



Personal Data Management for a
Decentralized Web

Drake White, Gopika Kini, Mia Schneider-Martin, Tanay Mahindru

Advisor: Kay Asholu

Master of Information Management and Systems, Capstone Report, 2024

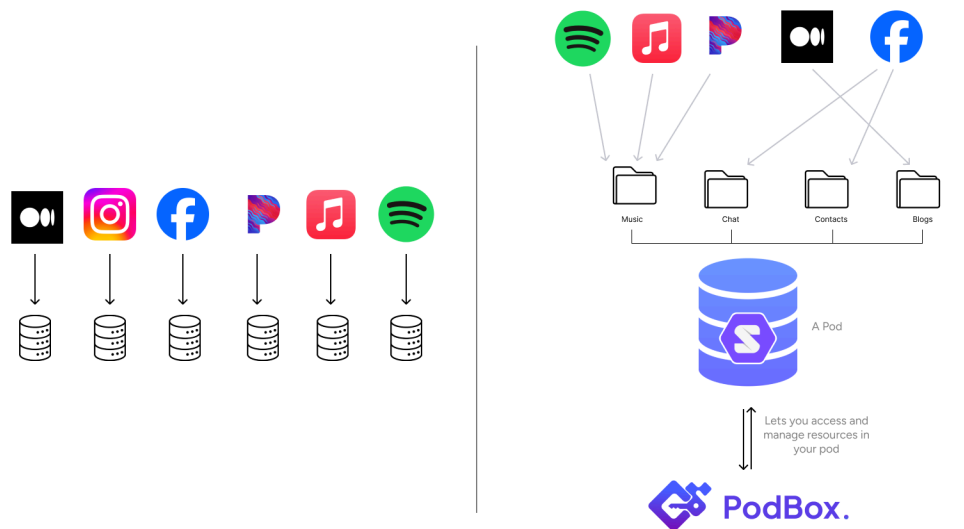
	1
Introduction	2
PodBox	3
Features	3
User Research	4
Objective & Goal	4
Exploratory User Research and Usability Tests	4
Interview Key Takeaways	6
Design Goals	7
User Journey Overview	8
Technical Architecture	11
Learnings	15
Next Steps	15
Project Links	16
Resources	16
Appendix	17
Jobs to be done	17

Introduction

There is an increasing recognition that users should be able to exert greater control over their data – in law as well as broader civil society. However, current technical architectures underpinning platform ecosystems may not be designed to address the requirements of decentralized data ownership.

Tim Berners-Lee, creator of the World Wide Web, reflects on its evolution: "The first decade held promise, but the last 20 years haven't lived up to it." Currently, users grapple with a lack of control over their personal data, often collected without explicit consent and stored in inaccessible digital "silos". Limited choices and dense terms and conditions accompany the free services we rely on, scattering our data across platforms and making it hard to track our footprint. Over time, we become locked into these services, unable to break free. Rethinking the web, Sir Berners Lee initiated the Solid project at the Massachusetts Institute of Technology, with a vision to decentralize the web and put data ownership back in the hands of its creators.

Solid, also known as "social linked data," is a set of protocols and tools aimed at constructing decentralized social platforms adhering to Linked Data principles. To decentralize is to decouple existing applications and data. At its core, Solid enables individuals and communities to securely store their data in containers known as Personal Online Datastores (POD). With PODs, users have full control over the access permissions granted to both individuals and applications seeking to interact with their data.



Centralized Network

A Decentralized Solid Network
(Apps used here are purely for illustrative purposes)

PodBox

Solid, a working group at the W3C, continues to advance development for a broader worldwide adoption, although its uptake for general purposes has been slow. Currently, only a few Solid applications exist, tailored primarily for a technical audience. These currently lack readiness for widespread adoption due to significant usability and experience challenges, particularly for non-technical users.

In response, PodBox, built on top of the Solid framework, offers a one-stop dashboard with a primary goal to simplify and streamline decentralized data management, making it accessible to all. By abstracting the complexities of the underlying protocol, PodBox aims to create a user-friendly interface for managing resources and permissions within one's POD, providing a unified and straightforward experience.



Features

- **POD:** A POD serves as a container for all user data, functioning similarly to their own personal database. Users can store various types of data within their POD.
- **POD Provider:** Typically hosted by a POD provider, a POD can also be self-hosted by users. Different POD providers offer varying services and features.
- **Solid Applications:** These are applications built on the Solid protocol, such as PodBox. They access user data through the POD, enabling functions like reading, writing, and appending data.
- **WebID:** A WebID serves as a unique identifier for a POD, akin to an email address within the decentralized network.

In addition to everything Solid offers (data ownership, interoperability, and secure sharing), PodBox offers-

- **Abstraction of Technicalities:** PodBox simplifies the complexities of decentralized data management by abstracting technical details with a vision to contribute to a wider adoption of the protocol. It offers an intuitive user interface that eliminates the need for users to engage with intricate protocols or frameworks directly.
- **Data Access:** PodBox provides users with a one-stop hub where they can conveniently access all their data on the internet (documents, music playlists, rental agreements, etc) in one place. Users can also view who has access to their data. Users have the flexibility to download, delete, and manage their data according to their preferences.
- **Access Control:** With PodBox, users have granular control over which applications can access their data. They can easily manage access permissions and revoke access as needed

User Research

Objective & Goal

The intent of our research was exploratory and the primary objective of our user research was to understand how users interact with the existing POD management tool and the use cases they use the application for. The findings of our research were used to guide our design and development decisions.

