Project 1

TrustChainKYC:

A Decentralised KYC Protocol for Privacy-Preserving Cross-Border Verification

By Group 1

1. Rowida Abdelrahhman Elhawary
2. Khaled Hassan Abdelhafez
3. Fouad Shawky
4. Abdelrahman Amin
5. Ahmed Mahmoud Hamza Aly Ahmed

Supervised by

Dr. Muhammad A. Othman

This paper was submitted in partial fulfilment of the requirements of the

Managing Successful Fintech & Blockchain Program Capstone Project FT 624 Course

ESLSCA PGD FT04

July 2025

Acknowledgement

We thank Dr Muhammad A. Othman for guidance throughout the capstone, ESLSCA University for its academic support and learning environment, and our classmates for their reviews, testing, and constructive feedback. We also appreciate the industry mentors who provided practical insights on onboarding, compliance, and privacy.

Warm regards,

The researchers

Executive Summary

KYC processes are duplicated across different institutions and jurisdictions, which increases onboarding times and costs while exposing personal data to unnecessary risk. TrustChainKYC introduces a decentralised protocol that allows financial institutions to trust and reuse verified identity attestations without sharing the underlying documents. The design integrates: a permissioned consortium blockchain (such as Hyperledger Fabric or enterprise Ethereum), W3C Decentralised Identifiers (DIDs), Verifiable Credentials (VCs), off-chain encrypted storage (IPFS), and zero-knowledge proofs (zk-SNARKs/zk-STARKs) for selective disclosure.

The platform enables a user to complete KYC once with a participating institution (Issuer). That institution signs a VC and records a cryptographic commitment on-chain. Any other institution (Relying Party) can request proof of attributes (e.g., “over 18”, “resides in EU”, “KYC verified <12 months”) and receive a privacy-preserving proof verified by a smart contract.

Consent is explicit, revocable, and recorded on-chain. No personal data is stored on the ledger; only hashes, commitments, and events are kept.

We outline the architecture, governance, compliance mapping (GDPR/FATF), smart-contract and ZKP design, a prototype implementation plan, and evaluation measures. The expected outcomes include: reduced duplication, faster onboarding, enhanced privacy by design, and interoperability across borders.

Table of Contents

List of Abbreviations

Abbreviations	Item
AML	Anti-Money Laundering
DID	Decentralised Identifier
DPIA	Data Protection Impact Assessment
EIDAS	Electronic Identification, Authentication and Trust Services
FATF	Financial Action Task Force
GDPR	General Data Protection Regulation
IPFS	InterPlanetary File System
KYC	Know Your Customer
PII	Personally Identifiable Information
RP	Relying Party (institution requesting verification)
SSI	Self-Sovereign Identity
VC	Verifiable Credential
VP	Verifiable Presentation
ZK	Zero-Knowledge
zk-SNARK/zk-STARK	Zero-knowledge succinct/non-interactive proofs
PII	Anti-Money Laundering

List of Tables

Table of Figures

1. Introduction

Customer due diligence and KYC are essential to mitigate money-laundering and terrorism-financing risks (Hanif et al., 2025; Inaltong, 2025; Ismail et al., 2025). Yet, globally, institutions repeat the same verifications for the same customer, leading to extended onboarding timelines, elevated operational costs, and increased exposure of sensitive documents (Fugkeaw et al., 2025; Xiong et al., 2025). Cross-border activity adds jurisdictional complexity (data residency, localisation and transfer restrictions), creating further friction for legitimate clients (Omar & Khan, 2025).

Blockchains offer shared truth without a central operator; SSI (DIDs/VCs) offers user-centric identity with cryptographic attestations (Femenias et al., 2025; Nguyen et al., 2025; Wang et al., 2025). Combining these with modern ZK proofs enables a design where attestations are reusable, evidence remains off-chain, and only the minimum required facts are disclosed (Abou et al., 2025; Eshan et al., 2025). This report details such a design and its path to a working prototype for the FT624 capstone.

2. Project Objectives

1. **Eliminate KYC duplication** through blockchain-enabled verified identity reusability.
2. **Accelerate onboarding** by creating a shared, interoperable identity ledger.
3. **Ensure compliance** with GDPR and international data sharing standards via zero-knowledge proofs (zk-SNARKs).
4. **Enable federated collaboration** among financial institutions through a permissioned consortium blockchain.

3. Problem Statement

- Duplication: Multiple institutions repeat KYC for the same user; documents are re-collected and re-validated.
- Latency: Onboarding can take days to weeks, affecting user experience and revenue recognition.
- Privacy risk: Centralised repositories are breach-prone; documents are overshared.
- Interoperability gaps: Lack of a common trust fabric across borders and sectors.

4. Related Work & Theoretical Foundations

- Centralised KYC Utilities reduce duplication but create new honeypots and governance issues (Dhanorkar et al., 2025; Hamza & Smolander, 2025).
- SSI & VCs (W3C) define portable credentials signed by issuers and presented to RPs (Kaiiali et al., 2025; Vanella, 2025).
- Blockchain provides immutable anchoring of commitments and audit events without storing PII (Ahmed et al., 2025; Bureacă et al., 2025).
- Zero-Knowledge enables attribute proofs without revealing raw data (e.g., over-18 without date of birth) (Huang et al., 2025; Mikołajczyk et al., 2025).
- This project applies these principles in a consortium model suitable for regulated finance.

