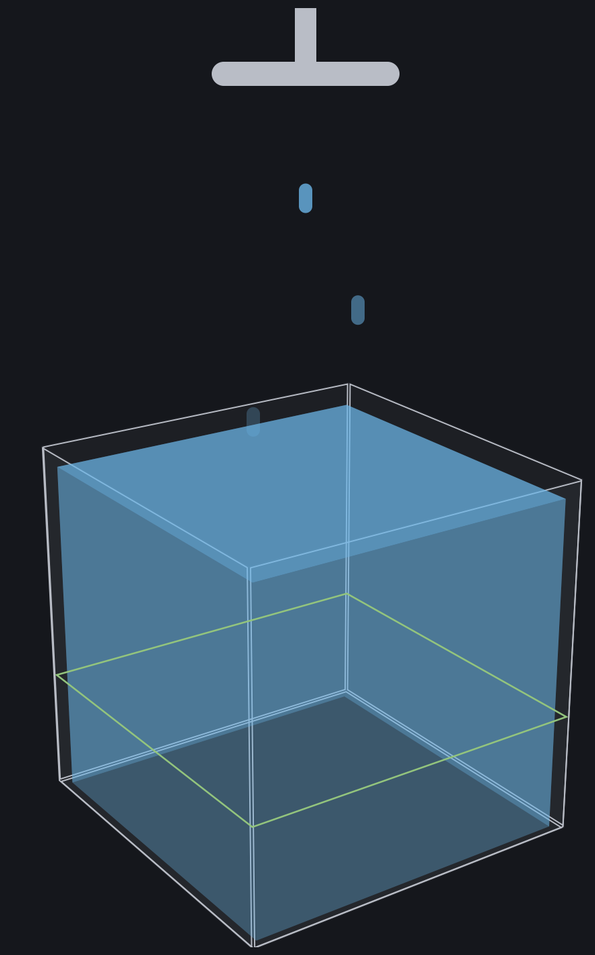
python project / sustainability

Shower Water Tracker

I take really long showers. This little program shows me how much water that is.



one shower fills this: 150 L
green line = my goal, 50 L

Ricardo Lua Villanueva

November 2026

part one: my problem

How much water is one shower?

15

minutes in one shower

10

liters every minute (my estimate)

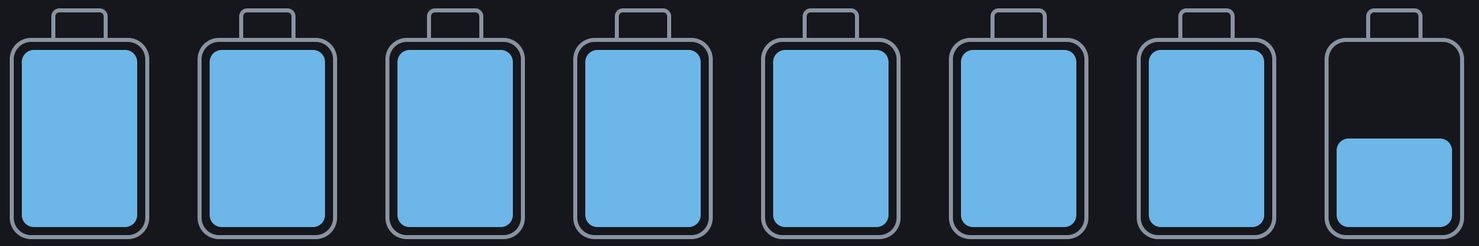
150

liters each time

50

liters is my goal (a 5-minute shower)

15 × 10 = 150 liters. That’s more than seven of those big 20-liter water jugs, every single morning.



7.5 jugs of 20 L

part one: my problem

So what’s the problem?

what happens

My showers last about 15 minutes. I just lose track of time in there.

where and when

At home, every morning before school.

what it causes

About 150 liters go down the drain each time. That’s over 1,000 liters a week.

why it happens

I never time my showers, and I had no clue how many liters that was.

what I want

To see my number every day and get it under 50 liters.

my problem statement

“In my daily life, taking 15-minute showers happens, which causes me to use about 150 liters of water each time. This happens because I don’t time my showers and I don’t know how many liters I use, and I would like to improve this situation by tracking my shower minutes every day and checking if I stay under my goal of 50 liters.”

part one: my problem

How do I measure that?

```
liters = minutes * liters_per_minute
```



01 time it

I time my shower with my phone.

02 multiply

Minutes times liters per minute. That gives me the liters.

03 compare

Then I check if I stayed under 50 liters.

part two: looking for ideas

I asked ChatGPT four things

1. help me understand my problem

It said my real problem isn’t only the water. It’s that I don’t measure anything.

2. what could affect it? what am I assuming?

The minutes and the water flow matter most. And I’m only guessing the 10 liters per minute.

3. what data do I need? how do I keep it simple?

The day, the minutes, the water flow and the liters. And only count showers, nothing else.

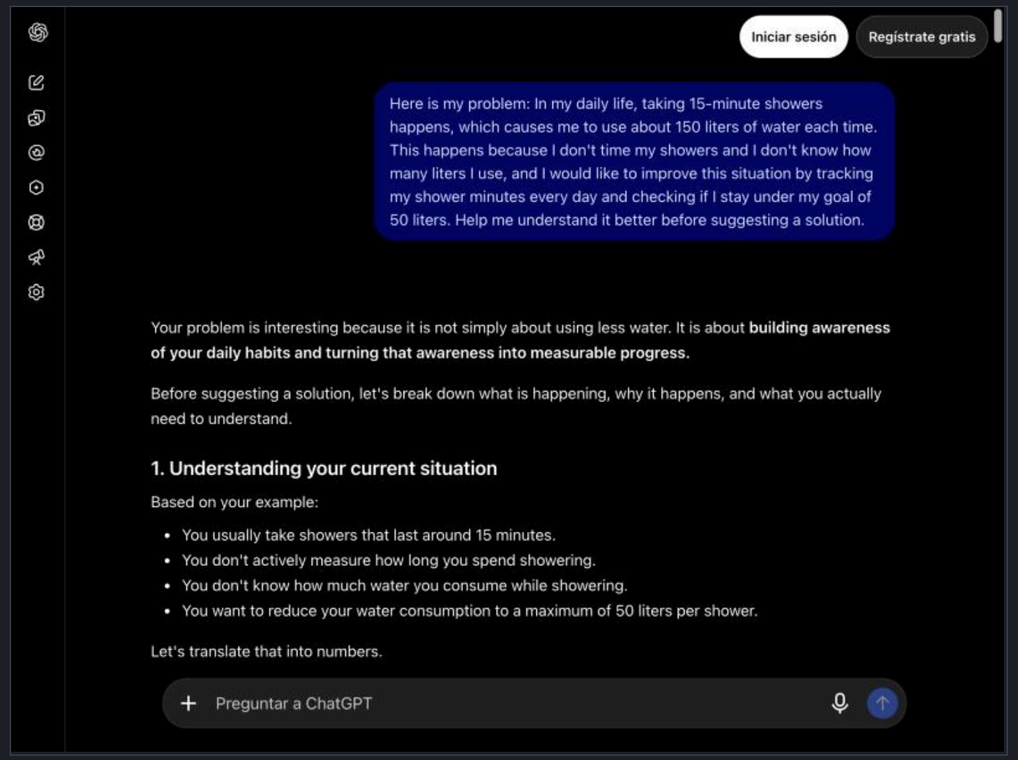
4. give me three ideas I could build in Python

