VPBank Technology Hackathon 2025

General Brief

Please fill up this table and use this document as a template to write your proposal.

Challenge Statement	#16 AI Agents for the CRM System
Team Name	Team 22 - **Agentify**

Team Members

Full Name	Role	Email Address	School Name	Faculty/ Area of Study	LinkedIn Profile URL
Phan Thanh Dieu Linh	Leader	phanthanhdieulinh.lucy@gmail.com	Foreign Trade University	Finance and Banking	https://www.linkedin.com/in/phanthanhdieulinh/
Pham Viet Hoang	Member	viethoangpham.contact@gmail.com	Hanoi University of Science and Technology	Computer Science	https://www.linkedin.com/in/viethoangpham14/
Nguyen Quang Vinh	Member	vinhnguyen21101109@gmail.com	University of Engineering and Technology - Vietnam National University	Information Technology	https://www.linkedin.com/in/vinh-n-32b4491a2/
Nguyen Ngoc Tan	Member	liam.nguyen306@gmail.com	Hanoi University of Science and Technology	Information Technology	https://www.linkedin.com/in/liam-nguyen-41215b331/

CONTENT OUTLINE

PROBLEMS

A Strained, Inefficient Model

1. Target Audience

Relationship Management Specialists (RMs), an audience that is critical to a bank's success but is increasingly burdened by low-value work.

2. The Problems

2.1. The Lack of Personalization

Problem: Banks and their RMs are failing to meet client expectations for true personalization, leading to a breakdown in trust and client retention.

Supporting Data:

- **Clients Demand Personalization:** The standard for "good" service is now set by tech giants, not other banks. **72% of customers rate personalization as "highly important"** in financial services, and 62% agree that personalized recommendations are superior to generic ones (Zendesk, 2025).
- **Generic Advice Is the Norm:** Despite this demand, banks are failing. A 2024 study found that **nearly 90% of High-Net-Worth Investors (HNWI) feel the advice they receive from their advisor is "too generic"** (swissQuant, 2024).
- **Lack of Personalization Erodes Trust:** This failure to personalize is not just a missed opportunity; it's a threat. **69% of customers state that personalization is "important" for building trust** with their financial institution (Smart Communications, 2024). **Agentify**'s solution's "brain" is designed to solve this by learning an RM's authentic voice, moving beyond generic templates to rebuild trust.

2.2. The Reliance on “Gut Feel”

Problem: RMs lack the tools to effectively prioritize their portfolio. They rely on "gut feel" to decide who to call, which is highly inefficient and leads to missed opportunities with high-potential clients.

Supporting Data:

- **Inefficient Targeting Wastes Time:** This lack of data-driven prioritization is a direct drain on the RM's most valuable resource: their time.
- **AI Prioritization Drives Measurable Returns:** Firms that use AI and predictive analytics to solve this exact problem see significant gains. On average, wealth firms using these techniques saw a **23% increase in ROI** (CoinLaw, 2025).
- **Better Prioritization Equals Higher Retention:** This feature directly impacts the bottom line. Custom AI models designed for client prioritization and personalization have enabled firms to **achieve 27% higher client retention rates** (CoinLaw, 2025).

2.3. The Lack of Real-time Context

Problem: When an RM contacts a client, they often lack the complete, real-time context to make a relevant, valuable recommendation. This results in the "generic advice" (swissQuant, 2024) that clients ignore and represents a massive, missed cross-selling opportunity.

Supporting Data:

- **Massive Upside from Contextual Offers:** The value of solving this is enormous. One "Next Best Action" (NBA) case study in banking found that customers who received relevant, data-triggered offers showed a **100% increase in second mortgages** and a **93% increase in auto loans** (WWT, 2021).
- **High, Fast ROI:** The business case for this feature is exceptionally strong. An analysis of NBA solutions found that banks could achieve a **162% ROI over five years** and see a payback on their investment in **less than six months** (Latinia, 2025). **Agentify** delivers this value on demand.

2.4. The Uncomfortable Navigation within Complexity

Problem: RMs must navigate multiple, complex, and slow-moving internal systems to get simple answers about their performance, which is a key part of the administrative burden that pulls them away from clients.

Supporting Data:

- **"Tool Sprawl" Kills Productivity:** RMs are forced to be "swivel-chair" operators. A typical RM uses an **average of five tools or applications for each key activity** (Accenture, 2022). **Agentify**'s ability to consolidate this data into a simple chat interface directly attacks this inefficiency.
- **AI Co-pilots Excel at This:** This is a prime use case for generative AI, which excels at synthesizing data from multiple sources and "running custom financial analyses" to provide instant, natural-language answers (Bain & Company, 2024).

2.5. The Time-consuming Manual Input

Problem: Manual CRM data entry is the single biggest bottleneck to a bank's entire data strategy. RMs hate it, so they do it poorly, late, or not at all. This creates "bad data" that poisons all other reporting, prioritization, and AI initiatives.

Supporting Data:

- **"Bad Data" is Pervasive:** This is a universal problem. Organizations estimate that **about one-third (33%) of their customer and prospect data is inaccurate** (Experian, 2024).
- **The High Cost of "Bad Data":** This isn't just a minor issue. "Inaccurate financial and operational data" can lead to **"disastrous business strategies"** and "poor lending decisions" (NetSuite, 2025).
- **The #1 Blocker to AI: Agentify** solves the foundational problem that cripples most banks' tech strategies. **35% of CFOs cite "poor data quality" as the key inhibitor** for adopting high-value AI (Gartner, cited in NetSuite, 2025). **Agentify**'s "painless" feature fixes the data-input problem at the source.

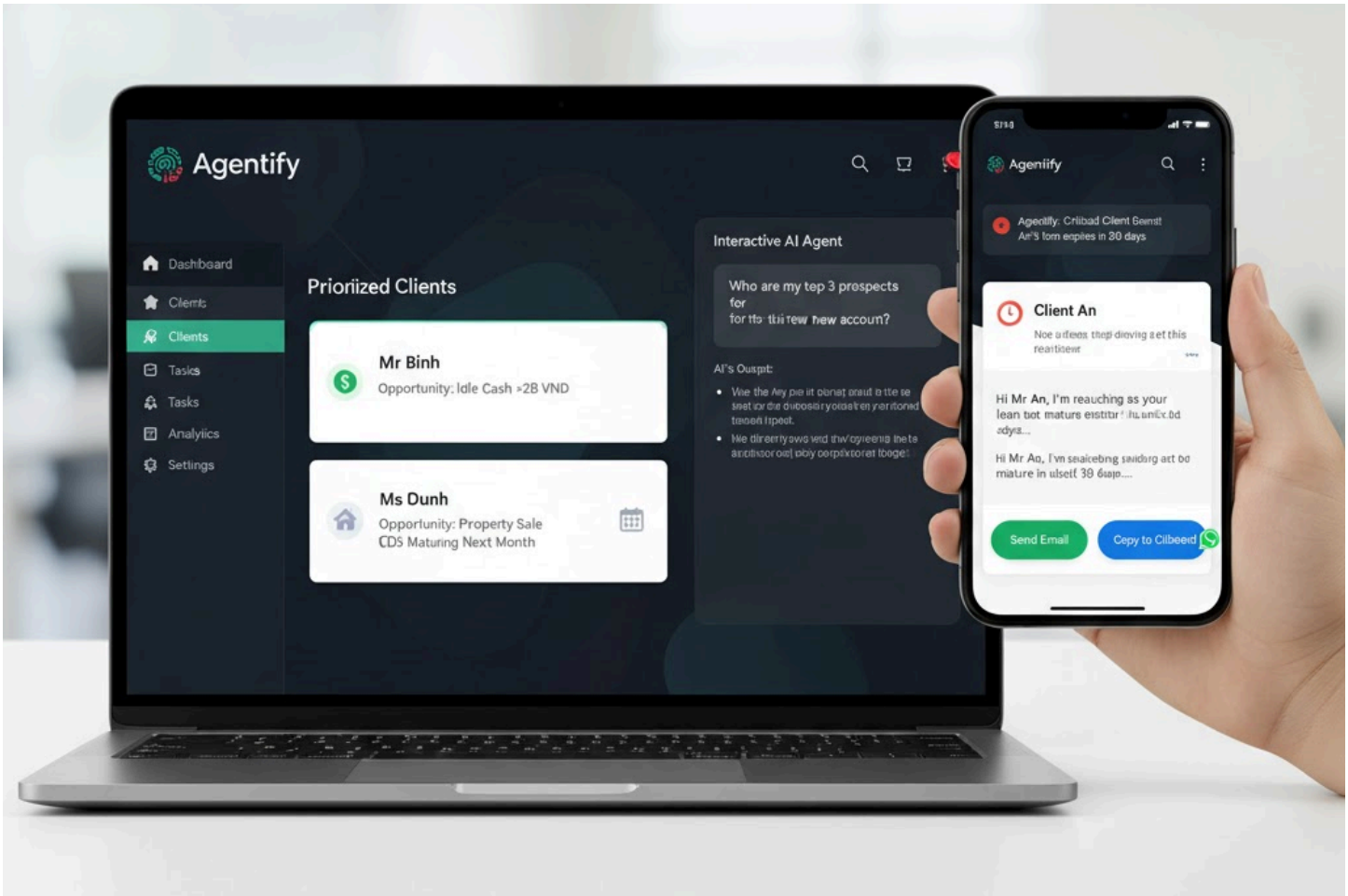
SOLUTION INTRODUCTION

Agentify - AI Co-pilot for Relationship Management

Our solution is an intelligent, multi-layered platform designed to act as an AI Co-pilot for Relationship Managers (RMs). It integrates deeply with the CRM system to automate administrative tasks, provide strategic insights, and ensure timely, highly personalized client communication.

It works by combining two core features with **a sophisticated personalized engine** underneath:

- A **Proactive Agent** that notifíe RMs of critical client events and write personalized ready-to-send email;
- **Interactive AI Agent Co-pilot** to serve as an on-demand strategic advisor & a helpful assistant for RM.



1. The Foundation: The Personalization Engine

This is the **"brain"** of our solution **core feature 01 and many other future features**, responsible for the quality and authenticity of every AI-generated message. It ensures that every communication sounds like it came directly from the RM, not a robot. It operates in two phases:This is the **one-time setup process** that makes all other features "magical". We offer two simple paths for this:

- **Passive Learning (The Effortless Method):** During the initial setup, the RM grants one-time permission for the AI to scan their third party's social platform (**E.g.** Outlook, Facebook, Zalo, etc.) from the past 6-12 months. The AI then analyzes thousands of real-world messages to passively learn their unique tone, formality, common phrases, and sign-offs.

