Bridging Erasmus Without Paper and Self-Sovereign Identity: A Verifiable Credential Issuer for Academic Mobility

João Carreira
University of Leiria and Oeste
Leiria, Portugal
joao.r.carreira@ipleiria.pt

Lourenço Ferreira
University of Leiria and Oeste
Leiria, Portugal
lourenco.ferreira@ipleiria.pt

Carlos Machado Antunes
University of Leiria and Oeste
Leiria, Portugal
carlos.machado@ipleiria.pt

Marisa Maximiano
University of Leiria and Oeste
CIIC
Leiria, Portugal
marisa.maximiano@ipleiria.pt

Ricardo Gomes
University of Leiria and Oeste
Leiria, Portugal
ricardo.p.gomes@ipleiria.pt

Vítor Távora
University of Leiria and Oeste
Leiria, Portugal
vitor.tavora@ipleiria.pt

***Abstract*—The Erasmus Without Paper (EWP) initiative facilitates the digital exchange of student mobility data between higher education institutions across Europe. However, these records remain under institutional custody: students have no native mechanism to hold, carry, or present their own academic records (Transcript of Records) without routing through institutional services. This paper presents a verifiable credential issuer that extends the lifecycle of an academic document into the Self-Sovereign Identity (SSI) ecosystem. The system ingests a Transcript of Records in the EWP-compatible European Learner Mobility Objects (ELMO) Extensible Markup Language (XML) format, preserves an institutional review step, and issues a signed digital Verifiable Credential to the student through the OpenID for Verifiable Credential Issuance (OID4VCI) protocol. Two issuance profiles are supported, a standard OID4VCI producing Selective Disclosure for JSON Web Tokens (SD-JWT) credentials, and an European Blockchain Services Infrastructure (EBSI)-oriented profile that produces W3C Verifiable Credentials in JWT format. This distinction reflects the heterogeneous interpretation of metadata and credential formats across real-world wallets. The proposed prototype also provides public verification and StatusList2021-based revocation. It was evaluated using 22 XML fixtures in automated tests and end-to-end interoperability executions involving two wallet implementations. Results show that the pipeline is technically feasible and that serving multiple profiles from a single transcript accommodates the current fragmentation of the SSI wallet ecosystem without forcing the issuer to choose a single credential format.**

***Index Terms*—Erasmus Without Paper, Self-Sovereign Identity, Verifiable Credentials, OID4VCI, EBSI, Academic Mobility**

I. Introduction

In 2024, Erasmus+ provided mobility opportunities to almost 1.5 million participants across education, training, youth, and sport across more than 30 countries [1]. Traditionally, each mobility required the exchange of multiple paper-based documents such as learning agreements, transcripts of records and arrival confirmations between sending and receiving institutions. The Erasmus Without Paper (EWP) initiative [2] addressed this by building a decentralized peer-to-peer network of Application Programming Interfaces (APIs) enabling higher education institutions to exchange structured academic data digitally, eliminating paper-based workflow.

EWP solves the system-to-system interoperability problem. For example, a Transcript of Records issued by a host university can be transmitted through the EWP network and received by the home institution in a structured XML format based on the Electronic Management of Records Exchange (EMREX) ELMO schema [3]. However, EWP does not address the student-to-system portability problem. Once the data is received by the home institution, it remains under institutional custody. The student, although the subject of the record, cannot hold, transport, or present it to third parties without re-requesting it from the academic services. Consequently, ownership of the record effectively resides within the institution rather than with the student.

This gap is precisely what the Self-Sovereign Identity (SSI) paradigm and the W3C Verifiable Credentials (VC) data model [4] are designed to address. In an SSI ecosystem, individuals hold digital credentials issued by trusted entities and present them to verifiers without involving the issuer at each transaction. The European Blockchain Services Infrastructure (EBSI) [5] provides the distributed trust layer for this model across the European Union by giving access to a permissioned blockchain that anchors decentralized identifiers (DIDs) and trusted issuer registries, enabling verifiers to confirm the provenance of a credential without contacting the issuing institution.

This paper presents a credential issuer that bridges these two domains. The system processes a Transcript of Records expressed in the EWP-compatible ELMO XML format, preserves an institutional validation step, and issues a Verifiable Credential directly to the student through an OpenID for Ver-

ifiable Credential Issuance (OID4VCI) flow [6]. Two issuance profiles are supported, standard OID4VCI producing SD-JWT credentials, and an EBSI draft 11 profile producing JWT-VC credentials. This dual approach accounts for variability in how real-world wallets interpret metadata. The system also provides public verification and supports credential revocation through the StatusList2021 specification [7].