A daily tracker, a 5-minute challenge and a weekly report.

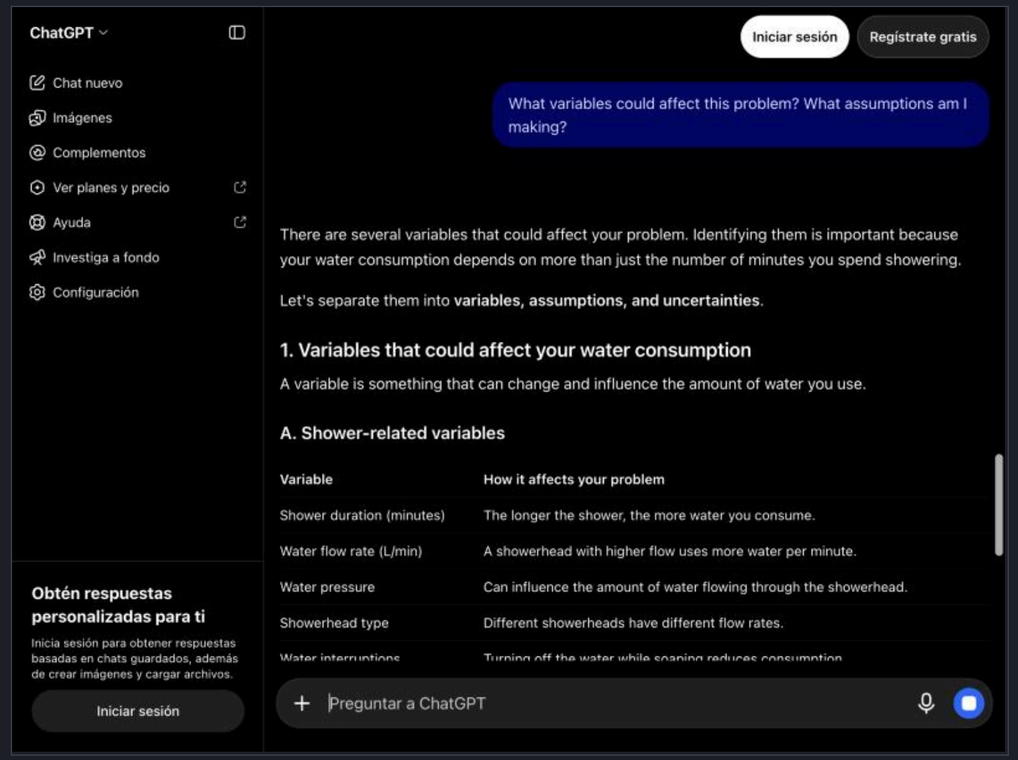
New chat, no account, November 18, 2026. The full chat is in my document.

part two: looking for ideas

Proof: the real chat (1 of 2)