Exploratory User Research and Usability Tests

We conducted a combination of usability tests and interviews to better understand the issues with current POD management tools.

Recruitment

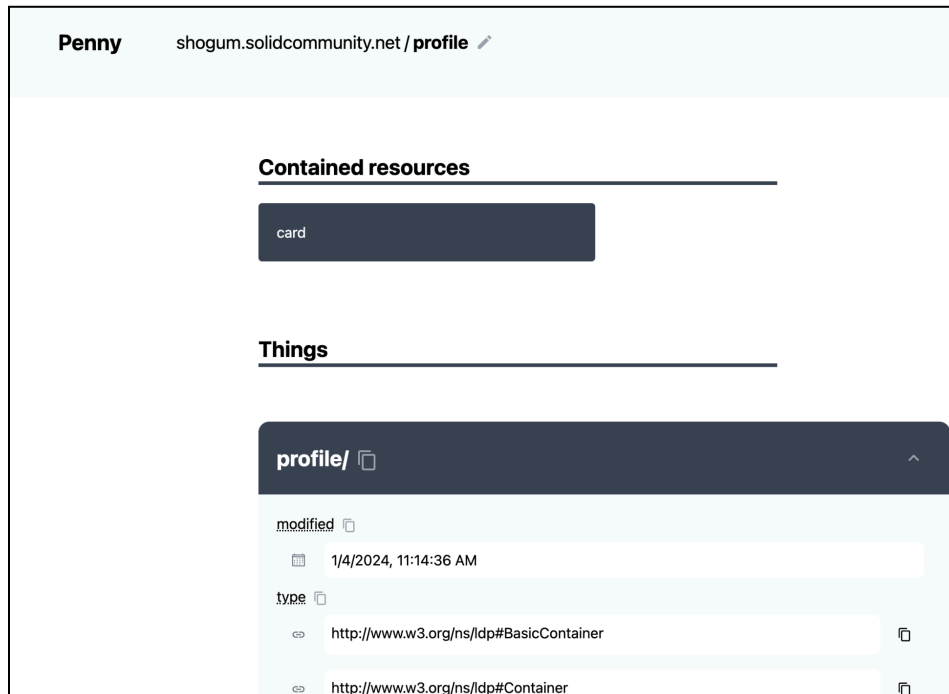
We recruited four participants from the iSchool community, specifically targeting individuals who demonstrated an interest in data ownership and privacy. The selection criteria aimed to gather perspectives from a diverse group of users currently utilizing POD management tools. Participants were compensated for their time.

Consent Process

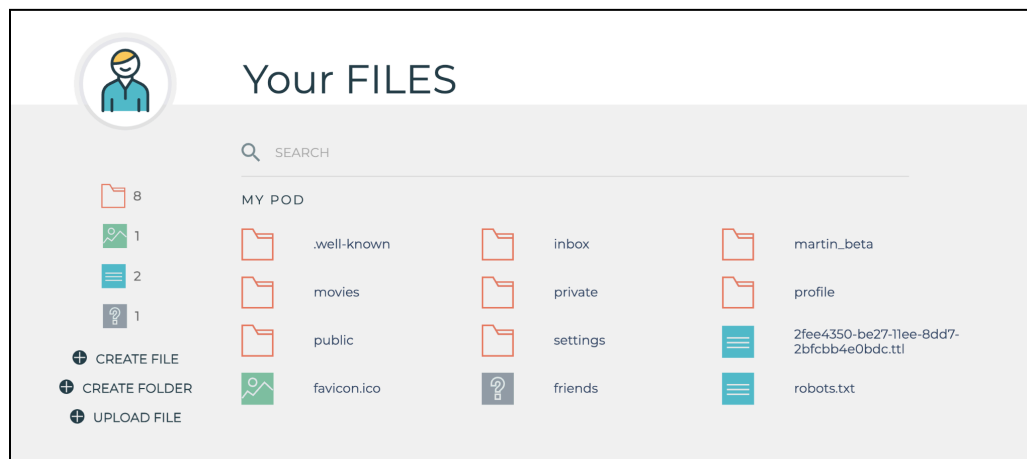
Prior to commencing the study, we provided participants with comprehensive information about the research, and its purpose, and obtained their informed consent. This step ensured adherence to ethical guidelines and transparency regarding the study's objectives.

User Interviews and Usability Tests

We conducted one-on-one interviews and usability tests with the participants, lasting approximately one hour each. During these sessions, participants were presented with a set of tasks related to the onboarding process of existing POD management tools. We encouraged participants to walk us through their thought processes, allowing us to observe their interactions and gather qualitative data.



(Home Page of an existing Pod Browser designed specifically for a technical audience)



(Existing Pod Management Tool lacking satisfactory user experience)

Data Collection

Throughout the interviews and usability tests, we diligently collected qualitative data, focusing on user perceptions, pain points, and potential areas for improvement within the current onboarding process. This data provided valuable insights into the user experience and served as the foundation for identifying opportunities for enhancement.

Data Analysis

Using grounded coding, we listened back to recordings of the interviews and identified key themes emerging from the conversations. We used Dovetail to transcribe the interviews and tag interviews with relevant themes.

