Exploring
Self-Sovereign Identity

SRS Document

VERSION 4

Index

Appendix A: External Links

1. User Manual
2. Technical Installation Manual
3. Coding Standards Document
4. Github Repo
5. Code of Duty SSI Wiki
6. Team Member Profiles
7. SSI Project Board
8. Literature Review
9. Deployment Model
10. Architectural Structural Design & Requirements

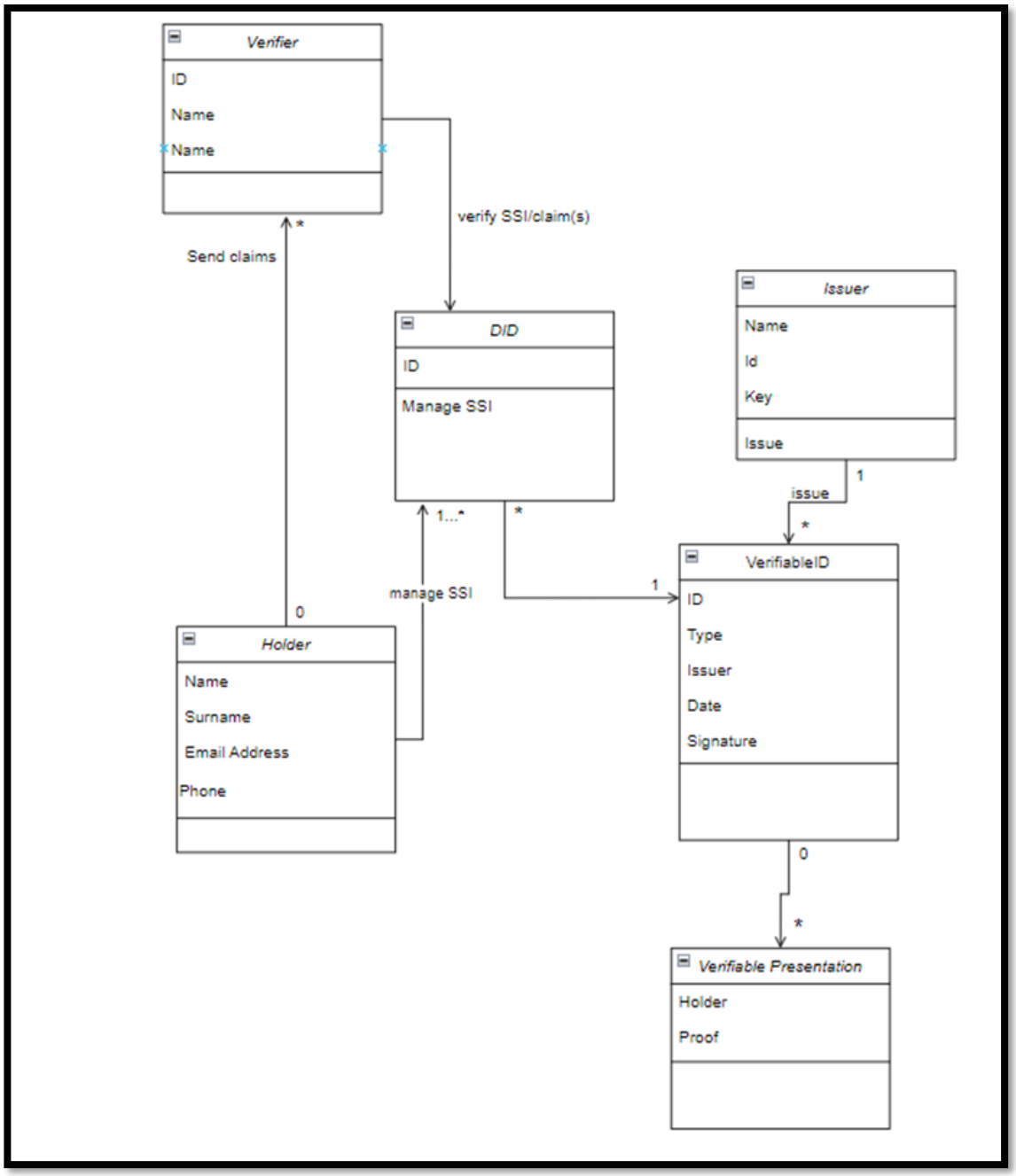
1. Introduction

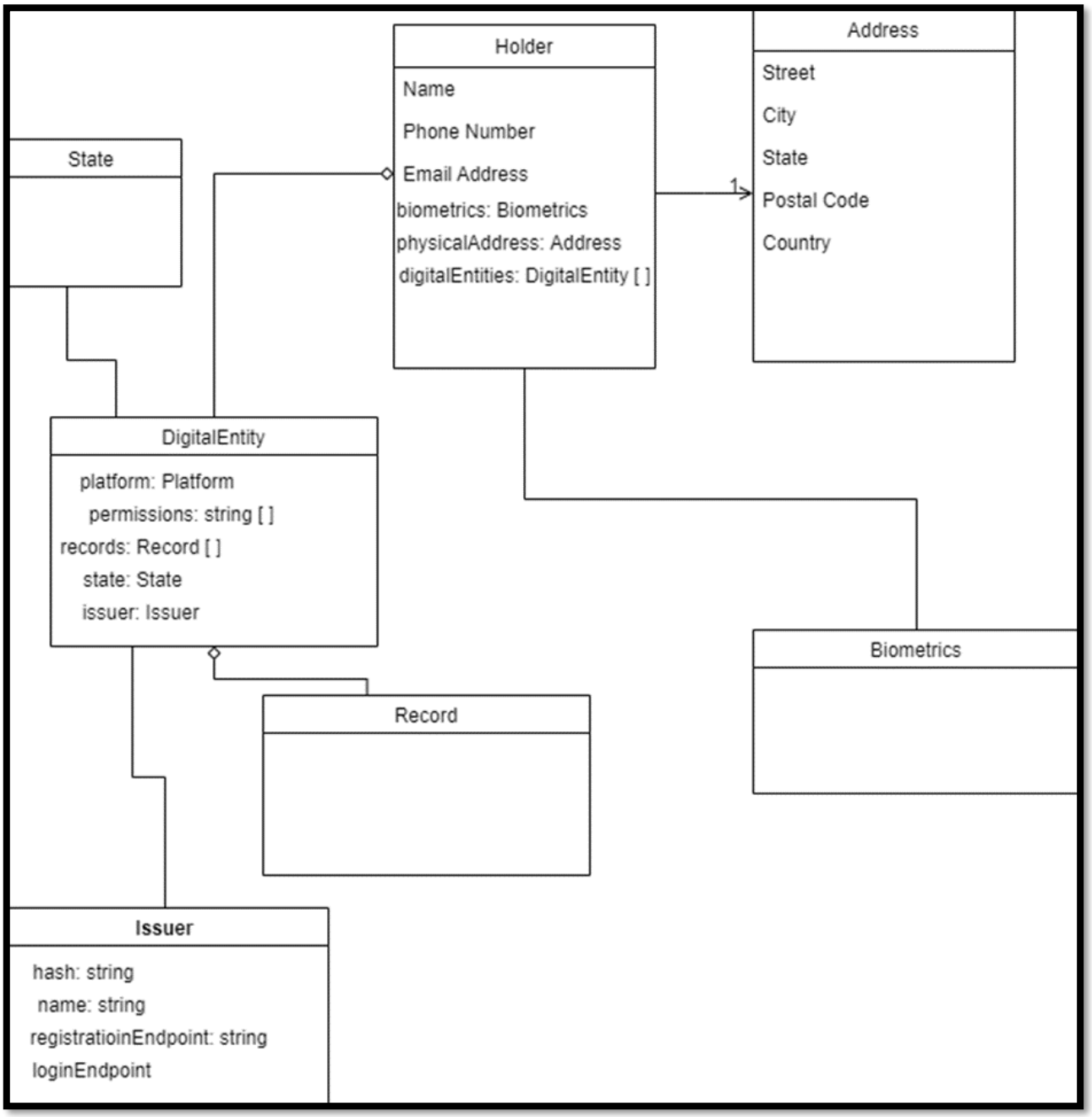
1.1. Abbreviations

1.1.1. Self-Sovereign Identity – SSI

The SSI system will act as a steppingstone into the world of digital identities, allowing users to have full control over their online assets. Digital Identities will be linked to users using real-world properties (such as biometrics) which will only allow the legitimate user access to their information. The legitimate user will also have control over the information made available to the public. The main purpose of the project is to explore the world and use cases around SSI and come up with exciting ideas to solve it.

2. Class Diagrams




Holder
Name
Phone Number
Email Address
biometrics: Biometrics
physicalAddress: Address
digitalEntities: DigitalEntity [ ]
Address
Street
City
State
Postal Code
Country
1
State
DigitalEntity
platform: Platform
permissions: string [ ]
records: Record [ ]
state: State
issuer: Issuer
Biometrics
Record
Issuer
hash: string
name: string
registratioinEndpoint: string
loginEndpoint

3. User Characteristics

Version 1

3.1. Holder

Holders will be the owners of the digital identities. Holders will be able to manage identities they own, sign off (or decline) on new ones and have access to those identities and the data they possess. Identities will be linked to real-world data (like biometrics).

