

Fundación Universitaria  
**SAN MATEO**



Fundación Universitaria  
**SAN MATEO**

**FACULTAD DE INGENIERÍA**  
**TÉCNICO PROFESIONAL EN SOPORTE DE SISTEMAS INFORMATICOS**

**GRAFICÓDIGO MYSQL: UNA APLICACIÓN PARA LA GENERACIÓN DE CÓDIGO FUENTE COMPATIBLE CON MYSQL**  
**WORKBENCH 6.3**  
**TRABAJO DE GRADO**

**LEONARDO GUSTAVO PARRA AREVALO**

**DIRECTOR**  
**JHON ALEXANDER LOPEZ FAJARDO**

**BOGOTÁ D.C.**  
**2019**

## **NOTA DE SALVEDAD DE RESPONSABILIDAD INSTITUCIONAL**

*“La Fundación Universitaria San Mateo NO se hace responsable de los conceptos emitidos en el presente documento, el departamento de investigaciones velará por el rigor metodológico de la investigación”.*



# CONTENIDO

<b>INTRODUCCIÓN</b>	14
<b>CAPITULO I</b>	16
<b>DESCRIPCIÓN DEL PROYECTO</b>	16
Presentación del problema de investigación	16
Justificación	17
Objetivos	18
<i>Objetivo General</i>	18
<i>Objetivos Específicos</i>	18
<b>CAPITULO II</b>	19
<b>MARCO TEÓRICO</b>	19
Antecedentes de la investigación	¡Error! Marcador no definido.19
Bases teóricas o fundamentos conceptuales	21
<b>CAPITULO III</b>	23
<b>DISEÑO METODOLÓGICO</b>	23
Tipo de investigación	24
Población	24
Técnicas e instrumentos de recolección de datos	25
<b>CAPITULO III</b>	26
<b>RESULTADOS DE LA INVESTIGACIÓN</b>	26
Resultados del objetivo específico no. 1	29
Resultados del objetivo específico no. 2	30
<b>CAPÍTULO V.</b>	31
<b>CONCLUSIONES Y RECOMENDACIONES</b>	31
<b>BIBLIOGRAFÍA</b>	33

# ÍNDICE DE ILUSTRACIONES

Figura 1: Diagrama de clases UML .....	24
Figura 2: Diagrama de casos de uso UML .....	25
Figura 3: Menú de configuración .....	30

# ÍNDICE DE TABLAS

Tabla 1: Caso de uso "Nombrar base de datos" .....	25
Tabla 2: Caso de uso "Nombrar tabla" .....	26
Tabla 3: Caso de uso "Crear relación" .....	26
Tabla 4. Caso de uso "Generar código" .....	27

# DEDICATORIA

Quiero dedicar esta tesis a mis padres Gustavo y Yormary, ya que ellos han dado razón a mi vida, por sus consejos, su apoyo incondicional y su paciencia, todo lo que he logrado es gracias a ellos.

A mis hermanos Juan, Camilo y Karen que más que mis hermanos son mis amigos.

A toda mi familia que es lo mejor que Dios me ha dado.

# AGRADECIMIENTOS

Doy gracias a Dios porque este trabajo ha sido resultado de su ayuda, gracias porque has estado presente en esta etapa tan importante de mi vida.

Agradezco también al mi asesor de este proyecto de grado, profesor Jhon Alexander Lopez Fajardo, por haberme dedicado tanto tiempo para impartirme sus conocimientos y por haberme enseñado tanto, y por haber tenido la paciencia para explicarme y guiarme durante todo este tiempo.

# ABREVIATURAS

UML: Son las siglas correspondientes al Lenguaje Unificado de Modelado. Es el conjunto de ilustraciones que se usan para trazar la planificación de un software, este lenguaje facilita la comprensión para el programador y del cliente.

Modelo ER: (Modelo Entidad Relación): Es el modelo con el cual se representa la estructura de una base de datos de forma gráfica para mayor entendimiento y comprensión del desarrollador de bases de datos.

GUI: (Graphical User Interface) o (Interfaz Gráfica de Usuario), Son las formas e imágenes que permiten al usuario interactuar con un programa.

SQL: Lenguaje de consulta estructurado: Se refiere a el lenguaje de base de datos normalizado que utilizan los motores de bases de datos para ejecutar operaciones.

