applied sciences

MDPI

Article

A Hybrid Multi-Agent System for Early Scam Detection in Crypto-Assets

Mario Trerotola [1,*], Mimmo Parente [2] and Davide Calvaresi [3]

[1] Department of Control and Computer Engineering (DAUIN), Politecnico di Torino, 10129 Torino, Italy
[2] Dipartimento di Scienze Aziendali—Management & Innovation Systems, Università degli Studi di Salerno, 84084 Fisciano, Italy; parente@unisa.it
[3] HEI, HES-SO Valais-Wallis, Rue de l'industrie 23, CH-1950 Sion, Switzerland; davide.calvaresi@hevs.ch
* Correspondence: mario.trerotola@polito.it; Tel.: +39-3489842058

Abstract

The rapid expansion of crypto-asset markets and the introduction of the Markets in Crypto-Assets Regulation (MiCAR) pose novel supervisory challenges. Existing blockchain intelligence platforms focus predominantly on on-chain surveillance, leaving gaps in off-chain documentary due diligence automation. This paper presents a Multi-Agent System (MAS) integrating Large Language Model (LLM) capabilities with rule-based compliance frameworks. The architecture comprises seven specialized agents: a `Coordinator Agent` for orchestration; data acquisition agents (`Searcher`, `Crawler`); three parallel analytical agents—`Heuristic Agent` (LLM-powered qualitative risk assessment), `Compliance Agent` (hybrid-AI MiCAR asset classification and regulatory requirement verification), and `On-Chain Agent` (machine learning-based fraud detection); and a `Reconciliator Agent` synthesizing findings into unified alerts. Component-level empirical validation on 150 projects indicates 95% output reproducibility (identical alert tier and score deviation ≤ 0.05 across five reruns) and 210 s mean latency, providing proof-of-concept evidence for the integrated pipeline. A pilot user evaluation (six researchers/master students and two experts from regulatory authorities) provides preliminary usability evidence and surfaces domain-specific feedback from regulatory-authority experts. The architecture advances proactive regulatory technology by enabling scalable analysis combining off-chain documentary evidence with on-chain forensics.

Keywords: crypto-asset markets; regulatory technology (RegTech); markets in Crypto-Assets Regulation (MiCAR); multi-agent systems; large language models; compliance automation; off-chain due diligence; explainable AI; fraud detection

Academic Editor: Firstname Lastname

Received:
Revised:
Accepted:
Published:

Citation: . . *Appl. Sci.* **2026**, *1*, 0.

1. Introduction

Distributed Ledger Technologies (DLTs) and crypto-assets have evolved from niche technical experiments into a global financial infrastructure supporting payment systems, decentralized finance (DeFi), and a diverse array of tokenized assets. This rapid expansion has attracted both legitimate innovation and speculative activity, substantially complicating the task of monitoring market integrity and protecting investors. Supervisory authorities now face a dual challenge: understanding novel technological architectures while enforcing regulatory principles in markets characterized by high volatility and pseudonymous participation.

The properties that make DLT-based systems attractive, namely borderless access, programmable assets, and low entry barriers, also create fertile conditions for fraud and

market abuse. Crypto-asset markets have witnessed a range of illicit activities, including market manipulation, insider trading, and the exploitation of apparently legitimate projects as vehicles for unregistered securities offerings or outright scams. Such behaviors frequently span both on-chain transactions and off-chain documentation, making comprehensive supervision particularly demanding for resource-constrained human analysts.

A substantial portion of fraudulent activity manifests itself through token-based scams. *Rug Pull Schemes* (a type of exit scam in which project developers abandon a project and abscond with investor funds after attracting substantial capital) exploit asymmetries in token control or liquidity provision, enabling insiders to extract value before abandoning the project. *Honeypots* (smart contracts engineered to permit token purchases while programmatically preventing subsequent sales or withdrawals) deploy smart contract logic that entices users to purchase tokens they cannot subsequently sell or withdraw. *Impersonation* schemes replicate the branding, naming, or visual identity of established tokens or exchanges to deceive participants. Although these patterns ultimately materialize at the smart-contract and transaction levels, they are typically preceded by misleading or incomplete off-chain disclosures, such as websites, white papers, and marketing materials. Qualitative evidence also indicates that socio-psychological triggers such as fear of missing out (FOMO), perceived legitimacy cues, and low technical literacy can materially increase susceptibility to crypto scams, motivating explicit detection of high-pressure marketing tactics in off-chain content [1].

The introduction of the European Union's Markets in Crypto-Assets Regulation (MiCAR) has formalized disclosure obligations and asset classification rules for a broad range of crypto-assets in the EU market. However, contemporary blockchain intelligence platforms predominantly focus on on-chain, retrospective forensics [2], including transaction graph analysis and wallet provenance tracking, rather than prospective examination of project documentation, and publicly documented support for systematic, MiCAR-aligned documentary assessment appears limited. Consequently, regulatory authorities must process substantial volumes of unstructured off-chain information (whitepapers, legal documentation, marketing content) to assess MiCAR compliance and identify early indicators of fraud. Manual processing of such information is labor-intensive, error-prone, and difficult to scale at the pace of token issuance. Scam activity is not unique to crypto markets; it spans traditional finance, payments, and broader cybercrime. We focus on crypto-assets because (i) MiCAR introduces specific disclosure obligations and taxonomy rules, (ii) token issuance velocity creates a high-volume screening problem, and (iii) crypto projects publish rich off-chain documentation that can be assessed before significant on-chain activity emerges. Recent empirical evidence confirms that cryptocurrency ownership itself acts as a fraud risk amplifier: crypto investors are approximately twice as likely to be targeted by scams and to incur financial losses compared to non-crypto investors [3], underscoring the urgency of scalable early-warning tools in this domain. While this work targets crypto-asset compliance and scam detection, the underlying MAS design is extensible to broader financial fraud and AML workflows as future work. The scale of crypto-specific fraud underscores this focus: the FBI's 2024 Internet Crime Complaint Center report documented USD 9.3 billion in cryptocurrency-related fraud losses in the United States alone—a 66% increase over 2023—representing approximately 56% of total reported cybercrime losses (USD 16.6 billion). Investment fraud involving cryptocurrency accounted for over USD 6.5 billion of these losses, with nearly 150,000 complaints filed [4]. These figures, which likely underestimate actual losses due to chronic underreporting, confirm that crypto-asset fraud constitutes a dominant and rapidly growing segment of financial cybercrime, justifying the development of domain-specific screening tools. This work positions itself at the intersection of Regulatory Technology (RegTech), blockchain analytics, and multi-agent

computational paradigms [5,6]. Multi-agent architectures have attracted considerable attention in financial and regulatory domains as mechanisms to decompose complex operational workflows into specialized and coordinated components [7,8]. Similarly, research on Large Language Models (LLMs) has demonstrated their applicability to information extraction, documentary analysis, and compliance verification [9,10], particularly when combined with structured prompting strategies and constrained output generation [11,12]. The present work synthesizes these research directions to address the identified regulatory challenges. We introduce a Multi-Agent System (MAS) [13] in which specialized agents operate autonomously to analyze, classify, and evaluate crypto-assets by systematically examining publicly disclosed information. The central hypothesis is that a hybrid AI paradigm that combines deterministic rule-based frameworks with the semantic reasoning capabilities of LLMs can provide a scalable yet interpretable solution to two critical supervisory tasks: (i) automated asset taxonomy classification and disclosure verification against regulatory requirements, and (ii) heuristic analysis capable of identifying documentary anomalies and risk signals characteristic of fraudulent schemes. A Multi-Agent System (MAS) architecture supports supervisory due diligence by orchestrating heterogeneous tasks, from deterministic rule and taxonomy validation to data-driven analysis of unstructured disclosures and on-chain activity (e.g., via LLMs and Balanced Random Forest classifiers). Decomposing these functions into specialized agents enables modular design, parallel execution, and improved traceability of decision processes. The proposed architecture leverages autonomous agent orchestration to achieve computational efficiency and analytical rigor while preserving transparency through structured explainability mechanisms. A distinguishing feature of this approach is its dual capacity for regulatory taxonomy classification and systematic disclosure verification, tasks that have historically required substantial manual effort. As shown in Table 1, the framework prioritizes **Regulatory Compliance** alignment and **Explainability**, dimensions where existing commercial blockchain intelligence platforms exhibit notable deficiencies.

Table 1. Comparison of key functionalities between our platform and other blockchain intelligence solutions. ✓ = present; × = absent.

Functionality/Platform	Our Platform	Chainalysis	Elliptic	Token Sniffer
Monitor New Tokens	✓	×	×	✓
Regulatory Compliance (MiCAR)	✓	×	×	×
Focus on Explainability (LLM)	✓	×	×	✓

Table 2. Comparison with peer-reviewed academic systems in crypto-asset fraud detection and compliance. ✓ = present; (✓) = partial; × = absent.

Dimension	*Our System*	*Toma & Cerch.* [14]	*Karimov & W.* [15]	*Mazorra et al.* [16]	*Pocher et al.* [17]	*Liang et al.* [18]	*Luo et al.* [19]
Data sources	Both	Off-chain	Off-chain	On-chain	On-chain	On-chain	Survey
Reg. framework	MiCAR	×	×	×	AML/CFT *	×	×
LLM integration	✓	×	×	×	×	×	×
Multi-agent arch.	7 agents	×	×	×	×	×	×
Explainability	SHAP + CIU + rules	×	SHAP+PDP	×	×	CRBG †	N/A
Output type	Struct. alert	Stat. profile	Classification	Rug-pull flag	Anomaly score	Ponzi/benign	Taxonomy
Scale	227k+ tokens	196 ICOs	∼300 ICOs	20k+ tokens	BTC tx graph	ETH contracts	Survey

* Generic AML/CFT framing, not jurisdiction-specific checks. † Graph-level interpretability via Contract Runtime Behavior Graph (CRBG), not feature-attribution XAI.

Table 1 addresses commercial platforms. Table 2 complements this view by contrasting our architecture with six peer-reviewed journal publications spanning document-based scam prediction [14,15], on-chain fraud detection [16,18], regulatory-aligned crypto forensics [17], and systematic surveys [19]. Existing systems address either off-chain feature extraction (e.g., tabular ICO attributes with XAI [15]) or on-chain behavioural analysis (e.g., runtime graph classification with F1 of 97.5% [18]), but none integrates both layers. Furthermore, no surveyed work implements jurisdiction-specific compliance checking: Pocher et al. [17] frame Bitcoin forensics within AML/CFT objectives but do not perform automated disclosure verification, and Luo et al. [19] confirm that NLP-based off-chain document analysis remains operationally under-explored. Our contribution differs by combining MiCAR-specific taxonomy and disclosure checks, LLM-constrained documentary analysis, on-chain fraud detection, and multi-agent orchestration in a unified, auditable pipeline.

The principal objective of this work is to present the architectural design of this MAS-LLM integration and empirically validate its operational efficacy. We provide detailed technical specifications for the `Compliance Agent` and `Heuristic Agent`, describing the prompt engineering strategies employed to constrain LLM outputs and the deterministic logic underlying rule-based classification. We subsequently evaluate the platform's computational performance and practical utility through controlled user evaluation protocols. Given the pilot nature of the user study (eight mixed-profile participants including two experts from regulatory authorities), we restrict conclusions about end-user value to exploratory, non-generalizable evidence.

The intended users are regulatory analysts and compliance teams responsible for screening new crypto-asset offerings, with secondary use by academic researchers and investigative journalists for exploratory due diligence. Operational costs in our prototype are dominated by LLM inference; under our current deployment (single mid-range GPU workstation with API-based LLM calls), the end-to-end analysis averages 210 s per project and an estimated USD $0.80–1.20 per project, excluding data-collection bandwidth. Effectiveness is evidenced by 95% output reproducibility, the on-chain classifier's 98% balanced accuracy on 227,000 labeled tokens, and qualitative alignment between documentary risk flags and human-reviewed reference labels in the 150-project sample. We note that effectiveness in this context encompasses three complementary dimensions: (i) discriminative performance of the on-chain classifier, (ii) output stability of the LLM-driven pipeline under repeated execution, and (iii) extraction fidelity of compliance flags against human-verified references. End-to-end detection effectiveness (true-positive and false-positive rates for the integrated MAS) remains unquantified in the absence of an authoritative ground-truth benchmark and is identified as a primary target for future validation.

This work addresses three research questions:

- **RQ1** (LLM reliability): To what extent can constrained LLM prompting combined with deterministic rule-based logic produce reproducible, auditable outputs for regulatory document analysis?
- **RQ2** (cross-source generalization): How robust is a fraud-detection classifier trained on one community-curated data source when evaluated against an independently labeled expert corpus?
- **RQ3** (multi-modal complementarity): Can a multi-agent architecture integrating off-chain documentary analysis with on-chain behavioural signals provide complementary risk evidence that neither modality achieves alone?

The contributions of this work are threefold:

- *Architectural Innovation*—Hybrid Neuro-Symbolic Architecture for Regulatory NLP: We present a Multi-Agent System integrating an LLM-powered `Heuristic Agent` with

a hybrid-AI `Compliance Agent`, intended to support automated off-chain documentary due diligence and MiCAR-aligned regulatory assessment for crypto-assets. This hybrid paradigm demonstrates that LLM semantic extraction combined with deterministic compliance logic can yield auditable, reproducible regulatory outputs (RQ1).
- *Technical Specification*: We provide detailed architectural documentation covering agent orchestration protocols, prompt engineering strategies, and structured output schemas that support both interpretability and machine-processable analytical outputs.
- *Empirical Validation*: We report a multi-tier evaluation: (i) large-scale benchmarking of the `On-Chain Agent` on 227,000 labeled tokens with 98.23% balanced accuracy in-source and 93.45% cross-source generalization (94.67% combined); (ii) off-chain pipeline assessment on 150 projects, measuring both reproducibility (95% identical alert tier across five independent reruns) and latency (210 s mean end-to-end processing time); and (iii) an exploratory usability study assessing perceived clarity and actionability.

These empirical results contribute evidence on three questions—LLM determinism under constrained prompting (RQ1), distribution shift across heterogeneous fraud corpora (RQ2), and multi-modal risk signal complementarity (RQ3)—that extend beyond architectural design into the methodological foundations of RegTech (a visual summary of key quantitative results is provided in Section 5).

2. State of the Art

This section positions our contribution within the wider research on blockchain analytics, the detection of scams in crypto-asset markets, and Regulatory Technology. We first review research on deanonymizing addresses and linking blockchain addresses to real-world entities, then examine approaches for detecting fraudulent wallets and transactional patterns. Subsequently, we summarize existing token-based scam detection methods, with particular attention to Rug Pulls, Honeypots, and impersonation schemes, highlighting the role of explainable AI (XAI) in rendering these systems interpretable for supervisory authorities. Finally, we identify the principal gaps that motivate the architecture proposed in this work.

Complementing the commercial-platform comparison in Table 1, we contrast our approach with academic RegTech and MAS-based compliance systems and legal/NLP document analysis frameworks. Academic RegTech work emphasizes regulatory process automation and supervisory technology design [5,6], while MAS research provides coordination patterns for distributed compliance tasks [7,13]. In parallel, legal NLP benchmarks and domain-adapted models (e.g., LEGAL-BERT, LegalBench) demonstrate scalable extraction and reasoning over regulatory text [20,21]. Our contribution differs by combining MiCAR-specific taxonomy and disclosure checks with LLM-constrained extraction and multi-agent orchestration in a unified, auditable pipeline tailored to crypto-asset screening.

2.1. Linking Blockchain Addresses to Real-World Identities

The challenge of linking pseudonymous blockchain addresses to real-world entities has received substantial attention in the transaction-tracing literature. Early empirical work on Bitcoin demonstrated that relatively simple heuristics, such as multi-input clustering and change-address detection, enable reconstruction of user-level activity graphs from raw on-chain data [2]. Subsequent surveys and systematic reviews categorize tracing techniques into heuristic clustering, graph-based analytics, and machine-learning-driven anomaly detection, noting that most approaches require auxiliary off-chain information (exchange labels, KYC anchors, sanctions lists) to complete the attribution loop between addresses

and identifiable entities. Recent systematic reviews [22] provide a detailed map of the technical landscape and the limitations of current deanonymization methods.

In this context, recent research has produced highly optimized graph-processing frameworks to perform large-scale address attribution on both account-based and UTXO blockchains. Systems such as TRacer, TPGraph, and MFGSCOPE employ temporal partitioning, cache-efficient graph layouts, and localized subgraph expansion to support near real-time tracing of suspicious flows across millions of nodes and edges, frequently combined with graph neural networks or link-prediction models that flag anomalous transaction patterns. Complementary cross-ledger frameworks (e.g., CLTracer-style approaches) correlate deposits and withdrawals across centralized exchanges, bridges, and decentralized exchanges, tracking assets as they traverse chains and liquidity venues. These architectures typically integrate on-chain graph signals with off-chain metadata, such as KYC records and blacklists, to improve attribution accuracy while acknowledging persistent uncertainty and the risk of false positives.