3.2. Institute

The Institute can manage what the identity is (like add/remove attributes). Institutes will have unique codes to distinguish from each other and prevent fraudulent claims which are linked to domains. The Institutes will be able to implement functionality from our API and effortlessly incorporate our system into theirs to allow a smooth and easy transition for any interested users.

3.3. Verifier (to be expanded)

Verifiers are 3rd party users that will be able to lookup the affiliation on an identity to ensure its validity.

Version 2, 3 & 4

3.1. Issuer

Issuers will be the owners of the digital identities. Issuers will be able to manage identities they own, sign off (or decline) on new ones and have access to those identities and the data they possess. Identities will be linked to real-world data in the form of biometrics. Fingerprint biometrics will be used at this stage of the project, but further expansion into retinal and face recognition is proposed.

3.2. Organization

The Organization can manage what the identity is (like add/remove attributes). Organizations will have unique codes to distinguish from each other and prevent fraudulent claims which are linked to domains. The Organizations will be able to implement functionality from our API and effortlessly incorporate our system into theirs to allow a smooth and

easy transition for any interested users. The binding between the Issuer and Organization exists in the form of a contract with the user's signature, a basic foundation for Smart Contracts.

3.3. Credential Authority

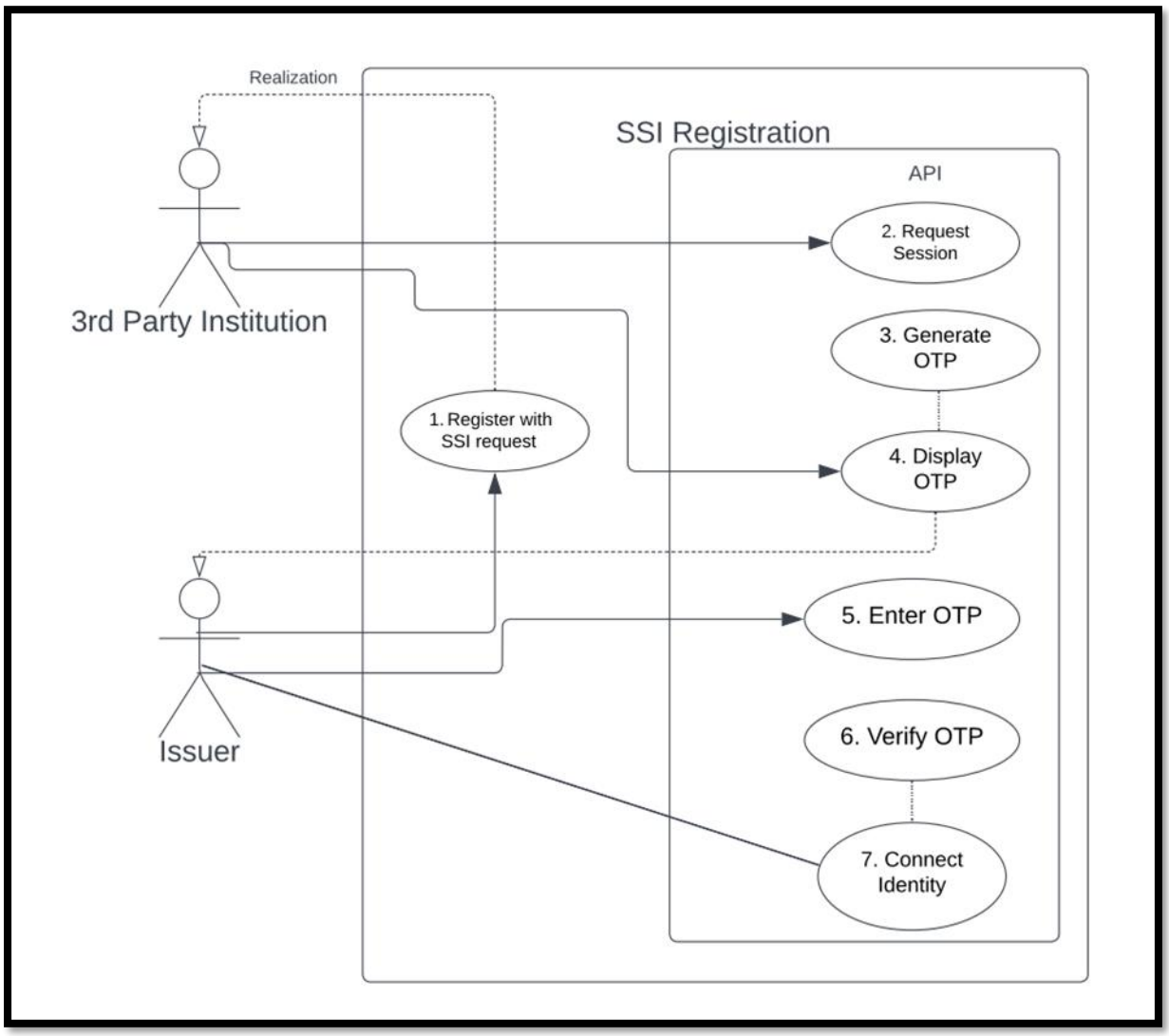
Credential Authorities are 3rd party users that will be able to lookup the affiliation on an identity to ensure its validity.

