RESUMEN DEL TRABAJO FIN DE GRADO

Universidad de Salamanca

Escuela Politécnica Superior de Zamora

Título: Cerebro Digital: Aplicación de notas basada en Markdown con integración RAG y análisis semántico topológico.

Autor: Oswaldo Enrique González Suárez

Tutores: María Luisa Pérez Delgado (USAL), Nicolás García Martín (HP SCDS), Adrián Fernández Roldan (HP SCDS)

Titulación: Grado en Ingeniería Informática en Sistemas de Información

Fecha: Julio de 2026

Resumen Ejecutivo

El presente Trabajo Fin de Grado describe el diseño, desarrollo y despliegue de *Cerebro Digital*, una plataforma de Gestión del Conocimiento Personal (PKM) que incorpora procesamiento de lenguaje natural de forma nativa y estrictamente local. Desarrollado bajo la tutela tecnológica de la empresa HP SCDS, el sistema implementa una arquitectura de microservicios eficiente. El *backend*, construido en Go (Golang), gestiona la concurrencia y la persistencia en MongoDB; mientras que el *frontend* en React proporciona una experiencia interactiva bajo el paradigma *Glassmorphism*. El proyecto destaca por su motor de Generación Aumentada por Recuperación (Graph-RAG híbrido) y el uso de Inteligencia Artificial en el borde (*Edge AI*), erradicando los riesgos de brechas de seguridad en la nube y garantizando la soberanía de los datos.

Palabras clave: PKM, Edge AI, RAG, Teoría de Grafos, Golang, React, DevOps.

Abstract

This capstone project details the design, development, and deployment of *Cerebro Digital*, a Personal Knowledge Management (PKM) platform that incorporates Natural Language Processing natively and strictly locally. Developed under the technological mentorship of HP SCDS, the system leverages a highly efficient microservices architecture. The backend, built in Go (Golang), manages concurrency and persistence in MongoDB; meanwhile, the React frontend provides an interactive experience following the *Glassmorphism* design paradigm. The project stands out for its hybrid Graph-Retrieval-Augmented Generation (Graph-RAG) engine and the use of Edge AI, eliminating the risks of cloud security breaches and ensuring full data sovereignty.

Keywords: PKM, Edge AI, RAG, Graph Theory, Golang, React, DevOps.

1. Introducción y Objetivos

En la actual era de la información, los profesionales de la ingeniería de software y sistemas de información se enfrentan a un volumen de documentación que crece de forma exponencial. La gestión tradicional de notas, basada en estructuras jerárquicas rígidas, ha demostrado ser insuficiente para emular la forma en la que el cerebro humano asocia ideas de manera orgánica y reticular. Si bien existen herramientas que introducen los enlaces bidireccionales, la mayoría carecen de analítica avanzada o dependen de APIs de Inteligencia Artificial alojadas en la nube, lo que compromete la privacidad de la información corporativa o del código fuente.

El objetivo principal de *Cerebro Digital* es resolver esta problemática mediante una plataforma que devuelva el control total de los datos al usuario. Para ello, se combina la agilidad del formato de texto plano (Markdown) con la potencia de los Modelos de Lenguaje Grandes (LLMs) ejecutados de manera estrictamente local (*Edge Computing*), garantizando el cumplimiento normativo del Reglamento General de Protección de Datos (RGPD).

2. Arquitectura del Sistema y Stack Tecnológico

El sistema se ha diseñado siguiendo los principios de la **Arquitectura Limpia** (*Clean Architecture*), estructurándose en tres capas funcionales fuertemente desacopladas para garantizar su mantenibilidad y escalabilidad:

- **Capa de Presentación (Frontend):** Desarrollada como una *Single Page Application* (SPA) utilizando React y Vite. Se elude el uso de gestores de estado complejos, apoyándose en *Hooks* nativos y patrones de Actualización Optimista (*Optimistic UI*) para enmascarar la latencia de red. El diseño visual implementa el paradigma *Glassmorphism* mediante Tailwind CSS.

