

fighting chargebacks for local stores

guide 9: win back the money stores lose to card disputes

the offer

when a customer disputes a card payment, the store gets a chargeback. the money is pulled back out of their account straight away, and they get a set number of days to send evidence to the card network if they want it back. most small stores either miss that deadline or send the wrong evidence, because nobody ever told them what the networks actually accept, so they lose disputes they could have won.

the stores that do fight usually pay a firm that keeps a cut of whatever it wins back. so like the invoice guide, the owner is already used to paying someone a share to get money back, they just don't have anyone doing it for them yet.

the offer is that you fight those disputes for them and take a cut of what you win back, and nothing for the ones you lose. the agent gathers the evidence and writes the response, and a decision model called jev checks every dispute against the rules first, so you only fight the ones that can be won and you never miss a deadline. every step below comes with the exact prompt to paste in.

step 1

set up the agent

the whole pipeline runs on an agent, and there are two good options for it. hermes agent at hermes-agent.nousresearch.com is open source and free and runs on your own computer or a server, and grok bot at x.ai/bot runs on its own computer in the cloud, so it keeps working when your laptop is closed. either way you message it the same way you'd message a person, so pick whichever one you like because everything in this guide works the same on both.

to set up hermes, open a terminal and run `curl -fsSL https://hermes-agent.nousresearch.com/install.sh | bash` or download the desktop app for your platform from the site, then sign in through nous portal when it asks. you talk to it from telegram, whatsapp, or discord, and if you want it running all day without your computer on, you can put it on a server like railway.

to set up grok bot, sign up at x.ai/bot, create a bot, give it a name and tell it what its job is, then message it from the desktop app or your phone. it comes with its own browser and terminal and picks the model for you, so there's nothing to install.

hermes needs one more step, which is connecting a model, because the model is what does the thinking. the simplest route is pasting in an anthropic or openai api key when setup asks for it, and the nous portal plans also come with monthly model credits and access to hundreds of models. use a strong model, because this pipeline involves judgment calls and a weak model makes expensive mistakes with real customers and real money.

after that, give it the keys for the job. in chat, tell the agent which accounts it's working with and paste the credentials when it asks: a google places api key from google cloud console with the places api enabled and a budget cap on it, a postcard api account with lob, postgrid, or stannp, and a vercel ai gateway key for jev, which the next step covers. once a store signs you'll also connect their stripe or shopify account so the agent can see their disputes and orders. on grok bot you can also log into a website once in the bot's browser and it stays logged in.

then just talk to it. there's no dashboard and no workflow builder, you describe the job in plain language the way you'd brief an employee, and the agent writes its own scripts, saves what works as a reusable skill, and can put itself on a schedule, which grok bot calls a routine. the exact brief to paste in is at the end of this guide.

step 2

connect jev

jev is a decision model from typesafe. the agent does the gathering and the writing, and jev is the part that decides. you give it a situation, like one dispute with its reason code, the order, and what evidence the store has, and you ask it a question with a fixed set of answers. it picks one and gives you a probability for how sure it is, from 0 to 1, and the agent only acts when jev is sure and sends everything else to you.

you get jev through vercel ai gateway. make a vercel account, open ai gateway from the dashboard, create an api key, and paste it into the agent when it asks. the model name is typesafe-ai/jev, and the agent calls it the same way it would call any other model. it costs almost nothing to run, when i tested all three of the results-based services in this series, 472 jev decisions came to \$0.0735 in total.

a jev question for a dispute looks like this:

```
situation:  
dispute on order #5531, $289, reason: item not received
```

```
deadline in 9 days
tracking shows delivered to the billing address on the 12th
no signature, customer has not messaged the store
```

```
question: what should the store do?
answers: fight, accept, gather more
```

before building anything else, have the agent make a few test disputes and run them through jev so you know the connection works.

```
connect to jev through my vercel ai gateway key (model typesafe-ai/jev). make up 3
disputes: one "item not received" where tracking shows delivery to the billing address, one
"fraud" where the billing address and shipping address don't match and there's no cvv
match, and one "not as described" where the customer emailed the store happy with the
product a week after delivery. ask jev for each: fight, accept, or gather more. show me
each answer and its probability.
```

