

Biblioteca

Contextualización

AtlasBooks es una cadena de bibliotecas con presencia mundial. Su modelo de negocio se asemeja al de **Blockbuster**: los usuarios se registran, eligen un libro, lo toman prestado y, después de un período determinado, lo devuelven. Sin embargo, no pueden tomar prestado otro libro hasta que hayan devuelto el anterior. A pesar de la popularidad de **AtlasBooks** y su crecimiento en términos de sucursales, enfrentan desafíos en la gestión de usuarios y stock. Actualmente, cada vez que los usuarios visitan una nueva sucursal, deben registrarse nuevamente, lo que ha generado cierto descontento en los usuarios. Ante esta situación, la gerencia de **AtlasBooks** ha contactado a Antares en busca de una solución.

Objetivo

La necesidad principal de **AtlasBooks** es implementar un sistema que centralice los datos de los usuarios, permitiendo que estén disponibles en cualquier sucursal alrededor del mundo. De esta manera, con un único registro, cualquier sucursal podrá acceder a la información del usuario simplemente con su documento de identidad. Adicionalmente, **AtlasBooks** desea una aplicación de software de escritorio que pueda monitorear los préstamos de libros. Esta aplicación requerirá información detallada de cada usuario, de cada libro y del período de préstamo acordado.

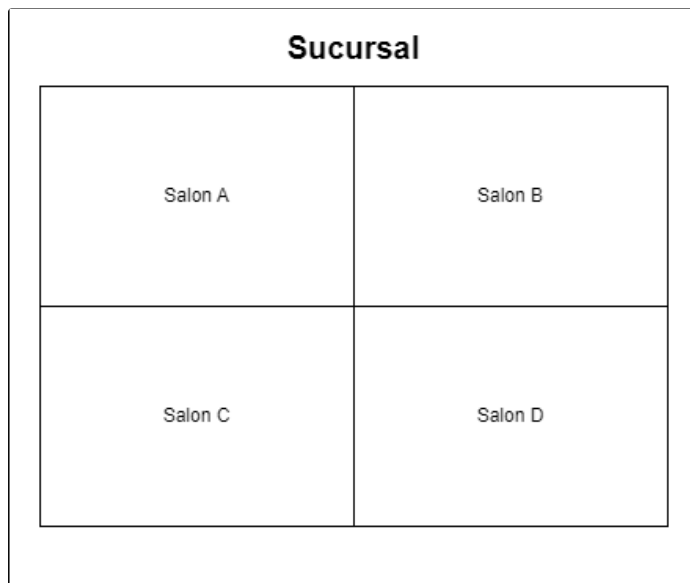
Es importante mencionar que las sucursales de **AtlasBooks** tienen una estructura estandarizada para organizar sus libros:

- **Salón:** Conjunto de estanterías.
- **Estantería:** Conjunto de estantes.
- **Estante:** Conjunto de secciones.
- **Sección:** Conjunto de libros.

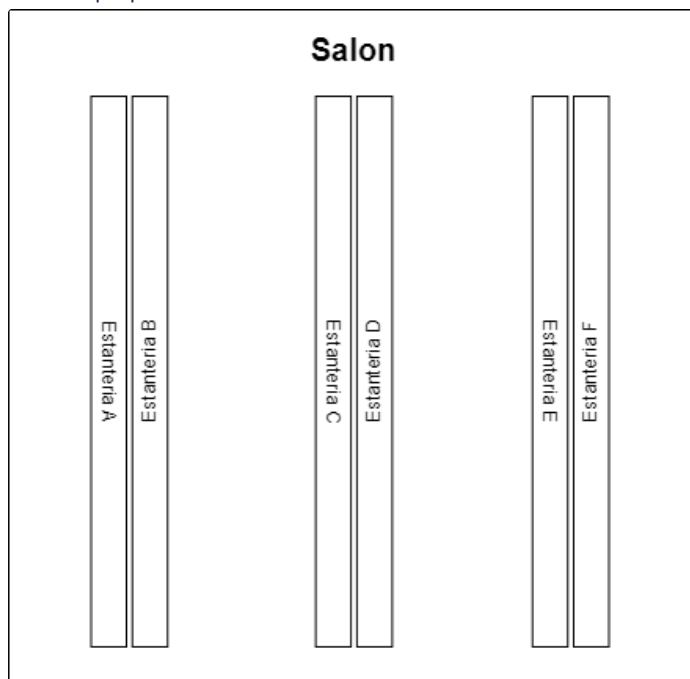
El objetivo es que, mediante la aplicación, los usuarios puedan localizar rápidamente un libro en los salones con solo ingresar el nombre del libro, evitando así la necesidad de interactuar directamente con el personal de la biblioteca.

