Give your ESP32s the gift of serviceability

microcontroller

Give your ESP32s the gift of serviceability

Toit

Kasper Lund, co-founder and CEO of Toit

Lots of fantastic experiences at YOW!

Presented at YOW! 2011

Toit

Presenting at YOW! 2021

Flutter

Presented at YOW! 2017

Our founding team comes from Google and Uber ...

Toit

Kasper Lund
Co-founder & CEO

Senior Staff Engineer, Google
Co-led development of Google's V8
Led development of Dart

Erik Corry
Co-founder

Senior Engineer, Google
Built the world's fastest regex engine

Anders Johnsen
Co-founder

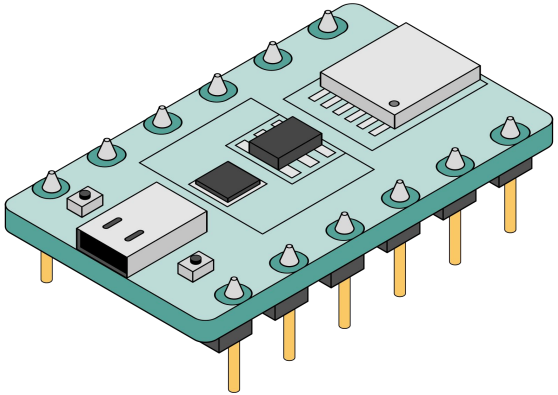
Software Engineer, Google
Senior Engineer II, Uber

Florian Loitsch
Co-founder

Senior Engineer, Google
PhD in Computer Science

Decades of experience implementing the world's most widely used software platforms

In 2018, we learned of the ESP32 ...



Powerful

Dual-core 240 MHz RISC CPU
520 KB RAM, 4MB+ Flash
Built in WiFi / Bluetooth

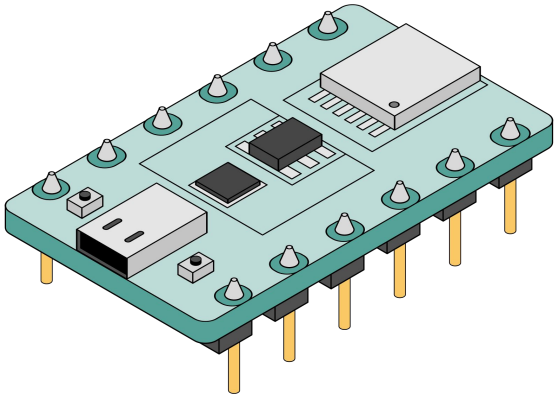
Runs on batteries

Practical drain in sleep mode is ~10 uA
Runs for years on AA batteries

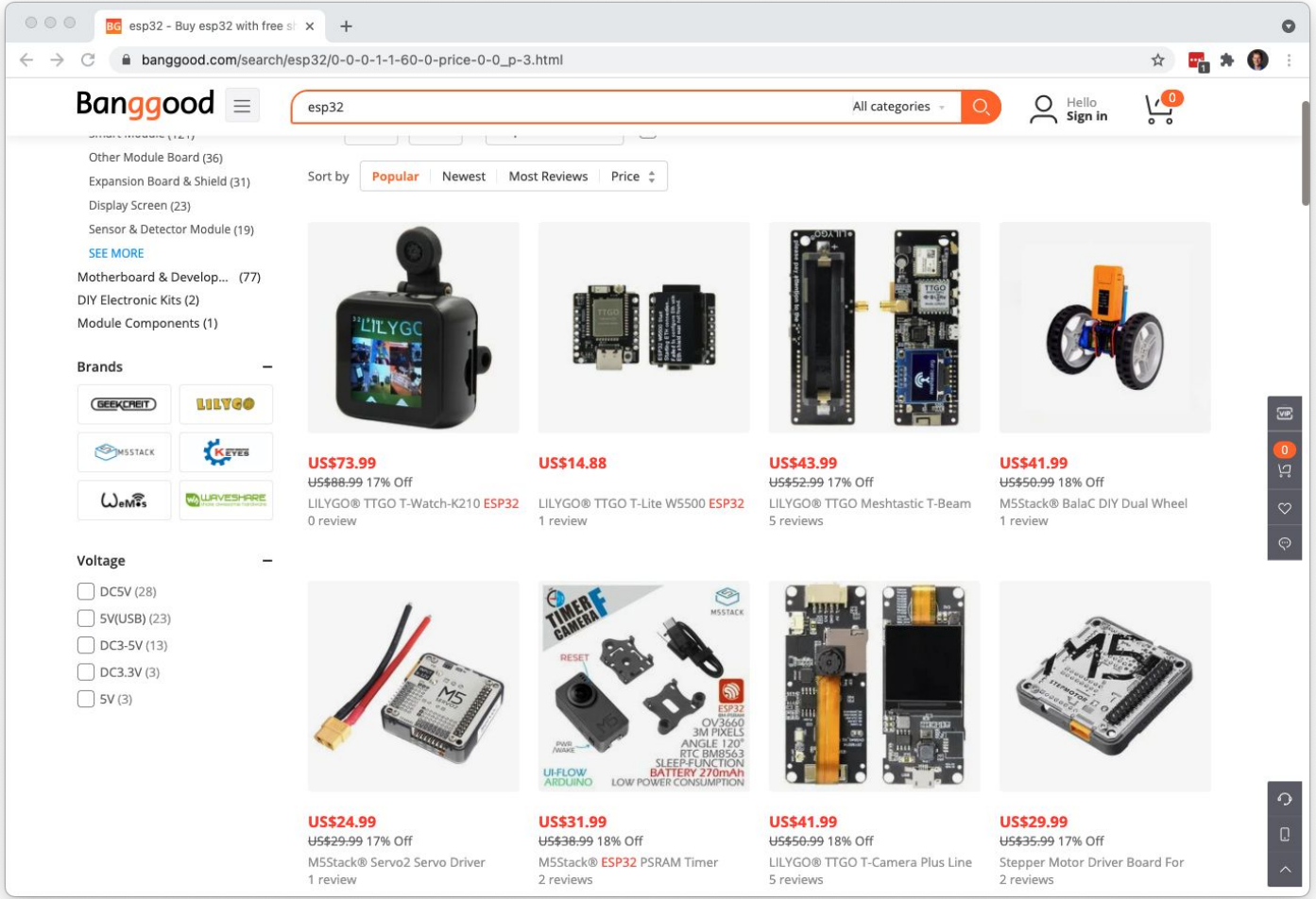
Inexpensive

Development kit costs $10
Standalone chip costs less than $2

For a lot of interesting use cases, this is a ...

