România
Ministerul Apărării Naționale
Academia Tehnică Militară „Ferdinand I”

Facultatea de Sisteme Informatice și Securitate Cibernetică
Calculatoare și sisteme informatice pentru apărare și securitate națională

Unelte software bazate pe mecanisme de inteligență artificială aplicată în criptografie

Specificația Cerințelor Software pentru platforma software de unelte AI în criptografie

Disciplina: Ingineria Programării

Profesor Coordonator
Prof. Dr. Ing. Mihai Togan

Student
Sd.Sg.Maj. Moldovan Andrei

București
4 Ianuarie 2026

Control Document

Titlu	Specificația Cerințelor Software pentru platforma software de unelte AI în criptografie
Data	4 Ianuarie 2026
Status	Draft
Versiune	1.0
Pregătit pentru	Academia Tehnică Militară „Ferdinand I”
Referință	SRS_AI_CRYPTO_V1.0_Jan_2026

Disclaimer

Acest document este pregătit în scopuri academice pentru proiectul de curs la disciplina Ingineria Programării. Documentul conține specificațiile detaliate ale cerințelor software pentru Unelte software bazate pe mecanisme de inteligență artificială aplicată în criptografie.

Cuprins

Glosar

Termen	Definiție
API	Application Programming Interface
RAG	Retrieval-Augmented Generation
RBAC	Role-Based Access Control
SRS	Software Requirements Specification
JWT	JSON Web Token
MFA	Multi-Factor Authentication
TLS	Transport Layer Security
mTLS	Mutual TLS
RL	Reinforcement Learning
LLM	Large Language Model
ML	Machine Learning
PQC	Post-Quantum Cryptography
HIBP	Have I Been Pwned
YAFU	Yet Another Factorization Utility
CSP	Content Security Policy
CSRF	Cross-Site Request Forgery
XSS	Cross-Site Scripting
TTL	Time to Live
K8s	Kubernetes
Docker	Containerization Platform
GDPR	General Data Protection Regulation
SIEM	Security Information and Event Management

1 Introducere

1.1 Scopul Documentului

Acest document SRS oferă o descriere completă a platformei software de unelte AI în criptografie, un sistem de inteligență criptografică de următoarea generație, bazat pe o arhitectură multi-agent autonomă. Platforma orchestrează agenți AI specializați pentru analiza securității parolelor, factorizare numere prime, asistență teoretică în criptografie, operațiuni criptografice și detecție de criptosisteme cu scor de încredere.

Platforma facilitează:

- Evaluarea securității parolelor folosind ansambluri ML
- Verificarea primalității și factorizarea numerelor mari
- Asistență teoretică în criptografie prin RAG
- Execuția de operațiuni criptografice (simetrice, asimetrice, PQC)
- Detectarea automată a criptosistemelor din ciphertext

Acest SRS va servi drept bază pentru fazele ulterioare de proiectare, dezvoltare și testare ale sistemului.

1.2 Convenții Document

Acest document urmează convențiile RFC 2119 cu următoarele traduceri în limba română:

Termen	Descriere
TREBUIE	Indică o cerință obligatorie care trebuie implementată.
AR TREBUI	Indică o cerință recomandată.
POATE	Indică o cerință opțională.

Cerințele sunt categorisite astfel:

Număr Cerință	Descriere
FR-XXX-NNN	Cerințe Funcționale
NFR-XXX-NNN	Cerințe Non-Funcționale
SR-XXX-NNN	Cerințe de Securitate
ML-XXX-NNN	Cerințe AI/ML
INF-XXX-NNN	Cerințe de Infrastructură
CMP-XXX-NNN	Cerințe de Conformitate

1.3 Cadrul Proiectului

1.3.1 În Cadrul Proiectului

Platforma software de unelte AI în criptografie include următoarele componente principale:

1. Modul de Identity și Access Management (IAM)
2. Orchestrator Multi-Agent
3. Agenți Specializați (6 agenți)
4. Sistem Notificări și Conversații
5. Interfață Web React și CLI

1.3.2 În Afara Domeniului Proiectului

Următoarele aspecte sunt excluse explicit din acest proiect:

1. Sistem de plăți sau monetizare
2. Aplicație mobilă nativă (iOS/Android)
3. Integrare cu sisteme externe de învățământ
4. Sistem de management al utilizatorilor la scară enterprise
5. Suport pentru mai multe limbi (doar engleză)

2 Descriere Generală

2.1 Descrierea produsului

Platforma este o soluție integrată de unelte AI aplicate în criptografie, complet funcțională, care unifică agenți specializați pentru criptanaliză, audit de parole și suport educațional. Sistemul include un orchestrator central pentru rutare, scheduling și execuție asincronă a job-urilor; un agent de decizie pentru extragerea intenției și a entităților; un detector de criptosisteme care are la baza RL cu scor de încredere; un modul Hash Breaker cu integrare HashCat/John și generare controlată de parole; un agregator de scoruri de robustețe (ML, zxcvbn, verificări de tip breach); un serviciu de primalitate/factorizare (YAFU + FactorDB); un modul RAG local pentru asistență teoretică; și un CTF Tool dedicat scenariilor educaționale și testare reproductibilă. Arhitectura este bazată pe microservicii containerizate, orchestrate în Kubernetes, expune API/CLI/Web, include autentificare și autorizare RBAC, audit și monitorizare continuă, iar pipeline-ul CI/CD și publicarea open-source pe GitHub susțin livrarea production-ready atât local, cât și în cloud.

2.2 Perspectiva Produsului

Platforma software de unelte AI în criptografie este dezvoltată ca proiect de licență la Academia Tehnică Militară „Ferdinand I". Sistemul operează ca o aplicație web centralizată, accesibilă prin browsere standard și CLI, fără a necesita instalarea de software client.

2.3 Funcționalitățile Produsului

Platforma software de unelte AI în criptografie oferă următoarele funcționalități majore:

1. **Autentificare și Managementul Accesului**
   - Autentificare securizată cu email și parolă
   - Sistem RBAC cu 3 roluri (Anonymous, User, Admin)
   - Suport MFA pentru admini
   - Token-uri JWT cu rotație automată
2. **Orchestrare Multi-Agent**
   - Detectare automată a intenției utilizatorului
   - Routing inteligent către agenții potriviți
   - Execuție paralelă pentru operații independente
   - Agregare răspunsuri multiple
3. **Verificare Parole (Password Checker)**
   - Evaluare ML cu PassGPT, zxcvbn, PassStrengthAI
   - Verificare HIBP (k-anonymity)
   - Scor unificat 0-100
   - Recomandări acționabile
4. **Verificare Primalitate (Prime Checker)**
   - Test Miller-Rabin deterministic

- Factorizare cu YAFU și FactorDB
- Cache LRU în memorie și persistent

5. **Specialist Teorie (Theory Specialist)**
   - RAG pentru criptografie cu ChromaDB
   - Ingestie PDF, Markdown, Text
   - Reranking cu cross-encoder
   - Istoric conversații cu context
6. **Executor Comenzi (Command Executor)**
   - Operații crypto: AES, RSA, HMAC, PQC
   - Hashing: SHA-256/384/512, SHA3, BLAKE2
   - Encoding: Base64, Hex
   - Implementare în Rust pentru siguranță
7. **Selector Alegeri (Choice Maker)**
   - NLP pentru clasificare intenții
   - Extracție entități cu SecureBERT 2.0
   - 10+ clase de intenție
8. **Detectare Criptosisteme**
   - Integrare CyberChef Magic
   - Euristici dcode-like
   - Scor de încredere 0-1

2.4 Caracteristici Utilizatori

Platforma deservește următoarele categorii de utilizatori:

Rol	Caracteristici și Nevoi
Anonymous	• Utilizatori neautentificați • Acces limitat la operații demo • Nevoi: Testare funcționalități de bază
User	• Utilizatori înregistrați • Competențe IT: medii-avansate • Nevoi: Acces complet la toate agenții, istoric conversații
Admin	• Administratori sistem • Competențe IT: avansate • Nevoi: Management utilizatori, configurări globale, audit, monitorizare

2.5 Mediul de Operare

Platforma va opera în următorul mediu:

1. **Mediu Tehnic**
    - Aplicație web accesibilă prin browsere moderne
    - CLI pentru power users
    - Design responsive
    - Hosting: on-premise sau cloud (AWS/GCP/Azure)
2. **Mediu Hardware**
    - Server(e) cu capacitate de procesare adecvată pentru 100+ utilizatori
    - Stocare: minimum 100GB pentru baze de date și modele ML
    - Backup automat
3. **Mediu Software**
    - Containerizare: Docker + Docker Compose
    - Orchestrare: Kubernetes (producție)
    - Limbaje: Go, Rust, Python, TypeScript
    - Baze de date: PostgreSQL 16, Redis 7, ChromaDB, BoltDB
    - Frontend: React
    - Observabilitate: Prometheus + Grafana

3 Cerințe Funcționale

Această secțiune detaliază cerințele funcționale ale platformei software de unelte AI în criptografie, organizate pe module majore.

