Aterges: AI and Agents Plan

Version: 1.0
Lead: Solopreneur (acting as Head of AI / Architect)
Objective: To define the strategy, architecture, and development roadmap for the core intelligence of the Aterges platform: the AI Orchestrator and the Hub of extensible Agents.

1. Core Objective

To build an intelligent and autonomous core that can accurately understand user intent, decompose complex problems into actionable steps, and delegate those steps to a suite of specialized, reliable agents. The goal is to move beyond simple data retrieval and enable true workflow automation.

2. Guiding Principles

- **Pragmatism over Dogma:** We will use the best AI model for each specific task (the "Mixed Fleet" strategy). The goal is the best possible result for the user, not loyalty to a single provider.
- **Autonomy, Not Shortcuts:** Agents are designed as flexible, parameter-driven **capabilities**, not as rigid, pre-defined commands. The AI Orchestrator determines *how* to use these capabilities based on the user's query.
- **Grounded in Fact:** The primary role of the AI is to reason and synthesize. It should always ground its final answers in the factual data retrieved by the Data Agents, minimizing the risk of "hallucinations."
- **Transparency:** Whenever possible, the system should be able to explain its reasoning, detailing which agents it used to arrive at a conclusion. This builds user trust.

3. Core Architecture: The Orchestrator & The Hub

Our AI architecture is a classic "hub and spoke" model, designed for modularity and scalability.

1. **The AI Orchestrator (The Hub):**
    - This is the central "brain" of the system, residing in our FastAPI backend.
    - It is responsible for the entire "Tool Calling" lifecycle:
        1. Receiving the user's natural language prompt.
        2. Describing the available Agents (tools) to the chosen LLM.
        3. Interpreting the LLM's decision to call a specific Agent with specific parameters.
        4. Executing the requested Agent.
        5. Sending the results from the Agent back to the LLM for final synthesis.
2. **The Hub of Agents (The Spokes):**

    - This is our library of specialized Python classes. Each agent is a self-contained module with a clear purpose.
    - **Data Agents (Read-Only):** These agents are experts at connecting to external APIs to extract and retrieve data. They form the basis of our MVP.
    - **Action Agents (Read/Write):** These agents interact with digital environments to perform tasks. They are the key to our long-term vision of automation and will be introduced in later phases.

4. AI Model Strategy (The "Mixed Fleet")

The Orchestrator will be designed to be model-agnostic, allowing us to leverage the strengths of different LLM providers.

- **Primary Workhorse (Initial): Google Gemini (via Vertex AI)**
    - **Role:** The default choice for most tasks, especially for interpreting user intent and interacting with the Google ecosystem. We will leverage Codey models for high-quality SQL generation for BigQuery.
    - **Reason:** Deep integration with our GCP infrastructure and access to startup credits.
- **Specialist Synthesizer: Anthropic Claude**
    - **Role:** To be called upon for generating long, coherent, and well-structured reports or complex analyses. Its large context window is a strategic advantage.
- **Reliable Benchmark: OpenAI GPT Series**
    - **Role:** Serves as a performance benchmark and a robust fallback option. Its "Function Calling" is very mature.
- **Strategic Long-Term Play: Open-Source Models (e.g., Llama)**
    - **Role:** For specific, fine-tuned tasks or to be offered as part of a self-hosted "Enterprise" plan for clients with extreme data privacy requirements.

5. Agent Development Roadmap

The capabilities of Aterges will grow as we build out our Hub of Agents in logical packs.

Phase 1: The Google Pack (MVP)

- **Goal:** Unify the core marketing and web analytics ecosystem.
- **Agents to Build:**
    - GoogleAnalyticsAgent: query_ga4(metrics, dimensions, date_ranges)
    - GoogleSearchConsoleAgent: get_seo_performance(...)
    - GoogleTagManagerAgent: list_tags(...), get_container_status(...)
    - GoogleBusinessProfileAgent: get_local_reviews(...)

    - PageSpeedInsightsAgent: run_performance_audit(url)

Phase 2: The Business Operations Pack

- **Goal:** Bridge the gap between marketing efforts and business results.
- **Agents to Build:**
    - **Microsoft Pack:** PowerBIAgent, Dynamics365Agent.
    - **CRM Pack: SalesforceAgent (highest priority)**, HubSpotAgent.

Phase 3: The Automation Pack

- **Goal:** Enable true workflow automation by introducing Action Agents.
- **Agents to Build:**
    - WebNavigatorAgent (Puppeteer): browse_to_url(url), click_element(selector), fill_form(data).
    - WebContentScraperAgent (Firecrawl): crawl_site(url).
    - CodeRepositoryAgent (Git/GitHub): clone_repo(url), list_recent_commits().

6. Key Deliverables

- A functional AI Orchestrator service within the FastAPI backend.
- A well-defined class structure and API for creating and registering new Agents.
- A complete and robust implementation of all Phase 1 "Google Pack" Data Agents.
- The ability for the platform to handle a multi-step query that requires calling multiple agents in sequence to formulate a single answer.