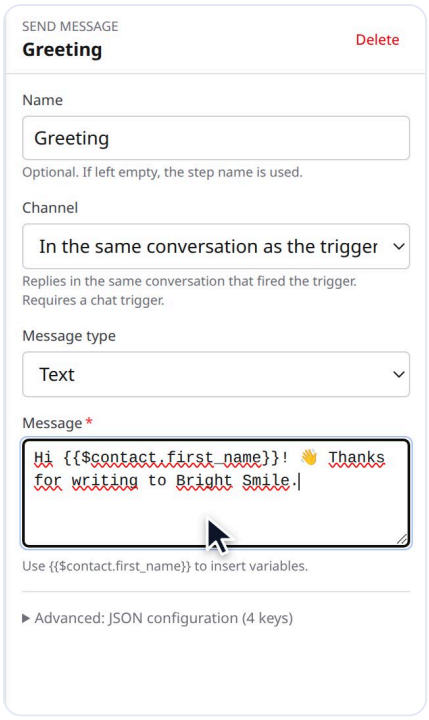
> **TIP**
> Connect the Holiday output too (for example, to the same after-hours message, or to one that says “we’re closed today for the holiday”). That way no day goes without a reply.

CHAPTER 15

15 Send message

The node you’ll use most, with its inspector field by field.

1. On the **Nodes** tab, add **Send message**.
2. Connect it from the **In hours** output of Business hours.
3. Click the node: its **inspector** opens on the right.
4. In **Name**, type “Greeting”: that way you’ll recognize it on the canvas.
5. Fill in the channel and the message.



Send message “Greeting”: Name, Channel, Message type and Message with the name variable.

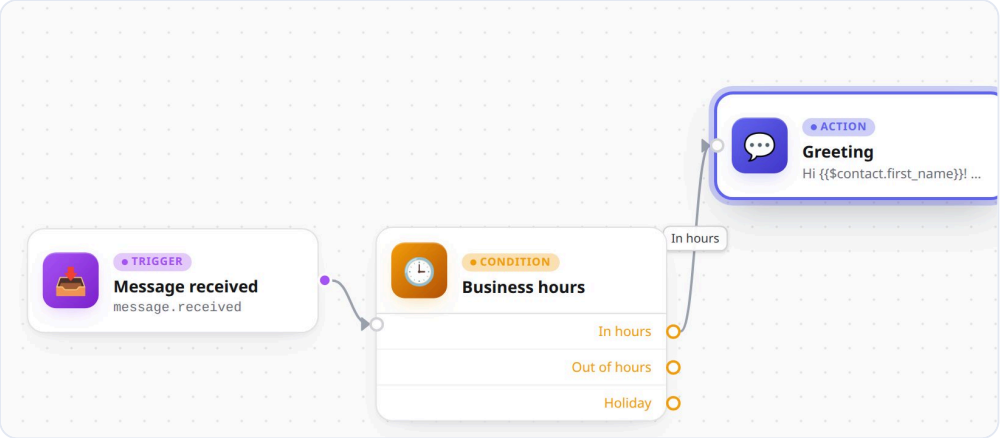
Field	What it is
Name	Optional. If you leave it empty, the step’s name is used.
Channel	Where the message goes out (see below).
Message type	**Text**, **Image** or **File**. With Image or File, **Image or file URL** appears (and the file name); the message text goes as the caption.
Message *	The text, with variables. Required.
Advanced	“JSON configuration”: the raw node, for advanced users.

Channel

Option	What it does
In the same conversation as the trigger	Replies in the conversation that started the workflow. Requires a chat trigger (message received, conversation opened...). It’s the one for a welcome.
On a specific channel	Finds or reuses an open conversation with the contact on that channel; if there isn’t one, it creates one.
Start a new conversation on a channel	Always creates a new conversation on the chosen channel. Useful for proactive outreach and reminders.
On all of the contact’s channels	Sends one message on each channel the contact is linked to.

The greeting message:

Hi `{{$contact.first_name}}` ! 👋 Thanks for writing to Bright Smile.



The greeting connected to the In hours output.

Notice the patient's name: it's a **variable**. When sending, Omnifox replaces it with the real name.

> **WATCH OUT**
> On WhatsApp, Instagram and Messenger you can only write freely within 24 hours of the customer's last message. A welcome replies to a message that just arrived, so that's no problem; to write later, use a template (chapter 36).

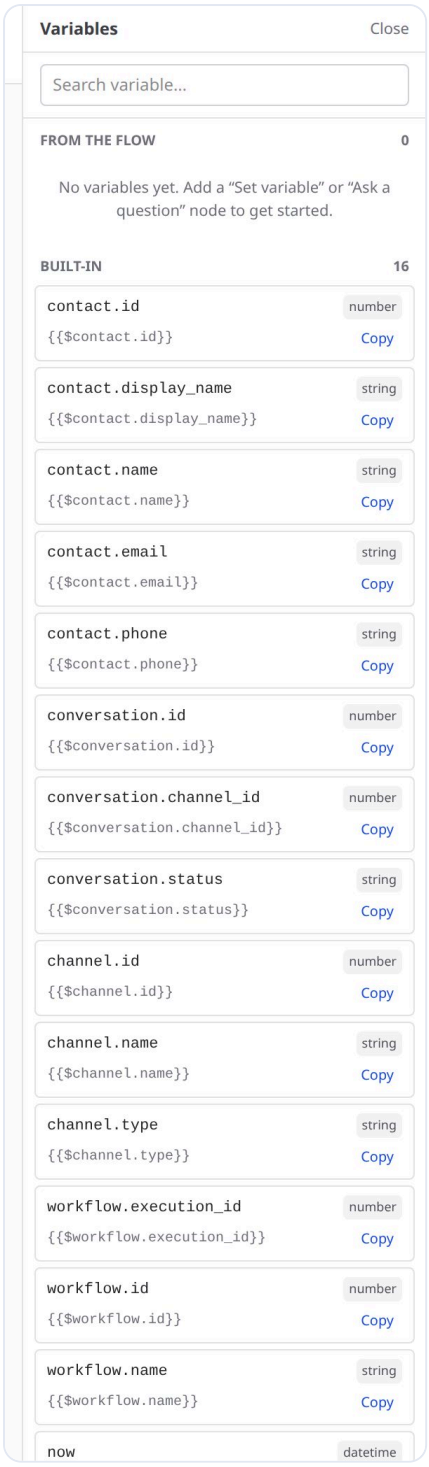
16

CHAPTER 16

Variables

Data that changes in every conversation, between double curly braces.

Variables are written between **double curly braces with a dollar sign**: `{{$contact.first_name}}`. When sending, Omnifox replaces it with the real data. **If a piece of data doesn't exist, it stays empty**, never the code: the patient never sees strange braces.



The Variables panel: search box, From the flow and Built-in, with Copy on each one.

The **Variables** button on the bar opens the panel with all the ones you can use, with a search box and three groups:

- **From the trigger**: the ones the event brings (chapter 11).
- **From the flow**: the ones created by the nodes of this workflow, such as the answer to a question, with **Go to node** so you know where they come from. If there aren’t any yet, the panel suggests adding “Set variable” or “Ask question”.
- **Built-in**: always available.

Each one has **Copy** (“Copy as {{variable}}”): you paste it into the message and you’re done.

The most used variables

Variable	What it brings
`{{$contact.first_name}}` · `{{$contact.last_name}}` · `{{$contact.display_name}}` · `{{$contact.name}}`	The contact's first name, last name, display name and full name.
`{{$contact.email}}` · `{{$contact.phone}}` · `{{$contact.company}}` · `{{$contact.city}}` · `{{$contact.country}}` · `{{$contact.language}}`	Profile data.
`{{$conversation.id}}` · `{{$conversation.status}}` · `{{$conversation.priority}}` · `{{$conversation.assignee}}` · `{{$conversation.assignee_team}}`	The conversation: its status, priority, assignee and team.
`{{$channel.name}}` · `{{$channel.type}}` · `{{$conversation.channel_name}}`	The channel.
`{{$message.text}}` · `{{$message.type}}` · `{{$message.media_url}}`	The message that triggered the workflow.
`{{$user.name}}` · `{{$team.name}}`	The assigned person and team.
`{{$now}}` · `{{$today}}` · `{{$now.hour}}` · `{{$now.date}}` · `{{$now.time}}`	The date and time, in the workspace's time zone (`{{$now.iso}}` is in UTC).
`{{$workflow.name}}` · `{{$workflow.id}}` · `{{$workflow.execution_id}}`	The workflow and the current run.
`{{$opcion}}` · `{{$answer}}` · `{{$answer.label}}`	The answer to a question (with the variable name you chose).
`{{$ai_category}}` · `{{$ai.response}}`	What the AI decided or wrote (chapter 40).
`{{$webhook.payload.campo}}`	A piece of data sent by another system (chapter 45); you can reach into lists and objects with dots.
`{{steps.last.created_deal_id}}`	The result of the previous step (for example, the deal that was just created).

TIP

A price with a dollar sign, such as "Cleaning: $35", is written as is: only what goes between double curly braces, or a known variable like $contact.first_name, gets replaced.

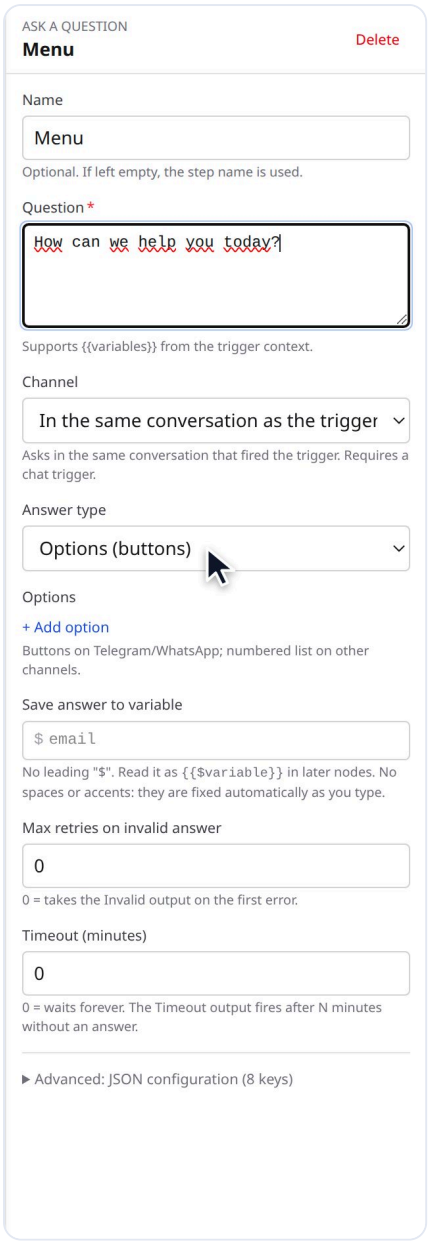
17

CHAPTER 17

Asking a question

Ask, pause the workflow and continue based on the answer.

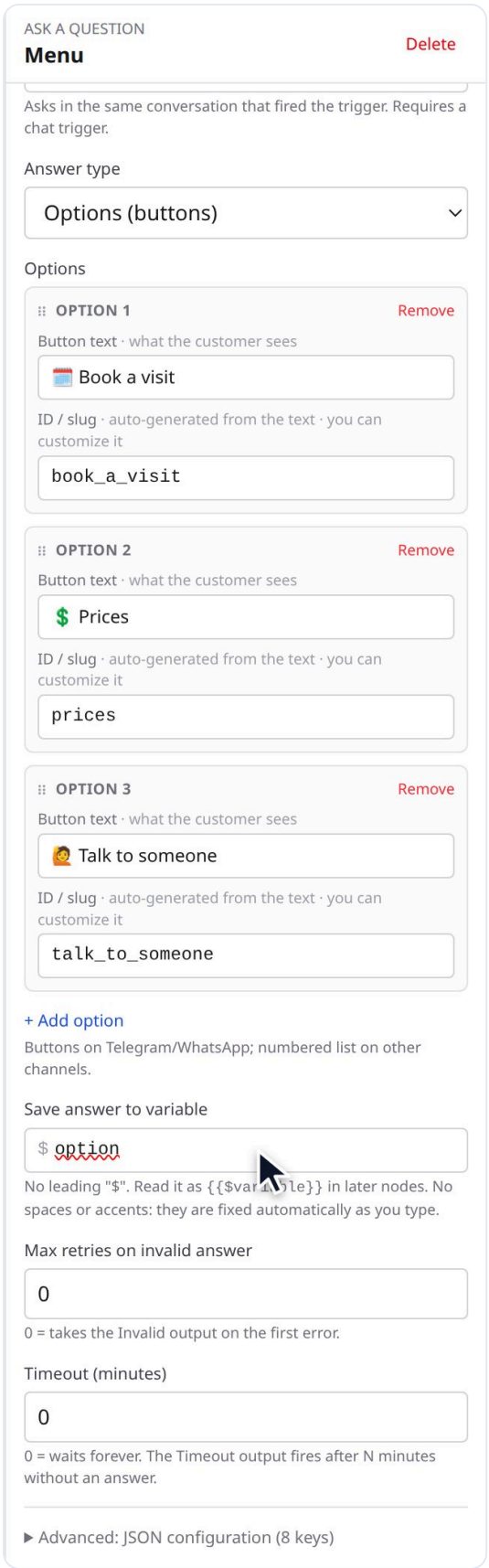
Ask question sends a question and **pauses the workflow** until the patient replies. We connect it after the greeting, call it **Menu** and write the question: "How can we help you today?".



Ask question “Menu”: the question, the channel and the Options (buttons) answer type.

Field	What it is
Question *	The text. Supports variables.
Channel	Same as in Send message: in the same conversation, on a specific channel or starting a new one.
Answer type	**Free text**, **Options (buttons)**, **Number**, **Date**, **Phone**, **Email** or **URL / website**. Except for Free text, Omnifox validates the answer.
Save answer to variable	The name, without a leading “$”. No spaces or accents: they’re fixed automatically as you type.
Max retries on invalid answer	0 = takes the Invalid output on the first mistake.
Retry message (optional)	What gets sent on a retry, for example “I didn’t understand, please try again”.
Timeout (minutes)	How long to wait. 0 = waits forever; with a number, the Timeout output fires after that many minutes without a reply.

Options with buttons



The three options with their text and ID, the opcion variable, the retries and the time limit.

With **Options (buttons)** you write each option with **+ Add option**. Each one has:

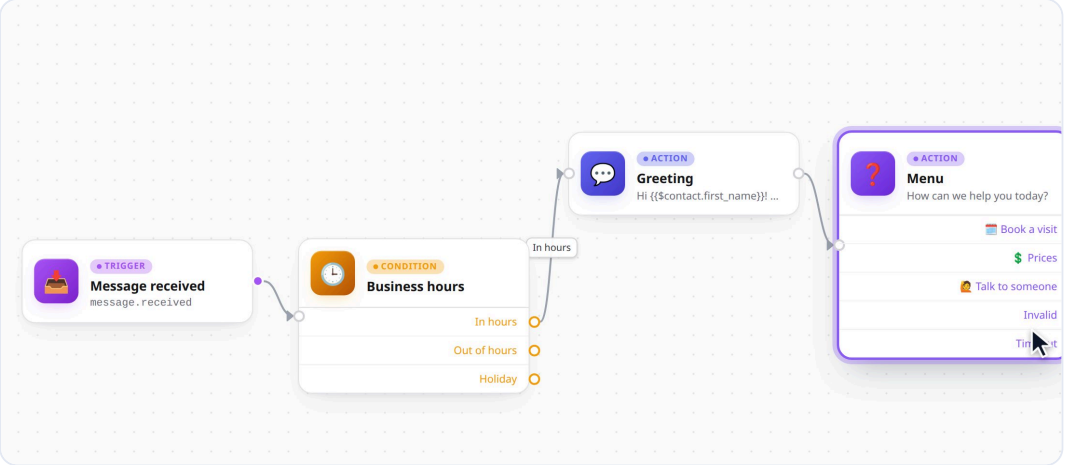
- **Button text**, what the customer sees. Up to **20 characters** so it fits in full as a WhatsApp button; if you go over, the inspector warns you.
- **ID / slug**, generated automatically from the text (you can change it). Use letters, numbers, hyphens and underscores.

Buttons go out as buttons on WhatsApp and Telegram, and as a numbered list on other channels.

Our three options: **Book a visit**, **Prices** and **Talk to someone**. The answer is saved in the variable **opcion**, to use it later.

How can we help you today?

🗓️ Book a visit | 💲 Prices | 🙋 Talk to someone



Each option is an output of the node, plus Invalid and Timeout.

Each option becomes an output of the node. On top of that, Ask question always has two more outputs:

Output	When
One per option	The customer chose that option (by tapping the button or typing its text or its number).
Invalid	The answer isn't valid and the retries have run out.
Timeout	The Timeout minutes passed without a reply.

TIP

If you leave Timeout at 0, an abandoned conversation leaves the workflow waiting forever at that step. For a menu, 60 or 120 minutes is usually reasonable; connect Timeout to a message or to a person.

18

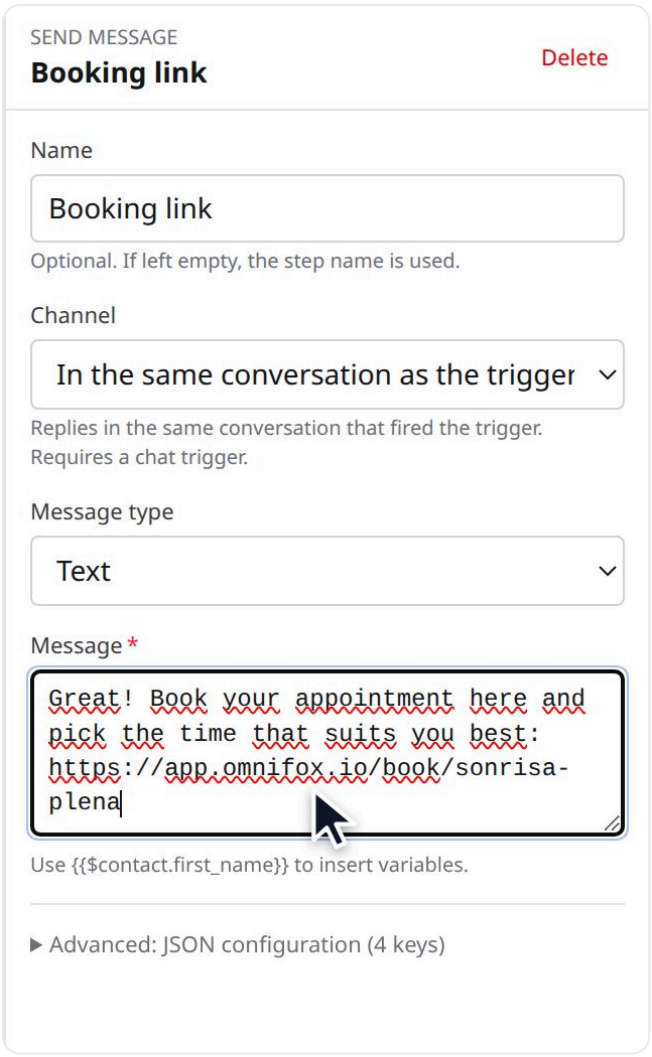
CHAPTER 18

One branch per option

Connect each button to its reply.

