

En la unidad 3 de nuestro curso hemos aplicado mecanismos de reutilización como la **herencia**, el **polimorfismo** y el uso de **interfaces**. Estas técnicas enriquecen el comportamiento de las clases modeladas como parte de una solución en el diagrama de clases.

En la unidad 4, aprenderemos a utilizar el concepto de **matriz** como elemento de modelaje, pudiendo así agrupar elementos del modelo de la solución en una estructura contenedora de dos dimensiones de tamaño fijo; y hemos empleado **estructuras contenedoras lineales de tamaño variable** como elementos de modelado que permiten manejar una secuencia de objetos.

Esta tarea integradora presenta una actividad en la cual se requiere aplicar todos los conocimientos adquiridos hasta el momento. Por tanto, esta tarea es un instrumento para verificar el cumplimiento de los objetivos planteados para las unidades 3 y 4 descritos en el programa del curso. Para llevar a cabo este ejercicio es necesario realizar las actividades listadas a continuación:

Actividades

Lleve a cabo las siguientes actividades de cada una de las etapas de desarrollo de software:

1. **Análisis del problema.** [Tabla de especificación y Lista de requerimientos funcionales](#) en el formato visto en la clase de Ingeniería de Software I.
2. **Diseño de la solución.**
 - a. Elabore un **diagrama de clases** que modele la solución del problema de acuerdo con las buenas prácticas y los patrones de diseño revisados hasta el momento en el curso. Su diagrama debe incluir el paquete modelo (model) y el de interfaz de usuario (ui).
 - b. **Trazabilidad del Análisis al Diseño.** Una [tabla a tres columnas](#) en la que se relaciona cada requerimiento con las clases y métodos que permiten satisfacer dicho requerimiento.
 - c. Recuerde que TODOS los artefactos generados en la fase de diseño deben **escribirse en inglés** y ubicarse en la carpeta doc de su proyecto
3. **Implementación en Java.** Incluya en la implementación, los comentarios descriptivos sobre los atributos y métodos de cada clase. Recuerde que TODOS los artefactos generados en la fase de implementación deben **escribirse en inglés**.
4. **Documentación en javadoc** (Debe entregarse el javadoc generado y ubicarlo en la carpeta API dentro de la carpeta doc).
5. **Usar GitHub como repositorio de código fuente y documentación** utilizando la estructura de carpetas aprendida en clase. En la entrega se debe adjuntar un enlace URL al repositorio. Se **debe poder identificar commits en diferentes fechas**.
6. **Elaborar un video con la ejecución del programa.** En la entrega se debe adjuntar un enlace al video.
7. Subir a Intu todos los entregables.

Recuerde que puede encontrar la **Rúbrica de la tarea integradora** en el siguiente [enlace](#).

Nota:

- Usted debe entregar la URL de su repositorio GitHub donde se deben encontrar los archivos de codificación en sus respectivos paquetes.
- Tenga en cuenta que su repositorio GitHub debe presentar una estructura base como por ejemplo:

`ReadX/
src/
bin/
doc/`

- Dentro de los directorios `src/` y `bin/` estarán presentes estos directorios (representando cada uno de sus paquetes):

`ui/
model/`

- El directorio `src` (source code) contiene sus clases `.java` dentro del directorio `ui/` y `model/`. Por otro lado, el directorio `bin` (binary files) contiene los archivos `.class` en el directorio `ui/` y `model/`. El directorio `doc` tendrá toda la documentación de análisis y diseño
- Su código debería compilar de acuerdo con lo explicado en la diapositiva 15 de esta presentación: <http://tinyurl.com/y3bd9bg2>

Enunciado

Debido a su vasta experiencia como programador, usted ha sido contratado por ReadX, un conglomerado egipcio, para desarrollar un prototipo de software que les permita gestionar su negocio de publicaciones a nivel mundial. A continuación se presentan las funcionalidades a desarrollar:

Gestión de productos bibliográficos: El modelo de negocio de la empresa se basa, por ahora, en la venta de dos tipos de productos bibliográficos: libros y revistas.