Interview Key Takeaways

Here are some takeaways from our interviews, organized by topics covered in our conversation. We used these insights to develop our wireframe and complete feature prioritization.

Understanding the concept of PODs and Solid Ecosystem

- Users struggled to differentiate between the POD itself and the POD management tool, indicating a need for clearer distinctions and explanations
- Users lack understanding of POD providers and which one to choose
- Users express confusion about the concept of WebID and its necessity for interacting with PODs
- Users encounter difficulties in navigating between logging into the POD and the POD management tool, indicating a need for streamlined navigation

Data/File Management

- The presence of index and metadata files within a user's Solid POD led to confusion among users. Users were not certain about the purpose of these files and whether they were essential to the functioning of their POD. Additionally, they were also apprehensive about accidentally deleting these files and causing disruptions to their data storage.
- The naming of folders in Solid is determined by corresponding RDF schemas. For example, all chat applications might be written to a folder named "Chat." However, users find this naming convention less intuitive. Users expressed a need for more intuitive file names within a POD to streamline navigation and file management.
- Currently, PODs lack the functionality to sort and organize files based on any criteria, leading to confusion among users accustomed to conventional file hierarchy and sorting methods.

- Users desire the capability to organize files according to various preferences such as title, file type, or application usage, allowing for greater flexibility
- In a traditional network of applications, which are typically app-centric, users are accustomed to an app-first network structure. As a result, they find it crucial to identify which applications are accessing or modifying specific files for easier location and tracking of data.
- Users appreciate the ability to choose their storage location within Solid, as it provides them with greater flexibility and control over where their data is stored and accessed.

“If it was just right there, like in front of me, if I could just see the message that I've sent you and it's like, oh, this is the message that you send Mia and then, you know, here's the data right in front of you right there when I open you know, like my, I open a thing which says Liquid Chat and I know that, you know, that's where my Liquid Chat data lives.”

Access Control

- Lack of understanding of what Access control means and what each action means (write, append, edit, etc.)
- Concerns over the unintuitive interface and difficulty in locating the needed controls to manage access

Design Goals

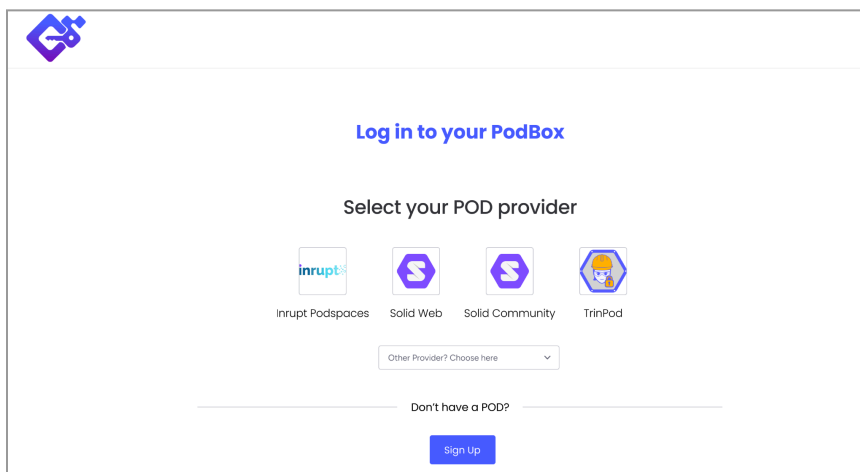
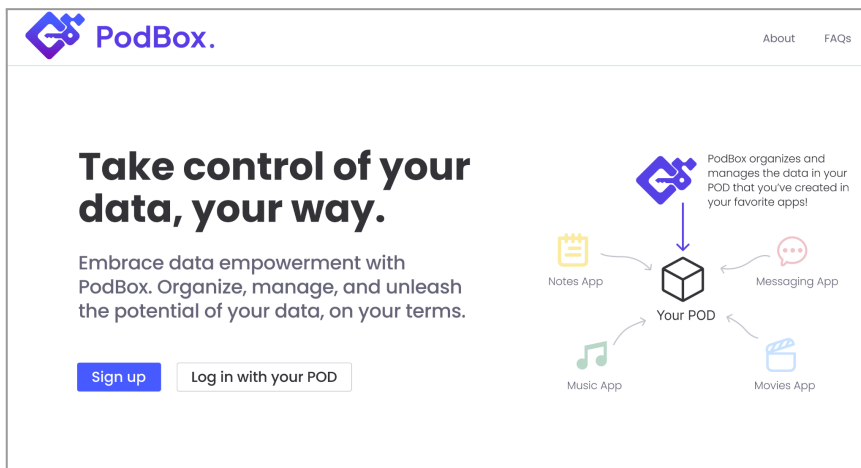
From our research, we identified three key design features:

1. App-first design:
 - a. Users were confused about the file organization of the data and expected to find data within a particular app, instead of within the file structure that the POD management tool used.
 - i. “I didn't know where to look for things or what things are for and what things meant? I was lost and that sums up most of it.”
 - b. User mental framework of data is tied to apps
 - i. “I know that the data from this app lives in this place. And then when I go in there, [I would prefer that] it's just presented to me like upfront rather than just being so deep in there.”
 - ii. “It could also indicate that the things that have access to it are Liquid Chat, some other chat.”
2. Educational onboarding:
 - a. Lack of understanding of key terms, confusion between POD and POD management tool so a unique onboarding experience is needed given that it is a novel technology

