


Jami and how it empowers users

LibrePlanet 2021

Amin Bandali



What is Jami?



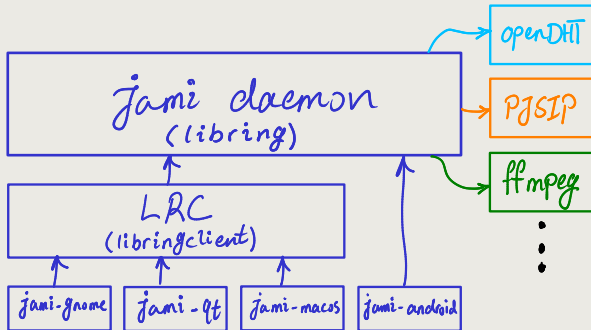
Jami is free software for universal communication
that respects the freedom and privacy of its users



Jami is developed by **Savoir-faire Linux**® based in Montréal, Canada. Savoir-faire Linux is a team of free software consultants providing training, consulting, development, and support services for free software technologies.

- one-on-one conversations
- file sharing
- audio/video calls & conferences
- screen sharing in video calls & conferences
- recording & sending audio/video messages
- SIP phone software functionality
- cross-platform communication framework

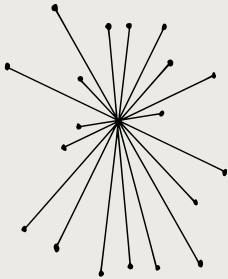
Architecture overview



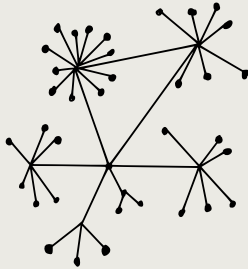
Architecture & components of Jami

Jami is

an **end-to-end encrypted** secure
and **distributed** voice, video,
and chat **communication platform**
that requires **no central server**
and leaves the power of **privacy**
and **freedom** in the hands of *users*



