

Estrategia

[Diagrama de entidad - relación](#)

[Arquitectura para el acceso a datos](#)

[Comunicación entre aplicación y base de datos](#)

[Diseño de la interfaz gráfica](#)

[Identificación del usuario](#)

Introducción

Este documento acompaña a la entrega del trabajo práctico FrbaBus y su objetivo es poder facilitar la comprensión de cómo se desarrolló la solución propuesta.

Secciones

La sección Diagrama de entidad - relación muestra el diseño de nuestro modelo de datos.

La sección Arquitectura para el acceso a datos presenta la organización en capas lógicas que tiene la aplicación para abstraerse del acceso a datos.

La sección Comunicación entre aplicación y base de datos explica el funcionamiento del componente creado para vincular la aplicación con el motor de base de datos.

La sección Diseño de la interfaz gráfica comenta que lineamientos seguimos para crear los formularios.

La sección Identificación del usuario explica el manejo de la seguridad en la aplicación.

Estructura de la entrega

La estructura cumple con las condiciones dadas:

- TP1C2013 K3051 SI_NO_APROBAMOS_HAY_TABLA 16 ZIP123
 - data
 - script_creacion_inicial.sql
 - src
 - FrbaBus.sln
 - GestionDeDatos.AccesoDatos
 - app.config
 - Estrategia.pdf
 - Readme.txt

De forma tal que los entregables obligatorios se encuentran en:

- Estrategia
 - TP1C2013 K3051 SI_NO_APROBAMOS_HAY_TABLA 16 ZIP123\Estrategia.pdf

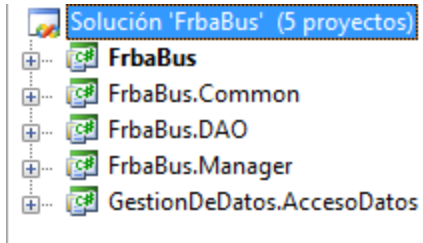
- Readme
 - TP1C2013 K3051 SI_NO_APROBAMOS_HAY_TABLA 16
ZIP123\Readme.txt
- Script de creación inicial
 - TP1C2013 K3051 SI_NO_APROBAMOS_HAY_TABLA 16
ZIP123\data\script_creacion_inicial.sql
- Solución de VisualStudio 2008 Express
 - TP1C2013 K3051 SI_NO_APROBAMOS_HAY_TABLA 16
ZIP123\src\FrbaBus.sln
- Archivo de configuración de la aplicación
 - TP1C2013 K3051 SI_NO_APROBAMOS_HAY_TABLA 16
ZIP123\src\GestionDeDatos.AccesoDatos\app.config

Diagrama de entidad - relación

Se deja adjunto el diagrama DER en la misma carpeta que este documento para facilitar su visualización.

Arquitectura para el acceso a datos

La aplicación se encuentra dividida en capas de abstracción para facilitar el desarrollo y las pruebas.



