



# ESCUELA POLITÉCNICA NACIONAL

## ESCUELA DE FORMACIÓN DE TECNÓLOGOS



### INFORME DEL PROYECTO

#### SEGUNDO BIMESTRE

#### TÍTULO DEL PROYECTO:

Programa CRUD para el Registro de Libros con Inicio de Sesión

**ASIGNATURA:** Programación

**PROFESOR:** Ing. Yadira Franco Rocha

**ESTUDIANTE:** Sebastian Vinicio Mañay Cevallos

Quito - Ecuador

2026

# ÍNDICE

<b>1. Introducción.....</b>	<b>3</b>
<b>2. Objetivos .....</b>	<b>3</b>
2.1 Objetivo general.....	3
2.2 Objetivos específicos.....	3
<b>3. Descripción del Proyecto.....</b>	<b>3</b>
<b>4. Motivación.....</b>	<b>4</b>
<b>5. Alcance del Proyecto.....</b>	<b>4</b>
<b>6. Desarrollo Técnico.....</b>	<b>4</b>
6.1 Arquitectura general del sistema.....	4
6.2 Estructuras y componentes utilizados.....	4
6.3 Tipos de datos empleados .....	5
<b>7. Funcionalidad del Sistema .....</b>	<b>5</b>
7.1 Crear registro.....	5
7.2 Mostrar registros.....	5
7.3 Actualizar registro.....	5
7.4 Eliminar registro .....	6
<b>8. Conclusiones.....</b>	<b>6</b>
<b>9. Recomendaciones y Trabajos Futuros.....</b>	<b>6</b>
<b>10. Repositorio del Proyecto .....</b>	<b>6</b>
<b>11. Bibliografía .....</b>	<b>7</b>

# **1. Introducción**

El presente proyecto consiste en una aplicación desarrollada en C++ con Qt, cuyo objetivo es la de gestionar un registro de libros mediante los procedimientos CRUD (crear, leer, actualizar y eliminar) y donde el sistema tiene un módulo de autenticación (login), que permite al usuario iniciar sesión para poder acceder a la aplicación principal. La información de usuarios y libros se almacena en archivos de texto (.txt), de forma que el guardado de datos no requiera una base de datos.

# **2. Objetivos**

## **2.1 Objetivo general**

Desarrollar una aplicación en C++ con interfaz gráfica que permita gestionar información mediante operaciones CRUD, almacenando los datos en archivos de texto (.txt).

## **2.2 Objetivos específicos**

- Implementar el registro de nuevos datos.
- Permitir la visualización de los registros almacenados.
- Actualizar y eliminar información existente.
- Manejar archivos planos para el guardado de datos.

# **3. Descripción del Proyecto**

El proyecto a desarrollar consiste en la creación de un programa, hecho en C++, con el uso de Qt, que permite tener una gestión de libros con procedimientos CRUD (crear, leer, actualizar y eliminar). El sistema también tiene un módulo de autenticación de usuarios (login), donde es necesario iniciar sesión o registrarse antes de entrar a la biblioteca. El programa permite a los usuarios llevar un control básico de libros. La información se almacena en archivos de texto (.txt), permitiendo el guardado de datos de libros sin necesidad de utilizar una base de datos.

El sistema maneja los siguientes campos para el registro de libros:

- ID del libro
- Título
- Autor
- Año de publicación

## 4. Motivación

La motivación para el desarrollo de este proyecto surge de la necesidad de aplicar de manera práctica los conocimientos adquiridos en la asignatura de Programación, especialmente en el uso del lenguaje C++, el manejo de archivos y la implementación de interfaces gráficas. El sistema desarrollado permite reforzar conceptos fundamentales como la lógica de programación, validación de datos y estructuración de un sistema CRUD, contribuyendo así al fortalecimiento de las habilidades técnicas.

## 5. Alcance del Proyecto

El proyecto permite realizar la gestión básica de libros mediante un sistema CRUD, incorporando autenticación de usuarios a través de un login y un registro previo. El sistema contempla las siguientes funcionalidades:

- Registro e inicio de sesión de usuarios.
- Almacenamiento de datos en archivos de texto.
- Creación, visualización, actualización y eliminación de libros.
- Interfaz gráfica amigable desarrollada con Qt.

No forma parte del alcance del proyecto la implementación de una base de datos, el uso de cifrado avanzado de contraseñas ni el acceso multiusuario en red. Estas características podrían considerarse como mejoras para versiones futuras.

## 6. Desarrollo Técnico

### 6.1 Arquitectura general del sistema

El programa fue desarrollada en C++ utilizando el framework Qt, organizado por módulos independientes que trabajan juntos. El sistema se divide en tres módulos principales:

- Login (MainWindow): encargado de la autenticación de usuarios.
- Registro de usuarios (registroLogin): permite la creación de nuevas cuentas.
- Gestión de biblioteca (biblioteca): módulo principal donde se ejecutan las operaciones CRUD sobre los libros.

La comunicación entre ventanas se realiza mediante la apertura y cierre controlado de formularios, garantizando una navegación adecuada dentro del sistema.

### 6.2 Estructuras y componentes utilizados

Para la interfaz gráfica se utilizaron los siguientes componentes de Qt:

- QMainWindow y QDialog para las ventanas.

