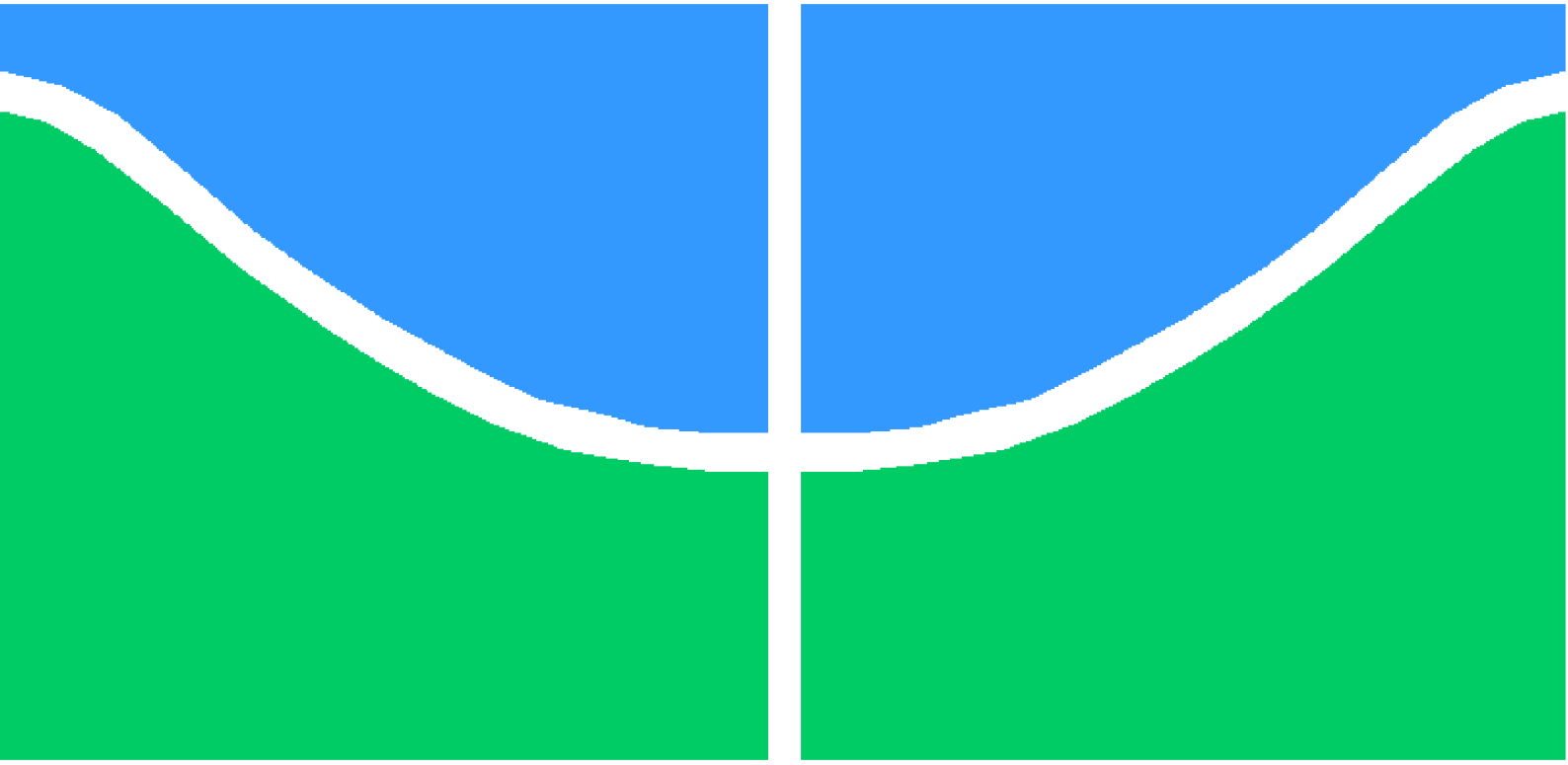


Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Engenharia de Software

# Implementando a Identidade Auto Soberana Através do Solid

Autor: Victor Hugo Siqueira Costa

Orientador: Prof. Dr. Fernando William Cruz

Brasília, DF

2024

Victor Hugo Siqueira Costa

# Implementando a Identidade Auto Soberana Através do Solid

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Prof. Dr. Fernando William Cruz

Brasília, DF

2024

Victor Hugo Siqueira Costa

# Implementando a Identidade Auto Soberana Através do Solid

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Brasília, DF
2024

# Agradecimentos

Minha motivação de todo dia é a mesma que me motiva a concluir esse trabalho, para provar diariamente minhas competências e porquê mereço ser o companheiro da minha motivação diária e esposa, Caroline Marinho Cardoso, a qual dedico meus esforços e minha vida. Assim como minha mãe Myllena, minha avó Denise e toda minha família, que me ajudaram em vários momentos difíceis, que apesar das dificuldades, me permitiram estar presente nesse momento para poder concluir essa etapa importante em minha vida. A ajuda do meu orientador Fernando William Cruz, assim como de todos os professores que me ajudaram em diferentes momentos da minha formação pessoal e profissional, também foi indispensável.

*"Até aqui sobrevivi... No meio de incertezas eu me fortaleci."*

# Resumo

A centralização de dados em grandes monopólios tecnológicos tem facilitado vazamentos e violações de privacidade, enquanto práticas como o bloqueio injustificado de contas e o aumento de fraudes *online*, como *phishing* e *spoofing*, complicam a segurança da identidade digital. Diante disso, o conceito de Self-Sovereign Identity (SSI), ou Identidade Auto Soberana, emerge como uma solução inovadora, devolvendo aos usuários o controle sobre suas informações, ao mesmo tempo que aborda questões como privacidade e segurança online. Este trabalho demonstra a implementação de uma solução para SSI, através do protocolo Solid, comparando o com outras alternativas de arquitetura para SSI e aplicando-a no contexto educacional do projeto SmartUnB.ECOS. A proposta visa descentralizar a gestão de identidades e dados, promovendo privacidade e autonomia em ambientes digitais complexos. Através do desenvolvimento de um servidor Solid e da análise das necessidades do usuários da *Web* comum, demonstra-se a viabilidade de identidades e armazenamentos descentralizados, incentivando futuras pesquisas e a adoção de uma internet mais segura e alinhada com os princípios de soberania dos dados.

**Palavras-chave**: Identidade Auto Soberana, Solid, Pod, Identificadores Descentralizados, Identidade Digital, Privacidade, Segurança, Descentralização, *Web*, Internet, *blockchain*.

# Abstract

The centralization of data in large technology monopolies has facilitated data leaks and privacy violations, while practices such as unjustified account blocking and the rise of online fraud, such as phishing and spoofing, complicate digital identity security. Therefore, the concept of Self-Sovereign Identity (SSI) emerges as an innovative solution, giving users back control over their information while addressing issues such as privacy and online security. This work demonstrates the implementation of an SSI solution using the Solid protocol, comparing it with other SSI architecture alternatives and applying it to the educational context of the SmartUnB.ECOS project. The proposal aims to decentralize identity and data management, promoting privacy and autonomy in complex digital environments. Through the development of a Solid server and the analysis of the needs of common *Web* users, the viability of decentralized identities and storage is demonstrated, encouraging future research and the adoption of a more secure internet aligned with the principles of data sovereignty.

**Keywords**: Self-Sovereign Identity, Solid, Pod, Decentralized Identifiers, Digital Identity, Privacy, Security, Decentralization, *Web*, Internet, *blockchain*.

# Lista de ilustrações

# Lista de tabelas

# Sumário

# 1 Introdução

A era digital trouxe consigo avanços tecnológicos sem precedentes, mas também expôs fragilidades significativas no que diz respeito à privacidade e ao controle dos dados pessoais. No livro A Era do Capitalismo de Vigilância, (ZUBOFF, 2019) destaca a falta de controle dos usuários sobre suas informações, um problema que tem gerado efeitos colaterais negativos para a sociedade. Entre esses efeitos, está a prática de prestadores de serviços negarem o acesso aos titulares dos dados, muitas vezes sem justificativas claras, resultando em encerramentos de contas e desativações abruptas. Além disso, a centralização de dados em grandes monopólios tecnológicos tem facilitado vazamentos de informações, seja por ataques cibernéticos ou falhas sistêmicas, violando a privacidade dos usuários de forma recorrente.

Paralelamente, (SOLTANI; NGUYEN; AN, 2021) explica como a ascensão de fraudes online, *phishing* e *spoofing* por exemplo, tem complicado ainda mais a questão da identidade digital. Em um cenário onde é cada vez mais difícil garantir que um usuário é realmente quem diz ser, a necessidade de um modelo mais seguro e descentralizado de gerenciamento de identidades torna-se urgente. É nesse contexto que surge o conceito de Self-Sovereign Identity (SSI), ou Identidade Auto Soberana, uma abordagem que busca devolver aos usuários o controle sobre suas identidades digitais, inspirada em princípios de privacidade e segurança que remontam à década de 1970.

Este trabalho tem como foco explorar o potencial da SSI para a Internet, junto com o protocolo Solid, que é uma iniciativa que visa descentralizar a *Web* e garantir maior autonomia aos usuários. A proposta é aplicada no contexto educacional, utilizando como contextualização o projeto SmartUnB.ECOS, que desenvolve e une tecnologia para redes acadêmicas, onde em certos contextos é necessário autorização para utilizar algum recurso. A integração entre o protocolo Solid e o e demais tecnologias de comunicação, busca não apenas promover a descentralização da *Web*, mas também criar um ambiente educacional que respeite a privacidade e a soberania dos dados dos usuários.

O objetivo é implementar um sistema que seja tecnicamente viável e descentralizado, permitindo aos usuários controlar suas identidades e dados, no ambiente acadêmico onde costumam se identificar para acessar algum site, se matricularem, pagar refeições, mesmo em um ambiente complexo como a universidade. Além disso, o trabalho busca motivar futuras pesquisas e desenvolvimentos na área, destacando a importância de uma Internet mais segura, privativa e alinhada com os princípios de descentralização.

Ao longo desta monografia, serão abordados os desafios e as oportunidades relacionados à adoção da SSI, bem como os resultados obtidos com a integração proposta.

Espera-se que este estudo contribua para a discussão sobre a necessidade de uma transformação digital que coloque os usuários no centro, garantindo-lhes o controle sobre suas identidades e dados em um mundo cada vez mais conectado e dependente da tecnologia.

## 1.1 Problema

A centralização de dados na Internet moderna tem gerado uma série de problemas relacionados à privacidade, segurança e controle das informações pessoais dos usuários, como menciona (ZUBOFF, 2019). Grandes monopólios tecnológicos concentram vastas quantidades de dados, tornando-se alvos frequentes de ataques cibernéticos e vazamentos, muitas vezes expondo informações sensíveis sem o consentimento dos titulares. O fato desses dados estarem sob controle de terceiros, abre espaço para muitas práticas questionáveis de comercialização de informações pessoais. Diante desse cenário, surge a necessidade de um modelo alternativo que devolva aos usuários o controle sobre suas identidades e dados, garantindo maior privacidade, segurança e autonomia.

## 1.2 Objetivos

Os objetivos desta pesquisa visam abordar a problemática da centralização de dados, e propor uma solução baseada em tecnologias descentralizadas, comparando tecnologias de arquitetura como *blockchains* e o projeto Solid, para devolver aos usuários o controle sobre suas identidades digitais.

### 1.2.1 Objetivo Geral

Este trabalho tem como objetivo geral explorar e propor uma solução para a Identidade Auto Soberana (SSI) com o protocolo Solid, no contexto educacional, em situações que requerem identificação, fazendo menção ao projeto SmartUnB.ECOS, a fim de promover a descentralização de dados, a privacidade e a autonomia dos usuários.

### 1.2.2 Objetivos Específicos

- **OE1:** Realizar buscas em bases de dados científicas reconhecidas com palavras chaves aos temas mencionados, com o objetivo de achar referencial aprofundado sobre os temas.
- **OE2:** Selecionar artigos recentes e relevantes sobre SSI, assim como comunidades voltadas ao tema. Em conjunto, utilizar alguma obra principal que reconheça as preocupações de 'A Era do Capitalismo de Vigilância'.

- **OE3:** Estudar e desenvolver sob tecnologias para favorecer a descentralização da *Web*.
- **OE4:** Elaborar uma proposta de desenvolvimento utilizando tecnologias que favoreçam a SSI, conseguindo agregar valor ao contexto educacional.

## 1.3 Metodologia

Por meio da criação da elaboração de um referencial teórico com bases de periódicos reconhecidos, os conceitos de SSI serão aprofundado, bem como o conhecimento a cerca do protocolo Solid. A metodologia adotada inclui o desenvolvimento, instalação e configuração de um servidor Solid, a análise de aplicações existentes e a adaptação de funcionalidades para o contexto educacional. A metodologia foi divida em Objetivo Geral e Objetivos Específicos, para direcionarem o Referencial Teórico 2, servindo como validação para a 3 e 4. As etapas são:

- Realizar buscas na base de dados da CAPES, IEEE Xplore, ScienceDirect, Scopus, ResearchGate e arXiv utilizando palavras-chave como 'identidade auto soberana', 'Solid', 'sistemas de gerenciamento de identidade digital', 'riscos de privacidade', 'segurança de identidade online' e 'centralização de dados'.
- Selecionar artigos recentes para garantir o entendimento das aplicações do conceito de SSI. Além disso pesquisar em comunidades voltadas para SSI ou DID, se baseando em um dos principais livros do tema do (PREUKSCHAT; REED, 2021).
- Além de pesquisar nessas bases por assuntos que envolvam 'Solid' com 'SSI' e derivados, investigar casos de uso reais onde o protocolo Solid foi implementada para descentralizar dados e proteger a privacidade dos usuários. Estudar o projeto SmartUnB.ECOS para entender como o ecossistema está sendo implementado e como colaborar, para entender suas possibilidades e funcionamento.
- Elaborar alguma proposta de implementação que utilize de Pods e aplicações Solid, agregando valor para o contexto educacional, integrando de forma útil demais tecnologias.

