

VERIFIED WORK HISTORY FOR INDEPENDENT SOFTWARE ENGINEERS

Your shipped work is public. Your work history is not portable.

Independent engineers already leave high-signal evidence in merged GitHub pull requests. What disappears is the context: who confirmed the contribution, what authority they had, and whether the receipt is still valid when a platform or client relationship ends.

The Web2 failure

A link proves an artifact exists. A testimonial names no artifact. A platform review cannot travel. Recruiters must either trust screenshots or repeat the verification from scratch.

1

Artifact proof

GitHub API facts for one public merged PR, normalized and hashed.

2

Issuer proof

An identifiable issuer signs the exact, frozen contribution claim; maintainer authority is a separate assurance level.

3

Chain proof

SAS exposes credential authority, signer, network, expiry and closure state.

[Dashboard](#)
[Import PR](#)
[Trust model](#)

[Sign in](#)

tang-vu submitted a self-issued claim for bug fix work represented by PR #2016; no independent confirmation.

Claim authorized 8/10/2026 Credential 8cd4d8-b0f7-42f8-97cc-bf4ca57a2bf8 ACTIVE

Technical receipt, not independent confirmation. The subject and issuer are the same GitHub identity. This receipt proves the signed payload and SAS lifecycle work end to end; it does not prove that a maintainer independently confirmed the contribution.

1 Artifact proof

Machine-derived GitHub facts frozen into an immutable normalized snapshot.

REPOSITORY Osmantic/ODS · ID 1153579168	PULL REQUEST #2016 - fix(dashboard): stop the usage sparkline losing the month you came from ✓
CONTRIBUTOR @tang-vu · GitHub ID 145498528	AUTHOR ASSOCIATION CONTRIBUTOR
MERGE TIMESTAMP 7/25/2026, 6:59:48 PM	MERGED COMMIT SHA 5717705bf6241eff3a2ade1f8fa4527e0e619e8c
SNAPSHOT HASH 942ae14729861c48ad8693407c9a1c75eb5dab51ba2c14d1dc0315e93c9bb71d9	EVIDENCE SOURCE GitHub REST API

GitHub authorship identifies the pull request author; it does not prove they wrote every line in a collaborative pull request.

Portable credential

The downloadable bundle includes the complete public claim, normalized artifact, issuer signature, and chain references needed for independent checking.

```
PAYLOAD HASH
39ae52211cb66c1eda9db364edfb
4627cae9fc157ffef8a3518123449e
f66ef71d7
```

CREDENTIAL
Bug fix

Download JSON

Verify without ProofPort

View @tang-vu

Live verifier-first receipt from GitHub OAuth + Phantom + SAS devnet, plus finalized two-key fee sponsorship. The public receipt is explicitly self-issued—technical validation, not independent maintainer confirmation.

ISSUER

@tang-vu · GitHub ID 145498528

IDENTIFICATION

Self-issued: issuer and subject are the same GitHub identity; no

Why Solana

A neutral registry lets unrelated issuers, engineers and hiring products inspect the same credential state without depending on ProofPort's database. Each issuer owns an isolated credential; a sponsor can pay gas without gaining issuer authority.

What Solana cannot do

It cannot make an issuer honest.
ProofPort exposes identity and
assurance instead of turning every
onchain claim into a green check.

One receipt. Three proof layers. No verifier account.

Start with the strongest public software artifact: a merged pull request. Keep private code, contracts, emails, OAuth tokens and personal documents offchain.

01
Developer signs in with GitHub

02
Import one public merged PR

03
Freeze API evidence and claim hash

04
Maintainer reviews a single-use request

05
Sponsor pays; issuer wallet remains required signer

06
Anyone verifies and downloads the bundle

P ProofPort

DashboardImport PRTrust model

@tangSign out

DEVELOPER FLOW · STEP 1 OF 2

Import a merged pull request

GitHub—not the form—supplies the artifact facts that become the receipt.

Development account: only the labeled fixture artifact is available. It will never be shown as GitHub API verified.

Describe your contribution

NORMALIZED

Fixture artifact · not fetched from the live GitHub API

REPOSITORY

proofport-demo/ledger

PULL REQUEST

#281 · Add idempotent receipt issuance API

AUTHOR

@tang · ID 1001

MERGED

7/19/2026, 9:42:18 PM

GitHub proves

Public artifact metadata, PR author identity and merged state.

Issuer claims

The named contribution description is accurate.

Solana proves

The configured signer created the integrity reference and its current lifecycle.

Live execution evidence

Founder–problem fit backed by public work

Tang Minh Vu is a Ninh Binh-based full-stack engineer with 3+ years shipping SaaS and developer tools, 10+ production systems, 10+ upstream merged PRs, ContribAI (244★), and a 2026 agent hackathon win. He built this complete TypeScript/SAS vertical slice. Public evidence includes a browser receipt, close trace and two sponsor-paid transactions. The unfunded issuer owned and signed its credential; receipt two reused the credential/schema. **Pre-revenue; no users or partnerships are claimed.**

Business model

Free

Developer passport and public verification

Recruiter SaaS/API

Verification workflow, search and audit trail

Managed issuance

Organization controls and bulk receipts

The passport is the wedge; the fundable product is neutral receipt infrastructure embedded by hiring products, bounty ecosystems and engineering organizations.

Next 90 days

- Run 10 Vietnam-based engineer, maintainer and recruiter interviews; measure comprehension, time saved and willingness to pay.
- Secure one independent maintainer receipt; ship a fine-grained GitHub App and verifier SDK.
- Deploy PostgreSQL-backed production and pilot managed issuance; keep payments and private repos out.