Masters Programmes: Group Assignment Cover Sheet

Student Numbers: Please list numbers of all group members	**5669869, 5673783, 5613239, 5583421**
Module Code:	**IB9YW0**
Module Title:	**Fintech: Digital Currencies and Decentralised Finance**
Submission Deadline:	**April 8, 2025**
Date Submitted:	**April 8, 2025**
Word Count:	**4993**
Number of Pages:	**18**
Question Attempted:	**In reference to guidance questions, 1, 2, 3**
Have you used Artificial Intelligence (AI) in any part of this assignment?	**No, AI has not been used for any part of this assignment.**

Academic Integrity Declaration- We're part of an academic community at Warwick. Whether studying, teaching, or researching, we're all taking part in an expert conversation which must meet standards of academic integrity. When we all meet these standards, we can take pride in our own academic achievements, as individuals and as an academic community. Academic integrity means committing to honesty in academic work, giving credit where we've used others' ideas and being proud of our own achievements.

In submitting my work, I confirm that:

- I have read the guidance on academic integrity provided in the Student Handbook and understand the University regulations in relation to Academic Integrity. I am aware of the potential consequences of Academic Misconduct.
- I declare that this work is being submitted on behalf of my group and is all our own, , except where I have stated otherwise.
- No substantial part(s) of the work submitted here has also been submitted by me in other credit bearing assessments courses of study (other than in certain cases of a resubmission of a piece of work), and I acknowledge that if this has been done this may lead to an appropriate sanction.
- Where a generative Artificial Intelligence such as ChatGPT has been used I confirm I have abided by both the University guidance and specific requirements as set out in the Student Handbook and the Assessment brief. I have clearly acknowledged the use of any generative Artificial Intelligence in my submission, my reasoning for using it and which generative AI (or AIs) I have used. Except where indicated the work is otherwise entirely my own.
- I understand that should this piece of work raise concerns requiring investigation in relation to any of points above, it is possible that other work I have submitted for assessment will be checked, even if marks (provisional or confirmed) have been published.
- Where a proof-reader, paid or unpaid was used, I confirm that the proof-reader was made aware of and has complied with the University's proofreading policy.

Upon electronic submission of your assessment, you will be required to agree to the statements above

EduTrust Scale: A Decentralized Infrastructure for Cross-Border Academic Credentialing

Group 4

April 8, 2025

Submitted in partial fulfilment of the requirements for

IB9YW0 – Fintech: Digital Currencies and Decentralised Finance

Contents

Abstract

This whitepaper first details Blockchain and how it overcomes limitations of traditional finance, backed with two real world examples. Two case studies have been done to cover in-depth on how diversified the use of blockchain is. Our novel protocol, a decentralized credentialing framework, EduTrust enables verifiable, tamper-proof academic records without relying on centralized authorities. Using permissioned blockchains, NFT tokenization, and zero-knowledge proofs, the protocol ensures privacy-compliant, user-controlled credentials. Transactions are timestamped, hashed, and stored via Merkle roots, enabling scalable validation. Smart contracts govern issuance, revocation, and employer querying. EduTrust achieves cross-jurisdictional interoperability and real-time verification with minimal infrastructure. Credentials are cryptographically portable and can be selectively disclosed, providing a robust, trustworthy alternative to traditional systems.

Section 1 Alternative Uses of Blockchain

1.1 Introduction and motivation

While blockchain is often associated with cryptocurrencies like Bitcoin and Ethereum, its potential extends beyond digital assets. Blockchain structures and shares data in a way that enables decentralized trust and secure record-keeping. This is especially valuable in areas like supply chain management and carbon credit trading, where data integrity and transparency are essential.

Traditional supply chains lack real-time visibility, making risk management and timely responses difficult. Similarly, carbon markets struggle with fraudulent claims and high transaction costs, undermining trust and effectiveness.

Blockchain offers a solution by providing access to a shared, tamper-proof ledger with consistent and verifiable data in real time. This paper explores the current inefficiencies in traditional systems and how blockchain technology can improve performance.

1.2 Traditional Finance and Its Limitations

Traditional supply chains rely on fragmented networks and siloed systems. This fragmentation results in inconsistent data, delays in communication, and a lack of transparency throughout the production and distribution process. Stakeholders often cannot view the full history of a product. This creates information asymmetries, which increase operational risk and make it more difficult to resolve disputes or trace the origin of goods in the event of a recall or quality issue. Walmart's experience before adopting blockchain is a notable example of this inefficiency. The company once required nearly a week to trace the origin of contaminated mangoes during a food safety investigation, a delay that posed serious public health and reputational risks (Kamath, 2018). This inefficiency stems from the manual effort to reconcile data from various parties across the supply chain. In financial terms, these inefficiencies also raise costs. Lenders and insurers involved in supply chain finance often make decisions based on partial or outdated information, leading to higher interest rates and stricter collateral requirements for small and medium-sized enterprises (Chod et al., 2020).

Carbon credit trading systems face different challenges but with similarly problematic outcomes. These markets are designed to support climate mitigation by allowing firms to offset emissions by purchasing certified carbon credits. However, traditional systems rely heavily on centralized registries and third-

party verifiers to authenticate and track credits. These intermediaries add cost and time to the process and are not immune to fraud.

In addition to fraud, inefficiencies in the carbon market are driven by slow settlement times, poor transparency, and limited access for smaller or less-developed project developers. Many carbon credits are sold through opaque over-the-counter markets, where buyers have little assurance that the credits they purchase represent real, additional, and verified emissions reductions. The high cost of validation also acts as a barrier to entry for small environmental projects, which undermines the system's inclusiveness and scalability.

1.3 How Blockchain Enhances Efficiency

Blockchain enhances these systems through a shared, decentralized ledger that is immutable, transparent, and accessible to authorized participants. In supply chains, it allows each process step to be securely and verifiably recorded. Integrated with sensors, IoT devices, or ERP systems, blockchain can track goods' movement, condition, and ownership in real time. This visibility lowers the risk of counterfeiting, improves disruption response, and streamlines coordination. A leading example is Walmart's adoption of IBM's Food Trust blockchain. The company reduced traceability time from seven days to just 2.2 seconds (Kamath, 2018). This leap enhances food safety and reduces recall-related losses. For financial institutions, access to real-time, immutable data improves risk assessment, enabling more efficient capital allocation and reduced financing costs (Gaur & Gaiha, 2020).

In carbon markets, blockchain improves integrity by tokenizing offsets into unique digital assets tracked from issuance to retirement. This prevents double spending and supports real-time auditing—crucial for ensuring environmental credibility. Tokenized credits can be traded directly between buyers and sellers, cutting costs and broadening access for smaller projects.

The Toucan Protocol exemplifies this approach, converting traditional credits into fungible tokens on Ethereum. This increases liquidity, traceability, and transparency in carbon markets (Swinkels, 2023). Such systems can also automate compliance through smart contracts that retire credits once emission thresholds are met or penalties are triggered.

Another key efficiency gain comes from blockchain's support for automation via smart contracts. These self-executing agreements operate on predefined rules, eliminating manual enforcement and third-party validation. In supply chains, they can automate payments once delivery is confirmed. In carbon markets, they can verify offset criteria or trigger programmatic climate actions based on emissions data (Saraji & Borowczak, 2021). Blockchain significantly improves systems like supply chains and carbon trading, where inefficiencies such as fraud, delayed traceability, and high costs limit transparency and trust. In contrast, blockchain provides real-time, secure access to shared data, reducing operational friction and expanding participation.

Through immutable ledgers, smart contracts, and tokenization, blockchain offers a more efficient foundation for tracking goods and verifying environmental outcomes. Despite challenges like interoperability and regulatory uncertainty, blockchain adoption is growing, increasingly positioning the technology as a backbone of a transparent and efficient global infrastructure.