# 2 Referencial Teórico

Para a base do desenvolvimento da solução proposta, é necessário compreender o crescente uso de plataformas digitais e a massiva troca de dados na Internet, a forma como as identidades são gerenciadas e controladas tem se tornado um tema central de debate. Esse capítulo aborda a evolução do conceito de Identidade Digital, explorando suas características, os modelos de controle e autenticação que dominam o setor, e o impacto que isso gera para os usuários. Além disso, são feitas análises sobre as implicações tecnológicas do controle dos dados e identidades, de forma centralizado, destacando as desvantagens e os riscos associados à centralização de dados pessoais em poucas corporações, como perda de privacidade, manipulação de informações e a exploração comercial das identidades digitais.

Na sequência, o capítulo explora as possíveis alternativas ao modelo centralizado de identidades digitais, com foco nas implementações de Identidade Auto Soberana (SSI) e nas tecnologias descentralizadas, como a *blockchain* e o protocolo Solid. Casos de uso práticos e atuais serão apresentados para ilustrar como essas tecnologias emergentes oferecem novas soluções para proteger a privacidade do usuário e devolver o controle sobre sua identidade. Esses tópicos visam proporcionar uma base sólida para entender as oportunidades e os desafios que envolvem a implementação de novos modelos de gestão de identidade digital em um cenário cada vez mais globalizado e digitalizado.

## 2.1 Digitalização da Identidade

A definição moderna de identidade foi primeiramente utilizada em 1950 por Erik Eriksson, sendo definida por Glasser e Vajihollahi como uma representação lógica de uma presença física de uma pessoa ou um objeto. Wang e Filippi também definem identidade como todos os atributos de uma pessoa que unicamente definem essa pessoa ao longo do seu tempo de vida fornecendo uniformidade e continuidade apesar de aspectos e condições variados, como é explicado por (SOLTANI; NGUYEN; AN, 2021).

Com a digitalização da informação, devidos aos avanços tecnológicos dos dois últimos séculos, que tornaram a Internet globalmente acessível, era de se esperar que a forma de identificação das pessoas, em todos os meios, também seria digital. Por questões de conveniência e da natureza até então complexa da tecnologia da informação, os dados e as informações relacionado a essas identidades, era centralizada no domínio dos serviços *online* onde eram necessárias.

Atualmente existem várias definições de identidade digital. Por exemplo, (SOLTANI; NGUYEN; AN, 2021) menciona que a União Internacional de Telecomunicações (UIT) define identidade digital como uma representação digital da informação conhecida sobre um indivíduo, grupo ou organização em especifico. Para cada entidade, pode haver múltiplas identidades parciais e também não são necessariamente iguais às identidades do mundo real, uma vez que as identidades digitais representadas *online* podem diferir das características representadas no mundo físico.

Outra definição similar é explicada por (KASSEM, 2019), que a caracteriza como um estabelecimento digital dos dados disponíveis sobre uma entidade, e o sistema de gerenciamento de identidade garante que apenas os usuários válidos estejam autorizados a obter acesso a essas informações. Além disso devido ao uso crescente, a capacidade de autenticar identidade digital tornou-se mais significativa para o desenvolvimento de tecnologias, serviços e padrões.

A identidade digital se tornou essencial para a vida cotidiano das pessoas ao redor do mundo. Com base nas estimativas do Banco Mundial, existem mais de 1,1 bilhões de indivíduos que não possuem uma identidade oficial e 3,5 bilhões de pessoas em todo o mundo que não têm acesso a serviços bancários. Essas identidades são relacionadas aos governos e agências bancárias que as utilizam pra conceder benefícios, direitos ou restrições, de acordo com seus critérios.

As identidades não são criadas somente para casos essenciais como uma identidade nacional ou bancária, como é o caso das contas *online* de redes sociais que possuem bilhões de usuários ao redor do mundo que compartilham suas informações pessoais ou outros assuntos com os demais usuários sendo eles conhecidos ou não. Para os indivíduos, os modelos de identidade existentes não os colocam no controle imediato dos seus dados de identidade, ou possuem visibilidade limitadas sobre como os seus dados pessoais são geridos, partilhados e descartados pelos prestadores de serviços.

Também existe a possibilidade da comercialização dos dados relacionados a identidades dos usuários, o que possibilitou a existência de um dos mercados mais lucrativos atualmente, que é o dos provedores de identidade que geram renda coletando dados comportamentais de seus usuários. Esses dados são usados para desenvolver sistemas sofisticados de análise e previsão do comportamento do usuário, que em seguida, são negociados em mercados onde os anunciantes podem selecionar um público-alvo adequado para seus produtos. Além disso, a maioria dos provedores de serviços *online* apoiam seu próprio conjunto específico de políticas e práticas de gerenciamento de dados, levando a monopólios industriais e aprisionamento de fornecedores.

### 2.1.1 Problemas da Identidade Centralizada

No livro 'A Era do Capitalismo de Vigilância', de Shoshanna Zuboff, ela trata a falta de controle do usuário sobre seus dados, como já mencionado, pelos efeitos colaterais negativos que isso gera para toda a sociedade. Um deles, é o de fazer com que alguns prestadores de serviços neguem o acesso aos titulares dos dados a qualquer momento, por motivos que considerem justificados. Isto levou a muitos casos de encerramento de contas e desativação por parte dos prestadores de serviços, com pouca explicação aos titulares dos dados, sendo mencionado por (SOLTANI; NGUYEN; AN, 2021).

Embora certas revogações e encerramentos de contas de usuários possam ser considerados aceitáveis, a análise e o compartilhamento não autorizado de dados pessoais por provedores de serviços viola a privacidade do usuário. As vezes esse compartilhamento autorizado não ocorre intencionalmente, mas devido a centralização de dados nesses monopólios mencionados, eles se tornam alvos principais de diversos ataques cibernéticos, ou de simplesmente ocorrer falhas nos seus sistemas, que geram vazamentos de dados.

Outros problemas ganham evidência à medida que o usuário também precisa estar atento para fraudes *onlines*. É explicado por (KASSEM, 2019), sobre a 'era da engenharia social', onde existem técnicas que envolvem disfarçar uma tela de acesso falsa, mensagens de alertas enganosos, vírus de computador, entre outros conhecidos como *phishing* ou *spoofing*, para capturar informações pessoais privadas. O termo identidade fica mais complexo com os desafios de saber, com segurança, se o usuário é de fato quem ele diz ser, graças a diversas fragilidades no modelo de dados centralizado vigente.

#### 2.1.1.1 Autenticação a Base de Senhas

Um dos principais motivos de ocorrer esses vazamentos, é a necessidade de sistemas de identificação a base de senhas, onde normalmente se utilizam senhas fracas ou de fácil adivinhação relacionados a informações já públicas dos usuários, como endereços, datas de aniversários, nomes de parentes entre outros. Isto se deve em parte ao número esmagadoramente grande de serviços *online* que precisam ser gerenciados por um sujeito, o que faz com que também sejam reutilizadas em mais de um serviço. Caso ocorra um vazamento em um serviço, muito provavelmente a senha daquele usuário será a senha de outras plataformas.

Em média, um usuário empresarial do aplicativo gerenciador de senhas LastPass precisa rastrear 191 senhas. Devido às dificuldades associadas à gestão de contas, muitos usuários utilizam as suas contas sociais, como o Facebook e o Google, para se autenticarem em serviços *online*, tornando as suas contas de redes sociais sujeitas a mais riscos do que antes e mais dependentes de poucos serviços centralizados (SOLTANI; NGUYEN; AN, 2021). Alternativas mais seguras como autenticação de dois fatores e biometrias precisam

ser mais facilmente utilizáveis, como no caso de computadores pessoais, além do próprio *smartphone*.

#### 2.1.1.2 Dados Pessoais Fragmentados

Outro ponto importante, este principalmente relacionado a venda dos dados pessoais já citado, é a crescente dificuldade de gerenciar os próprios dados, assim como as senhas de acesso, nos milhões de repositório de dados onde estes acabam sendo espalhados e com diferentes regras e permissões de gerenciamento. Se torna exponencialmente impossível saber onde todos os dados pessoais estão localizados para posteriormente poder apagá-los caso fosse possível o usuário fazer isso em algum momento.

Um usuário *online* comum tem seus dados dispersos entre vários centros de dados governamentais, financeiros e sociais, de formas duplicadas, incompatíveis e provavelmente desatualizados. Muitas vezes, uma única organização está fragmentada por unidades de negócios ou produtos, o que aumenta esta complexidade (SOLTANI; NGUYEN; AN, 2021). Essa fragmentação, em conjunto com a falta de padrões universais e interoperáveis de gerenciamento de dados entre diferentes repositórios, tornam a privacidade do usuário médio cada vez menos possível.

#### 2.1.1.3 Vazamentos e Fraudes de Identidade

Uma das principais preocupações atuais quando se trata da Internet, é o constante risco de fraudes de identidade. Os vazamentos de dados anteriormente mencionados, tornam ataques de fraudes virtuais com contas *online* se passando por outros usuário bastante possíveis. Em 2018, mais de 2,1 bilhões de registos do Facebook foram potencialmente comprometidos e 336 milhões de credenciais do Twitter foram expostas em texto simples. Dados governamentais também podem ser fraudados, como em 2017, onde mais de 16 milhões de consumidores nos Estados Unidos foram afetados por fraude de identidade, que resultou em danos de 16,8 mil milhões de dólares.

Além disso, os métodos ultrapassados de lidar com documentos de identidade física introduzem problemas próprios de privacidade e segurança. Os documentos físicos podem ser falsificados, alterados, perdidos ou roubados, e a sua apresentação e transferência podem levar a erros humano, como explica (SOLTANI; NGUYEN; AN, 2021). Dentre os casos citados, percebe-se a importância de alternativas para lidar com identidades que sejam interoperáveis, descentralizadas e de fácil gerenciamento.

### 2.1.2 Evolução do Gerenciamento de Identidades

É possível construir uma linha temporal dos diferentes modelos de gerenciamento de identidade e acesso, relacionando como elas lidam com a privacidade do usuário em

relação aos seus dados, embora existam diferentes motivações por trás de cada uma delas, ambas acompanhamento a evolução e adoção da Internet. Para isso, é importante ter o conhecimento do trilema de Zooko, ou triângulo de Zooko, que se refere ao artigo publicado por Zooko Wilcox O' Hearn em 2001, que diz que um sistema de identificação de usuários só pode ter duas dessas três características: segurança, descentralização e significado humano.

Não necessariamente esse trilema diz que é impossível, mas destaca a dificuldade de de se obter ambas as características em conjunto, em um *design* de sistema. Um paralelo pode ser feito entre o trilema de Zooko com as características demandadas por alternativas mais privativas, mencionadas no fim do capítulo anterior. Interoperável representaria segurança, onde para que diferentes sistemas se comuniquem, essa comunicação precisa ser a mais inviolável possível. Descentralizada é a própria descentralização mencionada no trilema, onde não há um ou poucos agentes que controlem essas identidades, embora descentralização possa ser um espectro. Por fim, fácil gerenciamento equivaleria ao significado humano, para que não seja necessário muita tecnicidade para utilizar esses sistemas.

Os sistemas precisam melhorar o nível de segurança, o fluxo de controle de informações, simplificar a autenticação e a afirmação de processos de credenciais, como destaca (KASSEM, 2019). A abordagem descentralizada é um ponto de viragem crucial na gestão de identidades para resolver alguns problemas, tais como *logins* com múltiplas palavras-passe, gestão eficiente de identidades e delegação da camada de autenticação. Essa linha do tempo dos diferentes modelos de gerenciamento de identidade e acesso é descrita na Figura 1, embora não necessariamente tenhas datas bem definidas ou uniformemente aceitas.

Figura 1 – Evolução dos Modelos de Gerenciamento de Identidade


Fonte: (SOLTANI; NGUYEN; AN, 2021)

#### 2.1.2.1 Identidade Isolada

O modelo de identidade isolada é um dos modelos iniciais e mais primitivos em que o provedor de serviços é responsável por todas as operações de gerenciamento de identidade de seus usuários. As principais desvantagens desta abordagem são o grande número de credenciais que devem ser mantidas por cada usuário e os riscos associados ao armazenamento de grandes quantidades de dados de usuários por cada provedor de serviços (SOLTANI; NGUYEN; AN, 2021).

#### 2.1.2.2 Identidade Centralizada

Em comparação com a identidade isolada, a diferença é que o provedor de identidade e os provedores de serviços são dissociados, mas gerenciados pela mesma organização, e cada interação do usuário com os provedores de serviços deve ser autenticada através do provedor de identidade central.

A abordagem centralizada deste modelo não é uma abordagem eficiente e ideal para usuários da Internet ou usuários de organizações significativamente grandes. Exemplos destas desvantagens são o Kerberos e a arquitetura baseada em infraestrutura de chave pública (PKI), como hierarquia de camada única e outras opções de hierarquia que dependem de uma única autoridade central, ou *central authority* (CA) raiz para emitir, manter e revogar dados digitais (SOLTANI; NGUYEN; AN, 2021).

Mais notavelmente esse modelo é utilizado no sistema de nomes de domínio, ou *Domain Name System* (DNS), que identificam todos os endereços de máquinas utilizados nos servidores *Web* da Internet, que é a característica de significado humano mencionado no trilema de Zooko. Uma CA pode fazer uso indevido de certificados, emitir certificados inválidos ou tornar-se vítima de uma violação de segurança que leva a uma ampla ramificação, incluindo emissão fraudulenta de certificados e ataques do homem do meio, ou *man-in-the-middle* (MITM) e de personificação (SOLTANI; NGUYEN; AN, 2021).

#### 2.1.2.3 Identidade Federada

No modelo federado, um conjunto de provedores de serviços e provedores de identidade forma uma federação confiável. Isto permite que o usuário tenha a opção de autenticação através de um dos provedores de identidade federados para acessar qualquer um dos provedores de serviços participantes (SOLTANI; NGUYEN; AN, 2021).