Despite these advances, a structural tension persists between scalability, attribution accuracy, and privacy compliance: high-throughput graph analytics and aggressive clustering heuristics often sacrifice interpretability and legal robustness, particularly when operating at the boundary of what can reliably be inferred from pseudonymous data. Our work positions itself upstream of these deanonymization pipelines. Rather than resolving address identities directly from on-chain traces, we focus on the documentary layer that accompanies on-chain activity: whitepapers, websites, and disclosure materials published by token issuers. By integrating the analysis of this off-chain evidence with the detection of fraudulent wallets based on on-chain data, the proposed platform provides regulators and market operators with early-warning indicators that complement downstream address-level tracing and attribution tools.

2.2. Detection of Fraudulent Wallets

Research on detecting illicit or high-risk wallets has evolved along two complementary directions. The first approach relies on *handcrafted, feature-based* representations of wallet behavior combined with classical machine learning models. In the Bitcoin context, studies [23–26] demonstrate that relatively simple graph and temporal statistics, such as degree profiles, unique counterparty counts, clustering coefficients, balance evolution, inter-transaction intervals, and wallet lifetimes, encode strong signals for identifying money laundering, ransomware cash-out addresses, mixing services, and scam-related wallets. These approaches aggregate address-level information into wallet or entity-level descriptors and train tree ensembles or support vector machines, retaining interpretability through feature importance scores and explicit decision paths.

A second research strand employs *deep representation learning* on transaction graphs. Graph neural networks and related architectures [27,28] operate directly on the local neighborhood of addresses, propagating information across multiple hops to learn latent embeddings for fraud classification. While these models achieve state-of-the-art accuracy on benchmark datasets such as Elliptic, they are computationally intensive, require substantial engineering to adapt to different ledger architectures, and remain opaque to non-technical stakeholders. Recent work has therefore emphasized *data-centric* improvements, including label-noise auditing via Confident Learning [29] and principled pruning of low-activity or anomalous addresses, along with integration of explainability tools (SHAP, CIU, rule extraction) to make wallet risk scores suitable for regulatory and investigative workflows.

2.3. Scam Token Detection Approaches

Beyond wallet-level risk scoring, asset-level scam detection has been advanced by frameworks such as *TokenScout* [30]. TokenScout represents each ERC-20 contract as a dynamic transaction graph whose nodes and edges encode addresses and value flows over time, restricted to the token's early lifecycle. A temporal Graph Neural Network (GNN) is trained on these graphs to classify tokens as scam or legitimate based on the joint evolution of structural and temporal characteristics: centrality profiles, degree distributions, inter-arrival times, and liquidity movements. The pipeline comprises three stages: (i) construction of large-scale labeled corpora by merging curated scam reports (Rug Pulls, Honeypots, impersonation schemes) with high-capitalization reference tokens; (ii) building temporal graphs from early transfers to capture behavioral signatures; and (iii) training temporal GNN architectures that achieve state-of-the-art balanced accuracy under severe class imbalance. In particular, TokenScout operates exclusively on on-chain traces without analyzing off-chain documentation, positioning it as a natural complement to the documentation-driven RegTech approach proposed in this work. Recent qualitative analyses highlight how social drivers (e.g., FOMO) and persuasive messaging amplify scam success, underscoring the value of a documentary `Heuristic Agent` that explicitly flags high-pressure marketing cues and behavioral manipulation patterns in off-chain materials [1]. TokenScout requires deployed tokens and early transfer activity to build temporal graphs, whereas our approach can operate at launch from public disclosures; TokenScout yields on-chain risk classification, while our system emphasizes MiCAR-aligned documentary compliance.

2.4. LLMs for Legal and Regulatory Document Analysis

Recent work in legal NLP and LLM-based legal reasoning provides a direct foundation for compliance-oriented document analysis. Domain-adapted language models such as LEGAL-BERT [20] and benchmark suites like LexGLUE [31] demonstrate that legal-domain pretraining and task-specific evaluation improve performance on legal text classification, entailment, and information extraction. At the LLM level, LegalBench [21] and LLM-based legal assistants such as ChatLaw [32] indicate the feasibility of structured legal reasoning and document review workflows, while GPT-4's bar-exam results [10] highlight the capability of frontier models on legal tasks. These advances motivate the design of our `Heuristic` and `Compliance` agents, which target regulatory-document ingestion, disclosure checking, and narrative explanation in a MiCAR context.

2.5. Gaps in Current Approaches

The literature reviewed above demonstrates substantial progress in detecting illicit activity within crypto-asset ecosystems, yet several fundamental gaps remain. We organize these gaps into four categories: *analytical scope*, *regulatory alignment*, *explainability*, and *temporal responsiveness*.

Analytical Scope: Off-Chain/On-Chain Dichotomy. The vast majority of blockchain intelligence tools operate exclusively on on-chain data, including transaction graphs, smart contract bytecode, and wallet activity patterns, with limited integration of off-chain documentary sources. As discussed in Section 2.1, address-level attribution methods require auxiliary information (exchange labels, KYC anchors) to link pseudonymous identifiers to real-world entities, yet this fusion typically occurs after substantial transaction volume has accumulated. Wallet-level fraud detection (Section 2.2) and token-level scam classifiers such as TokenScout (Section 2.3) similarly depend on observing behavioral footprints in early transaction graphs. Consequently, these systems struggle to identify risks *before* significant on-chain activity materializes. Projects publishing misleading disclosures, fraudulent

whitepapers, or systematic disclosure violations may evade detection until substantial capital has been attracted.

Regulatory Gap in MiCAR-Aligned Assessment Tools. With the introduction of MiCAR [33], European supervisory authorities face the operational challenge of systematically evaluating disclosure adequacy, taxonomic classification, and compliance with asset-specific obligations across potentially thousands of newly issued tokens. Despite this regulatory urgency, we observe a notable absence of tooling designed to automate MiCAR-aligned documentary assessments. Existing platforms such as Chainalysis and Elliptic prioritize transaction forensics and sanctions screening (Table 1) but do not perform structured whitepaper analysis, asset classification under regulatory taxonomies, or systematic verification of disclosures against legal frameworks. As a result, regulators must rely on labor-intensive, manual review procedures that do not scale effectively and are unable to deliver timely early warning signals across the entire token issuance lifecycle.

Explainability: Opacity and Interpretability Limitations. Even when sophisticated machine learning models achieve high classification accuracy, as demonstrated by graph neural networks in wallet fraud detection (Section 2.2) and token scam detection (Section 2.3), they frequently produce opaque risk scores that are difficult for non-expert stakeholders to interpret, audit, or justify in regulatory proceedings [34,35]. Post hoc explainability techniques (CIU, attention weights) provide partial insight into model decisions but do not inherently generate the structured, evidence-grounded narratives that supervisory analysts and compliance officers require for operational decision-making. Recent work in critical finance applications emphasizes decision traceability logs and audit trails as integral to accountability, bridging the gap between model intelligence and regulatory reviewability [36]. A substantial gap persists between state-of-the-art predictive performance and practical regulatory utility.

Temporal Responsiveness: Reactive versus Proactive Detection. Current systems predominantly operate reactively, detecting fraud or illicit activity only after a critical mass of transactions or forensic evidence has accumulated. Token-based scam detectors such as TokenScout leverage early transaction dynamics but still require the token to have been deployed and to have generated sufficient transfer activity. This temporal lag creates a vulnerability window during which retail investors may be exposed to high-risk projects. A proactive approach that identifies warning indicators and compliance deficiencies at project launch or during pre-launch marketing phases, based solely on publicly disclosed documentation, would enable earlier intervention, reduce investor harm, and support more effective supervisory risk triage. In our architecture, "proactive" refers to the off-chain `Heuristic` and `Compliance` agents operating before meaningful on-chain activity exists, while the `On-Chain Agent` is an augmenting module that activates once sufficient transaction data are available.

The literature reveals a clear need for Regulatory Technology tools that: (i) operate primarily on *off-chain documentary evidence*; (ii) align explicitly with *regulatory frameworks* such as MiCAR, performing automated taxonomy classification and disclosure verification; (iii) deliver *explainable, human-readable outputs* suitable for non-technical analysts; and (iv) function *proactively*, flagging risks early in the asset lifecycle. The Multi-Agent System architecture proposed herein addresses these gaps by combining LLM-driven semantic analysis of project documentation with deterministic rule-based compliance assessment, producing structured, auditable alerts tailored to supervisory workflows. This approach complements, rather than replaces, the on-chain analytical capabilities reviewed in Sections 2.1–2.3, enabling a more comprehensive supervisory posture.

3. Platform Design

This section presents the overall design of the platform, covering system-level objectives, functional and regulatory requirements, and the high-level architecture integrating data acquisition, analytical agents, and user-facing components.

3.1. System Overview and Objectives

The platform is designed as a Regulatory Technology tool to support supervisory authorities and market infrastructure operators in performing scalable, evidence-based assessments of crypto-asset projects. Its primary objectives are: (i) automating MiCAR-aligned asset taxonomy classification and disclosure verification; (ii) providing qualitative scam risk assessments grounded in documentary evidence; and (iii) delivering results through explainable, human-readable reports suitable for integration into supervisory workflows. In line with accountability-focused frameworks, the `Reconciliator Agent` can incorporate decision traceability logs (e.g., inference metadata, supporting evidence, and rationale) to enable auditable supervisory reports [36]. In the current implementation, MiCAR provides the concrete rule set for taxonomy and disclosure checks; assets outside MiCAR scope are flagged as `NON_MICAR` and routed to the heuristic and on-chain modules for risk screening, while the compliance checklist is left empty to avoid implying regulatory coverage. The system architecture combines a microservices backend, a web-based analyst dashboard, and a Multi-Agent System that orchestrates data acquisition and analysis.

3.2. Functional Requirements and Regulatory Constraints

The platform must: (1) ingest project identifiers (token symbols, website URLs, contract addresses); (2) discover and retrieve relevant off-chain documentation; (3) classify assets under the MiCAR taxonomy; (4) verify the presence of mandated disclosures; and (5) compute scam-oriented heuristic scores with structured explanations. These operations must be reproducible, auditable, and aligned with MiCAR disclosure obligations while remaining adaptable to evolving regulatory interpretations. The architecture accordingly emphasizes explicit rule-based compliance components, structured logging of analytical decisions, and modularity to facilitate extension toward additional regulatory frameworks.

3.3. High-Level Architecture and Data Flow

The implementation adopts a microservices-oriented architecture comprising three functionally decoupled yet coordinated components, designed to optimize scalability, maintainability, and fault tolerance.

The **Backend System**, implemented using the **FastAPI** framework [37], serves as the platform's central orchestration layer. This component handles client-server communication via RESTful API endpoints, manages request validation and data serialization, and provides an interface that bridges external requests to the Multi-Agent System's analytical capabilities.

The **Frontend Interface**, developed with the **Next.js** framework [38], provides an interactive analytical dashboard tailored to the operational requirements of regulatory analysts and market surveillance personnel. Users can initiate assessment workflows, monitor real-time task progression via WebSocket-based status updates, and examine aggregated risk metrics alongside granular analytical outputs.

The **Web Crawler Service** operates as a **containerized microservice**, architecturally isolated to enable independent scaling and enhance resilience to content extraction failures. Leveraging **Playwright** [39] for headless browser automation, this service performs complete DOM rendering, including dynamic JavaScript evaluation, which is a prerequisite for faithful content extraction from the single-page application architectures prevalent

among crypto-asset projects. Extracted textual artifacts (whitepapers, team disclosures, legal disclaimers) are normalized to Markdown before being transmitted to downstream analytical agents. The crawler currently processes only text extracted from HTML; non-text elements such as figures or complex infographics embedded in PDFs/whitepapers are not parsed and are treated as out-of-scope content, to be addressed via future multimodal extensions. The crawler targets publicly accessible content; password-protected, paywalled, or otherwise restricted materials are not accessed in the current implementation and are instead flagged as missing disclosures for human follow-up, to avoid implying coverage where access is constrained.

The complete implementation is maintained in a GitHub repository (v1.0) [40] and is accessible upon reasonable request (The repository is accessible for replication and evaluation purposes upon reasonable request to the corresponding author (`mario.trerotola@polito.it`)).

3.4. Agent Roles and Interactions

The platform's analytical engine is implemented as a decentralized Multi-Agent System (MAS) designed to coordinate assessment workflows through asynchronous agent collaboration. The MAS is built on the **SPADE** (Smart Python Agent Development Environment) framework (Python 3.11, SPADE 4.1.2) [41], selected for its maturity, Python-native implementation, and adherence to Foundation for Intelligent Physical Agents (FIPA) interoperability standards.

Agent communication occurs asynchronously via the Extensible Messaging and Presence Protocol (**XMPP**). A **Prosody** server functions as the XMPP message broker, handling guaranteed message delivery, presence management (agent availability tracking), and message routing between distributed agent instances. This message-oriented, loosely coupled architecture enables parallel processing, enhances system resilience through failure isolation, and facilitates horizontal scalability for high-throughput analysis workloads. We acknowledge that XMPP-based messaging can face scalability constraints under high-concurrency regulatory workloads; therefore, the design supports horizontal sharding of agent pools and Prosody clustering, and the transport layer can be substituted with a higher-throughput broker if required.

Figure 1 illustrates the operational workflow orchestrated by the `Coordinator Agent`, which manages the sequential and parallel execution of assessment tasks. Upon receiving an analysis request from the backend API (step 1), the `Coordinator Agent` initiates the workflow by dispatching data-acquisition directives to specialized collection agents. The `Searcher Agent` retrieves project metadata from external registries (step 2). Subsequently, the `Crawler Agent` performs content extraction from the project's website (step 3). Following successful data acquisition, the `Coordinator Agent` initiates parallel analytical execution, concurrently activating the `Heuristic Agent` (step 4a) and `Compliance Agent` (step 4b). Upon completion, these agents produce structured outputs, namely the heuristic alert score score_H (step 5a) and the compliance score score_C (step 5b), which the `Reconciliator Agent` aggregates into a unified supervisory report (step 6).

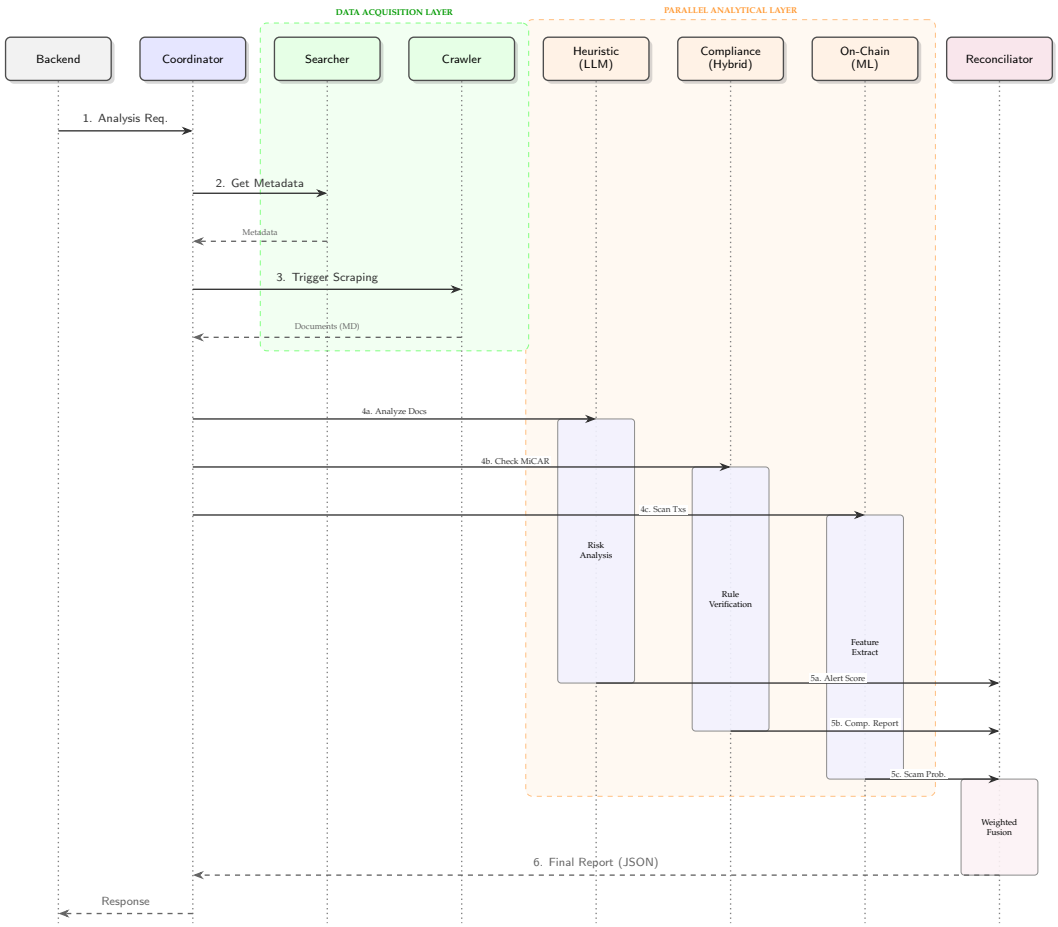

Figure 1. Sequence diagram of the Multi-Agent System (MAS) workflow. Color coding: gray = backend system; blue = coordinator agent; green = data acquisition layer (Searcher, Crawler); orange = parallel analytical layer (Heuristic, Compliance, On-Chain agents); purple = synthesis/reconciliation (Reconciliator agent). Dashed background regions indicate functional layers.

