EBSI-VECTOR
Education and work reloaded

D3.5 – Specification for the reference implementation of an issuer and verifier wallet for EBSI. Identification of quick wins and fast track development to support 1st implementation phase.

Project title:	**EBSI-VECTOR** - EBSI enabled VErifiable Credentials & Trusted Organisations Registries
Grant Agreement No.	101102512 - DIGITAL-2022-DEPLOY-02-EBSI-SERVICES
Deliverable Title	D3.5: Specification for the reference implementation of an issuer and verifier wallet for EBSI. Identification of quick wins and fast track development to support 1st implementation phase
Version:	1.2
Date:	31/05/2024
Responsible Partner:	IZERTIS
Authors:	Urko Larrañaga Piedra, Cristina Martinez Hernandez, Florentín Perez Gonzalez (IZERTIS)
Contributing Partners:	IGRN,PROTOKOL,VALIDATE ID, walt.id, INAIL, Goldman, NASK, CERTSIGN, SKS, GUNET, FNMT, DANUBETECH, CERTH, Gataca, ENG
Reviewers:	Katie Phillips (PROTOKOL), Carlo Lentini (INAIL)
Dissemination Level:	PU – Public

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Document Change History

Version	Date	Author (organisation)	Description
v0.1	09/01/2024	Urko Larrañaga (IZERTIS)	Table of contents definition
v0.2	14/01/2024	Urko Larrañaga (IZERTIS)	Requirements, use cases. Enterprise Wallet specification, architecture, working flows, sequence diagrams and integrations
v0.3	17/01/2024	Florentin Perez Gonzalez (IZERTIS)	Enterprise wallet features
v0.4	22/01/2024	Cristina Martinez Hernandez (IZERTIS)	Enterprise Wallet specifications, Capabilities
V1.0	26/01/2024	Urko Larrañaga (IZERTIS)	Review and pre-final version
V1.1	31/01/2024	Urko Larrañaga (IZERTIS)	Final Version
V1.2	30/05/2024	Florentin Perez Gonzalez, Urko Larrañaga (IZERTIS)	Added technical details

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Table of Contents

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

List of Figures

List of Tables

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

List of Terms and Abbreviations

Abbreviation	Definition
API	Application Interface
DID	Decentralized Identifier
DLT	Distributed Ledger Technology
EBSI	European Blockchain Infrastructure
TAO	Trusted Authority Organization
VC	Verifiable Credential
VP	Verifiable Presentación
W3C	World Wide Web Consortium
NTH	Nice to have

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Executive Summary

This document is the result of the work carried out in Task T3.2, whose objective is to develop an "Enterprise wallet and legal entities service". The objective of the document is outlining requirements, necessary capabilities, and to present a collection of specifications for the Enterprise Wallet. From these specifications, a reference implementation of the Enterprise Wallet will be developed. To this end this deliverable constitutes the initial version of what will be a living document for the remaining duration of T3.2 activities for the implementation of an issuer and verifier wallet for EBSI. The evolution will be included in the next deliverable D3.6

In an ecosystem like EBSI's one, it is essential to guarantee a certain homogeneity among the developments carried out. Otherwise, it will be difficult to guarantee interoperability among systems. EBSI's Conformance Tests are a basis for advancing on this interoperability, however, a deep knowledge and technical effort are required for have an own implementation. Because of that, a reference implementation of the Enterprise Wallet it is a requirement so, it can be adopted by the stakeholders of the different WPs of the EBSI-VECTOR project, as well as by other additional entities.

This specifications collection has been made in an incremental way. First, an initial proposal of the capabilities and a draft architecture of the Enterprise Wallet was created. After that, we focused on collecting the business requirements of both WP4 and WP5. In addition to it, it has been collected the approach of some partners involved on the Task 3.2 on how they are building their Enterprise Wallet. Finally, it has been worked on a final architecture and specification of the Enterprise Wallet.

As a result of all this work, a set of capabilities and requirements have been identified which ones the reference Enterprise Wallet of the EBSI-VECTOR project should satisfy. Also, components that the Enterprise Wallet should include have been defined, identifying the relation among the components so, the interfaces required for the integration can be developed. On the other hand, as a result of analyzing how the Enterprise Wallet could be applied, use cases for its adoption have been identified. Based on all this work, the specification of a solution that will be easy to apply, adopt and integrate has been defined.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

However, once the Enterprise Wallet is developed and begins to be implemented by the different stakeholders, improvements may be identified due to issues not contemplated. Which, it is understood, will be resolved in the development and implementation phase of the Enterprise Wallet.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Introduction

Stakeholders that issue and verify VCs require an Enterprise Wallet for managing those operations. Although there are standards, specifications and guidelines for the development of an Enterprise Wallet, there is still a need for additional tooling or support for entities that do not have the resource or capability to utilise these. Because of that, a reference Enterprise Wallet that could be adopted, applied and integrated it necessary.

This Enterprise Wallet should be offered and it could be used in many different ways since, the requirements of the different stakeholders from both Working Packages WP4 and WP5 differ among them. Because of that, the business requirements identified on WP4 and WP5 have been taken in consideration when working in this deliverable. Also, the work done by the partners of the WP3 that have their own Enterprise Wallet have been taken in consideration.

The following document's main objective is to describe and specify the approach of the Enterprise Wallet that the EBSI VECTOR project partners will build. Based on the general context and work done, this deliverable presents a detailed specification of the Enterprise Wallet, as well as how it can be used.

The first chapter presents an analysis of the status for the development of enterprise wallets and its offering. In the chapter four, there are presented the features and requirements that the Enterprise Wallet should fulfil, as well as the use cases on how it could be applied. Chapter 5 details the specification of the Enterprise Wallet itself, outlining the capabilities, architecture and functional flow of the Enterprise Wallet. Finally, the conclusions related to the work completed in this task is outlined in the final chapter.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Current status and context

EBSI Conformance Guidelines

In order to build an EBSI-conformance and compliant Enterprise Wallet it is necessary to follow its development guidelines. These guidelines cover both verifiable credentials issuance and the verification of verifiable presentations, as well as accreditation of the network participants.

Verifiable Credentials Issuance

Verifiable Credentials issuance follows the OpenID for Verifiable Credential Issuance specifications. It distinguishes between three different flows, which intervene depending on the characteristics of the use case:

- **In-Time Flow**: A user requests a verifiable credential and it is issued in response to his/her request.
- **Deferred Flow**: A user requests a verifiable credential, but it cannot be issued and delivered at that time due to use-case related issues (e.g., it might require additional information), so the user receives an acceptance token instead. This token will be exchanged later once the verifiable credential is available.
- **Pre-Authorized Flow**: No authentication by the user is required, as he/she has already done it, prior to making the request for the issuance of a verifiable credential. Note that both in-time flow and deferred flow issuance are compatible with this approach.

Any of the three flows could be started by the action of the Issuer itself or the end user. In all the cases, however, there is a discovery phase characterized by the collection of metadata, which is essential to determine the valid formats for effective communication, as well as the supported signature algorithms.