5. Requirements

5.1 Functional

- F1 User enrolment with DID creation and wallet binding.
- F2 Issuer verifies KYC and issues VC; cryptographic commitment anchored on-chain.
- F3 Consent workflow: grant/revoke to named RPs with expiry and scope.
- F4 RP requests proof; user supplies VP/zk-proof via wallet; smart contract verifies.
- F5 Audit log events: consent, issuance, revocation, proof-verified.
- F6 Revocation/expiry lifecycle for credentials.
- F7 Regulator read-only view for oversight (no PII).

5.2 Non-Functional

- Privacy by design; data minimisation; purpose limitation.
- High availability of verifier endpoints; deterministic verification (<3s) for proofs.
- Interoperability with DID/VC standards; portability of wallets.
- Observability and tamper-evident logs.
- Cryptographic agility (ability to upgrade circuits and keys).

6. Technology Stack Justification

Architecture & Components

Layer	**Tech / Tool**	**Rationale**
Blockchain Platform	Hyperledger Fabric (or Ethereum)	Fabric for permissioned control & modularity; Ethereum for public testnets
Smart Contracts	Solidity (Ethereum) / Chaincode (Fabric)	Enforces access, consent, KYC verification
Storage	IPFS	Secure off-chain encrypted document storage

Privacy Layer	zk-SNARKs or zk-STARKs	Selective disclosure + data minimisation
Backend	Node.js + Express.js	API gateway for blockchain interaction
Frontend	React.js	Responsive client portal for institutions & users
Consortium Identity	DID + Verifiable Credentials (W3C)	Supports decentralised identifiers & interoperability

Table 1 Architecture & Components by the researchers.

7. Architecture Overview (High-Level)

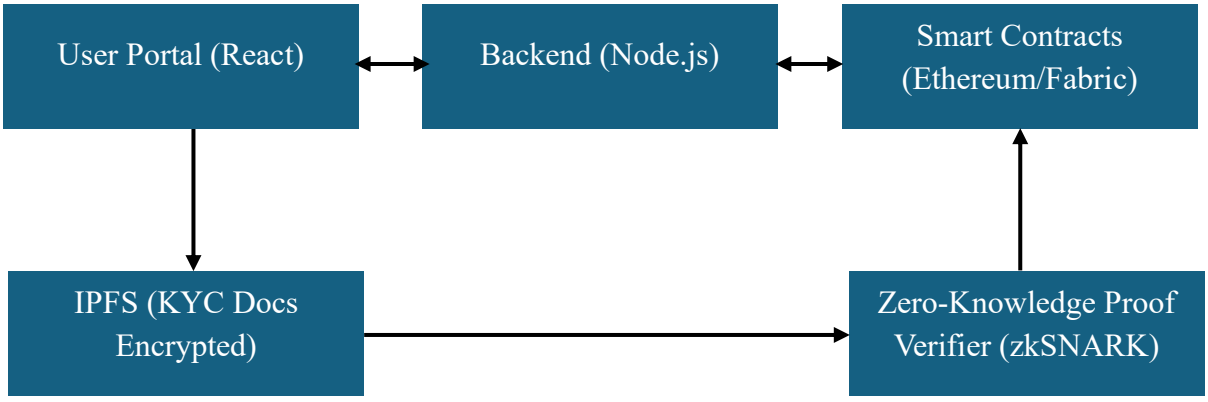

Figure 1 Architecture of TrustChainKYC by the researchers.

8. Key Functionalities

- **User Identity Creation**: Generates DID and stores verifiable credentials.
- **KYC Verification**: Institutions validate identity and sign with their key.
- **Consent Management**: User gives/revokes access to institutions (via smart contract).
- **Selective Disclosure**: Based on zk-SNARK, share only necessary data.
- **Audit Log**: Immutable log for data access & consent actions (visible only to data subject and regulators).

9. Data Model & Flows

9.1 Core On-Chain Data (no PII)

- `credentialCommitment`: Poseidon/Keccak hash of VC payload.
- `revocationRegistry`: bitmap or Merkle tree of revoked indices.
- `consentRecord`: mapping (user DID, RP DID, scope, expiry) → state.
- `events`: `Issued`, `Revoked`, `ConsentGranted`, `ConsentRevoked`, `ProofVerified`.

9.2 Key Flows (Narrative)

- **Enrolment & Issuance**: User provides evidence to Issuer; Issuer validates; generates VC; encrypts and stores artefacts in IPFS; posts `credentialCommitment` to chain.
- **Consent**: User signs a consent VP specifying RP, scope and time window; contract updates `consentRecord`.
- **Verification**: RP sends request; user wallet generates VP/zk-proof; verifier contract checks ZK proof, consent scope, revocation status and timestamp; emits `ProofVerified` event.
- **Revocation/Expiry**: Issuer updates revocation registry; proofs check membership to confirm validity.

