Analiza szczegółowa załączonego dokumentu

1. Streszczenie źródła

Dokument wejściowy zawiera kilka warstw naraz:

- architekturę portalu pierwotnie projektowanego pod /ai-truth;
- pakiet PPP / współpracy publiczno-technologicznej;
- mapę incydentów cyber/AI;
- mapę prawa i obowiązków;
- propozycję wspólnego źródła prawdy;
- moduły evidence, gap/abuse, legal live i effectiveness proof;
- komunikat dla dev/security/hacker/open-source/AI builders;
- propozycję rojów agentowych i dashboardów;
- warstwę szkoleń, ludzi, odpowiedzialności, szkód i regresu;
- potrzebę oddzielenia symulacji od faktów.

2. Najważniejsze ustalenie

Największą wartością dokumentu nie jest sama lista stron. Największą wartością jest mechanizm łączenia prawa, faktów, dowodów, incydentów, ludzi, agentów, BigTech, szkód i procedur w jednym wspólnym źródle prawdy.

Dlatego rekomendowana nazwa głównego programu:

K0-CST - Common Source of Truth

3. Elementy, które dokument ma już dobrze opisane

Obszar	Ocena	Uzasadnienie
Architektura modułowa	Mocna	Dokument jasno rozbija portal na start, CST, incydenty, mapy, PPP, prawo, dev/security, szkolenia, dowody i dashboardy.
Priorytety P0/P1/P2	Mocna, ale wymaga korekty URL	W źródle P0/P1/P2 są sensowne, ale trzeba przenieść je z /ai-truth na nowy root.
PPP/P0	Mocna	Jest gotowy język bezpłatnej analizy P0 i pilotażu GO CONTROLLED.
Evidence layer	Mocna	Pojawia się claim, source, evidence, status, owner, timestamp, appeal path.
Dev/Security	Mocna	Jest komunikat odpowiedzialnego testowania i rozróżnienie ethical hacking vs brak autoryzacji.
Ludzie i odpowiedzialność	Mocna	Dokument identyfikuje człowieka jako słabe ogniwo i potrzebę matrixa odpowiedzialności.
Ryzyko mieszania symulacji z faktami	Bardzo ważne	To powinno stać się zasadą architektoniczną P0.

4. Luki do uzupełnienia przed wdrożeniem

Luka	Priorytet	Co dopisać
Brak jednoznacznego nowego rootu	P0	Wszystko nowe pod /Common-Source-of-Truth/start; /ai-truth jako archiwum/odnośnik.
Brak modelu danych w formacie dev	P0	JSON schema dla claim, source, evidence, incident, actor, damage, action, legal_trigger.
Brak RACI dla ludzi	P0/P1	Operator, owner, legal, security, dev, reviewer, publisher, incident lead.
Brak granic automatyzacji agentów	P0	Agent Action Inventory, human approval, audit log, kill switch, read-only default.
Brak rozdzielenia REAL/SIMULATION w URL i danych	P0	Osobne statusy i namespace, zakaz mieszania w dashboardach.
Brak acceptance criteria dla zadań dev	P0	Każdy task musi mieć output i kryterium odbioru.
Brak polityki publikacji danych wrażliwych	P0	Widoczność public/internal/restricted, anonimizacja, minimalizacja.
Brak jednoznacznej polityki braku przypisywania winy	P0	Nie mylić źródła technicznego z osobą odpowiedzialną; claim bez dowodu = GAP.

5. Rekomendacja architektoniczna

Nie budować jednej długiej strony. Zbudować portal modułowy z krótkimi stronami, których każda ma:

- cel;
- problem;
- rozwiązanie K0NSULT;
- tabelę danych;
- procedurę;
- status wdrożenia;
- link do formularza;
- link do dokumentu;
- link do dashboardu;
- ownera i kolejny krok.

6. Najważniejsze ryzyka

Ryzyko	Skutek	Kontrola
Nadmiar materiału w jednym miejscu	Chaos, trudność publikacji	Pakiet P0 jako landing + 6 stron; reszta jako roadmap.
Przypisanie winy bez dowodu	Ryzyko prawne i reputacyjne	Claim <= Proof, status GAP/DISPUTED, appeal path.
Automatyzacja bez nadzoru	Nieustalona odpowiedzialność	Agent Action Inventory, human-in-the-loop, operator veto.
Mieszanie symulacji z incydentami realnymi	Fałszywe alarmy SOC/SIEM	Status SIMULATION i separacja środowisk.
Aktualność prawa	Błędy w obowiązkach	Legal Live Board + data aktualizacji + źródła oficjalne.
Vendor lock-in	Uzależnienie technologiczne	Open-source/EU stack review i mapa zależności.

7. Wniosek

Dokument jest wystarczający, aby rozpocząć P0. Nie jest jeszcze wystarczający do uruchamiania aktywnych agentów, automatycznego monitoringu lub przypisywania odpowiedzialności. Najpierw

trzeba zbudować warstwę dowodową, model danych, zasady publikacji i granice testowania.