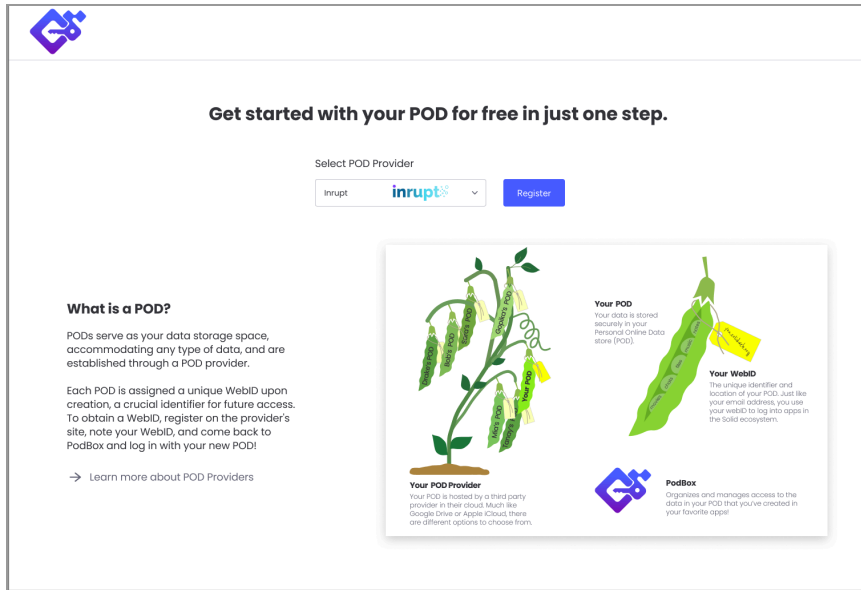
- i. “Hard to understand when you use big words, which makes me feel like a caveman and my caveman brain isn't working.”
- 3. Deeper access control explanations
 - a. Users have a lack of understanding of what the different access control options are and how their data will be shared. We must explain in everyday language what the impact of access controls are so that users understand the results of their selections.
 - i. “I don't know what this means to me right now. I don't know what this part is gonna do, so I feel very lost right now.”

User Journey Overview

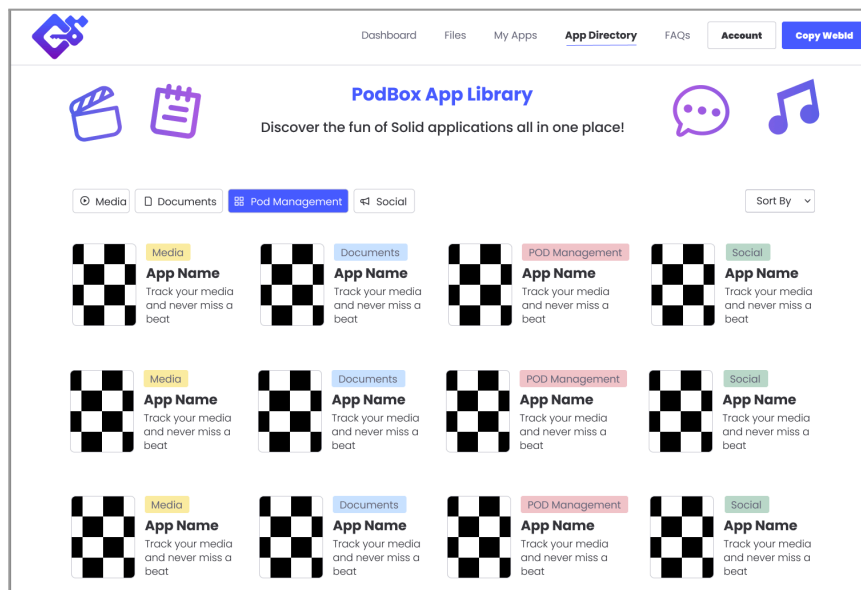
- I. **Streamlined Onboarding:** PodBox offers a user-friendly onboarding process that introduces newcomers to the concepts of Solid and PodBox in simple terms. It explains the benefits of using Solid and guides users through the creation of their own pods step by step.



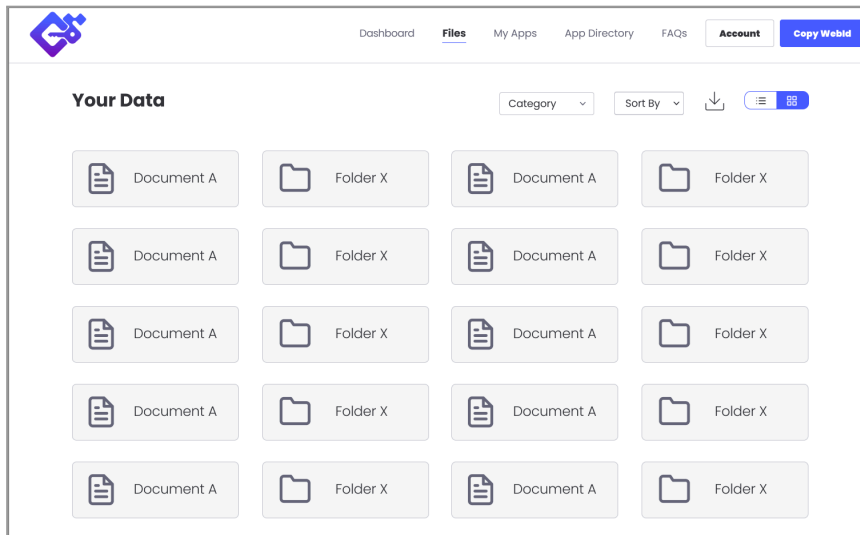
II. **New User Guidance:** For those unfamiliar with Solid, PodBox provides comprehensive guidance on what a pod is and how to create one, ensuring a smooth transition into the Solid ecosystem. We have also created a comprehensive guide with FAQs for all users.