Centralized



Decentralized

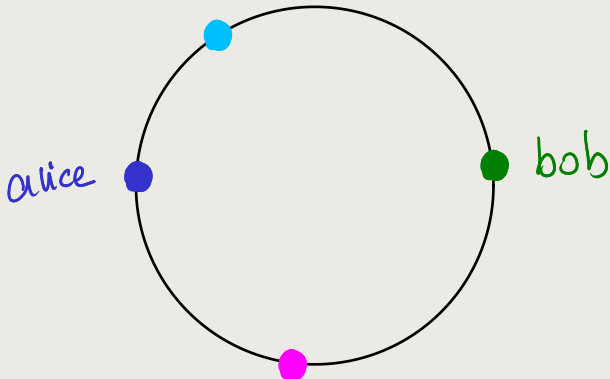


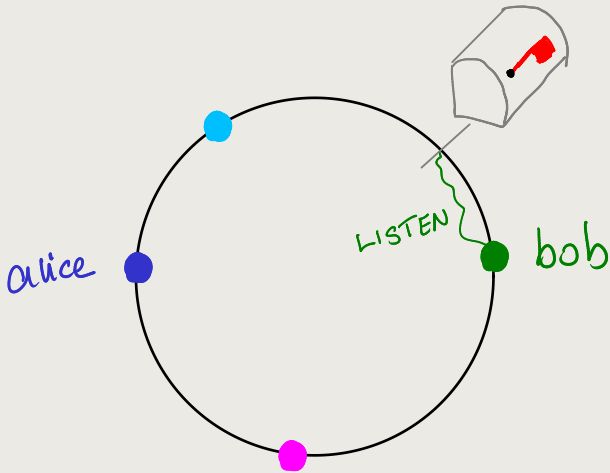
Distributed

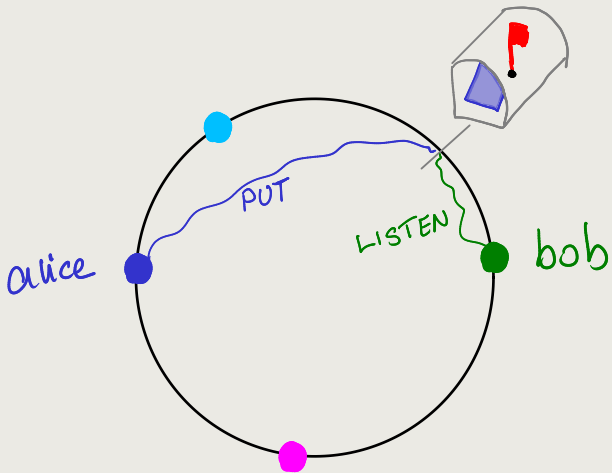
OpenDHT



- lightweight and scalable, designed for large networks and small devices
- high resilience to network disruption
- IPv4 and IPv6 support
- clean and powerful C++14 *map* API
- C, Rust, and Python 3 bindings
- public key crypto layer providing optional data signing and encryption (using GnuTLS)
- REST API with optional HTTP client+server and push notification support









TLS
ICE



SIP
TLS
ICE



SRTP

SIP

TLS

ICE



Development news

- many connectivity improvements
- automatic video bitrate adjustment
- detached host rendezvous points
- notification improvements
- bidi text in the chat view
- video renderer improvements
- conference call moderation
- improved error handling
- object copy reductions
- plugin API overhaul
- ... and more!

Rendezvous points

Meeting meeting9876 Talking 17:26

Find or start a conversation... +

- Meeting meeting9876 Talking
- Guillaume skp276 Outgoing 09:07
- Emeline emeline@revelia Outgoing 2021-03-19
- Cake cake@revelia Find 2021-03-19
- David David@revelia Find 2021-03-19
- Myrtille myrtille@revelia Outgoing 2021-03-19
- Harry harry@revelia Outgoing 2021-03-19
- Louane louane@revelia Outgoing 2021-03-19
- Laurent laurent@revelia Outgoing 2021-03-19
- Matt matt@revelia Outgoing 2021-03-17
- Caramel ghy@revelia Outgoing 2021-03-17
- Lena Moure lena@revelia Outgoing 2021-03-17
- Christophe christophe@revelia Outgoing 2021-03-17
- Marie-zoé marie-zoe@revelia Outgoing 2021-03-16
- Lara lara@revelia Outgoing 2021-03-16
- Nicolas nicolas@revelia Outgoing 2021-03-16
- Kids kids@revelia Outgoing 2021-03-16

Laurent Gagne

Meeting controls: Mute, Video, Chat, Screen Share, End Call

Meeting interface showing a list of participants on the left and a video grid on the right.

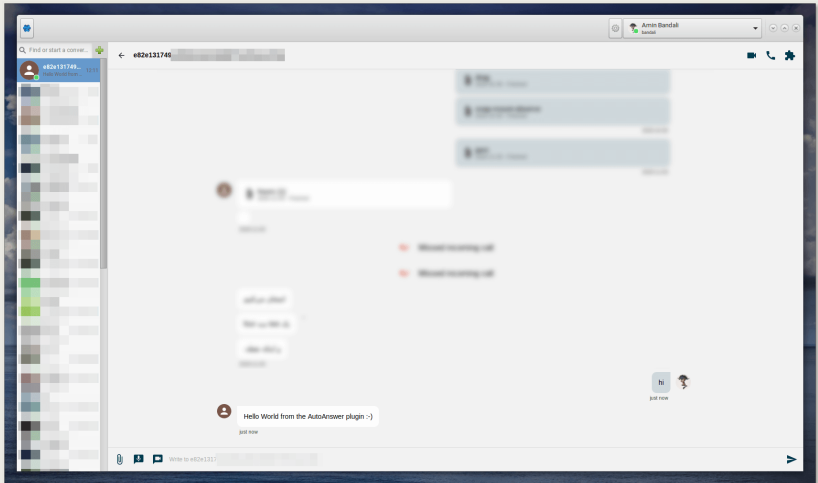
Participant List:

Name	Avatar	Status	Last Seen
Guillaume		Outgoing	09:07
Emeline		Outgoing	2021-03-19
Cake		Outgoing	2021-03-19
David		Outgoing	2021-03-19
Myrtille		Outgoing	2021-03-19
Harry		Outgoing	2021-03-19
Louane		Outgoing	2021-03-18
Laurent		Outgoing	2021-03-18
Matt		Outgoing	2021-03-17
Caramel		Outgoing	2021-03-17
Lena Moure		Outgoing	2021-03-17
Christophe		Outgoing	2021-03-17
Marie-zoé		Outgoing	2021-03-16
Lara		Outgoing	2021-03-16
Nicolas		Outgoing	2021-03-16
Kids		Outgoing	2021-03-16

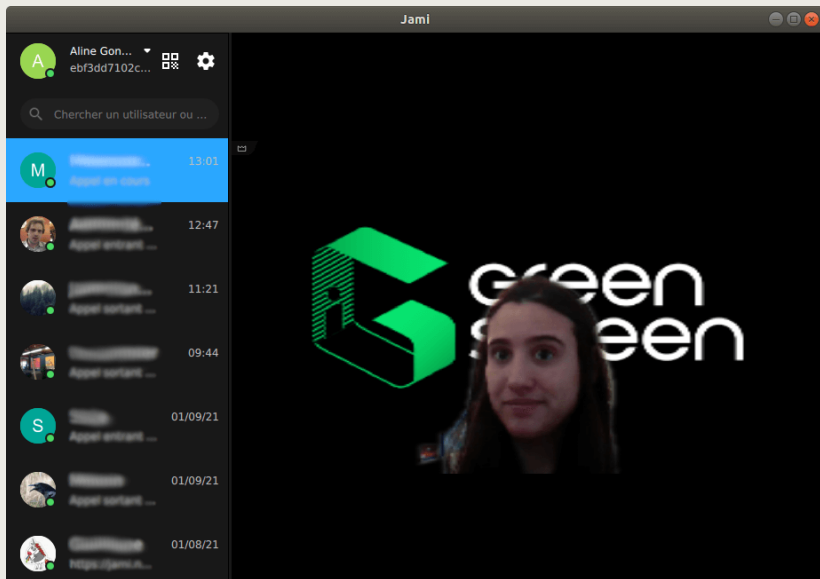
Video Grid:

- Top row: 7 small video thumbnails of participants.
- Main view: Large video of a participant wearing headphones and a microphone.