3.1 Identity și Access Management (IAM)

3.1.1 Autentificare și Autorizare

ID	Cerință
FR-IAM-001	Sistemul **TREBUIE** să permită înregistrarea utilizatorilor cu email, parolă și confirmare email.
FR-IAM-002	Sistemul **TREBUIE** să implementeze autentificare prin email/parolă cu rate limiting (max 5 încercări/minut).
FR-IAM-003	Sistemul **TREBUIE** să suporte MFA (TOTP RFC 6238) pentru rolul Admin.
FR-IAM-004	Sistemul **AR TREBUI** să suporte WebAuthn/FIDO2 pentru passwordless authentication.
FR-IAM-005	Sistemul **TREBUIE** să emită token-uri JWT (access: 15min, refresh: 7 zile) cu rotație automată.
FR-IAM-006	Sistemul **TREBUIE** să permită generarea și revocarea de API keys cu scope-uri configurabile.
FR-IAM-007	Sistemul **TREBUIE** să implementeze RBAC cu 3 roluri predefinite: Anonymous, User, Admin.
FR-IAM-008	Sistemul **TREBUIE** să permită resetarea parolei prin email cu token unic (TTL: 1 oră).
FR-IAM-009	Sistemul **AR TREBUI** să suporte OAuth2/OIDC pentru autentificare externă (GitHub, Google).
FR-IAM-010	Sistemul **TREBUIE** să invalideze toate sesiunile active la schimbarea parolei.

3.2 Orchestrare Multi-Agent

3.2.1 Orchestrator

ID	Cerință
FR-ORC-001	Orchestratorul **TREBUIE** să detecteze intenția utilizatorului folosind agentul Choice Maker.
FR-ORC-002	Orchestratorul **TREBUIE** să ruteze cererile către agentul/agenții potriviți pe baza intenției detectate.
FR-ORC-003	Orchestratorul **TREBUIE** să suporte execuție paralelă pentru operații independente.
FR-ORC-004	Orchestratorul **TREBUIE** să agregeze răspunsurile de la mai mulți agenți într-un răspuns unificat.
FR-ORC-005	Orchestratorul **TREBUIE** să implementeze timeout configurabil per agent (default: 30s).

ID	Cerință
FR-ORC-006	Orchestratorul **TREBUIE** să implementeze circuit breaker pentru agenți cu probleme.
FR-ORC-007	Orchestratorul **TREBUIE** să expună health endpoints pentru fiecare serviciu gestionat.
FR-ORC-008	Orchestratorul **AR TREBUI** să ofere fallback logic când agenții sunt indisponibili.
FR-ORC-009	Orchestratorul **TREBUIE** să suporte selectarea dinamică a provider-ului LLM per cerere.
FR-ORC-010	Orchestratorul **AR TREBUI** să permită configurarea priorităților de rutare per agent.

3.3 Agent Verificare Parole

3.3.1 Password Intelligence

ID	Cerință
FR-PWD-001	Agentul **TREBUIE** să calculeze scorul de securitate unificat (0-100) din ansamblu ML.
FR-PWD-002	Agentul **TREBUIE** să integreze PassGPT pentru analiza probabilistică a parolelor.
FR-PWD-003	Agentul **TREBUIE** să integreze zxcvbn pentru evaluarea heuristică.
FR-PWD-004	Agentul **TREBUIE** să verifice parola contra bazei HIBP (k-anonymity).
FR-PWD-005	Agentul **AR TREBUI** să integreze PassStrengthAI (CNN) pentru evaluare suplimentară.
FR-PWD-006	Agentul **TREBUIE** să returneze recomandări acționabile pentru îmbunătățirea parolei.
FR-PWD-007	Agentul **TREBUIE** să dezactiveze automat PassGPT pentru parole ¿ 10 caractere.
FR-PWD-008	Agentul **AR TREBUI** să aplice penalizări pentru parole scurte (¡ 8 caractere).
FR-PWD-009	Agentul **TREBUIE** să limiteze lungimea parolei acceptate la 128 caractere.
FR-PWD-010	Agentul NU **TREBUIE** să stocheze sau să logheze parola în clar.

3.4 Agent Verificare Primalitate

3.4.1 Prime Factorization

ID	Cerință
FR-PRM-001	Agentul **TREBUIE** să verifice primalitatea numerelor folosind Miller-Rabin deterministic pentru numere ¡ 2^64.

ID	Cerință
FR-PRM-002	Agentul **TREBUIE** să integreze YAFU pentru factorizare avansată.
FR-PRM-003	Agentul **TREBUIE** să utilizeze FactorDB ca fallback pentru numere mari.
FR-PRM-004	Agentul **TREBUIE** să implementeze cache LRU in-memory + persistent BoltDB.
FR-PRM-005	Agentul **TREBUIE** să returneze factorii primi și metoda folosită.
FR-PRM-006	Agentul **TREBUIE** să limiteze numărul maxim de cifre acceptate (default: 1000).
FR-PRM-007	Agentul **TREBUIE** să implementeze timeout-uri per backend (YAFU: 5s primality, 8s factor).
FR-PRM-008	Agentul **AR TREBUI** să raporteze timpul de calcul în răspuns.
FR-PRM-009	Agentul **TREBUIE** să expună endpoint /history pentru ultimele rezultate.
FR-PRM-010	Agentul **TREBUIE** să gestioneze concurența YAFU cu semaphore (default: 2).

3.5 Agent Specialist Teorie

3.5.1 RAG pentru Criptografie

ID	Cerință
FR-RAG-001	Agentul **TREBUIE** să suporte ingestia documentelor PDF, Markdown și Text.
FR-RAG-002	Agentul **TREBUIE** să stocheze embeddings în ChromaDB cu persistență.
FR-RAG-003	Agentul **TREBUIE** să utilizeze FastEmbed (BAAI/bge-small-en-v1.5) pentru vectorizare.
FR-RAG-004	Agentul **TREBUIE** să implementeze reranking cu cross-encoder (BAAI/bge-reranker-base).
FR-RAG-005	Agentul **TREBUIE** să mențină istoricul conversațiilor cu context tracking.
FR-RAG-006	Agentul **TREBUIE** să returneze surse (citări) pentru fiecare răspuns generat.
FR-RAG-007	Agentul **AR TREBUI** să suporte hybrid retrieval (vector + BM25).
FR-RAG-008	Agentul **TREBUIE** să suporte multiple LLM providers (Ollama, OpenAI, Gemini).
FR-RAG-009	Agentul **TREBUIE** să permită auto-ingestia documentelor noi din folder monitorizat.
FR-RAG-010	Agentul **AR TREBUI** să permită selectarea direct_rag pentru bypass LLM.

3.6 Agent Executor Comenzi

3.6.1 Operații Criptografice

ID	Cerință
FR-CMD-001	Agentul **TREBUIE** să suporte operații de encoding: Base64, Hex.
FR-CMD-002	Agentul **TREBUIE** să suporte hashing: SHA-256/384/512, SHA3, BLAKE2, MD5, HMAC.
FR-CMD-003	Agentul **TREBUIE** să suporte criptare simetrică AES-CBC + HMAC (Encrypt-then-MAC).
FR-CMD-004	Agentul **TREBUIE** să suporte criptare asimetrică RSA cu OAEP padding.
FR-CMD-005	Agentul **TREBUIE** să suporte semnături post-quantum (ML-DSA/Dilithium, Falcon).
FR-CMD-006	Agentul **TREBUIE** să valideze toate inputurile contra injection attacks.
FR-CMD-007	Agentul **TREBUIE** să redacteze secretele din logs/erori.
FR-CMD-008	Agentul **TREBUIE** să returneze comanda OpenSSL executată (scop educațional).
FR-CMD-009	Agentul **TREBUIE** să implementeze timeout per operație (default: 30s).
FR-CMD-010	Agentul **TREBUIE** să raporteze disponibilitatea PQC provider la /pqc/health.

3.7 Agent Selector Alegeri

3.7.1 NLP Intent Classification

ID	Cerință
FR-NLP-001	Agentul **TREBUIE** să clasifice intenția utilizatorului cu confidence score.
FR-NLP-002	Agentul **TREBUIE** să extragă entități relevante (numere, algoritmi, parole, chei).
FR-NLP-003	Agentul **TREBUIE** să utilizeze SecureBERT 2.0 pentru clasificare.
FR-NLP-004	Agentul **TREBUIE** să suporte minim 10 clase de intenție (encrypt, decrypt, hash, etc.).
FR-NLP-005	Agentul **TREBUIE** să returneze threshold de confidence configurabil.
FR-NLP-006	Agentul **AR TREBUI** să detecteze cereri ambigue și să solicite clarificare.
FR-NLP-007	Agentul **TREBUIE** să proceseze cereri în limba engleză.

ID	Cerință
FR-NLP-008	Agentul **POATE** să suporte input multilingv cu traducere automată.

3.8 Agent Detectare Criptosisteme

3.8.1 Cryptosystem Detection

ID	Cerință
FR-CRY-001	Agentul **TREBUIE** să detecteze tipul de criptosistem din ciphertext.
FR-CRY-002	Agentul **TREBUIE** să integreze CyberChef Magic detector.
FR-CRY-003	Agentul **AR TREBUI** să integreze euristici inspirate din dcode.fr.
FR-CRY-004	Agentul **TREBUIE** să agregeze rezultatele de la mai mulți detectori.
FR-CRY-005	Agentul **TREBUIE** să returneze scor de incredere (0-1) pentru fiecare detecție.
FR-CRY-006	Agentul **TREBUIE** să returneze top N candidați ordonați după scor (N configurabil).