Section 2 Case Study: Air Carbon Exchange

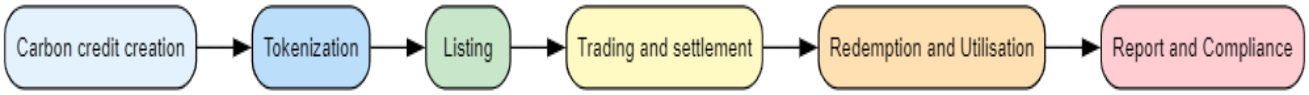
2.1 Introduction to Carbon credits

Carbon credit is a tradeable permit – a right to emit 1 metric ton of CO2 or equivalent GHG (Green House Gases). There are two types of carbon credit markets – Compliance Market and Voluntary Market. Compliance Market is based on legally binding government mandated emission requirements. Examples are EUETS and UKETS (Emissions Trading System) while voluntary Market is not legally binding, and companies trade to reduce emissions, achieve carbon neutral or achieve ESG (Environmental, Social and Governance) goals. Companies with projects that reduce or remove carbon, or GHG, through renewable projects, reforestation, methane capture projects, etc earn credits. Companies emitting less than their permitted emissions earn credits. Companies emitting more than their limit or to meet their emission targets can purchase carbon credits with carbon credits sale help companies to invest in renewables, conservation and capture projects.

2.2 Air-Carbon Exchange

ACX is a block-chain based digital exchange, similar to stock exchange, for carbon credits, operates in the Voluntary Carbon Market category. It leverages block chain technology to digitally trade carbon credits through tokenization of the credits, thus can transform the traditional carbon credit market. This helps in overcoming several challenges compared to traditional system of carbon credit sale.

2.3 ACX Process



1) **Carbon credit creation**

Carbon credits generated by accredited environmental projects are verified by third parties such as Gold Standard, Verra, etc., and issued through carbon registries after authentication.

2) **Tokenization of Carbon credits**

ACX converts the carbon credits into fungible digital tokens. Tokenization involves minting ERC-20 tokens through smart contracts. One token is equivalent to one ton of CO_2 reduction.

3) **Listing**

ACX functions like any stock exchange, offering real-time order books, market depth analysis and trading pairs. Sellers list their carbon tokens with price and conditions, while buyer can place their bids.

4) **Trading and settlement**

When a buy order matches a sell order, a smart contract executes the transaction automatically. Tokens are transferred and balances are updated instantly and immutably. The entire transaction is recorded on blockchain, ensuring traceability.

5) **Redemption & Utilization**

Token holders can either trade/hold the tokens or redeem the tokens to offset their emissions. Redemption of tokens involves burning, which removes the tokens permanently from the system, ensuring there is no double counting. Block chain ledger stores traceability and closure of the action, ensuring integrity.

6) Report & Compliance

All transaction data are logged automatically and immutably. Automatic compliance reports are generated showing complete transaction history, which are available to share with the regulators and stakeholders.

2.4 ACX and Block Chain - Unique features

A) Smart contracts: These are decentralized, cost-effect, self-executed and automated programs, ensuring trust, transparency, security, efficiency. They handle issue, transfer and redemption of carbon credits without intermediaries.

B) Permissioned blockchain: Unlike public blockchains like Bitcoin/Ethereum, access is restricted to only authorized entities. Only verified users such as accredited project developers, regulatory authorities, can access the system.

C) Community voting: Decision making is decentralized through community voting for policy changes, protocol upgrades and proposals. A rule can be set to run an activity if 70% of the members vote for.

D) Oracle integration: Enables connectivity to real world emissions data which helps in regulatory compliance and reporting. External data feeds are integrated which help in real time update of credit status.

E) KYC (Know your customer): Identity of customers are verified before they join, ensuring regulatory and compliance requirements.

F) AML (Anti money laundering): AML ensures prevention of money laundering, fraud and terrorist fundings.

2.4 ACX vs Traditional Market

Feature	ACX	Traditional Carbon Market
Tokenization	Credits are tokenized based on Digital ERC-20 tokens	Credits are paper/registry-based
Trading	Automated via smart contracts	Manual or broker-mediated
Settlement Time	Near-instant	Days to weeks
Transparency	All transactions are recorded in block chain with full transparency	Limited transparency specifically in tracing chain of transactions
Double Counting Prevention	Tokens are unique and are burnt upon redemption preventing double counting.	Due to lack of visibility of related transactions, there is a high risk of double counting
Access & Liquidity	Global platform available 24x7	Access is restricted and usually it is OTC trading

Fees & Intermediaries	Very low transaction costs and there are no intermediaries	Very high broker and agent fees
Reporting	Detailed, auto generated and auditable	Manual and fragmented reports

Table 1: Comparison of ACX and Traditional carbon markets

2.5 Summary for ACX

ACX integrates traditional exchange processes with blockchain-based tokenization, enabling secure, transparent, and automated carbon credit transactions. It enhances efficiency while meeting compliance and regulatory standards, addressing key limitations of legacy systems.

Section 3 Case Study: IBM Food Trust

3.1 Problems in the traditional food supply chain

As several reports and articles (Mol, 2013; Mintel, 2020) have noted, there is a growing demand for transparency in food industry. However, it is difficult to tracking foods' source in tradition supply chain, due to the complex and fragmented nature of production, logistics and sales process, a large number of suppliers, and their incommunicable databases. From seed supply and cultivation to selling, each process is isolated in terms of information. While data from the pre-stage may be accessible via QR codes and other methods, the entire process remains inaccessible to end consumers (Singh & Sharma, 2022). The difficulties in tracing food origins can also result in slow and costly recalls of unsafe products.

Food fraud, such as document falsification and the improper addition, removal, or substitution of ingredients for economic benefits (Du et al., 2018), also poses risks to consumer food safety. These practices have remained unresolved for many years. An instance is 2008 Chinese milk scandal. Sanlu Group illegally added melamine to pass protein tests, leading to severe health consequences to infants. Both problems can lead to consumer distrust, resulting in declining sales and reputational damage that ultimately harm the brand.

3.2 Protocol of IBM Food trust

IBM Food Trust is built on the Hyperledger Fabric framework, a permissioned blockchain governed by the Linux Foundation. Unlike public blockchains such as Bitcoin or Ethereum, Fabric operates as a private blockchain, where only authorized participants can join the network and transact. Its nodes are controlled by participating groups, and transactions are recorded on the blockchain only after being endorsed by multiple designated nodes. The ledger is hosted on IBM Cloud, and organizations retain ownership and control over the encrypted data they upload. In terms of governance, Food Trust's governance model and advisory council define and maintain its decentralized nature.

Users of IBM Food Trust can complete processes such as data onboarding, information tokenization, blockchain tracking, smart contract execution, and verification through platform modules like Trace, Certifications, and Data Entry & Access. During tokenization, each batch of food is assigned a unique digital token containing metadata about the product's journey, which consumers can verify by scanning a QR code. Blockchain tracking ensures that each touchpoint records its own updates on food

transportation, helping to prevent data manipulation. And the use of smart contracts ensures that payments are automatically released once conditions are met, reducing costs and improving efficiency.

3.3 Properties of IBM Food trust

Compared with other types of food supply chain, IBM Food Trust has five unique features that make it competitive.

A) Real-time scalability & a transparency tool

This means producers, distributors, retailers and customers can instantly access to product history. It helps build consumer trust while reducing the cost of tracking and recalling food for providers.

B) Advanced analytics and insights

Through IBM Food Trust, suppliers can perform predictive analysis to optimize supply chains, including inventory management and dispatch, reducing cost and waste.

C) IoT Devices

IBM Food Trust adopts standardized APIs, enabling integration with IoT devices that monitor temperature and humidity. This helps ensure the quality and safety of goods—especially perishable food—during transportation and storage.

D) Seamless integration

IBM Food Trust enables users to efficiently upload, record, and verify food-related information. The adoption of standardized APIs expands the platform's capacity to support new functionalities as user needs evolve. Furthermore, the system is designed based on internationally accepted GS1 standards, facilitating information sharing between different companies and even integration with other supply chain management systems.

E) Decentralization and safety

The protocol and decentralized governance design ensures the security of data and transactions.

3.4 Summary for IBM Food Trust

The introduction of blockchain into food supply chain offers several benefits, including improved food safety, more precise recall management, fraud prevention, and greater efficiency and collaboration. In the future, the IBM Food Trust is expected to expand its network of participants and areas of application. Nevertheless, the platform also faces several challenges. The first concern is data gaps caused by the unwillingness of certain participants to engage, which can undermine traceability (Lopez, 2018). Another risk is that increasing transaction volumes may lead to potential latency issues. In addition, Food Trust may face competition from other frameworks. These issues could be addressed through enhanced consensus and cooperation, sharding or layered architectures, and other proposed solutions. The success of IBM Food Trust has also encouraged the application of blockchain solutions in other industries, such as Carrefour's efforts to apply it in the textile sector.