# RESUMEN

Algunas veces, por diferentes motivos, no se cuenta con el tiempo suficiente para hacer la escritura del código para una base de datos, esto se debe a la duración que demanda cada actividad en el proceso, desde el diseño de la base de datos hasta la creación de las tablas y relaciones en lenguaje SQL, teniendo en cuenta que el desarrollador debe dedicar la mayoría de su tiempo a la programación de la aplicación que se conecta con dicha base de datos , este problema requiere de la investigación para encontrar alternativas que permitan su solución, es necesario agilizar el proceso de generación del código SQL, este proyecto dará como resultado un programa de código abierto escrito en el lenguaje de programación java permitiendo mediante una interfaz gráfica (GUI), escoger las tablas, atributos, llaves, relaciones y otros elementos de la bases de datos, eliminando el esfuerzo de escribir línea por línea, obteniendor el código fuente de la base de datos similar a como lo hace MySQL workbench 6.3 y logrando de este modo que el usuario haga la misma tarea más sencilla y rápidamente.

## **PALABRAS CLAVE:**

**Base de datos, Diseño de bases de datos, SQL, Generación código SQL, Lenguaje, Java, GUI, SQL.**

# ABSTRACT

Sometimes, for different reasons, you do not have enough time to write the code of a database it may be a task that in the most of the times it demands a long time, including the programmer often has a schedule where he can not spend time enough to this activity, the person affected by this problem needs to find an alternative solution for that problem, it becomes necessary to speed the process of code creation, for this reason this project will be a code creation in an open source application written in the programming language java, using a graphical interface (GUI), where the user could choose tables, attributes, keys, relationships and other necessary elements , it is no longer mandatory to write line by line, the user can generate the source code of the database similar to MySQL workbench 6.3, thus achieving, that the user Make the same task easier and faster.

**KEY WORDS:**

**Database, Database Design, SQL, Generation SQL code, Language, Java, GUI, SQL.**

# INTRODUCCIÓN

Las bases de datos son un conjunto de información organizada y almacenada en un dispositivo, entre sus principales funciones se encuentran: mostrar la información requerida por solicitud de una consulta enviada, ya sea específica o global, también se pueden modificar los datos, ya sea eliminando, agregando o actualizando la información almacenada en ella. Pero antes de ser implementada para poder ser utilizada por uno o varios usuarios y clientes este software debe ser programado por los desarrolladores de bases de datos.

Existen varias herramientas que han sido diseñadas para gestionar el diseño de las bases de datos con el fin de reducir la cantidad de tiempo invertido en el proceso de la creación de las mismas, actualmente, hay algunos programas diseñados para cumplir esta funcionalidad, como MySQL Workbench, el cual permite emplear una interfaz gráfica para construir el modelo ER.

En este proyecto se programará una aplicación en el lenguaje Java , donde el usuario crea un modelo ER de acuerdo a la base de datos que se desea generar, una vez terminado el modelo, la aplicación será la encargada de convertirlo en un archivo de extensión .sql el cual contiene las líneas de código necesarias para crear la base de datos deseada, siendo compatible con MySQL 6.3. Este proceso de conversión hará que el usuario ya no tenga que realizar el paso de verificar la sintaxis y tener que ordenar las líneas de código, los cuales habitualmente, suelen tener errores humanos; ese cambio del proceso de escritura directa del código SQL al diseño del modelo ER, hará la tarea más simple y rápida.

A pesar de que son varias las herramientas que cumplen esta función, aún no se ha visto en el lenguaje de programación de Java, por lo tanto es importante ampliar este proceso de conversión en otros lenguajes para que más usuarios tengan acceso a este tipo de herramientas.

Se hizo una búsqueda aplicada donde se encontraron diferentes tipos de herramientas creadas con el mismo objetivo, en base a esto se determina que se debe utilizar una interfaz gráfica, usando las bibliotecas de Java.swing y Java.awt donde se puede representar el modelo ER con sus elementos como las relaciones y las tablas con sus campos, para que el usuario lo manipule dentro de los parámetros necesarios para estructurar correctamente el código generado para posteriormente ser ejecutado correctamente en MySQL Workbench 6.3.

# CAPITULO I

## DESCRIPCIÓN DEL PROYECTO

### I. Presentación del problema de investigación

Formulación del problema:

La implementación de cualquier base de datos requiere dos pasos fundamentales: el diseño y la programación SQL, el diseño tiene una correspondencia directa con el lenguaje SQL que es desaprovechada por los responsables de su desarrollo. El problema de investigación plantea cómo traducir de un diseño modelo Entidad Relación a una secuencia de instrucciones en SQL.

Descripción del contexto:

La traducción de un lenguaje gráfico a un lenguaje escrito corresponde a un problema de Ingeniería de software, es decir el conjunto de técnicas de programación que permiten resolverlo empleando conceptos de programación orientada a objetos, diagrama de clases UML, diseño de casos de uso, entre otros, así como la aplicación de los conceptos de bases de datos y su analogía con el lenguaje SQL.

