

Diseño e Implementación de un Sistema de Gestión para Carteleras Cinematográficas con MySQL y VBA Excel

Design and Implementation of a Management System for Cinema Showtimes Using MySQL and VBA Excel

  Campoverde Huaman Daniel Enrique ¹,   Hoyos Llanos Yhordan Job ¹,
  Pineda Morales Joseph Steve ¹,   Suclupe Tirado Victor Manuel ¹,
  Verastegui Paredes Zucet Abigail ¹

¹ Escuela de Ingeniería Informática, Universidad Nacional de Trujillo, La Libertad, Perú.

*Autor correspondiente: decampoverdeh@unitru.edu.pe (D-CampoverdeH)

RESUMEN

Este artículo describe el desarrollo de un sistema para la administración de varias carteleras cinematográficas mediante la integración de una base de datos MySQL con una interfaz de usuario implementada en VBA Excel. El sistema permite gestionar de manera eficiente la información relacionada con películas, participantes de las producciones, cines, salas, funciones, promociones, entradas y opiniones de los espectadores. La metodología empleada comprendió el análisis de requerimientos, el diseño del modelo Entidad-Relación, su transformación al modelo físico, la implementación de la base de datos en MySQL y el desarrollo de formularios en VBA Excel conectados mediante ODBC. Asimismo, se aplicaron procesos de normalización para garantizar la integridad y consistencia de los datos. Como resultado, se obtuvo una aplicación capaz de realizar operaciones de registro, consulta, actualización y análisis de información, además de generar indicadores como el promedio de calificaciones de las películas, el monto total invertido en cada producción y reportes estadísticos sobre asistencia, promociones y preferencias del público. Se concluye que la integración de MySQL y VBA Excel constituye una alternativa eficiente para la administración de una cartelera cinematográfica, al facilitar la organización de la información, optimizar los procesos de gestión y proporcionar datos relevantes para la toma de decisiones.

Palabras clave: Cartelera cinematográfica, MySQL, VBA Excel, Base de datos, Sistema de gestión.

ABSTRACT

This study aimed to develop a movie theater management system by integrating a MySQL database with a user interface implemented in VBA Excel. The system efficiently manages information related to movies, production participants, cinemas, screening rooms, showtimes, promotions, tickets, and audience reviews. The methodology included requirements analysis, Entity-Relationship modeling, transformation into relational and physical database models, implementation of the MySQL database, and the development of VBA Excel forms connected through ODBC. Database normalization techniques were also applied to ensure data integrity and consistency. As a result, an application capable of performing data registration, querying, updating, and

analysis was obtained, generating indicators such as average movie ratings, total production investment, and statistical reports on attendance, promotions, and audience preferences. It is concluded that the integration of MySQL and VBA Excel provides an efficient solution for managing a movie theater schedule by improving information organization, streamlining management processes, and supporting decision-making through reliable data analysis.

Keywords: *Movie theater management, MySQL, VBA Excel, Database, Management system*

1. INTRODUCCIÓN

En la actualidad, las tecnologías de la información desempeñan un papel fundamental en la administración de empresas y organizaciones, permitiendo optimizar procesos, reducir errores y mejorar la disponibilidad de la información. En el sector cinematográfico, la gestión de una cartelera involucra el manejo de grandes volúmenes de datos relacionados con películas, funciones, salas, promociones, ventas de entradas y opiniones de los espectadores. Cuando estos procesos se realizan de forma manual o mediante herramientas poco integradas, pueden presentarse inconvenientes como la redundancia de información, dificultades para generar reportes y una menor eficiencia en la toma de decisiones.

Diversos relacionales como una solución para garantizar la organización, integridad y disponibilidad de la información [1]. Asimismo, herramientas como MySQL, por su confiabilidad y rendimiento, y Excel VBA, por su facilidad para desarrollar interfaces de usuario, constituyen una alternativa accesible para implementar aplicaciones orientadas a la administración de datos en organizaciones de pequeña y mediana escala.

En este contexto, el presente trabajo propone el desarrollo de un sistema para la administración de una cartelera cinematográfica mediante la integración de una base de datos MySQL y una interfaz desarrollada en Excel VBA, conectadas a través de ODBC. El sistema permite gestionar la información de películas, participantes de las producciones, cines, salas, funciones, promociones, ventas de entradas y opiniones del público, además de generar reportes estadísticos que apoyan el análisis de la información y la toma de decisiones.

El objetivo de este artículo es presentar el diseño e implementación de la solución propuesta, describiendo el proceso de modelado de la base de

datos, el desarrollo del proyecto y las funcionalidades implementadas para mejorar la administración de una cartelera cinematográfica mediante una herramienta organizada, eficiente y de fácil utilización.

1.1 Fundamento Teórico

A. Sistemas de Gestión de Bases de Datos (SGBD)

Un Sistema de Gestión de Bases de Datos (SGBD) es un sistema informático diseñado para almacenar, organizar, administrar y recuperar información de manera eficiente [2]. Permite gestionar grandes volúmenes de datos garantizando su integridad, seguridad y consistencia, además de facilitar el acceso simultáneo por múltiples usuarios. Los SGBD relacionales organizan la información en tablas relacionadas mediante claves primarias y foráneas, lo que reduce la redundancia de datos y mejora la eficiencia en las consultas [3].

B. MySQL como gestor de bases de datos

MySQL es un sistema de gestión de bases de datos relacional de código abierto ampliamente utilizado en el desarrollo de aplicaciones. Emplea el lenguaje SQL (Structured Query Language) para la creación, manipulación y consulta de datos [4]. Entre sus principales características destacan su estabilidad, escalabilidad, facilidad de administración y compatibilidad con diferentes plataformas y lenguajes de programación, lo que lo convierte en una alternativa adecuada para sistemas de gestión de información [5].

