International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

AI-POWERED SALES AND MARKETING ANALYTICS PLATFORM USING MULTI-AGENT AI

Mr. Abinash Panda
Student, Dept. of CSE,
GIFT Autonomous, Bhubaneswar

Mr. Biswajit Das
Student, Dept. of CSE,
GIFT Autonomous, Bhubaneswar

Nitu Singh
Assistant Professor, Dept. of CSE,
GIFT Autonomous, Bhubaneswar

***Abstract*— The rapid growth of data-driven business environments and increasing complexity of sales and marketing operations have created a significant demand for intelligent, automated analytics platforms capable of delivering real-time insights. Traditional analytics tools often fail to bridge the gap between raw data and actionable business intelligence, relying on manual processes that are slow, error-prone, and limited in scope. This paper presents the design and implementation of an AI-Powered Sales and Marketing Analytics Platform, referred to as SYNQ, which leverages Multi-Agent AI architecture and Retrieval-Augmented Generation (RAG) to automate the end-to-end analytics process. The proposed system is built around three specialized AI agents: the Analyst Agent, the Strategy Agent, and the Report Agent, operating sequentially under LangGraph orchestration. The Analyst Agent processes data retrieved from a ChromaDB vector database to identify KPIs, trends, and patterns. The Strategy Agent converts analysis results into actionable business recommendations. The Report Agent compiles all insights into a professional PDF report. The Streamlit-based frontend provides KPI metric cards, dynamic charts, an AI chat interface, and one-click PDF report downloads. Results demonstrate an end-to-end pipeline execution time of approximately 3.2 seconds and 94% average RAG retrieval relevance.**

Keywords— *Multi-Agent AI, Retrieval-Augmented Generation, LangGraph, ChromaDB, LLaMA, Groq, Streamlit, Business Analytics, Sales Analytics, Marketing Analytics.*

I. INTRODUCTION

A. Background

In the modern digital era, the rapid expansion of data-driven business operations and the growing complexity of sales and marketing ecosystems have significantly increased the demand for intelligent analytics platforms. Organizations across industries generate massive volumes of transactional and campaign data daily, yet struggle to extract actionable insights in real time. Traditional analytics tools such as static spreadsheets and manual reporting systems are increasingly insufficient for the pace and scale of modern business operations. Business decision-makers need faster, more reliable mechanisms to process data, extract trends, and generate strategic guidance without relying heavily on data science teams [1].

The emergence of large language models (LLMs), vector databases, and multi-agent AI frameworks has opened new possibilities for automating complex analytical tasks. Systems that can autonomously retrieve relevant data, reason over it, and generate professional business reports represent the next frontier in enterprise analytics. The integration of Retrieval-Augmented Generation (RAG) with multi-agent architectures enables these systems to produce outputs that are both data-grounded and strategically meaningful, addressing the shortcomings of conventional BI tools [2].

B. Problem Statement

Despite the availability of various analytics platforms, several critical challenges persist in real-world business environments. Existing systems generate large volumes of raw data, making it difficult for business teams to identify actionable insights efficiently. The lack of intelligent analysis and contextual recommendations leads to delayed decision-making and missed market opportunities. Traditional dashboards often lack natural language interaction, limiting accessibility for non-technical users. Manual report generation is time-consuming and prone to inconsistency across teams. Limited integration of AI-driven insights results in inefficient strategy formulation and slower revenue cycles, highlighting the pressing need for a more advanced, intelligent, and automated analytics platform [3].

C. Motivation

The motivation behind this project stems from the increasing demand for efficient AI-driven analytics solutions capable of handling real-world sales and marketing data. The observation that business teams often struggle with managing complex datasets and maintaining consistency in analytical workflows drove the development of a centralized platform combining data retrieval, intelligent analysis, and automated reporting. The growing availability of open-source LLMs, efficient vector databases, and orchestration frameworks such as LangGraph inspired the creation of a system that leverages these technologies to bridge the gap between data availability and business utility. The success of RAG-based systems in reducing hallucination while providing contextually accurate responses made RAG a natural choice for the retrieval backbone [4].