  E.g. For Outlook, the AI analyzes these real-world emails to automatically learn:

  - **Salutations:** Do they use "Hi [Name]," "Dear Mr. Smith," or "Hi Anh/Chi"?
  - **Formality:** Is their tone corporate and formal or casual and friendly?
  - **Signature Phrases:** What are their common sign-offs (e.g., "Best regards," "Trân trọng," etc.)?
  - **Structure:** Do they write short, direct sentences or more descriptive ones?
- **Active Prompting (Advanced Control):** For power users, RMs can go to Settings to provide direct, rules-based prompts. The prompting can be made with a simple message, no need to click many buttons. We will provide guidance for the RMs to create the most suitable prompt to their means, which may include:
  - **Global Tone (Default):** The RM can set a single master prompt for all communications (e.g., "My default style is friendly, professional, and approachable."). This is supplemented by hidden system prompts that enforce bank policies.
  - **Segment Tone:** RMs can set specific rules for client segments (e.g., "For all 'Diamond' clients, use a warmer, more personal, and appreciative tone.").
  - **Per-Client Tone:** RMs can set rules for specific high-value individuals.

This foundation step is easy to work on, taking up **only around 05 minutes once**. However, if the RM skips this step, the core feature can still function properly with the default specialized prompt that was created by us.

2. Core Feature: The Proactive Catch-up Agent

This feature acts as the RM's automated lookout, ensuring no opportunity or client event is missed.

- **How it Works:** The platform constantly monitors the bank's CRM and core banking data for critical, time-sensitive client events.
- **Triggers:** These include birthdays, loan maturity dates, Certificate of Deposit (CD) expiry, significant new deposits, or "inactive" clients who haven't been contacted in xx days.
- **The Workflow:**
    1. An event is triggered (e.g., "Client An's loan expires in 30 days")
    2. The RM receives a single push notification.
    3. The RM clicks on the notification. The notification then opens the app to a prioritized list of clients who need attention (auto-sorted and filtered by AI).
    4. The RM selects a client instantly and reveals a pre-drafted message (Email or/and Zalo/WhatsApp suggestion). This draft is already personalized using the RM's tone (using the **Personalization Engine**) and the client's specific event context.
    5. The RM can review, edit, or approve the message. With one click, they can **"Send Email"** directly from the platform. For chat apps (Zalo, WhatsApp, etc.), we provide a **"Copy to Clipboard"** function, allowing the RM to paste the message into the native app, which maintains compliance and user control.

3. Core Feature: The Interactive AI Agent

This is an interactive, on-demand **chat-based assistant** or even “Chief of Staff” within the app, allowing the RM to use a simple message to **get strategic advice, query data, and perform repetitive administrative tasks** instantly. 04 main use cases are covered within VPBank Hackathon Hack2Hire.

Others are listed as the future development and vision to make RM’s life simpler and bring more benefits to the customers.

3.1. Use Case 1: Strategic Prioritization

- RMs can ask the AI to analyze a specific client pool (e.g., their entire portfolio, a specific segment) against a particular goal or banking document (like a new product offering).

- The AI will cross-reference the client data with the goal and return a prioritized shortlist of the highest-potential clients, along with a brief explanation for *why* each is a top opportunity.

Example:

- **RM's Input:** The RM asks **Agentify**, "Analyze my 'Diamond' clients. Who are my top 3 prospects for this new account?"
- **AI's Output:**
    1. **Mr. Binh:** His primary savings account balance has exceeded 2B VND for over 90 days, earning minimal interest. He is a perfect candidate to move this idle cash.
    2. **Ms. Dung:** Just received a large inbound transfer (based on transaction alerts) from a property sale. She will likely need a safe, high-yield place to park this cash short-term.
    3. **Ms. Oanh:** Has multiple Certificates of Deposit (CDs) maturing next month totaling 1.5B VND. This new account offers better liquidity and a competitive rate.

3.2. Use Case 2: Next Best Action (Nurture)

- An RM can select a single client and ask the AI for the "next best action."
- The AI will analyze that specific client's entire profile, recent interactions, product holdings, and upcoming milestones to recommend the most logical and valuable next engagement step, helping the RM nurture the relationship proactively.

Example:

- **RM's Input:** The RM asks **Agentify**, "Client: Mr. Son Pham. 0823912039. What's my next best action?"
- **AI's Output:**
  Based on CRM information, the client has two critical, time-sensitive events.
  Recommended Action: Prioritize the upcoming CD maturity.
  Reason: The 12-day deadline is urgent.
  Suggested Message (to copy):
  "Hi Mr. Son,
  I'm reaching out as I see Agentify's 1.5B VND Certificate of Deposit is set to mature in about 12 days. I'd love to discuss Agentify's options, whether that's rolling it over or looking at some new, competitive rates we have. We also have Agentify's home loan renewal coming up in a few months, so it might be a

good time for a full portfolio review.
Are you free for a brief call early next week?
Best,
[RM Name]"

3.3. Use Case 3: Instant Performance Reporting

- RMs can use natural language to ask questions about their own performance or portfolio data.
- The AI queries integrated systems and dashboards to provide immediate, specific answers (e.g., "What's my total deposit value this month?" or "How many clients haven't I contacted?"), saving the RM from navigating complex reporting tools.

Example:

- **RM's Input:** What's my performance? How am I doing in plain English?
- **AI's Output:**
  Here's a simple summary of Agentify's Q3 performance based on that data:
  Overall Portfolio: You are **On Target**. You are at 450B VND/ 420B VND goal. (Excellent!)
  New Loans: You are **Exceeding Target**. You brought in a 25B VND/ 20B VND goal.
  New Deposits: You are **Below Target**. You brought in a 15B VND/ 20B VND goal.
  Client Activity: You are **Below Target** on client contacts (120/ 150).

3.4. Use Case 4: Painless CRM Data Entry

- After a client interaction, the RM can simply type or dictate an unstructured summary of what happened.
- The AI understands this natural language, automatically identifies the correct client, and translates the summary into structured data, creating the official meeting log and scheduling any follow-up tasks in the CRM without manual data entry.

Example:

- **RM's Input:** Record this information on the CRM. Met Mr. Son Pham 0823912039 at a coffee house, talked about a renovation loan for his apartment, which needs about 800m vnd. He seemed keen. I need to send him the info pack by tomorrow and call him next Wednesday.
- **AI's Output:**

Interaction and action have been logged on CRM by **Agentify**

- Client: Mr. Hung
- Location: Coffee House
- Summary: Discussed a potential Home Renovation Loan (~800M VND). The client is highly interested.
- New Tasks:
+ Send home renovation loan info package - Due: Tomorrow (25/10).
+ Follow up call with Mr. Hung - Due: Next Wednesday (29/10).

3.5. Other Use Cases

Category	Use Case	Description
Deeper Client Strategy	1. Attrition Risk Identification	The RM asks the AI to "scan my portfolio for flight risks." The AI analyzes transaction data, login frequency, and recent service calls to flag the top clients who show signs of leaving (e.g., "Ms. Oanh: Transferred 40% of her balance to an external account last week.").
	2. "Smart Briefing" Before Call	The RM asks, "Brief me on Client X, I'm calling them in 2 minutes." The AI provides a "smart briefing" with their recent transactions, last interaction summary, upcoming milestones, and personal details (e.g., "Their son's CD matures next week. Remember to ask about their recent trip to Da Nang.").
	3. Goal-Based Scenario Modeling	The RM asks, "Client An wants to save $50,000 for a down payment in 3 years. Model this." The AI uses the client's current financial profile to instantly create 2-3 scenarios (e.g., "Option 1: Increase monthly savings by $X. Option 2: Move $Y to this new fund...").
Research & Knowledge	4. Complex Product & Policy Q&A	The RM asks a complex question, saving them from searching internal wikis. "What's the eligibility for our new 'Green Biz' loan, and how does it compare to the standard one?" The AI provides a clear, side-by-side comparison.
	5. Personalized Market Intelligence	The RM asks, "What market news today affects my clients?" The AI scans news and market data, cross-references it with the RM's portfolio, and provides a summary (e.g., "Tech stocks are down 4%. This may impact Mr. Binh. Suggested Action: Draft a check-in?").

Interaction & Comms	6. Client Inquiry Triage	The RM is flooded with emails/messages and asks the AI, "Summarize my unread client messages and prioritize them." The AI groups messages by urgency and even pre-drafts replies for common requests (e.g., "Urgent: Ms. Dung (loan rate query)...").
	7. Automated Portfolio Review Prep	The RM asks, "Prepare my Q4 review deck for Mr. Binh." The AI gathers all the client's performance data, charts, and new opportunities (from Use Case 3.1) and compiles them into a client-ready presentation summary.
Compliance & Admin	8. Pre-Communication Compliance Check	Before sending a complex wealth management email, the RM asks, "Check this message for compliance." The AI scans the text for promissory language, suitability issues, or missing disclaimers and suggests edits (e.g., "Flag: 'guaranteed return'").
	9. Intelligent Task Management	The RM dictates a complex follow-up. "Remind me to check on Mr. Son's loan application every Monday and Friday until it's approved." The AI creates the recurring task and links it to the CRM, automatically closing the task when the loan status changes to "Approved."
	10. Performance Gap Analysis	The RM asks, "I'm behind on my 'New Deposits' goal. Why?" The AI analyzes the RM's activity (from Use Case 3.3) and reports, "Data shows your contact rate for clients with maturing CDs is 35% lower than the bank average. Suggestion: Focus on this segment."

IMPACT OF THE SOLUTION

1. Impacts on Stakeholders

At a macro level, **Agentify** is targeting one of the largest opportunities in the financial sector. McKinsey estimates that generative AI could add **$200 billion to $340 billion in value annually** to the global banking industry, equivalent to a **9% to 15% increase in operating profits** (McKinsey, 2023). **Agentify** platform provides the precise tools to capture a share of that value.

The solution provides distinct, cascading benefits for all three stakeholders:

1.1. End User (Relationship Managers)

- **Drastically Reduces Administrative Work: Agentify** automates the most time-consuming, low-value tasks. RMs spend, on average, **half of their time on non-revenue-generating activities** (Accenture, 2022). **Agentify** directly attacks this by providing "Instant Performance Reporting" (Use Case 3.3) and, most critically, "Painless CRM Data Entry" (Use Case 3.4), saving RMs from "manual data entry" and "navigating complex reporting tools."
- **Increases Effectiveness and Reduces Stress:** The "Proactive Catch-up Agent" acts as an "automated lookout," ensuring "no opportunity or client event is missed." This turns a chaotic, reactive job into a managed, proactive one, allowing the RM to focus on high-value conversations rather than data-hunting.
- **Makes Them a "Strategic Advisor":** The interactive co-pilot elevates the RM's role. It provides "Strategic Prioritization" (Use Case 3.1) and "Next Best Action" (Use Case 3.2) on demand. This empowers every RM with the insights of a top-tier analyst, helping them nurture relationships more effectively.