4. Functional Requirements

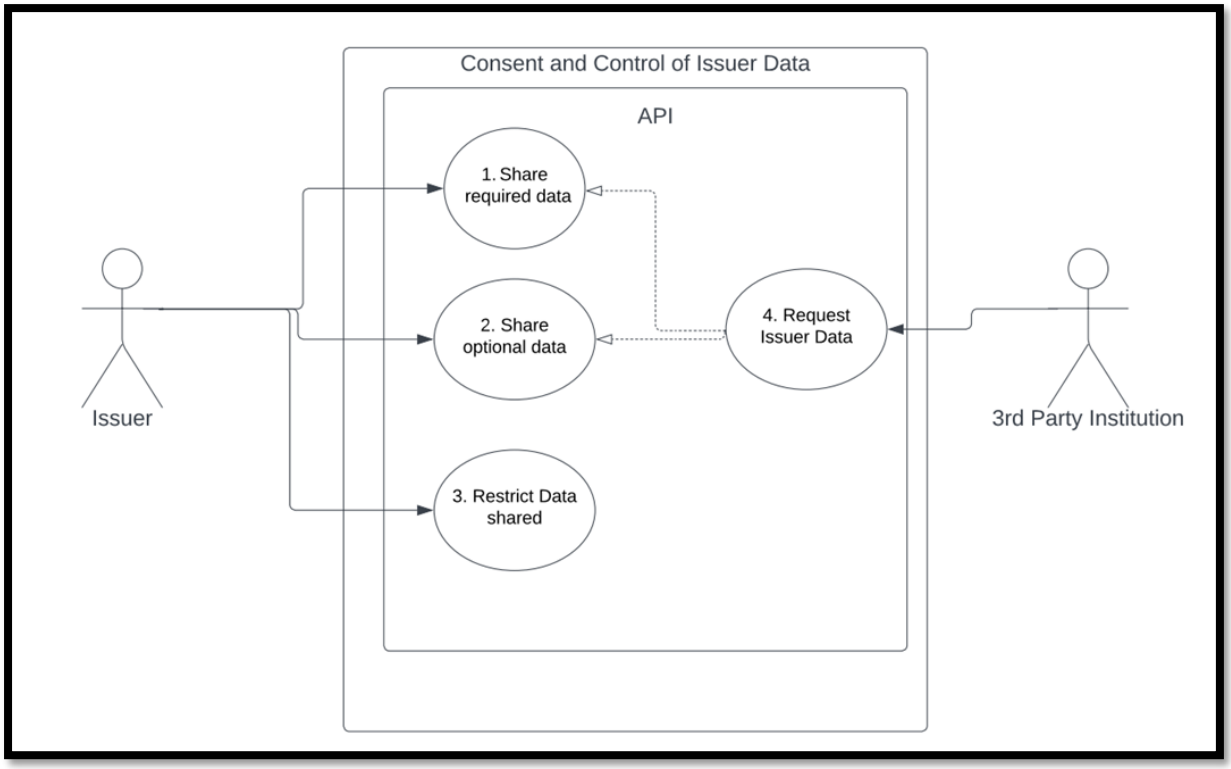
4.1. Use Cases

Version 1

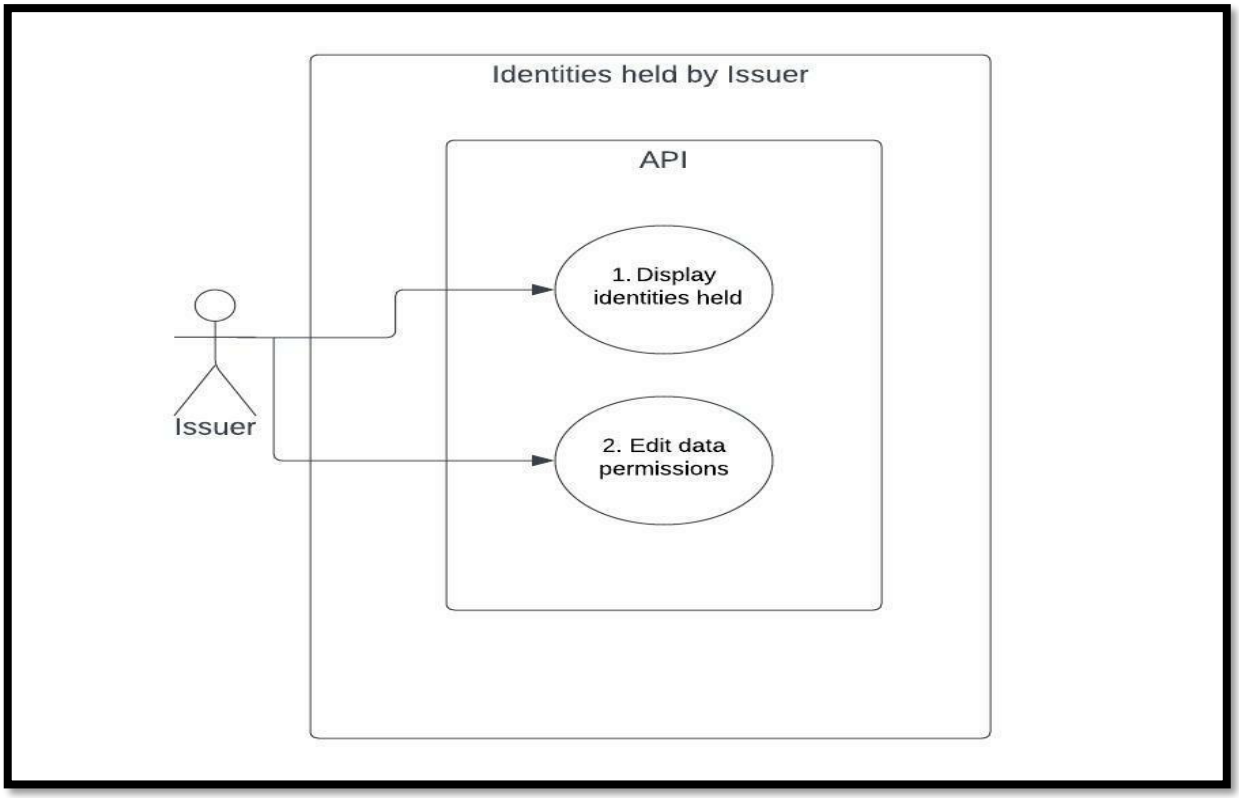
UCD1: SSI Registration



UCD2: Consent and Control of Issuer Data