step 3

write the rulebook

every dispute comes with a reason code, which is the reason the customer gave their bank, and each reason code has its own evidence that wins. the rulebook is that list written in plain words, and the agent turns it into questions for jev. this is the one i used:

```
fraud / card not recognized: address and cvv match, delivery to the billing address, proof
the customer used what they bought
item not received: tracking that shows delivery to the billing address, a signed receipt
for expensive orders
not as described: the product page and photos at the time of sale, the customer's own
messages
canceled or refunded: the policy shown at checkout, proof the customer agreed to it
fight if the winning evidence is there, let it go if it isn't
only list evidence that exists at the time you submit
always respond before the network's deadline
```

for every dispute jev answers four questions. fight, accept, or gather more evidence. how likely a win is, low, medium, or high. is the evidence complete. and is the deadline at risk. the agent only prepares a fight when jev is more than 0.8 sure and the win chance is at least medium, and everything else comes to you with jev's answers next to it.

the line about evidence that exists at the time you submit came from my own test. my first rulebook asked for a document that can't exist until after you've submitted, and jev kept flagging disputes as incomplete until i fixed the line. every mistake you find gets added to the rulebook the same way.

here is my rulebook for chargebacks, written in plain words: [paste your rulebook]

turn it into questions for jev (model typesafe-ai/jev through my vercel ai gateway key). for each dispute jev should answer: fight, accept, or gather more; win likelihood (low, medium, high); is the evidence complete (yes/no); is the deadline at risk (yes/no).

decision rules: prepare a fight only when jev is above 0.8 and win likelihood is at least medium. everything else goes to my review list with jev's answers.

save it as a skill called "chargeback rulebook" so i can make a copy per client.

step 4

write the response

when jev says fight, the agent writes the response the card network actually reads. it's short and factual: what was ordered, when, how it was paid, where it was delivered, and then each piece of evidence with a one-line note on what it shows. the evidence goes in as attachments, like the tracking page, the order confirmation, and screenshots of the customer's messages, and nothing gets listed that the store doesn't actually have.

for every dispute marked fight, write the response for the card network: one short paragraph with the order number, date, amount, product, and delivery details, then a numbered list of each piece of evidence with one line on what it shows. attach the evidence files from the order: [tracking, receipt, product page, customer messages]. only use evidence that exists right now. show me the response and attachments before anything is submitted.

step 5

find the stores

this fits businesses that take a lot of card payments and ship or deliver things, so pull furniture stores, home goods stores, boutiques, jewelers, and restaurants that take online orders. filter for a rating of 4.0 or above, a phone listed, and a website, and put the ones with an online shop first, because online orders are where most chargebacks come from.

use my google places api key to pull every [furniture store / home goods store / boutique / jeweler / restaurant with online ordering] in [city]. save name, address, phone, website, rating, and review count. visit each website and note whether it has an online shop and what it runs on (shopify, square, woocommerce, other). filter to rating 4.0 or higher with a phone and website, put online shops first, and wait for my approval.

step 6

build the demo page

you don't have their real disputes yet, so the agent makes three mock disputes that fit the store's type, like a furniture order marked as not received, a jewelry order marked as fraud, and a clothing order marked as not as described. it fills in the evidence checklist for each one, runs them through jev, and builds a page showing jev's verdict and how sure it is next to each, plus the full response for the strongest one.

at the top of the page is their name and logo, and at the bottom is the total that would have been won back. mark it clearly as mock data and end with one line: send us your last 3 disputes and we'll run them for real, free.

for [store name], make 3 mock disputes that fit a [store type]: realistic products and amounts, three different reason codes. for each, fill in which evidence the store has and doesn't have, and run it through my chargeback rulebook skill with jev. build a demo page: their name and logo from [website] at the top, each dispute with its reason code, the evidence checklist with ticks and crosses, jev's verdict and probability, and the full response for the strongest fight. put the total that would be won back at the bottom. mark it clearly as mock data, end with "send us your last 3 disputes and we'll run them for real, free." and my number: [your number]. host it on its own link.

step 7

the postcard

postcard apis like lob, postgrid, and stannp mail real printed cards from an api call for a dollar or two each. the front is a screenshot of the demo page with the verdicts showing, and the back says "we set up chargeback fighting for [store name]. scan to see which disputes you'd win." with the qr to the demo page and your phone number. store owners also live in their instagram dms, so the same link can go there too.

design a 6x9 postcard per store. front: a screenshot of their demo page showing the 3 disputes and verdicts. back: "we set up chargeback fighting for [store name]. scan to see which disputes you'd win." with a qr to the demo page and my name and number: [your name], [your number]. show me 3 samples. after i approve, send the batch through my [lob / postgrid / stannp] account and log every send. also give me a one-line dm for each store's instagram with the same link.