10. Compliance Strategy

- **GDPR Alignment**: Off-chain data storage with user revocation rights.
- **Immutability vs Privacy**: Use hashes and pointers only on-chain.
- **Data Minimisation**: Share only attestations, not raw documents.
- **Governance**: consortium charter (membership criteria, SLAs, key ceremonies, audit processes, dispute resolution).

11. Evaluation Criteria

Area	How it will be Demonstrated
Security	Penetration testing + ZKP implementation validation
Interoperability	Demo with multiple institution roles (Bank, Fintech, Regulator)
UI/UX	Walkthrough of user onboarding and KYC request/approval flow
Code Quality	Documented smart contracts + GitHub repo link

Table 2 Evaluation Criteria by the researchers.

12. Smart Contracts Design

12.1 Contracts

- `KYCRegistry`: stores credential commitments; manages revocations.
- `ConsentManager`: user-controlled consent states (grant/revoke with scope & expiry).
- `Verifier`: ZK verifier (generated from circuit); pure function returning boolean; emits events.

12.2 Key Functions (pseudocode)

issueCommitment(issuer, holderDID, commitment, expiry)

revokeCredential(issuer, commitment)

grantConsent(holder, rpDID, scopeHash, expiry)

revokeConsent(holder, rpDID, scopeHash)

verifyProof(rpDID, proof, publicSignals) -> bool

12.3 Events

`Issued`, `Revoked`, `ConsentGranted`, `ConsentRevoked`, `ProofVerified`.

12.4 Security

role-based access (Issuers only for issue/revoke), EIP-712 typed data for off-chain signatures, pausable & upgradable via proxy pattern in pilots.

13. Zero-Knowledge Proof Mechanisms

- **Proof Types:** (a) Age ≥ X from DOB; (b) Residency in jurisdiction set; (c) "KYC-verified within N days" from issuance date; (d) Liveness bound to credential public key.
- **Circuits:** Poseidon-based Merkle membership for commitment; range proofs for age; set-membership for country codes; time-window checks.
- **Selective Disclosure:** RP requests minimal predicates; wallet compiles circuit with only necessary public signals.
- **Revocation Checking:** proof includes non-membership in revocation registry at block height t.
- **Performance:** client-side proving for small predicates (<5s on laptop/modern phone); server-assisted proving available in pilot.

14. Off-Chain Storage & Key Management

- **Encrypt-then-Store:** artefacts encrypted with user-controlled keys; IPFS CID stored in encrypted VC only.
- **Key Management:** wallet-based keys with social recovery; optional institutional custodianship for retail users.
- **Rotation:** re-encrypt on key rotation; commitments unaffected.
- **Pinning:** consortium IPFS pinning services; retention under policy.
- **Data Deletion:** delete encrypted blobs upon erasure request; retain non-identifying on-chain events.

15. Interoperability & Standards

- W3C DID & VC Data Model; OIDC4VP for transport; WACI-DIDComm optional for secure interactions.
- Schema: KYC-Basic (name hash, DOB hash, nationality code, risk tier, issuance/expiry).
- Financial Messaging: optional mapping to ISO 20022 elements for downstream systems.
- Cross-Chain: anchors can be mirrored across Fabric↔Ethereum via relays if needed.

16. Security & Threat Model

Assets: credential commitments, revocation registries, consent states, keys.

Adversaries include malicious RP/Issuer, external attackers, insiders, curious consortium nodes, and colluding parties.

Threats & Controls

- Replay of proofs → include nonce, RP DID and expiry in public signals.
- Key theft → hardware-backed wallets; social recovery; anomaly detection.
- Corrupt Issuer → issuer reputation, slashing/de-listing, regulator oversight.
- Linkability → unlinkable presentations; fresh DIDs; minimal public signals.
- Node compromise → HSM for validator keys; audit trails; least-privilege ops.
- Denial of service → rate limiting; circuit quotas; autoscaling verifiers.
- Pen-Test Plan: static analysis of contracts; fuzzing; ZK verifier tests; red-team exercises; DPIA findings addressed prior to production.

17. Implementation Plan & DevOps

- **Stack**: React + Node/Express; Fabric (v2.x) or Quorum/Go-Ethereum (permissioned); IPFS; ZoKrates/snarkJS.
- **Environments**: Dev (local), Test (Docker compose), Pilot (Kubernetes).
- **CI/CD**: linting, unit tests (Jest/Mocha), contract tests (Hardhat/Truffle), security scans (Slither/Mythril).

- **Observability**: Prometheus metrics; ELK logs; chain explorer; audit dashboards.
- **Documentation**: OpenAPI specs; run-books; key ceremonies SOP.

18. UI/UX Overview

- **User Wallet**: consent screen (scopes, RP, duration), "show proof" flow, activity log.
- **Issuer Console**: verify & issue VC; anchor commitment; revoke; reports.
- **RP Portal**: request predicate(s); verification result with reference ID.
- **Regulator View**: read-only events, issuer list, policy snapshots.
- **Accessibility**, clear copy, and audit transparency are prioritised.