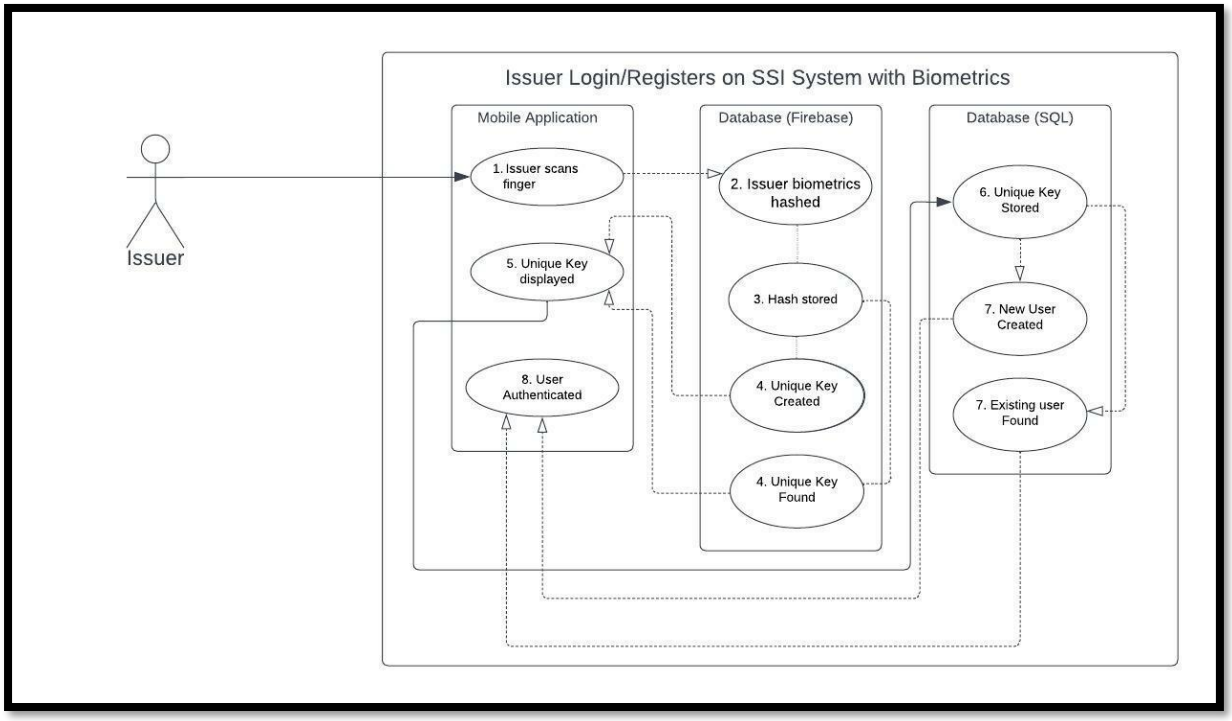


UCD3: Identities held by Issuer

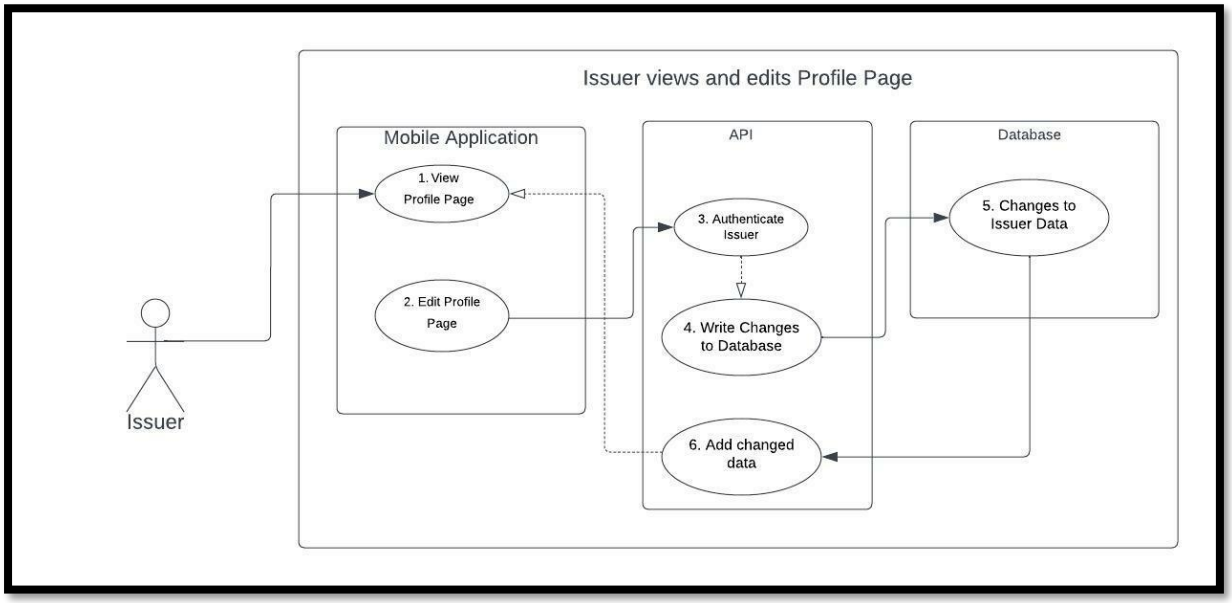


Version 2

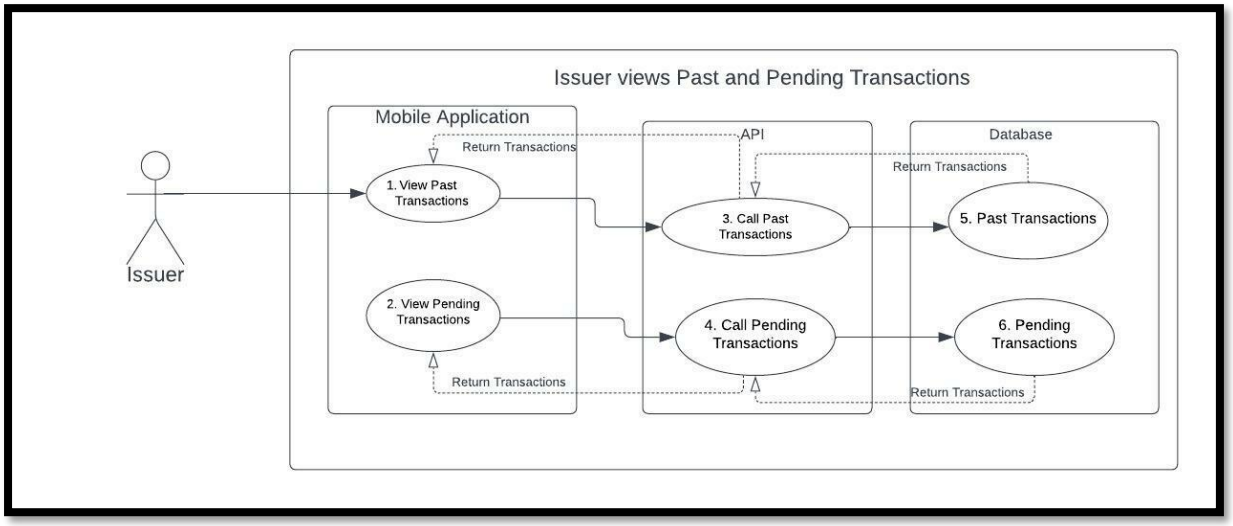
UCD1 & UCD2: Issuer Login/Registers on SSI System with Biometrics



