



SENG 491 – SENIOR PROJECT I

Stratify: AI-Powered Multi-Agent Business Planning Platform

High-Level Design Report

Team Members:

- Shahdad Peykani – 99111760026 – Software Engineering
- Muhammad Ozar Mirza – 99687965054 – Software Engineering
- Abdirahman Tahlil Farah Hussein – 99717977422 – Software Engineering

Supervisor: Prof. Ayşenur Birtürk

Jury Members:

- Prof. Eren Ulu
- Prof. Elif Kurtaran Özbudak

1. Introduction.....	3
1.1 Design Goals	3
2. Current Software Architecture.....	3
3. Proposed Software Architecture.....	3
3.1 Overview	4
3.2 Subsystem Decomposition	4
3.3 Hardware and Software Mapping.....	5
3.4 Persistent Data Management	6
3.5 Access Control and Security	6
3.6 Global Software Control.....	7
3.7 Boundary Conditions.....	8
4. Subsystem Services	8
4.1. Pipeline Orchestrator Subsystem Services	8
4.2. User Interface Subsystem Services.....	8
4.3. Idea Agent Subsystem Services (Mode A Only).....	9
4.4. Market & Risk Agent Subsystem Services.....	9
4.5. Financial Agent Subsystem Services.....	9
4.6. Web Generation Agent Subsystem Services	9
4.7. PDF Compiler Subsystem Services.....	10
5. Object and Class Model.....	11
6. Conclusion	11
7. References	12

1. Introduction

The purpose of this High-Level Design (HLD) document is to convey the architectural framework and key design choices for the Stratify system. Stratify addresses the high failure rate of early-stage startups caused by subjective validation and knowledge gaps. It operates as a unified "Business-in-a-Box" pipeline that transforms a user profile or a raw idea into a validated business plan. The platform uses Retrieval-Augmented Generation (RAG) and a multi-agent architecture to automate areas like market research, financial forecasting, and initial web presence creation.

1.1 Design Goals

The primary goals of the Stratify architecture are:

- Enforce discipline and data-driven rigor from initial input to final execution.
- Eliminate subjective bias in early-stage validation using objectively retrieved market data.
- Consolidate fragmented business planning tooling into a single, cohesive multi-agent pipeline.
- Provide actionable, deployable assets (Feasibility Report and Starter Website) to lower execution barriers.

2. Current Software Architecture

Entrepreneurs and students face critical challenges during the venture planning phase. Founders with existing ideas often lack objective data to verify market need, leading to bias-driven decisions. Furthermore, individuals frequently lack the structured knowledge necessary to conduct formal risk assessments, financial modeling, or compliance checks. Current tools are highly fragmented, lacking a unified system to integrate market research, financial forecasting, and technical execution. Non-technical founders also struggle to bridge the gap between planning and establishing an initial online presence.

3. Proposed Software Architecture

This part provides the top-level architecture proposal for the Stratify system. The structure emphasizes software engineering discipline by breaking down responsibilities into modules with a clear separation of concerns.

3.1 Overview

The system is managed by a central Orchestrator that handles two distinct operational scenarios. Mode A (Ideation) generates personalized startup concepts based on user profiles (skills, budget, risk tolerance). Mode B (Validation) bypasses idea generation to directly validate a user-supplied business description. The system then utilizes a team of AI agents to process the concept through a Retrieval-Augmented Generation (RAG) pipeline.

