

Database Manager Agent System Documentation

Overview

The Database Manager Agent System is a comprehensive business management platform that orchestrates database operations for inventory, orders, suppliers, and business registration. Built using Google ADK (Agent Development Kit) with Gemini 2.0 Flash models, the system provides automated business administration through specialized AI agents.

Technology Stack

- **AI Framework:** Google ADK (Agent Development Kit)
 - **Model:** Gemini 2.0 Flash
 - **Database:** MySQL
 - **Integration:** Telegram Bot API
 - **Programming Language:** Python
-

Main Orchestrator Agent

Orchestrator

- **Name:** Orchestrator
- **Model:** gemini-2.0-flash
- **Role:** Database Administrator for Business Owners

Description

Acts as the central database administrator orchestrating all database-related tasks including inventory management, order processing, supply chain management, and business account administration. The Orchestrator ensures data integrity, handles user authentication, and provides structured business reports.

Capabilities

- **Account Management:** Creates, updates, and manages business accounts
- **Data Orchestration:** Coordinates between inventory, orders, and supply management
- **Report Generation:** Compiles comprehensive business reports
- **User Authentication:** Handles login/logout processes and account verification

- **Bot Integration:** Manages Telegram bot creation and messaging
- **Order Fulfillment:** Processes customer and supply orders with inventory updates

Sub-Agents

1. **Inventory Agent** (`inventory_agent`)
2. **Order Manager** (`order_manager`)
3. **Supply Agent** (`supply_agent`)
4. **The Registrar** (`the_registrar`)

Agent Tools

- (`inventory_agent`) - Inventory management operations
 - (`order_manager`) - Order processing and management
 - (`supply_agent`) - Supplier relationship management
 - (`the_registrar`) - Business registration and account management
 - (`run`) - Telegram messaging utility
 - (`get_contact_details`) - Customer service contact retrieval
-

Specialized Agents

1. Inventory Manager

- **Name:** (`inventory_manager`)
- **Model:** (`gemini-2.0-flash`)
- **Role:** Product Inventory Management

Description

Handles all inventory management tasks including adding, modifying, and deleting products. Monitors stock levels, tracks expiry dates, and provides inventory reports with threshold alerts.

Capabilities

- **Product Management:** Add, update, delete inventory items
- **Stock Monitoring:** Track quantity levels and minimum thresholds
- **Expiry Tracking:** Monitor product expiration dates
- **Category Management:** Organize products by categories

- **Inventory Reporting:** Generate comprehensive stock reports
- **Low Stock Alerts:** Identify items below minimum thresholds

Tools

- `get_rows_with_exact_column_values` - Database query utility
- `view_items` - Display inventory items
- `update_table` - Database update operations
- `add_item` - Add new inventory items
- `get_expired_goods` - Retrieve expired products
- `execute_query_in_str` - Custom query execution
- `get_minimum_threshold` - Get low-stock items

Database Tables

- **Primary:** `product` table
 - **Fields:** `business_id`, `item_name`, `category`, `brand`, `quantity_in_stock`, `minimum_threshold`, `selling_price`, `minimum_selling_price`, `expiry_date`, `metadata`, `active`
-

2. Order Manager

- **Name:** `order_manager`
- **Model:** `gemini-2.0-flash`
- **Role:** Order Processing and Management

Description

Manages both customer and supplier orders, handling order creation, status updates, and fulfillment processes. Coordinates with inventory management for stock updates during order processing.

Capabilities

- **Order Creation:** Process new supply orders
- **Order Tracking:** Monitor order status and fulfillment
- **Order Updates:** Modify existing orders and status
- **Fulfillment Processing:** Handle order completion with inventory updates
- **Order Reporting:** Generate unfulfilled order reports

- **Multi-type Support:** Handle both customer and supplier orders

Tools

- `add_supply_order` - Create new supply orders
- `execute_query` - Database query operations
- `update_table` - Order status updates
- `get_supplier_id_by_mail` - Supplier identification
- `get_product_id` - Product identification
- `get_supplier_id_by_name` - Supplier lookup by name
- `get_unfulfilled_supplier_order` - Pending supply orders
- `get_unfulfilled_customer_orders` - Pending customer orders
- `get_rows_with_exact_column_values` - Database filtering

Database Tables

- **Primary:** `supply_order`, `customer_order`
 - **Related:** `product`, `supplier`, `customer`, `business`
-

3. Supply Manager

- **Name:** `supply_manager`
- **Model:** `gemini-2.0-flash`
- **Role:** Supplier Relationship Management

Description

Manages all supplier-related operations including supplier registration, inventory management, and supply chain coordination. Handles supplier product catalogs and availability tracking.