UCD3 & UCD4: Issuer views and edits Profile Page

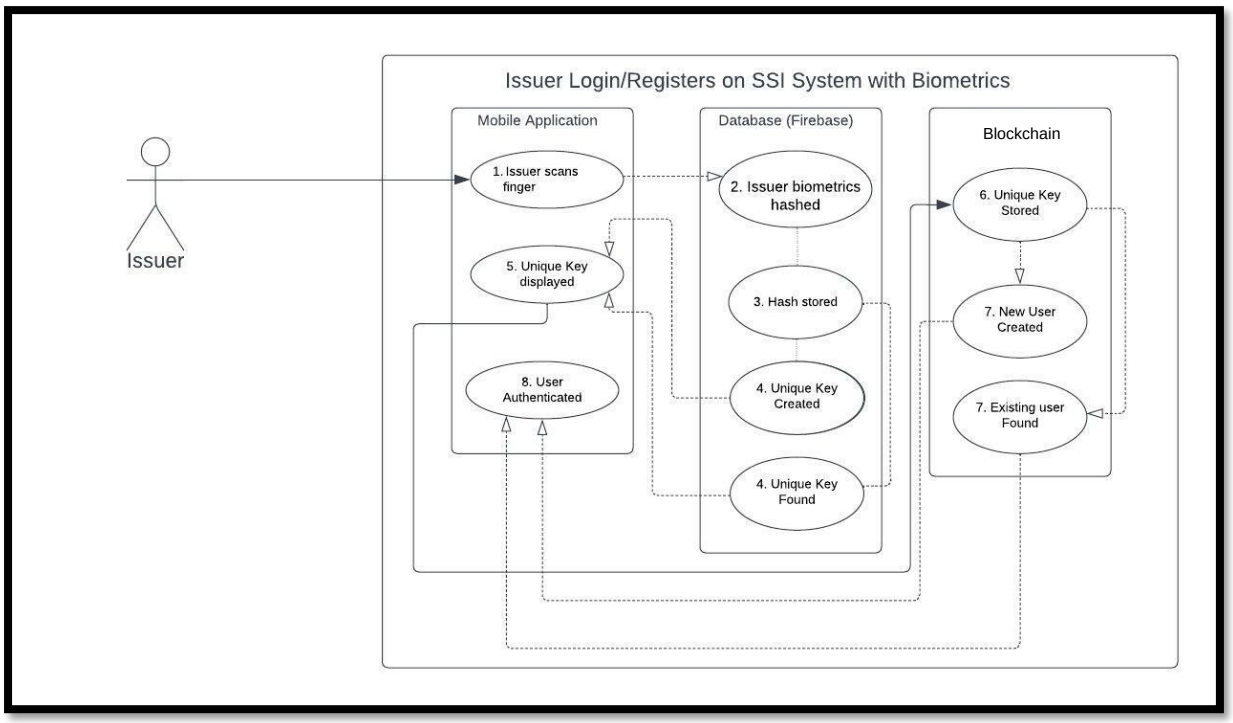


UCD5: Issuer views Past and Pending Transactions

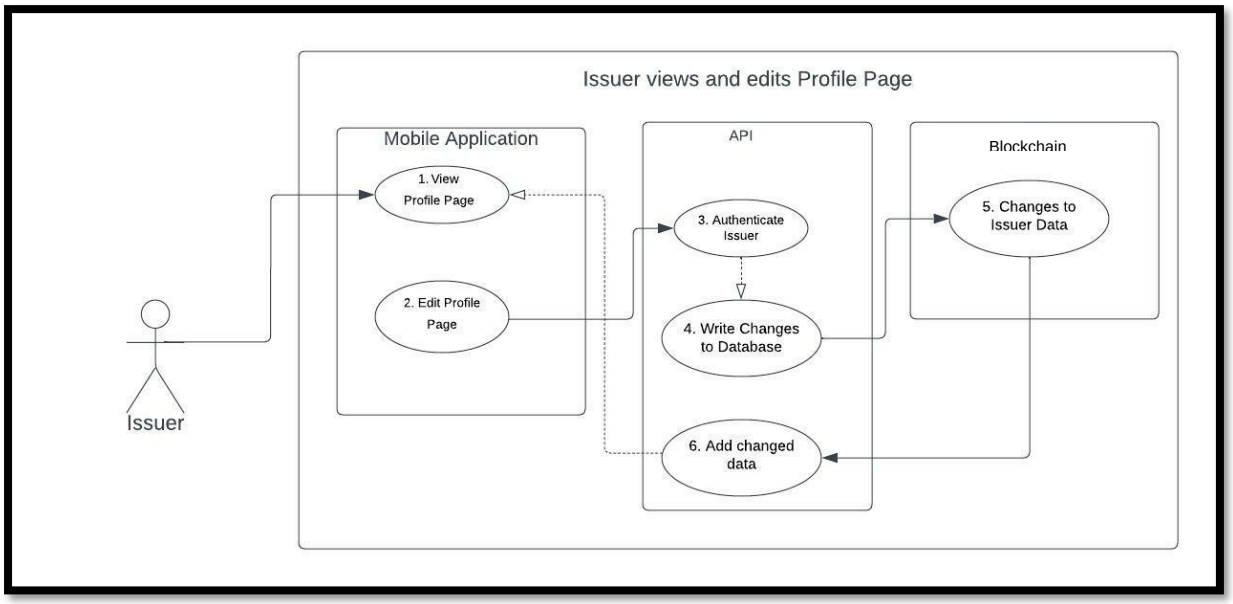


Version 3: Same User Functionality Modified for Blockchain

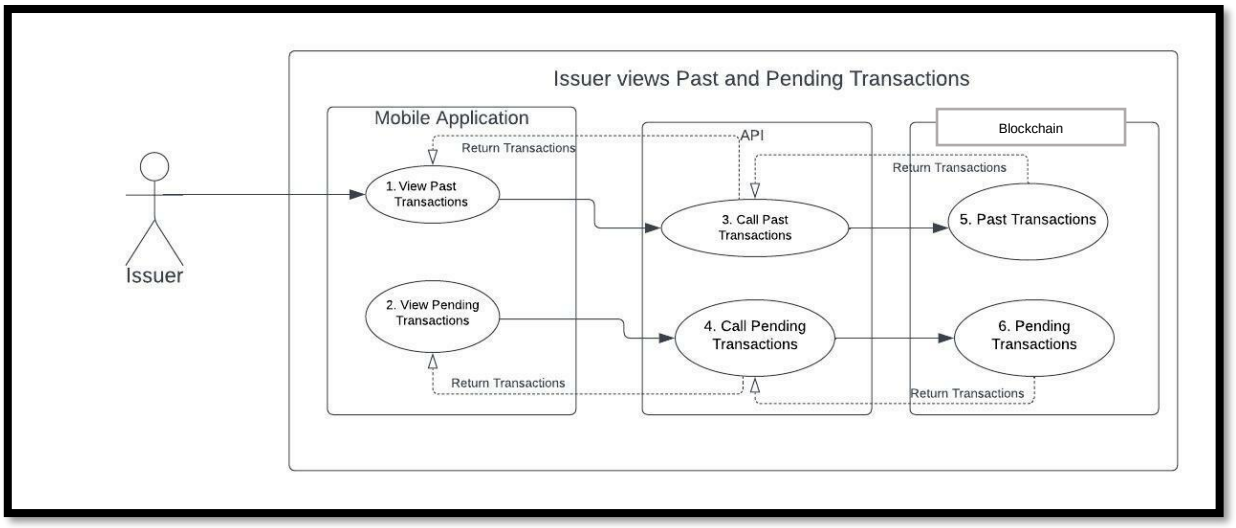
UCD1 & UCD2: Issuer Login/Registers on SSI System with Biometrics



