

# UNIVERSIDAD ESTATAL A DISTANCIA VICERRECTORÍA ACADÉMICA ESCUELA DE CIENCIAS EXACTAS Y NATURALES



Código: 00830

# Cátedra Tecnología de Sistemas

Programación Avanzada con C#

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 de subprocesamiento múltiple, bases de datos, colecciones y redes. Además, de continuar con la práctica de conceptos de interfaz gráfica y el manejo de excepciones.

# Software de Desarrollo

Visual Studio Community 2022 (C# Net 8.0)

#### Desarrollo

Los encargados de la empresa **45GAMES4U** están muy contentos por la aplicación desarrollada, y ahora les gustaría agregarle nuevas funcionalidades, como el almacenamiento de los datos en una base de datos SQL Server para guardar la información de forma permanente. Por el otro lado, también quieren agregar una funcionalidad para poder reservar videojuegos y separarlos del inventario. Para lograr la incorporación de esta nueva funcionalidad se debe desarrollar una solución cliente/servidor, que permita a los clientes conectarse mediante una aplicación cliente al servidor central para hacer dichas reservas.

El sistema por desarrollar debe estar compuesto mínimo por dos soluciones; un cliente y un servidor, ambos deben comunicarse a través de la red mediante el protocolo TCP. La solución cliente será utilizada por los clientes para realizar la reserva de videojuegos disponibles en la tienda, mientras que la solución servidor será utilizada por un único administrador en las oficinas centrales para dar

I Cuatrimestre 2025 Página 1 de 18

mantenimiento a las funcionalidades de Tipos de Videojuegos, Videojuegos, Administradores, Tiendas, Inventario y Clientes.

Código: 00830

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

#### La solución 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 videojuegos 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.
  - o 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 un videojuego:

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

- Se debe enviar solitud al servidor para consultar las tiendas, esta información se debe mostrar en un control ComboBox.
  - Solo se deben mostrar las tiendas activas
- Una vez seleccionada la tienda se debe enviar solicitud al servidor para consultar todos los videojuegos asociados a la tienda seleccionada. La información de los videojuegos disponibles se debe mostrar en un control DataGridView.

I Cuatrimestre 2025 Página 2 de 18

 Se debe validar que al menos exista un videojuego disponible en la tienda seleccionada.

Código: 00830

- Una vez seleccionada la tienda y el videojuego el usuario debe seleccionar una fecha de la reservación (datetimepicker) y la cantidad de videojuegos (textbox) que se desee reservar.
- La cantidad de videojuegos digitada debe ser mayor a 0.
- o Debe validar que tenga existencias de videojuegos disponibles para realizar la reserva.
- Una vez validada la reservación, y tomando en consideración que el videojuego se debe rebajar del campo existencias de Videojuegos por Tienda la disponibilidad de ese videojuego asociado.
- o 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).
  - Videojuego por Tienda (Objeto VideojuegosXTiendaEntidad).
  - Cliente (Objeto ClienteEntidad).
  - Fecha Reserva (DateTime).
  - Cantidad (int).
- 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.
- o El cliente muestra en pantalla las reservas realizadas por el cliente.

I Cuatrimestre 2025 Página 3 de 18

 La información debe mostrar el id de la reserva, la información de la Tienda, la información del videojuego, la fecha de la reserva y la cantidad reservada.

Código: 00830

- Se debe implementar dos formas (funcionalidades) de consultar reservas, la primera es consultar todas las reservas realizadas por el cliente; la segunda es 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

# La solución 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.
- Se 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 <u>todos los eventos</u> 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 aceptar un máximo de 5 conexiones simultáneas, una sexta conexión debe esperar hasta que alguna de las 5 conexiones establecidas se desconecte.

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

- Registrar y consultar Tipos de Videojuegos
- Registrar y consultar Videojuegos
- Registrar y consultar Administradores de las Tiendas
- Registrar y consultar Tiendas

I Cuatrimestre 2025 Página 4 de 18

- Registrar y consultar Inventario (Existencias de los Videojuegos por Tiendas)
- Registrar y consultar Clientes

# Registrar Tipos de Videojuegos

Está opción le permitirá a la persona usuaria registrar o agregar los datos de los tipos a los que puede pertenecer un videojuego, por ejemplo: Acción, Aventura, Deportivo, Estrategia, etc. Para ello se debe crear una clase llamada "TipoVideojuegoEntidad".

Se debe poder registrar la siguiente información:

Nombre (string), nombre del tipo de videojuego. Por ejemplo: Acción, Aventura, Deportivo, Estrategia, etc.