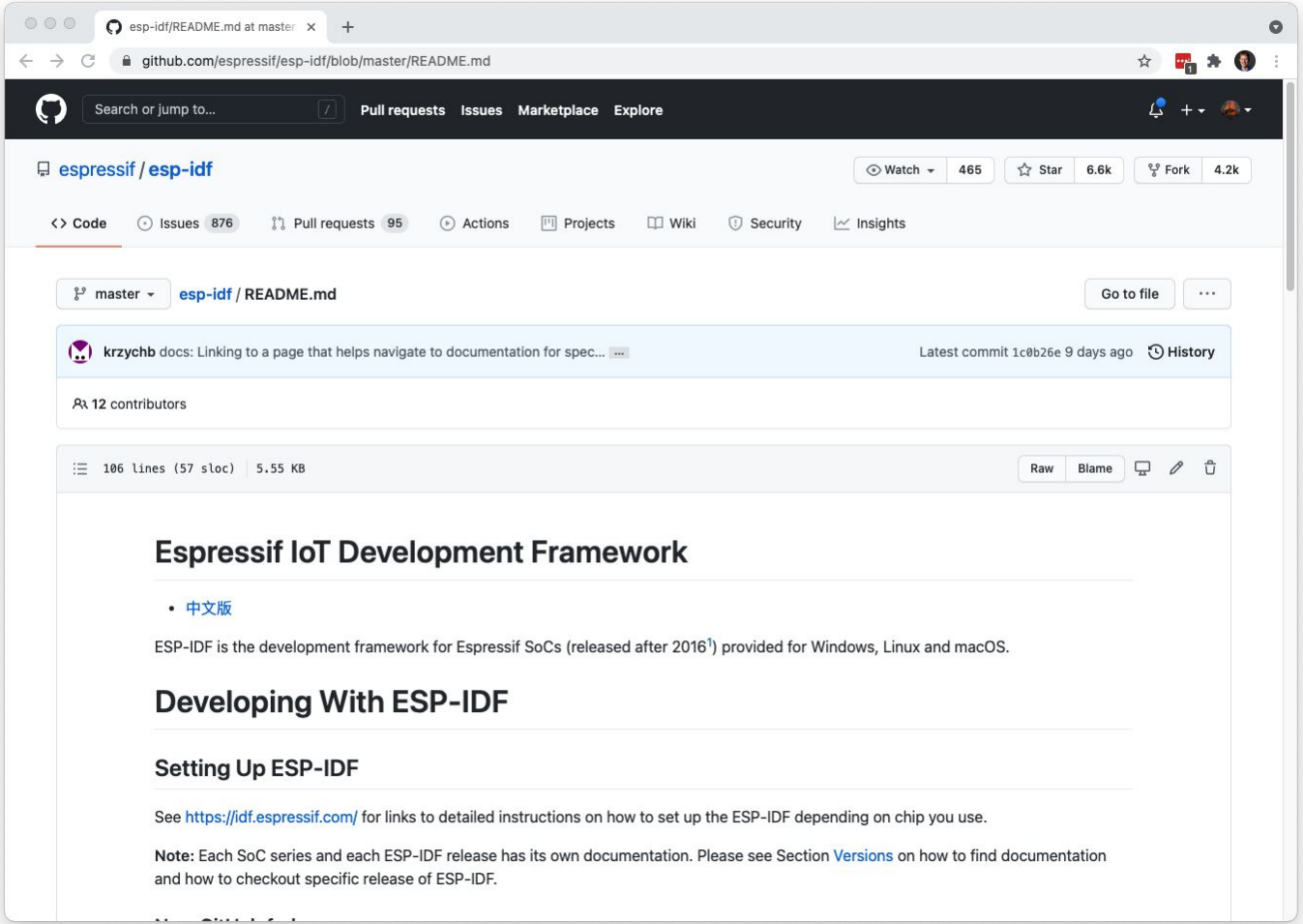


... compelling alternative to the Raspberry Pi


esp32 - Buy esp32 with free s
banggood.com/search/esp32/0-0-0-1-1-60-0-price-0-0_p-3.html
Banggood
esp32
All categories
Hello
Sign in
Other Module Board (36)
Expansion Board & Shield (31)
Display Screen (23)
Sensor & Detector Module (19)
SEE MORE
Motherboard & Develop... (77)
DIY Electronic Kits (2)
Module Components (1)
Brands
Voltage
DC5V (28)
5V(USB) (23)
DC3-5V (13)
DC3.3V (3)
5V (3)
Sort by
Popular
Newest
Most Reviews
Price
US$73.99
US$88.99 17% Off
LILYGO® TTGO T-Watch-K210 ESP32
0 review
US$14.88
LILYGO® TTGO T-Lite W5500 ESP32
1 review
US$43.99
US$52.99 17% Off
LILYGO® TTGO Meshtastic T-Beam
5 reviews
US$41.99
US$50.99 18% Off
M5Stack® BalaC DIY Dual Wheel
1 review
US$24.99
US$29.99 17% Off
M5Stack® Servo2 Servo Driver
1 review
US$31.99
US$38.99 18% Off
M5Stack® ESP32 PSRAM Timer
2 reviews
US$41.99
US$50.99 18% Off
LILYGO® TTGO T-Camera Plus Line
5 reviews
US$29.99
US$35.99 17% Off
Stepper Motor Driver Board For
2 reviews

I am a software engineer

Do I have what it takes to
write code for an ESP32?


esp-idf/README.md at master
github.com/espressif/esp-idf/blob/master/README.md
Search or jump to...
Pull requests
Issues
Marketplace
Explore
espressif / esp-idf
Watch 465
Star 6.6k
Fork 4.2k
Code
Issues 876
Pull requests 95
Actions
Projects
Wiki
Security
Insights
master
esp-idf / README.md
Go to file
krzychb docs: Linking to a page that helps navigate to documentation for spec...
Latest commit 1c0b26e 9 days ago
History
12 contributors
106 lines (57 sloc) 5.55 KB
Raw
Blame
Espressif IoT Development Framework
中文版
ESP-IDF is the development framework for Espressif SoCs (released after 2016[1]) provided for Windows, Linux and macOS.
Developing With ESP-IDF
Setting Up ESP-IDF
See https://idf.espressif.com/ for links to detailed instructions on how to set up the ESP-IDF depending on chip you use.
Note: Each SoC series and each ESP-IDF release has its own documentation. Please see Section Versions on how to find documentation and how to checkout specific release of ESP-IDF.

