

...
**MICROSERVICES,
¿NIGHTMARE OR
BLESSING?**

A Guide for angry Devs



About me

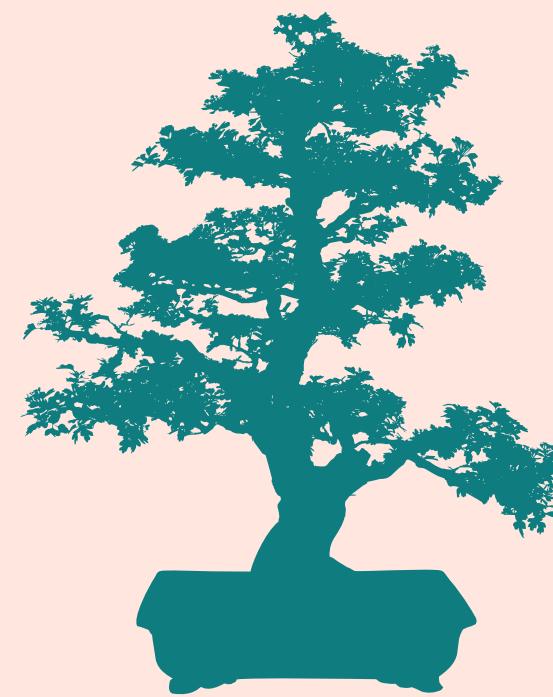
DEVELOPMENT LEAD AT
PROCIBERNÉTICA





I love plants, this is my goal... or something like this...

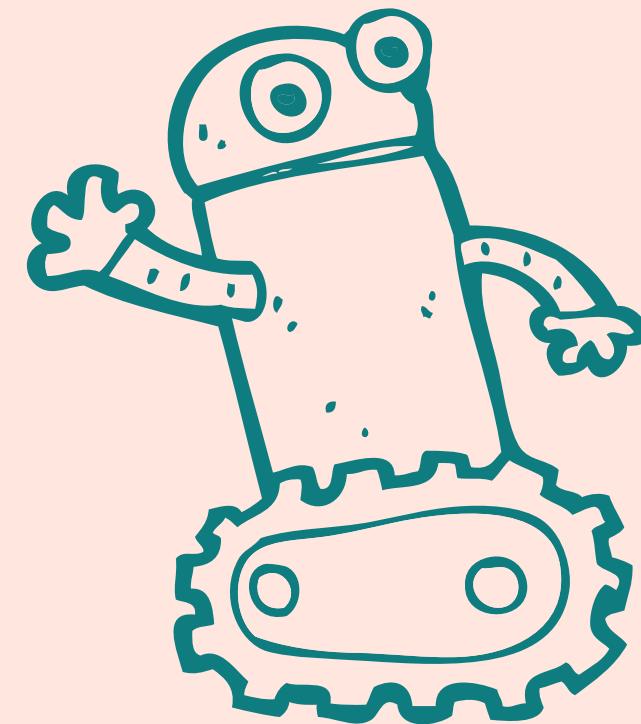
But I still need some things...



Learn a lot
about plants



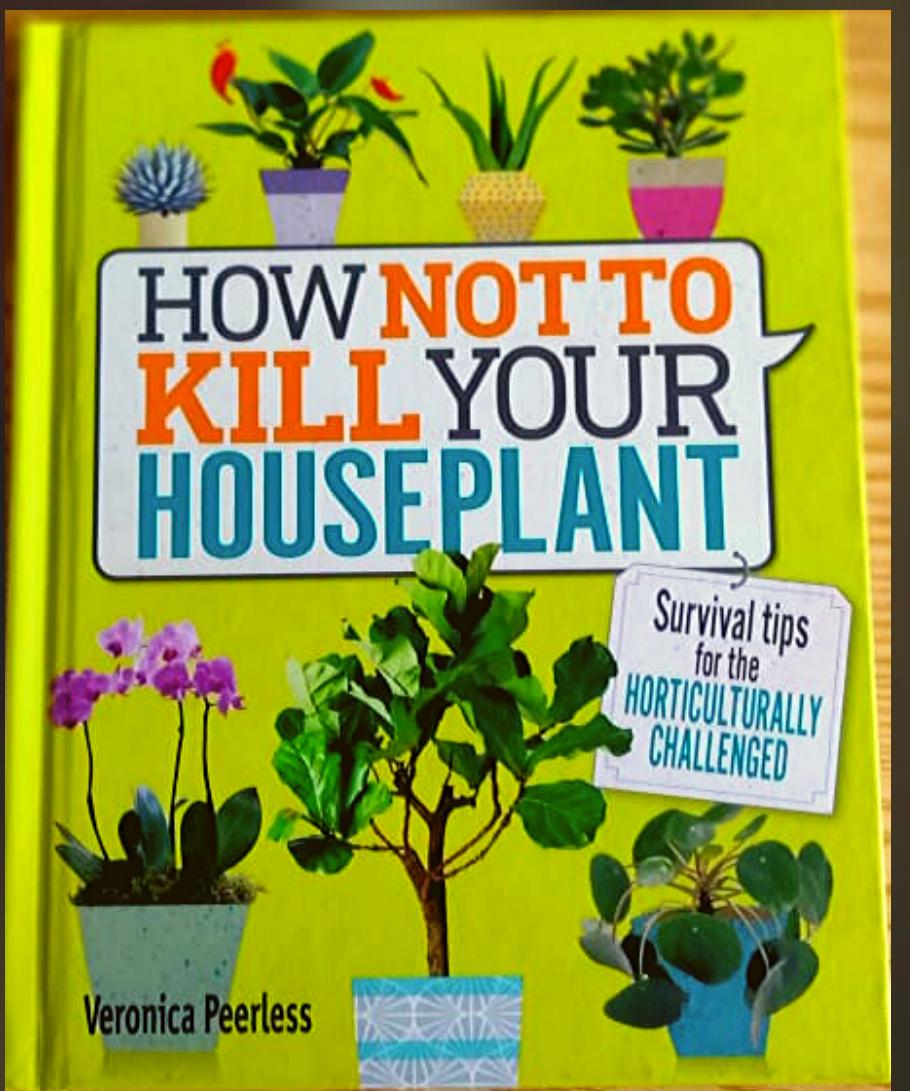
Buy a house



Get some help...

1. Learn a lot about plants

I'VE BEEN READING THIS...