Entidades identificadas

- Usuarios: entidad encargada de diferenciar a un sujeto registrado de uno no registrado, asimismo también sirve para determinar si un usuario tiene más permisos que otro. (No usuario, Usuario y Administrador)
 - **No usuario**: Sujeto donde su DNI no figura en la base de datos
 - **Usuario**: Sujeto donde su DNI figura en la base de datos
 - **Administrador**: Sujeto donde su DNI figura en la base de datos y dispone de funcionalidades únicas además de las que tiene un **Usuario**
- Libros: entidad encargada de representar un libro físico todo libro en una sección debe estar representado digitalmente en la base de datos esto implica tener la posibilidad de indicar su ubicación a través de las entidades nombradas a continuación
- Salon: Representación de los salones de una sucursal



- Estanteria: Representación de las estanterias distribuidas en los salones, es decir, una estanteria no puede existir sin un salon al que pertenecer



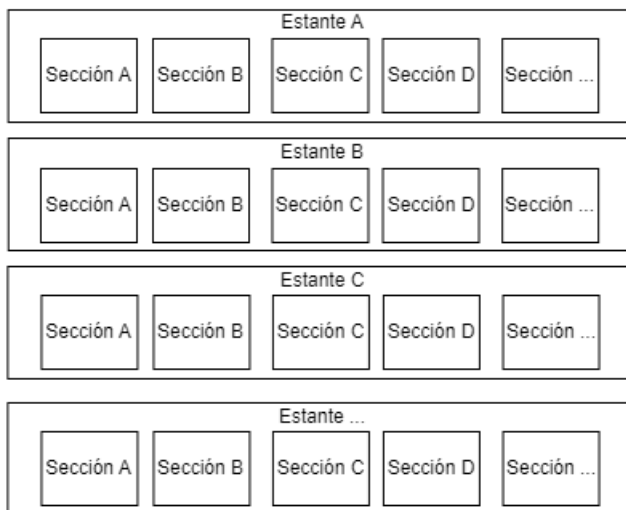
- Estante: Representación de un segmento de una estanteria en particular, es decir, un estante no puede existir sin una estanteria a la que pertenecer

Estanteria

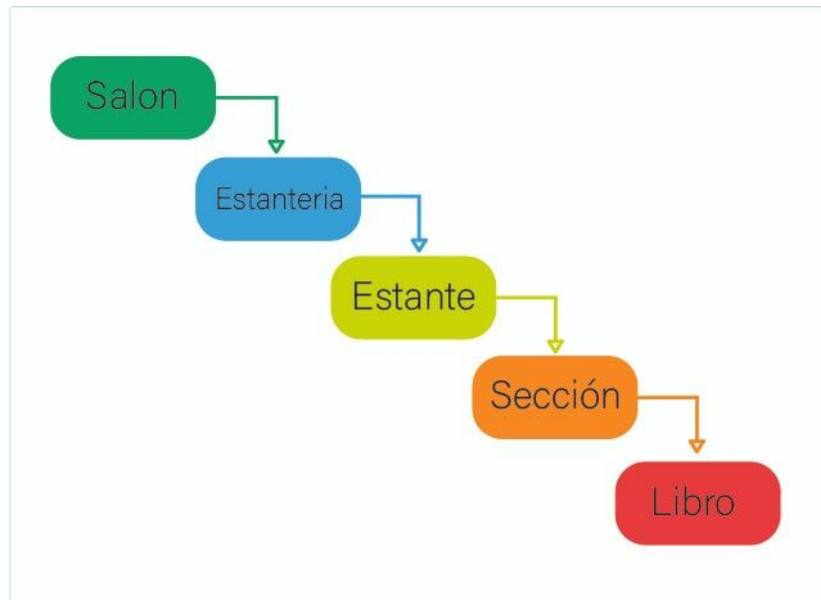
Estante A
Estante B
Estante C
Estante D
Estante E
Estante F
Estante G
Estante H
Estante ...
Estante ...
Estante ...
Estante ...
Estante ...

- Sección: Representación de un segmento de un estante en particular, es decir, determinación de un area dentro de un estante que involucra el conjunto de uno o mas libros.