Developing for microcontrollers kinda sucks ...

Toit

- Thin, if any, separation between application, drivers, and OS
- Monolithic, close-knit system software tied to specific hardware
- C and assembly are the common source languages
- Application errors often result in crashing the entire device
- Development cycles are looooong

In spite of the hardware advances in microcontrollers, the development experience just doesn't compare favorably to server, desktop, or mobile development.

… because it is all about firmware

Toit

Fantastically firm

Get it right the first time!

Sufficiently soft

Learn fast and feel free to change your mind

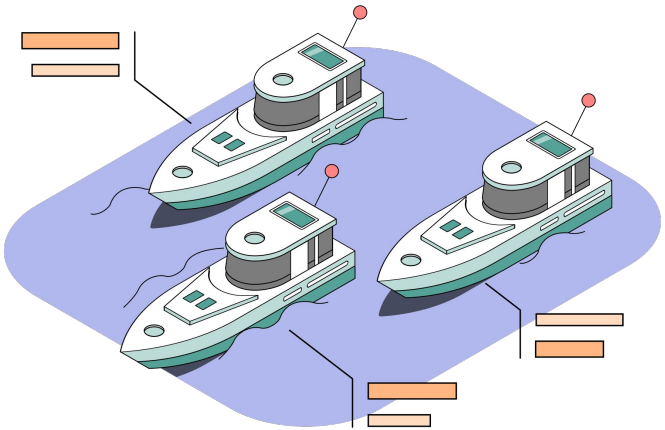
Scared of bricks?

Definition

Brick

An electronic device that, due to **corrupted firmware** or similar, **cannot be serviced** or even function, and thus is "bricked". The device becomes as technologically **useful as a brick**, hence the name.

Serviceability is more than observability



sɜːvɪsəˈbɪlɪtɪ

The ability to install, configure, and monitor computer products, identify exceptions or faults, debug or isolate faults to root cause analysis, and provide hardware or software maintenance in pursuit of solving a problem and restoring the product into service.


It's a UNIX system.
I know this.

How do you get serviceability for an ESP32?

Keep on truckin'

The software must be robust and **resilient** in the presence of bugs and faults. There is no way to service a bricked device.

Tell what's going on

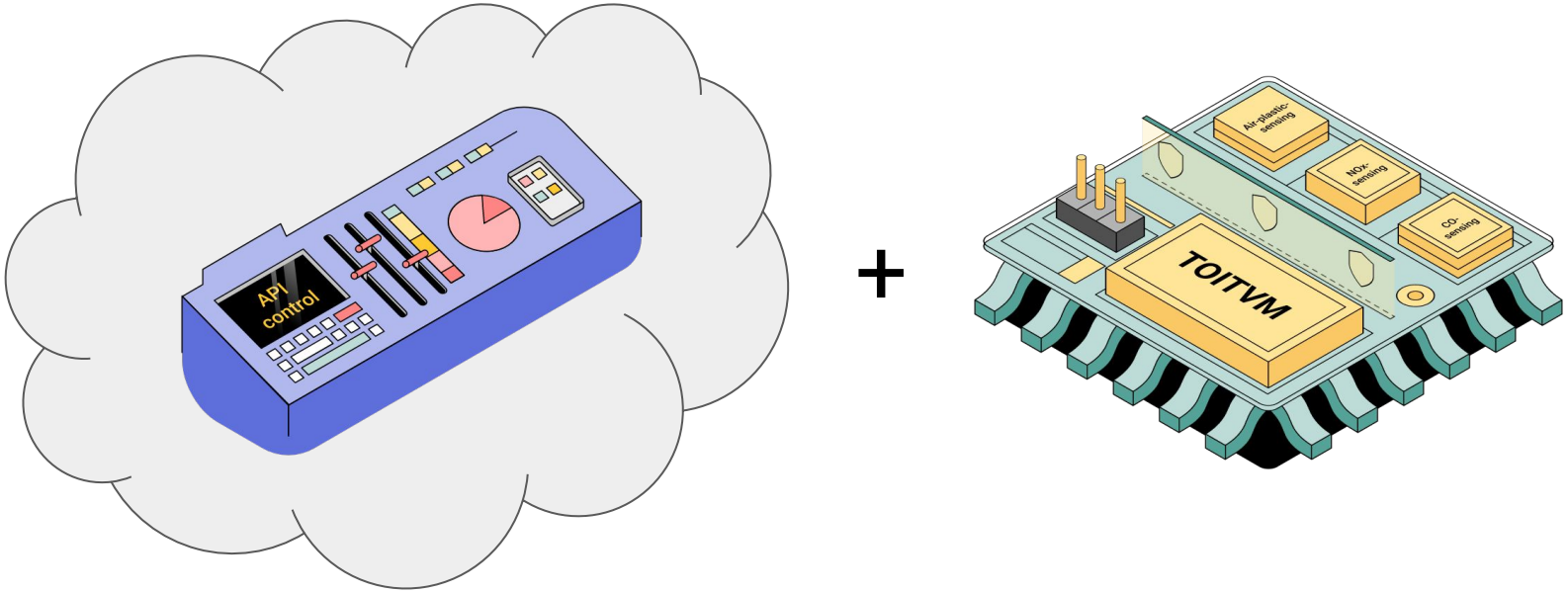
Event logging and telemetry metrics are critical tools to **understand** the behavior of the code running on the device.

Ask for direction

The system needs to prioritize taking direction and requests from an external orchestrator. This way you can **upgrade** and configure even in production.

