



Cátedra Tecnología de Sistemas

[Programación Avanzada con C#]

Código: [00830]

Proyecto 2. Valor 2%

Temas de Estudio

1. Tema 5 Subprocesamiento múltiple
2. Tema 6 Bases de datos y componentes ADO.NET
3. Tema 7 Colecciones
4. Tema 8 Redes

Objetivo

Poner en práctica los conceptos subprocesamiento múltiple y redes, así como retomar los conceptos ya aprendidos de base de datos y colecciones. Además, poner en práctica los conceptos de interfaz gráfica, así como el manejo de excepciones.

Software de Desarrollo

Visual Studio Community 2022 (C#, .Net Framework 4.8 o Net 6.0)

Desarrollo

A los encargados **DESARROLLO-UNED** están contentos con la primera etapa de la automatización de sus procesos, sin embargo, han decidido hacer algunos cambios como almacenar de manera permanente la información en una base de datos SQL Server, además quieren poner en funcionamiento lo antes posible reserva de artículos. Ahora bien, para lograr la incorporación de esta nueva funcionalidad debe implementarse una solución cliente/servidor que permita a los clientes conectarse mediante una aplicación cliente al servidor central para solicitar las reservas.

El sistema por desarrollar debe estar compuesto mínimo por dos proyectos; un cliente y un servidor, ambos deben comunicarse a través de la red mediante el protocolo TCP. El proyecto cliente será utilizado por los clientes para registrar la reserva de artículos disponibles en la tienda, mientras que el proyecto servidor será utilizado por un único administrador en las oficinas centrales para dar mantenimiento a las funcionalidades de Categoría, Artículos, Vendedor, Sucursal, Artículo por Sucursal y Cliente.

Toda la información debe ser almacenada en una base de datos SQL Server.

El proyecto cliente debe cumplir con los siguiente:

Conexión y Validación.

- Debe ser utilizado por los clientes para conectarse con el servidor central y realizar reservas de los productos que hay en el inventario.
- Debe tener una opción para conectarse y desconectarse del servidor.
- Al iniciar el cliente se debe solicitar el número de identificación del cliente
 - Se debe validar que el cliente exista en base de datos antes de mostrar las funcionalidades de la aplicación cliente.
 - La aplicación cliente envía al servidor la solicitud de verificación de la identificación del usuario cliente que ingresó.
 - El cliente debe haber sido registrado previamente por el administrador en el servidor.
 - Si la identificación existe, es decir, es un cliente registrado se permite el ingreso a la aplicación cliente.
- Una vez que el cliente ha sido validado, en la aplicación cliente se debe mostrar el nombre del cliente con los dos apellidos, esta información debe ser siempre visible en la aplicación cliente. Además, una vez que el cliente ha sido validado, en la aplicación cliente se muestran desbloqueadas las funcionalidades de la aplicación.

Reserva de artículo:

Se debe mostrar una opción que permita al cliente registrar la reserva de artículos que se encuentren en el inventario por Sucursal. Para esto el sistema debe crear la reserva de artículos (Reserva). Para esto se debe realizar lo siguiente:

- Se debe enviar solicitud al servidor para consultar las sucursales, esta información se debe mostrar en un control ComboBox, o bien, en un control DataGridView.
 - Solo se deben mostrar las sucursales activas

- Una vez seleccionada la sucursal se debe enviar solicitud al servidor para consultar todos los artículos asociados a la sucursal seleccionada. La información de artículos disponibles se debe mostrar en un control DataGridView.
 - Se debe validar que al menos exista un artículo disponible en la sucursal seleccionada.
- Una vez seleccionada la sucursal y el artículo el usuario debe seleccionar una fecha de la reservación.
- Debe validar que tenga la cantidad de artículos disponibles para realizar la reserva, al menos debe contar con un artículo en el valor de cantidad.
- Una vez validada la reservación, y tomando en consideración que el artículo se debe rebajar del campo cantidad de artículo por Sucursal la disponibilidad de ese artículo asociado.
- En esta funcionalidad no se selecciona el cliente, dado que el cliente seleccionado es el que está ejecutando el programa. La aplicación cliente solo puede registrar reserva del cliente que ejecuta la aplicación.
- Al hacer clic en el botón “Realizar Reserva” debe enviarse al servidor, mediante la red, la información de la reserva. El servidor es el encargado de hacer los cambios en base de datos.
- Se debe validar los datos en todos los pasos y las posibles excepciones.
- La información por registrar es la siguiente:
 - Id de reserva (int) Se debe validar que no existan id's repetido.
 - Artículo por Sucursal (Objeto ArticuloxSucursal),
 - Cliente (Objeto Cliente)
 - Fecha Reserva (DateTime)
- Mostrar en pantalla si el proceso fue exitoso.