Estanteria

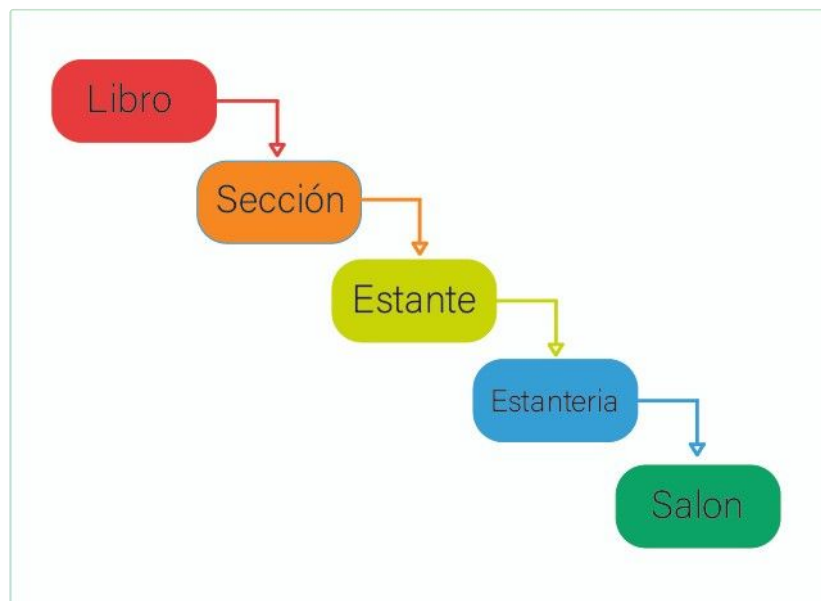


De esta manera se puede comprender que un libro estara situado en una sección de un estante que pertenece a una estanteria que esta dentro de un salon.



Decantación del filtro para llegar hasta uno o mas libros desde el Salon

Y de forma inversa tambien se podría llegar a la ubicación completa con el nombre del libro unicamente



Decantación del filtro para llegar a la ubicación de un libro desde el libro

i Solo hay una forma en la que un libro puede existir en la base de datos y no aparecer en la búsqueda antes mencionada y es **si solo si** ese libro este en caracter de prestamo con un usuario.

- Prestamos: Representación de un acuerdo entre el usuario con la empresa **AtlasBooks** donde la empresa se propone a darle un libro a un usuario mientras el usuario se proponga a entregarlo antes del tiempo de prestamo pactado. Es decir, esta entidad se encargara de llevar un registro de todos los prestamos hechos al igual de los prestamos que ya fueron concretados (devolución del libro por parte del usuario)

Logica de negocio para generar un prestamo



Acciones identificadas por los roles de usuario [↗](#)

Roles → Entidades ↓	No usuario	Usuario	Administrador
Libro	<input checked="" type="checkbox"/> Leer <input type="checkbox"/> Agregar <input type="checkbox"/> Editar <input type="checkbox"/> Eliminar	<input checked="" type="checkbox"/> Leer <input type="checkbox"/> Agregar <input type="checkbox"/> Editar <input type="checkbox"/> Eliminar	<input checked="" type="checkbox"/> Leer <input checked="" type="checkbox"/> Agregar <input checked="" type="checkbox"/> Editar <input checked="" type="checkbox"/> Eliminar
Usuario	<input type="checkbox"/> Leer <input type="checkbox"/> Agregar <input type="checkbox"/> Editar <input type="checkbox"/> Eliminar	<input checked="" type="checkbox"/> Leer (Propio) <input type="checkbox"/> Agregar <input checked="" type="checkbox"/> Editar (Propio) <input type="checkbox"/> Eliminar	<input checked="" type="checkbox"/> Leer <input checked="" type="checkbox"/> Agregar <input checked="" type="checkbox"/> Editar <input checked="" type="checkbox"/> Eliminar
Ubicaciones	<input checked="" type="checkbox"/> Leer <input type="checkbox"/> Agregar <input type="checkbox"/> Editar <input type="checkbox"/> Eliminar	<input checked="" type="checkbox"/> Leer <input type="checkbox"/> Agregar <input type="checkbox"/> Editar <input type="checkbox"/> Eliminar	<input checked="" type="checkbox"/> Leer <input checked="" type="checkbox"/> Agregar <input checked="" type="checkbox"/> Editar <input checked="" type="checkbox"/> Eliminar
Prestamos	<input type="checkbox"/> Leer <input type="checkbox"/> Agregar <input type="checkbox"/> Editar <input type="checkbox"/> Eliminar	<input type="checkbox"/> Leer <input type="checkbox"/> Agregar <input type="checkbox"/> Editar <input type="checkbox"/> Eliminar	<input checked="" type="checkbox"/> Leer <input checked="" type="checkbox"/> Agregar <input checked="" type="checkbox"/> Editar <input checked="" type="checkbox"/> Eliminar

Informacion que se requiere guardar [↗](#)

• Propiedades del Libro

- Título
- Sinopsis
- Puntaje de la critica
- Ubicacion (sección a la que pertenece)

• Propiedades de un Usuario

- Nombre
- Apellido
- Fecha de Nacimiento
- Numero de telefono
- Email
- Rol

• Propiedades de un Rol

- Nombre
- Descripción

Tenga en cuenta que los datos a guardar son los que la biblioteca necesita de base, es decir, **no pueden faltar bajo ningun motivo**. De todas formas tomese la libertad de realizar las conexiones, segmentacion de tabla, o la estructura de la base de datos que mas le guste.