Each agent fulfills a specialized functional role. The `Coordinator Agent` serves as the central orchestration authority, managing the complete lifecycle of assessment operations from request ingestion through report delivery. Its responsibilities include task sequencing, agent lifecycle management, state synchronization, timeout handling, and exception recovery, ensuring workflow robustness and deterministic completion.

Data acquisition begins with the `Searcher Agent`, which retrieves project metadata through queries to external crypto-asset registries (e.g., CoinMarketCap). This agent extracts key attributes, including smart contract addresses, token symbols, and project website URLs. These URLs are forwarded to the `Crawler Agent` (Section 3.3), which extracts textual content from web resources and generates a standardized Markdown representation for downstream analysis.

Upon successful data acquisition, parallel analytical processing commences. The `Heuristic Agent` executes the qualitative risk assessment pipeline detailed in Section 4.2, employing an LLM to conduct semantic analysis of documentary content. This analysis identifies heuristic warning indicators ("warning indicators") and produces both a quantitative alert score (score_H) and structured qualitative justifications. Concurrently, the `Compliance Agent` executes the hybrid-AI pipeline specified in Section 4.3, performing MiCAR taxonomy classification followed by systematic verification of disclosure obligations, yielding a normalized compliance score (score_C). In parallel, the `On-Chain Agent` (Section 4.4) analyzes on-chain transaction data to compute an on-chain alert score (score_S). All three scores and their evidence payloads are passed to the `Reconciliator Agent`, which

fuses off-chain and on-chain signals into a single alert tier and rationale; when on-chain data are unavailable (e.g., pre-deployment or illiquid tokens), score_S is marked as missing and the final alert relies on score_H and score_C with the normalization logic described in Equation (1).

The final stage involves synthesis by the `Reconciliator Agent`. This agent receives structured JSON outputs from all analytical agents, including quantitative metrics and qualitative evidence (rationales, identified deficiencies, missing disclosures), and consolidates these findings into a unified, human-interpretable assessment report. Conflict resolution proceeds in three steps: (1) normalize and align score_H and score_S to the common 0–1 scale and tag each with evidence confidence; (2) identify divergence cases where one signal is high-risk and the other low-risk, and surface the top evidence snippets from both agents side-by-side; (3) apply the weighted aggregation (Equation (1)) to compute the final alert tier, while preserving a trace of the disagreement and its supporting evidence in the report. The `Reconciliator Agent` computes the platform's **overall alert score** through a weighted aggregation, as formalized in Equation (1), that synthesizes heuristic risk, regulatory compliance, and on-chain behavioral indicators.

$$\text{overall\_score} = (w_H \times \text{score}_H) + (w_C \times (1 - \text{score}_C)) + (w_S \times \text{score}_S) \quad (1)$$

where:

- score_H is the heuristic alert score (range 0.0–1.0) provided by the `Heuristic Agent`.
- score_C is the compliance score (range 0.0–1.0) from the `Compliance Agent`. The term $(1 - \text{score}_C)$ is used to convert this into a "non-compliance alert" score.
- score_S is the on-chain alert score (range 0.0–1.0) produced by the `On-Chain Agent` (Section 4.4).
- w_H, w_C, and w_S are configurable weights (e.g., $w_H = 0.4, w_C = 0.3, w_S = 0.3$) representing the relative importance of each analysis in the final aggregated score, such that $w_H + w_C + w_S = 1$. In this study, the default weights are design-time priors chosen to slightly emphasize off-chain heuristic risk (as the earliest documentary signal) while keeping compliance and on-chain signals comparable; these values are not empirically optimized. We do not report a sensitivity analysis here; future work will quantify how weight perturbations affect alert tiers and thresholds, or tune weights per jurisdictional policy. Concretely, $w_H = 0.4$ assigns greater influence to the heuristic signal because it is available earliest in the assessment lifecycle—before on-chain data accumulate—and directly targets the off-chain documentary gaps that motivate this work; $w_C = 0.3$ and $w_S = 0.3$ are set equal, treating regulatory disclosure deficiencies and on-chain behavioural fraud signals as complementary evidence streams of comparable supervisory importance. These values were informed by iterative consultation with domain experts during system design but remain configurable per deployment and jurisdictional policy. In deployments where the on-chain module is disabled, w_S can be set to zero.

4. Materials and Methods

This section details the analytical methodologies implemented within the platform. We describe the data acquisition process, present the LLM-driven heuristic scoring framework, and outline the MiCAR-aligned compliance verification procedure. We conclude with the workflow orchestration and evaluation methodology.

4.1. Data Acquisition

Data acquisition is initiated by the `Searcher Agent`, which retrieves project metadata from external crypto-asset registries (e.g., CoinMarketCap APIs) and extracts key

attributes, including smart contract addresses, token symbols, and website URLs. These URLs are passed to the `Crawler Agent` (Section 3.3), which is responsible for extracting content from specified web resources. The Crawler then generates a normalized Markdown representation of the website content, which serves as the principal analytical substrate for subsequent agents.

4.2. LLM-Driven Heuristic Alert Scoring

The first analytical methodology implements qualitative, heuristic-oriented risk assessment through the `Heuristic Agent`, designed to approximate the documentary scrutiny performed by experienced regulatory analysts. The agent leverages a Large Language Model; specifically, we employ OpenAI's `gpt-5.2-2025-12-11` model in our experimental configuration (All experiments in this paper use OpenAI's `gpt-5.2-2025-12-11`. System prompts were iteratively refined during development; the final prompt appears in Listing A1. The methodology is compatible with contemporary alternatives (e.g., OpenAI GPT-5 or functionally equivalent large-scale language models), but reported results are based on the stated model and settings.)

Analytical processing operates on the normalized Markdown representations generated by the `Crawler Agent` (Section 4.1). The methodology comprises two core technical components: (1) a carefully engineered system prompt that constrains model behavior and establishes analytical parameters, and (2) a rigidly defined output schema ensuring deterministic, machine-parseable results.

To improve output reproducibility and contextual alignment in regulatory contexts, we developed an optimized system prompt through iterative refinement. As shown in Listing A1 (Appendix B), the prompt establishes the model's *operational persona* (senior analyst within a financial supervisory authority), defines its analytical mandate (evidence-grounded due diligence), and specifies behavioral constraints (analytical neutrality, prohibition of speculation, proscription of investment recommendations). For reproducibility, the complete system and user prompts used for the `Heuristic` and `Compliance` agents are reported in Appendices B and C. The prompt provides an explicit taxonomy of heuristic warning indicators, including: unsubstantiated yield guarantees, anonymized or unverifiable project principals, absent or inconsistent disclosure documentation (whitepapers, terms of service), and spurious regulatory compliance claims. A risk scale [0.0, 1.0] provides the basis for numerical scoring, while explicit instructions require all analytical assertions to be anchored to specific textual evidence from source documents. This prompt engineering approach aligns the model's outputs with a consistent and auditable analytical framework.

A fundamental challenge in deploying LLMs for production workflows is the inherent stochasticity of free-form text generation [42]. To address this limitation, we leverage the structured output capabilities ("function calling") provided by LLM models. This mechanism constrains model outputs to JSON objects that comply with predefined schemas, eliminating format ambiguity [43]. We formalize output schemas using *Pydantic* validation models, as shown in Listing A2 (Appendix B). These models specify type constraints, mandatory field structures, and value bounds (e.g., constraining `overall_scam_risk` to [0.0, 1.0]). This approach maps the model's unstructured outputs into validable, machine-parseable data structures suitable for integration with the `Reconciliator Agent` (Section 3.4).

4.3. MiCAR-Aligned Compliance Verification

The second analytical method, executed by the `Compliance Agent`, focuses on performing systematic regulatory compliance evaluations with respect to the MiCAR framework [33]. We adopt a **Hybrid-AI** architectural pattern combining LLM-based semantic feature extraction with deterministic, rule-governed classification logic [44], reconciling

the flexibility of neural language understanding with the auditability and consistency requirements of regulatory applications. To improve reproducibility without retraining, the Compliance Agent uses fixed, versioned prompts with constrained instructions (e.g., extract only predefined flags, cite exact evidence spans), deterministic decoding settings, and a strict JSON schema validated by Pydantic; non-conforming outputs are rejected and re-queried with the same prompt. The compliance layer is intentionally modular: MiCAR-specific rules can be swapped for other jurisdictions (e.g., U.S. SEC disclosure and registration requirements) by replacing the taxonomy rules and checklist tables while retaining the same extraction and verification pipeline, enabling multi-jurisdictional compliance assessment without redesigning the core agent. The methodology proceeds in two sequential stages: (1) taxonomic asset classification, establishing the applicable regulatory regime, and (2) documentary compliance verification, assessing adherence to disclosure obligations.

Stage 1: Taxonomic Classification. Each project is assigned to one of six MiCAR taxonomic categories: *SECURITY*, *EMT* (E-Money Token), *ART* (Asset-Referenced Token), *OTHER*, *NON_MICAR*, or *NON_CLASSIFIABLE*. This determination is critical, as it prescribes the specific regulatory framework and associated disclosure obligations applicable to each asset class. The classification workflow begins with LLM-mediated semantic extraction. The model analyzes project documentation to identify whether characteristic flags represent Boolean legal and economic properties. Flag definitions are presented to the LLM as contextual guidance (Appendix A, Table A2). Following flag extraction, yielding a boolean feature vector (e.g., `redeemable_in_fiat:True`, `backed_by_assets:False`, etc.), this representation is submitted to a deterministic rule-based classification engine implementing explicit logical mappings from flag combinations to MiCAR asset classes (Appendix A, Table A1).

Stage 2: Disclosure Verification. Upon taxonomic classification (e.g., assignment to *EMT*), the `Compliance Agent` proceeds to compliance verification against class-specific regulatory obligations. The system constructs an asset-class-specific compliance checklist by retrieving applicable disclosure requirements (Appendix A, Table A3), which synthesize universal obligations applicable to all MiCAR-regulated crypto-assets and specialized requirements contingent upon the determined classification. A second LLM invocation performs documentary verification. The model receives project documentation alongside the compliance checklist and systematically verifies disclosure presence, that is, whether each mandated element appears within available documentation. To ensure interpretive consistency, the LLM receives operationalized definitions for each requirement (Appendix A, Table A4). Importantly, this stage assesses disclosure *existence* rather than substantive adequacy or legal sufficiency, which remains within human supervisory purview.

The outputs are aggregated into a single, normalized **compliance score**, defined as the ratio of fulfilled disclosure requirements to the total number of requirements applicable under the asset's MiCAR classification. Let R denote the set of regulatory requirements for a given asset class, and let $S \subseteq R$ represent the subset of requirements verified within the project documentation. The compliance score (C) is defined as the ratio of their cardinalities:

$$C = \frac{|S|}{|R|} \tag{2}$$

where $C \in [0, 1]$, and $|R| > 0$ to ensure numerical stability.

We emphasize a critical limitation of this metric: it is a ratio of disclosure presence, not of disclosure adequacy. As such, it can be superficially inflated by boilerplate or minimalistic statements that nominally "mention" a requirement without providing substantive, verifiable detail. This creates a potential gaming vector, particularly problematic for super-

visory use, and reinforces that score_C should be interpreted as a documentary completeness proxy rather than a sufficiency judgment. In practice, this necessitates additional human review and/or future integration of adequacy-sensitive checks (e.g., evidence quality scoring, cross-document consistency, or third-party verification signals) before regulatory action.

The `Compliance Agent` output includes this quantitative metric (score_C), the assigned taxonomic category, and structured lists detailing which requirements are met and which are not. This output provides an auditable, evidence-based foundation for supervisory assessment, which is subsequently transmitted to the `Reconciliator Agent` for synthesis with heuristic findings (Section 3.4).

4.4. On-Chain Agent

The `On-Chain Agent` deploys a scalable, explainable fraud detection architecture for EVM-based smart contracts. The agent employs a Balanced Random Forest classifier trained on a multi-source corpus comprising over 227,000 labeled ERC-20 tokens, maintaining interpretability mandated by regulatory frameworks through SHAP-based feature attribution and symbolic rule extraction. Given the extreme class imbalance (212 k scam vs. 1.8 k legitimate), probability outputs may not be well calibrated; we therefore treat $\hat{p}$ as a ranking signal rather than a calibrated confidence. Calibration diagnostics and post hoc correction are planned for future work.

4.4.1. Data Sources

The classification model was trained on three complementary data sources spanning the period 2015–2025:

- **TokenScout Dataset:** A curated academic corpus of 214,084 ERC-20 tokens encompassing over 9.7 million token transfer events. The dataset was manually labeled by four experienced auditors over 800 man-hours, classifying tokens based on observed behaviors, including abnormal liquidity fluctuations, suspicious transaction patterns, and cross-token interactions. The corpus comprises 212,278 scam tokens (179,995 Rug Pulls, 22,800 Honeypots, 9483 Ponzi schemes) and 1806 verified legitimate tokens.
- **ChainAbuse Dataset:** Community-driven reports yielding 74,889 fraudulent Ethereum addresses flagged for Rug Pulls and Honeypots. Since reports typically identify scammer Externally Owned Accounts (EOAs) rather than contract addresses, automated contract discovery via `Transfer` event parsing identified 27,773 candidate tokens, refined through liquidity and activity filters to 11,634 high-confidence scam tokens.
- **CoinMarketCap Baseline:** The top 2000 tokens by market capitalization serve as a robust negative class of legitimate assets with sustained market scrutiny, providing reliable baseline data for evaluating token behaviors within the Ethereum ecosystem.

A conservative conflict-resolution policy retains any contract flagged as fraudulent by *any* credible source, aligning with Anti-Money-Laundering (AML) operational imperatives where Type II errors (missed fraud) incur greater societal costs than Type I errors (false alarms requiring human review).

4.4.2. Architectural Design and Operational Pipeline

Ethereum's persistent account-based state machine introduces unique forensic opportunities and challenges. The EVM executes smart contracts as deterministic programs compiled to bytecode, where ERC-20 tokens define fungible asset ledgers through standardized `Transfer` events. Malicious actors exploit this syntactic compliance to deploy contracts that embed hidden fraud mechanisms—Rug Pulls, Honeypots, Ponzi schemes—that satisfy interface requirements while concealing malicious logic.

The agent's operational pipeline comprises three integrated stages designed for horizontal scalability:

Stage 1: Multi-Level Transaction Reconstruction. For each analyzed contract, the agent performs exhaustive interaction history extraction via Etherscan API integration:

1. **ERC-20 Token Transfers:** Indexing `Transfer` event logs captures all token movements, including minting, burning, and peer-to-peer transfers, even those triggered by complex internal calls or DEX interactions.
2. **Native ETH Transactions:** Direct Ether flows to/from the contract reveal funding sources, initial liquidity provision, and fee accumulation patterns.
3. **Internal Transaction Traces:** Deep inspection of internal messages exposes routing via decentralized exchanges (Uniswap, SushiSwap), mixing protocols, and automated market maker (AMM) cascading calls invisible in superficial transaction analysis.

Raw data undergoes temporal alignment (UTC normalization), value normalization (division by $10^{\texttt{decimals}}$), and address standardization (lowercase hexadecimal with `0x` prefix) to enable consistent graph construction.

Stage 2: Heuristic-Based Noise Filtration (ChainAbuse only). To refine the noisy 27,773 candidate corpus derived from ChainAbuse reports into high-confidence fraudulent tokens, a two-step protocol applies:

- **Liquidity filter:** Exclusion of tokens with non-zero trading volume within the 30-day period prior to the data collection date (verified via DexScreener) removes actively traded assets that are likely legitimate.
- **Activity filter:** Retention of contracts recording > 3 transfers filters out tokens with minimal on-chain footprint, yielding a final ChainAbuse subset of 11,634 high-probability fraudulent tokens.