The main contributions of this work are: (i) a documented end-to-end architecture that extends the EWP data pipeline into the EBSI SSI ecosystem while preserving the institutional review role, and (ii) empirical findings on wallet interoperability between OID4VCI and EBSI credential profiles, validated with two production wallets.

The remainder of this paper is organized as follows. Section II reviews the background and related work, covering the Erasmus Without Paper infrastructure, the Self-Sovereign Identity paradigm, and the role of EBSI as a trust anchor. Section III presents the proposed system architecture, detailing the components and overall design. Section IV describes the implementation, including the dual-profile issuance mechanism. Section V reports the validation methodology and experimental results, with particular emphasis on interoperability across wallet implementations. Section VI discusses the main findings and limitations, and Section VII concludes the paper by outlining future research directions.

II. Background and Related Work

This section provides the conceptual and technical background necessary to contextualize this study, along with a review of the most relevant related work.

A. Erasmus Without Paper

EWP is coordinated by the European University Foundation and operates as a peer-to-peer network in which each institution keeps its own systems and exposes a set of standardized Hypertext Transfer Protocol (HTTP) APIs, rather than feeding a single centralized database [2]. A central EWP Registry acts as a public catalogue of participating institutions, publishing manifest files that describe which APIs and versions each institution supports, and allowing any institution to discover, at runtime, what a partner system is capable of exchanging [2]. The API contracts themselves are specified as XML Schemas, which is what lets independent implementations, commercial or open source, interoperate without ambiguity [3].

The API surface covers most of the mobility lifecycle: Institutions and Organizational Units APIs expose basic institutional data, the Inter-Institutional Agreements API manages bilateral agreements, Online Learning Agreements handles the Learning Agreement workflow, and the Transcript of Records API, still in pilot phase, supports the exchange of academic records after a mobility period ends [3].

Even with this coverage, the model keeps every transcript inside the sending and receiving institution's own systems. A student cannot pull their record out of that exchange, store it anywhere, or hand it to someone outside the two institutions involved. Verification only works while both systems are online and willing to talk to each other, which makes it synchronous, bilateral, and entirely dependent on institutional infrastructure remaining available. These are exactly the constraints that make self-sovereign identity and verifiable credentials a natural complement rather than a competing solution.

B. Self-Sovereign Identity and Verifiable Credentials

Self-sovereign identity shifts control of digital identity from central providers to the individual. Rather than depending on federated login systems where identity is contingent on an institutional relationship, SSI gives the user a persistent, portable digital identity [8,13].

The W3C Verifiable Credentials Data Model 2.0 [4] standardizes how assertions about a subject can be expressed in a cryptographically secure, machine-verifiable format. A VC consists of claims made by an issuer about a holder, signed with the issuer's private key. DIDs [13] provide globally unique, user-controlled identifiers that resolve to DID Documents containing public keys and service endpoints.

Together, VCs and DIDs form a three-party trust model where: (1) the issuer creates and signs the credential, (2) the holder stores and selectively presents it, and (3) the verifier validates it cryptographically against a Verifiable Data Registry (VDR), without contacting the issuer. This decoupling grants the holder lifelong custody of the credential which remains valid and portable even after the holder's relationship with the issuing institution has ended.

C. EBSI as Trust Anchor

The EBSI is a permissioned blockchain network operated by the EU Member States, Norway, and Liechtenstein [5]. The network runs Hyperledger Besu (Ethereum client) and restricts block validation to pre-approved institutional nodes through a Proof-of-Authority consensus mechanism [14].

EBSI provides the Verifiable Data Registry at the base of the trust triangle. Its Trusted Issuers Registry (TIR) maintains an on-chain, tamper-evident record of which institutions are authorized to issue which credential types. The registration chain originates with a Trusted Accreditation Organization (typically a national quality-assurance agency) that attests an institution's authority and registers its DID in the TIR. When a verifier receives a credential, it queries the TIR to confirm that the issuer's DID is registered for that credential type, and validates the cryptographic proof. No direct communication with the issuer is required. EBSI natively supports the W3C VC Data Model and defines credential schemas as JSON Schemas registered in the TSR. Its alignment with the eIDAS 2.0 regulation and the forthcoming EU Digital Identity Wallet positions it as the reference trust infrastructure for credential exchange within the European education area [5,15].

