**Instituto Tecnológico de Costa Rica**

****

**Lenguajes de Programación**

**Proyecto programado #1**

**Desarrollado por:**

**Darío Vargas**

**María Rubio**

**II Semestre**

**2021**

**Tabla de contenidos**

[**Manual de usuario**](#_k9zla72foeen) **3**

[Compilación](#_u48qsczgde4t) 3

[Ejecución](#_ifxso4s9d170) 3

[Uso](#_kwpg67iopdsx) 3

[**Pruebas de funcionalidad**](#_maqxmgoq5fu1) **5**

[**Descripción del problema**](#_bcbl0wm9r0m6) **9**

[**Diseño programa**](#_q4fybvknc5f7) **9**

[Decisiones de diseño](#_crdx2t8zqrg6) 9

[Algoritmos usados](#_y1nqy0vz31d6) 12

[Diagramas de Bases de Datos](#_tgdcshd23puf) 15

[**Librerías Usadas**](#_glazazkcx16q) **16**

[**Resultados**](#_h56254dwhta0) **16**

[Objetivos alcanzados](#_2btbd5gkzs3e) 16

[Objetivos no alcanzados](#_hsxqn5g0mbnj) 16

[**Bitácora**](#_ba2m3opwh2si) **16**

# Manual de usuario

## Compilación

Para compilar el proyecto en linux se necesita:

1. Tener el sistema operativo Linux o una máquina virtual
2. Instalar Visual Studio Code
3. Instalar la extensión de C/C++ para VS Code
4. Verificar en la terminal gcc -v
5. Actualizar lista de paquetes sudo apt-get update
6. Instalar herramientas de compilación de GNU y debugger sudo apt-get install build-essential gdb
7. Descargue el proyecto de GitHub y ábralo en VS Code
8. Abra la terminal y escriba el siguiente comando gcc main.c -o main (Ya se compila el archivo)

## Ejecución

Una vez compilado el programa solamente es necesario escribir en la terminal:

1. ./programa (ejecuta el programa)

## Uso

Para el uso del programa lo único que se necesita es ir escribiendo el número de la opción y los datos que le piden en formato y orden que este lo solicita. >>> Se muestra como se debería de responder.

Menú

1.) Opciones operativas

2.) Opciones generales

3.) Salir

>>> 1

--------------------------------------------------------------------------

Opciones operativas

1.) Información Aulas

2.) Información de Profesores

3.) Información de Cursos

4.) Cursos por Período

5.) Reservación de aula

6.) Cancelar reservación

7.) Estadísticas

8.) Volver

>>> 1

--------------------------------------------------------------------------

Información de aulas

1.) Incluir en lote

2.) Incluir aula

3.) Listar aula

4.) Volver

>>> 2

--------------------------------------------------------------------------

Ingrese el nombre del aula

>>> A12

Ingrese la capacidad

>>> 15

--------------------------------------------------------------------------

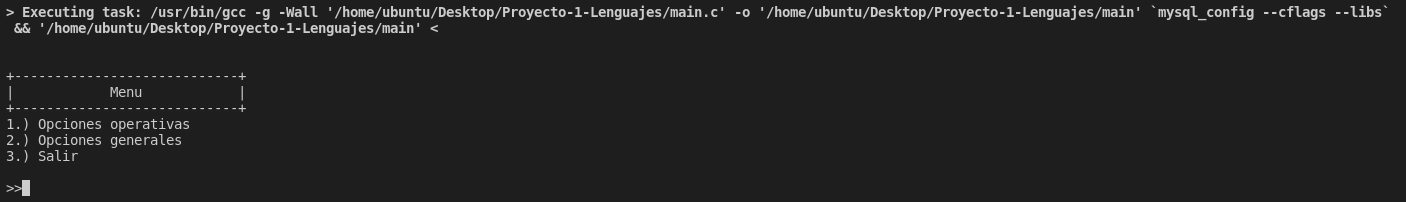
A12 5

# 

# 

# Pruebas de funcionalidad

Las pruebas de funcionalidad del sistema se realizan con constantes ejecuciones del programa, esperando a que responda de la mejor manera y en caso de no hacerlo corregir los errores. En los siguientes screenshots podemos observar algunas pruebas de funcionalidad del sistema.