The question already has its three outputs; now each one leads to its reply.

Book a visit → Booking link



The message with the link to the clinic's booking page.

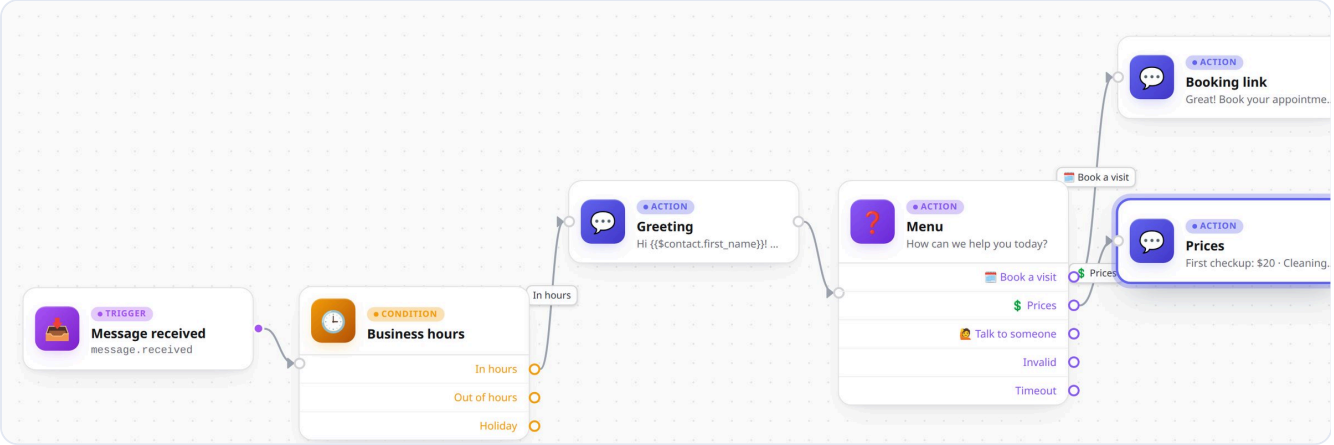
1. Add a **Send message** and connect it to the **Book a visit** output.
2. Call it **Booking link**.
3. Write the message with the link to the clinic's booking page, where the patient picks the day and time that suit them best.

Great! Book your appointment here and pick the time that suits you best: https://app.omnifox.io/book/sonrisa-plena

Prices

Another **Send message**, connected to the **Prices** output, with the prices of the checkup, the cleaning and the whitening. The dollar sign is no problem.

First checkup: $20 · Cleaning: $35 · Whitening: $180. Shall I book you a checkup?



Book a visit leads to the Booking link; Prices, to the message with the prices.

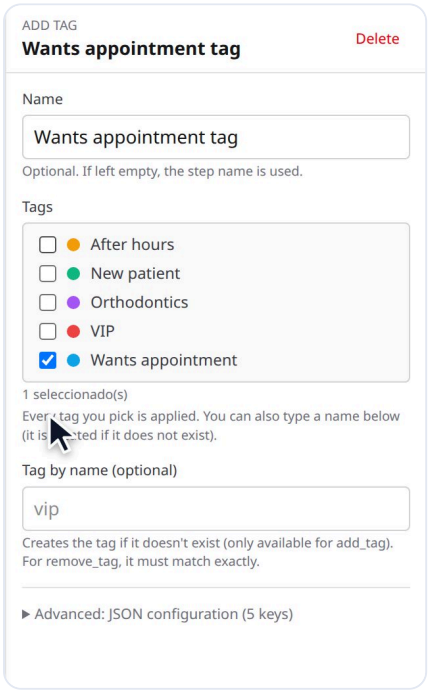
> **TIP**
>
> If a message is similar to one you already have, duplicate it with Ctrl + D and change only the text (chapter 22).

19

CHAPTER 19

Adding and removing tags

Mark the contact so your team knows what they want.



Add tag: the workspace's list of tags, with Wants appointment checked, and the field to type a new one.

After the booking link, **Add tag** gives the contact the **Wants appointment** tag:

1. Add **Add tag** and connect it after "Booking link".
2. Give it a name: "Wants appointment tag".
3. Under **Tags**, check **Wants appointment**.

That way, when the front desk opens the inbox or Contacts, they'll know who wants an appointment.

Field	What it is
Tags	The workspace's list of tags, with their colors. You can check **several**: all of them are applied.
Tag by name (optional)	Type a name: if the tag doesn't exist, it's created automatically. Supports variables.

Remove tag works the same way, in reverse: it removes the checked tags (or the one typed by name, which has to match exactly).

> **NOTE**
> Tags also trigger workflows ("Tag added") and act as a brake ("Contact has tag", chapter 32). A tag like "Do not contact" can stop all your automatic campaigns.

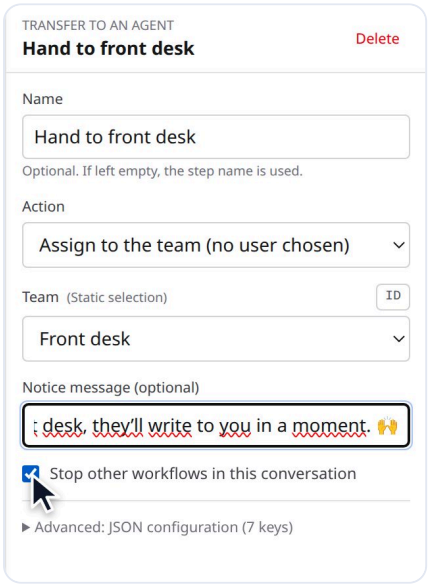
20

CHAPTER 20

Transferring to a person or a team

Hand the conversation to the front desk, and let the patient know.

The third option, **Talk to someone**, goes to **Transfer to an agent**, which we connect to that output and call "Hand to front desk".



Transfer to an agent: Assign to the team, the Front desk team, the notice to the patient and Stop other workflows.

Field	What it is
Action	**Assign to a specific user**, **Assign to a user in a specific team**, **Assign to the team (no user chosen)** or **Remove assignment**.
User / Team	Who to. We choose the **Front desk** team. You can pick from the list or use a variable.
Assignment logic	For a team: **Round-robin (rotate)**, **Fewest open contacts** or **Manual (team queue)**.
Notice message (optional)	What the patient receives, for example "I'm handing you over to the front desk, they'll write to you in a moment. 🙌".
Stop other workflows in this conversation	Checked by default: no other workflow keeps writing in the conversation.

Logic	What it does
Round-robin (rotate)	Spreads conversations evenly among team members, taking turns.
Fewest open contacts	Assigns to the member with the lightest load right now.
Manual (team queue)	Doesn't pick a person: the conversation stays in the team queue for someone to take it.

🙋 Talk to someone

I'm handing you over to the front desk, they'll write to you in a moment. 🙌

Conversation assigned to the Front desk team

21

CHAPTER 21

The after-hours branch

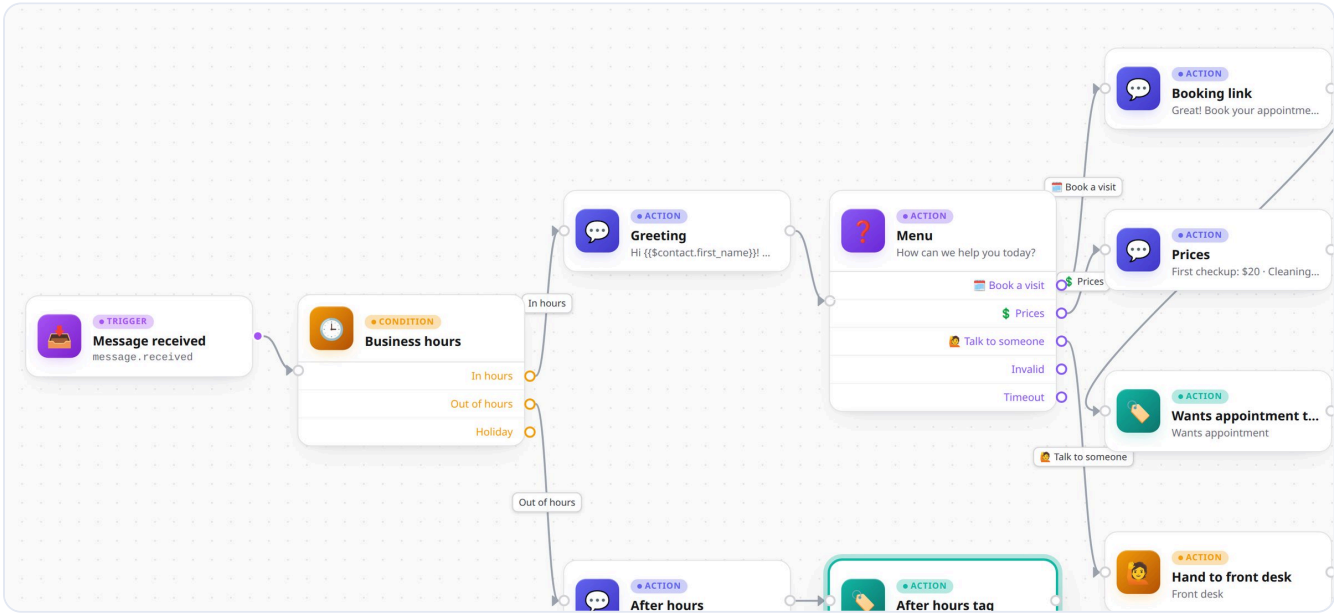
An honest reply when the clinic is closed.

The **Out of hours** branch is still missing:

1. From the **Out of hours** output of Business hours, add another **Send message** and call it "After hours".
2. Write the clinic's opening hours, with the promise to reply first thing.
3. Then **Add tag** again, with the **After hours** tag, so the front desk sees these patients first when they open in the morning.

Good evening, are you open tomorrow?

Thanks for writing! We're closed right now: we're open Monday to Friday 9 AM to 7 PM and Saturdays until 2 PM. We'll reply first thing tomorrow.



The workflow with its two branches: the daytime menu and the after-hours reply with its tag.

That way, during the day the patient gets the menu, and at night, an honest reply with the opening hours.

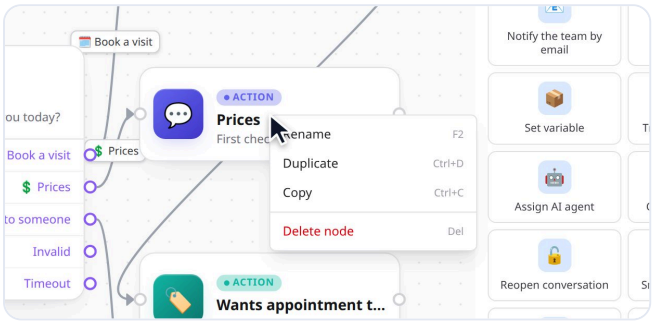
TIP

In the morning, filter the inbox or Contacts by the “After hours” tag to serve whoever wrote at night first. Once you’ve helped them, remove the tag.

CHAPTER 22

22 Move, rename, duplicate, copy and delete

Everything you can do with a node that’s already in place.



A node’s menu: Rename, Duplicate, Copy and Delete node, with their shortcuts.

Action	Menu (right-click on the node)	Shortcut
Rename	**Rename**	—
Duplicate	**Duplicate**	Ctrl + D
Copy and paste	**Copy** and, on the canvas, **Paste**	Ctrl + C Ctrl + V
Delete	**Delete node**	Del

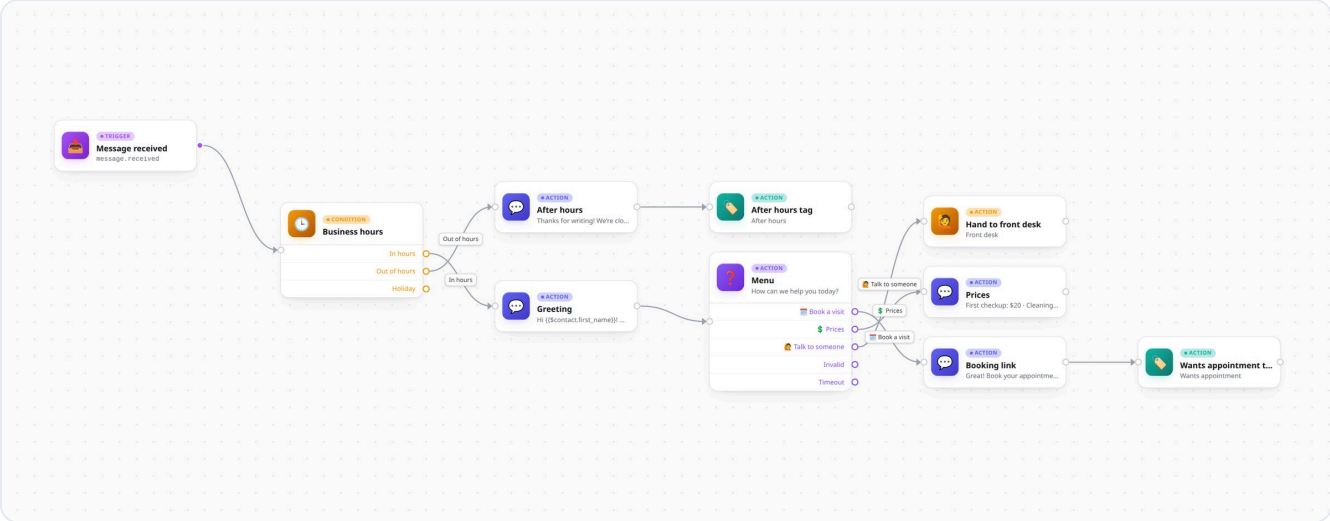
- **Move**: drag the node; its connections follow it.
- **Select several**: holding Shift, drag a box over the canvas, or use **Select all** (Ctrl + A). With several selected, the menu offers Duplicate, Copy or Delete the N nodes.
- **Copy between workflows**: copy in one and paste in another; for entire workflows, use Import/Export (chapter 51).
- Each node’s inspector also has the **Delete** button, at the top right.

CHAPTER 23

23 Undo, redo and auto-arrange

Making mistakes costs nothing.

- **Undo**: Ctrl + Z or the arrow on the bar.
- **Redo**: Ctrl + Y (or Ctrl + Shift + Z) or the other arrow.
- **Auto-arrange** lays out the whole workflow from left to right, with tidy branches, and fits it on the screen. If you don’t like how it turned out, undo it.



After Auto-arrange: everything from left to right, with tidy branches.

WATCH OUT

Undo works within the editor session. Once you save and leave, there's no version history for the workflow: before a big change, duplicate the workflow or export it as a backup.

24

CHAPTER 24

Validate, save and turn on

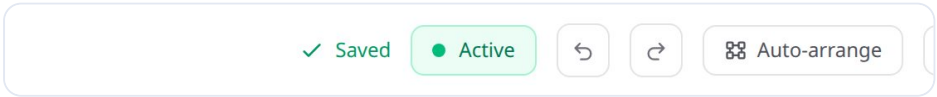
Nothing fires until you save it as active.

Validate

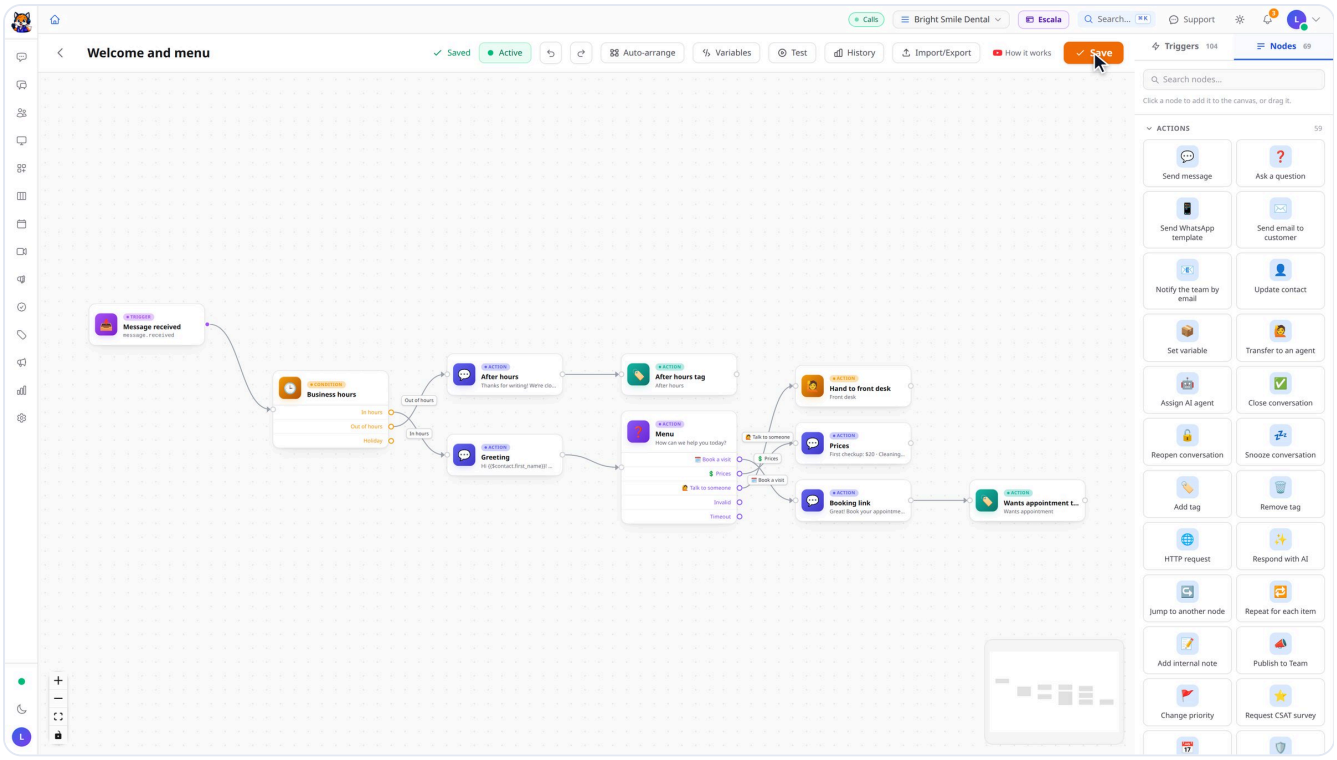
While something required is missing, the node is marked in red with what it lacks (for example, "Still missing: message") and the main button says **Fix errors**: you can't save. Once everything is filled in, it says **Save** again.

Save and turn on

1. Press **Save**. The status changes to **Saved**. The workflow is still **inactive**.
2. Click the **Inactive** pill: it changes to **Active**.
3. Press **Save** again. From now on, the workflow works on its own.



Saved and Active: the workflow now works on its own.



Welcome and menu, saved and active.

- **There's no autosave**: if you close or press Back with pending changes, the editor warns you.
- **A workflow with errors can't be turned on**: if you try, it's saved as an inactive draft and Omnifox tells you what to fix.
- You can't save while a test is running (chapter 26).

> **TIP**
>
> Before turning on a workflow that writes to customers, test it with your own number or with a test conversation (chapter 25).