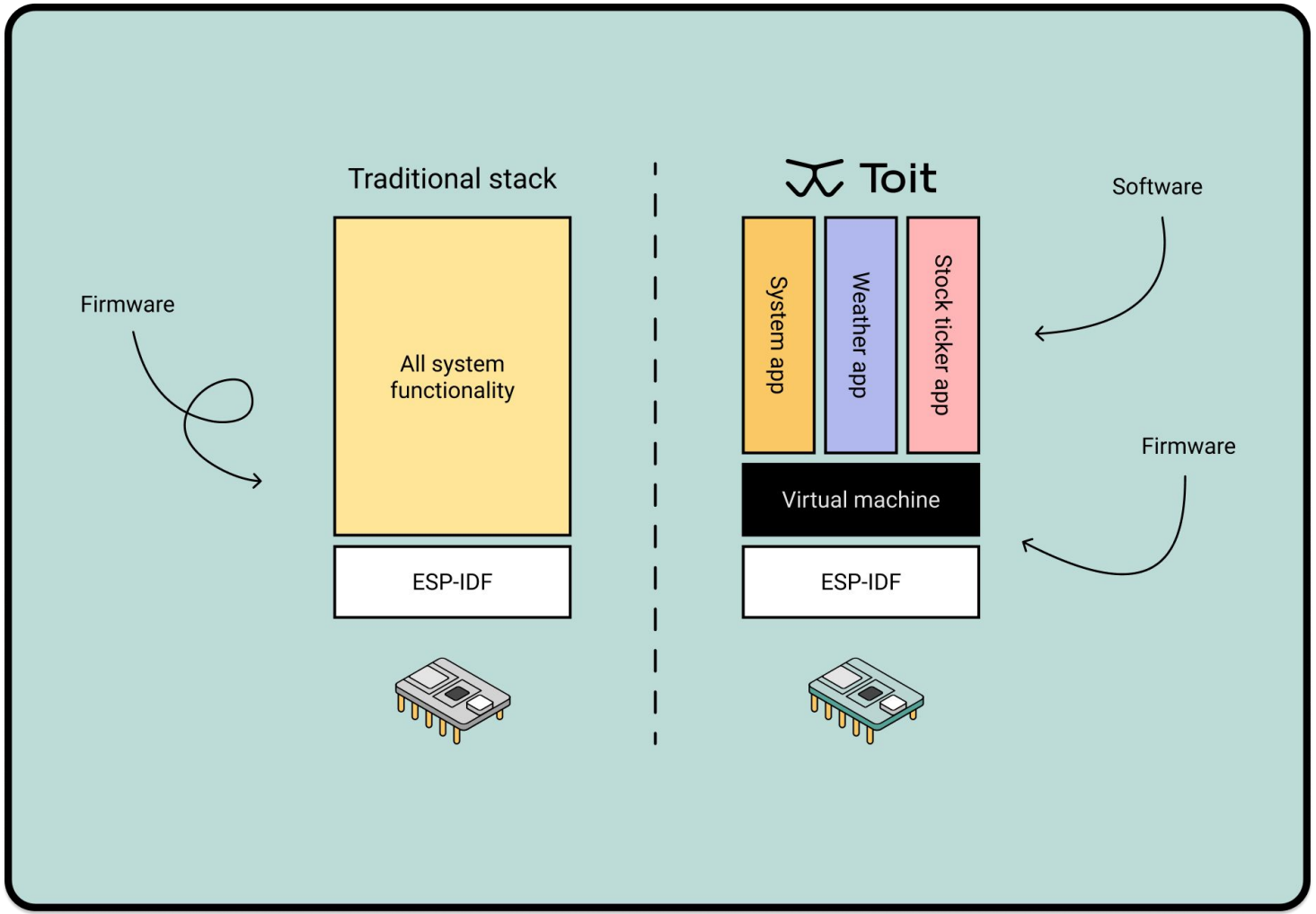
Luckily we've got a hammer ...

… in the form of a virtual machine



Cloud-managed containers on microcontrollers

Sandboxed environment for your ESP32 code,
fully controlled through a rich cloud API.

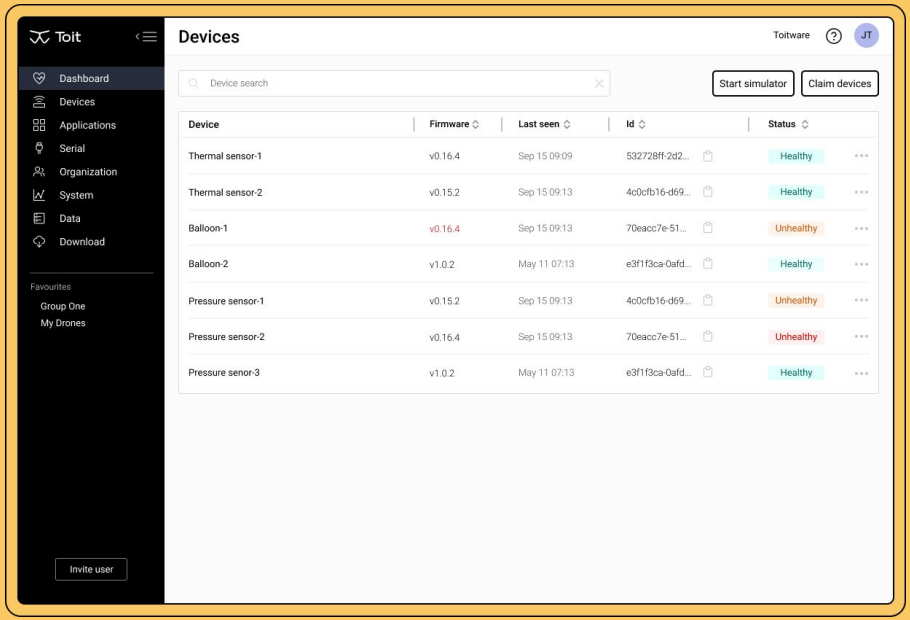

Traditional stack
Firmware
All system functionality
ESP-IDF
Toit
System app
Weather app
Stock ticker app
Virtual machine
ESP-IDF
Software
Firmware