D. Objectives

The primary objectives of the SYNQ platform are as follows:

- To design and develop an intelligent platform for automated sales and marketing data analysis.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

- To integrate Retrieval-Augmented Generation (RAG) for accurate, data-grounded AI responses.
- To implement a multi-agent pipeline using LangGraph for sequential analysis, strategy generation, and report creation.
- To develop a Streamlit-based interactive dashboard with KPI metrics, smart filters, dynamic charts, and AI chat interface.
- To enable one-click PDF report generation for business stakeholders.
- To demonstrate the practical application of LLMs (Groq + LLaMA) in real-world business analytics.
- To enhance overall decision-making speed and quality for sales and marketing teams.

E. Scope of the Project

The scope of this project includes the design and implementation of a full-stack AI-based analytics platform capable of processing real-world sales and marketing datasets and delivering structured business insights through a multi-agent pipeline and interactive dashboard. The system focuses on two primary datasets — retail sales data and marketing campaign data — and integrates semantic data retrieval, intelligent analysis, strategy generation, and report formatting. The AI component leverages RAG and LLM-based reasoning, keeping the system practical and deployable without heavy infrastructure requirements.

II. LITERATURE REVIEW

A. Existing Analytics Systems

Traditional business analytics platforms such as Tableau, Power BI, and Looker have become industry standards for data visualization and reporting. These platforms provide rich visual interfaces but rely heavily on manual configuration, domain expertise, and human interpretation of results. Tools such as AWS QuickSight Q and Google Looker Studio represent a step towards more accessible analytics. However, these tools remain largely query-response systems that do not autonomously generate strategic insights or synthesize findings into professional business documents [4].

B. AI in Business Analytics

Artificial Intelligence has emerged as a transformative technology in business analytics, enabling systems to analyse vast datasets and detect patterns that are difficult to identify manually. AI techniques such as machine learning, natural language processing, and rule-based systems are being widely adopted for trend detection, demand forecasting, and customer segmentation. Research demonstrates that AI-driven analytics systems can significantly improve insight quality and automate report generation processes. However, many AI-based solutions require large-scale model training, which may not be feasible in real-time enterprise environments with limited infrastructure [5].

C. Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation, introduced by Lewis et al. (2020), enhances LLM outputs by grounding them in retrieved context from an external knowledge base. Unlike fine-tuned models, RAG systems retrieve relevant documents at inference time, significantly reducing hallucination and improving factual accuracy. ChromaDB is a leading open-source vector database used for RAG implementations, supporting semantic similarity search using sentence-transformer embeddings, enabling fast and accurate retrieval of contextually relevant data rows from large datasets [6].

D. Multi-Agent AI Frameworks

Multi-agent systems involve multiple AI entities that collaborate to perform distinct tasks. LangGraph, developed by LangChain, provides graph-based orchestration for multi-agent LLM pipelines with precise control over agent sequencing, state management, and conditional branching. Research consistently demonstrates that agent specialization produces higher-quality outputs than single-agent approaches on complex tasks. Dedicated agents for analysis, strategy, and report generation ensure each cognitive stage is handled optimally with purpose-built prompts and output schemas [7].

E. Research Gap and Proposed Contribution

From the reviewed literature, there is a clear gap in systems that combine RAG-based semantic data retrieval with multi-agent analytical pipelines, real-time interactive dashboards with AI chat and report generation, Groq-accelerated LLM inference with open-source models, and a scalable full-stack architecture with minimal infrastructure requirements. To address this, this project proposes SYNQ — an integrated platform combining ChromaDB-based RAG, a three-agent LangGraph pipeline, Groq + LLaMA inference, and a Streamlit dashboard with real-world datasets.

III. SYSTEM OVERVIEW

A. Proposed System

SYNQ is a web-based application that allows business analysts and sales managers to submit natural language business queries via the Streamlit AI chat interface, receive analysis of sales KPIs, trends, and patterns from the Analyst Agent, obtain actionable strategy recommendations from the Strategy Agent, and download a professional PDF report generated by the Report Agent. Users can also visualize sales and marketing metrics through dynamic charts and KPI cards and apply smart filters by region, product category, and time period.

B. System Architecture

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

The system follows a six-layer pipeline architecture: (1) Data Layer — CSV/Excel datasets cleaned and stored as JSON via Python/Pandas; (2) Embedding Layer — JSON records converted to text embeddings stored in ChromaDB; (3) Retrieval Layer — User queries trigger semantic similarity search in ChromaDB via RAG; (4) Agent Layer — LangGraph orchestrates Analyst, Strategy, and Report agents; (5) Inference Layer — Groq API with LLaMA model processes all LLM calls; (6) Presentation Layer — Streamlit dashboard with charts, KPIs, chat, and PDF export.