1.2. Customer (The Bank)

- **Drives Measurable Revenue and ROI:** The solution is a revenue-generation engine. The "Strategic Prioritization" feature is proven to **increase client retention rates by 27%** (CoinLaw, 2025). Furthermore, the "Next Best Action" (NBA) feature directly drives cross-selling, a strategy that has shown a **162% ROI** and a **100% increase in specific loan products** in banking case studies (Latinia, 2025; WWT, 2021).
- **Solves the "Bad Data" Problem:** The "Painless CRM Data Entry" feature (Use Case 3.4) is a fundamental strategic benefit. By making data entry effortless, it solves the root cause of "bad data" - a

problem that 35% of CFOs cite as the **key inhibitor for AI adoption** (Gartner, cited in NetSuite, 2025). **Agentify** provides the clean, reliable data the bank needs for all other strategic initiatives.

- **Improves RM Productivity and Scalability:** By automating administrative work, the bank makes its entire RM workforce more efficient. This allows RMs to manage larger portfolios more effectively, increasing the bank's scalability and profitability without a linear increase in headcount.

1.3. Society (The Bank's Customers)

- **Receives Authentic, Personalized Service:** Customers benefit from "timely, highly personalized client communication." The "Personalization Engine" ensures messages have "quality and authenticity," solving a major industry complaint: **nearly 90% of high-net-worth investors feel the advice they receive is "too generic"** (swissQuant, 2024).
- **Benefits from Proactive Financial Guidance:** The "Proactive Catch-up Agent" ensures customers are contacted at critical financial moments, such as a "loan maturity date" or "CD expiry." This proactive service helps them make better, more timely financial decisions, fostering trust and financial well-being.

2. Competitors Analysis

Agentify is a good solution because it's an intelligent, multi-layered platform that moves beyond being a simple, reactive tool to become a comprehensive co-pilot that is both proactive and strategic.

It is better than existing solutions in three key ways:

2.1. Authentic Personalization vs. Generic Templates

Competitors use basic "formal/informal" toggles that produce robotic messages. **Agentify**'s "Personalization Engine" is far more sophisticated. Its **"Passive Learning"** method analyzes *thousands* of an RM's real-world past messages from third-party apps (like Outlook, Zalo, etc.) to learn their unique, authentic "tone, formality, common phrases, and sign-offs." This means the AI-generated drafts sound human and trustworthy.

2.2. Proactive Action vs. Reactive Alerts

Most CRM tools are reactive; they are dashboards that require an RM to manually pull information. **Agentify**'s "Proactive Catch-up Agent" is an "automated lookout." It doesn't just alert the RM to an event (like a client's

birthday); it completes the entire workflow by writing the personalized, ready-to-send email for that event, turning hours of work into a one-click approval.

2.3. A Simple Chat UX vs. Complex Tools

Competitors force RMs to use "complex reporting tools" and clunky CRM data-entry screens. **Agentify**'s "Interactive AI Agent" consolidates these functions into a simple, natural-language chat interface. An RM can get a full performance breakdown or log an entire client meeting by simply typing or dictating an unstructured summary. This simplicity ensures adoption and solves the root cause of poor data entry. A traditional dashboard shows you what happened. Our AI Agent tells you what to do next. We bridge the critical gap between data and action.

3. Unique Selling Point

Agentify's competitive advantage lies in its three-part system that creates a complete, end-to-end co-pilot for Relationship Managers (RMs).

3.1. The "Passive Learning" Personalization Engine

This is the "brain" that makes every AI-generated message sound authentic.

While competitors use generic "formal/informal" toggles, **Agentify**'s **Passive Learning** (Effortless Method) is its secret sauce. During a one-time setup, the AI scans the RM's actual past sent messages from **third-party apps like Outlook, Zalo, and Facebook**.

This allows it to learn the RM's **unique, real-world tone, common phrases, and sign-offs**. As a result, when the AI drafts a message, it has genuine quality and authenticity, making the client feel they are talking to their RM, not a robot.

3.2. Painless, Unstructured CRM Data Entry

This is the "killer feature" that solves the most hated administrative task in banking. Instead of manually filling out complex CRM forms after a meeting, an RM can **simply type or dictate an unstructured summary** (Use Case 3.4). This saves hours of admin work and solves the bank's "bad data" problem or tedious work logging at the source.

3.3. The Dual-AI Co-pilot System

Agentify isn't just one tool; it's a "multi-layered" system that works in two ways:

- **A Proactive Agent (The "Lookout"):** This feature *pushes* work to the RM. It constantly monitors CRM data for triggers (like a "CD expiry" or "inactive client"). It doesn't just send an alert; it **writes the personalized, ready-to-send email** (using the Personalization Engine) so the RM can review and send it in one click.
- **An Interactive Agent (The "Chief of Staff"):** This feature *pulls* information for the RM. It's an on-demand advisor that the RM can chat with to get **"Strategic Prioritization"** (e.g., "Who are my top 3 prospects for this new account?") or a **"Next Best Action"** (e.g., "What's my next move for Mr. Son?").

This closed loop - where the AI proactively finds opportunities and interactively provides on-demand strategy - creates a complete co-pilot that makes the RM more efficient and effective.

DEEP DIVE INTO THE SOLUTION

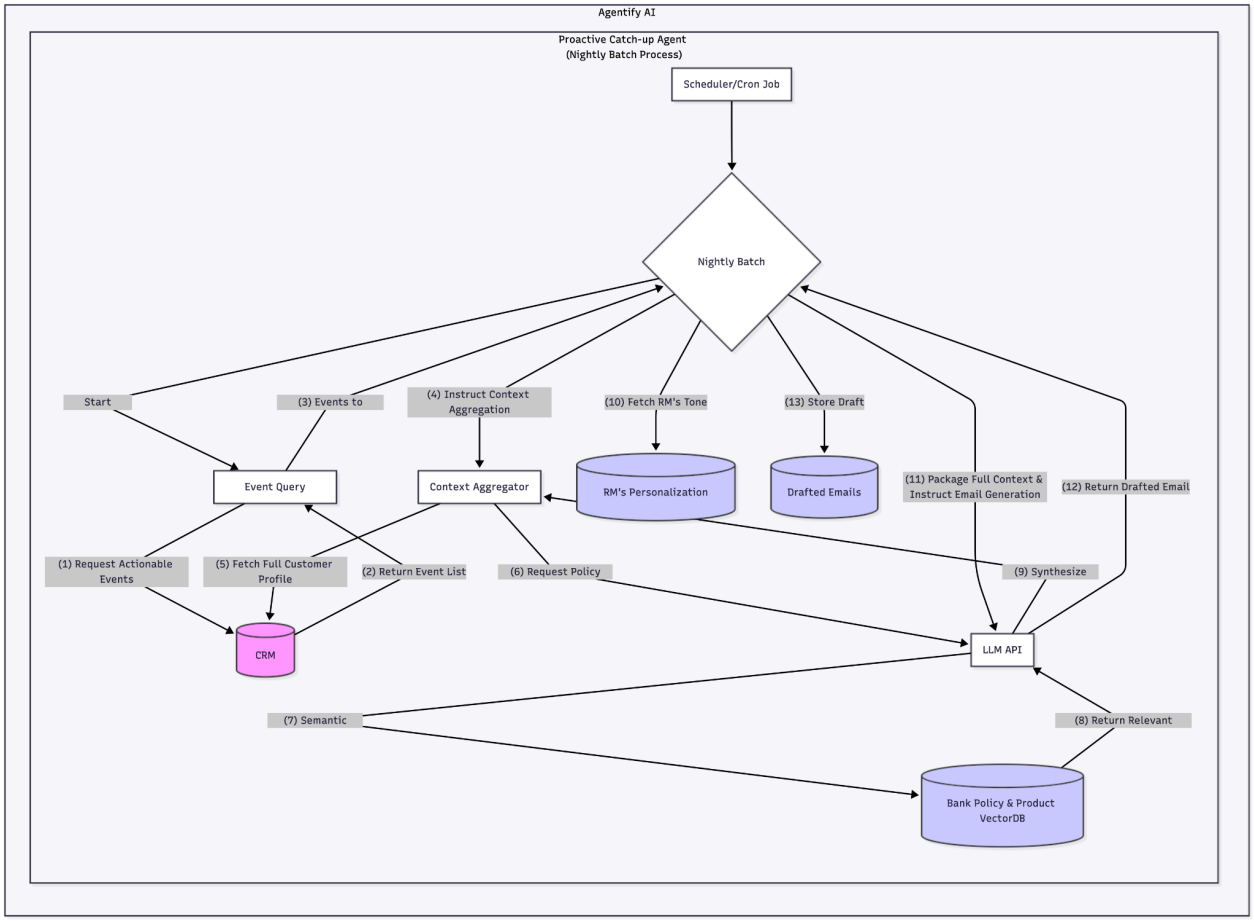
Our proposed solution, **Agentify**, is built on a sophisticated, multi-layered architecture designed to be both powerful and modular. It comprises two primary systems working in concert: **The Proactive Catch-up Agent**, which automates outbound communication preparation, and **The Interactive AI Agent**, which serves as an on-demand strategic co-pilot for the RM. Both are managed through our Model Context Protocol (MCP) Server, ensuring seamless data flow and context awareness.

1. The Proactive Catch-up Agent

This feature is the engine behind **Agentify**'s ability to ensure no opportunity is missed. It operates as a nightly batch process, meticulously preparing personalized, context-aware communications so that RMs can start their day with actionable, ready-to-send drafts.

- **Nightly Trigger (Cron Job / Scheduler):** An automated scheduler (e.g., AWS EventBridge) that initiates the entire workflow at a set time, such as midnight.
- **Event Query Module:** A service that queries the core database to identify all time-sensitive client events for the upcoming day (e.g., birthdays, loan maturities, CD expiries).
- **CRM API:** The secure, managed gateway to the bank's core systems. This API is the single source of truth for all customer profiles, transaction histories, and product holdings, ensuring data integrity and adherence to security protocols.
- **Nightly Batch Orchestrator:** The central controller that receives the list of events and manages the sequence of tasks required to generate an email for each one.
- **Context Aggregator:** A crucial service that gathers and assembles a complete "context package" for the LLM, including customer data, event details, bank policies, and RM preferences.
- **Bank Policy Documents & Vector Database:** A secure repository of internal documents (product terms, compliance rules) that are pre-processed and stored in a VectorDB (e.g., ChromaDB). This allows for rapid, semantic retrieval of the most relevant policy information for any given situation.
- **LLM API (OpenAI / Claude):** The Large Language Model performs two key roles:
    1. **Retrieval & Synthesis:** Intelligently queries the Vector Database and synthesizes the retrieved policy information into a concise, relevant summary.
    2. **Email Writing:** Uses the complete context package to draft a natural, personalized, and compliant email in the RM's unique voice.

- **RM's Personalization Engine:** Stores the pre-learned (or pre-set) communication style and tone prompts for each RM.
- **Drafted Emails DB:** A dedicated database where the final, ready-to-send emails are stored, linked to the customer, event, and RM.



