

# Laboratorio de Programación II

## BiblÁnima v2.0

Prof: Juan Bordagorry  
Prof: Lucía Cárcamo  
Prof: Silvina Menéndez

August 25, 2019

## 1 Introducción

Este documento pretende presentar el proyecto final de Programación II del curso 5to año TIC. BiblÁnima surge de la necesidad de tener un gestor de bibliotecas online, capaz de gestionar usuarios y libros, así como también gestionar notificaciones relacionados con prestamos.

## 2 Objetivos

La presente tarea tiene los siguientes objetivos de aprendizaje por parte del estudiante:

1. Aprender las construcciones básicas del lenguaje Java.
2. Realizar una interfaz gráfica (GUI) en Swing.
3. Utilizar herramientas avanzadas para el desarrollo de software: entornos de desarrollo integrados (IDEs).
4. Aplicar prácticas usuales en el desarrollo de software:
  - Trabajo en equipos.
  - Planificación de actividades.
  - Registro de actividades.

## 3 BiblÁnima

Se desea implementar una aplicación en java, cuya función esté enfocada en la gestión de una biblioteca. Para esto, el sistema debe reconocer a los Usuarios, particularmente busca conocer el nombre, el apellido, el mail, la CI, un password y un id UNICO.

Sabemos que los Usuarios se distinguen en tres tipos: los Estudiantes, los Profesores y los Bibliotecarios. De los Estudiantes se sabe que pueden retirar como máximo 2 libros en paralelo.

De los Profesores se sabe a que orientación pertenecen (puede ocurrir que pertenezcan a las 2).

Los Bibliotecarios serán los administradores del sistema, siendo estos los que darán de alta y baja a los usuarios, los libros y los prestamos.

Por otra parte, el sistema debe mantener una colección de todos los Libros del sistema, incluyendo a todos aquellos Libros que se agreguen en el futuro. De los Libros, nos interesa saber los autores, el año de publicación, el numero de edición, la editorial, una breve descripción, la cantidad de ejemplares, saber si hay un ejemplar disponible para retirar, el código ISBN, código del libro, el genero y un link a la imagen perteneciente al cover del libro.

Una funcionalidad vital, es la posibilidad de generar Prestamos de libros a los distintos Usuarios del sistema. De cada Préstamo, es necesario conocer de quien es, fecha de solicitado, fecha de devolución y que libro retiró.

El sistema debe mantener una colección de Notificaciones generadas por el préstamo de un libro. Cada Notificación debe almacenar de quien es, el mail, la fecha de solicitado y la fecha de devolución.

La aplicación debe poder persistir el estado global del sistema, siendo este la totalidad de los libros, los usuarios, los prestamos y las notificaciones.

## 4 ISBN Search

Se va a proveer un FICHERO java, el cual se conectará con un servidor y podrá extraer la información de un libro solicitado.

Al solicitar la información de un libro, se retornará un String con el siguiente formato:

*"NombreLibro,Autor1:Autor2:Autor3:...:AutorN,  
AñoPublicación,AñoEdición,Editorial,LinkImg,Descripción"*

## 5 UML

En el siguiente link se encuentra el UML a implementar:

<https://drive.google.com/file/d/1twoGqUM9rxvvOw0uvfdz2t8xFCm-qomR/view?usp=sharing>

## 6 Actores

Se distinguen 3 actores principales: Bibliotecario, Estudiante, Profesor.

## 7 Casos de Uso Mínimos

El sistema deberá contemplar estos casos de usos para el correcto funcionamiento del software.

**Caso de Uso:** Alta Usuario

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando un Bibliotecario quiere dar de alta a un nuevo Usuario en el sistema. Para ello se indica el nombre, apellido, mail y un password. Luego, selecciona que tipo de usuario es (Bibliotecario, Estudiante o Profesor y se carga la información correspondiente. Si el mail corresponde a uno ya existente, el sistema deberá alertar de esto y pedirle al usuario que ingrese uno nuevo. Finalmente el sistema da de alta a un Usuario y los persiste en la base de datos.

**Caso de Uso:** Consulta Usuario

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea consultar el perfil de un usuario. Para ello el sistema muestra la lista de todos los usuarios y el administrador elige uno. Luego, el sistema muestra todos los datos básicos del usuario. Además, se muestra toda la información relevante a sus prestamos, diferenciando aquellos que están activos, del histórico de prestamos.

**Caso de Uso:** Buscar Usuario

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea consultar el perfil de un usuario. Para ello el sistema mantiene un campo para introducir la CI del usuario y el sistema busca el usuario con esa CI asignada. Luego, el sistema muestra todos los datos básicos del usuario. Además, se muestra toda la información relevante a sus prestamos, diferenciando aquellos que están activos, del histórico de prestamos.