Cada libro tiene un identificador único (3 caracteres hexadecimales), un nombre, un número de páginas, una reseña corta, una fecha de publicación, un género, una URL que lleva a un repositorio con la portada del libro, el valor de venta (en dólares), el número de ejemplares vendidos y el acumulado de páginas leídas. Los posibles géneros existentes en el prototipo son: Ciencia Ficción, Fantasía y Novela Histórica.

Cada revista tiene un identificador único (3 caracteres alfanuméricos), un nombre, un número de páginas, una fecha de publicación, una categoría, una URL que lleva a un repositorio con la portada de la revista, el valor de la suscripción (en dólares), la periodicidad de emisión, el número de suscripciones activas y el acumulado de páginas leídas. Las posibles categorías existentes en el prototipo son: Variedades, Diseño y Científica.

ReadX anticipa cambios en la industria de la publicación por lo que le solicita que el diseño del prototipo contemple la creación futura de otros tipos de productos bibliográficos.

Ejemplo de Libro		Ejemplo de Revista	
Identificador:	4AF	Identificador:	Z1T
Nombre:	A Game of Thrones	Nombre:	Vogue
Número de páginas:	694	Número de páginas:	40
Reseña:	Join adventurers across the seven kingdoms	Fecha de publicación:	01/06/2021
Fecha de publicación:	01/08/1996	Categoría:	Variedades
Género:	Fantasía	URL:	LMV2021.jpg
URL:	AGOT.png	Valor suscripción:	\$ 4,99
Valor de venta:	\$ 19,99 USD	Periodicidad de emisión:	Mensual
# de ejemplares vendidos:	212.018.485	# de suscripciones activas:	122.867
# de páginas leídas:	4.888.789.521.362	# de páginas leídas:	7.533.456.212

El prototipo debe permitir **registrar, borrar y modificar** productos bibliográficos en el sistema.

Gestión de usuarios: El prototipo debe contar, por ahora, con dos tipos de usuario: regular y premium. El usuario Regular podrá comprar 5 libros, suscribirse hasta a 2 revistas y durante su uso de la plataforma, se le presentarán anuncios publicitarios. Por su parte, el usuario Premium podrá adquirir libros y suscribirse a tantas revistas como desee y pueda pagar. Para realizar el registro de usuarios en la plataforma se necesitan los siguientes datos: nombre, cédula y fecha de vinculación. ReadX anticipa la evolución y crecimiento dinámico de su plataforma por lo que le solicita que el diseño del prototipo contemple la creación futura de otros tipos de usuario.

Compra de libros o suscripción a revistas: Cuando los usuarios adquieren un libro o se suscriben a una revista, se debe almacenar la fecha de la operación y el monto pagado en una factura, así como también actualizarse el número de ejemplares vendidos o las suscripciones activas, según corresponda al tipo de producto bibliográfico. Adicionalmente, se debe permitir al usuario cancelar la suscripción a una revista en cualquier momento.

Presentación de la Biblioteca: ReadX requiere que su prototipo cuente con un menú que le permita al usuario visualizar su colección de productos bibliográficos, de manera rápida, semejando a una biblioteca. La Biblioteca deberá representarse a través de matrices 5x5 que gráficamente presenten el código de los productos bibliográficos asociados a la cuenta del usuario. Los productos deberán estar ordenados por fecha de publicación, de la más antigua a la más nueva. Conforme el usuario

amplíe su colección, se deberá permitir al usuario navegar en la misma (anterior o siguiente página). El usuario deberá poder seleccionar un producto bibliográfico ya sea por su coordenada x,y en la matriz presentada o el identificador de 3 caracteres del producto con el fin de iniciar una sesión de lectura. A continuación se presenta un ejemplo de interfaz gráfica de la Biblioteca.

Biblioteca de Juan

	0	1	2	3	4
0	4AF	2BA	Z1T	PQ4	___
1	___	___	___	___	___
2	___	___	___	___	___
3	___	___	___	___	___
4	___	___	___	___	___

Digite la coordenada x,y o el código correspondiente del producto bibliográfico para iniciar una sesión de lectura
 Digite A para ir a la anterior página
 Digite S para ir a la siguiente página
 Digite E para salir

Simulación de una sesión de lectura: La simulación de la sesión de lectura no es más que presentar por consola el nombre del producto bibliográfico, la página actual que se está leyendo y unas opciones de navegación para leer la página anterior, la página siguiente y para volver a la Biblioteca. A continuación se presenta un ejemplo de la simulación.

