

TAREA 4: Grape Loop

Manuel González<sup>†</sup>, Pablo Gómez<sup>†</sup>, Ayrton Morrison<sup>†</sup> y Emmanuel Velásquez<sup>†</sup>

<sup>†</sup> Universidad de Magallanes

Este informe fue compilado el 14 de diciembre de 2024

Resumen

Keywords:

■ Índice

1	Introducción	2
2	Objetivo principal	2
3	Planteamiento del desarrollo del proyecto	3
4	Implementación	4
5	Gestión del equipo de trabajo	5
6	Posibles Mejoras a futuro	6
7	Ejemplo de uso	7
8	Conclusiones	8

## 1. Introducción

La presente tarea tiene como propósito consolidar los conocimientos adquiridos mediante el diseño e implementación de un sistema práctico. En esta ocasión, el proyecto se centra en el desarrollo de una red social simulada, un entorno que permitirá implementar y evaluar diversos algoritmos de recomendación y técnicas relacionadas con redes sociales. Este entorno incluye el uso de estructuras de datos avanzadas como grafos y listas enlazadas, integrando también algoritmos de similitud de Jaccard. El objetivo general es simular un sistema que facilite recomendaciones personalizadas basadas en la importancia y similitud de nodos en una red, resolviendo desafíos técnicos relacionados con la gestión dinámica de datos en estructuras complejas.

## 2. Objetivo principal

El objetivo principal es diseñar e implementar un sistema funcional que permita gestionar perfiles de usuario, conexiones entre ellos y operaciones asociadas como publicaciones, búsquedas y recomendaciones. Este proyecto busca aplicar conceptos fundamentales de estructuras de datos y algoritmos avanzados para resolver problemas reales relacionados con la gestión y optimización de redes sociales simuladas.

### 3. Planteamiento del desarrollo del proyecto

El desarrollo de este proyecto se centra en la implementación de una red social simulada basada en grafos. Los nodos representan perfiles de usuario y los bordes, sus conexiones (seguidores y seguidos). Los algoritmos principales empleados incluyen:

- 1.- Similitud de Jaccard: Implementado para calcular la semejanza entre los intereses o conexiones de diferentes usuarios.
- 2.- Recomendaciones personalizadas: Basadas en conexiones compartidas entre usuarios y análisis de interacciones.

El proyecto incorpora una estructura modular, separando tareas como el almacenamiento de datos en archivos, la simulación de conexiones entre usuarios y el cálculo dinámico de métricas para optimizar el rendimiento.

## 4. Implementación

La implementación se desarrolla en etapas:

### 1.- Estructuras de datos:

- Uso de grafos para modelar las relaciones entre usuarios.
- Implementación de listas enlazadas y tablas hash para gestionar datos asociados a cada usuario.

### 2.- Algoritmos:

- Adaptación del algoritmo de Dijkstra para encontrar rutas más cortas.
- Cálculo de la similitud de Jaccard para sugerencias basadas en intereses comunes.

### 3.- Manejo de datos:

- Almacenamiento estructurado en directorios para cada usuario.
- Archivos separados para guardar seguidores, seguidos y datos generales de los usuarios.

### 4.- Generadores dinámicos:

- Creación automática de usuarios y simulación de publicaciones.
- Generación de conexiones aleatorias y recomendaciones personalizadas.

### 5.- Interfaz de interacción:

- Desarrollo de una interfaz de línea de comandos para manejar la creación, modificación y búsqueda de usuarios.

- Ejecución a través de parámetros para una interacción dinámica.

## 5. Gestión del equipo de trabajo

El equipo de trabajo está compuesto por los siguientes integrantes: Ayrton Morrison, Emmanuel Velásquez, Manuel González y Pablo Gómez. La división de responsabilidades se organizó en dos grupos:

Equipo de programación: Responsable del diseño y desarrollo del código del sistema.

Equipo de documentación: Encargado de la redacción del informe y la elaboración de diagramas y documentación adicional.

## 6. Posibles Mejoras a futuro

En esta sección se describen algunas posibles mejoras que se podrían implementar en el futuro para optimizar y ampliar el proyecto:

- **Optimización del rendimiento:** Revisar y mejorar el código para reducir el tiempo de ejecución y el uso de recursos.
- **Ampliación de funcionalidades:** Añadir nuevas características que puedan ser útiles para los usuarios, basadas en el feedback recibido.
- **Mejora de la documentación:** Asegurarse de que toda la documentación esté actualizada y sea fácil de entender para nuevos desarrolladores.
- **Pruebas automatizadas:** Implementar un conjunto completo de pruebas automatizadas para garantizar la estabilidad y la calidad del código.
- **Interfaz de usuario:** Mejorar la interfaz de usuario para hacerla más intuitiva y fácil de usar.
- **Seguridad:** Revisar y mejorar las medidas de seguridad para proteger los datos y la integridad del sistema.

## 7. Ejemplo de uso

A continuación, se presentan algunos ejemplos prácticos de cómo los usuarios podrían interactuar con el sistema:

### 1.- Recomendación de conexiones:

- El usuario "Juan" se registra en la red social y sigue a tres perfiles.
- El sistema, utilizando la similitud de Jaccard, sugiere a "Pedro" como posible conexión debido a intereses comunes.

### 2.- Simulación de publicaciones:

- "Ana" publica un mensaje en su perfil.
- El sistema utiliza PageRank para determinar la relevancia de "Ana" y mostrar su publicación en la página principal de sus seguidores.

### 3.- Búsqueda personalizada:

- Un usuario busca perfiles relacionados con "tecnología".
- El sistema prioriza resultados basados en la relevancia calculada por PageRank.

## 8. Conclusiones

El proyecto 'Grape Loot' representa una sólida aplicación de estructuras de datos y algoritmos en un entorno simulado de redes sociales. La implementación de grafos, listas enlazadas y algoritmos avanzados como PageRank y similitud de Jaccard permite abordar problemas reales de gestión de datos y personalización de recomendaciones.