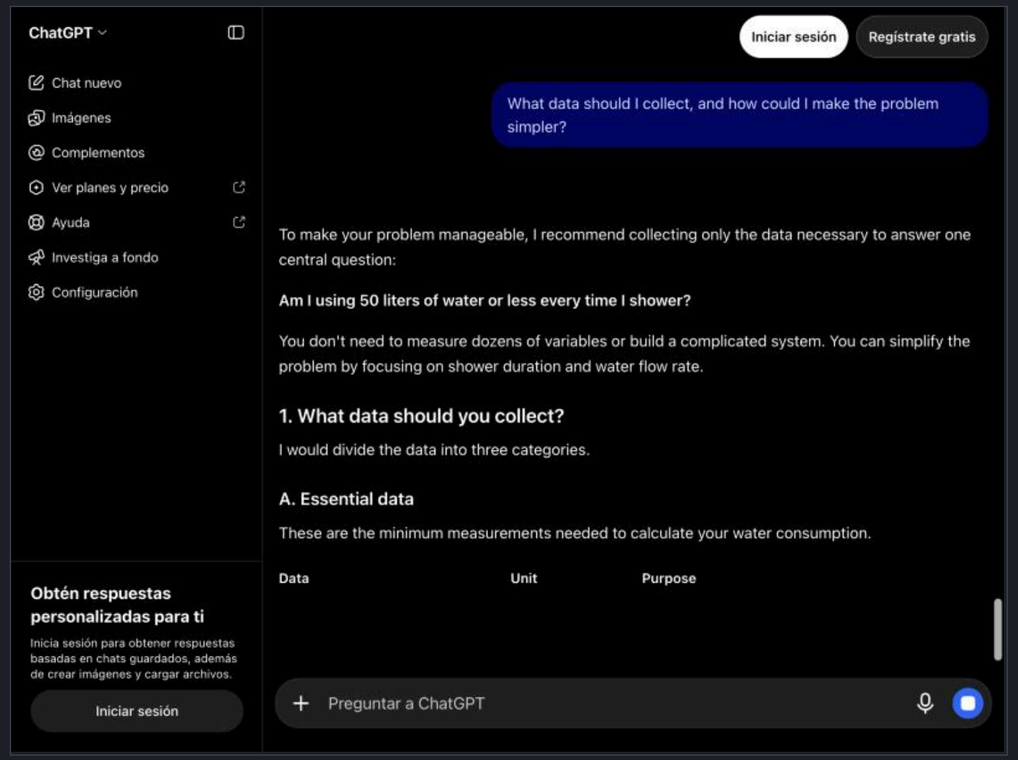
question 1



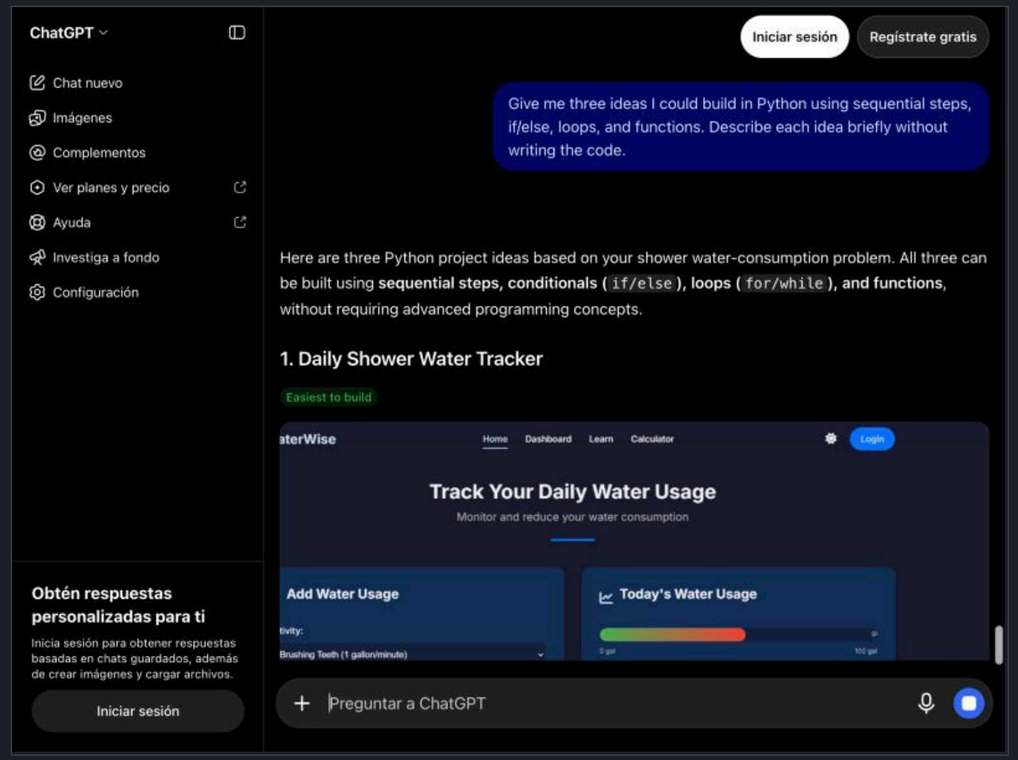
question 2

part two: looking for ideas

Proof: the real chat (2 of 2)



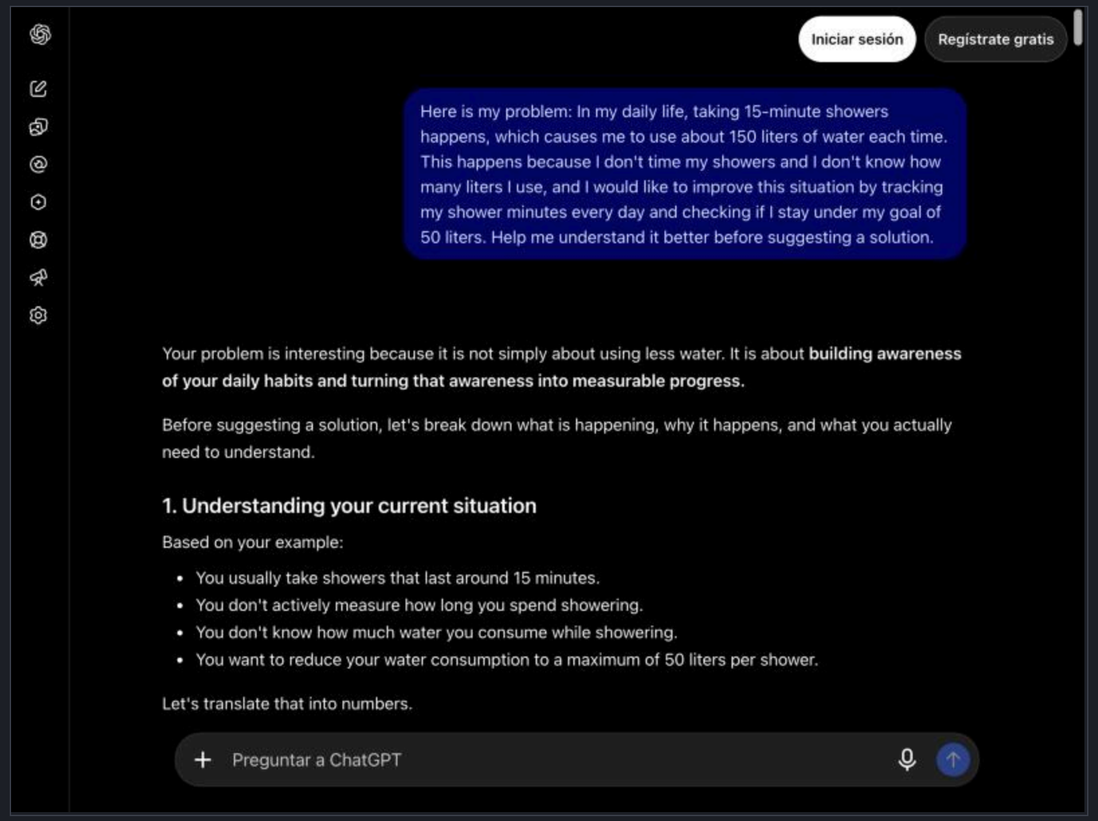
question 3



question 4

part two: looking for ideas

Replay of the chat



This clip scrolls down through my real screenshots of the chat: my four questions and the start of each answer.