Screenshot 1. Menú principal



Screenshot 2. Opciones operativas



Screenshot 3. Información de aulas



Screenshot 4. Incluir un lote de agua



Screenshot 5. Incluir un aula



Screenshot 6. Listar aulas



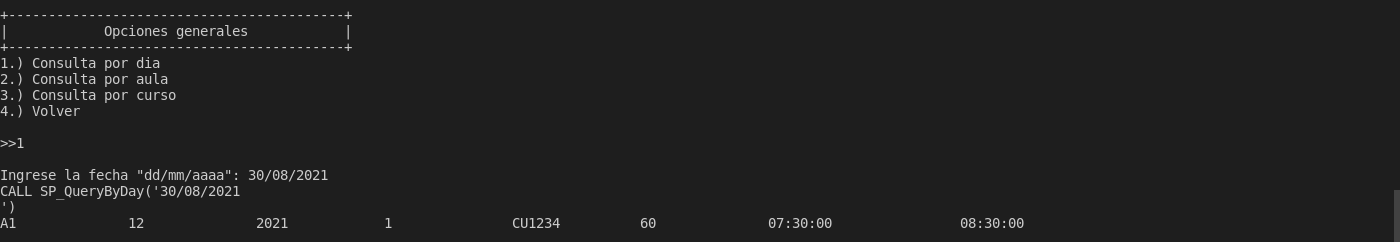
Screenshot 7. Eliminar aula



Screenshot 8. Incluir profesor



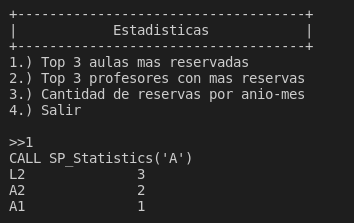
Screenshot 9. Incluir curso por periodo



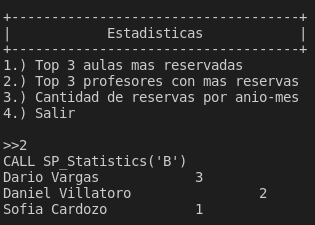
Screenshot 10. Consulta por dia



Screenshot 11. Consulta por aula



Screenshot 12. Estadística de top3 de aulas reservadas



Screenshot 13. Estadística de top3 de profesores con reservas

# Descripción del problema

Dado a la pandemia por COVID-19 que se enfrenta, muchos centros educativos tuvieron que cerrar y dar lecciones por la virtualidad, una vez teniendo las vacunas se plantea el regresar a clases, pero si bien es cierto en el virus sigue por lo que debe mantener distancias de 1.8m entre cada persona, con esto se reduce el aforo de las aulas y hay muchos cambios en los horarios.

Por lo que se plantea realizar un sistema que permita a la administración manejar sus infraestructuras, así, como la reserva de aulas y que esto no tenga choques de horario o sobrepase el aforo permitido.

# Diseño programa

## Decisiones de diseño

Entre las decisiones de diseño que se tomaron, se tomó en cuenta la base de datos, donde inicialmente se plantea un diagrama, pero en el camino este se cambia, por razones de facilidad de manejo de datos y mantener la base de datos normalizada. Imagen 1 es la forma inicial, imagen 2 como quedó. La base de datos que se seleccionó fue MySQL ya que uno de los integrantes tenía un poco de conocimiento en ella.

