

**UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA
CENTRO UNIVERSITARIO SAN BENITO, PETÉN
FACULTAD DE INGENIERÍA EN SISTEMAS DE
INFORMACIÓN.**



CATEDRÁTICO: Luis Antonio Garcia Aguilar.

CURSO: Programación I

TRABAJO:

**Proyecto C++ con Conexión a
Base de Datos MySQL y GitHub**

Alumno:

Mario Rigoberto Cortez Rodas.

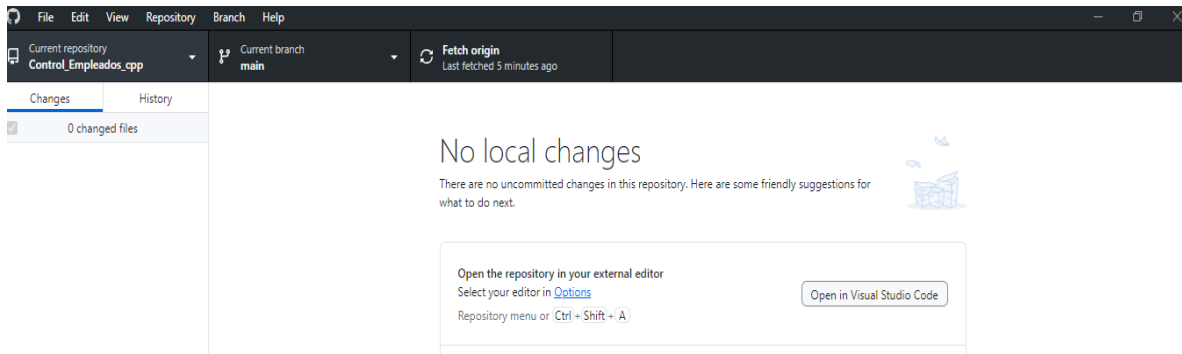
Carné:

1690-22-12638

Paso a Paso del Proyecto con Conexión a Base de Datos

1. Creación del repositorio en GitHub Desktop

- Abrí GitHub Desktop.
- Seleccioné la opción para crear un nuevo repositorio.
- Asigné un nombre “control_empleados_cpp” al proyecto y lo guardé en una carpeta local.
- Cree una nueva carpeta “Base_Datos”



2. Creación de tablas en MySQL usando la consola de comandos

- Abrí la terminal de MySQL.



- Ejecuté los comandos SQL para crear la nueva base de datos “*Create databases Control_Horario*” luego cree las tablas necesarias con su estructura (campos, tipos de datos, claves foráneas).

```

+-----+
| Database |
+-----+
| control_horario |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.01 sec)

mysql> use control_horario;
Database changed
mysql> show tables;
+-----+
| Tables_in_control_horario |
+-----+
| empleados |
| personas |
| registros |
+-----+
3 rows in set (0.01 sec)

mysql> DESCRIBE empleados;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id | int | NO | PRI | NULL | |
| Personas_id | int | YES | MUL | NULL | |
| Cargo | varchar(50) | YES | | NULL | |

```

3. Conexión y verificación de tablas en HeidiSQL

- Realice las conexiones.