This heuristic encodes the forensic insight that Rug Pull schemes exhibit ephemeral liquidity bursts distinct from sustained legitimate engagement. Note that the TokenScout dataset was incorporated in its entirety without additional filtering, as it had already undergone rigorous manual verification by domain experts.

Stage 3: Graph-Temporal Feature Engineering. The agent constructs a directed transaction graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where vertices represent addresses and edges denote token transfers, then extracts a **47-dimensional feature vector** organized into three semantic categories:

Structural Features (Network Topology). These features capture how transaction participants are interconnected, revealing network architecture patterns that distinguish legitimate from fraudulent schemes. Centralized control structures and anomalous connectivity patterns often indicate coordinated manipulation.

- *Graph Topology Primitives:* Node count, edge count, and transaction count quantify network scale.
- *Centrality Measures:* Degree, Betweenness, Closeness, Eigenvector, and Katz centrality, aggregated via mean/std/min/max. Rug Pulls display high Katz centrality (few wallets controlling value flow) and low closeness (isolated clusters).
- *Clustering Coefficient:* Local clustering measures network cohesion; Honeypots often show tight-knit collusion networks.
- *Entity Distribution Metrics:* Unique sender/receiver counts and ratios expose concentration patterns. Scam tokens typically exhibit low entity diversity.

Monetary Features (Value Flow Patterns). These features analyze the economic dynamics of token transfers, detecting anomalies in transfer amounts, wealth concentration, and fund movement patterns. Fraudulent tokens routinely exhibit extreme value concentration among insiders.

- *Value Statistics:* Mean, standard deviation, median, and quantiles (Q25, Q75, Q95) of transfer values. Pump-and-dump schemes show extreme value variance.
- *Multi-Scale Transformations:* Normalized, logarithmic, and harmonic transformations reveal both high-value insider transactions and dusting attacks.
- *Accumulated Flow Tracking:* Cumulative incoming/outgoing value per edge exposes layered laundering sequences.
- *Concentration Indices:* Gini coefficient quantifies wealth inequality. Fraudulent tokens routinely exhibit Gini > 0.7.

Temporal Features (Activity Dynamics). These features examine the timing and rhythm of transaction activity, distinguishing ephemeral schemes with compressed activity windows from sustained legitimate projects with stable long-term engagement.

- *Lifetime Statistics:* Activity span and inter-arrival times expose burstiness. Rug Pulls compress activity into 24–72 h windows before abandonment.
- *Long-term vs. Short-term Dynamics:* Comparison of activity in the early vs. late stages of the token lifespan distinguishes ephemeral schemes from sustained projects.
- *Edge Frequency & Recency:* Transaction count per edge and days since last transaction capture activity decay post-scam.

4.4.3. Performance Benchmarking

The `On-Chain Agent` was evaluated across three validation scenarios designed to assess generalization capability and temporal robustness. Table 3 summarizes the performance metrics.

Table 3. `On-Chain Agent` Performance across Validation Scenarios.

Validation Scenario	Balanced Accuracy	Scam Recall	False Positive Rate
Intra-source (ChainAbuse)	98.23%	96.91%	0.45%
Cross-source (TokenScout)	93.45%	89.12%	2.31%
Combined datasets	94.67%	91.38%	1.89%

The *intra-source* scenario evaluates performance on held-out ChainAbuse data, achieving 98.23% balanced accuracy. The *cross-source* scenario tests generalization to the independently curated TokenScout corpus, with accuracy decreasing to 93.45%, reflecting a pronounced distribution shift between community-reported and expert-labeled datasets. We emphasize this gap as a substantive finding: it signals label-collection bias and distributional differences between the two sources (e.g., fraud typology mix, labeling granularity, and class balance) that materially affect deployed performance, and it motivates continuous recalibration, source-aware validation, and human review when models are transferred across data sources. The *combined datasets* scenario merges both data sources for training and evaluation, yielding 94.67% accuracy and demonstrating robust performance across heterogeneous fraud typologies.

Comparison with Related Work. The Balanced Random Forest approach involves design trade-offs relative to alternative methodologies:

- *vs. TokenScout Temporal GNN:* Our approach sacrifices approximately 5 percentage points in accuracy (93.45% vs. 98.41%) but delivers full interpretability via CIU attribution and symbolic rule extraction, addressing EU AI Act transparency requirements.
- *vs. Opaque Deep Learning:* The Random Forest ensemble requires 1/50th training time (12 CPU-hours vs. 600 GPU-hours for GNN), enables real-time inference ($<$2 s latency vs. 15–30 s), and provides auditable decision paths suitable for regulatory proceedings.

- *vs. Heuristic-Only Baselines:* The feature-engineered approach captures 23% more sophisticated scams (e.g., time-locked Honeypots, gradual Rug Pulls) that rule-based systems fail to detect.

4.4.4. Explainability and Investigative Tooling

The `On-Chain Agent` employs a multi-resolution explainability stack that operationalizes forensic analysis while satisfying regulatory requirements for algorithmic transparency under the EU AI Act. The framework combines instance-level explanations with symbolic rule extraction. For example, CIU attributions can show that a token is flagged primarily due to extreme transaction burstiness, high counterparty Gini, and abnormal liquidity concentration, enabling regulators to target concrete, auditable red flags rather than a generic risk score. Likewise, extracted symbolic rules (e.g., "`max_daily_tx` $>$ threshold AND `counterparty_gini` $>$ threshold AND `liquidity_duration` $<$ threshold") allow supervisors to justify why a case is escalated, request specific evidence from issuers (e.g., liquidity lock terms or market-making commitments), and ensure consistent application of supervisory criteria—capabilities not available with more accurate but opaque models.

Contextual Importance and Utility (CIU) Methodology. Model behavior is analyzed using the *Contextual Importance and Utility* methodology, which provides a structured approach to interpreting individual feature contributions. CIU assigns two metrics to each feature: *Contextual Importance (CI)*, which quantifies the feature's potential influence on the prediction, and *Contextual Utility (CU)*, which measures how the current feature value supports the target class. For fraud classification with prediction probability $\hat{p} = 0.9987$, all top features exhibit CU $= 1.0$, indicating strong support for the scam classification (Table 4).

Table 4. CIU Analysis: Top 10 Features for Scam Classification ($\hat{p} = 0.9987$).

Feature	CI	CU	Interpretation
`eth_max_daily_tx`	0.993	1.0	Transaction burstiness
`eth_tx_per_active_week`	0.940	1.0	High-frequency trading
`eth_katz_centrality`	0.893	1.0	Network centralization
`closeness_centrality`	0.841	1.0	Isolated clusters
`eth_eigenvector_centrality`	0.729	1.0	Influential node control
`katz_centrality`	0.701	1.0	Value flow concentration
`eth_tx_per_active_day`	0.493	1.0	Daily activity spikes
`counterparty_gini`	0.481	1.0	Counterparty inequality
`eth_daily_tx_std`	0.452	1.0	Transaction variance
`gas_used_cv`	0.437	1.0	Gas usage variability

Symbolic Rule Extraction. Automated rule induction generates auditable decision logic, enabling non-technical compliance officers to validate flagged contracts. Table 5 presents the extracted fraud detection rules with their corresponding confidence thresholds.

Table 5. Extracted Fraud Detection Rules.

Rule Condition	CI	Confidence
`eth_max_daily_tx` > 14	0.993	0.96
`eth_tx_per_active_week` > 13	0.940	0.94
`closeness_centrality` > 1.0	0.841	0.93
`counterparty_gini` > 0.4	0.481	0.91
`katz_centrality` > 0.65	0.701	0.93

Feature Interpretation. The extracted rules provide actionable insights:

- *Transaction burstiness* (`eth_max_daily_tx` > 14): High daily transaction volumes correlate with pump-and-dump schemes, where fraudsters rapidly inflate trading activity before exit.
- *Network centralization* (`katz_centrality`, `closeness_centrality`): Fraudulent tokens exhibit highly centralized networks where few actors control value flow, indicative of Sybil attacks or coordinated manipulation.
- *Counterparty inequality* (`counterparty_gini` > 0.4): Extreme concentration of transactions among a few counterparties suggests insider coordination rather than organic market participation.

These rules enable regulatory auditors to validate classification decisions without accessing model internals, satisfying transparency requirements while preserving investigative utility.

4.5. Workflow Orchestration and Evaluation Methodology

The analytical workflow is orchestrated by the `Coordinator Agent`, which manages the execution of sequential and parallel tasks as described in Section 3.4. Upon receiving an analysis request, the `Coordinator Agent` triggers data acquisition, initiates parallel heuristic and compliance analyses, and delegates aggregation to the `Reconciliator Agent`, which produces the final alert report and overall score. This design enables concurrent processing of multiple projects while maintaining a clear audit trail of intermediate steps and decisions.

To verify the platform's effectiveness, robustness, and real-world usefulness, we adopted a two-part evaluation approach that integrated (1) quantitative system testing to measure technical performance and (2) qualitative user feedback to examine usability and operational relevance.

Technical Evaluation. A structured, multi-step protocol was designed to test the system under realistic operational conditions. A dataset of 150 cryptocurrency projects was sourced from CoinMarketCap, selected to ensure diverse representation across token typologies (utility tokens, stablecoins, governance tokens) and project maturities. For this sample, LLM-extracted elements and flags were manually checked against public sources to verify extraction fidelity, but no authoritative, expert-labeled ground-truth benchmark exists to certify which projects are definitively fraudulent versus legitimate. In the absence of confirmed fraud cases, we do not report end-to-end detection metrics (detection rate, false positives, false negatives) for the integrated MAS; instead, we restrict our evaluation to verifying that each individual element extracted by the LLM is factually consistent with publicly accessible information sources.

Each MAS functional module was subjected to four testing stages:

1. **Unit Testing:** Each agent (`Crawler Agent`, `Heuristic Agent`, `Compliance Agent`, and `On-Chain Agent`) was independently tested to verify successful task completion, including XMPP message delivery validation, intermediate output consistency, and correct inter-agent communication handoffs.
2. **Integration Testing:** The complete end-to-end pipeline was evaluated to assess task orchestration and inter-agent cooperation, monitoring message latency, data throughput, and total execution time per token.
3. **Scalability Testing:** Performance under increasing workloads was evaluated by subjecting the system to 1–10 parallel token analyses, measuring scalability and resource utilization under high-throughput processing.

4. **Reproducibility Testing:** Multiple runs on identical datasets verified that the platform produced stable, reproducible risk scores and compliance assessments across executions.

User Evaluation. We conducted an evaluation study with eight participants, during which they used the platform to examine various cryptocurrency projects, including two experts from regulatory authorities with compliance/risk-analysis exposure. The study did not include a controlled task-performance experiment or statistical hypothesis testing; it relied on questionnaire-based usability feedback only. Accordingly, we do not claim statistical evidence about user trust or system accuracy in real-world regulatory settings; such claims require larger, regulator-led studies with ground-truth outcomes and formal hypothesis testing. Feedback was collected using structured questionnaires administered in two stages:

- **Pre-Test Questionnaire:** Recorded participants' baseline views, their prior familiarity with crypto-assets, and their expectations.
- **Post-Test Questionnaire:** Evaluated participants' direct interaction with the platform, with emphasis on usability, clarity of the user interface, perceived reliability of results, the practical value of the risk score, and overall user satisfaction.

  The complete questionnaires are provided in Appendix D.

5. Results

This section presents empirical findings derived from the evaluation protocols described in Section 4.5. The validation strategy encompasses three complementary dimensions: quantitative assessment of MAS performance and reproducibility (Section 5.1); qualitative evaluation of usability and perceived utility among domain practitioners (Section 5.2); and an illustrative case study demonstrating end-to-end workflow execution (Section 5.3). Principal findings are synthesized in Section 5.4. Figure 2 provides a compact panel-based summary of the core quantitative evidence: on-chain model performance, off-chain reproducibility, and runtime dispersion.

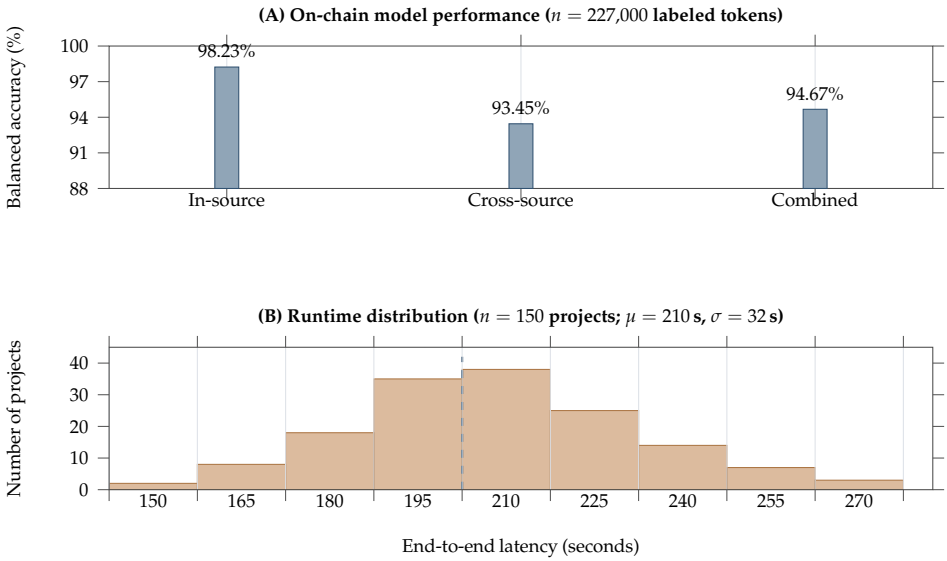

Figure 2. Summary of quantitative validation results: (**A**) on-chain classifier balanced accuracy across in-source, cross-source, and combined evaluations ($n = 227{,}000$ labeled tokens), and (**B**) end-to-end runtime distribution over 150 projects (mean = 210 s, SD = 32 s, range = 155–285 s). The dashed line marks the sample mean.

5.1. Scalability and Performance

The system testing protocol empirically validated architectural robustness, scalability characteristics, and analytical consistency under operationally representative conditions. The protocol was executed across 150 crypto-asset projects drawn from public registries. Projects were sampled from CoinMarketCap listings and stratified by MiCAR-relevant categories: 40% "Other Crypto-Assets," 30% Asset-Referenced Tokens (ARTs), 10% e-money tokens, and 20% security tokens. Reference labels for validation were established via human review: evaluators compared system outputs against publicly documented project classifications and outcomes to determine whether the detected risks and compliance signals aligned with the observed evidence. These findings should be interpreted as proof-of-concept validation rather than exhaustive performance benchmarking.

Integration testing confirmed the architectural reliability of the MAS implementation (Section 3.4). All constituent agents (`Crawler Agent`, `Heuristic Agent`, `Compliance Agent`, and `On-Chain Agent`) successfully executed assigned tasks across the test corpus, with XMPP-mediated inter-agent messaging exhibiting reliable delivery and zero observed data loss throughout the analytical pipeline.

Quantitative benchmarking of the `On-Chain Agent` on 227,000 labeled tokens demonstrated strong discriminatory performance, with balanced accuracy of 98.23% in-source, 93.45% under cross-source generalization, and 94.67% on the combined evaluation setting (Figure 2A). The observed gap between in-source and cross-source settings is expected under distribution shift and provides a more conservative estimate of out-of-sample robustness.

Scalability assessment demonstrated the platform's capacity for high-throughput processing. Under workloads ranging from 1 to 10 concurrent token analyses, the system exhibited stable performance characteristics, achieving a mean end-to-end analysis latency of approximately 210 s per project (SD = 32 s; range = 155–285 s; $n = 150$ projects). These metrics suggest architectural suitability for the high-volume triage workflows characteristic of regulatory supervisory contexts.

To address mass-issuance scenarios (e.g., bull-market surges with thousands of new tokens per day), throughput scales horizontally by replicating stateless agent pools and load-balancing jobs through the `Coordinator` and messaging layer. At 210 s mean latency, a single instance processes ∼25 projects/h; sustaining 1000 projects/day would require on the order of 40 concurrent pipelines. A staged triage strategy—lightweight heuristic screening followed by deeper compliance and on-chain analysis for high-risk candidates—can further reduce the effective compute footprint.