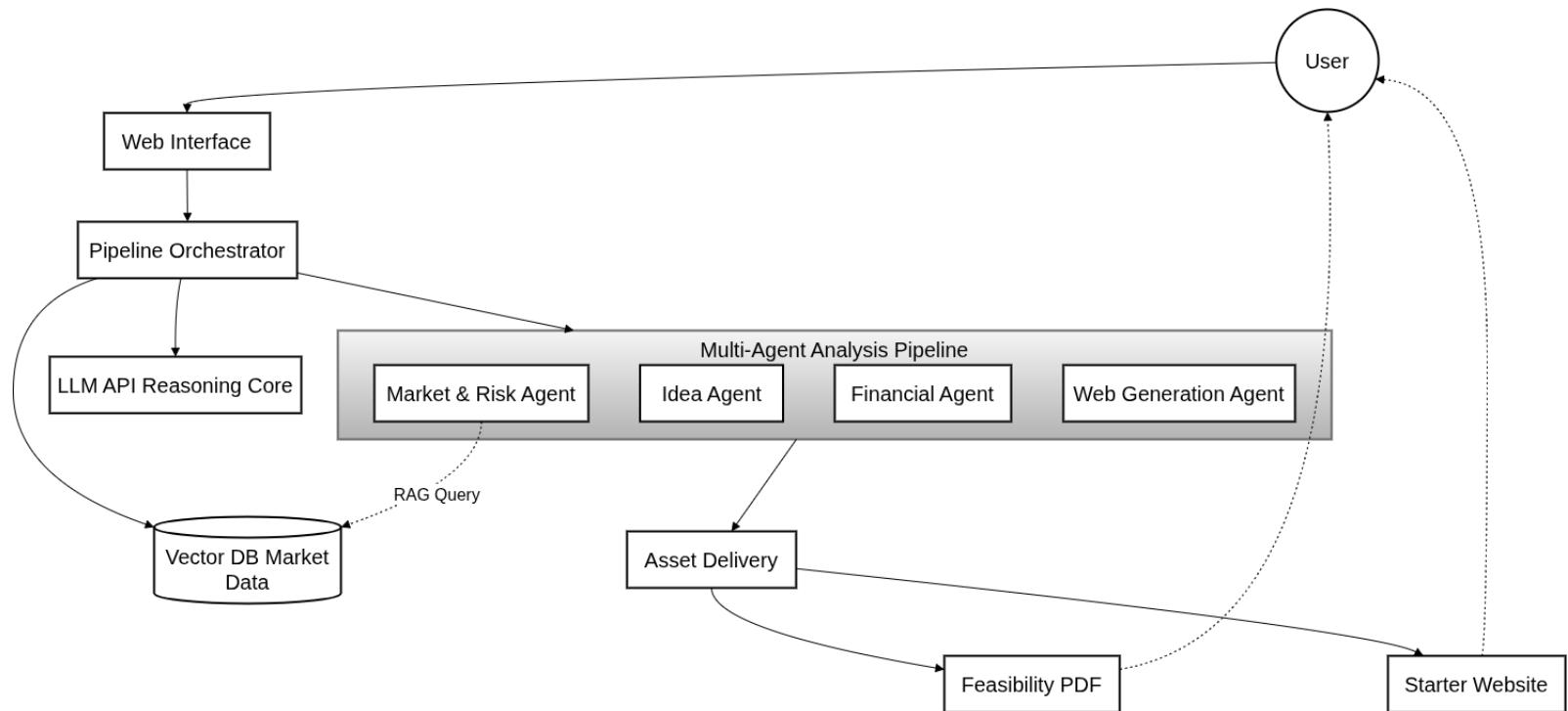


Figure 1 - High-level system architecture

3.2 Subsystem Decomposition

This part provides the top-level architecture proposal for the Stratify system. The structure emphasizes software engineering discipline by breaking down responsibilities into modules with a clear separation of concerns.

- **User Interface Subsystem:** Handles all user interactions and presents the generated elements for example the main dashboard.
- **Pipeline Orchestrator:** Controls execution flow, state transitions, and error recovery. It ensures a strict execution order and checks that each subsystem has run successfully before the next one starts.

- **Idea Agent Subsystem:** Operates in Mode A to generate at least three personalized, market-backed startup concepts based on user profiling questions like skills, budget, and risk tolerance.
- **Market & Risk Subsystem:** Uses a RAG pipeline to fetch real-time industry data and competitor landscapes. It generates SWOT analyses and identifies at least three potential operational or market risks.
- **Financial Subsystem:** Performs numeric modeling to estimate initial Capital Expenditure (CAPEX) and calculate theoretical break-even points.
- **Web Generation Subsystem:** Produces valid, responsive HTML/CSS code for a tailored, static single-page landing page for validation.