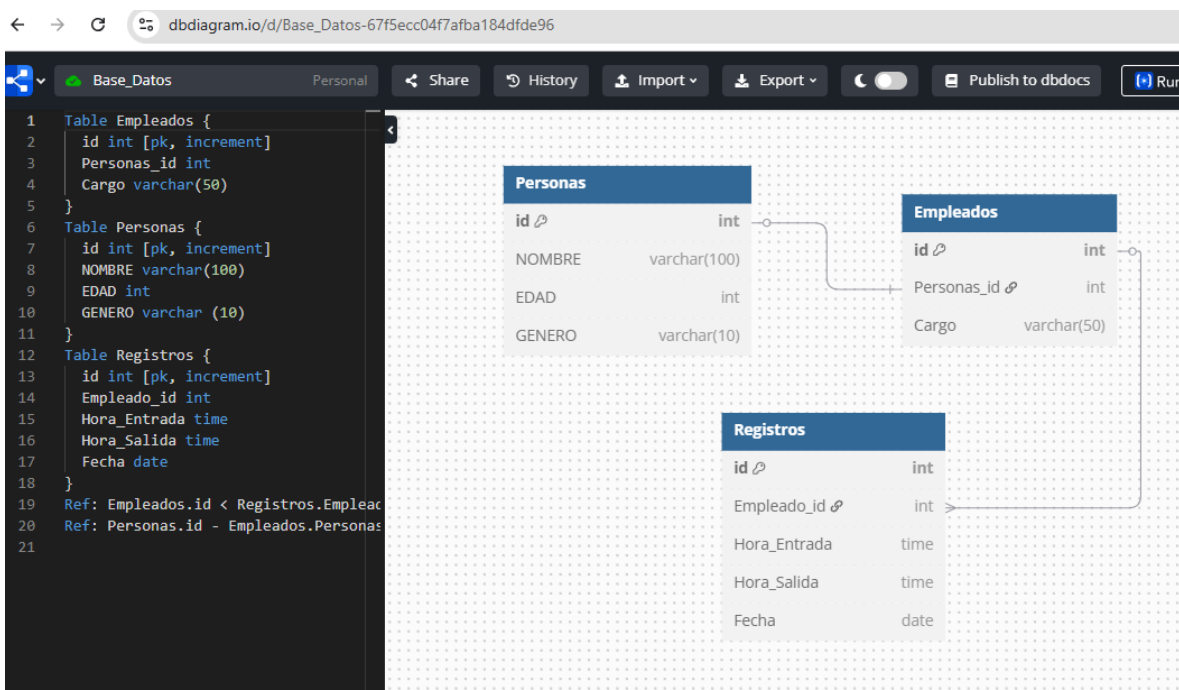
```
1 ALTER TABLE registros
2 ADD CONSTRAINT fk_registros_empleados
3 FOREIGN KEY (empleado_id) REFERENCES empleados(id);
```

```
1 ALTER TABLE registro
2 ADD CONSTRAINT fk_registro
3 FOREIGN KEY (empleado_id) REFERENCES empleados (id);
4
```

```
1 ALTER TABLE empleados
2 ADD CONSTRAINT fk_empleados_personas
3 FOREIGN KEY (personas_id) REFERENCES personas(id);
```

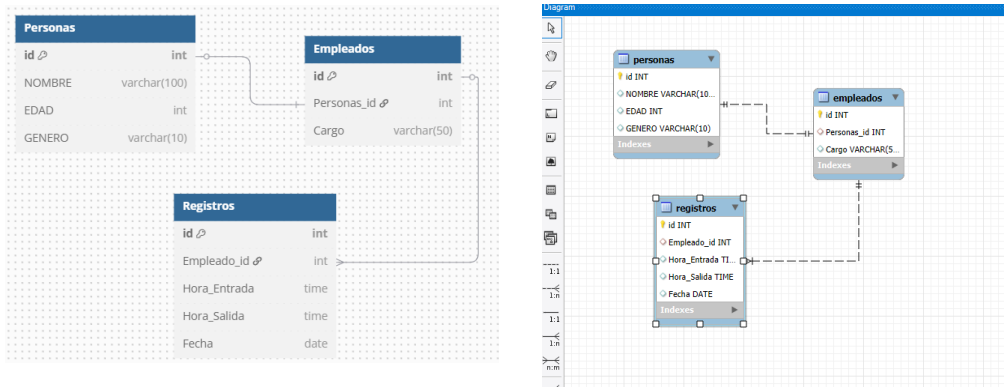
4. Creación del diagrama entidad-relación en dbdiagram.io

- Ingresé a dbdiagram.io y realicé el diagrama.



5. Comparación de la estructura de la base de datos en MySQL Workbench

- Comparé el diagrama creado con la base de datos en Workbench



6. Inserción de datos en las tablas usando HeidiSQL

- En HeidiSQL, seleccioné las tablas deseadas e ingresé los datos.

control_horario.empleados: 5 filas en total (exacto)

#	id	Personas_id	Cargo
1	1	5	GERENTE
2	2	1	TESORERO
3	3	4	DIRECTOR DE INFORMATICA
4	4	3	DIRECTORA RECURSOS HUMANOS
5	5	5	PRESIDENTE

control_horario.personas: 5 filas en total (exacto)






#	id	NOMBRE	EDAD	GENERO
1	1	ELMER ROMARIO	23	MASCULINO
2	2	ISAIAS RODAS	35	MASCULINO
3	3	LUZ MAGALLON	20	FEMENINO
4	4	MARIO CORTEZ	19	MASCULINO
5	5	JOSIAS CORTEZ	21	MASCULINO

control_horario.registros: 30 filas en total (exacto)