25

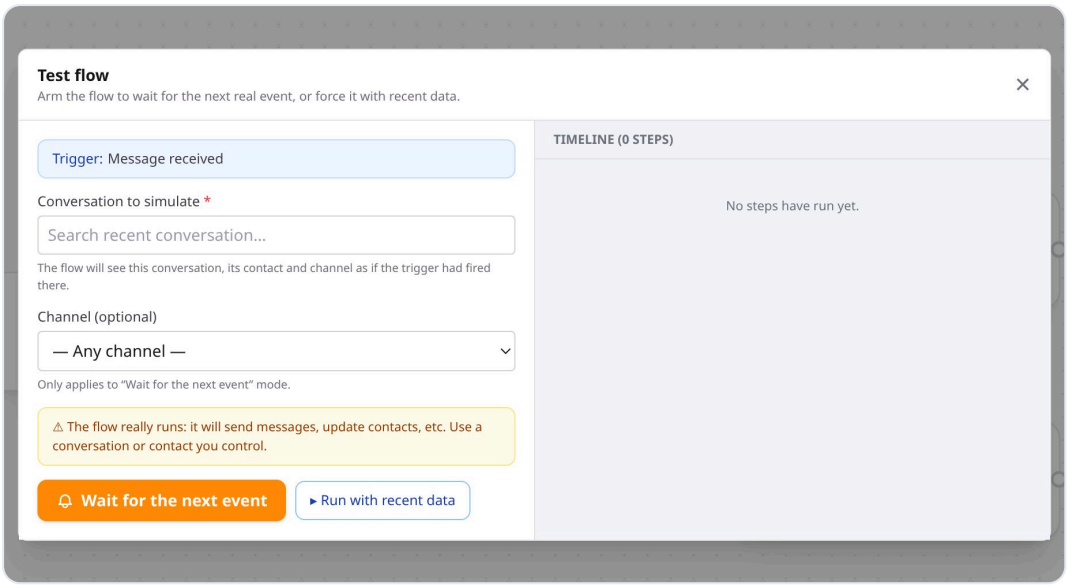
CHAPTER 25

Testing the workflow

Before you let it work, you test it.

Test opens the **Test flow** window with two modes:

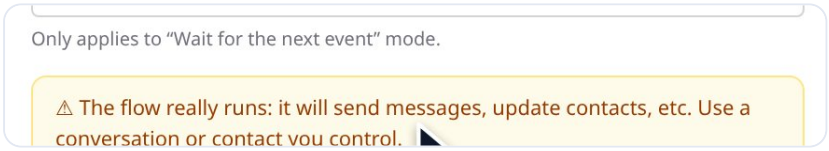
Mode	What it does	When to use it
Wait for the next event	Leaves the workflow "armed", watching for the next real event that comes in (for example, the next message you write from your phone).	To test the full journey, the way the customer will experience it.
Run with recent data	Runs it right now on the conversation or contact you choose.	For a quick test, without waiting.



Test flow: Wait for the next event or Run with recent data, on the conversation you choose.

Depending on the trigger, you choose what to simulate:

- **Conversation to simulate** ("Search recent conversation..."): the workflow will see that conversation, its contact and its channel as if the event had happened there.
- **Contact to simulate**: for contact triggers, such as Contact created.
- **Deal to simulate**: for CRM triggers.
- **Trigger data**: triggers without a real-time event (scheduled, webhook, manual) are tested with data you type in; you can attach an optional conversation, contact and channel.



The notice: the test is real, it sends messages and really changes data.

> **WATCH OUT**
> **The test is real**: it sends the messages and really changes contacts. Use your own number or a test conversation, such as a web chat one you open yourself.

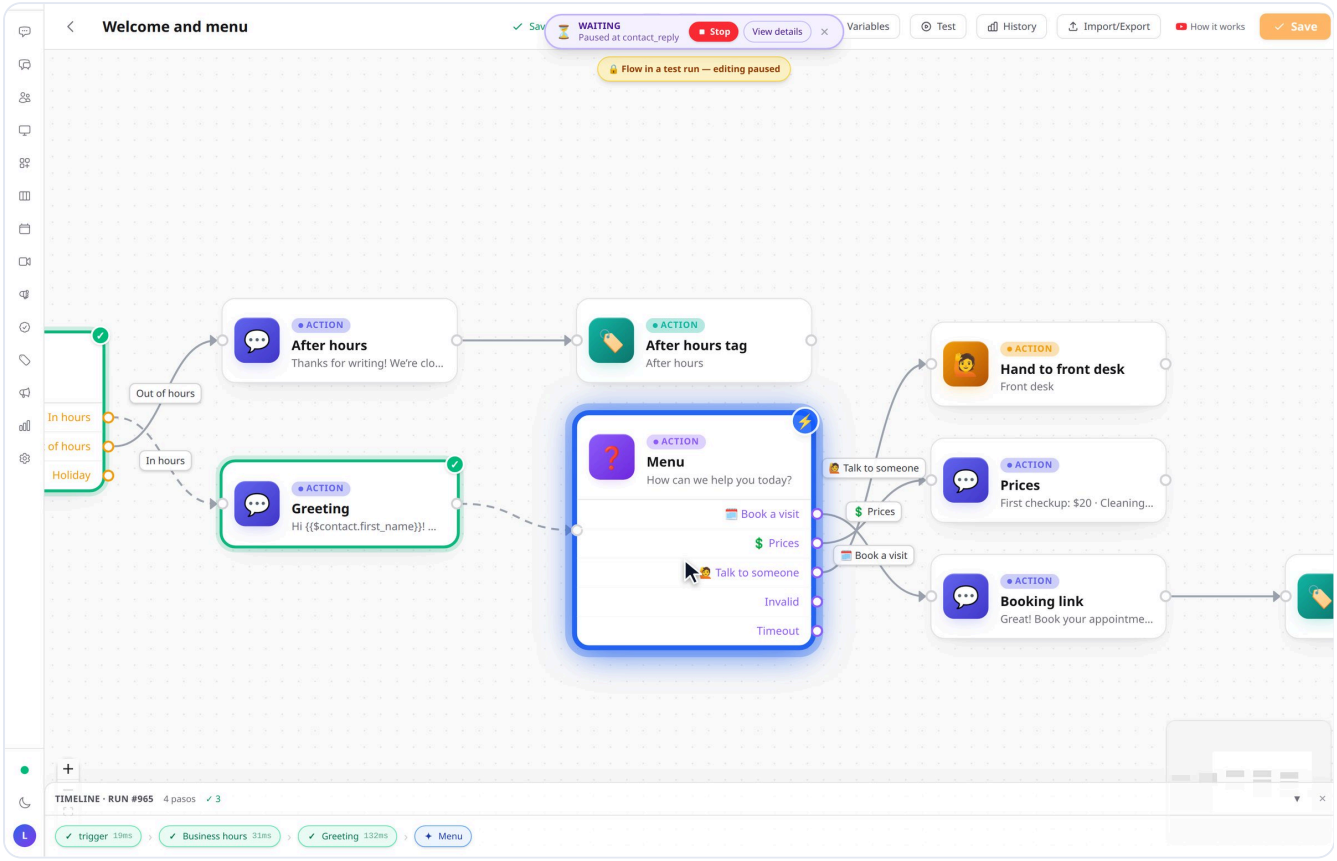
While the test lasts, the normal run of that same workflow on that same conversation is paused, so no duplicate messages go out.

26

CHAPTER 26

The live run

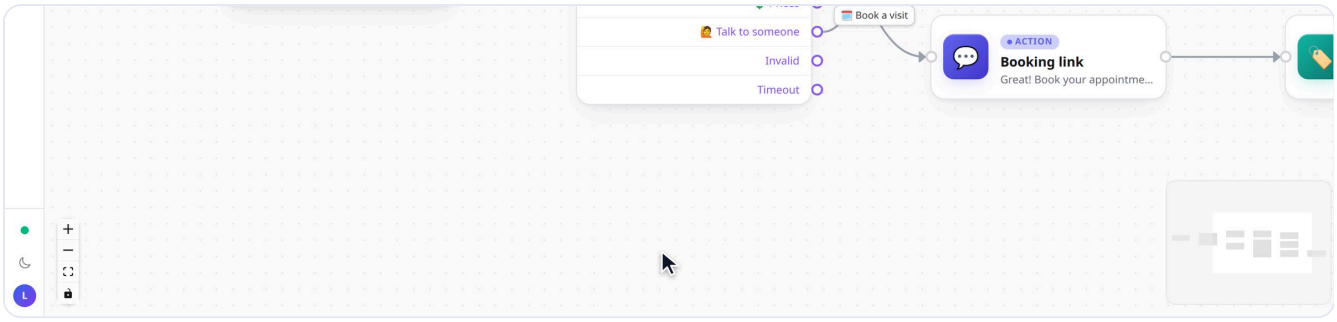
Watch the workflow work, step by step.



The live run: the editor locked and each node lighting up as it runs.

- The editor **locks** so you don't change anything halfway through: "Flow in a test run — editing paused".
- Each node **lights up** as it runs ("Running now") and stays marked when it finishes ("Node executed").
- At the top, a **pill** shows the status, with the **Stop**, **View details** and **Close** buttons.
- At the bottom, the **Timeline** with each step; **Center #...** moves the view to that node.

Status	What it means
🟡 ARMED · Waiting for the next event	"Wait for the next event" mode: it hasn't arrived yet.
🟢 RUNNING / Flow running	It's going through the nodes.
⏳ WAITING / Paused at ...	It stopped at a question or a wait.
✓ COMPLETED / Run finished	It reached the end, with its duration.
✕ FAILED / Run with errors	A step failed; the details say which one.
🛑 CANCELLED	You stopped it, or it stopped (for example, because an agent replied).



The timeline with each step; here, paused at the question.

In the example, the workflow is **paused at the question**, waiting for the patient to choose an option. You can close the details: the pill and the timeline keep showing the progress live.

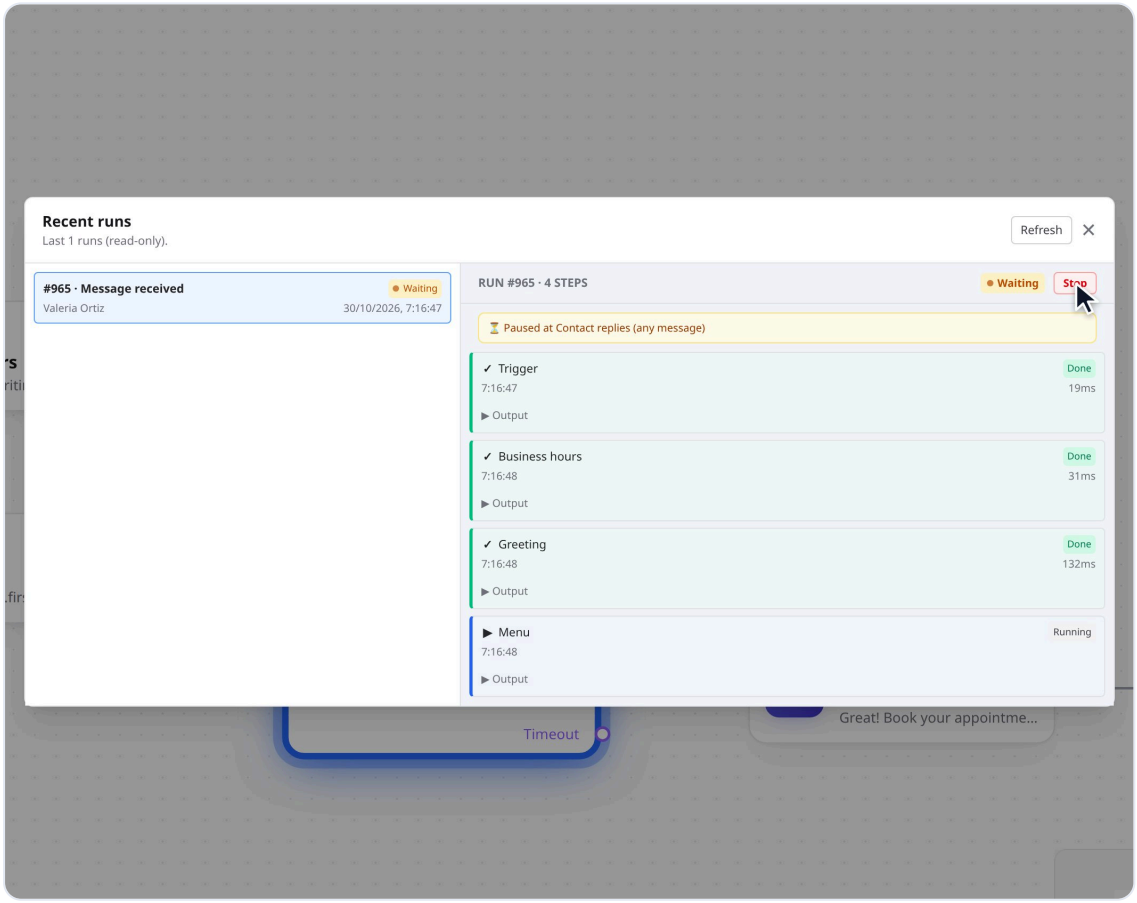
27

CHAPTER 27

The run history

What the workflow did, each time, and where it failed.

History opens **Recent runs**: the workflow's latest runs (test and real), in read-only mode, with the **Refresh** button.



Recent runs: the list on the left and the steps of the selected run on the right.

Part	What it shows
The list (left)	Each run with its status (finished ✓, failed ✕, cancelled, waiting or running), the date and "Duration: N s".
The details (right)	"Run #... · N steps" and each step in order, with its result: ✓ completed, ▶ entered, ✕ failed, – skipped.
Error	If something failed, the error appears in red, at the exact step.
Paused at	If the run is waiting (for an answer, for some time), at which node and for what.
Stop	Stops a run that's running or waiting.
Cancellation reason	Why it stopped: an agent replied or took the conversation, it was stopped by hand, the test expired...

From the workflow list, the **View run #...** link on a card with failures opens this window directly with that run.

Filtering the history

Filter	Options
Status	**All**, **Failed**, **Finished OK**, **Running or waiting** and **Cancelled**.
Workflow	**This workflow** or **All workflows** in the workspace. With all, each run carries its workflow's name and the **Open its workflow** button.
From / **To**	A date range, in the workspace's time zone.
Clear filters	Back to the full view. If nothing matches: "No executions match these filters.".

Retrying a failed run

A run that **failed** shows the **Retry** button. First fix the cause (the red step tells you what it was) and click Retry: Omnifox starts a **new execution** of the whole workflow, with the same trigger data and the workflow's **current version** ("Retry started: execution #...").

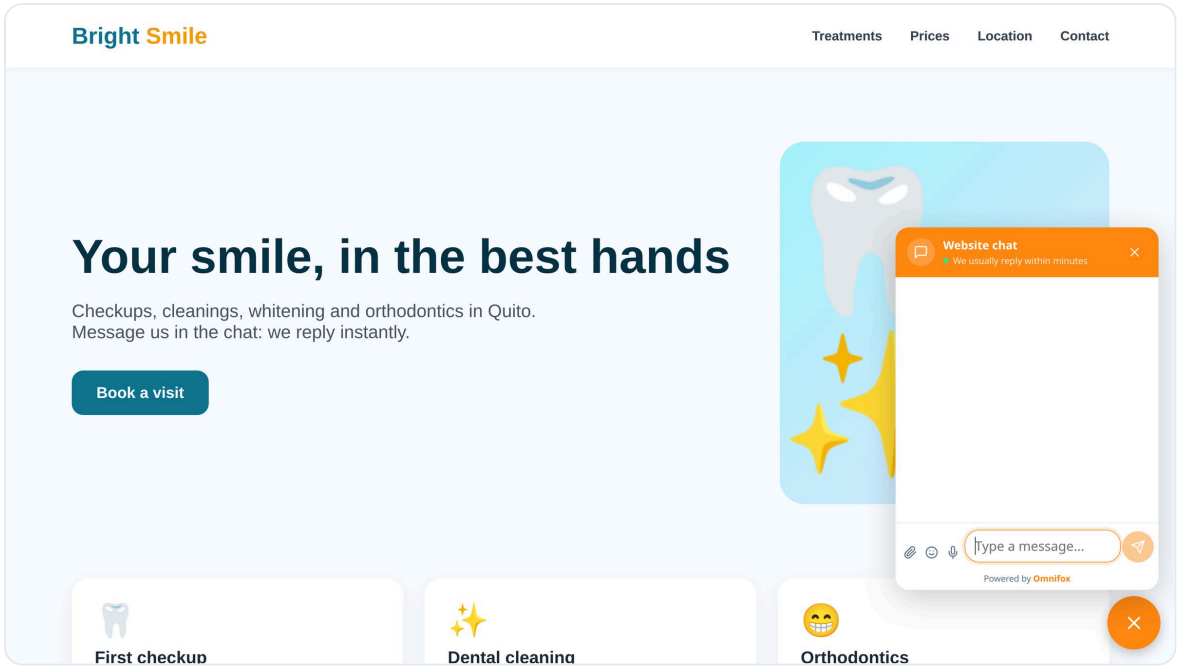
> **WATCH OUT**
>
> Retry runs the workflow from the start: steps that had already succeeded (for example, messages sent) are done again. That's why Omnifox asks you to confirm first. Only failed runs can be retried, and you need permission to manage workflows.

28

CHAPTER 28

In real life: a patient writes from the website

The workflow at work, without anyone touching anything.



The clinic's website with the Omnifox chat.

This is the clinic's website with the Omnifox chat. A patient writes "Hi". Instantly:

Hi

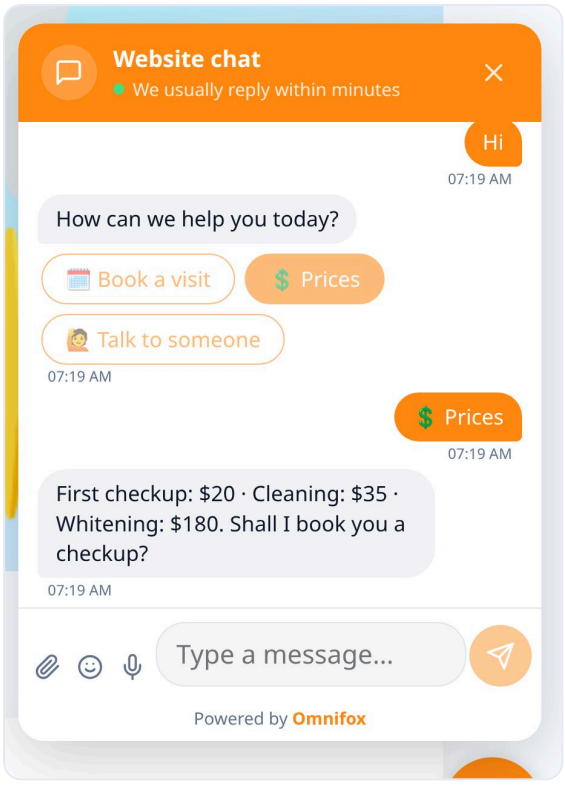
Hi `{{$contact.first_name}}` ! 👋 Thanks for writing to Bright Smile.

How can we help you today?

🗓️ Book a visit | 💲 Prices | 💁 Talk to someone

💲 Prices

First checkup: $20 · Cleaning: $35 · Whitening: $180. Shall I book you a checkup?



The patient writes “Hi” and gets the welcome and the menu; they choose Prices.