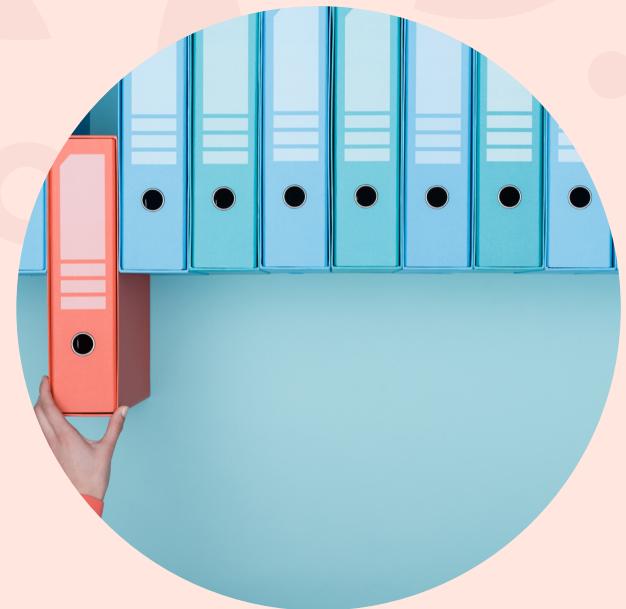




2. Buy a house

:(
:(

3. Get some help



DATABASE



HOW TO NOT KILL
MY PLANTS

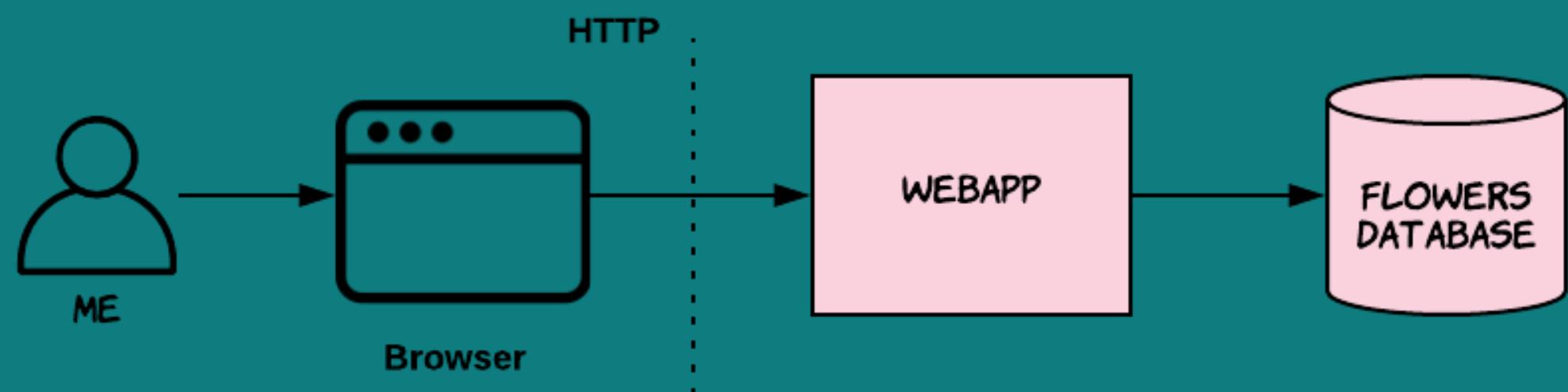


INTERFACE

Notifications

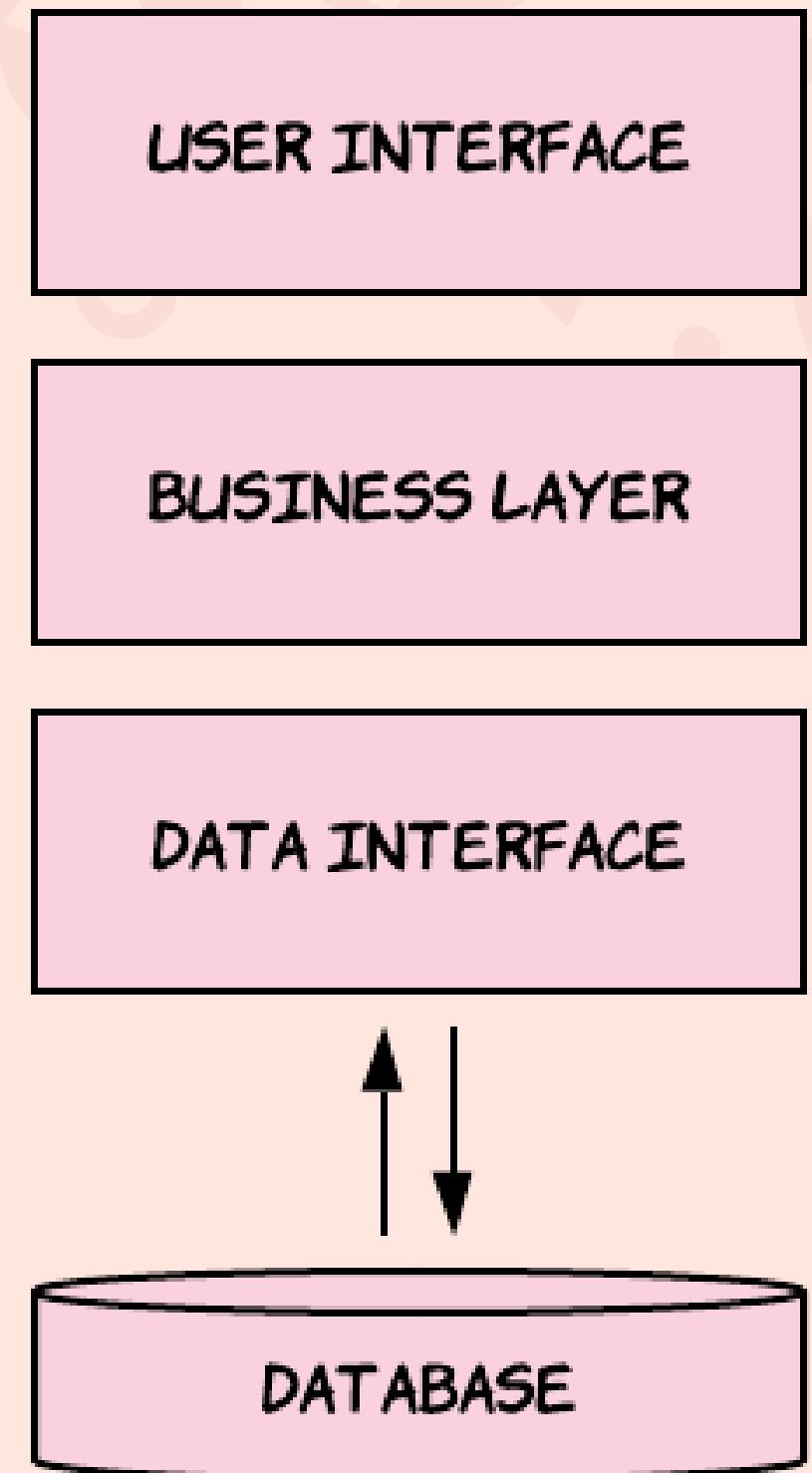
Plant
information

Front End



My monolith

- CI/CD
- EASY TO UNDERSTAND
- MAINTAINABLE
- SIMPLE TO TEST
- SIMPLE TO SCALE

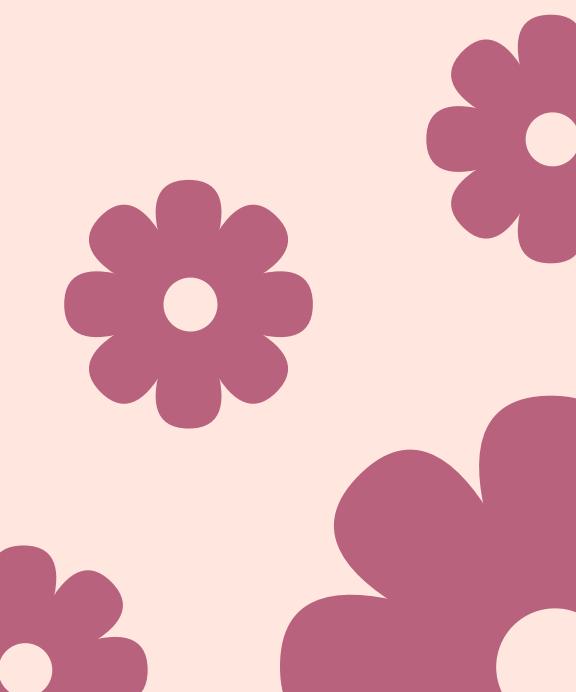




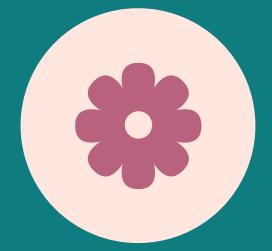
Lets talk about SOA

SERVICE-ORIENTED ARCHITECTURE

- It logically represents a business activity with a specified outcome.
- It is a black box for its consumers.
- It may consist of other underlying services (hierarchy)