Section 4 EduTrust Scale

Where Academic Credentials Meet Global Standards

4.1 Introduction EduTrust Scale

In a world driven by digital transformation, academic credentialing remains shockingly antiquated. Fraudulent diplomas, delayed verifications, and fragmented record systems have plagued both institutions and employers. According to the World Education Services (2022), over 70% of global employers face difficulty verifying foreign academic records. 'EduTrust Scale' proposes a blockchain-based protocol designed to solve this crisis. By combining permissioned blockchains, ERC-721 tokenization, and zero-knowledge proofs, EduTrust offers a trustworthy, privacy-preserving framework for issuing, verifying, and storing academic credentials. The protocol enables instant employer validation, student-controlled disclosure, and institutional-grade governance through a DAO model.

4.2 EduTrust Vision & Mission

EduTrust Scale envisions future where academic credentials are universally verifiable, tamper-proof, and user controlled. The protocol's mission is to replace slow paper-based systems with a decentralized, programmable, and privacy-preserving credential infrastructure. At its core, EduTrust operates as a multi-layer protocol that tokenizes academic records into secure, permissioned NFTs (ERC-721 and ERC-1155), governed by smart contracts and verifiable through decentralized identifiers (DIDs).

The mission of EduTrust is defined by five foundational goals:

1. Transparency – Full auditability and traceability of records via on-chain hashes
2. Fraud Resistance – Immutable credentials issued by verified institutions
3. Portability – Cross-border compatibility with DID-based wallet access
4. Automation – Smart contract enforcement of issuance, revocation, and expiration rules
5. Privacy Compliance – Zero-knowledge selective disclosure and FERPA/GDPR readiness

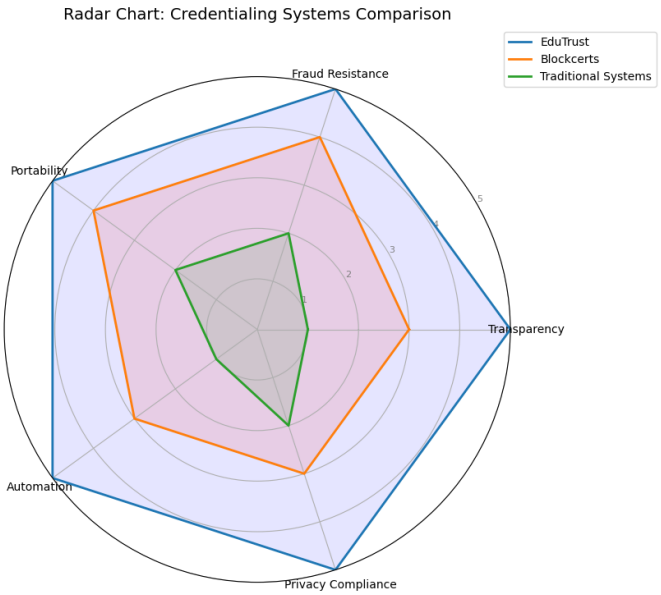


Figure 1: Credentials System Comparison

System	Transparency	Fraud Resistance	Portability	Autonomy	Privacy
EduTrust	5	5	5	5	5
Blockcert	3	4	4	3	3
Tradfi	1	2	2	1	2

Note: See appendix 1, for an expanded and academically referenced scores.

Table 2: Credentials System Comparison

As illustrated in Figure 1, EduTrust outperforms both traditional systems and earlier blockchain-based efforts like Blockcerts across these five core dimensions. While others offer digital representation, EduTrust delivers a comprehensive protocol capable of embedding trust, control, and automation directly into the academic record. Additionally, the platform will showcase a training/demo version for students and universities as a starting platform.

4.3 System Architecture

EduTrust Scale is built on a modular three-layer protocol architecture that ensures secure credential storage, verifiability, and global interoperability. Each layer is functionally distinct yet collectively interlinked to support a trustless academic credentialing ecosystem. This section outlines the operational logic and technologies that power EduTrust, emphasizing cryptographic integrity, digital asset generation, and decentralized governance pathways.

4.3.1 Overview of the stack

The protocol is composed of three vertically integrated layers:

- Layer 1 – Settlement (infrastructure & base storage)
- Layer 2 – Asset (credentials as on-chain assets)

- Layer 3 – Protocol (interaction logic, identity, analytics)

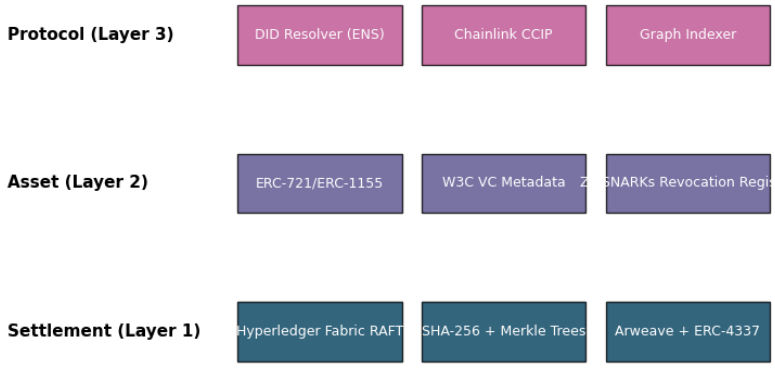


Figure 2: Architecture Stack

This stack is visualized in Figure 2, where each component contributes to EduTrust's transparency, automation, and data privacy mandates.

4.3.2 Layer 1 : Settlement Layer

This foundational layer anchors the system's security and data integrity. Academic records are hashed using the SHA-256 algorithm and stored as digital proofs. To enable batch verifiability, the system employs Merkle Trees, allowing large credential datasets to be verified via a single root hash—minimizing gas fees and improving scalability.

For example, a credential such as;

$$"John\ Doe, MSc\ Finance, Warwick, 2025"$$

yields a secure on-chain fingerprint:

$$752ff89140cc1ce82fc525d87b042bcf9e6090791311697cd2c12d6d8dbed733$$

Multiple records are then hashed into a Merkle Root like:

$$b9c39935ae6d2354c7ab5e452e668c99ba02fc95891b97444e48b925e3c89620.$$

The system uses Hyperledger Fabric (RAFT consensus) for permissioned execution, while Arweave offers immutable off-chain archival storage. To optimize transactions, EduTrust integrates ERC-4337 account abstraction, reducing the cost and friction of credential submission. EduTrust uses Hyperledger Fabric (RAFT consensus) because its permissioned model aligns with academic institutions' need for controlled validator access. Unlike public blockchains (e.g., Ethereum), Fabric offers:

- Higher throughput (1,000+ TPS vs. Ethereum's ~15 TPS).
- Lower latency (instant finality vs. PoW/PoS probabilistic finality).
- GDPR compliance (off-chain data storage). Trade-off: Centralization risk mitigated by multi-university validator nodes.

4.3.3 Layer 2 : Asset Layer

At this stage, verified credentials are tokenized using standards such as ERC-721 or ERC-1155. Each token is embedded with W3C-compliant Verifiable Credential (VC) metadata, ensuring semantic interoperability across institutions.

To enhance user privacy and selective data sharing, EduTrust employs Zero-Knowledge Proofs (ZK-SNARKs) (Karuna, J. and Banerjee, A. 2019). This enables a student to disclose specific information without revealing full identity. Additionally, Revocation Registry Smart Contracts are used to track invalidated or outdated credentials.

4.3.4 Layer 3 : Protocol Layer

The final layer governs user interaction. Every student is assigned a Decentralized Identifier (DID), allowing verifiable linkage between the blockchain credential and real-world identity. For instance, a query such as:

$$did: edutrust: bob456$$

resolves:

$$ipfs://QmXyzCredentialHash2,$$

allowing employers or universities to instantly verify credentials.

EduTrust also integrates Chainlink's Cross-Chain Interoperability Protocol (CCIP) for verification across multiple blockchain networks, while the Graph Protocol indexes on-chain data for real-time searchability.