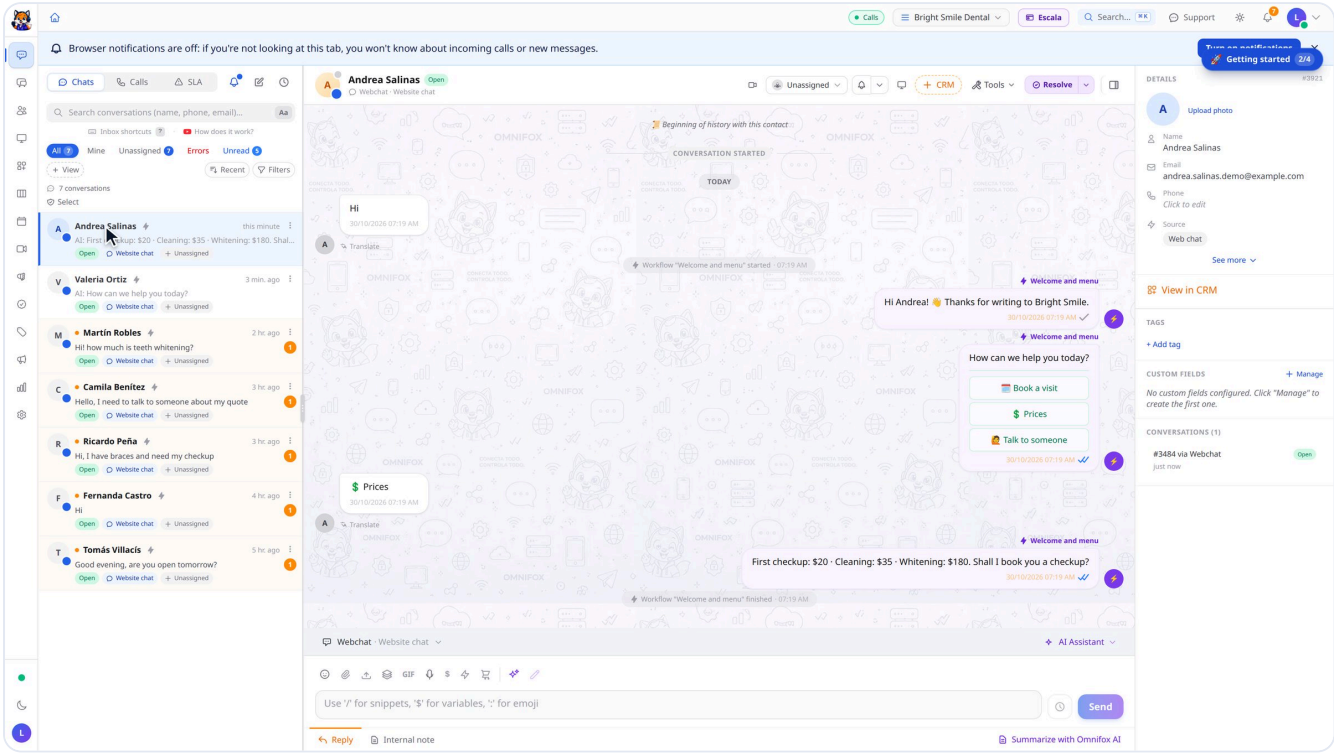
They get the welcome with their name and the menu; they choose **Prices** and the workflow replies with the list. Nobody at the clinic touched a thing.

29

CHAPTER 29

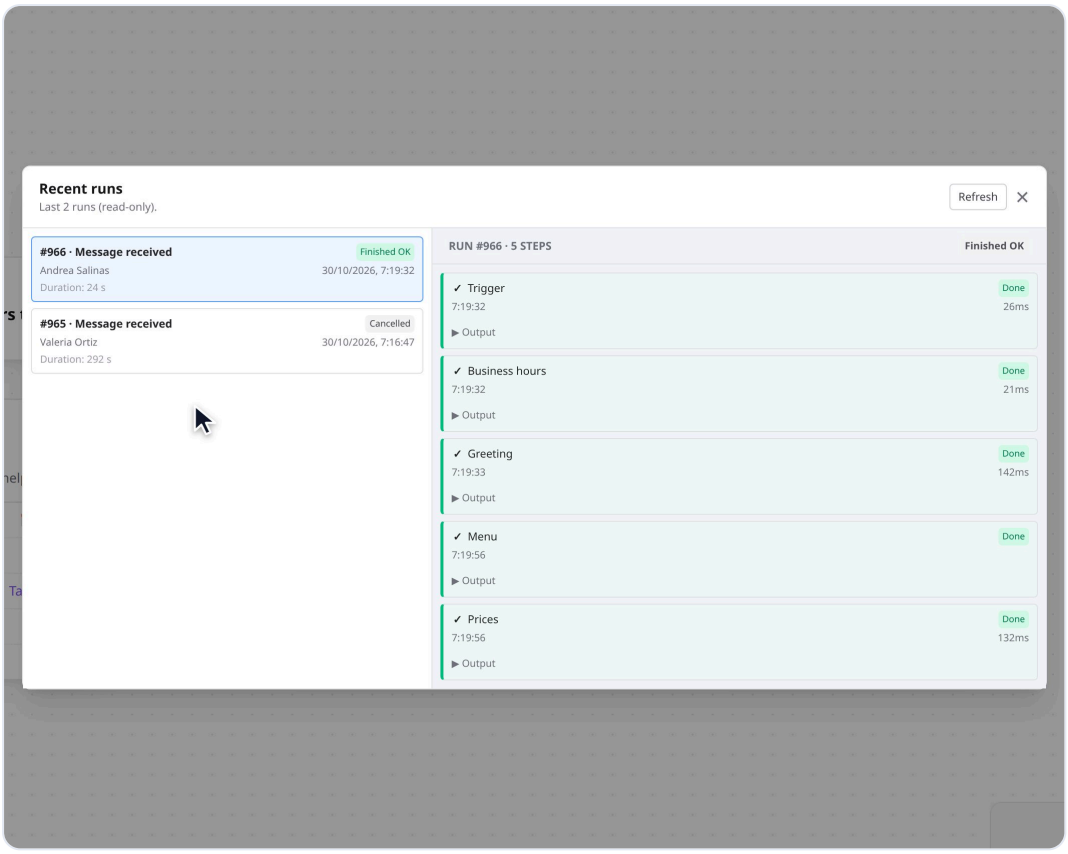
The workflow in the inbox

What your team sees while a workflow is active.



The same conversation in the inbox, with the active workflow notice and the button to stop it.

- The conversation shows **everything the workflow sent**, just like a person's messages.
- While a workflow is active in the conversation (for example, waiting for the menu answer), a **notice** appears at the top with the workflow's name and the button to **stop it**, if a person wants to take over.
- If the workflow has "Stop this workflow when the agent replies" turned on (chapter 11), just writing is enough: it steps back on its own.
- The workflow's **History** keeps the full run, step by step.



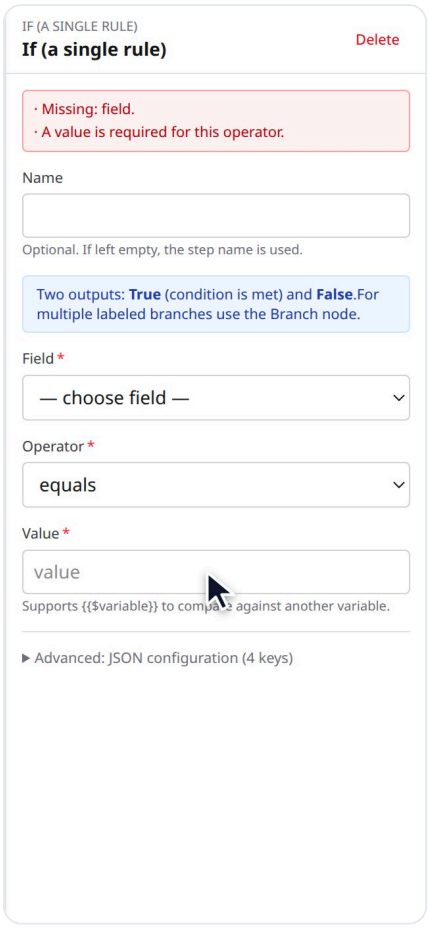
The workflow's History keeps the full run, step by step.

30

CHAPTER 30

If condition: a single rule

Two paths: it matches or it doesn't.



If (a single rule): Field, Operator and Value; Yes and No outputs.

If (a single rule) has two outputs: **Yes** and **No**. It's built with three fields:

- **Field**: what to look at. They're grouped: Conversation (status, priority, assignee, team, channel, channel type, unread), Contact (name, email, phone, company, language...), Message (text, type), AI, Answer (to a question), Loop, Error, Workspace, User, Team, AI handoff and Now (time, day of the week, weekend, within business hours). Under **Advanced → Custom variable...** you can type any variable.
- **Operator**: equals, does not equal, contains, does not contain, starts with, ends with, greater than, less than, is empty, is not empty.
- **Value**: what to compare against. Supports {{$variable}} to compare two pieces of data with each other.

Example	Field	Operator	Value
Did they write on WhatsApp?	Channel type	equals	whatsapp
Did they ask for a price?	Message text	contains	price
Is it the weekend?	Now · weekend	equals	true
Did they leave their email?	Contact · email	is not empty	—

TIP

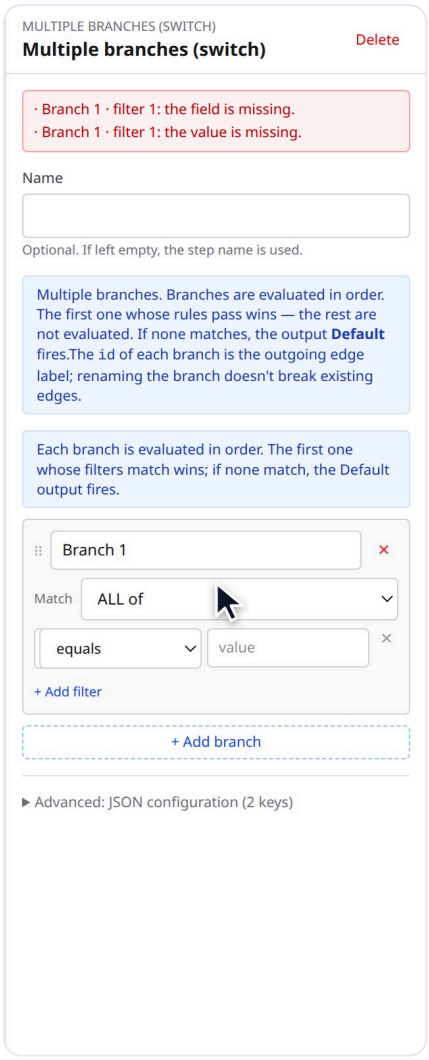
For more than two paths, don't chain lots of Ifs: use Multiple branches (chapter 31).

31

CHAPTER 31

Multiple branches

Like several If conditions together, in order.



Multiple branches (switch): each branch with its name and its rules, plus the default branch.

Multiple branches (switch) has a list of branches. Each branch has:

- a **name**, which is the name of its output on the canvas (renaming it doesn't break its connections);
- its **rules** (field, operator, value, as in If), joined with **AND** (all) or **OR** (any).

Branches are evaluated **in order** and **the first one that matches wins**: the rest aren't even checked. If none matches, the workflow continues along the **default** branch.

Each branch is an output you connect to whatever you want.

Example: based on **Channel type**, one branch "WhatsApp", another "Web chat" and the default branch for everything else.

WATCH OUT

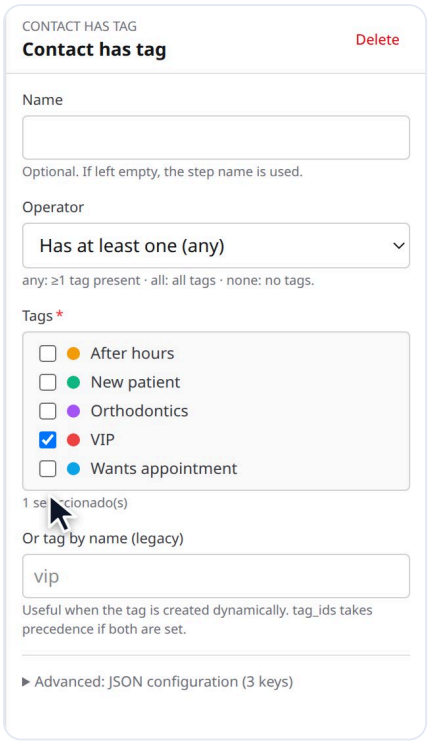
Put the most specific branches first. If the first branch is very broad, it takes every case and the ones below are never used.

32

CHAPTER 32

Contact has tag

Ask about the contact's tags.



Contact has tag: any, all or none of the chosen tags.

Contact has tag looks at the tags you choose and continues along **Yes** or **No**, depending on the **Operator**:

- **any**: has at least one of the tags;
- **all**: has all of them;
- **none**: has none of them.

You pick the **Tags** from the list; you can also type one by name.

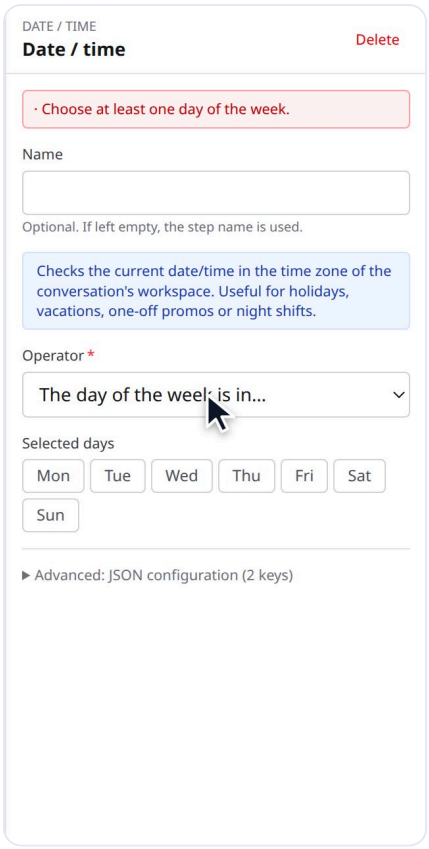
> **TIP**
>
> A trick: with "none" it works as a brake. If the patient has the "Do not contact" tag, the workflow doesn't continue. Put it at the start of your follow-ups and automatic campaigns.

33

CHAPTER 33

Date and time

Reply differently depending on the day, the month or the time.



Date / time: the operator and its values, in the workspace's time zone.

Operator	What it compares
Exact date	Today is that day.
Before / After	Before or after a date.
Between dates	Between two dates (From / To).
Days of the week	Certain days (Monday to Sunday).
Days of the month	Numbers separated by commas, e.g. 1, 15.
Months	Certain months of the year.
Time range	Between two times. If "To" is earlier than "From", it crosses midnight (22:00 → 06:00).

Date / time compares the current moment, **in the workspace's time zone**, and continues along Yes or No. For example, to reply differently during a December promotion, or only on Saturdays.

> **NOTE**
> For your everyday opening hours use Business hours (chapter 14), which already knows your schedule and your holidays. Date / time is for exceptions.

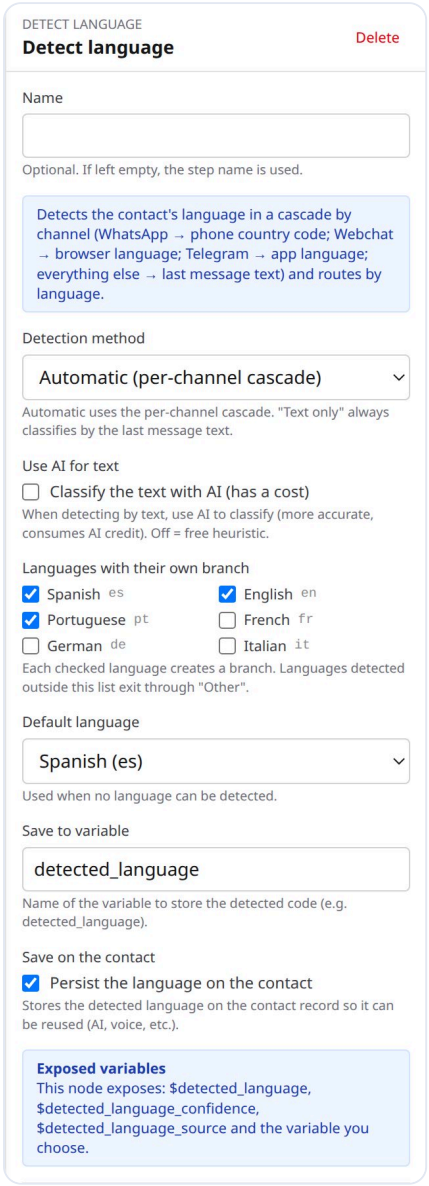
34

CHAPTER 34

Detect language and agent in team

Branches by language and by the agent's team.

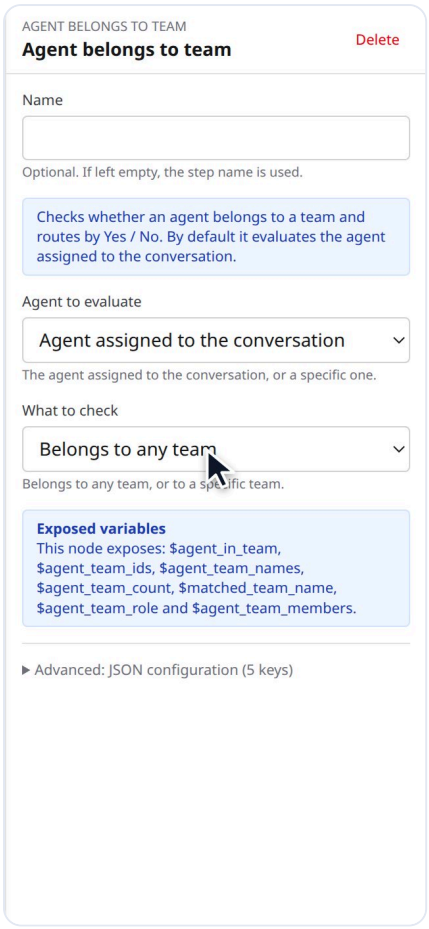
Detect language



Detect language: method, languages with their own branch and default language.

Field	What it is
Detection method	**Automatic (per-channel cascade)**: on WhatsApp, by country code; on web chat, by browser language; otherwise, by the text. **By message text only**: always by the text.
Use AI for text	Classifies the text with AI: more accurate, but uses AI credit.
Languages with their own branch	Each checked language (Spanish, English, Portuguese, French, German, Italian) is a branch; the rest go out through "Other".
Default language	If it can't detect any.
Save to variable · Save on the contact	Saves the code (es, en, pt...) in a variable and, if you want, on the contact's profile, to reuse it (AI, voice, etc.).

Agent belongs to team



Agent belongs to team: which agent to evaluate and what to check.

Checks whether an agent is on a team and continues along **Yes** or **No**:

- **Agent to evaluate**: the **Agent assigned to the conversation**, or **A specific agent**.
- **What to check**: whether they **Belong to any team** or to **a specific team**.

Handy so you don't reassign someone who's already on the right team: if the assigned agent is already on Front desk, there's no need to transfer.