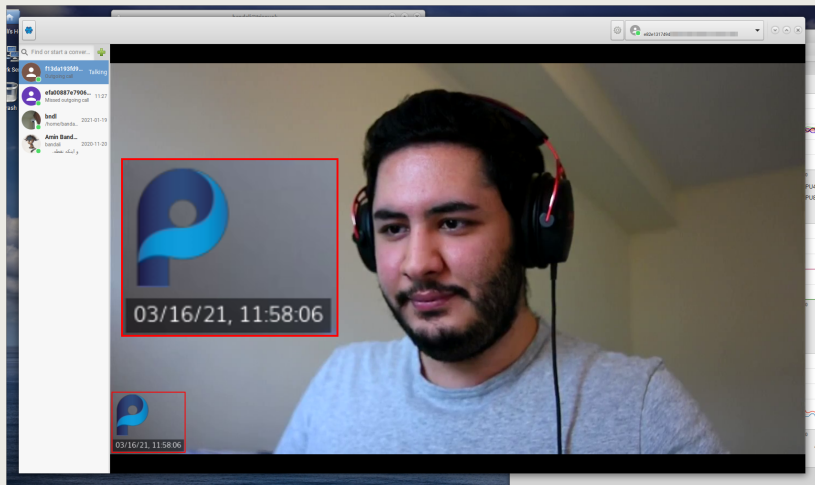
Plugins



AutoAnswer



GreenScreen



WaterMark

discover more Jami plugins

jami.net/plugins

learn more about the plugins SDK

jami.net/plugins-sdk

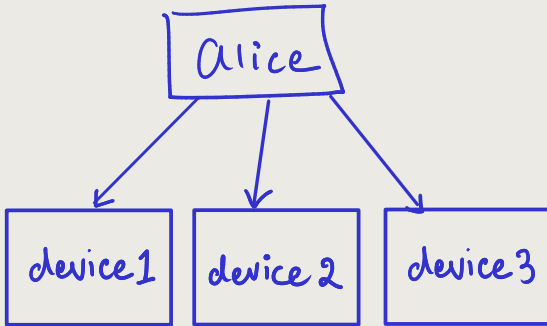
discover more Jami plugins

jami.net/plugins

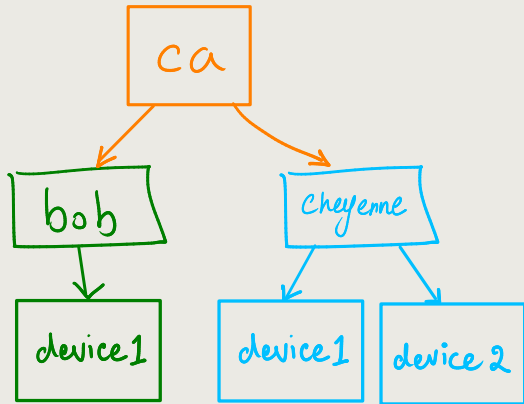
learn more about the plugins SDK

jami.net/plugins-sdk

JAMS

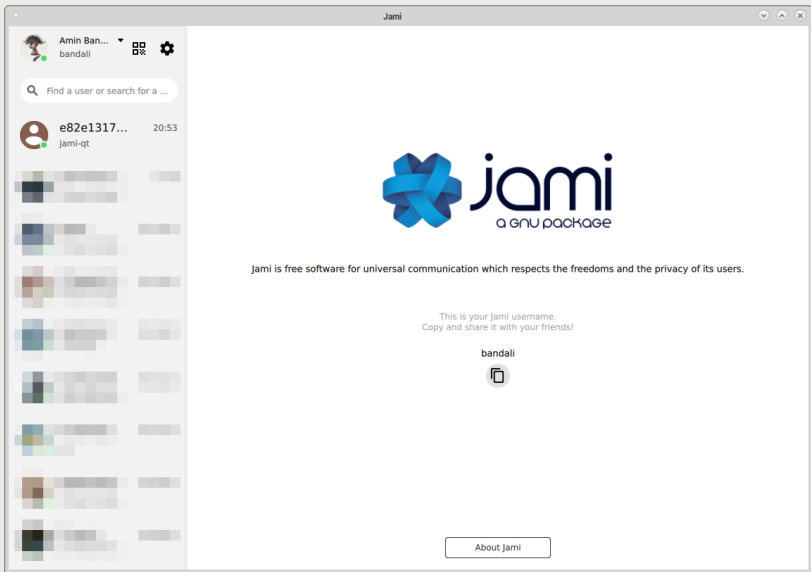


Jami account & device management

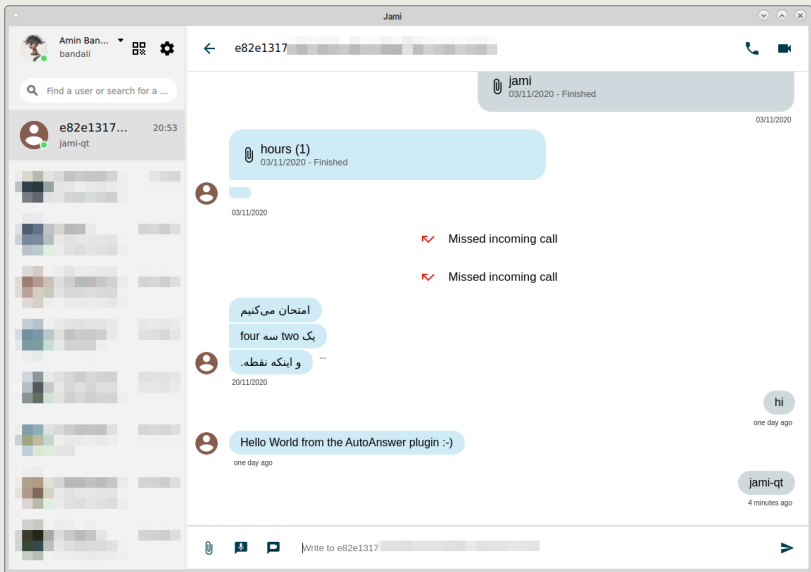


Jami account & device management (with JAMS)