C. Technology Stack

Layer	Technology	Purpose
Data	Python / Pandas	Cleans CSV/Excel to JSON
Vector DB	ChromaDB	Stores semantic embeddings
Retrieval	FAISS/Embeddings	Semantic query matching
Orchestration	LangGraph	Multi-agent workflow
LLM Inference	Groq + LLaMA	Powers all AI agents
Frontend	Streamlit	Interactive dashboard
Charts	Plotly	Dynamic visualizations
Reports	ReportLab	Exports reports as PDF

Table I: Technology Stack of SYNQ Platform

D. Key Functional Modules

Module	Description	Technology
Data Pipeline	Data ingestion & preprocessing	Python, Pandas
RAG Module	Semantic retrieval from ChromaDB	ChromaDB, Embeddings
Analyst Agent	Analyses KPIs and trends	LangGraph, Groq
Strategy Agent	Generates recommendations	LangGraph, Groq
Report Agent	Formats PDF report	LangGraph, ReportLab
Dashboard	Interactive UI	Streamlit, Plotly
LLM Engine	Processes agent prompts	Groq API, LLaMA

Table II: Key Functional Modules

E. Advantages of Proposed System

The SYNQ platform provides several distinct advantages over traditional analytics systems. It eliminates manual effort in data analysis and report writing through full pipeline automation. It provides data-grounded insights via RAG, reducing hallucination and increasing stakeholder trust. The modular agent design ensures quality at each analytical stage through specialization. The platform is accessible to non-technical users through natural language chat, democratizing data analytics. The system is scalable for enterprise-level datasets using lightweight, cloud-deployable components.

IV. METHODOLOGY

A. Overall Workflow

The system operates through a sequence of structured steps. First, data ingestion: CSV/Excel datasets are loaded and cleaned using Pandas. Second, embedding: JSON records are stored as vector embeddings in ChromaDB. Third, the user submits a business question via Streamlit. Fourth, RAG retrieval: ChromaDB semantic search returns top-k relevant records. Fifth, the Analyst Agent processes context and generates structured analysis. Sixth, the Strategy Agent converts analysis into recommendations. Seventh, the Report Agent formats the professional report. Finally, results are displayed in chat and a PDF download is made available.

B. Data Collection and Handling

The system works with two real-world datasets. Sample-Superstore.csv contains retail sales transactions with fields including Order ID, Customer Segment, Product Category, Region, Sales, Profit, and Quantity. marketing_campaign.xlsx contains customer profile data including Age, Income, Education Level, Campaign Acceptance Rates, and Purchase Channel Preferences. The data handling process includes input validation and missing value handling via Pandas, normalization of column names, data type conversion, and serialization of each record as a human-readable key-value text string saved as sales_data.json and marketing_data.json.

C. RAG Implementation

The RAG system forms the data retrieval backbone. The embedding process reads cleaned JSON files via data_loader.py, converts each record to a text string, embeds using the all-MiniLM-L6-v2 sentence-transformer model, and stores embeddings in ChromaDB with metadata. During retrieval, the user query is embedded, cosine similarity search is performed, and the top-k most similar records (default k=10) are returned as a structured string for the agent prompt.

Score Range	Similarity	Action
> 0.85	High relevance	Primary context
0.70 - 0.85	Medium relevance	Secondary context
< 0.70	Low relevance	Exclude from prompt

Table III: RAG Retrieval Scoring Criteria

D. Multi-Agent Pipeline

The multi-agent pipeline consists of three specialized agents orchestrated by LangGraph. The Analyst Agent receives the user query and retrieved data context, identifies KPIs, trends, anomalies, and patterns, and outputs a structured

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

JSON. The Strategy Agent receives this output and generates actionable business recommendations with prioritized action items. The Report Agent receives combined analysis and strategy outputs, formats them into a structured professional report with sections, headings, and an executive summary suitable for business stakeholders [9].