The process executes nightly for each identified event:

1. **Initiate Process:** At midnight, the **Nightly Trigger** activates the workflow.
2. **Scan for Triggers:** The **Event Query Module** calls the **CRM API** to request all actionable client events scheduled for the day.
3. **Return Event List:** The **CRM API** returns a list of events (e.g., customer_id: 123, event_type: CD_EXPIRY).
4. **Orchestrate Tasks:** The **Nightly Batch Orchestrator** iterates through each event. For each one, it instructs the **Context Aggregator** to begin its work.

5. **Gather Full Context:** The **Context Aggregator** performs several fetches:
    - It retrieves the full customer profile and specific event details by making targeted calls to the **CRM API**.
    - It invokes the **LLM API** to perform a retrieval task. The LLM generates a semantic query (e.g., "renewal options for high-net-worth client CD") and sends it to the **Vector Database**.
    - The VectorDB returns relevant policy document chunks. The LLM **synthesizes** these into a clear summary (e.g., "Offer promotional rate X% for Diamond-tier clients").
6. **Fetch RM's Tone:** The orchestrator retrieves the specific RM's pre-configured **tone and style prompt** from the Personalization Engine.
7. **Generate Draft Email:** The **Orchestrator** packages all assembled information—customer profile, event details, synthesized policy, and RM's tone—into a final, comprehensive prompt. It calls the **LLM API** in its "email writer" mode.
8. **Store Draft:** The LLM returns a fully drafted, personalized, and compliant email. The **Orchestrator** stores this in the **Drafted Emails DB**, ready for the RM's review in the morning.

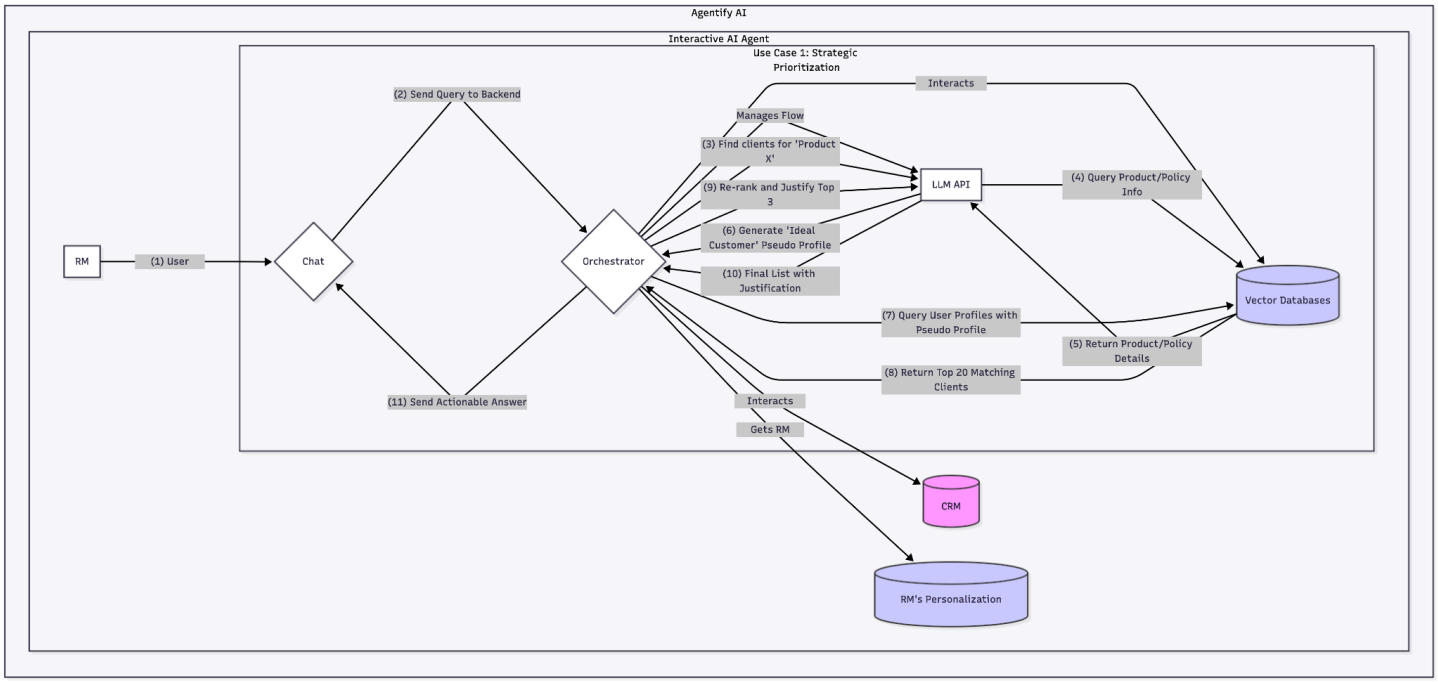
2. The Interactive AI Agent

This is the RM's on-demand "Chief of Staff," a chat-based co-pilot for strategic advice, data queries, and administrative tasks. Its power lies in a flexible, multi-stage architecture that leverages a common set of core components orchestrated differently depending on the RM's query.

- **Chat UI:** The RM's intuitive interface for asking questions in natural language.
- **Orchestrator:** The brain of the operation. It receives the RM's request, interprets the intent, and invokes the correct sequence of processes, managing the flow of data between the LLM and various data sources.
- **LLM API (OpenAI / Claude):** The core reasoning engine used to understand user intent, generate database queries, synthesize information from multiple documents, and formulate natural language responses.
- **Vector Databases (VectorDB):** For lightning-fast semantic search. We utilize three distinct VectorDBs:
    - **Product VectorDB:** Contains detailed information on all of VPBank's products.
    - **Policy VectorDB:** Securely stores all internal bank policies and compliance rules.
    - **User Profile VectorDB:** Stores a vectorized "essence" of every client, capturing a holistic view of their financial status, behavior, and relationship history for rapid matching.

- **CRM API:** The secure interface for accessing the bank's source of truth. It's used to fetch real-time, detailed client data and execute read-only queries against the CRM database.
- **RM's Personalization Engine:** Stores the pre-learned (or pre-set) communication style and tone prompts for each RM.

2.1. Use case 1 - Strategic Prioritization

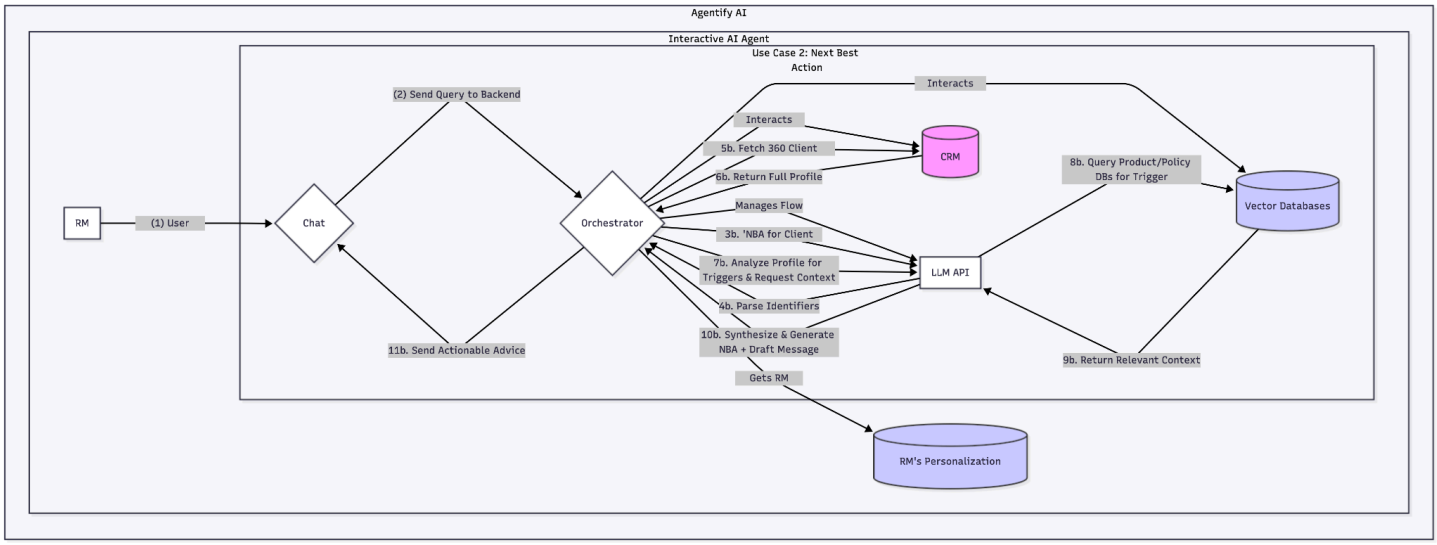


To answer "Who are my top prospects for this new product?", the system uses an intelligent funneling approach instead of a brute-force search.

1. **User Query:** The RM asks in the **Chat UI**, "Find top clients for the new 'Premier Invest' account."
2. **Product & Policy Retrieval:** The **Orchestrator** directs the **LLM API** to query the **Product VectorDB** and **Policy VectorDB** to understand the product's features, target demographic, and associated sales rules.
3. **Synthesize Context:** The LLM synthesizes the retrieved information into a rich, consolidated summary of the product and its constraints.
4. **Create "Pseudo Profile":** The **Orchestrator** tasks the **LLM API** with a critical instruction: "Based on the synthesized context, generate a detailed natural language description of the ideal customer for this product."

5. **Vectorize Ideal Profile:** This "pseudo profile" is converted into a vector embedding, representing the perfect customer mathematically.
6. **Similarity Search:** This "pseudo-vector" is used to query the **User Profile VectorDB**, which instantly returns a list of the top 20 clients who are the closest mathematical match.
7. **Re-rank and Justify:** This list is passed back to the **LLM API** for final filtering. The LLM applies the policy rules (from Phase 1) to the matched profiles, re-ranks them based on business logic (e.g., de-prioritizing clients with recent complaints), and selects the top 3. Crucially, it generates a concise justification for *why* each client is a top prospect.
8. **Return Actionable Answer:** The **Orchestrator** sends the final, justified list to the **Chat UI**, empowering the RM with a strategic, data-driven action plan.