Código: 00830

Descripción (string), breve descripción del tipo de videojuego.

Todos los datos son requeridos. El ld se incrementa automáticamente en la base de datos. Una vez obtenida la información, se debe almacenar en la tabla llamada TipoVideojuego.

#### Registrar Videojuegos

Está opción le permitirá a la persona usuaria registrar o agregar los datos de los videojuegos. Para ello se debe crear una clase llamada "VideojuegoEntidad".

Se debe poder registrar la siguiente información:

- Nombre (string), nombre del videojuego.
- > TipoVideojuego (Objeto TipoVideojuegoEntidad), debe realizarse por medio de un Combobox, el cual desplegará los tipos de videojuegos registrados previamente.
- Desarrollador (string), nombre de la empresa desarrolladora del videojuego.
- Lanzamiento (int), año del lanzamiento del videojuego.
- Físico (bool), indica si el videojuego es físico (true) o virtual (false). Se debe utilizar estrictamente un combobox con los textos Físico y Virtual, el cual se debe utilizar el Físico como true y el Virtual como false.

Todos los datos son requeridos. El ld se incrementa automáticamente en la base de datos. Una vez obtenida la información, se debe almacenar en la tabla llamada Videojuego.

# Registrar Administradores de las Tiendas

Está opción le permitirá a la persona usuaria registrar o agregar los datos de los administradores de las tiendas. Para ello se debe crear una clase llamada "AdministradorEntidad".

I Cuatrimestre 2025 Página 5 de 18

Se debe poder registrar la siguiente información:

Identificación (int), identificador del administrador. Este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos.

Código: 00830

- Nombre (string), nombre del administrador.
- PrimerApellido (string), primer apellido del administrador.
- SegundoApellido (string), segundo apellido del administrador.
- ➤ FechaNacimiento (datetime), fecha de nacimiento del administrador. Se debe utilizar el componente de DateTimePicker y asegurarse que el administrador es mayor de edad. Debe mostrar solo la fecha, sin el tiempo.
- ➤ FechaContratación (datetime), fecha de la contratación del administrador. Se debe utilizar el componente de DateTimePicker y no puede ser mayor al día en que se está registrando el administrador en la aplicación (o sea, tiene que ser <= hoy). Debe mostrar solo la fecha, sin el tiempo.

Todos los datos son requeridos. Una vez obtenida la información, se debe almacenar en la tabla llamada Administrador.

# **Registrar Tiendas**

Está opción le permitirá a la persona usuaria registrar o agregar los datos de las tiendas de la cadena. Para ello se debe crear una clase llamada "TiendaEntidad".

Se debe poder registrar la siguiente información:

- Nombre (string), nombre de la tienda.
- Administrador (Objeto AdministradorEntidad), debe realizarse por medio de un Combobox, el cual desplegará a todos los administradores registrados previamente. El combobox debe mostrar el nombre y los dos apellidos del administrador.
- Dirección (string), dirección de la tienda.
- Teléfono (string), teléfono de la tienda.
- Activa (bool), estado de la tienda, si se encuentra activa o no. Se debe utilizar estrictamente un combobox con los textos Si y No, el cual se debe utilizar el Si como true y el No como false.

Todos los datos son requeridos. El ld se incrementa automáticamente en la base de datos. Una vez obtenida la información, se debe almacenar en la tabla llamada Tienda.

I Cuatrimestre 2025 Página 6 de 18

#### **Registrar Clientes**

Está opción le permitirá a la persona usuaria registrar o agregar los datos de una persona cliente. Para ello se debe crear una clase llamada "ClienteEntidad".

Se debe poder registrar la siguiente información:

➤ Identificación (int), identificador del cliente. Este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos.

Código: 00830

- Nombre (string), nombre del cliente.
- PrimerApellido (string), primer apellido del cliente.
- SegundoApellido (string), segundo apellido del cliente.
- FechaNacimiento (datetime), fecha de nacimiento del cliente. Se debe utilizar el componente de DateTimePicker. Debe mostrar solo la fecha y no el tiempo.
- ➤ JugadorEnLinea (bool), almacena si el jugador juega en línea o solo local. Se debe utilizar estrictamente un combobox con los textos Si y No, el cual se debe utilizar el Si como true y el No como false.

Todos los datos son requeridos. Una vez obtenida la información, se debe almacenar en la tabla llamada Cliente.