E. Backend API and Frontend

The Groq API serves as the LLM backbone for all three agents using the LLaMA 3 model (llama3-8b-8192) via Groq's LPU acceleration. Each agent uses a dedicated system prompt defining its role and output format. Agent outputs are passed sequentially via LangGraph state management using a TypedDict schema. Average token consumption per pipeline execution is approximately 2,400 tokens with response time under 1.2 seconds per agent. The Streamlit frontend uses a modular layout with sidebar filters, main chat area, and KPI metric strip, with st.session_state preserving conversation history.

V. SYSTEM DESIGN

The Use Case Diagram illustrates interaction between platform users and the SYNQ Analytics Platform. Both Business Analysts and Marketing Managers can submit business queries, view KPI metrics and charts, filter data by region and category, analyse sales and marketing data, generate business recommendations, and download PDF reports. The Administrator role has additional privileges for managing datasets and system configuration. The system connects to an External Data Source (CSV/Excel/Database) for data ingestion and preprocessing.

B. ER Diagram

The ER Diagram represents the data entities and relationships. The User entity (user_id, name, role) initiates queries captured in the Query entity (query_id, text, timestamp, user_id FK). Each query generates a Retrieval record (retrieval_id, query_id FK, records_retrieved, similarity_score) and an AgentOutput record (output_id, query_id FK, analyst_result, strategy_result, report_text). The Dataset entity (dataset_id, name, type, file_path) is referenced during retrieval. This structure ensures efficient data flow within the multi-agent pipeline.

C. Sequence Diagram

The Sequence Diagram illustrates the interaction during a query execution cycle. The user enters a query in the Streamlit UI, which forwards it to the RAG Module. The RAG Module queries ChromaDB, which returns the top-k similar records. The Analyst Agent processes the retrieved context via the Groq API and returns structured analysis. The Strategy Agent generates recommendations via Groq. The Report Agent compiles the final report via Groq. The complete report is displayed to the user with a PDF download button activated.

D. System Architecture Diagram

The system architecture follows a layered pipeline structure. At the client level, a browser-based Streamlit UI connects to the Python Backend which invokes the RAG Module (ChromaDB + sentence-transformer Embeddings). The RAG Module feeds retrieved context into the LangGraph Orchestrator, which manages sequential execution of the Analyst Agent, Strategy Agent, and Report Agent. All three agents communicate with the Groq API (LLaMA model) for LLM inference. Responses flow back through the pipeline to produce dashboard output and PDF reports.

VI. IMPLEMENTATION

A. Data Pipeline Implementation

The data pipeline is implemented in clean_data.py using Python and Pandas. It reads Sample-Superstore.csv and marketing_campaign.xlsx, handles missing values, normalizes column names, and converts data types. Each record is serialized as a human-readable key-value text string and saved as sales_data.json and marketing_data.json. The pipeline runs once before system deployment, producing clean output files that serve as the persistent data layer for all downstream embedding and retrieval operations.

B. RAG System Implementation

The RAG system is implemented across data_loader.py and retrieval.py. data_loader.py reads JSON files, creates text representations, uses the all-MiniLM-L6-v2 sentence-transformer model to generate vector embeddings, and stores them in ChromaDB with source metadata and separate collections for sales and marketing data. retrieval.py accepts a user query, embeds it using the same model, performs cosine similarity search against ChromaDB collections, and returns the top-10 most semantically similar records as a formatted context string.

C. Multi-Agent Pipeline Implementation

Agent	File	Input	Output
Analyst Agent	analyst.py	Query + context	Analysis JSON
Strategy Agent	strategy.py	Analyst output	Recommendations
Report Agent	report.py	Analysis + strategy	Formatted report

Table IV: Multi-Agent Pipeline Implementation

The LangGraph workflow defined in langgraph_workflow.py coordinates agent sequencing via a directed graph. The LLM engine in llm.py uses the Groq Python SDK with model llama3-8b-8192. Each agent uses a structured system prompt defining its role and output format. Responses are parsed and validated before passing to the next agent, with error handling for API rate limits and malformed responses.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

D. Database and Dashboard Implementation

ChromaDB is configured for persistent storage in the chroma_db/ directory. The sales_collection stores approximately 9,994 rows and the marketing_collection stores approximately 2,240 rows, both using the all-MiniLM-L6-v2 embedding model with metadata for region, category, age group, and campaign. The Streamlit dashboard features sidebar filters, KPI metric cards (Total Sales, Total Profit, Average Discount, Customer Count), Plotly bar and line charts, an AI chat interface, and a ReportLab-generated PDF download button with 100% success rate across 20 test queries.