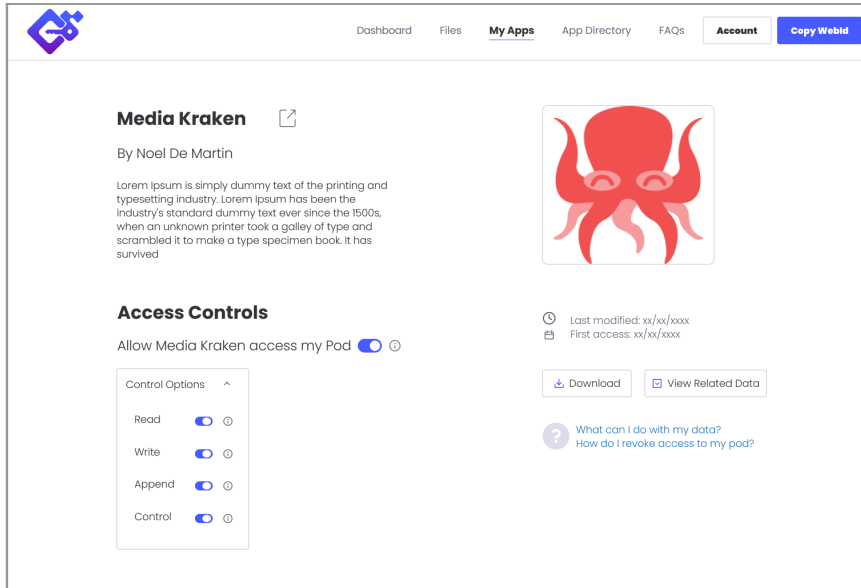
III. **App Library:** Recognizing the need for a single place to discover all Solid applications, PodBox features an App Library where users can discover and access various Solid applications in one convenient location, akin to traditional app stores.



- IV. **Personal Data Management:** PodBox offers users a comprehensive interface to view all the contents stored within their pod. Users also get insights into their pod, including the history of app permissions granted, data storage details, and modification timestamps, enabling users to stay informed and in control of their digital footprint. Users can also upload, download, and manage their data as they would with a personal drive



- V. **Access Control:** With PodBox, users can easily view who has access to their pods and adjust permissions as needed, including the ability to revoke access entirely. We also provide granular control over access levels, allowing users to specify permissions such as read, write, and append, ensuring complete control over how their data is accessed and utilized.




Dashboard Files **My Apps** App Directory FAQs Account [Copy Webid](#)

Media Kraken

By Noel De Martin

Lorem ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived



Access Controls

Allow Media Kraken access my Pod ⓘ

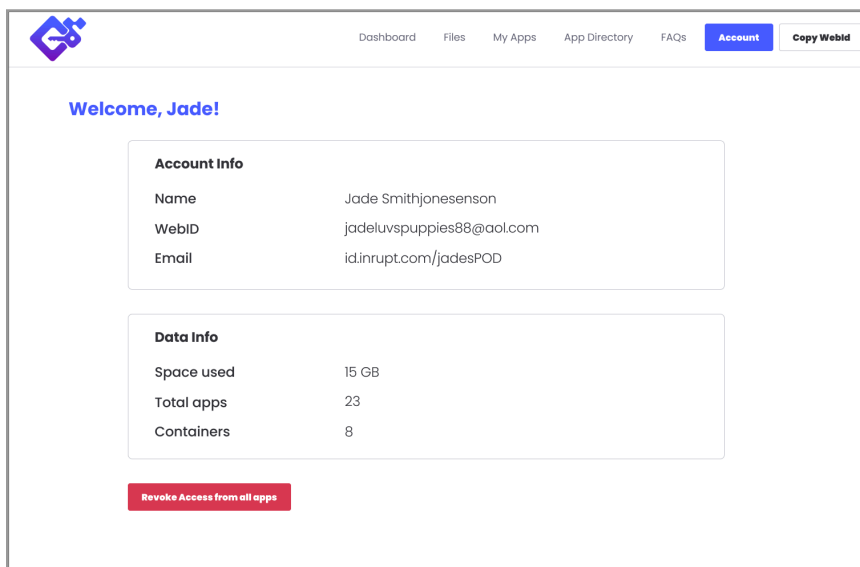
Control Options ^

- Read ⓘ
- Write ⓘ
- Append ⓘ
- Control ⓘ

Last modified: xx/xx/xxxx
First access: xx/xx/xxxx

[Download](#) [View Related Data](#)

? What can I do with my data?
How do I revoke access to my pod?



Dashboard Files My Apps App Directory FAQs **Account** [Copy Webid](#)

Welcome, Jade!