2.2. Use case 2 - Next Best Action

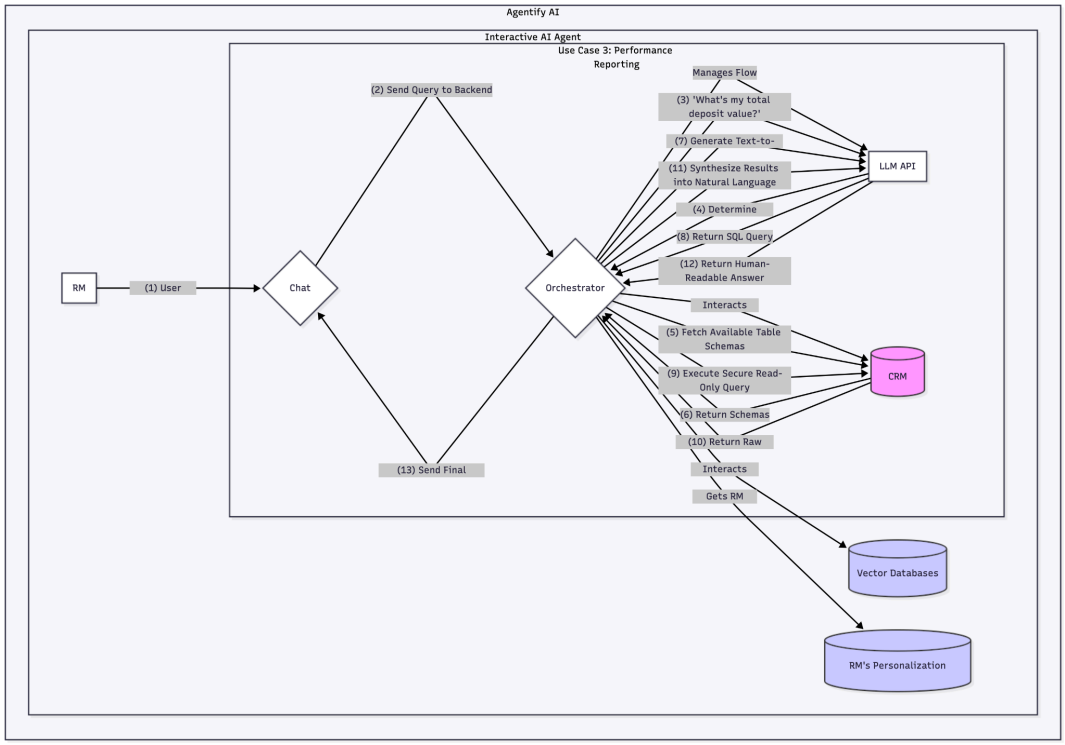


While Strategic Prioritization is a *one-to-many* problem (one product, many clients), the Next Best Action (NBA) feature solves the inverse: a *many-to-one* problem (many possible actions for one client).

1. **User Query:** The RM asks, "Client: Mr. Son Pham, 0823912039. What's my next best action?"
2. **Identify and Fetch:** The **Orchestrator** uses the **LLM API** to parse identifiers (name, phone) and then executes a comprehensive query against the **CRM / Core Banking DB** to aggregate a full 360-degree client profile (financials, history, upcoming milestones).

3. **Analyze Profile:** The **Orchestrator** analyzes the client's profile for key triggers, such as an upcoming CD maturity.
4. **Retrieve Relevant Context:** It then directs the **LLM API** to query the **Product** and **Policy VectorDBs** for context specific to that trigger (e.g., "renewal options for CD XYZ," "compliance rules for maturing deposits").
5. **Synthesize Situation:** The LLM returns a concise summary of the client's immediate situation, combining their profile with relevant product and policy information.
6. **Generate NBA:** The **Orchestrator** sends the complete context package (360-profile + situational summary) to the **LLM API** with the prompt: "Given this complete view, prioritize all potential actions by urgency and value, determine the single best next action, and draft a message in the RM's voice."
7. **Reason and Draft:** The LLM reasons that the 12-day CD maturity deadline is the most urgent and valuable action. It generates the recommendation, a justification, and a personalized, ready-to-send message using the RM's style from the **Personalization Engine**.
8. **Return Actionable Advice:** The **Orchestrator** delivers the complete recommendation to the **Chat UI**, transforming the RM into a proactive, trusted advisor.

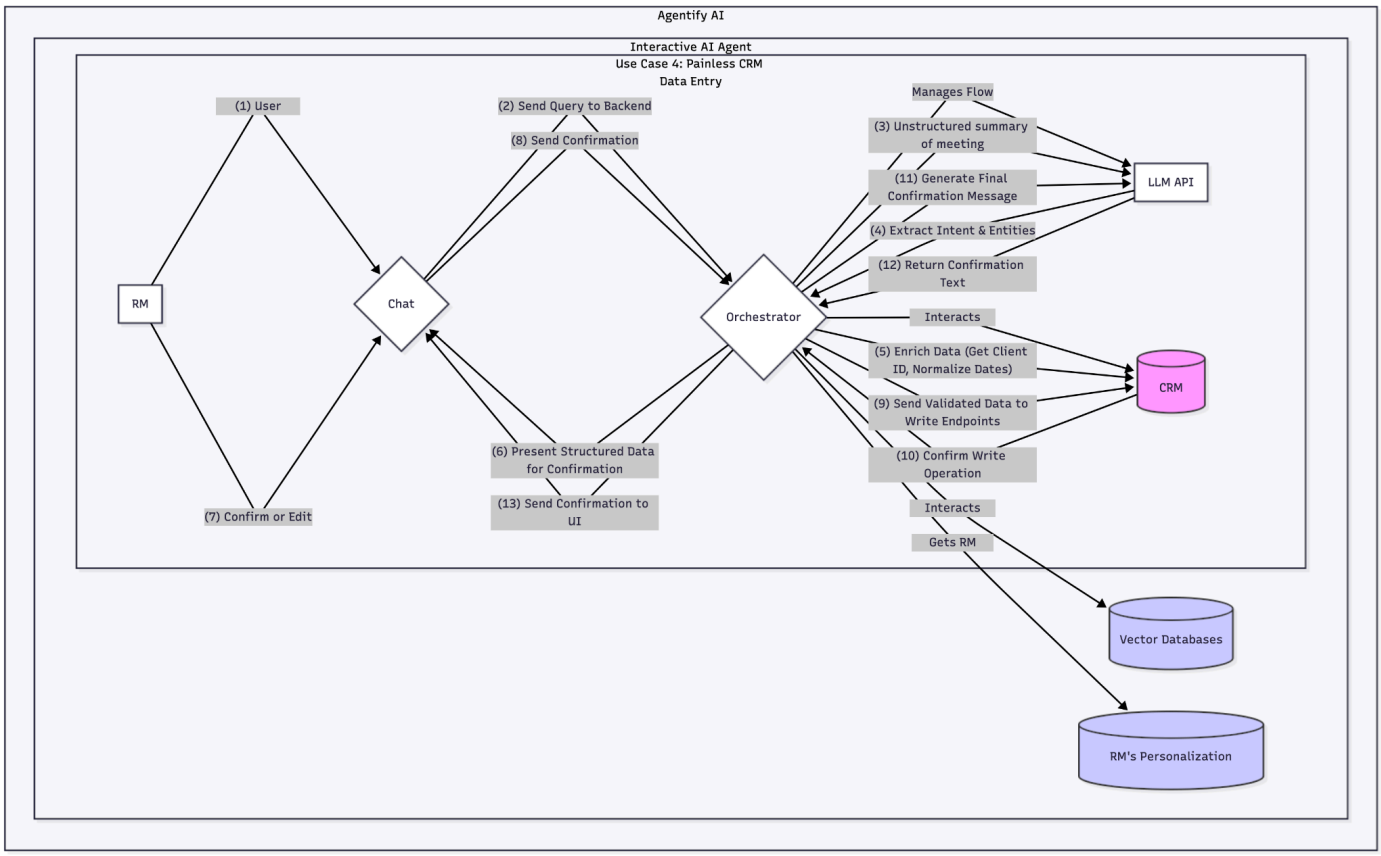
2.3. Use case 3 - Instant Performance Reporting



This use case transforms how RMs access their performance data, replacing manual report-pulling with a simple, conversational interface. The system is designed to safely convert natural language questions into precise database queries that are executed *through* the **CRM API**.

1. **User Query:** The RM asks a question in the **Chat UI**, such as, "What's my total deposit value this month?"
2. **Intent Recognition and Table Identification:** The **Orchestrator** receives the query. The **LLM API** determines the "performance reporting" intent. The **Orchestrator** then calls a dedicated endpoint on the **CRM API** (e.g., GET /api/crm/data/tables) to fetch a list of available, queryable data views or tables.
3. **Schema Retrieval and Column Filtering:** The **Orchestrator** makes further calls to the **CRM API** (e.g., GET /api/crm/data/tables/{table_name}/schema) to fetch the schemas for the relevant tables. This schema information, along with the user query, is passed to the **LLM API** to select only the most relevant columns.
4. **Text-to-SQL Generation:** The Orchestrator sends the user's query, the filtered schemas, and a carefully crafted prompt to the **LLM API**, instructing it to write a precise, executable SQL query.
5. **Secure Query Execution:** The generated SQL query is first validated internally. Then, the **Orchestrator** sends the query to a specific, secure endpoint on the **CRM API** (e.g., POST /api/crm/data/query). This API endpoint is responsible for running the read-only query against the database. This ensures the AI agent operates within the bank's established security and access control policies and can never modify production data.
6. **Result Synthesis and Answer Generation:** The **CRM API** returns the raw, structured data from the query. This data is passed back to the **LLM**, which interprets the table of results and synthesizes it into a clear, natural language summary.
7. **Deliver Final Answer:** The **Orchestrator** sends the final, human-readable response to the **Chat UI**. The entire process happens in seconds, providing the RM with an instantaneous and secure way to access performance metrics.

2.4. Use case 4: Painless CRM Data Entry



This feature is designed to eliminate the most tedious part of an RM's job: manual data entry. It transforms the process from a structured, form-filling exercise into a simple, conversational task, directly solving the "bad data" problem at its source by incorporating a crucial human-in-the-loop validation step.

1. **User Query:** The RM provides a natural language summary of a client interaction in the **Chat UI**, such as, "Met Mr. Son Pham 0823912039 at a coffee house, talked about a renovation loan for his apartment, which needs about 800m vnd. He seemed keen. I need to send him the info pack by tomorrow and call him next Wednesday."
2. **Intent and Entity Extraction:** The **Orchestrator** receives the unstructured text and sends it to the **LLM API**. The **LLM** parses the text to identify the core intent ("log a client interaction and create follow-up tasks") and extracts all relevant entities:
    a. **Client Identifiers:** Name ("Mr. Son Pham") and Phone ("0823912039").
    b. **Interaction Details:** Location ("coffee house"), Topic ("renovation loan"), Amount ("800m vnd"), and Sentiment ("seemed keen").

c. **Action Items:** Two distinct tasks are identified: "send him the info pack by tomorrow" and "call him next Wednesday."