Consultar mis reservas: Está opción le permite al cliente consultar desde la aplicación cliente las reservas realizadas.

- Está opción solo muestra las reservas realizadas por el cliente que está ejecutando la aplicación cliente, no debe mostrar reservas realizadas por otros clientes.

- Debe enviar la solicitud de consultar reservas del cliente hacia el servidor a través de la red.
- El servidor es quien consulta la base de datos y devuelve la información al cliente a través de la red.
- El cliente muestra en pantalla las reservas realizadas por el cliente.
- La información debe mostrar el id de la reserva, la información de la Sucursal, la información artículo y la fecha de la reserva.
- Se debe implementar dos formas (funcionalidades) de consultar reservas, una pudiendo consultar todas las reservas realizadas por el cliente; la otra consultando reserva específicamente mediante el id de la reserva.
- **La aplicación cliente no tiene conexión directa a la base de datos, todas sus solicitudes deben pasar a través de la aplicación servidor, este último es el que tiene conexión hacia la base de datos**

El proyecto servidor debe de cumplir con lo siguiente:

- Debe implementar un servidor que escuche y responda las solicitudes de red de múltiples clientes TCP de forma simultánea utilizando subprocesamiento múltiple.
- Debe utilizar sockets en la dirección 127.0.0.1 y el puerto 14100 para escuchar solicitudes.
- La aplicación cliente debe almacenar toda la información en base de datos SQL Server.
- El servidor almacena en la base de datos toda la información indicada desde la aplicación cliente por los usuarios clientes.
- El servidor consulta y almacena en la base de datos la información solicitada por los clientes.
- **Solo la aplicación servidor se comunica con la base de datos, la aplicación cliente no tiene conexión con la base de datos.**
- El servidor debe ser utilizado por un único usuario administrador en las oficinas centrales, el cual no requiere registrarse, tampoco requiere usuario y contraseña.
- Debe tener una pantalla principal donde muestre en bitácora **todos los eventos** enviados por todos los clientes conectados en tiempo real, como, por ejemplo: conexión al servidor, desconexión, registro de reserva, consulta de reserva.

- Siempre debe ser visible la cantidad de clientes conectados.

El servidor debe contar con un menú principal con las siguientes opciones:

- Registrar y consultar Categoría de Artículos.
- Registrar y consultar Artículos.
- Registrar y consultar Vendedor.
- Registrar y consultar Sucursal.
- Registrar y consultar Artículos por Sucursal.
- Registrar y consultar Cliente.

Registrar Categoría Artículos

Esta opción le permitirá a la persona usuaria registrar o agregar los datos de la categoría a la que puede pertenecer un artículo, por ejemplo: ropa, zapatos, balones, etc.

Se debe poder registrar la siguiente información:

- Id de categoría (int), este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos.
- Nombre de la categoría(string)
- Descripción (string)

Una vez capturada la información se debe agregar a un arreglo de objetos de tipo "Categoría". El arreglo debe ser de 10 posiciones.

Una vez capturada la información se debe agregar en la tabla "Categoría".

Registrar Artículo

Esta opción le permitirá a la persona usuaria registrar o agregar los datos de un artículo.

Se debe poder registrar la siguiente información:

- Id (int), este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos.
- Descripción(string).
- Categoría artículo (Objeto Categoría) Debe validar que por lo menos exista registrado una categoría de artículo o manejarlo con un Combobox.
- Marca (string). Debe indicar un texto con la marca.
- Activo (bool).