Account Info

Name	Jade Smithjonesenson
WebID	jadeluvspuppies88@aol.com
Email	id.inrupt.com/jadesPOD

Data Info

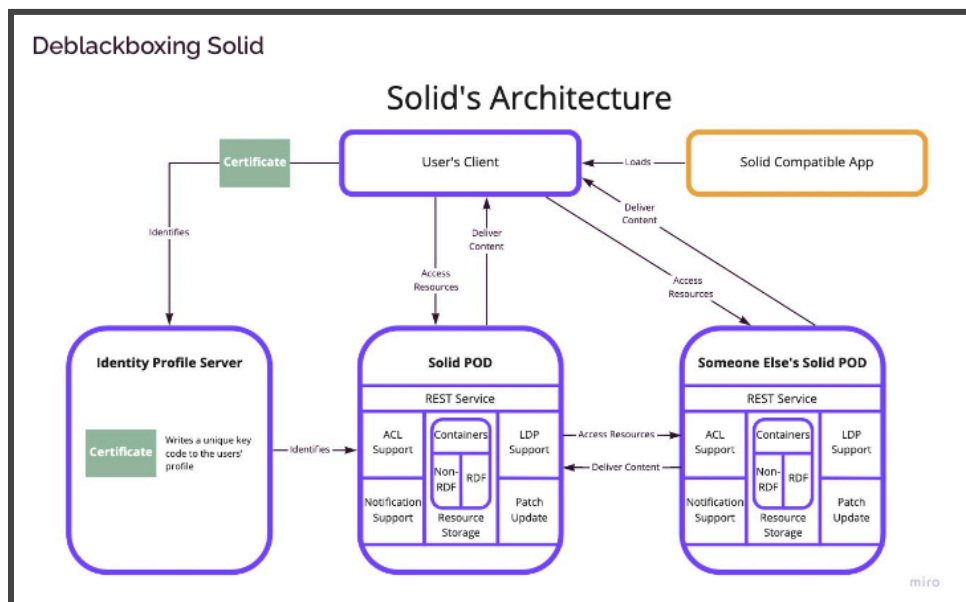
Space used	15 GB
Total apps	23
Containers	8

[Revoke Access from all apps](#)

Technical Architecture

Solid aims to break up ‘identity authentication’, ‘data hosting’, and ‘application interface’, functions typically all implemented by a single platform. This is an attempt to prevent vendor lock-in and enable greater degrees of interoperability between different service providers. For example, a large issue with the existing architecture is that if I would like to leave a platform like X (perhaps because of ideological differences with their chief executive), a user

would also be giving up the 'data' that was created through the platform, such as Tweets or replies or comments.



Architecture of solid framework

The separation of these functions into different entities would allow for users to seamlessly switch between different platforms – and even identity providers – as they please. Applications would thus compete on the quality of services and nature of user interface offered and cannot depend only on the data they have accumulated to maintain their hold on users. It may thus enable greater competition between newer app developers and expand user options beyond existing large platforms in return to more diverse web interfaces.

	Traditional web architecture	Solid-enabled web architecture	Podbox project
Identity authentication	Platform (eg. Facebook)	OIDC Provider (eg. Inrupt OIDC, Solid OIDC)	Inrupt OIDC
Data hosting	Platform (eg. Facebook)	Pod provider (eg. solidcommunity.net, Inrupt Pod Spaces)	Inrupt Pod Spaces
Application interface	Platform (eg. Facebook)	Any Solid compliant / enabled application	PodBox

The Podbox application attempts to experiment with this 'front-end only' architecture for application development, in which developers may focus on creating rich interfaces without

the need to manage back-end data management by leaving those to Solid and Pod providers.

Application interface

The project team undertook the development of a web application leveraging Next.js and React. Next.js is a popular React framework that provides server-side rendering and other performance optimizations, while React offers a component-based approach to building user interfaces.

We decided to use React due to the need for a dynamic web application and prior experience with working with Solid. Out of the [broader suite of libraries](#) available, we utilized the [JavaScript Client Libraries](#) for Solid application development. The libraries provided modules to read and write to users' Pod, authenticate users, and create and update existing information.

Identity Authentication

For authentication the Solid project uses the [WebID-OIDC protocol](#) which adds a layer on top of the OpenID Connect protocol which in turn is built on top of the OAuth 2.0 protocol. The WebID-OIDC protocol specifies a mechanism for getting a WebID URI from an OIDC ID Token.

When the user attempts to log in to a Solid-enabled web application, the user is redirected to an authentication process hosted by an 'identity provider'. The identity provider authenticates the username and password entered into the form and redirects them back to the web application.

Data hosting

POD providers host user and community pods in servers – which may themselves be hosted by some cloud provider or even in a rack at home. These POD providers are responsible for the secure storage of the data on the POD and implementing read and write access.

POD providers are currently using different implementations of the Solid server. The NSS (Node Solid Server) implementations are currently deprecated, but ESS (Enterprise Solid Server) and CSS (Community Solid Server) are updated regularly and also enforce cross-compatibility.

A long list of pod providers can be [found here](#). A snapshot is below.