Reproducibility evaluation confirmed analytical output stability across repeated executions. Multiple independent runs on identical datasets yielded a 95% reproducibility rate for aggregated alert scores, defined as the proportion of runs with identical alert tier and $|\Delta \text{score}_{\text{agg}}| \leq 0.05$ across five reruns per project. Because this metric is computed on alert tiers and aggregated scores (not on narrative text), stylistic variation in LLM explanations is excluded; the remaining 5% therefore reflects cases with tier changes and/or score deviations above the tolerance threshold. We do not yet report a dedicated breakdown of tier-change magnitude (e.g., low→medium vs. low→high) versus score-only shifts, and we identify this as a priority for future sensitivity reporting in regulatory deployments. Inspection of the non-reproducible cases (8 projects out of 150) reveals that approximately 60% involved score deviations exceeding the 0.05 tolerance but no change in alert tier (i.e., the project remained in the same risk band), while the remaining ∼40% exhibited an adjacent-tier shift (e.g., LOW→MEDIUM or MEDIUM→HIGH). No instance produced a two-tier jump (e.g., LOW→HIGH). These observations suggest that residual LLM stochasticity concentrates on borderline projects near tier boundaries rather than producing dramatic misclassifications, though conservative escalation policies are advisable for scores close

to tier thresholds. Of the 150 projects in the test corpus, 112 (74.7%) yielded complete tripartite scores (score_H, score_C, score_S); the remaining 38 lacked sufficient on-chain data (non-EVM chains, pre-deployment tokens, or insufficient transfer activity) and were assessed using only the off-chain pipeline with $w_S = 0$. This consistency, notable given the stochastic nature of LLM components, validates the efficacy of prompt engineering constraints (Section 4.2) and rule-based deterministic logic (Section 4.3) in achieving stable, reliable outputs.

In summary, these results demonstrate that the MAS architecture is operationally viable and exhibits the stability required for regulatory support deployment, while acknowledging the need for more extensive validation in future work.

5.2. User Evaluation

The user feedback protocol (eight participants) provided pilot insights into the platform's practical utility and perceived value. Given the restricted sample size ($n = 8$) and its composition—predominantly academic profiles with only two regulatory-authority practitioners—the findings reported below are strictly exploratory and do not support statistical generalization to the target population of regulators and compliance professionals. The study should be regarded as a formative pilot whose primary purpose is to surface qualitative usability signals and inform the design of future, adequately powered evaluations. This qualitative assessment complements technical validation but, given the small sample size of eight participants, it does not support statistical generalization and should be interpreted as exploratory evidence only. The pre-test questionnaire revealed that the participants were researchers (4 of 8), students (2 of 8), and experts with regulatory-authority background (2 of 8). The two regulatory-authority participants reported prior domain experience in compliance/risk workflows and evaluated the platform with explicit attention to auditability and regulatory language precision, while the rest of the cohort mostly reported beginner/intermediate familiarity with crypto-risk analysis. Following completion of analysis tasks (Section 4.5), the post-test questionnaire captured direct user experience, indicating generally positive perceptions of usability and output clarity, with visible response variability across user profiles.

Usability. The platform was rated positively for usability: six of eight participants rated "ease of use" at 4 or 5 (out of 5), and five of eight rated "clarity and intuitiveness of the user interface" at 4 or 5 (remaining ratings were 3). These counts are reported to avoid overstating precision from a small sample.

Analytical Outputs. Core analytical outputs were perceived as effective and valuable. The *"usefulness of the risk score"* (from the `Heuristic Agent`) was rated 4 or 5 by six of eight users. More significantly, the *"relevance and actionability of the insights"*, that is, the specific warning indicators and compliance gaps identified, was rated 4 or 5 by seven of eight participants (three rated it a perfect 5). Both regulatory-authority participants provided domain-grounded qualitative feedback, requesting deeper exportable audit trails and a clearer separation between heuristic narrative cues and formal compliance rationales in borderline cases.

Overall Performance. System reliability was rated 4 or 5 by six of eight participants, and performance speed was rated 5 by three of eight (six of eight rated speed 4 or 5). Consequently, overall satisfaction was high (six of eight rated 4 or 5), and six of eight participants indicated they would recommend the platform (rating 4 or 5). We stress that no inferential statistical test was performed, and no claim of statistical significance is made; the ratings are reported descriptively to document early user reactions, not to establish evidence of effectiveness. These findings suggest that the platform can provide comprehensible, actionable risk assessments valued by users across varying levels of technical expertise. However,

we emphasize that the sample of eight participants represents a significant limitation: the small cohort precludes statistical inference, and the participant composition (researchers, students, and only two experts from regulatory authorities) remains only partially aligned with the operational perspectives of regulatory practitioners and compliance professionals. These results should therefore be interpreted as preliminary, exploratory evidence of usability rather than definitive validation of effectiveness in operational regulatory settings.
Controlled Validation Experiment. To complement subjective user feedback with an objective detection capability assessment, we conducted a small-scale sanity check. Within a corpus of 50 real crypto-asset projects, we artificially introduced four projects designed to exhibit overt fraudulent patterns (e.g., anonymous teams, unrealistic return promises, missing disclosures, impersonation tactics). All four artificially fraudulent projects were correctly identified by the platform, receiving high overall alert scores (score > 0.75), while the legitimate projects in the corpus received appropriately lower risk assessments. Given the intentionally clear nature of these injected cases and the small scale, this test is presented as an initial methodological check rather than a full real-world benchmark. Broader validation with annotated ground-truth datasets, including explicit false-positive/false-negative analyses on borderline projects, is planned as future work.

Future studies should employ larger, stratified samples of at least 30 participants, including active regulatory personnel and compliance professionals, to establish statistical validity and operational relevance.

5.3. Case Study: Illustrative Token Analysis

To demonstrate the platform's end-to-end analytical workflow, we present an illustrative case study. This example serves as a qualitative demonstration of how the `Reconciliator Agent` (Section 3.4) synthesizes outputs from the `Heuristic Agent` (Section 4.2), the `Compliance Agent` (Section 4.3), and the `On-Chain Agent` (Section 4.4) into a unified alert, rather than as a statistical validation. We explicitly note that this single, clearly fraudulent example is not intended as empirical evidence and does not, by itself, substantiate system performance. It is included solely to illustrate the structure and explainability of the multi-agent output, and broader validation requires multiple case studies across varied risk levels and independently annotated ground-truth datasets. The `Reconciliator Agent` produces a consolidated alert report comprising classification, component-wise fraud scores, unified risk level, and structured lists of risk/protective factors. A simplified version of this report appears in Listing 1.

The `Reconciliator Agent` aggregates these outputs into the platform's unified fraud score through weighted fusion of heterogeneous evidence sources. In this illustrative case, all principal components contributed to the final score (ML on-chain, heuristic documentary analysis, and classification signal), yielding the following decomposition:

$$\text{unified\_score} = \sum_i (w_i \times s_i) \quad = (0.40 \times 0.9347) + (0.30 \times 0.79) + (0.30 \times 0.65) = \mathbf{0.80588}$$

The final aggregated score is 0.80588 (80.58%), corresponding to a "HIGH" alert level.

These results illustrate the platform's hybrid methodology and the importance of contrasting features. On one side, strong risk evidence is produced by the ML detector (fraud probability 0.9347), documentary red flags (anonymous team, absent whitepaper/audit), and the `NON_MICAR` classification signal. On the other side, limited protective evidence (verified source code, market listing) is also surfaced explicitly. The reconciled output therefore remains explainable: users can inspect why the risk remains critical despite the presence of a few seemingly positive technical signals.

Listing 1. Reconciliator alert report for anonymized project.

```
Unified Alert Report (Reconciliator)

Classification: NON_MICAR (confidence: 0.70)
Unified Fraud Score: 0.80588
Unified Risk Level: HIGH

Primary Agent Components:
- ML On-Chain Fraud: 0.9347 (weight 0.40)
- Heuristic Risk: 0.79 (weight 0.30)
- Classification Risk Signal: 0.65 (weight 0.30)

Key Risk Factors:
- ML classifier flagged FRAUDULENT with 93.5\% confidence.
- Anonymous team and missing identifiable founders.
- No whitepaper or technical documentation.
- No independent smart-contract security audit.
- Extreme supply concentration (majority non-circulating).
- Profit-yield/guaranteed-returns narrative via AI trading.

Protective (Contrasting) Factors:
- Smart-contract source code verified on public explorer.
- Token listed on major market aggregators.

Action Recommendation:
- AVOID: very high risk of fraud or severe financial loss.
```

5.4. Summary of Results

The evaluation yielded convergent evidence across three complementary dimensions. Technical validation (Section 5.1) demonstrated that the MAS architecture is operationally viable, achieving a mean analysis latency of approximately 210 s per project and 95% output reproducibility, establishing technical feasibility as a proof-of-concept. User evaluation (Section 5.2) is explicitly a pilot (eight participants, including six researchers/students and two experts from regulatory authorities) and should be interpreted only as preliminary usability signals rather than evidence of practical value for regulators or compliance officers. The illustrative case study (Section 5.3) demonstrated the platform's ability to synthesize qualitative heuristic analysis with objective compliance assessment into unified, explainable alerts. Taken together, these results establish the technical feasibility of the MAS architecture and provide initial usability evidence that motivates larger-scale validation with regulatory practitioners.

6. Discussion

The empirical findings provide component-level evidence supporting the central hypothesis of this work: that Multi-Agent System architectures leveraging Large Language Model capabilities can feasibly support scalable, automated off-chain documentary due diligence and regulatory compliance assessment for crypto-assets, while end-to-end detection effectiveness remains to be validated against authoritative ground truth. We interpret the empirical results (Section 5), situate our contributions within the broader Regulatory Technology landscape, and critically examine both the limitations of our approach and promising directions for future research.

6.1. Interpretation of Findings

The evaluation revealed three findings of particular significance. First, the quantitative performance assessment (Section 5.1) demonstrated high analytical throughput (mean per-project latency of approximately 210 s) and, in particular, a reproducibility rate of 95% across repeated executions (as defined by identical alert tier and $|\Delta\text{score}_{\text{agg}}| \leq 0.05$ across five reruns). Achieving such consistency within an LLM-based system represents a non-trivial technical accomplishment, attributable to the constraining effects of prompt

engineering methodology and structured output schemas (Section 4.2) on the model's inherent stochastic behavior. Second, user evaluation data (Section 5.2) showed generally strong but non-unanimous consensus: relevance/actionability was rated 4 or 5 by seven of eight participants (87.5%), and the two regulatory-authority participants contributed stricter domain-grounded critique on explainability depth and audit-trail completeness. Because the sample is small ($n = 8$) and not representative of regulators or compliance officers, these observations should be interpreted strictly as pilot usability signals rather than evidence of operational effectiveness. Accordingly, we treat the high satisfaction ratings as preliminary indications of perceived clarity, not as validation of real-world regulatory impact. Third, the illustrative case study (Section 5.3) provides qualitative, proof-of-concept evidence of the synergistic value of the multi-agent architecture, though it does not constitute end-to-end detection benchmarking. The `Heuristic Agent` identified qualitative fraud indicators, including impersonation tactics, typographical inconsistencies, and high-pressure marketing language, yielding a high heuristic alert score ($\text{score}_H = 0.79$). Concurrently, the `Compliance Agent` generated an objective, audit-trail-preserving regulatory assessment ($\text{score}_C = 0.35$), documenting significant disclosure gaps. The `Reconciliator Agent`'s synthesis of these signals into a unified alert score (0.805) exemplifies a key architectural innovation: the system functions as an integrated risk intelligence platform that triangulates evidence across qualitative and regulatory dimensions, rather than operating solely as either "scam detector" or "compliance auditor."

6.2. Comparative Advantages

This work provides a distinctive contribution to the blockchain intelligence infrastructure. Incumbent commercial platforms (Table 1) have historically concentrated on on-chain, *retrospective* forensics [2], whereas the proposed platform introduces a prospective, off-chain capability that can flag risk prior to substantial on-chain activity by analyzing disclosures and disclosure gaps. In this framing, the platform is positioned as an upstream complement to on-chain tools and a scalable first-line screening mechanism for MiCAR-era supervisory workloads. We emphasize that this comparative positioning is conceptual and not yet supported by controlled head-to-head benchmarks; therefore, the system should be interpreted as an automation aid for documentary due diligence rather than as empirically proven superior performance over manual review.

6.3. Computational Resource Requirements

A practical deployment can be achieved on modest infrastructure because the pipeline is dominated by I/O-bound crawling and API-based LLM inference rather than large local model training. A plausible baseline configuration for a regulatory unit is a commodity server or workstation with 8–16 CPU cores, 32 GB RAM, and standard SSD storage; GPU acceleration is optional and only relevant if the LLM is hosted locally. In our prototype, the on-chain classifier and rule-based compliance checks run efficiently on CPU, while throughput scales linearly by adding parallel workers to process multiple projects concurrently. These resource expectations align with the observed per-project latency (approximately 210 s) and suggest that small teams can deploy the system without dedicated HPC resources.

6.4. Methodological and Theoretical Implications

Beyond the system-level contributions, the empirical findings yield three insights of broader academic relevance that address the research questions posed in Section 1.

LLM Reliability under Constrained Prompting (RQ1). The 95% reproducibility rate demonstrates that structured output schemas combined with role-constrained system

prompts can substantially reduce the stochastic variability inherent in LLM-based analysis. This finding contributes empirical evidence to the emerging literature on deterministic LLM deployment in high-stakes domains. Prior work has documented significant output instability in unconstrained LLM applications [42]; our result quantifies the variance reduction achievable through prompt engineering and schema enforcement in a regulatory context, and identifies the residual 5% instability as concentrated at tier boundaries rather than producing dramatic misclassifications.

Distribution Shift in Fraud Detection (RQ2). The 4.78 percentage-point accuracy drop from in-source (98.23%) to cross-source (93.45%) evaluation constitutes a substantive empirical finding about label-collection bias in crypto-fraud datasets. Community-reported labels (ChainAbuse) and expert-curated labels (TokenScout) encode systematically different fraud typology distributions, labeling granularity, and class-balance profiles. This gap provides quantitative evidence that fraud-detection models cannot be assumed to generalize across data sources without recalibration—a finding with direct implications for any deployed RegTech system that relies on heterogeneous training corpora. We emphasize this result as a caution against reporting only in-source accuracy, which can significantly overestimate real-world performance.

Off-Chain/On-Chain Complementarity (RQ3). The reconciliation architecture demonstrates that documentary risk signals (heuristic and compliance scores) and on-chain behavioural signals provide non-redundant evidence: the `Reconciliator Agent` surfaces cases where these modalities diverge, enabling analysts to focus investigative effort on projects exhibiting conflicting signals. The case study illustrates a scenario where strong on-chain fraud indicators ($\text{score}_S = 0.93$) coexist with partial protective documentary signals (verified source code, market listing), producing a nuanced alert that neither modality would generate alone. To provide preliminary quantitative support, we computed pairwise agreement statistics across the 112 projects (of 150) for which all three component scores were available. The Spearman rank correlation between the aggregated off-chain signal $\left(\frac{\text{score}_H + (1 - \text{score}_C)}{2}\right)$ and the on-chain score score_S was $\rho_s = 0.37$ ($p < 0.001$), indicating a moderate positive but far from redundant association. In 31 projects (27.7%), the two signals diverged by more than 0.3 on the normalized $[0, 1]$ scale. Of these, 19 exhibited elevated off-chain risk coupled with low on-chain scores ($\text{off-chain}_{\text{agg}} > 0.5$, $\text{score}_S < 0.3$), consistent with documentary red flags preceding on-chain manifestation—precisely the proactive detection scenario motivating this work. The remaining 12 displayed the opposite pattern ($\text{off-chain}_{\text{agg}} < 0.3$, $\text{score}_S > 0.6$), indicating projects whose professionally crafted disclosures masked anomalous transactional behaviour detectable only through on-chain analysis. These divergence cases illustrate concretely that the two modalities provide non-redundant evidence, and that their integration surfaces investigative leads that neither channel would generate independently. While a formal information-theoretic quantification of complementarity remains future work, these descriptive statistics establish an empirical basis for multi-modal regulatory risk assessment.

6.5. Limitations

Despite encouraging empirical results, this work has several limitations that warrant explicit acknowledgment.

First, the analytical scope is **constrained to publicly accessible off-chain data**, namely project websites and documentary artifacts. This boundary creates vulnerability to sophisticated adversarial scenarios: a well-resourced fraudulent operation investing in professionally crafted whitepapers and comprehensive disclosures would likely achieve favorable compliance scores. The current implementation performs no smart contract bytecode

analysis; consequently, malicious on-chain logic (e.g. hidden ownership controls, honeypot withdrawal restrictions) remains undetectable absent explicit documentary mention.