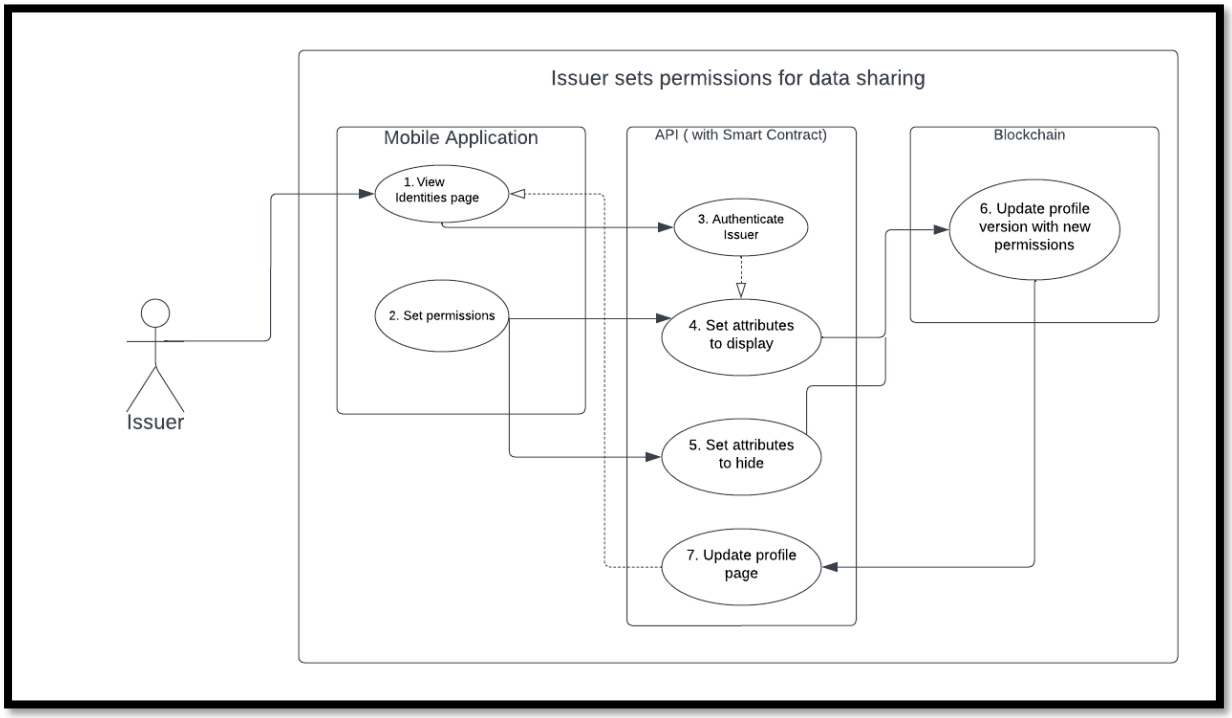
UCD3 & UCD4: Issuer views and edits Profile Page



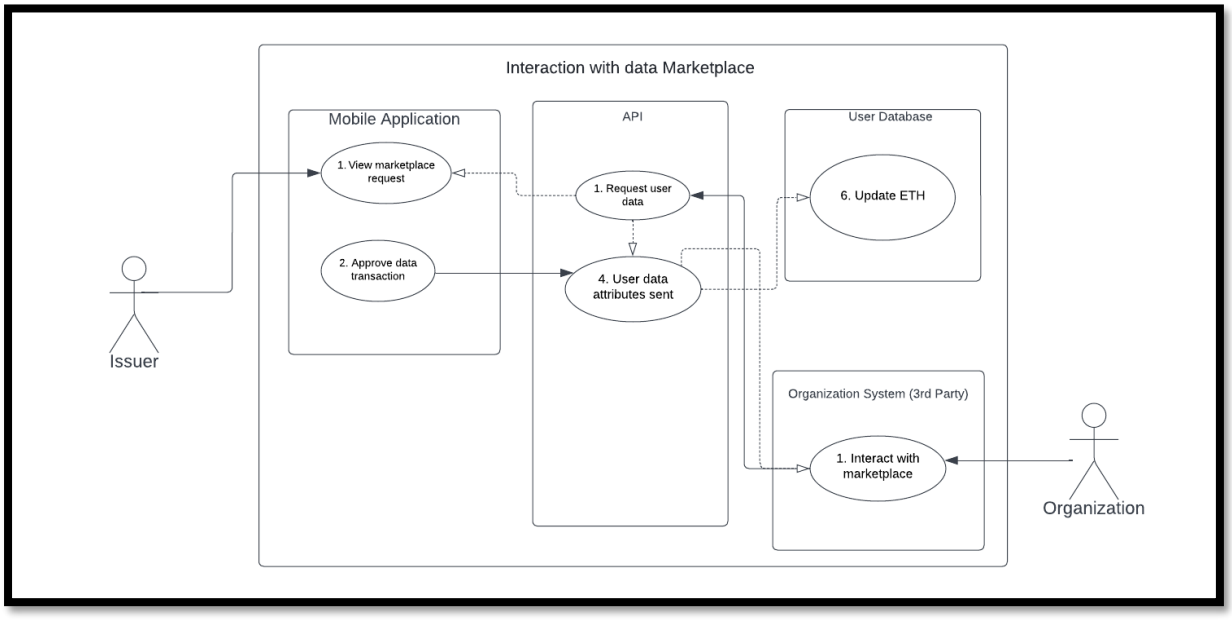
UCD5: Issuer views Past and Pending Transactions



UCD6: Issuer sets permissions for data sharing



Version 4: All previous accumulated use cases and new feature



4.2. Requirements

Functional

- R1. The SSI-System shall allow Users to manage their data.
    - R1.1. The SSI-System shall allow Users to register with biometric data.
    - R1.2. The SSI-System shall allow Users to login with biometric data.
    - R1.3. The SSI-System shall allow Users to approve requests for stored and encrypted-data.
    - R1.4. The SSI-System shall allow Users to deny requests for stored and encrypted-data.
    - R1.5. The SSI-System shall allow Users to update stored encrypted-data.
        - R1.5.1. The SSI-System shall allow Users to edit stored attribute values.
        - R1.5.2. The SSI-System shall allow Users to add and store attributes.
        - R1.5.3. The SSI-System shall allow Users to remove stored attributes.
- R2. The SSI-System shall allow Institutes/organizations to manage their credentials.
    - R2.1. The SSI-System shall allow Organizations to register credentials.
    - R2.2. The SSI-System shall allow Organizations to manage credential templates.
        - R2.2.1. The SSI-System shall allow Organizations to add attributes within credentials.
        - R2.2.2. The SSI-System shall allow Organizations to remove attributes within credentials.
- R3. The SSI-System shall allow data requests.
    - R3.1. The SSI-System shall allow Organizations to request data.
    - R3.2. The SSI-System shall allow Users to request data.
- R4. The SSI-System shall allow User-Organization authentication.

    - R4.1. The SSI-System shall allow Organizations to create sessions.
    - R4.2. The SSI-System shall allow Users to join sessions.
    - R4.3. The SSI-System shall allow data requests through sessions.

Non-Functional

- NF1. The SSI-System shall be hosted on Azure.
- NF2. The SSI-System shall store identity data in a decentralized manner.