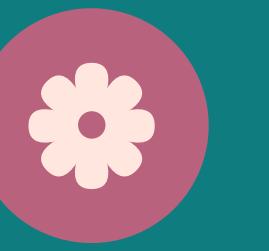
From SOA to MSA



DEPENDENCY

TO

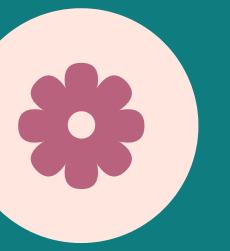
INDEPENDENCY



SERVICES

TO

MICROSERVICES



HIGH COUPLING

TO

LESS COUPLING

What is a Microservice?

- JON EAVES

" The service can be rewritten and redeployed in 2 weeks. "

- WERNER VOGELS

" It must be possible to feed a team that maintains a service with two pizzas. "



FOR ME...

Is a small part of code that is:

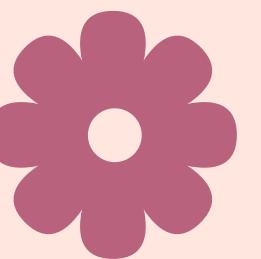
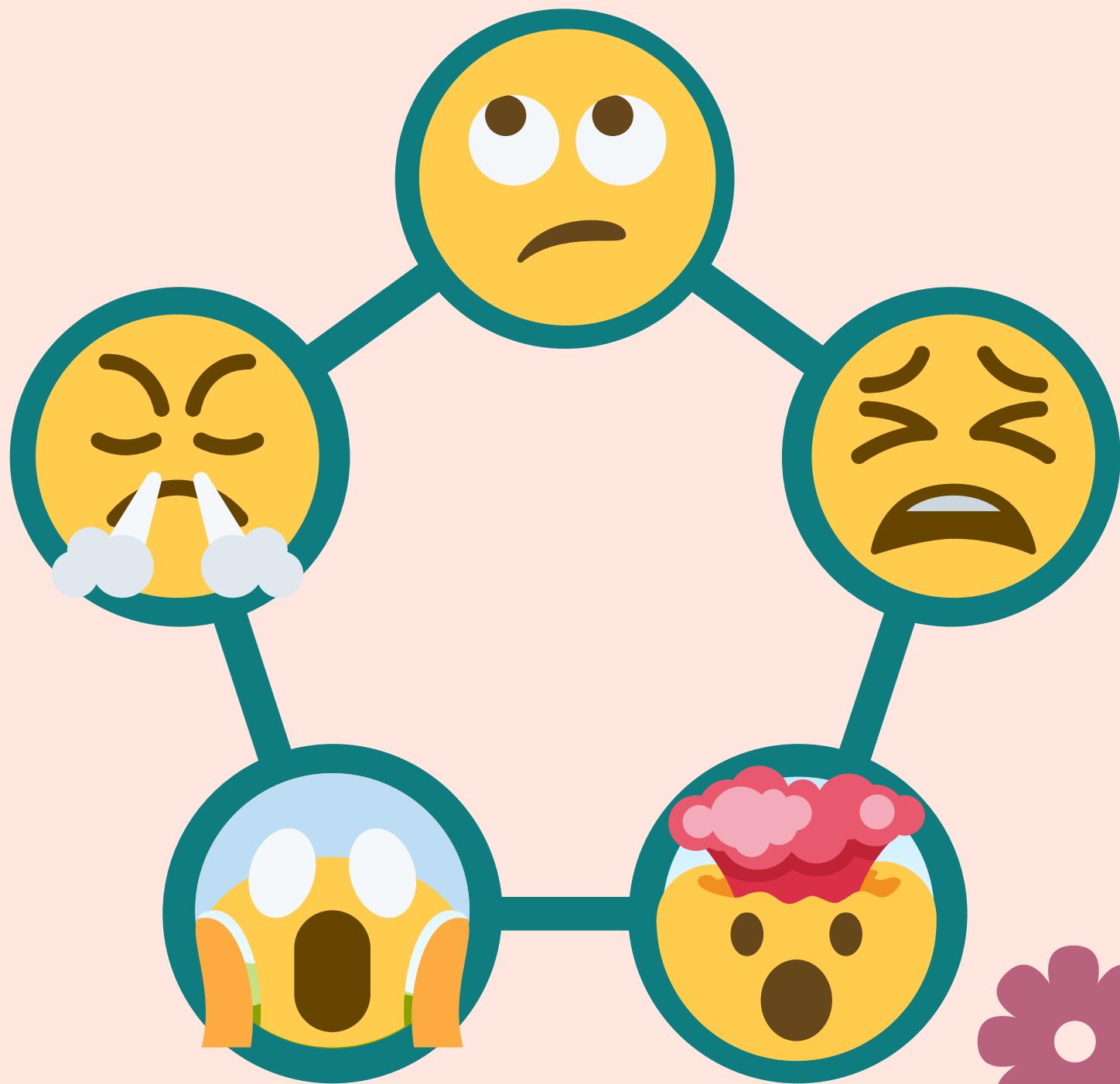
- Independent
- Meets a goal
- Have an interface to be consumed
- Is a business need

- NATHALY VILLAMOR

What means to have a MSA?