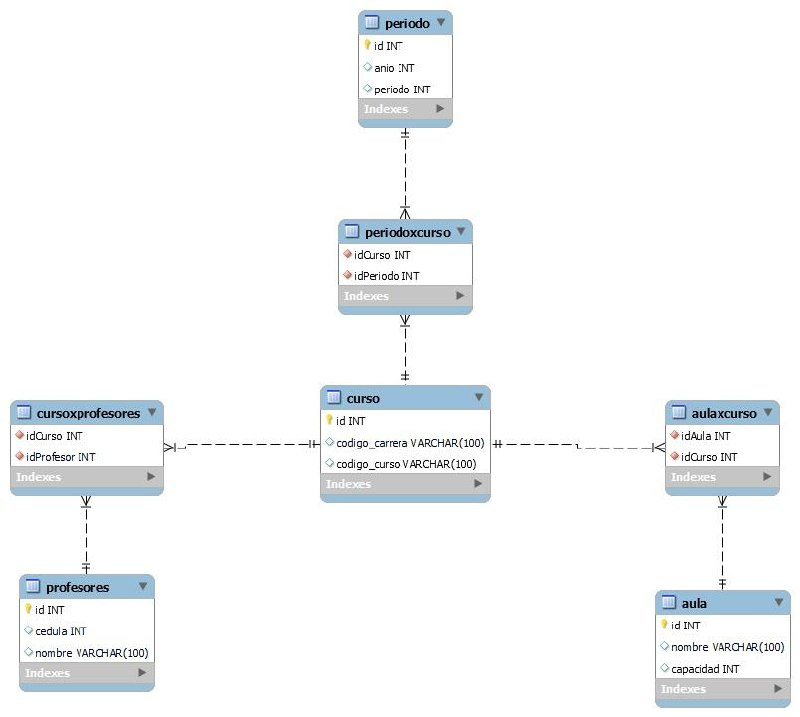


Imagen 1. Diagrama de base de datos inicial.

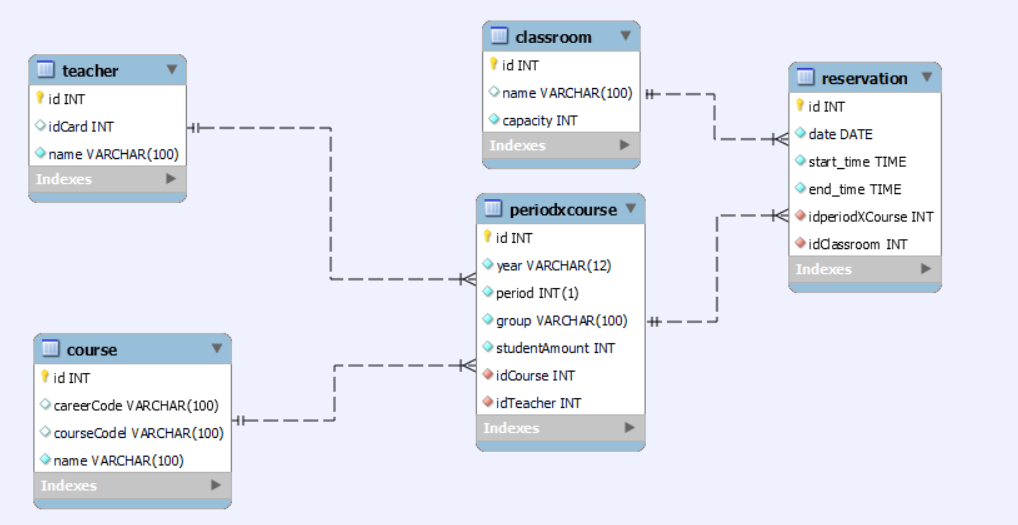


Imagen 2. Diagrama de base de datos final.

Después, en las mismas bases de datos se dificultó el uso de DATE y YEAR por lo que se debió cambiar el año, para que fuera tipo varchar y no tipo DATE, ya que no lo leía.

Siguiendo con esto, y al proyecto al ser en consola se planteó que todos los menús funcionen por medio de impresión de cada sub-menú u opción junto a un número y que el usuario digitara el número de esta.