The three dots mark where a long answer keeps going and the clip jumps ahead.

The chat had no account, so it can’t be opened again. That’s why this is a replay and not a live recording.

screenshots from November 18, 2026

part two: looking for ideas

It gave me three ideas

idea 1

Daily Shower Water Tracker

I type my minutes each day. It tells me the liters and if I passed my goal.

Needs: my minutes, the liters per minute and my goal.

easy to code, shows the liters

idea 2

Shower Time Challenge

It gives me a 5-minute limit and checks it for a week.

Needs: only my minutes.

easy too, but no liters

idea 3

Weekly Water Consumption Analyzer

It saves the whole week in a list and makes a report.

Needs: 7 days of minutes, the liters per minute and my goal.

most complete, hardest to code

part two: looking for ideas

I went with idea 1

The Daily Shower Water Tracker. It's simple, and it shows the one thing I didn't know: the liters.

why not idea 2

It only looks at the minutes. But my whole problem is that I don't know the liters.

why not idea 3

It needs lists and more functions. We've barely used those in class.

what I kept

ChatGPT told me to pick idea 3. I didn't, but I took one thing from it: the total at the end.

new statement

So this is what my program does

what I type

- My shower minutes, day by day
- The liters per minute of my shower
- My goal in liters

what it does

- Multiplies minutes by liters per minute
- Checks the liters against my goal
- Adds everything up

what I see

- The liters for each day
- A message: under my goal or over it
- The total at the end

part three: the plan

First I split it into small parts

Decomposition = cutting one big problem into small ones.

1. Ask for the data: the days, the water flow and my goal.
2. Go day by day and ask for my shower minutes.
3. Do the math: minutes times liters per minute. That’s my function.
4. Check if that day was under my goal.
5. Add it all up and show the total.

part three: the plan

What I need, and what I can ignore

Abstraction = keeping only the details that change the result.

I need

- How many days I'm tracking
- My minutes for each day
- How many liters per minute my shower uses
- My goal

I can ignore

- The water pressure and the temperature
- Whether I washed my hair or not
- What time I showered
- The rest of the water we use at home

part three: the plan

The steps, in order

Algorithm = the recipe the program follows, step by step.

1. Ask how many days I want to track.
2. Ask for the liters per minute and for my goal.
3. Start the total at 0.
4. For each day, ask for my shower minutes.
5. Use the function to get the liters of that day.
6. Add those liters to the total.
7. Under my goal, or exactly on it? Say so. Over it? Warn me.
8. After the last day, show the total. Steps 4 to 7 repeat for every day.

part three: the plan

What keeps repeating?

Pattern recognition = spotting what repeats, so I only write it once.

every day

I do the exact same thing for each day. So I wrote it once and put it in a for loop.

the same math

The liters are always minutes times liters per minute. So that went into a function.

the same question

Every day ends with the same question: did I pass my goal? That's my if/else.

part three: the plan

IPO table

IPO = Input, Process, Output. What goes in, what happens, what comes out.

Input	Process	Output
days: how many days I track	1. Repeat these steps for each day.	The liters I used each day
liters_per_minute: how fast the water comes out	2. Multiply minutes by liters per minute.	A message: under my goal or over it
daily_goal: the liters I don’t want to pass	3. Compare the liters with my goal.	The total liters at the end
minutes: asked once for each day	4. Add the liters to the total.	

part three: the plan

My variables

Variable name	Data type	Why I chose this type
days	int	You can’t track half a day.
liters_per_minute	float	It could be something like 9.5.
daily_goal	float	Same thing, my goal could have decimals.
minutes	float	A shower can be 7.5 minutes. I learned that the hard way.
liters	float	It’s a float times a float.
total_liters	float	It’s all the liters added up.
day	int	It just counts: 1, 2, 3...

part four: build and test

The code, part 1: function and inputs

```
def calculate_liters(minutes, liters_per_minute):
    liters = minutes * liters_per_minute
    return liters

# Input
days = int(input("How many days do you want to track? "))
liters_per_minute = float(input("Liters per minute of your shower: "))
daily_goal = float(input("Your daily goal in liters: "))

total_liters = 0
```

The function does the math. The three input() lines are the questions it asks me.

part four: build and test

The code, part 2: the loop

```
# Process
for day in range(1, days + 1):
    minutes = float(input(f"Shower minutes on day {day}: "))
    liters = calculate_liters(minutes, liters_per_minute)
    total_liters = total_liters + liters

    if liters <= daily_goal:
        print("You used", liters, "liters. You stayed under your goal!")
    else:
        print("You used", liters, "liters. You went over your goal.")

# Output
print("Total liters:", total_liters)
```