Sesión de Lectura en progreso:

Leyendo: Las mil y una noches

Leyendo página 4 de 225

Digite A para ir a la anterior página
 Digite S para ir a la siguiente página
 Digite B para volver a la Biblioteca

Cada página leída en la simulación incrementa el número de páginas leídas del producto bibliográfico correspondiente en la plataforma. Finalmente, se debe tener en cuenta que durante la sesión de lectura, para el caso de los usuarios regulares, se deben presentar anuncios publicitarios¹ en dos momentos: al iniciar la sesión de lectura y luego de cada 20 páginas leídas de un libro o 5 páginas leídas de una revista.

Generación de reportes: con el fin de generar contenidos dirigidos, ReadX solicita que el prototipo pueda generar los siguientes reportes en tiempo real:

1. Para cada tipo de producto bibliográfico, libro y revista, informar el acumulado total de páginas leídas en toda la plataforma (tipo de producto y número de páginas leídas).
2. Informar el género de libro y categoría de revista más leídas para toda la plataforma (nombre del género o categoría y número de páginas leídas).

¹ El prototipo debe contener los siguientes anuncios publicitarios, que se presentarán de manera aleatoria:

- ¡Suscríbete al Combo Plus y llévate Disney+ y Star+ a un precio increíble!
- Ahora tus mascotas tienen una app favorita: Laika. Los mejores productos para tu peludito.
- ¡Estamos de aniversario! Visita tu Éxito más cercano y sorpréndete con las mejores ofertas.

3. Informar el Top 5 de libros y el Top 5 de revistas más leídas en la plataforma (nombre del libro, nombre del género o categoría y número de páginas leídas).
4. De cada género, informar el número de libros vendidos y el valor total de ventas (\$).
5. De cada categoría, informar el número de suscripciones activas y el valor total pagado por suscripciones.

Gestión de pruebas: Con el fin de facilitar al usuario final la realización de pruebas, el prototipo debe contener al menos una funcionalidad que se encargue de generar automáticamente en el sistema, cuando el usuario lo solicite, al menos un objeto para cada tipo de usuario y producto bibliográfico.

En resumen, el prototipo a desarrollar debe ofrecer las siguientes funcionalidades:

1. Registrar usuarios, regulares y premium.
2. Gestionar productos bibliográficos: registrar, modificar y borrar libros y revistas.
3. Generar automáticamente objetos en el sistema para cada tipo de usuario y producto bibliográfico.
4. Permitir a un usuario comprar un libro.
5. Permitir a un usuario suscribirse a una revista.
6. Permitir a un usuario simular una sesión de lectura (regular y premium).
7. Presentar al usuario su Biblioteca de Productos Bibliográficos.
8. Generar informes con los datos registrados:
 - a. Para cada tipo de producto bibliográfico, libro y revista, informar el acumulado total de páginas leídas en toda la plataforma (tipo de producto y número de páginas leídas).
 - b. Informar el género de libro y categoría de revista más leídas en la plataforma (nombre del género o categoría y número de páginas leídas).
 - c. Informar el Top 5 de libros y el Top 5 de revistas más leídas en la plataforma (nombre del libro o revista, nombre del género o categoría y número de páginas leídas).
 - d. De cada género, informar el número de libros vendidos y el valor total de ventas (\$).
 - e. De cada categoría, informar el número de suscripciones activas y el valor total pagado por suscripciones (\$).

Entregas

Primera entrega

Se deben entregar la primera versión del diagrama de clases y todos los artefactos de análisis, diseño e implementación que le permitan completar los requerimientos 1 al 6.

Segunda entrega

Se deben entregar la versión final del diagrama de clases y los artefactos de análisis, diseño e implementación que le permitan completar todos los requerimientos del 1 al 8 (del 8a al 8e).

Sustentación

Se realizará una sustentación individual ante sus respectivos profesores de Algoritmos y Programación I.