Descripción de la dificultad: Existen herramientas que resuelven el problema, sin embargo su naturaleza impide el acceso tanto al código fuente como a su documentación del diseño. Este inconveniente plantea dos dificultades: primero el tiempo necesario para lograrlo y segundo los conceptos necesarios para su implementación requieren un nivel avanzado de programación y de conocimientos de Diseño de Software.

Para solucionar esto el programa contará con dos elementos principales. El primero es la interfaz gráfica, siendo el lugar donde tiene lugar la gestión y visualización del modelo entidad relación, donde el usuario crea las tablas, añade los campos y establece las relaciones entre las tablas. Esta información será guardada por el programa haciendo uso de varios objetos de tipo ArrayList. El segundo elemento utilizado es un algoritmo que cumple la función de generar el código SQL, donde se tendrá acceso a la información contenida en los arreglos, para organizarla y transformarla en las líneas de código, y posteriormente se llevarán a un nuevo documento .sql preparado para ser usado en MySQL Workbench 6.3.

- Descripción del Contexto: Escribir el código para crear una base de datos en MySQL workbench 6.3 puede tomar mucho tiempo para el o los programadores, y se convierte en un desafío si el periodo asignado a resolver esta actividad es corto, por ejemplo si en el lugar de trabajo no se le da la prioridad suficiente a este tipo de tareas para cumplirlas satisfactoriamente.
- Descripción de la dificultad: Existen otras herramientas de generación de código SQL, pero algunas de ellas están obsoletas, otras son de código privativo y no hay alguna herramienta divulgada que esté programada en el lenguaje Java.
- Formulación de la pregunta problema: Si por algún motivo, tengo poco tiempo disponible ¿puedo crear un código fuente para crear mi base de datos utilizando Java y de una manera fácil y rápida?.

## II. Justificación

Escribir todo el código para generar una base de datos desde el principio es una tarea que puede implicar una gran cantidad de tiempo para el programador, dependiendo del tamaño de ésta. Parte de ese tiempo se invierte en corregir errores humanos, como puede ser la sintaxis incorrecta o la organización de las líneas de código. Teniendo en cuenta a las personas con un horario corto para la realización de estas actividades y en otros casos simplemente no saben o no se encuentran interesados o son principiantes en cuanto a la programación en lenguaje SQL y necesitan una base de datos, agilizar el proceso para reducir el tiempo de esta labor se establece como la prioridad más importante para este proyecto.

Este proyecto será una herramienta adecuada para cualquier usuario que desee crear una base de datos sin la necesidad de digitar todo el código fuente, en vez de esto, será con la diagramación del modelo ER, lo cual disminuye la cantidad de trabajo realizado y de tiempo transcurrido, se aumentará la productividad, la facilidad de esta tarea y se logrará una mayor comprensión de la base de datos. El usuario podrá obtener fácilmente el código fuente listo para ejecutar en MySQL workbench 6.3, sin la necesidad de verificar y corregir los errores de sintaxis ya que el programa será el encargado de generar el código funcional. Se contará con una interfaz gráfica simple y minimalista para que la tarea de generar las tablas, atributos y relaciones sea mucho más sencilla para el consumidor.

Este proyecto está dirigido principalmente para cualquier desarrollador de bases de datos y entusiasta del tema con el objetivo de obtener el código SQL a partir del un modelo ER creado por ellos mismos para poder crear su propia base de datos.

### III. **Objetivos**

#### A. *Objetivo General*

- Crear un archivo de extensión SQL con el código compatible y listo para usar en MySQL workbench 6.3, por medio del programa Gráficoódigo MySQL.

#### B. *Objetivos Específicos*

- Interpretar el modelo entidad relación hecho en el programa por el usuario para generar el código SQL.
- Diseñar una interfaz gráfica utilizando java para insertar las tablas y relaciones.

# CAPITULO II

## MARCO TEÓRICO

Existe el programa gratuito y de código abierto llamado DBDesigner, este software exporta el código solamente para bases de datos de tipo MySQL a partir del modelado del modelo ER, está programado en el lenguaje Pascal y tiene como última versión la 4.0.3.15, la cual se encuentra publicada en su sitio web. Este programa se define como "DBDesigner 4 es un sistema visual de diseño de bases de datos que integra el diseño, modelado, creación y mantenimiento de bases de datos en un entorno único y sin interrupciones.