**Caso de Uso:** Modificar Datos Usuario

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea modificar el perfil de un usuario. Para ello el sistema muestra la lista de todos los usuarios y el administrador elige uno. Luego, el sistema muestra todos los datos básicos del usuario. El administrador puede editar todos los datos básicos, menos la fecha de nacimiento. Cuando termina la edición, el sistema actualiza los datos del usuario.

**Caso de Uso:** Listar Usuarios Existentes

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea ver un listado de todos los usuarios del sistema. El administrador indica esto y el sistema lista los mails de todos los usuarios.

**Caso de Uso:** Alta Préstamos

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea dar de alta un Préstamo. Para esto, el sistema debe listar todos los usuarios del sistema, y luego seleccionar el usuario que solicitó un préstamo. Junto con esto, poder seleccionar fechas validas, tanto de préstamo como de devolución y el resto de los datos pertinentes a los préstamos. El sistema debe persistir el préstamo.

**Caso de Uso:** Consultar Préstamo

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea consultar los préstamos que un Usuario tiene. Para esto el sistema debe listar todos los Usuarios para que luego el Bibliotecario pueda seleccionar uno, y el sistema lista la información de los préstamos activos y el histórico.

**Caso de Uso:** Listar Préstamos

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea consultar todos los préstamos del sistema. Para esto, el sistema debe desplegar todos los préstamos con la información pertinente.

**Caso de Uso:** Alta Notificación

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea dar de alta un Préstamo. Para esto, el sistema debe generar tres Notificaciones, la primera es una notificación que detalla que se generó un préstamo para el usuario solicitante, otra para el día previo a la devolución del libro que solicitó y la tercera, corresponde cuando un usuario no devolvió el libro en fecha y debe generarse una alerta por esto.

**Caso de Uso:** Alta Libro

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea dar de alta un Libro. Para esto, el Bibliotecario pasa el código ISBN, y luego el sistema da de alta un libro con el contenido pertinente. Luego del alta, el sistema persiste el libro.

**Caso de Uso:** Listar Libros

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea listar todos los libros del sistema. Para esto, el sistema debe listar todos los libros que se encuentren en el sistema, detallando el nombre, el autor, la edición, la cantidad de ejemplares disponibles y el código del mismo.

**Caso de Uso:** Dar de baja un Préstamo

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea dar de baja un préstamo ya que el usuario solicitante devolvió el libro asociado al préstamo, para esto, lista los usuarios del sistema y selecciona el necesario (o bien, busca el mismo), luego lista sus préstamos y luego lo da de baja.

## 8 Casos de Uso Adicionales

**Caso de Uso:** Alta Datos

**Actor:** Bibliotecario

**Descripción:** El caso de uso comienza cuando el Bibliotecario desea cargar los datos en el sistema desde un CSV. Para esto, el Bibliotecario tendrá un archivo CSV para los libros, uno para los usuarios y uno para los préstamos, donde se tendrán que cargar los datos en dichos archivos y persistirlos. Se debe mantener la integridad y la coherencia del sistema.

## 9 Plazo de Entregas

El día **lunes 2/9**, se debe entregar en programación un informe que describa la correctitud y consistencia del UML entregado, así como también posibles mejoras que se le pueden hacer y posibles errores o problemas que puedan surgir.

El resto de los lunes se van a dedicar a la implementación del sistema, para esto se realizará una entrega intermedia para evaluar el progreso de los grupos. La fecha esta por confirmarse.

## 10 Forma de trabajo

Se formarán grupos de 2 o 3 personas. Cada integrante deberá implementar el código de forma individual.

## 11 Grupos de Trabajo:

A confirmar.

## 12 Evaluación

Durante la fecha de entregas, el grupo pasará a la etapa de defensa. En la misma se probarán ciertas funcionalidades de la aplicación y la misma deberá correr sin errores.

A su vez, se tendrá una evaluación individual oral, en la que se preguntarán cosas puntuales acerca de la implementación de la aplicación.

## 13 Aclaraciones

El incumplimiento de la sección 4. referente a la implementación individual del código por parte de cada alumno, impactará de forma negativa en la calificación final.

Será de utilidad para la interfaz de Swing, utilizar elementos básicos como los siguientes:

- JList - <https://docs.oracle.com/javase/8/docs/api/javax/swing/JList.html>
- Menu - <https://docs.oracle.com/javase/tutorial/uiswing/components/menu.html>
- JPanel - <https://docs.oracle.com/javase/7/docs/api/javax/swing/JPanel.html>
- JFrame - <https://docs.oracle.com/javase/7/docs/api/javax/swing/JFrame.html>