In cases where an authentication process is necessary, EBSI does not impose a specific method, but the responsibility of choice falls again on the characteristics and requirements of the use case. However, EBSI presents two possibilities: ID token authentication and VP token authentication. While the first one is focused on proving the user's control over a specific DID, the second one allows to check a number of claims on it that the issuer can trust thanks to the

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

underlying trust framework supported by the EBSI infrastructure. Note that to request verifiable presentations an Entity needs to possess the verification capabilities required for verifiers.

The aforementioned framework of trust is based on the construction of chains of trust with a clear hierarchy, in which the members in a position as TAO place their trust in Trusted Issuers by allowing them to issue certain credentials. Consequently, an Issuer will be able to act as an issuer once it has been previously accredited by another member of the chain. Before it, the Issuer should have registered the DID that will be used, as well as its verification methods, essentially a series of cryptographic keys that the Issuer will use to perform different operations, one of them being the issuance of credentials, for which an ES256 key is required.

Verification of Verifiable Presentations

The verification process consists of requesting certain information from the user in the form of credentials, which are delivered through verifiable presentations according to the W3C specification and following the indications of OpenID for Verifiable Presentations. In general, the same verifier can support several alternative processes, each of them linked and/or identified to a specific "scope".

Note that verification can be performed as a single process, i.e., as an isolated process. Also, it can be part of another process, using the verification of verifiable credentials for authentication. For example, it would be possible to perform verifications during the credential issuance process if verifiable submissions are requested by the entity to the user for authenticating him/her. In those cases, the result of the process usually will be a code or token for accessing a previously restricted resource.

In order to achieve a successful verification of the verifiable credentials, it is necessary for the entity to act as a verifier and have access to the EBSI APIs. Thanks to the EBSI APIs the verifier could query existing information stored in the registries, as if the issuer of the credential is a participant in the network, if the issuer has been accredited to issue credentials of a specified type requested by the verifier, as well as the signature verification method registered by the issuer. It is also essential to actively check the status of the credential, i.e., whether it has been revoked or has expired. The information related to the revocation process is variable so, due to the non-fixed condition of this capability, credentials should include a field specifying how to obtain revocation information.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Issuer Accreditation

To achieve the accreditation and authorization of an issuer to operate on the EBSI, the Issuer should apply to a TAO that has been previously registered in the system. This TAO should be linked to the Issuer's use case of interest and should have been similarly accredited to act as a TAO by another TAO with at higher hierarchy or, ultimately, by the Root TAO directly. The TAO establishes the authorization and authentication requirements that the prospective Issuer must satisfy in order to obtain the accreditation credential.

Once the corresponding credentials have been obtained, the Issuer can submit them to the EBSI authorization API to obtain an access token. With this access token the Issuer will be able to interact directly with the EBSI registry RPC APIs. Note that each access token is linked to a specific set of operations, which depends on the scope requested during authorization by the issuer. In the specific case of accreditation, the credential should be registered in the Trusted Issuer Registry in the attribute reserved for the user defined in the credential received.

Solution providers

Following, a summary where is presented the current offering of the partners of EBSI Vector.

Solution Provider	Open Source	Capabilities	Components	EBSI Conformance
Izertis	YES	VC-Issuance and verification, OpenID provider configuration	Enterprise Wallet, vc-microservice and Holder Wallet	2024 Q1
Walt.ID	YES	VC issuance and verification (W3C, OIDC),	Enterprise Wallet (Issuer, Verifier, Holder), Wallet (B2C)	2024 Q1

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

		wallet, credential management, etc.		
Inail	Yes: specific components and libraries	VC-Issuance and Verification; compliance with OpenID protocols and W3C credential formats; credential encryption; compliance with GDPR eIDAS regulation; access and delegation management	Enterprise wallet reference implementation in progress; holder wallet supported by HSM	In progress
Protokol	Yes	VC issuance and verification. Credential management, Access management etc.	Reference implementation enterprise wallet & holder wallet in progress	In progress
IGrant.io	YES: Specific components and libraries	Provides both individual (android and iOS) and organisation	iOS, Android SDKs Permissions and Consents, Issuer and verifier	Q3 2023 (v3 compliance)

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

		al wallets (web/cloud based)		
Gataca	-Based on Open Specifications	ID Wallet	-	-YES, aligned with eIDASv2 specifications
Goldman	No	Accredit & Authorise Verify Issue Request & present credentials	Organization and Individual wallet. Mobile (Android, iOS), and Web versions	July 2023 (WCT V3 compliant)
ValidatedID	No	VC Issuance VC Verification VC Presentation VIDwallet	Enterprise wallet, holder wallet (mobile App), verifier,	WCTv3 compliant in Q3 2023 for the Issuer, rest is planned for Q1 2024
NASK	-	-	-	-
CERTSIGN	Yes	VC Issuance VC Verification VC Presentation	Enterprise wallet reference implementation (contribution to development and testing effort)	
SKS	-	-	-	-

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

GUNET	Yes	WebAuthN HW-RoT: Security Keys, (Remote HSM in progress) Same-device and Cross-device capabilities for VC issuance and verification flows	wwWallet (browser-based)	January 2024
FNMT	No, under consideration	VC-Issuance and verification	Issuer and verifier services	Yes
DANUBETECH	Partially	VC Issuance and Verification; DID Creation and Resolution; Enterprise Wallet	Various JDK open-source libraries; Universal Issuer/Verifier/Resolver/Registrar services	yes
CERTH	Partially	Accredit & Authorize, Verify, Issue, Request & Present	Organization Wallet, Individual Cross-Platform Mobile Wallet, Credential Issuer, Verifiable Credentials Verification Service	v3.0 compliant (Q3 2023)

Table 1: Current status of solution providers in Enterprise Wallet offering

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Izertis

Izertis is a Spanish technological company. Izertis works with many emerging technologies, building blocks for the next internet generation. In that sense, digital identity and verifiable credentials are fundamental technologis on which Izertis is working on. Izertis is working with many digital identity models and tris to publish the components publicly so, they can be used by the mainstream,

Walt.ID

Walt.id is a company building open source decentralized identity and wallet infrastructure already used by thousands of developers and organizations. The solutions are aligned with EU standards and ecosystems (EBSI, eIDAS2).

Inail

Inail, (Istituto Nazionale per gli Infortuni sul Lavoro - the National Institute for Accidents at Work), is an Italian non-profit public body that protects workers from physical injuries and occupational diseases. Its many goals include reducing accidents, protecting workers in dangerous jobs, and facilitating a return to work for people injured on the job.

There is an internal structure that has designed an Enterprise Wallet solution with the aim of supporting INAIL's commitment to the safe and effective management of insurance policies, in particular in the event of injuries to posted workers, where it is essential to be able to share traceable information and information verified with companies from other member countries. Furthermore, the use of the Enterprise Wallet and Verifying Credential paradigm guarantees, within a corporate organization, the identity and accreditation of both employees and users who access the services.