3.9 Management Date și Conversații

3.9.1 Data Persistence

ID	Cerință
FR-DAT-001	Sistemul **TREBUIE** să stocheze istoricul conversațiilor per utilizator.
FR-DAT-002	Sistemul **TREBUIE** să permită reluarea conversațiilor anterioare.
FR-DAT-003	Sistemul **TREBUIE** să permită exportul rezultatelor în JSON.
FR-DAT-004	Sistemul **AR TREBUI** să permită exportul rapoartelor în PDF.
FR-DAT-005	Sistemul **TREBUIE** să implementeze TTL configurabil pentru cache (default: 1h).
FR-DAT-006	Sistemul **TREBUIE** să permită ștergerea datelor utilizatorului la cerere (GDPR).
FR-DAT-007	Sistemul **TREBUIE** să anonimizeze datele în log-uri.
FR-DAT-008	Sistemul **AR TREBUI** să implementeze backup automat al bazelor de date.
FR-DAT-009	Sistemul **TREBUIE** să definească retention policy pentru date (default: 90 zile).

ID	Cerință
FR-DAT-010	Sistemul **AR TREBUI** să permită exportul metadatelor conversațiilor.

3.10 Administrare și Audit

3.10.1 System Administration

ID	Cerință
FR-ADM-001	Sistemul **TREBUIE** să logheze toate acțiunile administrative în audit log.
FR-ADM-002	Sistemul **TREBUIE** să înregistreze timestamp, user ID, acțiune, resursa afectată, IP.
FR-ADM-003	Sistemul **TREBUIE** să ofere UI de administrare pentru utilizatori și roluri.
FR-ADM-004	Sistemul **TREBUIE** să ofere dashboard pentru management API keys.
FR-ADM-005	Sistemul **TREBUIE** să implementeze rate limiting configurabil per endpoint.
FR-ADM-006	Sistemul **TREBUIE** să implementeze quota per utilizator/API key.
FR-ADM-007	Sistemul **AR TREBUI** să alerteze la pattern-uri anormale (brute force, anomalii).
FR-ADM-008	Sistemul **TREBUIE** să permită configurări centralizate per mediu (dev/staging/prod).
FR-ADM-009	Sistemul **TREBUIE** să păstreze audit log-ul minim 5 ani.
FR-ADM-010	Sistemul **AR TREBUI** să ofere export audit log în format SIEM-compatible.

3.11 Interfață Utilizator

3.11.1 UI și UX

ID	Cerință
FR-UI-001	Interfața web **TREBUIE** să ofere input conversațional pentru cereri.
FR-UI-002	Interfața **TREBUIE** să afișeze rezultatele într-un format structurat și lizibil.
FR-UI-003	Interfața **TREBUIE** să afișeze sursele (citări) pentru răspunsurile RAG.
FR-UI-004	Interfața **TREBUIE** să permită navigarea între conversații anterioare.
FR-UI-005	Interfața **TREBUIE** să fie responsive pentru desktop, tabletă și mobil.
FR-UI-006	Interfața **AR TREBUI** să ofere mod întunecat (dark mode).

ID	Cerință
FR-UI-007	Interfața **TREBUIE** să afișeze status de loading pentru operații async.
FR-UI-008	Interfața **TREBUIE** să afișeze erori într-un mod user-friendly.
FR-UI-009	CLI-ul **TREBUIE** să ofere acces la toate funcționalitățile core.
FR-UI-010	CLI-ul **AR TREBUI** să suporte output în format JSON pentru scripting.

4 Cerințe Non-Funcționale

4.1 Performanță

4.1.1 Timp de Răspuns

ID	Cerință
NFR-PRF-001	Endpoint-urile lightweight (health, status) **TREBUIE** să răspundă în p95 ¡ 100ms.
NFR-PRF-002	Clasificarea intenției (Choice Maker) **TREBUIE** să se finalizeze în p95 ¡ 500ms.
NFR-PRF-003	Evaluarea parolei **TREBUIE** să se finalizeze în p95 ¡ 2s.
NFR-PRF-004	Verificarea primalității pentru numere ¡ 64 biți **TREBUIE** să fie ¡ 100ms.
NFR-PRF-005	Operațiile criptografice standard **TREBUIE** să se finalizeze în ¡ 1s.
NFR-PRF-006	Generarea RAG **TREBUIE** să returneze răspuns în p95 ¡ 10s (dependent de LLM).
NFR-PRF-007	Sistemul **TREBUIE** să suporte minim 5 de cereri concurente.
NFR-PRF-008	Sistemul **AR TREBUI** să suporte minim 5 utilizatori concurenți activi.
NFR-PRF-009	Cache-ul **TREBUIE** să reducă latența pentru cereri repetitive cu minim 80%.
NFR-PRF-010	Operațiile heavy (factorizare, RAG extins) **TREBUIE** să fie async cu polling.

4.2 Scalabilitate

4.2.1 Horizontal Scaling

ID	Cerință
NFR-SCL-001	Arhitectura **TREBUIE** să permită scalare orizontală pentru toți agenții.
NFR-SCL-002	Sistemul **TREBUIE** să funcționeze corect cu minim 2 replici per agent critic.
NFR-SCL-003	Baza de date **TREBUIE** să suporte connection pooling eficient.
NFR-SCL-004	Sistemul **AR TREBUI** să implementeze auto-scaling pe bază de load în K8s.
NFR-SCL-005	Sistemul **TREBUIE** să gestioneze backpressure la cereri excesive.

4.3 Fiabilitate

4.3.1 Availability și Recovery

ID	Cerință
NFR-REL-001	Disponibilitatea target pentru orchestrator și backend: $\geq 99.5\%$.
NFR-REL-002	Disponibilitatea target pentru agenți individuali: $\geq 99\%$.
NFR-REL-003	Sistemul **TREBUIE** să implementeze retry cu exponential backoff pentru dependențe externe.
NFR-REL-004	Sistemul **TREBUIE** să implementeze circuit breaker cu threshold configurabil.
NFR-REL-005	Sistemul **TREBUIE** să funcționeze în mod degradat când agenți non-critici sunt indisponibili.
NFR-REL-006	MTBF target pentru servicii critice: ≥ 720 ore.
NFR-REL-007	MTTR target: ≤ 30 minute.
NFR-REL-008	Backup-urile bazelor de date **TREBUIE** să fie automate și testate periodic.
NFR-REL-009	RTO (Recovery Time Objective): ≤ 4 ore.
NFR-REL-010	RPO (Recovery Point Objective): ≤ 1 oră.

4.4 Observabilitate

4.4.1 Monitoring și Logging

ID	Cerință
NFR-OBS-001	Toate serviciile **TREBUIE** să expună metrici Prometheus pe /metrics.
NFR-OBS-002	Sistemul **TREBUIE** să colecteze metrici RED (Rate, Errors, Duration).
NFR-OBS-003	Sistemul **TREBUIE** să colecteze metrici USE (Utilization, Saturation, Errors).
NFR-OBS-004	Toate serviciile **TREBUIE** să emită loguri structurate (JSON).
NFR-OBS-005	Log-urile **TREBUIE** să includă: timestamp, level, service, trace_id, message.
NFR-OBS-006	Sistemul **AR TREBUI** să implementeze distributed tracing (OpenTelemetry).
NFR-OBS-007	Sistemul **TREBUIE** să ofere dashboards Grafana pentru monitorizare.

ID	Cerință
NFR-OBS-008	Sistemul **TREBUIE** să configureze alerting pentru metrici critice.
NFR-OBS-009	Alertele critice **TREBUIE** să fie notificate în ¡ 5 minute de la incident.
NFR-OBS-010	Sistemul **AR TREBUI** să implementeze anomaly detection pentru pattern-uri neobișnuite.

5 Cerințe de Securitate

5.1 Transport Security

5.1.1 TLS și mTLS

ID	Cerință
SR-TLS-001	Toate comunicațiile externe **TREBUIE** să utilizeze TLS 1.2+.
SR-TLS-002	Comunicațiile inter-servicii în producție **TREBUIE** să utilizeze mTLS.
SR-TLS-003	Certificatele **TREBUIE** să aibă minimum 2048-bit RSA sau ECDSA P-256.
SR-TLS-004	Sistemul **TREBUIE** să implementeze certificate rotation automată.
SR-TLS-005	Sistemul **TREBUIE** să forțeze HSTS cu max-age $\geq$ 1 an.

5.2 Data Security

5.2.1 Protecția Datelor

ID	Cerință
SR-DAT-001	Datele sensibile at-rest **TREBUIE** să fie criptate (AES-256-GCM).
SR-DAT-002	Parolele **TREBUIE** să fie hashuite cu bcrypt/Argon2 (cost $\geq$ 12).
SR-DAT-003	API keys **TREBUIE** să fie stocate hashuite, afișate o singură dată.
SR-DAT-004	Secretele NU **TREBUIE** să fie stocate în cod sau imagini container.
SR-DAT-005	Sistemul **TREBUIE** să utilizeze secrets management (Vault/K8s Secrets).
SR-DAT-006	Log-urile NU **TREBUIE** să conțină date sensibile în clar.
SR-DAT-007	Baza de date **TREBUIE** să fie accesibilă doar din rețeaua internă.

5.3 Input Validation și Injection Prevention

5.3.1 Validare și Sanitizare

ID	Cerință
SR-INJ-001	Sistemul **TREBUIE** să prevină SQL Injection prin parametrizare.

ID	Cerință
SR-INJ-002	Sistemul **TREBUIE** să prevină Command Injection prin validare strictă.
SR-INJ-003	Sistemul **TREBUIE** să prevină XSS prin sanitizare input și output encoding.
SR-INJ-004	Sistemul **TREBUIE** să prevină CSRF prin token-uri per sesiune.
SR-INJ-005	Sistemul **TREBUIE** să prevină Path Traversal cu validare și sandboxing.
SR-INJ-006	Sistemul **TREBUIE** să implementeze allowlist pentru algoritmi și operațiuni.
SR-INJ-007	Sistemul **TREBUIE** să valideze toate inputurile server-side.
SR-INJ-008	Sistemul **TREBUIE** să implementeze request size limits (default: 1MB).