WHY PEOPLE HATES MSA?

- High complexity
- Double of everything
- Documentation
- Upgrades
- Maybe more people
- Maybe more money





Everything works
- every body
loves you



You add CI /CD
to have peace



First code
duplication, don't
fool yourself,
you'll do it again.



First Bug, you
have the
double of bugs
now



Business needs,
it'll happen
again, I
promise you.



You realize that
you need to
monitorize two
things now



There are new
updates in the
technology you
are using



You are
spending more
money than
you expected



Your main app
dies after your
microservice
dies



Now you
regret and
hate
microservices

Is there something good about MSA?

AGILITY

You can add features faster

SCALING

Scale is easier, you have to scale only the critical part

CHOOSE TECH

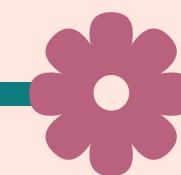
You can use the technology that fits better with your problem

STRONG

Your app is stronger now that you have isolated functionalities.

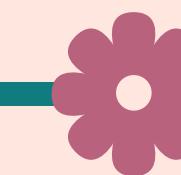
My advice could be...

RUN, RUN B**CH



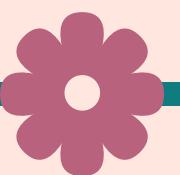
NEVER, NEVER

Start your
architecture thinking
on microservices.



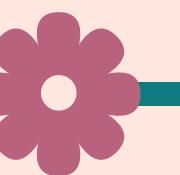
BE SURE

Your microservice
should accomplish
only one goal, only
one functionality,
only one!



DO NOT

FORGET
You are adding
latency when you
add a microservice,
It should worth.



ASK YOUR
SELF

Is it really necessary?
Is it really necessary?
Is it really necessary?

Is the microservice a business **need**?

Is it going to be written in a **different** language?

Can my team **deal** with the new challenge?

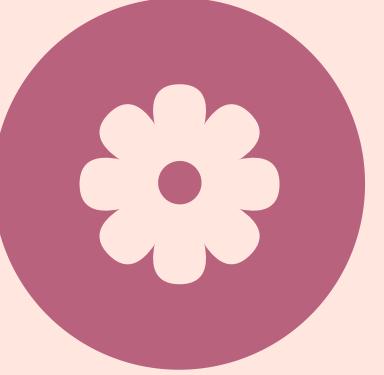
Which is going to be my communication protocol?

Can I **pay** extra money for the new microservice?

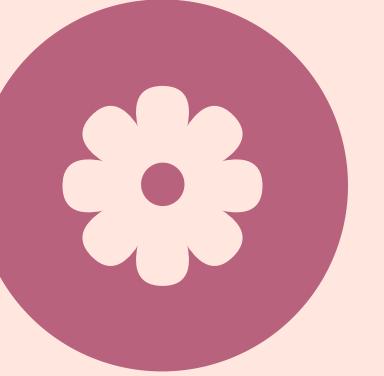
Can I handle the new technical **debt**?

How hard is it to **find** devs to maintain my new microservice?

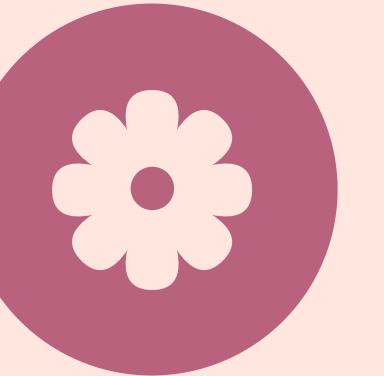
Is it a good **moment** to add a microservice?



