**SISTEMA DE GETIÓN DE ESTUDIANTES**

**Cristian Manuel Velasquez Tovar**

**TABLA DE CONTENIDO**

Tabla de contenido

[RESUMEN 3](#_Toc184411627)

[**INTRODUCCIÓN** 4](#_Toc184411628)

[**PLANTEAMIENTO DEL PROBLEMA** 5](#_Toc184411629)

[**JUSTIFICACIÓN** 6](#_Toc184411630)

[**OBJETIVOS** 7](#_Toc184411631)

[**METODOLOGÍA** 8](#_Toc184411637)

[**RESULTADOS** 9](#_Toc184411639)

[**CONCLUSIÓN** 12](#_Toc184411640)

[**REFENCIAS** 13](#_Toc184411641)

RESUMEN

Este proyecto, llamado **Sistema de Gestión de Estudiantes** , es un programa en C++ que se enfoca en simplificar la administración de información estudiantil. Su función principal es permitir a los usuarios manejar datos básicos como nombres, identificaciones y promedios de los estudiantes. Entre sus principales características están registrar nuevos datos, consultar información ya guardada, modificar registros y eliminarlos si es necesario. El programa funciona a través de un menú interactivo que hace que su uso sea intuitivo. La idea es que el sistema sea una solución accesible para escuelas pequeñas o personas que quieran automatizar estas tareas sin complicarse demasiado.

# **INTRODUCCIÓN**

La gestión de estudiantes es una tarea básica pero muy importante en cualquier institución educativa. Aunque parezca algo simple, muchas veces se vuelve un problema cuando se hace de forma manual, ya que pueden surgir errores o pérdida de información. Por eso, este proyecto propone un sistema digital que facilita este trabajo. El programa permite realizar tareas básicas como registrar nuevos estudiantes, consultar datos ya existentes, modificarlos o eliminarlos según sea necesario. Todo esto se hace de manera sencilla y desde un menú interactivo que guía al usuario paso a paso. Con esta herramienta, se busca hacer que las tareas administrativas sean más rápidas y organizadas, especialmente en lugares donde no se cuenta con software más avanzado.

**PLANTEAMIENTO DEL PROBLEMA**

En muchas escuelas o instituciones educativas, especialmente las más pequeñas, el manejo de la información de los estudiantes se hace a mano o con herramientas muy básicas. Esto puede generar varios problemas, como desorganización, pérdida de datos o errores al buscar información. Por ejemplo, si se necesita saber los datos de un estudiante específico o actualizar su promedio, puede ser complicado encontrar la información de forma rápida y sin errores. Este tipo de problemas afecta tanto a los encargados de la administración como a los estudiantes. Este proyecto surge para resolver esos inconvenientes mediante un programa que organiza y gestiona esta información de forma digital y accesible, evitando los problemas que se presentan con los métodos tradicionales.

**JUSTIFICACIÓN**

La razón principal para desarrollar este sistema es que muchas instituciones necesitan herramientas simples y efectivas para organizar la información de los estudiantes. No siempre se cuenta con los recursos para adquirir software costosos o especializados, por lo que un programa sencillo como este puede ser una buena solución. Además, este proyecto también sirve como un ejercicio de aprendizaje. Al crearlo, se pueden practicar habilidades importantes de programación, como trabajar con menús, manejar estructuras de datos y diseñar un programa funcional que pueda resolver un problema real. Por último, el sistema es una base que se puede mejorar en el futuro para incluir funciones más avanzadas, como bases de datos o generación de informes.

**OBJETIVOS**

**Objetivo General:**  
Desarrollar un programa que permita gestionar de manera eficiente la información de los estudiantes, asegurando que sea fácil de usar y cumplir con las necesidades básicas de registro, consulta, actualización y eliminación de datos.

**Objetivos específicos:**

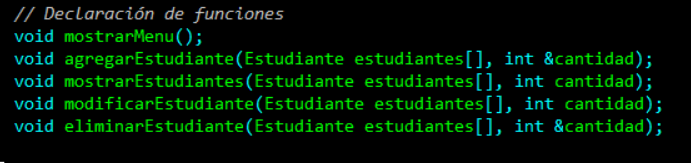
1. Cree un menú interactivo que facilite la navegación y el uso del sistema para los usuarios.
2. Diseñar funciones específicas que permiten registrar nuevos estudiantes, consultar información, modificar datos y eliminar registros.
3. Realizar pruebas al programa para asegurarse de que funcione correctamente en diferentes escenarios y con distintos tipos de datos.

**METODOLOGÍA**

**Cascada**

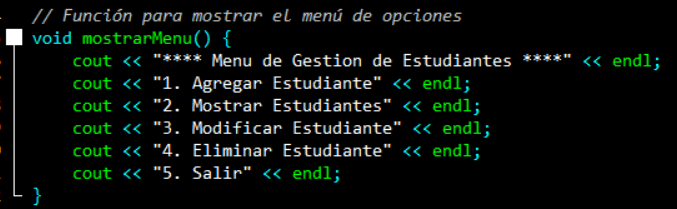
El **programa de gestión de estudiantes** sigue la metodología de cascada para su desarrollo. Primero, en la fase de **requerimientos**, se definieron las funcionalidades del sistema, como la capacidad de agregar, eliminar, modificar y consultar estudiantes, además de almacenar información básica como nombre, ID, promedio y edad. En la fase de **diseño**, se organizó el código en funciones modulares para cada operación, lo que facilita su implementación y mantenimiento. Luego, en la fase de **implementación**, se desarrolló el programa de manera secuencial, primero creando la estructura de datos y luego implementando las funciones para cada operación. Después, en la fase de **verificación**, se probaron las funciones para asegurarse de que el sistema cumpliera con los requisitos, como permitir la gestión correcta de estudiantes. Finalmente, en la fase de **mantenimiento**, el sistema podrá ser actualizado fácilmente para agregar nuevas funcionalidades o corregir errores. Este enfoque de cascada garantiza que el programa sea desarrollado de manera ordenada y que cada fase se complete antes de pasar a la siguiente.