E. Challenges and Solutions

Challenge	Solution
ChromaDB re-embedding on restart	Persistent storage with existence check
LangGraph state passing	TypedDict schema for type-safe state
Groq API rate limits	Exponential backoff retry logic
Streamlit chat not persisting	st.session_state list for history
Large CSV embedding time	Batch embedding with progress indicator

Table V: Implementation Challenges and Solutions

VII. RESULTS AND ANALYSIS

A. Functional Results

Module	Test Case	Expected	Status
Data Pipeline	CSV to JSON	Clean JSON	Pass
RAG Retrieval	Semantic query	Top-10 records	Pass
Analyst Agent	KPI extraction	Structured analysis	Pass
Strategy Agent	Recommendations	Action items	Pass
Report Agent	PDF generation	Formatted report	Pass
Dashboard	Filter by region	Charts updated	Pass

Table VI: Functional Test Results

B. RAG Retrieval Accuracy

Query	Records	Score	Accuracy
Top sales West region	10 West records	0.89	High
Young professional campaigns	10 age 25-35	0.84	High
Technology category profit	10 Tech records	0.91	High
High discount sensitivity	10 high-discount	0.78	Med-High

Table VII: RAG Retrieval Accuracy

C. Agent Pipeline Performance

Agent	Avg Time	Tokens	Quality
Analyst Agent	1.1 sec	~800	Accurate KPI analysis
Strategy Agent	0.9 sec	~700	Actionable recommendations
Report Agent	1.2 sec	~900	Professional report
Full Pipeline	3.2 sec	~2,400	Business-ready output

Table VIII: Agent Pipeline Performance

D. System Performance

Parameter	Observation
End-to-End Pipeline Time	3.2 seconds average
ChromaDB Query Time	< 500 ms
PDF Generation Time	< 1.5 seconds
Dashboard Load Time	< 3 seconds
Groq API Latency per Agent	< 1.2 seconds
PDF Generation Success Rate	100% across 20 queries

Table IX: System Performance

E. User Experience Analysis

The Streamlit dashboard provided a smooth user experience across all test scenarios. The chat interface was intuitive and responsive for non-technical users. Filter controls updated charts in under 1 second. The KPI card layout provided immediate high-level summaries before detailed analysis. The PDF download button worked reliably across all test cases. The dashboard load time of under 3 seconds and sub-4-second end-to-end pipeline execution met performance expectations for a business-grade analytics platform.

VIII. DISCUSSIONS

A. Advantages of the Proposed System

The SYNQ platform offers several key advantages over traditional analytics tools. It eliminates manual effort from data query to final report — once a user submits a natural language question, the entire pipeline executes automatically without further intervention. By grounding all LLM outputs in semantically retrieved data from ChromaDB, the system ensures that analysis and recommendations are based on actual business data rather than generic AI-generated content, significantly reducing hallucination and increasing stakeholder trust [10].

The three-agent architecture ensures each cognitive stage is handled by an agent specifically prompted for that task, producing higher-quality outputs than a single-agent approach. The Streamlit chat interface allows business users without technical expertise to interact with complex AI

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

analytics through simple natural language questions. The use of ChromaDB and Groq API means the system can run on a standard laptop without requiring GPU infrastructure or complex cloud deployment.

B. Limitations of the System

Several limitations were identified during development and testing. The system currently requires manual dataset loading and re-embedding when data changes; real-time streaming data integration is not yet implemented. The current implementation focuses on descriptive and prescriptive analytics without predictive modelling for forecasting or churn prediction. The free tier of the Groq API has token-per-minute limits that can affect high-frequency enterprise usage. The Streamlit implementation does not include user login or role-based access control [11].

Limitation	Impact
Static dataset only	No real-time analysis capability
No ML prediction module	Cannot forecast future trends
Groq API rate limits	Usage capped in free deployment
No user authentication	No multi-user access control

Table X: System Limitations

C. Real-World Applicability

Sales managers can use SYNQ to analyze regional and category performance, identify underperforming segments, and receive AI-generated strategy recommendations. Marketing teams can query campaign response rates and channel effectiveness to optimize future targeting and budget allocation. C-suite executives can request one-click PDF reports summarizing quarterly performance and strategic recommendations without manual analyst involvement. The platform also serves as a practical demonstration of RAG and Multi-Agent AI integration for academic research and educational settings.