4.3.5 Inter-Layer flow

When a credential is submitted, Layer 1 hashes and stores it, Layer 2 tokenizes it, and Layer 3 enables real-time querying and validation. Each layer is modular yet tightly coupled, making EduTrust scalable and upgradeable over time. (**Appendix 2 for simulation**).

4.4 Tokenization of Academic Records

EduTrust Scale turns academic credentials into programmable, on-chain digital assets, enabling a new paradigm of verifiable, privacy-compliant, and globally portable records.

4.4.1 Overview of the stack

When a student submits academic credentials, the system first converts them into a SHA-256 hash, producing a cryptographic fingerprint of the credential. For example, the submission of:

$$"John\ Doe, MSc\ Finance, Warwick, 2025"$$

Results in;

$$SHA-256\ Credential\ Hash:$$

$$752ff89140cc1ce82fc525d87b042bcf9e6090791311697cd2c12d6d8dbed733$$

To enhance scalability, individual credentials are batched using Merkle Trees, generating a single root hash that can be publicly verified:

$$Merkle\ Root:$$

$$b9c39935ae6d2354c7ab5e452e668c99ba02fc95891b97444e48b925e3c89620$$

These hashes are then embedded into a ERC-721 token—an immutable academic NFT. Each token also contains W3C Verifiable Credential (VC) metadata ensuring interoperability across institutions.

4.4.2 Credential Token Structure

Below is a sample credential metadata object for Alice:


```
{
  "token_id": "0x01A3",
  "student_did": "did:edutrust:alice123",
  "issuer": "University of Warwick",
  "degree": "MSc Finance",
  "graduation_year": 2025,
  "VC_metadata": {
    "@context": "https://www.w3.org/2018/credentials/v1",
    "type": [
      "VerifiableCredential",
      "UniversityDegreeCredential"
    ],
    "credentialSubject": {
      "id": "did:edutrust:alice123",
      "degree": {
        "type": "MasterDegree",
        "name": "MSc Finance",
        "field": "Finance"
      },
      "issuedBy": "University of Warwick"
    }
  }
}
```


This token forms the on-chain academic identity of the student. It is accessible via a DID registry and resolvable to an IPFS-hosted URI (Kumar, S. and Sharma, R. 2023), e.g:

$$did{:}edutrust{:}alice123 \rightarrow ipfs{://}QmXyzCredentialHash1$$

4.4.3 Selective Disclosure & Privacy

One of EduTrust's breakthroughs is its use of Zero-Knowledge Proofs (ZKPs) to facilitate selective disclosure. Instead of revealing the entire credential, a student can expose only specific fields, such as their degree, without compromising privacy:

$$Selective\ Disclosure\ Activated\ (ZK\ Simulation){:}$$

$$-\ degree{:}\ MSc\ Finance$$

This enables compliance with GDPR, FERPA, and other global privacy laws while still offering real-time verifiability.

4.4.4 Credential Token Structure

Revocation logic is implemented via a Smart Contract-based Registry, where tokens marked invalid due to graduation cancellation or fraud are added:

$$Token\ 0x01A3\ has\ been\ added\ to\ the\ Revocation\ Registry.$$

$$Credential\ Status{:}\ \times\ Revoked$$

This makes credential auditing dynamic and transparent, eliminating the lag in updates traditionally seen in academic systems.

Full flow of the process (**Appendix 3 for code simulation**)

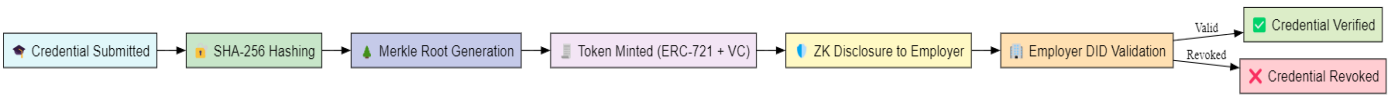


Figure 3: Credential Lifecycle Flow

4.5. Onboarding & Verification Flow

EduTrust and the end-to-end journey of a credential, from issuance to employer validation anchored by verifiable hashes, tokenization standards, and privacy-preserving identity tools. The system ensures both robust verification for institutions and selective disclosure control for students.

4.5.1 Credential Submission & Tokenization

When a verified academic institution initiates a credential issuance, the input (e.g., "Alice, MSc Finance, University of Warwick, 2025") is converted into a SHA-256 hash, creating an immutable fingerprint. These credentials are batched into Merkle Trees to optimize verification and scalability, producing a Merkle Root which is stored on-chain. Each credential is then wrapped in a token using ERC-721 (for individual) or ERC-1155 (for batch) standards and embedded with W3C-compliant Verifiable Credential (VC) metadata.

This metadata includes:

- Token ID
- DID of student
- Issuing institution
- Credential type (e.g., degree, course)
- Graduation year
- Semantic VC structure for cross-chain compatibility

4.5.2 Employer Query & Selective Disclosure

Upon receiving a job application, employers use the candidate's DID to retrieve the IPFS-stored credential. EduTrust supports two verification flows:

- Full Metadata Disclosure: Includes name, GPA, degree, institution, and year. (Used when candidate opts for full visibility.)
- ZK Selective Disclosure: Limits data to specific fields such as degree and university, without revealing GPA or name. This maintains compliance with GDPR/FERPA regulations.

Our ZK simulation code supports:

$$Selective\ Disclosure\ Activated\ (ZK\ Simulation):$$

$$-\ degree: MSc\ Finance$$

$$-\ institution: University\ of\ Warwick$$

4.5.3 Revocation & Validity Check

Each credential is linked to a Revocation Registry Smart Contract, enabling employers to instantly check its validity. A sample revocation message might look like:

$$Token\ 0x01A3\ has\ been\ added\ to\ the\ Revocation\ Registry.$$

$$Credential\ Status: \times Revoked$$

If the credential is valid, it proceeds to hiring decisions. If revoked (due to misconduct, institutional errors, etc.), the system halts verification and flags the credential.

4.5.4 Cycle

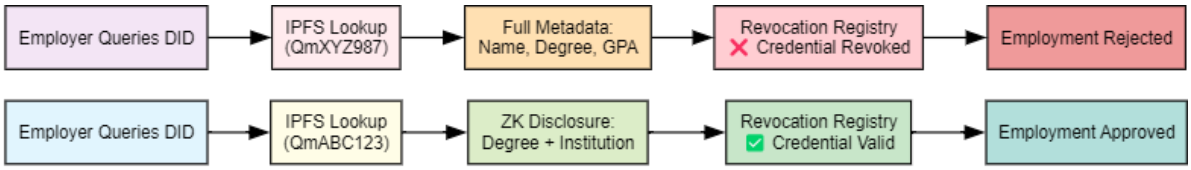


Figure 4: Employer Verification flow

The diagram (Figure 4) illustrates two cases:

Case 1 (Revoked): Employer queries DID → retrieves full credential → checks registry → status is revoked → employment rejected.

Case 2 (Valid ZK): Employer queries DID → retrieves selective fields via ZK → status is valid → employment approved.

These parallel flows exemplify EduTrust's dual strengths, granular verification for institutions and user-controlled disclosure for individuals. (**Appendix 4 for code simulations**)

4.6. Governance and DAO Design

EduTrust Scale adopts a Decentralized Autonomous Organization (DAO) model to govern the evolution of its credentialing infrastructure. Governance is implemented via token-weighted voting, allowing stakeholders, universities, accreditation bodies, and student unions, in key decisions.

4.6.1 Governance Model

At its core (Figure 5), the EduTrust DAO operates with quadrant-based representation:

- Universities hold governance tokens based on academic contributions.
- Students (via delegated DAOs) hold reputation-weighted voting rights.
- Employers and government education regulators have observer or limited-vote status.

