



Universidad de Margarita

Alma Mater del Caribe

Decanato de Ingeniería de Sistemas

Cátedra: Programación II

INFORME PROYECTO BIBLIOTECA UNIMAR

Integrantes:

Gustavo Gutiérrez C.I.: 27.547.080

Juan García C.I.: 30.707.234

Carlos García C.I.: 30.065.517

Docente:

César Requena

El Valle del Espíritu Santo, febrero del 2024

Para iniciar el proyecto se creó el Windows Form llamado “MenuPrincipal”. Esto debido a que al evaluar los requisitos solicitados para el proyecto se determinó que era el punto de partida correcto. En este formulario se crearon ocho botones cada uno con la funcionalidad de hacer que el programa abriera otros Windows Forms. También se le agregó un label como título representativo del proyecto.

Seguidamente, se creó un nuevo formulario que se le asignó el nombre de “frmAlumnoRegistro”. En este formulario, se le solicita al alumno que ingrese sus datos como el nombre, apellido, cédula y carrera mediante el uso de TextBoxes y un ComboBox para seleccionar la carrera. También se configuró un botón para que cumpla la función de guardar los datos ingresados en las cajas de textos y otro botón para cerrar el formulario actual y volver al menú principal.

Para poder guardar los datos ingresados por el usuario se creó una clase “Estudiante” cuyos atributos corresponden a los datos que debe ingresar el usuario para registrarse en el sistema. También se creó el método constructor de la clase y los métodos getters y el método ToString para retornar todos los datos del estudiante.

Luego de la creación de la clase “Estudiante” se decidió crear otro formulario llamado “frmRegistroLibro” con el propósito de pedirle al usuario que maneje el sistema la información del libro que desee ingresar en el sistema, es decir, del libro que se quiera añadir a la biblioteca. En este formulario nuevamente se utilizaron cajas de texto para el título del libro, los datos del autor y un ComboBox para el género del libro. También se añadió un botón que al ser presionado guardara la información ingresada.

Se creó una clase “Libro” para poderle asignar la información ingresada por el usuario a un objeto de dicha clase.

Para ambas clases, una vez registrado un alumno o un libro, el objeto correspondiente se guardaba en una lista dependiendo de su tipo, es decir, inicialmente se crearon dos listas, una para los datos de los estudiantes registrados y otra para los de los libros ingresados. Estas listas trabajan a través del uso de nodos. Para ello se creó la clase

“Nodo”. En esta clase se realizó una sobrecarga de constructores para que el nodo pueda guardar un valor diferente dependiendo de la lista en la que se use.

La decisión de usar listas se tomó debido a que estas permiten el manejo fácil de objetos, son estructuras dinámicas y es relativamente sencillo encontrar algún valor dentro de la lista cuando el tamaño no es muy grande.

Por supuesto, para cada lista se creó una clase. La clase “ListaEstudiantes” contenía sus respectivos atributos haciendo uso de nodos que almacenaban objetos de la clase “Estudiante” y la clase “ListaLibros” usaba nodos capaces de almacenar objetos de la clase “Libro”. Ambas clases de listas se crearon con sus respectivos métodos, en su mayoría similares en funcionalidad.

Después de crear las clases de listas, se decidió proceder con la creación de otro Windows Form para mostrar los libros disponibles. Dentro de este formulario, se colocó una caja de texto de solo lectura en donde se mostraría la lista de libros que han sido registrados y también se agregaron tres botones, uno para volver al menú principal, otro para mostrar todos los libros disponibles en la biblioteca y finalmente un botón para filtrar los libros según su género. Este último botón trabaja en conjunto a una ComboBox donde se selecciona el género que se quiere filtrar. Al presionar el botón de filtrar o el botón para mostrar todos los libros se muestran los libros guardados en la lista en la caja de texto.

Al llegar a este punto del proyecto se determinó que era necesario la utilización de archivos de texto que guardaran los datos de las listas, de manera que los datos de las listas se escribieran en sus respectivos archivos para que la información no se perdiera de manera definitiva al cerrar el programa. Mediante el uso de archivos de texto, se podía asegurar que los datos de los usuarios, los libros y el sistema en general se recuperara siempre que se volviera a iniciar el programa y así poder continuar con la manipulación de dichos datos.

A continuación, se creó la clase “Préstamo” cuyos atributos sería objetos de la clase “Estudiante”, “Libro” y la fecha en la cual se realice el préstamo correspondiente. Luego de crear esta clase se creó una clase que se le asignó el nombre de “ListaPrestamos”

con el propósito de guardar los datos de los préstamos realizados en la lista y también en un archivo de texto. A continuación, se creó el formulario donde se le piden al estudiante, su cédula y el título del libro que desea pedir prestado de la biblioteca.

Luego de esto, se creó el formulario llamado “frmPrestamosActivos” donde se colocó una caja de texto de lectura que al cargar el formulario muestra los datos de los préstamos activos junto con el estado actual del estudiante que realizó el préstamo. Seguidamente, se creó otro formulario con el nombre “frmDevoluciones”. En este formulario se le pide al usuario ingresar la cédula del estudiante cuyo préstamo se quiere retornar. Usando la cédula del estudiante se busca el préstamo en la lista de préstamos y se elimina el préstamo correspondiente.

Continuando con el programa, se decidió crear el formulario de renovación de préstamos. En este formulario se le pide la cédula al usuario, se busca el préstamo correspondiente en la lista de préstamos y se modifica la fecha para que no ocurra una sanción por días sin regresar el libro.

Finalmente, se creó el formulario de estudiantes sancionados. En este formulario se muestra únicamente una TextBox de solo lectura donde se muestran los datos de todos los estudiantes cuya fecha de realización de un préstamo se compara con la fecha actual del sistema y se determina que la diferencia es mayor a tres días, lo que equivale a que no se retornó el libro prestado dentro del tiempo estipulado.