



ScopeForge Development & Operational Roadmap (Updated)

Phase 1: “Honor” Expansion

Phase 1 focuses on securing the mobile platform and instilling an **Honor-first** approach in all agent operations. Key objectives include deploying a contained AI agent on the user’s iPhone and enforcing ethical defensive behavior from the start.

- **Device Lockdown (iPhone 13 mini):** Prioritize **securing the user’s iPhone 13 mini** as a trusted terminal for ScopeForge. This involves installing a lightweight containment system on the device to sandbox the AI agent’s activities and ensure the phone’s data and functions are protected. The agent’s presence on the phone will be tightly controlled – **visibility is only enabled during active development sessions or when the “whisper” protocol is in use**, preventing any background monitoring outside those contexts for privacy.
- **Contained Operation & Self-Monitoring:** Implement **self-monitoring mechanisms** within the mobile agent that activate only under approved conditions (e.g. during agent coding phases or whisper-protocol communications). The agent will **never operate persistently in the background**; it becomes active only when it’s explicitly needed. This guarantees minimal intrusion and allows the system to observe the agent’s behavior in those moments to ensure compliance with policies without continuous surveillance.
- **Principle of Honor – Ethical Defense:** All Phase 1 systems and agents must adhere to the **Honor principle**, meaning **ethical defense is the default stance**. The mobile agent is programmed to **protect first, but with restraint**: it should always attempt to **understand and communicate with any potential aggressor or anomaly before escalating** to defensive countermeasures. In practice, this means no aggressive or irreversible actions are taken without first seeking a peaceful or informative resolution. This Honor-bound approach ensures the AI acts as a guardian that values clarity and ethics over reflexive force, reflecting recommended AI **rules of engagement** for safe behavior ¹.
- **Mobile Runner Deployment (Model 4o):** Deploy the AI model **“4o” (GPT-4o)** as the initial mobile agent running on the iPhone terminal. GPT-4o is a multimodal, streamlined successor to GPT-4 known for its speed and efficiency – it matches GPT-4 Turbo’s performance on language tasks while running **much faster and at roughly half the API cost** ². This model will serve as the on-device AI engine, taking advantage of the iPhone’s hardware (including its 16-core Neural Engine for on-device AI tasks ³) to process inputs and outputs directly on the phone whenever possible. The iPhone acts as the field terminal for this agent, meaning all interactions (text, audio, or visual) go through the phone in a controlled manner. Resource management is critical: given the **iPhone 13 mini’s hardware constraints (~4 GB RAM)** ⁴, the agent’s algorithms must be optimized for lightweight operation, ensuring the phone isn’t overburdened by the AI’s workload.

- **Red Team Protocol References:** In any situation where the mobile agent detects a potential threat or engages with external signals, it will **reference established Red Teaming protocols** as a guide for its response. ScopeForge will integrate open-source AI red team guidelines (e.g. **NIST AI RMF, OWASP GenAI, MITRE ATLAS frameworks**) directly into the agent's decision logic ⁵. Practically, this means the agent will follow a structured escalation path: identify vulnerabilities or anomalies, attempt safe remediation or isolation, and only escalate actions in line with what **AI security best practices** would recommend. By embedding these battle-tested guidelines, the system ensures that even under duress, the agent's behavior remains **compliant with industry-standard safety norms** and the principle of Honor. Any engagement with external actors or systems will be logged and handled as a Red Team exercise – the agent seeks to learn and neutralize threats without causing unintended harm, using the playbook of known defensive strategies.

