

UNIVERSIDAD PRIVADA DE TACNA



INGENIERIA DE SISTEMAS

TITULO:

TRABAJO FINAL UNIDAD I - BIBLIOTECA

CURSO:

BASE DE DATOS II

DOCENTE(ING):

Patrick Cuadros Quiroga

Integrantes:

Condori Gutierrez, Flor de Maria

(2015053227)

TRABAJO FINAL

SISTEMA BIBLIOTECA

1. PROBLEMA

Nos pide sistematizar una biblioteca, para una determinada reservación, o prestamos de algún libro y atención a los usuarios de la universidad de manera satisfactoria y darle una solución a través del visual studio.

1.1. Título Descriptivo del Proyecto

Para este proyecto, tiene como nombre BIBLIOTECA que esto forma parte de la sistematización de la biblioteca de la upt. podría ser en alguna otra biblioteca que lo necesiten este tipo de modelo, que será desarrollado de manera grupal y sería más que todo en nuestras casas o en la universidad misma donde lo desarrollaremos.

1.2. Formulación del Problema

Para la solución del problema de este dicho proyecto sería más que todo para que los usuarios se sientan conforme con la atención que le damos con este software y así no tenga ninguna queja al momento de ingresar a la biblioteca virtual.

2. MARCO TEÓRICO

2.1. MVC

En este diseño de biblioteca de software utilizada para implementar sistemas donde se requiere el uso de interfaces de usuario. Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.

Su fundamento es la separación del código en tres capas diferentes, acotadas por su responsabilidad, en lo que se llaman Modelos, Vistas y Controladores, o lo que es lo mismo, Model, Views & Controllers, si lo prefieres en inglés.

MVC es un "invento" que ya tiene varias décadas y fue presentado incluso antes de la aparición de la Web. No obstante, en los últimos años ha ganado mucha fuerza y seguidores gracias a la aparición de numerosos frameworks de desarrollo web que utilizan el patrón MVC como modelo para la arquitectura de las aplicaciones web.

2.2. Entity Framework

Es un conjunto de tecnologías de ADO.NET que permiten el desarrollo de aplicaciones de software orientadas a datos. Los arquitectos y programadores de aplicaciones orientadas a datos se han enfrentado a la necesidad de lograr dos objetivos muy diferentes. Deben modelar las entidades, las relaciones y la lógica de los problemas empresariales que resuelven, y también deben trabajar con los motores de datos que se usan para almacenar y recuperar los datos. Los datos pueden abarcar varios sistemas de almacenamiento, cada uno con sus propios protocolos; incluso las aplicaciones que funcionan con un único sistema de almacenamiento deben equilibrar los requisitos del sistema de almacenamiento con respecto a los requisitos de escribir un código de aplicación eficaz y fácil de mantener.

