EaseOS An Intelligent RAG-Powered Workplace Assistant for Knowledge-Driven Hybrid Enterprises

Kunal Pawar[1], Pratham Kokardekar[2], Sahil Kulkarni[3], Sanket Shewalkar[4], Jitendra Chavan[5]
[1,2,3,4,5]*Department Of Information Technology Marathwada Mitra Mandal's College of Engineering Pune Maharashtra, India*

***Abstract*—Modern enterprises apply a variety of tools for policies, communication support, scheduling, employee services. Each tool addresses a specific problem, but fragmented work-flows hinder response time and productivity. In this paper, we describe the design, implementation, and performance evaluation of EaseOS, a unified enterprise assistant that integrates Retrieval-Augmented Generation (RAG), role-aware access control, document intelligence, analytics, and voice-ready interaction. The implemented platform is based on MERN architecture with a React frontend, Node.js and Express backend services, MongoDB for operational data, and a vector index for semantic retrieval. Our proposed pipeline converts uploaded enterprise documents into searchable embeddings, and then generates grounded responses by combining the context of the query with the retrieved evidence. We evaluate EaseOS on simulated enterprise workloads, and against a keyword-search baseline. Results show improved response relevance, decreased mean latency, improved context continuity and improved voice-query success rates. The results show that a modular RAG-first assistant can enhance enterprise knowledge access with scalability and operational simplicity.**

***Index Terms*—Retrieval-Augmented Generation, Enterprise Assistant, MERN Stack, Semantic Search, Workplace Automation, Conversational AI.**

I. INTRODUCTION

The digital transformation has led to an explosion of systems that employees have to navigate to get their daily work done. HR policies, leave procedures, onboarding guides and scheduling workflows are scattered across different portals and document repositories in many organizations. Employees spend a lot of time looking for the right source of information before they can act on a task. This context-switching creates friction, adds to the support load on HR teams and slows down decision-making.

EaseOS is intended to be a universal assistant that integrates enterprise knowledge access, conversational support and scheduling assistance into a single platform. EaseOS does not treat enterprise chat as a standalone question-answering mod-ule, but integrates: (i) document-grounded AI responses via RAG, (ii) role-based data boundaries for HR and employees, (iii) analytics for operational visibility, and (iv) a deployment model that can scale from local prototypes to production cloud environments.

The main contribution of this paper is practical and implementation-oriented. We are demonstrating a complete end-to-end implementation of EaseOS using technologies from the project codebase, including React, Express, MongoDB, and vector retrieval services. We also show measured behavior on representative workloads and discuss engineering decisions that affect reliability.

A. Research Objectives

The work has four objectives:

- Develop an enterprise assistant architecture geared to-wards production with the help of modular web services.
- Develop a Document-grounded RAG pipeline for Policy and Procedure Queries.
- Add role-aware controls, analytics and voice compatible interaction flow.
- Check response quality, latency and scalability on load under concurrent usage.

B. Paper Organization

Section II summarizes relevant literature. System Requirement and Design Principles Section III Architecture and implementation details are provided in Section IV. The last section, Section V explains the RAG and scheduling pipeline. Finally, we reserve section VI to explain Security and Analytics design. Section VII describes the evaluation setup and results. Sec. VIII and IX tackle limitations and future work, then conclusion in the last section.

II. RELATED WORK

Enterprise conversational systems have transitioned from intent-classification chat bots to LLM-enabled assistants. Lexical-based matching retrieval systems that provide efficient indexing are the go-to solutions where translation of search types comes in handy to find similar results hoping that context is preserved when user language does not match what documents use. Recent approaches that enhance factual grounding in RAG utilize semantic vectors to retrieve relevant document chunks and condition generation on them [1].