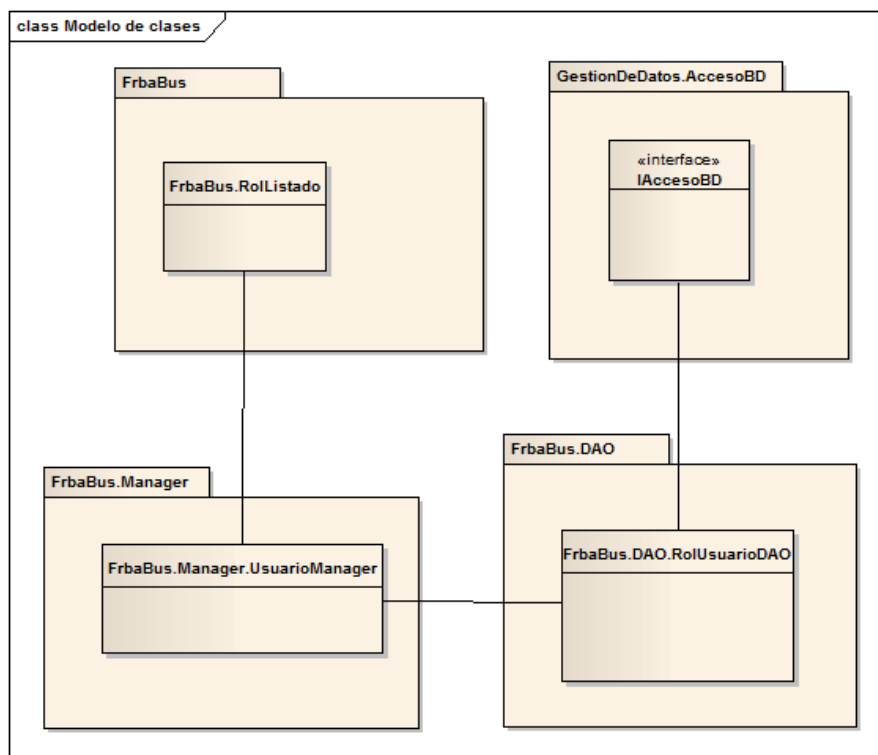
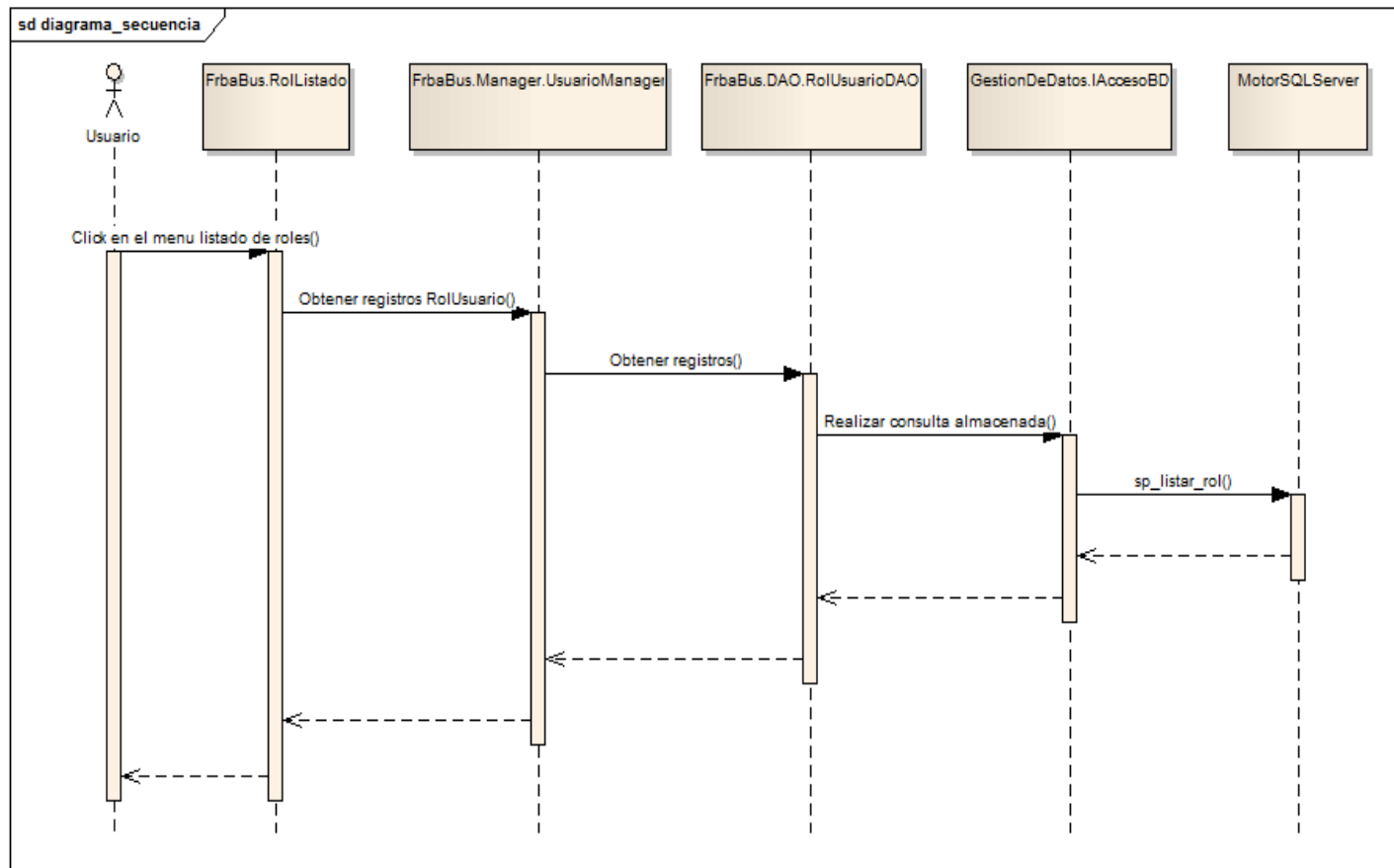
La capa superior es la interfaz gráfica realizada con WinForms del framework .NET. Por lo general cada entidad del modelo tiene su pantalla de listado que permite, además de filtrar, poder dar de baja y modificar instancias de la entidad.

