Prepare the business logic for the platform

Based on the comprehensive system architecture designed for the Enterprise Agentic Workflow Engine (EAWE) and the detailed pain-point analysis provided, the following document outlines the core business logic that will govern the platform's functionality. This logic serves as the blueprint for development, ensuring the system is intelligent, secure, auditable, and directly solves the enterprise challenges identified[1].

Guiding Principles of the Business Logic

The entire system will operate on a set of core principles that inform every logical decision.

- **Human-Centricity:** Automation serves to augment human capabilities, not replace them. Every process must have clear points for human oversight, intervention, and final authorization.
- **Modularity and Reusability:** Logic is encapsulated into modular components (agents, tools, templates) that can be configured and reused across different departments and workflows.
- **Security by Design:** Access control, data permissions, and credential management are fundamental to the logic, not an afterthought.
- **Full Auditability:** Every action taken by an agent and every decision made by a human is immutably logged to ensure transparency and compliance.
- **Continuous Improvement:** The system is designed to learn from human feedback, becoming more efficient and accurate over time.

Core Business Logic by Architectural Component

1. Workflow Lifecycle Management

This logic governs how workflows are created, initiated, executed, and concluded.

A. Workflow Creation (in the Workflow Studio)

- **Template-Based Start:** Users will initiate workflow creation by selecting a pre-built template corresponding to a departmental pain point (e.g., "HR Onboarding," "Sales Proposal Generation," "AP Invoice Processing").
- **Visual Composition:** The user composes a workflow by dragging agents onto a canvas and connecting them. Each connection represents the flow of data and control.
- **Tool Authorization:** When an agent is added to a workflow, the designer must explicitly grant it access to specific tools (e.g., granting the "Communications Agent" access to the "Send Email" tool). The agent can only use the tools authorized for that specific workflow.

- **Validation Logic:** Before a workflow can be saved as "Active," the system runs a validation check:
  - Ensures the workflow graph has a defined start and end point.
  - Verifies that all required parameters for each agent and tool are configured.
  - Confirms that all connections are valid (e.g., you cannot pass a text file to a tool expecting a number).

B. Workflow Triggering & Initiation

- Workflows are initiated based on one of three trigger types:
  - **Event-Driven:** The Core Orchestration Engine listens for events from integrated systems (e.g., a new row in a Salesforce table, a new email with "invoice" in the subject line). Upon detecting a matching event, it instantiates and runs the corresponding workflow. This is the default and preferred method.
  - **Scheduled:** Workflows can be configured to run on a schedule (e.g., "Run the 'Sales Performance Report' workflow every Friday at 5 PM").
  - **Manual/Ad-Hoc:** A user can manually trigger a workflow from the UI or via a voice/text command that is interpreted as an actionable intent.

C. Workflow Execution (by the Core Orchestration Engine)

- **State Management:** The engine maintains the state of every active workflow instance in a database. This includes the current step, all collected data (the "payload"), and a history of completed steps.
- **Task Queuing:** The engine places tasks for agents onto a message queue. This ensures that even if an agent service temporarily fails, the task will be processed once the service recovers.
- **Conditional Logic:** Workflows can have decision points (e.g., "If invoice amount > $10,000, send to Director for approval; otherwise, send to Manager"). The engine evaluates these conditions based on the current data payload to determine the next step.
- **Error Handling:**
  - **Retry:** If a tool fails due to a transient error (e.g., temporary network issue), the agent will automatically retry up to 3 times.
  - **Escalation:** If a tool fails permanently or the agent cannot achieve its goal, the workflow is paused, and a task is created in the Human-in-the-Loop (HITL) interface for manual review.

2. Agentic & Retrieval Logic

This logic defines how agents "think" and how the system retrieves knowledge.

A. Agent Reasoning Loop

- Each agent operates on a **Goal-Oriented Action** loop:

1. **Receive Goal:** The agent is assigned a goal by the Orchestration Engine (e.g., "Extract details from the attached invoice").
2. **Reason:** The agent uses its underlying LLM to break the goal down into a sequence of steps.
3. **Select Tool:** The agent selects the most appropriate tool from its authorized toolset for the first step.
4. **Execute & Observe:** The agent executes the tool and observes the result (e.g., the extracted text from an OCR tool).
5. **Repeat:** The agent assesses if the goal has been met. If not, it uses the observation from the previous step to reason about the next step and repeats the cycle.

B. Knowledge Retrieval & Synthesis (RAG) Logic

- **Permission-Aware Ingestion:** When the IDDR module ingests data into the vector database, it also ingests and attaches the source system's access control permissions as metadata to each data chunk.
- **Query Logic:** When a user queries the system via voice or text:
  1. The user's identity is authenticated.
  2. The query is converted into a vector embedding.
  3. The vector database is queried for semantically similar chunks. **Crucially, the query is filtered to only return chunks where the metadata permissions match the user's credentials.**
  4. The retrieved chunks (the context) and the original query are passed to the Retrieval Agent's LLM to synthesize a final answer. This ensures a user can never see information they are not authorized to access.
- **Intent Recognition:** The Retrieval Agent is trained to differentiate between a knowledge query ("Who is the account manager for Acme Corp?") and an action-oriented command ("Generate a proposal for Acme Corp and send it for approval"). If an action is detected, it will suggest initiating the relevant workflow.

3. Human-in-the-Loop (HITL) & Approval Logic

This logic governs the critical interface between automation and human control.

- **Trigger Conditions for HITL:** A task is routed to the HITL dashboard if:
  - It is an explicit "Approval" step in a workflow.
  - An agent encounters a persistent error it cannot resolve.
  - An agent's confidence score for a decision (e.g., categorizing an expense) is below a configurable threshold (e.g., 90%).
- **Multi-Tier Approval Routing:** The system uses rules to route approvals to the correct individuals, directly reflecting the organizational hierarchy:
  - **Rule Engine:** A simple rule engine will be used (e.g., `IF 'invoice_amount' > 5000 AND 'department' == 'Marketing' THEN route_to 'Marketing Director'`).

    - **Dynamic Routing:** The system can look up a user's manager in the HRIS (via a tool) for dynamic escalation.
- **Actionable Feedback Loop:**
    - **Approve:** The workflow proceeds. The system logs the approval.
    - **Reject:** The workflow terminates. The user must provide a reason, which is logged for analysis.
    - **Edit & Approve:** The user can modify data processed by the agent (e.g., correct an OCR error on an invoice line item) before approving. **This is the most critical feedback.** The system logs the `(original_agent_output, human_corrected_output)` pair. This data is flagged and added to a dataset for future fine-tuning of the agents, creating a continuous learning loop.

4. Administration & Security Logic

- **Role-Based Access Control (RBAC):**
    - **Administrator:** Can manage users, billing, system-wide integrations, and security settings.
    - **Designer:** Can create, edit, and test workflows. They cannot manage users or system settings.
    - **Operator/User:** Can manually run workflows they have permission for and handle tasks in the HITL approval queue. They cannot design or edit workflows.
- **Credential Management:** The Integration & Tooling Layer includes a secure vault. Agents do not have direct access to API keys or passwords. They request access to a tool, and the vault provides a temporary, scoped token for the duration of the execution.
- **Immutable Audit Trail:** Every event is logged with a timestamp, the actor (user or agent), the action taken, and the data payload involved. This log is write-only and cannot be altered, ensuring a complete and compliant audit history for any workflow instance.

⁂

1. Organizational-Workflow-Automation-Analysis_-Ident.pdf