19. Evaluation & Testing Plan

Criterion	Method	Target
Security	Contract audits, verifier unit tests	0 criticals outstanding
Privacy	DPIA; data-flow review	No PII on-chain confirmed
Performance	Proof verify latency	≤ 3s median
Interoperability	End-to-end with two RPs, one Issuer	100% pass
Usability	Task-completion time (3 tasks)	≤ 5 min avg

Table 3 Evaluation & Testing Plan by the researchers.

20. Business Case & Adoption

- **Value**: lower onboarding cost, faster revenue capture, improved compliance posture, better user control.
- **Stakeholders**: banks, fintechs, payment providers, and regulators.
- **Incentives**: reduced repeat KYC, shared utilities, compliance analytics, and reputational benefits.
- **Operating Model**: membership fees, tiered API SLAs, and an optional managed verifier service.
- **Go-to-Market**: start with bilateral corridors (e.g., Egypt–EU fintech), expand to regional networks.

21. Risks & Mitigations

Risk	Category	Mitigation
Immutability vs erasure	Legal	Keep PII off-chain; commitments non-identifying; delete off-chain blobs
ZK integration complexity	Technical	Start with proven libraries; limit predicates in pilot
Consortium bootstrapping	Governance	Clear charter; early anchor members; regulator MOU
Issuer reliability variance	Operational	Accreditation; SLAs; audits; slashing/de-listing
User key loss	UX	Social recovery; custodial options; backup guidance

Table 4 Risks & Mitigations by the researchers.

22. Deliverables

1. **Web App Demo** with multiple user types.
2. **Smart Contracts Repository** with documentation.
3. **Whitepaper (8–15 pages)**: Includes architecture, threat model, protocols, diagrams.
4. **Demo Video (3–5 min)**: Simulates onboarding and cross-institution sharing.
5. **Slide Deck (PPT)** for 5 Sept presentation.

23. Limitations & Future Work

- Expand proof set (income band, sanctions screening attestations).
- Hardware-backed wallets on mobile; passkeys.
- Cross-chain anchoring and interoperability with national digital ID schemes.
- Privacy-preserving analytics for regulators.

24. Milestone Plan

Milestone	Date
Finalise Architecture & Tools	20 July
Smart Contract Development Begins	21 July
IPFS + zk-SNARK Setup	25 July
Frontend + Backend Integration	1 August
Internal Demo	15 August
Whitepaper Draft + Pen Testing	25 August
Demo Video + Final Polishing	1 September
Submit Final Project + PPT	4 September
Project Presentation Day	5 September

Table 5 Milestone Plan by the researchers.

25. Conclusion

TrustChainKYC demonstrates a practical path to reusable, privacy-preserving KYC across borders. By anchoring attestations, not documents, on a permissioned ledger, and leveraging DIDs/VCs with ZK proofs, the design improves onboarding speed and privacy while aligning with regulatory expectations. The outlined architecture and plan are implementation-ready for a pilot.

References

Abou, Z., Houda, E., Eren, H., Karaduman, Ö., Tuncay, M., & Glu, G. ˇ. (2025). Security Challenges and Performance Trade-Offs in On-Chain and Off-Chain Blockchain Storage: A Comprehensive Review. *Applied Sciences 2025, Vol. 15, Page 3225*, *15*(6), 3225. https://doi.org/10.3390/APP15063225

Ahmed, I., Toyoda, K., Nakano, T., Kasahara, S., Goyal, S. R., & Tran, T. H. (2025). A Systematic Review on Blockchain-Enabled eKYC: Leveraging SSI and DID for Secure and Efficient Identity Verification. *IEEE Internet of Things Journal*, 1–1. https://doi.org/10.1109/JIOT.2025.3597356

Bureacă, E., Leancă, R. A., Ciobanu, I., Brînzea, A., & Aciobăniței, I. (2025). Unlinkable Revocation Lists for Qualified Electronic Attestations: A Blockchain-Based Framework. *Electronics 2025, Vol. 14, Page 2795*, *14*(14), 2795. https://doi.org/10.3390/ELECTRONICS14142795

Dhanorkar, T., Bhaskara, V., Kotapati, R., & Sethuraman, S. (2025). Programmable Banking Rails:: The Next Evolution of Open Banking APIs. *Journal of Knowledge Learning and Science Technology ISSN: 2959-6386 (Online)*, *4*(1), 121–129. https://doi.org/10.60087/JKLST.V4.N1.013

Eshan, S., Shirish, A., & Lav, U. (2025). The Power I Know: Zero-Knowledge Proofs and their Transformative Role in the Future of Cryptography. *IEEE Access*. https://doi.org/10.1109/ACCESS.2025.3599555

Femenias, G., Francisca, H. H., Riera-Palou, F., Ferrer-Gomila, J. L., & Jaume-Barceló, A. (2025). A Multi-Leader Multi-Follower Stackelberg Game for Dynamic Spectrum Sharing in a Blockchain Enabled Cell-Free Massive MIMO Scenario. *IEEE Open Journal of the Communications Society*, *6*, 5359–5383. https://doi.org/10.1109/OJCOMS.2025.3578102

Fugkeaw, S., Sungchai, S., Nakprame, S., & Sreekongpan, P. (2025). Enabling Secure and Scalable GDPR-Compliant Blockchain-based e-KYC with Efficient Redaction. *IEEE Access*. https://doi.org/10.1109/ACCESS.2025.3594656

