



UNIVERSIDAD ESTATAL A DISTANCIA
VICERRECTORÍA ACADÉMICA
ESCUELA DE CIENCIAS EXACTAS Y NATURALES
Cátedra Tecnología de Sistemas



Programación Avanzada

Código: 00830

Proyecto. Valor 30%

Temas de Estudio

1. Tema 2 Particularidades del lenguaje C#
2. Tema 3 Manejo de excepciones en C#
3. Tema 4 Conceptos de Interfaz Gráfica
4. Tema 5 Subprocesamiento múltiple
5. Tema 6 Bases de datos y componentes ADO.NET
6. Tema 7 Colecciones
7. 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.

Software de Desarrollo

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

Desarrollo

Debido al éxito que se ha obtenido con la segunda versión del sistema de información entregada para la empresa **AUTOTRANSPORTES-UNED**, usted ha sido contratado nuevamente para hacer mejoras en su sistema de información, entre las cuales se destaca que la información, sea almacenada de forma permanente en una base de datos SQL Server. A la fecha los conductores no saben cómo

estará su rol, hasta que lleguen a las oficinas centrales de la empresa, y los supervisores de forma manual, tienen que estar realizando los respectivos roles.

Por lo anterior, se requiere incorporar nuevas funcionalidades que permitan a los conductores, conocer cuál será su rol, y los supervisores, realizar los roles correspondientes.

El sistema para 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 tanto por los conductores normales como supervisores, con el fin de verificar mis rutas o todas las rutas en el caso de los supervisores, así como crear los respectivos roles, mientras que el proyecto servidor será utilizado por un único administrador en las oficinas centrales para dar mantenimiento a las funcionalidades de Registrar Terminales, Registrar Conductores, Registrar Autobuses y Tarifas. Toda la información debe ser almacenada en una base de datos SQL Server.

El proyecto cliente debe cumplir con lo siguiente:

- Debe ser utilizado por los conductores para conectarse con el servidor central y conocer sus rutas asignadas y en el caso de los supervisores para crear los roles.
- Debe tener una opción para conectarse y desconectarse del servidor.
- Al iniciar el programa cliente debe solicitar el número de identificación del cliente
 - Se debe validar que el conductor 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 conductor que ingresó.
 - El conductor debe haber sido registrado previamente por el administrador en el servidor.
 - Si la identificación existe, es decir, es un conductor registrado y activo se permite el ingreso a la aplicación cliente.

Una vez que el conductor ha sido validado, en la aplicación cliente se debe mostrar el nombre del conductor 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 las funcionalidades de la aplicación dependiendo del perfil de la persona que se conectó:

Para un Conductor Supervisor deberá habilitar las siguientes opciones:

- Registrar Roles
- Consultar Roles

Para un Conductor Normal:

- Consultar Roles

1. Registrar Roles:

Funcionalidad que debe permitir al conductor supervisor crear los roles de forma diaria, antes de ingresar a esta opción el sistema deberá realizar las siguientes validaciones:

Se debe poder registrar la siguiente información:

- Fecha (date). Por regla del negocio los roles deben ser creados al menos con 2 días de antelación, por ende, debe validarse que la fecha de registro corresponda al menos a los dos días posteriores a la fecha que tiene registrado el sistema
- Hora de Salida (datetime), deberá validarse que la hora digitada sea válida.
- Id Ruta, (9999) debe validarse que el Id de la Ruta exista, en caso de que no exista, indicar mensaje de error de que la ruta no existe, si es válida se deberá desplegar:
 - Terminal de Origen
 - Terminal Destino
 - Tarifa
 - Tipo

Se recomienda el uso de un ComboBox, que contenga las rutas y que automáticamente al escoger la ruta, se llene automáticamente el ID de acuerdo con el nombre de la ruta seleccionada.

- Número de identificación del Autobús (9999), debe validarse que número de autobús exista, en caso de que no exista, indicar mensaje de error de que el autobús no existe, si es válida se deberá desplegar:
 - Placa
 - Marca
 - Capacidad del autobús

Además, se deberá validar que el autobús, no tenga programada una salida a la misma hora, en otra terminal y dado que todas las rutas son mayores a 2 horas, se deberá validar que en se lapso no puede asociarse otra salida al mismo autobús.