D. Related Work

Europass Digital Credentials (EDC) [10] provide a mature, eIDAS-compliant format for digitally sealed diplomas. The European Learning Model (ELM) underpins EDC and aligns

conceptually with W3C VC, though EDCs rely on centralized electronic seals rather than DIDs and do not integrate with the EWP API layer.

Open Badges 3.0 [11] adopted W3C VC Data Model 2.0 as its native signature format, making badges cryptographically verifiable. However, the specification does not mandate a DID method or a particular trust registry, leaving interoperability with EBSI and EWP to individual implementers.

Blockcerts [12] pioneered blockchain-anchored credentials by publishing credential hashes to Bitcoin. The standard predates W3C VCs and uses a proprietary format. Storing hashes of personal academic records on a public, immutable ledger also raises unresolved GDPR compliance questions.

EBSI Diploma Pilot [9] demonstrated cross-border verification of transcripts of records using EBSI between KU Leuven and the University of Bologna, confirming the viability of W3C VCs anchored to the EBSI TIR for academic credentials.

Although these solutions address individual parts of the academic credential lifecycle, none combines EWP-compatible Transcript of Records data, institutional approval, student-held verifiable credentials, and interoperable issuance to digital wallets. EWP enables institutional exchange but not student portability. Europass, Open Badges, Blockcerts, and the EBSI pilot do not provide this complete EWP-to-wallet pipeline. This work addresses that gap through an issuer that transforms an approved EWP/ELMO transcript into a verifiable credential delivered via OID4VCI.

III. System Architecture

The proposed architecture supports the complete lifecycle of a verifiable academic credential, from the import of a Transcript of Records to its issuance, delivery to the student, third-party verification, and possible revocation. It comprises an administrative interface for academic services, services for processing and issuing credentials, persistent storage, wallet-compatible public interfaces, credential-offer delivery, and a public verification service.

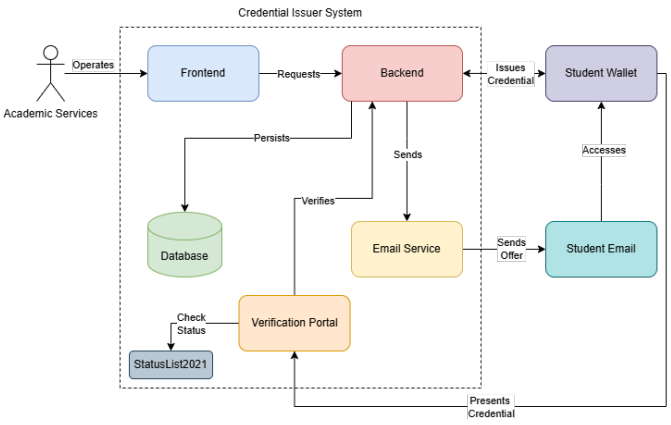


Fig. 1. General component diagram of the credential issuer.

Figure 1 presents an overview of the proposed credential issuer and its main interactions. The system receives a Transcript of Records in an EWP-compatible XML format, which is imported, validated, and made available for review by academic services. Only an approved transcript can originate a credential offer, which the student accepts through a compatible digital wallet.

The architecture separates administrative operations from public interactions. Academic services manage transcript review, approval, offer creation, and revocation, thereby retaining institutional control over the academic data. Public interfaces support wallet-based credential issuance and allow third parties to verify a credential presented by the student. This separation preserves the institutional validation process while enabling the student to hold and present the resulting verifiable credential independently.

A. Credential Lifecycle

The credential lifecycle starts with the import of a Transcript of Records in an EWP-compatible XML format [3]. The backend computes a SHA-256 hash over the original XML document, validates the file against local XML schemas, resolves the relevant EWP and ELMO namespaces, and extracts the academic data required to build an internal transcript representation. A successfully processed document is stored as an imported transcript and mapped to a normalized Transcript of Records, initially awaiting administrative review.

The administrative review step represents a deliberate architectural boundary. XML validation confirms the structural validity of the document, but it does not replace the institutional decision regarding the use of academic data for credential issuance. Academic services therefore review the parsed transcript and either approve or reject it. Only approved transcripts can progress to credential offer creation.