Una vez capturada la información se debe agregar en la tabla "Artículo".

Registrar Vendedor

Esta opción le permitirá a la persona usuaria registrar o agregar los datos de una persona encargada de administrar una sucursal.

Se debe poder registrar la siguiente información:

- Identificación (int) este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos.
- Nombre(string)
- Primer apellido(string)
- Segundo apellido(string)
- Fecha de nacimiento (datetime)
- Fecha de ingreso (datetime)

Una vez capturada la información se debe agregar en la tabla "Vendedor".

Registrar Sucursal

Esta opción le permitirá a la persona usuaria registrar o agregar los datos de una sucursal.

Se debe poder registrar la siguiente información:

- Id de sucursal (int), este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos.
- Nombre(string)
- Vendedor (Objeto Vendedor)
- Dirección (string)
- Teléfono (string)
- Activo (bool), sí o no, (mostrar en "Combobox")

Una vez capturada la información se debe agregar en la tabla "Sucursal".

Registrar Cliente

Esta opción le permitirá a la persona usuaria registrar o agregar los datos de una persona cliente.

Se debe poder registrar la siguiente información:

- Identificación (int) este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos.
- Nombre(string).
- Apellido1(string).
- Apellido2(string).
- Fecha de nacimiento (datetime).

- Activo (bool), Sí o No, (mostrar en “Combobox”)

Una vez capturada la información se debe agregar en la tabla “Cliente”.

Registrar Artículo por Sucursal

Funcionalidad que debe permitir al usuario registrar (asociar) los artículos que estarán disponibles en cada Sucursal. Para ello se debe crear una clase “ArticulosxSucursal”.

Para esta opción se debe poder registrar la siguiente información para Artículos por Sucursal:

- Código (int) este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos.
- Sucursal (Objeto Sucursal), este valor en combinación con Artículo debe ser únicos por lo que se debe validar en el sistema que no existan valores repetidos.
- Artículo (Objeto Artículo), este valor en combinación con Sucursal debe ser únicos por lo que se debe validar en el sistema que no existan valores repetidos.
- Cantidad (int).

Una vez capturada la información se debe agregar en la tabla “ArticulosxSucursal”.

Para esto se debe realizar lo siguiente:

- Mostrar en pantalla en un “ComboBox” las sucursales disponibles.
 - Se debe validar que al menos exista una sucursal registrada para realizarle asociación.
 - Para cada sucursal registrada debe mostrar los siguientes datos: Nombre
 - Solo debe mostrar las sucursales que se encuentran activas.
- Permitir a la persona usuaria seleccionar desde “ComboBox” la sucursal.
- Mostrar en un control “DataGridView” los artículos disponibles para asignar a la sucursal.
 - Se debe validar que al menos exista un artículo registrado para ser asociado.
 - Para cada artículo de la lista se debe mostrar toda la información de los objetos: Id, descripción, categoría, marca.
 - Solamente se puede desplegar los artículos activos.
- La persona usuaria puede seleccionar uno o más artículos del control “DataGridView” para asociar a la sucursal.
- La persona usuaria debe contar con un control de entrada de datos “TextBox”, donde podrá definir la cantidad de inventario que contará cada artículo.

- Se debe validar que la cantidad ingresada sea mayor a cero.
- Si se selecciona varios artículos a asociar a todas se les asignará la misma cantidad de inventario.

Consultar Categoría Artículo

Esta opción le permitirá a la persona usuaria visualizar en pantalla todas las categorías de artículos registradas, para tal fin debe hacer uso del control “DataGridView”, se debe mostrar todos los atributos.

Consultar Artículos

Esta opción le permitirá a la persona usuaria visualizar en pantalla todos los artículos registrados, para tal fin debe hacer uso del control “DataGridView”, se debe mostrar todos los atributos incluso los datos de la categoría a la que pertenece.

Consultar Vendedor

Esta opción le permitirá a la persona usuaria visualizar en pantalla todos los vendedores registrados, para tal fin debe hacer uso del control “DataGridView”, se debe mostrar todos los atributos.