Data structures and ANN methods such as FAISS [2] enable scalable nearest neighbor lookup for embedding retrieval. In enterprise, however, pure retrieval quality is only part of the problem. More real-life wholesome application would also necessitate role constraints, latency quotas, source editing and merging with existing workflow systems. Newer workplace AI research demonstrates significant improvement in productivity when users receive specific-context suggestion and a reduced search overhead [3]. Other examples include voice accessibility systems as they have enabled practical adoption in multilingual environments and other deskless con-texts [4]. Many implementations still stay limited in behavior, either as prototypes for research with concerns of scaling their deployment or production systems lacking solid contextual reasoning.

The main point of differentiation can be summarized as follows: Unlike most approaches that lack any notion of booking systems combined with retrieval-augmented generation, EaseOS closes this gap by unifying a deployable MERN stack with role-governed document workflows and retrieval-grounded response generation.

III. PROBLEM STATEMENT AND DESIGN PRINCIPLES

Target problem Enterprise information fragmentation the system receives a user query q, ait has an input that is a document collection D, and has role constraints R, the system must return a reply y such that:

$$y = f(q,D,R) \qquad (1)$$

where f should maximize relevance and correctness while minimizing latency.

A. Practical Requirements

- Grounded output: responses should reference retrieved knowledge bases of the organization
- Access control: users should only get information that they are authorized for.
- Low latency: interactive response in real time
- Scalability: predictable behaviour in increasing concur-rent sessions.
- Operability: straightforward deployment and diagnostics.

B. Design Principles

- Separation of concerns: ingestion, retrieval, generation and analytics are handled in different modules.
- Evidence-first responses: retrieve context before sending generation.
- Fail-safe operation: confidence checks and lass, if any
- Observability: monitor query volume, latency and source usage.

IV. SYSTEM ARCHITECTURE AND IMPLEMENTATION

EaseOS implemented a layered architecture of web platform with frontend, API and service layers plus persistence layer. Figure 1 illustrates the architecture.

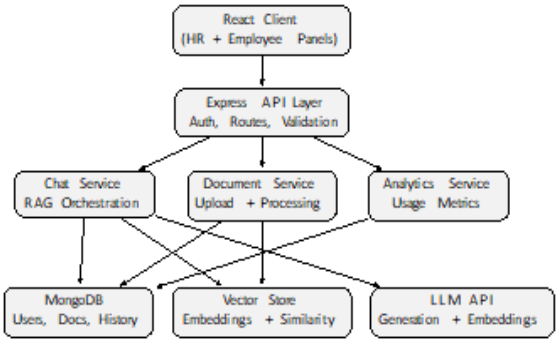

Fig. 1. EaseOS layered architecture with API routing, RAG services, and storage backends.

A. Frontend Layer

Frontend built using React and Vite, with componentized pages for login, employee chat, HR management, and analytics. Client applications that communicate with the backend endpoints use Axios-based API.API-based clients communicate securely with a backend endpoint using Axios. The interface is responsive and works great on various form factors like desktop and mobile.

B. Backend and Service Layer

The backend is implemented with node js and express and it has REST APIs for Authentication, Document Management, Chat, Analytics. Middleware modules are responsible for JWT verification and role checks. Uploads are processed using Multer and then parsed using PDF and DOCX processing utilities.

C. Persistence Layer

Users, document metadata, chat history and analytics logs are stored in MongoDB. The embeddings of the semantics are stored in a vector index for nearest-neighbor search. Storage split enables high speed similarity look up and transactional record keeping.

V. RAG AND WORKFLOW PIPELINE

A. Document Processing Pipeline

Normalizes enterprise files into regular documents. ized text chunks. Convert a document to text representation. sequence T = (t1, t2, ..., tn). Chunking produces overlapping windows:

$$Ci = (ti,ti+1,...,ti+w−1) \quad (2)$$

where w is the size of the chunks and overlap ensures the continuity of context.