O modelo de identidade federada permite que um usuário use um único conjunto de credenciais para autenticar-se com o provedor de identidade, para acessar facilmente qualquer um dos provedores de serviços federados, mesmos que sejam de diferentes entidades, organizações ou pessoais. Existem exemplos de redes sociais desse modelo, que procuram ser alternativas para redes sociais centralizadas que lucram com os dados dos usuários, como o Mastodon, Odissey e o Matrix. Isto é feito pelo provedor de identidade fornecendo uma prova de autenticação ao provedor de serviços onde esses recurso são conhecidos como *login* único. Protocolos como *Security Assertion Markup Language* (SAML) são aproveitados para desenvolver sistemas de identidade federados.

#### 2.1.2.4 Identidade Centrada no Usuário

Muitas vezes expresso como uma iteração pouco discutida, mas significativa, nos modelos IAM, o modelo de identidade centrado no usuário permite que os usuários tenham

alguma liberdade na seleção de seu provedor de identidade preferido e do atributo de identidade que gostam de compartilhar, juntamente com as condições sob as quais esses atributos podem ser compartilhados (SOLTANI; NGUYEN; AN, 2021).

Neste modelo, os prestadores de serviços e os fornecedores de identidade nem sempre podem ter uma relação de confiança pré-existente. Exemplos desse modelo são o uso do recurso de login do Facebook e do Google para acessar outros serviços online. Protocolos como OAuth e OpenID Connect são as tecnologias dominantes neste modelo. OAuth e OpenID Connect são protocolos abertos, bastante utilizados por toda a infraestrutura da Internet, para permitir autorização segura de maneira simples e padronizada em aplicativos *Web*, *mobile* e *desktop*. Embora a praticidade aumente, a centralização definitivamente aumenta, se concentrando nas plataformas já consolidadas no mercados de dados digitais.

## 2.2 Identidade Auto Soberana

O modelo de identidade auto soberana é o próximo passo seguindo a linha dos modelos de identidade centralizada e federada e o foco desse trabalho. O termo identidade auto soberana originou-se de uma postagem em 2012 por Devon Loffreto para a lista de discussão do Vendor Relationship Management (VRM) com o título "Autoridade de Fonte Soberana". Embora a SSI tenha ganhado reconhecimento nos últimos anos, ela tem suas raízes na privacidade na década de 1970, com a introdução do protocolo de troca de chaves Diffie-Hellman, que permite aos usuários proteger sua privacidade no mundo digital usando criptografia de chave pública (SOLTANI; NGUYEN; AN, 2021).

O conceito de Identidade Auto Soberana, também chamada de *Self-Sovereign Identity* (SSI), propõe uma forma inovadora de gerenciar essas identidades, ao mesmo que tempo que lembra a forma como as pessoas lidavam com suas identidades antes da digitalização e da Internet. O objetivo da SSI é conectar os sistemas de identidade online ao mundo real, e dar aos utilizadores o controle sobre as suas identidades (SHUAIB, 2022).

Por ser um conceito que pode ser implementado de diferentes formas, é comum imaginar a implementação de uma aplicação de SSI como uma carteira digital, assim como as presentes nos dispositivos mobiles Android e Iphone, que armazenam diferentes tipos de identidade, cada uma pra uma situação própria. Tendo como base algumas definições de grupos de trabalho da *World Wide Web Consorcium* (W3C) e outras comunidades relacionadas ao tema de SSI, ela possui dez princípios como é apresentado na Tabela 1.

Embora não exista uma definição única e formal do que é uma SSI, ela pode ser considerada como uma identidade de longa duração onde somente o indivíduo responsável por ela pode ter o total controle, sem que uma autoridade externa possa revogar a sua existência. É necessário o consentimento, no contexto online dos próprios usuários, para que suas informações relacionadas sejam utilizadas por terceiros e quais informações

Tabela 1 – Princípios da Identidade Auto Soberana

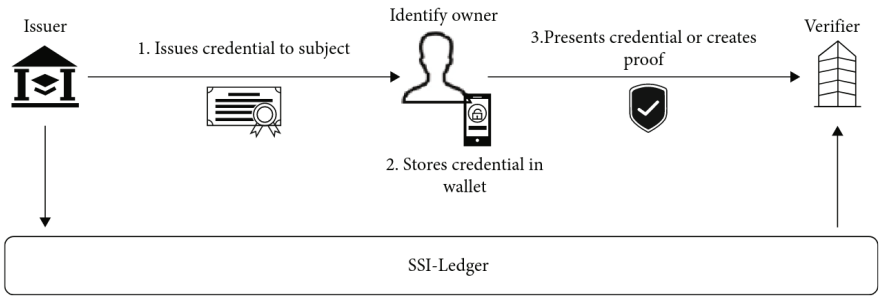
| | |
|---|---|
| Controle | Os usuários devem controlar suas identidades. |
| Acesso | Os usuários devem ter acesso aos seus próprios dados. |
| Transparência | Sistemas e Algoritmos Transparentes. |
| Persistência | As identidades devem ter vida longa. Somente o usuário pode remover suas identidades. |
| Portabilidade | As informações e serviços de identidade devem ser transportáveis. |
| Interoperabilidade | As identidades devem ser usadas tão amplamente quanto possível. |
| Consentimento | Os usuários devem concordar com o uso de sua identidade. |
| Existência | Os usuários devem ter uma existência independente. |
| Minimalidade | A divulgação de reivindicações deve ser minimizada. |
| Proteção | Os direitos dos usuários devem ser protegidos. |

Fonte: (SHUAIB, 2022)

especificamente (SHUAIB, 2022), que sua por sua vez, permite que tenham autonomia dos seus dados.

A natureza descentralizada da SSI permite que também sejam chamadas de Identificadores Descentralizados ou *Decentralized Identitifiers* (DIDs) onde existem três entidades principais, também chamados de agentes: O emissor ou *issuer* que emite alguma informação, que pode ser chamada de credencial ou *Verifiable Credential* (VC), sobre alguém ou algo; o portador dessa credencial, onde o usuário porta suas próprias credenciais que são atreladas ao seu DID; e por fim o verificador, aquele que necessita verificar alguma credencial em algum contexto. A Figura 2 demonstra a relação entre as entidades.

Figura 2 – Relação Entre as Entidades da SSI




Fonte: (SHUAIB, 2022)

Para possibilitar esse sistema, é necessário uma base de registros, também descentralizada, onde o verificador pode confirmar se uma credencial é legitima e emitida pelo emissor mencionado, essa base pode ser um livro razão distribuído, como uma *blockchain*

(SHUAIB, 2022), ou algum protocolo interoperável, como será mostrado na Seção 2.3. Embora haja espaço para discussão das possíveis formas de implementar uma base de dados, até mesmo com partes sob controle centralizado mas que mantém os princípios da SSI, as *blockchains* são comumente mencionadas por serem a implementação mais testada de se criar um livro razão distribuído, além de seus benefícios para sistemas SSI, detalhados de forma adaptada na Tabela 2.

Tabela 2 – Benefícios das *Blockhains* na SSI

| | |
|---|---|
| Integridade de dados | Alta integridade e imutabilidade pois dados são invioláveis devido a *hashes* criptográficos. |
| Propriedade | Os dados estão totalmente contidos na *blockchain*, logo todos podem acessá-la caso ela seja pública. |
| Controle de acesso | Somente o usuário pode executar transações e definir permissões, em seu nome, com sua chave privada |
| Interoperabilidade | Através de protocolos, como as DIDs desenvolvida pela W3C, dados podem ser trocados em diferentes *blockchains* e plataformas |
| Segurança de dados | Aprimorado por métodos criptográficos e tecnologia de contabilidade distribuída, reduzindo pontos únicos de falha. |
| Custo | Embora haja custos de configuração inicial e taxas contínuas em uma rede *blockchain*, no longo prazo haverá economia na administração e maior segurança dos dados. |
| Escalabilidade | Dependendo da arquitetura escolhida; algumas *blockchains* oferecem alta escalabilidade com custos de transação mais baixos. |
| Autonomia do Usuário | Os usuários podem gerenciar e compartilhar seus próprios dados com segurança e eficiência, desde que a aplicação que utilize a *blockchain* também seja segura. |
| Privacidade | *blockchain* e SSI podem suportar tecnologias de preservação de privacidade, como provas de conhecimento zero, permitindo a verificação sem revelar dados pessoais subjacentes. |
| Transparência e Confiança | A natureza imutável e verificável da *blockchain* gera confiança entre as partes interessadas, onde os registros podem ser verificados de forma independente, sem a necessidade de confiar na instituição emissora. |

Fonte: (FAIZAN; RIAZ; SAIF, 2024)

Afim de garantir a interoperabilidade entre DIDs, de diferentes métodos, para que elas não fiquem limitadas em algumas poucas plataformas, notavelmente, um novo identificador uniforme de recursos, ou *Uniform Resource Identifier* (URI), surgiu para compor o conjunto de URIs, assim como o *Uniform Resource Locator* (URL), bastante conhecido, sendo ele o DID. A W3C propôs essa estrutura para credenciais verificáveis em maio de 2017 e lançou o DID 1.0 em dezembro de 2019, logo o surgimento deste padrão

permitiu o uso de (DIDs) para alcançar a SSI através dessa direção de pesquisa (LIN, 2024).

As DIDs são uma sequência de texto simples que consiste em três partes: o identificador do esquema URI DID; o identificador do método DID e; o identificador específico do método DID; esse identificador resolve uma referência de um documento DID armazenado através do método. Esse documento contém informações associadas ao DID, como formas de autenticar criptograficamente um controlado. A Figura 3 exemplifica essa sequência e a Figura 4 exemplifica esse documento. Essa estrutura pode gerar DIDs temporárias e *Peer to Peer* (P2P), sendo útil, por exemplo, para dispositivos de *Internet of Things* (IoT), pois eles estão sendo bastante utilizados diariamente em casas por todo o mundo, e precisam de uma forma de se identificarem sem comprometerem os dados de seus usuários, demonstrando a utilidade dos identificadores.

Figura 3 – Exemplo de Um Identificador Descentralizado (DID)


Fonte: (W3C, 2022)

Figura 4 – Exemplo de Um Documento DID

```
{
  "@context": [
    "https://www.w3.org/ns/did/v1",
    "https://w3id.org/security/suites/ed25519-2020/v1"
  ]
  "id": "did:example:123456789abcdefghi",
  "authentication": [{
    // used to authenticate as did:...fghi
    "id": "did:example:123456789abcdefghi#keys-1",
    "type": "Ed25519VerificationKey2020",
    "controller": "did:example:123456789abcdefghi",
    "publicKeyMultibase": "zH3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV"
  }]
}
```

Fonte: (W3C, 2022)

### 2.2.1 Blockchains

O conceito de SSI é mais facilmente enxergado como um aplicativo de carteira nos dispositivos mobiles que guardam diferentes tipo de identidades, até outras informações, para você se autenticar ou provar uma credencial em alguma ocasião. Como também mencionado, as carteiras nativas dos sistemas Android e Iphone, da Google e Apple

respectivamente, gerenciam as identidades do usuário, garantindo um aspecto de autonomia apesar de ainda terem seus dados controlados por essas plataformas, e não totalmente interoperáveis entre si.

Sendo a interoperabilidade como essencial para a adoção da SSI, e entendendo que *blockchains* são a principal forma de implementar a base de dados descentralizada que garante essa interoperabilidade, (PREUKSCHAT; REED, 2021) menciona no seu livro sobre Identidade Auto Soberana que a tecnologia *blockchain* é a mãe da SSI, pura e simples. Ele também menciona que uma linha de pensamento, no ambiente online, surgiu junto com o início da adoção dessa tecnologia, que insistia nas virtudes *blockchain* como uma arquitetura para dados baseados em consenso, apesar dos seus propósitos originalmente financeiros, que serão mencionados mais pra frente. Isso vai de acordo com (GIMENEZ-AGUILAR, 2021), que explica que as *blockchains* são formadas por nós *online* que cooperam para atualizar dados, baseando-se em uma prova de consenso

Os dois pilares da adoção da tecnologia *blockchain* são o Bitcoin (BTC) e a Ethereum (ETH), e embora ambas as comunidades tenham estado ativas no tema de SSI, Ethereum tem sido um foco particular devido ao poder e flexibilidade da sua tecnologia de criar um computador universal no estado da *blockchain*. Como exemplo, (TRUONG, 2021) utiliza a Ethereum desenvolver um sistema de confiança para serviços descentralizados no qual os participantes (clientes e provedores de serviços) interagem entre si na cadeia de maneira *peer-to-peer*(P2P). Por outro lado, o ecossistema *blockchain* começou com o lançamento da rede Bitcoin em 2009, o que faz dela a *blockchain* mais antiga e que suportou testes de estresse do tempo, resistindo até hoje (PREUKSCHAT; REED, 2021). Ambas *blockchains* possuem origens e visões de mundos diferentes, porém, (PREUKSCHAT; REED, 2021) destaca que existem mais soluções relacionadas a aplicações descentralizadas, ou *Decentralized Applications* (DApps) no ecossistema da Ethereum, principalmente devido a sua alta programabilidade.

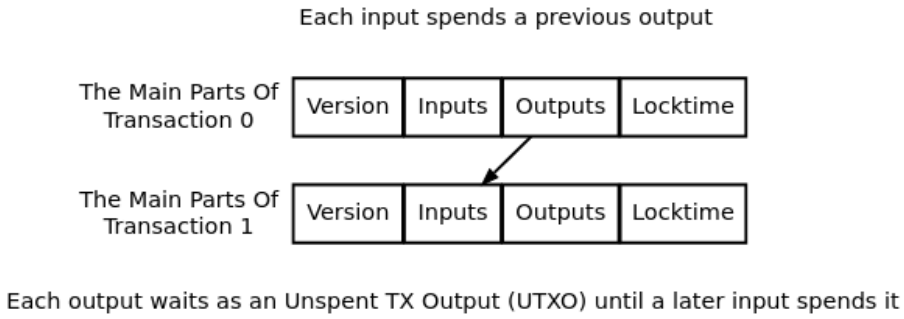
As DApps são um conjunto programas executados de forma descentralizada, como em uma *blockchain*, que constituem uma aplicação para seus usuários. Entretanto, para essa alta configuração e descentralização, existe a desvantagem do custo financeiro em se utilizar essas redes, quando se se deseja alterar algum dado. Esse custo é chamado de *Gas*, no caso da Ethereum, e mais especificamente, refere-se à unidade que mede a quantidade de esforço computacional necessário para executar operações específicas na rede. Como cada transação requer recursos computacionais para ser executada, esses recursos devem ser pagos para garantir que o Ethereum não seja vulnerável a spam e não fique preso em *loops* computacionais infinitos (ETHEREUM-DOCS, 2024).