Hamza, M., & Smolander, K. (2025). *From ideology to implementation : real-world blockchain use cases and the European Union's strategies for adoption*. https://lutpub.lut.fi/handle/10024/169346

Hanif, R., Ahmad, H. S., & Ali, A. (2025). Developing an Integrated AML Risk Management Framework for Commercial Banks Based on Customer Risk Profiling and Enhanced Due

Diligence. *Advance Journal of Econometrics and Finance*, *3*(3), 206–215. https://ajeaf.com/index.php/Journal/article/view/114

Huang, K., Sai Narajala, V., Yeoh, J., Ross, J., Lambe, M., Raskar, R., Harkati, Y., Huang, J., Habler, I., & Hughes, C. (2025). A Novel Zero-Trust Identity Framework for Agentic AI: Decentralized Authentication and Fine-Grained Access Control. *ArXiv*. https://arxiv.org/pdf/2505.19301

Inaltong, N. U. (2025). *Anti-Money Laundering Practices in the Scope of Risk Mitigation and Comparison with Anti-Money Laundering Regulations*. https://doi.org/10.2139/SSRN.5215578

Ismail, S., Abdou, R. M., & Ibrahim, M. S. (2025). Who is Better in Practicing Customer Due Diligence as an Anti-Money Laundering Tool for Financial Institutions: Can Internal Auditors Be Forensic Accountants? Evidence from MENA Region. *Studies in Big Data*, *171*, 11–23. https://doi.org/10.1007/978-3-031-83911-5_2

Kaiiali, M., Sette, I. S., Wazan, A. S., Chadwick, D. W., & Alfandi, O. (2025). On the interoperability of verifiable credentials: simple universal verifier (SUV). *Annales Des Telecommunications/Annals of Telecommunications*, *80*(7), 639–657. https://doi.org/10.1007/S12243-025-01066-4/TABLES/3

Mikołajczyk, P., Hassanizadeh, P., & Ebrahimi, S. (2025). Towards Trustless Provenance: A Privacy-Preserving Framework for On-chain Media Verification. *Cryptology EPrint Archive*, *1*. https://eprint.iacr.org/2025/1024

Nguyen, T. L., Nguyen, L., Hoang, T., Bandara, D., Wang, Q., Lu, Q., Xu, X., Zhu, L., & Chen, S. (2025). Blockchain-Empowered Trustworthy Data Sharing: Fundamentals, Applications, and Challenges. *ACM Computing Surveys*, *57*(8). https://doi.org/10.1145/3718082/ASSET/1E756987-9780-4E4C-AD05-166C6B984213/ASSETS/IMAGES/LARGE/CSUR-2023-1031-F09.JPG

Omar, M., & Khan, F. (2025). *Blockchain-Enabled Secure, Fast and Accessible Cross-Border Money Transfer Platform*. https://www.doria.fi/handle/10024/192738

Vanella, A. (2025). *Evolution of Digital Identity in Europe: Experimenting with the eIDAS 2.0 Framework and the EU Digital Identity Wallet*.

Wang, Q., Qian, C., Mia, S., Zhang, H., Zhao, H., Lu, Y., & Zhu, H. (2025). Blockchain-Enabled Credible Multi-Operator Spectrum Sharing in UAV Communication Systems. *IEEE Transactions on Vehicular Technology*, *74*(6), 8989–9001. https://doi.org/10.1109/TVT.2025.3531541

Xiong, X., Huth, M., & Knottenbelt, W. (2025). REGKYC: Supporting Privacy and Compliance Enforcement for KYC in Blockchains. *Cryptology EPrint Archive*. https://eprint.iacr.org/2025/579