Provider	Responsible for Domain Name and Terms	Responsible for Hosting	Hosting Location	Solid Implementation
Inrupt Pod Spaces	Inrupt, Inc.	Amazon	Germany	ESS
solidcommunity.net	Solid Project	Digital Ocean	UK	NSS
solidweb.org	Solid Grassroots	Hosteurope	France	NSS
trinpod.us	Graphmetrix, Inc.	Amazon	USA	TrinPod
use.id	Digita	DigitalOcean	EU	CSS
solidweb.me	Meisdata	Hosteurope	France	CSS
Data Pod	iGrant.io, Sweden	RedPill Linpro, AWS, GCP or Azure	EU	NSS
redpencil.io	redpencil.io	Hetzner	Finland	CSS
teamid.live	Meisdata	Hosteurope	France	CSS

A snapshot of pod providers.

Data Standards

A number of features of Solid are enabled by the nature of data storage. Data on a POD is stored as “Structured data”, which in this context, refers to Resource Description Framework (RDF) data that are represented as Things in SolidDatasets, where SolidDatasets are organized in Containers. The storage of data in this structured format allows for interoperability.

In PODs, data is structured as properties of a Thing, where a Thing refers to a data entity. Different attributes of Things are stored as ‘properties’ as ‘triples’ defined by RDF, enabling the creation of ‘Linked Data’. Containers and SolidDatasets are identified by a unique URL (Uniform Resource Locator) or URI (Uniform Resource Identifier). More information about Solid

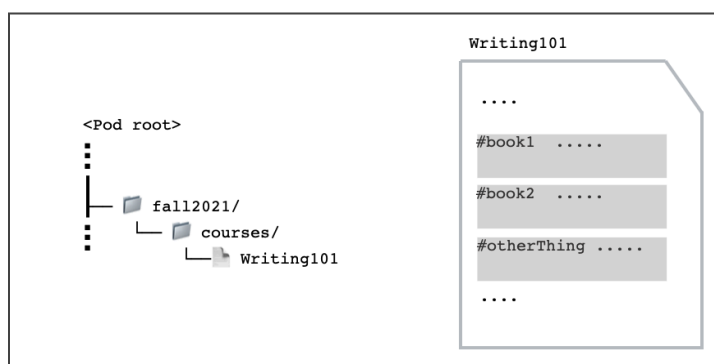


Illustration of data storage format

Here, ‘fall2021’ is the container, which contains another container, containing a ‘SolidDataset’ that corresponds to the information stored about a single course. For example, in the above, if Pod

URL is given by “https://storage.inrupt.com/<Root Container>/:”, the Container, has the URL “https://storage.inrupt.com/<Root Container>/fall2021/.” More [information is here](#).

Vocabularies for data are defined by entities such as ‘[schema.org](#)’ and are required to be referenced during any interaction with user data stored in a POD. Solid leverages schema.org vocabularies to define the structure and semantics of data stored within SolidDatasets. These vocabularies provide a standardized way to describe entities and their relationships, ensuring interoperability and meaningful data exchange across applications. By adhering to schema.org vocabularies, Solid enables seamless integration and interpretation of data structured in RDF format, facilitating efficient data management and utilization within the framework.

In the context of our project, we utilize the vocabulary for ‘name’ and ‘email address’ when entering this information into the user’s profile, stored on their pod. Information about their favorite applications are also stored in the Pod using a specific vocabulary.

Learnings

- Convincing users to embrace data control and migrate to a decentralized platform requires substantial education and behavior change efforts.
- There is an increasing concern among users about their privacy and ownership, but also high friction in adoption when introducing new systems like PodBox
- The absence of established precedents and novelty of the project presented some engineering challenges. However, the community has provided exceptional support which has been instrumental in the successful completion.
- Decentralized applications like Bluesky, Flipboard, and Threads (incorporating ActivityPub) are gaining recognition, the primary friction lies in conveying the value proposition and onboarding users.

Next Steps

- Refine the PodBox application and showcase it on the official Solid website to enhance accessibility and credibility.
- Collaborate with industry experts to gather insights on how PodBox can contribute to broader adoption and address user needs effectively.
- Improve access controls within PodBox, focusing on providing users with more granular control over their personal data.
- Develop visualization within PodBox to help users visualize and understand their personal data, empowering them to make informed decisions.

Project Links

[Wireframes on Figma](#)

[Figma Designs](#)

[Github Repository](#)

[Project Website](#)

Resources

- [Monday.com UI kit for Design Systems](#)
- [Assessing the Solid Protocol in Relation to Security and Privacy Obligations](#)
- [Attribute-based Access Control on Solid Pods using Privacy-friendly Credentials](#)
- [SOLID pods for social media - Master's Thesis](#)
- [Automating the response to GDPR's Right of Access](#)
- [A Policy-Oriented Architecture for Enforcing Consent in Solid](#)
- [Investigating decentralized management of health and fitness data](#)
- [CiTizen-centric DATA pLatform \(TIDAL\): Sharing distributed personal data in a privacy-preserving manner for health research - IOS Press](#)
- [A Solid-Powered Collaborative Rule Management Tool for Sharing Patient Data](#)