Consultar Sucursal

Esta opción le permitirá a la persona usuaria visualizar en pantalla todas las sucursales registradas, para tal fin debe hacer uso del control “DataGridView”, se debe mostrar todos los atributos.

Consultar Cliente

Esta opción le permitirá a la persona usuaria mostrar en pantalla todos los clientes registrados, para tal fin debe hacer uso del control “DataGridView”, se debe mostrar todos los atributos.

Consultar Artículo por Sucursal

Esta opción le permitirá a la persona usuaria visualizar en pantalla todas las asociaciones registradas, para tal fin debe hacer uso del control “DataGridView”, se debe mostrar todos los atributos de la sucursal y los artículos.

Consideraciones técnicas

Uso de comentarios:

Se deberá implementar el uso de comentarios cuando se realiza la codificación del programa, en la parte superior indicar la universidad, el cuatrimestre, el nombre y la descripción del proyecto, el nombre del estudiante y la fecha en que está realizando la programación. Lo anterior debe estar en cada archivo cs (clase, program, etc) que se realiza el código de programación; como se puede observar en el siguiente ejemplo:

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  /* UNED III Cuatrimestre
8   * Proyecto XX: Descripción del Proyecto
9   * Estudiante:
10  * Fecha:
11
12  */
13
14  namespace CapaEntidades
15  {
16  }
```

Debe utilizar POO (Programación orientada a objetos) para resolver el problema.

- Las clases de objetos que usted considere como entidades (Categoría, Artículo, Vendedor, Sucursal, Cliente, Artículo por Sucursal, etc.) no deben contener lógica para solicitar información a la persona usuaria, solo debe tener la estructura de la clase y sus propiedades.
- Implementar en la Solución proyectos independientes con el fin de tener las distintas capas que usted considere: Capa de Entidades, Capa de Acceso a Datos, Capa de Lógica Negocio, Capa de Presentación (Interfaz del Usuario) etc.
- Cada clase debe ser creada en archivo por separado
- Las clases no deben tener métodos vacíos y recuerde utilizar nombres significativos a sus variables.

Uso de colecciones

- Se deben usar colecciones genéricas (List<>) cuando se requiera consultar de base de datos y retornar más de un registro.

- Las colecciones genéricas se usan como “DataSource” en los controles “combobox” y “DataGridView”.
- **No use “DataSet” ni “DataTable”.**