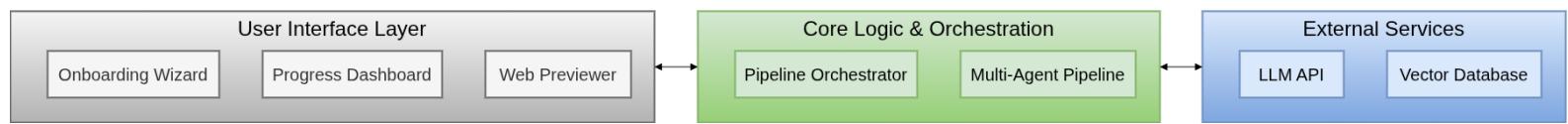


Figure 2 - Subsystem Decomposition

3.3 Hardware and Software Mapping

The system is designed as a web-based application utilizing standard cloud computing resources.

- **Frontend:** A modern web framework (e.g., React or Vue.js) accessible via standard browsers.
- **Backend:** A Python-based server (e.g., FastAPI or Flask) responsible for agent orchestration.
- **External APIs:** Generative tasks are handled by an LLM API (e.g., OpenAI or Gemini), while contextual retrieval utilizes a Vector Database (e.g., Pinecone or ChromaDB).

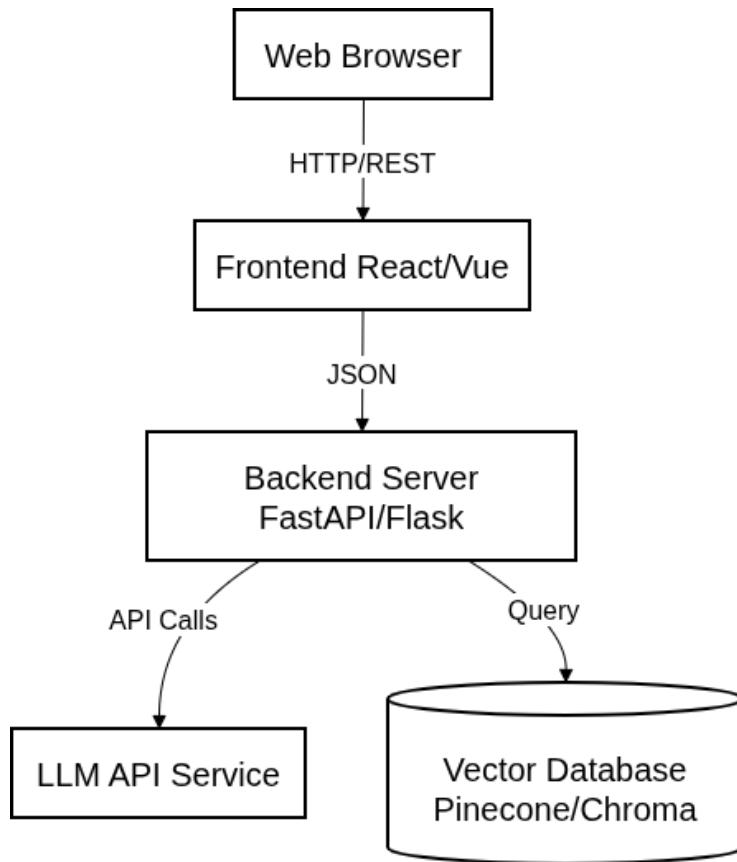


Figure 3 - Hardware/Software Mapping

3.4 Persistent Data Management

Data persistence is focused on the knowledge base required for the RAG pipeline. The Vector Database permanently stores industry data, competitor landscapes, and regulatory guidelines used for retrieval. User inputs, intermediate agent context, and generated artifacts (PDF reports and HTML files) are managed momentarily by the Orchestrator to compile the final "Business-in-a-Box" output.

3.5 Access Control and Security

Stratify incorporates strict security and ethical guidelines:

- **Data Privacy:** User business ideas are treated as intellectual property and must not be used to train public models without consent. User data must be encrypted in transit.
- **Credential Management:** API Keys for LLM services are securely stored in environment variables, never in frontend code.
- **Ethical Guardrails:** The system includes content filtering to prevent the generation of illegal or harmful business models. Furthermore, mandatory disclaimers emphasize that

outputs are for educational/planning reference, subject to AI hallucination, and do not constitute certified financial advice.