part four: build and test

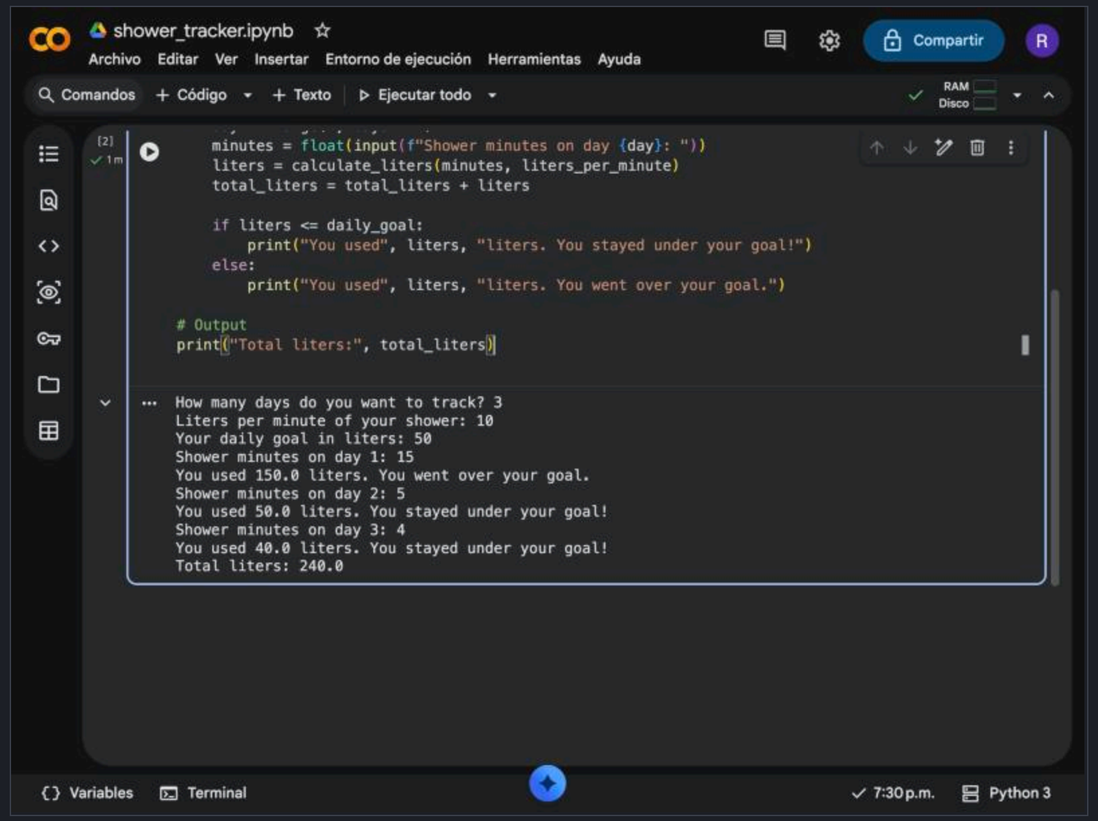
Here it is running

```
How many days do you want to track? 3
Liters per minute of your shower: 10
Your daily goal in liters: 50
Shower minutes on day 1: 15
You used 150.0 liters. You went over your goal.
Shower minutes on day 2: 5
You used 50.0 liters. You stayed under your goal!
Shower minutes on day 3: 4
You used 40.0 liters. You stayed under your goal!
Total liters: 240.0
_
```

Yellow is what I typed. Day 2 is exactly 50 liters, and that still counts as reaching my goal.

part four: build and test

The same run, recorded in Colab



1. I press run in Google Colab.
2. It asks for the days, the liters per minute and my goal.
3. Then it asks for my minutes, day by day.
4. At the end it prints the total: 240 liters.

screen capture of my notebook, November 23, 2026

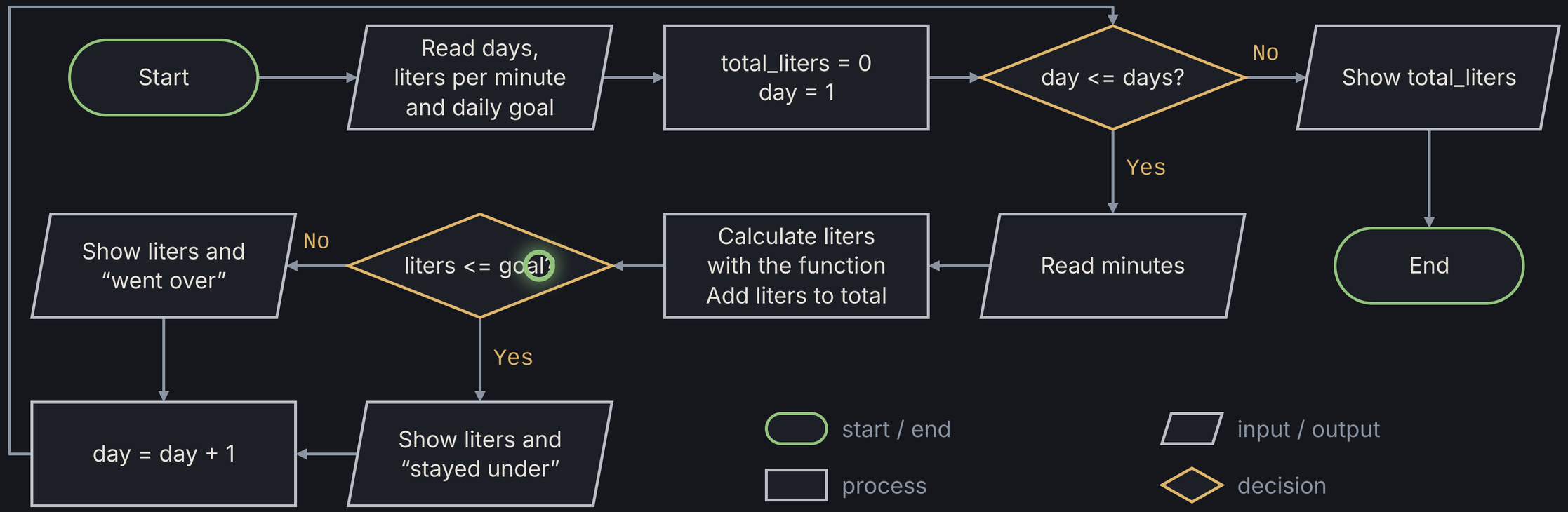
part four: build and test

Pseudocode

```
FUNCTION calculate_liters(minutes, liters_per_minute)
    liters = minutes * liters_per_minute
    RETURN liters
READ days
READ liters_per_minute
READ daily_goal
total_liters = 0
FOR day FROM 1 TO days
    READ minutes
    liters = calculate_liters(minutes, liters_per_minute)
    total_liters = total_liters + liters
    IF liters <= daily_goal
        SHOW liters, "You stayed under your goal!"
    ELSE
        SHOW liters, "You went over your goal."
SHOW total_liters
```

part four: build and test

Flowchart



part four: build and test

Bug 1: it asked for “day 0”

FIXED

Step	What happened
What I tested	A normal run with 3 days.
What I expected	The first question should say “day 1”.
What happened	It said “Shower minutes on day 0”.
Why	range(days) starts counting at 0, not at 1.
What I changed	I wrote range(1, days + 1).
After the fix	Now it asks for day 1, day 2 and day 3.

part four: build and test

Bug 2: it crashed with 7.5

FIXED

Step	What happened
What I tested	I typed 7.5 as my shower minutes.
What I expected	75 liters.
What happened	The program crashed with a ValueError: invalid literal for int().
Why	int() can’t read a number with decimals.
What I changed	I used float() instead of int() for the minutes.
After the fix	It shows 75.0 liters.