#### Registrar Inventario

Funcionalidad que debe permitir al usuario registrar (asociar) los videojuegos que estarán disponibles en cada una de las Tiendas. Para ello se debe crear una clase llamada "VideojuegosXTiendaEntidad".

Para esta opción se debe poder registrar la siguiente información:

- ➤ Tienda (Objeto TiendaEntidad), este valor, en combinación con Videojuego, debe de ser único por lo que se debe validar en el sistema que no existan valores repetidos.
- Videojuego (Objeto VideojuegoEntidad), este valor, en combinación con Tienda, debe de ser único por lo que se debe validar en el sistema que no existan valores repetidos.
- Existencias (int), cantidad disponible del videojuego ubicado en la tienda.

Todos los datos son requeridos. Una vez obtenida la información, se debe almacenar en la tabla llamada VideojuegosXTienda.

Para esto se debe realizar lo siguiente:

Mostrar en pantalla en un "ComboBox" las tiendas activas.

I Cuatrimestre 2025 Página 7 de 18

 Se debe validar que al menos exista una tienda registrada activa para realizar asociación.

Código: 00830

- Solo debe mostrar las tiendas que se encuentran activas.
- o Se debe visualizar el nombre de la tienda en el ComboBox.
- Mostrar en un control "DataGridView" los videojuegos disponibles para asignar a la tienda.
  - Se debe validar que al menos exista un videojuego registrado para ser asociado.
  - Para cada videojuego de la lista se debe mostrar la siguiente información del objeto:
     id, nombre, nombre del tipo de videojuego, nombre del desarrollador y año de lanzamiento.
  - Solamente se puede desplegar los videojuegos que se encuentren en su versión física.
- La persona usuaria puede seleccionar uno o más videojuegos del control "DataGridView" para asociar a la tienda.
- ➤ La persona usuaria debe contar con un control de entrada de datos "TextBox", donde podrá definir la cantidad de existencias que contará cada videojuego.
  - o Se debe validar que la cantidad ingresada sea mayor a cero.
  - Si selecciona varios videojuegos para asociar, se les asignará la misma cantidad de existencias a todos.

#### Consultar Tipos de Videojuegos

Está opción le permitirá a la persona usuaria visualizar en pantalla todos los tipos de videojuegos registrados, para tal fin debe hacer uso del control "DataGridView", se debe mostrar todas las propiedades de la clase. (3 columnas)

#### Consultar Videojuegos

Está opción le permitirá a la persona usuaria visualizar en pantalla todos los videojuegos registrados, para tal fin debe hacer uso del control "DataGridView", se debe mostrar todas las propiedades de la clase. Para la propiedad Tipo de videojuego se debe mostrar nombre del tipo de videojuego al que pertenece. (6 columnas)

# Consultar Administradores de las Tiendas

Está opción le permitirá a la persona usuaria visualizar en pantalla a todos los administradores registrados, para tal fin debe hacer uso del control "DataGridView", se debe mostrar todas las

I Cuatrimestre 2025 Página 8 de 18

propiedades de la clase. La fecha de nacimiento y la fecha de contratación no deben mostrar el tiempo, solo la fecha. (6 columnas)

Código: 00830

# **Consultar Tienda**

Está opción le permitirá a la persona usuaria visualizar en pantalla todas las tiendas registradas, para tal fin debe hacer uso del control "DataGridView", se debe mostrar todas las propiedades de la clase. Para la propiedad administrador se debe mostrar el nombre completo del administrador y la fecha de contratación. La fecha de contratación no debe mostrar el tiempo, solo la fecha. Para la columna Activa debe mostrar Si cuando es true y No cuando es false. (7 columnas)

# **Consultar Cliente**

Está 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 todas las propiedades de la clase. La fecha de nacimiento no debe mostrar el tiempo, solo la fecha. Para la columna Jugador en línea debe mostrar Si cuando es true y No cuando es false. (6 columnas)

# **Consultar Inventario**

Está 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 las siguientes propiedades:

- De la Tienda:
  - o ld
  - Nombre
  - o Dirección
- Del Videojuego:
  - o Id
  - Nombre
  - Nombre del tipo de videojuego
- Existencias
- > (7 columnas)

I Cuatrimestre 2025 Página 9 de 18

#### <u>Reservas</u>

La solución servidor debe tener la posibilidad de guardar las reservas que son enviadas desde la solución cliente. Esta información se debe guardar en la tabla llamada Reserva y se debe descontar la cantidad de existencias del inventario. Se debe implementar un sistema de bloqueo para que guarde y actualice el inventario un cliente a la vez, y con esto evitar reservar más cantidad de videojuegos que existencias. Las existencias del inventario no deben quedar en negativo. Debe verificar que haya existencias suficientes antes de guardar la reserva, en caso contrario debe enviarle al cliente un mensaje de error. El id de la reserva se genera automáticamente desde la base de datos.