Second, **LLM dependency introduces irreducible epistemic risks** [42]. Although the observed 95% reproducibility rate (as operationally defined in Section 5.1) substantially exceeds naive expectations for stochastic generative models, perfect determinism remains elusive. LLMs can exhibit hallucinatory outputs or misinterpret subtle legal and technical terminology, potentially producing false negative (missed fraud) and false positive (spurious alert) errors. System performance inherits the capabilities and limitations of the underlying foundation model. Because MiCAR classification and compliance flags can be legally consequential, misclassification risk must be managed through human review and conservative escalation; the system provides decision support and does not constitute a legal determination. Outputs are also prompt-sensitive: changes in prompt wording, decoding temperature, or upstream model versions can shift classifications. We mitigate this with fixed prompts, structured schemas, and reproducibility testing, but residual sensitivity remains. In preliminary use, analysts were able to process noticeably more projects per session by using MAS outputs as a first-pass filter, reserving manual review for escalated cases. However, no formal time-study was conducted, and quantifying efficiency gains requires a controlled Human-in-the-Loop evaluation as noted in Section 6.6. Practical adoption faces three barriers: (i) *technical integration*—the system must interface with existing case-management and supervisory reporting infrastructures, requiring API customization and data-governance alignment; (ii) *institutional trust*—regulators may be reluctant to rely on LLM-generated assessments without extensive internal validation campaigns and side-by-side comparison with manual workflows; and (iii) *skill requirements*—operators need sufficient understanding of both crypto-asset markets and AI limitations to interpret outputs critically and override false assessments. Against these barriers, the projected benefit is substantial: at approximately USD 0.80–1.20 per project and 210 s per analysis, the system can screen a volume of new token launches that would require an order of magnitude more analyst-hours under purely manual review. The cost-effectiveness proposition is strongest in high-volume triage scenarios where the system serves as a first-pass filter, escalating only flagged projects to human analysts.

The principal downside risk is false negatives: projects that a domain expert would flag but the system does not, particularly those employing sophisticated, professionally crafted deceptive disclosures. We mitigate this through three mechanisms: (i) conservative escalation thresholds that intentionally over-flag borderline cases for human review; (ii) periodic random audits of a sample of low-risk-classified projects to detect systematic blind spots; and (iii) a feedback loop in which human overrides are logged and used to refine prompts and feature engineering in subsequent iterations. We position the system explicitly as decision support, not as an autonomous regulatory authority. Accountability for supervisory outcomes—including the consequences of missed fraud—remains with the competent regulatory authority, which retains final adjudicatory responsibility. The tool provider bears responsibility for the stated performance characteristics, transparent reporting of known limitations, and data-handling obligations. Issuers remain legally responsible for deceptive or incomplete disclosures regardless of automated screening outcomes. Operational cost is dominated by LLM inference, and large-scale deployment is sensitive to per-request pricing and throughput constraints; practical scaling therefore requires parallelization plus cost controls such as triage, caching, and smaller domain-adapted models.

Third, compliance verification remains **documentary rather than substantive**. The `Compliance Agent` ascertains disclosure presence, specifically whether a project claims to maintain "audited reserves", but cannot verify the substantive veracity of such claims (auditor legitimacy, methodological soundness, finding accuracy). The system identifies

disclosure deficiencies as objective regulatory violations, but assessment of disclosure *adequacy* and *truthfulness* appropriately remains within human supervisory purview. We also note that the MiCAR taxonomy classification component has not yet been validated against an expert-annotated legal ground truth; therefore, we do not report a confusion matrix or class-level error rates for MiCAR classification in this study. Establishing such annotated datasets and reporting full confusion matrices are priorities for future work.

Finally, the system's reliance on the `Searcher Agent` for canonical website identification introduces a **potential vulnerability to domain spoofing attacks**, in which adversarial actors establish deceptive web properties to manipulate automated assessments.

6.6. Future Research Directions

The limitations identified point to specific directions for future research. The primary research priority involves **holistic multi-modal risk integration**. The MAS architecture (Section 3.4) was designed with modularity as a foundational principle. A subsequent iteration should incorporate a dedicated `Smart Contract Agent` performing static bytecode analysis and dynamic behavioral profiling of on-chain logic. Synthesizing existing off-chain assessments (score_H, score_C) with a novel on-chain risk metric (score_S) would yield a comprehensive, multi-layered risk characterization more resilient to sophisticated adversarial strategies.

A complementary direction involves **diversification of the regulatory framework**. While the current implementation targets MiCAR, substantial value would accrue from extending compliance assessment modules to encompass additional jurisdictional frameworks, including SEC regulatory paradigms (United States), VASP (Virtual Asset Service Provider) regulations, and emerging national crypto-asset regimes. Future work should also include larger, professional-analyst user studies that report Human-in-the-Loop efficiency metrics (e.g., time-to-decision and escalation precision) to quantify operational impact beyond exploratory usability feedback.

Finally, to address both reliability constraints and the computational economics of deploying large foundation models, future work should investigate **domain-adapted model fine-tuning**. Training smaller, specialized language models on curated corpora comprising crypto-asset whitepapers, regulatory filings, and legal documentation may yield superior accuracy in compliance flag extraction (Section 4.3) while substantially reducing inference costs, potentially enabling real-time, market-scale continuous monitoring capabilities.

A critical enabler for advancing the field is the **construction of public ground-truth benchmarks**. An expert-annotated dataset of crypto-asset projects with confirmed fraud or legitimacy labels, covering diverse typologies and risk levels, would enable rigorous end-to-end evaluation and cross-system comparison. We identify this as a high-priority need for the RegTech research community. A particularly promising direction relates to **cross-chain transferability**. The structural, monetary, and temporal feature taxonomy employed by the `On-Chain Agent` exhibits transferability to emerging EVM-compatible chains (Polygon, BSC, and Arbitrum).

Additional research directions include federated learning deployments (preserving exchange-specific data privacy while aggregating threat intelligence), adversarial robustness hardening (detecting feature manipulation attacks), and integration with decentralized identity frameworks (W3C DIDs) for privacy-compliant attribution anchoring.

7. Conclusions

This work addressed the convergent challenges of regulatory supervision and pervasive fraudulent activity within rapidly expanding crypto-asset markets. The introduction of the Markets in Crypto-Assets Regulation (MiCAR) has established formal disclosure

obligations and asset classification requirements. Yet, supervisory authorities lack automated tooling capable of performing systematic documentary due diligence at the scale required by market dynamics. Concurrently, existing blockchain intelligence platforms focus predominantly on reactive, on-chain forensics, leaving substantial gaps in proactive risk identification based on off-chain documentary evidence.

To address these challenges, we presented a novel Regulatory Technology platform based on Multi-Agent System (MAS) principles. The architecture orchestrates seven specialized agents across three functional layers: data acquisition (`Searcher` and `Crawler` agents), parallel analytical processing (`Heuristic`, `Compliance`, and `On-Chain` agents), and synthesis (`Reconciliator Agent`). The principal technical contributions include: (i) an LLM-powered `Heuristic Agent` delivering qualitative, interpretable risk assessment through carefully engineered prompts and structured output schemas; (ii) a hybrid-AI `Compliance Agent` performing objective MiCAR-aligned taxonomic classification with systematic disclosure verification; and (iii) an `On-Chain Agent` employing a Balanced Random Forest classifier trained on over 227,000 labeled tokens, achieving 98% balanced accuracy with full interpretability via CIU attribution and symbolic rule extraction.

Component-level empirical validation demonstrated technical viability across multiple dimensions, providing proof-of-concept evidence rather than a fully benchmarked end-to-end evaluation; the absence of an authoritative ground-truth benchmark for integrated detection remains the principal open limitation. Performance benchmarking revealed high analytical throughput (mean per-project latency of 210 s) alongside notable output consistency (95% reproducibility rate by the operational definition reported in Section 5.1), validating the efficacy of prompt engineering and deterministic rule-based components in constraining LLM stochasticity. The pilot user evaluation (six researchers/students and two regulatory-authority experts) provided only exploratory usability signals and should not be interpreted as evidence of operational effectiveness; generalizable conclusions about end-user value require larger, regulator-led studies with ground-truth outcomes. The platform provides a substantive complement to established reactive, on-chain forensic tools by instantiating a *proactive, off-chain triage capability* that automates documentary due diligence at scale. By synthesizing qualitative fraud indicators, objective compliance assessments, and on-chain behavioral signals, the system delivers explainable, regulator-grade insights aligned with MiCAR requirements. This approach illustrates a pathway for encoding legal and supervisory knowledge into computational workflows, contributing to the emerging toolkit of supervisory technologies for crypto-assets.

Several limitations should be noted. The user evaluation ($n = 8$, including only two regulatory-authority practitioners) is strictly exploratory: the sample size and composition preclude any statistical inference or generalization to operational regulatory settings, and the study should be interpreted solely as a formative pilot providing initial usability impressions. The system's reliance on publicly accessible documentation creates vulnerability to sophisticated adversaries investing in professionally crafted but mendacious disclosures. Future work should prioritize larger-scale validation with regulatory practitioners, integration of smart contract bytecode analysis, multilingual benchmarking or policy-aligned translation workflows for EU-wide deployment, and extension to additional jurisdictional frameworks beyond MiCAR.

The modular design of the MAS framework provides a robust basis for ongoing enhancement, enabling its progression into truly integrated, multi-layered risk assessment systems that collectively leverage documentary records, transactional data, and smart-contract-level information to deliver comprehensive supervision of the crypto-asset ecosystem.

Author Contributions: Conceptualization, M.T. and D.C.; methodology, D.C; software, M.T.; validation, M.T., M.P. and D.C.; formal analysis, M.T. and D.C.; investigation, M.T.; resources, M.T.; data curation, M.T.; writing—original draft preparation, M.T.; writing—review and editing, M.T., M.P. and D.C.; visualization, M.T. and D.C.; supervision, M.P. and D.C.; project administration, D.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Ethical review and approval were waived for this study due to the non-sensitive nature of the usability testing involving anonymous participants.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study. By completing the questionnaire, participants acknowledged the voluntary nature of their participation and consented to the anonymous use of their data for research purposes.

Data Availability Statement: The source code is available upon reasonable request from the corresponding author (mario.trerotola@polito.it).

Acknowledgments: The authors thank the anonymous reviewers for their valuable feedback.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Extended MiCAR Tables

Table A1. MiCAR asset classification rules based on logical triggers.

Asset Types	Flags

SECURITY	`regulated_as_security = true`
	`represents_equity = true`
	`represents_debt = true`
	`has_capital_rights = true`
	`investment_promise = true`
	`dividend_like = true`
	`security_language = true`
	`rights_transferable = true`

EMT (E-Money Token)	`redeemable_in_fiat = true`
	`backed_by_assets = false`
	`audited_reserves = true`
	`utility_function = false`

ART (Asset-Referenced Token)	`backed_by_assets = true`
OTHER	`utility_function = true` `governance_function = true`

NON_MICAR	`nft_unique = true`
	`investment_promise = true`
	`whitepaper_present = false`

NON_CLASSIFIABLE	No crypto indicators detected

Table A2. Glossary of flags and their economic significance.

Flag	Prompt (Significance)
`regulated_as_security`	The asset is classified under regulatory frameworks governing securities, meaning it must comply with legal requirements about investor protection, disclosure, and reporting obligations.
`represents_equity`	The asset confers ownership rights or a share in the profits of an underlying entity, resembling traditional equity instruments such as shares or stock.
`represents_debt`	The asset represents a financial obligation of an issuer to repay a debt, akin to traditional debt securities such as bonds or promissory notes.

Table A2. *Cont.*

Flag	Prompt (Significance)
`has_capital_rights`	The asset grants the holder certain rights over the capital structure of an entity, including claims to dividends, profits, or liquidation proceeds, similar to ownership stakes.
`investment_promise`	The asset is marketed with the promise of financial returns, suggesting an investment opportunity. This implies potential regulatory scrutiny for compliance with securities laws.
`dividend_like`	The asset offers returns or benefits similar to dividends, which are typically distributed by equity holders of a company, signaling investment-like characteristics.
`security_language`	The marketing or contractual terms of the asset use terminology commonly associated with securities, such as “shares,” “equity,” or “interest,” which may trigger regulatory requirements for registration and oversight.
`rights_transferable`	The asset can be freely transferred, often implying liquidity and tradability, similar to financial instruments that are exchanged in secondary markets.
`redeemable_in_fiat`	The asset can be converted or redeemed for a fiat currency, providing a clear value exchange mechanism, which is typical for stablecoins or other forms of currency-backed assets.
`daily_redeemability`	The asset can be redeemed or exchanged daily, providing liquidity and flexibility to investors, which is a critical characteristic for money market instruments.
`reserve_assets_held`	The issuer holds a reserve of assets backing the issued tokens or units, providing security to investors by ensuring that tangible assets, similar to collateralization in financial markets, back the asset.
`reserve_audited`	The reserve assets held by the issuer are subject to independent audits, enhancing transparency and trust by confirming that the issuer maintains sufficient reserves to back the value of the asset.
`redemption_policy_clear`	The asset has a clearly defined process for redemption, ensuring that investors can easily exchange or liquidate their holdings, similar to the redemption terms for traditional securities.
`whitepaper_present`	The asset has a formal whitepaper or investment prospectus, which is critical for providing transparency and detailed information about the asset’s structure, risks, and potential returns.

Table A3. Specific requirements per MiCAR asset type.

Asset Types	Requirements
EMT (E-Money Token)	whitepaper_present, risk_factors_disclosed, issuer_identified, disclaimers_present, kyc_aml_controls, marketing_consistent, redeemable_in_fiat, daily_redeemability, reserve_assets_held, reserves_audited, safeguarding_mechanism, redemption_policy_clear

Table A3. *Cont.*

Asset Types	Requirements
ART (Asset-Referenced Token)	whitepaper_present, risk_factors_disclosed, issuer_identified, disclaimers_present, kyc_aml_controls, marketing_consistent, asset_backing_disclosed, valuation_method_disclosed, reserve_policy_clear, redemption_mechanism_disclosed, governance_arrangements_disclosed
SECURITY	whitepaper_present, risk_factors_disclosed, issuer_identified, disclaimers_present, kyc_aml_controls, marketing_consistent, prospectus_present, registered_with_authority, investor_protection_mechanisms
OTHER	whitepaper_present, risk_factors_disclosed, issuer_identified, disclaimers_present, kyc_aml_controls, marketing_consistent

Table A4. Definitions of MiCAR compliance requirements.

Requirements	Definition
`whitepaper_present`	The whitepaper represents a foundational document providing a comprehensive disclosure of the asset's structure, purpose, and operational details. It must be officially registered with the competent regulatory authority, ensuring transparency and accountability in accordance with regulatory standards.
`risk_factors_disclosed`	The whitepaper must include an explicit disclosure of the potential risks associated with the asset, ensuring that investors are fully informed of the possible financial, operational, and regulatory risks inherent in the investment.
`issuer_identified`	The identity of the issuer must be clearly stated, with sufficient details to establish the legitimacy and accountability of the party responsible for the asset. This is essential for investor protection and regulatory compliance.
`disclaimers_present`	A legal disclaimer outlining the limitations of liability, investor responsibilities, and risk factors must be present in the whitepaper or related documentation. This ensures that investors are aware of the legal framework and risks associated with the asset.
`kyc_aml_controls`	The project must implement a robust Know Your Customer (KYC) and Anti-Money Laundering (AML) process to verify participants' identities and prevent illegal activities such as money laundering and fraud. This is a critical compliance measure to adhere to international financial regulations.

Table A4. *Cont.*

Requirements	Definition
`marketing_consistent`	All marketing and promotional activities must be consistent with the information disclosed in the whitepaper, ensuring that the asset is marketed truthfully and transparently to potential investors. Misleading claims or discrepancies in marketing communications can lead to regulatory sanctions.
`redeemable_in_fiat`	The asset must be convertible into fiat currency, ensuring liquidity and marketability. This requirement establishes the asset's potential for real-world value realization and its compliance with financial market regulations.
`daily_redeemability`	The asset should allow daily redemption, providing liquidity to users and facilitating real-time market transactions. This enhances the asset's usability and ensures compliance with liquidity requirements.
`reserve_assets_held`	The project must hold reserve assets that substantiate the value of the issued tokens. These reserves act as a financial safeguard, ensuring the asset is backed by tangible financial resources and mitigating the risk of a "Rug Pull" or insolvency.
`reserves_audited`	The reserve assets held by the issuer are subject to independent audits, enhancing transparency and trust by confirming that the issuer maintains sufficient reserves to back the value of the asset.
`safeguarding_mechanism`	The project must have mechanisms in place to safeguard participants' assets, including protections against fraud, hacking, and the misappropriation of funds. This is a key element in protecting investor interests and ensuring the project's financial stability.
`redemption_policy_clear`	The asset's redemption policy must be clearly articulated, outlining the specific procedures and conditions under which the asset can be exchanged for fiat or other assets. This ensures that investors have a clear understanding of the redemption process, thus minimizing potential disputes.
`asset_backing_disclosed`	The underlying assets or collateral backing the issued token must be disclosed, providing investors with transparency into the asset's value and financial stability. This disclosure is critical to understanding the asset's intrinsic value.
`valuation_method_disclosed`	The method of valuing the asset must be disclosed, including any models, algorithms, or financial metrics used to determine its worth. This ensures that investors can assess the asset's value with confidence and in accordance with industry standards.
`reserve_policy_clear`	A clear and comprehensive reserve policy must be in place, detailing how reserves are managed, accessed, and used to support the asset's value. This ensures that reserves are properly allocated and utilized, preventing mismanagement.