C. VBA Excel para el desarrollo de interfaces

Visual Basic for Applications (VBA) es un lenguaje de programación integrado en Microsoft Excel que permite automatizar tareas y desarrollar aplicaciones mediante formularios interactivos [6]. A través de controles como cuadros de texto, botones y listas, es posible diseñar interfaces amigables que facilitan la captura, consulta y actualización de información. Su integración con Excel permite desarrollar aplicaciones de escritorio sin necesidad de software adicional.

D. Modelo Entidad-Relación y modelo relacional

El modelo Entidad-Relación (E-R) es una técnica de diseño conceptual utilizada para representar la estructura de una base de datos mediante entidades, atributos y relaciones [7]. Este modelo facilita la identificación de los elementos que intervienen en un sistema y la forma en que interactúan entre sí. Posteriormente, el modelo E-R se transforma en un modelo relacional, donde la información se organiza en tablas relacionadas mediante claves primarias y claves foráneas, permitiendo una implementación eficiente en sistemas gestores de bases de datos relacionales [8] [9].

E. Conectividad mediante ODBC

Open Database Connectivity (ODBC) es un estándar que permite establecer comunicación entre aplicaciones y diferentes sistemas gestores de bases de datos mediante controladores específicos [10]. Esta tecnología proporciona una interfaz común para ejecutar consultas, insertar, modificar y recuperar información desde aplicaciones externas, favoreciendo la interoperabilidad entre herramientas de desarrollo y bases de datos.

F. Álgebra Relacional

El álgebra relacional es un lenguaje formal de consulta para bases de datos relacionales, propuesto por Edgar F. Codd, que proporciona un conjunto de operaciones para manipular y recuperar información almacenada en tablas [11]. Estas operaciones, como selección, proyección, unión, diferencia, producto cartesiano y combinación (join), permiten expresar consultas de manera lógica e independiente de su implementación [12]. El álgebra relacional constituye la base teórica del

lenguaje SQL y es fundamental para el diseño y optimización de consultas, ya que facilita la obtención de información precisa a partir de relaciones entre los datos.

2. MATERIALES Y METODOS

El desarrollo del sistema se llevó a cabo mediante las etapas metodológicas que se describen a continuación.

2.1 MATERIALES

Para el desarrollo del sistema de administración de una cartelera cinematográfica se utilizaron recursos de hardware y software que permitieron el diseño, implementación y validación del proyecto. Entre los materiales empleados se encuentran una computadora personal con sistema operativo Windows, el sistema gestor de bases de datos MySQL para el almacenamiento y administración de la información, Microsoft Excel con Visual Basic for Applications (VBA) para el desarrollo de la interfaz de usuario, y el controlador Open Database Connectivity (ODBC) para establecer la comunicación entre la aplicación y la base de datos. Asimismo, se utilizaron herramientas de modelado para el diseño del modelo Entidad-Relación, el modelo relacional y el modelo físico de la base de datos, además de un editor de código para la implementación y ejecución de las consultas SQL y los procedimientos necesarios para el funcionamiento del sistema.

2.2 METODOS

A. Diseño de la base de datos

Se elaboró el Modelo Entidad-Relación (E-R) para representar gráficamente las entidades, sus atributos y las relaciones entre ellas.