Convenciones

- Idioma: ingles
- Tablas: Ademas de las propiedades de cada tabla en particular, se deben agregar tres propiedades extra a todas las tablas que se construyan
 - Fecha de creación del registro
 - Fecha de actualizacion del registro
 - Booleano para saber si el registro esta activo o no (borrado logico)

Preparación del entorno con docker

Desde Antares recomendamos el uso de Docker para instanciar las bases de datos por su portabilidad y versatilidad y facilidad de instanciacion. **No es obligatorio el uso de Docker y el trabajo sera calificado por igual sobre una persona que levante la instancia a travez del metodo convencional.**

Sitio oficial de Docker: [🐳 Docker: Accelerated Container Application Development](https://www.docker.com/)

Link a la imagen de SQL Server 2019: [🐳 Microsoft SQL Server - Ubuntu based images by Microsoft | Docker Hub](https://mcr.microsoft.com/mssql/server/2019-latest)

Comando en terminal de cortesia para descargar e instanciar un contenedor con dicha imagen:

```
1 docker run -e "ACCEPT_EULA=Y" -e "SA_PASSWORD=Antares123" \  
2 -p 1433:1433 --name sql_server_container \  
3 -d mcr.microsoft.com/mssql/server:2019-latest  
4
```

Entrega esperada



1. Estructura:

- a. **Descripción:** Diagrama entidad-relación de la base de datos, detallando las relaciones, llaves foráneas y llaves primarias.
- b. **Entregable:** Una imagen del diagrama junto con un archivo `.SQL` que contenga las instrucciones para la generación de la base de datos.

2. Stored procedures:

- a. **Descripción:** Procedimientos almacenados para la inserción, edición y eliminación de las entidades definidas previamente.

b. **Entregable:** Integrar estos procedimientos en el archivo `.SQL` del primer entregable.

3. Vistas:

a. **Descripción:** Vistas que faciliten la visualización conjunta de entidades. Ejemplo: la combinación de libros y sus ubicaciones, o préstamos, libros y usuarios.

b. **Entregable:** Estas vistas deben ser añadidas al archivo `.SQL` que incluye la estructura de la base de datos y los stored procedures.

4. Funciones:

a. **Descripción:** Funciones para cumplir con las especificaciones del cliente, como la búsqueda de libros a través de parámetros como la ubicación o el nombre. Estas funciones deben gestionar adecuadamente los valores nulos.

b. **Entregable:** Las funciones deben incorporarse al archivo `.SQL` mencionado en los puntos anteriores.

5. Triggers:

a. **Descripción:** Triggers para manejar eventos ocasionales basados en condiciones específicas, como prevenir el registro de préstamos si un usuario ya tiene un libro en su poder o si un libro ya ha sido prestado y aún no ha sido devuelto.

b. **Entregable:** Estos triggers deben ser añadidos al archivo `.SQL` consolidado que se ha estado construyendo en las etapas anteriores.

Desarrollo esperado

- **Archivo de Entrega:** El producto final debe ser un único archivo con extensión `.SQL`, diseñado para ser compatible y ejecutable en SQL SERVER 2019.
- **Convenciones y Nomenclatura:** Se espera coherencia y claridad en la nomenclatura utilizada. Esto incluye la denominación de tablas, propiedades, stored procedures y vistas. El candidato deberá justificar y defender la elección de estas convenciones durante la revisión.
- **Diseño y Relaciones:** Es vital que el diseño de la base de datos refleje adecuadamente la estructura y relaciones entre las entidades identificadas, como usuarios, libros, ubicaciones, entre otros. Se evaluará la lógica detrás de las conexiones y relaciones establecidas.
- **Propiedades de las Tablas:** Las tablas deben contener propiedades relevantes y necesarias para satisfacer las necesidades del sistema. Durante la revisión, se espera que el candidato explique las razones detrás de la inclusión o exclusión de ciertas propiedades.
- **Metodología en Stored Procedures y Vistas:** El enfoque y la estructura utilizados para construir los stored procedures y vistas serán un punto central de la revisión. Se espera que el candidato pueda defender sus decisiones metodológicas.
- **Revisión Adicional:** Aparte de los puntos mencionados anteriormente, el candidato debe estar preparado para responder preguntas adicionales basadas en el desarrollo realizado. Estas preguntas buscarán evaluar la profundidad del entendimiento del candidato y su capacidad para adaptar soluciones a las necesidades del cliente.