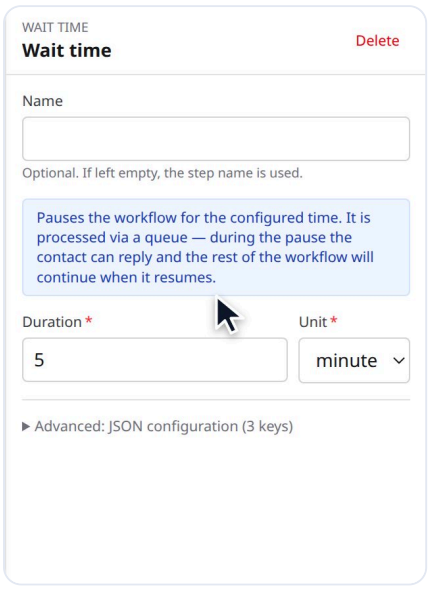
35

CHAPTER 35

Wait time and wait for an event

Pause the workflow for a while, or until something happens.

Wait time



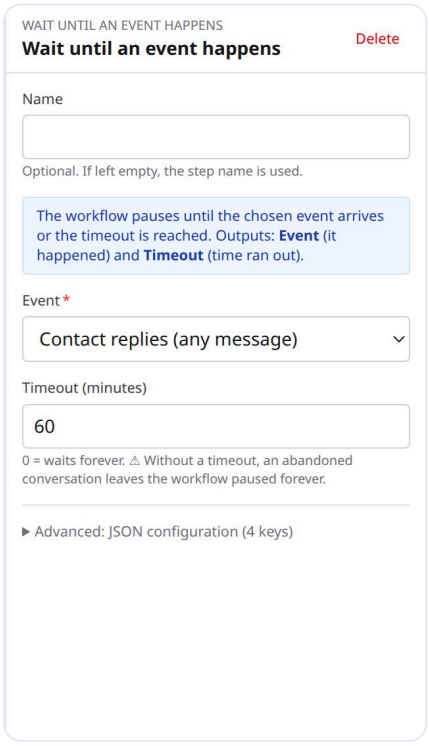
Wait time: amount and unit.

Pauses the workflow for some **minutes**, **hours** or **days** and then continues. During the pause, the contact can reply and the conversation carries on as normal.

For example, one day after sending the booking link, ask whether they managed to book:

> Were you able to book your appointment, `{{$contact.first_name}}`? If I can help with anything, I'm here. 😊

Wait until an event happens



Wait until an event happens: the event and the time limit.

Field	What it is
Event	The customer replies, the conversation is assigned (to an agent), closed, resolved or gets a tag.
Timeout (minutes)	The longest it will wait. 0 = forever.
Outputs	**Event** (the event arrived) and **Timeout** (time ran out).

WATCH OUT

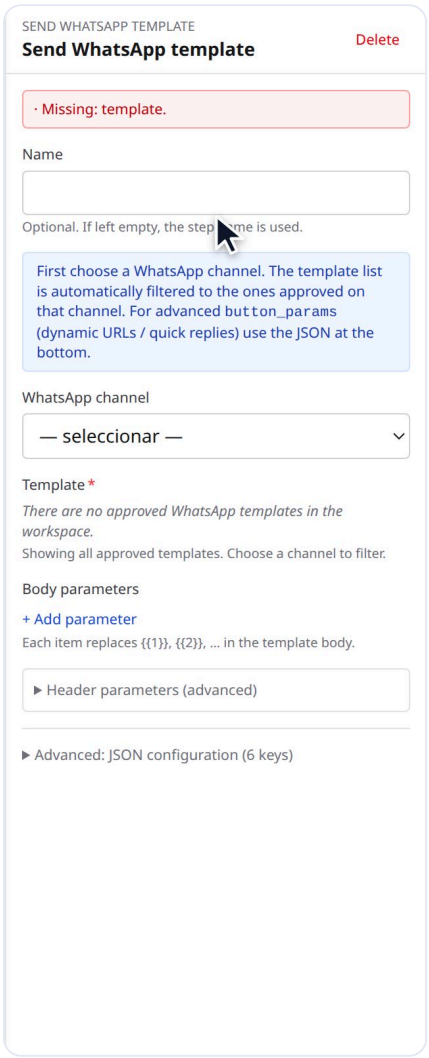
Always set a time limit on event waits: without one, an abandoned conversation leaves the workflow waiting forever.

CHAPTER 36

36 WhatsApp templates and the 24-hour window

Writing outside the 24-hour window.

On WhatsApp you can only write freely **within 24 hours of the customer's last message**. Outside that window you need a **Meta-approved template**. An appointment reminder, which goes out a day before, almost always falls outside the window.



Send WhatsApp template: channel, approved template and body parameters.

Field	What it is
WhatsApp channel	Pick the channel first: the template list is filtered to the ones approved on that channel.
Template *	The approved template. If there are none, create one in WhatsApp Templates.
Body parameters	Each row replaces {{1}}, {{2}}... in the body, in order, with a value or a variable such as `{{$contact.first_name}}`.
Header params	Only if the template has a header with a variable.

It's what Bright Smile's "WhatsApp appointment reminder" workflow uses:

Hi `{{$contact.first_name}}` 👋 A reminder of your `{{$appointment.service_name}}` appointment tomorrow at `{{$appointment.time}}` at Bright Smile. Reply YES to confirm.

TIP

Create templates designed for their purpose (a reminder says it's a reminder). Utility templates get approved faster and cost less than marketing ones.

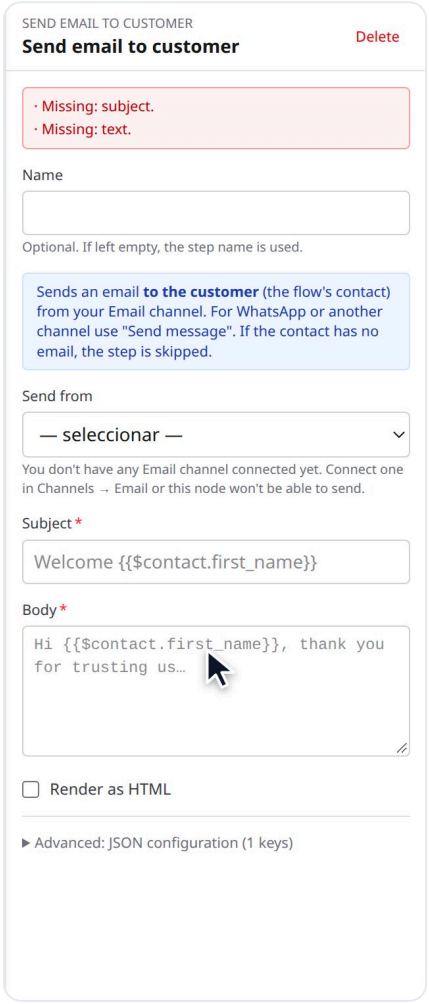
37

CHAPTER 37

Emails: to the customer and to the team

Two email nodes: one for the customer and one internal.

Send email to customer

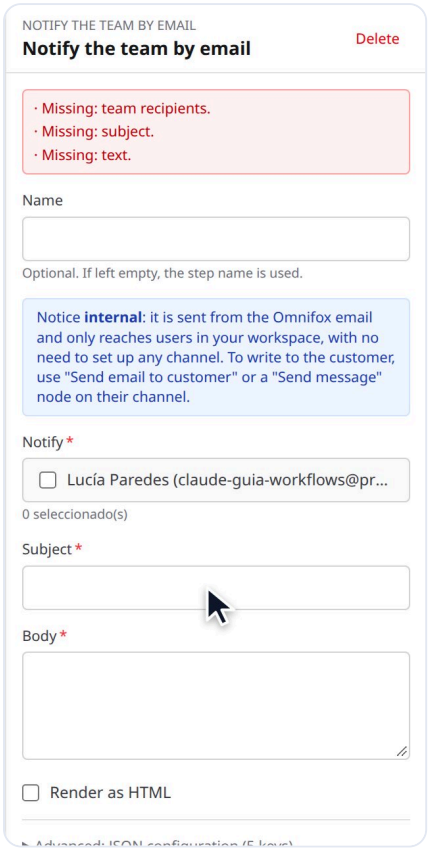


Send email to customer: which Email channel it's sent from, subject and body.

Field	What it is
Send from	Your Email channel. Empty = the conversation's one if there is one, otherwise the first active Email channel.
Subject · Body	With variables, for example "Welcome `{{$contact.first_name}}`".
Render as HTML	For a formatted email.

It writes **to the workflow's contact** from your Email channel. If the contact has no email address, the step is skipped. For WhatsApp or another channel, use Send message.

Notify the team by email



Notify the team by email: who to, subject and body.

Field	What it is
Notify	The people in your workspace who will get the notice (required).
Subject · Body	The notice, with variables.
Render as HTML	Optional.

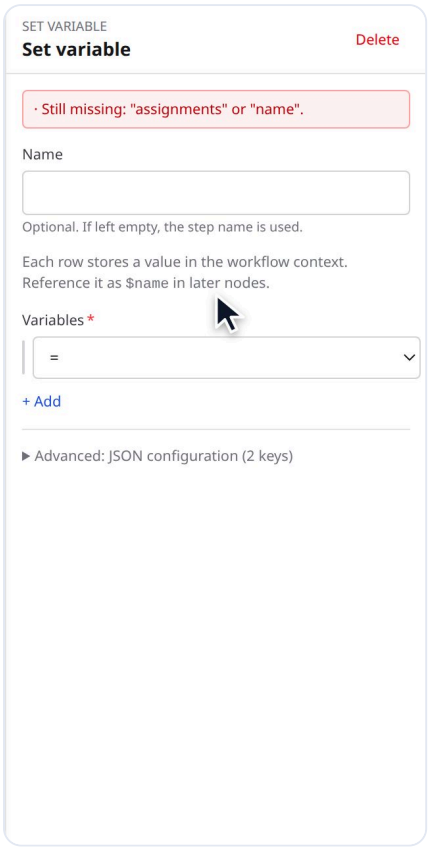
It's an **internal** notice: it goes out from Omnifox's email and only reaches users in your workspace, without setting up any channel. For example, "VIP patient writing in: `{{$contact.display_name}}`".

38

CHAPTER 38

Contacts and variables

Write to the profile, store values and find the contact.



Update contact: each row writes a profile field.

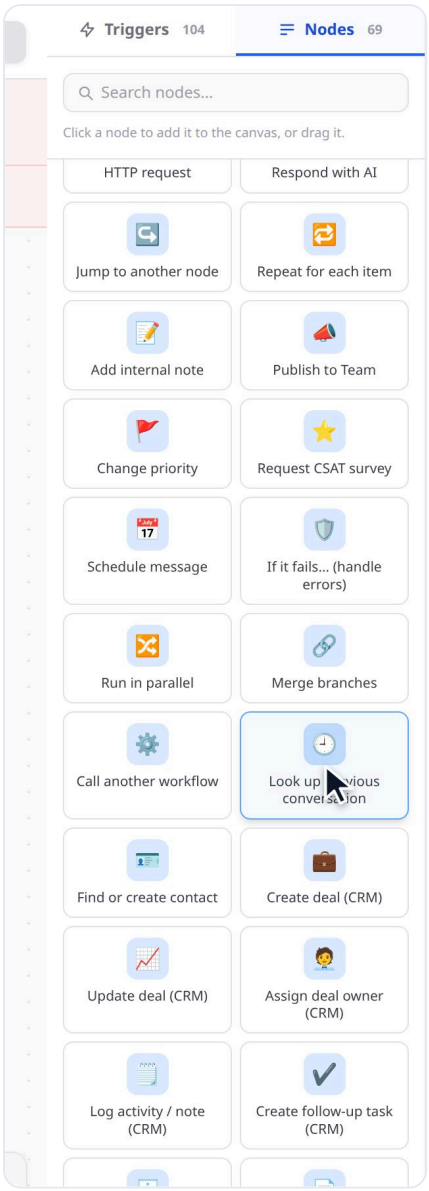
Update contact

Each row under **Assignments** writes a profile field (first name, last name, display name, email, phone, company, notes, language or a custom field) with a fixed value or a variable, even with the answer the patient gave ({{$opcion}}).

Set variable

Each row stores a value in the workflow, to use it later as {{$nombre}}. Supports other variables in the value.

Find or create contact



Find or create contact: by phone or email, with the data to create it.

Field	What it is
Look up by	**Phone** or **Email**. The phone number is normalized: 0960... and +593960... are the same.
Identifier	The data, usually a variable, for example `{{$webhook.payload.cliente.telefono}}`.
First name, Last name, Display name, Company (only when creating)	Only used if the contact didn't exist; they never overwrite data.
Open a conversation	Opens (or reuses) a conversation with the contact, on the channel you choose or the automatic one.
Outputs	**Created**, **Found** or **Not identified**. Leaves `{{$contact_id}}` and `{{$conversation_id}}`.

Look up prior conversation

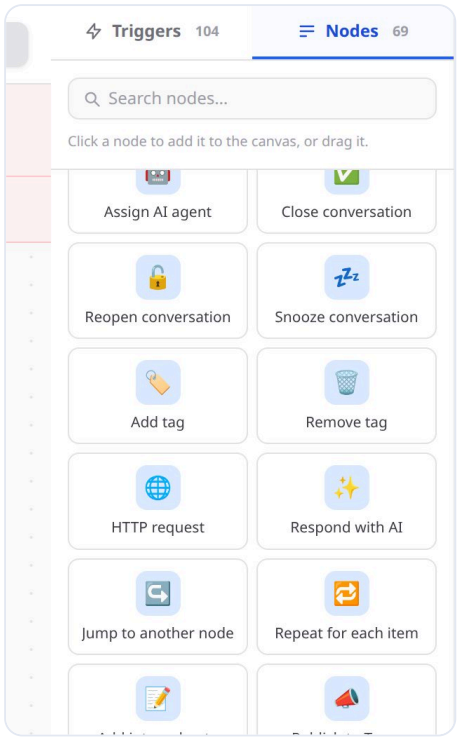
Finds the contact's most recent conversation that matches the filters (**Statuses to search**, **Filter by team**, **Max age (hours)**, **Exclude the conversation that triggered the workflow**) and continues along **Found, with agent**, **Found, no agent** or **Not found**. It leaves variables such as `{{$last_agent_id}}`: with them you can **send the patient back to the same person** who helped them last time.

39

CHAPTER 39

Conversation actions

Close, reopen, snooze, prioritize, take notes and send surveys.



An action on the conversation.

Node	What it does and what it asks for
Close conversation	**Resolution**: close it or mark it as resolved.
Reopen conversation	Sets it back to open.
Snooze conversation	Takes it out of the open inbox for the duration you set; when it's up, it comes back on its own.
Set priority	No priority, low, medium, high or urgent, on the conversation where the workflow is running.
Add internal note	A note with variables that **only your team sees**; it's never sent to the contact.
Request CSAT survey	Sends the workspace's satisfaction survey (there's one per workspace). It's skipped if it was already sent in that conversation. On WhatsApp, Instagram, Messenger and TikTok it only goes out within the 24 hours.
Schedule message	Leaves a message ready to be sent later: at an exact time or after a certain amount of time. Variables are resolved when it's scheduled.

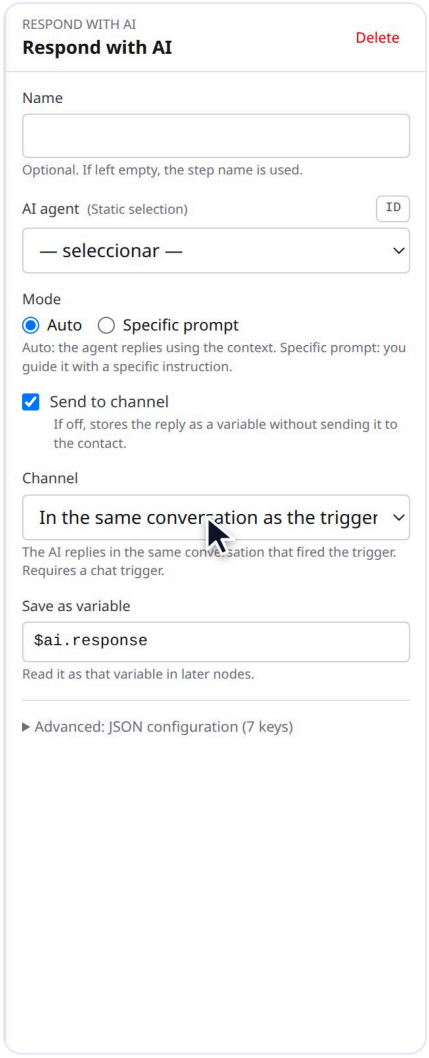
For example, Bright Smile's "Survey when the conversation closes" workflow: trigger **Conversation closed**, a one-hour wait and **Request CSAT survey**.

CHAPTER 40

40 Respond with AI and AI classifier

AI inside the workflow: to reply or to decide the path.

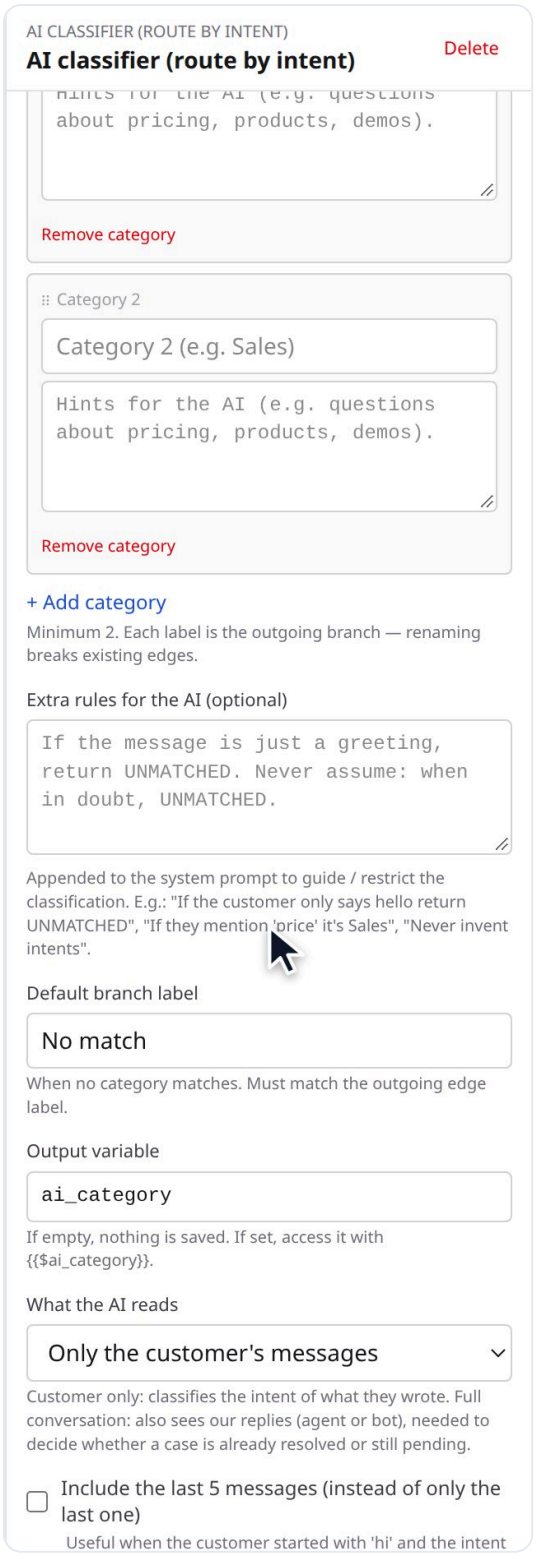
AI respond



AI respond: the agent, the mode and whether it's sent or stored.

