

THE BROKER TIMES

[START HERE](#)

Start here

AI WhatsApp Agent Engine · thebrokertimes.com.au

An AI agent that answers your WhatsApp enquiries the moment they arrive, qualifies them, and books the good ones — built on Make, so you can see and change every step without writing code.

Built by The Broker Times.

What this is

A client messages your WhatsApp business number at 9:47pm.

They get an instant reply that tells them they are talking to an assistant and asks their name. Once they answer, they get a menu of four — Business loan, Vehicle & equipment, Home loan, Personal loan — so one tap tells you what they want.

From there, every message Meta receives goes to a Make scenario. It looks up who the person is and what you have already discussed, sends that to an AI model along with your knowledge base and your rules, and posts the reply back to WhatsApp — in a few seconds, in your voice, addressed to them by name, without quoting a rate or promising anything it shouldn't.

Hot enquiries get pushed to you immediately. Everything is logged.

What this is not

- Not a done-for-you service. You build it, in your own accounts.
 - Not an adviser. It gives general information and books calls. The guardrails are the product.
 - Not a way to send marketing messages. It only ever responds to people who message you first.
-

Before you start

You need:

- A Make account (any paid plan)
- A Meta Business account you can verify
- A phone number that is **not** currently on the consumer WhatsApp app
- An OpenRouter account with a small amount of credit
- About an hour to write your own knowledge base
- Your licensee's willingness to look at it before you go live

Do it in this order

- 1. Start Meta business verification right now.** It's the only step with a queue. `guides/01-meta-whatsapp-setup.pdf`
 - 2. While that runs, write your knowledge base.** The highest-value hour in the build. `prompts/knowledge-base-template.md`
 - 3. Create the two data stores.** Before importing anything. `data-structures/data-stores.pdf`
 - 4. Import the blueprint and wire it up.** `guides/02-make-setup.pdf`
 - 5. Add your OpenRouter key and pick a model.** `guides/03-openrouter.pdf`
 - 6. Test it properly, then go live narrowly.** `guides/04-testing-and-go-live.pdf`
 - 7. When something breaks:** `guides/05-troubleshooting.pdf`
- Read** `compliance/compliance-pack.pdf` **before step 6, not after.**
-

What's in the box

```
START-HERE.pdf           this file
blueprints/
  01-whatsapp-ai-agent.blueprint.json  the importable Make scenario
prompts/
  system-prompt.pdf           the agent's brain and its hard rules
  system-prompt.md           the same, as plain text to copy from
  knowledge-base-template.pdf  the part only you can write
  knowledge-base-template.md  the same, as plain text to fill in
guides/
  01-meta-whatsapp-setup.pdf
  02-make-setup.pdf
  03-openrouter.pdf
  04-testing-and-go-live.pdf
  05-troubleshooting.pdf
data-structures/
  data-stores.pdf           create these first
compliance/
  compliance-pack.pdf       read before launch
  pre-launch-checklist.pdf  print this one
```

Everything is a PDF so it reads and prints properly. The two files you need to copy text out of — the system prompt and the knowledge base template — are included as plain `.md` as well, so you can select and paste without fighting a PDF.

How the scenario works

```

WhatsApp message arrives
    ↓
[1] Watch Events           Meta's webhook fires
    ↓
[2] Set Variables          pull out sender, text, name; detect opt-out keywords
                           (filter: real messages only, not delivery receipts)
    ↓
[3] Search wa_contacts     who is this, and have they opted out?
    ↓
[4] Router (four routes, only one ever fires)
    |— they said STOP → mark opted_out, send confirmation, stop
    |   [20] Add wa_contacts     opted_out = true, stage = STOP
    |   [21] Send WhatsApp       the unsubscribe confirmation
    |
    |— first contact → AI disclosure, then ask their name
    |   [50] Send WhatsApp       "you're chatting with an assistant... your name?"
    |   [51] Add wa_contacts     create the record, stage = AWAITING_NAME
    |   [52] Add wa_messages     log their opening message
    |
    |— they just gave their name → save it, send the loan-type menu
    |   [60] Add wa_contacts     first_name = their answer, stage = AWAITING_TYPE
    |   [61] Add wa_messages     log their name message
    |   [62] Send list message   Business loan / Vehicle & equipment /
    |                           Home loan / Personal loan
    |
    |— everyone else → the AI route
    |   [28] Search wa_messages   last 15 messages for this person
    |   [29] Text aggregator      build the transcript
    |   [30] Transform to JSON    escape the system prompt safely
    |   [31] Transform to JSON    escape the history and new message
    |   [32] HTTP → OpenRouter    ask the model
    |   [33] Parse JSON           reply / stage / handover / capture
    |   [34] Send WhatsApp message the reply goes out
    |   [35] Add wa_messages      log what they said
    |   [36] Add wa_messages      log what we said
    |   [37] Add wa_contacts      update stage and notes
    |   [38] HTTP → your webhook  only if HOT, handover or opt-out
  
```

Roughly eight to ten Make operations per message on the AI route. A first contact costs four, because it never touches the model.

The one thing people get wrong

The temporary access token on Meta's API Setup page expires in about 24 hours. Everything works, then it silently stops the next day. Use a permanent System User token from the start. Guide 1, Step 3.

Support and updates

Updates are free for life and go out by email to the address you bought with.

If you get stuck, email us with the scenario name and a screenshot of the failing module's input and output bundles from Make's execution history. We'll point you at the right part of the guide. We don't log into your accounts.

Legal

Educational and productivity software. Not legal, compliance or credit advice. Does not create a credit representative relationship.

WhatsApp is a trademark of Meta Platforms, Inc. Make is a product of Celonis. The Broker Times is not affiliated with, endorsed by or sponsored by either, or by OpenRouter or any model provider.

You are responsible for your obligations under the National Consumer Credit Protection Act, the Privacy Act 1988 and the Spam Act 2003, for your licensee's policies, and for everything sent from your business number.