- QLineEdit para el ingreso de datos.
- QTableWidget para mostrar los registros de libros.
- QMessageBox para mostrar mensajes informativos y de error.

Además, se utilizaron delegados personalizados (QStyledItemDelegate) para validar la entrada de datos numéricos en la tabla, como el ID y el año de publicación.

### **6.3 Tipos de datos empleados**

Durante el desarrollo del sistema se emplearon los siguientes tipos de datos:

- QString para el manejo de texto (usuarios, contraseñas, títulos, autores).
- int para valores numéricos como el año de publicación.
- bool para validar condiciones (por ejemplo, usuario encontrado o validaciones).
- QStringList para dividir y manejar los datos leídos desde los archivos de texto.

## **7. Funcionalidad del Sistema**

El sistema creado permite la administración de libros por medio de funciones CRUD, las cuales se llevan a cabo por medio de una interfaz gráfica intuitiva. Se detalla a continuación el funcionamiento de cada una de las opciones disponibles:

### **7.1 Crear registro**

Esta opción permite ingresar un nuevo libro al sistema. El usuario debe completar los campos correspondientes como el ID, título, autor y año de publicación. Antes de guardar el registro, el sistema realiza las siguientes validaciones:

- Verifica que todos los campos estén completos.
- Valida que el ID contenga únicamente valores numéricos.
- Controla que el año de publicación no sea mayor al año permitido.

Si las validaciones son correctas, el libro se agrega a la tabla de registros y se guarda automáticamente en el archivo libros.txt.

### **7.2 Mostrar registros**

Esta funcionalidad permite visualizar todos los libros almacenados en el sistema. Al acceder a la ventana principal, el sistema carga automáticamente los datos desde el archivo libros.txt y los muestra en una tabla (tableWidget). La tabla presenta los registros de forma ordenada, permitiendo al usuario seleccionar cualquier libro para su consulta o edición.

### **7.3 Actualizar registro**

La opción de actualizar permite modificar la información de un libro previamente registrado. Para realizar esta acción, el usuario debe seleccionar un registro de la tabla, lo que carga automáticamente los datos en los campos de edición. Una vez realizados los cambios, el

sistema actualiza la información tanto en la tabla como en el archivo libros.txt, garantizando que los datos se mantengan actualizados.

#### 7.4 Eliminar registro

Esta opción permite eliminar un libro del sistema. El usuario debe seleccionar el registro que desea eliminar y presionar el botón correspondiente. El sistema elimina el registro de la tabla y actualiza el archivo libros.txt, asegurando que el libro eliminado no vuelva a aparecer en ejecuciones posteriores de la aplicación.

## 8. Conclusiones

- Se desarrolló exitosamente un programa en C++ con Qt que cumple con las operaciones CRUD (crear, leer, actualizar y eliminar) para la gestión de libros.
- El sistema implementa correctamente un módulo de autenticación, permitiendo el registro e inicio de sesión de usuarios antes de acceder a la funcionalidad principal.
- El uso de archivos de texto (.txt) permitió almacenar la información de manera persistente sin necesidad de una base de datos.
- Se aplicaron validaciones que garantizan la integridad de los datos ingresados por el usuario.
- El proyecto contribuyó al fortalecimiento de conocimientos en programación estructurada, manejo de archivos y desarrollo de interfaces gráficas.

## 9. Recomendaciones y Trabajos Futuros

- Implementar una base de datos para mejorar la seguridad y la capacidad de crecimiento del sistema.
- Incorporar cifrado de contraseñas para aumentar la protección de la información de los usuarios.
- Agregar diferentes roles de usuario, como administrador y usuario estándar.
- Mejorar la interfaz gráfica para ofrecer una experiencia de usuario más intuitiva.
- Implementar validaciones adicionales y manejo avanzado de errores.
- Ampliar el sistema para permitir la gestión de préstamos y devoluciones de libros.

## 10. Repositorio del Proyecto

El proyecto se encuentra disponible en el siguiente repositorio de GitHub:

<https://github.com/sebassm0528-dev/Proyecto-de-Programaci-n.git>

## 11. Bibliografía

TutorialsPoint. (s.f.). *C++ File Streams*. Recuperado de  
[https://www.tutorialspoint.com/cplusplus/cpp\\_files\\_streams.htm](https://www.tutorialspoint.com/cplusplus/cpp_files_streams.htm)

Bodnar, J. (2023). *Qt5 tutorial – Learn C++ GUI programming with Qt*. ZetCode.  
Recuperado de <https://zetcode.com/gui/qt5/>

*Basic Qt programming tutorial (español)*. Qt Wiki. Recuperado de  
[https://wiki.qt.io/Basic\\_Qt\\_Programming\\_Tutorial/es](https://wiki.qt.io/Basic_Qt_Programming_Tutorial/es)

*Qt Examples and Tutorials*. Felgo Documentation. Recuperado de  
<https://felgo.com/doc/qt/qtexamplesandtutorials/>

Qt Project. (2025). *Getting Started Programming with Qt Widgets*. Qt Documentation.  
<https://doc.qt.io/qt-6/widgets-getting-started.html>