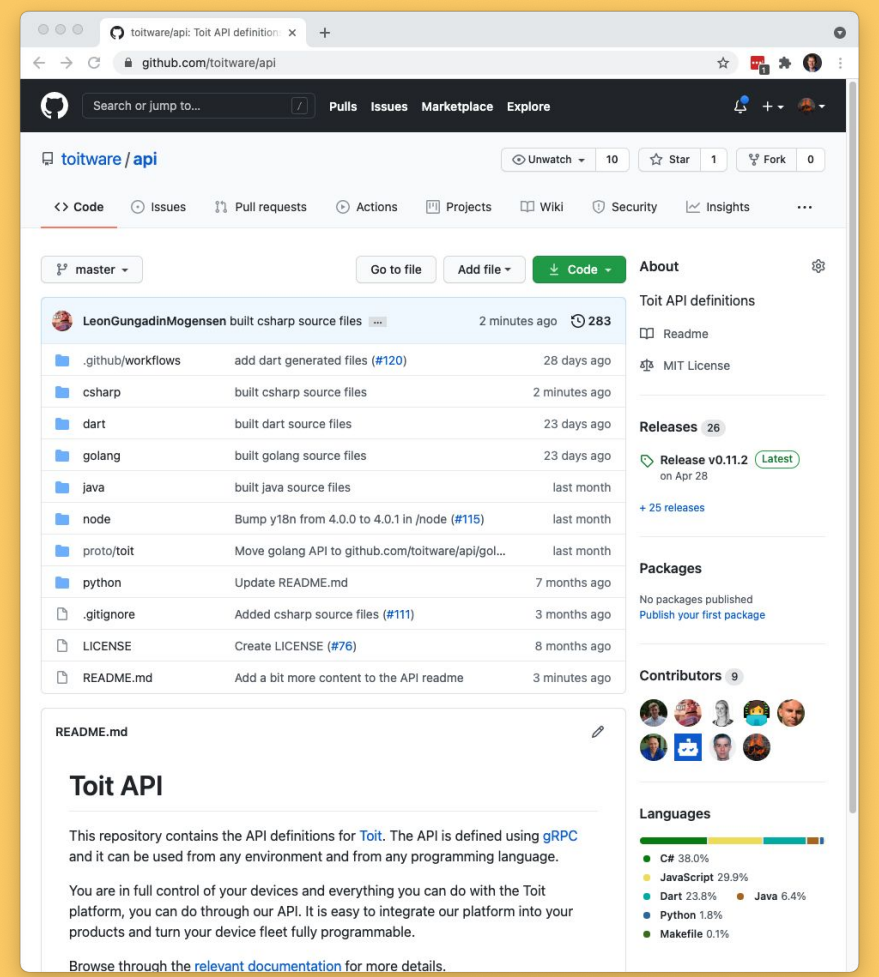
Get an overview through the **console**

Dashboard for your device fleet

Monitor and gain insights

Change and experiment





Rich **API** for servicing your devices

Update configurations

Install, update, and remove applications

Publish or subscribe to data

Built using gRPC

gRPC

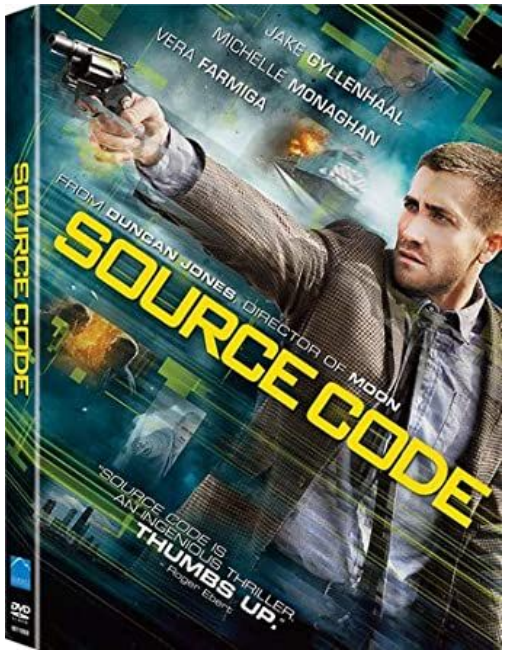
Google's high performance, open source universal RPC framework.

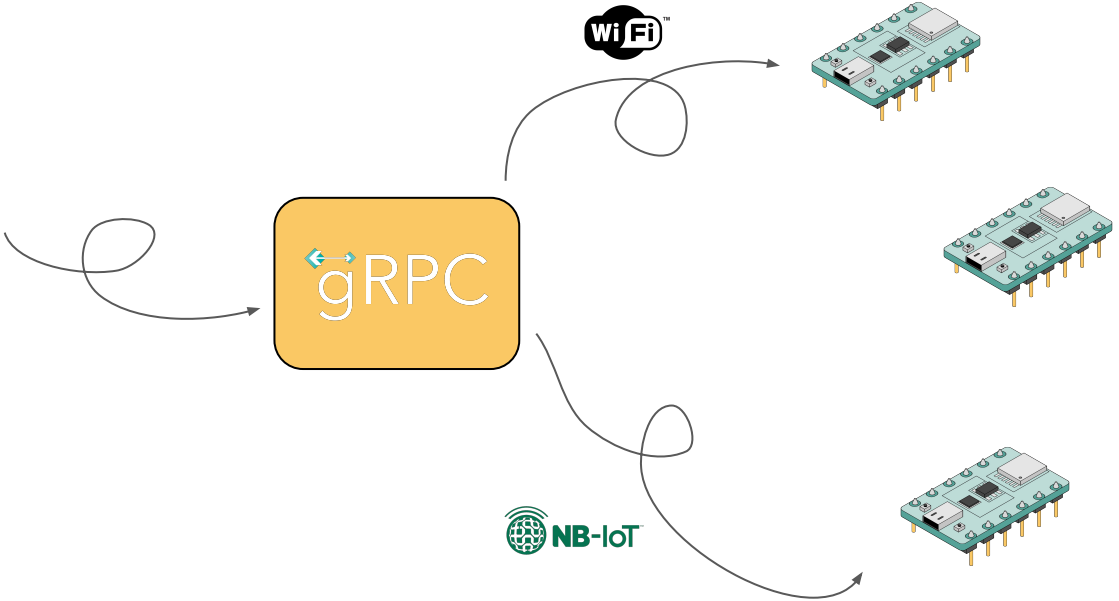
Reasonably idiomatic client libraries in 10 languages

