

Private AI assistant: performance review

Ready for use

We installed an AI chat assistant on a single computer in our own office, then load-tested it to find out how many people it can serve before it slows down. This is what we measured. No customer data left the building at any point, and there is no per-message fee.

Bottom line

18 Sep 2026

The machine comfortably supports a **20-30 person team** for everyday use. Across roughly 2,000 test messages it never returned a single error, never ran out of memory, and never dropped a request it had accepted.

0

FAILED REQUESTS IN 2,000+

0.07s

TYPICAL WAIT BEFORE IT
STARTS REPLYING

~2,000

WORDS PER MINUTE
DELIVERED TO EACH PERSON

74-247

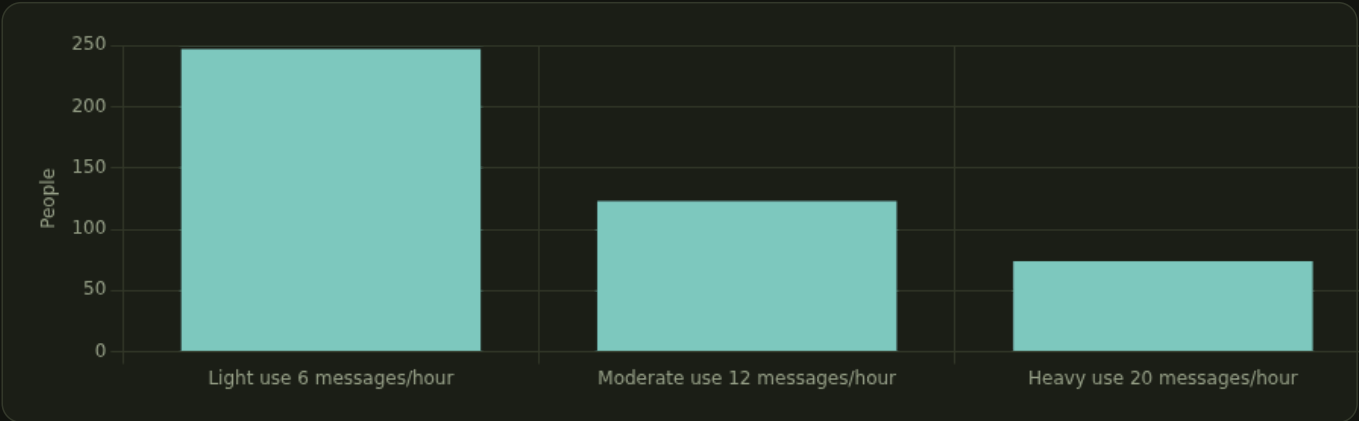
STAFF SUPPORTED,
DEPENDING ON HOW HEAVILY
THEY USE IT

How to read the numbers

TERM	WHAT IT MEANS IN PRACTICE
Wait before replying	How long you stare at the screen after pressing enter, before the first word appears. Under a tenth of a second feels instant.
Words per minute	How fast the answer is written out on screen. A fast human reader manages about 240 words per minute, so 2,000 is roughly eight times faster than anyone can read.
Simultaneous users	People whose messages are being answered in the same moment — not people logged in. In a 30-person office only a handful overlap at any instant.
Context limit	The biggest document the assistant can consider at once. Ours holds about 24,000 words, roughly a 95-page report.

How many staff it can serve

The same machine serves very different headcounts depending on how intensively people use it. These figures are calculated from the measured throughput at a realistic office load, and they assume everyone keeps working at that rate all day, which in practice nobody does.