Código: 00830

#### 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, formulario, etc) que se realiza el código de programación; como se puede observar en el siguiente ejemplo:

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

- ➤ Las clases de objetos que usted considere como entidades (TipoVideojuegoEntidad, VideojuegoEntidad, AdministradorEntidad, TiendaEntidad, ClienteEntidad, VideojuegosXTiendaEntidad, ReservaEntidad) no deben contener lógica para solicitar información a la persona usuaria, solo debe tener la estructura de la clase y sus propiedades. El nombre del objeto y las propiedades no deben de llevar tilde ni ñ.
- ➤ Implementar en las soluciones proyectos independientes con el fin de tener las distintas capas recomendadas: 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.

I Cuatrimestre 2025 Página 10 de 18

Las clases no deben tener métodos vacíos y recuerde utilizar nombres significativos a sus variables.

Código: 00830

Las entidades AdministradorEntidad y ClienteEntidad tienen varias propiedades iguales, por lo cual se puede crear una entidad nueva llamada PersonaEntidad, se debe colocar las propiedades en común en esta nueva entidad y aplicar herencia.

#### Propiedades Id e Identificación

No se debe crear una variable que se incremente con cada inserción, en el caso de los Ids la base de datos los crea automáticamente. En el caso de la identificación se debe permitir que el usuario la digite.

#### Uso de combobox

Los combobox no se deben poder editar o escribir en él, solo debe permitir seleccionar una opción de la lista.

#### Uso de colecciones

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

#### Interfaz de usuario

- Debe usar interfaz de usuario con formularios GUI (Proyecto Windows Form).
- > Se debe cuidar con el orden de los formularios.
- Se debe cuidar con la ortografía.
- Se debe crear un menú para acceder a las opciones solicitadas.
- ➤ Cada vez que se agrega un objeto se deben limpiar los campos de la pantalla y mostrar el siguiente mensaje: "El registrado se ha ingresado correctamente". Esto con el fin de que el usuario pueda ingresar otro registro.

#### 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.

I Cuatrimestre 2025 Página 11 de 18

#### 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.

Código: 00830

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

#### Uso de DataGridView

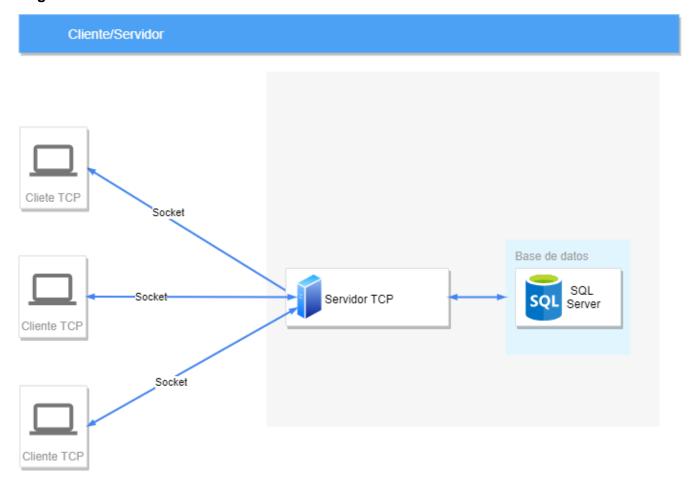
- Los títulos de las columnas deben de estar previamente establecidos para que se muestren de forma correcta, y no se debe dejar que tomen el nombre de la propiedad de la clase.
- Los DataGridView debe de ser de solo lectura, no se pueden modificar los datos que se muestren allí. Tampoco debe permitir agregar o borrar nuevos datos.
- Cuando se seleccione un dato del DataGridView se debe seleccionar toda la fila.

#### 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 proyecto independiente de tipo biblioteca de clases para las entidades que se van a utilizar en las dos soluciones, de esa manera no tendrá que duplicar las clases de entidades en la solución cliente y servidor, en estos solo se agrega la referencia del proyecto o del archivo 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.