Interfaz de usuario

- Debe usar interfaz de usuario con formularios GUI (Proyecto Windows Form).

Base de datos

- Debe almacenar y consultar la información en una base de datos SQL Server. No se permite otro motor de base de datos.
- Para la cadena de conexión use seguridad integrada de Windows y un archivo de configuración “App.config”.
- **El script de creación de la base de datos se publicará en la plataforma. No debe modificar el script de base de datos.**

Debe implementar el manejo de excepciones

- Si ocurre una excepción, el sistema no debe cerrarse, se debe mostrar un mensaje al usuario y manejar la excepción de forma apropiada.
- Interfaz de usuario
- Debe usar interfaz de usuario con formularios GUI (Proyecto Windows Form).

Otras consideraciones

- La solución será probada utilizando al menos 5 aplicaciones clientes simultáneamente y una única instancia de la aplicación servidor.
- Se recomienda crear un tercer proyecto de entidades tipo biblioteca de clases para las entidades a utilizar en la solución, de esa manera no tendrá que duplicar las clases de entidades en el proyecto cliente y servidor, en estos solo se agrega la referencia o dll.
- Revise constantemente los foros de consulta y el foro de interés, ahí se aclaran dudas del proyecto o se realizan anuncios que le pueden ser de mucha utilidad en el desarrollo del proyecto.
- Los proyectos en donde se identifique una copia textual o similar, tanto del material digital, como de código fuente, serán anulados y se aplicará el reglamento para dichos casos. Si usa código fuente que no es de su autoría debe indicarlo, aunque sea código del tutor.
- Debe presentar un pequeño manual con los pasos para probar la solución entregada.

Diagrama cliente/servidor

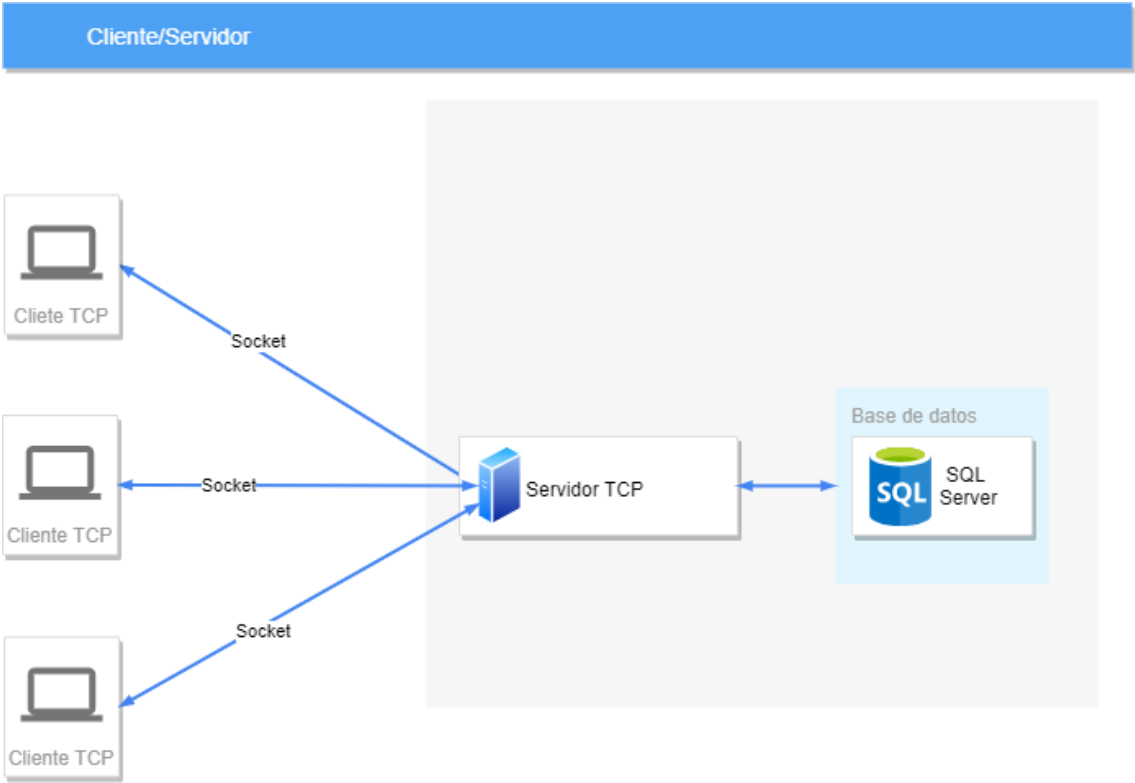
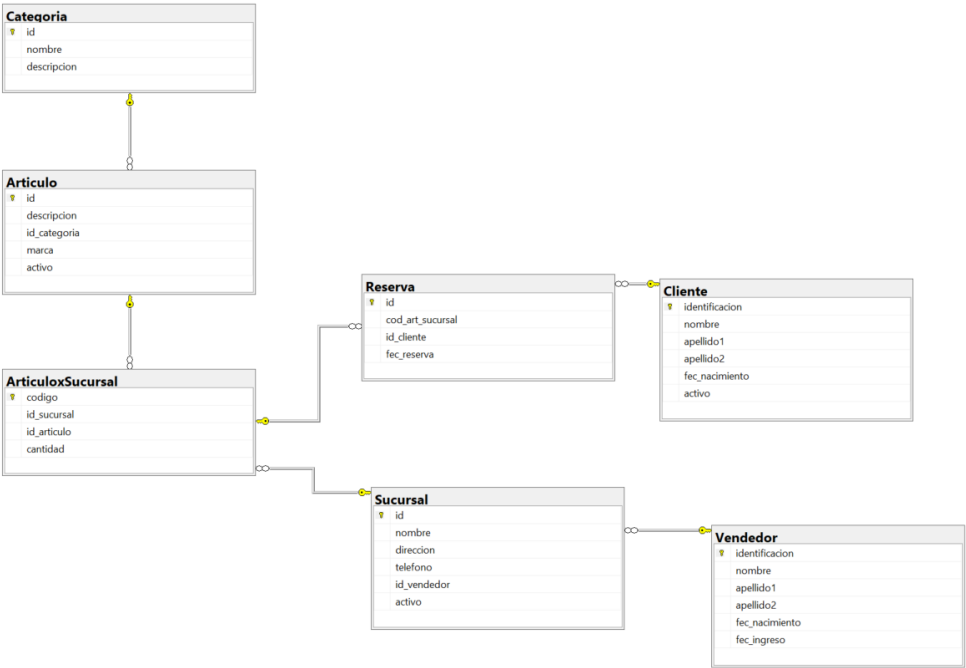


Diagrama base de datos



Honestidad Académica



<https://audiovisuales.uned.ac.cr/play/player/23048>

Nota Importante

Cada estudiante es responsable del contenido que entrega, si no es el archivo correcto, no podrá entregarlo posterior a la fecha establecida.

Si el contenido del archivo coincide con algún otro estudiante, o se comprueba que no es de su autoría, se aplicaría lo indicado en la plataforma en el documento [Lineamientos ante casos de plagio](#)

Indicaciones Importantes

- Es obligatorio que incluya todo el directorio donde se encuentra el proyecto 1.