Combina características profesionales y una interfaz de usuario clara y simple para ofrecer la forma más eficiente de manejar sus bases de datos.

DBDesigner 4 se compara con productos como Oracle's Designer®, Rational Rose® de IBM, ERwin® de Computer Associates y DataArchitect® de la compañía, pero es un proyecto de código abierto disponible para Microsoft Windows® 2k / XP y Linux KDE / Gnome. Se lanza en la GPL." (fabforce.eu, 2018). Pero tiene el inconveniente de que actualmente ya llegó al fin de su desarrollo, por ésta carencia y también por la falta de mantenimiento se está convirtiendo en un programa obsoleto. Adicionalmente, en la página web oficial el enlace de descarga tanto del programa como del código fuente ya no se encuentran disponibles, aunque en GitHub se encuentran: el código con licencia de tipo GPL-2.0 y algunos proyectos derivados (forks) de DBDesigner.

MySQL Workbench es una herramienta programada en C++, C# y Objective-C, diseñada para convertir el modelo ER en un archivo .sql. Este programa es el sucesor del software anteriormente mencionado, DBDesigner 4 de fabFORCE.net. Es el reemplazo del anterior conjunto de software, MySQL GUI Tools Bundle. Éste trae incluida una herramienta integrada para generar bases de datos compatibles con MySQL donde el usuario, por medio de una interfaz gráfica crea su propio modelo ER para la base de datos deseada o personalizando las plantillas predeterminadas, cuando desea generar el código sql, inicia un asistente que lo guiará en establecer unos parámetros para poder terminar de configurar el modelo entidad relación a su gusto. MySQL Workbench compila ese modelo ER y luego lo transforma en el código de lenguaje de consulta estructurado que el usuario necesita para ejecutarlo. Este programa fue desarrollado en el año 2006.

Existe una página web llamada <https://www.dbdesigner.net/> que cumple una tarea similar que propone este anteproyecto de grado, ofrece un servicio para crear el código de una base de datos. "DbDesigner es una aplicación web que le permite diseñar su esquema de base de datos sin escribir SQL. Nuestra aplicación web es la herramienta más avanzada y estable disponible en el mercado que le brinda el poder del diseño visual de bases de datos en línea. Comience desde cero al igual que crear un diagrama UML o importar una base de datos existente. Utilice nuestras potentes funciones para ahorrar tiempo y reducir errores. Comparte y colabora con tu equipo. Cuando haya terminado, genere un script SQL con un clic para cualquiera de los siguientes motores de base de datos compatibles" (dbdesigner.net, 2019). Teniendo la ventaja de generar el código fuente no solamente para MySQL, también para otros sistemas de administración de bases de datos como Oracle, MSSql, SQLite, PostgreSQL y Microsoft SQL Server, pero tiene las siguientes desventajas:

la más destacable es que se debe tener conexión a internet y se debe estar registrado con el correo electrónico, solamente se pueden guardar hasta 3 plantillas de bases de datos para un usuario que use el servicio de manera gratuita. No es un programa de código abierto.

También se encuentra el programa SQLyog, programado en C++, compatible con el sistema operativo Microsoft Windows. "SQLyog es la herramienta de administración de MySQL más potente para DBA, desarrolladores y arquitectos de bases de datos. SQLyog proporciona herramientas de monitoreo y administración para bases de datos relacionales de código abierto. Las soluciones de Webyog incluyen SQL Diagnostic Manager para la optimización del rendimiento de MySQL y SQLyog para la administración de MySQL. Más de 35,000 compañías (incluidas Amazon, IBM, Salesforce, AT&T, eBay y GE) y 2.5 millones de usuarios confían en las soluciones de Webyog para proporcionar información valiosa sobre sus bases de datos." (webyog.com, 2019). Entre sus herramientas cuenta con un programa para generar el código de base de datos a partir del modelo ER. Cuenta con una versión gratuita, una versión de 14 días de prueba y una versión de pago. El programa sigue actualmente en desarrollo activo.

