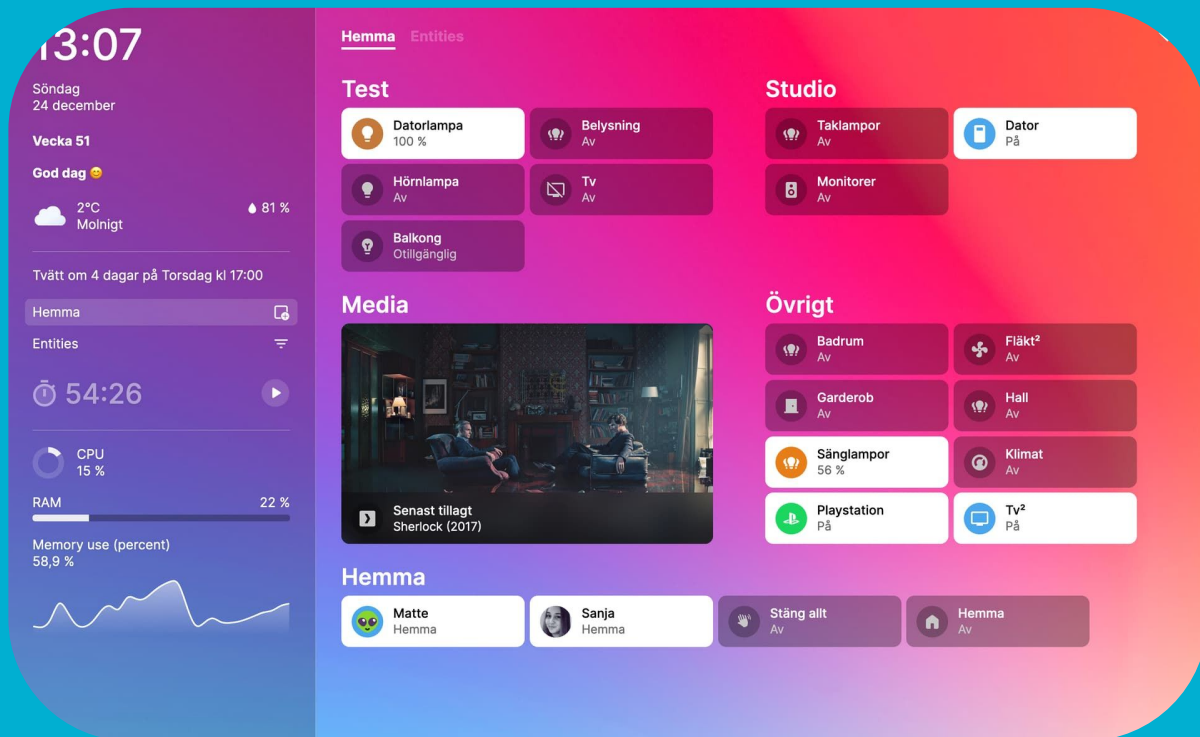


Home Assistant



What is the purpose of the project?

“It’s the brain of a smart home”



What is the purpose of the project?

For Paulus Schoutsen: “It’s the brain of a smart home”

- Local Control
- Privacy
- Interoperability



Integrations

The screenshot shows the Home Assistant integrations page. At the top, there's a navigation bar with links for 'Getting started', 'Documentation', 'Integrations', 'Blog', and 'Need help?'. Below this, there are filters for 'All', 'Featured', and 'Partners'. A table with columns 'Category', 'Version', 'IoT Class', and 'Quality Scale' is visible. Below the table is a search bar labeled 'Search integrations...'. The main content area displays a grid of integration cards, each with a logo and name. The cards are arranged in rows and columns, showing a variety of supported devices and services.

Category	Version	IoT Class	Quality Scale
All	All	All	All

Search integrations...

- 1-Wire
- 17TRACK
- 3 Day Blinds
- A. O. Smith
- Abode
- Acala
- AccuWeather
- Acer Projector
- Acomax
- Actiontec
- Adax
- AdGuard Home
- ADS
- Advantage Air
- AEMET OpenData
- AEP Ohio
- AEP Texas
- AfterShip
- Agent DVR
- AI Task
- Air quality
- air-Q
- AirGradient
- Airly
- AirNow
- Airthings
- Airthings BLE
- AirTouch 4
- AirTouch 5
- AirVisual Cloud
- AirVisual Pro
- Airzone
- Airzone Cloud
- Alarm control panel
- AlarmDecoder
- Alert
- Alexa Devices
- Alpha Vantage
- Altruist
- Amazon Alexa



ADD INTEGRATION TO MY

The Apple TV integration was introduced in Home Assistant 0.49, and it's used by **15.4%** of the active installations.