3.6 Global Software Control

System control is centralized within the Orchestrator, which acts as a Mode Manager. Depending on the input stimulus (Profile for Mode A, or Idea for Mode B), the Orchestrator triggers the appropriate agent pipeline execution path. The execution is strictly linear following idea selection, ensuring Market, Financial, and Web generation phases occur synchronously or in controlled parallel execution.

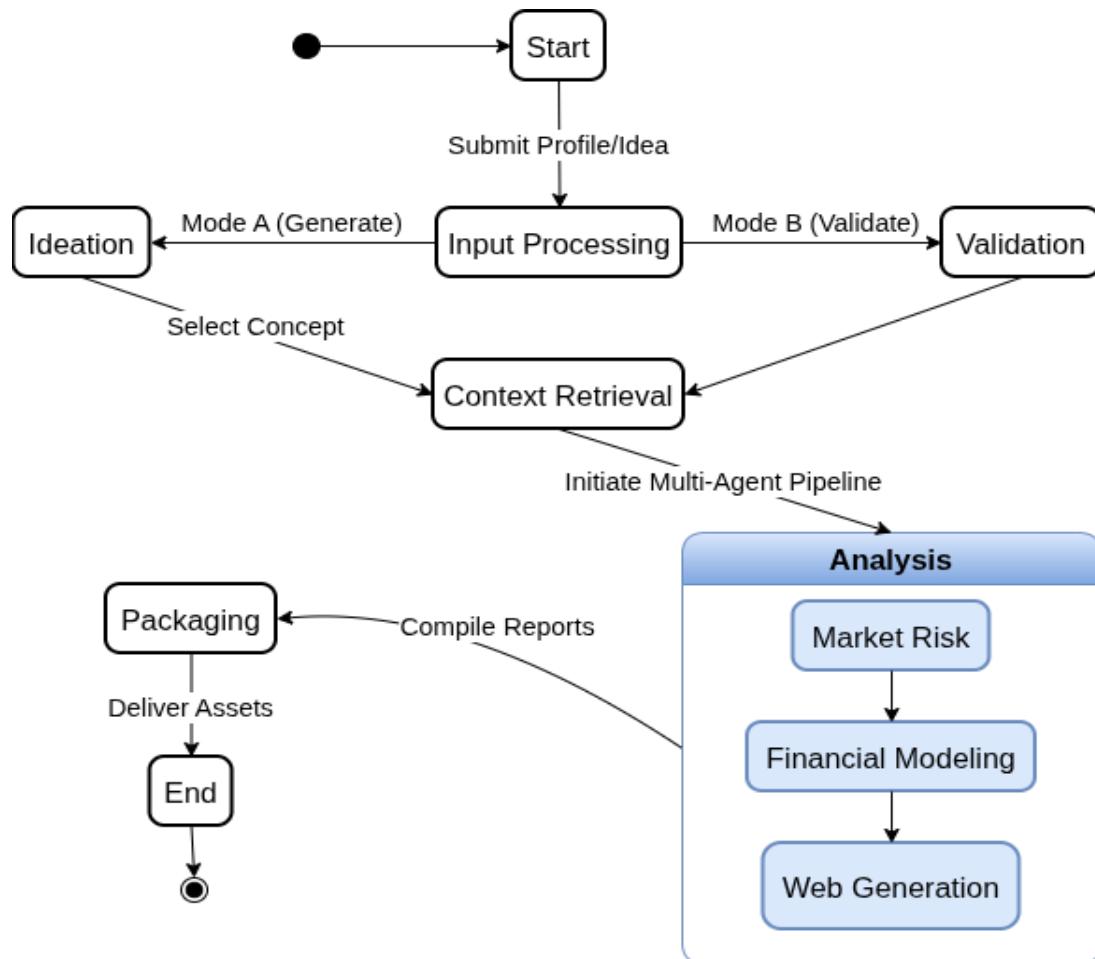


Figure 4 - State Diagram

3.7 Boundary Conditions

- **Initialization:** The system provides a wizard to capture user constraints or raw ideas, establishing the context required for agent execution.
- **Normal Operation:** Agents communicate sequentially, passing contextual data (e.g., Market Agent sending risk data to the Financial Agent) to build the final deliverables.
- **Termination:** Upon successful execution, the PDF Report Compiler and Web Generation Agent package the results, providing the user with direct download links and an interactive web preview.

