

## **Segundo Proyecto**

Escuela de Ingeniería en Computadores

Algoritmos y Estructuras de Datos I

Profesor:

Leonardo Araya

Estudiante:

Dylan Guerrero González | 2022016016

Paula Melina Porras Mora | 2023082886

10 de octubre del 2024

## Tabla de contenidos

Tabla de contenidos .....	2
Introducción .....	3
Descripción del problema.....	4
Descripción de la solución.....	5
Diagrama UML .....	6
Enlace a GitHub .....	7

# Introducción

El proyecto TinySQLDb tiene como objetivo principal diseñar e implementar un motor de bases de datos relacional sencillo que proporcione una comprensión práctica de cómo funcionan los sistemas de gestión de bases de datos. Por lo cual se deberán cumplir con ciertas cosas:

**Implementar Componentes Básicos:** La primera fase del proyecto se centra en la creación de los componentes fundamentales de un motor de bases de datos relacional, esto incluye el desarrollo de un cliente en PowerShell que permita a los usuarios interactuar con el servidor de base de datos de forma intuitiva. Además, se implementará un servidor que contendrá varias capas, cada una desempeñando un rol específico. La interfaz API facilitará la comunicación entre el cliente y el servidor, el procesamiento de consultas se encargará de interpretar y ejecutar las instrucciones SQL, y el gestor de datos almacenados se encargará de la persistencia de la información.

**Investigar Motores de Bases de Datos:** Al estudiar sus características, diseño y funcionamiento, se podrá identificar las mejores prácticas y técnicas que pueden ser aplicadas en nuestro propio motor. Este análisis permitirá entender cómo se gestionan las consultas, cómo se optimizan los índices y cómo se asegura la integridad de los datos en sistemas más complejos.

**Desarrollo en C#:** El proyecto se desarrollará utilizando el lenguaje de programación C#, que es ampliamente utilizado en la creación de aplicaciones de software, al emplear C#, se tendrá la oportunidad de aplicar conceptos de programación orientada a objetos (POO), haciendo uso de estructuras de datos, algoritmos de búsqueda y ordenación, y patrones de diseño.

En resumen, TinySQLDb no solo es un proyecto académico, sino también una oportunidad comprender los principios fundamentales detrás de los motores de bases de datos.

## Descripción del problema

TinySQLDb es un motor de base de datos diseñado para proporcionar una solución simple y eficiente para el almacenamiento y gestión de datos. Su arquitectura se basa en un modelo cliente-servidor, lo que permite una clara separación de responsabilidades y facilita la interacción entre los usuarios y el sistema de gestión de bases de datos.

**Arquitectura Cliente-Servidor:** El proyecto se compone de un cliente y un servidor. El cliente actúa como la interfaz de usuario, donde se puede introducir consultas y recibir los resultados de estas consultas. Por otro lado, el servidor es el núcleo del motor de base de datos, es responsable de procesar las consultas SQL enviadas por el cliente, ejecutar las operaciones solicitadas y devolver los resultados.

**Estructuras de Almacenamiento:** El almacenamiento es un componente esencial en TinySQLDb. Para gestionar los datos de manera eficiente, el servidor implementa estructuras de datos adecuadas, como los árboles B, estas estructuras son árboles autobalanceados que permiten realizar búsquedas, inserciones y eliminaciones de forma rápida, minimizando el número de accesos al disco, lo que es crucial para mantener un rendimiento óptimo.