For each approved transcript, the academic services staff has the option to then create a credential offer bundle containing two concurrent offers. One offer targets an OID4VCI [6] profile that issues a `dc+sd-jwt` credential, while the other targets an EBSI-oriented [5] profile that issues a `jwt_vc` credential [4]. This design reflects a practical interoperability constraint observed during development: different wallets may expect different metadata structures, endpoint paths, and credential formats. By associating the profile with the offer itself, the backend can adapt the issuance behaviour to the offer selected by the wallet.

Each offer contains a pre-authorized code, an expiration timestamp, a credential offer URI, and a link to a credential record. The pre-authorized code is not stored in clear text, instead, only its hash is persisted. The offer URI can be delivered to the student by email, together with a QR code. Email delivery is treated as an auxiliary channel and if, for example, sending fails, the offer remains valid and can be resent while it is still active and unexpired.

When the student opens one of the offers in a compatible wallet, the wallet follows the OpenID4VCI pre-authorized code flow [6]. It retrieves issuer metadata, exchanges the pre-authorized code for an access token, and requests the credential from the corresponding credential endpoint. The

backend validates the offer state, expiration, token binding, and requested profile before assembling and signing the credential. After successful issuance, the consumed offer is marked as used, the corresponding credential record becomes issued, and the alternative offer in the same bundle is invalidated. This prevents the same Transcript of Records from producing two simultaneously valid credentials.

B. Verification and Revocation

Verification is supported through a public verifier hosted by the issuer. A verifier can submit a credential through the portal. The system first detects whether the input corresponds to a `dc+sd-jwt` or a `jwt_vc` format. It then validates the issuer signature and checks relevant JWT claims. It subsequently confirms the credential corresponds to a known issuance record in the database. A credential is considered valid only if the signature is correct, the issuer is expected, the credential is registered internally, and the record remains in an issued and non-revoked state.

Revocation is represented at two levels. Internally, a revoked credential record stores its revocation state, timestamp, and an optional reason. For EBSI-oriented [5] `jwt_vc` credentials, the system also uses StatusList2021-based revocation [7]. In this case, a status list index is reserved during issuance and embedded in the credential through the `credentialStatus` field. If the credential is later revoked, the corresponding bit in the compressed status list is updated and the signed status list credential is republished through a public endpoint. The list does not expose personal data, transcript contents, or revocation reasons. It only allows wallets and verifiers to check whether a given index has been revoked.

C. Trust Model

The prototype follows the trust model used by Verifiable Credentials [4] in which an issuer signs a credential, a holder stores it in a wallet, and a verifier validates the credential using public cryptographic material and status information. In the current implementation, issuer identity is represented through configurable issuer metadata and public key material exposed by the backend. This approach effectively enables controlled interoperability testing and validation of the end-to-end credential lifecycle.

Full integration with the EBSI TIR [5] remains a direction for future work rather than a completed feature of the proof of concept. In a production deployment, the issuer DID would be registered and accredited through the appropriate EBSI trust registries, allowing verifiers to check not only that the credential signature is valid, but also that the issuer is authorized to issue that credential type. The current architecture is designed to align with that model, but it does not claim production-level EBSI trust anchoring. This direction is important. The proof of concept does not store academic records on a blockchain, nor does it rely on blockchain storage for credential contents. Trust is instead established through signed credentials, issuer metadata, internal issuance records, and revocation status. In an EBSI-integrated deployment, the distributed ledger would support issuer trust and accreditation, not the storage of student academic data.

IV. Implementation

The implementation realizes the credential lifecycle defined by the proposed architecture. As shown in Fig. 2, the process starts with the import of an EWP-compatible Transcript of Records. The transcript is structurally validated and reviewed by academic services, only an approved record can proceed to credential-offer creation.

For each approved transcript, the system creates alternative credential offers targeting different wallet interoperability profiles. The student receives the offers and selects one through a compatible digital wallet. Once one offer is consumed, a single credential is issued and the alternative offer is invalidated. Issuance executes within a single transaction that acquires a pessimistic write lock on the offer bundle, so concurrent consumption attempts on sibling offers are serialized and the losing request is rejected by state validation, ensuring that one transcript yields at most one credential. The resulting credential can subsequently be presented for public verification and, if necessary, revoked by academic services.