Protokol

Protokol are a specialist Web3 technology partner that builds Custom Web3 Solutions, products and dApps. Our global team collaborates with start-ups, enterprises and public administrations to help them understand and unlock the blockchain opportunity through our range of development and consulting services. Protokol are a keen participant in the EBSI VECTOR project and are in the initial stages of designing and developing an open-source reference EBSI enterprise wallet and an EBSI conformant holder wallet, to support the VC/Diploma use case.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Protokol's reference implementation of the enterprise wallet will contain features that enable VC issuance and verification. There are also features related to the management and display of credentials, user onboarding, access management and more. The wallet will be accessed by users through a user-friendly frontend web interface. Further functionality and capabilities can be added based on the requirements specified.

IGrant.io

iGrant.io is a Swedish data intermediation service provider based out of Stockholm Sweden. All the components and libraries used are based on open specifications and many are open sourced. As a data intermediation service provider iGrant.io enables digital wallets based consented data exchange supporting multiple trust anchors. The current supported trust anchors include EBSI, Hyperledger Indy and x.509 based infra. The supported credential profiles include OID4VCI/VP+SIOPv2 with JWT/SD-JWT, Anon Creds, ICAO DTC, PKPASS, and X.509 credential formats.

EBSI Compliance: Yes for both individual wallets (Data Wallets available in Playstore and Appstore) as well organisational wallets.

Gataca

Gataca is a cybersecurity company specializing in user-centric identity management technologies. Gataca's ID Wallet and enterprise solution is based on open standards and strong privacy-by-design practices. Their low-code platform is EBSI conformant, aligned with eIDASv2 specifications, and has been deployed by multiple governments and higher education institutions across Europe.

Goldman

AC Goldman Solutions & Services Ltd is an SME offering high quality services in Information Communication Technologies. Its main operating areas: Software Development, Network, Security & Infrastructure Services, and Project management. Its vision is focused on developing skills and competences capable of performing in new technologies and applications. Participated in several EU-funded (CEF) projects, including the EBSI Early Adopters program, successfully delivering one of the first six EBSI success stories – the License to Practice (Diploma) cross border Use Case – CYP/GRE.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Goldman developed a Holder (Individual) EBSI-V3 web-wallet application as well as Android and iOS mobile versions, based on TALAOs open-source implementation, supporting: -

- VC/VP jwt format;
- Protocol OIDC4VCI: pre-authorized code flow, user pin, authorization code flow, deferred endpoint;
- Protocol OIDC4VP and OIDC, presentation definition, presentation submission.
- Support of did:key, did:ebsi.

Also developed an organization wallet for issuers and verifiers based on NodeJs. Both Holder and Organization wallets have passed all EBSI conformance tests (V3).

ValidatedID

Validated ID (VID) is a Spanish Qualified Trust Service Provider (QTSP) with the headquarters in Barcelona. VID provides besides their state-of-the art and legal compliant qualified electronic signature services, qualified electronic timestamping service, e-invoicing service, a full SSI technology stack consisting of components and tool for all actors within an SSI ecosystem such as an issuer and verifier service as well as an enterprise and holder wallet. Besides aiming for legal compliance with relevant laws, VID aims to be fully interoperable and follows the eIDAS 2.0 developments. The SSI stack so-called VIDchain, implements the OpenID connect for verifiable credentials standards family such as OIDC4CI, OIDC4VP, and OIDC SIOPv2.

VID is currently working on getting all components compliant with EBSI's WCT v3. Besides the EBSI related activities, VID currently implements also an QEAA issuing and verification service, that is mentioned in eIDAS 2.0.

CERTSIGN

certSIGN is the most important Romanian provider of trust services (QTSP), and a pioneer on the Romanian market by introducing the electronic identification means and paperless technologies in the business processes. certSIGN is a producer of proprietary software solutions, cyber security, digital identity and archiving solutions successfully implemented in 20 countries and, the main developer and provider of cryptographic solutions and trust services in Romania in compliance with the requirements of the eIDAS Regulation.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

certSIGN is the only provider in Romania - through the “Paperless” system of qualified remote electronic signature and remote electronic seal - which has obtained the certification Qualified Signature and Seal Creation Device (QSCD), the list of devices approved by the European Commission, the only reference at European level and, the provider of “certME Identity Wallet”, the first Romanian Electronic Identification Means in accordance with the eIDAS Regulation.

At this moment, our identity wallet solution is not EBSI compliant. We will get involved in the development and quality assurance testing activities related to the EBSI wallet enterprise.

SKS

Skaitos kompiuterių servisas is an IT service provider focusing on innovative solutions for the market and is an active social partner for the Lithuanian government. With the support of the national EBP representative and CEF funding implemented the EBSI node and organized multiple workshops in Lithuania to promote and explain the EBSI ecosystem for public institutions.

Despite the fact that SKS don't have their own enterprise wallet, it will contribute to this task for the specifications improvements.

GUNET

GUnet (Greek Universities Network) is a non-profit company with members from all 25 public universities in Greece. In its role as a central service and platform provider, GUnet’s solutions portfolio includes Qualified Trust Services for e-Signing and e-Sealing (QTSP), and Identity Management Services that operate in 20 out of 25 Greek Universities. GUnet has been participating in the EBSI Early Adopters Program since 2021, showcasing cross-border scenarios in the education domain.

Following the release of the EBSI conformant eDiplomas.gr Wallet for individuals and organisations, in Q3 2023, GUnet has shifted its focus to the development of wwWallet.org, in partnership with SUNET and Yubico. This aims to deliver an open-source implementation of a non-custodial yet web-wallet solution, utilizing Progressive Web App (PWA) technologies to provide an installable app-like experience with offline capabilities. It leverages FIDO2 security keys and hardware-based RoT to ensure a high Level of Assurance (LoA) in authentication and prioritize privacy.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

FNMT

FNMT is a Spanish public corporation and instrumental resource of the General Government Administration and acting as a true public service. It produces identification documents (e.g. driving licenses, passports, national ID documents, etc.) and offers a wide range of digital services, acts as a Qualified Trust Service Provider and is in charge of the securitization of electronic communications with the public administration. It is undertaking the design and development of verifiable credential issuance and verification services and will act as the solution's PID provider."