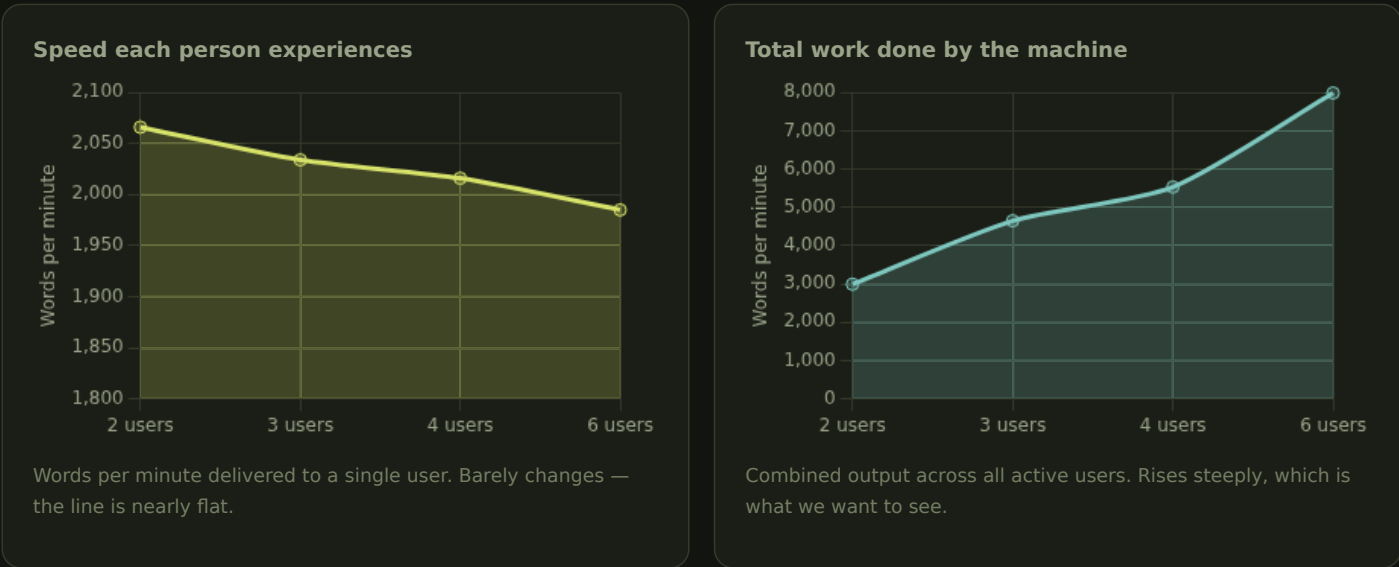
Staff supported at each usage intensity. Derived from the 4-user steady load test (24.8 messages/minute sustained). Source: run 14:37, 18 Sep 2026.

For our current team size, this is not close to the limit

At 30 people sending a message every five minutes, we would be using roughly a quarter of what the machine can deliver.

What happens as more people use it at once

This is the most important chart in the report. As we went from 2 to 6 simultaneous users, each individual’s reply speed dropped by only 4%, while the total work done by the machine nearly tripled. In other words, extra users are served almost for free until the hardware genuinely fills up.



Simultaneous users, sustained load with realistic pauses between messages. Source: run 14:37, 18 Sep 2026.

The worst case: everyone hits enter at the same second

This is a deliberately unfair test — an entire team pressing send in the same instant, with no gaps at all. It is where we expect the experience to degrade, and it does, but predictably and without failures.



Seconds before a reply begins, by number of people sending simultaneously. “Unluckiest” is the 95th percentile — the slowest 1 in 20.
Source: run 14:37, 18 Sep 2026.

Up to 16 at once: still feels immediate

Everyone waits under half a second. All 16 replies finish within 25 seconds.

24-30 at once: most are fine, a few wait

The typical person still starts reading in under half a second, but the unluckiest waits about 7 seconds. All 30 are fully answered within 33 seconds.

Beyond capacity, it queues politely instead of breaking

We pushed 48 messages in at once — more than the system is configured to accept. It answered 32 and asked the other 16 to retry a moment later. Nothing crashed, nothing was silently lost, and no user would see an error page.

Large documents and long-form answers

Two questions matter here: can it read a big document, and can it write a long one? We tested both, and the second only after an earlier version of our tests was found to be skipping it entirely.

Taking in a 90-page report

Reading

5.2s

TO READ ~23,000 WORDS AND ANSWER

We pasted in a document just under the maximum size the assistant can hold. It absorbed the whole thing and responded in about five seconds. We also hid a specific access code at five different depths in an 18,000-word document; it found the code every time.

Producing a 6,000-word document

Writing

2.9 min

TO WRITE ~6,000 WORDS IN ONE GO

Long answers stream out at essentially the same speed as short ones — writing 6,000 words is only 1.5% slower per word than writing 750. Length is not a performance problem; it simply takes proportionally longer.

One clear limit, worth knowing before anyone hits it

The document you send and the answer it writes share a single budget of about 24,000 words. Ask it to write a long report about an already-long document and it will refuse immediately with a clear message, in a fifth of a second, rather than failing halfway through.

Two real faults the testing caught

Both were found before anyone outside the project used the system. Neither would have been visible in casual testing, and the first would eventually have taken the service down during normal business use.