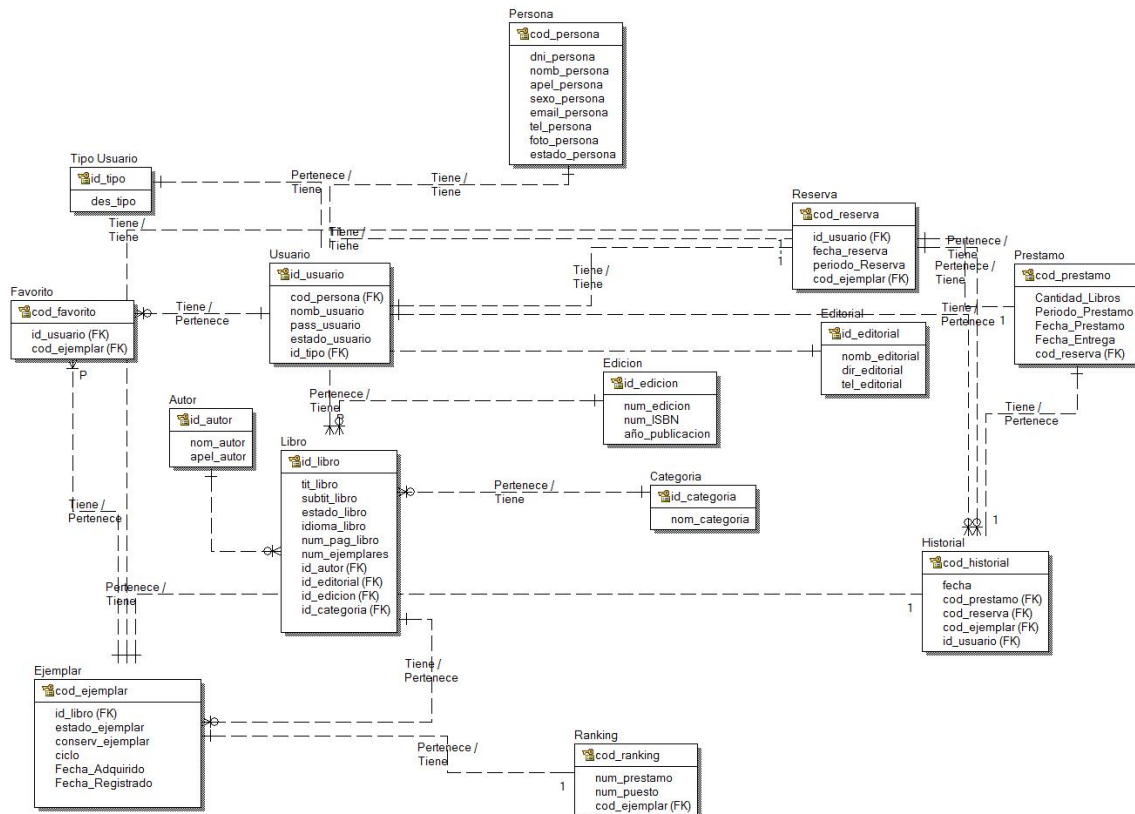
Entity Framework permite a los desarrolladores trabajar con datos en forma de objetos y propiedades específicos del dominio, como clientes y direcciones de cliente, sin tener que preocuparse por las tablas y columnas de la base de datos subyacente donde se almacenan estos datos. Con Entity Framework, los desarrolladores pueden trabajar en un nivel mayor de abstracción cuando tratan con datos, y pueden crear y mantener aplicaciones orientadas a datos con menos código que en las aplicaciones tradicionales. Dado que Entity Framework es un componente de .NET Framework, las aplicaciones de Entity Framework se pueden ejecutar en cualquier equipo en el que esté instalado .NET Framework a partir de la versión 3.5 SP1.

3. DESARROLLO

3.1. Análisis

- Modelo entidad relación

Este diagrama es muestra la cardinalidad, relaciones y atributos de las Clases existentes en el proyecto. Este diagrama fue hecho en Erwin Data Modeler.



3.2. ENTIDADES

Clase Autor:

```
public class ClsAutor
{
    2 referencias
    public int id_autor { get; set; }
    4 referencias
    public string nom_autor { get; set; }
    3 referencias
    public string apel_autor { get; set; }

    1 referencia
    public static ClsAutor RegistrarAutor(string nomb, string apel)
    {
        return new ClsAutor()
        {
            nom_autor = nomb,
            apel_autor = apel
        };
    }

    0 referencias
    public void ModificarAutor(string nomb, string apel)
    {
        nom_autor = nomb;
        apel_autor = apel;
    }
}
```

Clase Categoria:

```
public class ClsCategoria
{
    2 referencias
    public int id_categoria { get; set; }
    3 referencias
    public string nomb_categoria { get; set; }

    0 referencias
    public static ClsCategoria RegistrarCategoria(string asnomb_categoria)
    {
        return new ClsCategoria()
        {
            nomb_categoria = asnomb_categoria
        };
    }

    0 referencias
    public void ModificarCategoria(string asNuevonomb_categoria)
    {
        nomb_categoria = asNuevonomb_categoria;
    }
}
```

Clase Edicion:

```
public class ClsEdicion
{
    2 referencias
    public int id_edicion { get; set; }
    3 referencias
    public string num_edicion { get; set; }
    3 referencias
    public string num_ISBN { get; set; }
    3 referencias
    public string anio_publicacion { get; set; }

    0 referencias
    public static ClsEdicion RegistrarEdicion(string asnum_edicion, string asnum_ISBN,
        string asanio_publicacion)
    {
        return new ClsEdicion()
        {
            num_edicion = asnum_edicion,
            num_ISBN = asnum_ISBN,
            anio_publicacion = asanio_publicacion
        };
    }

    0 referencias
    public void ModificarEdicion(string asnum_edicion, string asnum_ISBN,
        string asanio_publicacion)
    {
        num_edicion = asnum_edicion;
        num_ISBN = asnum_ISBN;
        anio_publicacion = asanio_publicacion;
    }
}
```