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Requirements

Enterprise Wallet features

The following features are the result of the collaborative work of the partners of work packages 3, 4 and 5, with the aim of aligning the Enterprise Wallet with stakeholder expectations. However, it should be mentioned that in WP5 are working on the identification of those requirements so

Feature class	Feature	Required (YES / NTH / NO)	Notes
User admin	User management	NO	No required at Enterprise level. However, Enterprise users should log in to access the platform.
	User roles / access control	NO	
	Key storage and management (kms)	NO	
Holder identification	Identification thorough existing legacy systems	NTH	
	Identification thorough a VC presentation = VP	NO	
Issuance	Generation of "Credential offering requests" - VCI request process will be started by the User	YES	
	Issue diplomas, certificates, transcripts, micro-credentials	YES	
	Issuance VC with a Holder ("requesting a single credential for multiple DIDs")	NO	
	Compliance with EBSI W3C VC and VP	YES	

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

	Customizable credential-Schemas	YES	Capacity to be used with different credentials so, it is no linked to any use case
	Support for various credential types	YES	
	Full portability of credentials with different wallets	YES	It is a requirement that should be fulfilled by the Holder Wallets
	Issue credentials in bulk	YES	
	Multi-signature and multi-issuer credential management	YES	
Management	List issued credentials	YES	
	View details on an issued credential	YES	
Secure credential management & SSI Storage	BC integration for verification and immutability	YES	
	Credential encryption for enhanced security	YES	
	Automatic timestamping eIDAS compliant through blockchain	YES	
	Full integration with TAO/TAR for issuer verification and trust management	YES	
Credential revocation	Non-business-driven revocation system but agnostic revocation capability that can be plugged in different business processes to match different MS requirements.	YES	
	Revoke issued credential (enterprise/university revokes)	NO	

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

	Modify issued credential status	NO	Related to revocation and EBSI planification
Verify	Generation of "Presentation offering requests"	NO	
	Machine-based real-time verification	YES	
	Verification revocation (user can stop verification from selected parties)	YES	
	EBSI Verification process: schema, signature by Issuer (verify if Issuer is authorized...)	NO	
	Ability to accept/verify selective disclosure	NTH	
	Ability to verify that someone is a student at a particular institution	NO	
Data privacy	GDPR	YES	
	User consent management	YES	
	Permanent data export in open formats and deletion options	YES	
Interoperability and Standards Compliance	Interoperability - Support for widely accepted standards like EBSI W3C Verifiable Credentials.	YES	
	Interoperability - Compatibility with other digital credential systems.	YES	
	Interoperability - Open format	YES	
	Integration APIs - educational institutions' systems and HR platforms	YES	
	SSO login	NTH	
	Full regulation compliance with EU regulations	YES	

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Other	Notification and alert feature to alert issuers/verifiers of actions that require attention	YES	
	Feature for users to be able to submit feedback	YES	

Table 2: Enterprise Wallet features

Use cases

According to the proposal that is presented below, the Entities will have the chance to use the Enterprise Wallet in three different ways:

- Isolated Enterprise Wallet: On the one hand, Entities will be able to use Enterprise Wallet without doing any integration with their current legacy systems. For that, an Enterprise Wallet that combines a web portal and a Backoffice will be provided so, the entities can start up their own instance of an Enterprise Wallet without requiring any modification or integrations. In this case, as it has not been done any integration, the data required to complete a verifiable credential would be introduced manually.
- Integrated Enterprise Wallet: On the other hand, the entities will be able to integrate the Enterprise Wallet capabilities and functionalities with their systems so, the Holders could use a familiar and known system to interact with the entity. In this case, some integrations should be done. The interfaces required for the integration there are explained in the "Integrations" chapter.
- Verifiable credentials Microservice: Finally, the entities could have the chance to develop their own systems and use only the verifiable credentials microservice, this microservice will be a stateless microservice.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Enterprise Wallet specification

The following section presents the specification of the Enterprise Wallet itself. First of all, here are presented the capabilities that the Enterprise Wallet will fulfil. Then the architecture of the Enterprise Wallet is described. After that, a sequence diagram is provided to explain how the Enterprise Wallet will be used. Finally, as the result of the previous content, will be described the main interfaces that the Enterprise Wallet will have for its usage and the integration with legacy systems.

Capabilities

The enterprise Wallet developed under EBSI VECTOR project should be include the following capabilities:

- Enterprise Wallet must be conformant with EBSI services.

  Must be approved by EBSI, based on conformances process, detailed on EBSI website. https://hub.ebsi.eu/wallet-conformance . 'Accredit and Authorize tests'; 'Issue to Holder tests'; 'Verify VC'. In case, verifying VC will depend on the use cases.

  The credentials must remain EBSI conformant even if they are issued by different entities or organizations.

- Legal Entity Accreditation

  Enterprise Wallet must be conformant to https://hub.ebsi.eu/wallet-conformance/accredit-authorise .

  The work developed should be linked with the work done in T3.3. Besides, the Enterprise Wallet should include the different levels of accreditations into the eIDAS2 context. In this way, we will simplify the process, since due to the participation of so many countries in the project and universities we could develop too complex interactions.

- DID for Legal Entity Management

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Must conform to https://hub.ebsi.eu/vc-framework/did/did-methods/legal-entities: This capability probably Might not be inside Enterprise Wallet implementation. It is probably a one-time action.

1. Authorization
2. DID creation
3. DID resolution
4. DID document lifecycle
5. DID authority

The Enterprise Wallet will use an open source Key Management System

- Enterprise Wallet Administration Console

  The admin console will manage, store, issue and verify received Verifiable Credentials/Presentations..

  The administration panel to access the enterprise Wallet, will be accessed via username and password of each of the different organizations and universities. Having been registered by the technical administrator previously. On the other hand, the user's Holder Wallet will have different authentication methods out of scope in this document.

  The Enterprise Wallet will provide the ability to issue credentials that have not been issued correctly or to issue them repeatedly to the same user, for example, if a user has lost his wallet. This is because it will be possible to issue and manage these credentials also from the enterprise wallet for the different organizations participating in the project.

- Data Management, storage and key management system

  The enterprise Wallet must provide a secure storage in order to store the data whenever required.

- Verifiable Credential Management

  Within the VC management, the ability to revoke VCs by the organizations should be included. For the first version of the MVP, this functionality will not be included, because it requires an additional research and development effort not contemplated at this early stage of the project.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

The data model will take into account the WP4 and WP5 use cases. I.e. Diploma and Social security schema.

High-level Architecture

In the following chapter there is detailed the architecture of the Enterprise Wallet. The Enterprise Wallet will be composed of various components so, below there are listed those components and the relation among them.

The following diagram shows the components that make up the Enterprise Wallet organized and presented by layers. We have two main layers: the first one, the fronted layer, where the different user interfaces that will be available are presented; secondly, there are presented the components that will be on the backend. Finally, in the picture there is presented EBSI at the Distributed System layer.