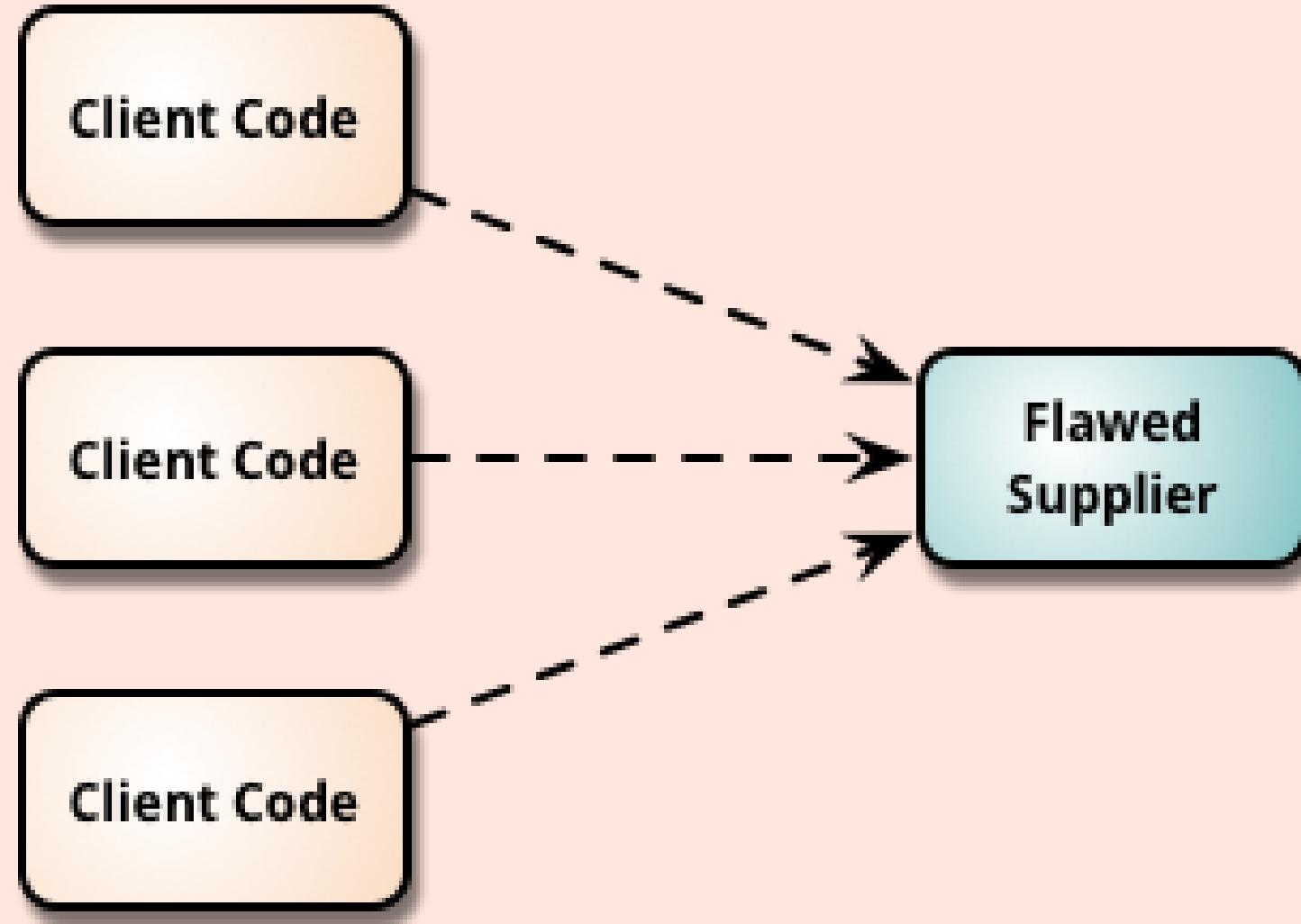
Integration tests



Measure your
requests



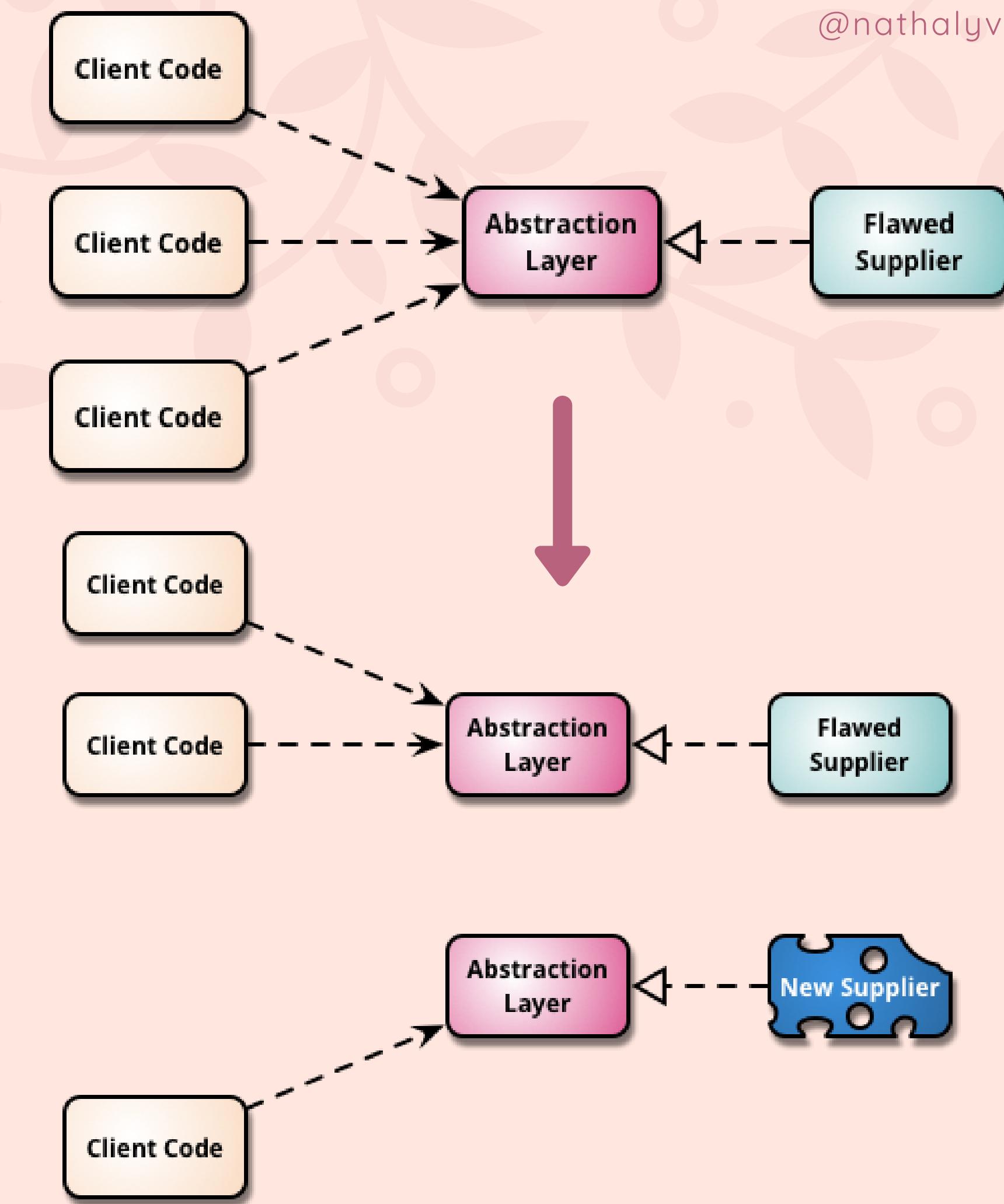
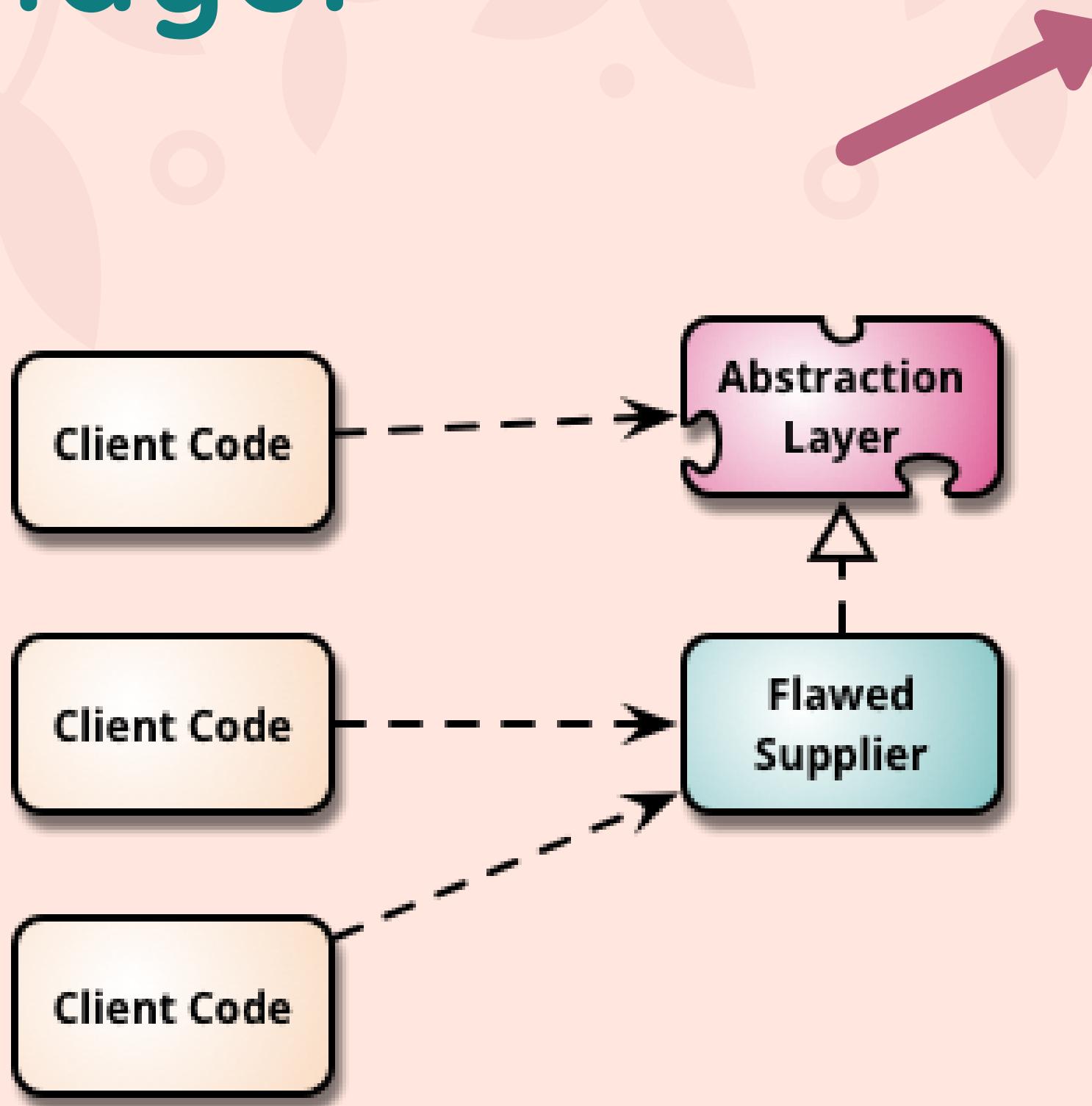
Add
documentation

