**Enhancing Jira Ticket Retrieval with Knowledge Graphs**

**Dissertation Report (Mid Semester)**

Submitted in partial fulfillment of the requirements of the Degree: M.Tech in Artificial Intelligence and Machine Learning

By  
**Boda Dinesh Kumar Reddy**  
BITS ID No.: 2023AA05149

Under the supervision of  
**Shivajith Mutteal K**  
Engineer, Senior Staff  
Qualcomm India Pvt. Ltd.

Birla Institute of Technology and Science, Pilani (Rajasthan) INDIA  
(June, 2025)

### Abstract

This dissertation explores the integration of Knowledge Graphs (KGs) into Jira ticket retrieval systems to enhance the performance of existing Retrieval-Augmented Generation (RAG) pipelines. Traditional vector-based RAG systems primarily capture surface-level semantic similarity but struggle with deeper contextual understanding and relationship modeling between issue attributes like reporter, assignee, project, and issue type.

The proposed solution involves constructing a domain-specific Knowledge Graph from historical Jira data, modeling explicit relationships among ticket entities. This KG-enhanced retrieval approach supports complex, multi-entity queries and improves explainability by offering traceable reasoning paths for each result. The Knowledge Graph is integrated with an existing RAG system, enabling hybrid retrieval mechanisms that leverage both vector-based semantic similarity and graph-based contextual reasoning.

This mid-semester report details the motivation, theoretical framework, literature review, methodology, and system design for the project, along with a roadmap for future work.

**Keywords:** Knowledge Graph, Retrieval-Augmented Generation, Jira Ticket Retrieval, Explainable AI, Semantic Search

### List of Abbreviations

| Abbreviation | Description |
| --- | --- |
| KG | Knowledge Graph |
| RAG | Retrieval-Augmented Generation |
| MRR | Mean Reciprocal Rank |
| XAI | Explainable AI |
| CRM | Customer Relationship Management |

### Table of Contents

1. Chapter 1: Introduction
2. Chapter 2: Literature Review
3. Chapter 3: Research Methodology
4. Chapter 4: System Design and Architecture
5. Directions for Future Work
6. References

## Chapter 1: Introduction

### 1.1 Background

Modern software development environments rely on issue tracking systems like Jira to manage bugs, feature requests, and support tickets. Retrieving relevant historical tickets for a given query assists in faster issue triaging and knowledge reuse. Conventional vector-based RAG systems perform semantic similarity searches using dense embeddings but lack mechanisms to model explicit relationships between ticket attributes.

Knowledge Graphs (KGs) provide a structured representation of entities and their interconnections, making them ideal for improving retrieval precision and explainability in such systems. Integrating KGs with RAG frameworks can improve the handling of complex, multi-entity queries and increase user trust through transparent reasoning paths.

### 1.2 Problem Statement

The current vector-based Jira ticket retrieval system struggles to capture explicit relationships between ticket attributes and lacks explainability. It cannot effectively handle multi-entity, relationship-driven queries, leading to suboptimal ticket matches and slower issue resolution.

### 1.3 Objectives

* To construct a domain-specific Knowledge Graph from Jira ticket data.
* To integrate the KG with an existing RAG framework.
* To enable complex, multi-entity query handling.
* To improve retrieval accuracy and explainability.
* To evaluate the system using metrics like MRR and user feedback.

### 1.4 Scope of the Research

The project involves designing, developing, and integrating a Knowledge Graph with a RAG system for Jira ticket retrieval, evaluated through offline experiments. On-premise deployment options and future multilingual support are scoped for post-midsem phases.

### 1.5 Significance of the Study

The system will enhance retrieval relevance, reduce triaging time, and improve user trust through explainable reasoning paths. The modular design ensures scalability and extensibility for enterprise environments.

## Chapter 2: Literature Review

**Existing work** highlights limitations in purely vector-based RAG pipelines for knowledge-intensive and explainable retrieval. Several studies demonstrated that KG integration improves contextual understanding, multi-hop reasoning, and transparency:

* Linders & Tomczak (2025) used KG-extended RAG for QA systems improving explainability.
* Sanmartin (2024) developed KG-RAG to bridge unstructured text and structured knowledge.
* Xu et al. (2024) applied KG-RAG to customer service, achieving notable improvements in MRR.

These findings provide a theoretical foundation for this dissertation, validating KG integration for Jira ticket retrieval systems.

## Chapter 3: Research Methodology

### 3.1 Research Design

A phased methodology:

1. Literature review and KG schema design
2. Prototype KG construction from Jira data
3. Integration with RAG framework
4. Query interface development
5. Evaluation and refinement

### 3.2 Data Collection Methods

Historical Jira data containing tickets, issue types, components, assignees, and relationships are used to construct the KG.

### 3.3 Tools and Frameworks

* Neo4j for Knowledge Graph modeling.
* Existing vector-based RAG system.
* Python libraries for graph integration.

### 3.4 Evaluation Metrics

* **Mean Reciprocal Rank (MRR)** for retrieval effectiveness.
* User feedback for qualitative insights.

## Chapter 4: System Design and Architecture

### 4.1 System Overview

The system integrates a Knowledge Graph with an existing RAG framework to enhance Jira ticket retrieval. Queries are executed using both vector similarity and graph-based filtering.

### 4.2 System Architecture Diagram

+--------------------+  
 | User Query |  
 +--------------------+  
 |  
 +-----------+------------+  
 | |  
 Vector Similarity KG-based Filtering  
 | |  
 +-----------+------------+  
 |  
 Merged Results  
 |  
 Final Retrieval

### 4.3 Components

* **Knowledge Graph:** Neo4j-based, models ticket entities and relationships.
* **RAG Pipeline:** Performs vector-based retrieval.
* **Hybrid Retrieval Module:** Merges KG-filtered results with vector similarity results.

## Directions for Future Work

* KG population expansion with multilingual and cross-project tickets.
* Integration of explainability modules for result justification.
* Performance benchmarking against pure RAG systems.
* CRM system integration for customer issue management.
* On-premise deployment implementation.

## References

1. Linders & Tomczak, (2025). Knowledge Graph-extended Retrieval Augmented Generation for Question Answering. *arXiv:2504.08893*
2. Sanmartin, (2024). KG-RAG: Bridging the Gap Between Knowledge and Creativity. *arXiv:2405.12035*
3. Xu et al., (2024). Retrieval-Augmented Generation with Knowledge Graphs for Customer Service. *arXiv:2404.17723*

*End of Mid Semester Report.*