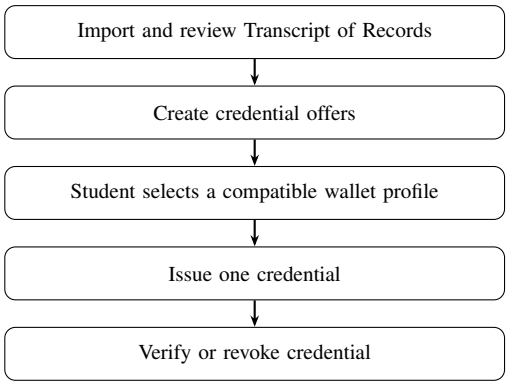


Fig. 2. Simplified functional flow of the credential lifecycle.

A. Dual-Profile Issuance

Once a transcript is approved, the system creates a credential offer bundle containing one offer per supported profile. Two profiles are defined: a standard OID4VCI profile targeting the `dc+sd-jwt` format, and an EBSI draft 11 profile targeting the `jwt_vc` format [4,5]. Each offer is backed by its own credential record, and both offers are grouped under the same bundle. The bundle enforces the rule that only the first consumed offer results in an issued credential: once one offer is used, the competing offer is automatically invalidated, as exemplified in Figure 3.

The dual-profile design responds to a practical finding encountered during wallet interoperability tests. The Sphereon Wallet expects OID4VCI metadata at the default `/.well-known/` paths and obtains a `dc+sd-jwt` credential through the standard pre-authorized code flow [6]. The EBSI-compatible wallet tested expects metadata and token endpoints under an `/ebsi/` prefix, a `jwt_vc` credential



Fig. 3. Flow of the Dual-Profile Issuance with the issuance of the OID4VCI credential

format, and an `id_token` together with a `c_nonce` in the token response. Rather than forcing one wallet to adapt to the other's conventions, the bundle approach lets the same approved transcript serve both profiles. The student receives a single email containing links to both offers, the wallet that opens the link determines the profile and format of the resulting credential.

Each offer carries a pre-authorized code generated from 32 cryptographically random bytes. Only the SHA-256 hash of the code is persisted, the clear-text code is returned once in the API response and embedded in the offer URI. During issuance, the wallet posts the code to the token endpoint. The backend hashes the received value, queries the database for a matching offer, validates that the offer is active and unexpired, and returns a Bearer access token. In the prototype, token sessions are held in memory, keyed by token hash, with a configurable time-to-live bounded by the remaining lifetime of the offer. For the EBSI profile, the token response additionally includes a self-issued `id_token` and a `c_nonce` to satisfy the EBSI diagnostic draft 11 conventions expected by the tested wallet.

The credential endpoint resolves the access token session, retrieves the associated offer and profile, and delegates to the appropriate issuance method. Two construction paths exist. For the OID4VCI profile, the service builds an SD-JWT using the Authlete SD-JWT library. The academic claims (student name, number, identifier, home and host universities, period, issue date, total European Credit Transfer and Accumulation System (ECTS), grade scale, courses, and evidence) are encoded as SD-JWT disclosures, giving the holder selective disclosure capability (show what you want) while keeping structural JWT claims such as issuer, subject, issue time, and credential type public. The claims are assembled, signed with the issuer's elliptic-curve (EC) P-256 key, and serialized with the disclosures appended after a tilde separator.

For the EBSI profile, the service constructs a W3C Verifiable Credential envelope inside a JWT. The `vc` claim contains the `@context`, `type`, `issuer`, `credentialSubject` with the academic fields, a `credentialStatus` entry referencing a Status List 2021 index, and an `evidence` block that records the SHA-256 hash of the source EWP XML document. The resulting JWT carries the JSON Object Signing and Encryption (JOSE) type `JWT` and is signed with the same EC P-256 key. Both issuance paths record the credential in the internal database for later verification, allowing the verifier to later distinguish credentials actually issued by this system from credentials that are merely cryptographically well-formed.

B. Verification and Revocation

Verification is exposed through a static HyperText Markup Language (HTML) page at `/verify` and a companion API at `/api/verify`. Neither endpoint requires authentication. The service normalises the input then detects the format: a credential containing an isolated tilde character is treated as `dc+sd-jwt` otherwise it is treated as `jwt_vc`. The JWT payload before the first tilde is parsed and validated in sequence: the algorithm must be ES256, the key ID must match the issuer's configured key, the signature must verify against the issuer public key, the issuer DID must equal the configured credential issuer DID, the subject DID must be present and start with `did:`, and temporal claims must fall within acceptable bounds with a 300-second clock skew tolerance.