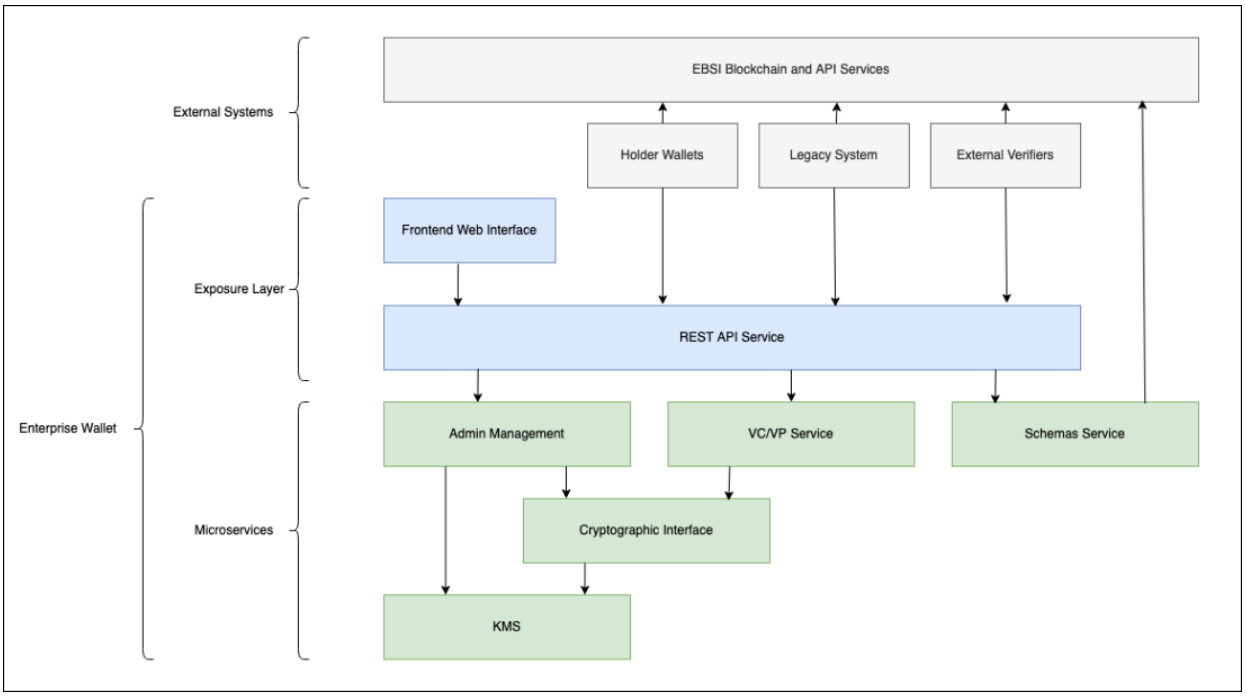

Figure 1: Enterprise Wallet Architecture

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Following, there is described each component, as well as the interaction and relation that its component will have with the rest ones:

Frontend layer: all the component of this layer will communicate with the rest of the components through the API Rest provided by the Backoffice.

- **Administration dashboard**: an administrative platform for the management of the required information for the management of the enterprise wallet. If any integration is done, they could be configured through this administrative platform.
- ***Entity's legacy Web***: this is the Web page or User Interface provided by the Entities. It is not a component of the Enterprise Wallet itself. However, it is shown in the picture, to confirm that if the entities would like to use their existing systems, they could use them. The interaction between the users and the entities could be done through this component.
- **Enterprise Wallet Web**: this component would be a User Interface provided directly by the Enterprise Wallet so it can be used without any integrations. The interaction between the users and the entities could be done through this component.

Backend layer

- **Rest API interface**: to facilitate the usage of the services and functionalities of the enterprise wallet. Moreover, if any integration is done, the communication among the legacy systems and the enterprise wallet will be managed through this component.
- **Backoffice / Middleware**: the core of the enterprise wallet where all the functionalities and orchestration among the components will be done.
- ***Entity's legacy service and data store***: these are the existing components of the entities.
- **Verifiable credentials microservice**: it is a stateless microservice for the management of the main operations related to verifiable credentials management.
- **Database**: a database will be deployed for the storage of the data, as the issued verifiable credentials.
- **KMS**: a key management system will be integrated in the Enterprise Wallet to store securely the cryptographic keys.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

For the issuance of a VC there is required to get the required data from the user to complete the VC. Following we describe the issue and approach related to data retrieving for completing the VC.

Data retrieving for completing the VC

There are two ways to do it:

- The first one considers manual data entry, thus includes a form to entities so, they can introduce metadata of the VC that they want to issue.
    - In this case, the Enterprise Wallet should issue the VC once it has received the request of a VCI from a user. With that request, the Entity will complete the form. Due to it, Enterprise Wallet needs to be able to manage the requests received from users.
- The second one require entities to have a data repository that can be invoked from the Enterprise Wallet, in order to load the data necessary to prepare the VC. In the case of the "Integrated Enterprise Wallet" there will be required an integration with existing legacy system for data retrieving. Data must be structured based on a predefined model/schema so to avoid mismatch when loading fields and preparing the VC.

Verifiable credentials issuance working flow

This section is presenting the approach that will fulfil the Enterprise Wallet for the verifiable credentials' issuance. The request of the issuance of a Verifiable Credential could be initiated by the User / Holder or the Issuer. In spite of it, the approach presented below is based on a process started by the User - this would be the approach of EBSI Vector at the current phase.

However, first of all, it would be decided how to manage the User identification. Following, there is presented the issue and scenarios regarding to user identification.

Identification of the user / holder

Issue: How is identified the user? How we know to whom we are going to issue the VC?

- Ways to manage that identification:

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

  - 1) The Issuer could require to the user to present a VC, what it is a VP itself.
  - 2) Having a user authentication/management system through which the user authenticates and then requests the VC.
  - 3) Identifying the user in person, what it is a phase-to-face identification process.
- If user identification is NOT required, the question to answer is the following one: how do we know to whom we issue the VC? Although there is a flow data allows it, at the EBSI VECTOR project we understand that users will be identified

Approach: Identification of the user– in one of these two ways

- (Online) The request for the issuance of a VC will start once the user has been authenticated / logged in to the corresponding system. The user would start up that process thorough (1) scanning a QR or (2) clicking on a link at the Entity's page.
- (Offline – face-to-face) The request of the issuance of a VC will start once the user has been authenticated in person, the, a person from the Enterprise would show to the user a QR to scan.

In both cases, the control to activate the "VC Issuance request" process would be in the hands of the Enterprise. The User / Holder will only be able to request the Issuance of a VC after authenticated.

Sequence diagram

According to the approach presented below, in the following picture it is shown a Sequence Diagram on how will work for the "Integrated Enterprise Wallet" use case for the Issuance of a Verifiable Credential.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