La capa de interfaz gráfica se apoya en distintos administradores de entidades, que se encuentran en la capa del dominio, que son los encargados de proveer las entidades desde la base de datos. Generalmente, cada entidad tiene su propio administrador en esta capa.

La capa de dominio devuelve las entidades en forma de objetos preparados con getters y setters para que la interfaz gráfica pueda mostrarlos en controles de Textbox, Combobox, CheckListBox, RadioButton y otros. Para ello cada administrador conoce uno o mas objetos de la capa de acceso a datos que son los que conocen como solicitarle los datos a la base de datos.

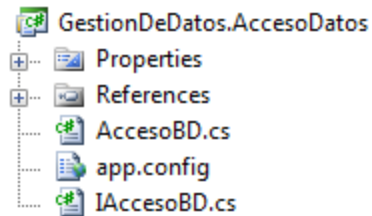
Esta capa de acceso a datos del modelo a su vez, se abstrae de la administración de la conexión contra la base de datos, delegando ésta responsabilidad en el componente GestionDeDatos.AccesoDatos, que sabe como abrir un canal de comunicación, obtener datos mediante consultas o procedimientos almacenados, cerrar la comunicación y devolver resultados.

Este componente se conecta a la base de datos mediante un archivo de configuración que contiene una cadena de conexión con todos los datos necesarios.



Comunicación entre aplicación y base de datos

El componente desarrollado para la aplicación llamado GestionDeDatos.AccesoDatos es el más ligado al motor de la base de datos. Toda petición para obtener entidades de la base de datos, provenientes de alguna pantalla, terminará en una solicitud a este componente.



Contrato del componente

El componente tiene una interfaz IAccesoBD donde expone todas las operaciones disponibles contra el motor de base de datos. Estas operaciones son las siguientes:

- RealizarConsulta
 - Permite realizar una sentencia en lenguaje SQL y obtener el resultado de la misma en un objeto DataSet del framework .NET
- EjecutarComando
 - A diferencia de RealizarConsulta, se ejecuta un comando, que puede ser una sentencia en lenguaje SQL o una llamada a un procedimiento, pero no se devuelven resultados. Solo devuelve la cantidad de filas afectadas por la ejecución.
- RealizarConsultaAlmacenada
 - Permite invocar un procedimiento almacenado, pudiendo pasar parámetros y devuelve un DataSet con el resultado de la ejecución del procedimiento.

El componente maneja los parámetros de las consultas y comandos mediante un diccionario de SqlParameter y su valor, que puede ser cualquier objeto.

Diseño de la interfaz gráfica

Se optó por una pantalla principal con menús para cada entidad. Dentro de esta pantalla principal se permite realizar la compra de pasajes/encomiendas, ya que es la actividad principal del negocio. En la parte inferior de la pantalla se encuentra una barra de estado, con la información de seguridad correspondiente al usuario activo.

Para cada entidad se tiene un menú dentro de la pantalla principal, donde se despliegan las opciones para listar y dar de alta.

FrbaBus - Pantalla Principal

Archivo Rol Ciudad Recorrido Micro Viaje Cliente Pasaje/Encomienda Estadísticas Premios

Compras

Viaje

Fecha de salida: junio 21, 2013 - 11:03:13

Ciudad origen: Banfield

Ciudad destino: Banfield

Buscar micros

Micros

Micro:

Butacas libres:

Kgs disponibles:

Tipo de servicio:

Elegir micro

Medio de pago

Detalles de la compra

☐ Pasajes ☐ Encomiendas

Pasajeros

Cantidad de pasajes:

Agregar

Modificar

Quitar

Encomiendas

Agregar

Modificar

Quitar

Usuario: admin2 Rol: Administrativo (Sesión iniciada)

El listado permite visualizar todas las instancias de la entidad y contiene a su vez filtros para reducir la cantidad de resultados. Desde este mismo listado podemos seleccionar una instancia y poder tanto realizarle una modificación como una baja lógica.

