GT-Acredita

Aplicação de CREDenciais verificáveis para Identidade digital e Acesso

Emerson Ribeiro de Mello

mello@ifsc.edu.br

02 de junho de 2026

INSTITUTO FEDERAL
Santa Catarina
Câmpus São José

RNP

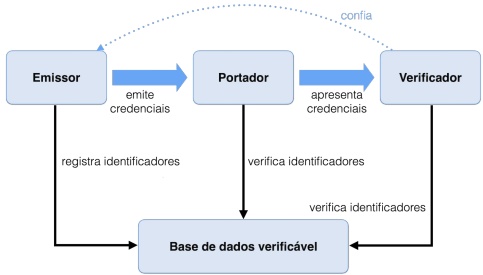
GT-Acredita

- Emerson Mello (IFSC)
- Rodrigo Lira (IFPE)
- Kauan Freitas (IFSC)
- Marcos Wagner (IFSC)
- João Vitor (UFRGS)

- Execução de 01/04/2025 a 31/12/2025

Objetivos do Grupo de Trabalho

- Uma **prova de conceito** de aplicação emissora e verificadora de credenciais verificáveis
  - Base: Projeto Ilíada[1] (infraestrutura blockchain e APIs)
- Explorar o uso de **credenciais verificáveis** para autenticação de usuários no JEMS3.
  - SBC como emissora e verificadora inicial



Fonte: Adaptado de W3C VC 2.0 Data Model

[1] https://iliadablockchain.org.br/

Prova de conceito

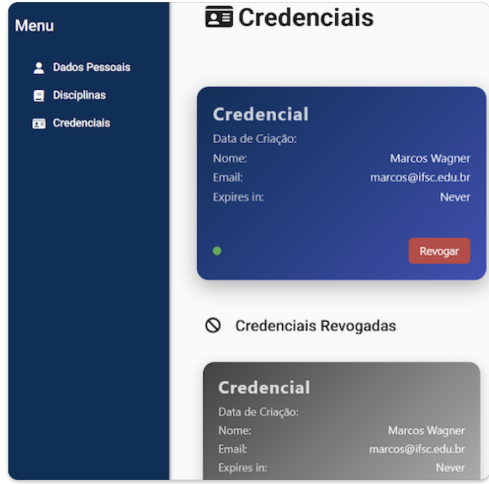
Uso de VCs no processo de autenticação de usuários

1. Usuário solicita emissão de credencial verificável (VC) na aplicação emissora
2. Uso de carteira digital (PWA) para armazenar a VC
3. Usuário fornece uma VC para autenticação na aplicação verificadora

Prova de conceito

Uso de VCs no processo de autenticação de usuários

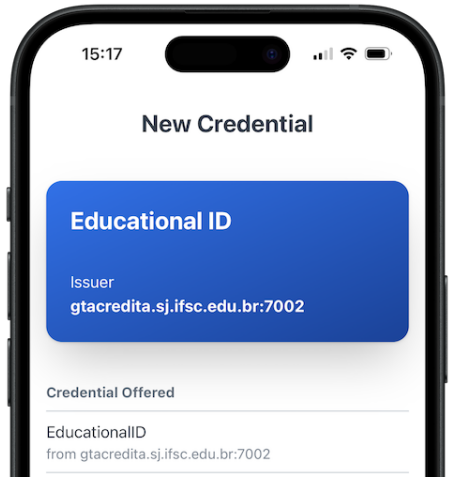
1. Usuário solicita emissão de credencial verificável (VC) na aplicação emissora
2. Uso de carteira digital (PWA) para armazenar a VC
3. Usuário fornece uma VC para autenticação na aplicação verificadora



Prova de conceito

Uso de VCs no processo de autenticação de usuários

1. Usuário solicita emissão de credencial verificável (VC) na aplicação emissora
2. Uso de carteira digital (PWA) para armazenar a VC
3. Usuário fornece uma VC para autenticação na aplicação verificadora



Prova de conceito

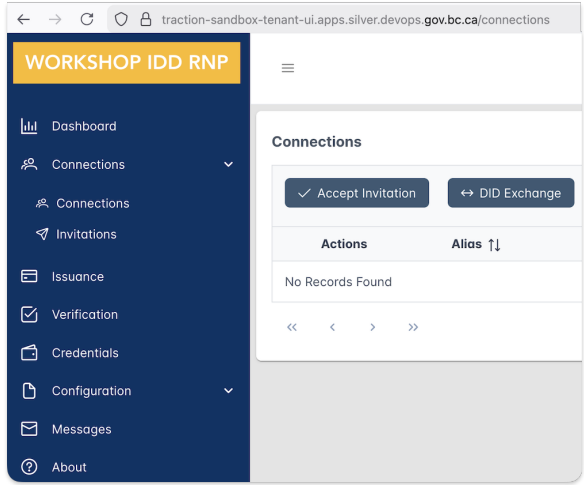
Uso de VCs no processo de autenticação de usuários

1. Usuário solicita emissão de credencial verificável (VC) na aplicação emissora
2. Uso de carteira digital (PWA) para armazenar a VC
3. Usuário fornece uma VC para autenticação na aplicação verificadora