As vantagens mencionadas em se escolher o modelo *blockchain* como base de dados descentralizada, mencionadas por (FAIZAN; RIAZ; SAIF, 2024), também possuem desafios até conseguirem serem aproveitadas, como complexidade técnica, custo, escalabilidade,

interoperabilidade, adoção e treinamento de usuários. Esses desafios são bastante relacionados ao fato de ser uma novo paradigma de desenvolvimento e utilização que se distância do modelo convencional e centralizado da Internet moderna. Para tornar ainda mais complexo o desafio da implementação da SSI com *blockchains*, essas vantagens anteriormente mencionadas, tem como foco dois modelos de *blockchain*, as de consórcio, onde instituições escolhidas compartilham o controle da rede, e as privadas, com controle mais seleto. Esses modelos acabam trazendo de volta os problemas da centralização e da dependência de terceiros.

A complexidade técnica e o custo operacional ainda se mantém quando se trata de *blockchains* públicas, podendo até serem consideravelmente maiores, apesar de favorecerem a descentralização. É o caso da rede do *Bitcoin*, voltada para fins financeiros como reserva de valor e transações de dinheiro, apesar de ser possível armazenar outras informações nela, esse foco mais direto favorece a simplicidade para usuários e desenvolvedores desse novo protocolo, assim como o custo das transações. Entretanto, tal simplicidade termina dificultando outras aplicações de existirem sem dependerem de redes de terceiros para aumentar a escalabilidade desses dados que não seguem o protocolo. A Figura 5 mostra a pouca quantidade de informação contida em uma transação, onde *inputs* e *outputs* são diretamente transferências anteriores de Bitcoin.

Figura 5 – Partes de Uma Transação de Bitcoin




Fonte: (BITCOIN-DOCS, 2025)

O mesmo problema ocorre com a Ethereum, que ao ter maior customização para aplicações nativas, causando um custo financeiro maior pela complexidade de cada transação, optou por uma atualização que permitiu a integração de redes *blockchains* externas, chamadas de *Layer 2*, onde estas armazenariam informações relevantes para os seus propósitos e a Ethereum serviria como backup e segurança dessas demais redes (ETHEREUM-DOCS, 2024).

### 2.2.2 Implementações

Revisando o conceito de SSI, assim como a associação relevante de *blockchains*, que (PREUKSCHAT; REED, 2021) estabelece com o tema, percebe-se vantagens e desvanta-

gens, levando em consideração a experiência tanto do usuário quanto do desenvolvedor. Algumas dessas limitações podem ter sido evitadas ou reduzidas em outras aplicações e protocolos, que envolvam outras tecnologias de redes descentralizadas, ou ao menos favoreçam privacidade, segurança e autonomia dos dados. Com a popularização dos problemas relacionados ao uso dos dados dos usuários da Internet, vale a pena revisar, também, outras implementações que abordam esse tema da autonomia dos usuários, inclusive, algumas que já existem na Internet/ *Web* moderna, analisando seus potenciais:

- **PKI/DPKI**: A *Public Key Infrastructure* (PKI) ou infraestrutura de chave pública consiste em um conjunto de serviços, ferramentas, processos e tecnologias que facilitam o desempenho de operações criptográficas baseadas em criptografia de chave pública. O modelo de certificado PKI mais comumente usado é conhecido como PKI X.509 ou PKIX (SOLTANI; NGUYEN; AN, 2021). DPKI seria uma PKI descentralizada.

- **W3C DID**: Como explicado em 2.2, DIDs são um meio de verificar assinaturas criptográfica, por meio de chaves públicas assimétricas, validando a autenticidade de um documento, por meio de uma base de registros que conteria essas chaves. O design das DIDs são baseadas nos padrões W3C para garantir interoperabilidade e verificação de identidade entre diferentes sistemas/plataformas (LIN, 2024).

- **Ethereum**: Ethereum é uma *blockchain* com um computador embutido nele. É a base para construir aplicativos e organizações de forma descentralizada, sem permissão e resistente à censura. Existe um único computador canônico, chamado de *Ethereum Virtual Machine* (EVM) cujo estado todos na rede Ethereum concordam e é responsável por executar os seus programas, chamados de contratos inteligentes. As solicitações de computação são chamadas de solicitações de transação e são pagas com a moeda nativa, chamada de Ether (ETH) (ETHEREUM-DOCS, 2024).

- **Hyperledger**[1]: Hyperledger é uma fundação que desenvolve *frameworks* para o desenvolvimento de *blockchains*, principalmente privadas. A Hyperledger Fabric é uma *blockchain* privada popular, enquanto Hyperledger Indy é uma instância de livros razão distribuídos públicos sob permissão, voltado para identidades digitais descentralizados (SOLTANI; NGUYEN; AN, 2021).

- **Matrix**[2]: Matrix é uma rede descentralizada que permite que você converse com amigos, familiares, comunidades e colegas de trabalho usando vários clientes e serviços. Você também pode construir aplicativos de comunicação avançados e ricos em cima do protocolo Matrix usando a especificação de código aberto e *Software Development Kits* (SDKs).

[1] <https://www.hyperledger.org/>
[2] <https://matrix.org/>

- **Solid**[3]: O protocolo *Social Link Data* (Solid) é um rascunho de especificação para gerenciar dados pessoais na *Web*. O Solid foi proposto para descentralizar as redes sociais e tirar os dados das mãos das corporações, ao mesmo tempo em que aumenta a soberania dos dados, ou seja, capacita os proprietários dos dados em relação ao acesso aos seus próprios dados, alavancando os padrões W3C reutilizáveis para a Semântica da *Web*, por meio de servidores *Web* interoperáveis chamados de Pods (ESPOSITO, 2024).

Dentre essas implementações, é possível fazer alguns destaques quanto a interação usuário e desenvolvedor, assim como descentralização e custos operacionais. A PKI já esta sendo implementada na Internet, em algumas regiões do planeta mais que outras, garantindo a validade criptográfica dos certificados de nomes de domínio, aumentando a segurança, entretanto, ainda é necessário garantir a existência de uma base de registros descentralizada para poder chegar a uma DPKI. As W3C DIDs são úteis para garantir interoperabilidade entre múltiplas bases, aumentando a autonomia de escolhas dos usuários e desenvolvedor, porém, o design, assim como as *blockhains* ou outros modelos de bases, ainda não se popularizaram como alternativas para as identidades digitais centralizadas. Ethereum favorece bastante a descentralização do armazenamento, mas impõe complexidade de uso de tecnologias incomuns para o usuário médio, assim como o custo elevado para grandes quantidades de dados por transação. Hyperledger serviria melhor para serviços particulares ou especializados, pois ainda seria necessário entidades possivelmente centralizadas para operar uma nova *blockhain* sem os mesmos problemas das *blockhains* públicas. Matrix demonstra ser uma boa alternativa descentralizada para comunicação em redes sociais, desde que mantenha-se a observabilidade dos operadores da rede, já que é uma rede federada.

O protocolo e projeto Solid procura ser uma nova especificação, que é conveniente tanto para usuários e desenvolvedores. Ele se mantêm próximo da pilha de tecnologias da Internet/ *Web* tradicional, como a de servidores comuns, ao mesmo tempo em que favorece a descentralização, permitindo a autonomia da criação e seleção de Pods pelo próprio usuário. Como será explicado na Seção 2.3, os propósitos de cada Pod podem variar, visto que a especificação suporta diversos tipos de dados a serem operados e compartilhados, servindo para mais que redes sociais. Além disso, os trabalhos de (JUVITO L. DE A.; SOARES, 2023) e (SILVA, 2024) abordam sobre o projeto Solid no contexto educacional, utilizando de Pods.

[3] <https://solidproject.org/>

## 2.3 Solid

Solid é uma especificação que permite que as pessoas armazenem seus dados com segurança em armazenamentos de dados descentralizados chamados Pods. Pods são como servidores *Web* pessoais seguros para seus dados. As entidades controlam o acesso aos dados em seu Pod. As entidades decidem quais dados compartilhar e com quem (sejam indivíduos, organizações, aplicativos, entre outros), e podem revogar o acesso a qualquer momento. Para armazenar e acessar dados em um Pod, os aplicativos habilitados para Solid usam formatos e protocolos de dados padrão, abertos e interoperáveis (SOLID-PROJECT, 2025). O protocolo padroniza interfaces entre aplicativos Solid que usam dados, Pods que armazenam dados e serviços que emitem identidades para usuários e outros agentes que participam do ecossistema Solid. Ao fazer isso, o Solid traz uma camada de confiança, autenticação e autorização para a *Web*, que visam garantir a integridade contextual dos fluxos de informações pelo ecossistema (ESPOSITO, 2024). O projeto foi criado por Tim-Berners Lee, o criador da *Web*, em 2016.

Parte da justificativa para focar em cenários onde o Pod contém os dados pessoais do proprietário é que o Solid foi apresentado como uma solução para aumentar a soberania de dados. A ideia básica é que as empresas que armazenam dados pessoais mantenham uma cópia dos dados relativos ao titular dos dados dentro de um Pod de propriedade do titular dos dados. Dessa forma, os dados mantidos sobre os titulares dos dados se tornam mais transparentes, sendo reunidos em um sistema onde o titular dos dados tem controle de acesso direto. O protocolo Solid fornece meios técnicos para o proprietário conceder ou revogar permissões de acesso para usar os dados no Pod. Dessa forma, o protocolo se torna uma ferramenta para habilitar a soberania de dados ao servir como uma interface entre organizações que desejam ser transparentes sobre os dados pessoais que mantêm e processam (ESPOSITO, 2024).

O ecossistema Solid é um sistema distribuído que fornece serviços para estabelecer relações de confiança necessárias para entregar dados pessoais. Cuidar de questões de confiança e privacidade é ainda mais importante para o Solid, já que, tradicionalmente, a tecnologia de gráfico de conhecimento, como dados vinculados dereferenciáveis e bancos de dados *Resource Description Framework* (RDF), que é utilizada pelo protocolo, foram projetadas com dados abertos em mente, onde os dados são publicados *on-line* sob licenças adequadas. Assim, se reutilizada diretamente a tecnologia existente, algumas das diretrizes e configurações padrão são fundamentalmente opostas aos requisitos de privacidade (ESPOSITO, 2024).

### 2.3.1 Arquitetura

Antes que dados pessoais em um Pod Solid possam ser transmitidos, o primeiro passo é que algum agente (por exemplo, o usuário de dados ou o proprietário do Pod) faça login em um aplicativo Solid. Isso é obtido usando um protocolo de autenticação que verifica se o aplicativo, o agente e algum emissor do agente concordam que o login é válido. Como parte desses protocolos, o agente, o emissor e o aplicativo concordarão mutuamente sobre o conteúdo de um *token* de ID emitido para o aplicativo. O *token* afirma criptograficamente o aplicativo no qual o agente está conectado e o escopo das operações que o aplicativo pode executar em nome do agente. As entidades envolvidas, no caso o aplicativo, agente e emissor, são todas externas ao Pod, e a política do aplicativo é interna a ele mesmo e não especificada no protocolo Solid. O *token* resultante é usado em seguida na fase de autorização ao acessar recursos no Pod Solid.

Vários protocolos podem ser usados nessa fase de autenticação, como o OpenID Connect, que é usado regularmente em outros meios pela Internet. Um recurso atual específico do fluxo do OpenID Connect[4] usado no Solid, é o uso de criptografia de chave pública entre o aplicativo Solid e o emissor, em vez de um segredo compartilhado para estabelecer confiança, como era nas primeiras versões do Solid. As chaves públicas necessárias, são suportadas por outra camada de PKI, onde a proposta dominante atual se chama WebIDs, que são simplesmente URIs de *Hyper Text Transfer Protocol Secure* (HTTPS), ou URLs, que servem como uma identidade forte para uma entidade (usuário de dados, proprietário do Pod, aplicativo, emissor, etc.), que resolve para um documento RDF. Outro protocolo emergente, proposto a ser suportado, são as credenciais verificáveis das DIDs, explicado na Seção 2.2.

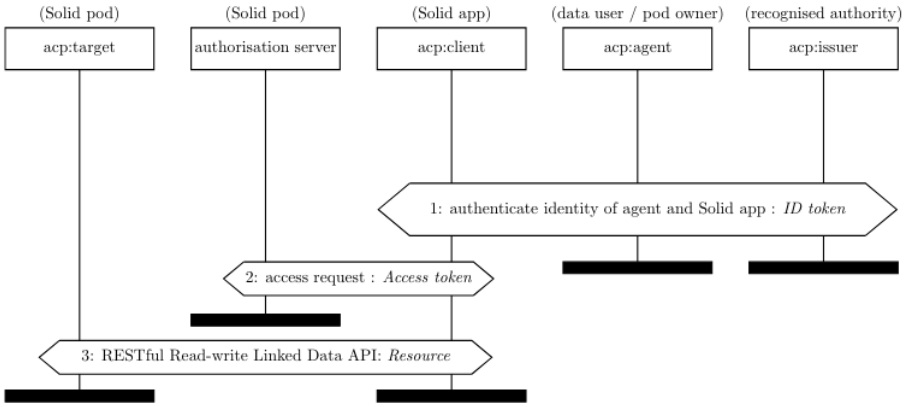
Após o primeiro passo, uma vez que o aplicativo Solid esteja em posse de um *token* de ID, afirmando por meio de assinatura criptográfica o aplicativo, usuário, emissor e escopo delegado ao aplicativo, o aplicativo Solid tenta acessar os Pods relevantes. Nesse segundo passo, o aplicativo Solid contata um endpoint de autorização, que autentica o aplicativo com base no *token* de ID. A decisão sobre conceder, ou não, acesso aos recursos aos quais o aplicativo solicita acesso, é determinada por uma política de controle de acesso. O Solid atualmente suporta dois mecanismos de controle de acesso, separados com base na ontologia *Web Access Control* (WAC), e na ontologia *Access Control Policy* (ACP). Um Pod Solid é equipado com bibliotecas que permitem que o proprietário do Pod, ou um agente a quem ele concede privilégios, defina políticas de controle de acesso. Se o acesso for concedido pelo servidor de autorização ao aplicativo Solid para acessar o Pod, um *token* de acesso de recursos será emitido para o aplicativo Solid.

