

# **Taller de Programación 2021**

## **Trabajo Práctico Final**

## Introducción

El trabajo práctico final consta de la realización de una aplicación de **gestión de préstamos de una biblioteca**.

La aplicación deberá ser capaz de realizar **acciones** para **gestionar los préstamos, como ser administración de usuarios, alta libros, registro de préstamos y devoluciones**. Además tendrá **funcionalidades de reportes y notificaciones para facilitar la gestión de los mismos**.

## Requerimientos funcionales

### Alta de libros

La aplicación deberá ser capaz de dar de **alta libros** para que los ejemplares estén disponibles para el préstamo, para ello tiene que **acceder a un catálogo online** con el fin de obtener los metadatos del libro. De estos **metadatos** solo deben ser persistidos lo que considere el **alumno** sean relevantes

Utilizaremos el **catálogo online de Open Library** (<https://openlibrary.org/>) que es un sitio de internet el cual posee datos gratuitos de libros, publicaciones, revistas, etc.

Este sitio permite además acceder a los datos mediante una **API publica** de fácil utilización devolviendo los datos en **formato JSON** (JavaScript Object Notation). A continuación se encuentra el link de acceso a la **documentación de la API**:

<https://openlibrary.org/developers/api>

Utilizaremos esta funcionalidad con el fin de **obtener la información de los libros** para ser incorporadas a la aplicación. Un ejemplo de cómo **acceder a los datos** de estos libros desde una aplicación C# lo podemos encontrar en el siguiente repositorio de **GitHub**

<https://github.com/utn-frcu-isi-tdp/openlibrary-test-client>

Un **usuario administrativo** de la aplicación deberá poder la realizar **importación y actualización de datos de los libros**.

## Alta de usuarios

La aplicación deberá poder dar de alta usuarios para que puedan ser utilizados en la misma. Los usuarios pueden ser de dos tipos, administrador o simple. Un usuario de tipo administrador será un usuario que pueda interactuar con la aplicación con el fin de realizar las diferentes acciones para gestionar los préstamos. En cambio un usuario o usuario simple no puede interactuar en el sistema y su sola presencia en el mismo es debido a que será objetivo los préstamos.

## Préstamos y devoluciones de ejemplares

La aplicación deberá ser capaz de registrar un préstamo de un ejemplar de un libro a un usuario, estableciendo datos acerca del préstamo como ser fecha de préstamo, fecha de devolución, etc. Además la aplicación deberá poder registrar también la devolución de los ejemplares prestados.

## Reporte de préstamos próximos a vencer

La aplicación deberá ser capaz de generar un reporte de préstamos próximos a vencer. El objetivo de este reporte es que un usuario administrador pueda acceder a él para visualizar los ejemplares a ser devueltos en los próximos 7 días.

## Scoring de prestamos

La aplicación deberá mantener un scoring de préstamos de manera de establecer premios y/castigos en el uso de los servicios de la aplicación.

Las puntuaciones serán las siguientes:

- Devolver un ejemplar en malas condiciones: - 10 puntos
- Devolver un ejemplar fuera del límite de tiempo: - 2 puntos por día hábil de mora.
- Devolver un ejemplar de manera correcta: + 5 puntos.

El préstamo de un ejemplar tiene una duración base de 5 días hábiles. Pero mediante el scoring un usuario si dispone de un buen puntaje puede extender el préstamo 1 día cada 5 puntos ganados en el scoring. Esto se pueda extender hasta un máximo de 15 días hábiles.

## Notificación de vencimiento de préstamo

Cuando el préstamo se encuentre entre los dos días hábiles a ser vencido, se le enviará una notificación por mail al usuario avisando de esta situación. El envío de notificaciones a los usuarios deberá quedar registradas en la aplicación.

## Requerimientos no funcionales

- La aplicación deberá ser robusta ante cualquier tipo de errores.
- La aplicación deberá ser fácil de usar e intuitiva.
- La interfaz del usuario deberá ser consistente y no deberá tener errores de interacción ni de visualización de información.
- La aplicación deberá ser desarrollado sobre la plataforma .NET y en lenguaje C#.
- El programa deberá tener una interfaz gráfica, se sugiere el uso de WinForms, integrando los conocimientos y técnicas adquiridos durante la cátedra. La incorporación de conocimientos no adquiridos durante la cátedra serán también bienvenidos.
- El programa deberá persistir las configuraciones y otros datos en una Base de Datos relacional, utilizando el gestor es a elección del alumno. Se espera que en un futuro puedan configurarse persistencia en distintos gestores de Bases de Datos u otras formas de persistencia (como por ejemplo archivos, Bases de Datos No-SQL, entre otras), por lo que el software debe estar preparado para ello.
- La aplicación deberá contener una bitácora de monitoreo (archivo de log), que permita hacer diagnósticos ante la ocurrencia de errores.
- **Se espera que en un futuro la aplicación pueda acceder a diferentes catálogos online de libros. El sistema deberá estar diseñado para que los cambios a realizar para incorporar catálogos sea el menor posible.**
- **Se espera además incorporar en un futuro otros mecanismos de notificación de vencimiento de préstamos, por lo que se debe tener esto en cuenta en el desarrollo de la aplicación.**
- El código fuente deberá estar correctamente comentado y documentado con los formatos correspondientes.

## Entregables

Los entregables del trabajo final son:

- Código ejecutable.
- Código fuente.
- Diagrama de clases y paquetes en formato digital de alguna herramienta UML conocida.
- Guía de instalación y uso del programa.

**Importante: Se debe entregar todos los mencionados elementos grabados en un CD cuando se defienda el trabajo.**

## **Evaluación**

Se evaluará el uso de buenas técnicas de desarrollo y documentación, legibilidad del código, utilización correcta del lenguaje y finalmente la resolución empleada para satisfacer los requerimientos.

Se deberá realizar una defensa grupal e individual del trabajo en el momento de la entrega, en donde los alumnos deberán explicar la solución empleada y responder a preguntas de forma individual del equipo docente.