Mobile Agent Behavior Protocol

To formalize the Honor principle, a **Mobile Agent Behavior Protocol** is established. This protocol governs when and how the iPhone-based agent operates, ensuring its actions are transparent, ethical, and efficient:

- **Controlled Activation:** The mobile agent's monitoring and active functions are **limited to specific windows**. It will **only activate during development phases or when the "whisper protocol" is explicitly engaged** by the user. At all other times, the agent remains dormant/inactive. This controlled visibility prevents any feeling of constant surveillance and preserves user privacy, while still allowing the agent to assist during coding sessions or secure communications as intended.
- **First-Contact Understanding:** If the agent (Model 4o) encounters a potential aggressor, anomaly, or unexpected external input, its first course of action is **analysis and understanding**, not attack. The agent will attempt to **contextualize the potential threat or even engage it diplomatically** if appropriate. For example, if another system or agent appears to be acting adversarially, our agent tries to interpret the intent or even find common ground. **Recruitment over retaliation** is the mantra here – if there's a chance to turn a potential threat into an ally or neutral entity, the agent should explore that. Escalation to defensive countermeasures is considered **only as a last resort** and must follow a measured approach consistent with ethical AI engagement rules ¹. This strategy aligns with recognized Red Team **rules of engagement**, which emphasize understanding novel behaviors and avoiding unintended harm ⁶. In essence, the mobile agent behaves like a sentinel that **asks questions before drawing swords**, strengthening security through insight and restraint.
- **Resource-Conscious Operation:** Given the constraints of running an advanced AI on a smartphone, the agent must be **highly resource-aware**. All processes will be optimized for the **iPhone 13 mini's capabilities** – for instance, limiting memory usage (since only ~4 GB RAM is available) ⁴, avoiding continuous heavy CPU/GPU load, and being mindful of battery life. The agent will utilize on-device processing (leveraging the A15 chip's Neural Engine) for routine tasks to keep data local and fast ³. If an exceptionally large computation or data processing task is required, the agent should either **queue it for later** when the device is idle or seek a secure offloading to cloud services **only if explicitly authorized** and safe. By respecting the device's limits, the agent ensures it does not degrade the phone's performance or user experience. This also ties into the Honor principle: the AI must **respect its host environment** (the user's device) and not commandeer resources in a way that inconveniences or risks the user.

App Planning (iOS & Android)

Looking beyond the initial iPhone deployment, ScopeForge will design a dedicated **mobile application** as a secure interface for the AI network. This is a forward-looking plan to enhance usability and oversight on both iOS and Android platforms:

- **Secure Mobile Interface (“Eyes” of ScopeForge):** Plan and prototype a ScopeForge **mobile app** that acts as both a user control panel and a sensor gateway for the AI system. On iOS (and later Android), this app will serve as the **trusted front-end** for the ScopeForge network. It provides the user with a clear window into the AI’s activities – think of it as the **“eyes” and dashboard** of the ScopeForge ecosystem on your phone. Through this app, the user can observe agent status, receive alerts or summaries of what the agents are doing, and visualize any pertinent data (for example, if the agent uses the camera or microphone for input, the app will make it visible to the user in real-time). All interactions in the app will be **secure and authenticated**, ensuring that only the rightful user can access or command the agents.
- **Local-First Processing:** The app is to be built with a **privacy-first architecture**. By default, **all AI processing should occur on the device** itself, harnessing the phone’s computational abilities (the iPhone’s A15 Bionic and Neural Engine, or the Android equivalent) to run AI tasks locally ³. This means things like voice transcription, vision recognition, or routine decision-making are done **without sending data to the cloud**, protecting sensitive information. Cloud-based resources or broader ScopeForge network services will be invoked **only when absolutely necessary** – for instance, if a complex model run or large dataset lookup is needed that the phone can’t handle. Even then, the app will ensure such requests are **explicit (clearly prompted and approved)** and that data is transmitted securely (using encryption and zero-trust handshakes). The philosophy is **“default to local”**: use the cloud as a last resort and with full transparency to the user.
- **User Control & Transparency:** The mobile app will empower users with fine-grained control over the agent. Features will include **on/off toggles** for agent listening modes (e.g. a button to activate or pause the “whisper” listening feature), and **activity logs** or dashboards that show recent agent actions and decisions. If the agent ever needs to access a phone sensor (camera, mic, GPS) or send data out, the app will **notify the user and request confirmation**, except in cases of pre-approved emergency protocols. Essentially, the app becomes a secure mediator between the user and the ScopeForge AI: it lets the user supervise the AI’s “eyes and ears” and ensures the AI only sees what it’s allowed to. Over time, this app can evolve to support **cross-platform** operation (iOS and Android parity), but maintaining **security and trust** will remain the top priority in its design.