Highly efficient on wire and with a simple service definition framework

Bi-directional streaming with HTTP/2 based transport

Pluggable auth, tracing, load balancing and health checking


JAKE GYLLENHAAL
MICHELLE MONAGHAN
VERA FARMIGA
FROM DUNCAN JONES, DIRECTOR OF MOON
SOURCE CODE
"SOURCE CODE IS AN INGENIOUS THRILLER. THUMBS UP."
SOURCE CODE

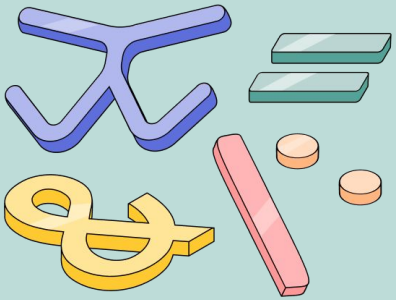

Wi Fi
gRPC
NB-IoT

Custom programming language

We built the **Toit language** to enable high-level programming for microcontrollers.

Hello

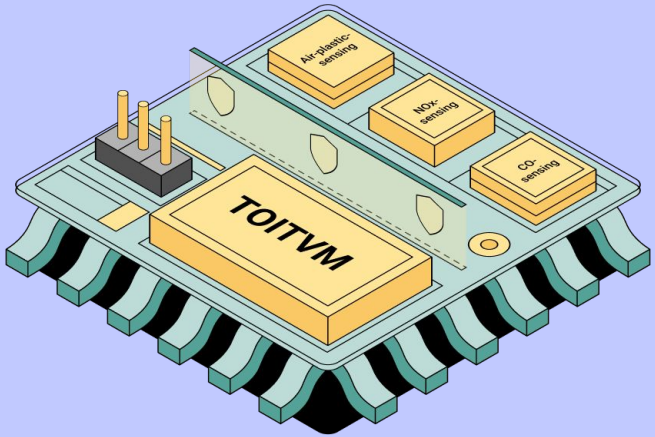
```
main:
  message := "Hello World"
  print message
```

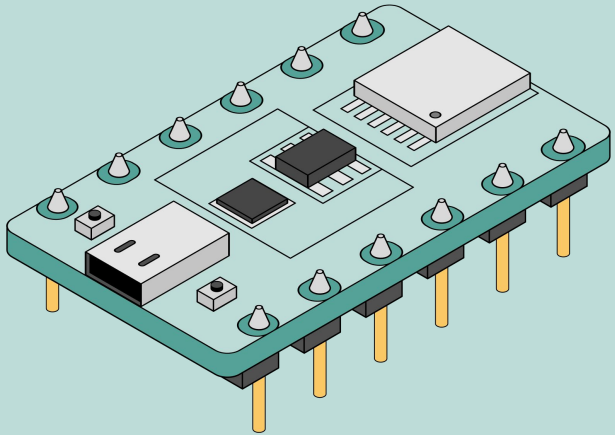


Functions

```
/// Returns the square of the given $x.
square x:
  return x * x

/// Returns the double of the given $x.
twice x/int -> int:
  return x + x
```





Classes

```
interface Vehicle:
  drive speed/int -> none

class Car implements Vehicle:
  drive speed:
    print "Driving $speed km/h"

main:
  car := Car
  car.drive 70
```

Blocks

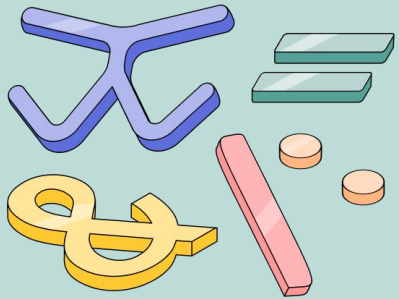
```
when condition [body]:
  if condition:
    body.call

main:
  when true:
    print "All is well!"
  when false:
    print "Oh, noes."
```

Cooperative tasks

```
main:
  unsorted := List 10: random 1000
  print unsorted
  unsorted.do: |value|
    task::
      sleep --ms=value
      print value
```

```
$ toit execute sleep-sorting.toit
[224, 812, 107, 690, 895, 71, 780, 630, 460, 624]
71
107
224
460
624
630
690
780
812
895
```