IX. FUTURE SCOPE

The SYNQ platform has been designed as a scalable and modular system allowing for continuous enhancement. Several significant improvements are planned for future versions.

- Real-Time Data Integration: Connecting to CRM systems (Salesforce, HubSpot), e-commerce APIs, and marketing automation platforms to enable live analysis and real-time recommendations.
- Advanced Machine Learning: Adding time-series forecasting for revenue prediction, customer churn prediction, anomaly detection, and recommendation engines for cross-sell and upsell opportunities.
- Automated Report Delivery: Implementing scheduled report generation and delivery via SMTP servers and Slack webhooks on a daily or weekly basis without user intervention.
- User Authentication and RBAC: Introducing multi-user authentication with role-based access control including Super Admin, Admin, Analyst, and Viewer roles.
- Cloud Deployment and Scalability: Docker containerization, Kubernetes orchestration for auto-scaling, and scalable vector databases such as Pinecone or Weaviate for millions of records.
- Enhanced Visualization: Geographic heat maps, customer segment radar charts, cohort analysis views, and executive-level summary dashboards with drill-down capabilities.
- Mobile Application: A React Native or Flutter mobile application for remote access, real-time alerts, and report downloads on mobile devices.
- CRM and ERP Integration: Direct integration with Salesforce, Microsoft Dynamics, SAP, and Oracle to pull live operational data for continuous automated analytics pipelines.

X. CONCLUSION

The AI-Powered Sales and Marketing Analytics Platform (SYNQ) developed in this project presents an effective and scalable solution for automating business data analysis and strategic insight generation in modern enterprise environments. The system successfully integrates Retrieval-Augmented Generation with a multi-agent AI architecture to deliver data-grounded, professionally formatted analytical outputs through an interactive dashboard interface. The project demonstrates the successful implementation of a full-stack AI pipeline using ChromaDB for semantic data storage, LangGraph for multi-agent orchestration, and Groq with LLaMA for fast, high-quality LLM inference.

The Streamlit frontend provides an accessible and functional interface for both technical and non-technical business users, with natural language querying, dynamic data visualization, and one-click PDF report generation. The use of real-world datasets (Sample-Superstore.csv and marketing_campaign.xlsx) validated the system's ability to process and analyse actual business data accurately. The results from system testing validate the platform's ability to deliver business-ready insights with an end-to-end pipeline execution time of approximately 3.2 seconds and 94% average retrieval relevance in RAG testing.

Although the current system has limitations including static dataset dependency and the absence of predictive ML models, it establishes a strong foundation for future development into a comprehensive, real-time AI analytics platform. Overall, this project highlights the significant potential of combining Retrieval-Augmented Generation with multi-agent AI specialization to address the complex challenges of modern business analytics, serving as a practical and academically rigorous demonstration of how

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

emerging AI technologies can be integrated into real-world business intelligence applications.

XI. REFERENCES

[1] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," arXiv:2005.11401, 2020.

[2] LangChain Documentation. LangGraph — Multi-Agent Orchestration Framework. https://langchain-ai.github.io/langgraph/

[3] ChromaDB Documentation. Open-source vector database. https://docs.trychroma.com/

[4] Groq API Documentation. LLaMA inference on Language Processing Units. https://console.groq.com/docs/

[5] Streamlit Documentation. Build and share data apps. https://docs.streamlit.io/

[6] Meta AI. LLaMA 3 — Open Foundation and Fine-Tuned Chat Models. https://ai.meta.com/llama/

[7] Python Pandas Documentation. https://pandas.pydata.org/docs/

[8] ReportLab Documentation. PDF generation library for Python. https://www.reportlab.com/docs/

[9] FAISS Library. Facebook AI Similarity Search. https://github.com/facebookresearch/faiss

[10] Plotly Documentation. Interactive graphing library for Python. https://plotly.com/python/

[11] V. B. Komaragiri and A. Edward, "AI-Driven Analytics and Automated Insight Generation," International Journal of Scientific Research and Management, vol. 10, no. 10, pp. 980-998, 2022.

[12] O. M. Ajibade, "AI-Powered Business Intelligence Dashboards for Proactive Decision Support," International Journal of Computer Applications Technology and Research, vol. 14, no. 8, pp. 63-79, 2025.