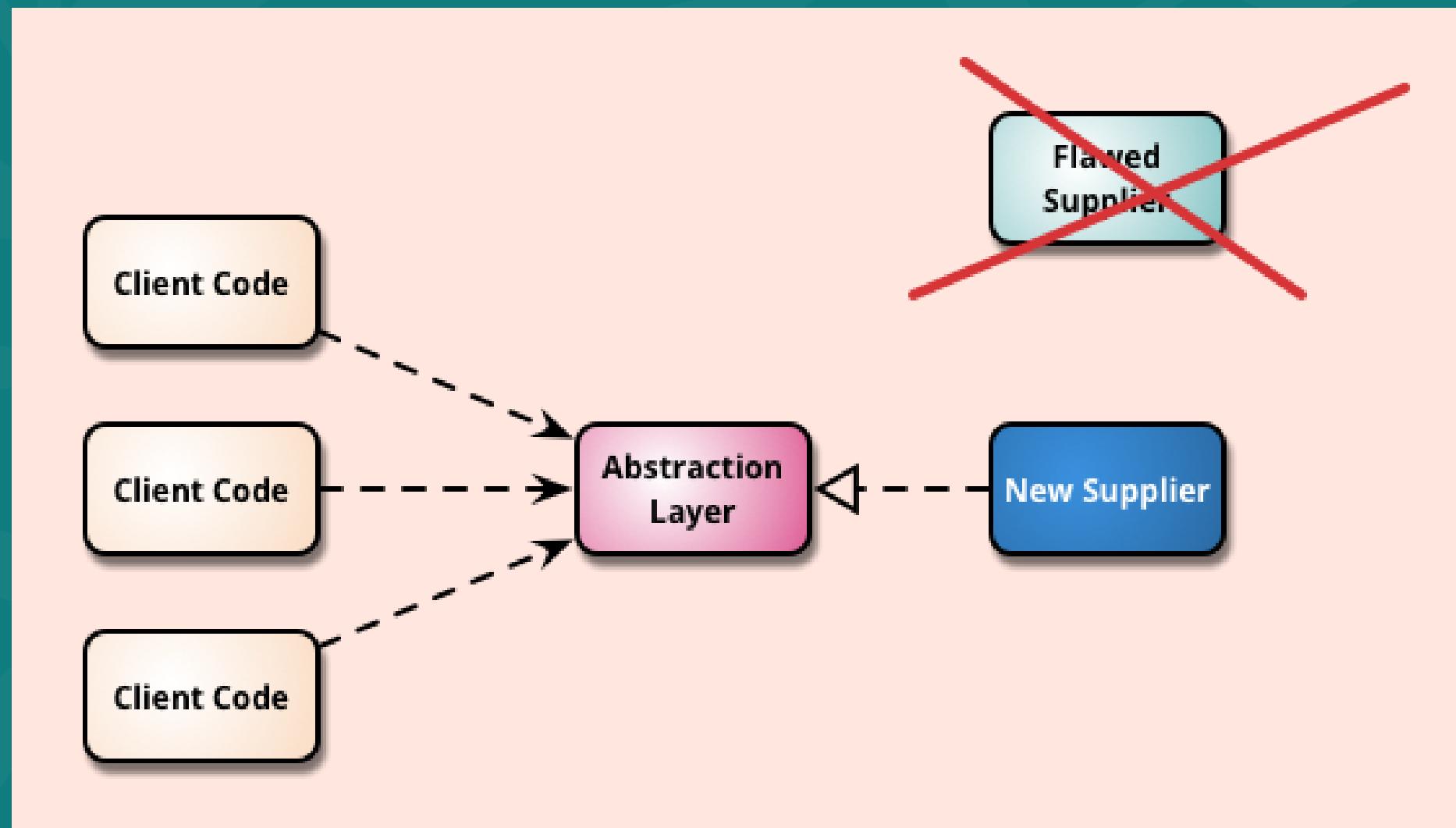


Branch by abstraction

What can we do when we have a monolith and
we need to create a microservice from an
existing feature?