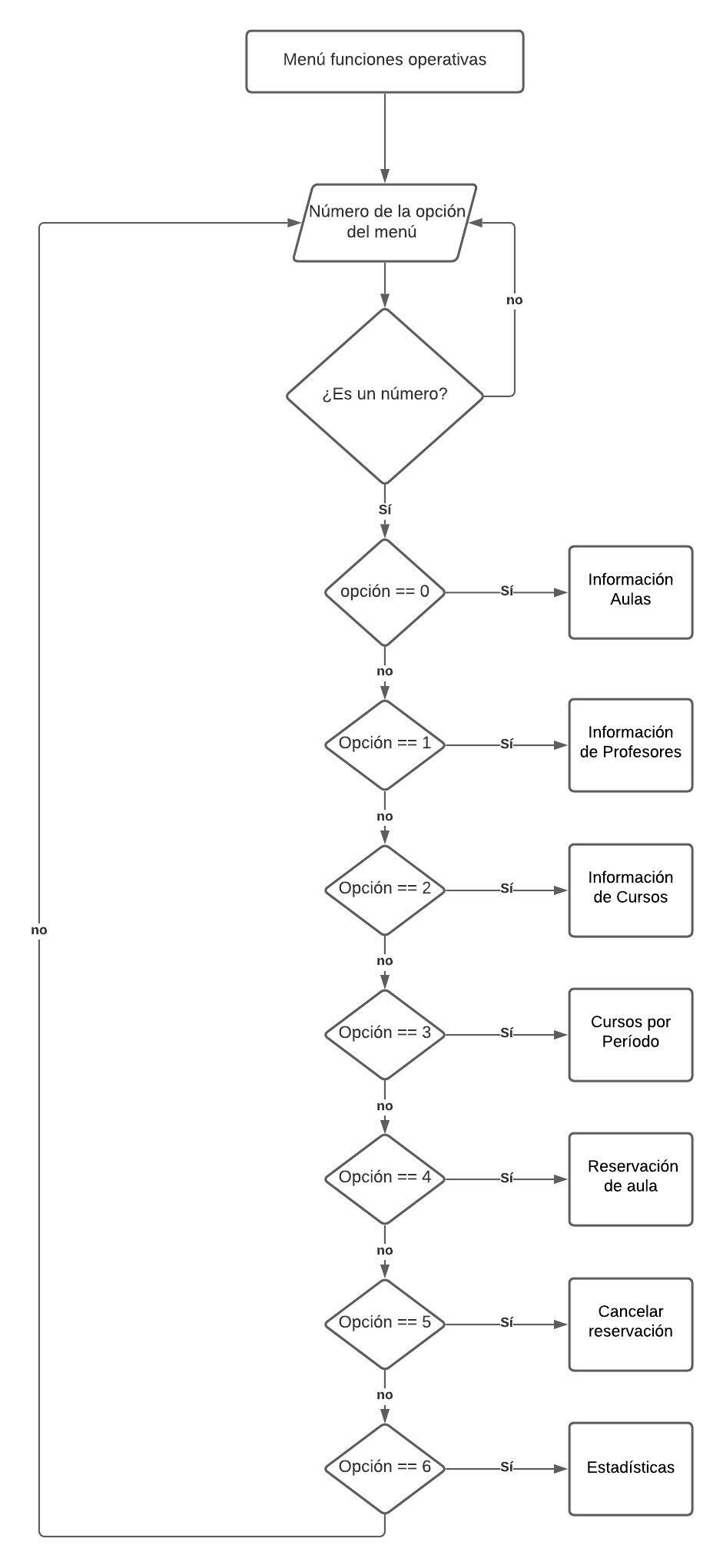


Imagen 3. Algoritmo de menú de funciones operativas

En esta imagen observamos un ejemplo de esto, el usuario si digita 0 va a información de aula, 1 información de profesores y así sucesivamente.