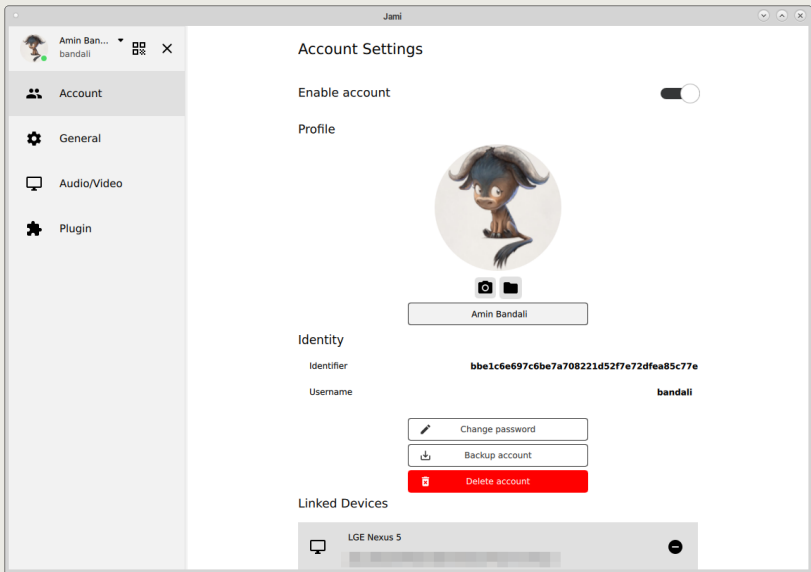
jami-qt **client**



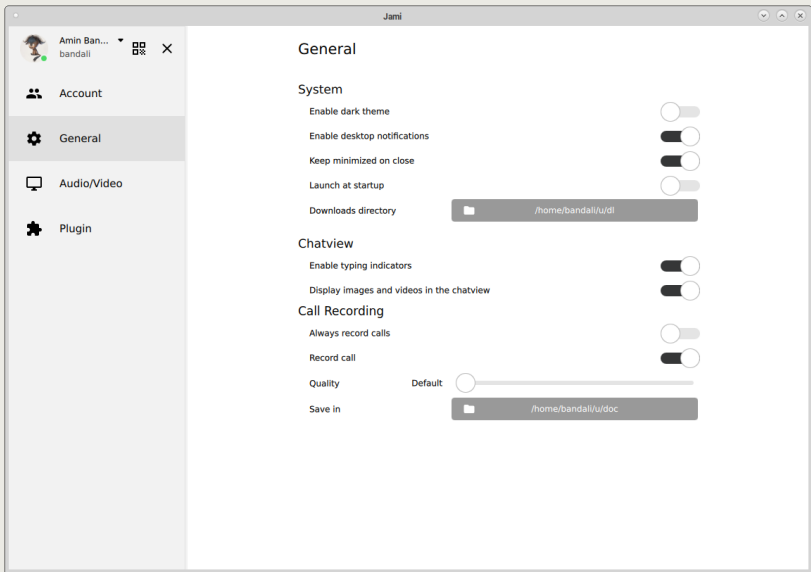
jami-qt - main window



jami-qt - chat view

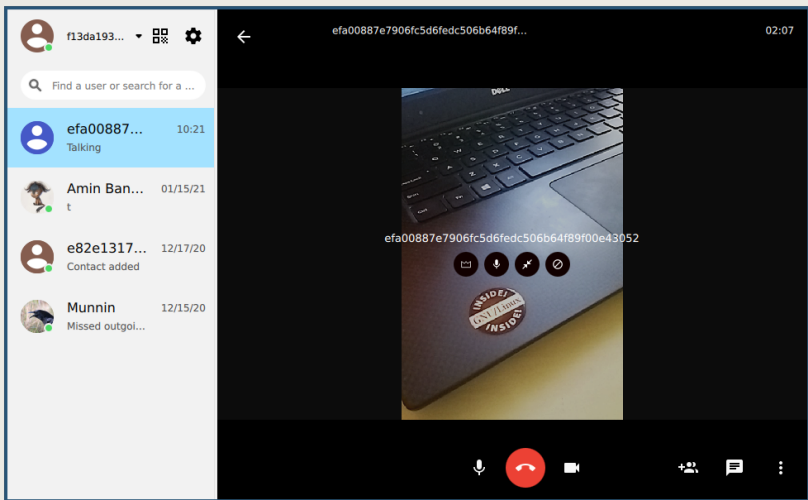


jami-qt - settings view (account)



jami-qt - settings view (general)

Moderation



jami-qt - moderation

Swarm chat

characteristic features of swarm chats:

- split and merge based on connectivity
- message history synchronization
- no central authority or server
- non-repudiation: each device must be able to verify validity of old messages and replay the whole history
- PFS during transport
- device-local storage

characteristic features of swarm chats:

- split and merge based on connectivity
- message history synchronization
- no central authority or server
- non-repudiation: each device must be able to verify validity of old messages and replay the whole history
- PFS during transport
- device-local storage

characteristic features of swarm chats:

- split and merge based on connectivity
- message history synchronization
- no central authority or server
- non-repudiation: each device must be able to verify validity of old messages and replay the whole history
- PFS during transport
- device-local storage

characteristic features of swarm chats:

- split and merge based on connectivity
- message history synchronization
- no central authority or server
- non-repudiation: each device must be able to verify validity of old messages and replay the whole history
- PFS during transport
- device-local storage

characteristic features of swarm chats:

- split and merge based on connectivity
- message history synchronization
- no central authority or server
- non-repudiation: each device must be able to verify validity of old messages and replay the whole history
- PFS during transport
- device-local storage

characteristic features of swarm chats:

- split and merge based on connectivity
- message history synchronization
- no central authority or server