Its IoT class is **Local Push**.

[View source on GitHub](#)

[View known issues](#)

INTEGRATION OWNERS

We are incredibly grateful to the following contributors who currently maintain this integration:



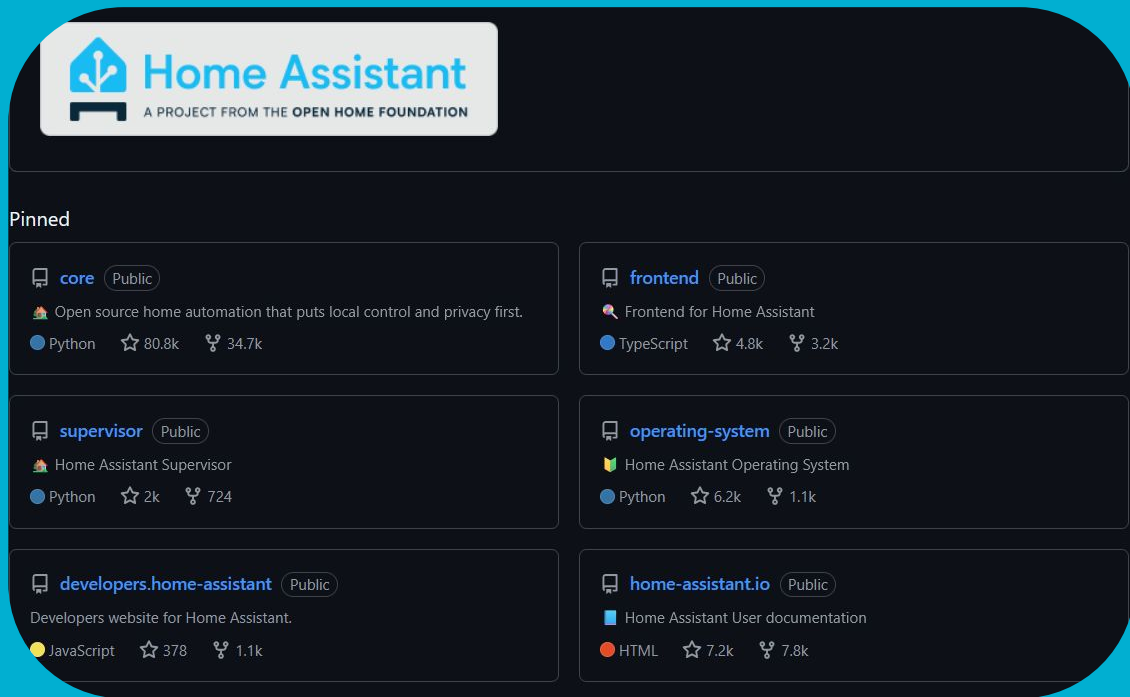
@postlund

What is the purpose of the project?

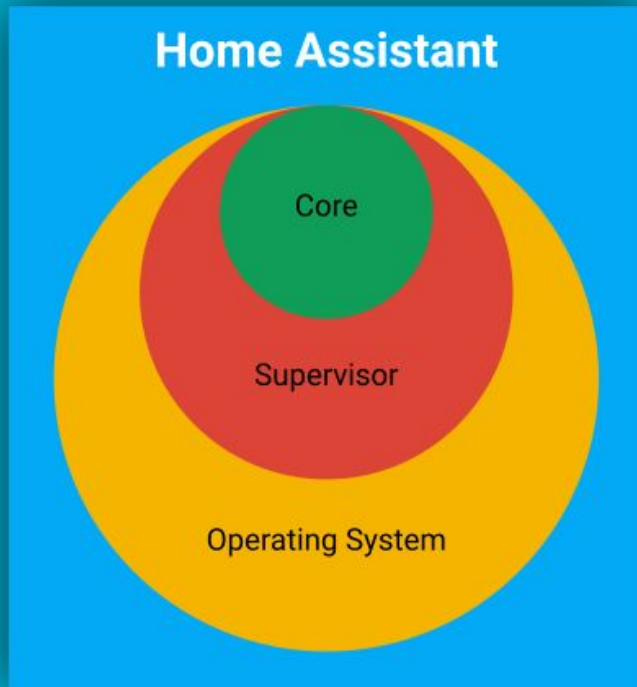
Top 10 public projects by contributors on GitHub

Project	Contributor count
home-assistant/core	>21K
microsoft/vscode	>20K
ProvableHQ/leo	>20K
firstcontributions/first-contributions	>13k
flutter/flutter	>10K
tensorflow/tensorflow	>10K

Where is the source code located?



