AI Financial Analyst - High Level Plan

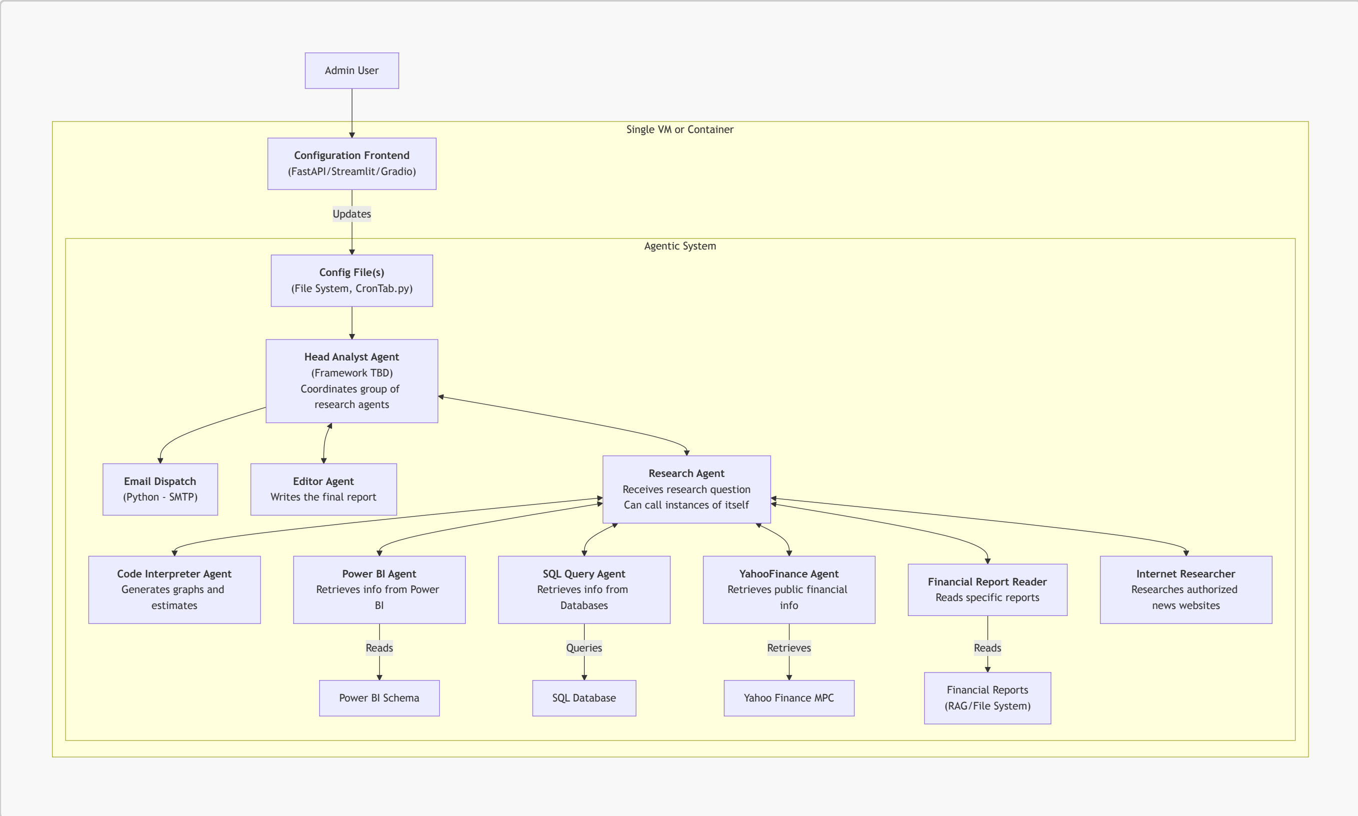
Proposed Report Structure

The following is the report structure I have pictured so far. I personally prefer relatively simple structures, but please let me know if you would prefer to break it down further. My notes on the content assume we will be doing a single report, but the structure can also be used if the target is one report per KPI as well.