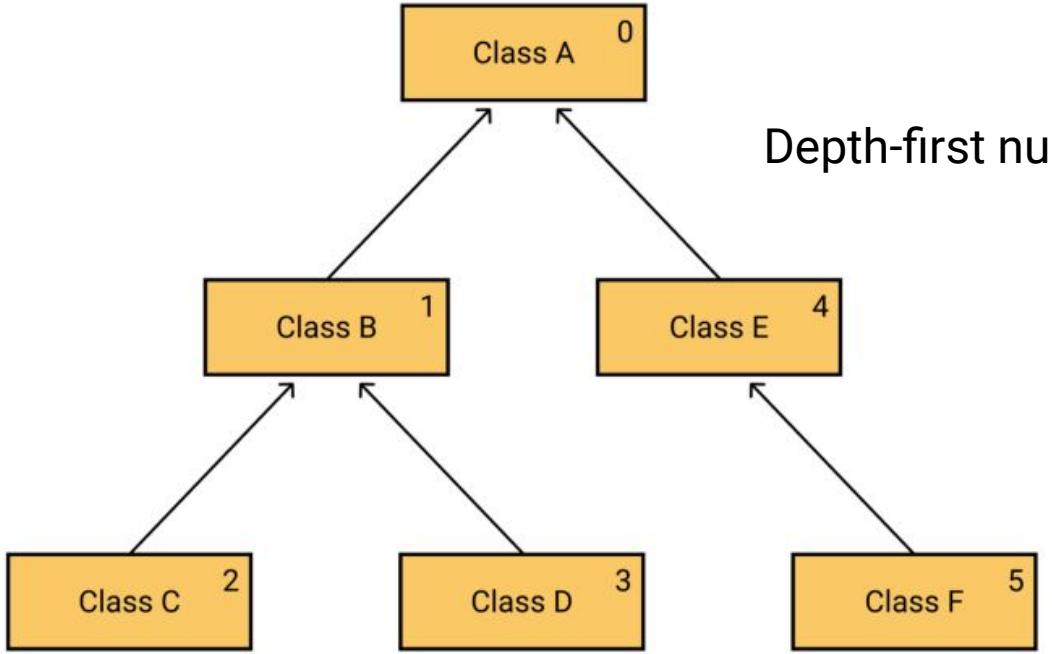
Inside the virtual machine

Sneak peek into the **engine** room.

```
months.append “April”
```

Optimized virtual dispatching without RAM-based caching

Compressed using selector-based row displacement

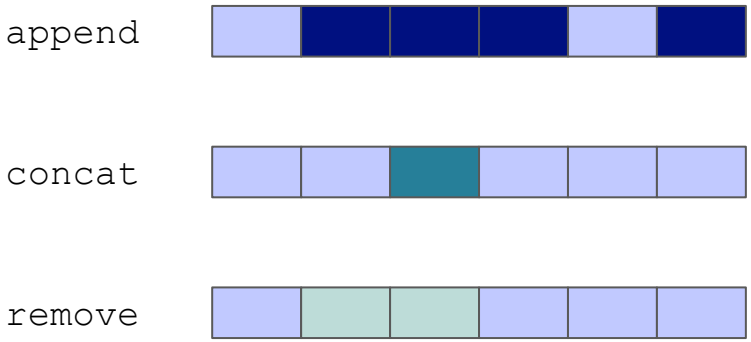


Depth-first numbered classes

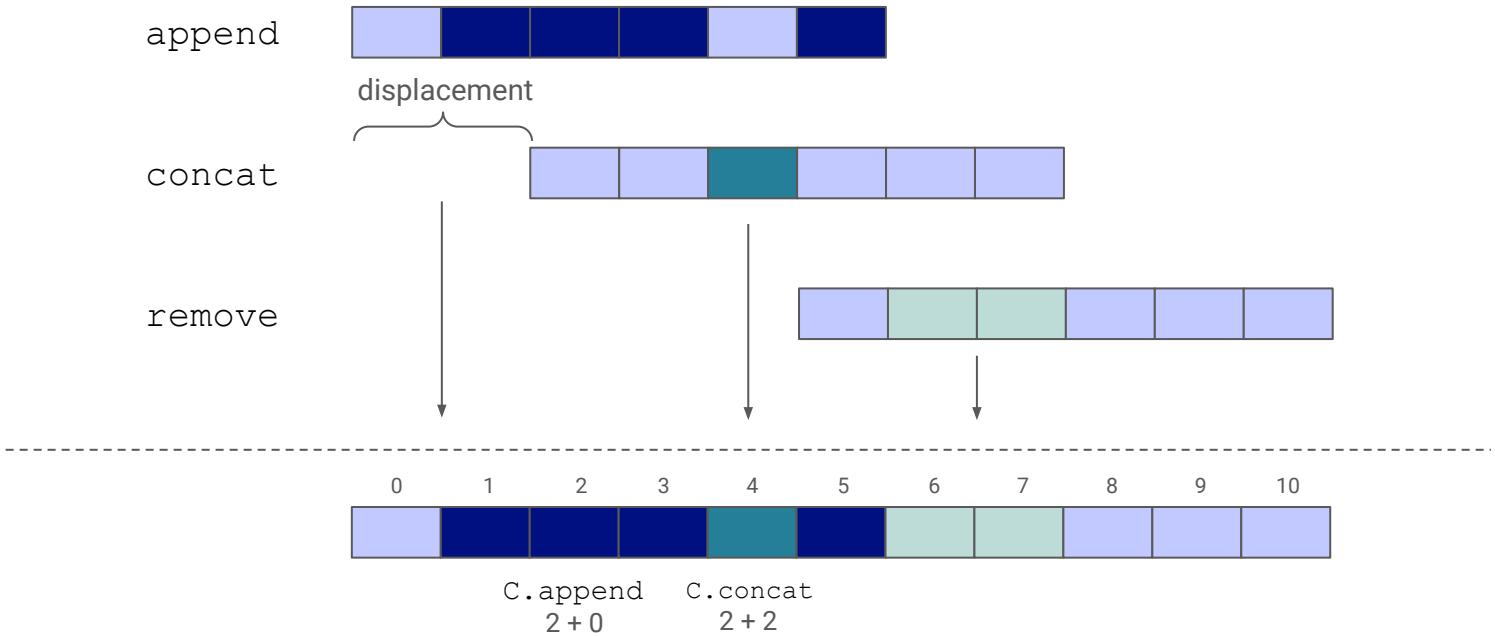
- 0	B.append 1	B.append 2	B.append 3	- 4	F.append 5