El programa Visual Paradigm es una herramienta multiplataforma, ya que es compatible con los sistemas operativos: MacOS, Linux y Microsoft Windows, cumple con las mismas funciones de los programas mencionados anteriormente, pero a diferencia de éstos, Visual Paradigm es un programa muy completo, debido a la cantidad de herramientas que no solo incluyen la generación de una base de datos a partir del modelo ER sino otras características de modelado como lo son las conversiones de los diagramas UML a las líneas de código de un software para los lenguajes de programación más utilizados. "Entity Relationship Diagram (ERD), a database design tool that provides graphical representation of database tables, their columns and inter-relationships. ERD is the most popular database design tool. A well-developed ERD can provide sufficient information for database administrator to follow when developing and maintaining database" (visual-paradigm.com, 2019). Incluye una versión gratuita, de uso no comercial llamada Community edition, también las versiones de pago llamadas Modeler Edition, Standard Edition, Professional Edition y Enterprise Edition

#### **IV. Bases teóricas o fundamentos conceptuales**

Por otro lado, en investigaciones recientes como Zapata(2011) se realiza una conversión del lenguaje natural al modelo ER y al código sql utilizando el lenguaje de programación controlado de la universidad Nacional de Colombia UN-Lencep (Universidad Nacional de Colombia—Lenguaje Controlado para la Especificación de Esquemas Preconceptuales) cercano al lenguaje natural y de fácil comprensión para el usuario interesado. esto para obtener automáticamente, el diagrama entidad-relación y su correspondiente representación en código SQL. Este lenguaje hace uso de la heurística y de la probabilidad para poder traducir el lenguaje natural y generar el código de la base de datos correspondiente al modelo ER que el usuario realice.

En la investigación realizada por Nazlia(2004) se establecen una serie de nuevas heurísticas que se utilizaron en la generación del modelo ER con el fin de tener completo el modelamiento de las bases de datos a partir de una descripción hecha por una persona en lenguaje natural y que describen la implementación de dicho sistema llamado ER-Converter.

# CAPITULO III

## DISEÑO METODOLÓGICO

Para obtener el resultado esperado, primero se creó una interfaz gráfica, hecha con las bibliotecas de Java.swing y Java.awt, en la parte superior cuenta con un menú de opciones, una de ellas es la opción de crear una tabla, al dar a esa opción usando una ventana emergente tipo JOptionPane donde se le pide al usuario que le asigne un nombre a la tabla, luego de esto se muestra en el lienzo de la ventana, la tabla creada, esta tabla contiene dos botones, el primero es para borrar la tabla y el segundo botón es para crear un campo, este campo está compuesto por un JTextField y un JButton. En el JTextField, el usuario debe ingresar el nombre del campo o atributo para poder editar el campo y posteriormente asignarle propiedades, al oprimir el botón de editar el campo, se abrirá un menú hecho como un JPanel, donde se alojan una serie de JLabel, JButton, JCheckBox, Y JComboBox.

Las opciones que tiene el usuario para asignar propiedades o modificar el campo son las siguientes:

1. Un botón para eliminar el campo.
2. Una lista para elegir el tipo de variable.
3. Un cajón de texto para asignar la longitud máxima que pueda tener ese campo.
4. Un check para asignar el campo como llave primaria.
5. Un check para establecer el incremento automático.
6. Un check para no permitir que el campo sea nulo.
7. Establecer el campo como una llave foránea.

Si se marca el check 7 se habilitarán otras dos listas con las cuales el usuario termina de configurar la llave foránea

1. Elegir la tabla referida.
2. Elegir el campo de la tabla referida con el cual se quiera establecer conexión.

## V. Tipo de investigación

La investigación realizada es de tipo aplicada, debido a que se busca solucionar el problema de investigación, lo cual es algo específico, pero al mismo tiempo, teniendo en cuenta las otras herramientas que ya existen para tal fin, cuales son sus ventajas e inconvenientes, en qué lenguajes de programación fueron escritos, las investigaciones publicadas sobre la solución al problema y los resultados obtenidos. También se probaron para ver la estructura de la interfaz gráfica, las particularidades frente a los otros programas, si es o no compatible con MySQL Workbench 6.3, el tipo de licencias que ofrece y si es software libre o privativo.

Esta búsqueda de información fue documental, ya que se buscaron publicaciones y documentos relacionados al tema de generación del código SQL a partir del modelo ER, además de investigar la documentación de los programas mencionados anteriormente.

## VI. Población

Se establecieron dos tipos de usuarios para los cuales está diseñada la aplicación Gráficoódigo MySQL.

El primer grupo de personas está conformado por las personas que saben programar bases de datos, pero que necesitan darle celeridad al proceso ya que el periodo de tiempo designado para esta labor podría ser insuficiente debido al gran tamaño que pueden alcanzar sus bases de datos, dependiendo el entorno donde se vaya a implementar, por ejemplo, el corporativo, o por lo menos complejas o muy extensas al momento de escribir línea por línea.