Se recomienda el uso de un ComboBox, que contenga todos los autobuses y que automáticamente al escoger el autobús, se llene automáticamente el ID de acuerdo con la descripción del autobús seleccionado.

- Conductor varchar(15): se deberá validar que el conductor exista, que este activo y que no tenga programada una salida a la misma hora ni en las próximas dos horas, en otro autobús, ni que se trate de un supervisor conductor.

Se recomienda el uso de un ComboBox, que contenga todos los conductores y que automáticamente al escoger el conductor, se llene automáticamente el ID de acuerdo con la descripción del conductor seleccionado.

2. Consultar Roles

Conductor Normal: Se deberá presentar un formulario, donde solicite el rango de fechas, de la consulta a realizar, las cuales deben ser fechas válidas, y la fecha Inicial debe ser Mayor a la final, una vez captura esta información, Deberá enviar una solicitud al servidor para consultar los roles asignados de acuerdo a la cédula del conductor que está conectado a la aplicación cliente, en las fechas solicitadas, en caso de que no existan datos, se deberá desplegar un mensaje indicado que no se obtuvieron datos en las fechas solicitadas

Conductor Supervisor: Se deberá presentar un formulario, donde solicite el rango de fechas, de la consulta a realizar, las cuales deben ser fechas válidas, y la fecha Inicial debe ser Mayor a la final, una vez captura esta información, así solicitar el id de la Ruta, para la cual se desea consultar los roles, así como un combo box, donde se solicite el tipo de consulta a Realizar: Por Ruta, Conductor o Id del Autobús, en el caso

- Consulta por Ruta: debe validarse que el Id de la Ruta exista, en caso de que no exista, indicar mensaje de error de que la ruta no existe, si es válida se deberá desplegar:
 - Terminal de Origen
 - Terminal Destino

- Tarifa
- Tipo

Se recomienda el uso de un ComboBox, que contenga las rutas y que automáticamente al escoger la ruta, se llene automáticamente el ID de acuerdo con el nombre de la ruta seleccionada.

- Consulta Conductor: Se deberá validar que el conductor exista, en el caso de que el conductor no exista, deberá enviar un mensaje de error indicado que el conductor no existe, una vez validado deberá desplegar el nombre del conductor asignado.

Se recomienda el uso de un ComboBox, que contenga todos los conductores y que automáticamente al escoger el conductor, se llene automáticamente el ID de acuerdo con la descripción del conductor seleccionado.

- Consulta por Número de identificación del Autobús, debe validarse que número de autobús exista, en caso de que no exista, indicar mensaje de error de que el autobús no existe, si es válida se deberá desplegar:
 - Placa
 - Marca
 - Capacidad del autobús

Se recomienda el uso de un ComboBox, que contenga todos los autobuses y que automáticamente al escoger el autobús, se llene automáticamente el ID de acuerdo con la descripción del autobús seleccionado.

Una vez seleccionada la información del rango de fechas, así como el tipo de consulta a realizar, deberá realizar una petición al servidor para presentar la información, por medio de un control de tipo DataGridView todas las Roles Registrados de acuerdo al criterio solicitado.

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

- 1. Registrar Terminales**
- 2. Registrar Conductores**
- 3. Registrar Autobuses**
- 4. Registrar Tarifas**

1. Registrar Terminales:

Esta opción del menú le permite al administrador del sistema registrar o agregar los datos relacionados con las terminales de la empresa. En la base de datos. Se debe poder registrar la siguiente información:

- Id de terminal numeric(3,0), este deberá ser generado de forma automática por el sistema
- Nombre de terminal varchar(40)
- Dirección de terminal varchar(40)
- Teléfono de terminal varchar(8)
- Hora de Apertura (time)
- Hora de Cierre (time)
- Estado (bit), True=Activo, False=Inactivo, deberá utilizarse un combo box, que incluya las opciones Activo e Inactivo.
- Es importante indicar que los horarios deberán ser capturados en formato militar, de acuerdo con el siguiente ejemplo: Si la terminal abre a las 5:00 am y cierra a las 6:00 pm, el horario debe escribirse de la siguiente forma: De: 08:00-18:00

Una vez capturada la información se debe agregar a un arreglo de objetos de tipo Terminal. El arreglo debe ser de 20 posiciones