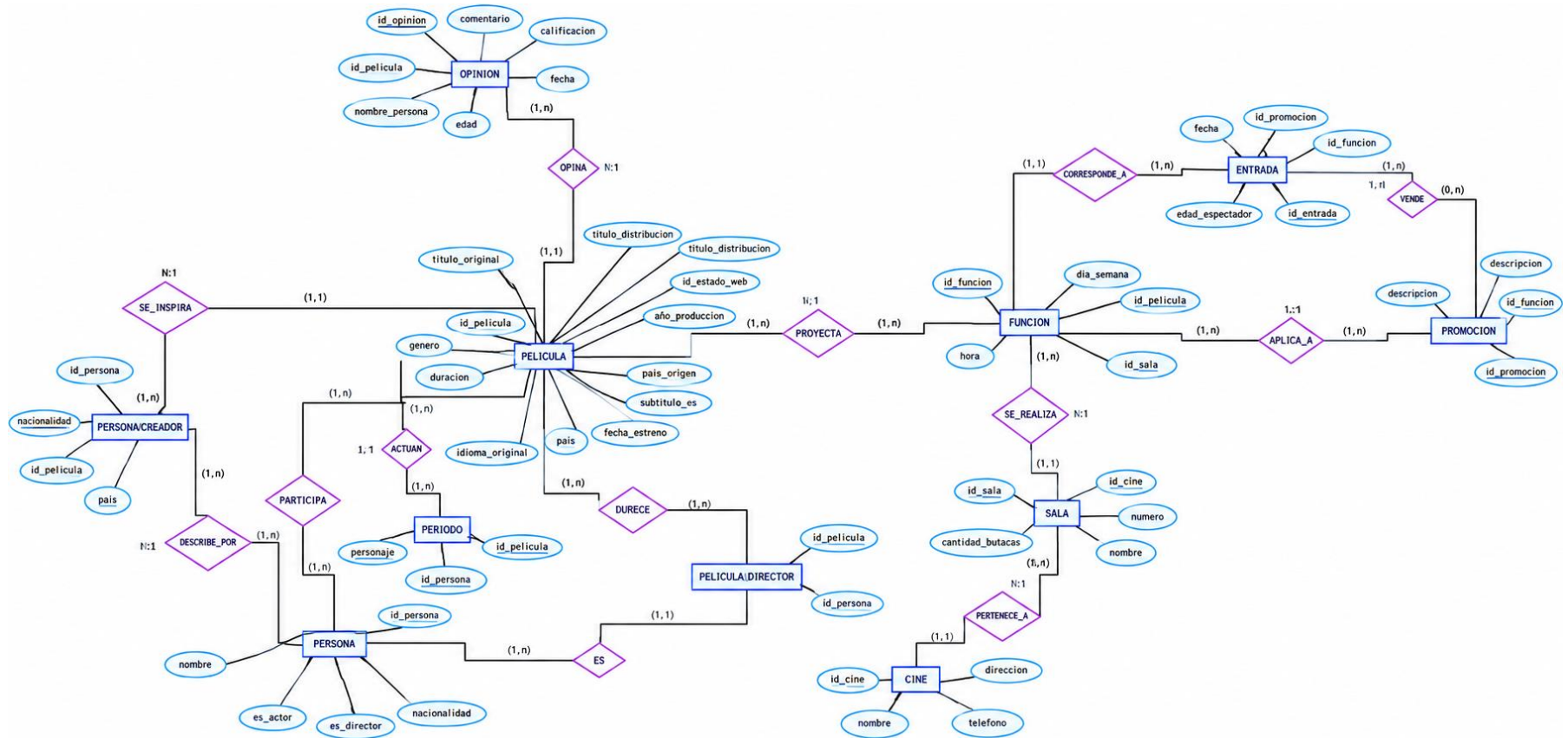


Figura 1. Diagrama Entidad-Relación (E-R) del sistema de gestión de cartelera cinematográfica. Se muestran las entidades, atributos, relaciones y cardinalidades que conforman el diseño conceptual de la base de datos.

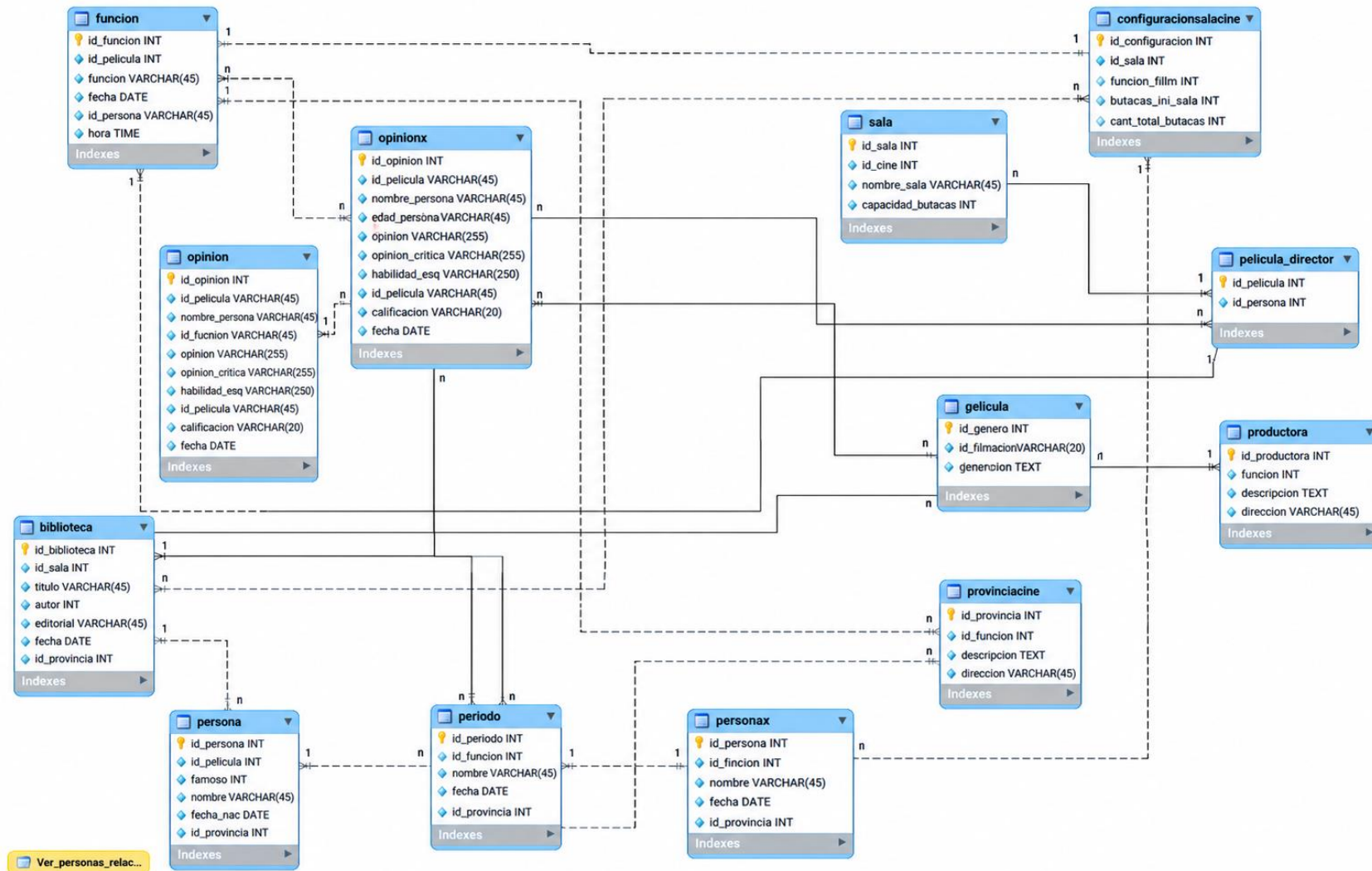


Figura 2. Modelo físico del sistema de gestión de cartelera cinematográfica. Se presentan las tablas junto con sus claves primarias (PK), claves foráneas (FK).

B. Normalización de la base de datos

Se aplicó el proceso de normalización para optimizar la estructura de la base de datos y reducir la redundancia de información. Se verificó el cumplimiento de la Primera Forma Normal (1FN), Segunda Forma Normal (2FN), Tercera Forma Normal (3FN) y la Forma Normal de Boyce-Codd (FNBC), garantizando la consistencia de los datos y evitando anomalías de inserción, actualización y eliminación.