- El **proyecto 1** debe estar desarrollado en **Visual Studio Community 2022** que es la herramienta oficial del curso.
- El programa debe ser modular, utilizando de la mejor manera funciones definidas por usted.
- **Los trabajos deben realizarse en forma individual. Dentro del código del programa debe de indicar la documentación que explique cómo fue realizado el programa.**
- **Si utiliza código de algún ejemplo del libro, del algún profesor o de otra fuente que no sea de su autoría, debe de indicarlo.**
- Comprima todos los archivos en un solo archivo .zip o .rar.
- **Nombre del archivo que envía:** debe ser nombre y primer apellido del estudiante, y nombre de la tarea. **Ejemplo: JuanRojas-proyecto1.**
- La entrega del proyecto las fechas establecidas en la plataforma de aprendizaje en línea Moodle en el apartado que se indique.

Rúbrica de Evaluación

Criterio	Cumple a satisfacción lo indicado en la evaluación	Cumple medianamente en lo indicado en la evaluación	Cumple en contenido y formato, pero los aportes no son significantes	No cumple o no presenta lo solicitado
Presentación del código: Correcto uso de variables, nombres de métodos significativos, no presenta métodos vacíos.	5	3	2	1
POO y creación de las clases (Categoría, Artículo, Vendedor, Sucursal, Cliente, Artículo por Sucursal y Reserva).	5	3	2	1
Registra Categoría y Artículos en el servidor.	5	3	2	1
Registra Vendedor y Clientes en el servidor.	5	3	2	1
Registra Sucursal y Artículos por Sucursal en el servidor.	10	7	3	1
Consulta Categoría y Artículos en el servidor.	5	3	2	1
Consulta Vendedor y Clientes en el servidor.	5	3	2	1
Consulta Sucursal y Artículos por Sucursal en el servidor.	5	3	2	1
Conexión de la aplicación cliente con el servidor mediante TCP y administración de las conexiones en el servidor.	10	7	3	1
Validación del usuario cliente en la aplicación cliente.	5	3	2	1
Realizar solicitud de reserva en la aplicación cliente.	10	7	3	1

Consulta de préstamos en la aplicación cliente	5	3	2	1
En la aplicación servidor se muestran los eventos enviados por todos los clientes conectados.	5	3	2	1
Implementación de un método para administrar la concurrencia en el servidor.	5	3	2	1
Correcto uso del manejo de excepciones.	5	3	2	1
Interfaz de usuario es fácil de usarse y presenta buen diseño.	5	3	2	1
Manual de uso de la solución.	5	3	2	1
TOTAL	100			