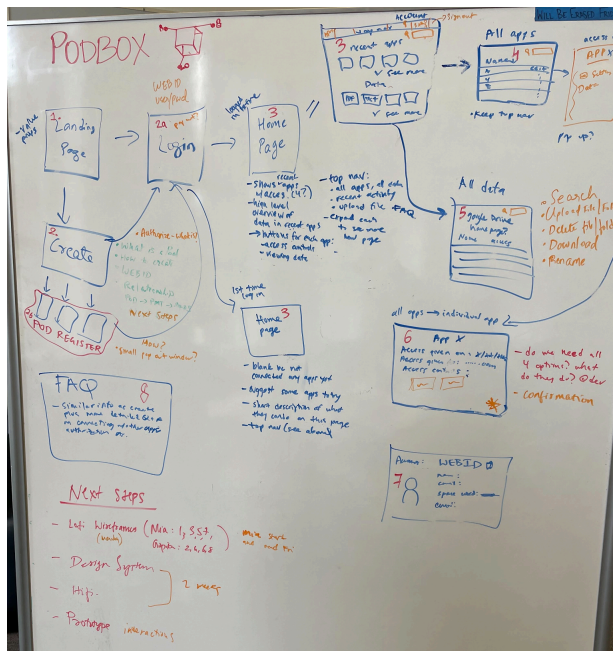
Appendix

Jobs to be done

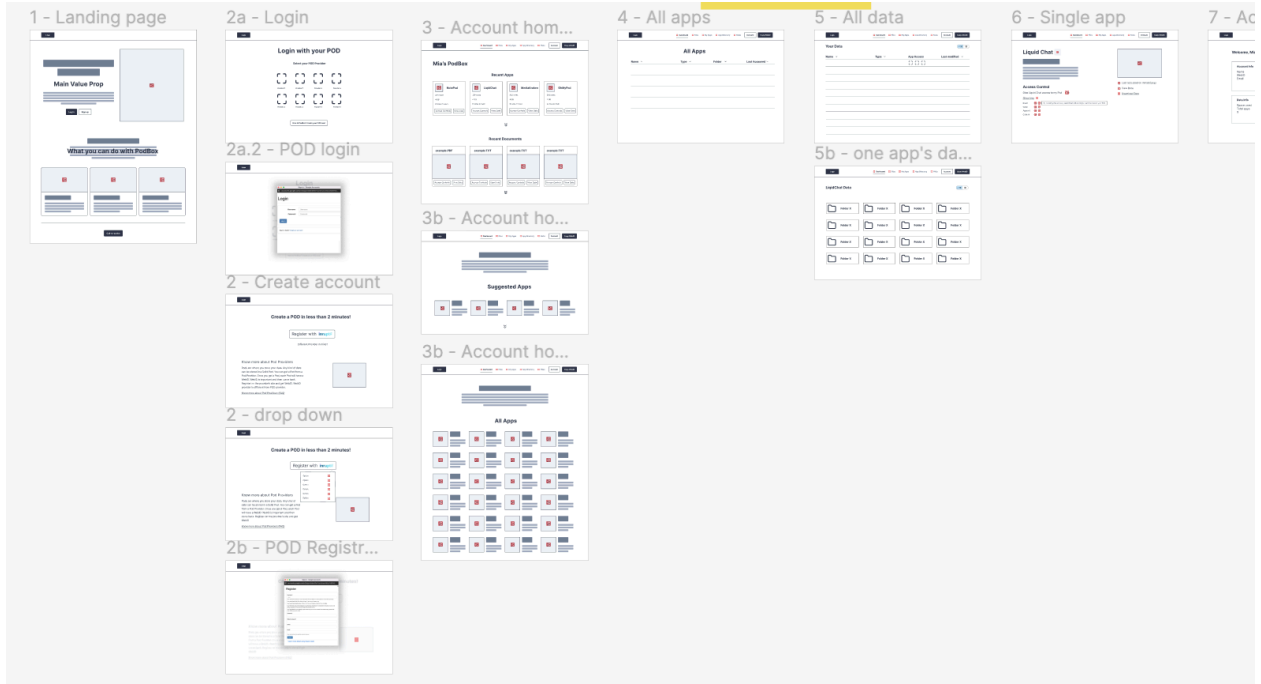
Based on our User Interviews and Initial Research, we came up with the following User Stories-

Feature	User Story	Priority
User Onboarding and Login	As a user, I want to login into PodBox with my username/password	High
	As a user, I want to see my POD provider on the login page	High
	As a user, if I don't have a POD, I want to redirect to the page to create a POD	High
	As a user, I don't want to login every time I open the app	High
	As a user, I want to sign-out when my task is accomplished	High
File Management System	As a user, I want to see a high level folder structure based on topics	High
	As a user, I want to download my files	Low
	As a user, I want to see/download my data in different formats	Low
	As a user, I want a search option to look for my file	High
	As a user, I want to be able to identify newly added files	High
	As a user, I want to be able to identify newly added folders	High
	As a user, I want to delete folder	High
	As a user, I want to delete files	High
	As a user I want to sort my folder/files	Medium
	As a user, I want to share files/folders	Low
	As a user, I don't want to see files/folders that is not my data	High
Access Management	As a user, I want to see what apps have what access control	High
	As a user, I want to change access different apps have	High

	As a user, I want to be notified when there's a change in access control of my apps	Low
Apps	As a user, I want to see all Solid apps in one place	High
	As a user, I want to see a recommended app to access my data	Medium
FAQ	As a user, I want to know more about PODs in FAQs	High
Profile	As a user, I want to see my profile details	Low
	As a user, I want to update my profile	Low
Home	As a user, I want to copy my WebID	High



User flow brainstorming



Original wireframe of user flow