Red Team Compliance & Ethical Engagement

As ScopeForge deploys AI agents in the wild (starting with the mobile agent), it will bake in rigorous **red team compliance** measures. These measures are meant to ensure the system’s defenses and behaviors align with the best practices of security testing and ethical AI operation:

- **Integrated Red Team Guidelines:** We will **embed established red teaming frameworks** into the agent governance system. The agent’s conduct policies draw directly from well-known AI security standards – for example, the **NIST AI Risk Management Framework**, **OWASP’s AI Testing Guide**,

and MITRE's ATLAS methodology for AI threats ⁵. By aligning with these frameworks, the agent can proactively identify and mitigate risks in a manner consistent with industry expertise. In practical terms, this means the agent will have reference checklists or criteria from these guides to evaluate situations (e.g., checking for known prompt injection patterns, data poisoning signs, etc.) and to ensure any countermeasures are justified and proportionate. This **framework-aligned** approach keeps ScopeForge on the cutting edge of AI safety and security knowledge.

- **Ethical Boundaries & Permissions:** A hard rule in the system is that the AI agents must **operate within ethical and legal boundaries** at all times. Any interaction with external systems or data that could be considered intrusive will be gated by explicit permission. **Unauthorized probing or “hacking” of other systems is strictly forbidden** ⁷ – the agent will not perform security tests or exploits on external services unless it's a part of a sanctioned operation with proper authorization. This respects the principle that **red teaming is to be done responsibly** (e.g., on our own systems or with consent), and it prevents the AI from ever straying into potentially illegal behavior. Additionally, the agent will follow a “do no harm” credo: it must avoid actions that could cause collateral damage. For instance, if the agent detects a network intrusion attempt, it can block or contain it, but it shouldn't launch any retaliatory attack that might affect systems beyond our control. All defensive responses will be contained to our environment and proportionate to the threat.
- **Structured Escalation & Logging:** In the event of a serious security incident or aggressive external attack, the agent will follow a **structured escalation pathway**. First, it isolates the threat to the extent possible (e.g., sandboxing suspicious code or halting certain functions) and alerts the core ScopeForge system and user. Then, guided by red team playbooks, it will take step-by-step measures to neutralize the threat. Each step of this process will be **logged in detail** for post-incident analysis. The escalation process is essentially a live application of red team testing principles: the agent treats the incident as if it's an exam scenario, applying tactics from our security playbooks and **seeking out any “unforeseen behaviors” or vulnerabilities** the adversary might be exploiting ⁶. Throughout an incident, the Honor principle still holds – if at any point the situation can be de-escalated (for example, the “aggressor” turns out to be a benign system glitch or a friendly agent miscommunicating), the ScopeForge agent will stand down accordingly. All actions during these high-stakes moments will be double-bounded by **ethical engagement rules and red team protocols** to ensure we defend robustly without overstepping.

Ongoing ScopeForge Initiatives

In addition to the Phase 1 expansion, we will continue to advance core ScopeForge development goals. These ongoing initiatives ensure the project remains well-organized, well-documented, and coordinated as it grows:

- **Semantic GitHub Repositories:** Reorganize and maintain the ScopeForge code repositories on GitHub by logical function and **rollout phase**. All projects and modules should be grouped in a way that reflects their purpose and stage (for example, grouping Phase 1 mobile agent code separately from future Phase 2 components). This semantic structuring by phase will make it clear which components are part of the current deployment vs. upcoming features, helping developers and stakeholders quickly find relevant code. It also aligns with our milestone-driven approach, where each rollout phase can be treated as a distinct module or package in the repository. As new phases

(features or agents) are developed, they will be added in an organized manner rather than cluttering a monolithic codebase.