If all cryptographic and claim-level checks pass, the service looks up the credential in the issuance database. A credential whose record is not found is classified as `UNRECOGNIZED`, even if the signature is valid. This distinction prevents the verifier from accepting credentials that may be valid JWTs but were not issued by this system. If the credential is found, its record status is `ISSUED`, and its optional Status List 2021 entry is not revoked, it is classified as `VALID` otherwise it is classified as `INVALID`.

Revocation happens at two levels. Internally, each credential record stores a revocation timestamp and an optional reason. Only records in the `ISSUED` state can be revoked. Records already revoked return their current state idempotently. For `jwt_vc` credentials issued through the EBSI profile, revocation is also reflected in the Status List 2021 infrastructure [7]. During issuance, the system reserves an index in a shared bitstring list. The list is compressed with GZIP, encoded in Base64URL, wrapped in a signed JWT of type `StatusList2021Credential`, and published at `/credentials/status-lists/{id}`. When a credential is revoked, the bit at the reserved index is toggled to 1 and the list credential is re-signed, as exemplified in Figure 4. The status list contains no personal data, no transcript contents, and no revocation reasons. A verifier fetches the list, checks the indexed bit, and determines whether the credential has been revoked. That list exposes status bits but contains no transcript content, revocation reason, or direct index-to-student mapping.

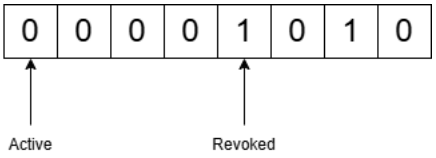


Fig. 4. Simplified example of Status List 2021 indexes

V. Validation and Results

The validation assessed whether the prototype supports the intended end-to-end transformation of an EWP-compatible Transcript of Records into a verifiable credential that can be

held by a student and validated by third parties. The main objectives were to confirm the correctness of transcript import and processing, preservation of the institutional approval boundary, controlled issuance through the two supported profiles, and enforcement of single issuance for each approved transcript.

The evaluation also assessed the supporting lifecycle operations: delivery of credential offers, public verification of issued credentials, and effective revocation. Validation combined automated backend tests, manual checks of the administrative interface, and end-to-end interoperability scenarios with external wallets. This combination was intended to evaluate not only isolated functions, but also the consistency of the complete credential lifecycle in realistic execution conditions.

A. Automated Testing and End-to-End Validation

Automated coverage spans seven functional areas across more than twenty test classes, covering XML import and schema validation, administrative review, credential offer creation, email notification, credential issuance through both OID4VCI and EBSI profiles [5,6], public verification, and Status List 2021 revocation [7]. The suite comprises 182 unit tests, all of which passed in the validation run.

A dedicated set of 22 XML fixtures was prepared with 20 structurally valid transcripts (`valid-tor-01.xml` through `valid-tor-20.xml`), one base file used as a reusable assertion target, and one deliberately malformed document. The fixtures exercise the import pipeline with SHA-256 hash computation, local XSD validation against the EWP iMobility ToR v3 schemas, namespace-aware XPath parsing, duplicate detection, and persistence of the normalized transcript. They also confirmed that the parser handles variations in course count, institution names, grading scales, and missing English translations of course titles.

The following domain invariants were verified programmatically: (i) a transcript in any state other than `APPROVED` cannot generate a credential offer; (ii) an expired or already-consumed offer is rejected at the *token endpoint*; (iii) a pre-authorized code whose *hash* does not match any persisted offer returns an OAuth error; (iv) a competing offer in the same bundle is invalidated when its sibling is used; (v) only records in `ISSUED` state can be revoked; and (vi) a revoked credential is rejected by the public verifier even when its cryptographic signature remains intact.

End-to-end validation covered seven scenarios executed against a running instance of the full stack with backend, PostgreSQL, Mailpit, the Electron administrative interface, and the public verification page. The scenarios trace the complete credential lifecycle. The process begins with XML import, followed by administrative review and approval. It then proceeds to dual-profile offer creation and email delivery, inspected through the Mailpit web interface. The final stages include issuance through two external wallets, public verification of the resulting credential, administrative revocation, and subsequent re-verification of the revoked credential.

B. Wallet Interoperability

The wallet interoperability tests revealed the most significant differences. Table I summarises the differences observed between the two issuance profiles and the two wallets tested.