3. **Data Structuring and Enrichment:** The extracted entities are returned to the **Orchestrator**, which enriches the data for clarity and system compatibility:
    a. **Client Disambiguation:** It makes a call to the **CRM API** (GET /api/crm/clients?phone=0823912039) to retrieve the unique client ID, ensuring the log is associated with the correct person.
    b. **Date Normalization:** It uses the **LLM** or a built-in utility to convert relative dates ("tomorrow," "next Wednesday") into absolute, standardized formats (e.g., "2025-10-25," "2025-10-29").
4. **Present for User Confirmation and Revision:** Before any data is written to the CRM, the **Orchestrator** sends the structured, human-readable summary back to the **Chat UI**. The interface displays a clear confirmation card. The system now pauses and waits for the RM's input. This step is vital as it gives the user full control, allowing them to verify the AI's interpretation and make corrections if needed.
5. **API Payload Generation:** Only after the RM clicks **"**Confirm**"** does the **Orchestrator** proceed. It assembles the validated data into a formal JSON payload ready for the CRM, separating the interaction log from the actionable tasks.
6. **Secure CRM Write Operations:** The **Orchestrator** sends the formatted payload to the secure **CRM API** write endpoints (e.g., POST /api/crm/interactions and POST /api/crm/tasks). These endpoints are responsible for applying all business logic and validation before committing the data, ensuring adherence to the bank's data governance rules.
7. **Final Confirmation:** Once the **CRM API** confirms that the records have been successfully created, the **Orchestrator** instructs the **LLM API** to generate a final, concise confirmation message. This message replaces the confirmation card in the UI, providing a clear audit trail of the action taken.

3. The LLMOps Pipeline

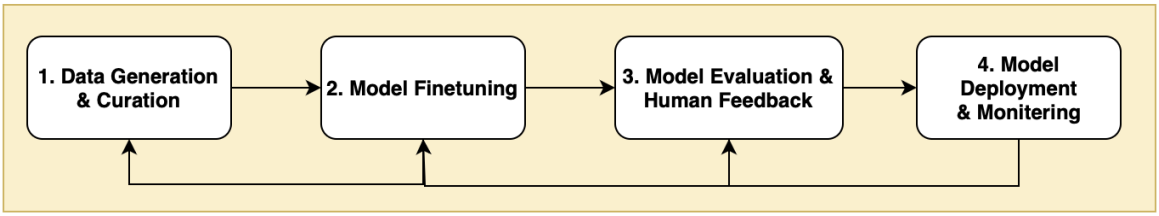
3.1. Motivation to train the dedicated agentic models.

In banking, third-party LLM APIs pose the problems of leaking customers data or data protection policy violation. We propose our solution to train our own models to meet business and regulatory requirements.

- **Security & Compliance:** Only a self-hosted model ensures full control, data privacy, and regulatory compliance (GDPR, data residency), eliminating third-party risk.

- **Accuracy & Specialization:** Generic models lack banking expertise. A fine-tuned model becomes a domain expert, improving accuracy and preventing harmful errors.
- **Control & Efficiency:** Our own model delivers predictable costs, low latency, and reliability without dependency on external APIs.

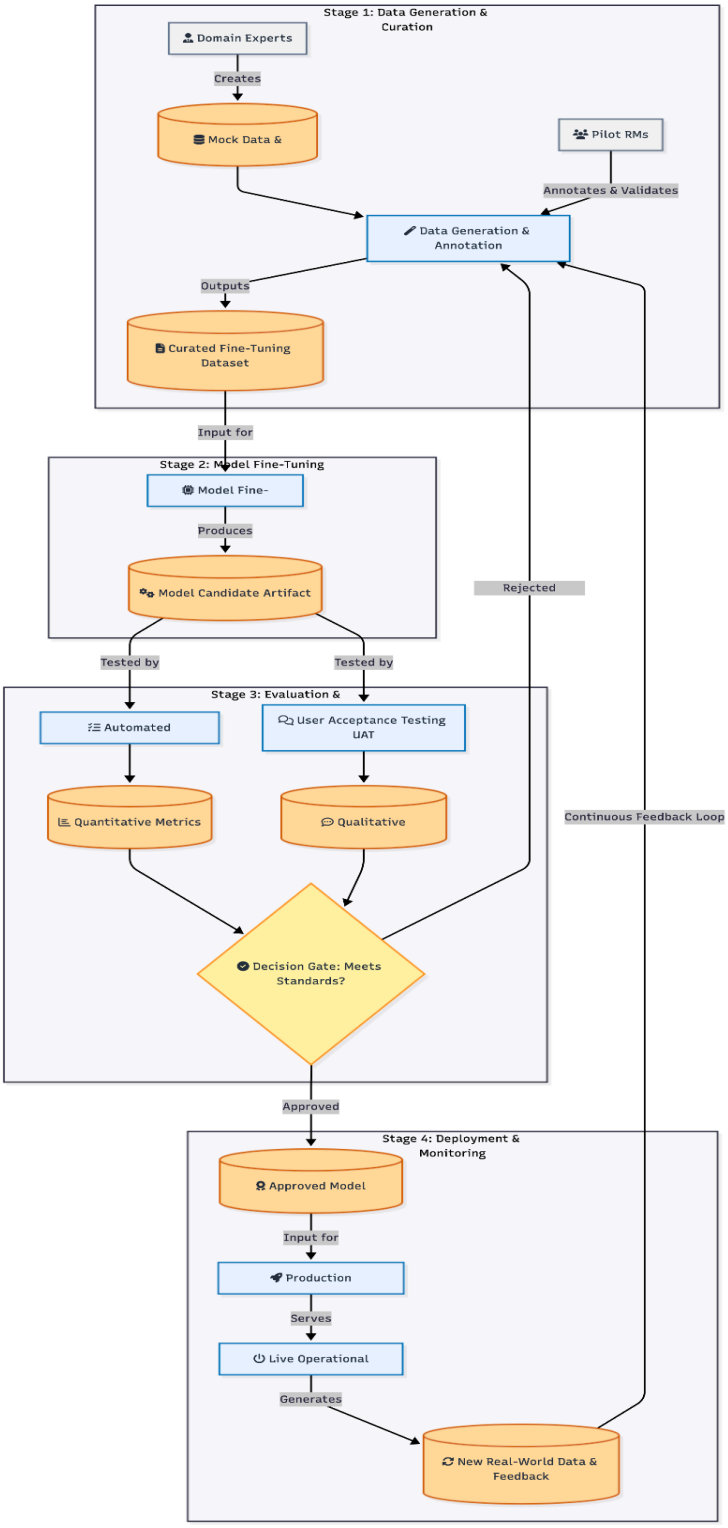
This LLMOps pipeline is the foundation for a secure, compliant, and intelligent AI co-pilot built for bank systems.



Overall there are four key stages in the pipeline:

- Stage 1: Agentic Data Generation & Curation
- Stage 2: Model Fine-Tuning
- Stage 3: Evaluation & Human-in-the-Loop (HITL) Feedback
- Stage 4: Production Deployment & Continuous Monitoring

3.2 Details of each stage



3.2.1. Stage 1: Data Generation & Curation

This initial stage is foundational, creating the high-quality, specialized data needed to teach the model.

- **Input:**
    1. **Domain Expertise:** Banking experts design realistic, anonymized mock customer data and interaction scenarios.
    2. **User Queries:** A large set of synthetic queries representing typical RM tasks is created (e.g., "Summarize customer X's recent activity and suggest a suitable investment product").
- **Process:**
    1. A powerful "teacher" LLM (e.g., Claude 3 Opus) is configured as an agent backbone with tools to interact with the mock data, following the pipeline present in the section 1 and 2.
    2. The teacher agent processes the synthetic queries, generating detailed logs of its entire workflow: its reasoning ("thoughts"), the tools it selects, the specific API calls it makes, and its final, synthesized answer.
    3. This prototype system is exposed to a pilot group of RMs who validate the responses, provide corrections, and give quality ratings. This human annotation is critical for capturing nuance.
- **Output:** A curated and structured instruction dataset (~10,000+ examples) containing high-quality examples of agentic behavior (input, thought process, tool use, final output).
- **Decision Gate:** Has the dataset reached the target size and quality threshold as validated by experts? If yes, proceed to fine-tuning.

3.2.2. Stage 2: Model Fine-Tuning

Here, the knowledge captured in the curated dataset is distilled into a smaller, efficient "student" model.

- **Input:** The curated fine-tuning dataset from Stage 1.
- **Process:**
    1. A smaller, open-source model (e.g., Llama 3 8B) is selected as the base "student" model.
    2. A managed fine-tuning job is executed, training the student model on the curated dataset.
    3. The process is tracked as an experiment, logging all hyperparameters, data versions, and performance metrics.
- **Output:** A **Model Candidate**, a trained model artifact, along with its training metrics. This artifact is versioned and stored in a central model registry.

3.2.3. Stage 3: Evaluation & Human-in-the-Loop Feedback

This crucial quality assurance stage validates the Model Candidate before it can be considered for production.

- **Input:** The versioned Model Candidate from the model registry.
- **Process:**
    1. **Automated Evaluation:** The model is tested against a reserved test set (~500+ conversations). Its performance on quantitative metrics (e.g., tool-use accuracy, response relevance, BLEU/ROUGE scores) is measured.
    2. **User Acceptance Testing:** If the model passes the automated benchmarks, it is deployed to a controlled staging environment integrated with the CRM. A select group of RMs use it for real-world tasks and provide qualitative feedback and ratings.
- **Output:** A comprehensive evaluation report containing both quantitative scores and qualitative human feedback.
- **Final Decision Gate:** Does the model meet the predefined quality, accuracy, and safety standards based on both automated and human evaluation? If yes, the model artifact is approved for production. If not, the feedback is used to return to Stage 1 (to generate more specific data) or Stage 2 (to adjust tuning parameters).

3.2.4. Stage 4: Deployment & The Continuous Feedback Loop

The approved model is released into production, where its performance is monitored to fuel the next cycle of improvement.

- **Input:** The approved and versioned Model Artifact from the model registry.
- **Process:**
    1. An automated CI/CD pipeline retrieves the approved model artifact and deploys it to a scalable, resilient production endpoint. Strategies like A/B testing can be used to compare the new model against the incumbent.
    2. The live model's operational performance (latency, errors) and model quality (data drift, concept drift) are continuously monitored.
    3. **Feedback Loop:** All user interactions, generated outputs, and explicit user feedback (e.g., thumbs up/down) from the production environment are logged.
- **Output:**

1. A live, operational model serving all RMs.
2. A continuous stream of new, real-world data that is fed back into the data lake, becoming the input for the next iteration of the entire cycle, starting again at Stage 1.