5.4 Access Control

5.4.1 Autorizare și Rate Limiting

ID	Cerință
SR-ACC-001	Sistemul **TREBUIE** să implementeze principiul privilegiilor minime.
SR-ACC-002	Sistemul **TREBUIE** să verifice autorizarea pentru fiecare cerere.
SR-ACC-003	Sistemul **TREBUIE** să implementeze rate limiting per IP și per user.
SR-ACC-004	Sistemul **TREBUIE** să blocheze conturile după 5 încercări eșuate (30 min).
SR-ACC-005	Sistemul **AR TREBUI** să implementeze IP reputation și blacklisting.
SR-ACC-006	Sistemul **AR TREBUI** să detecteze și să blocheze brute force attacks.

5.5 Security Headers

5.5.1 HTTP Security

ID	Cerință
SR-HDR-001	Sistemul **TREBUIE** să seteze Content-Security-Policy restrictiv.
SR-HDR-002	Sistemul **TREBUIE** să seteze X-Frame-Options: DENY.
SR-HDR-003	Sistemul **TREBUIE** să seteze X-Content-Type-Options: nosniff.

ID	Cerință
SR-HDR-004	Sistemul **TREBUIE** să seteze Referrer-Policy: strict-origin-when-cross-origin.
SR-HDR-005	Sistemul **TREBUIE** să configureze CORS restrictiv (nu wildcard în producție).

5.6 Audit și Incident Response

5.6.1 Logging și Forensics

ID	Cerință
SR-AUD-001	Sistemul **TREBUIE** să logheze toate accesele la resurse sensibile.
SR-AUD-002	Sistemul **TREBUIE** să logheze toate operațiunile administrative.
SR-AUD-003	Sistemul **TREBUIE** să păstreze audit logs imutabile pentru investigații.
SR-AUD-004	Sistemul **AR TREBUI** să alerteze la comportament suspect (anomalii).
SR-AUD-005	Sistemul **TREBUIE** să permită investigație și forensics post-incident.
SR-AUD-006	Sistemul **AR TREBUI** să ofere export pentru SIEM integration.

6 Cerințe AI/ML

6.1 Model Specification

6.1.1 Modele ML Utilizate

ID	Cerință
ML-MOD-001	PassGPT **TREBUIE** să utilizeze model pre-antrenat (javirandor/passgpt-10characters).
ML-MOD-002	SecureBERT **TREBUIE** să utilizeze versiunea 2.0 pentru clasificare.
ML-MOD-003	Embedding model pentru RAG **TREBUIE** să fie BAAI/bge-small-en-v1.5.
ML-MOD-004	Reranker **TREBUIE** să fie BAAI/bge-reranker-base (ONNX).
ML-MOD-005	Toate modelele **TREBUIE** să aibă checksum verificat la încărcare.
ML-MOD-006	Modelele **TREBUIE** să fie versionate și etichetate în registry.

6.2 Data Management

6.2.1 Gestiunea Datelor ML

ID	Cerință
ML-DAT-001	Documentele ingestate **TREBUIE** să fie clasificate și etichetate.
ML-DAT-002	Sistemul **TREBUIE** să păstreze metadata pentru fiecare document.
ML-DAT-003	Sistemul **AR TREBUI** să permită actualizarea incrementală a vectorilor.
ML-DAT-004	Sistemul **TREBUIE** să permită ștergerea selectivă din vector store.
ML-DAT-005	Dataset-urile de antrenament **TREBUIE** să fie documentate și versionate.

6.3 Guardrails și Safety

6.3.1 Protecție și Siguranță ML

ID	Cerință
ML-GRD-001	Sistemul **TREBUIE** să valideze inputul înainte de procesare ML.
ML-GRD-002	Sistemul **TREBUIE** să limiteze lungimea inputului acceptat (context window).

ID	Cerință
ML-GRD-003	Sistemul **TREBUIE** să filtreze output-urile pentru conținut harmful.
ML-GRD-004	Sistemul **TREBUIE** să implementeze limită de acțiuni per sesiune.
ML-GRD-005	Sistemul **AR TREBUI** să detecteze și să blocheze prompt injection attempts.
ML-GRD-006	Sistemul NU **TREBUIE** să expună informații sensibile prin model outputs.

6.4 Model Lifecycle (MLOps)

6.4.1 Management și Deployment Modele

ID	Cerință
ML-OPS-001	Sistemul **TREBUIE** să suporte blue-green deployment pentru modele.
ML-OPS-002	Sistemul **TREBUIE** să monitorizeze drift-ul modelelor.
ML-OPS-003	Sistemul **AR TREBUI** să implementeze A/B testing pentru modele noi.
ML-OPS-004	Sistemul **TREBUIE** să permită rollback rapid la versiunea anterioară.
ML-OPS-005	Sistemul **TREBUIE** să păstreze metrici de performanță per versiune model.

6.5 Ethics și Transparency

6.5.1 Transparență AI

ID	Cerință
ML-ETH-001	Sistemul **TREBUIE** să informeze utilizatorii că răspunsurile sunt generate de AI.
ML-ETH-002	Sistemul **TREBUIE** să ofere confidence scores pentru predicții.
ML-ETH-003	Sistemul **AR TREBUI** să documenteze limitările cunoscute ale modelelor.
ML-ETH-004	Sistemul NU **TREBUIE** să pretindă certitudine pentru rezultate probabilistice.

7 Cazuri de Utilizare

7.1 Anonymous

ID	Cerință
UC-ANON-001	Utilizatorul Anonymous **TREBUIE** să poată accesa prezentarea publică a platformei și lista de unelte/agenți.
UC-ANON-002	Utilizatorul Anonymous **TREBUIE** să poată crea un cont nou.
UC-ANON-003	Utilizatorul Anonymous **TREBUIE** să poată iniția autentificarea și recuperarea parolei.
UC-ANON-004	Utilizatorul Anonymous **TREBUIE** să poată rula o demonstrație limitată pentru Password Checker.
UC-ANON-005	Utilizatorul Anonymous **TREBUIE** să poată rula o demonstrație limitată pentru Prime Checker (numere mici).
UC-ANON-006	Utilizatorul Anonymous **TREBUIE** să poată rula o demonstrație limitată de detectare criptosisteme cu scor de încredere.
UC-ANON-007	Utilizatorul Anonymous **TREBUIE** să poată consulta statusul serviciilor și documentația publică API/CLI.

7.2 User

ID	Cerință
UC-USER-001	Utilizatorul User **TREBUIE** să poată gestiona sesiunea (login, refresh token, logout).
UC-USER-002	Utilizatorul User **TREBUIE** să poată trimite cereri către Orchestrator pentru rutare multi-agent și răspuns agregat.
UC-USER-003	Utilizatorul User **TREBUIE** să poată audita parole prin agregatorul ML (PassGPT, zxcvbn, HIBP).
UC-USER-004	Utilizatorul User **TREBUIE** să poată iniția job-uri Hash Breaker și să urmărească statusul lor asincron.
UC-USER-005	Utilizatorul User **TREBUIE** să poată verifica primalitatea și factorizarea numerelor și să acceseze istoricul cererilor.
UC-USER-006	Utilizatorul User **TREBUIE** să poată executa operații criptografice (encode/hash/AES/RSA/PQC) prin Command Executor.
UC-USER-007	Utilizatorul User **TREBUIE** să poată utiliza modul educațional (Theory Specialist RAG + CTF Tool) pentru întrebări și exerciții.

7.3 Admin

ID	Cerință
UC-ADMIN-001	Utilizatorul Admin **TREBUIE** să poată administra utilizatorii (creare, blocare, resetare).
UC-ADMIN-002	Utilizatorul Admin **TREBUIE** să poată configura roluri și permisiuni RBAC.
UC-ADMIN-003	Utilizatorul Admin **TREBUIE** să poată genera și revoca chei API pentru integrări externe.
UC-ADMIN-004	Utilizatorul Admin **TREBUIE** să poată configura politici de acces și rate limiting.
UC-ADMIN-005	Utilizatorul Admin **TREBUIE** să poată administra corpusul RAG (ingestie, reindexare, ștergere).
UC-ADMIN-006	Utilizatorul Admin **TREBUIE** să poată monitoriza serviciile, metricile și alertele operaționale.
UC-ADMIN-007	Utilizatorul Admin **TREBUIE** să poată consulta și exporta audit logs pentru investigații.

7.4 Sistem Extern (API Client)

ID	Cerință
UC-EXT-001	Sistemul extern **TREBUIE** să poată autentifica cereri folosind API key sau token JWT.
UC-EXT-002	Sistemul extern **TREBUIE** să poată apela endpoint-ul /v1/orchestrate și să primească răspuns agregat.
UC-EXT-003	Sistemul extern **TREBUIE** să poată apela direct agenții (password/prime/cryptosystem/command executor).
UC-EXT-004	Sistemul extern **TREBUIE** să poată iniția job-uri Hash Breaker și să interogheze statusul lor.
UC-EXT-005	Sistemul extern **TREBUIE** să poată interoga Theory Specialist (RAG) și să primească sursele folosite.
UC-EXT-006	Sistemul extern **TREBUIE** să poată consulta endpoint-uri de health/status pentru monitorizare.
UC-EXT-007	Sistemul extern **TREBUIE** să primească coduri de eroare standard și headere de rate limiting pentru retry/backoff.