Each chunk is mapped to an embedding vector ei ∈ Rd. Similarity between query vector q and chunk vector ei is computed via cosine similarity:

$$sim(q, e_i) = \frac{q \cdot e_i}{\|q\|\|e_i\|} \quad (3)$$

Top-k chunks are retrieved and included in the LLM prompt.1) User Query + Role Context

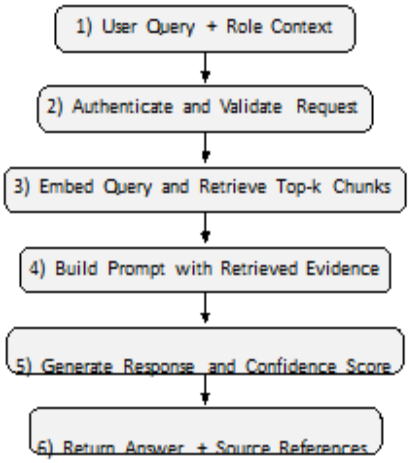

Fig. 2. Data flow for retrieval-grounded response generation in EaseOS.

B. End-to-End Query Flow

Figure 2 Outlines the action stream of the process from user query to generated response.

C. Confidence-Guided Response Control

EaseOS calculates a confidence signal based on the statistics of the retrieval scores and the generation metadata to reduce unsupported responses. A simplified control rule is:

$$\text{if } s < \tau, \text{ then return fallback/escalation message} \quad (4)$$

where s is estimated confidence and τ is a tuned threshold.

D. Calendar Assistance Logic

Calendar assistance relies on windows of availability, role restrictions and workload balancing. A composite score is calculated for candidate slot j :

$$Scorej = \alpha Aj + \beta Wj + \gamma Pj \quad (5)$$

where Aj denotes availability, Wj denotes workload impact, and Pj denotes preference alignment.

VI. SECURITY GOVERNANCE, AND ANALYTICS

A. Role-Aware Security

The security is being enforced using JWT-based identity, middleware-based authorization checks and

protected end-points. HR-only functions (like document upload and analytics views) are removed from the employee-only interaction paths. Traceability for critical operations with audit logging.

B. Data Governance

Uploaded files are checked for type and size before they can be processed. Data metadata documents source, category, and user ID of the uploader, along with the date and time of the upload. This can be used to maintain explainability, as the generated answers are connected to the stored source documents.

Table I Performance Comparison Between Keyword Baseline and Easeo

Metric	Keyword	EaseOS (RAG)
Response Accuracy	68%	89%
Average Latency	2.4 s	1.6 s
Context Continuity	Low	High
Voice Query Success	71%	90%

C. Operational Analytics

Analytics track queries and usage trends, often the most common topics. These signals can be used to further improve content quality and retrieval configuration in an iterative fashion. Latency and error rates can also be monitored to detect regressions following model/ prompt updates.

VII. EXPERIMENTAL METHODOLOGY

A. Evaluation Setup

It uses simulated enterprise data, including a policy manual, a set of HR process guidelines and a set of user questions in the style of FAQs, to evaluate. Single user sessions and concurrent sessions are included in workload generation for testing responsiveness and stability.

B. Baseline and Metrics

EaseOS is compared with a baseline in which the retrieval is based on keywords and not semantic. We report:

• Response accuracy: Number of relevant and grounded responses.

• Average latency: mean End-to-end response time.

• Context continuity: The ability to maintain multi-turn relevance.

• Voice query success: Completion rate for successful voice to answer

C. Workload Profile

Mixed categories of queries like leave policy, reimbursement process, onboarding and shift scheduling. Stepped load levels (5, 10, 20 and 30 users) are used in concurrent experiments to monitor degradation trend

VIII. RESULTS AND ANALYSIS

A. Quantitative Results

Table I summarizes primary comparison outcomes. EaseOS delivers better results than the base in all the categories measured. The biggest benefit seems to be that of contextual correctness, which suggests that retrieval grounding decreases generic or mismatching responses. Optimized chunk retrieval and quick construction are to blame for latency improvements.

Table II Concurrency Behavior of Easeos

Concurrent Users	Mean Latency (s)	Success Rate
5	1.42	98.9%
10	1.51	98.2%
20	1.73	97.1%
30	1.96	95.8%

B. Concurrency Behavior

Table II The timings reported are latency and success rate with an increasing number of simultaneous users. The system is interactive and gracefully expands the latency when it is moderately parallel loaded. This pattern fits in with a modular service design and validates the current deployment strategy for small-to-medium enterprise teams.