ARCHITECTURE OF THE SOLUTION

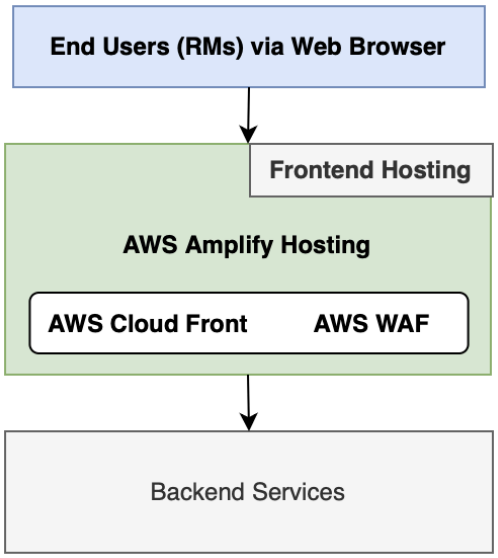
Overall architecture of the system

The system architecture consists of four integrated services:

- **Frontend Service:** Provides a secure web interface for Relationship Managers (RMs) to interact with backend systems via APIs.
- **Proactive Notification & Action Agent Service:** Automates background workflows, proactive alerts, and policy lookups using containerized components on AWS Fargate.
- **Copilot Agent Service:** An AI-powered assistant that connects LLMs with CRM data to deliver contextual insights and guided actions.
- **LLMOps Pipeline:** Manages continuous model improvement through data generation, fine-tuning, evaluation, and deployment.

Together, these services enable intelligent automation, seamless user interaction, and continuous learning across the platform. Below we present the detailed architecture of each service.

1. Architecture of the Frontend Service

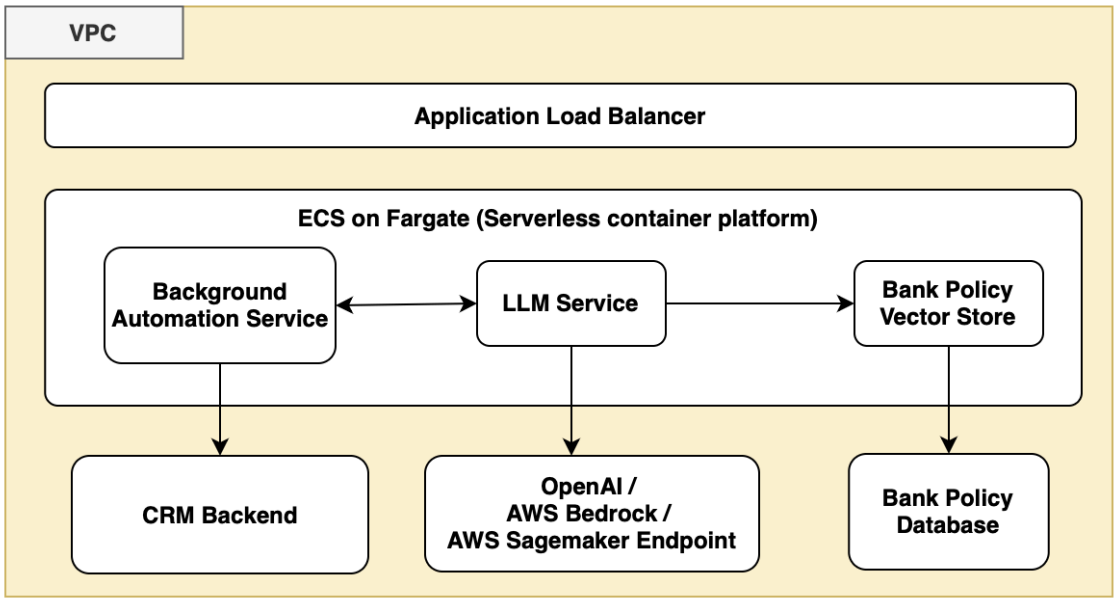


- General Description: The web interface provides tools for RM to interact with the backend system via API calls. End users (RM) access the web application through their browser. Requests are routed via **AWS CloudFront**, which serves cached content for speed and passes traffic through **AWS WAF** for inspection and protection. The traffic then reaches **AWS Amplify Hosting**, where the frontend content is hosted and served back to the user.

- Why choose the services:

- **AWS Amplify Hosting:** A managed service for deploying and hosting modern web apps with built-in CI/CD, scaling, and HTTPS support, making it ideal for reliable and fast frontend delivery.
- **AWS Cloud Front:** A global CDN that caches and delivers content from edge locations, reducing latency and improving speed and availability for users worldwide.
- **AWS WAF (Web Application Firewall):** A security layer that filters and blocks malicious web traffic, protecting the app from common attacks like SQL injection and XSS.

2. Architecture of Proactive Notification & Action Agent Service



- General Description:

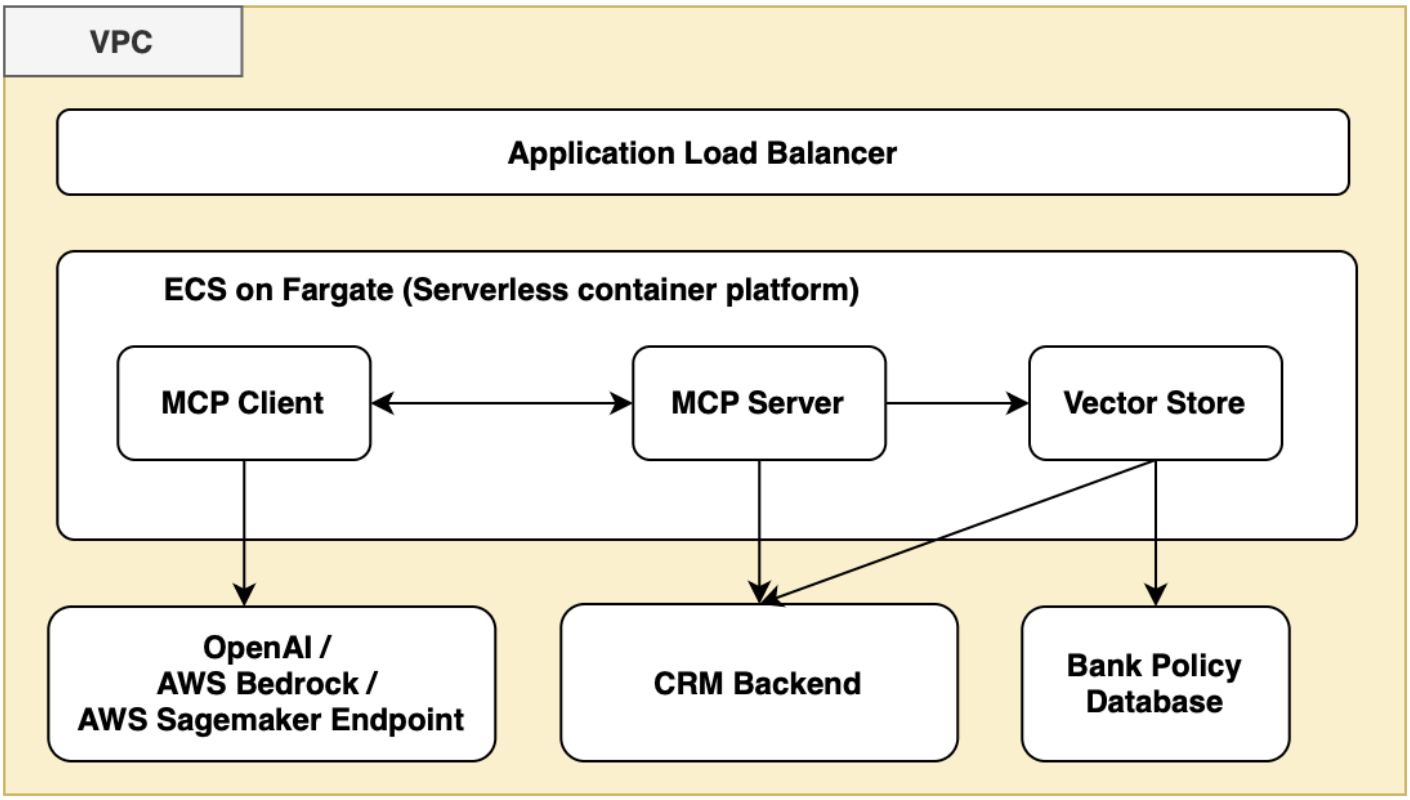
- The architecture consists of three main containerized components deployed on **Amazon ECS using AWS Fargate**, running within a **Virtual Private Cloud (VPC)** and fronted by an **Application Load Balancer (ALB)**.
    - The **Background Automation Service** handles recurring and asynchronous workflows, including scheduled cron jobs and retrieval of user context and data from the **CRM Backend**.
    - The **LLM Service** manages all interactions with external large language model providers such as **OpenAI**, **AWS Bedrock**, or **AWS SageMaker Endpoints**, processing queries and orchestrating responses.
    - The **Bank Policy Vector Store** acts as a specialized retrieval component that stores and indexes vectorized policy data sourced from the **Bank Policy Database**, providing quick and context-aware reference lookups for the LLM Service.
      This design ensures modularity, operational isolation, and scalability across each functional area—automation, language processing, and data retrieval.

- Why choose the services:

- **Amazon ECS on AWS Fargate:** Provides a fully managed, serverless container orchestration platform. By using Fargate, teams avoid manual server management and gain automatic scaling, simplified deployment, and built-in fault tolerance for each service. This reduces infrastructure overhead while maintaining consistent performance.
- **Amazon ECR (Elastic Container Registry):** Serves as the secure repository for all container images (Background Automation Service, LLM Service, and Bank Policy Vector Store). It integrates seamlessly with ECS, enabling controlled image versioning, scanning, and automated deployments.
- **Application Load Balancer (ALB):** Manages inbound traffic and distributes requests efficiently across ECS tasks. It ensures availability, secure HTTPS termination, and supports routing rules to direct traffic to the right service endpoints.
- **Amazon VPC (Virtual Private Cloud):** Provides a secure and isolated network environment. ECS tasks and backend services run in private subnets, protected from external access, while the ALB resides in a public subnet to safely handle incoming requests. This isolation ensures compliance and security for sensitive banking and CRM data.
- **CRM Backend:** A prebuilt system integrated through secure APIs. The Background Automation Service interacts with it to fetch user context, transactional data, and other relevant information to support downstream automation and decision-making.

- **Bank Policy Vector Store and Database:** The Vector Store maintains embeddings of bank policy documents for fast semantic search and contextual retrieval by the LLM Service. The underlying Policy Database serves as the source of truth for all bank regulations and policy documents, ensuring that the LLM always references accurate and updated data.