step 8

when they scan

the agent watches the demo links and messages you when one is opened. a store owner who looked at the page for a while has almost always lost a dispute recently and remembers it, so

call them the same day and ask if they've had any chargebacks lately.

the two questions you'll get are how you get paid, which is a cut of what you win back and nothing if you lose, and whether they need to give you access to anything, which for the free first job is just the details of their last few disputes.

message me whenever a chargeback demo page is opened, with the store name, their phone, and time on page. for stores that opened the page but didn't call after 3 days, draft one short follow up text from me for my approval. only one follow up per store.

step 9

the free first job

when a store says yes, ask for their last three disputes, which they can see in their stripe, shopify, or square dashboard, plus the order details for each. the agent runs them exactly the same way as the demo, pulls whatever evidence the store has, and shows you the verdicts and the response for every fight before anything is submitted. if a dispute is still open and inside its deadline, that one gets fought for real straight away.

when i tested this on 30 disputes for a home goods store, run twice, jev got 27 of 30 right both times and called the deadline risk correctly on all 30. it never told me to fight a dispute that couldn't be won and never told me to give up on one that could, and the three it got wrong were disputes it sent to gather more evidence, which cost nothing but a second look.

step 10

fulfillment

once a store signs, connect their stripe or shopify account so the agent sees new disputes the day they come in, along with the order details, tracking, and customer messages it needs as evidence. then put it on a daily schedule, which on grok bot is a routine and on hermes is a scheduled task. every morning it checks for new disputes, asks jev, builds the response for every fight, and shows it to you well before the deadline.

when a dispute is won, the money goes back to the store, and the agent keeps a running total of what was won. at the end of each month it sends you and the owner a statement listing every dispute, what happened, and the total won back, and you bill your cut from that.

[store name] connected their [stripe / shopify] account. every morning, check for new disputes. for each one, pull the order, payment details, tracking, and any customer messages, run it through their copy of my chargeback rulebook with jev, and write the response for every fight. put everything in front of me at least [n] days before each

deadline. after i approve, submit the response through their [stripe / shopify] dashboard. at the end of each month, send me and the owner a statement of every dispute, the outcome, and the total won back.

step 11

adding more clients

every new store is the same skill with a copy of the rulebook and their own account connected, so one agent can run all of them. the daily routine loops through each store in turn, and your review list shows which store each dispute belongs to. this also stacks well with guide 8, because a store that's losing money to chargebacks usually has other money going missing too.

the full prompt

once every step above works on its own, this is the whole pipeline in one message. paste it in, fill in the brackets, and the agent saves it as one skill you can rerun on a new city with a single line.

you are my chargeback agent. run this pipeline and check in with me at the marked points.

niche: [furniture / home goods / boutiques / jewelers / restaurants with online ordering]
city: [your city]

1. use my google places api key to pull every [niche] in [city]. save name, address, phone, website, rating, and whether the website has an online shop.

2. filter to: rating 4.0 or higher, phone and website listed, online shop first. show me the list and wait for my approval.

3. for each approved store, make 3 mock disputes that fit their store type, run them through my chargeback rulebook skill with jev (model typesafe-ai/jev through my vercel ai gateway key), and build a demo page: their name and logo, each dispute with its reason code, the evidence checklist, jev's verdict and probability, and the full response for the strongest one, marked clearly as mock data, ending with "send us your last 3 disputes and we'll run them for real, free."

4. design postcards: front is a screenshot of the demo page, back says "we set up chargeback fighting for [store name]. scan to see which disputes you'd win." plus a qr to the demo page and my number: [your number].

5. show me 3 sample demo pages and 3 sample postcards. after i approve, send the batch through my [lob / postgrid / stannp] account and log every send in a spreadsheet.

6. message me whenever a demo page is opened.

7. when a client connects their [stripe / shopify] account, check for new disputes every day, gather the evidence from their orders, run jev, and show me every response before it's submitted, at least [n] days before the deadline. keep a running total of what's won and send a monthly statement.

rules: never submit a dispute response without my approval. never list evidence the store doesn't actually have. no mail without my batch approval. cap api spend at [\$x] without asking. save this workflow as a skill.

guide 9 of the local business series. next up: bill audits for local businesses.