1FN PRIMERA FORMA NORMAL:

- a) **Pelicula_Base:** (id_pelicula, titulo_original, titulo_distribucion, genero, idioma_original, subtítulos_es, año_produccion, url_sitio_web, duracion, calificación_pelicula, fecha_estreno, resumen)
- b) **Pelicula_Pais**(id_pelicula, pais_origen)
- c) **Pelicula_Persona_Rol** (id_pelicula, id_persona, rol, pago, nombre_persona, nacionalidad_persona)
- d) **Reparto** (id_pelicula, id_persona personaje)
- e) **Opinion_Pelicula**(id_pelicula, id_opinion, nombre_opinion, edad_opinion, fecha_opinion, calificación_opinion, comentario_opinion)
- f) **Cine_Base** (id_cine, nombre_cine, direccion_cine, telefono_cine)
- g) **Cine_Sala** (id_cine, numero_sala, id_sala, nombre_sala, cantidad_butacas)
- h) **Funcion_Base** (id_funcion, id_pelicula, id_sala, fecha_funcion, hora_funcion)
- i) **Funcion_Promocion**(id_funcion, id_promocion, descripcion_promocion, descuento)
- j) **Entrada_Base** (id_entrada, id_funcion, fecha_entrada, edad_espectador, id_promocion)

2FN - SEGUNDA FORMA NORMAL

- a) **PersonaPeliculaRol** (id_persona, id_pelicula, rol, pago)
- b) **Persona** (id_persona, nombre_persona, nacionalidad_persona)
- c) **Promoción** (id_promocion, descripcion_promocion, descuento)

- d) **FuncionPromocion** (id_funcion id_promocion)

3FN - TERCERA FORMA NORMAL

- a) **Pelicula** (id_pelicula, titulo_original, titulo_distribucion, genero, idioma_original, subtítulos_es, año_produccion, url_sitio_web, duracion, calificación_pelicula, fecha_estreno, resumen)
- b) **Pais** (id_pais, nombre_pais)
- c) **PelículaPaís** (id_pelicula, pais_origen)
- d) **CINE** (id_cine nombre_cine direccion_cine telefono_cine)
- e) **Sala** (id_sala, id_cine, numero_sala, nombre_sala, cantidad_butacas)
- f) **Funcion** (id_funcion, id_pelicula, id_sala, fecha_funcion, hora_funcion)
- g) **Entrada** (id_entrada, id_funcion, fecha_entrada, edad_espectador, id_promocion)
- h) **Opinion** (id_pelicula, id_opinion, nombre_opinion, edad_opinion, fecha_opinion, calificación_opinion, comentario_opinion)
- i) **Persona** (id_persona nombre_persona nacionalidad_persona)
- j) **PersonaPeliculaRol** (id_pelicula, id_persona, personaje)
- k) **Promocion** (id_promocion, descripcion_promocion, descuento)
- l) **FuncionPromocion** (id_funcion id_promocion)

FNBC – FORMA NORMAL DE BOYCE -Codd

- a) **Pelicula**(id_pelicula, titulo_original, titulo_distribucion, genero, idioma_original, subtítulos_es, año_produccion, url_sitio_web, duracion, calificación_pelicula, fecha_estreno, resumen)
- b) **Pais** (id_pais, nombre_pais)
- c) **PeliculaPaís** (id_pelicula, id_pais)
- d) **Persona** (id_persona, nombre_persona, nacionalidad_persona)
- e) **PersonaPeliculaRol** (id_persona, id_pelicula, rol, pago)
- f) **Cine** (id_cine, nombre_cine, direccion_cine, telefono_cine)
- g) **Sala** (id_sala, id_cine, numero_sala, nombre_sala, cantidad_butacas)
- h) **Funcion**(id_funcion, id_pelicula, id_sala, fecha_funcion, hora_funcion)
- i) **Promocion** (id_promocion, descripcion_promocion, descuento)
- j) **FuncionPromocion**(id_funcion, id_promocion)

- k) **Entrada**(id_entrada, id_funcion, fecha_entrada, edad_espectador, id_promocion)
- l) **Opinion**(id_opinion, id_pelicula, nombre_opinion, edad_opinion, fecha_opinion, calificacion_opinion, comentario_opinion)

C. IMPLEMENTACIÓN EN MYSQL

Una vez validado el diseño, se implementó la base de datos en MySQL mediante la creación de las tablas, claves primarias, claves foráneas y tipos de datos adecuados. Además, se desarrollaron consultas SQL para realizar operaciones de registro, búsqueda, actualización y generación de reportes.

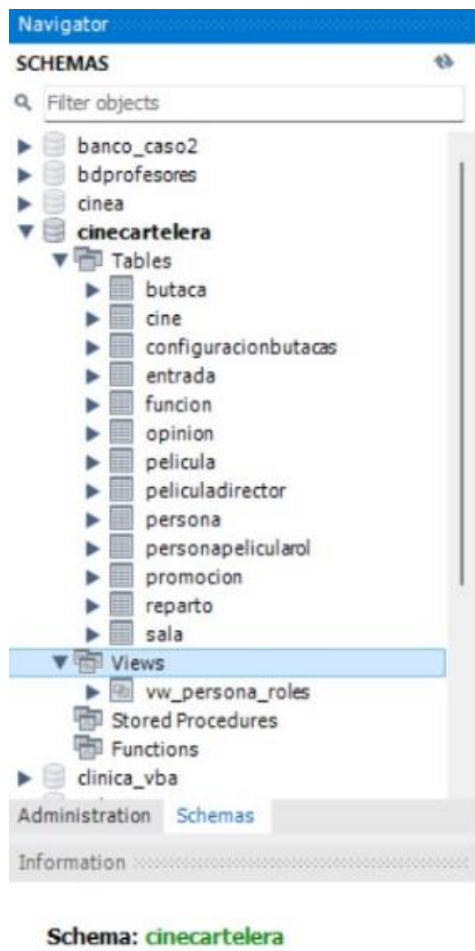


Figura 3. Esquema de la base de datos cinecartelera implementado en MySQL. Se muestran las tablas, la vista creada y la estructura lógica del sistema de gestión de cartelera cinematográfica.