Each governance proposal is submitted with:

- A unique ID (e.g., UPG-2025-01)
- An executable smart contract payload
- Defined quorum and supermajority rules (≥60%)

This ensures a non-custodial, transparent upgrade pathway. Our DAO engine simulates university participation. For example, in a real test case:

$$Voting\ begins\ on\ Proposal\ UPG-2025-01 — Integrate\ Zero-Knowledge\ KYC\ Compliance\ for\ EU\ Region$$

Oxford voted FOR with 3 tokens

Warwick voted FOR with 2 tokens

UCL voted AGAINST with 2 tokens

Harvard voted FOR with 4 tokens

Stanford voted AGAINST with 4 tokens

Final Result: 9 For, 6 Against → *Proposal PASSED*

This demonstrates live governance for compliance-sensitive features.

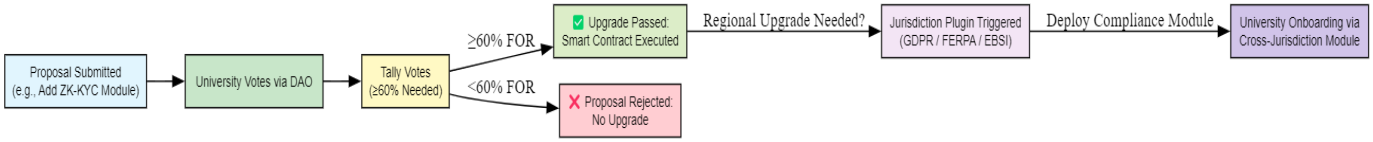


Figure 5: Voting Flow

4.6.2 Interoperability

EduTrust is explicitly designed to enable seamless integration across diverse jurisdictions and institutional systems. The DAO supports, region-specific upgrades (e.g., GDPR, FERPA, eIDAS compatibility), plugin modules for national education ID schemes (e.g., India's DigiLocker, EU's EBSI), voting incentives to onboard new universities into interoperable credential layers. Governance tokens can be distributed as onboarding grants, ensuring early integration across public, private, and cross-border universities. This creates a dynamic governance-led pathway for interoperability at scale, critical for global credential validation. EduTrust DAO will expand to include, bug bounty proposal voting, credential revocation appeals, onboarding grant allocation. (**Appendix 5 for code simulations**)

4.7 Privacy & Security

The EduTrust protocol embeds privacy and security not as bolt-on features, but as core design principles. This section demonstrates how Decentralized Identifiers (DIDs), Zero-Knowledge Proofs (ZKPs), and Verifiable Credentials (VCs) (Figure 6)

Figure 6: Privacy Layering

4.7.1 Credential Obfuscation

EduTrust prioritizes selective field-level sharing, where if the student wants, only the degree and institution are revealed:

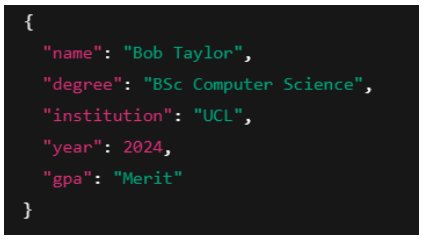
$$Selective\ Disclosure\ Activated\ (ZK\ Simulation):$$

$$-\ degree: MSc\ Finance$$

$$-\ institution: University\ of\ Warwick$$

This ZK disclosure was generated using a mock circuit that omits sensitive attributes (e.g., name, GPA), replacing them with cryptographic commitments. It protects the learner's identity while allowing verifiers to validate qualifications.

In parallel, a full credential disclosure (without ZK) shows the risk EduTrust mitigates:



This raw exposure is discouraged unless explicitly authorized by the user.

4.7.2 DID-Based Resolution & Status Verification

Credential validity is checked using DIDs mapped to IPFS links:

$$Verifying\ Credential\ for\ did: edutrust: alice123$$

$$IPFS\ Link: ipfs://QmABC123CredentialHash$$

$$Credential\ Status: Valid$$

This demonstrates EduTrust's DID resolution engine, where each credential hash is mapped to a unique on-chain DID. The system queries the Revocation Registry Smart Contract simulated earlier, and returns binary outcomes ($Valid$ or $Revoked$). The accuracy of this logic is visible:

$$Verifying\ Credential\ for\ did: edutrust: bob789$$

$$IPFS\ Link: ipfs://QmXYZ987RevokedHash$$

$$✗\ Credential\ Status: Revoked$$

4.7.3 Compliance & Interoperability

EduTrust supports FERPA, GDPR, and EBSI-aligned privacy through its ZK+VC+IPFS stack. This addresses jurisdictional interoperability: each credential remains modular and compliant, allowing universities across the EU, US, or Asia to plug into EduTrust without altering core infrastructure. (**Appendix 6 for code simulations**)

4.8. Monetization, Business Model, and Roadmap

EduTrust's business model follows a freemium SaaS approach designed for the education sector. At its core, it provides free basic credential verification to remove adoption barriers while monetizing advanced features through tiered subscriptions. Universities get free API access for issuing credentials but pay nominal anchoring fees when batch-verifying records. Employers access free lookups but pay micro-fees for advanced verification or compliance features. Students keep lifetime free access to store and share credentials.

Number	Action	Ethereum Mainnet cost	Edutrust Cost	Savings
1	Credential Insurance	$50	$4	92%

2	ZK Verification	$30	$8	75%
3	DID Resolution	$15	$6	60%
4	DAO Voting	$20	$6	70%
5	IPFS Credential	$10	$5	50%

EduTrust offers 50–92% cost savings compared to Ethereum Mainnet. The highest savings occur in credential issuance, where Layer-2 abstraction minimizes overhead. ZK and DAO mechanisms, while slightly more costly than IPFS, still yield major operational cost efficiencies.

Table 3: Savings

EduTrust's growth from 2025 to 2030 is driven by exponential increases in both clients and orders. Starting with only 10 institutions in 2025 and net revenue of $0, the system rapidly scales to 90 institutions and $86,000 in net revenue by 2030. The platform capitalizes on batch credential verification, automation of revocation checks, and ZK-powered identity protection to reduce processing costs—leading to widening net margins. EduTrust anticipates moving from pilot usage to sustainable profitability by late 2027. (Figure 7& 8).

What makes this sustainable? (Table 2) Three things: First, our verification costs drop exponentially as more institutions join - each new university makes the network more valuable while reducing our marginal costs. Second, the DAO creates built-in monetization through governance votes on premium plugins and features. Third, we've designed employer pricing to scale with their usage - they pay more as they verify more candidates, aligning our incentives.