Clase Editorial:

```
public class ClsEditorial
{
    2 referencias
    public int id_editorial { get; set; }
    3 referencias
    public string nomb_editorial { get; set; }
    3 referencias
    public string dir_editorial { get; set; }
    3 referencias
    public string tel_editorial { get; set; }

    0 referencias
    public static ClsEditorial RegistrarEditorial(string as nomb_editorial, string as dir_editorial,
        string as tel_editorial)
    {
        return new ClsEditorial()
        {
            nomb_editorial = as nomb_editorial,
            dir_editorial = as dir_editorial,
            tel_editorial = as tel_editorial
        };
    }

    0 referencias
    public void ModificarEditorial(string as nomb_editorial, string as dir_editorial,
        string as tel_editorial)
    {
        nomb_editorial = as nomb_editorial;
        dir_editorial = as dir_editorial;
        tel_editorial = as tel_editorial;
    }
}
```

Clase Ejemplar:

```
public class ClsEjemplar
{
    2 referencias
    public int cod_ejemplar { get; set; }
    4 referencias
    public int id_libro { get; set; }
    3 referencias
    public string estado_ejemplar { get; set; }
    3 referencias
    public string conserv_ejemplar { get; set; }
    3 referencias
    public string ciclo { get; set; }
    3 referencias
    public DateTime Fecha_Adquirido { get; set; }
    3 referencias
    public DateTime Fecha_Registrado { get; set; }
    1 referencia
    public ClsLibro libro { get; set; }

    0 referencias
    public static ClsEjemplar RegistrarEjemplar(int as id_libro, string as estado,
        string as conservacion, string as ciclo,
        DateTime as Fecha_Adquirido, DateTime as Fecha_Registrado)
    {
        return new ClsEjemplar()
        {
            id_libro = as id_libro,
            estado_ejemplar = as estado,
            conserv_ejemplar = as conservacion,
            ciclo = as ciclo,
            Fecha_Adquirido = as Fecha_Adquirido,
            Fecha_Registrado = as Fecha_Registrado
        };
    }

    0 referencias
    public void ModificarEjemplar(int as id_libro, string as estado,
        string as conservacion, string as ciclo,
        DateTime as Fecha_Adquirido, DateTime as Fecha_Registrado)
    {
        id_libro = as id_libro;
        estado_ejemplar = as estado;
        conserv_ejemplar = as conservacion;
        ciclo = as ciclo;
        Fecha_Adquirido = as Fecha_Adquirido;
        Fecha_Registrado = as Fecha_Registrado;
    }
}
```

Clase Favorito:

```
public class ClsFavorito
{
    2 referencias
    public int cod_favoritos { get; set; }
    4 referencias
    public int id_usuario { get; set; }
    4 referencias
    public int cod_ejemplar { get; set; }
    1 referencia
    public ClsUsuario usuario { get; set; }
    1 referencia
    public ClsEjemplar ejemplar { get; set; }

    0 referencias
    public static ClsFavorito AgregarFavorito(int asid_usu, int ascod_ejemplar)
    {
        return new ClsFavorito()
        {
            id_usuario = asid_usu,
            cod_ejemplar = ascod_ejemplar
        };
    }

    0 referencias
    public void EliminarFavorito()
    {
        id_usuario = 0;
        cod_ejemplar = 0;
    }
}
```

Clase Historial:

```
public class ClsHistorial
{
    2 referencias
    public int cod_historial { get; set; }
    3 referencias
    public int cod_prestamo { get; set; }
    3 referencias
    public int cod_reserva { get; set; }
    3 referencias
    public int id_usuario { get; set; }
    3 referencias
    public int cod_ejemplar { get; set; }
    2 referencias
    public DateTime fecha { get; set; }
    1 referencia
    public ClsPrestamo prestamo { get; set; }
    1 referencia
    public ClsReserva reserva { get; set; }
    1 referencia
    public ClsUsuario usuario { get; set; }
    1 referencia
    public ClsEjemplar ejemplar { get; set; }

    0 referencias
    public static ClsHistorial AgregarHistorial(int ascod_prestamo, int ascod_reserva,
        int asid_usu, int ascod_ejemplar, DateTime asfecha)
    {
        return new ClsHistorial()
        {
            cod_prestamo = ascod_prestamo,
            cod_reserva = ascod_reserva,
            id_usuario = asid_usu,
            cod_ejemplar = ascod_ejemplar,
            fecha = asfecha
        };
    }
}
```