FrbaBus - Roles :: Listado

Filtros de búsqueda

Rol

Funcionalidad

Filtrar

	id_rol	nombre	inhabilitado
▶	1	Administrativo	<input type="checkbox"/>
	2	Cliente	<input type="checkbox"/>
	5	adf	<input type="checkbox"/>
	6	uuiuadf	<input type="checkbox"/>
	7	uuijhfftyfadt	<input type="checkbox"/>
	8	uuijhfftyfhghadt	<input type="checkbox"/>

Modificar Dar de baja

Para el alta se tiene un formulario con todos los campos requeridos. A su vez, si se necesitan campos que hagan referencia a otra tabla, se cargarán las descripciones para que el usuario pueda elegir e internamente se guarda el id de esa entidad.

FrbaBus - Roles :: Alta

Nuevo Rol

Nombre

Funcionalidades

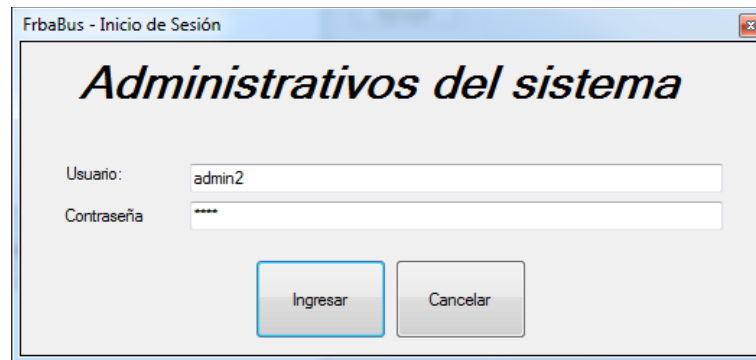
- ☒ tsmRolAlta
- ☐ tsmRolListado
- ☐ tsmCiudadAlta
- ☐ tsmCiudadListado
- ☐ tsmRecomidoAlta
- ☐ tsmRecomidoListado
- ☐ tsmMicroAlta
- ☐ tsmMicroListado
- ☐ tsmClienteListado
- ☐ tsmClienteAlta
- ☐ tsmClientePasajeroFrecuenteConsultar
- ☐ tsmViajeAlta
- ☐ tsmViajeListado
- ☐ tsmPasajeEncomiendaAlta
- ☐ tsmPasajeEncomiendaCancelar
- ☐ tsmEstadisticasT5ClientesMasPuntos
- ☐ tsmEstadisticasT5DestMasCancelados
- ☐ tsmEstadisticasT5DestMasMicrosVacios
- ☐ tsmEstadisticasT5DestMasVendidos
- ☐ tsmEstadisticasT5MicrosMasDiasFueraDeServicio
- ☐ tsmPremiosAlta
- ☐ tsmPremiosListado

Limpiar Guardar

Identificación del usuario

Sesiones de usuarios administrativos

Al iniciar la aplicación se visualiza en la pantalla principal el formulario de compra. Desde aquí los clientes pueden realizar sus operaciones e identificarse al momento de poner su DNI. Pero para los usuarios administrativos existe la opción de iniciar su sesión en el sistema, a través del menú Archivo\Sesión\Iniciar sesión.



The screenshot shows a Windows-style dialog box titled 'FrbaBus - Inicio de Sesión'. Inside the dialog, the text '*Administrativos del sistema*' is displayed in a large, bold, italicized font. Below this, there are two input fields: 'Usuario:' with the text 'admin2' and 'Contraseña:' with five asterisks. At the bottom, there are two buttons: 'Ingresar' (highlighted with a blue border) and 'Cancelar'.

Al solicitar identificarse desde esta pantalla, la aplicación termina invocando un procedimiento que, partiendo del nombre de usuario y el password, informa por medio de un código de retorno el resultado de la identificación.

La regla de los códigos de retorno, se modeló en un enumerador y es la siguiente:

- Usuario Inválido
 - Valor -2
- Usuario Bloqueado
 - Valor -1
- Usuario Identificado
 - Valor 0

Encriptación de las contraseñas

Para cumplir con los requerimientos del trabajo práctico, desarrollamos una clase helper que se encarga de recibir una cadena de caracteres y devuelve un array de bytes correspondientes al hash encriptado por el algoritmo de seguridad.

La tabla de usuarios nunca conoce los valores reales de las contraseñas porque almacena en la columna password los bytes suministrados al procedimiento de alta de usuario.

Por lo tanto, cuando un usuario se identifica, se invoca al procedimiento de identificación de usuarios con el nombre y el hash, para que el motor solo realice una comparación de valores.

Usuarios provistos para las pruebas

Se dejan a disposición para las pruebas, los siguientes usuarios administrativos. Todos con la contraseña “w23e”.

- admin1
- admin2
- admin3
- admin4