4.3. Subsystems

1. Blockchain layer

This is a decentralized system where SSI is stored and managed, this gives individuals control of their digital identities and ensure that it does not depend on any centralized authority and can never be taken away.

2. Session Subsystem

This subsystem will handle the issuer and verifier portal and session for managing the entity

3. Mobile Subsystem

This is where the Holders will interact with their data in the form of sharing data, granting, and denying access. Holders will have access to the Marketplace functionality to monetize their data by selling it to third-party organizations.

4. Third-Party Integration

This sub-system will integrate the parties that will be the third-party institutions and credential authorities. These institutions can use the Marketplace functionality to buy user data.

5. Quality Requirements

5.1 Non-functional Metrics

Version 1

- The software must accommodate for secure storage of user info and certificates
- The software must be scalable for a large user base
- The software must be able to retrieve data at reasonable speeds
- The software must be straight to the point and easy for the user with no prior blockchain knowledge to use
- The software must be easily accessible from anywhere
- The software must be reliable in fetching the correct user data

Version 2

a. Security

- A core requirement is that of data and identity security. The system must only allow access to sensitive data through use of a decryption key, which can only be accessed via biometric data (thus unique to each user).
- Data transfers over the internet should be secure, thus HTTPS protocol with TLS should be used on Layer 4.

b. Reliability

- Since data is linked to biometrics, the chance for data loss is limited to the user losing their biometric identity (nearly impossible).
- The decryption needs to validate successful data decryption to ensure no invalid data is transmitted. Thus a value will be stored and tested each time data is decrypted.

c. Availability

- Since having access to your data anywhere and anytime is crucial, the system should have an ideal uptime of between 95%-99%.

d. Scalability

- The decentralized version of the app will allow high scalability, however at its current centralized state, it should be able to allow at least 20 requests per minute.

e. Portability

- The SSI app/system needs to be used interchangeably between different devices for the same user, retaining the same data/functionality. 1 user with multiple supported devices should have the same functionality on each separate device.
- Since the app/system will be a cloud hosted service, it will be available/usable/installable on devices with internet connection. Thus the app will be 100% available to anyone with internet access and supported devices.

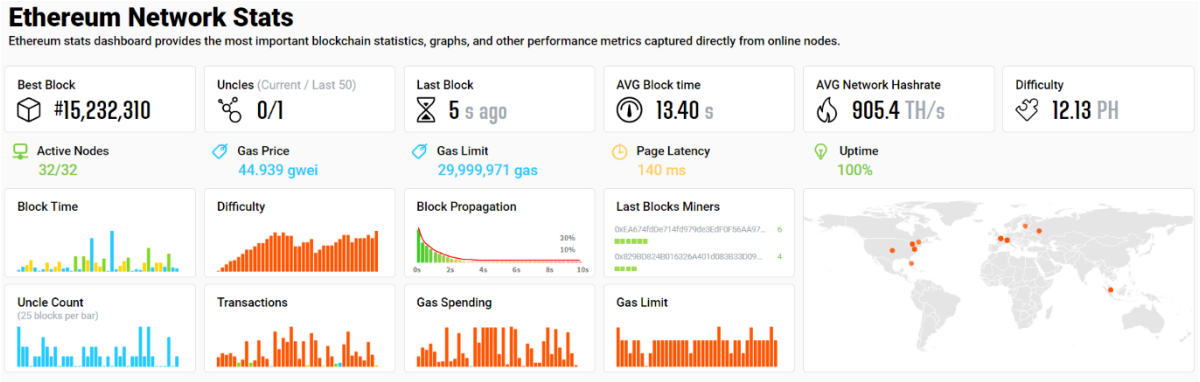
f. Usability

- The system gives a user full control of their data and identity. This control should not overwhelm the user, but be easily and efficiently managed.
- The system should be accessible anywhere where the user has access to internet and a smart phone (for biometric authentication).

g. Compatibility

- The system/app should not be constricted by use of different devices. Smart phones should be able to interact with different versions/brands as long as both have biometric capabilities.
- Ensure different biometric data from different devices are interchangeable.

Version 3



a. Security

- A core requirement is that of data and identity security. The system must only allow access to sensitive data through use of a decryption key, which can only be accessed via biometric data (thus unique to each user).
- Data transfers over the internet should be secure, thus HTTPS protocol with TLS should be used on Layer 4.
- The Ethereum blockchain is heavily secured and has experienced only four large scale attacks in its long history. Recovery from these breaches was efficient, with minimal data loss.

b. Reliability

- Since data is linked to biometrics, the chance for data loss is limited to the user losing their biometric identity (nearly impossible).
- The decryption needs to validate successful data decryption to ensure no invalid data is transmitted. Thus a value will be stored and tested each time data is decrypted.
- The Ethereum blockchain is reliable by nature, as seen by the uptime and very few attacks and issues.

c. Availability

- The Ethereum Blockchain on which the system is based has an uptime of 100%. The high uptime can be attributed to decentralization and security. Decentralization ensures availability of the blockchain in many instances, simultaneously. Ethereum, as one of the older cryptocurrencies, is well-known by developers, allowing them to plan for and solve unforeseen issues before they occur.

d. Scalability

- The decentralized version of the app allows high scalability, where its previous centralized state, allowed for at least 20 requests per minute.
- Ethereum Blockchain allows for 30 transactions per second

e. Portability

- The SSI app/system needs to be used interchangeably between different devices for the same user, retaining the same data/functionality. One user with multiple supported devices should have the same functionality on each separate device.
- Since the app/system will be a cloud hosted service, it will be available/usable/installable on devices with internet connection. Thus the app will be 100% available to anyone with internet access and supported devices.
- The blockchain is also accessible to anyone using the app, via the Smart Contract, set up for that purpose. The Decentralization of the app and Ethereum itself contributes to its portability.

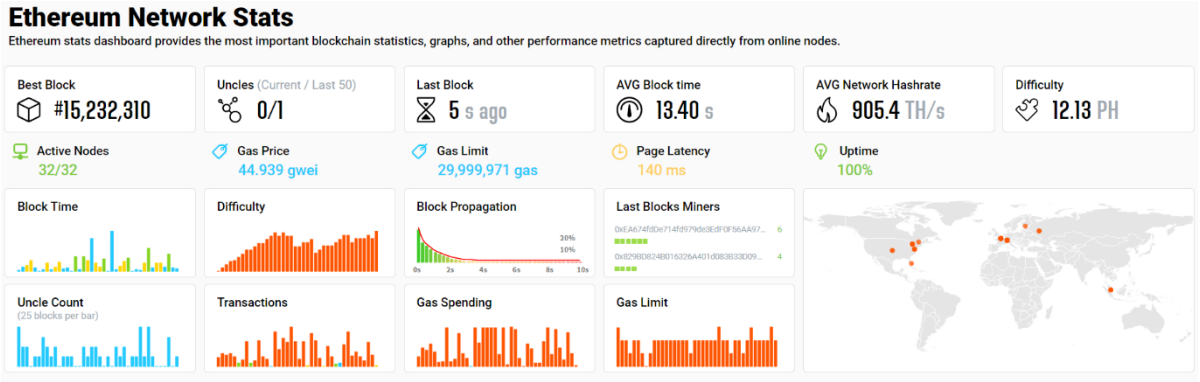
f. Usability

- The system gives a user full control of their data and identity. This control should not overwhelm the user but be easily and efficiently managed.
- The system should be accessible anywhere where the user has access to internet and a smart phone (for biometric authentication).
- The user does not have to understand blockchain or deal with the technicalities surrounding Ethereum. They can simply transact, and the system handles the complexities.

g. Compatibility

- The system/app should not be constricted by use of different devices. Smart phones should be able to interact with different versions/brands as long as both have biometric capabilities.
- Ensure different biometric data from different devices are interchangeable.