8 Cerințe de Infrastructură și DevOps

8.1 Containerizare și Orchestrare

8.1.1 Kubernetes

ID	Cerință
INF-K8S-001	Sistemul **TREBUIE** să ruleze în Kubernetes cu namespace segregation.
INF-K8S-002	Sistemul **TREBUIE** să definească resource limits pentru toate container-ele.
INF-K8S-003	Sistemul **TREBUIE** să implementeze network policies pentru izolare.
INF-K8S-004	Sistemul **TREBUIE** să utilizeze non-root containers.
INF-K8S-005	Sistemul **TREBUIE** să implementeze pod security standards.
INF-K8S-006	Sistemul **AR TREBUI** să utilizeze service mesh (Istio/Linkerd).
INF-K8S-007	Sistemul **TREBUIE** să implementeze health checks (liveness/readiness).
INF-K8S-008	Sistemul **AR TREBUI** să suporte horizontal pod autoscaling.

8.2 CI/CD Pipeline

8.2.1 Automatizare Build și Deploy

ID	Cerință
INF-CIC-001	Pipeline **TREBUIE** să execute build automat la fiecare commit.
INF-CIC-002	Pipeline **TREBUIE** să execute unit tests cu coverage $\geq 70\%$.
INF-CIC-003	Pipeline **TREBUIE** să execute static analysis (linters).
INF-CIC-004	Pipeline **TREBUIE** să execute security scanning (Trivy, Snyk).
INF-CIC-005	Pipeline **TREBUIE** să execute integration tests.
INF-CIC-006	Pipeline **AR TREBUI** să execute SAST și DAST.
INF-CIC-007	Pipeline **TREBUIE** să genereze și să publice imagini cu tag semantic.
INF-CIC-008	Pipeline **TREBUIE** să implementeze deployment automat în staging.

ID	Cerință
INF-CIC-009	Pipeline **AR TREBUI** să suporte canary deployments în producție.
INF-CIC-010	Pipeline **TREBUIE** să permită rollback rapid (¡ 5 minute).

8.3 Deployment și Portability

8.3.1 Portabilitate și Instalare

ID	Cerință
INF-DEP-001	Sistemul **TREBUIE** să suporte deployment on-premise.
INF-DEP-002	Sistemul **AR TREBUI** să suporte deployment în cloud (AWS/GCP/Azure).
INF-DEP-003	Sistemul **TREBUIE** să funcționeze pe Linux (Ubuntu 22.04+, Debian 12+).
INF-DEP-004	Sistemul **AR TREBUI** să suporte air-gapped deployment.
INF-DEP-005	Configurația **TREBUIE** să fie externalizată prin env vars/ConfigMaps.
INF-DEP-006	Sistemul **TREBUIE** să ofere documentație completă pentru deployment.

8.4 Operational Readiness

8.4.1 Producție

ID	Cerință
INF-OPS-001	Sistemul **TREBUIE** să aibă runbook-uri pentru incident response și operațiuni critice.
INF-OPS-002	Procedurile de backup și restore **TREBUIE** să fie documentate și testate periodic.
INF-OPS-003	Sistemul **TREBUIE** să aibă plan de disaster recovery cu RTO/RPO validate.
INF-OPS-004	Audit log-urile **TREBUIE** să fie protejate împotriva modificării și accesate doar cu roluri dedicate.
INF-OPS-005	Release-urile **TREBUIE** să treacă prin quality gates (teste, scanări, verificări de securitate).
INF-OPS-006	Sistemul **TREBUIE** să efectueze audit de securitate periodic (cel puțin anual sau per release major).
INF-OPS-007	Alertele critice **TREBUIE** să fie rutate către un canal de on-call.

9 Cerințe de Conformitate

9.1 Security Standards

9.1.1 Conformitate Securitate

ID	Cerință
CMP-SEC-001	Sistemul **TREBUIE** să respecte OWASP Top 10 (2021).
CMP-SEC-002	Sistemul **AR TREBUI** să respecte CIS Benchmarks pentru containerizare.
CMP-SEC-003	Sistemul **AR TREBUI** să respecte NIST Cybersecurity Framework.
CMP-SEC-004	Operațiunile criptografice **AR TREBUI** să respecte NIST SP 800-57.
CMP-SEC-005	Post-quantum crypto **AR TREBUI** să respecte NIST PQC standards.

9.2 Data Protection

9.2.1 GDPR Compliance

ID	Cerință
CMP-GDP-001	Sistemul **TREBUIE** să permită exercitarea dreptului la ștergere (Art. 17 GDPR).
CMP-GDP-002	Sistemul **TREBUIE** să permită exportul datelor personale (Art. 20 GDPR).
CMP-GDP-003	Sistemul **TREBUIE** să documenteze fluxurile de date personale.
CMP-GDP-004	Sistemul **TREBUIE** să minimizeze colectarea datelor (Art. 5 GDPR).
CMP-GDP-005	Sistemul **AR TREBUI** să implementeze pseudonimizare unde posibil.

10 Anexă

10.1 Diagrame UML

10.1.1 Arhitectură generală

Arhitectura platformei software de unelte AI în criptografie este organizată pe niveluri:

1. **Nivel Frontend** - React Web, CLI Client, API Consumers
2. **Nivel API Gateway** - Go Backend (Auth, Rate Limiting, Routing)
3. **Nivel Orchestrare** - Orchestrator (Intent Routing, LLM)
4. **Nivel Agenți** - agenți specializați
5. **Nivel Date** - PostgreSQL, Redis, ChromaDB, BoltDB