O terceiro e último passo, após ser autenticado e autorizado com sucesso, o aplicativo Solid usa uma *Aplicattion Programming Interfac* (API) capaz de utilizar *Representational*

[4] <https://openid.net/developers/how-connect-works/>

*State Transfer* (RESTful), para acessar recursos. Essa API RESTful se baseia na Linked Data Platform (LDP) , que maximiza o uso de verbos HTTP (GET, PUT, POST, etc.) e códigos de resposta HTTP para implementar uma interface de dados vinculados de leitura-gravação. O protocolo deve ser executado somente por HTTPS e os cabeçalhos comunicam o *token* de acesso relevante ao recurso. Resumidamente, o Solid fornece um repositório online para armazenar conteúdo de qualquer tipo (documentos, imagens, vídeos, etc.) enquanto fornece meios claros e facilmente gerenciáveis para conceder/revogar permissões de acesso a esse conteúdo. Ele não proíbe que outras interfaces, por exemplo, *endpoints* SPARQL, sejam fornecidas. A Figura 6 demonstra o primeiro, segundo e terceiro passo, ou fases, explicados nessa Seção.

Figura 6 – Fases do Protocolo Solid




Fonte: (ESPOSITO, 2024)

### 2.3.2 Aplicações

Existem outras entidades envolvidas no ecossistema Solid, além dos atores diretamente exigidos pelo protocolo Solid, descritos na Seção 2.3.1. Embora seja possível instalar e manter seu próprio servidor Solid, a maioria dos usuários da Internet poderiam esperar que serviços que utilizem Pods, fossem empregados, também, como provedores de Pods para fornecer a sua infraestrutura de utilidade, essencialmente como serviços de nuvem. Vários provedores de Pods, como Inrupt[5], use.id[6] ou Solidcommunity[7], fornecem infraestrutura para hospedar e gerenciar Pods, permitindo que o proprietário de um Pod de dados escolha seu próprio provedor de Pod. Apesar do usuário controlar e gerir seu Pod, a infraestrutura e acesso a estes seria gerenciado por um terceiro, algo que o usuário deverá estar ciente e saber balancear autonomia com conveniência.

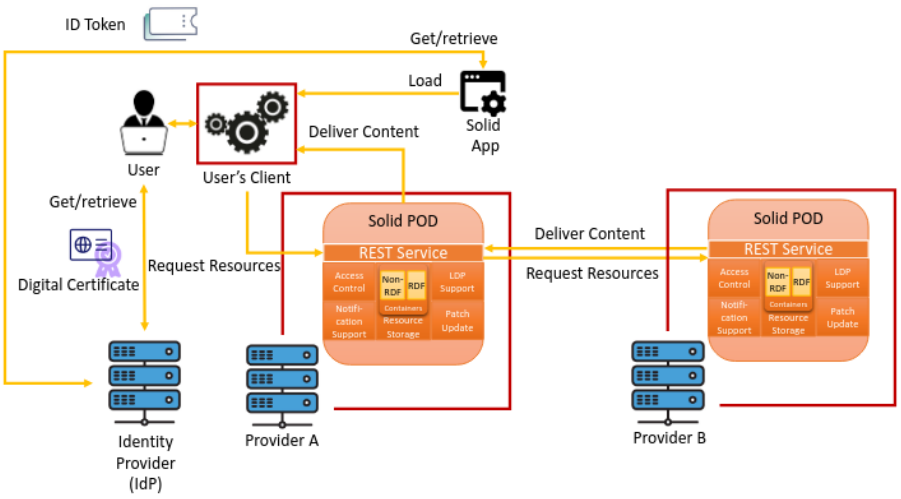
Uma fato interessante é que o criador do projeto Solid também o criador da empresa Inrupt.net responsável por prover Pods, documentação, SDKs e promover o ecossistema.

[5] <inrupt.com>
[6] <get.use.id>
[7] <https://solidcommunity.be/>

Nota-se que existe o reconhecimento de que a conveniência, de já ter o serviço oferecido por um terceiro, também é importante, assim como a independência e controle do usuário dos seus dados, também é fundamental. O fato de que os aplicativos Solid e os Pods são interoperáveis, independentemente do provedor ou da organização que fornece os aplicativos, pode ajudar a evitar o bloqueio de dados, pois aplicativos adequados podem ser usados para migrar dados entre provedores de Pods, que assim é incentivado a ser implementado pelo desenvolvedor da aplicação.

Da mesma forma que os serviços de armazenamento em nuvem, os provedores de Pods são responsáveis pelo registro e manutenção em nível de rede, como aplicação de atualizações em tempo real e garantia de disponibilidade (ESPOSITO, 2024). A Figura 7 ilustra essa dinâmica, onde um provedor, contendo um ou mais Pods, responde a uma requisição por conteúdo, no formato *Resource Description Framework* (RDF)[8], para um usuário através da aplicação cliente, ou outro provedor Pod, utilizando um serviço REST. O usuário por sua vez, é identificado através de um *token* ID validado por um certificado digital, similar ao modelo de VCs explicado na Seção 2.2, fornecido por provedor de identidade.

Figura 7 – Arquitetura das Aplicações Solid




Fonte: (ESPOSITO, 2024)

O protocolo Solid favorece qualidades como privacidade, segurança, descentralização e autonomia do usuário sob seus dados, mesmo quando seus dados estão em um Pod de um terceiro. Pode-se dizer que o principal fator que favorece essas qualidades, é a interoperabilidade entre os Pods, independente de qual provedor ou propósito da aplicação, e também o fato de serem práticas incentivadas pelo ecossistema. Entretanto, para garantir que a SSI seja mais incentivada e adotada, seria importante analisar tecnologias para possibilitar, além de facilitar, o controle inteiro e total desses dados por cada usuário,

[8] <https://www.w3.org/TR/rdf12-concepts/>

amenizando o máximo possível os riscos e inconveniências, das práticas e conhecimentos que seriam necessários. Para isso, é útil analisar como utilizar aplicações Pods, configurando-as para uso pessoal, e para situações que exigem mais disponibilidade, como configurar um servidor *Web* também. A própria página do (SOLID-PROJECT, 2025) recomenda aplicações, externas ao projeto Solid, para diversos propósitos, com código aberto, ou *open source*, que podem ser executadas a partir de qualquer Pod. A Figura 8 apresenta algumas delas.

Figura 8 – Exemplo de Aplicações Solid

| Application | Description |
|---|---|
| Media Kraken | Track your media and never miss a beat.<br>- Documentation. GNU General Public License v3.0 © 2020 Noel De Martin<br>- Source code. GNU General Public License v3.0 © 2020 Noel De Martin<br>- Provide Feedback. |
| Penny | A general Pod Browser by Vincent Tunru |
| Solid IDE | File manager and IDE.<br>- Source code MIT License Copyright © 2018 Jeff Zucker |
| Solid File Manager | A Solid app that help you manages files in your Pod.<br>- Source code MIT License Copyright © 2019 Otto AA |
| Pod Pro | An IDE for editing Solid Pods by Jasmine Leonard. |
| graphMetrix | Allows you to browse your Solid Pod offering multiple views of information including overview, graph, doc, gallery and grid as well as easy to use Solid collaboration control and file management. 30 day free trial followed by a $10 per user/per month. © 2018 graphMetrix |

Fonte: (SOLID-PROJECT, 2025)

Algumas aplicações Solid podem ser customizáveis para um propósito específico, através de SDKs e outras ferramentas de desenvolvimento, principalmente as disponibilizados pela Inrupt[9], o qual disponibilizam servidores Pod para uso pessoal e empresarial, além de bibliotecas e APIs, para as linguagens Java e JavaScript. É possível criar novas ferramentas e APIs para aplicações Solid, desde que mantenha o protocolo compatível com os demais Pods, explicado em 2.3. Alguns casos de uso reais, por organizações, empresas e demais entidades, são mencionados como exemplos por provedores de Pods terceiros.

Notavelmente, a corporação pública de rádio e televisão do Reino Unido, *British Broadcasting Corporation* (BBC) utiliza o protocolo Solid para melhorar a experiência de seus telespectadores[10], assim como outras organizações privadas com seus consumidores. Entretanto, para o desenvolvimento desse trabalho, existe um foco maior no meio acadêmico com soluções abertas para acesso e utilização pública, e para isso, foram selecionados aplicações e casos de estudo, voltados para melhorar o ambiente educacional, mencionados

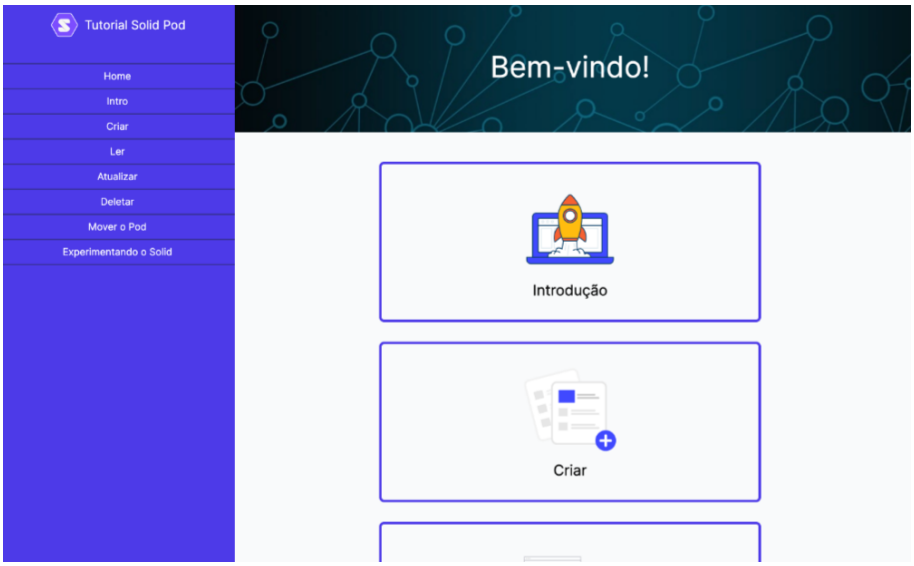
[9] <https://docs.inrupt.com/>
[10] <https://www.inrupt.com/case-study/bbc-improves-viewing-experience-with-solid-pods>

em 2.2.2. No capítulo 3, será demonstrado como executar um servidor Pod pessoal, e usá-lo para armazenar dados pessoais com outras aplicações, e essa demonstração é válida para executar qualquer aplicação Solid.

#### 2.3.2.1 Tutorial *Web*

Convenientemente, foi desenvolvido por (SILVA, 2024), um material no formato *Web* para introduzir o estudante do ensino superior, ao projeto Solid. Com o material, busca-se um aprofundamento técnico relacionado tanto ao funcionamento do projeto, quanto às disciplinas dos currículos de computação, e ainda indicar ferramentas para gerenciar o pod. Foram utilizadas tecnologias voltadas para o desenvolvimento *Web*, como: TypeScript[11], ReactJS[12], Tailwind CSS[13], Next.js[14] e Vercel[15]. Essas mesmas tecnologias foram utilizadas nas aplicações seguintes, com algumas variações mas mesmas utilidades, como o *framework* Vue.js[16]. O tutorial está organizado em partes que englobam uma introdução para situar o projeto e segmentos que abordam a utilização do pod. Esses segmentos abrangem as etapas de operações CRUD com os recursos disponíveis. Além disso, são tratados temas relacionados a mover e experimentação do pod, ainda que o projeto seja relativamente novo e, possivelmente, necessite de atualizações no futuro. A Figura 9 mostra a página inicial do tutorial.

Figura 9 – Tutorial *Web* Para o Solid



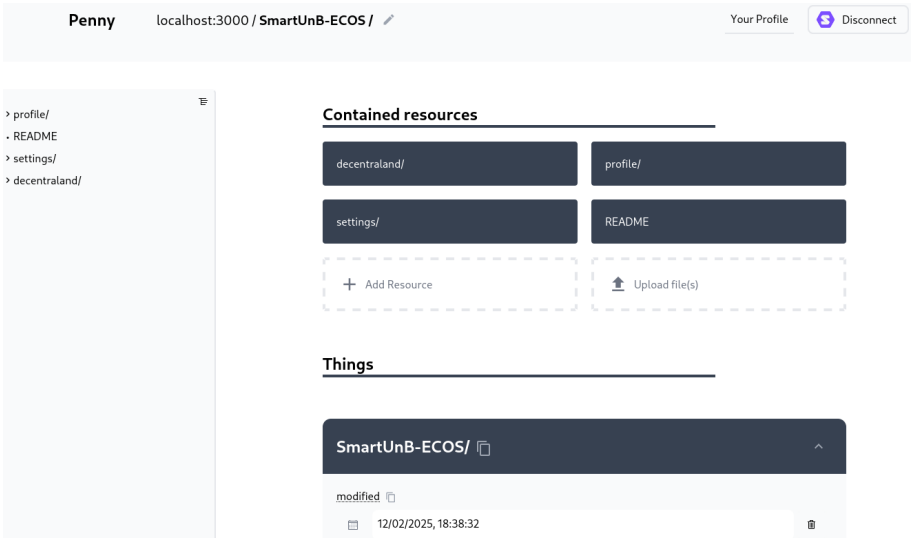

Fonte: (SILVA, 2024)

[11] <https://www.typescriptlang.org/>
[12] <https://react.dev/>
[13] <https://tailwindcss.com/>
[14] <https://nextjs.org/>
[15] <https://vercel.com/>
[16] <https://vuejs.org/>

#### 2.3.2.2 Penny

Tomando como exemplo a aplicação Penny[17], que tem como propósito ser um gerenciador de arquivos e *Integrated Development Enviroment* (IDE), ou ambiente de desenvolvimento integrado, podemos usá-la para editar algum Pod que tenhamos criado. A criação de um Pod em um servidor *Web*, e seus detalhes, será detalhada no capítulo 3. Basta conectar com uma conta de usuário com um WebID, que é exibida como URL, conceder permissão de acesso, e logo em seguida escolher qual Pod, o qual esse WebID possui permissões, para gerenciarmos. Podemos visualizar um Pod como um diretório com subdiretórios e arquivos, e editamos ele ao executar operações de *Create, Read, Update, Delete* (CRUD), como mostra a Figura 10. Essa é uma aplicação útil para gerir Pods de propósito geral.