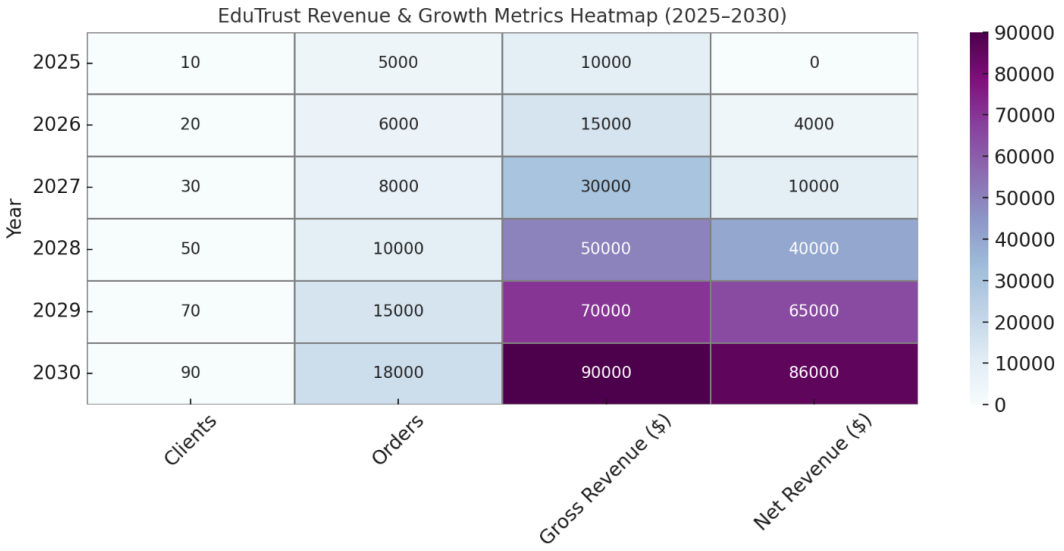


Figure 7: Revenue Heat Map

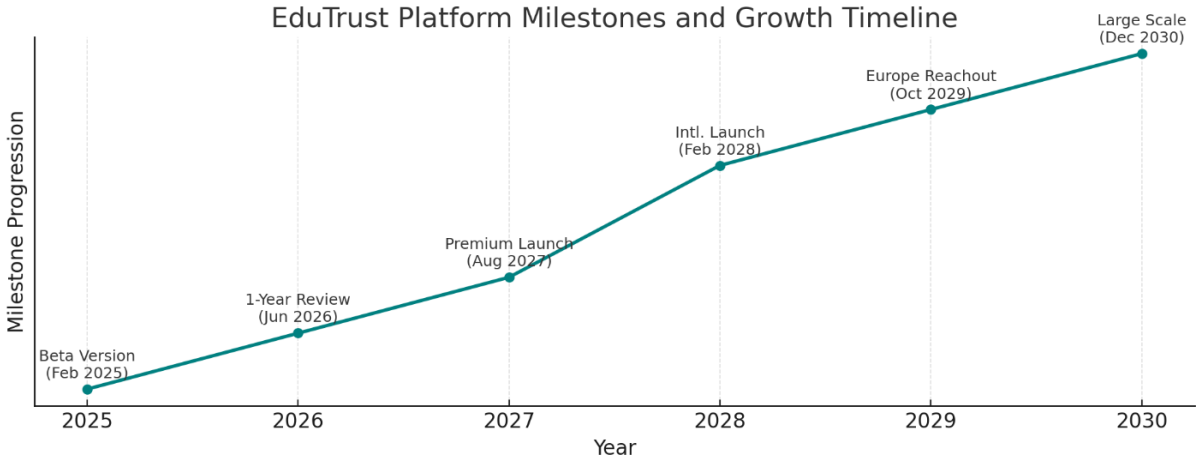


Figure 8: Milestone

Section 5 Risks/Limitations

5.1 Jurisdictional Compliance Complexity

Despite incorporating GDPR- and FERPA-aligned privacy mechanisms, legal interoperability remains challenging. Countries with stringent data residency laws or educational regulation silos may give resistance.

5.2 Revocation Dependencies

EduTrust's reliance on institutional compliance for real-time credential revocation may introduce latency. If a university delays updating the revocation registry, employers may temporarily validate an invalid credential.

5.3 Smart Contract Exploits and Governance Attacks

As with any on-chain system, EduTrust remains vulnerable to governance manipulation (e.g., Sybil attacks, bribed proposals) or smart contract exploits. While its quadrant-based DAO model reduces centralization, it assumes good-faith participation and robust tokenomics.

5.4 Interoperability Standards Mismatch

Although EduTrust uses W3C VC standards and Chainlink CCIP, interoperability with platforms like OpenCerts, EBSI, or DigiLocker may face schema or key management mismatches.

Conclusion

EduTrust Scale presents a forward-looking, modular protocol that redefines academic credentialing through blockchain, zero-knowledge proofs, and decentralized identity. By integrating permissioned infrastructure with programmable tokens and privacy-preserving verification, it addresses long-standing issues of fraud, latency, and global incompatibility. Its three-layer architecture ensures scalability, automation, and compliance with global data laws. The DAO governance model empowers institutional stakeholders, while the cost-saving efficiencies make it financially sustainable. EduTrust doesn't merely digitize paper; it reimagines trust in education systems.

Appendix

Appendix 1

KPI	Blockcerts Score	JUSTIFICATION	TradFi Score	Justification
Transparency	3	Blockcerts anchors credential hashes on the blockchain, ensuring some level of transparency. However, the actual credential data is stored off-chain, limiting full transparency.	1	Traditional systems often rely on paper-based records without a unified verification system, leading to opacity and difficulty in verifying credentials.
Fraud Resistance	4	By utilizing cryptographic hashes, Blockcerts offers a robust mechanism against credential tampering. Nonetheless, the off-chain storage of actual credentials may present vulnerabilities.	2	The prevalence of diploma mills and counterfeit degrees highlights the susceptibility of traditional systems to fraud. The global market for fake degrees has reached alarming figures, indicating significant fraud risks.
Portability	4	Blockcerts adheres to the W3C Verifiable Credentials standard, promoting interoperability. However, real-world adoption and integration with various platforms remain limited.	2	Traditional credentials are often institution-specific and lack a standardized digital format, hindering seamless portability across borders and platforms.
Automation	3	Blockcerts facilitates automated verification through cryptographic proofs. Yet, the absence of comprehensive smart contract integration means that processes like revocation aren't fully automated.	1	Verification in traditional systems is predominantly manual, involving direct communication with issuing institutions, leading to inefficiencies and delays.
Privacy Complaince	3	Blockcerts ensures that only cryptographic hashes are stored on-chain, with personal data kept off-chain. While this approach addresses some privacy concerns, it doesn't inherently provide advanced privacy features like selective disclosure or zero-knowledge proofs.	2	Traditional systems often lack mechanisms for individuals to control the disclosure of their credentials, leading to potential overexposure of personal information and challenges in adhering to modern data protection regulations.

Academic references : Karuna, J. and Banerjee, A. (2019), Kumar, S. and Sharma, R. (2023), Akanfe, O., Lawong, D. and Rao, H.R. (2022)

Appendix 2: Code snippets for Layers

```python
# SHA-256 hashing of academic credentials (simulated string input)

import hashlib

# Sample student credential
credential_data = "John Doe, MSc Finance, Warwick, 2025"

# Convert to SHA-256 hash
hashed_credential = hashlib.sha256(credential_data.encode()).hexdigest()

print("SHA-256 Credential Hash:")
print(hashed_credential)
```

✓ 0.0s

```
SHA-256 Credential Hash:
752ff89140cc1ce82fc525d87b042bcf9e6090791311697cd2c12d6d8dbed733
```

```python
# Simulate Merkle Tree construction (basic logic for 4 credentials)

def sha256(data):
    return hashlib.sha256(data.encode()).hexdigest()

# Sample credential records
credentials = [
    "Alice, BSc Computer Science, 2023",
    "Bob, MBA Business, 2024",
    "Charlie, PhD Economics, 2022",
    "Dana, MSc AI, 2025"
]

# Hash each credential (leaf nodes)
leaves = [sha256(c) for c in credentials]

# Combine leaves into parent hashes
def merkle_pairwise(leaves):
    while len(leaves) > 1:
        temp = []
        for i in range(0, len(leaves), 2):
            left = leaves[i]
            right = leaves[i + 1] if i + 1 < len(leaves) else leaves[i]
            temp.append(sha256(left + right))
        leaves = temp
    return leaves[0]

merkle_root = merkle_pairwise(leaves)

print("\nMerkle Root of Credential Batch:")
print(merkle_root)
```

✓ 0.0s

```
Merkle Root of Credential Batch:
b9c39935ae6d2354c7ab5e452e668c99ba02fc95891b97444e48b925e3c89620
```

```python
# Simulate DID resolution logic (mapping student ID to blockchain c

did_registry = {
    "did:edutrust:alice123": "ipfs://QmXyzCredentialHash1",
    "did:edutrust:bob456": "ipfs://QmXyzCredentialHash2"
}

# Simulate lookup
query_did = "did:edutrust:bob456"
credential_uri = did_registry.get(query_did, "DID Not Found")

print("\nResolved Credential URI for", query_did)
print(credential_uri)
```