Por último, el uso de los struct se planteó de la siguiente manera

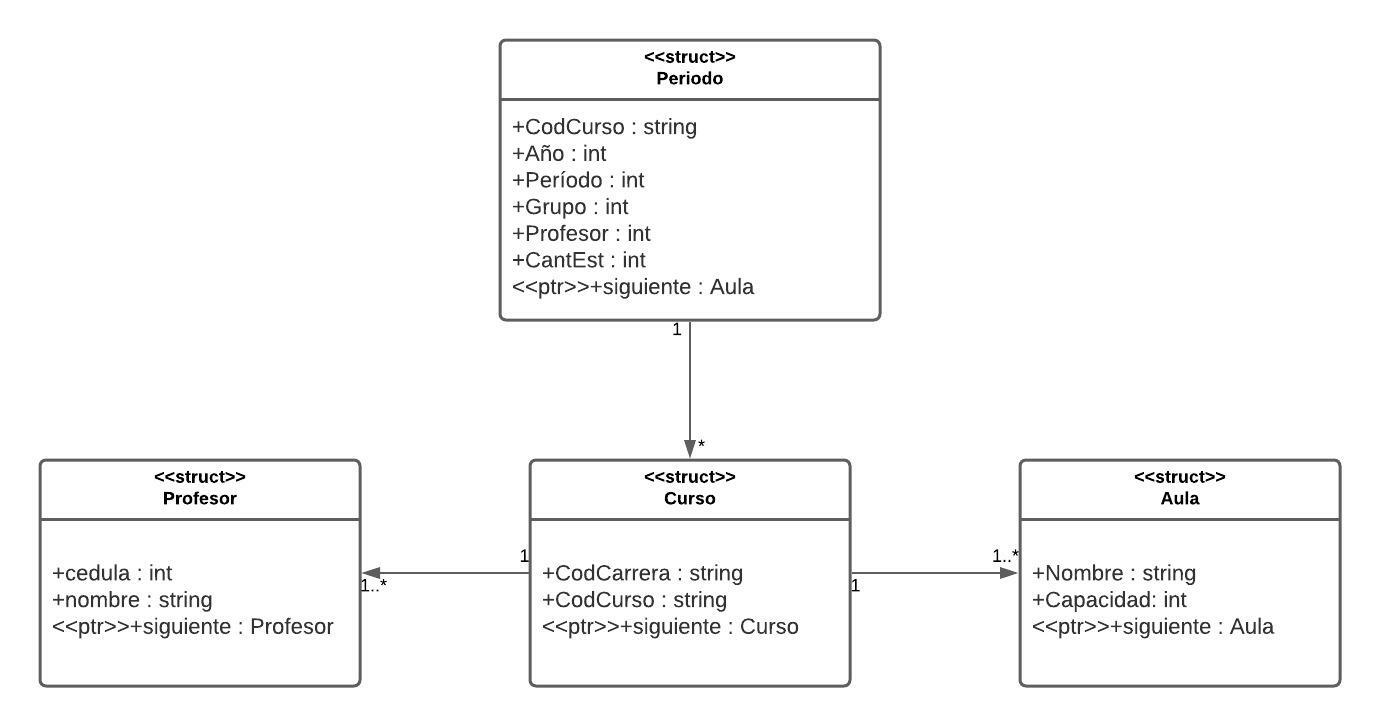


Imagen 4. Diagrama de structs

## Algoritmos usados

Los algoritmos utilizados fueron muy básicos como mencionados anteriormente, básicamente es la impresión de opciones y el usuario decide por medio del número que acompaña la opción. A Continuación se mostrarán 3 ejemplos de los algoritmos usados en los diferentes menús.