El segundo grupo y posiblemente el más beneficiado, por el contrario al caso anterior, no sabe programar en SQL o tienen conocimientos muy básicos sobre el tema, se identificó que no necesitan bases de datos complejas como las que pueden haber en una empresa, sino más bien una base de datos pequeña, y sea para un proyecto personal o una base de datos muy básica o con el fin de aprender a diagramar el modelo ER usando modelos simples y básicos.

## VII. Técnicas e instrumentos de recolección de datos

Para encontrar las herramientas que cumplen la misma función que Gráficoódigo MySQL, se hizo una búsqueda en la base de datos de la página de internet (<https://alternativeto.net/>), se han encontrado algunos de los programas mencionados en este documento. En cuanto a servicios web que cumplan la misma función ,en la base de datos del motor de bpúsqueda de Bing se encontró el sitio (<https://blog.dbdesigner.net/>), además se revisó la documentación de MySQL Workbench.

Para los artículos e investigaciones se buscó en la base de datos de EBSCO, en Ebarry, en IEEE Xplorer Digital Library, en a base de datos de la Universidad Nacional y en Google académico.

En los documentos y programas consultados se concluye que son fundamentales para este trabajo de grado, pues estos cumplen satisfactoriamente con el objetivo de generar el código SQL y solucionar el problema de investigación, estando relacionada con la pregunta de investigación.

## CAPITULO III

# RESULTADOS DE LA INVESTIGACIÓN

Se llevó a cabo la creación de varios ArrayList, uno para guardar la información de las tablas, dentro de este ArrayList hay otro arreglo donde serán almacenados los campos de cada tabla y dentro de cada campo, un tercer arrayList donde permanecen las propiedades de cada campo, por ejemplo, el tamaño y el tipo de variable que se debe utilizar para con ese campo.

También se creó otro ArrayList para almacenar las relaciones entre tablas, donde se guarda la información correspondiente a las llaves foráneas, la tabla dependiente, la tabla independiente, si es una relación uno a muchos, muchos a muchos o uno a uno.

Se utiliza la sentencia for de forma anidada, para recorrer la información del esquema que contiene las tablas, en cada campo del ArrayList hay otro for para leer los campos de las tablas, y otro for para leer las propiedades de los campos, estos se van sumando a una variable de tipo String, en la cual se van guardando las líneas de código de la fuura base de datos.

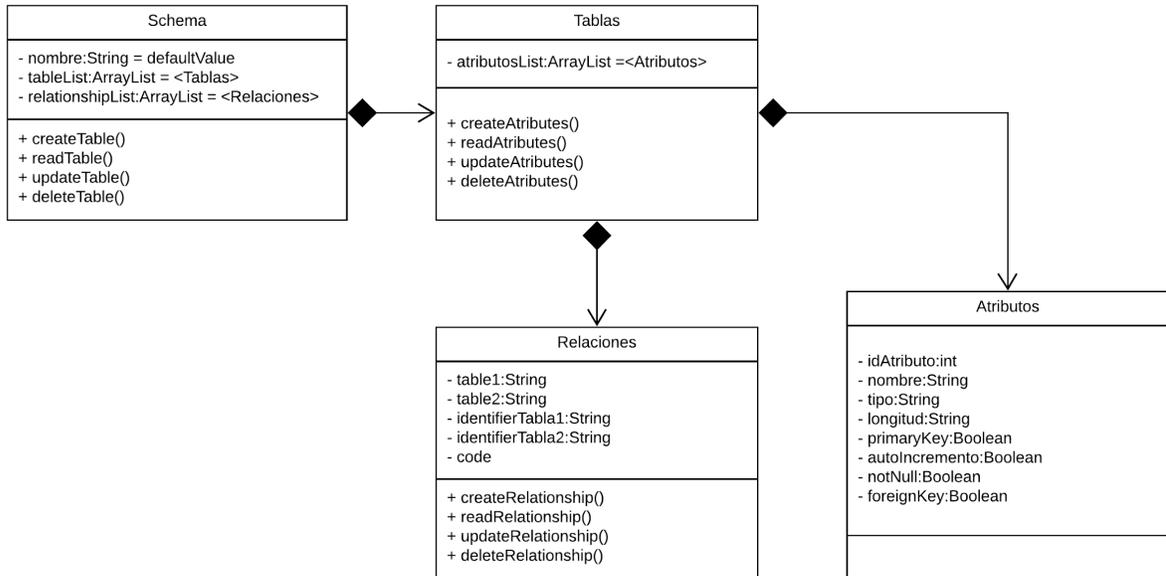


Figura 1. Diagrama de clases UML.