Figura 10 – Aplicação Solid Penny




Fonte: (SOLID-PROJECT, 2025)

#### 2.3.2.3 RUview

Outras aplicações foram desenvolvidas juntamente, também, com temática voltada para o contexto educacional, ou seja, onde fariam sentido em fazer parte do cotidiano de alunos e professores de escolas e universidades. O RUview (SILVA, 2024) é uma aplicação na qual alunos podem visualizar as refeições do Restaurante Universitário (RU) disponíveis no dia do acesso. Caso autorizem o acesso a seus Pods, estes alunos podem ainda avaliar essas refeições caso gostem ou desgostem delas, e também ver o que seus amigos compartilharam sobre as mesmas refeições. Existe também um sistema simples de edição de amigos em seu perfil Solid, e ainda uma área para administradores, onde é possível criar e modificar as refeições presentes no sistema e atualizar o cardápio com elas. Esta última parte não é

[17] <https://penny.vincenttunru.com/>

descentralizada, com as informações das refeições, cardápios e de autenticação armazenadas no Firebase[18], um Sistema Gerenciador de Banco de Dados (SGBD) hospedado em nuvem. A Figura 11 mostra uma tela, como exemplo, do RUview.

Figura 11 – Tela da Aplicação RUview


Fonte: (SILVA, 2024)

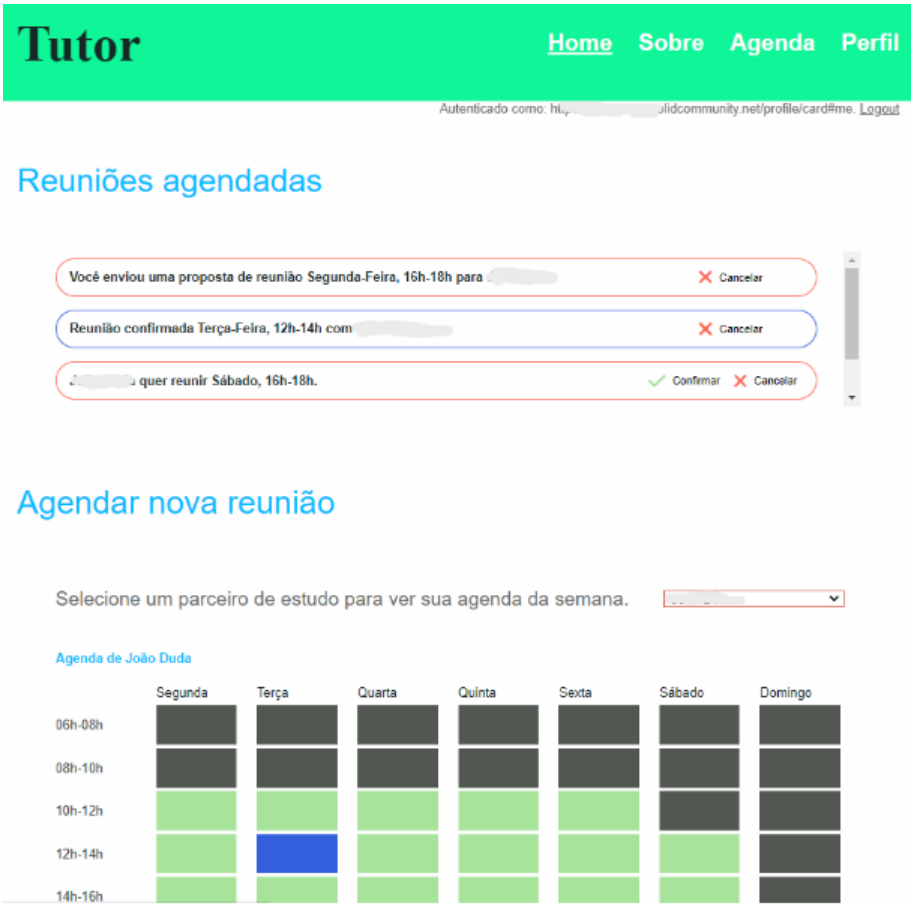
#### 2.3.2.4 Tutor

No contexto do agendamento de reuniões de estudos entre alunos, foi desenvolvido o Tutor (SILVA, 2024), que consiste em um outro aplicativo Solid para facilitar esse costume. Uma vez que o aluno autoriza a aplicação a acessar seu Pod, ele pode criar sua agenda com seus horários livres da semana e também ver as agendas de seus amigos para enviar propostas de reunião para cada um em seus horários livres. Através do acesso ao seu WebID, uma vez autenticado, a aplicação busca os *JavaScript Object Notation* (JSONs)[19] da agenda e dos compromissos do Pod do usuário, e acessa a lista de amigos dele. Para cada amigo nessa lista, a aplicação busca os arquivos JSONs da agenda, dos compromissos e dos pratos curtidos pelo RUview daquele amigo (caso existam) para poder exibir as agendas e compromissos atualizados. A interoperabilidade se mostra presente

[18] <https://firebase.google.com/?hl=pt-br>
[19] <https://json.org/json-pt.html>

no funcionamento do Tutor, que procura por dados criados pelo RUview nos Pods dos amigos e do próprio aluno para sugerir possíveis oportunidades de reunião, com a ideia por trás sendo se ambos gostam daquela refeição, então os dois estarão pela universidade naquele horário, sendo uma boa oportunidade para se reunirem (SILVA, 2024). A Figura 12 mostra uma tela, como exemplo, do Tutor.

Figura 12 – Tela da Aplicação Tutor



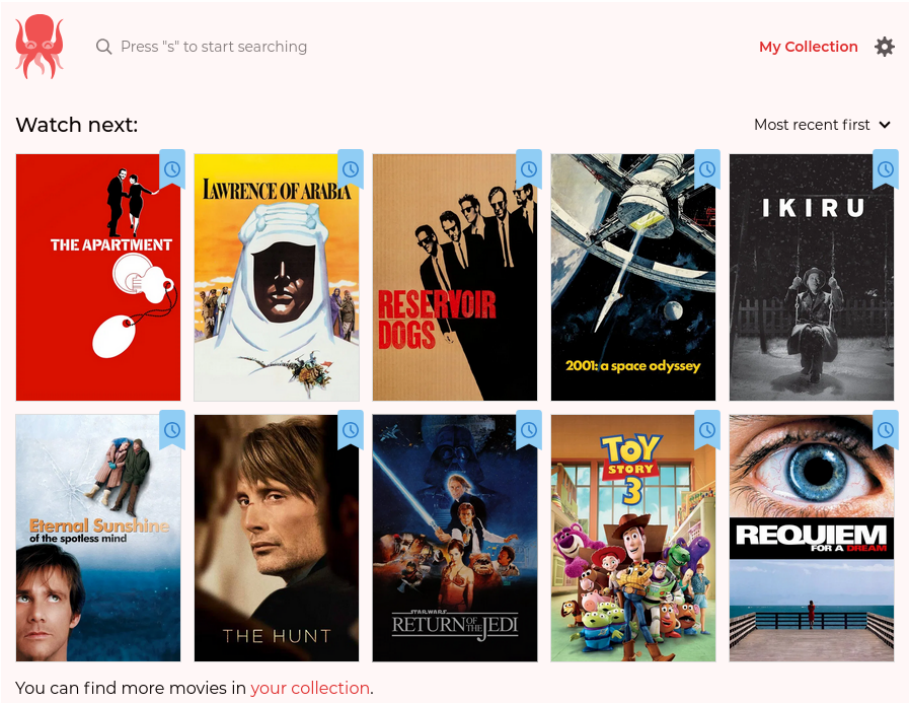

Fonte: (SILVA, 2024)

#### 2.3.2.5 Media Kraken

A aplicação Media Kraken[20], serve para organizar mídias, como filmes favoritos, para assistir depois, como mostra a Figura 13. A utilização é bem simples e direta, podendo sugerir filmes recomendados ou importar a própria biblioteca de filmes. Assim como as aplicações anteriores, basta conectar com uma conta de usuário com um URL de um WebID, conceder permissão de acesso, e logo em seguida escolher qual Pod, o qual esse WebID possui permissões, para salvar a mídia.

[20] <https://noeldemartin.github.io/media-kraken/>

Figura 13 – Filmes Organizados no Media Kraken




Fonte: Autoria Própria

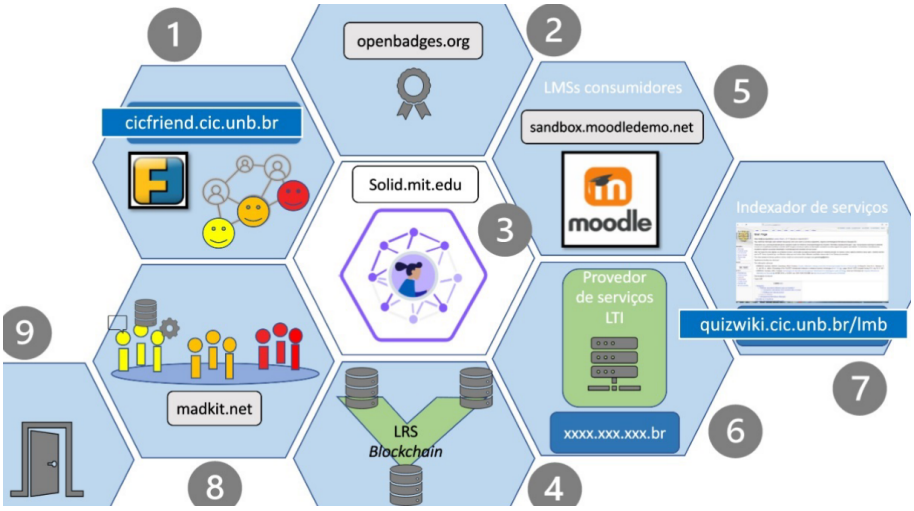
## 2.4 SmartUnB.ECOS

SmartUnB.ECOS é um projeto de ecossistema educacional digital para atendimento à comunidade de campus universitário que busca prover a interoperabilidade de ferramentas de comunicação e de educação a fim de fomentar a socialização e a aprendizagem nesse contexto. No projeto, são contemplados temas de impacto social que têm sido tratados também pela comunidade que trata sobre Interação Humano Computador (IHC), com abordagens enriquecedoras para estudantes de computação, a exemplo das redes sociais descentralizadas e da privacidade de dados. (NÓBREGA; SILVA; SILVA, 2022).

Os elementos que compõem o ecossistema alicerçam uma estrutura sobre a qual se busca fomentar uma dinâmica de conexão entre a aprendizagem formal e a informal. No que diz respeito a escolha das temáticas, ela vai ao encontro do compromisso que se acredita ser necessário para o educador(a) em computação, nas perspectivas técnicas e éticas, de acordo com (NÓBREGA; SILVA; SILVA, 2022). A Figura 14 representa os elementos que compõe o ecossistema do SmartUnB.ECOS. O item 3 representa um servidor Pod Solid para abrigar dados dos usuários, aplicando a descentralização para resguardar privacidade.

O protocolo Solid é mencionado por (JUVITO L. DE A.; SOARES, 2023) no contexto do SmartUnB.ECOS, onde são apresentados trabalhos que estudam a adoção da tecnologia Solid pela sociedade, e como aplicações Solid podem agregar valor nesse

Figura 14 – Ecossistema SmartUnB.ECOS




Fonte: (NÓBREGA; SILVA; SILVA, 2022)

ambiente educacional. Do ponto de vista pedagógico, esse ambiente pode ser categorizado como um ambiente que une aspectos do ensino remoto digital e ensino presencial. Ao se utilizar diferentes ferramentas digitais de comunicação, que compartilham dados pessoais entre si, por necessidade do usuário, encontra-se um espaço favorável para incentivar a adoção de tecnologias *Web* descentralizadas, demonstrando na prática a importância da privacidade, segurança, e soberania dos usuários com seus dados pessoais.

# 3 Proposta

Para a utilização das aplicações Solid, é necessário que haja um provedor de Pod, podendo esse ser um terceiro ou o próprio usuário hospedando um servidor onde armazenará esses Pods. Embora a última opção exija mais conhecimento técnico, o que limita a base de usuários, ela é a que mais favorece a SSI, visto que o usuário tem controle total dos seus dados. Vale ressaltar que ainda existem vantagens em utilizar um Pod, mesmo que a partir de um provedor externo, pelo fato do protocolo ser interoperável, favorecendo a liberdade de transferência dos dados, e o estabelecimento de confiança entre usuário e a empresa ou organização.