1. **Executive Summary**
    - Key findings
    - Current value of KPIs and comparison to last period
    - No chars or tables
2. **Overview**
    - Quick snapshot of the value of the KPIs for the current period
    - Comparison with the previous period
    - Comparison vs. target/benchmark if we have one
    - Key insights or highlights about performance
3. **Trends and Context**
    - Short-term trend chart or table (last 3–5 periods) for each KPI
    - High-level explanation of what drives changes in each KPI (e.g., volume, price, cost impact, broader market events)
    - Relevant operational metrics directly influencing the KPI
    - Highlight any negative trends, changes in trends, or outliers (”special cases”)
4. **In depth analysis**
    - For each KPI with a "special case" (this section could be excluded if there isn’t any):
        - An explanation of the main causes or business events impacting the KPI
        - Any relevant operational or financial events linked to the variance
5. **Forward Outlook and Recommendations**
    - Forecast or outlook for the KPI based on current data
    - For each KPI with special cases:
        - Suggested actions or focus areas to remedy / maintain
        - Forecast for the KPI if actions are implemented

High Level Architecture

The following is the high level architecture I have imagined so far.



Proposed Implementation Process

I propose an incremental plan for developing this system, so we can ensure we get to a working MVP first, even if it does not include all the potential functionalities, and add on to it as time allows after.

You will notice I am proposing to start with retrieving data only from a PowerBI file, I propose this for two reasons: it looks like the toughest problem in terms of agent access and it seems like the most important to A&M, so for both reasons I would prefer to handle it early.

1. Define core stack:
    - Inference provider
    - Agent framework(s)
    - Structure of data provision (e.g., how are we going to retrieve the PowerBI)
    - Agent inputs (e.g., is the list of KPIs in prompt or set as configuration)
2. Create a basic agent system that can achieve the following process:
    1. Take information from a PowerBI
    2. Create a simplified report (Executive Summary, Overview and Recommendations)
    3. Self-review the report and ensure it has the required information
    4. Email to a set address
3. Create configuration frontend:
    - Set up the system to run autonomously
    - Update configuration values set so far (e.g., the list of KPIs or the URL to the PowerBI)
4. Add basic forecasting (CAGR or similar) to the report
5. Add “special case” detection

    1. Identify special cases (changes in trend, outliers, or negative trends)
    2. Define research plan with available tools, including exploring operational information if available
    3. Implement research plan
    4. Add sections to the report
6. **Up to here we will have the MVP; then we can continue adding additional sources of information, which I would suggest we add in the following order:**
    1. Add database retrieval
    2. Add document retrieval (define if using RAG or another agent)
    3. Add access to Yahoo Finance
    4. Add internet access to the internet in general (defining a list of approved sites)
    5. Potentially: Add more advanced forecasting

Agent Development Process

When development each agent, I would like to approach in line with "Evaluation Driven Design" (similar to TDD); in summary:

1. Define the expected behaviour
2. Define a set of scenarios to evaluate
3. Define the dataset available for those scenarios
    - We can consider creating custom datasets that match important scenarios if the data is not available, but this could add a lot of time to the process
4. Define the type of evaluation (e.g., LLM as judge, a defined set of rules, or just manual revision) for each scenario
5. Write evaluation code
6. Write the actual agent / new behaviour

This would make the development process a bit slower and maybe require a bit more data, but I believe give confidence to experiment with different prompts / models.

Questions / Asks for A&M Team

- Report Structure:
    - Are we writing a single report for a set of KPIs, or a report per KPI?
    - Are the KPIs a given (and thus can they be included in the prompt) or should they be configured by the user?
- Technology Stack:
    - KEY ASK: Do you have preferred inference provider (e.g., OpenAI or Azure)? If so, I do need access to models.
        - I am asking about the inference provider because that will define the list of models available (e.g., if its OpenAI I can only use their models). If you don't, we might want to look at Azure as I've seen Microsoft Fabric has a PowerBI agent which be helpful.
    - Do you have a preferred Agent Framework to use in this project (e.g., LangGraph or OpenAI's SDK)?
    - Low priority for now but can help define which stack to use for the frontend:
        - Are there any specific needs for authentication (e.g., a specific OIDC provider)?
        - Does the frontend need to be accessible as a public website?

- PowerBI:
  - I understood we will be using a defined schema you have already designed, if so, can you please share?
  - Do you have a preferred way of connecting to the PowerBI?
    - To clarify what I mean, I could create a .pbi file in the same folder as the agent, but I imagine the preference would be to provide a URL, but then we might need to deal with Microsoft Authentication.
  - Ideally, at least one sample PowerBI file.
- Database:
  - When you refer to a Database, I am not 100% sure if that is the same Database the PowerBI file takes their information from, or if its a different Database.
    - If it's a different Database, do you have a set schema? And a Database type (Postgres, MySQL, etc.)
- Financial Reports
  - How would the financial reports be provided (e.g., URL, as files to be uploaded to the agent, etc.)