

Programación II

Trabajo Práctico Final

Requisitos generales:

- El trabajo deberá realizarse en grupos de **3 o 4 integrantes**.
- La defensa del trabajo será **individual** (cada uno de los integrantes del grupo deberá tener conocimiento de cómo funciona todo el código, y ser capaz de explicar las decisiones de diseño tomadas).
- Cada exposición durará máximo **20 minutos** y consta de una **explicación del sistema** y **preguntas a cada integrante** del grupo. La exposición se tendrá en cuenta para la evaluación final del trabajo práctico.
- Deberán implementarse todos los conceptos aprendidos durante el desarrollo de las clases del cuatrimestre siendo de carácter **OBLIGATORIO** lo siguiente:
 - Realización y entrega de un **Diagrama UML** en el cual quede diseñado la estructura del sistema.
 - Utilización de **los 4 pilares de la POO** (Herencia, Polimorfismo, Abstracción y Encapsulamiento).
 - Creación de un **mínimo de 5 clases** en su sistema (Sin contar la clase Main, interfaces ni excepciones).
 - Crear mínimamente **1 Clase Abstracta** y **1 interfaz** para implementar.
 - Utilización de Listas/Arreglos/Colecciones. **OBLIGATORIO el uso de al menos una colección de tipo List, otra de tipo Set**

y otra de tipo Map (a elección del grupo, y justificando en cada caso el por qué de ese tipo de colección para ese tipo de datos)

- Aplicación del concepto de **manejo de errores** con al menos 2 clases de excepción personalizadas.
 - Manejo de **JSON** para la persistencia de datos, utilizando **json.org**.
 - Debe manejarse el concepto de **baja lógica**
 - El proyecto ya debe tener **cargados en JSON al menos 15 datos** de c/u de los que manejen en su sistema, para poder probar c/u de las funcionalidades del sistema.
 - El código fuente debe estar correctamente documentado/comentado.
 - Deberá haber **ABMCL de cada una de las clases que maneje el sistema y que así lo requieran (como mínimo, de cuatro de ellas si es que no todas requieren ABMCL**, ya que, por ejemplo, las clases abstractas no requieren ABMCL).
 - Deben existir **DOS tipos distintos de acceso y manejo del sistema para distintos tipos de usuarios:** administrador y usuario).
- Será de carácter **OPCIONAL** lo siguiente:
 - Creación y uso de una **Clase Genérica**.
 - Utilización de la herramienta **GIT y GITHUB** para el versionado de código y el trabajo colaborativo. Todos los integrantes del grupo deben commitear sus cambios a medida que van subiendo funcionalidad o van arreglando código. No puede haber un solo commit con toda la funcionalidad o que un solo integrante haya subido todo el código. Todos deben de participar.
 - Implementación de una **interfaz gráfica de usuario** (GUI) con la utilización de **JavaFX**.

Temática del proyecto:

- El tema podrá ser elegido por cada grupo, **respetando las condiciones mínimas antes expuestas (AMBCL de cada una de las cinco clases, manejo del sistema para distintos tipos de usuarios, etc)**
- **Deberá seguirse como orientación las opciones de temas brindados por la cátedra en cuanto a las funcionalidades mínimas que deberá tener el proyecto y la complejidad mínima que deberá tener el mismo.**
- También podrá optarse por alguno de los dos temas que siguen a continuación como propuestas de la cátedra.

Opción 1: Sistema de administración de un hotel.

El sistema deberá organizar un hotel, administrando sus habitaciones y los pasajeros/clientes del hotel.

El sistema deberá permitir realizar reservas de habitaciones, hacer el check-in y check-out de los pasajeros, listar las habitaciones actualmente ocupadas y los datos de los ocupantes, las habitaciones disponibles, y las habitaciones que no estén disponibles por algún motivo (limpieza, reparación, desinfección, etc. Detallar el motivo).

Deberá informar si es posible ocupar una habitación en un período determinado (consultando la ocupación y las reservas).

Para los pasajeros se pide un informe que incluya nombre, DNI, origen, domicilio de origen. Opcionalmente se puede dar información sobre la historia del pasajero en el hotel (detalles de los períodos en los que estuvo alojado, la habitación que ocupó, etc).

- **Check-in:** Es el proceso de registro de un pasajero en el hotel. Se realiza cuando el pasajero llega al hotel para tomar posesión de la habitación.
- **Check-out:** Es cuando el pasajero deja la habitación.
- **Pasajero:** Es la persona que ocupa físicamente la habitación.
- **Ocupación:** La ocupación es cuando el pasajero está pagando por la habitación. Normalmente comienza cuando el pasajero toma posesión de la misma, durante el check-in, y termina cuando el pasajero abandona la misma en el check-out.
- **Reserva:** Una reserva consiste en un período de tiempo en el que la habitación será ocupada por un pasajero. Una habitación reservada no puede ser ocupada, salvo por el pasajero que la reservó, a no ser que se cancele la reserva.

Tipos de usuario del sistema:

Se prevén por lo menos 3 tipos de usuario

- **Administrador:** Es el encargado de las funciones administrativas del sistema. Dentro de sus funciones está la realización del backup de la información, la creación de otros usuarios, la asignación de permisos a usuarios, etc.
- **Conserje o Recepcionista:** Es la persona que atiende a los pasajeros, realiza los check-ins y check-outs, las reservas, etc. Debe poder conocer el estado de cada habitación en todo momento, y tener acceso a la carga de datos de los pasajeros del sistema.
- **Pasajero:** Es la persona que ocupa la habitación. Podrían preverse accesos al sistema de los pasajeros para realizar reservas remotas de habitaciones, o para realizar consumos en las habitaciones ya ocupadas.

Opción 2: Sistema de administración de una biblioteca.

El sistema deberá organizar una biblioteca, administrando sus libros y los usuarios que la utilizan.

El sistema deberá permitir el préstamo de libros, la devolución, y la realización de reservas de libros.

Deberá mantener un registro de los libros actualmente prestados y los datos de los usuarios que los han solicitado.

Para los usuarios se pide un informe que incluya el nombre del usuario, su número de identificación (DNI), dirección y datos de contacto. Opcionalmente, se puede incluir información sobre el historial de préstamos de cada usuario, como los libros que han solicitado y las fechas de préstamo y devolución.

El sistema debe permitir a los usuarios buscar libros por título, autor o género, y verificar la disponibilidad de un libro en particular.

Asimismo, deberá informar si es posible solicitar un préstamo de un libro en un periodo determinado, teniendo en cuenta las fechas de préstamo y devolución de otros usuarios.

Tipos de usuario del sistema: se prevén al menos 3 tipos de usuario:

- **Administrador:** Es el encargado de las funciones administrativas del sistema. Dentro de sus funciones está la realización del backup de la información, la creación de otros usuarios, la asignación de permisos a usuarios, etc. El administrador también tiene acceso a la base de datos de libros y usuarios para mantener el catálogo actualizado.
- **Bibliotecario:** Es la persona que atiende a los usuarios, realiza los préstamos, devoluciones y reservas de libros. Debe poder conocer el estado de cada libro en todo momento y tener acceso a la carga de datos de los usuarios del sistema.
- **Usuario:** Es la persona que utiliza la biblioteca. Los usuarios pueden realizar búsquedas en el catálogo de libros, solicitar préstamos y devolver

libros prestados. También pueden hacer reservas de libros que estén actualmente prestados.