Abstraction layer

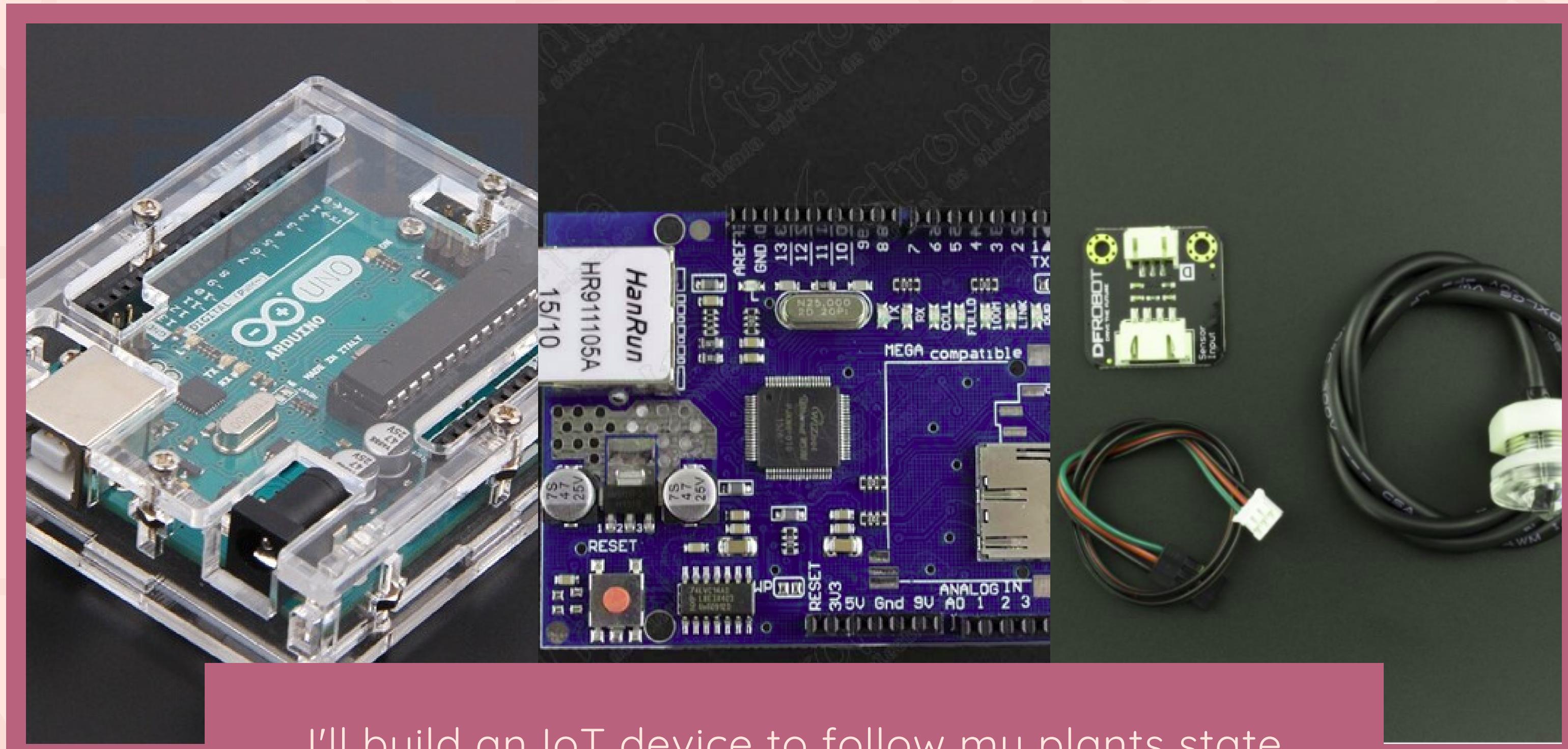




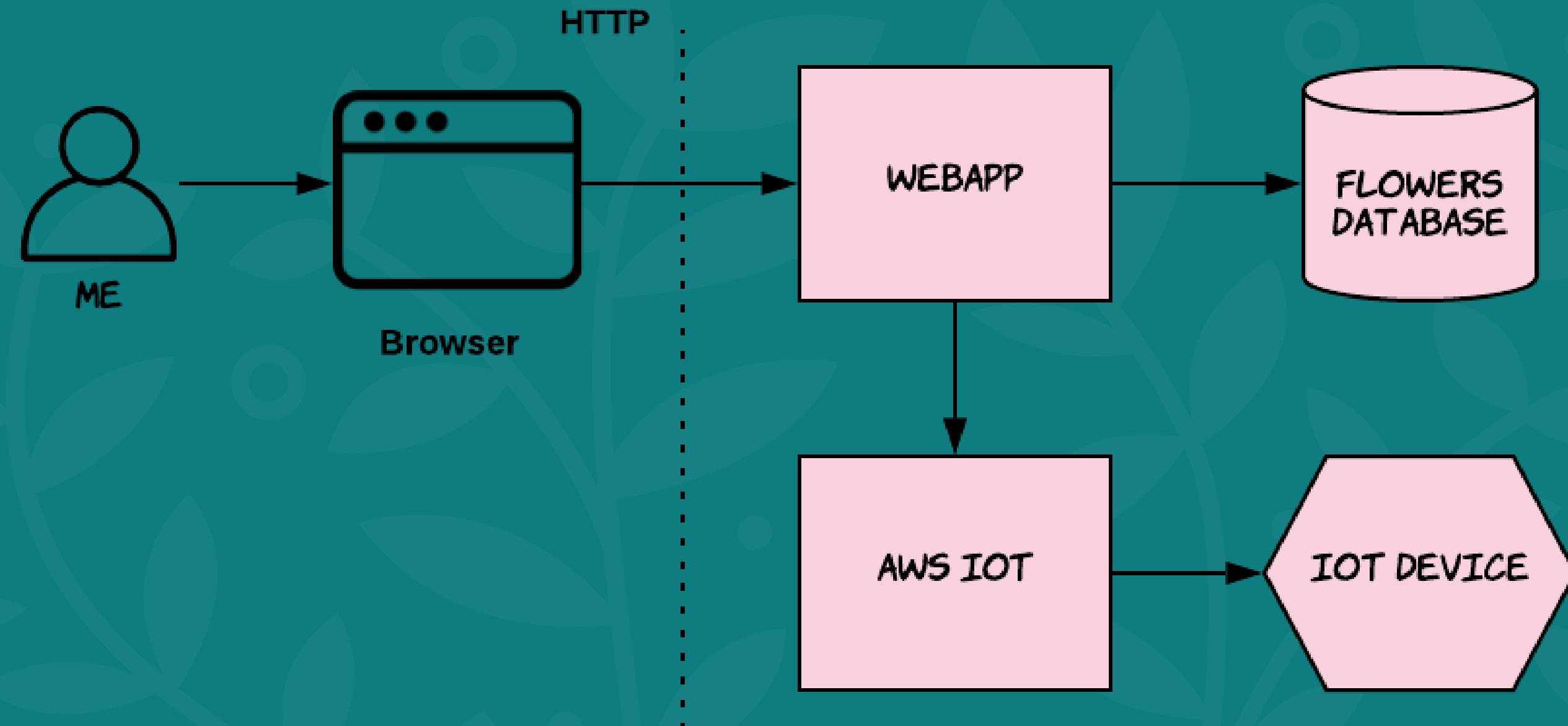
Now we can migrate the new supplier to a microservice.