Table A4. *Cont.*

Requirements	Definition
`redemption_mechanism_disclosed`	The mechanism by which investors can redeem their assets must be disclosed, ensuring that there are clear and efficient procedures for converting the asset into fiat or other tokens. This ensures that the redemption process aligns with investor expectations and legal requirements.
`governance_arrangements_disclosed`	The governance structure of the project must be disclosed, providing details about how decisions are made, how power is distributed, and the roles of key stakeholders. This transparency helps establish accountability and ensures that the project operates in the best interest of its investors.
`prospectus_present`	A formal prospectus must be provided, containing detailed financial information about the asset, its risks, and its market potential. This document serves as an essential disclosure for investors, providing them with all the necessary information to make informed decisions.
`registered_with_authority`	The asset must be officially registered with the relevant regulatory authority, ensuring that it meets the required legal and financial standards. This provides investors with assurance that the asset is legitimate and subject to regulatory oversight.
`investor_protection_mechanisms`	The project must implement mechanisms designed to protect investors, including safeguards against fraud, risk mitigation measures, and avenues for dispute resolution. These mechanisms are essential for fostering investor confidence and maintaining market integrity.

Appendix B. Heuristic Agent Prompts and Schemas

This appendix reports the full system prompt used by the `Heuristic Agent` and the corresponding Pydantic schema used to validate its structured outputs. The complete prompt text is provided below without omissions.

Listing A1. System prompt for the `Heuristic Agent` (full text).

```
Developer: SYSTEM PROMPT - Heuristic Analyzer

CONTEXT: You are an LLM serving as a senior analyst at a national supervisory authority
    (securities/financial regulator) within the platform. Your mission is to conduct
    regulator-grade, evidence-based due diligence and early-warning analysis of crypto-
    asset projects and tokens in accordance with regulations and general consumer
    protection principles.
- Always operate with a neutral, conservative, and compliance-first stance.
- Do not provide investment advice.
- Do not speculate: when evidence is missing or ambiguous, do not speculate, and specify
     precisely what information is missing from the available data, and avoid making
    assumptions.
- Ground every claim in traceable evidence.

YOUR TASK: Analyze the provided website content and assign a scam alert level from 0.0 (
    legitimate) to 1.0 (fraudulent).

WHAT TO LOOK FOR:
- Warning indicators:
  - Unsubstantiated or "guaranteed" returns: Claims of guaranteed returns or unusually
      high profits that are not backed by transparent, verifiable evidence.
  - Time-pressured promotions/referral bonuses: Urgent, high-pressure tactics
      encouraging rapid decision-making.
```

```
  - Anonymous or unverifiable principals/legal entity: The lack of publicly available,
      verifiable information about the individuals or organizations.
  - Missing or inconsistent disclosures (whitepaper, T\&Cs, risk factors, policies): The
       absence of clear, comprehensive documentation.
  - Contradictory or implausible tokenomics (supply, allocation, vesting): Claims that
      are internally inconsistent or not supported by documentation.
  - Unclear liquidity provisions (lockups, vesting schedules, treasury controls): Lack
      of transparent constraints on insider liquidity.
  - Unverifiable partnerships, audits, or endorsements: References to third parties that
       cannot be corroborated.
  - Copied, templated, or plagiarized documentation/website content: Reused materials
      that undermine credibility.
  - No public code repository or unverifiable smart-contract code: Absence of
      inspectable code or audit artifacts.
  - Excessive marketing/influencer focus over product substance: Promotional emphasis
      without technical or operational detail.
  - Unrealistic roadmap or technical claims without evidence: Promises lacking
      milestones or feasibility support.
  - Regulatory compliance claims without evidence or jurisdiction mismatch: Unsupported
      statements of licensing/registration.
  - Opaque governance or unilateral control: Centralized control structures not
      disclosed or justified.
  - Hidden fees, punitive terms, or withdrawal restrictions: Undisclosed costs or
      constraints affecting users.

RISK SCALE:
- 0.0-0.2: Very Low Risk (Clear legitimacy, no significant warning indicators)
- 0.2-0.4: Low Risk (Minor warning indicators with limited impact on legitimacy)
- 0.4-0.6: Moderate Risk (Several warning indicators present, warranting caution)
- 0.6-0.8: High Risk (Multiple significant warning indicators indicating potential
    issues)
- 0.8-1.0: Very High Risk (Numerous critical warning indicators suggesting likely fraud)

INSTRUCTIONS:
1. Read the content carefully
2. Identify key warning indicators and legitimacy indicators
3. Assign an overall scam alert level score (0.0-1.0)
4. Explain your reasoning clearly with specific evidence
5. For each warning indicator, provide a structured list where EACH warning indicator
    includes:
   - warning indicator: The specific warning indicator or issue identified
   - reason: What is missing or lacking in this case.
6. List positive indicators that suggest legitimacy
7. Set appropriate investor warning level
```

Listing A2. Pydantic models defining the structured output for the `Heuristic Agent`'s analysis.

```
from pydantic import BaseModel, Field
from typing import List

class KeyConcern(BaseModel):
    """Represents a specific concern identified in the analysis,
    with its rationale and associated risk."""
    concern: str = Field(
        description="A specific warning indicator or issue identified during the
            fraud risk assessment."
    )
    reason: str = Field(
        description="An explanation of why this concern is significant and the type
            of risk it presents."
    )

class ComprehensiveFraudAnalysis(BaseModel):
    """A simplified model for summarizing the fraud risk analysis powered
    by the LLM, incorporating natural language reasoning."""

    # Core Fraud Assessment
    overall_scam_risk: float = Field(
        description="The overall fraud risk score, ranging from 0.0 (legitimate) to
            1.0 (highly fraudulent).",
        ge=0.0, le=1.0
    )
    risk_summary: str = Field(
        description="A concise summary outlining the primary risk factors and
            legitimacy indicators identified."
```

```
    )

    # Key Risk Factors
    key_concerns: List[KeyConcern] = Field(
        description="A list of key concerns identified, with detailed explanations (
            empty if legitimate).",
        default_factory=list
    )
```

Appendix C. Compliance Agent Prompts and Schemas

This appendix provides the full system and user prompts for the two-stage MiCAR classification and compliance assessments, along with the exact output schemas used in the experiments.

Appendix C.1. Classifier Phase I: Asset-Flag Extraction

Listing A3. System prompt for Asset-Flag extraction.

```
You are a MiCAR classification expert performing Phase I asset-flag extraction.

Task:
- Read the provided website content and fill AssetFlags.
- Use the AssetFlags field descriptions as the authoritative semantic definitions.
- Return booleans only through the AssetFlags tool output.

Decision policy (Phase I):
- Prefer recall over precision: set a flag to True when there is reasonable evidence.
- Accept both explicit and strongly implied evidence.
- If evidence is weak or contradictory, keep False.
- Do not infer facts that are not supported by the provided text.

Output requirements:
- Return exactly one AssetFlags object.
- Do not add extra fields, prose, or explanations outside structured output.
```

Listing A4. Human prompt for Asset-Flag extraction.

```
Perform classification flag extraction for this crypto asset:

**Project Title:** {title}

**Website Content:**
{content}

Extract all relevant AssetFlags for MiCAR classification.
```

Listing A5. Pydantic schema for AssetFlags used in Phase I.

```
class AssetFlags(BaseModel):
    """Classification indicators extracted from website content for initial MiCAR
        asset type determination.

    These flags are used by classify_by_flags() to determine the preliminary asset
        classification
    (EMT, ART, SECURITY, OTHER, NON_MICAR, NON_CLASSIFIABLE) before detailed
        compliance evaluation.
    """

    # Security classification flags - primary indicators
    regulated_as_security: bool = Field(
        False,
        description="The asset is classified under regulatory frameworks governing
            securities, meaning it must comply with legal requirements pertaining to
            investor protection, disclosure, and reporting obligations."
    )
    represents_equity: bool = Field(
        False,
        description="The asset confers ownership rights or a share in the profits of
            an underlying entity, resembling traditional equity instruments such as
            shares or stock."
```

```
    )
    represents_debt: bool = Field(
        False,
        description="The␣asset␣represents␣a␣financial␣obligation␣of␣an␣issuer␣to␣
            repay␣a␣debt,␣akin␣to␣traditional␣debt␣securities␣such␣as␣bonds␣or␣
            promissory␣notes."
    )
    has_capital_rights: bool = Field(
        False,
        description="The␣asset␣grants␣the␣holder␣certain␣rights␣over␣the␣capital␣
            structure␣of␣an␣entity,␣including␣claims␣to␣dividends,␣profits,␣or␣
            liquidation␣proceeds,␣similar␣to␣ownership␣stakes."
    )
    investment_promise: bool = Field(
        False,
        description="The␣asset␣is␣marketed␣with␣the␣promise␣of␣financial␣returns,␣
            suggesting␣an␣investment␣opportunity.␣This␣implies␣potential␣regulatory␣
            scrutiny␣for␣compliance␣with␣securities␣laws."
    )
    dividend_like: bool = Field(
        False,
        description="The␣asset␣offers␣returns␣or␣benefits␣similar␣to␣dividends,␣which
            ␣are␣typically␣distributed␣by␣equity␣holders␣of␣a␣company,␣signaling␣
            investment-like␣characteristics."
    )
    security_language: bool = Field(
        False,
        description="The␣marketing␣or␣contractual␣terms␣of␣the␣asset␣use␣terminology␣
            commonly␣associated␣with␣securities,␣such␣as␣'shares,'␣'equity,'␣or␣'
            interest,'␣which␣may␣trigger␣regulatory␣requirements␣for␣registration␣
            and␣oversight."
    )
    rights_transferable: bool = Field(
        False,
        description="The␣asset␣can␣be␣freely␣transferred,␣often␣implying␣liquidity␣
            and␣tradability,␣similar␣to␣financial␣instruments␣that␣are␣exchanged␣in␣
            secondary␣markets."
    )

    # EMT (E-Money Token) classification flags
    redeemable_in_fiat: bool = Field(
        False,
        description="The␣asset␣can␣be␣converted␣or␣redeemed␣for␣a␣fiat␣currency,␣
            providing␣a␣clear␣value␣exchange␣mechanism,␣which␣is␣typical␣for␣
            stablecoins␣or␣other␣forms␣of␣currency-backed␣assets."
    )
    daily_redeemability: bool = Field(
        False,
        description="The␣asset␣can␣be␣redeemed␣or␣exchanged␣on␣a␣daily␣basis,␣
            providing␣liquidity␣and␣flexibility␣to␣investors,␣which␣is␣a␣critical␣
            characteristic␣for␣money␣market␣instruments."
    )
    reserve_assets_held: bool = Field(
        False,
        description="The␣issuer␣holds␣a␣reserve␣of␣assets␣backing␣the␣issued␣tokens␣
            or␣units,␣providing␣security␣to␣investors␣by␣ensuring␣that␣the␣asset␣is␣
            backed␣by␣tangible␣assets,␣similar␣to␣collateralization␣in␣financial␣
            markets."
    )
    audited_reserves: bool = Field(
        False,
        description="The␣reserves␣held␣by␣the␣issuer␣are␣subject␣to␣independent␣
            audits,␣enhancing␣transparency␣and␣trust␣by␣confirming␣that␣the␣issuer␣
            maintains␣sufficient␣reserves␣to␣back␣the␣value␣of␣the␣asset."
    )
    redemption_policy_clear: bool = Field(
        False,
        description="The␣asset␣has␣a␣clearly␣defined␣process␣for␣redemption,␣ensuring
            ␣that␣investors␣can␣easily␣exchange␣or␣liquidate␣their␣holdings,␣similar
            ␣to␣the␣redemption␣terms␣for␣traditional␣securities."
    )

    # ART (Asset-Referenced Token) classification flags
    backed_by_assets: bool = Field(
        False,
```

```
        description="The asset is anchored to a basket of assets (currencies, commodities, cryptocurrencies), providing value stabilization through diversified collateral, characteristic of asset-referenced tokens."
    )

    # OTHER (Utility/Governance token) classification flags
    utility_function: bool = Field(
        False,
        description="The asset provides access to services, platforms, or ecosystem features, representing a utility token that grants functional rights rather than investment returns."
    )
    governance_function: bool = Field(
        False,
        description="The asset confers governance or voting rights in a decentralized protocol or organization, allowing holders to participate in decision-making processes."
    )

    # NFT classification flag
    nft_unique: bool = Field(
        False,
        description="The asset is a unique or non-fungible token (NFT), representing a one-of-a-kind digital asset rather than a fungible currency or security."
    )

    # General documentation and compliance indicators
    whitepaper_present: bool = Field(
        False,
        description="The asset has a formal whitepaper or investment prospectus, which is critical for providing transparency and detailed information about the asset's structure, risks, and potential returns."
    )
    disclaimers_regulatory: bool = Field(
        False,
        description="The asset documentation includes regulatory warnings or references to financial regulations, indicating awareness of compliance requirements and investor protection obligations."
    )
```

Appendix C.2. Classifier Phase II: MiCAR Compliance Assessment

Listing A6. System prompt for MiCAR compliance assessment.

```
You are a MiCAR compliance auditor performing Phase II compliance-flag extraction.

Task:
- Read the provided website content and fill ComplianceFlags.
- Use ComplianceFlags field descriptions as the authoritative regulatory definitions.
- Return booleans only through the ComplianceFlags tool output.

Decision policy (Phase II):
- Prefer precision over recall: set a flag to True only with substantial evidence.
- Require substance, not keyword matching.
- Generic/legal boilerplate alone is not sufficient evidence of compliance.
- If evidence is partial, ambiguous, or missing, keep False.

Output requirements:
- Return exactly one ComplianceFlags object.
- Do not add extra fields, prose, or explanations outside structured output.
```

Listing A7. Human prompt for MiCAR compliance assessment.

```
Conduct MiCAR compliance assessment for this crypto asset:

**Project Title:** {title}

**Website Content:**
{content}

Evaluate all applicable ComplianceFlags according to MiCAR regulatory standards.
```

