
Final Report of the Project

Context and Objectives of the Project:

This project is part of research on Self-Sovereign Identity (SSI) and the Digital Identity Wallet. The main objective is to design a taxonomy that structures and organizes the various aspects of these concepts based on the available scientific publications. This project aims to compare and bridge the gaps between academic and industrial approaches to digital identity management.

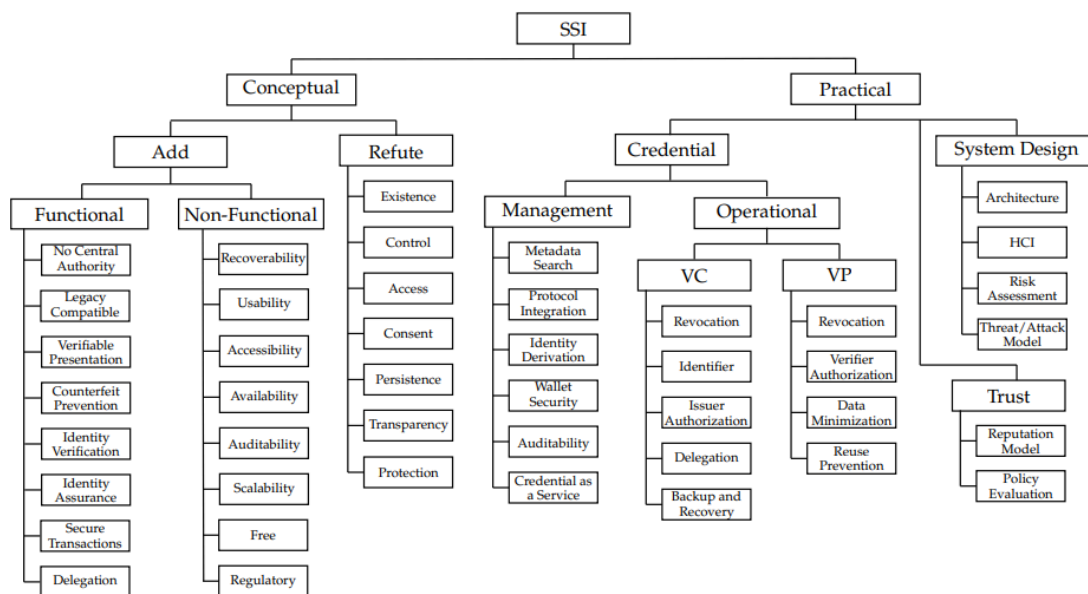


Figure 4: Taxonomy of SSI.

Project Methodology:

To successfully complete this project, we established a final methodology (see **Methodologie_Thales.drawio**). For more information, we have another document that provides more detailed information on the different stages of the methodology (see **All_Methodologie_Thales.drawio**).

The project is divided into four distinct phases:

- **Phase I:** Literature Review and Project Definition
- **Phase II:** Data Acquisition and Filtering
- **Phase III:** Taxonomy and Systematic Mapping
- **Phase IV:** Final Deliverables

Research Papers:

To understand the different scientific concepts and terms around SSI/Digital Identity Wallet, we have a set of research papers (see the **ResearchPaper** folder). This will also help us better understand the structure of a research paper, which will be very useful for the rest of the project.

Analysis and Extraction of Research Papers from an Academic Database:

To develop this taxonomy, we need data, specifically research papers. To retrieve these papers, we need to extract them from an academic database. We chose the **Semantic Scholar** academic database. This choice was based on several criteria, but the main reason is that it provides a free API that allows for easy data manipulation.

We provide you with the link to the API documentation:

[Semantic Scholar API Documentation](#)

We will now focus on the notebook

ExtractResearchPaperSemanticScholar.ipynb. In this notebook, we proceeded in small steps to reach a final stage where all these small steps, combined (result pagination, application of criteria, etc.), form a complete code (see the section **Final Step: Retrieving and Saving Papers in Excel, JSON, and CSV Formats via the Semantic Scholar API**). When we talk about retrieving research papers, we are specifically referring to the fields we extract (authors, title, abstract, id, etc.).

A small clarification regarding the **Query**: when querying the database, we use two terms, "**Self-Sovereign Identity**" and "**Digital Identity Wallet**". We run the code once for each term. After extracting the research papers, we add a **references** column and fill it with the paperId of the papers referenced in each article (see the section **Updating Paper References via the Semantic Scholar API**). More detailed explanations are available directly in the notebook..

IMPORTANT: The addition of references has not been done for all the research papers, whether they are on **Self-Sovereign Identity** or **Digital Identity Wallet**. You simply need to run the following code sections: "**Updating Paper References via the Semantic Scholar API**" and "**Updating Paper References with Pagination and Error Handling via the Semantic Scholar API**", by loading the respective files.

Algorithm and Clustering:

We will now focus on the notebook **Cluster_Algorithm.ipynb**. We used completely different data, on another topic, for several reasons:

- We already know the results of this data.
- The purpose of these various algorithms and methods is to test their effectiveness and ensure they function correctly on data with known results.

In the short term, the goal is to apply these algorithms and methods to the previously extracted research papers (with references) on the topics of **Self-Sovereign Identity** and **Digital Identity Wallet**.

Link to the site (explaining the algorithms and methods that I have adapted to my use case): <http://brandonrose.org/clustering>

In this notebook, as in the previous one, we proceeded in small steps to reach a final stage where all these steps, combined (tokenization, stemming, stopword removal, etc.), form a complete code (see section **"Final Step: Cosine Similarity Calculation for Combined Titles and Abstracts Using TF-IDF"**). The objective of this notebook is to evaluate the similarity between the documents. To do this, we decided to verify this similarity both semantically and by the number of shared references between pairs of documents (see section **"Calculating Document Similarity Based on References"**).

To assess the similarity of documents based on shared references, we applied a first method (see section **"Calculating and Visualizing Reference Similarity Between Documents"**) that involves taking the number of common references between documents and dividing it by the total number of references in both documents. We then multiply the result by 100 to obtain a percentage of similarity. The problem with this method is that when documents have a very different number of references (for example, if document 1 has around ten references and document 2 has about fifty), the percentage of similarity can be significantly skewed.

We then turned to a second method, based on the research paper **Van2007** (in the **ResearchPaper** folder), which proposes a logarithmic formula for calculating co-reference similarity between documents. However, we encountered a problem with this method, as it produces inconsistent results. This may be due to errors in the script or incorrect application of the formula (see section **"Calculating Co-Reference Similarity Between Documents"** — this section is therefore not operational at the moment).

Finally, it is important to note that we calculate document similarity both semantically and by references to see if there is a correlation between these two types of

similarity. In the schema presented in the section "**Correlation Between Semantic and Reference-Based Similarity**", if the results tend to form a diagonal, this would indicate a correlation between these two types of similarity.

Extraction of Application Domains and Technologies:

We also have another notebook (**Extract_Domain_Technologies.ipynb**) that extracts two key pieces of information from the abstracts of research papers: the application domain and the technologies used. This is done using an AI API to process the abstracts and extract these details (see the section "**Automatic Extraction of Application Domains and Technologies from Abstracts**"). In the short term, we aim to find a correlation between the application domains and the references.

The folder **DigitalIdentityWallet_ApplicationDomain_TechnologyUsed** contains the initial results of these extraction scripts. The extraction will need to be done for each year and for both terms.