2. Registrar Conductores:

Esta opción del menú le permite al administrador del sistema registrar o agregar los datos relacionados a los conductores de la empresa, en la base de datos. Se debe poder registrar la siguiente información:

- Identificación varchar(15), se debe validar que no existan identificaciones repetidas
- Nombre varchar(40)
- Primer apellido varchar(40)
- Segundo apellido varchar(40)
- Fecha de nacimiento (date)
- Género (char) M=Masculino, F=Femenino, deberá utilizarse un combo box, que incluya las opciones Masculino y Femenino
- Conductor Supervisor (bit)

3. Registrar Autobuses:

Esta opción del menú le permite al administrador del sistema, registrar o agregar los datos relacionados a los Autobuses de la empresa, en la base de datos. Se debe poder registrar la siguiente información:

- Número de identificación numeric(4,0) este deberá ser generado de forma automática por el sistema
- Id Placa varchar(10) se debe validar que no existan id's de placas repetidas
- Marca varchar(30)
- Modelo numeric(4,0)
- Capacidad numeric(3,0)
- Estado (bool), True=Activo, False=Inactivo, deberá utilizarse un combo box, que incluya las opciones Activo e Inactivo.

4. Registrar Rutas:

Está opción del menú le permite al administrador del sistema, registrar o agregar los datos relacionados a las tarifas de la empresa, en la base datos. Se debe poder registrar la siguiente información:

- Id numeric (4,0), este deberá ser generado de forma automática por el sistema
- Id Terminal de Salida numeric(3,0): (Objeto Terminal), debe validarse que el Id de la terminal exista, en caso de que no exista, indicar mensaje de error de que la terminal de salida no exista, si es válida se deberá desplegar el nombre de la terminal.
Se recomienda el uso de un ComboBox, que contenga las terminales y que automáticamente al escoger la terminal, se llene automáticamente el ID de acuerdo con el nombre de terminal de Salida seleccionada.
- Id Terminal de Llegada numeric(3,0): (Objeto Terminal), debe validarse que el Id de la terminal exista, en caso de que no exista, indicar mensaje de error de que la terminal de llegada no existe, si es válida se deberá desplegar el nombre de la terminal. Adicionalmente se debe validar que el Id de la terminal de salida, sea diferente al de la terminal de llegada.
Se recomienda el uso de un ComboBox, que contenga las terminales y que automáticamente al escoger la terminal, se llene automáticamente el ID de acuerdo con el nombre de terminal de salida seleccionada.
- Tarifa numeric(8,2)
- Descripción de la Ruta varchar(40)
- Tipo de Ruta numeric(2,0) 1=Directo, 2=Regular, deberá utilizarse un combo box, que incluya las opciones Directo y Regular
- Estado (bit), True=Activo, False=Inactivo, deberá utilizarse un combo box, que incluya las opciones Activo e Inactivo.

5. Consultar Terminales:

Esta opción permite al usuario consultar en pantalla, por medio de un control de tipo DataGridView, todas las terminales registradas, para cada Terminal se debe mostrar todas las propiedades de las terminales registradas.

6. Consultar Conductores:

Esta opción permite al usuario consultar en pantalla, por medio de un control de tipo DataGridView, todos los datos de conductores registrados, para cada conductor se debe mostrar todas las propiedades.

7. Consultar Autobuses:

Esta opción permite al usuario consultar en pantalla, por medio de un control de tipo DataGridView, todos los autobuses registrados, para cada autobús se debe mostrar todas las propiedades

8. Consultar Rutas

Esta opción permite al usuario consultar en pantalla, por medio de un control de tipo DataGridView, todas las rutas registradas, para cada ruta se debe mostrar todas las propiedades de las tarifas y los datos de la terminal de salida y destino (Nombre, Dirección, Horario de Apertura, Horario de Cierre)

Consideraciones técnicas

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

- Las clases de objetos (Terminal, Conductor, Autobús, Tarifa, Roles), no deben contener lógica para solicitar información al usuario, solo debe tener la estructura de la clase y sus propiedades.
- Las clases no deben tener métodos vacíos

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.

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 utilizar DataSet ni DataTable.

Interfaz de usuario

- Debe almacenar y consultar la información en una base de datos SQL Server. No se permite otro motor de base de datos.

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.
- **El script de creación de la base de datos se publicará en la plataforma. No de be modificar el script de base de datos**

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.