**RESULTADOS**

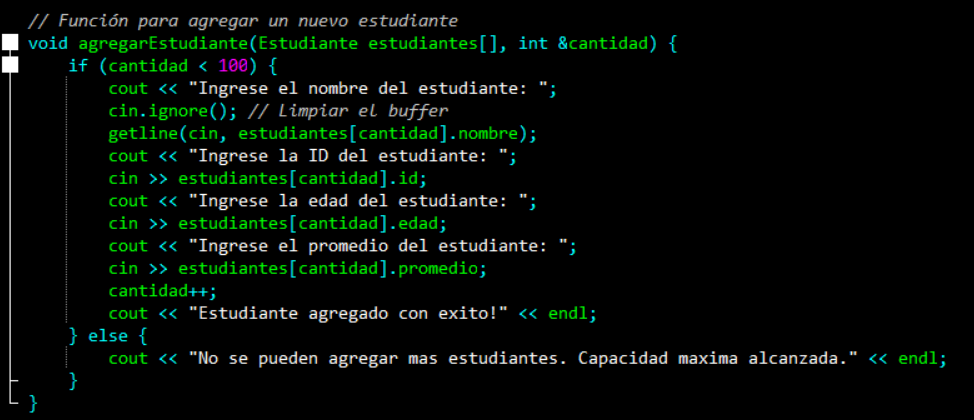
En el desarrollo del programa de **gestión de estudiantes**, utilicé código en C++ dividido en varias secciones para lograr un sistema funcional y organizado. El núcleo del programa se basa en el uso de estructuras para almacenar los datos de cada estudiante, como nombre, ID, promedio y edad. Las principales operaciones (agregar, mostrar, modificar y eliminar estudiantes) se implementan a través de funciones específicas, lo que ayuda a mantener el código modular y fácil de leer. 

El programa también incluye un **menú interactivo** que utiliza un bucle do-while, permitiendo al usuario realizar múltiples operaciones hasta seleccionar la opción de salir. Las decisiones dentro del menú se manejan mediante un **switch-case**, que dirige al usuario a la funcionalidad correspondiente. Para la gestión de datos, se emplea un arreglo de estructuras, que actúa como una base de datos básica dentro de la memoria del programa.

Texto

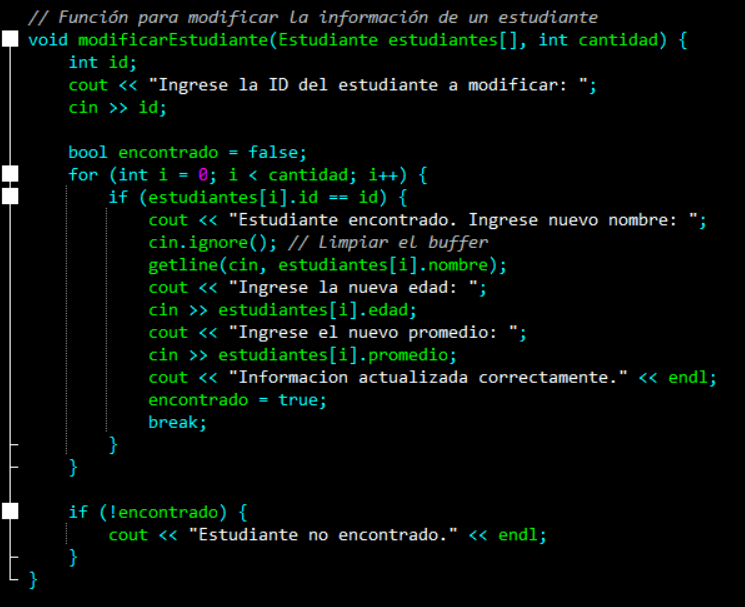
Descripción generada automáticamente 

Por ejemplo, la función agregarEstudiantesolicita al usuario los datos básicos del estudiante, incluyendo la edad, y los almacena en el arreglo. Otras funciones, como mostrarEstudiantes, iteran sobre este arreglo para listar los datos registrados, mientras que modificarEstudiantey eliminarEstudiantepermiten actualizar o borrar registros según sea necesario.



Pantalla de computadora con fondo verde

Descripción generada automáticamente con confianza baja



Texto

Descripción generada automáticamente

# **CONCLUSIÓN**

El desarrollo del **sistema de gestión de estudiantes** permitió implementar de manera práctica los conceptos fundamentales de la programación estructurada en C++. A través del uso de estructuras, funciones y control de flujo, se logró construir un programa funcional que permite gestionar de forma eficiente la información de los estudiantes, incluyendo su nombre, ID, promedio y edad. El enfoque basado en la metodología de cascada garantizó un desarrollo ordenado, donde cada fase se ensambla antes de avanzar, lo que facilitó la organización y ejecución del proyecto.

Entre los aportes principales del trabajo destaca la posibilidad de usar el programa como una herramienta básica de gestión para entornos educativos pequeños, mostrando cómo los fundamentos de programación pueden resolver problemas cotidianos. Además, la implementación modular del código asegura que el sistema sea escalable, permitiendo futuras mejoras o nuevas funcionalidades como la integración con bases de datos externos o el cálculo automático de estadísticas sobre los estudiantes registrados.

# **REFENCIAS**

1. Documentación oficial de C [++ : https://cplusplus.com](https://cplusplus.com)

2. [Libro o guía utilizada como referencia]

3. [Otros recursos usados ​​en el desarrollo del proyecto]