4. Subsystem Services

This section describes the specific services and micro-operations provided by each major subsystem within the Stratify architecture. By delegating complex business planning tasks to specialized agent subsystems, the platform maintains strict modularity and high execution reliability.

4.1. Pipeline Orchestrator Subsystem Services

The Pipeline Orchestrator serves as the central system of Stratify, managing data flow and agent lifecycles.

- **Dual-Mode Routing:** Determines whether the pipeline follows the generative Mode A or the validation-focused Mode B based on initial input.
- **State and Context Management:** Maintains the overarching UserContext and passes sanitized data payloads sequentially between analyzing agents.
- **Parallel Execution Control:** Manages the synchronous execution of the analysis agents and the parallel execution of the asset generation (PDF and Web) phases.
- **Error Handling:** Monitors external API timeouts (LLM or Vector DB) and manages internal retries or fallback mechanisms.

4.2. User Interface Subsystem Services

The UI acts as the sole interaction layer, abstracting the complexity of the multi-agent pipeline from the entrepreneur.

- **Structured Onboarding:** Captures constrained parameters (skills, budget, risk tolerance) using a step-by-step wizard for Mode A.
- **Unstructured Idea Parsing:** Accepts and cleans long-form business idea descriptions for Mode B.
- **Execution Monitoring:** Displays real-time progress indicators as individual background agents complete their specific tasks.

- **Interactive Code Preview:** Renders the generated HTML/CSS landing page in a live frame, permitting the user to review and edit text directly within the browser.

4.3. Idea Agent Subsystem Services (Mode A Only)

- **Constraint-Based Synthesis:** Analyzes user skills and budget limits to map out realistic market opportunities.
- **Concept Generation:** Produces a minimum of three distinct and viable startup concepts for the user to evaluate and select.

4.4. Market & Risk Agent Subsystem Services

The subsystem is responsible for applying grounding techniques to enhance the LLM's reasoning in factual data.

- **RAG Context Retrieval:** Formulates queries and fetches relevant industry documents, competitor landscapes, and trend data from the Vector Database.
- **SWOT Generation:** Synthesizes the retrieved context into a formal Strengths, Weaknesses, Opportunities, and Threats framework.
- **Hazard Identification:** Flags at least three specific venture risks, categorizing them into operational, regulatory, or market saturation threats.

4.5. Financial Agent Subsystem Services

This subsystem executes the numeric modeling required to project venture feasibility.

- **Capital Expenditure (CAPEX) Estimation:** Calculates the initial startup costs required to launch, adjusted to the specific business model and user budget constraints.
- **Operational Expenditure (OPEX) Modeling:** Estimates recurring monthly expenses.
- **Break-Even Calculation:** Computes the theoretical break-even timeline based on projected unit economics and revenue streams.

4.6. Web Generation Agent Subsystem Services

- **Targeted Copywriting:** Incorporates persuasive hero content and context-aware value propositions that are specific to the approved concept.
- **Code Synthesis:** Generates valid, responsive HTML and CSS structural code.
- **Lead-Capture Integration:** Implements a "Coming Soon" email capture form into the static landing page to allow immediate user validation testing.

4.7. PDF Compiler Subsystem Services

- **Data Aggregation:** Collects the contrasting outputs from the Market, Risk, and Financial agents.

- **Document Formatting:** Structures the aggregated data into a formal, professional layout (Executive Summary, Strategy, Financials, Risk).
- **Artifact Export:** Generates the final, downloadable PDF Feasibility Report.