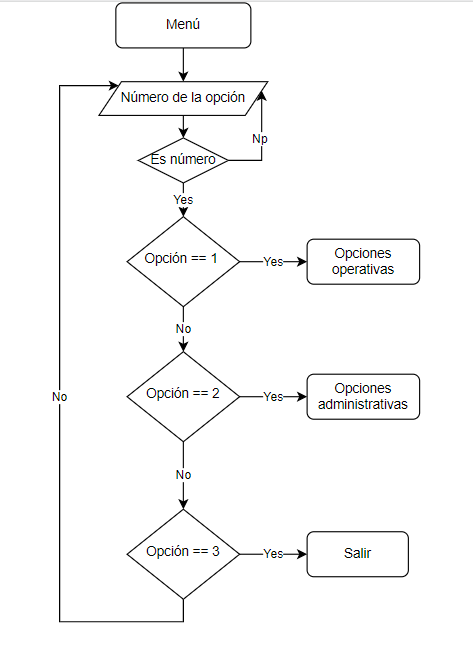


Imagen 5. Algoritmo Menú

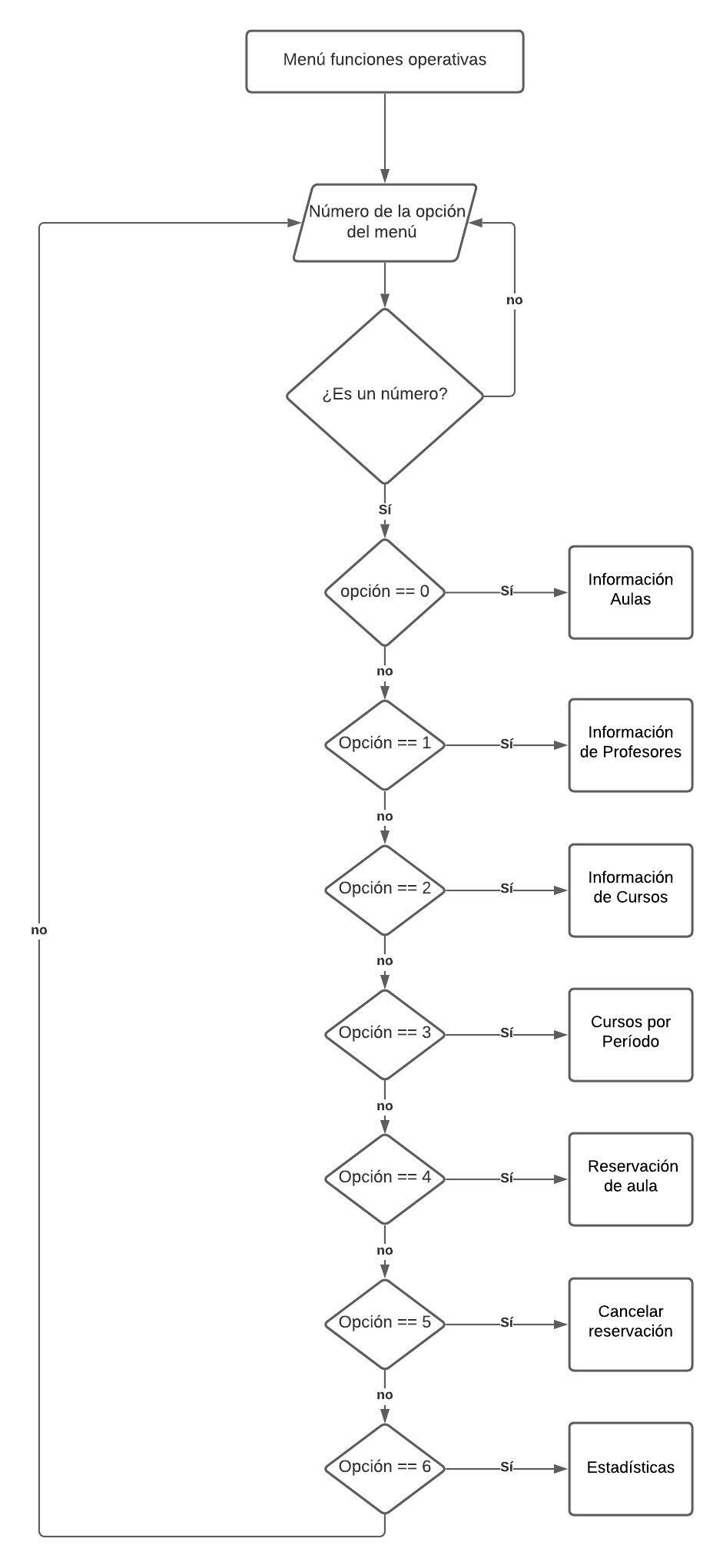


Imagen 6. Algoritmo de menú de funciones operativas

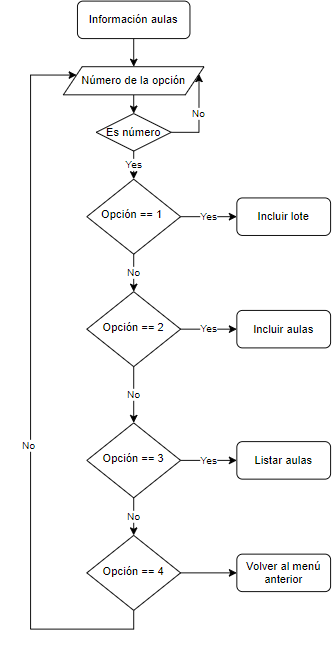


Imagen 7. Algoritmo de menú de funciones operativas

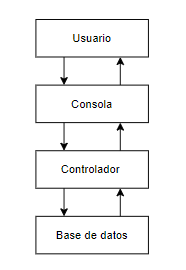


Imagen 8. Funcionamiento del programa   
  
Aquí se puede ver básicamente el funcionamiento del sistema, el usuario ingresa la opción que desea, la consola lo pasa al controlador, el controlador busca el procedimiento almacenado en la base de datos y retorna los datos necesarios. Luego el controlador muestra los datos en la consola y el usuario lo recibe.

## Diagramas de Bases de Datos

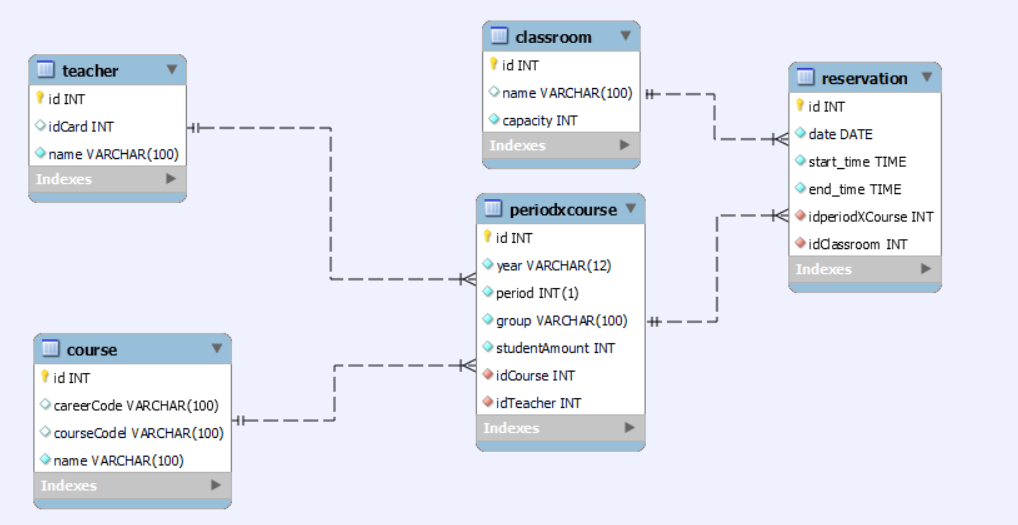


Imagen 9. Diagrama de bases de datos

# Librerías Usadas

**mysql.h**

Esta librería permite realizar la conexión con la base de datos de MySQL y además, realizar las consultas necesarias para el funcionamiento del programa.

**stdio.h**

Significa Standard Input Output. Esta librería nos permite tener la información de los datos de las funciones de entrada y salida. Cómo imprimir y obtener datos del teclado, además, de abrir archivos.

**stdlib.h**

Una biblioteca estándar, esta librería permite la asignación de memoria, control de procesos, además, de permitir la conversión de los datos.

**string.h**

Esta librería nos permite hacer uso de funciones para el manejo de cadenas y manejo de memoria.

Una biblioteca estándar, esta librería permite la asignación de memoria, control de procesos, además, de permitir la conversión de los datos.

# Resultados

## Objetivos alcanzados

Ingresar todo a la base de datos

Listar todo lo que se le solicita

Eliminar elementos

Realizar consultas

## Objetivos no alcanzados

| **Objetivo no alcanzado** | **Razón** |
| --- | --- |
| Que pudiera reservar varias aulas y verificar el choque de horario. | Falta de tiempo. |
| No toda la funcionalidad está correcta | Falta de tiempo, problemas con los programas y una computadora. |

# Bitácora

<https://github.com/Dasperless/Proyecto-1-Lenguajes>