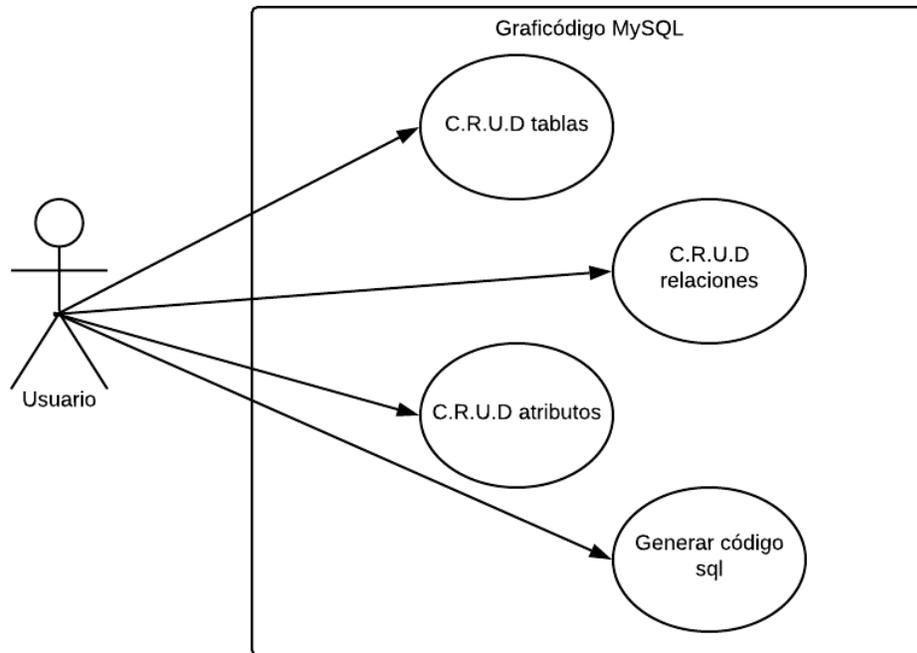


Figura 2. Diagrama de casos de uso UML.

<b>ID</b>	1	
<b>Nombre</b>	Nombrar base de datos	
<b>Actores</b>	Usuario	
<b>Descripción</b>	El usuario ingresa el nombre de la base de datos	
<b>Evento activador</b>	El usuario da click en el botón "Nombre DB"	
<b>Precondiciones</b>	Ninguna	
<b>Poscondiciones</b>	Ninguna	
<b>Flujo Normal</b>	<b>Actor</b>	<b>Sistema</b>
		1. Muestra el menú "Herramientas".
	2. Abre el menú	
		3. Muestra el botón "Nombre DB" y "Nombre Tabla".
	4. Oprime el botón "Nombre DB".	
		5. Muestra una ventana para escribir el nombre de la base de datos.
	6. Ingresar el nombre de la base de datos.	
	7. Oprime el botón aceptar	
		8. Verifica si el nombre de la base de datos es un campo nulo
		9. Guarda el nombre de la base de datos.

Tabla 1: Caso de uso "Nombrar base de datos".

<b>ID</b>	2	
<b>Nombre</b>	Nombrar tabla	
<b>Actores</b>	Usuario	
<b>Descripción</b>	El usuario ingresa el nombre de la base de datos	
<b>Evento activador</b>	El usuario da click en el botón "Nombre Tabla"	
<b>Precondiciones</b>	Ninguna	
<b>Poscondiciones</b>	Ninguna	
<b>Flujo Normal</b>	<b>Actor</b>	<b>Sistema</b>
		1. Muestra el menú "Herramientas".
	2. Abre el menú	
		3. Muestra el botón "Nombre Tabla".
	4. Ingresar el nombre de la tabla.	
	5. Oprime el botón Aceptar	
		6. Verifica si el nombre de la tabla es vacío.
		7. Guarda el nombre de la tabla.

Tabla 2: Caso de uso "Nombrar tabla".

<b>ID</b>		3
<b>Nombre</b>	Crear relación	
<b>Actores</b>	Usuario	
<b>Descripción</b>	El usuario ingresa el nombre de la base de datos	
<b>Evento activador</b>	El usuario da click en el botón "Crear relación"	
<b>Precondiciones</b>	Ninguna	
<b>Poscondiciones</b>	Ninguna	
<b>Flujo Normal</b>	<b>Actor</b>	<b>Sistema</b>
		1. Muestra el menú "Herramientas".
	2. Abre el menú	
		1. Muestra el botón "Crear Relación".
	2. Elige la tabla independiente.	
	3. Elige la tabla dependiente.	
	4. Oprime el botón Aceptar	
		3. Guarda la relación.
		4. Muestra gráficamente la relación.