Version 4



a. Security

- A core requirement is that of data and identity security. The system must only allow access to sensitive data through use of a decryption key, which can only be accessed via biometric data (thus unique to each user).
- Data transfers over the internet should be secure, thus HTTPS protocol with TLS should be used on Layer 4.
- The Ethereum blockchain is heavily secured and has experienced only four large scale attacks in its long history. Recovery from these breaches was efficient, with minimal data loss.
- User data and secrets are encrypted using secure encryption keys.

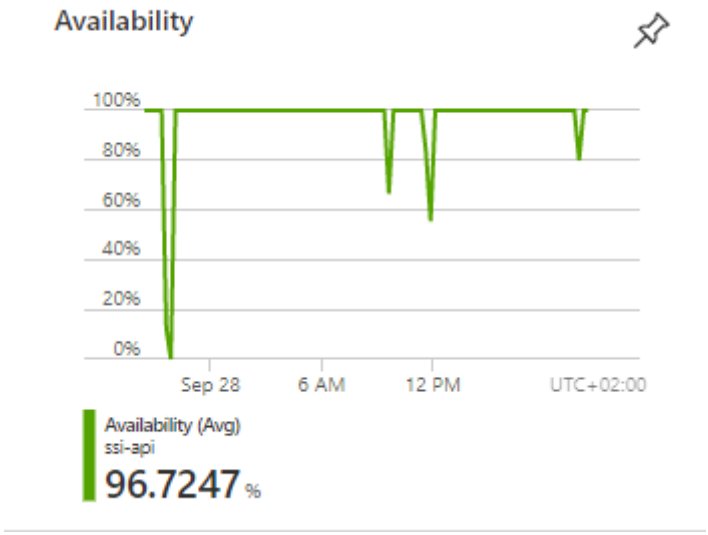
b. Reliability

- Since data is linked to biometrics, the chance for data loss is limited to the user losing their biometric identity (nearly impossible).
- The decryption needs to validate successful data decryption to ensure no invalid data is transmitted. Thus a value will be stored and tested each time data is decrypted.
- The Ethereum blockchain is reliable by nature, as seen by the uptime and very few attacks and issues.

c. Availability

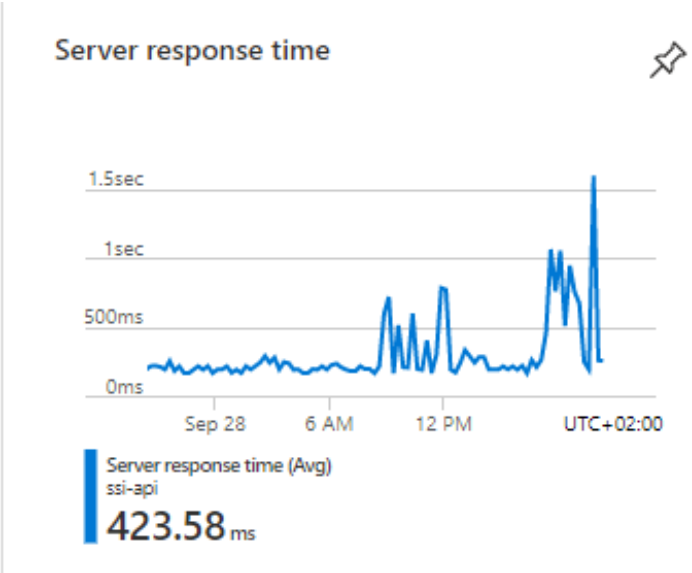
- The Ethereum Blockchain on which the system is based has an uptime of 100%. The high uptime can be attributed to decentralization and security. Decentralization ensures availability of the blockchain in many instances, simultaneously. Ethereum, as one of the older cryptocurrencies, is well-known by developers,

allowing them to plan for and solve unforeseen issues before they occur.



d. Scalability

- The decentralized version of the app allows high scalability, where its previous centralized state, allowed for at least 20 requests per minute.
- Ethereum Blockchain allows for 30 transactions per second



e. Portability

- The SSI app/system needs to be used interchangeably between different devices for the same user, retaining the same data/functionality. One user with multiple supported devices should have the same functionality on each separate device.
- Since the app/system will be a cloud hosted service, it will be available/usable/installable on devices with internet connection. Thus the app will be 100% available to anyone with internet access and supported devices.
- The blockchain is also accessible to anyone using the app, via the Smart Contract, set up for that purpose. The Decentralization of the app and Ethereum itself contributes to its portability.

f. Usability

- The system gives a user full control of their data and identity. This control should not overwhelm the user but be easily and efficiently managed.
- The system should be accessible anywhere where the user has access to internet and a smart phone (for biometric authentication).
- The user does not have to understand blockchain or deal with the technicalities surrounding Ethereum. They can simply transact and the system handles the complexities.

g. Compatibility

- The system/app should not be constricted by use of different devices. Smart phones should be able to interact with different versions/brands as long as both have biometric capabilities.
- Ensure different biometric data from different devices are interchangeable.

6. Trace-ability Matrix

Functional/Quality	BlockchainLayer	Session Subsystem	Mobile Subsystem	3rd Party Subsystem
R1.1.		x	x	
R1.2.		x	x	
R1.3.		x	x	
R1.4.		x	x	
R1.5.			x	
R2.1.		x		x
R2.2.		x		x
R3.1.	x	x		
R3.2.	x	x		
R3.3.		x	x	x

7. Technology Requirements

User

- The user will be able to access the SSI system and application via PC, Desktop, tablet and smart phone
- Since the system is decentralized, it has no device-specific requirements to support the system, aside from an internet connection.

Developer

- A developer wishing to run the system in their local environment must have a computer that can run the API, Front-end application, Smart Contract and test blockchain (Sufficient memory and computing power)
- An IDE such as Visual Studio is the recommended environment to set-up the repository
- See Technical Installation Manual for more specific requirements

Appendix A

Links to external documentation

1. User Manual: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-Identity/wiki/Manuals-&-Tutorials

2. Technical Installation Manual: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-Identity/wiki/Manuals-&-Tutorials

3. Coding Standards Document: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-Identity/files/9847262/Coding.Standards.-.Code.of.Duty.SSI.pdf

4. Github Repo: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-Identity

5. Code of Duty SSI Wiki: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-Identity/wiki

6. Team Member Profiles: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-Identity/wiki/Team-Profiles

7. SSI Project Board: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-Identity/projects?type=classic

8. Literature Review: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-Identity/files/9453546/Literature.Review.-.Code.of.Duty.Exploring.SSI.pdf

9. Deployment Model: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-Identity/files/9847268/Deployment.Model.-.Code.of.Duty.SSI.pdf

10. Architectural Structural Design & Requirements: https://github.com/COS301-SE-2022/Exploring-Self-Sovereign-