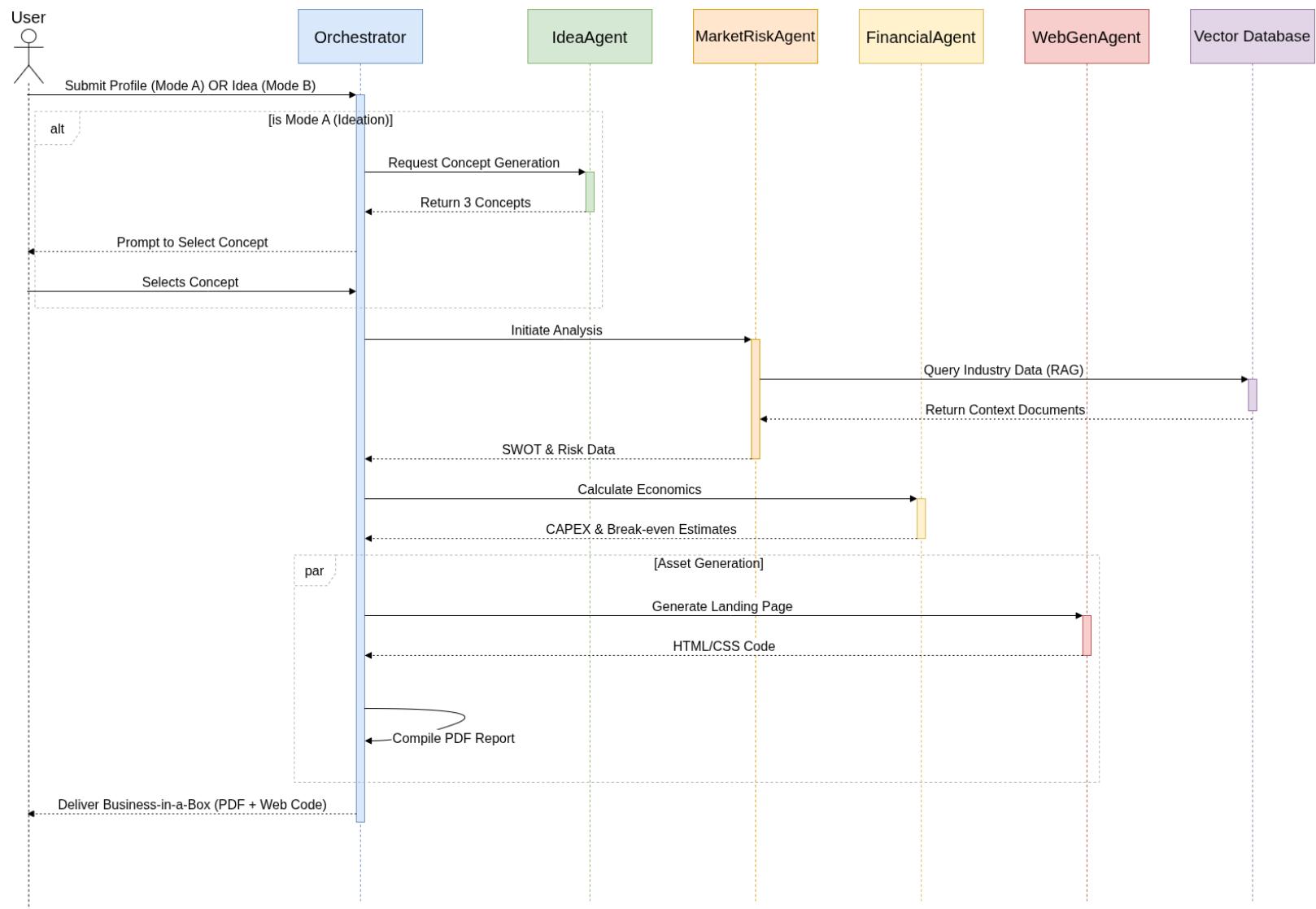


Figure 5 - Sequence Diagram

5. Object and Class Model

The core backend structure is built around an inheritance model for AI agents. The central *Orchestrator* manages a standard *UserContext* and determines the active pipeline route. The *AbstractAgent* base class provides a standard *execute(context)* method, which is implemented uniquely by the *IdeaAgent*, *MarketRiskAgent*, *FinancialAgent*, and *WebGenerationAgent*. The *Orchestrator* ultimately triggers the *PDFCompiler* to export the finalized report.