Diagrama de la Aplicación

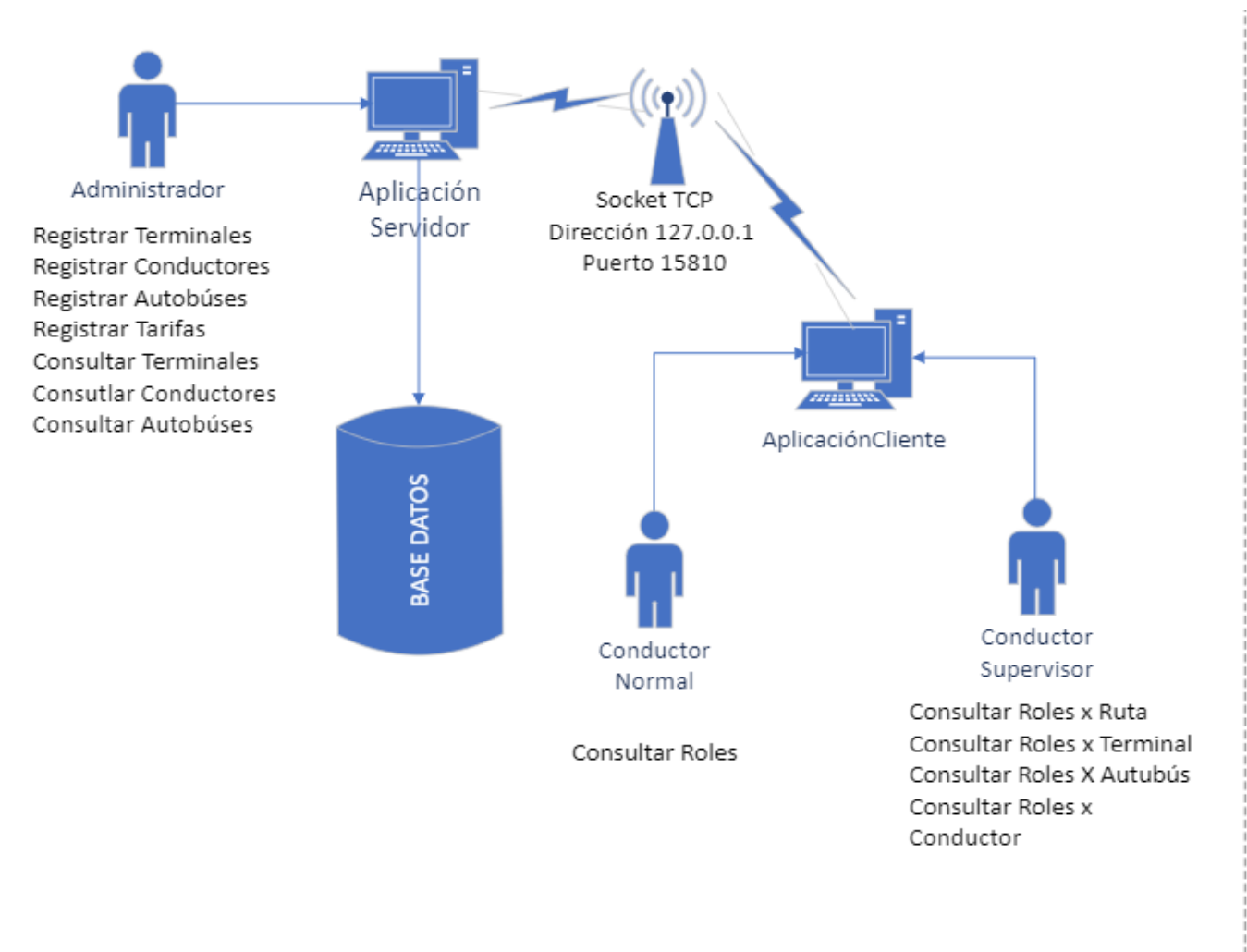


DIAGRAMA DE BASE DATOS BUSUNED

TERMINAL			
	Column Name	Data Type	Allow Nulls
🔑	COD_TERMINAL	numeric(3, 0)	<input type="checkbox"/>
	DSC_TERMINAL	varchar(40)	<input type="checkbox"/>
	NUM_TELEFONO	varchar(8)	<input type="checkbox"/>
	TIM_HORA_APERTURA	time(7)	<input type="checkbox"/>
	TIM_HORA_CIERRE	time(7)	<input type="checkbox"/>
	BOL_ESTADO	bit	<input type="checkbox"/>
			<input type="checkbox"/>