- **Lógica de Negocio (Backend):** El núcleo del servidor RESTful está construido en Go (Golang). Aprovechando las mejoras de enrutamiento de la librería estándar introducidas en Go 1.22, se ha prescindido de *frameworks* de terceros. La elección de este lenguaje compilado responde a la necesidad imperativa de gestionar operaciones concurrentes eficientes para el cálculo matemático de distancias vectoriales.

- **Persistencia Documental:** Se ha implementado MongoDB como motor de base de datos NoSQL. Su estructura BSON permite almacenar de manera ágil el texto desestructurado en Markdown, junto con las matrices numéricas de precisión decimal (vectores) requeridas por la Inteligencia Artificial.

3. Integración de Inteligencia Artificial (Graph-RAG)

El núcleo analítico del proyecto reside en la implementación de una estrategia de Generación Aumentada por Recuperación híbrida (**Graph-RAG**). Para superar las limitaciones de hardware de consumo y evitar el desbordamiento de la memoria VRAM, el sistema orquesta modelos fuertemente cuantizados a través del motor local Ollama.

El flujo de procesamiento semántico opera en dos fases críticas. En primer lugar, la vectorización del conocimiento se realiza utilizando el modelo especializado *Nomic-Embed-Text*, traduciendo los documentos a coordenadas espaciales multidimensionales. Posteriormente, la recuperación de información se ejecuta computando la Similitud del Coseno directamente en la memoria del servidor Go. Esta aproximación híbrida inyecta tanto los resultados de la métrica de distancia vectorial como las dependencias topológicas (notas enlazadas) directamente en la ventana de contexto del modelo de inferencia *Qwen3 (0.6b)*. Adicionalmente, se aplica un Ïmpulso Léxico.ªlgorítmico para coincidencias exactas, mitigando drásticamente las alucinaciones intrínsecas de los LLMs.

4. Representación Topológica y Búsqueda Neural

Para modelar computacionalmente la red de conocimiento, *Cerebro Digital* procesa el contenido redactado por el usuario extrayendo metadatos y adyacencias mediante Expresiones Regulares (Regex). Las referencias cruzadas (*Wiki-links*) se abstraen utilizando los fundamentos de la Teoría de Grafos.

La visualización interactiva en el cliente se estabiliza iterativamente mediante un motor de *Force-Directed Graphs*. El cálculo de las fuerzas de atracción (Ley de Hooke) y repulsión (Ley de Coulomb) se realiza utilizando la aproximación algorítmica de **Barnes-Hut**, reduciendo la complejidad espacial. Como innovación en la interfaz, se ha implementado un efecto de "Búsqueda Neural", el cual altera las propiedades físicas y visuales de los nodos en tiempo real mediante funciones trigonométricas cuando intersectan con el espacio semántico de la consulta del usuario.

5. Aseguramiento de Calidad y Entornos (DevOps)

En el contexto de la Ingeniería de Software moderna, garantizar la reproducibilidad del entorno de ejecución es indispensable. Se ha adoptado un ecosistema estrictamente determinista empleando **Nix** y **Direnv**. Esta infraestructura elimina el antipatrón de divergencia de configuraciones, asegurando una reproducibilidad binaria absoluta del código base sin contaminar el sistema operativo anfitrión, mientras que Docker se encarga de la orquestación aislada de la persistencia (MongoDB).

La viabilidad del sistema ha sido verificada mediante rigurosos *benchmarks* de rendimiento sobre hardware de consumo estándar. Los algoritmos de extracción y parseo en Go demostraron la capacidad de procesar iterativamente hasta 80.000 documentos por segundo sin bloqueos de hilo, mientras que la latencia de inferencia RAG completa oscila de manera estable entre los 5 y 10 segundos.

6. Conclusiones

El desarrollo de *Cerebro Digital* demuestra la viabilidad técnica y operativa de ejecutar sistemas de gestión del conocimiento impulsados por IA de forma estrictamente local (*Edge Computing*). El proyecto resuelve la principal deficiencia de las plataformas actuales al asegurar una soberanía absoluta de los datos, demostrando que mediante un uso arquitectónicamente eficiente de lenguajes concurrentes como Go y modelos cuantizados, es posible democratizar el acceso a la Inteligencia Artificial semántica en equipos ofimáticos estándar.