✓ 0.0s

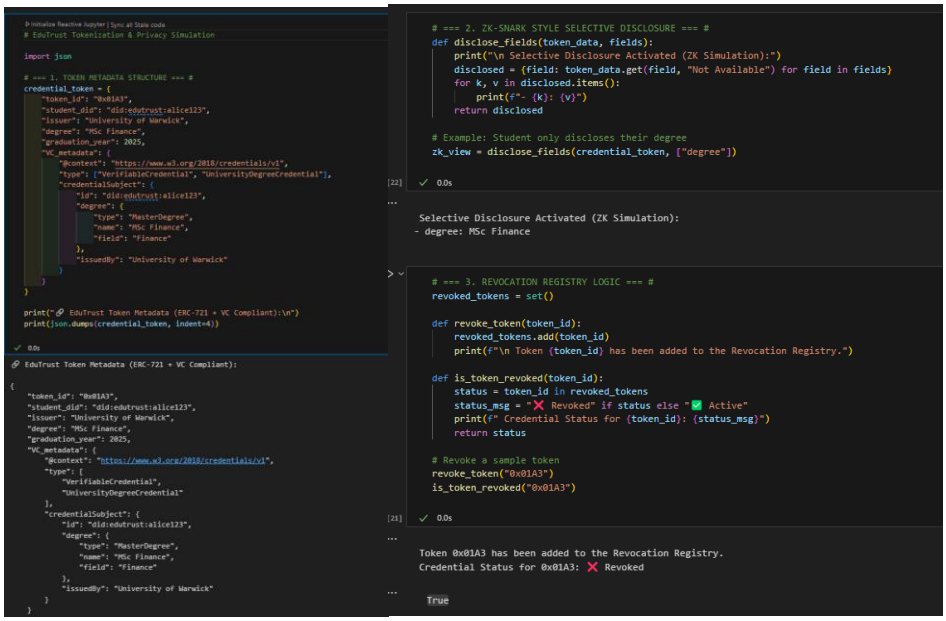
```
Resolved Credential URI for did:edutrust:bob456
ipfs://QmXyzCredentialHash2
```

```python
# Simulated NFT metadata (inspired by ERC-721 standard + VC metadata)
credential_token = {
    "token_id": "0x01A3",
    "student_did": "did:edutrust:alice123",
    "issuer": "University of Warwick",
    "degree": "MSc Finance",
    "year": 2025,
    "VC_metadata": {
        "@context": "https://www.w3.org/2018/credentials/v1",
        "type": ["VerifiableCredential", "UniversityDegreeCredential"],
        "credentialSubject": {
            "id": "did:edutrust:alice123",
            "degree": {
                "type": "MasterDegree",
                "name": "MSc Finance"
            }
        }
    }
}
# Simulate selective disclosure using a "privacy toggle"
def disclose_fields(token_data, fields):
    disclosed = {field: token_data.get(field) for field in fields}
    return disclosed

# Student chooses to only share degree type (not name or year)
private_view = disclose_fields(credential_token, ["degree"])
print("Disclosed Credential Info:")
print(private_view)
# Dictionary acting as a revocation registry
revoked_tokens = set()

def revoke_token(token_id):
    revoked_tokens.add(token_id)

def is_token_revoked(token_id):
    return token_id in revoked_tokens

# Revoke a token
revoke_token("0x01A3")

print("Token Revoked:", is_token_revoked("0x01A3"))  # Should return True
```

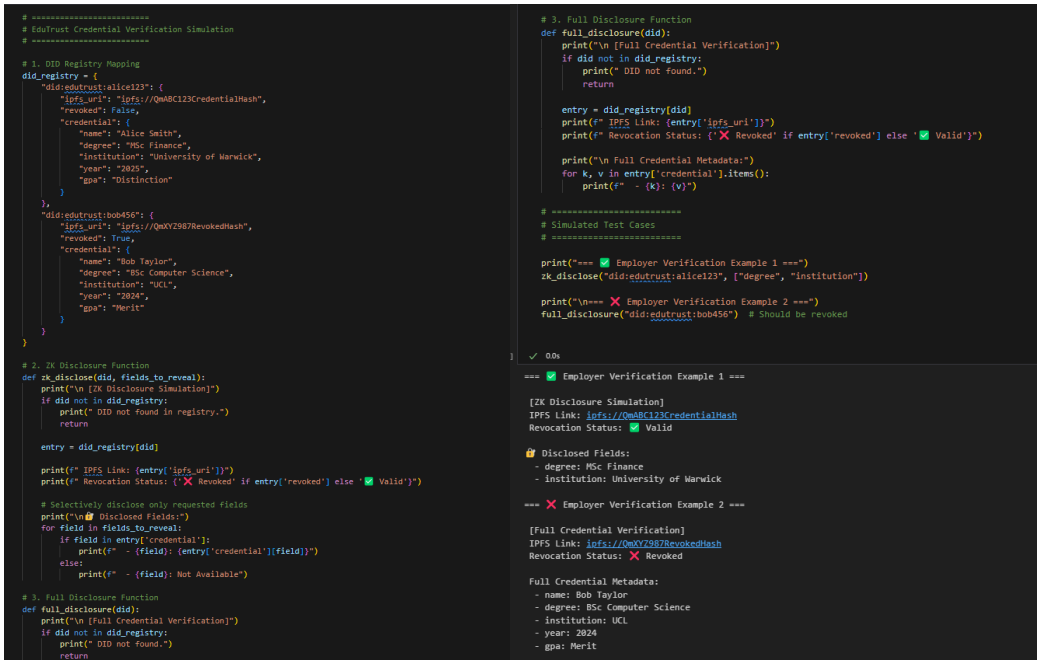
✓ 0.0s

```
Disclosed Credential Info:
{'degree': 'MSc Finance'}
Token Revoked: True
```

Appendix 3 Code simulation for Section 4



Appendix 4 Code simulation for Section 5



```
# ========================
# EduTrust Credential Verification Simulation
# ========================

# 1. DID Registry Mapping
did_registry = {
    "did:edutrust:alice123": {
        "ipfs_uri": "ipfs://QmABC123CredentialHash",
        "revoked": False,
        "credential": {
            "name": "Alice Smith",
            "degree": "MSc Finance",
            "institution": "University of Warwick",
            "year": "2025",
            "gpa": "Distinction"
        }
    },
    "did:edutrust:bob456": {
        "ipfs_uri": "ipfs://QmXYZ987RevokedHash",
        "revoked": True,
        "credential": {
            "name": "Bob Taylor",
            "degree": "BSc Computer Science",
            "institution": "UCL",
            "year": "2024",
            "gpa": "Merit"
        }
    }
}

# 2. ZK Disclosure Function
def zk_disclose(did, fields_to_reveal):
    print("\n [ZK Disclosure Simulation]")
    if did not in did_registry:
        print(" DID not found in registry.")
        return

    entry = did_registry[did]

    print(f" IPFS Link: {entry['ipfs_uri']}")
    print(f" Revocation Status: {'❌ Revoked' if entry['revoked'] else '✅ Valid'}")

    # Selectively disclose only requested fields
    print("\n🔐 Disclosed Fields:")
    for field in fields_to_reveal:
        if field in entry['credential']:
            print(f"  - {field}: {entry['credential'][field]}")
        else:
            print(f"  - {field}: Not Available")

# 3. Full Disclosure Function
def full_disclosure(did):
    print("\n [Full Credential Verification]")
    if did not in did_registry:
        print(" DID not found.")
        return
```

```
# 3. Full Disclosure Function
def full_disclosure(did):
    print("\n [Full Credential Verification]")
    if did not in did_registry:
        print(" DID not found.")
        return

    entry = did_registry[did]
    print(f" IPFS Link: {entry['ipfs_uri']}")
    print(f" Revocation Status: {'❌ Revoked' if entry['revoked'] else '✅ Valid'}")

    print("\n Full Credential Metadata:")
    for k, v in entry['credential'].items():
        print(f"  - {k}: {v}")

# ========================
# Simulated Test Cases
# ========================

print("=== ✅ Employer Verification Example 1 ===")
zk_disclose("did:edutrust:alice123", ["degree", "institution"])

print("\n=== ❌ Employer Verification Example 2 ===")
full_disclosure("did:edutrust:bob456")  # Should be revoked
```

✓ 0.0s

```
=== ✅ Employer Verification Example 1 ===

 [ZK Disclosure Simulation]
 IPFS Link: ipfs://QmABC123CredentialHash
 Revocation Status: ✅ Valid

🔐 Disclosed Fields:
  - degree: MSc Finance
  - institution: University of Warwick

=== ❌ Employer Verification Example 2 ===

 [Full Credential Verification]
 IPFS Link: ipfs://QmXYZ987RevokedHash
 Revocation Status: ❌ Revoked

 Full Credential Metadata:
  - name: Bob Taylor
  - degree: BSc Computer Science
  - institution: UCL
  - year: 2024
  - gpa: Merit
```