Bug 3: 50 liters counted as “over”

FIXED

Step	What happened
What I tested	A 5-minute shower, with a goal of 50 liters.
What I expected	“You stayed under your goal!”
What happened	It said “You went over your goal.”
Why	I used <, and 50 is not less than 50.
What I changed	I changed < to <=.
After the fix	Now exactly 50 liters counts as reaching my goal.

part four: build and test

My code and my videos

```
colab_link = "https://colab.research.google.com/drive/1jHE58xO5bDxpyxoVrNmUhL8K1j8lOhUU"
video_1 = "[link: I run the program and explain the result]"
video_2 = "[link: I show the bugs and how I fixed them]"
```

No Arduino or micro:bit here. It all runs in Google Colab.

part five: benefits and impacts

Is it actually a good solution?

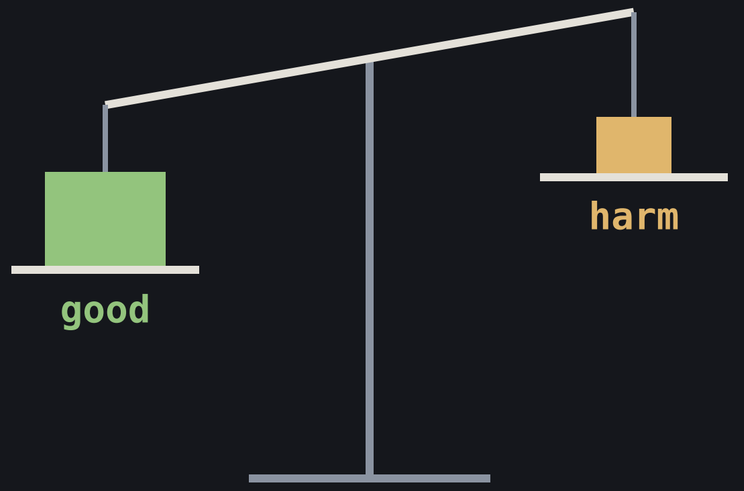
MOGO = Most Good, Least Harm. I asked ChatGPT to check my program with it.

the good

- I finally know how much water I use
- Less water, less energy to heat it
- It pushes me to take shorter showers
- It’s free and easy to use

the not so good

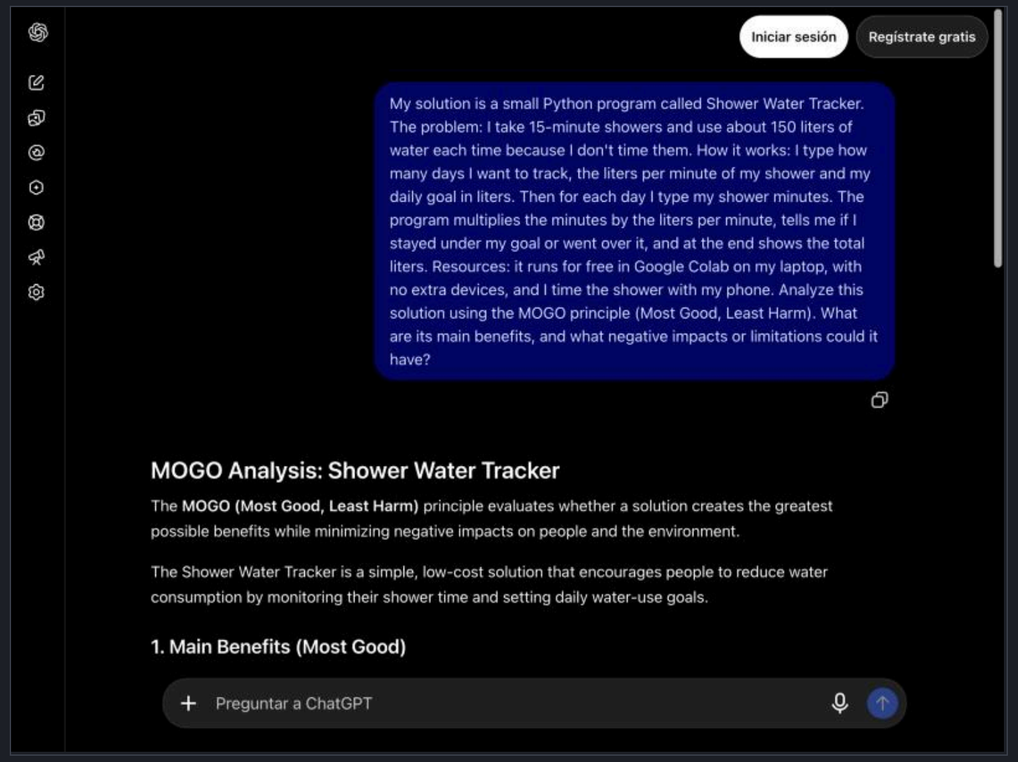
- It only estimates the water
- I type the minutes, so I can forget or cheat
- It needs a laptop and internet
- It only counts showers



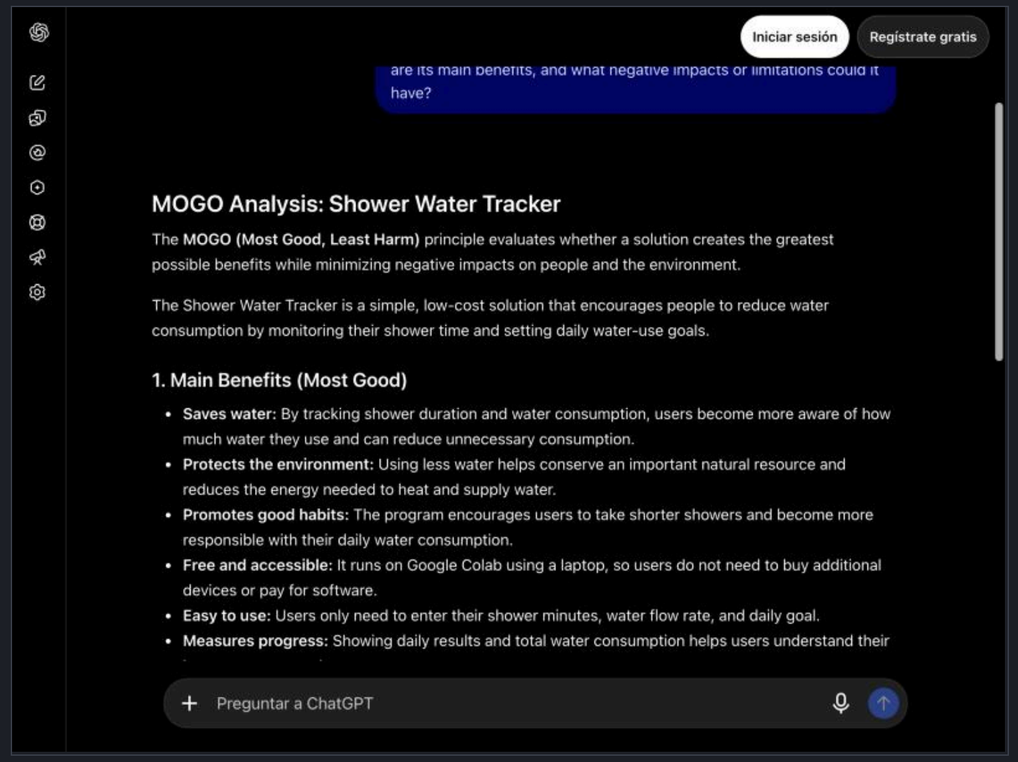
Its verdict: more good than harm.