Capabilities

- **Supplier Management:** Add, update, delete supplier information
- **Supplier Inventory:** Manage supplier product catalogs
- **Supply Chain Coordination:** Track supplier product availability
- **Cost Management:** Monitor supplier pricing
- **Supplier Reporting:** Generate supplier and inventory reports

- **Product Sourcing:** Identify products without suppliers

Tools

- `get_rows_with_exact_column_values` - Database filtering
- `get_no_suppliers` - Find unsupplied products
- `view_items` - View inventory items
- `add_supplier` - Register new suppliers
- `add_supplier_and_supplier_inv` - Add supplier with inventory
- `add_to_supplier_inv` - Add items to supplier inventory
- `get_supplier_id_by_mail` - Supplier identification
- `get_supplier_inv_id` - Supplier inventory lookup
- `get_product_id` - Product identification
- `execute_query` - Custom database queries
- `delete_supplier_and_supplier_inv` - Remove suppliers
- `delete_supplier_inv` - Remove supplier inventory items
- `get_single_value` - Single value retrieval
- `update_table` - Database updates
- `view_supplier_inventory` - Display supplier catalogs
- `view_suppliers` - List all suppliers
- `get_single_supplier_inventory` - Individual supplier inventory

Database Tables

- **Primary:** `supplier`, `supplier_inventory`
 - **Related:** `product`, `business`
-

4. The Registrar

- **Name:** `the_registrar`
- **Model:** `gemini-2.0-flash`
- **Role:** Business Registration and Account Management

Description

Handles business registration, account creation, and business profile management. Integrates with bot creation services to establish customer service channels for new businesses.

Capabilities

- **Business Registration:** Complete business account setup
- **Account Verification:** Validate business information
- **Profile Management:** Update business details
- **Bot Integration:** Create customer service bots
- **Account Lifecycle:** Handle account deletion and updates
- **Age Verification:** Ensure business owner eligibility

Sub-Agents

- **Bot Mama** (`bot_mama`) - Telegram bot creation service

Agent Tools

- (`bot_mama`) - Telegram bot creation and management

Tools

- (`add_business`) - Register new business
- (`verify_business`) - Validate business existence
- (`delete_business`) - Remove business accounts
- (`update_table`) - Update business information

Database Tables

- **Primary:** (`business`)
 - **Fields:** id, username, name, business_name, brief_description, date_joined, contact_details, physical_address, date_of_birth, active, tg_bot_username, tg_bot_link, tg_bot_token, chat_id
-

5. Bot Mama

- **Name:** (`bot_mama`)
- **Model:** (`gemini-2.0-flash`)
- **Role:** Telegram Bot Creation Service

Description

Specialized agent responsible for creating Telegram bots for businesses through interaction with Telegram's BotFather. Handles bot naming, username generation, and initial setup.

Capabilities

- **Bot Creation:** Interact with Telegram BotFather
- **Username Generation:** Create unique bot usernames
- **Bot Configuration:** Initial bot setup and activation
- **Token Management:** Retrieve and manage bot tokens
- **Link Generation:** Create bot access links

Tools

- `run` - Telegram messaging utility for BotFather interaction
-

System Architecture

Data Flow

1. **User Authentication** → Orchestrator validates user credentials
2. **Task Delegation** → Orchestrator routes requests to appropriate sub-agents
3. **Data Processing** → Sub-agents execute specialized operations
4. **Report Compilation** → Orchestrator aggregates results
5. **User Response** → Formatted output delivered to user

Database Integration

- **MySQL Backend:** Centralized data storage
- **Transaction Management:** Ensures data consistency
- **Active Record Pattern:** Soft deletion with `active` flags
- **Relational Integrity:** Foreign key relationships maintained

Security Features

- **Access Control:** Business-specific data isolation
- **Input Validation:** Comprehensive data verification
- **Soft Deletion:** Maintains data history with active flags
- **Age Verification:** Ensures business owner eligibility (18+ years)

Integration Points

- **Telegram API:** Bot creation and messaging
- **Database Layer:** MySQL connectivity and operations
- **Agent Communication:** Inter-agent data sharing
- **Report Generation:** Automated business analytics

Usage Patterns

Common Workflows

1. **Business Registration:** Account creation → Bot setup → Initial configuration
2. **Inventory Management:** Product addition → Supplier linking → Stock monitoring
3. **Order Processing:** Order creation → Inventory updates → Customer notification
4. **Supply Chain:** Supplier registration → Product catalog → Order fulfillment
5. **Reporting:** Data aggregation → Analysis → Formatted output

Error Handling

- **Data Validation:** Input verification at each step
- **Rollback Mechanisms:** Transaction safety
- **User Feedback:** Clear error messages and guidance
- **System Recovery:** Graceful handling of failures

This system provides a comprehensive solution for small to medium businesses seeking automated database management with AI-powered assistance.