APPENDIX

Appendix A: VC Templates (JSON, SD-JWT payload sketches)

```
{
  "urn:tc:kyc:person_identity:v1": {
    "sd_jwt_payload": {
      "iss": "did:issuer:bank-eg-001",
      "sub": "did:person:abc123",
      "nbf": "2025-08-01T00:00:00Z",
      "exp": "2027-08-01T00:00:00Z",
      "jti": "vc-person-123",
      "trusted_schema": "urn:tc:kyc:person_identity:v1",
      "claims": {
        "legal_name": "AHMED MAHMOUD HAMZA",
        "dob": "1985-03-01",
        "nationality": "EG",
        "id_type": "national_id",
        "id_hash": "sha256-..."
      },
      "revocation_id": "rvk:eg:person:123"
    }
  },
  "urn:tc:kyc:address:v1": {
    "sd_jwt_payload": {
      "iss": "did:issuer:bank-eg-001",
      "sub": "did:person:abc123",
      "nbf": "2025-08-01T00:00:00Z",
      "exp": "2026-02-01T00:00:00Z",
      "jti": "vc-address-456",
      "trusted_schema": "urn:tc:kyc:address:v1",
      "claims": {
        "address_line": "Flat 5, Building 10, Nasr City",
        "city": "Cairo",
        "country": "EG",
        "postal_code": "11371",
        "evidence": {"kind": "utility_bill_hash", "hash": "sha256-...",
"issued_at": "2025-07-15T00:00:00Z"}
      },
      "revocation_id": "rvk:eg:addr:456"
    }
  },
  "urn:tc:kyc:screening:v1": {
    "sd_jwt_payload": {
      "iss": "did:issuer:kycprov-eg-009",
      "sub": "did:person:abc123",
      "nbf": "2025-08-01T00:00:00Z",
      "exp": "2026-08-01T00:00:00Z",
      "jti": "vc-screen-789",
      "trusted_schema": "urn:tc:kyc:screening:v1",
```

```
      "claims": {"pep": "no_match", "sanctions": "no_match",
"adverse_media": "none", "run_at": "2025-08-01T10:15:00Z", "provider":
"Refinitiv"},
      "revocation_id": "rvk:eg:screen:789"
    }
  },
  "urn:tc:kyc:business_identity:v1": {
    "sd_jwt_payload": {
      "iss": "did:issuer:cr-eg",
      "sub": "did:org:eg:123456789",
      "nbf": "2025-08-01T00:00:00Z",
      "exp": "2028-08-01T00:00:00Z",
      "jti": "vc-biz-001",
      "trusted_schema": "urn:tc:kyc:business_identity:v1",
      "claims": {"legal_name": "TrustChainKYC Technologies LLC", "reg_no":
"EG-CR-123456", "jurisdiction": "EG", "legal_form": "LLC",
"incorporation_date": "2023-05-20", "registered_address": ",Ä¶", "status":
"active"},
      "revocation_id": "rvk:eg:biz:001"
    }
  },
  "urn:tc:kyc:tax_vat:v1": {
    "sd_jwt_payload": {
      "iss": "did:issuer:tax-eg",
      "sub": "did:org:eg:123456789",
      "nbf": "2025-08-01T00:00:00Z",
      "exp": "2027-08-01T00:00:00Z",
      "jti": "vc-tax-002",
      "trusted_schema": "urn:tc:kyc:tax_vat:v1",
      "claims": {"tax_id": "EG-TAX-998877", "vat_status": "registered",
"vat_number": "EG-VAT-112233"},
      "revocation_id": "rvk:eg:tax:002"
    }
  },
  "urn:tc:kyc:ubo_control:v1": {
    "sd_jwt_payload": {
      "iss": "did:issuer:kycprov-eg-009",
      "sub": "did:org:eg:123456789",
      "nbf": "2025-08-01T00:00:00Z",
      "exp": "2026-02-01T00:00:00Z",
      "jti": "vc-ubo-003",
      "trusted_schema": "urn:tc:kyc:ubo_control:v1",
      "claims": {"controllers": [{"name": "A. H. H.", "dob_month": 3,
"dob_year": 1985, "pct": 40}, {"name": "J. D.", "dob_month": 7,
"dob_year": 1982, "pct": 20}], "last_verified": "2025-08-01T11:00:00Z",
"evidence_ref": "opaque-ubo-register-ref"},
      "revocation_id": "rvk:eg:ubo:003"
    }
  },
  "urn:tc:kyc:authority_role:v1": {
    "sd_jwt_payload": {
      "iss": "did:issuer:org-admin-eg-123",
      "sub": "did:person:abc123",
```

```
      "nbf": "2025-08-01T00:00:00Z",
      "exp": "2026-08-01T00:00:00Z",
      "jti": "vc-role-004",
      "trusted_schema": "urn:tc:kyc:authority_role:v1",
      "claims": {"act_for": "did:org:eg:123456789", "role": "Authorised
Signatory", "scope": ["open_account", "sign_contracts"], "valid_from":
"2025-08-01", "valid_to": "2026-08-01"},
      "revocation_id": "rvk:eg:role:004"
    }
  }
}
```