- non-repudiation: each device must be able to verify validity of old messages and replay the whole history
- PFS during transport
- device-local storage

four modes for swarm chats:

- 1-to-1 (conversation with another Jami user)
- admin-invite-only (e.g. a class where the teacher can invite people, but not students)
- invite-only (a private group of friends)
- public (a public chat forum)

four modes for swarm chats:

- 1-to-1 (conversation with another Jami user)
- admin-invite-only (e.g. a class where the teacher can invite people, but not students)
- invite-only (a private group of friends)
- public (a public chat forum)

four modes for swarm chats:

- 1-to-1 (conversation with another Jami user)
- admin-invite-only (e.g. a class where the teacher can invite people, but not students)
- invite-only (a private group of friends)
- public (a public chat forum)

four modes for swarm chats:

- 1-to-1 (conversation with another Jami user)
- admin-invite-only (e.g. a class where the teacher can invite people, but not students)
- invite-only (a private group of friends)
- public (a public chat forum)

maintain a synchronized Merkle tree of messages

each conversation will be a git repository

maintain a synchronized Merkle tree of messages

each conversation will be a git repository

reasons for using git include:

- need to synchronize and order messages
- git is distributed by nature
- widely used and popular
- can verify commits using hooks and cryptographic signatures
- can be stored in a database if needed
- conflicts arising from splits/merges are much better handled by using commit messages rather than files

reasons for using git include:

- need to synchronize and order messages
- git is distributed by nature
- widely used and popular
- can verify commits using hooks and cryptographic signatures
- can be stored in a database if needed
- conflicts arising from splits/merges are much better handled by using commit messages rather than files

reasons for using git include:

- need to synchronize and order messages
- git is distributed by nature
- widely used and popular
- can verify commits using hooks and cryptographic signatures
- can be stored in a database if needed
- conflicts arising from splits/merges are much better handled by using commit messages rather than files

reasons for using git include:

- need to synchronize and order messages
- git is distributed by nature
- widely used and popular
- can verify commits using hooks and cryptographic signatures
- can be stored in a database if needed
- conflicts arising from splits/merges are much better handled by using commit messages rather than files

reasons for using git include:

- need to synchronize and order messages
- git is distributed by nature
- widely used and popular
- can verify commits using hooks and cryptographic signatures
- can be stored in a database if needed
- conflicts arising from splits/merges are much better handled by using commit messages rather than files

reasons for using git include:

- need to synchronize and order messages
- git is distributed by nature
- widely used and popular
- can verify commits using hooks and cryptographic signatures
- can be stored in a database if needed
- conflicts arising from splits/merges are much better handled by using commit messages rather than files

demo time!

download jami-qt
jami.net/download

Community

- `jami.net`
- `forum.jami.net`
- `jami@gnu.org`
- `#jami` on freenode
- `git.jami.net`
- `review.jami.net`
- `docs.jami.net`
- write and share your own plugins
- help translate Jami to your language(s)
- package Jami in your GNU/Linux distribution
- *use Jami and help spread the word <3*

- `jami.net`
- `forum.jami.net`
- `jami@gnu.org`
- `#jami` on freenode
- `git.jami.net`
- `review.jami.net`
- `docs.jami.net`
- write and share your own plugins
- help translate Jami to your language(s)
- package Jami in your GNU/Linux distribution
- *use Jami and help spread the word <3*

- jami.net
- forum.jami.net
- jami@gnu.org
- #jami on freenode
- git.jami.net
- review.jami.net
- docs.jami.net
- write and share your own plugins
- help translate Jami to your language(s)
- package Jami in your GNU/Linux distribution
- *use Jami and help spread the word <3*

- jami.net
- forum.jami.net
- jami@gnu.org
- #jami on freenode
- git.jami.net
- review.jami.net
- docs.jami.net
- write and share your own plugins
- help translate Jami to your language(s)
- package Jami in your GNU/Linux distribution
- *use Jami and help spread the word <3*

- `jami.net`
- `forum.jami.net`
- `jami@gnu.org`
- `#jami` on freenode
- `git.jami.net`
- `review.jami.net`
- `docs.jami.net`
- write and share your own plugins
- help translate Jami to your language(s)
- package Jami in your GNU/Linux distribution
- *use Jami and help spread the word <3*

- `jami.net`
- `forum.jami.net`
- `jami@gnu.org`
- `#jami` on freenode
- `git.jami.net`
- `review.jami.net`
- `docs.jami.net`
- write and share your own plugins
- help translate Jami to your language(s)
- package Jami in your GNU/Linux distribution
- *use Jami and help spread the word <3*

- `jami.net`
- `forum.jami.net`
- `jami@gnu.org`
- `#jami` on freenode
- `git.jami.net`
- `review.jami.net`
- `docs.jami.net`
- write and share your own plugins
- help translate Jami to your language(s)
- package Jami in your GNU/Linux distribution
- *use Jami and help spread the word <3*

- `jami.net`
- `forum.jami.net`
- `jami@gnu.org`
- `#jami` on freenode
- `git.jami.net`
- `review.jami.net`
- `docs.jami.net`
- write and share your own plugins
- help translate Jami to your language(s)
- package Jami in your GNU/Linux distribution
- *use Jami and help spread the word <3*

- jami.net
- forum.jami.net
- jami@gnu.org
- #jami on freenode
- git.jami.net
- review.jami.net
- docs.jami.net
- write and share your own plugins
- help translate Jami to your language(s)
- package Jami in your GNU/Linux distribution
- *use Jami and help spread the word <3*

Thanks !

Amin Bandali <bandali@gnu.org>

kelar.org/~bandali/2021/03/20/jami-empowers-users.html

(GPLv3+)