Field	What it is
AI agent	Which of your AI agents replies.
Mode	**Auto**: replies using the conversation's context. **Specific prompt**: you guide it with an instruction, for example "Apologize for the delay and ask for the order number".
Send to channel	On, the reply is sent. Off, it's only stored in a variable.
Save as variable	To use the reply in the next steps (`{{$ai.response}}`).

AI classifier (route by intent)



AI classifier: the categories, the extra rules and the output variable.

It reads the customer's message and picks **one** of your categories. It doesn't talk to the customer: it only classifies and moves on.

- **Categories**: at least 2, each with its name (which is its output) and hints for the AI, for example **Booking**, **Prices**, **Complaint**.
- **Extra rules for the AI (optional)**: "If the message is just a greeting, don't pick any".
- **Default branch label**: the output when none fits ("No match").
- **Output variable**: stores the category in {{$ai_category}}.
- **What the AI reads**: only the customer's messages or the whole conversation, and whether it includes the last 5 messages (useful when the customer opens with "hi").

WATCH OUT

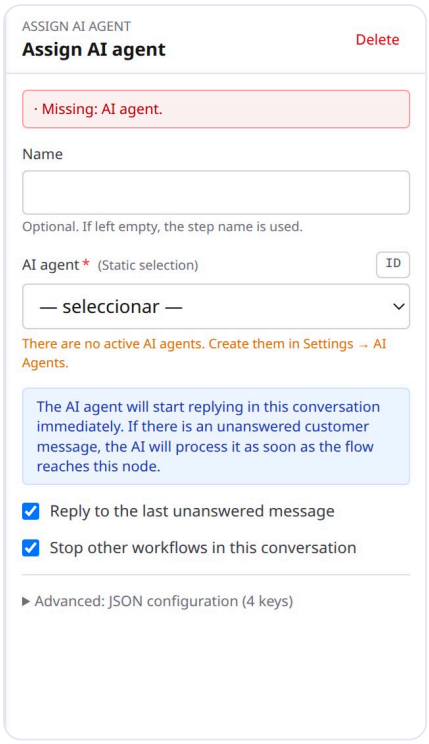
These nodes use AI credit (the classifier, very little per message). If the credit runs out, the classifier continues along the default branch.

41

CHAPTER 41

AI agents and workflows, together

A workflow that hands off to AI, and an AI that launches workflows.



Assign AI agent: the agent and Reply to the last unanswered message.

Assign AI agent

Hands the conversation to one of your AI agents, which **starts replying right away**. With **Reply to the last unanswered message**, the AI answers what the customer had already written. It also has **Stop other workflows in this conversation**.

And the other way around

- An AI agent can **launch a workflow** with its **Trigger workflows** tool: for example, the “log order” workflow when the customer confirms. The workflow uses the **Manual (called by another workflow or the AI)** trigger.
- The **The AI handed off the conversation** trigger fires when the AI passes a case to a person (with the handoff reason as a filter and as a variable): perfect for alerting the team or leaving a note.

TIP

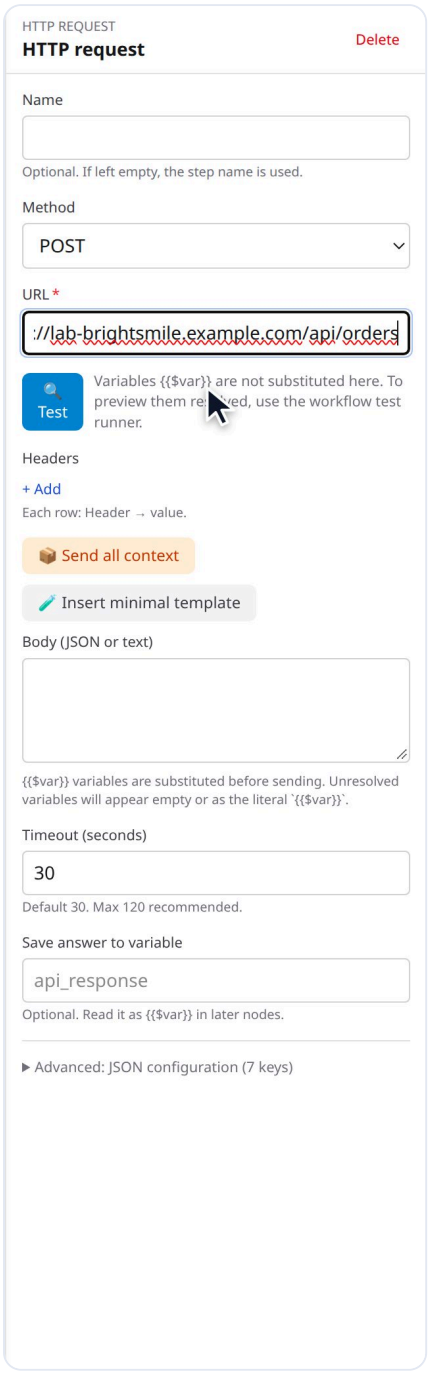
Split the work: the AI converses and understands; the workflow does the steps that always have to be the same (tagging, creating the deal, alerting).

42

CHAPTER 42

HTTP request

Connect Omnifox with any other system.



HTTP request: method, URL, headers, body, timeout and the Test button.

Field	What it is
Method	GET, POST, PUT, PATCH or DELETE.
URL *	The address, with variables if needed.
Headers	Each row: header → value (for example, the API key).
Body (JSON or text)	The body, with variables; **Send all context** or **Insert minimal template** build it for you, and the editor tells you whether the JSON is valid.
Timeout (seconds)	30 by default; recommended maximum 120.
Save response to variable	To read it in the following nodes, for example {{$api_response.body.estado}}.
Test	Sends the request now and shows you the response, before saving.

It has two outputs, **success** and **error**. At Bright Smile, “Notify the lab (HTTP)” sends the order to the lab’s system when a patient gets the Orthodontics tag.

```
POST https://lab-brightsmile.example.com/api/orders
{ "patient": "{{$contact.display_name}}", "phone": "{{$contact.phone}}" }
```

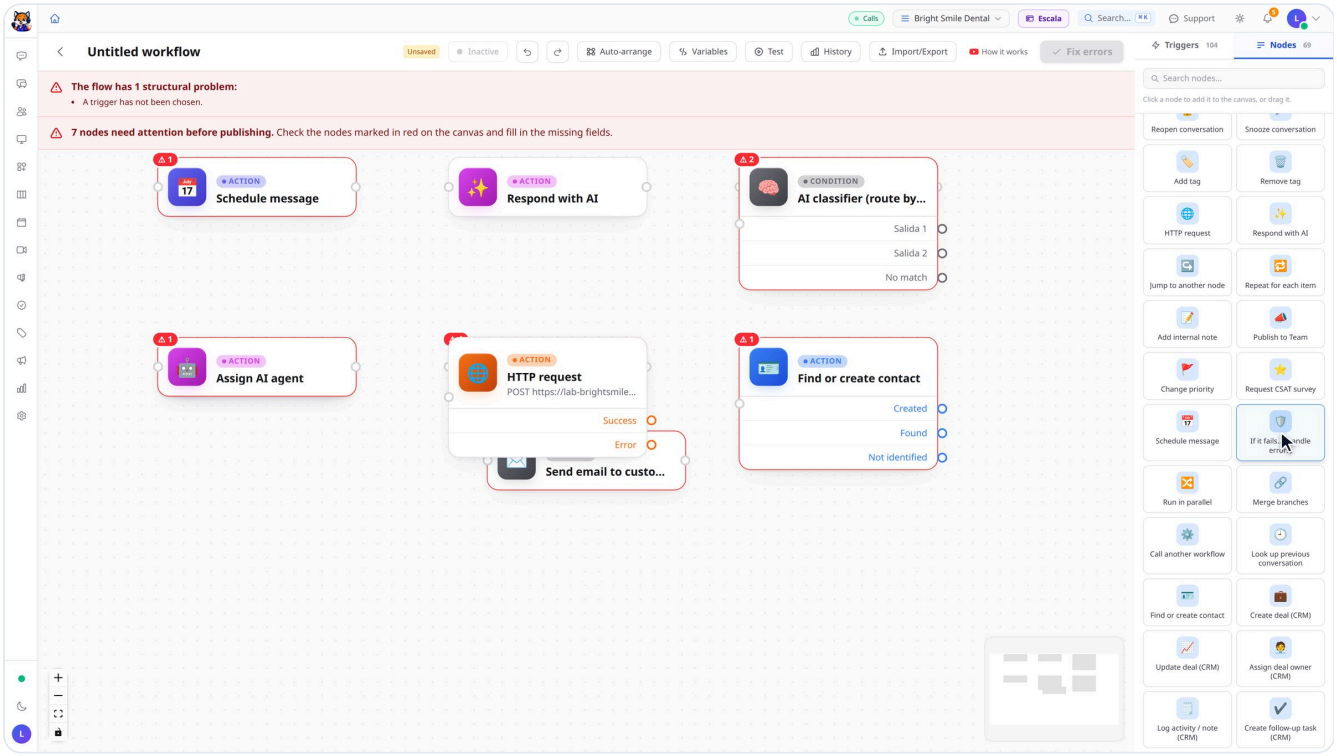
WATCH OUT

The node's Test button really sends the request, but without replacing the variables: to see them resolved, test the whole workflow (chapter 25). Connect the error output to a team alert, so you find out if the other system is down.

CHAPTER 43

43 Flow control

Organize large workflows and reuse steps.



A workflow with control nodes: repeat, parallel branches and merging them.

Node	What it does
Jump to another node	Continues at the node you choose: to go back (retry a question) or skip ahead.
Repeat for each item	Goes through a list (from an HTTP request or a variable): **Source variable**, **Item variable**, **Index variable** and **Max iterations**. The loop branch runs once per item ({{$item}}, {{$index}}) and the other continues when it's done.
If it fails... (handle errors)	Checks the **immediately preceding** step and continues along success or **On error**, with {{$error.message}} and {{$error.node}}. Place it right after the node you want to watch.
Run in parallel	Opens 1 to 5 branches (b1...b5).
Merge branches	Waits for all the connected branches to arrive and only then continues.
Call another workflow	Runs another of your workflows whose trigger is **Manual (called by another workflow or the AI)**, with an **Input (optional)** that the other one reads as {{$caller.input}}.

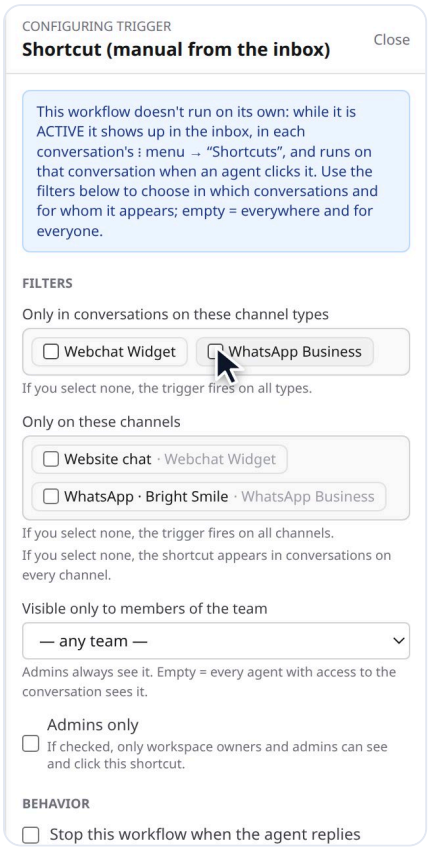
TIP

Call another workflow is ideal for shared steps: for example, a "Notify the front desk" workflow used by five different workflows. You change it in just one place.

CHAPTER 44

44 Shortcuts for your team in the inbox

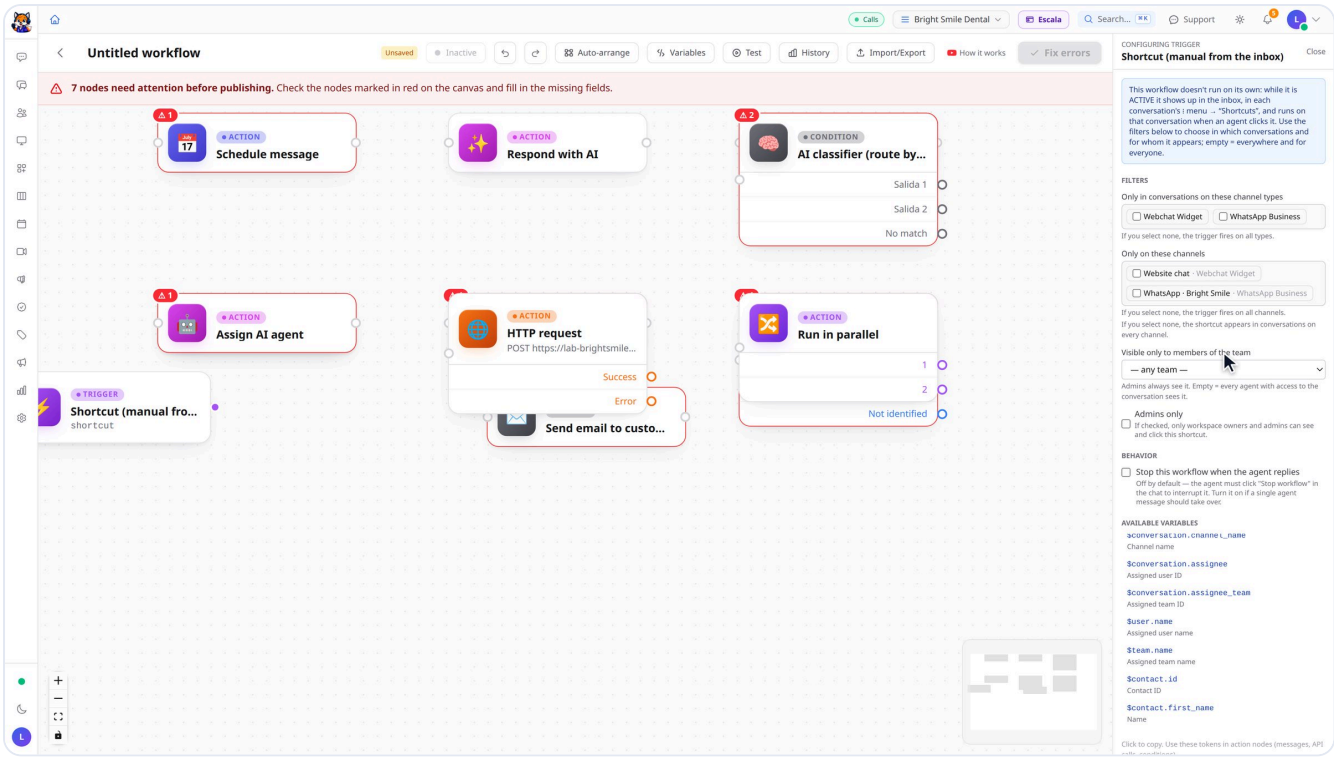
A button in the inbox that always does the same thing.



The Shortcut trigger and where it appears.

With the **Shortcut (manual from the inbox)** trigger, the workflow becomes a button: your team finds it in the conversation menu, under **Shortcuts**, and runs it with one click.

- **Only in conversations on these channel types** and **Only on these channels**: where it appears.
- **Visible only to members of the team**: who sees it (admins always see it).



The shortcut in the conversation menu, in the inbox.

For example, “Send instructions before the cleaning”: one click and the patient gets everything, always the same, without copying and pasting.

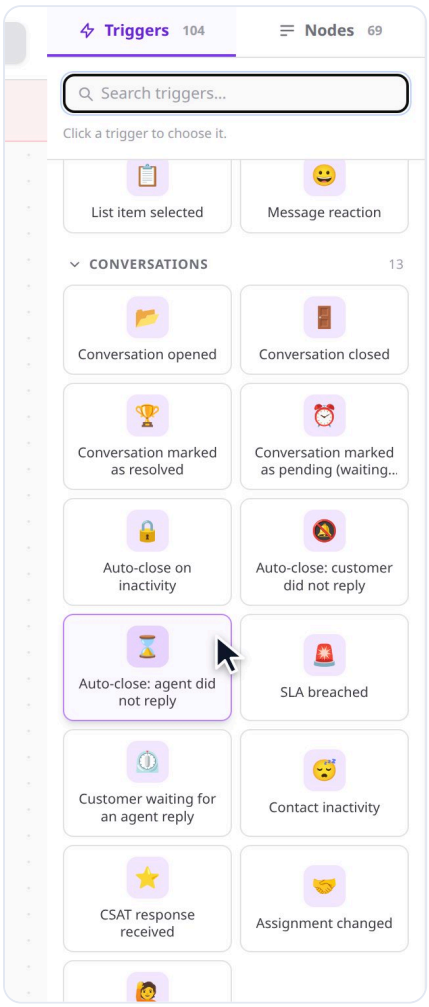
45

CHAPTER 45

Scheduled and incoming webhook

Workflows that start based on the time or on another system.

Scheduled (cron)



Scheduled (cron) with its Presets: every X minutes, hourly, daily, weekly or monthly.

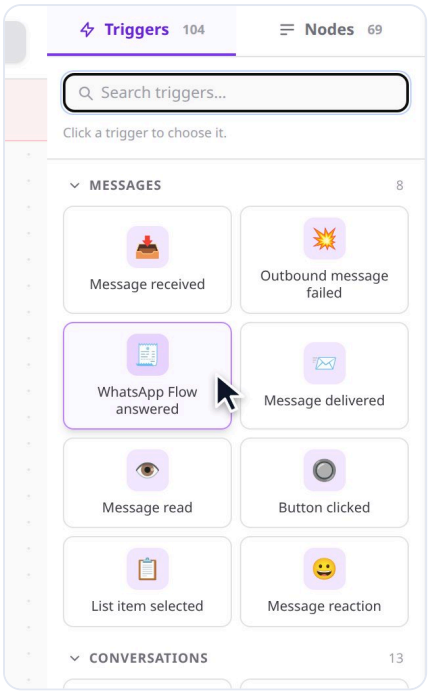
Runs the workflow on a schedule. With **Presets** you build the rule without knowing cron:

- **Every X minutes**;
- **Every hour (at minute...)**;
- **Daily at HH:MM**;
- **Weekly** (days of the week and time);
- **Monthly** (day of the month and time).

Below you see the **Generated cron**, for example `0 9 * * 1` = every Monday at 9:00.

Relative date (Time category) is for notices like “three days before something is due”: you choose the record type and the **Offset days** (negative = before, positive = after).

Incoming webhook



Incoming webhook: the Webhook URL with Copy and Regenerate URL.

Gives you its own **Webhook URL**, with **Copy** and **Regenerate URL**. When another system sends a POST (JSON) to that address, **only this workflow** runs, with that data in {{$webhook.payload.campo}}.

- Save the workflow so the URL gets generated.
- **Match by JSON path** filters, for example, `event.type == "order.paid"`.
- Regenerating the URL stops accepting the old one once you save.