Dispatch table row for `append`

Compressing the dispatch table rows



Compressing the dispatch table rows

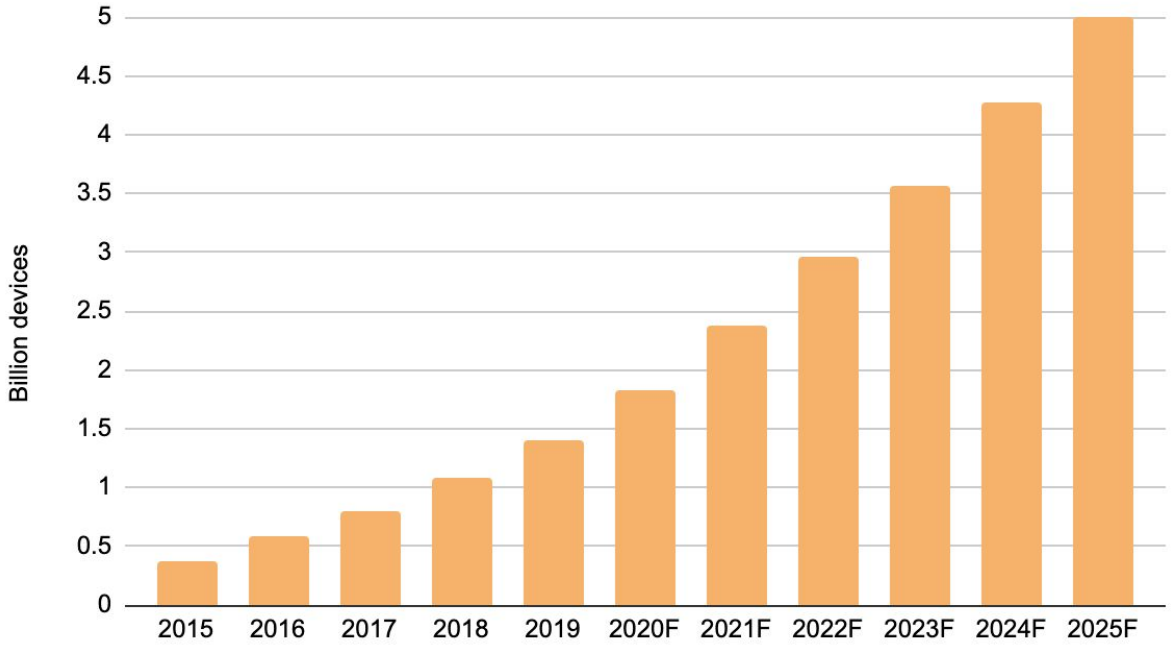


Demonstration

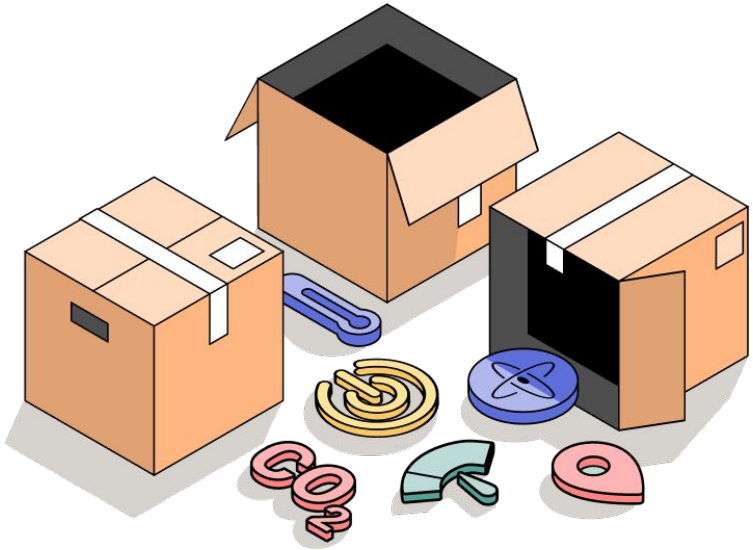
Catch a quick glimpse of the **Toit** experience.

```
 0:  load local 2
 1:  load smi 2
 3:  invoke <
 4:  branch to 11 if false
 7:  load local 2
 8:  return
11:  load local 2
12:  load smi 1
13:  invoke -
14:  invoke static fib test.toit:1:1
17:  load local 3
18:  load smi 2
20:  invoke -
21:  invoke static fib test.toit:1:1
24:  invoke +
25:  return
```

5 billion cellular connected devices need our help!



Source: Ericsson's "Cellular networks for Massive IoT" whitepaper



To get started, we have packaged up an end-to-end platform for your ESP32s.

You can deploy your solutions on microcontrollers and run for years on batteries without giving up on serviceability.

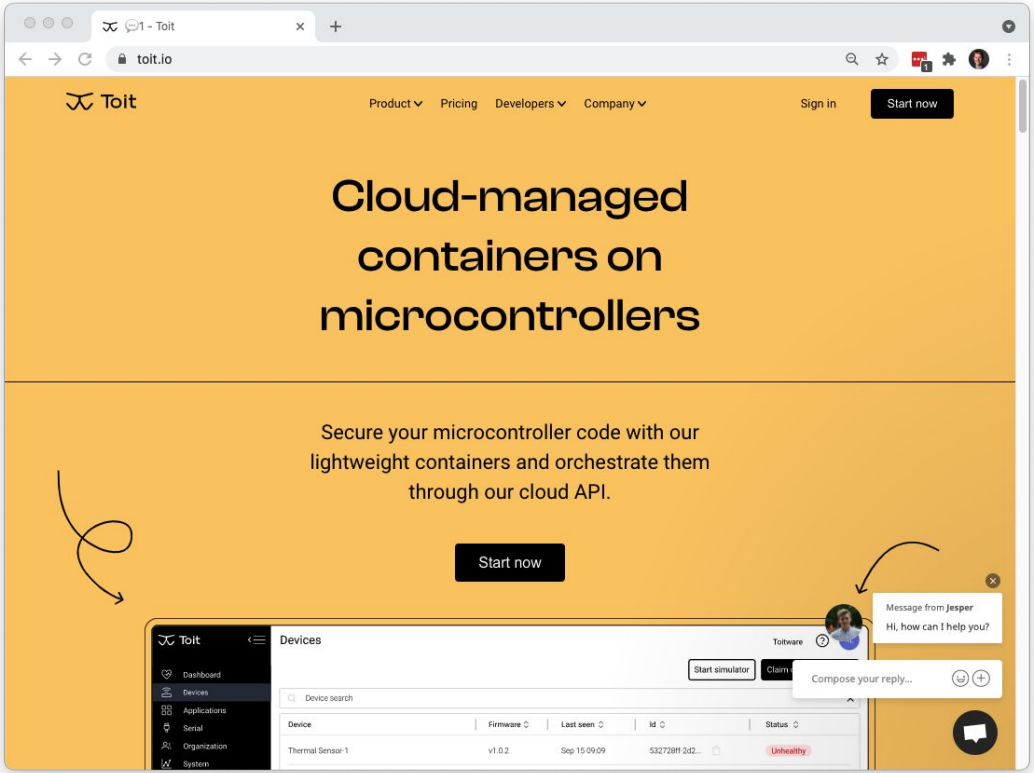
Toit 1.2 is here!

The full platform is open and easy to run on your own ESP32s.

You can sign up today for free via

https://toit.io/

and try a new development experience for microcontrollers.



Thank you!

Questions?