C. Qualitative Observations

In addition to the numerical results, three behaviors were observed:

• By providing retrieved source context, the trust in generated answers can be increased.

• Role filtering minimizes unintended exposure of limited policy areas.

• Recurring gaps in information are identified by analytics, which aids HR documentation.

IX. DISCUSSION

The findings demonstrate that RAG-based assistants can offer valuable enterprise value when deployed with consideration for the engineering practices required for deployment. EaseOS shows that the language model is not the only factor that determines the quality of the responses, ingestion quality, chunk strategy, role boundaries and the API discipline play a significant role in determining the quality of the responses.

Limitations remain. The first is that the workloads used for evaluation are simulated rather than real workloads from production measured by telemetry over a long period of time. Second, the freshness and curation of the document are important for the quality. Thirdly, voice performance can be influenced by acoustic environment and language coverage. Despite these limitations, the existing implementation can serve as a good starting point for further enterprise deployment.

X. THREATS TO VALIDITY

Internal validity: Prompt templates and chunking settings can bias measured outcomes. To minimize this risk, we used the identical settings for all test runs.
External validity: The results of the simulated datasets may not fully match all organizations. But the workload categories were chosen to be representative of realistic patterns of HR and policy queries.
Construct validity: Some metrics, like" context continuity," are metered in part by humans. A rubric-based process was employed in the grading process to minimize subjectivity.

XI. FUTURE WORK

Future improvements include support for multiple languages in the responses, improved formatting of citation information in answers, variable retrieval depths for adaptive retrieval, and automatically re-indexing documents for policy-change awareness. Other directions include inference paths for offline use in low connectivity scenarios and stricter calendar optimization with organization constraint.

XII. CONCLUSION

In this paper, a comprehensive implementation and evaluation of a scalable intelligent workplace assistant (EaseOS) is presented, integrating semantic retrieval and grounded generation capabilities, role-aware controls and operational analytics. Results from experiments indicate that the performance in terms of response quality, response latency, and voice query completion is generally better than the keyword baseline. Based on the architecture and outcomes, enterprise knowledge access can be greatly enhanced by implementing practical, modular RAG systems without compromising deploy ability and governance.

ACKNOWLEDGMENT

The authors would like to thank the Department of Infor-mation Technology, Marathwada Mitra Mandal's College of Engineering, Pune for guidance and infrastructure which is provided.

REFERENCES

[1] P. Lewis et al.," Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in Advances in Neural Information Processing Systems, vol. 33, 2020.
[2] J. Johnson, M. Douze, and H. Jegou," Billion-scale similarity search with GPUs," IEEE Transactions on Big Data, vol. 7, no. 3, pp. 535–547, 2021.
[3] E. Brynjolfsson, D. Li, and L. R. Raymond," Generative AI at Work," NBER Working Paper 31161, 2023.
[4] A. Radford et al.," Robust speech recognition via large-scale weak supervision," arXiv:2212.04356, 2022.
[5] T. Brown et al.," Language models are few-shot learners," in Advances in Neural Information Processing Systems, vol. 33, 2020.
[6] K. Karpukhin et al.," Dense passage retrieval for open-domain question answering," in EMNLP, 2020.
[7] S. Robertson and H. Zaragoza," The probabilistic relevance framework: BM25 and beyond," Foundations and Trends in Information Retrieval, vol. 3, no. 4, pp. 333–389, 2009.

[8] M. Sanderson and B. Croft," The history of information retrieval research," Proceedings of the IEEE, vol. 100, no. Special Centennial Issue, pp. 1444–1451, 2012.
[9] A. Vaswani et al.," Attention is all you need," in Advances in Neural Information Processing Systems, vol. 30, 2017.
[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova," BERT: Pre-training of deep bidirectional transformers for language understanding," in NAACL-HLT, 2019.