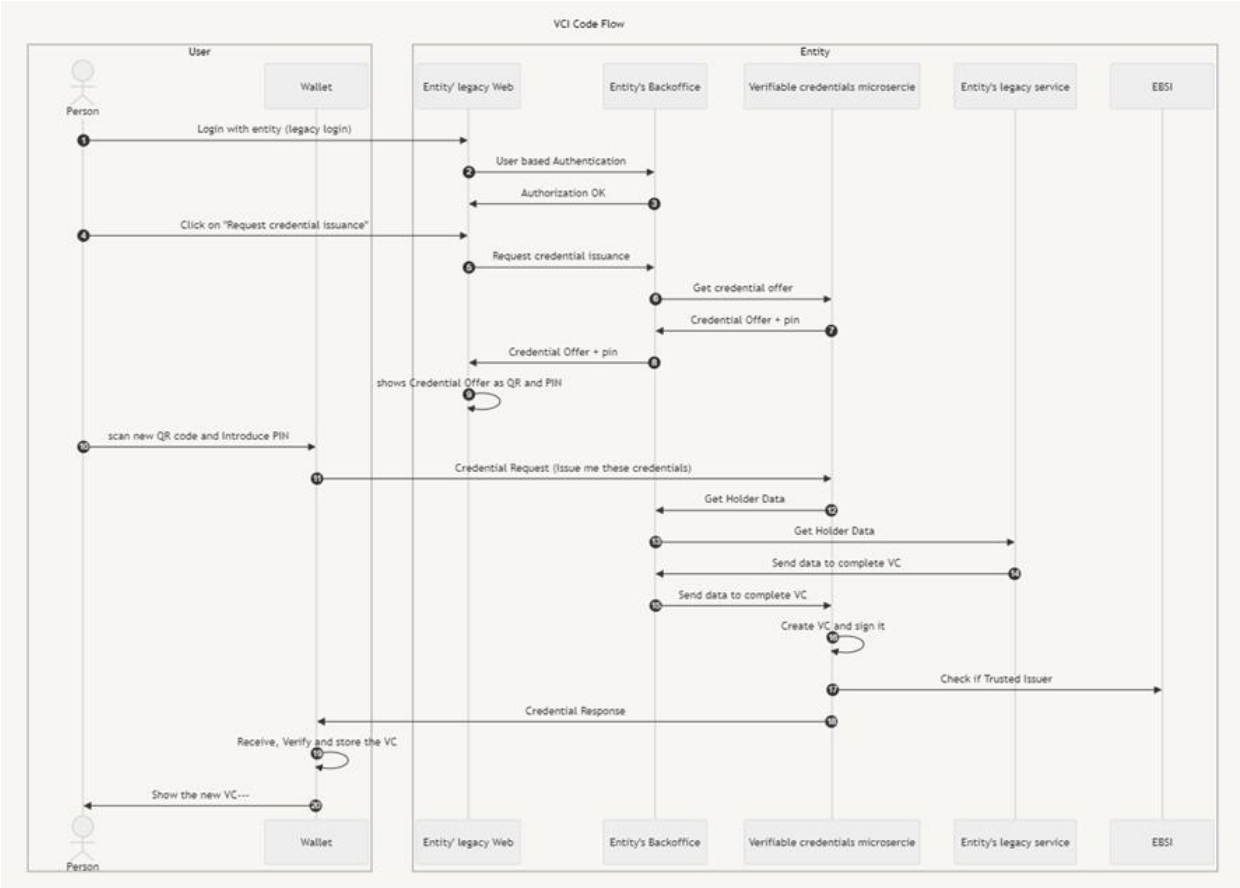

Figure 2: VCI Sequence diagram

Components' specification

As mentioned above, the Enterprise Wallet is divided into two functional blocks:

- On the one hand, we can define the Backoffice that acts as the central component of the solution and is fundamentally responsible for defining the issuance and verification flows that we want to support. In other words, this is the component that allows users to define which verifiable credentials they want to issue or verify, as well as the authorization requirements underlying these processes, if any.
- On the other hand, we have the verifiable credentials microservice, which is a stateless service that implements OID4VCI and OID4VP, consequently being responsible for the generation of VCs and their verification when they are presented in a VP. The service is

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

not designed to be used independently, since it requires its integration with a component that knows the characteristics of the issuance and verification processes underlying the use case that is to be supported, with the previously mentioned BackOffice being a possible candidate.

BackOffice

Built using the Django Framework, it is a component that provides an administration interface (Administration Dashboard) to users through which they can define the credential issuance processes they want to support, as well as the verification processes. The operations that can be performed through the Administration Dashboard provided by the Backoffice are listed below:

Define emission flow

The definition of an issuance flow requires that users indicate any issue related to it, with the exception of user data that will be included in the credential itself. This is, mainly, the following issues:

- The credential scheme.
- The most specific type of credential.
- Indicate if it is a delayed issuance credential.
- The type of authorization that will be requested from the user, the possibilities being ID Token and VP Token.

Additionally, it is also possible to define additional optional issues, such as the life time/validity of the VCs to be issued and their support for revocations.

- Scheme and type of the credential.
    - The scheme to be indicated corresponds to the URL address where it can be retrieved and which should correspond to one hosted by the EBSI Trusted Schemas Registry. For its part, the type of the VC corresponds to its specific type, that is, the one that most defines it at the user data level. All resulting credentials will include that type along with VerifiableAttestation and VerifiableCredential.
- Deferred credential issuance flow
    - A deferred credential corresponds to one that, even assuming correct operation, cannot be delivered to the user at the same moment in which the user has requested it due to issues usually related to the logic and characteristics of the

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

underlying use case. For example, it could be the case that the data to be included in the user's VC is not yet available or that the process requires some additional verification and/or validation that must be carried out by an external agent. In these cases, the user instead receives an acceptance token that can later be exchanged for the VC once it is ready. The BackOffice allows you to specify, therefore, whether this flow is the one that should be used to issue a VC. It is important to highlight, however, that the BackOffice is not responsible for the management of the acceptance token or its availability due to its lack of knowledge of the underlying use case, making it necessary to integrate the solution with an alternative or legacy system that is knowledgeable. and manager of this information (more information in the Integrations section).

- Type of authorization required.
    - Credential issuance processes require that the user requesting them be authorized and authenticated in the system using methods that depend on the characteristics of the business logic. Thus, in some cases, it may be sufficient for the user to prove that they are the controller of a DID, while, in others, they may be required to present certain credentials. From an authorization point of view, this corresponds to which token is going to be requested from the Holder, the possibilities being ID Token and VP Token. The first allows proving that the user is the owner of a certain DID by requesting a signed token with a verification method associated with it. For its part, a VP Token extends the above by additionally including a series of VCs that meet certain requirements presented in the prior authorization request. These requirements are defined in a data structure known as Presentation Definition and allow you to specify everything from the format of the VCs that the user must deliver (JWT or JLD, although only the first is currently supported) to the claims that they should have. In order to request a VP Token from the user, it is necessary to have previously defined at least one Presentation Definition, which can be done through the BackOffice. With respect to the types of authorization, it is important to highlight that the BackOffice is not responsible for resolving the OpenID processes with which they are related, this being the responsibility of the verifiable credentials service. The BackOffice, however, is responsible for managing this information and, through the integration of both

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

components, indicating it to it during its execution so that it can successfully carry out its task.