Appendix B: Issuer API (OpenAPI Stub)

```
openapi: 3.1.0
info:
  title: TrustChainKYC ,Äî Issuer API (MVP Stub)
  version: 0.1.0
servers:
  - url: https://api.trustchainkyc.example.com
components:
  securitySchemes:
    oauth2:
      type: oauth2
      flows:
        clientCredentials:
          tokenUrl: https://auth.trustchainkyc.example.com/oauth2/token
          scopes:
            issuer:issue: Issue credentials
            issuer:revoke: Revoke credentials
            issuer:read: Read status and registry
  schemas:
    IssueRequest:
      type: object
      required: [type, subject_did, claims]
      properties:
        type: { type: string }
        subject_did: { type: string }
        claims: { type: object, additionalProperties: true }
        evidence:
          type: array
          items:
            type: object
            properties:
              kind: { type: string }
              ref: { type: string }
              hash: { type: string }
              issued_at: { type: string, format: date-time }
        options:
          type: object
          properties:
            exp: { type: string, format: date-time }
            nbf: { type: string, format: date-time }
            aud: { type: string }
            tag: { type: string }
        idempotency_key: { type: string }
    IssueResponse:
      type: object
      properties:
        vc_jwt: { type: string }
        disclosures: { type: array, items: { type: string } }
        jti: { type: string }
        revocation_id: { type: string }
        exp: { type: string, format: date-time }
```

```
        nbf: { type: string, format: date-time }
        issued_at: { type: string, format: date-time }
    RevokeRequest:
      type: object
      required: [revocation_id, reason]
      properties:
        revocation_id: { type: string }
        reason: { type: string, enum: [user_request, suspected_fraud,
superseded, expired, other] }
        performed_by: { type: string }
        at: { type: string, format: date-time }
    StatusResponse:
      type: object
      properties:
        status: { type: string, enum: [active, revoked] }
        revoked_at: { type: string, format: date-time, nullable: true }
        reason: { type: string, nullable: true }
        issuer_did: { type: string }
        credential_type: { type: string }
    RotateKeysRequest:
      type: object
      properties:
        next_kid: { type: string }
        not_before: { type: string, format: date-time }
        not_after: { type: string, format: date-time }
paths:
  /v1/issue:
    post:
      summary: Issue a verifiable credential (SD-JWT)
      security: [ { oauth2: [issuer:issue] } ]
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/IssueRequest'
      responses:
        '201':
          description: Issued
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/IssueResponse'
  /v1/revoke:
    post:
      summary: Revoke a credential via its revocation id
      security: [ { oauth2: [issuer:revoke] } ]
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/RevokeRequest'
```

```
    responses:
      '200': { description: Revoked }
/v1/status/{revocation_id}:
  get:
    summary: Get revocation/status for a credential
    security: [ { oauth2: [issuer:read] } ]
    parameters:
      - in: path
        name: revocation_id
        required: true
        schema: { type: string }
    responses:
      '200':
        description: Status
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/StatusResponse'
/v1/rotate-keys:
  post:
    summary: Rotate issuer signing keys (maker,Äëchecker recommended)
    security: [ { oauth2: [issuer:revoke] } ]
    requestBody:
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/RotateKeysRequest'
    responses:
      '202': { description: Rotation scheduled }
/v1/trust-registry:
  get:
    summary: Fetch trust registry entries relevant to this issuer
    security: [ { oauth2: [issuer:read] } ]
    responses:
      '200': { description: Registry list }
```

Appendix C: Verifier API (OpenAPI Stub)

```
openapi: 3.1.0
info:
  title: TrustChainKYC ,Äî Verifier API (MVP Stub)
  version: 0.1.0
servers:
  - url: https://verifier.trustchainkyc.example.com
components:
  securitySchemes:
    oauth2:
      type: oauth2
      flows:
        clientCredentials:
          tokenUrl: https://auth.trustchainkyc.example.com/oauth2/token
          scopes:
            verifier:verify: Verify presentations
            verifier:read: Read requests/status
  schemas:
    ProofRequest:
      type: object
      required: [requested_credentials, expires_at]
      properties:
        requested_credentials:
          type: array
          items:
            type: object
            required: [type]
            properties:
              type: { type: string }
              purpose: { type: string }
              selective_disclosure: { type: boolean, default: true }
              constraints: { type: object, additionalProperties: true }
        nonce: { type: string }
        expires_at: { type: string, format: date-time }
        callback_url: { type: string }
    ProofRequestResponse:
      type: object
      properties:
        request_id: { type: string }
        oob_url: { type: string }
        qr_png: { type: string }
        expires_at: { type: string, format: date-time }
    ProofSubmission:
      type: object
      required: [request_id, presentation_type, vp_token]
      properties:
        request_id: { type: string }
        holder_did: { type: string }
        presentation_type: { type: string, enum: [sd-jwt-vc, json-ld-vc] }
        vp_token: { type: string }
        disclosures: { type: array, items: { type: string } }
```

```
        consent_receipt:
          type: object
          properties:
            purpose: { type: string }
            requested_by: { type: string }
            timestamp: { type: string, format: date-time }
    VerificationResult:
      type: object
      properties:
        result: { type: string, enum: [valid, invalid] }
        checks:
          type: object
          properties:
            signature: { type: string, enum: [pass, fail] }
            expiry: { type: string, enum: [pass, fail] }
            revocation: { type: string, enum: [pass, fail] }
            schema: { type: string, enum: [pass, fail] }
        extracted_claims: { type: object, additionalProperties: true }
        policy_outcome:
          type: object
          properties:
            decision: { type: string, enum: [approve, refer, deny] }
            reasons: { type: array, items: { type: string } }
paths:
  /v1/proof-requests:
    post:
      summary: Create a proof request for a wallet
      security: [ { oauth2: [verifier:verify] } ]
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/ProofRequest'
      responses:
        '201':
          description: Created
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/ProofRequestResponse'
  /v1/proof-requests/{id}/status:
    get:
      summary: Get the status of a proof request
      security: [ { oauth2: [verifier:read] } ]
      parameters:
        - in: path
          name: id
          required: true
          schema: { type: string }
      responses:
        '200': { description: Status }
  /v1/proof-submissions:
```

```
    post:
      summary: Submit a presentation from the wallet (callback target)
      security: [ { oauth2: [verifier:verify] } ]
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/ProofSubmission'
      responses:
        '200':
          description: Verification result
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/VerificationResult'
  /v1/policies/evaluate:
    post:
      summary: Evaluate a policy against extracted claims
      security: [ { oauth2: [verifier:verify] } ]
      requestBody:
        required: true
        content:
          application/json:
            schema:
              type: object
              properties:
                risk_tier: { type: string, enum: [low, medium, high] }
                context: { type: string, description: retail|business }
                claims: { type: object, additionalProperties: true }
      responses:
        '200':
          description: Decision
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/VerificationResult'
  /v1/status/{revocation_id}:
    get:
      summary: Pass,Äëthrough revocation check (convenience)
      security: [ { oauth2: [verifier:read] } ]
      parameters:
        - in: path
          name: revocation_id
          required: true
          schema: { type: string }
      responses:
        '200': { description: Status }
```