TABLE I
INTEROPERABILITY COMPARISON BETWEEN THE TWO SUPPORTED ISSUANCE PROFILES AND TESTED WALLETS.

Aspect	**OID4VCI profile**	**EBSI profile**
Wallet tested	Sphereon Wallet	Portuguese EBSI Wallet (PDW)
Credential format	`dc+sd-jwt`	`jwt_vc`
Metadata paths	`/.well-known/`	`/ebsi/.well-known/`
Selective disclosure	Yes, through SD-JWT disclosures	No, claims are inlined
Token response	`access_token`	`access_token`, `id_token`, `c_nonce`
Status list	Internal record only	Status List 2021 entry
Adaptations needed	No additional profile-specific adaptation	Separate metadata path, token fields, VC envelope
Issuance result	Successful in final validation run	Successful after profile adaptations

The Sphereon Wallet consumed the OID4VCI offer without additional issuer-side adjustments in the final validation run. It retrieved the credential issuer metadata from the default `/.well-known/` paths, exchanged the pre-authorized code for a Bearer token, and requested the credential at the standard credential endpoint. The resulting `dc+sd-jwt` was stored in the wallet with selective disclosure support.

The EBSI-compatible wallet required adaptations. It expected the credential issuer metadata under the `/ebsi/` prefix and returned an error when the default endpoint structure was presented. The adapted metadata included a different credential configuration identifier, the `jwt_vc` format declaration, and a `credential_definition` block absent from the OID4VCI profile. At the token endpoint, the wallet expected an `id_token` and a `c_nonce` alongside the access token. Omitting either field caused the wallet to abort. The credential endpoint required the `credentialStatus` field to be populated with a valid Status List 2021 entry. Once these elements were aligned, the wallet completed issuance and stored the `jwt_vc` credential.

The dual-profile bundle validated its intended purpose across both tests. In each end-to-end run, only one offer was consumed, the competing offer was automatically invalidated, and a single `CredentialRecord` transitioned to `ISSUED`. No run produced two valid credentials for the same transcript.

These results confirm the practical observation that drove the dual-profile architecture, the SSI wallet ecosystem does not yet behave uniformly, and small differences in metadata structure or response payloads can block issuance entirely. Serving multiple profiles from a single transcript allows the system to accommodate this diversity without imposing a single profile choice on the issuer or the wallet.

VI. DISCUSSION

The prototype confirms the technical feasibility of an EWP-to-SSI pipeline with institutional validation. The academic

services workflow remains intact, comprising the stages of importing, review and approval. What changes is the output. Instead of producing a transcript confined to institutional systems, the process results in a signed credential delivered to the student. This credential can be stored in a personal wallet and later verified through public verification mechanisms. EWP alone does not provide this portability layer.

Several constraints limit the current proof of concept. The issuer identity uses `did:web` rather than `did:ebsi`, the system publishes a DID Document and JWKS, but the DID is not registered in the EBSI Trusted Issuers Registry, so a verifier cannot independently confirm that the issuer is accredited for the credential type presented. Only two wallets were tested, and the EBSI-compatible wallet required profile-specific adaptations, a different EBSI-conformant wallet might fail on a different metadata field. Credentials in `dc+sd-jwt` format lack a public revocation mechanism in this version, their status is checked against an internal record, which works as long as the issuer's own verifier is used, but does not provide the same decentralised verification model expected by external verifiers.

VII. Conclusion

This work met its main objective of demonstrating the technical feasibility of bridging Erasmus Without Paper and Self-Sovereign Identity through a verifiable credential issuer. The developed proof of concept showed that an EWP-compatible Transcript of Records can be imported, reviewed under institutional control, transformed into a signed digital credential, delivered to the student, and later verified by third parties. In this sense, the prototype confirmed that the portability gap left by institution-centred mobility infrastructures can be addressed without removing the academic services from the approval process.

Three aspects are particularly important to retain. First, the institutional validation step was preserved as a mandatory condition for issuance, ensuring that only reviewed and approved academic data can originate a credential. Second, the resulting credential extends the usability of the Transcript of Records beyond the original administrative workflow, allowing the student to hold and present it independently. Third, the dual-profile issuance approach proved to be a practical response to the current fragmentation of the wallet ecosystem, since interoperability depended not only on standards compliance but also on profile-specific adaptations observed in real wallet implementations.