Which technologies are used?



Which technologies are used?

CORE

Backend:

- Python
- SQLite
- aiohttp
- YAML

Frontend:

- HTML/CSS
- JS
- Lit/Web components
- YAML



















Which technologies are used?

Supervisor

Backend:

- Python
- Docker

Official add-ons

 Almond The open-source, privacy-preserving voice assistant	 CEC Scanner Scan for HDMI CEC devices	 Check Home Assistant configuration Check your Home Assistant configuration against other versions	 deCONZ Control a Zigbee network with ConBee or RaspBee by Dresden Elektronik
 Duck DNS Free Dynamic DNS (DynDNS or DDNS) service with Let's Encrypt support	 File editor Simple browser-based file editor for Home Assistant	 Hey Ada! Voice assistant powered by Home Assistant	 HomeMatic CCU HomeMatic central based on OCCU
 Let's Encrypt Manage certificate from Let's Encrypt	 MariaDB A SQL database server	 Mosquitto broker An Open Source MQTT broker	 NGINX Home Assistant SSL proxy An SSL/TLS proxy
 OpenZWave Control a ZWave network with Home Assistant	 RPC Shutdown Shutdown Windows machines remotely	 Samba share Expose Home Assistant folders with SMB/CIFS	 TellStick TellStick and TellStick Duo service
 VLC Turn your device into a Media Player with VLC	 Z-Wave JS Control a ZWave network with Home Assistant Z-Wave JS		

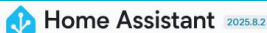
Which technologies are used?

OS

Backend:

- Linux
- Buildroot
- GRUB

License and contribution guidelines



Home » Developers »

The Apache 2.0 License

The Home Assistant source code is released under the following license.

Apache License

Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the license.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

Contributing

Hi there! We're thrilled that you'd like to contribute to this project. Your help is essential for keeping it great.

Contributions to this project are released to the public under the project's open source license ([Apache 2.0](#) for code, [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International](#) for everything else).

Please note that this project is released with a [Contributor Code of Conduct](#). By participating in this project you agree to abide by its terms.

Submitting a pull request

0. Fork and clone the repository

1. Create a new branch: `git checkout -b my-branch-name`
2. Make your change, add tests and make sure the tests still pass
3. Push to your fork and submit a pull request
4. Pat yourself on the back and wait for your pull request to be reviewed and merged.

Here are a few things you can do that will increase the likelihood of your pull request being accepted:

- Follow standards for style and code quality
- Write tests.
- Keep your change as focused as possible. If there are multiple changes you would like to make that are not dependent upon each other, consider submitting them as separate pull requests.
- Write a [good commit message](#).

Apache License 2.0

A permissive license whose main conditions require preservation of copyright and license notices. Contributors provide an express grant of patent rights. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions

- Commercial use
- Distribution
- Modification
- Patent use
- Private use

Conditions

- License and copyright notice
- State changes

Limitations

- Liability
- Trademark use
- Warranty

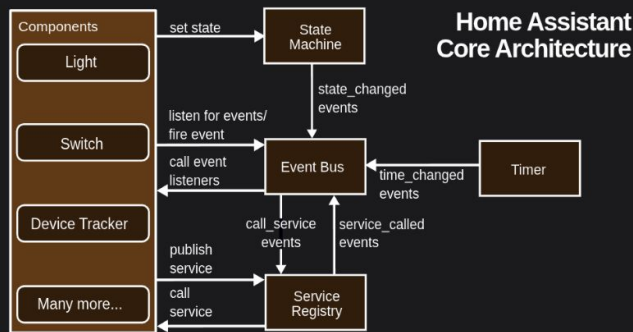
[View full Apache License 2.0 »](#)

Docs

Core architecture

The Home Assistant Core consists of four main parts. On top of this it includes many helper classes to deal with common scenarios, like providing an entity or dealing with locations.