1. Ejemplos de Consultas SQL en MYSQL

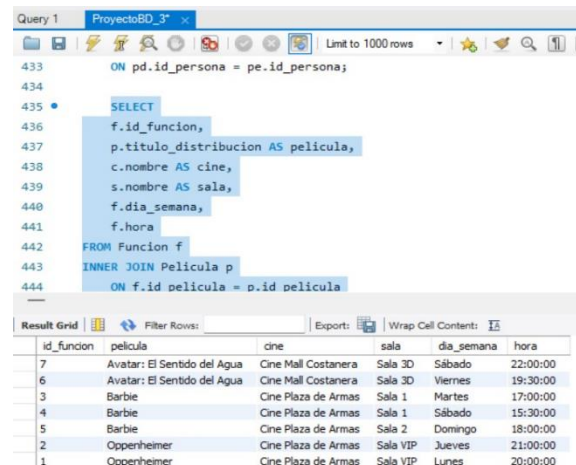


Figura 4. Consulta para obtener la programación de funciones, mostrando la película, el cine, la sala, el día de la semana y el horario mediante la combinación de tablas relacionadas.

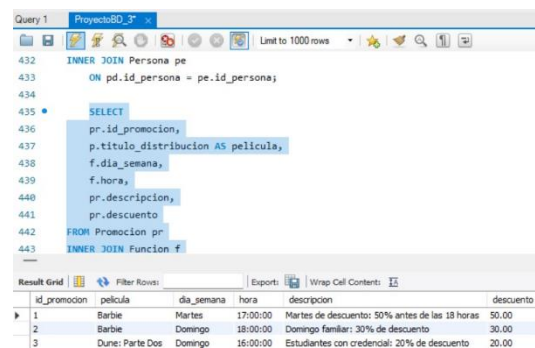


Figura 5. Consulta para obtener la información de las promociones asociadas a las funciones cinematográficas, mostrando la película, el día de la semana, el horario, la descripción y el porcentaje de descuento.

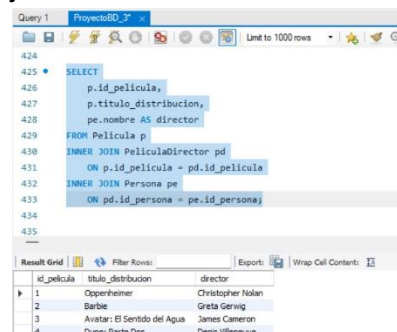


Figura 6. Consulta para obtener la información de las películas y sus respectivos directores mediante la relación entre las tablas Película, PelículaDirector y Persona.

```

1  USE CineCartelera;
2  SELECT *
3  FROM ConfiguraciónButacas
4  WHERE id_sala = 1;
    
```

id_configuración	id_sala	numero_filas	butacas_por_fila	ubicacion_pantalla
1	1	10	12	Arriba

Figura 7. Consulta para obtener la configuración de butacas de una sala de cine, mostrando el número de filas, butacas por fila y la ubicación de la pantalla.

D. Desarrollo de la interfaz en Excel VBA

Se diseñó varias interfaces de usuario utilizando formularios (UserForms) en Microsoft Excel con VBA. Estos formularios permiten ingresar, consultar y actualizar la información almacenada en la base de datos de forma intuitiva, evitando que el usuario interactúe directamente con MySQL.



Figura 8. Estructura del proyecto desarrollado en Excel VBA para el sistema de gestión de cartelera cinematográfica. Se observan los formularios utilizados.

1. Configuración de la conexión ODBC entre VBA Excel y MySQL

Para establecer la comunicación entre la interfaz desarrollada en VBA Excel y la base de datos MySQL, se configuró un origen de datos (DSN) mediante el controlador ODBC.

Esta configuración permitió ejecutar consultas SQL desde la aplicación y garantizar el intercambio de información entre ambos entornos durante la ejecución del sistema.

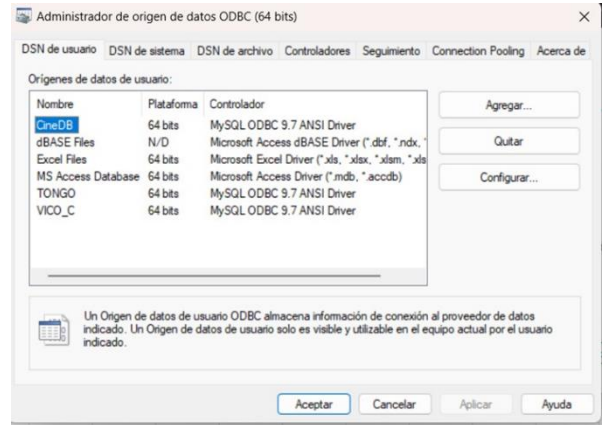


Figura 9: Creación del origen de datos (DSN) para establecer la conexión entre MySQL y VBA Excel.

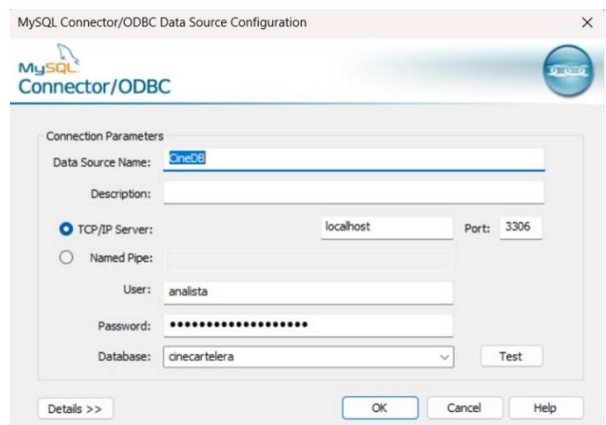


Figura 10: Configuración de los parámetros de conexión del controlador ODBC utilizando los datos de la base de datos MySQL.