part five: benefits and impacts

Proof: the MOGO chat (1 of 2)



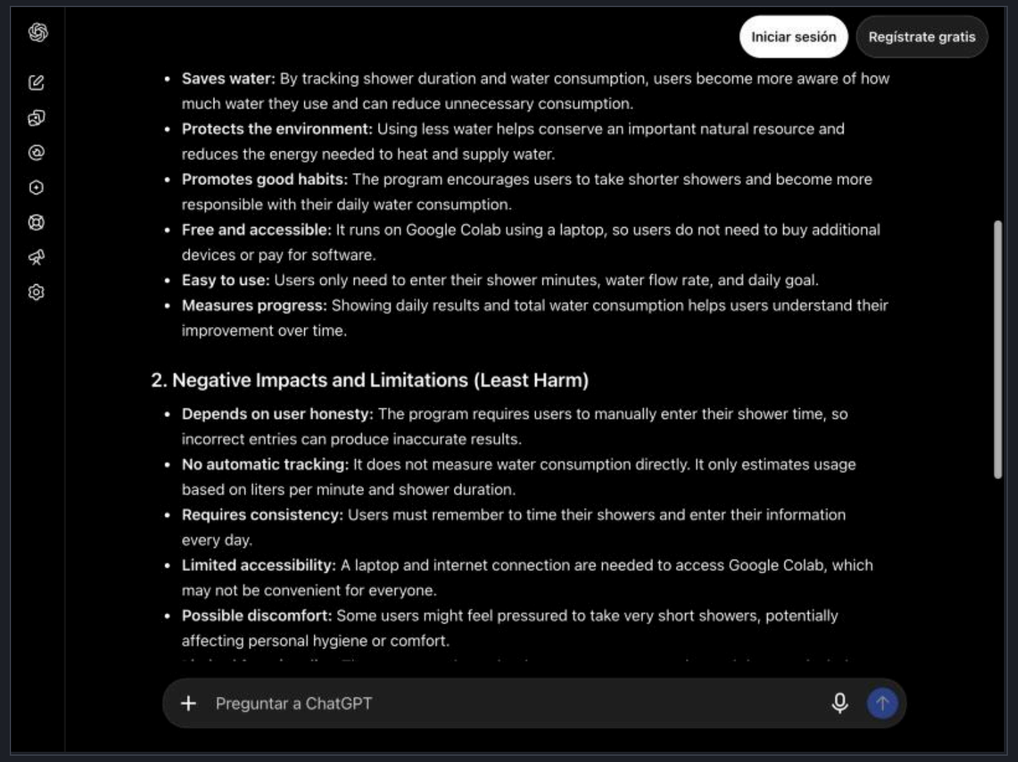
what I asked



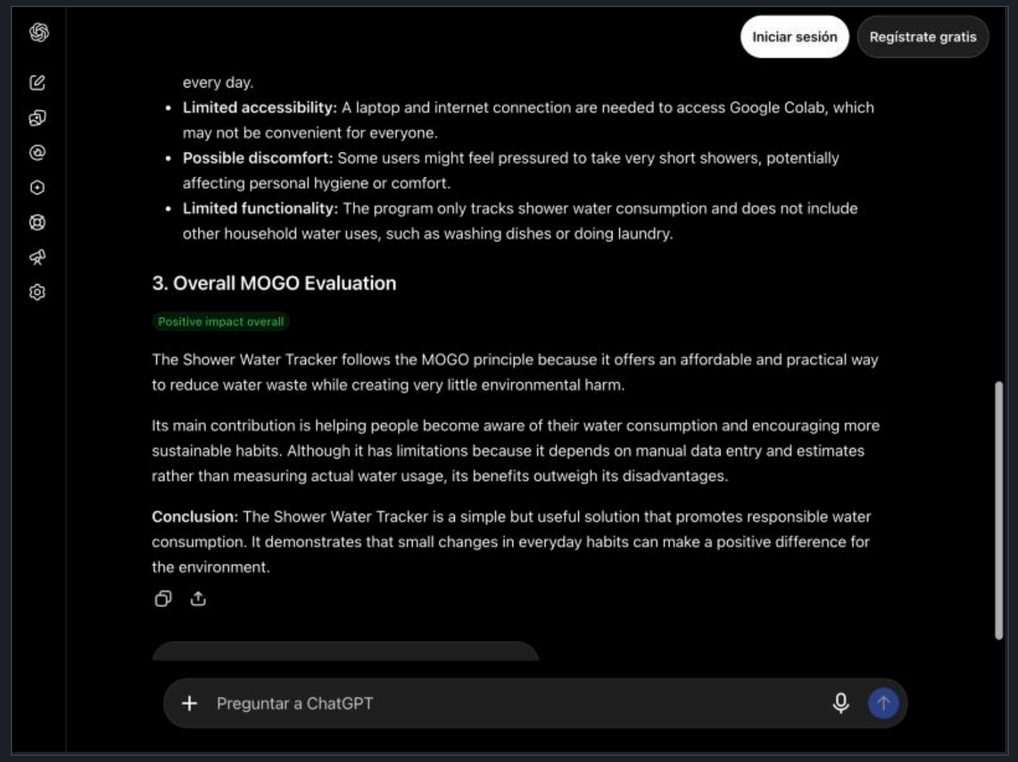
the good

part five: benefits and impacts

Proof: the MOGO chat (2 of 2)