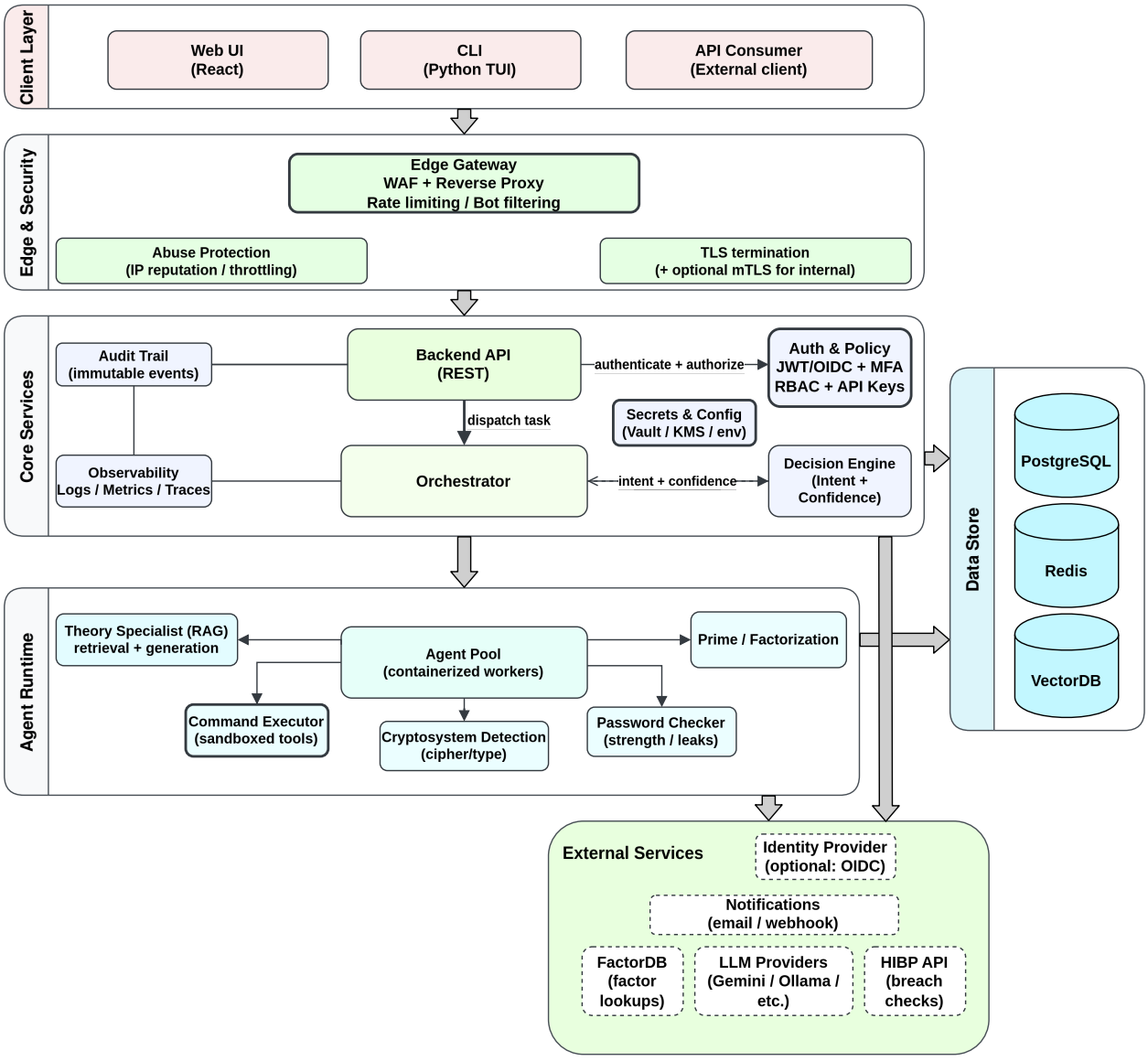

Figura 1: Arhitectura generală a platformei

10.1.2 Cazuri de utilizare - Anonymous

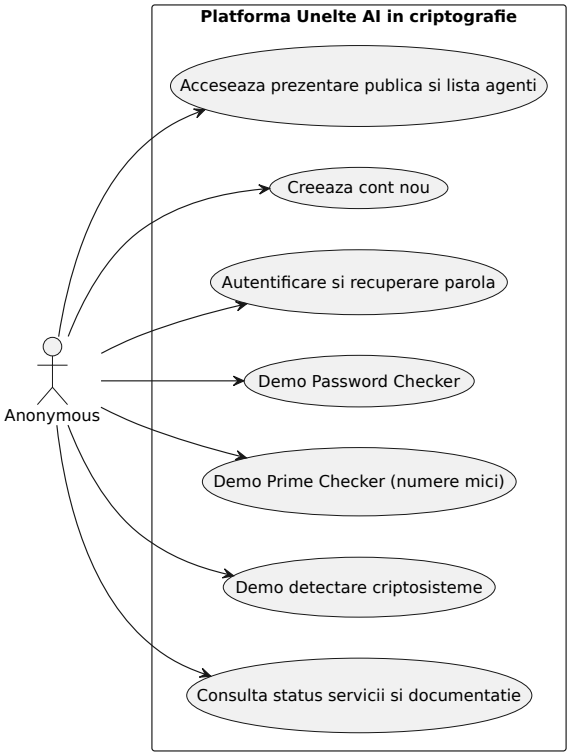

Figura 2: Cazuri de utilizare pentru Anonymous

10.1.3 Cazuri de utilizare - User

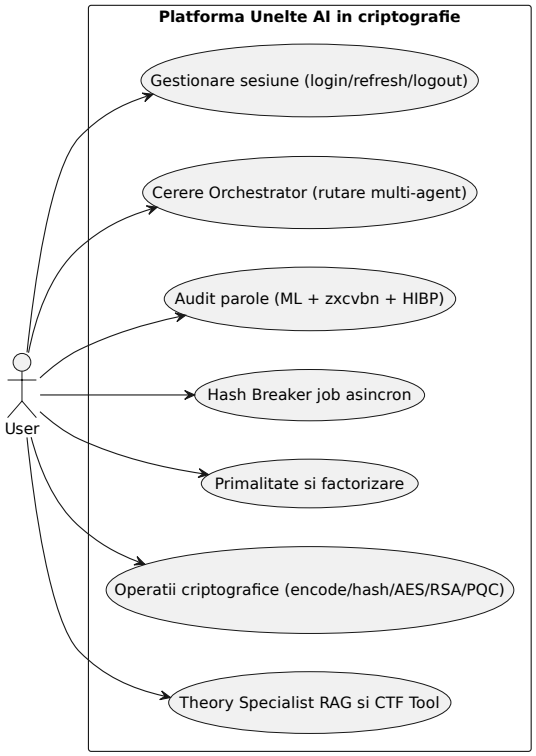

Figura 3: Cazuri de utilizare pentru User

10.1.4 Cazuri de utilizare - Admin

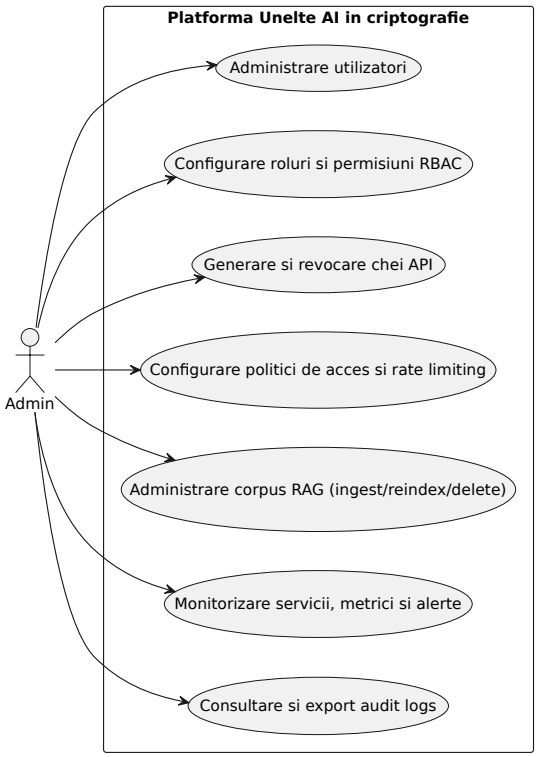

Figura 4: Cazuri de utilizare pentru Admin

10.1.5 Cazuri de utilizare - Sistem extern

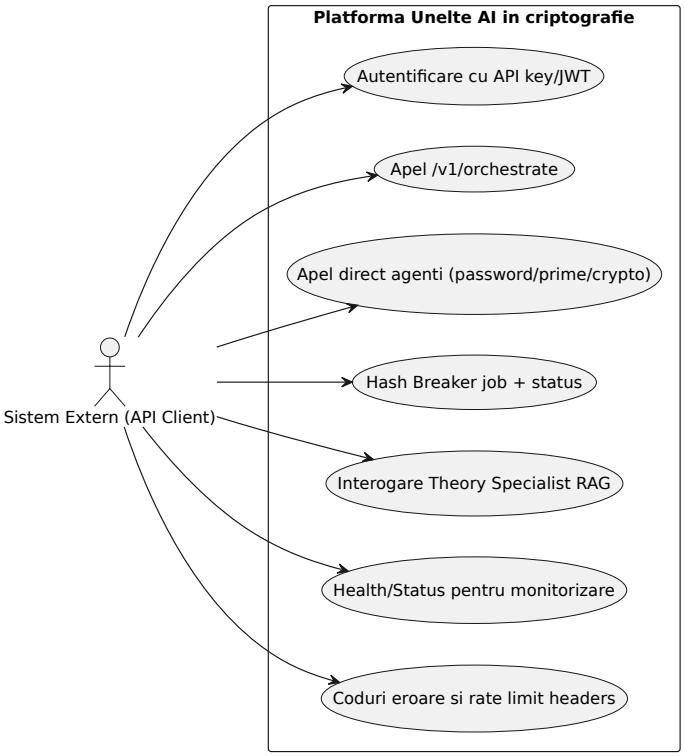

Figura 5: Cazuri de utilizare pentru Sistem Extern (API Client)