#	id	Empleado_id	Hora_Entrada	Hora_Salida	Fecha
1	1	1	07:45:00	16:10:47	2025-04-04
2	2	5	07:57:00	16:11:48	2025-04-04
3	3	3	07:58:00	16:10:23	2025-04-04
4	4	4	07:59:00	16:35:00	2025-04-04
5	5	2	07:59:28	16:05:39	2025-04-04
6	6	1	07:40:26	16:36:36	2025-04-05
7	7	2	07:42:02	16:37:11	2025-04-05
8	8	3	07:45:35	16:37:43	2025-04-05
9	9	4	07:47:06	16:38:23	2025-04-05
10	10	5	07:47:49	16:38:59	2025-04-05
11	11	5	07:50:33	16:39:39	2025-04-06
12	12	4	07:51:59	16:40:12	2025-04-06

7. Configuración Visual Studio Code

- Cree un archivo main.cpp, Descargué e instalé las librerías necesarias para conectarme a MySQL (MySQLConexion, EloquentORM.h) y las coloque en la misma carpeta de main.cpp

 EloquentORM	9/04/2025 19:08	Archivo de origen ...	10 KB
 libmysql.dll	16/12/2024 03:55	Extensión de la ap...	6,827 KB
 main	10/04/2025 00:08	Archivo de origen ...	1 KB
 main	10/04/2025 00:16	Aplicación	649 KB
 MySQLConexion	9/04/2025 19:00	Archivo de origen ...	3 KB

- Configuré los archivos de compilación (tasks.json) para permitir compilar el código correctamente desde VS Code.

```
vscode > {} tasks.json > [ ] tasks > {} 1 > [ ] args
 2     "tasks": [
 3         {
27     },
28     {
29         "type": "cppbuild",
30         "label": "C/C++: g++.exe compilar archivo activo 🤔 🤔",
31         "command": "C:\\msys64\\mingw64\\bin\\g++.exe",
32         "args": [
33             "-fdiagnostics-color=always",
34             "-g",
35             "${file}",
36             "-I",
37             "C:/Program Files/MySQL/MySQL Server 8.0/include",
38             "-L",
39             "C:/Program Files/MySQL/MySQL Server 8.0/lib",
40             "-llibmysql",
41             "-o",
42             "${fileDirname}\\${fileBasenameNoExtension}.exe"
43         ],
44         "options": {
45             "cwd": "${fileDirname}"
46         },
47         "problemMatcher": [
```































8. Desarrollo del código

- Realicé operaciones básicas (insertar, consultar, actualizar, eliminar).

```
main.cpp > ...
1  #include "MySQLConexion.h"
2  #include "EloquentORM.h"
3  #include <iostream>
4  using namespace std;
5
6  int main() {
7      MySQLConexion conn("root", "Marioco@2025.", "control_horario");
8      if (!conn.open()) {
9          cerr << "No se pudo conectar." << endl;
10         return 1;
11     }
12
13     vector<string> columnas = { "NOMBRE", "EDAD", "GENERO" };
14     EloquentORM personas(conn, "personas", columnas);
15
16     auto lista = personas.getAll();
17     for (auto &reg : lista) {
18         cout << "ID: " << reg["id"] << ", Nombre: " << reg["NOMBRE"] << endl;
19     }
20
21     return 0;
22 }
```

9. Finalización del proyecto

- Hice commit de todos los archivos y subí los cambios a GitHub.

Organice los archivos	
 marioco004 committed yesterday	6a7a6f1   <>
Insertados registros y comandos ejecutados en base de datos	
 marioco004 committed yesterday	c3b7810   <>
Commits on Apr 9, 2025	
Merge branch 'main' of https://github.com/marioco004/Control_Empleados_cpp	
 marioco004 committed yesterday	5c97e4f   <>
Registros y comandos 	
 marioco004 committed yesterday	615298b   <>
Primer Commit 	
 marioco004 authored yesterday	 bdd7d77   <>
Add files via upload	
 marioco004 authored yesterday	 7557a32   <>
Delete Base_Datos/Control_Horario_Empleados.sql	
 marioco004 authored yesterday	 d84c747   <>
Commits on Apr 8, 2025	
Control_Horario 	
 marioco004 committed 2 days ago	48ef8f4   <>

Acá está el enlace de GitHub.

https://github.com/marioco004/Control_Empleados_cpp.git