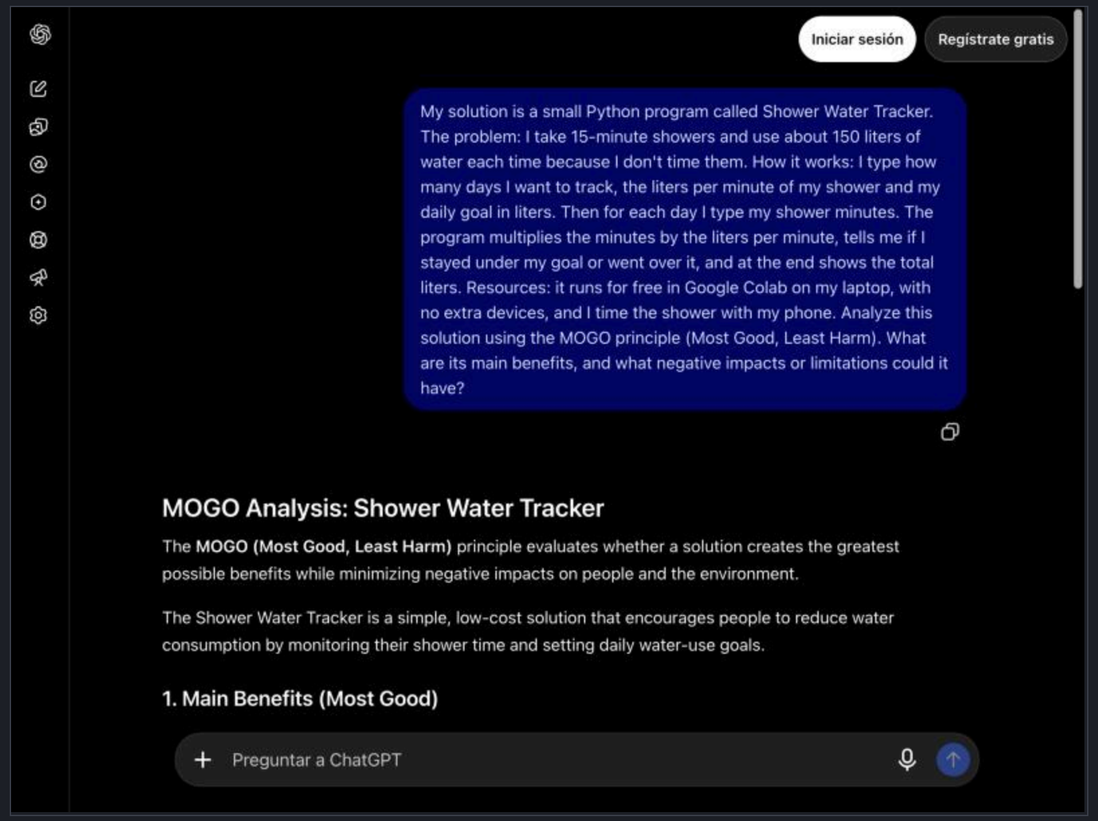
the not so good



its verdict

part five: benefits and impacts

Replay of the MOGO chat



My question and the whole answer, scrolled from top to bottom. It is put together from my real screenshots.

Same thing here: the chat had no account, so I can’t open it again to record it live.

screenshots from November 18, 2026

part five: benefits and impacts

What I think (1 of 2)

What is the greatest benefit of my solution?

Now I can actually see my liters. Once you see 150 next to a goal of 50, you want to hurry up.

What harm or limitation could it have?

It’s only an estimate. If my shower doesn’t really use 10 liters per minute, every number is off. And I type the minutes myself, so I can forget.

Does it use any unnecessary resources?

Not really. No extra devices, just my laptop and Colab. Those do use some energy and internet, so I’d type the whole week in one go instead of opening it every day.

part five: benefits and impacts

What I think (2 of 2)

How could I make it simpler?

Keep the water flow and my goal fixed in the code. Then it only asks for the minutes.

Did I use AI responsibly? Why?

I used AI for a big part of this project: to understand my problem, get ideas, help me write and organize it, and for the MOGO analysis. I think that’s only responsible if I say so, check that the answers make sense and can explain my program on my own. I didn’t put personal data in the chats.

What environmental footprint did using AI have in my process, and what could I do to reduce it next time?

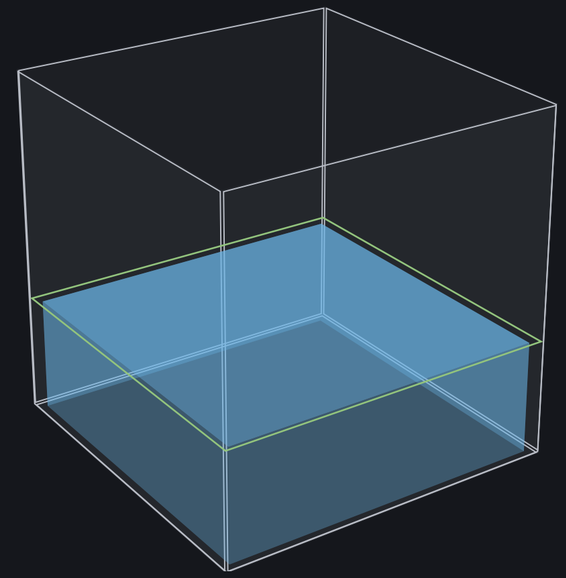
Every question to an AI uses electricity and water in a data center. I asked five questions in two chats, plus the AI help with the writing. I can’t say the exact amount. Next time: fewer, clearer questions, and I do the easy parts myself.

the end

Where I am now

```
[x] part one    define the problem
[x] part two    look for ideas and pick one
[x] part three  plan how the program works
[x] part four   write the code and fix the bugs
[x] part five   check the good and the not so good
[ ] to do       paste my Colab link and record my two videos
```

Next: measure the real water flow of my shower and try it with my own numbers for a week.



the plan: from 150 L down to 50 L