- Credentials with expiration time
    - If desired, an expiration time, in seconds, can be associated with each verifiable credential that is issued. This allows short-lived credentials to be represented, that is, credentials that act similarly to how an access token does; or other more complex ones whose validity is longer but still require periodic renewal, such as a personal identifier.
    - Failure to specify the expiration time will be interpreted as meaning that the credential to be generated will not expire over time, and will always be valid unless it is manually revoked.
- Revocation of issued credentials
    - In the definition of an issuance flow it is possible to indicate the data structure that will be used to manage revocations of issued credentials. If not specified, it is considered that said credentials cannot be revoked. Currently, there is only one option available, this being StatusList2021.
    - If selected, the system will internally create a 16 KB (131072 bits) binary list when the first VC with revocation support is issued and will relate the value of one of the available bits to the status of the credential, being a value of 1 that this has been revoked. The assignment of bits is done sequentially and when the list is completed a new one is generated. The generated VCs will include the "credentialStatus" field with the URL of the endpoint that a verifier must consult to obtain the status credentials associated with the original VCs, as well as the assigned index within the list. The state credentials, for their part, inform the verifier of the entire list, allowing them to then obtain, by knowing the associated index, the state of the VC they are verifying. The endpoint indicated in the original VC corresponds to one managed by BackOffice itself, without using EBSI as an intermediary or proxy, consequently the chosen strategy is one of direct communication between the verifier and the issuer. The latter is not able to know during the process which VC is being verified of all those related to the requested StatusList, but it does discover the IP used by the verifier.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

- To revoke a credential, the administration interface provides a screen in which the identifier of all the VCs that have been generated is recorded, along with their type. The user can interact with these identifiers to check the associated status in the list and, if desired, modify it to revoke the VC. Please note that revocation is not reversible. It is important to point out that, as indicated, only the ID of the generated VCs is stored and at no time all of them.

Issuer metadata

From the BackOffice it is possible to define some of the issuer's metadata, although the majority is self-defined by the verifiable credentials service, and it is only possible to indicate the endpoints that intervene in the OID4VCI process, which are communicated to a Holder Wallet during the discovery phase. That is, the endpoints that intervene in the authorization and authentication process, as well as the one that can be used to request a verifiable credential or exchange an acceptance token for a deferred credential.

Credential Offer

Once one or more issuance processes have been defined, the system is responsible for self-managing the corresponding Credential Offers, making user intervention unnecessary in this regard. In order to request a VC, the BackOffice offers an endpoint through which you can obtain a QR that internally has a URI with which you can obtain the Credential Offer necessary to begin, from the perspective of a Holder Wallet, the request for a VC.

Obtaining the QR through an endpoint allows it to be easily integrated into the applications with the business logic, making the interaction and integration more organic.

Pre-Authorized Issuance Flow

The BackOffice does not present any option or form to specify whether a VC should be issued following the pre-authorized flow or not. The reason is because this flow is evaluated at runtime, considering it possible for any VC flow. The Enterprise Wallet as a whole is not responsible for the management of pre-authorized codes nor does it know the underlying logic that generates

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

them. Instead, when it receives one, it communicates it with an external system with which it must have been integrated and which has the responsibility of determining its validity. This abstraction allows the code to follow any format and not influence the execution of the Enterpirse Wallet. If at the business logic level the pre-authorized flow is not possible, then it is enough for the integration to reject any code that is delivered to it.

For its part, a Holder Wallet has two ways of knowing the code to use. Firstly, this may have been communicated by methods external to the Enterprise Wallet and, secondly, you can obtain it directly from the Credential Offer. The latter is possible because the endpoint that generates the corresponding QR accepts an optional parameter that allows indicating the pre-authorized code to be included.

Verification of Verifiable Presentations

The verification of verifiable presentations can be integrated into a process of issuing verifiable credentials or independently. The first case occurs when the Holder is requested to deliver a VP Token either in the authorization or authentication phase. When verification occurs independently, it is usually because it is desired to verify certain information about the Holder before granting access to a protected resource, in other words, it occurs as an authorization process for resources that do not necessarily correspond to verifiable credentials.

The verification process is highly dependent on the underlying business logic and requires, consequently, that the different sets of credentials that will be requested from the user be defined in the form of a Presentation Definition. As mentioned in section 1.1.1.3, BackOffice offers a visual interface through which these can be specified. Once at least one is defined, the user can indicate, in a manner analogous to what occurs with the issuance process, the characteristics of the verification process when it occurs independently. However, the number of characteristics to indicate is considerably smaller, it being only necessary to indicate the definition to be used (the same one that will be delivered to the holder) and the associated “scope”, this acting as a unique identifier of the process and which must be included in the answers that include the VP Token by the Holder Wallet.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Verifiable credentials service

The verifiable credentials service implements OID4VCI and OID4VP, being consequently the component responsible for the Enterprise Wallet in managing the authorization and authentication processes, in addition to generating the VCs that will be delivered to the Holders and verifying the credentials delivered by them. in verification processes.

General characteristics.

- The authorization and authentication processes only support ES256 as a cryptographic algorithm.
- The credentials issued are in JWT format and signed using an ES256 key.
- It uses a fixed ES256 key pair that is defined at service startup time.
- It uses a specific EBSI DID or DID KEY that is defined at service startup time.
- It is a "stateless" service, it lacks access to any database.
- It requires integration with other systems to recover information associated with the business logic (VC data) such as determining how to manage its communication with an entity during an OpenID flow.

"Stateless" nature of the service.

Although the service is stateless as such and does not manage any database, in practice it needs to retrieve information about the activity it is performing to carry it out correctly. For example, when issuing a VC, you need to determine its outline and type among other issues. Also, and along the same lines, during an authorization process it is necessary to know what type of additional authorization request will be requested from the user to generate an authorization code. Because the service does not store this information, another system must do so, requiring its integration with it in order to operate. In the specific case of the Enterprise Wallet, this system corresponds to the BackOffice. This makes an endpoint available to the service through which it can retrieve the information defined for the issuance and verification flows.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Regarding the data related to the subjects of the credentials, these are unknown to the Enterprise Wallet, making it imperative to integrate the solution with an independent external system to obtain it (more information in the integrations section).

Nonce management.

During a process of issuing credentials or verifying presentations with OpenID, it is necessary to actively use nonces to link the actions of the Holder, avoid repetition attacks and guarantee the Holder's ownership of the cryptographic material. These nonces are managed internally by the service, responsible for keeping track of their use by potential Holders. However, this information, due to the stateless nature of the service, is not stored in the service itself. Instead, an additional model is defined in the BackOffice, which we can call Nonce Manager, which is responsible for its storage, acting for practical purposes as a shared remote storage.

This component provides a basic REST API that allows you to perform CRUD operations against nonces. The service stores, along with each nonce, the DID of the entity for which it was generated. In addition, it also associates a state that is actively used by the service to "remember" information between the different phases in which a nonce is used, as well as to determine if it has been used previously. In more concrete terms, the associated state corresponds to an array of strings where each index, except the first, depends on the phase of the issuance or verification process in which the service is. The first index, for its part, corresponds to an opcode that identifies the phase itself, allowing the meaning of the following to be easily deduced. A total of three opcodes are distinguished:

- A first opcode that is used to identify a nonce that has been generated in the authorization phase and that is expected to be included in the ID Tokens and VP Tokens that must subsequently be delivered by the Holder Wallet.
- A second opcode that identifies that a user has already passed the authorization phase and obtained an authorization code.
- A third opcode indicating that a user has successfully obtained an access token.

