Primer proyecto Indice Invertido

Estructura de datos - EIF205

Prof. Steven Brenes Chavarría

Escuela de Informática

Estudiantes:

Eslyn Andrés Jara Calderón

Jean Carlo Blanco Mora

Mishelle Rojas Chacón

2025



**Definición general del proyecto de software**

El programa consiste en un motor de búsqueda basada en la estructura de datos de índice invertido, para esto se utilizó el algoritmo de ordenamiento *Radix* y listas circulares. Con esto el programa es capaz de buscar lo que el usuario desee, y hacer una lista con el top 10 de los documentos con mayor similitud, con el número del documento, la similitud con la búsqueda y el URL del documento.

**Herramientas**: Para el proyecto se utilizó el lenguaje de programación C# (.NET Framework), con el IDE Visual Studio y para el manejo de versiones y trabajo en conjunto se utilizó GitHub y las fuentes de datos fueron archivos de texto/documentos brindados por el docente.   
  
**Arquitectura**: La arquitectura se diseñó en capas modulares, permitiendo separar las responsabilidades y facilitando la comprensión del funcionamiento del motor de búsqueda. Los módulos principales son:

**Ingreso de documentos:** Se encarga de leer los archivos de texto y les hace un preprocesamiento básico, separar términos, elimina caracteres especiales y normaliza las palabras. Este módulo devuelve una colección de documentos listos para indexar

**Índice invertido**: Este es el encargado de construir la estructura principal del proyecto. Aquí se almacena cada término junto con la lista de documentos en los que aparece y su frecuencia de aparición. Estas listas se gestionan con listas circulares para recorrer los documentos asociados a un término sin perder referencias y de manera más eficiente.

**Procesamiento de consultas:** Aquí se recibe la búsqueda del usuario y se localizan y recopilan los términos en el índice invertido. Cuando completa la recopilación de los términos calcula la similitud entre cada documento y la consulta realizada, generando una lista con los resultados.

**Ordenamiento**: Se aplica el algoritmo de Radix Sort para organizar los documentos conforme a su similitud, dicho ordenamiento es realizado del mayor al menor, haciendo que los resultados más relevantes sean priorizados automáticamente.

**Presentación y resultado**: Se extrae la lista ya ordenada y se construye el Top 10 de los documentos más relevantes, y se muestra el número del doc, el porcentaje de similitud con la consulta y el URL asociado a dicho documento. Este módulo devuelve y presenta la lista final para el usuario final.

**Diseño y modelo de datos**: Con un diseño centrado en el uso de estructura de datos para mejorar la optimización de la búsqueda, con una lógica basada en el índice invertido y uso de listas circulares.  
  
***Model***: En el modelo se manejan los documentos y se meten a la lista para poder hacer el índice invertido y saca las frecuencias de aparición.