AUTOBUS			
	Column Name	Data Type	Allow Nulls
🔑	NUM_IDENTIFICACION	numeric(4, 0)	<input type="checkbox"/>
	NUM_PLACA	varchar(10)	<input type="checkbox"/>
	DSC_MARCA	varchar(30)	<input type="checkbox"/>
	NUM_MODELO	numeric(4, 0)	<input type="checkbox"/>
	NUM_CAPACIDAD	numeric(3, 0)	<input type="checkbox"/>
	BOL_ESTADO	bit	<input type="checkbox"/>
			<input type="checkbox"/>

RUTA			
	Column Name	Data Type	Allow Nulls
🔑	NUM_RUTA	numeric(4, 0)	<input type="checkbox"/>
	COD_TERMINAL_SALIDA	numeric(3, 0)	<input type="checkbox"/>
	COD_TERMINAL_LLEGADA	numeric(3, 0)	<input type="checkbox"/>
	NUM_TARIFA	numeric(8, 2)	<input type="checkbox"/>
	DSC_RUTA	varchar(40)	<input type="checkbox"/>
	NUM_TIPO_RUTA	numeric(2, 0)	<input type="checkbox"/>
	BOL_ESTADO	bit	<input type="checkbox"/>
			<input type="checkbox"/>

CONDUCTOR			
	Column Name	Data Type	Allow Nulls
🔑	NUM_CEDULA	varchar(15)	<input type="checkbox"/>
	NOM_NOMBRE	varchar(40)	<input type="checkbox"/>
	NOM_APELLIDO_1	varchar(40)	<input type="checkbox"/>
	NOM_APELLIDO_2	varchar(40)	<input type="checkbox"/>
	FEC_NACIMIENTO	date	<input type="checkbox"/>
	TIP_GENERO	char(1)	<input type="checkbox"/>
	BIT_SUPERVISOR	bit	<input type="checkbox"/>
			<input type="checkbox"/>

ROL			
	Column Name	Data Type	Allow Nulls
🔑	FEC_ROL	date	<input type="checkbox"/>
🔑	TIM_HORA_SALIDA	time(7)	<input type="checkbox"/>
🔑	NUM_RUTA	numeric(4, 0)	<input type="checkbox"/>
🔑	NUM_IDENTIFICACION_BUS	numeric(4, 0)	<input type="checkbox"/>
	NUM_CEDULA_CONDUCTOR	varchar(15)	<input type="checkbox"/>
			<input type="checkbox"/>

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 proyecto final.
- El proyecto debe estar desarrollado en Visual Studio Community 2022 (C#, .Net Framework 4.8 o Net 6.0) 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, 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-Proyecto.**
- La entrega del proyecto debe realizarse 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.

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.	3	2	1	0.5
Registrar Terminales en Servidor	5	3	2	1
Registrar Conductores en el Servidor	5	3	2	1
Registrar Autobuses en el Servidor	5	3	2	1
Registrar Tarifas en el Servidor	5	3	2	1
Consultar Terminales en el Servidor	5	3	2	1
Consultar Conductores en el Servidor	5	3	2	1
Consultar Autobuses en el Servidor	5	3	2	1
Consultar Tarifas en el Servidor	5	3	2	1

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
Conexión de la aplicación cliente con el servidor mediante TCP y administración de las conexiones en el servidor	5	3	2	1
Validación del conductor, en la aplicación cliente	5	3	2	1
Consultar Roles en la aplicación cliente	10	5	2	1
Registro de Roles en la aplicación cliente	10	5	2	1
En la aplicación servidor se muestran los eventos enviados por todos los clientes conectados	5	3	2	1
Correcto uso de subprocesamiento múltiple en servidor	5	3	2	1
Implementación de un método para administrar la concurrencia de recursos compartidos en el servidor al registrar los roles	5	3	2	1

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
Correcto uso del manejo de excepciones	3	2	1	0.5
La Interfaz de usuario GUI, es fácil de usar y de buen diseño.	5	3	2	1
Manual de uso de la solución	4	2	1	0.5
TOTAL	100			