Appendix D: Pilot Verification Policy (Risk-based)

Context	Risk tier	Required credentials	Optional (risk-based)	Refresh
Retail	Low	`person_identity`, `screening`	`address`	Screening 12 m; ID 24–36 m
Retail	Medium	`person_identity`, `screening`, `address`	Liveness/match (OOB)	Screening 6–12 m; Address 12 m
Retail	High	All above + EDD flags	Source-of-funds pointer	Screening 3–6 m
Business	Low	`business_identity`, `tax_vat`, `authority_role` (≥1)	`ubo_control` summary	ID 24–36 m; Tax 24 m
Business	Medium	+ `entity_screening` (if available)	Account ownership proof	Entity screen 6–12 m
Business	High	+ **person KYC** for UBOs ≥25% & all signatories	Additional EDD evidence	UBO/Authority 6–12 m

Appendix E: Sequence Sketches

Issuance: Auth → Validate → Build claims → Sign SD-JWT → Deliver to wallet → Publish revocation → Audit.

Verification: Create request (nonce, expiry) → Wallet selects claims → Present SD-JWT + disclosures → Verify signature/expiry/revocation/trust → Policy → Audit.

Revocation: Revoke(reason) → Update status → Invalidate caches → Next checks reflect status.

Appendix F: DPIA Template (Starter)

F1 Purpose & scope; **F2 Controllers/Processors**; **F3 Processing description** (issuance, verification, revocation, audit); **F4 Lawful basis** (contract, legitimate interests, consent); **F5 Data minimisation** (selective disclosure, no raw docs on TrustChainKYC); **F6 Risk assessment**; **F7 Measures** (technical/organisational); **F8 Residual risk & approvals** (DPO sign-off).

Appendix G: UAT Test Matrix

ID	Scenario	Input	Expected result

U1	Valid retail proof	Person ID + Screening	Decision = Approve
U2	Expired VC	Person ID exp<today	Decision = Deny (expiry)
U3	Revoked VC	Revocation=revoked	Decision = Deny (revoked)
U4	Wrong schema	Unexpected claims	Decision = Deny (schema)
U5	Replay	Same VP + nonce used	Decision = Deny (replay)
U6	Missing authority	No matching role VC	Decision = Deny (authority)
U7	UBO threshold	UBO<required coverage	Decision = Refer
U8	Watchlist hit	Screening = hit	Decision = Refer + workflow
U9	Consent missing	No receipt	Deny + error

Appendix H: ISO/IEC 27001:2022 Mapping (Extract)

- **A.5–A.6** Policies & organisation; people controls; NDA & awareness.
- **A.8** Asset & data classification (PII, logs, keys).
- **A.9** Access control (RBAC/ABAC, MFA, least privilege, secrets vault).
- **A.10** Cryptography (HSM, key rotation, KID versioning, crypto agility).
- **A.12–A.14** Ops & SDLC (SAST/DAST, SCA, SBOM, IaC scans).
- **A.15** Supplier DPAs & due diligence.
- **A.16** Incident management (SEV ladder, runbooks).
- **A.17** Business continuity (RTO≤4 h, RPO≤1 h).
- **A.18** Compliance (privacy, records).

Appendix I: Change & Release Policy (Pilot)

Semantic versioning; additive schema changes only; weekly change windows; emergency hotfix path; migration plans for breaking changes (post-pilot).

Appendix J: Partner Integration Checklist

Contract & DPA; DPIA; keys/DIDs published; sandbox credentials; proof templates by risk tier; monitoring hooks; alert channels; runbook contacts; SEV ladder agreed.

Appendix K: Example Solidity-style Pseudocode

```
contract ConsentManager {
      event ConsentGranted(bytes32 holder, bytes32 rp, bytes32 scope,
      uint64 expiry);
      event ConsentRevoked(bytes32 holder, bytes32 rp, bytes32 scope);
      mapping(bytes32 => mapping(bytes32 => mapping(bytes32 => uint64)))
      public consentExpiry;
      function grant(bytes32 holder, bytes32 rp, bytes32 scope, uint64
      expiry) external {
       require(msg.sender == addressFromDID(holder));
       require(expiry > block.timestamp);
       consentExpiry[holder][rp][scope] = expiry;
       emit ConsentGranted(holder, rp, scope, expiry);
 }
      function revoke(bytes32 holder, bytes32 rp, bytes32 scope) external
{
      require(msg.sender == addressFromDID(holder));
      consentExpiry[holder][rp][scope] = 0;
      emit ConsentRevoked(holder, rp, scope);
 }
}
```