10.1.6 Flux secvențial cerere-răspuns

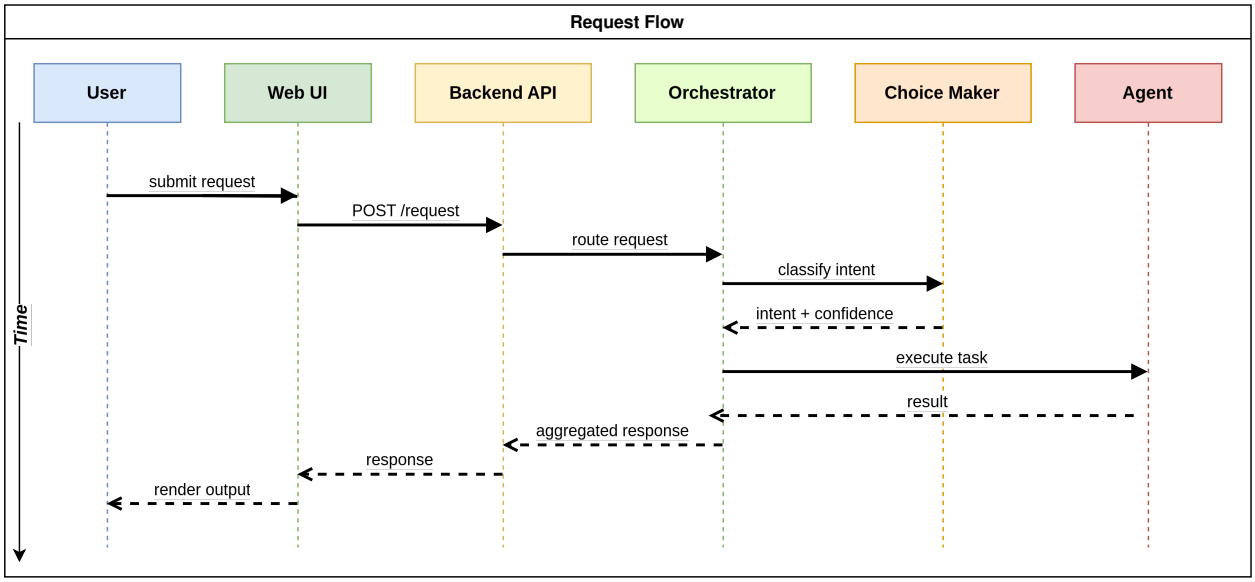

Figura 6: Flux secvențial pentru o cerere standard

10.1.7 Flux asincron pentru operații heavy

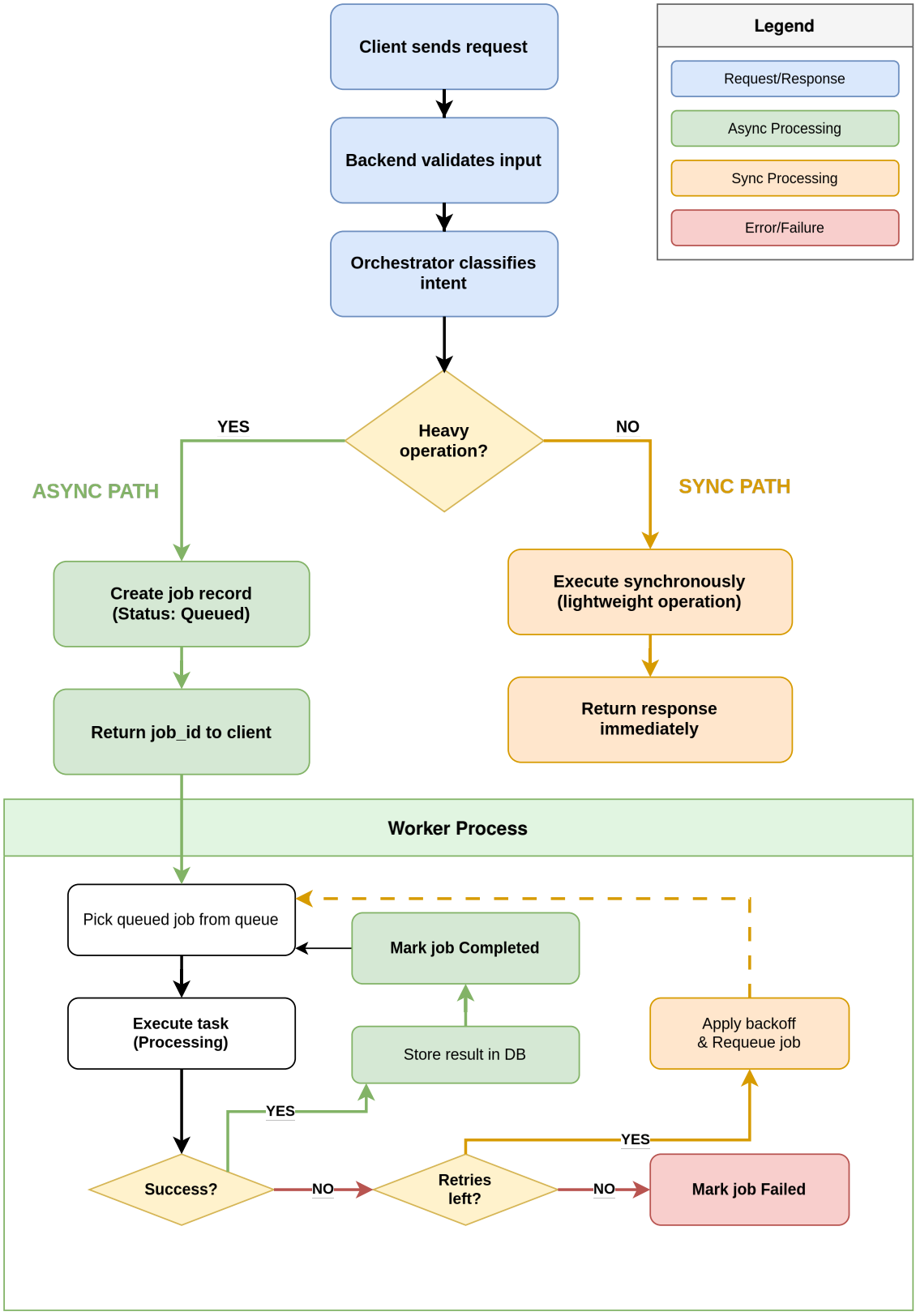

Figura 7: Flux asincron pentru operații heavy

10.1.8 Ciclul de viață al job-urilor

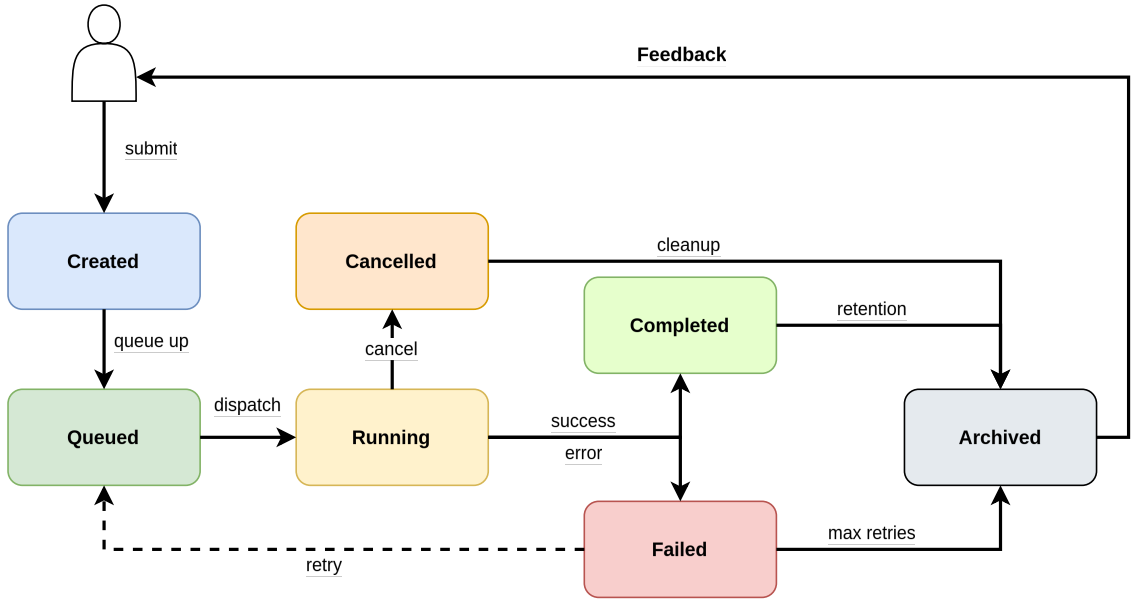

Figura 8: Lifecycle job-uri (queued, running, completed/failed)