WATCH OUT

Treat the webhook URL like a password: anyone who has it can launch the workflow. If it leaks, regenerate it.

46

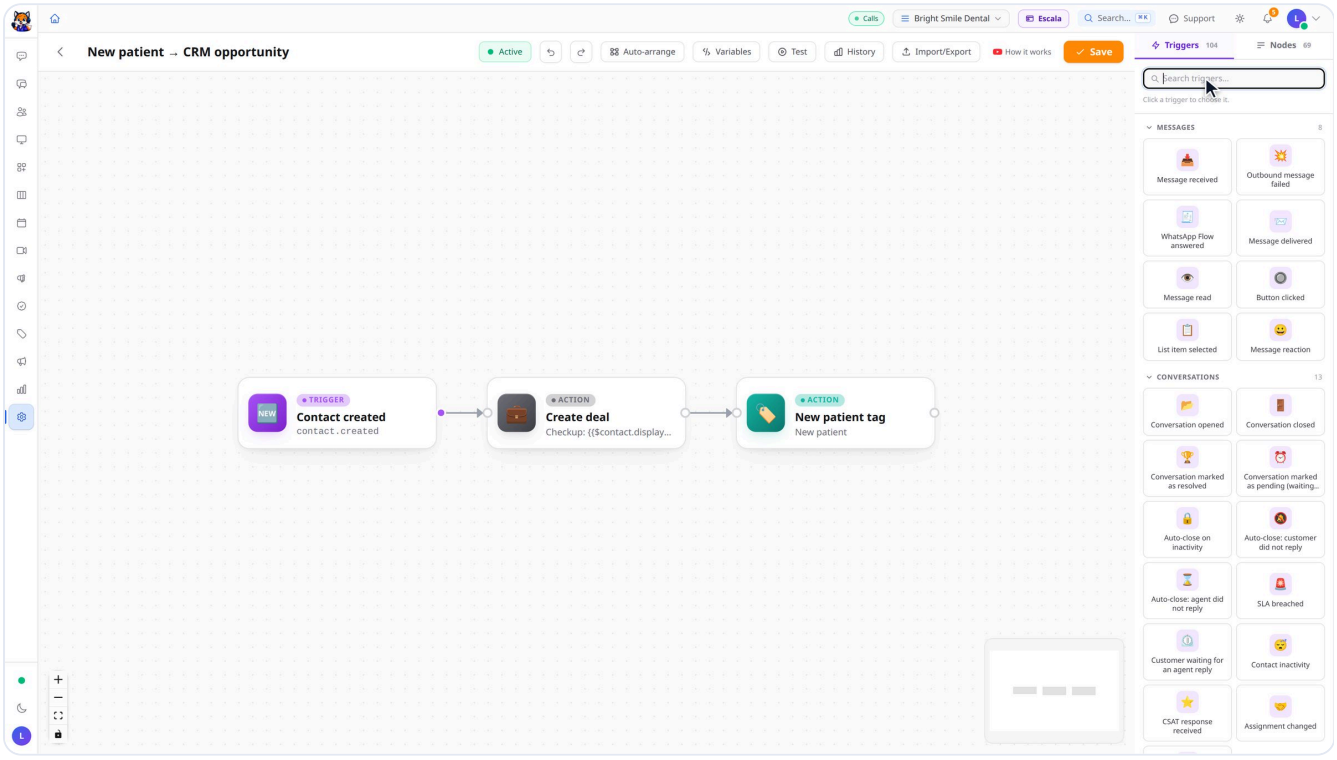
CHAPTER 46

CRM: triggers and nodes

Deals, tasks and activities, on autopilot.

With the CRM, workflows fire when a deal is **created**, **changes stage**, is **won** or **lost**, goes **days without activity**, gets **stuck in a stage**, changes priority, its customer doesn't reply, or an activity becomes overdue (table in chapter 9).

Node	What it does
Create deal (CRM)	Title (with variables), pipeline, stage (empty = the first one), amount, currency, contact, owner, tags and description. If the contact already has an open one in that pipeline, it reuses it.
Update deal (CRM)	Changes the stage (or marks it won/lost with a reason), the owner, the data and the tags. Each change is kept in its history.
Assign deal owner (CRM)	From the list or by variable.
Log activity / note (CRM)	A note, call, meeting or email on the profile (saved as already done).
Create follow-up task (CRM)	Type, title, due date (fixed or relative, respecting business hours), reminder and owner. It shows up in the owner's To-dos.



New patient → CRM opportunity.

The "New patient → CRM opportunity" workflow turns every new patient into a deal "Checkup: `{{$contact.display_name}}`" in the **Treatments** pipeline, at the "New patient" stage.

47

CHAPTER 47

Appointments: triggers and nodes

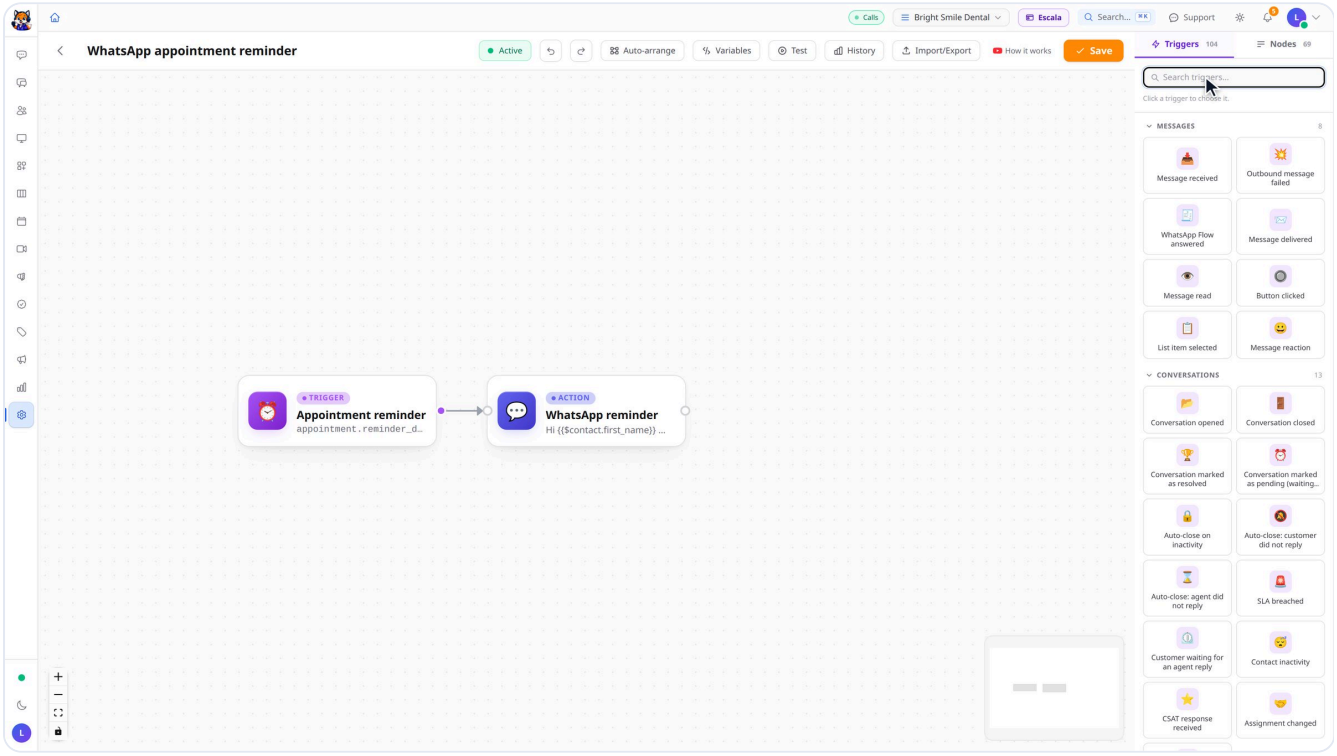
Reminders, no-shows and the waitlist.

Calendar triggers: **Appointment created**, **Appointment cancelled**, **Appointment rescheduled**, **Appointment confirmed by the customer**, **Appointment reminder**, **The customer did not show up (no-show)**, **Appointment completed**, **A spot opened up** and **Someone from the waitlist got a spot**. They bring variables such as

`{{$appointment.service_name}}` and `{{$appointment.time}}`.

Node	What it does
Schedule appointment (Calendar)	Start date and time (or a variable), calendar, service and title. It can be an Omnifox video call.
Reschedule appointment	Moves an appointment to another date; if the slot is taken, it doesn't move it.
Cancel appointment	Cancels it (with a reason) and frees the slot, which fires "A spot opened up".
Confirm appointment	Sets it to confirmed: useful on the "if they reply YES" branch.
Mark as no-show	Marks that person as a no-show.
Add to waitlist	If there's a slot they get in right away; if not, they stay waiting.
Check availability	Looks for free slots (like the booking page) to offer them.
Block time in the calendar	Nobody can book during that time.

Without an appointment ID, the nodes use the trigger's appointment or the contact's next appointment.



WhatsApp appointment reminder.

"WhatsApp appointment reminder" sends the day-before message using **Appointment reminder** and a template; "Missed appointment → reschedule" writes to the patient who didn't show up and creates a task for the front desk:

> Hi `{{$contact.first_name}}`, we were expecting you today and you couldn't make it. Shall we find you another time? 🦷

> **WATCH OUT**
>
> If you use a reminder workflow, turn off the calendar's built-in reminder, or the patient will get two.

Hours before the appointment

The **Appointment reminder** trigger has the **Hours before the appointment** field. Empty, the workflow runs with the workspace's general reminder (24 h before by default, or the hours set in the Medical module). With a number, for example 2 or 48, that workflow gets **its own reminder**: it runs once per appointment when that many hours are left. So you can have one workflow at 48 h and another at 2 h, each at its own time. It's checked every hour.

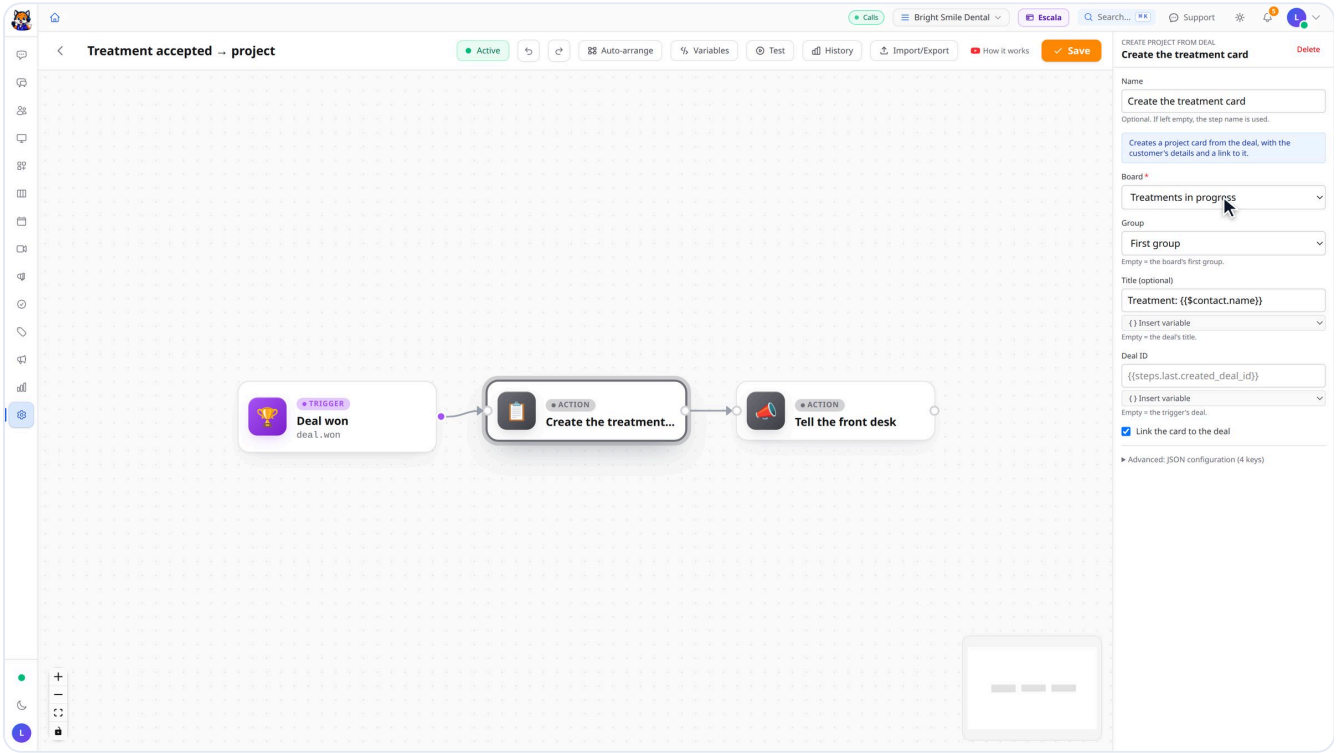
CHAPTER 48

48 Projects and Team

Boards and internal communication.

Projects

When a treatment is accepted, the "Treatment accepted → project" workflow (**Deal won** trigger) creates the patient's card on the **Treatments in progress** board, linked to the deal, with **Create project from deal**.



Treatment accepted → project.

Node	What it does
Create board item	A card on the board and group you choose.
Create project from deal · Create board card (from the conversation) · (from the appointment)	A card with the data already filled in and the link to the source record.
Update task cell · Complete task · Move task to another group	They change an existing card (the trigger's, or the one just created).
Link record (item ↔ deal)	Links a card with a deal or a contact.

And the **Boards** triggers: column changed, item created, entered a group, assigned, task status changed, subtask created, due soon and overdue.

The board node forms

These four nodes have their own form in the inspector. In a board workflow they use that board's columns; in a regular workflow you pick **Board · Column** from all the workspace's boards.

Node	What you fill in
Assign a person	The **person column**, the **people to assign** (workspace members) and, optionally, copy whoever is assigned in another column of the card.
Set a date	The **date column** and which date: **Today**, **In N days** (with the days; a negative number gives a past date) or **A fixed date** (YYYY-MM-DD or a variable).
Append row to Google Sheets	The **spreadsheet ID**, the **tab** and the **row cells** (one per line or separated by |, with variables). Left empty, they use the integration's sheet and tab. Requires Google Sheets connected.
Create/update HubSpot contact	**Email** (required), **first name**, **last name**, **phone** and **company**. If a contact with that email already exists, it updates it. Requires HubSpot connected.

> **NOTE**
> The Google Sheets and HubSpot integrations are still listed as "Coming soon" in Integrations: you can already set up these two nodes, and they'll start working once the integration is available to connect.

Team

Node	What it does
Publish to Team	Posts in a Team space, with variables and, if you want, the conversation card.
Send Team DM	A direct message to a person on the team.
Create Team reminder	A reminder for someone, on a date.
Schedule Team message	Posts in a space later.

Internal Team triggers: **You were mentioned in Team**, **Message posted in a Team Space**, **Team reminder is due** and **Team poll closed**. "VIP patient → notify the doctor" posts in the **Front desk & schedule** space when a patient with the VIP tag writes in.

49

CHAPTER 49

Orders, payments, calls and more

Catalog, payments and phone.

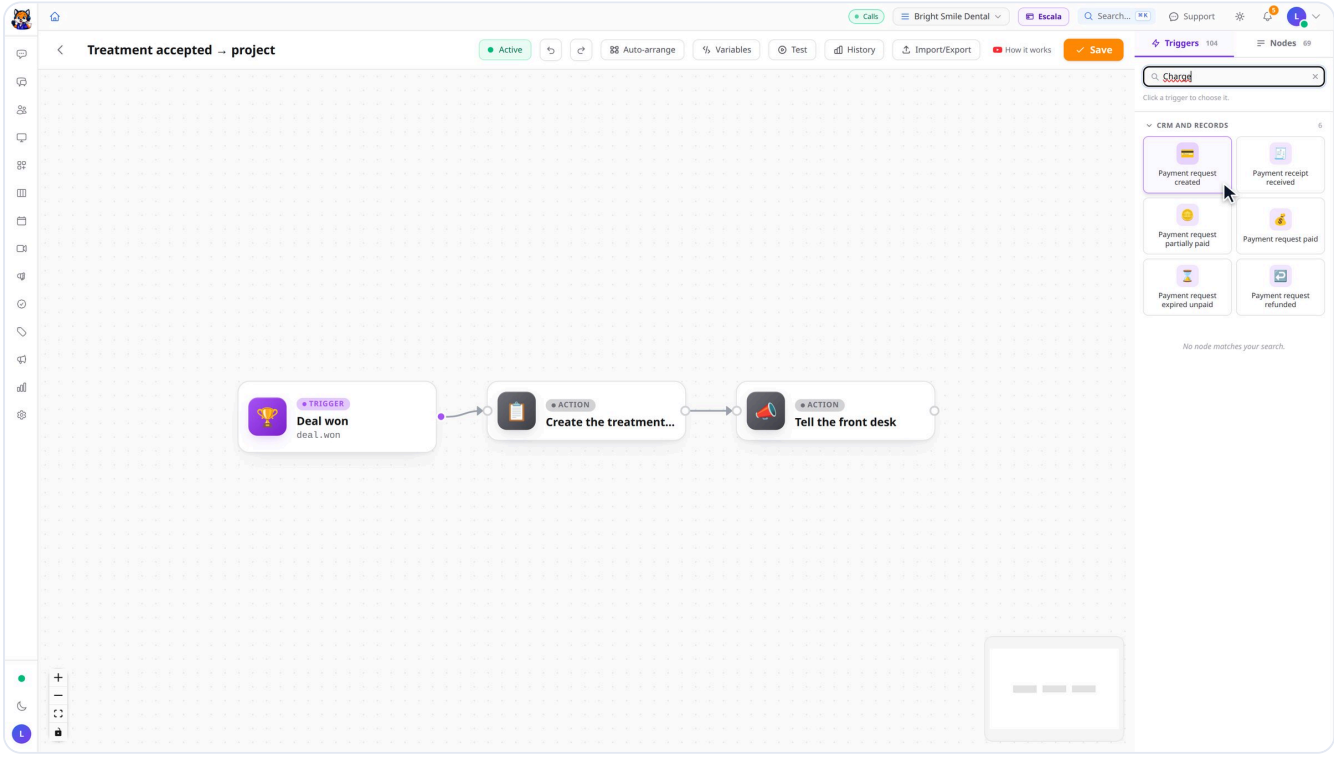
Orders, carts and payments

If you sell through the catalog, there are triggers for **orders** (created, confirmed, status changed, paid, canceled), **carts** (created, updated, abandoned, turned into an order, expired) and **payment requests** (created, receipt received, partially paid, paid, expired, refunded), with these nodes:

Node	What it does
Send cart link	Sends the customer the link to their cart, with what they already had in it. Ideal after "Abandoned cart".
Empty the cart	Leaves it with no products.
Turn the cart into an order	Creates the order with current prices.
Change order status	Confirmed, Preparing, Ready, Shipped, Delivered or Cancelled, with a reason.

Calls

With calls: **Missed call**, **Out-of-hours call**, **Recording ready**, **Voicemail received**, **IVR: option selected** and the rest in the table in chapter 9. For example, after a missed call, write to them on WhatsApp: "We saw your call, how can we help?". Call triggers are available on plans with calls.