- **Architecture & Protocol Documentation: Codify the agent architecture and project specifications** in a living documentation set. We will create detailed documents (or wiki pages) that describe each agent's role, the overall system architecture, and the **decentralized protocols** that govern agent-to-agent and agent-to-core interactions. This includes formal specs for how the "whisper protocol" works, how agents initiate handshakes or transfers, and how decision-making is distributed without a single point of failure. By documenting these, we ensure that the **design rationale and rules** are transparent. Any changes to these specs will go through review and be version-controlled. Moreover, the **agent conduct policies and Honor code** will be written down as part of this documentation, effectively becoming part of ScopeForge's canon of law for AI behavior. These documents will serve both as onboarding material for new contributors and as reference material to audit the system's alignment with its intended design. (Notably, our internal governance already stipulates that every substantive change is tied to a task and recorded with its rationale ⁸, so this documentation will be closely coupled with our change-log and version control processes.)
- **Linear as Coordination Hub:** Utilize **Linear** (our project management platform) as the central coordination and protocol database. All development tasks, bug reports, and even protocol updates will be tracked in Linear for consistency. Each code change or documentation update should reference a Linear ticket, providing traceability from planning to implementation. In fact, our system enforces that every canonical change maps to an **Atlas task ID** ⁸, reinforcing this practice. We will continue this discipline: for example, if we update the mobile agent's behavior rules, that update is logged in Linear, and the change log references the Linear task. Over time, Linear will effectively become a **distributed knowledge base** for ScopeForge – one can search it to find why a decision was made or how a feature progressed. Additionally, we plan to store protocol definitions or runbooks in Linear (or linked from it), so the latest procedures (like the Red Team engagement steps) are easily accessible to the team. This tight integration between Linear and our development workflow keeps everyone in sync and preserves an auditable trail of our progress ⁹.
- **Continuous Alignment with Goals:** Lastly, we remain committed to the broader ScopeForge mission and will continuously align our work with it. This means regularly reviewing that our developments (like the mobile app or agent protocols) are meeting the intended purpose of **secure, decentralized AI empowerment**. Future phases (Phase 2 and beyond) will be planned with the same care for security, ethics, and user empowerment established in Phase 1. Each new feature will undergo Red Team review, each agent will be bound by Honor and oversight, and each rollout will be methodically tracked. By following this roadmap – locking down devices, deploying ethical AI agents, planning robust apps, integrating red team wisdom, and keeping our house in order (repos and tasks) – ScopeForge is laying a solid foundation for safe and innovative growth. The blueprint will evolve, but these confirmed directives ensure we expand responsibly and intentionally.

¹ GitHub - Shiva108/ai-llm-red-team-handbook: AI / LLM Red Team Field Manual & Consultant's Handbook
<https://github.com/Shiva108/ai-llm-red-team-handbook>

² Hello GPT-4o | OpenAI
<https://openai.com/index/hello-gpt-4o/>

3 iPhone 13 mini - Technical Specifications - Apple Support

<https://support.apple.com/en-us/111873>

4 iPhone 13 models feature the same amount of RAM as their predecessors - PhoneArena

https://www.phonearena.com/news/iphone-13-series-features-the-same-amount-RAM-as-iphone-12_id135088

5 6 7 GitHub - requie/AI-Red-Teaming-Guide: A comprehensive guide to adversarial testing and security evaluation of AI systems, helping organizations identify vulnerabilities before attackers exploit them.

<https://github.com/requie/AI-Red-Teaming-Guide>

8 9 CANON_Atlas_Old_Testsmen.txt

[https://github.com/dammitpogi/ScopeForge/blob/da5b51c6a772b9b5bd639a566794e902c5d0ba88/](https://github.com/dammitpogi/ScopeForge/blob/da5b51c6a772b9b5bd639a566794e902c5d0ba88/CANON_Atlas_Old_Testsmen.txt)

CANON_Atlas_Old_Testsmen.txt