10.1.9 Agent Verificare Parole

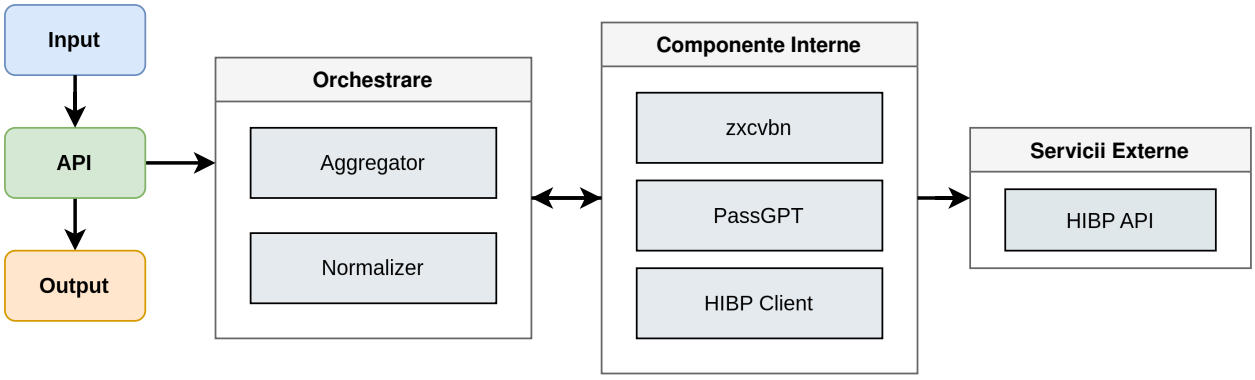

Figura 9: Diagramă componentă pentru Password Checker

10.1.10 Agent Verificare Primalitate

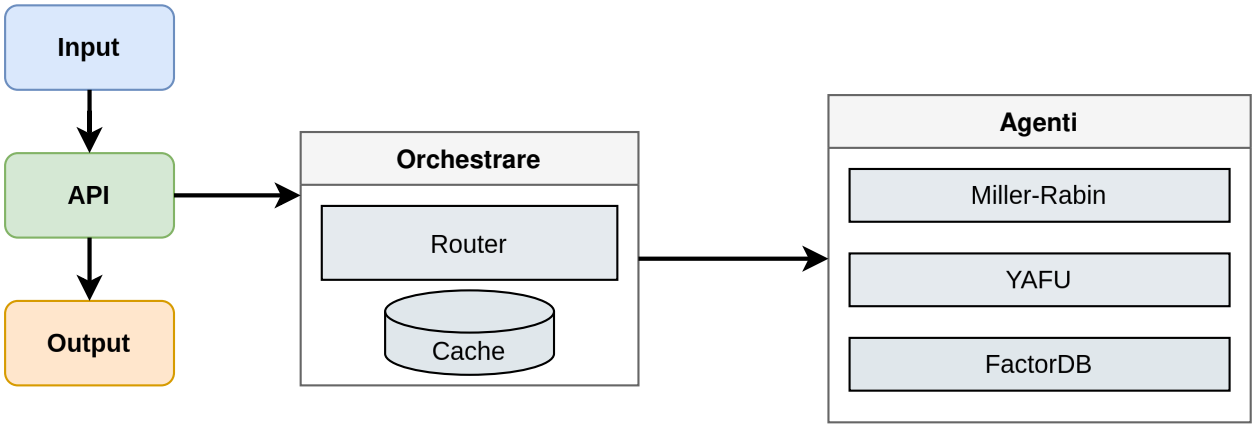

Figura 10: Diagramă componentă pentru Prime Checker

10.1.11 Agent Specialist Teorie (RAG)

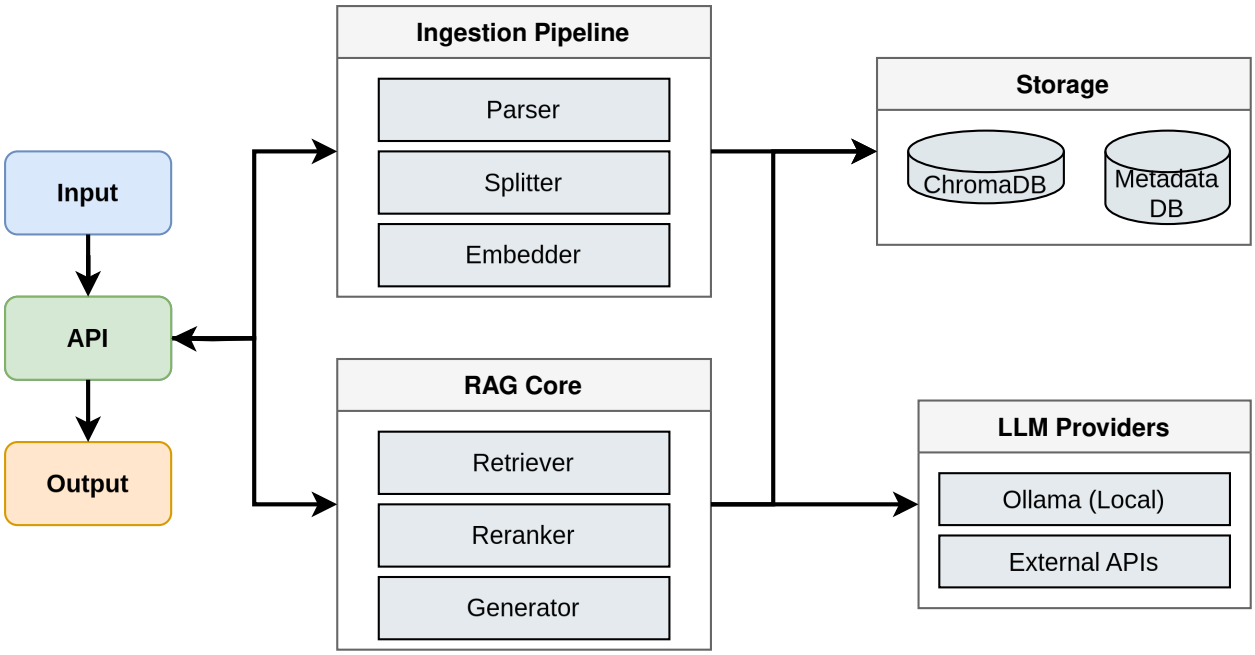

Figura 11: Diagramă componentă pentru Theory Specialist

10.1.12 Agent Executor Comenzi

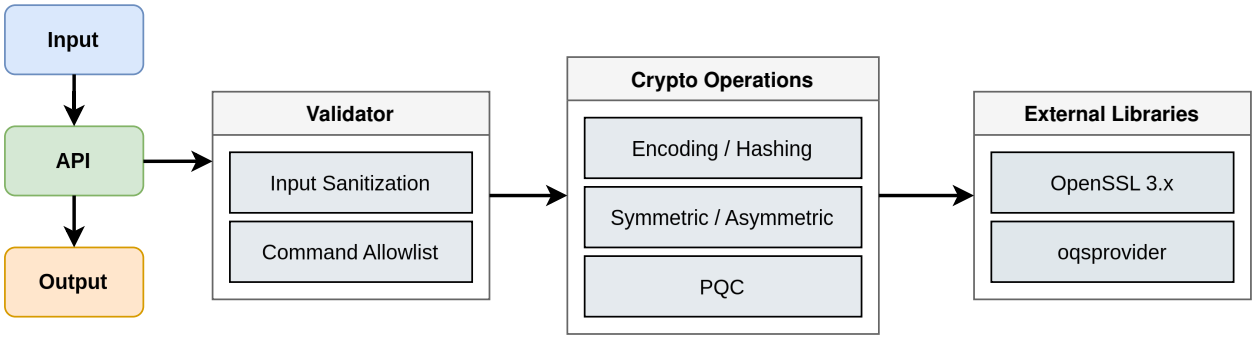

Figura 12: Diagramă componentă pentru Command Executor

10.1.13 Agent Choice Maker (clasificare)

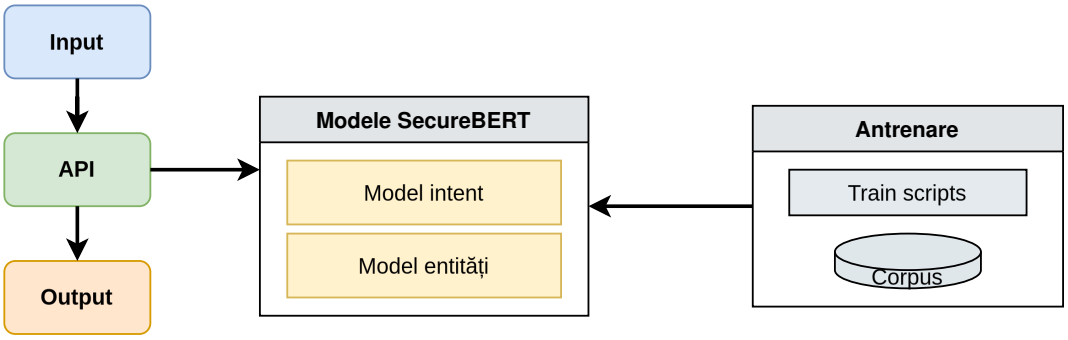

Figura 13: Diagramă componentă pentru Choice Maker (inferență)

10.1.14 Agent Choice Maker (generare întrebări)

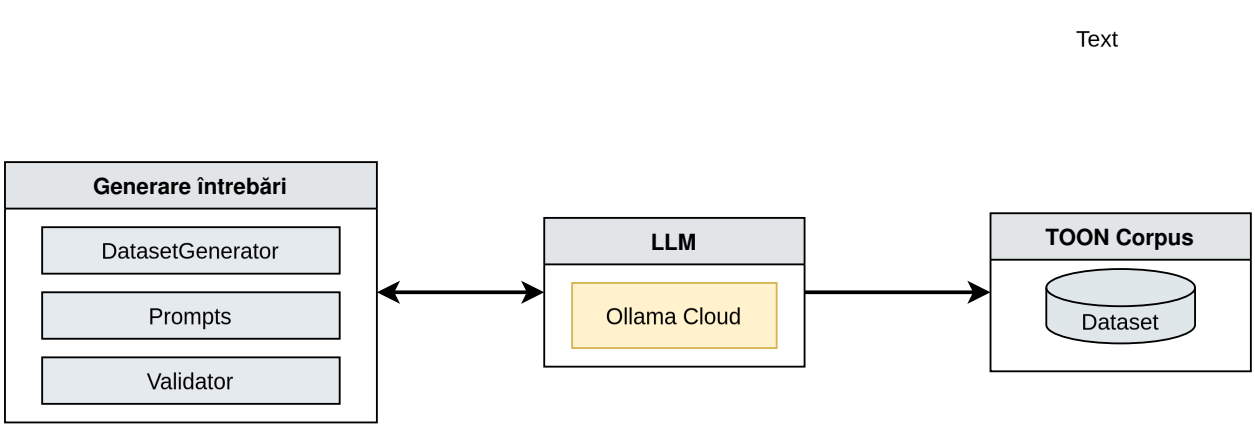

Figura 14: Diagramă componentă pentru Choice Maker (training)

10.1.15 Agent Detectare Criptosisteme

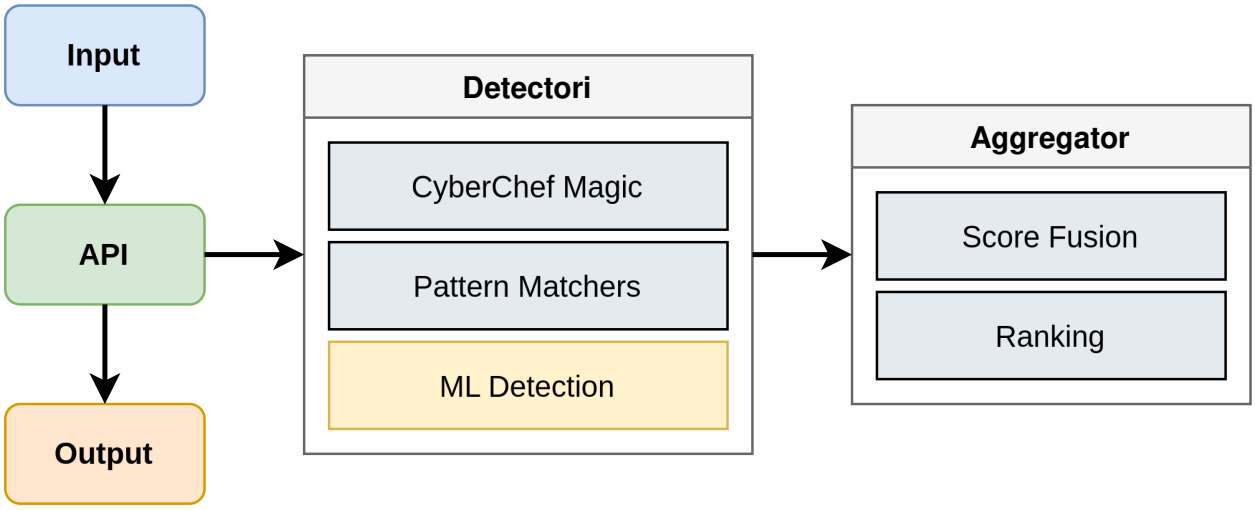

Figura 15: Diagramă componentă pentru Cryptosystem Detection