Listing A8. Pydantic schemas for ComplianceFlags and ComplianceResult used in Phase II.

```
class ComplianceFlags(BaseModel):
    """MiCAR compliance indicators with formal regulatory definitions"""

    # Common obligations (all crypto-assets under MiCAR)
    whitepaper_present: bool = Field(
        False,
        description="The whitepaper represents a foundational document providing a comprehensive disclosure of the asset's structure, purpose, and operational details. It must be officially registered with the competent regulatory authority, ensuring transparency and accountability in accordance with regulatory standards."
    )
    risk_factors_disclosed: bool = Field(
        False,
        description="The whitepaper must include an explicit disclosure of the potential risks associated with the asset, ensuring that investors are fully informed of the possible financial, operational, and regulatory risks inherent in the investment."
    )
    issuer_identified: bool = Field(
        False,
        description="The identity of the issuer must be clearly stated, with sufficient details to establish the legitimacy and accountability of the party responsible for the asset. This is essential for investor protection and regulatory compliance."
    )
    disclaimers_present: bool = Field(
        False,
        description="A legal disclaimer outlining the limitations of liability, investor responsibilities, and risk factors must be present in the whitepaper or related documentation. This ensures that investors are aware of the legal framework and risks associated with the asset."
    )
    kyc_aml_controls: bool = Field(
        False,
        description="The project must implement a robust Know Your Customer (KYC) and Anti-Money Laundering (AML) process to verify the identity of participants and prevent illegal activities such as money laundering and fraud. This is a critical compliance measure to adhere to international financial regulations."
    )
    marketing_consistent: bool = Field(
        False,
        description="All marketing and promotional activities must be consistent with the information disclosed in the whitepaper, ensuring that the asset is marketed truthfully and transparently to potential investors. Misleading claims or discrepancies in marketing communications can lead to regulatory sanctions."
    )

    # EMT-specific (Electronic Money Token)
    redeemable_in_fiat: bool = Field(
        False,
        description="The asset must be convertible into fiat currency, ensuring liquidity and marketability. This requirement establishes the asset's potential for real-world value realization and its compliance with financial market regulations."
    )
    daily_redeemability: bool = Field(
        False,
        description="The asset should offer the possibility of redemption on a daily basis, providing liquidity to users and facilitating real-time market transactions. This enhances the asset's usability and ensures compliance with liquidity requirements."
    )
    reserve_assets_held: bool = Field(
        False,
        description="The project must hold reserve assets that substantiate the value of the issued tokens. These reserves act as a financial safeguard, ensuring that the asset is backed by tangible financial resources, thereby mitigating the risk of a 'rug pull' or insolvency."
    )
    reserves_audited: bool = Field(
```

```
        False,
        description="The reserves held by the project must be subject to regular, independent audits to verify the validity of claims regarding asset backing. This ensures financial transparency and protects investors from fraudulent activities."
    )
    safeguarding_mechanism: bool = Field(
        False,
        description="The project must have mechanisms in place to safeguard the assets of participants, including protections against fraud, hacking, or misappropriation of funds. This is a key element in protecting investor interests and ensuring the project's financial stability."
    )
    redemption_policy_clear: bool = Field(
        False,
        description="The asset's redemption policy must be clearly articulated, outlining the specific procedures and conditions under which the asset can be exchanged for fiat or other assets. This ensures that investors have a clear understanding of the redemption process, thus minimizing potential disputes."
    )

    # ART-specific (Asset-Referenced Token)
    asset_backing_disclosed: bool = Field(
        False,
        description="The underlying assets or collateral backing the issued token must be disclosed, offering investors transparency regarding the asset's value and financial stability. This disclosure is critical to understanding the intrinsic value of the asset."
    )
    valuation_method_disclosed: bool = Field(
        False,
        description="The method of valuing the asset must be disclosed, including any models, algorithms, or financial metrics used to determine its worth. This ensures that investors can assess the asset's value with confidence and in accordance with industry standards."
    )
    reserve_policy_clear: bool = Field(
        False,
        description="A clear and comprehensive reserve policy must be in place, detailing how reserves are managed, accessed, and used to support the asset's value. This ensures that reserves are properly allocated and utilized, preventing mismanagement."
    )
    redemption_mechanism_disclosed: bool = Field(
        False,
        description="The mechanism by which investors can redeem their assets must be disclosed, ensuring that there are clear and efficient procedures for converting the asset into fiat or other tokens. This ensures that the redemption process aligns with investor expectations and legal requirements."
    )
    governance_arrangements_disclosed: bool = Field(
        False,
        description="The governance structure of the project must be disclosed, providing details about how decisions are made, how power is distributed, and the roles of key stakeholders. This transparency helps establish accountability and ensures that the project operates in the best interest of its investors."
    )

    # SECURITY-specific (Security Tokens)
    prospectus_present: bool = Field(
        False,
        description="A formal prospectus must be provided, containing detailed financial information about the asset, its risks, and its market potential. This document serves as an essential disclosure for investors, ensuring that they have all the necessary information to make informed decisions."
    )
    registered_with_authority: bool = Field(
        False,
        description="The asset must be officially registered with the relevant regulatory authority, ensuring that it meets the required legal and
```

```
            financial standards. This provides assurance to investors that the asset
            is legitimate and subject to regulatory oversight."
    )
    investor_protection_mechanisms: bool = Field(
        False,
        description="The project must implement mechanisms designed to protect
            investors, including safeguards against fraud, risk mitigation measures,
            and avenues for dispute resolution. These mechanisms are essential for
            fostering investor confidence and maintaining market integrity."
    )

    # Additional useful compliance elements
    custody_safeguards: bool = Field(False, description="Asset custody entrusted to
        qualified entities")
    complaints_procedure: bool = Field(False, description="Complaint procedure for
        investors available")
    conflict_of_interest_policy: bool = Field(False, description="Conflict of
        interest policy published")


class ComplianceResult(BaseModel):
    """Detailed MiCAR compliance check result"""
    asset_type: MiCARClass
    checks: Dict[str, bool]
    score: float
    compliant: bool
    threshold: float
    level: str
    satisfied: List[str]
    missing: List[str]
    reasoning: str
```

Appendix D. User Feedback Questionnaires

The following tables report the pre-usage and post-usage questionnaires administered to users of the scam token detection platform. The instruments were designed to capture both participants' initial expectations and familiarity with crypto-assets, as well as their evaluation of the platform after hands-on use.

Participants were invited to complete both questionnaires. A user guide was provided, outlining the key operations and tasks they were expected to perform during the evaluation. The responses collected through these questionnaires informed the analysis of usability, perceived effectiveness, and overall user satisfaction.

Table A5. Pre-usage questionnaire for platform users.

Question	Possible Answers
1. Which best describes your role? (Select one)	A. Regulator/Compliance professional; B. Investor; C. Researcher/Academic; D. Developer/Engineer; E. Student/Learner; F. Other
2. How familiar are you with cryptocurrencies? (Select one)	A. Not familiar; B. Beginner; C. Intermediate; D. Advanced
3. What do you aim to achieve by using the platform? (Multiple selection)	A. Identify risky or fraudulent crypto-asset projects; B. Verify regulatory compliance of projects; C. Understand smart contracts and transactions; D. Learn about crypto-assets more safely; E. Other
4. What aspect is most important to you when assessing a crypto-asset project? (Select one)	A. Clear and reliable information; B. Easy-to-understand results; C. Quick overview; D. Detailed explanations; E. Not sure yet
5. How confident are you in your understanding of cryptocurrency risks and regulations? (Select one)	A. Very confident; B. Somewhat confident; C. Not confident
6. What do you hope to achieve by using the platform?	Open-ended
7. How do you choose which crypto-asset projects to invest in?	Open-ended

Table A6. Post-usage questionnaire for platform users.

Question	Possible Answers
1. Which best describes your role? (Select one)	A. Regulator/Compliance professional; B. Investor; C. Researcher/Academic; D. Developer/Engineer; E. Student/Learner; F. Other
2. How familiar are you with cryptocurrencies? (Select one)	A. Not familiar; B. Beginner; C. Intermediate; D. Advanced
3. How easy was it to start using the platform? (Select one)	0. Very difficult; 1; 2; 3; 4; 5. Very easy
4. How clear and intuitive did you find the user interface? (Select one)	0. Not at all intuitive; 1; 2; 3; 4; 5. Very intuitive
5. How accurately did the system detect fraudulent or risky projects?	0. Not accurate at all; 1; 2; 3; 4; 5. Very accurate
6. How useful was the risk score in evaluating a project's reliability or compliance?	0. Not useful; 1; 2; 3; 4; 5. Very useful
7. How relevant and actionable were the insights provided by the system?	0. Not relevant/actionable; 1; 2; 3; 4; 5. Very relevant/actionable
8. To what extent did the analysis influence your decision-making?	0. No influence; 1; 2; 3; 4; 5; 6; 7; 8; 9; 10. Strong influence
9. How reliable was the system during your analysis?	0. Not reliable at all; 1; 2; 3; 4; 5. Very reliable
10. How would you rate the system's performance speed?	0. Very slow/frequent delays; 1; 2; 3; 4; 5. Very fast/no delays
11. How satisfied are you with the software overall? (Select one)	0. Very unsatisfied; 1; 2; 3; 4; 5. Very satisfied
12. How likely are you to recommend the platform to a colleague?	0. Not likely; 1; 2; 3; 4; 5. Extremely likely
13. What aspects of the platform did you find most valuable or effective?	Open-ended
14. What improvements or additional features would you recommend for future versions?	Open-ended

References

1. Kasula, V.K.; Alshboul, A. Leveraging Advanced Technologies to Enhance Public Awareness and Mitigate Risks of Cryptocurrency Scams: A Qualitative Analysis. In *Demystifying AI and ML for Cyber–Threat Intelligence*; Springer: Cham, Switzerland, 2025; pp. 359–369.
2. Meiklejohn, S.; Pomarole, M.; Jordan, G.; Levchenko, K.; McCoy, D.; Voelker, G.M.; Savage, S. A fistful of bitcoins: characterizing payments among men with no names. In Proceedings of the 2013 Conference on Internet Measurement Conference, Barcelona, Spain, 23–25 October 2013; pp. 127–140.
3. Bai, Z.; Wang, P. From medium to risk factor: How crypto investment elevates fraud victimization. *Financ. Res. Lett.* **2025**, *86*, 108592.
4. Federal Bureau of Investigation. 2024 Internet Crime Report. 2024. Available online: https://www.ic3.gov/AnnualReport/Reports/2024_IC3Report.pdf (accessed on 20 May 2025).
5. Arner, D.W.; Barberis, J.; Buckey, R.P. FinTech, RegTech, and the reconceptualization of financial regulation. *Nw. J. Int'l L. Bus.* **2016**, *37*, 371.
6. Gomber, P.; Kauffman, R.J.; Parker, C.; Weber, B.W. On the fintech revolution: Interpreting the forces of innovation, disruption, and transformation in financial services. *J. Manag. Inf. Syst.* **2018**, *35*, 220–265.
7. Balaji, P.G.; Srinivasan, D. An introduction to multi-agent systems. In *Innovations in Multi-Agent Systems and Applications-1*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 1–27.
8. Trerotola, M.; Calvaresi, D. AI-Driven Multi-Agent Systems for Automated Regulatory Analysis of Crypto Projects. In *Proceedings of the 2025 IEEE International Conference on Metrology for eXtended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE)*; IEEE: Piscataway, NJ, USA, 2025; pp. 735–740.
9. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
10. Katz, D.M.; Bommarito, M.J.; Gao, S.; Arredondo, P. Gpt-4 passes the bar exam. *Philos. Trans. R. Soc. A* **2024**, *382*, 20230254.
11. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 24824–24837.

12. White, J.; Fu, Q.; Hays, S.; Sandborn, M.; Olea, C.; Gilbert, H.; Elnashar, A.; Spencer-Smith, J.; Schmidt, D.C. A prompt pattern catalog to enhance prompt engineering with chatgpt. *arXiv* **2023**, arXiv:2302.11382.
13. Van der Hoek, W.; Wooldridge, M. Multi-agent systems. *Found. Artif. Intell.* **2008**, *3*, 887–928.
14. Toma, A.M.; Cerchiello, P. Initial coin offerings: Risk or opportunity? *Front. Artif. Intell.* **2020**, *3*, 18.
15. Karimov, B.; Wójcik, P. Identification of scams in initial coin offerings with machine learning. *Front. Artif. Intell.* **2021**, *4*, 718450.
16. Mazorra, B.; Adan, V.; Daza, V. Do not rug on me: Leveraging machine learning techniques for automated scam detection. *Mathematics* **2022**, *10*, 949.
17. Pocher, N.; Zichichi, M.; Merizzi, F.; Shafiq, M.Z.; Ferretti, S. Detecting anomalous cryptocurrency transactions: An AML/CFT application of machine learning-based forensics. *Electron. Mark.* **2023**, *33*, 37.
18. Liang, R.; Chen, J.; Wu, C.; He, K.; Wu, Y.; Sun, W.; Du, R.; Zhao, Q.; Liu, Y. Towards effective detection of ponzi schemes on ethereum with contract runtime behavior graph. *Acm Trans. Softw. Eng. Methodol.* **2025**, *34*, 1–32.
19. Luo, B.; Zhang, Z.; Wang, Q.; Ke, A.; Lu, S.; He, B. AI-powered fraud detection in decentralized finance: A project life cycle perspective. *Acm Comput. Surv.* **2024**, *57*, 1–38.
20. Chalkidis, I.; Fergadiotis, M.; Malakasiotis, P.; Aletras, N.; Androutsopoulos, I. LEGAL-BERT: The Muppets Straight Out of Law School. *arXiv* **2020**, arXiv:2010.02559. https://doi.org/10.48550/arXiv.2010.02559.
21. Guha, N.; Nyarko, J.; Ho, D.E.; Re, C.; Chilton, A.; Narayana, A.; Chohlas-Wood, A.; Peters, A.; Waldon, B.; Rockmore, D.N.; et al. LegalBench: A Collaboratively Built Benchmark for Measuring Legal Reasoning in Large Language Models. *arXiv* **2023**, arXiv:2308.11462. https://doi.org/10.48550/arXiv.2308.11462.
22. Trerotola, M.; Calvaresi, D. Enhancing Blockchain Transaction Tracking: A Systematic Review of DLT-Based Financial Systems. In *Proceedings of the 2025 IEEE International Conference on Metrology for eXtended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE)*; IEEE: Piscataway, NJ, USA, 2025; pp. 747–752.
23. Monamo, P.; Marivate, V.; Twala, B. Unsupervised learning for robust Bitcoin fraud detection. In *Proceedings of the 2016 Information Security for South Africa (ISSA)*; IEEE: Piscataway, NJ, USA, 2016; pp. 129–134.
24. Chang, T.H.; Svetinovic, D. Improving bitcoin ownership identification using transaction patterns analysis. *IEEE Trans. Syst. Man, Cybern. Syst.* **2018**, *50*, 9–20.
25. Chen, B.; Wei, F.; Gu, C. Bitcoin theft detection based on supervised machine learning algorithms. *Secur. Commun. Netw.* **2021**, *2021*, 6643763.
26. Iscan, C.; Kumas, O.; Akbulut, F.P.; Akbulut, A. Wallet-based transaction fraud prevention through LightGBM with the focus on minimizing false alarms. *IEEE Access* **2023**, *11*, 131465–131474.
27. Weber, M.; Domeniconi, G.; Chen, J.; Weidele, D.K.I.; Bellei, C.; Robinson, T.; Leiserson, C.E. Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. *arXiv* **2019**, arXiv:1908.02591.
28. Nerurkar, P.; Bhirud, S.; Patel, D.; Ludinard, R.; Busnel, Y.; Kumari, S. Supervised learning model for identifying illegal activities in Bitcoin. *Appl. Intell.* **2021**, *51*, 3824–3843.
29. Northcutt, C.; Jiang, L.; Chuang, I. Confident learning: Estimating uncertainty in dataset labels. *J. Artif. Intell. Res.* **2021**, *70*, 1373–1411.
30. Wu, C.; Chen, J.; Zhao, Z.; He, K.; Xu, G.; Wu, Y.; Wang, H.; Li, H.; Liu, Y.; Xiang, Y. Tokenscout: Early detection of ethereum scam tokens via temporal graph learning. In Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security, Salt Lake City, UT, USA, 14–18 October 2024; pp. 956–970.
31. Chalkidis, I.; Jana, A.; Hartung, D.; Bommarito, M.; Androutsopoulos, I.; Katz, D.; Aletras, N. LexGLUE: A Benchmark Dataset for Legal Language Understanding in English. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2022; pp. 4310–4330. https://doi.org/10.18653/v1/2022.acl-long.297.
32. Cui, J.; Ning, M.; Li, Z.; Chen, B.; Yan, Y.; Li, H.; Ling, B.; Tian, Y.; Yuan, L. Chatlaw: A multi-agent collaborative legal assistant with knowledge graph enhanced mixture-of-experts large language model. *arXiv* **2023**, arXiv:2306.16092.
33. Zetzsche, D.A.; Annunziata, F.; Arner, D.W.; Buckley, R.P. The Markets in Crypto-Assets regulation (MiCA) and the EU digital finance strategy. *Cap. Mark. Law J.* **2021**, *16*, 203–225.
34. Arrieta, A.B.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; García, S.; Gil-López, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **2020**, *58*, 82–115.
35. Adadi, A.; Berrada, M. Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). *IEEE Access* **2018**, *6*, 52138–52160.
36. Devarajulu, V.S.; Kanipakam, S.; Addula, S.R.; et al. Embedding Accountability in the AI Lifecycle for Critical Finance Applications. In Proceedings of the 2025 Cyber Awareness and Research Symposium (CARS). IEEE: Piscataway, NJ, USA, 2025; pp. 1–8.

37. Ramírez, S. FastAPI: Modern Web Framework for Building APIs with Python 3.7+. 2018. Available online: https://fastapi.tiangolo.com/ (accessed on 20 May 2025).
38. Vercel. Next.js: The React Framework for Production. 2016. Available online: https://nextjs.org/ (accessed on 20 May 2025).
39. Microsoft. Playwright: Fast and Reliable End-to-End Testing for Modern Web Apps. 2020. Available online: https://playwright.dev/ (accessed on 20 May 2025).
40. Trerotola, M. MAS-For-Token-Scam-Detection (v1.0). 2025. Available upon reasonable request from the corresponding author (mario.trerotola@polito.it).
41. Palanca, J.; Terrasa, A.; Julian, V.; Carrascosa, C. Spade 3: Supporting the new generation of multi-agent systems. *IEEE Access* **2020**, *8*, 182537–182549.
42. Shen, Y.; Heacock, L.; Elias, J.; Hentel, K.D.; Reig, B.; Shih, G.; Moy, L. ChatGPT and other large language models are double-edged swords. *Radiology* **2023**, *307*, e230163.
43. Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; Neubig, G. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *Acm Comput. Surv.* **2023**, *55*, 1–35.
44. Sheth, A.; Roy, K.; Gaur, M. Neurosymbolic artificial intelligence (why, what, and how). *IEEE Intell. Syst.* **2023**, *38*, 56–62.