



CREANDO APLICACIONES WPF

Consigna general: Desarrollar en Windows Presentation Foundation (WPF) los módulos de una aplicación para gestionar las operaciones de una Playa de Estacionamiento. La aplicación será un desarrollo escalable de acuerdo a las consignas y temas aprendidos en cada módulo de la materia.

Consigna particular: Crear la primera aplicación WPF. Familiarizarse con la herramienta de desarrollo. Crear y mostrar Ventanas. Crear e instanciar clases. Conocer el uso y manejo de controles comunes (Buttons, Labels, TextBoxs, etc.).

Ejercicios:

Crear una Solución en Visual Studio2010 denominada "LPOOII_GRUPOXX", donde "XX" representa el número del grupo al que pertenece. La solución deberá contener dos proyectos individuales. El primer proyecto será de tipo "Biblioteca de Clases" y el segundo proyecto será de tipo "Windows Presentation Foudation (WPF)". Para ello se debe tener en cuenta los siguientes pasos:

- 1) Crear un proyecto denominado "ClasesBase". Este proyecto contendrá las Clases a utilizar por el sistema.
- 2) Crear un *Diagrama de Clases* (utilizando el diseñador de clases) que contenga las siguientes Clases:

Clase	Propiedades
Sector	SectorCodigo (int) Descripcion (string) Identificador (string) (Ej: "E1", "E2") Habilitado (bool)
TipoVehiculo	TVCodigo (int) Descripcion (string) (Ej: "Auto", "Moto") Tarifa (decimal)
Cliente	ClienteDNI (int) Apellido (string) Nombre (string) Telefono (string)
Ticket	TicketNro (int) FechaHoraEnt (datetime) FechaHoraSal (datetime) ClienteDNI (int) TVCodigo (int) Patente (string) SectorCodigo (int) Duracion (double) Tarifa (decimal) Total (decimal)
Usuario	UserName (string) Password (string) Apellido (string) Nombre (string) Rol (string)



CREANDO APLICACIONES WPF

- 3) Agregar un nuevo proyecto denominado “Vistas”. Este proyecto contendrá todas las ventanas de la aplicación. Establecer este proyecto como **proyecto de inicio**.
- 4) Al iniciar la aplicación se mostrará una pantalla de bienvenida al sistema, la misma deberá permitir el ingreso de un nombre de usuario y una contraseña que se encontrarán en modo “hardcoded” en el método que se ejecuta al producirse el evento *click* del botón Ingresar (Ej: *btnclick_onClick*). Definir dos tipos de usuario: *Admin* y *Operador*.
- 5) Luego de validar el usuario y la contraseña de la ventana de login, se deberá llamar a una ventana donde se ofrecerán las distintas funciones del sistema de acuerdo al rol del usuario. Nota: el usuario Admin tendrá habilitada las funciones de gestión de Sectores y Tipos de Vehículo, y el Operador tendrá habilitada las funciones de gestión de Clientes y Gestión de Estacionamiento.
- 6) Desarrollar las funciones de ingreso de datos (Formulario que permitirá realizar el ABM de **Tipos de Vehículo y Clientes**). Nota: El formulario sólo permitirá capturar los valores de los textboxes (previo mensaje de confirmación) y almacenarlos en las propiedades de los objetos (por ej.: *oTipoVehiculo*, *oCliente*). Imprimir los valores de las propiedades de los objetos en un *MessageBox* para verificar que los datos ingresados se almacenaron en dichas propiedades. Posteriormente, en el TP N°3, estos datos capturados en variables se almacenarán en una base de datos.
- 7) Crear una ventana “Vehículos en Playa” como la que se muestra en la siguiente figura:



- En el método *Loaded* deshabilitar el sector “E7” cambiando su color de fondo a Rojo (ocupado).
- Al hacer click en un sector se deberá mostrar un mensaje según lo siguiente:

 UNIVERSIDAD NACIONAL DE JUJUY FACULTAD DE INGENIERÍA ANALISTA PROGRAMADOR UNIVERSITARIO (PLAN 2008)	LABORATORIO DE PROGRAMACIÓN ORIENTADA A OBJETOS II	TP N°
		1
		2023

CREANDO APLICACIONES WPF

- si está disponible (Verde) mostrar un mensaje “Sector Disponible. Registrar Entrada”.
 - si está ocupado (Rojo) mostrar un mensaje “Sector Ocupado. Registrar Salida”.
 - si está deshabilitado (Gris) mostrar un mensaje “Sector deshabilitado.”.
- Generar el manejador de eventos del botón salir de manera que se cierre la ventana.
- 8) Utilizar una imagen de fondo para el formulario principal.

Consideraciones:

- a) El trabajo podrá ser desarrollado en forma individual o grupal con un máximo de 7 integrantes por grupo para la comisión SSJ y de 4 integrantes por grupo para la comisión SP.
- b) El presente trabajo práctico **deberá ser entregado por un integrante de cada grupo en la fecha establecida como fecha de entrega** o de lo **contrario** el mismo se considerará como **desaprobado**.
- c) La entrega del trabajo deberá ser con ejercicios completos que compilen y ejecuten sin errores de compilación. Todo tipo de error que presente el trabajo deberá ser corregido previo a la fecha de entrega. Aquellos trabajos que se entreguen en forma incompleta o con errores se considerarán como **desaprobados**.
- d) La entrega del presente trabajo práctico deberá ser un archivo .rar que contenga el proyecto desarrollado en Visual Studio. Dicho archivo se deberá enviar al correo del docente responsable de la práctica de las comisiones de SP o SSJ respectivamente. Nota: Si el archivo no puede ser enviado, modificar la extensión del mismo, por ejemplo con un guion bajo: “.rar_”.