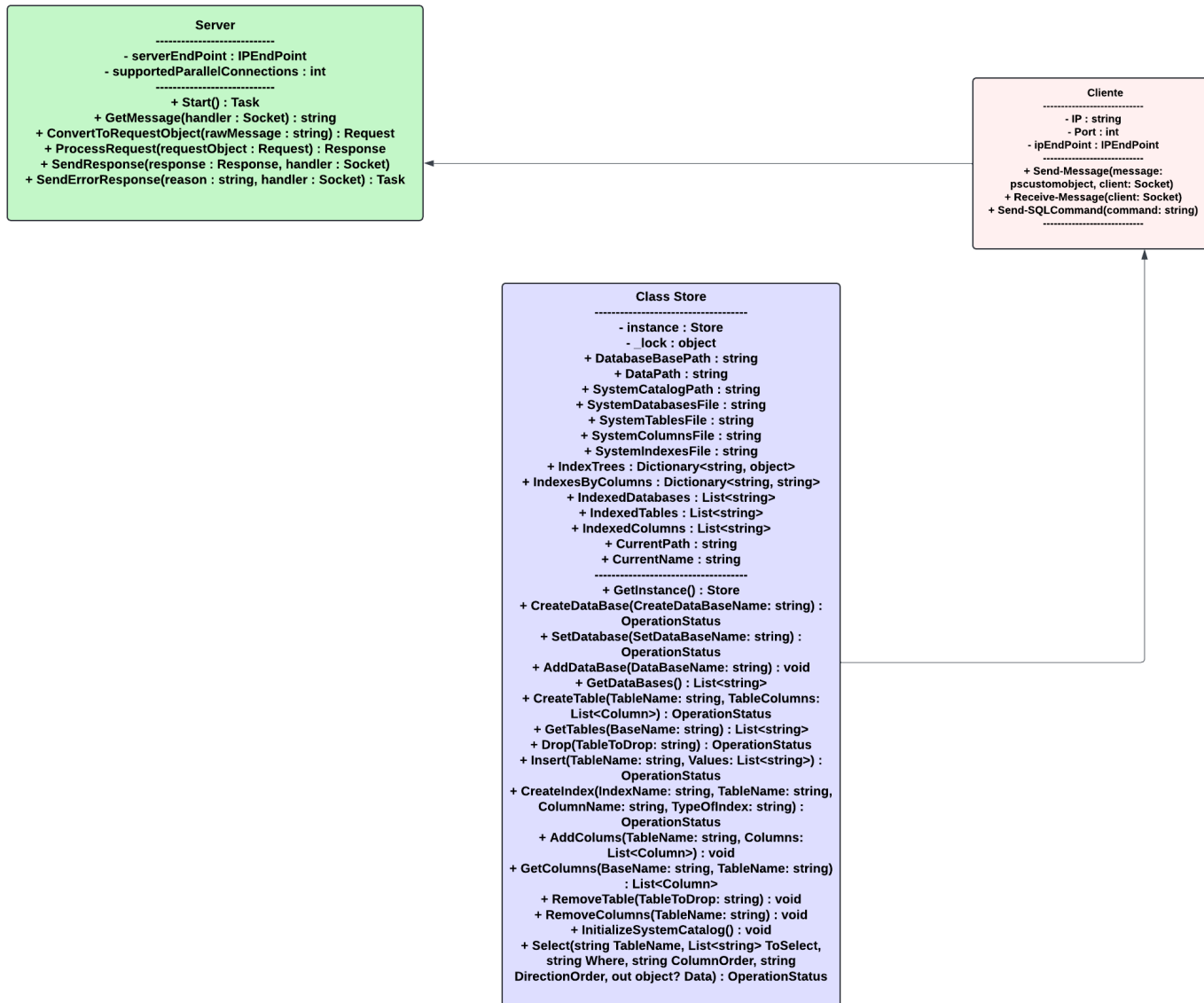
## Descripción de la solución

El objetivo de este proyecto es desarrollar un módulo de PowerShell que permita a los usuarios ejecutar sentencias SQL sobre una base de datos estructurada como un sistema de archivos. Este módulo proporcionará una interfaz para gestionar bases de datos, tablas, índices y realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre los datos almacenados, por lo cual se implementaron diferentes métodos y funciones para su solución.

- Se creó una función en PowerShell que permite ejecutar múltiples comandos SQL desde un archivo, facilitando la interacción con la base de datos.
- Se implementó una lógica para leer el contenido del archivo de comandos SQL especificado.
- El sistema está diseñado para almacenar datos como archivos binarios.
- Se utilizó un socket TCP para conectarse al servidor, utilizando la IP y el puerto proporcionado.
- Para cada sentencia SQL, se envió al servidor un objeto JSON que contenía la sentencia.
- Al recibir las respuestas del servidor, se procesó el estado de cada consulta (éxito, error, etc.) y se imprimieron los resultados en la terminal.
- Se creó un bucle que permite a los usuarios ingresar comandos SQL continuamente hasta que decidan salir.
- El cliente también está diseñado para manejar metadatos de la base de datos. Los comandos para crear bases de datos, tablas, índices, y para realizar operaciones CRUD (crear, leer, actualizar, eliminar) están implementados, permitiendo una gestión completa de la base de datos a través del cliente de PowerShell.
- La creación de índices en las tablas permite optimizar las consultas, haciendo que el sistema sea más eficiente en la recuperación de datos. Se implementaron sentencias para crear índices en columnas específicas.

Al finalizar el desarrollo, el módulo de PowerShell debe funcionar de manera efectiva, permitiendo a los usuarios gestionar bases de datos y realizar operaciones sobre los datos de manera intuitiva y eficiente. Esto facilitará el manejo de datos en un entorno de desarrollo o producción, y proporcionará herramientas para monitorear y optimizar el rendimiento de las consultas SQL.

## Diagrama UML



Enlace a GitHub

[Kasyrgan04/TinySQLDB \(github.com\)](https://github.com/Kasyrgan04/TinySQLDB)