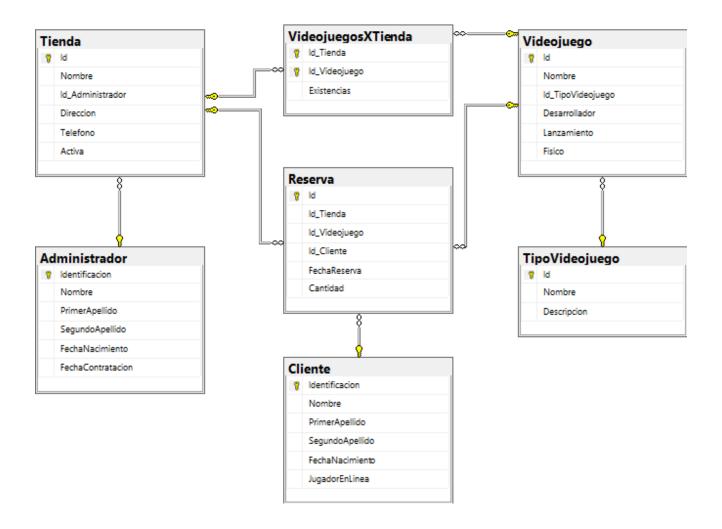
I Cuatrimestre 2025 Página 12 de 18

# Diagrama cliente/servidor



I Cuatrimestre 2025 Página 13 de 18

# Diagrama base de datos



I Cuatrimestre 2025 Página 14 de 18

#### Honestidad Académica



https://audiovisuales.un ed.ac.cr/play/player/230 48

Código: 00830



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 2.
- > El proyecto 2 debe estar desarrollado en Visual Studio Community 2022 que es la herramienta oficial del curso.
- > 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-proyecto2.
- La entrega del proyecto debe ser en las fechas establecidas en la plataforma de aprendizaje en línea Moodle en el apartado que se indique.
- Si no concluyó a tiempo la tarea, debe entregar lo que pudo hacer e incluir una carta explicando las razones por las cuales no finalizó.

I Cuatrimestre 2025 Página 15 de 18

# 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 y variables significativos, no presenta métodos vacíos. Entrega el archivo del				
programa con todo el directorio del programa con todo el directorio del programa desarrollado. Indica dentro del programa en los comentarios, si se utiliza código de algún ejemplo del libro o algún profesor o de otra fuente que no sea de autoría del estudiante. El nombre del archivo es el solicitado. Agrega los comentarios de información	5	3	2	0
del proyecto en las clases.  POO y creación de las clases (TipoVideojuegoEntidad, VideojuegoEntidad, AdministradorEntidad, TiendaEntidad, ClienteEntidad, VideojuegosXTiendaEntidad, ReservaEntidad). Utiliza propiedades y no atributos. No agrega	5	3	2	0

Código: 00830

I Cuatrimestre 2025 Página 16 de 18

constructores ni métodos				
adicionales. El proyecto está				
dividido en capas.				
Registra Tipos de Videojuegos y	5	3	2	0
Videojuegos en el servidor.				
Registra Administradores y	5	3	2	0
Clientes en el servidor.				
Registra Tiendas y Videojuegos	10	7	3	0
por Tienda en el servidor.				
Consulta Tipos de Videojuegos				
y Videojuegos desde el	5	3	2	0
servidor.				
Consulta Administradores y	5	3	2	0
Clientes desde el servidor.				
Consulta Tiendas y Videojuegos	5	3	2	0
por Tienda desde el servidor.				
Conexión de la aplicación				
cliente con el servidor mediante	10	7	3	0
TCP y administración de las				
conexiones en el servidor.				
Validación del usuario cliente en	5	3	2	0
la aplicación cliente.				
Realizar solicitud de reserva en la aplicación cliente.	10	7	3	0
Consulta de reservas en la				
aplicación cliente	5	3	2	0
En la aplicación servidor se				
muestran los eventos enviados				
por todos los clientes	5	3	2	0
conectados (bitácora).				
Implementación de un método				
para administrar la concurrencia	5	3	2	0
en el servidor.				

Código: 00830

I Cuatrimestre 2025 Página 17 de 18

TOTAL	100			
Manual de uso de la solución.	5	3	2	0
formularios.				
navegación entre los				
Presenta una correcta forma de				
satisfacción de forma amigable.				
mensajes de error y de	3	3		U
los DateTimePicker. Muestra	5	3	2	0
DataGridView, los Combobox y				
Presenta correctamente los				
usarse y presenta buen diseño.				
Interfaz de usuario es fácil de				
excepciones con try catch.	J	9	_	J
Correcto uso del manejo de	5	3	2	0

Código: 00830

I Cuatrimestre 2025 Página 18 de 18