Update operations are used to modify the state associated with a nonce within the manager, while delete operations only act when the issue or verification flow has finished. The latter implies that any data structure with an associated nonce becomes invalid after this point.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Additionally, all registered nonces have an associated TTL that, upon expiration, causes its deletion.

This Nonce Manager component can be constituted as its own entity in the solution design or can be integrated into another existing one. In the case in question, this responsibility has been attributed to the BackOffice.

REST API interface of the service.

The service offers a REST API interface that the Holder Wallets can use to interact with it, whether for the authorization, authentication processes or the issuance of the credential itself. The service's REST API follows the OID4VCI and OID4VP guidelines as well as the EBSI implementation guides and does not define any additional methods that are not included in them. Thus, the following is distinguished:

- A base authorization endpoint with which users can begin the OID4VCI or OID4VP process.
- An endpoint in which to deliver authorization responses to requests that the service may request to request additional information from the user, this being ID Tokens and VP Tokens.
- An endpoint for the authentication process that is responsible for the generation of access tokens and that supports as “grant types” an authorization code generated in the phase of the same name and by the service itself or a pre-authorization code.
- An endpoint to request verifiable credentials and that requires an access token generated in the authentication phase.
- An endpoint to exchange an acceptance token for a credential in a deferred issuance flow.

Access tokens and authorization codes

The authorization codes, like the access tokens, correspond to JWTs signed with the service's own cryptographic material. As long as this material is not exposed, this allows him to easily verify that both tokens have been generated by himself. To avoid replay attacks, the nonce associated with the DID that appears in the audience field is included in each token, which corresponds for practical purposes to the DID of the entity that is undergoing the authorization or authentication

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

process. As indicated in section 1.2.3, each nonce has an associated state and a TTL that together prevent them from being used more than once or after a period of time.

Finally, and due to the stateless nature of the service, it takes advantage of tokens to include certain information related to the process and that is not related to user data itself, such as the "scope" that is being used in the case of the authorization token and, in the case of access tokens, the type of credential that is allowed to be issued with that token.

Relationship between BackOffice and verifiable credentials service

The service is designed to be able to handle Holder Wallet requests directly if desired. However, in the case of the Enterprise Wallet, the service is not exposed to the outside and instead all interactions with the Holder Wallet are managed by the BackOffice, which implements the same REST API interface as the service but without underlying knowledge of OpenID, merely acting as a reverse proxy.

Additionally, and as previously mentioned in this document, the BackOffice acts as a database manager and provides interfaces that allow the service to consult the information it needs, as well as register and manage the nonces it generates during its activity.

Integrations between the Enterprise Wallet and Legacy systems

The Enterprise Wallet is agnostic about the use case and the business logic associated with it, making it necessary to integrate it with external systems to obtain or deliver certain information. In the following chapter there are explained the endpoint required for the integration of the Enterprise Wallet and the legacy systems.

The integrations will require some interfaces, some of them should be provided by the legacy systems and others by the Enterprise Wallet. Next, there are detailed the necessary integrations that should be carried out.

Required services of the Enterprise Wallet

There will be required a service for getting the Credential Offer that should be shown to the User.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

- Consumer of the endpoint: Entity's legacy Web.
- Endpoint offered by the "Enterprise Wallet".

Required services of the legacy systems

Following, there are explained 4 services required by the Enterprise Wallet.

- Consumer of the endpoint: Enterprise Wallet.
- Endpoint offered by the "Entity's legacy service and data store".

Obtaining data from the subjects of the credentials

At the time the VC is generated, the service launches a query to an external service that allows it to retrieve the data to be included in the credential. This service is given the type of credential that will be generated and the DID used by the Holder. The external service must respond with the corresponding data or reject the operation, the latter if it is determined, for example, that it lacks data for said user or for another reason.

As there are numerous integrations and to avoid indicating multiple URLs, the service associates the same URL with all the services with which integration is required, varying the "path" of the query as appropriate. In order to isolate the service as much as possible, in practice this URL is assigned to BackOffice itself, which acts in this case as a proxy when contacting external services and retrieving information for the verifiable credentials service.

Obtaining and managing the deferred code

When a credential is going to be issued following the deferred flow, it is necessary that it be made known to external services so that they can prepare the data. Likewise, the responsibility of assigning an internal identifier to said task and indicating which acceptance code should be provided to the Holder Wallet so that it can redeem the VC later is relegated to them. In this way, a first integration is necessary in which the service, when it is going to generate an acceptance token, communicates to the external services the DID of the Holder and the type of VC requested and these, instead of responding with the data from the VC itself, respond with any acceptance

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

code. The service does not return this code directly to the Holder, but rather wraps it in a JWT with certain data from the original request, such as the type of the requested VC.

When the Holder Wallet tries to exchange the token for its credential, the service will extract the acceptance code issued by the external services from it and consult its status with them through a second integration. If the VC is already available, they must respond with its data directly, so that the service can build it and deliver it to the Holder Wallet. If, on the other hand, it is not yet available, they must respond with an acceptance code again, which may be new or the same as the one previously provided. There is also the possibility that a code that has already been redeemed previously will be delivered. In this case, it is the responsibility of the external services, as handlers of said identifier, to detect such a situation and reject the operation.

Pre-authorized code management

Pre-authorized codes are generated by alternative authorization or legacy systems that the business application has and that are unknown to the Enterprise Wallet, for example, they could be a consequence of a classic login with username and password.

When one of these codes is delivered in the authentication phase, the service delivers it to an external service in charge of determining its validity. If so, it communicates this to the Enterprise Wallet and proceeds to generate an access token to the Holder Wallet.

Validation and delivery of data extracted from the VPs

When a VP is delivered, it contains credentials that are checked for validity by the verifiable credentials service. However, the latter can only validate its signature, the correlation between the “claims” delivered and those that were requested, its status and the validity of the accreditation of the corresponding issuers; In other words, it cannot check whether the data makes sense from a business logic point of view. Thus, when the service finishes verifying a VP, it delivers the data extracted from it (which depends on what was requested in the definition of the presentation that was delivered to the Holder Wallet) along with the Holder's DID to an external service that must perform a final validation on said data. In case of success, the validation of the delivered VP will be considered valid by the verifiable credentials service.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Conclusions

This document presents a summary of the work undertaken in task 3.2 over the past few months. The main conclusion of the work done is that the Enterprise Wallet adoption and application would be offered in many ways, as an isolated component, a microservice or an integrated system. The stakeholders could use the Enterprise Wallet as an independent component or as a microservice, integrating it or not.

Because of that, what it is presented is an architecture and the description of each component so, it can facilitate the adoption and application of it to the stakeholders. Also, there are mentioned the main features and capabilities that the Enterprise Wallet will fulfil. This deliverable constitutes the initial version of what will be a living document for the remaining duration of T3.2 activities for the implementation of an issuer and verifier wallet for EBSI. The evolution will be included in the next deliverable D3.6.

Project co-funded by the European Union under the Digital Europe Programme. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.