Tabla 3: Caso de uso "Crear relación".

<b>ID</b>		4
<b>Nombre</b>	Generar código	
<b>Actores</b>	Usuario	
<b>Descripción</b>	El usuario ingresa el nombre de la base de datos	
<b>Evento activador</b>	El usuario da click en el botón "Generar Código"	
<b>Precondiciones</b>	Ninguna	
<b>Poscondiciones</b>	Ninguna	
<b>Flujo Normal</b>	<b>Actor</b>	<b>Sistema</b>
		1. Muestra el menú "Archivo".
	2. Abre el menú	
		3. Muestra el botón "Generar Código".
	4. Oprime el botón.	
		5. Genera el código y lo muestra en pantalla
	5.1. Oprime el botón "Copiar código"	
	5.2 Oprime el botón "Exportar como .sql"	

Tabla 4. Caso de uso "Generar código".

## VIII. Resultados del objetivo específico no. 1

Se crea un JPanel usado como lienzo, el espacio de trabajo donde se insertarán las relaciones, se añaden los JPanel correspondientes a las tablas, estas tablas se pueden arrastrar por el JPanel principal. Se crean las líneas y puntas correspondientes a las relaciones entre dos tablas.

Además, se usa un menú para que el usuario edite el nombre de la base de datos, el nombre de la tabla y las propiedades de los atributos.

## **IX. Resultados del objetivo específico no. 2**

Se usa la clase ArrayList para almacenar la información sobre cada tabla, asimismo con sus atributos y sus relaciones. Al momento de que el usuario haya terminado el modelamiento de su base de datos, y decida generar el código .sql, el software Gráficoódigo MySQL hará un recorrido por todos los ArrayList e irá convirtiendo esta información en las líneas de código, se almacenarán en un String para posteriormente guardarse en un TextArea o en el archivo de texto plano de extensión .txt.

# CAPÍTULO V.

## CONCLUSIONES Y RECOMENDACIONES

La herramienta facilita la creación de bases de datos de forma gráfica frente a digitar una por una las sentencias de Definición de Datos (DDL).

La interfaz gráfica tiene un diseño simple que permite al usuario acceder fácilmente a las opciones para interactuar de manera óptima con el programa.

Aunque existen herramientas similares ésta se diferencia por la interfaz gráfica y el uso de licencia.

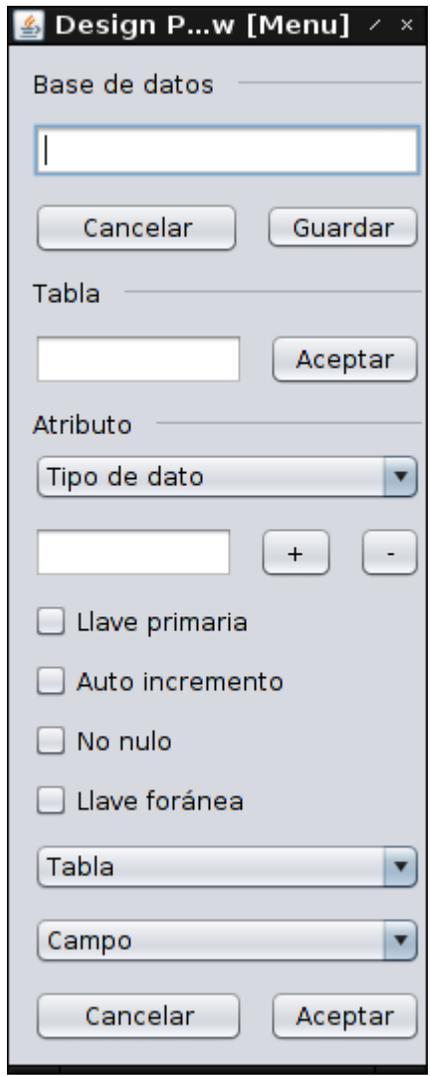


Figura 3. Menú de configuración.

# BIBLIOGRAFÍA

## Referencias

- [1] V. Medina Kern y A. L. Moraes Ramos, «Bridging the gap between natural and information modeling languages: an informal approach to information modeling learning».
- [2] C. M. Zapata Jaramillo, G. González Calderón y J. J. Chaverra Mojica, «GENERACIÓN AUTOMÁTICA DEL DIAGRAMA ENTIDAD-RELACIÓN Y SU REPRESENTACIÓN EN SQL DESDE UN LENGUAJE CONTROLADO (UN-LENCEP),» Ingenierías Universidad de Medellín, vol. 10, nº 18, pp. 127-135, 2011.