3. Architecture of the Copilot Agent Service



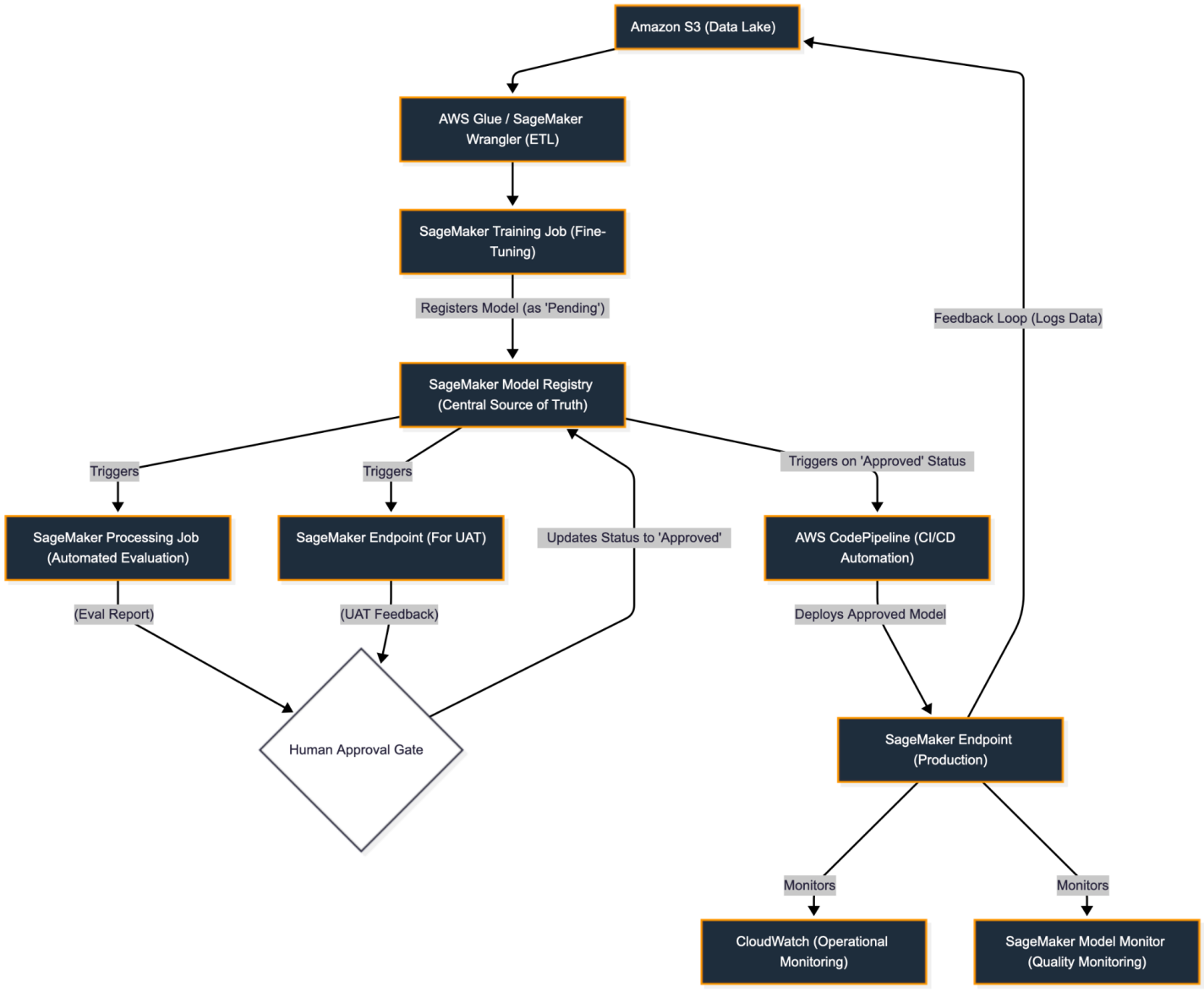
- General Description: The MCP architecture consists of two main containerized components: the **MCP Client** and the **MCP Server**, both deployed on **Amazon ECS using AWS Fargate** for a fully serverless and scalable container environment.

- The **MCP Client** communicates directly with external Large Language Model (LLM) services such as **OpenAI API**, **AWS Bedrock**, or **AWS SageMaker Endpoints** to process and generate responses.
- The **MCP Server** acts as a bridge between the MCP Client and the **CRM Backend,** the **Vector Store**, fetching relevant business data and contextual information to support intelligent responses.

- All components are deployed within a **Virtual Private Cloud (VPC)**, fronted by an **Application Load Balancer (ALB)** that securely routes incoming requests to the appropriate ECS tasks.
  This design ensures a modular, secure, and highly available platform for AI-driven CRM operations.

- Why choose the services: The justification for choosing the services is like the section for the architecture of Proactive Notification & Action Agent/

4. Architecture of the LLMOps Pipeline



Following the stages proposed in the Solution Deep Dive Section, we present the architecture for the LLMOps pipeline.

4.1. Stage 1: Agentic Data Generation & Curation

- **Data Lake: Amazon S3** serves as the central data lake, with distinct prefixes for raw logs, mock CRM data, processed datasets, and user feedback.
- **"Teacher" Model:** The powerful teacher agent is powered by a foundation model accessed via **Amazon Bedrock** (e.g., Claude 3 Opus), ensuring data remains within AWS.
- **Data Processing (ETL): AWS Glue** or **Amazon SageMaker Data Wrangler** is used for large-scale batch processing to transform raw logs into structured fine-tuning formats. **AWS Lambda** can be used for real-time processing of smaller data streams like user feedback.

4.2 Stage 2: Model Fine-Tuning

- **Managed Training: Amazon SageMaker Training Jobs** provide the fully managed, scalable environment for running fine-tuning scripts. It automatically provisions and tears down GPU instances.
- **Experiment Tracking: Amazon SageMaker Experiments** is used to log, track, and compare all training runs, including hyperparameters and output metrics.
- **Model Storage & Versioning:** The **Amazon SageMaker Model Registry** is the central repository for storing, versioning, and managing the lifecycle (e.g., pending, approved, rejected) of all trained model artifacts.
- **Custom Environments: Amazon ECR (Elastic Container Registry)** stores custom Docker container images if the training environment requires specific libraries or dependencies.

4.3 Stage 3: Evaluation & Human-in-the-Loop (HITL) Feedback

- **Automated Evaluation: Amazon SageMaker Processing Jobs** or **Batch Transform** are used to run the model against the holdout test set in a scalable manner.
- **Staging/UAT Deployment:** The Model Candidate is deployed to an **Amazon SageMaker Endpoint** configured for the staging environment. This endpoint is isolated from production traffic.
- **UAT Frontend:** A secure web application for RMs to interact with the model is hosted using **AWS Amplify** (for rapid development) or on **Amazon ECS/Fargate** for more control. It operates within the bank's VPC.

- **Feedback Collection:** User ratings and qualitative feedback from the UAT interface are captured in **Amazon DynamoDB** for low-latency writes and structured queries, with raw feedback logs also sent to **Amazon S3**.

4.4. Stage 4: Production Deployment & Continuous Monitoring

- **CI/CD Automation: AWS CodePipeline**, **AWS CodeCommit**, and **AWS CodeBuild** create a CI/CD pipeline that automates the deployment of an approved model from the SageMaker Model Registry to the production endpoint. Infrastructure is defined using **AWS CloudFormation** or **AWS CDK**.
- **Production Inference: Amazon SageMaker Endpoints** serve the model for real-time inference.
    - **Autoscaling** policies are configured to handle varying loads.
    - **Serverless Inference** can be used for workloads with intermittent traffic to optimize costs.
    - **Endpoint Variants** are used to deploy multiple models for A/B testing.
- **Operational Monitoring: Amazon CloudWatch** monitors endpoint metrics (latency, invocations, error rates), captures logs, and triggers alarms for anomalies.
- **Model Quality Monitoring: Amazon SageMaker Model Monitor** is configured to detect data and concept drift by comparing live production traffic against the training data baseline, ensuring the model's performance doesn't degrade over time.

REFERENCES

Accenture. (2022, October 13). *Why relationship managers are more crucial than ever*. Accenture Capital Markets Blog. Retrieved October 23, 2025, from https://capitalmarketsblog.accenture.com/why-relationship-managers-are-more-crucial-than-ever

Bain & Company. (2024). *Five functions where AI is already delivering*. Bain & Company. Retrieved October 23, 2025, from https://www.bain.com/insights/five-functions-where-ai-is-already-delivering-tech-report-2024/

CoinLaw. (2025, October 16). *AI in wealth management statistics 2025: Key AI milestones and industry shifts*. CoinLaw. Retrieved October 23, 2025, from https://coinlaw.io/ai-in-wealth-management-statistics/

eMoney Advisor. (2025). *Crafting a client retention strategy: Best practices for financial planners*. Retrieved October 23, 2025, from https://emoneyadvisor.com/blog/crafting-a-client-retention-strategy-best-practices-for-financial-planners/

Experian. (2024, March 12). *Your guide to tackling data quality issues across a CRM system's lifecycle*. Experian. Retrieved October 23, 2025, from https://www.experian.co.uk/blogs/latest-thinking/wp-content/uploads/sites/13/2024/03/guide-tackling-data-quality-issues-CRM-lifecycle.pdf

Goyal, A., Kareem, S., Singh, A., Vignesh, V., Arun, V., & Chidambaram, S. (2024, October 23). *FMOps/LLMOps: Operationalize generative AI and differences with MLOps*. AWS Machine Learning Blog. https://aws.amazon.com/blogs/machine-learning/fmops-llmops-operationalize-generative-ai-and-differences-with-mlops/

Kareem, S., Subramanian, R., Singh, A., Vignesh, V., Arun, V., & Chidambaram, S. (2024, August 14). *MLOps foundation roadmap for enterprises with Amazon SageMaker*. AWS Machine Learning Blog. https://aws.amazon.com/blogs/machine-learning/mlops-foundation-roadmap-for-enterprises-with-amazon-sagemaker/

Kitces.com. (2019, March 18). *How do financial advisors actually spend their time?* Retrieved October 23, 2025, from https://www.kitces.com/blog/how-do-financial-advise-spend-time-research-study-productivity-capacity-efficiency/

Latinia. (2025, August 9). *Examples of next best actions for banking marketing*. Latinia. Retrieved October 23, 2025, from https://latinia.com/en/resources/examples-next-best-actions-marketing-banking

NetSuite. (2025, February 18). *The 8 top data challenges in financial services (with solutions)*. NetSuite. Retrieved October 23, 2025, from https://www.netsuite.com/portal/resource/articles/financial-management/data-challenges-financial-services.shtml

SelectAdvistorsInstitute. (2025, February 27). *Client experience in investment management: The key to long-term success*. SelectAdvistorsInstitute. Retrieved October 23, 2025, from https://www.selectadvisorsinstitute.com/our-perspective/2025/2/27/client-experience-in-investment-management-the-key-to-long-term-success

Smart Communications. (2024). *Customer communications in financial services*. Retrieved October 23, 2025, from https://www.smartcommunications.com/wp-content/uploads/BM-24-FS-SC.pdf

swissQuant. (2024, January 18). *How to master the challenges for banking relationship managers in 2024*. swissQuant. Retrieved October 23, 2025, from https://swissquant.com/industry-insights/charting-the-course-mastering-the-challenges-banking-relationship-managers-face-in-2024/

WWT. (2021, September 7). *The next best action for banks*. WWT. Retrieved October 23, 2025, from https://www.wwt.com/article/the-next-best-action-for-banks

Zendesk. (2025, July 7). *5 banking customer experience trends to consider for 2025*. Zendesk. Retrieved October 23, 2025, from https://www.zendesk.com/blog/customer-experience-in-banking/