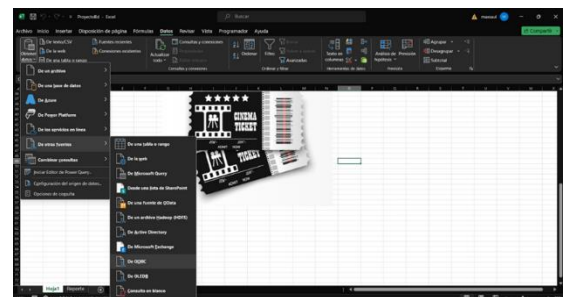


Figura 11: Establecimiento de la conexión desde Microsoft Excel mediante el controlador ODBC.



Figura 12: Verificación de la conexión entre la interfaz desarrollada en VBA Excel y la base de datos MySQL.



Figura 13: Ejecución de la prueba de conexión para validar la comunicación con la base de datos.

E. ALGEBRA Y CALCULO RELACIONAL

1. Seleccionar las películas de género "Ciencia Ficción"

Álgebra relacional:

$\sigma_{\text{genero} = \text{'Ciencia Ficción'}}(\text{Pelicula})$

Cálculo relacional orientado a tuplas:

$\{ t \mid t \in \text{Pelicula} \wedge t.\text{genero} = \text{'Ciencia Ficción'} \}$

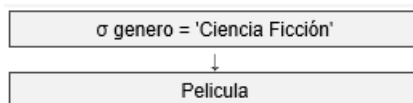


Figura 14: Diagrama de selección de películas de género "Ciencia Ficción"

Tabla 01. Resultado de las películas de género "Ciencia Ficción"

id_pelicula	titulo_distribucion	titulo_original	genero	idioma_original	duracion	cali
3	Avatar: El Sentido del Agua	Avatar: The Way of Water	Ciencia Ficción	Inglés	03:12:00	+9
4	Dune: Parte Dos	Dune: Part Two	Ciencia Ficción	Inglés	02:46:00	+15

2. Obtener el nombre y la nacionalidad de los actores

Álgebra relacional:

$\pi_{\text{nombre, nacionalidad}}(\sigma_{\text{es_actor} = \text{TRUE}}(\text{Persona}))$

Cálculo relacional orientado a tuplas:

$\{ t.\text{nombre}, t.\text{nacionalidad} \mid t \in \text{Persona} \wedge t.\text{es_actor} = \text{TRUE} \}$

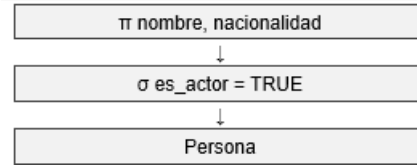


Figura 15: Diagrama de proyección de actores

Tabla 02. Resultado de los actores

nombre	nacionalidad
Cillian Murphy	Irlandés
Robert Downey Jr.	Estadounidense
Margot Robbie	Australiana
Ryan Gosling	Canadiense
Zendaya	Estadounidense
Timothée Chalamet	Estadounidense

3. Obtener las películas con sus directores

Álgebra relacional:

$\pi_{\text{titulo_distribucion, nombre}}(\text{Pelicula} \bowtie \text{PeliculaDirector} \bowtie \text{Persona})$

Cálculo relacional orientado a tuplas:

$\{ p.\text{titulo_distribucion}, per.\text{nombre} \mid p \in \text{Pelicula} \wedge pd \in \text{PeliculaDirector} \wedge per \in \text{Persona} \wedge p.\text{id_pelicula} = pd.\text{id_pelicula} \wedge pd.\text{id_persona} = per.\text{id_persona} \}$

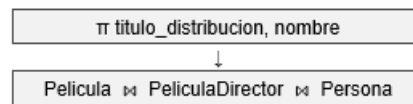


Figura 16. Diagrama de unión películas con directores.

4. Obtener las funciones que cuentan con promoción vigente

Álgebra relacional:

$\pi_{\text{titulo_distribucion, dia_semana, hora, descripcion, descuento}}(\text{Pelicula} \bowtie \text{Funcion} \bowtie \text{Promocion})$

Cálculo relacional orientado a tuplas:

$\{ p.\text{titulo_distribucion}, f.\text{dia_semana}, f.\text{hora}, pr.\text{descripcion}, pr.\text{descuento} \mid p \in \text{Pelicula} \wedge f \in \text{Funcion} \wedge pr \in \text{Promocion} \wedge p.\text{id_pelicula} = f.\text{id_pelicula} \wedge f.\text{id_funcion} = pr.\text{id_funcion} \}$

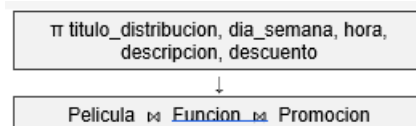


Figura 17. Diagrama de unión funciones con promoción

Tabla 03. Resultado de funciones con promoción

titulo_distribucion	dia_semana	hora	descripcion	descuento
Barbie	Martes	17:00:00	Martes de descuento: 50% antes de las 18 horas	50.00
Barbie	Domingo	18:00:00	Domingo familiar: 30% de descuento	30.00
Dune: Parte Dos	Domingo	16:00:00	Estudiantes con credencial: 20% de descuento	20.00

5. Obtener los actores y el pago recibido por película (financiamiento)

Álgebra relacional:

π titulo_distribucion, nombre, pago (σ rol = 'Actor' (PersonaPeliculaRol) $\bowtie$ Persona $\bowtie$ Pelicula)

Cálculo relacional orientado a tuplas:

$\{ \text{pel.titulo_distribucion, per.nombre, r.pago} \mid r \in \text{PersonaPeliculaRol} \wedge \text{per} \in \text{Persona} \wedge \text{pel} \in \text{Pelicula} \wedge r.\text{rol} = \text{'Actor'} \wedge r.\text{id_persona} = \text{per.id_persona} \wedge r.\text{id_pelicula} = \text{pel.id_pelicula} \}$

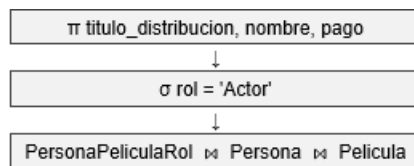


Figura 18. Diagrama de unión actores con su pago por película

Tabla 04. Resultado de actores y pago por película

titulo_distribucion	nombre	pago
Oppenheimer	Cillian Murphy	5,000,000.00
Oppenheimer	Robert Downey Jr.	8,000,000.00
Barbie	Margot Robbie	12,000,000.00
Barbie	Ryan Gosling	10,000,000.00
Dune: Parte Dos	Timothée Chalamet	9,000,000.00
Dune: Parte Dos	Zendaya	7,000,000.00

3. RESULTADOS

Se logró implementar interfaces en VBA Excel integradas con una base de datos MySQL mediante ODBC

3.1 Integración entre VBA Excel y MySQL

La integración entre VBA Excel y MySQL se realizó mediante el controlador ODBC y la biblioteca ADODB, permitiendo que los formularios ejecuten sentencias SQL para registrar, consultar y eliminar información de la base de datos. Con el fin de verificar el funcionamiento de la aplicación, se realizaron pruebas sobre la tabla Cine, ejecutando operaciones de registro y eliminación desde la interfaz desarrollada en VBA Excel y comprobando posteriormente los resultados mediante consultas en MySQL.

Figura 19. Formulario de registro de cine

La interfaz de la figura 19, permite insertar, eliminar y mostrar datos del cine.

```
sql = "INSERT INTO Cine (nombre, direccion, telefono) VALUES (' & _
      '" & txtNombreCine.Text & "'," & _
      '" & txtDireccion.Text & "'," & _
      '" & txtTelefono.Text & "')"

```

```
conn.Execute sql
```

Figura 20. Implementación de la instrucción INSERT INTO en VBA Excel para almacenar la información del cine (nombre, dirección y teléfono) en la base de datos.

```

----- CINES -----
INSERT INTO Cine (nombre, direccion, telefono) VALUES
SET @cine1 = LAST_INSERT_ID();

```

Figura 21 Sentencia SQL utilizada para insertar un nuevo registro en la tabla Cine y obtener el identificador generado automáticamente mediante la función LAST_INSERT_ID()

Figura 22: Registro de un nuevo cine desde la interfaz desarrollada en VBA Excel. Tras ingresar el nombre, dirección y teléfono, el sistema confirma el almacenamiento exitoso de la información mediante un mensaje de registro correcto.

4. DISCUSIÓN O ANÁLISIS DE RESULTADOS

Los resultados obtenidos evidencian que la implementación de una base de datos relacional permitió organizar la información de la cartelera cinematográfica de forma estructurada, reduciendo la redundancia de los datos y garantizando su integridad mediante el uso de claves primarias y claves foráneas. La aplicación de las formas normales hasta la Forma Normal de Boyce-Codd (FNBC) favoreció la eliminación de dependencias funcionales innecesarias y disminuyó la posibilidad de inconsistencias durante las operaciones de inserción, actualización y eliminación de registros. Estos resultados coinciden con los principios establecidos por Elmasri y Navathe [13], quienes sostienen que un adecuado proceso de normalización constituye uno de los pilares fundamentales para el diseño eficiente de bases de datos relacionales.

Asimismo, el empleo de MySQL como sistema gestor de bases de datos proporcionó un entorno estable para el almacenamiento y administración de la información. Las consultas realizadas permitieron recuperar datos relacionados con películas, participantes, funciones, promociones y opiniones de manera eficiente, facilitando la generación de reportes requeridos por el sistema. Estas características concuerdan con la documentación oficial de MySQL, la cual destaca el rendimiento, la confiabilidad y la escalabilidad del gestor para aplicaciones orientadas al manejo de información estructurada [14].

La utilización de Excel VBA como interfaz de usuario permitió desarrollar una aplicación de escritorio con formularios intuitivos que simplifican la interacción entre el usuario y la base de datos. La integración mediante ODBC permitió ejecutar consultas SQL directamente desde la interfaz, evitando que el usuario interactúe con el gestor de bases de datos de manera directa. Este enfoque coincide con las recomendaciones de Microsoft para el desarrollo de aplicaciones orientadas a la automatización y administración de datos mediante Visual Basic for Applications [15].

El diseño del sistema basado en el modelo Entidad-Relación facilitó la representación conceptual de las entidades involucradas en la administración de una cartelera cinematográfica, permitiendo posteriormente su transformación al modelo relacional e implementación física en MySQL. Según Silberschatz, Korth y Sudarshan [16], la correcta transición entre estos modelos contribuye significativamente a la consistencia lógica de la base de datos y simplifica su mantenimiento durante el ciclo de vida del sistema.

Otro aspecto relevante corresponde a la implementación de consultas SQL y operaciones fundamentadas en el álgebra relacional. Estas consultas permitieron obtener indicadores relacionados con el financiamiento de las películas, la asistencia de espectadores, la distribución por edades, las promociones más utilizadas y el promedio de calificaciones otorgadas por el público. De acuerdo con Codd [17], el álgebra relacional constituye la base teórica de los sistemas de bases de datos relacionales y permite expresar consultas de forma precisa e independiente de la implementación física [18].