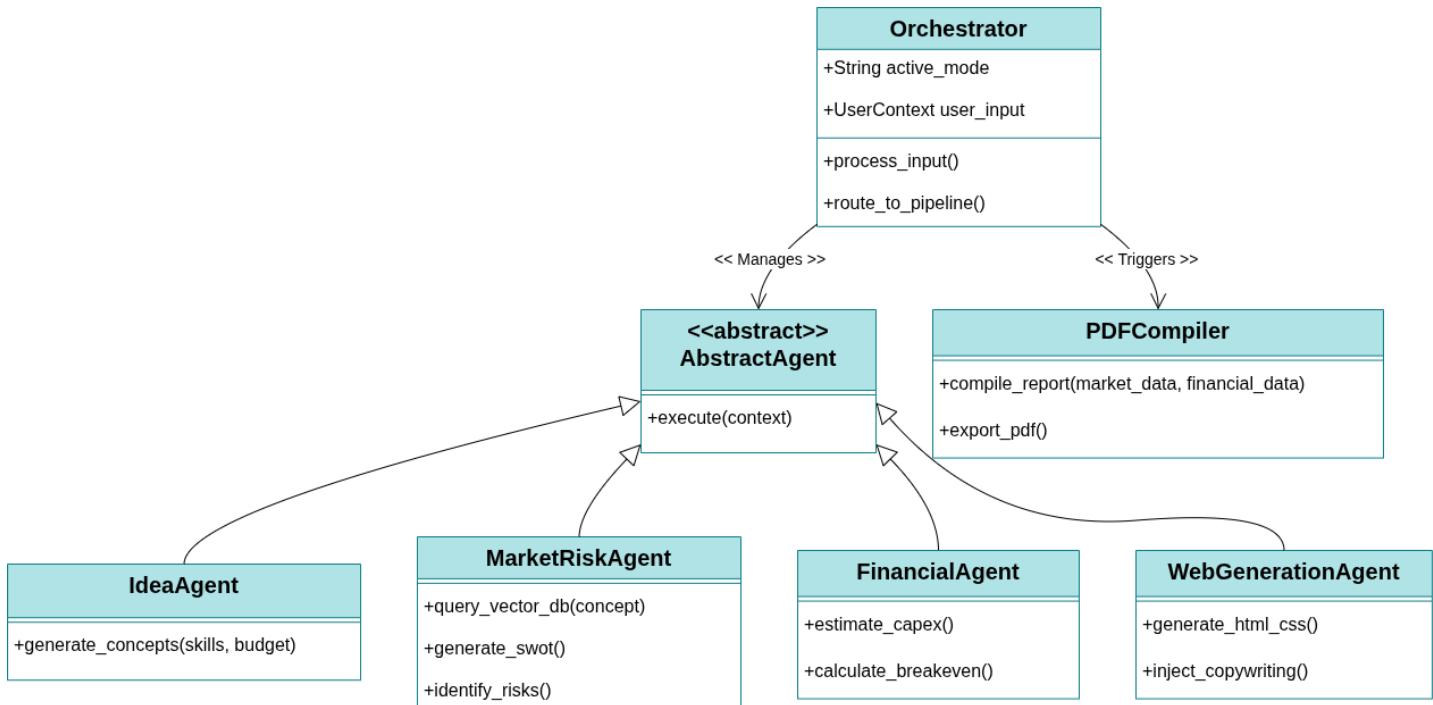


Figure 6 - Class Diagram

6. Conclusion

The High-level Design report for Stratify addresses an intelligent, multi-agent framework supporting business development from early-stages . By combining structured user onboarding modes with advanced RAG-driven market analysis, Stratify successfully reduces subjective bias and fragmentation. The modular agent pipeline ensures accurate financial forecasting and immediate technical execution capabilities, delivering a comprehensive "Business-in-a-Box" securely and efficiently.

7. References

- [1] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in Proc. NeurIPS, 2020.
- [2] IEEE, "IEEE Code of Ethics," IEEE Policies, Section 7.8, 2024.
- [3] A. Osterwalder and Y. Pigneur, Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers. Hoboken, NJ: John Wiley & Sons, 2010.
- [4] T. I. Tiangolo, "FastAPI High performance, easy to learn, fast to code, ready for production," FastAPI Documentation, 2024.
- [5] Meta Open Source, "React: The Library for Web and Native User Interfaces," 2024.
- [6] JGraph Ltd., "draw.io," 2026. [Online]. Available: <https://app.diagrams.net/>.