At the same time, the work exposed important difficulties and constraints. The main challenge was interoperability with external wallets, especially the differences in metadata expectations, endpoint structure, token responses, and credential formats between the tested OID4VCI and EBSI-oriented flows. Additional constraints arose from the proof-of-concept nature of the implementation, including temporary public exposure for testing, in-memory token handling, and dependence on development-oriented auxiliary services.

The prototype also leaves clear limitations and directions for future work. A first priority is full integration with the EBSI trust infrastructure, particularly issuer registration in the Trusted Issuers Registry, so that verification can rely on institutional accreditation in addition to signature validity. Further work should also expand interoperability testing with additional wallets, strengthen operational aspects such as persistent token storage, key rotation, and institutional email delivery, and refine revocation and verification mechanisms for more realistic production-oriented deployments.

Acknowledgment

This work was financially supported by Project BlockchainPT – Decentralize Portugal with Blockchain Agenda, WP9: Capacitation 02/C05-i01.01/2022.PC644918095-00000033, funded by the Portuguese Recovery and Resilience Program (PRR), the Portuguese Republic, and the European Union under the Next Generation EU program.

João Carreira and Lourenço Ferreira acknowledge the University of Leiria and Oeste for the financial support of their scientific research fellowships (BI).

The authors acknowledge the use of generative AI tools during the preparation of this manuscript for language editing and structural revision of early drafts. All AI-generated content was reviewed and edited by the authors. All technical content, experimental design, implementation, and data analysis were performed by the authors.

References

[1] European Commission, *Erasmus+ Annual Report 2024*. Publications Office of the EU, 2025, doi:10.2766/8201983.

[2] European University Foundation, "Erasmus Without Paper," 2024. [Online]. Available: https://uni-foundation.eu/erasmus-without-paper/

[3] Erasmus Without Paper Consortium, "EWP Incoming Mobility ToR API, v3," 2024. [Online]. Available: https://github.com/erasmus-without-paper/ewp-specs-api-imobility-tors

[4] M. Sporny *et al.*, "Verifiable Credentials Data Model v2.0," W3C Recommendation, 2025. [Online]. Available: https://www.w3.org/TR/vc-data-model-2.0/

[5] European Commission, "European Blockchain Services Infrastructure (EBSI)," 2025. [Online]. Available: https://digital-strategy.ec.europa.eu/en/policies/european-blockchain-services-infrastructure

[6] OpenID Foundation, "OpenID for Verifiable Credential Issuance — Draft 15," 2025. [Online]. Available: https://openid.net/specs/openid-4-verifiable-credential-issuance-1_0.html

[7] W3C Credentials Community Group, "Status List 2021, v1.0," 2025. [Online]. Available: https://www.w3.org/TR/2023/WD-vc-status-list-20230427/

[8] A. Grech, I. Sood, and L. Ariño, "Blockchain, Self-Sovereign Identity and Digital Credentials: Promise vs Praxis in Education," *Frontiers in Blockchain*, vol. 4, p. 616779, 2021, doi:10.3389/fbloc.2021.616779.

[9] E. Tan, E. Lerouge, J. Du Caju, and D. Du Seuil, "Verification of Education Credentials on EBSI: Action Research in a Cross-Border Use Case between Belgium and Italy," *Big Data and Cognitive Computing*, vol. 7, no. 2, p. 79, 2023, doi:10.3390/bdcc7020079.

[10] European Commission, "European Digital Credentials," Europass, 2025. [Online]. Available: https://europa.eu/europass/en/european-digital-credentials

[11] 1EdTech Consortium, "Open Badges Specification," 2025. [Online]. Available: https://www.1edtech.org/open-badges

[12] Blockcerts Community, "Blockcerts — The Open Standard for Blockchain Credentials," 2025. [Online]. Available: https://www.blockcerts.org/

[13] M. Sporny *et al.*, “Decentralized Identifiers (DIDs) v1.0,” W3C Recommendation, 2022. [Online]. Available: https://www.w3.org/TR/did-core/
[14] EBSI, “Understanding EBSI’s Blockchain Ecosystem,” EBSI Hub, 2026. [Online]. Available: https://hub.ebsi.eu/blockchain
[15] European Commission, “eIDAS Regulation,” Digital Strategy, s.d. [Online]. Available: https://digital-strategy.ec.europa.eu/en/policies/eidas-regulation