...

Now, back to my garden..

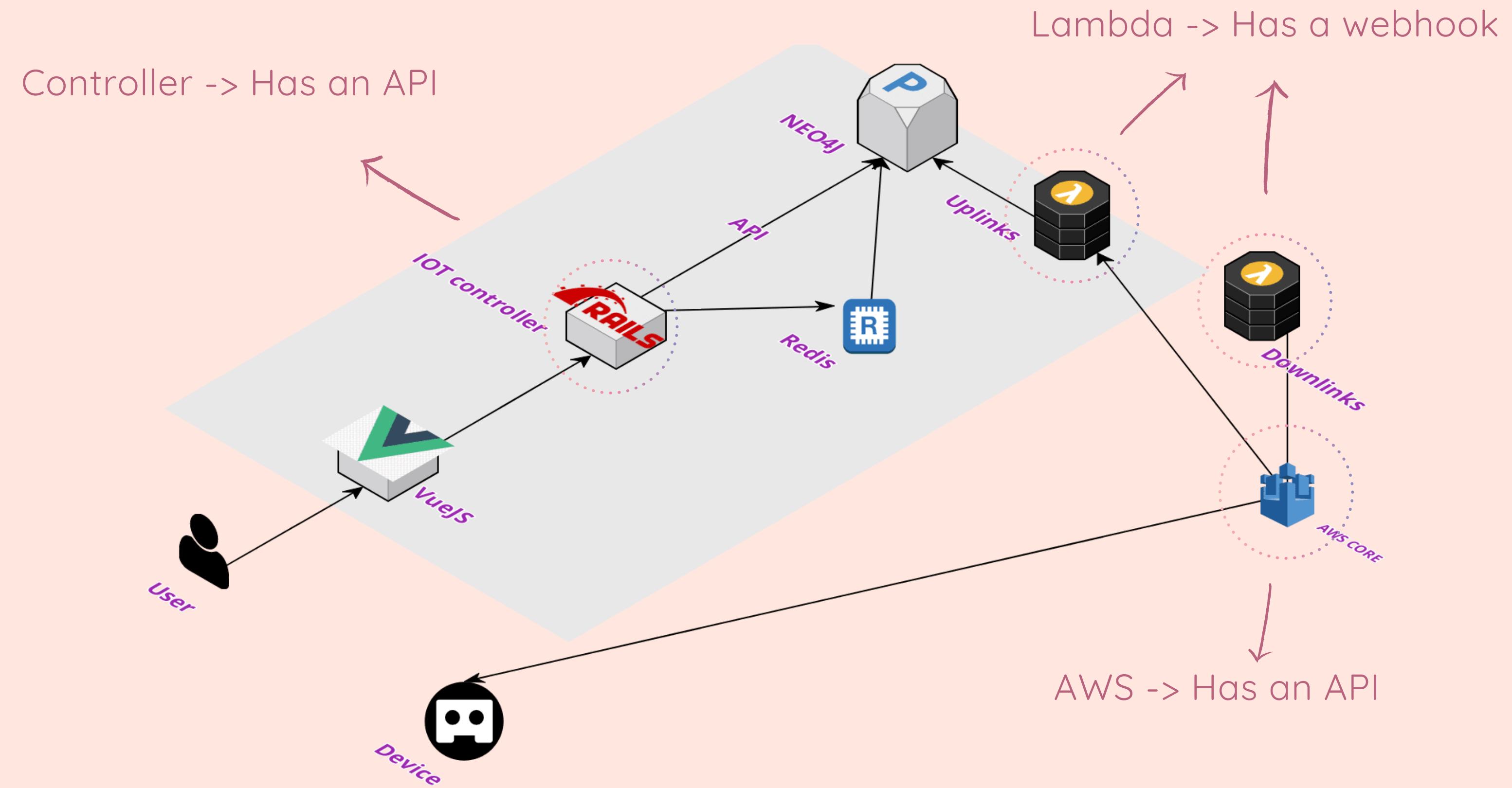


I'll build an IoT device to follow my plants state



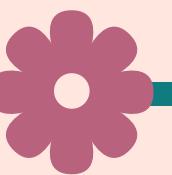
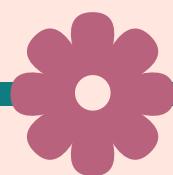
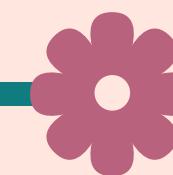
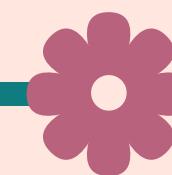
We'll have something like this

What about real life?



Conclusion

AT MY GARDEN...



1.

I built a MSA
because I had to, my
'business' needed it.

2.

I could develop my
own communication
device but it's not
worth it.

3.

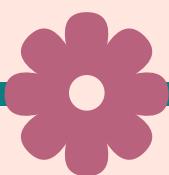
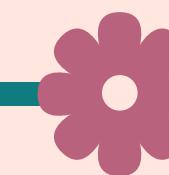
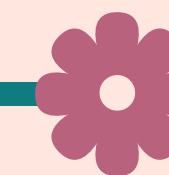
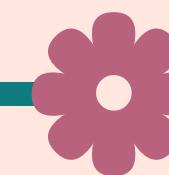
Now, I have to pay
more but I save time
for more features

4.

My microservice is
really useful and it
fits completely in my
architecture.

Conclusion

AT MY JOB...



1.

I built a MSA
because I thought i
needed it, and it is
too hard

2.

I could develop my
database
communication in the
monolith and
separate it later

3.

Now, I have to deal
with my technical
depth and maintain
4 microservices.

4.

Microservices are
great! but only if you
follow good
practices.



¿NIGHTMARE OR BLESSING?

Got any questions?

ADDITIONAL INFO

CONTRACT TESTING

<https://docs.pact.io/>

MICROSERVICES

<https://microservices.io>

FOR EVERYTHING

<https://martinfowler.com>

Follow Me!



@nathalyvillamor



@nathalyvillamor



@nathalyvillamor

nathalyvillamor6@gmail.com

rubybogotacomunity@gmail.com

nathaly@jsconf.co

Gracias!