- **Event Bus:** facilitates the firing and listening of events – the beating heart of Home Assistant.
- **State Machine:** keeps track of the states of things and fires a `state_changed` event when a state has been changed.
- **Service Registry:** listens on the event bus for `call_service` events and allows other code to register service actions.
- **Timer:** sends a `time_changed` event every 1 second on the event bus.



[Edit this page](#)

Last updated on Jul 16, 2024

Previous
« Introduction

Next
Integrations »

Governance



Governance



Paulus Schoutsen
President



Pascal Vizeli
Treasurer



J. Nick Koston
Member

Projects



Home Assistant

Home Assistant is an open-source, home automation platform that puts local control and privacy first. It is powered by a worldwide community of tinkerers and DIY enthusiasts, and exists to put users in control of their homes, their data, and their personal well-being.

[LEARN MORE](#)

Music Assistant

Music Assistant is a music library manager for your offline and online music sources, which can easily stream your favourite music to a wide range of supported players. It becomes incredibly powerful when combined with Home Assistant!

[LEARN MORE](#)

ESPHome

ESPHome is a system to control ESP8266, ESP32, and other microcontrollers to empower makers to repurpose or create custom smart home devices. It's based on simple, yet powerful low-code configuration files, and is compatible with Home Assistant and other home automation systems. It also works as a standalone system.

[LEARN MORE](#)

Minor projects

Wyoming



Peer-to-peer protocol for voice assistants.

Open standard

Voice

BTHome



Open standard for broadcasting sensor data and button presses over Bluetooth LE.

Open standard

Bluetooth

Improv Wi-Fi



Open standard for connecting devices to Wi-Fi using Bluetooth LE or Serial.

Open standard

Wi-Fi

Zigpy



Hardware Independent Zigbee integration that implements Zigbee as a Python library.

Driver

Zigbee

Python Matter Server



Matter Controller Server implementation over WebSockets.

Driver

Matter

Piper



Local neural text-to-speech system that sounds great and is optimized for the Raspberry Pi 4.

Library

Voice

ESP Web Tools



User friendly tools to manage ESP8266 and ESP32 devices in a web browser.

Library

ESP32

HACS



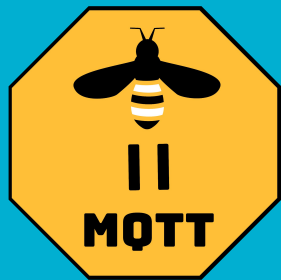
A custom integration designed to help the Home Assistant community discover and share projects.

Home Assistant Integration

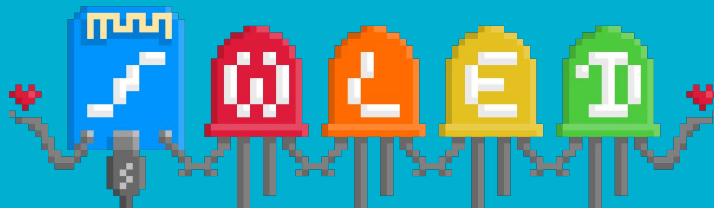
Visit us on GitHub



Colabs



Zigbee2Mqtt



WLED



Rhaspy

Who supports our work

The work of the Open Home Foundation is made possible through the contributions of these organizations:



NABU CASA

Commercial Partner



DuckDuckGo.

Donors



ESPRESSIF



Supporters

Who they support

AIOHTTP



Alpine Linux



Buildroot



NOW

NEXT

LATER



DEVICES

Bring context to devices

- "Device types"



AUTOMATION

Make automations easier to create

- Better triggers and conditions

Make default automations relevant and instant

- Device-based automation suggestions

Make automations more visual

- Better automation editor layout



DASHBOARD

THEME
Home-approved
dashboards

Make default dashboards more useful and relevant

- Default dashboard
- Area dashboards
- Function dashboards

Improve representation of devices

- Device dashboards and device pages

Improve privacy for user, guest, and public access

- ????

Make dashboards easier to share

- ????



VOICE & MUSIC

Make Assist more conversational

- Announcement
- Continued conversation

Make Music Assistant easier to set up

- Music Assistant onboarding



FRONTEND

Make features easier to discover and navigate to

- Sidebar revamp
- Settings navigation revamp

Design system overhaul