Order, payment and call triggers.

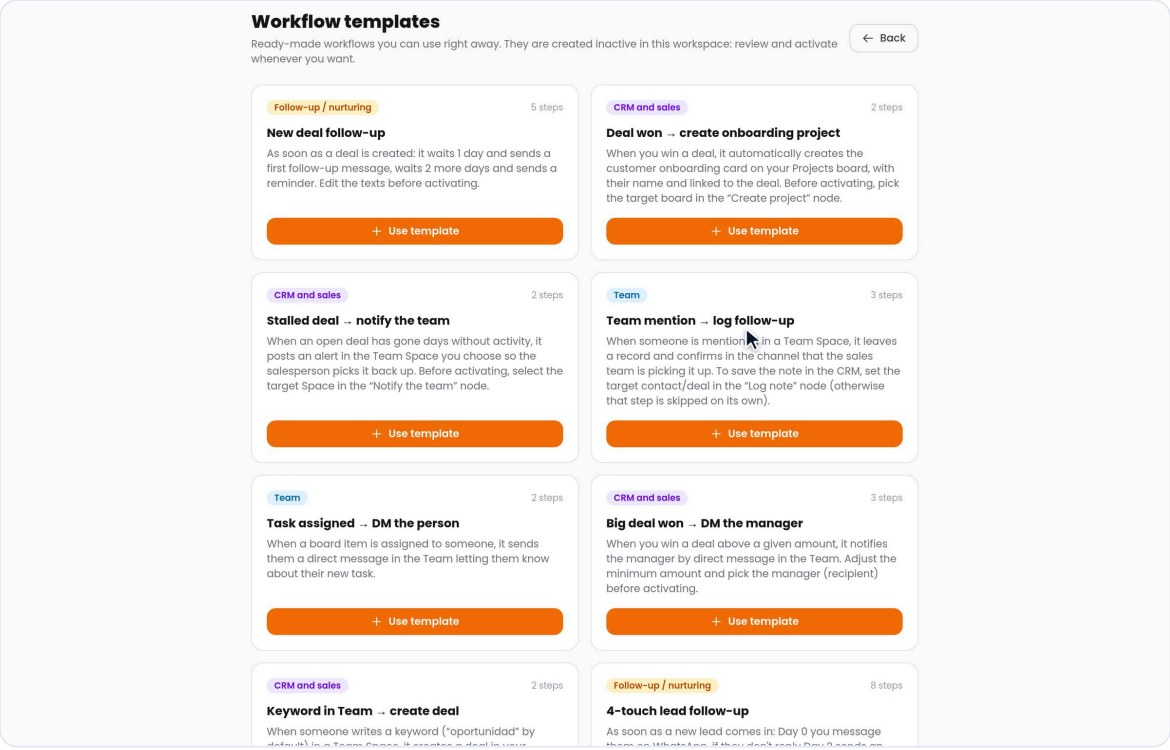
> **NOTE**
>
> With the Azur integration connected, the Invoice authorized in Azur and Invoice voided in Azur triggers also appear, along with nodes to register the customer and send them the invoice PDF.

CHAPTER 50

50 The template gallery

Ready-made workflows for everyday cases.

Under **Templates** (a button on the list) you have ready-made workflows, grouped into **CRM and sales**, **Follow-up / nurturing**, **Team** and **Appointments**.



The template gallery, grouped by topic.

Template	What it does	Fill in before turning on
Appointment created → WhatsApp confirmation	Confirms every new appointment.	Turn off the calendar's built-in notice.
Appointment reminder → WhatsApp	Reminds people of their appointment.	Turn off the built-in reminder.
New deal follow-up	Two messages with waits after a deal is created.	The texts.
4-touch lead follow-up	Message, email, task and message, with waits; it pauses if the lead replies.	The texts.
Stalled deal reactivation	Message, email and task for a cold deal.	The texts.
New lead welcome	Message, email and task for every new contact.	The texts.
Deal won → create onboarding project	Creates the project card.	The target board.
Stalled deal → notify the team	Posts in Team.	The Team space.
Big deal won → DM the manager	Sends a DM if the amount is high.	The amount and the manager.
Keyword in Team → create deal	Creates a deal from a Team message.	The keyword.
Team mention → log follow-up	Posts and logs the activity.	The Team space.
Task assigned → DM the person	Notifies whoever gets a task.	—

1. Press **Use template**.
2. It's created in your workspace, **inactive**, and opens in the editor with the notice "Template created. Review it and activate when you are ready.".
3. Check what it says needs filling in (the board, the Team space...) and adjust the texts to your business.
4. Test it and turn it on.

> **NOTE**
> If you'd already used that template, Omnifox takes you to your copy instead of creating another one.

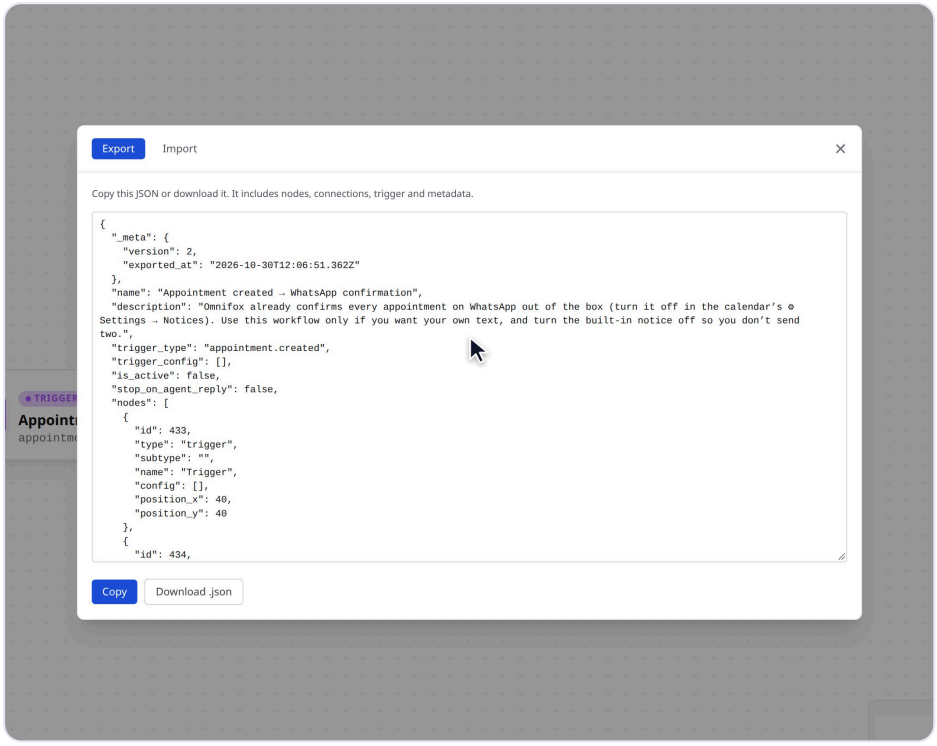
51

CHAPTER 51

Export and import

Take a workflow to another workspace or keep it as a backup.

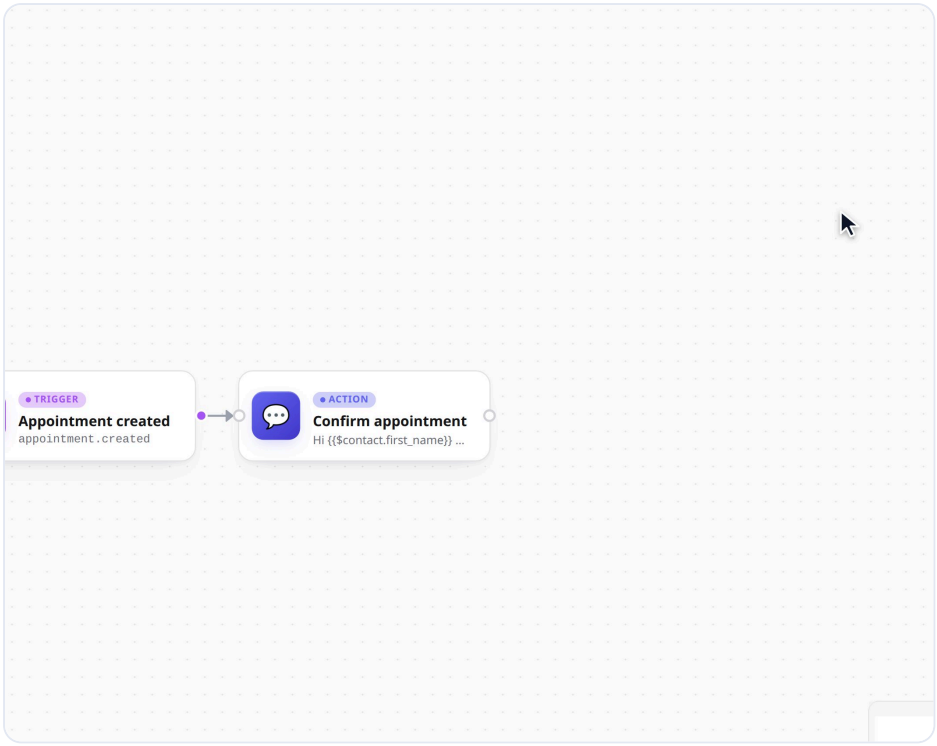
Export



Import/Export, Export tab: the workflow as JSON to copy or download.

In the editor, **Import/Export → Export** gives you the workflow as JSON text ("It includes nodes, connections, trigger and metadata") to **Copy** or **Download .json**. From the list, the Export button on the card downloads the same file.

Import



Import: map external references and choose the strategy.

1. In **Import/Export → Import**, use **Load .json** or paste the JSON. Omnifox validates it ("Valid file — N nodes / N connections").
2. Under **Map external references**, match whatever doesn't exist in this workspace: channels, teams, users, tags, AI agents, called workflows and WhatsApp templates. Leave one unmapped to skip it.
3. Choose the **Strategy**: **Add alongside the current flow** or **Replace the whole flow**.
4. Press **Apply import** and then **Save** (applying doesn't save).

From the list, the **Import** button does the same with a file (up to 2 MB) and creates a new, **inactive** workflow.

> **TIP**
> Exporting is also your backup: before a big change, download the .json. If something goes wrong, import it with "Replace the whole flow".

52

CHAPTER 52

Plans, limits and permissions

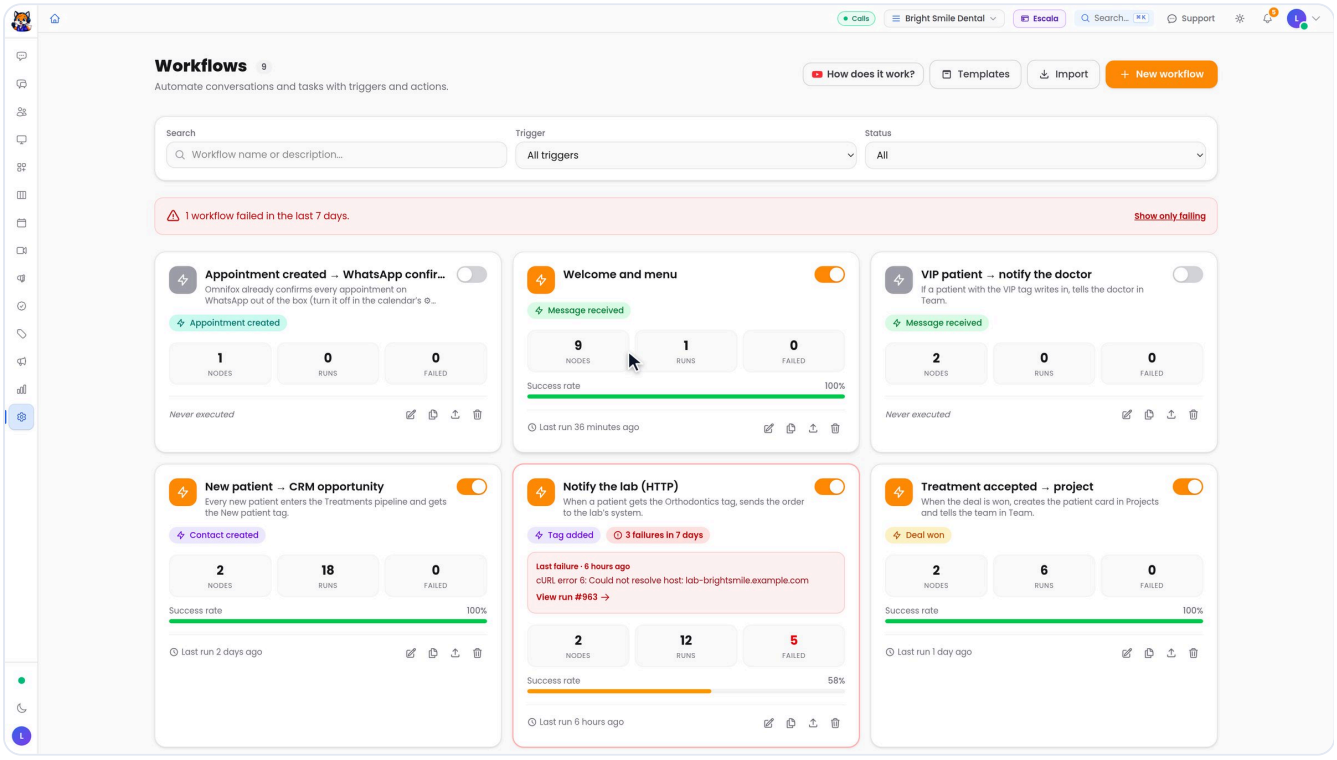
How much you can run and who can edit.

Runs per month

Each plan includes a number of **runs per month**. A run is each time a workflow starts, whether it does one step or twenty.

Plan	Runs per month
Lite	1,000
Impulsa	5,000
Conecta	15,000
Crece	30,000
Escala	50,000
Exclusivo	Unlimited

- At **80%** we let you know by email.
- When you reach the cap, workflows **pause until next month**: no new runs start (the reason is logged).
- The counter is shared by workflows and phone menus (IVR).
- Call triggers are on plans with calls (from Conecta up); some nodes may require a higher plan and show a padlock in the palette.



Workflow runs in Billing.

Permissions

Permission	What it allows
View workflows	See the list and open the editor in **read-only mode**.
Manage workflows	Create, edit and save, turn on and off, test, duplicate, delete, use templates, import and export.

On a phone, the editor opens in **read-only mode**, step by step: you can review a workflow, but editing is done on a computer.

> **NOTE**
> Blocked contacts don't trigger workflows (except shortcuts and manual workflows). And each workflow only works with its own workspace's data.

53

CHAPTER 53

Best practices

Workflows that work from day one.

- **Name every node** ("Greeting", "Menu", "Hand to front desk"): the canvas and the history explain themselves.
- **Start with short workflows** and add steps once they work.
- **Always test them** with your own number or a test conversation before turning them on.
- Use **Only on the first inbound message** for welcomes, or the customer will get the menu with every message.
- **Always set a time limit** on questions and event waits, and connect Timeout.
- Connect the **Holiday**, **Invalid** and **error** outputs: an unconnected path is a customer with no reply.
- On long follow-ups, turn on **Stop this workflow when the agent replies**.
- Outside WhatsApp's 24 hours, use **templates**.
- For steps repeated across several workflows, use **Call another workflow**.
- **Check every week** the failure notice on the list and the history of your most important workflows.
- Before a big change, **duplicate or export** the workflow.

54

CHAPTER 54

Troubleshooting

What to check when something doesn't go as expected, FAQ and glossary.

Symptom	What to check
The workflow doesn't fire	That it's **active and saved** (Active pill + Saved). That the event meets the trigger's **filters**: an unmet filter leaves no error or run. That "Run only once per contact" isn't checked with a contact who already went through. That the contact isn't blocked.
It stopped halfway	Open the **History**: the step in red tells you exactly what failed. Check whether it was left "Paused at" a question or a wait with no time limit.
A WhatsApp message didn't arrive	It may be the **24-hour window**: outside it, use Send WhatsApp template. Also check the "Outbound message failed" trigger.
The customer gets the menu with every message	Check **Only on the first inbound message of a conversation** in Message received.
A message went out with a gap where the name should be	That contact doesn't have that data: empty variables stay empty. Use `{{$contact.display_name}}` or check the profile.
I can't save	The button says **Fix errors**: complete the nodes in red. If a test is running, wait for it to finish or stop it.
I can't turn it on	The workflow has errors: it was saved as an inactive draft. Fix what the notice points out.
It fires twice	Check whether there are two workflows with the same trigger and similar filters (filter the list by Trigger).
The workflow kept writing when a person was already helping	Turn on **Stop this workflow when the agent replies**, or use Stop from the conversation notice.
All workflows paused	You reached the month's run cap (chapter 52).

Frequently asked questions

How many workflows can I have active at the same time?

Several at once; what your plan limits is the **runs per month** (chapter 52).

Can a workflow have two triggers?

No, just one. If you need two events, create two workflows and have both use **Call another workflow** with the shared steps.

Is there a version history?

No. Before a big change, duplicate the workflow or export it as a backup.

Can a workflow be generated with AI?

No: workflows are built in the editor or started from a template. What you can do is combine them with AI agents (chapter 41).

Can a failed run be retried?

Yes. In the **History**, open the failed run and click **Retry** (chapter 27): it runs the whole workflow again, with the same trigger data and the current version. Careful: steps that had already succeeded are repeated.

Does testing use up runs and send real messages?

Yes: a test is a real run. Use your own number or a test conversation.

Can I send an image or a PDF?

Yes: in Send message choose Message type Image or File and paste the URL; the text goes as the caption.

Can I edit workflows from my phone?

On a phone the editor opens in read-only mode. To edit, use a computer.

What happens to a paused workflow if I turn it off?

No new runs fire. The ones that were already waiting can be stopped from the History.

Do workflows work when the clinic is closed?

Yes, 24 hours a day. To reply differently at night, use Business hours.

Glossary

Workflow
An automation: a trigger and a chain of nodes.

Trigger
The event that sets the workflow in motion. One per workflow.

Filter
A trigger condition: if the event doesn't meet it, the workflow doesn't start.

Node
Each step of the workflow: action, condition or wait.

Action
A node that does something: send, tag, assign, create.

Condition
A node that chooses the path based on a rule (If, Multiple branches, Business hours...).

Wait
A node that pauses the workflow for some time or until an event.

Output / branch
Each path that leaves a node, with its name (Yes, In hours, Prices...).

Connection
The line that joins one node's output with the next one's input.

Inspector
The right-hand panel with the selected node's fields.

Palette
The panel with the Triggers and Nodes tabs.

Variable
A piece of data that changes on every run, such as `{{$contact.first_name}}`.

Run
Each time a workflow starts. They're counted per month according to your plan.

History
The list of a workflow's recent runs, with their steps.

Shortcut
A workflow your team launches with a button from the conversation.

Webhook
An address that receives data from another system and launches the workflow.

WhatsApp template
A Meta-approved message for writing outside the 24-hour window.

24-hour window
The time, from the customer's last message, during which WhatsApp lets you write freely.

Deal
A potential sale in a CRM pipeline.

Need help?

Write to us from the **Support** button in the app or visit the help center at omnifox.io/help. If you don't use Omnifox yet, try it free for 7 days at **omnifox.io**.