A crash triggered by three colleagues sharing long documents at once

Fixed and re-tested

The original setup used almost all of the graphics card's memory to hold conversations, leaving no room to actually do the work. Three people pasting long documents simultaneously exhausted the memory and stopped the service outright — everyone would have been disconnected, not just those three.

We reduced the memory reservation slightly and reduced how much text is processed per cycle. The exact scenario that previously crashed the service now completes cleanly, and we verified it under harder conditions than the original failure.

Our own tests were quietly measuring nothing

Fixed and re-tested

An early version of the long-document tests asked for big inputs but only ever requested one-sentence answers. The results looked healthy while the hardest case — reading a lot and writing a lot at the same time — was never actually being exercised.

When we fixed that, a second issue appeared: asked to write an exhaustive document, the assistant sometimes stopped after two paragraphs and explained that the task was too large — once even quoting our own instruction to “be concise” back at us. The tests now verify that the amount of text produced matches what was asked for, so this class of false result cannot hide again.

Hardware behaviour under sustained load

We ran the machine flat out for 36 minutes. Nothing overheated, nothing was throttled, and the computer never had to fall back on slow disk-based memory.

71°C

HOTTEST THE GRAPHICS CARD GOT

182W

PEAK POWER DRAW, ABOUT TWO DESKTOP PCS

0 MB

OVERFLOW TO SLOW DISK MEMORY

None

SPEED REDUCTIONS FROM HEAT

Measured continuously during the 36-minute, 43-scenario run. Source: run 14:37, 18 Sep 2026.

For context, the graphics card is rated well above the temperature we reached, and it idles at about 6 watts when nobody is using it. Running this service does not require any change to cooling or power provisioning.

How the testing progressed

Nine test runs were recorded and all of them are kept permanently, so any future change can be compared against today’s baseline rather than argued about.

RUN	SCOPE	WHAT IT ESTABLISHED
12:11 and 12:13	13 scenarios	First exploratory measurements to confirm the assistant worked and was being measured correctly.
12:52	33 scenarios, 505 messages	First full test. Confirmed everyday performance, and led to the memory crash being found and fixed.
13:50	43 scenarios, 522 messages	Added long-answer tests. Caught the assistant refusing to write long documents, which invalidated part of the run.
14:37	43 scenarios, 520 messages	The clean reference run all figures in this report come from. Zero errors, zero missed targets, all 70 accuracy checks passed.
4 short runs	1-24 scenarios	Verification checks used while fixing the two faults above.

What we have not proven

These are genuine gaps rather than formalities. None of them block rolling the system out to the team, but each would matter for a specific kind of use, and it is better to name them now.

NOT YET ESTABLISHED	WHO IT WOULD AFFECT
Answer quality was not assessed	Everything here measures speed and reliability. We confirmed the assistant can find a specific fact inside a long document, but nobody has graded the usefulness or accuracy of its writing.
Long answers are likely repetitive	To force it to produce 6,000 words on demand, we allowed near-duplicate sections. Treat the long-form results as a speed measurement, not evidence of good long-form writing.
Only short bursts, not a full working day	The longest continuous test was about three minutes of heavy load. We have not run a full day to see whether performance drifts.
Mixing long analysis with quick chat is slow	While two people run long document analyses, a colleague asking a quick question waits about 2.8 seconds instead of 0.1. This is the one result most likely to generate complaints, and it is worth deciding whether analysis work should be scheduled separately.
No authentication on the direct interface	The chat interface requires a login. The underlying technical interface does not, so anyone already on our private network can use the hardware freely. That is acceptable internally, but it is not a boundary we should rely on.

Recommendation

Proceed with rolling this out to the team. The hardware has substantial headroom at our size, it fails gracefully rather than catastrophically when overloaded, and the two faults found during testing are fixed and verified.

Two things worth an explicit decision. First, whether heavy document-analysis work should be kept separate from everyday chat, since the two interfere noticeably. Second, whether we want a proper access key on the technical interface before the number of people on the network grows.

Assistant: Qwen 3.5 (9 billion parameters, NVFP4 compressed). Hardware: one NVIDIA RTX 5060 Ti 16GB graphics card, AMD Ryzen 5 9600X processor, 30GB memory. Serving software: vLLM 0.26.0.

All figures from benchmark run suite-20260918-143709 unless stated otherwise. Word counts converted from tokens at 0.75 words per token. Full technical reports for every run are archived and browsable.