Tecnologias estudadas

PoC com Hyperledger Indy I

- Aplicativo móvel BCWallet
- BC Gov Traction e ACA-Py
- `did:indy` e protocolo DIDComm
- VCs no formato *Anoncreds*



PoC com Hyperledger Indy II

- Hyperledger Indy descontinuado e Anoncreds com problemas de escalabilidade
- Indy Besu – Hyperledger Besu com *Smart Contracts* para suporte a IDD
  - Anoncreds, `did:indy:besu` e DIDComm
- Experiência do CPqD e alinhamento com o projeto Ilíada

Escolhas tecnológicas para próxima PoC

- **Formato de credenciais**
  - W3C VC ou IETF SD-JWT
- **Procolo para emissão e verificação**
  - OID4VCI e OID4VP
- **DID Method**
  - `did:ethr` (com Besu), `did:jwk` ou `did:key`
- **Carteiras digitais**
  - wwWallet - por ser uma PWA
  - Walt.ID - PWA, solução completa e com boa documentação

> **Nota**
>
> Versão 1.0 da OID4VCI foi publicada em setembro de 2025

wwWallet

https://github.com/wwWallet

- *Progressive Web App* (PWA), open source e sendo usada por algumas iniciativas europeias
  - Suporte a chaves de acesso (*passkeys*), *Bitstring Status List* v1.0
  - Escrito em Express.js
- `did:key`, `did:ebsi`, OID4VC e SD-JWT VC
- Demo com emissor e verificador

wwWallet

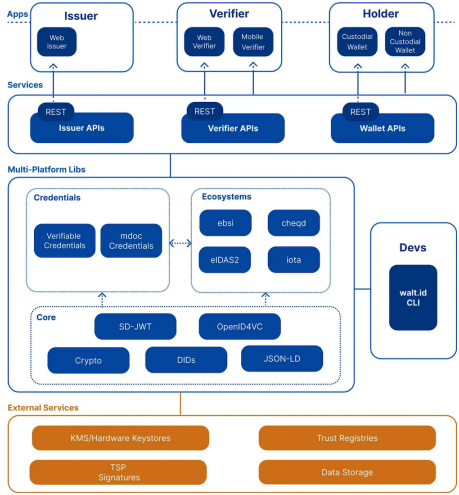
https://github.com/wwWallet

- *Progressive Web App* (PWA), open source e sendo usada por algumas iniciativas europeias
  - Suporte a chaves de acesso (*passkeys*), *Bitstring Status List* v1.0
  - Escrito em Express.js
- `did:key`, `did:ebsi`, OID4VC e SD-JWT VC
- Demo com emissor e verificador

> **Nota**
>
> Documentação incompleta, código com forte acoplamento entre componentes, dificultando a extensão ou modificação.

Walt.ID I

- SDK completo para emissão, verificação e gerenciamento de VCs
- Carteira no formato PWA
- Suporte a múltiplos formatos de credenciais e protocolos

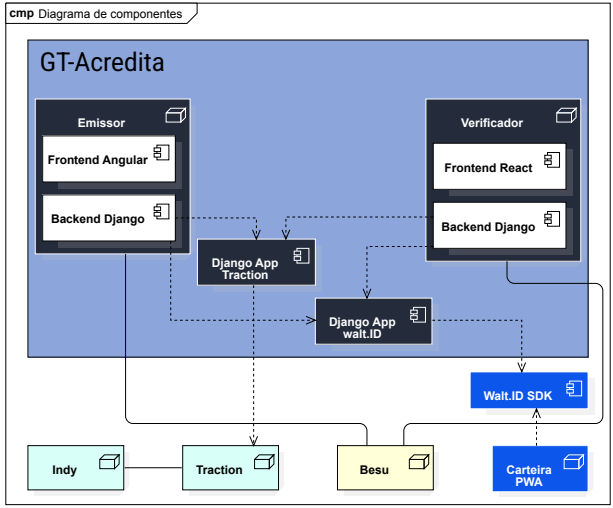


Fonte: https://docs.walt.id

Walt.ID II

- **PoC 1**
  - W3C VC 1.1, OID4VCI, Bitstring Status List v1.0, `did:jwk`
  - Verificador indica em quais DIDs confia como emissores de credenciais
  - Testado com duas carteiras digitais: Walt.ID e Sphereon
- **PoC 2**
  - `did:ethr` com Besu, W3C VC 1.1, OID4VC, Bitstring Status List v1.0
  - Solidity Smart Contracts e Hardhat ignition para implantação

Arquitetura da solução desenvolvida



Considerações finais

- Optamos em não realizar a integração com o JEMS3
  - Falta de maturidade da base tecnológica
- Uso de VCs para autenticação de usuários é viável, porém gera muito atrito
  - W3C Digital Credential API[2] ainda está em desenvolvimento
- Próximo passo é deixar VCs para a primeira autenticação e depois migrar para chaves de acesso (*passkeys*) para autenticações subsequentes
  - Sob eIDAS 2.0, carteiras digitais adotam *WebAuthn* por padrão e VCs somente quando legalmente necessário

[2] https://www.w3.org/TR/digital-credentials/

Obrigado!

Emerson Mello
https://emersonmello.me