Aprimorar o processo de hospedagem de um servidor Pod, pode facilitar a adoção de novos usuários, tanto de usuários da nuvem tradicional como desenvolvedores. Por meio da experimentação se soluções similares, a partir do servidor Community Solid Server (CSS)[1] foi realizado um *soft-fork* para elaboração de um servidor próprio, chamado SSI Solid Server (SSS), voltado para as necessidades próprias do contexto educacional da UnB, em conjunto com as implementações internas do projeto SmartUnB.ECOS. O principal problema que o CSS aborda, de acordo com (HERWEGEN J.; VERBORGH, 2024), é a necessidade de uma plataforma altamente flexível e fácil de usar, especialmente para novos usuários e pesquisadores. O SSS servidor destaca a importância da SSI, oferece mais acesso aos estudantes com documentação e testes detalhados, desenvolve uma interface mais assimilável, oferecendo outras opções de funcionalidades nas configurações internas, além de oferecer um repositório[2] centralizado para facilitar o desenvolvimento coletivo na universidade, com código aberto.

Como é mencionado por (DEDECKER, 2022), desenvolvedores de aplicativos Solid têm dificuldade em tomar decisões sustentáveis sobre como estruturar dados para reutilização, uma vez que suas escolhas individuais impactam a interoperabilidade de todo o ecossistema, principalmente através de APIs. Tais dificuldades também podem ser notadas na arquitetura de servidores Solid, dependendo da pilha de tecnologias utilizadas, principalmente ao se utilizar JavaScript para o desenvolvimento, o que torna importante o estudo dessas implementações para compreender seus detalhes técnicos, servindo como referência para o desenvolvimento de aplicações Solid em trabalhos futuros.

A Relação entre as Entidades da SSI incentiva a implementação de aplicações móveis para a utilização de credenciais verificáveis em situações cotidianas. Fazer uma assimilação entre o protocolo Solid com SSI significa, também, explorar a possibilidade do desenvolvimento de aplicações voltadas para o meio *mobile*. Embora esses servidores

[1] <https://communitysolidserver.github.io/CommunitySolidServer/>
[2] <https://github.com/8ifq3/SSI-Solid-Server/>

funcionem como *softwares* voltados para estações de trabalho e *desktops*, eles podem servir como prova de conceito e de funcionalidades, para serem futuramente compatibilizadas com *smartphones*, como novas aplicações.

A proposta segue um modelo tradicional de desenvolvimento *Web*, que se utiliza de um servidor *Web*, para executar operações CRUD em um Pod, por meio de aplicações *Web* que irão pedir por permissão para acessarem os dados dos Pods locais. O processo de instalação, configuração e execução do servidor será descrito nas etapas a seguir, podendo ser consultado com mais detalhe no repositório do projeto. Para isso é necessário ter o Node.js[3] instalado previamente, na versão 18 ou mais. Logo em seguida:

- Clonar o repositório.
- Executar o comando **npm i** na pasta do repositório, para instalar as dependências.
- Executar o comando **npx . -c @css:config/file.json -f data/**, para subir o servidor e persistir o conteúdo alterado dentro do Pod.
- Acessar o endereço local do servidor, através de um navegador, **localhost:3000**. Um nome de domínio público pode ser associado, para permitir o acesso do servidor remotamente, com as devidas configurações de portas e processos. A Figura 15 mostra a página inicial que se deve encontrar, ao acessar o endereço.
- Clicar em **Sign up for a account** para criar uma conta de usuário nesse Pod. A Figura 16 mostra o exemplo de uma conta sendo criada, para o *e-mail* já existente **180149598@aluno.unb.br**. A vantagem de utilizar um *e-mail* real, é que o SSS pode enviar um *e-mail* para recuperação de senha, caso ela seja perdida.
- Após a conta ter sido criada, é possível criar um Pod para essa conta, na própria página depois do cadastro ou *login*. A Figura 17 mostra um Pod chamado **SmartUnB.ECOS** sendo criado, juntamente com a opção de usar o Pod, no caso o link URL do Pod, como um WebID para aplicações Solid.
- Como resultado, a Figura 18 mostra a tela principal após as operações anteriores.

Para exemplificar e testar uma aplicação Solid, utilizando o SSS para armazenar dados em um Pod, será utilizado a aplicação Media Kraken, para popular o Pod com metadados de filmes e mídias favoritas, e servir de recordação. As etapas para utilizar a aplicação são:

- Acessar a aplicação e conectar o endereço do Pod, no caso **SmartUnB-ECOS**, como mostra a Figura 19.

[3] <https://nodejs.org/pt>

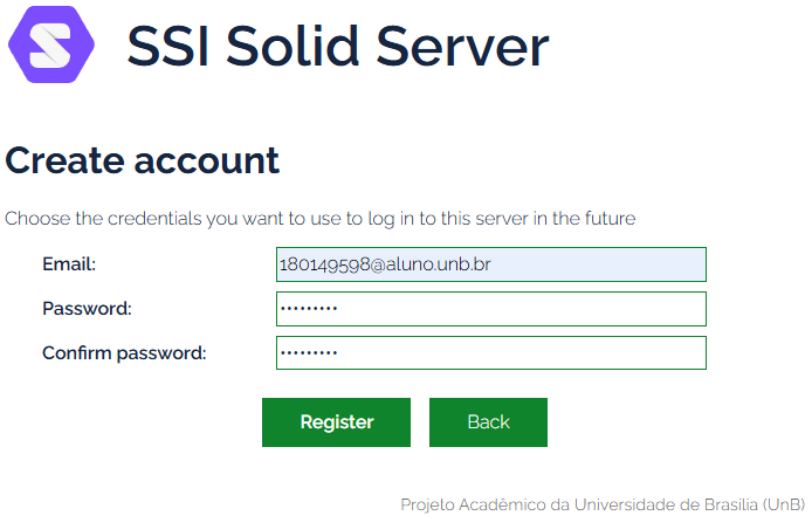
Figura 15 – Página Inicial do SSS


Fonte: Autoria Própria

- Conceder permissão de acesso da aplicação para o Pod pessoal, similar a Figura 20.
- É possível importar sua própria coleção de filmes, ou aceitar uma sugestão do Media Kraken, de acordo com a Figura 21. Para fins de exemplificação, a opção de recomendação será escolhida.
- Por fim, os filmes aparecem de forma organizada, sendo útil para o usuários acessar futuramente, como mostra a Figura 22.

Uma das funcionalidades explorada no CSS e aproveitada no SSS, é o *framework* Components.js, desenvolvido por (TAELMAN, 2023), um *framework* semântico para aplicações TypeScript e JavaScript que fornece semântica global para configurações de *software*. Components.js permite a construção de aplicações altamente modulares, conectadas dinamicamente com base em arquivos de configuração semântica, como utilizado no comando

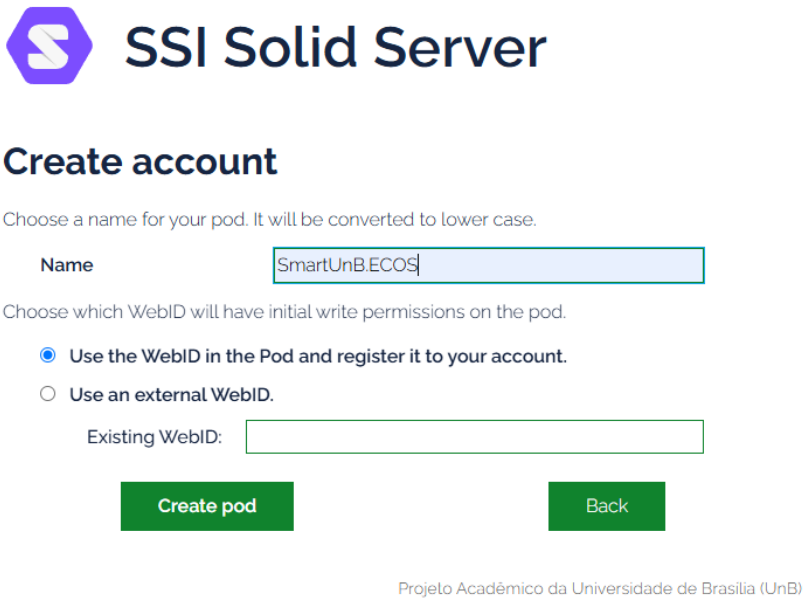
Figura 16 – Criando Uma Conta de Usuário no SSS




Fonte: Autoria Própria

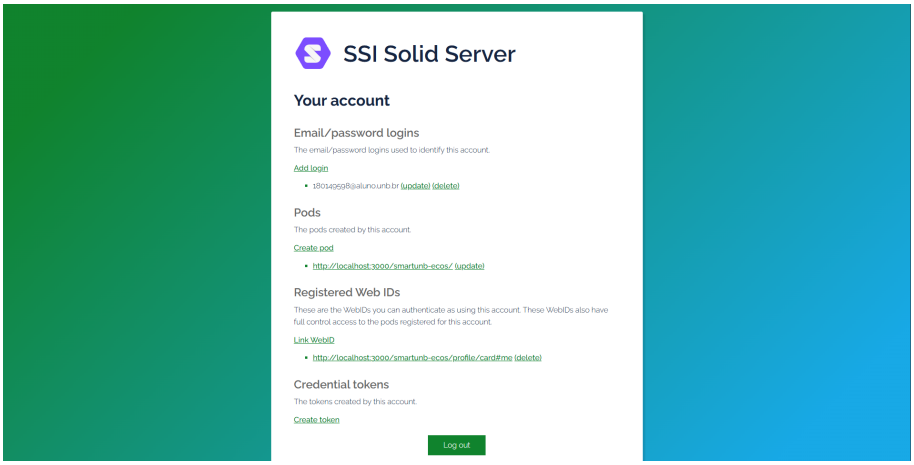
**npx . -c @css:config/file.json -f data/**, onde o arquivo de configurações permite tornar persistente o conteúdo dos Pods, assim como outras permissões. No repositório do SSS foi elaborado configurações para permitir o maior nível de permissões e funcionalidades disponíveis, como é explicado mais detalhadamente no próprio repositório. Essa funcionalidade, assim como a utilização e detalhes internos para modificações no código, são esforços importantes para se obter uma melhor compreensão do desenvolvimento com o Solid.

