



®



**TECNOLÓGICO
NACIONAL DE MÉXICO**

®

INSTITUTO TECNOLÓGICO DE TLAXIACO

ASIGNATURA:

SISTEMAS OPERATIVOS.

CARRERA:

INGENIERÍA EN SISTEMAS COMPUTACIONALES

ACTIVIDAD:

. DIAGRAMA DE CLASES EN ARCHITEC

DOCENTE:

ROMAN CRUZ JOSE ALFREDO.

ALUMNO:

CHARLY JOSHUA SANDOVAL HERNANDEZ.

GRUPO:

5US

FECHA:

07/099/2023

INTRODUCCION:



Architect es una potente herramienta de modelado que permite a los desarrolladores, analistas y diseñadores crear visualmente diagramas UML para modelar sistemas de software. Los diagramas de clases en Enterprise Architect son una parte fundamental de este proceso, ya que representan la estructura estática de un sistema.

A continuación, se presenta una breve guía sobre cómo crear un diagrama de clases en Enterprise Architect:

Crear un Proyecto: Antes de comenzar, debes crear un proyecto en Enterprise Architect para organizar tus modelos y diagramas. Puedes hacerlo seleccionando "File" (Archivo) > "New Project" (Nuevo proyecto) y siguiendo los pasos.





OBJECTIVOS:

Architect es una potente herramienta de modelado que permite a los desarrolladores, analistas y diseñadores crear visualmente diagramas UML para modelar sistemas de software. Los diagramas de clases en Enterprise Architect son una parte fundamental de este proceso, ya que representan la estructura estática de un sistema.

A continuación, se presenta una breve guía sobre cómo crear un diagrama de clases en Enterprise Architect:

Crear un Proyecto: Antes de comenzar, debes crear un proyecto en Enterprise Architect para organizar tus modelos y diagramas. Puedes hacerlo seleccionando "File" (Archivo) > "New Project" (Nuevo proyecto) y siguiendo los pasos.

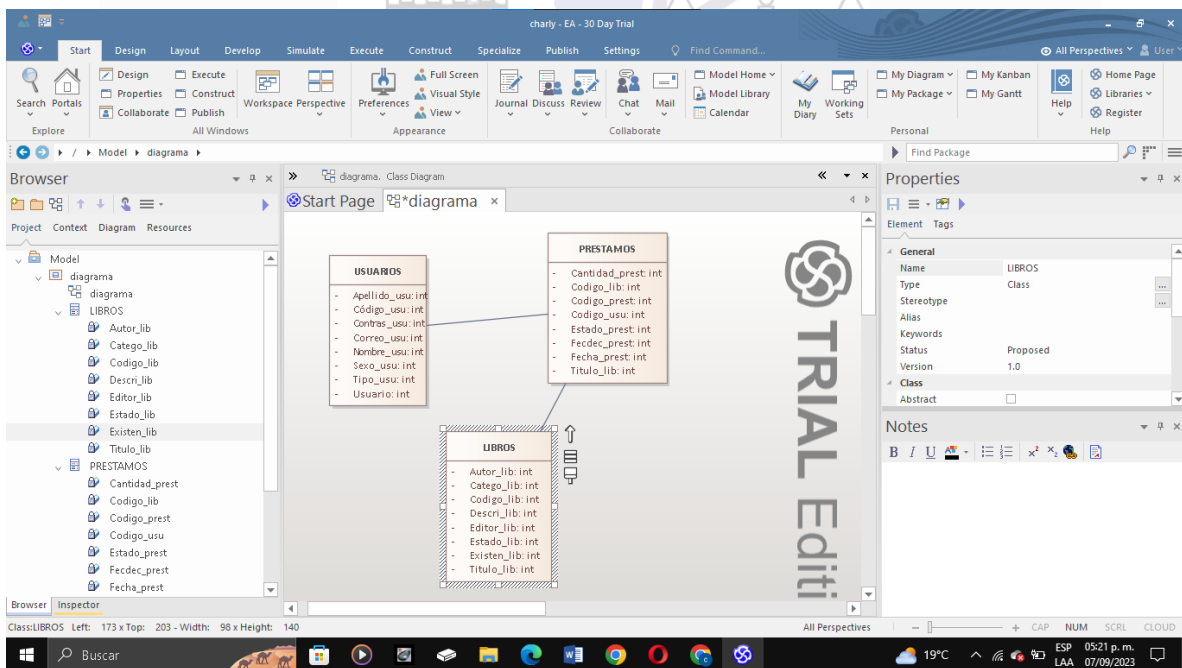
DESARROLLO:

Agregar Clases al Diagrama: Abre tu herramienta de modelado UML (como Enterprise Architect) y crea un nuevo diagrama de clases. Luego, arrastra y suelta las clases que identificaste en el paso anterior desde la paleta de herramientas o el catálogo de elementos a tu diagrama de clases.

Definir Atributos: Para cada clase, define sus atributos (propiedades). Por ejemplo, para la clase "Libro", podrías tener atributos como "Título", "Autor", "ISBN", "Precio", etc. Agrega estos atributos a las clases correspondientes en el diagrama.

Especificar Métodos: Define los métodos (comportamiento) que cada clase tiene. Por ejemplo, la clase "Libro" podría tener métodos como "prestar()", "devolver()", "consultarDisponibilidad()", etc. Agrega estos métodos a las clases correspondientes en el diagrama.

Relaciones entre Clases: Establece las relaciones entre las clases. Por ejemplo, podrías tener una relación entre la clase "Libro" y la clase "Cliente" para representar que un cliente puede tomar prestado un libro. Puedes utilizar notaciones como asociaciones, agregaciones o composiciones según corresponda.



USUARIOS

- Apellido_usu: int
- Código_usu: int
- Contras_usu: int
- Correo_usu: int
- Nombre_usu: int
- Sexo_usu: int
- Tipo_usu: int
- Usuario: int

La clase "Usuarios" se relaciona con la clase "Préstamo" a través del método "Realizar Préstamo", lo que indica que un usuario puede realizar préstamos de libros.

En el primer diagrama agregamos los primeros datos para el usuario, los usuarios desempeñan un papel esencial en una librería, ya que son quienes dan vida al espacio y utilizan los recursos que ofrece.

PRESTAMOS

- Cantidad_prest: int
- Codigo_lib: int
- Codigo_prest: int
- Codigo_usu: int
- Estado_prest: int
- Fedec_prest: int
- Fecha_prest: int
- Titulo_lib: int

En el siguiente diagrama La clase "prestamos" se relaciona con "libros" para indicar que un libro puede estar involucrado en uno o varios préstamos.

LIBROS

- Autor_lib: int
- Catego_lib: int
- Codigo_lib: int
- Descri_lib: int
- Editor_lib: int
- Estado_lib: int
- Existen_lib: int
- Titulo_lib: int

La clase "Libro" contiene atributos que describen los detalles de un libro, como su ISBN, título, autor, precio y cantidad en stock. Las clases "Categoría", "Autor" y "Editorial" se utilizan para categorizar y gestionar los libros y sus relaciones con otras entidades.

CONCLUSION:



La conclusión de un diagrama de clases de una librería debería resumir los aspectos clave de la representación visual de las clases y sus relaciones en el sistema de la librería. Aquí tienes un ejemplo de una conclusión para un diagrama de clases de una librería:

En resumen, el diagrama de clases de nuestra librería proporciona una visión completa y organizada de las clases y sus relaciones en el sistema. Hemos identificado las principales clases, como Libro, Cliente, Empleado y Biblioteca, junto con sus atributos y métodos relevantes. Además, hemos representado las asociaciones entre estas clases, como la relación entre Cliente y Libro para representar los préstamos, así como las relaciones de herencia para modelar diferentes tipos de empleados.