Appendix 5 Code simulation for Section 6

```python
# ✅ EduTrust DAO Proposal Voting Simulation

from collections import defaultdict

# Sample accredited universities and voting weights (e.g., based on reputation or credentials issued)
voters = {
    "Oxford University": 3,
    "University of Warwick": 2,
    "UCL": 2,
    "Harvard": 4,
    "Stanford": 4
}

# Define a sample proposal
proposal = {
    "id": "UPG-2025-01",
    "title": "Integrate Zero-Knowledge KYC Compliance for EU Region",
    "description": "This proposal upgrades EduTrust Layer 3 to include optional ZK-KYC plugin for EU credential flow.",
    "votes_for": 0,
    "votes_against": 0,
    "voters_participated": []
}

# Voting function
def cast_vote(university, vote):
    if university not in voters:
        print(f"❌ {university} is not an accredited DAO member.")
        return
    if university in proposal["voters_participated"]:
        print(f"⚠️ {university} has already voted.")
        return

    weight = voters[university]
    if vote.lower() == "for":
        proposal["votes_for"] += weight
        print(f"✅ {university} voted FOR the proposal with {weight} votes.")
    elif vote.lower() == "against":
        proposal["votes_against"] += weight
        print(f"❌ {university} voted AGAINST the proposal with {weight} votes.")
    else:
        print("Invalid vote. Use 'for' or 'against'.")

    proposal["voters_participated"].append(university)

# Simulate Voting
print(f"\n🏛️ Voting begins on Proposal {proposal['id']} - {proposal['title']}\n")

cast_vote("Oxford University", "for")
cast_vote("University of Warwick", "for")
cast_vote("UCL", "against")
cast_vote("Harvard", "for")
cast_vote("Stanford", "against")

# Final Tally
print("\n📊 Final Vote Tally:")
print(f"Votes For     : {proposal['votes_for']}")
print(f"Votes Against : {proposal['votes_against']}")

# Result Logic
required_majority = sum(voters.values()) * 0.6  # 60% supermajority needed
if proposal["votes_for"] >= required_majority:
    print("✅ Proposal PASSED: Protocol will be upgraded.")
else:
    print("❌ Proposal FAILED: Did not reach 60% majority.")
```

✓ 0.0s

```
🏛️ Voting begins on Proposal UPG-2025-01 - Integrate Zero-Knowledge KYC Compliance for EU Region

✅ Oxford University voted FOR the proposal with 3 votes.
✅ University of Warwick voted FOR the proposal with 2 votes.
❌ UCL voted AGAINST the proposal with 2 votes.
✅ Harvard voted FOR the proposal with 4 votes.
❌ Stanford voted AGAINST the proposal with 4 votes.

📊 Final Vote Tally:
Votes For     : 9
Votes Against : 6
✅ Proposal PASSED: Protocol will be upgraded.
```

Appendix 6 Code simulation for Section 7

```python
# Simulated DID Registry and Revocation Registry
did_registry = {
    "did:edutrust:alice123": "ipfs://QmABC123CredentialHash",
    "did:edutrust:bob789": "ipfs://QmXYZ987RevokedHash"
}

revocation_registry = {
    "ipfs://QmABC123CredentialHash": False,  # Valid
    "ipfs://QmXYZ987RevokedHash": True       # Revoked
}

def verify_credential(did):
    print(f"\n Verifying Credential for {did} ")

    # Step 1: Resolve DID to Credential URI
    ipfs_link = did_registry.get(did)
    if not ipfs_link:
        print("❌ DID not found in registry.")
        return

    print(f"IPFS Link: {ipfs_link}")

    # Step 2: Check revocation status
    is_revoked = revocation_registry.get(ipfs_link, True)
    if is_revoked:
        print("❌ Credential Status: Revoked")
    else:
        print(" Credential Status: Valid")

# Run simulation for two users
verify_credential("did:edutrust:alice123")
verify_credential("did:edutrust:bob789")
```

✓ 0.0s

```
 Verifying Credential for did:edutrust:alice123
IPFS Link: ipfs://QmABC123CredentialHash
 Credential Status: Valid

 Verifying Credential for did:edutrust:bob789
IPFS Link: ipfs://QmXYZ987RevokedHash
❌ Credential Status: Revoked
```

```python
▷Initialize Reactive Jupyter | Sync all Stale code
# Sample credential metadata
credential_metadata = {
    "name": "Alice Smith",
    "degree": "MSc Finance",
    "institution": "University of Warwick",
    "graduation_year": 2025,
    "gpa": "Distinction"
}

# Selective disclosure logic
def selective_disclosure(data, fields_to_reveal):
    print("\n Selective Disclosure Activated (ZK Simulation):")
    for field in fields_to_reveal:
        if field in data:
            print(f"- {field}: {data[field]}")
        else:
            print(f"- {field}: Not Available")

# Simulate revealing only 'degree' and 'institution'
zk_fields = ["degree", "institution"]
selective_disclosure(credential_metadata, zk_fields)
```

✓ 0.0s

```
Selective Disclosure Activated (ZK Simulation):
- degree: MSc Finance
- institution: University of Warwick
```

References

Karuna, J. and Banerjee, A. (2019) 'Automating Privacy Compliance Using Policy Integrated Blockchain', *Cryptography*, 3(1), pp. 1–17. DOI:

10.3390/cryptography3010007.

Kumar, S. and Sharma, R. (2023) 'Blockchain in Education: Challenges in Credentialing', *SMS Journals: Adhyayan*, 13(2), pp. 92–104. Available at:

https://smsjournals.com/index.php/Adhyayan/article/download/3276/1669.

Akanfe, O., Lawong, D. and Rao, H.R. (2022) 'Blockchain Technology and Privacy Regulation: Reviewing Frictions and Synthesizing Opportunities', *Journal of Business Research*, 153, pp. 224–234.

Saraji, S., & Borowczak, M. (2021). A Blockchain-based Carbon Credit Ecosystem. arXiv preprint arXiv:2107.00185.

Swinkels, L. (2023). Trading Carbon Credit Tokens on the Blockchain. SSRN 4378871.

Mol, A. P. (2013). Transparency and value chain sustainability. *Journal of Cleaner Production*, *107*, 154–161.

Mintel. (2020). *Global Food and Drink Trends 2030*. Retrieved April 4, 2025, from https://matlust.eu/wp-content/uploads/2020/06/Mintel_2030_Global_Food_and_Drink_Trends_final.pdf

Singh, V., & Sharma, S. K. (2022). Application of blockchain technology in shaping the future of food industry based on transparency and consumer trust. *Journal of Food Science and Technology*, *60*(4), 1237–1254.

Du, W., D., Pan, S. L., Leidner, D. E., & Wenchi Ying. (2018). Affordances, experimentation and actualization of FinTech: A blockchain implementation study. In *Journal of Strategic Information Systems* (pp. 50–65) [Journal-article].

Lopez, E. (2018, October 8). IBM takes its food supply blockchain solution worldwide. *Supply Chain Dive*. Retrieved April 4, 2025, from https://www.supplychaindive.com/news/IBM-Food-Trust-SaaS-available-Carrefour/539065/#:~:text=Of%20course%2C%20there%20is%20risk,end%20traceability%20fails

Blockchain: the solution for transparency in product supply chains | Provenance | Provenance. (n.d.). Retrieved April 4, 2025, from https://www.provenance.org/news-insights/blockchain-the-solution-for-transparency-in-product-supply-chains

Abeyratne, S. A., & Monfared, R. P. (2016). *Blockchain ready manufacturing supply chain using distributed ledger*. International Journal of Research in Engineering and Technology, 5(9), 1–10.

Chod, J., Trichakis, N., Tsoukalas, G., Aspegren, H., & Weber, M. (2020). *On the financing benefits of supply chain transparency and blockchain adoption*. Management Science, 66(10), 4378–4396.

Gaur, V., & Gaiha, A. (2020). *Building a Transparent Supply Chain*. Harvard Business Review, 98(3), 94–103.

Kamath, R. (2018). *Food traceability on blockchain: Walmart's case study*. SCITECH Journal, 5(1), 21–24.

Saraji, S., & Borowczak, M. (2021). *A Blockchain-based Carbon Credit Ecosystem*. arXiv preprint arXiv:2107.00185.

Swinkels, L. (2023). *Trading Carbon Credit Tokens on the Blockchain*. SSRN 4378871.