Figura 17 – Criando Um Pod e WebID no SSS



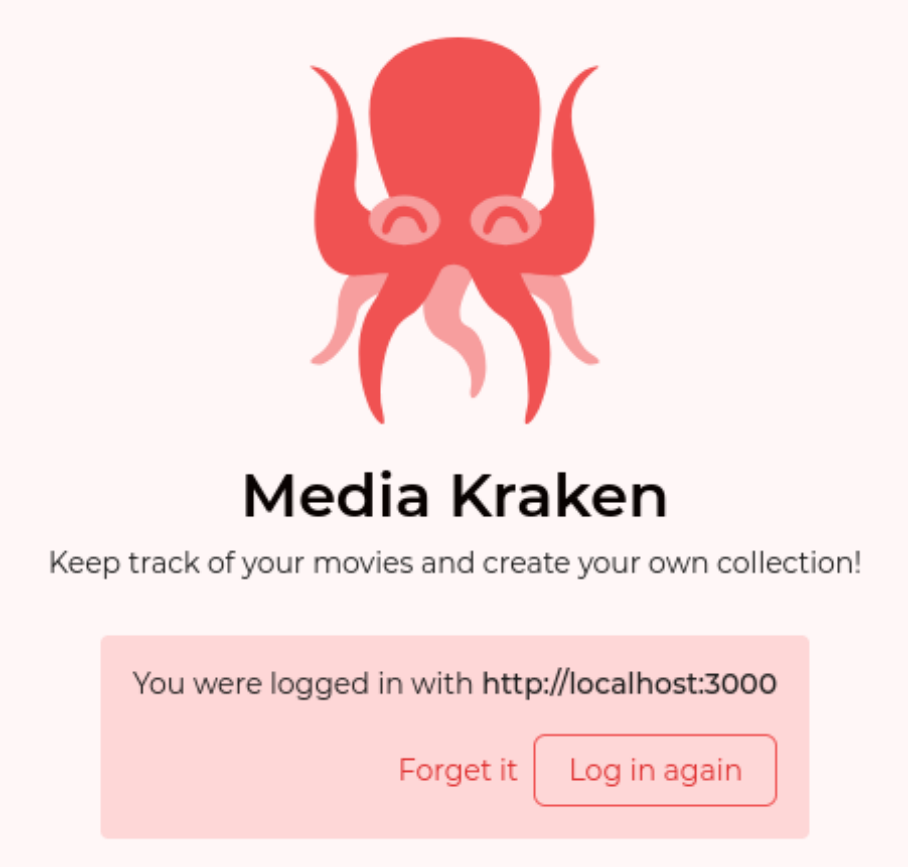

Fonte: Autoria Própria

Figura 18 – Tela Principal Após Operações no SSS



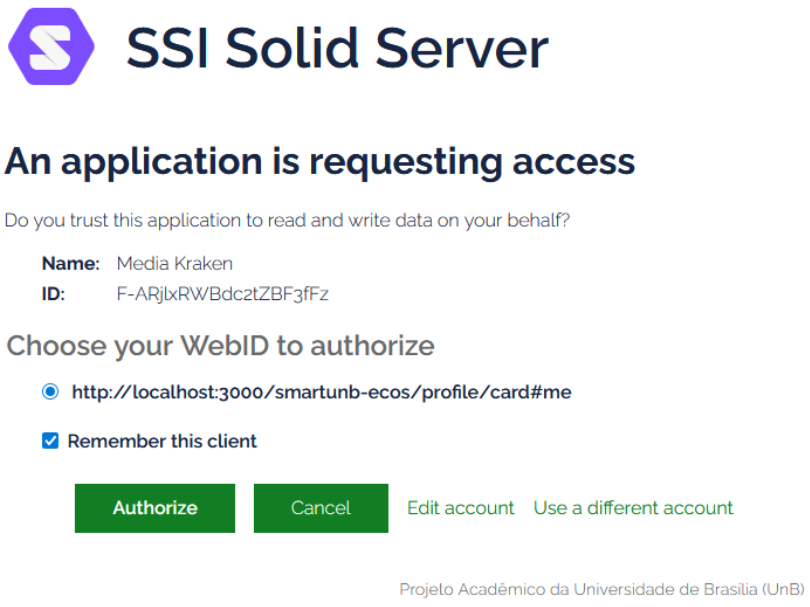

Fonte: Autoria Própria

Figura 19 – Conectar Pod no Media Kraken



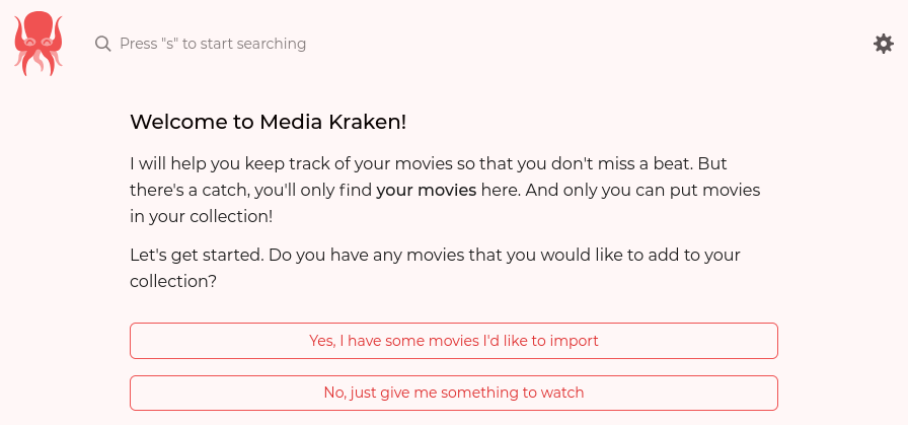

Fonte: Autoria Própria

Figura 20 – Permissão de Acesso ao Pod Pelo Media Kraken



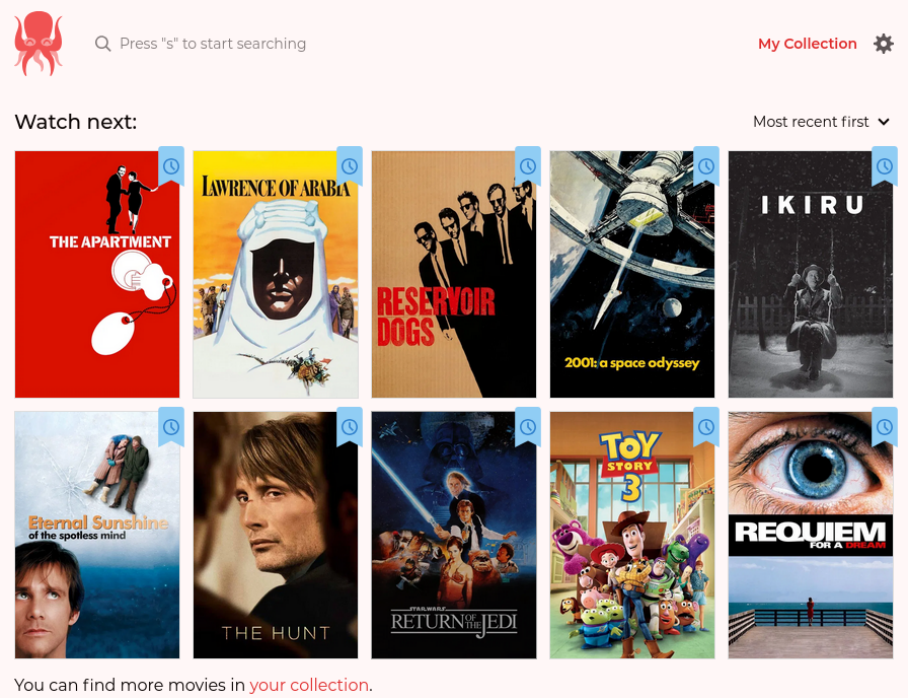

Fonte: Autoria Própria

Figura 21 – Organizar Filmes no Media Kraken




Fonte: Autoria Própria

Figura 22 – Filmes Organizados no Media Kraken




Fonte: Autoria Própria

# 4 Resultados

A proposta de experimentação das definições e aplicações do ecossistema Solid, com o contexto educacional 3, por meio do projeto SmartUnB.ECOS, seguiu um sistema de teste e execução de tarefas, que buscou provar a possibilidade de alcançar ambos os objetivos em conjunto, o da Descentralização da *Web* e o de promover a educação para o ambiente digital. Em conjunto com tema do trabalho, também foi posto em prática o amadurecimento e implementação da SSI, para alcançar novos setores digitais do uso cotidiano. Pensando em trabalhos futuros, esse projeto proposto foi planejado como adaptável, imaginando a experiência do usuário e desenvolvedor, onde os resultados esperados e obtidos, assim como suas partes, foram destacados a seguir:

- Entender o que é a SSI, e onde ela pode estar sendo implementada.
- Analisar como o protocolo Solid pode ser favorável para adoção da SSI.
- Analisar como o adequar o protocolo Solid para o projeto SmartUnB.ECOS.
- Instalação, configuração e execução de um servidor Solid para obter um Pod pessoal, hospedado na própria máquina.
- Analisar e testar o estado atual das aplicações integradas com Pods Solid.
- Possibilitar a descentralização da identidade do usuário, principalmente no contexto do SmartUnB.ECOS
- Relatar a experiência de pesquisa e desenvolvimento, motivando futuros trabalhos voltados para SSI.
- Demonstrar a portabilidade dentre diferentes provedores, serviços e aplicativos Solid, dada a natureza interoperável do protocolo.

Vale destacar alguns trabalhos futuros como a utilização do SSS por estudantes, desenvolvedores, e usuários comuns da Internet, para a utilização de aplicações Solid, como as mencionadas na Seção 2.3.2. Para isso seria importante o aprimoramento das interfaces e toda a experiência do usuário. Possíveis novas funcionalidades seriam bem vindas, como proteção de ataques de negação de serviços distribuídos, *Distributed Denial of Service* (DDoS), além de correções de problemas técnicos. Essas melhorias seriam voltadas para a qualidade do desenvolvimento *back-end*, em outras palavras, a infraestrutura de um contexto maior, como o SmartUnB.ECOS, pelos responsável por manter o sistema operacional. Para isso, a melhoria da documentação, como a do repositório, sempre será de

importância. Outra contribuição interessante seria a de estabelecer um provedor de Pod, similar aos serviços de nuvem existentes atualmente, como mencionado na Seção 2.3.2, visto que o conteúdo entre diferentes Pods são transferíveis entre si, graças ao protocolo. Provavelmente, a maior contribuição, será a de utilizar o servidor em paralelo com o desenvolvimento de aplicações *Web* para o protocolo Solid, esperando possibilitar caminhos para alcançar ou adaptar soluções, também, para o ambiente *mobile* dos *smartphones*, dado que como foi explicado na Seção 3, a SSI pode ser mais assimilável com o meio móvel.

Como referência desses trabalhos idealizados, alguns podem ser mencionados diretamente, como os exemplos reais de como o projeto Solid tem sido explorado em âmbito governamental por (SILVA, 2024); a análise do protocolo Solid em relação a obrigações de privacidade, como no caso da Lei Geral de Proteção dos Dados pelo Regulamento Geral de Proteção de Dados (RGPD) Europeu, feita por (ESPOSITO, 2024); o estudo do contexto educacional do *Metaverso*, junto com o Solid por (JUVITO L. DE A.; SOARES, 2023); o desenvolvimento e os problemas solucionados pelo CSS, por (HERWEGEN J.; VERBORGH, 2024), o qual motivou o desenvolvimento da solução proposta, o SSS; e o desenvolvimento de outras soluções *Web*, também aproveitadas nesse projeto, como o Components.js por (TAELMAN, 2023); o contexto educacional do SmartUnB.ECOS e soluções sendo utilizadas no âmbito universitário, por (NÓBREGA; SILVA; SILVA, 2022).

# 5 Conclusão

A centralização da Internet moderna é um risco, que tem como consequência a violação da privacidade e segurança de seus usuários. Muitos dados pessoais sensíveis, mesmo que não pareçam, acabam sendo compartilhando com terceiros indesejados, intencionalmente ou não. O termo SSI levanta a ideia de que essa realidade pode ser diferente, onde o usuário pode ter controle de seus dados, principalmente no que refere-se à sua identificação, onde ele assume o controle e se protege do risco de ser bloqueado de acessar suas próprias informações. Ele escolhe também com quem compartilhar e o que compartilhar. Entretanto, há desafios de aceitação, amadurecimento e concretização para a SSI, visto que a Internet necessitaria de uma transformação conveniente, para usuários e desenvolvedores perceberem a importância de cuidarem de seus dados.

O projeto SmartUnB.ECOS apresentou uma boa oportunidade de implementação para SSI, no contexto educacional universitário, ao conceder a oportunidade de implementar uma forma de identificação e armazenamento dos dados, que garantisse descentralização e autonomia dos usuários. Compreender a necessidade do usuário e a dificuldade de adaptação a novos conceitos, se provou desafiador devido a dificuldade técnicas e de usabilidade abrangência do tema, principalmente se trata de controle dos dados. O projeto Solid, mesmo ainda em estágio de evolução e adoção, se demonstrou muito atrativo por ser conveniente e similar, com as tecnologias atuais da *Web*, e por em prática ideias do seu criador, Tim Bernees-Lee, para concretizar o seu desejo de uma Internet descentralizada, como imaginava durante seu surgimento.

Esse trabalho serviu para testar e sistematizar uma proposta de uso, visando ser adaptável por trabalhos futuros, provando o valor do protocolo Solid para a adoção da SSI, mesmo que um contexto isolado. Tais ideias podem ser motivadoras para outros setores digitais, em que se busquem adotar responsabilidade com os dados de seus usuários, além de incentivar a evolução do protocolo. É esperado, por meio desse trabalho, a conscientização dos estudantes de cursos da área da tecnologia, sobre a importância dos dados pessoais, de como eles estão sendo utilizados contra nossa segurança e privacidade, e que ainda é possível transformar essa realidade perigosa em um futuro inspirador de soluções descentralizadas, onde todos podem utilizar uma Internet mais democrática e justa.

# Referências

BITCOIN-DOCS. 2025. <https://developer.bitcoin.org/devguide/>. Accessado em: 15/03/2025. Citado na página 25.

DEDECKER, R. e. a. What's in a pod? *IDLab, Department of Electronics and Information Systems, Ghent University – imec*, 2022. Citado na página 39.

ESPOSITO, C. e. a. Assessing the solid protocol in relation to security and privacy obligations. *MDPI Information*, 2024. Citado 5 vezes nas páginas 27, 28, 30, 31 e 47.

ETHEREUM-DOCS. 2024. <https://ethereum.org/en/developers/docs/>. Accessado em: 24-07-2024. Citado 3 vezes nas páginas 24, 25 e 26.

FAIZAN, M.; RIAZ, N.; SAIF, U. Transforming university records management: A comprehensive review of blockchain and self-sovereign identity applications. *International Journal of Science and Research Archive*, 2024. Citado 2 vezes nas páginas 22 e 24.

GIMENEZ-AGUILAR, M. e. a. Achieving cybersecurity in blockchain-based systems: A survey. *Future Generation Computer Systems*, 2021. Citado na página 24.

HERWEGEN J.; VERBORGH, R. V. The community solid server: Supporting research & development in an evolving ecosystem. *Semantic Web*, v. 15, n. 6, p. 2597–2611, 2024. Disponível em: <https://journals.sagepub.com/doi/abs/10.3233/SW-243726>. Citado 2 vezes nas páginas 39 e 47.

JUVITO L. DE A.; SOARES, R. T. A metaversidade chega ao campus: possibilidades e desafios do metaverso para educação superior. *Biblioteca Digital da Produção Intelectual Discente da Universidade de Brasília*, 2023. Citado 3 vezes nas páginas 27, 37 e 47.

KASSEM, J. e. a. A. Dns-idm: A blockchain identity management system to secure personal data sharing in a network. *Applied Sciences*, 2019. Citado 3 vezes nas páginas 15, 16 e 18.

LIN, I.-C. e. a. Designing a secure and scalable data sharing mechanism using decentralized identifiers (did). *Computer Modeling in Engineering and Sciences*, 2024. Citado 2 vezes nas páginas 23 e 26.

NÓBREGA, G. M. D.; SILVA, G. T. D.; SILVA, T. V. R. Um projeto estruturante para orientações de tcc em cursos de computação: que oportunidades para ihc? *Simpósio Brasileiro de Computação*, 2022. Citado 3 vezes nas páginas 37, 38 e 47.

PREUKSCHAT, A.; REED, D. *Self-Sovereign Identity*. [S.l.]: Manning Publications, 2021. Citado 3 vezes nas páginas 13, 24 e 25.

SHUAIB, M. e. a. Self-sovereign identity solution for blockchain-based land registry system: A comparison. *Mobile Information Systems*, 2022. Citado 3 vezes nas páginas 20, 21 e 22.

SILVA, O. E. B. M. e. a. Simpósio brasileiro de educação em computação (educomp). In: *Você decide quem POD: empoderando a/o estudante de computação quanto à propriedade de seus dados*. [S.l.: s.n.], 2024. Citado 6 vezes nas páginas 27, 33, 34, 35, 36 e 47.

SOLID-PROJECT. 2025. <https://solidproject.org/about>. Accessado em: 16/03/2025. Citado 3 vezes nas páginas 28, 32 e 34.

SOLTANI, R.; NGUYEN, U. T.; AN, A. A survey of self-sovereign identity ecosystem. *Security and Communication Networks*, 2021. Citado 9 vezes nas páginas 11, 14, 15, 16, 17, 18, 19, 20 e 26.

TAELMAN, R. e. a. Components.js: Semantic dependency injection. *IDLab, Department of Electronics and Information Systems, Ghent University – imec*, 2023. Citado 2 vezes nas páginas 41 e 47.

TRUONG, N. e. a. A blockchain-based trust system for decentralised applications: When trustless needs trust. *Future Generation Computer Systems*, 2021. Citado na página 24.

W3C. 2022. <https://www.w3.org/TR/did-core/>. Acessado em: 28-08-2024. Citado na página 23.

ZUBOFF, S. *The Age of Surveillance Capitalism*. [S.l.]: Profile Books, 2019. Citado 2 vezes nas páginas 11 e 12.