Desde el punto de vista funcional, el sistema desarrollado permitió integrar en una única aplicación procesos que generalmente se realizan de

manera independiente, como el registro de películas, la programación de funciones, la administración de salas, la gestión de promociones y el almacenamiento de opiniones de los espectadores. Esta integración favorece la centralización de la información y reduce el tiempo requerido para realizar operaciones administrativas, tal como señalan Pressman [19], al describir las ventajas de la ingeniería de software orientada al desarrollo de sistemas de información integrados

La generación automática de reportes estadísticos representa otro de los resultados relevantes del sistema. Los indicadores obtenidos permiten identificar las películas con mayor aceptación por parte del público, analizar el comportamiento de la asistencia según la edad de los espectadores y evaluar la efectividad de las promociones implementadas. En este sentido, Laudon y Laudon indican que los sistemas de información proporcionan información útil para apoyar la toma de decisiones en los distintos niveles de una organización [20]

Por otra parte, la integración de VBA Excel, ODBC y MySQL demuestra que es posible desarrollar soluciones de bajo costo para organizaciones que requieren administrar información sin recurrir a plataformas empresariales de mayor complejidad. Este tipo de soluciones resulta especialmente adecuado para pequeñas y medianas organizaciones debido a su facilidad de implementación, mantenimiento y escalabilidad, aspectos ampliamente documentados en la literatura relacionada con sistemas de gestión de información

Los resultados obtenidos también evidencian que la estructuración de la aplicación facilita futuras ampliaciones del sistema, como la incorporación de nuevos reportes, mecanismos de autenticación más robustos o funcionalidades web que permitan el acceso remoto a la información.

Finalmente, el sistema desarrollado cumplió con los objetivos planteados al proporcionar una herramienta capaz de administrar de manera organizada la información relacionada con una cartelera cinematográfica, optimizando el almacenamiento de datos y facilitando su consulta mediante una interfaz intuitiva. La integración de una base de datos relacional con herramientas de desarrollo ampliamente disponibles constituye una alternativa viable para la implementación de sistemas de información orientados a la gestión administrativa, resultados que coinciden con investigaciones recientes relacionadas con

aplicaciones de escritorio conectadas a bases de datos relacionales.

5. CONCLUSION

El desarrollo del sistema de gestión para varias carteleras cinematográficas permitió integrar una base de datos MySQL con una interfaz desarrollada en VBA Excel, proporcionando una solución eficiente para la administración de películas, funciones, salas, promociones, participantes, entradas y opiniones de los espectadores.

El diseño de la base de datos, sustentado en el modelo Entidad-Relación, su transformación al modelo físico y la aplicación de la Primera, Segunda y Tercera Forma Normal, así como de la Forma Normal de Boyce-Codd (FNBC), contribuyó a garantizar la integridad, consistencia y organización de la información, reduciendo la redundancia de los datos y favoreciendo un almacenamiento estructurado.

Finalmente, la implementación de consultas SQL y la generación de reportes estadísticos permitieron obtener información relevante para el análisis de la gestión cinematográfica, apoyando la toma de decisiones y demostrando que la solución desarrollada cumple con los objetivos planteados, al ofrecer una herramienta funcional, organizada y de fácil utilización para la administración de varias carteleras cinematográficas.

REFERENCIAS

- [1] E. G. y. Valle, Base de Datos, Lima: Todos Los Derechos Son Reservados, 2018.
- [2] I. Corporation, «¿Qué es un sistema de gestión de bases de datos?,» pp. 10-13, Sin Fecha Sin Fecha 2010.
- [3] T. L. D. S. Reservados, «¿Qué es una base de datos relacional? (RDBMS)?,» 18 Junio 2021.
- [4] T. Portal, «MySQL,» 2024.

- [5] M. López, «MySQL: qué es, para qué sirve y cuáles son sus características,» *IMMUNE Technology Institute*, 13 Enero 2024.
- [6] Y. R. Sinha, «Aprenda VBA en Excel: Automatice y mejore sus hojas de cálculo.,» *IntelliPat*, 3 Noviembre 2025.
- [7] T. L. D. S. Reservados, «Modelo de Entidad - Relación de una BBDD: qué es, elementos y ejemplos,» *ILERNA*, 24 Septiembre 2024.
- [8] T. L. D. S. Reservados, «Modelo relacional en sistemas de gestión de bases de datos,» *Geeksforgeeks*, 24 Abril 2026.
- [9] T. L. D. S. Reservados, «Diferencia entre el modelo ER y el modelo relacional en los sistemas de gestión de bases de datos (DBMS),» *Geeksforgeeks*, 15 Julio 2025.
- [10] T. L. D. S. Reservados, «Qué es ODBC - Conectividad abierta de bases de datos,» *insightsoftware*, 10 Junio 2023.
- [11] D. Radečić, «Álgebra relacional: La lógica detrás de cada consulta SQL,» *Datacamp*, 25 Febrero 2026.
- [12] L. V. Cabrera, «Álgebra Relacional». *Universidad de Sevilla*.
- [13] R. /. N. S. ELMASRI, de *FUNDAMENTOS DE SISTEMAS DE BASES DE DATOS*, PEARSON EDUCATION LIMITED, 2007, p. 1012.
- [14] M. SQL, «Manual de Referencia de MySQL».
- [15] Microsof, «Referencia del lenguaje de Visual Basic para Aplicaciones (VBA),» 2026.
- [16] H. F. K. S. S. Abraham Silberschatz, «DATABASE,» p. 1337.
- [17] T. L. D. S. Reservados, «El modelo relacional de Edgar F. Codd: por qué SQL ganó en los negocios,» *Koder*, 30 Agosto 2025.
- [18] T. L. D. S. Reservados, «Álgebra Relacional: Definición y Operaciones Esenciales,» *Studocu*, Sin Fecha Sin Fecha 2026.
- [19] P. Roger S. Pressman, «Ingeniería del software, Un Enfoque Práctico,» p. 805.
- [20] J. P. Kennet C. Laudon, de *SISTEMAS DE INFORMACIÓN*, Mexico, 2012, p. 643.
- [21] R. S. Pressman, «Ingeniería de Software, Un Enfoque Práctico».