UNIVERSIDAD
DE GRANADA

Prácticas y Documentación

Completa Comparador RadarBook

Ingeniería de Sistemas de Información

José Miguel Aguado Coca

Alejandro Bolivar Corpas

Adrián Martos Medina

Índice

Propuesta de Proyecto

Título y Temática del Proyecto

Como título para nuestro proyecto nos hemos decantado por "**RadarBook**". Este nombre da a entender sobre qué tema tratará el proyecto, un **comparador de precios de libros**.

Integrantes del Proyecto

- José Miguel Aguado Coca
- Alejandro Bolivar Corpas
- Adrián Martos Medina

Proyectos existentes en el mismo ámbito

- https://www.oklibros.com : Solo permite la búsqueda por isbn, se puede subir un documento con los ISBN de todos los libros que deseas buscar.
- https://www.letturalia.com/es: Permite búsqueda por título ISBN y autor, también permite subir un fichero con todo lo que quieras buscar.
- https://www.buscarlibros.com/: Permite muchas opciones de búsqueda además de especificar el formato, tipo de tapa o usado o nuevo, sin embargo, no permite la búsqueda en masa pasandole un archivo.

Fuentes de Datos

Nos hemos decantado por las siguientes fuentes de datos:

- Rainforest API
- Scrapper API (Extracción de datos de amazon y funcionalidades como saltarse captchas y aceptar cookies)
- Web Scrapping Librería San Pablo
- Web Scrapping Ebay

Descripción del Ámbito del Proyecto

En la actualidad, el mundo de los libros es tan grande que encontrar nuestro favorito al mejor precio se puede volver un auténtico dolor de cabeza. RadarBook es un proyecto de comparador de precios de libros que al contrario que los actuales comparadores de libros, se centra en proveer un servicio centrado en las opiniones, reseñas y experiencias de otros usuarios. De esta forma buscamos mejorar la experiencia de compra y selección de libros, saliéndose de la norma de leer únicamente una sinopsis.

En nuestra aplicación, será posible buscar un libro tanto por ISBN como por nombre, una vez introducido el criterio en nuestra barra de búsqueda y pulsado el botón enter, aparecerá en la parte inferior un listado con todas las coincidencias encontradas en nuestras fuentes de datos, de las cuales haya stock disponible, las que no tengan stock se descartarán y no se enseñarán. Junto con ellas, aparecerá el precio de las mejores ofertas, información del libro (incluyendo autor, editorial y número de páginas) y la valoración de los lectores junto con determinadas reseñas para que el comprador pueda orientarse en cuanto al contenido del libro sin necesidad de leerlo. Teniendo así el valor agregado de opiniones de terceros para proceder a la compra del libro a través de los enlaces a la compra a los que se accederá con un simple click en un botón “Compra Ya”.

Por el contrario, hay algunos aspectos que no se tendrán en cuenta para el desarrollo del proyecto RadarBook como por ejemplo la venta directa de libros. Se ha descartado por su elevado coste logístico y de almacenamiento y por la existencia de otras alternativas mejores como el programa de afiliados de Amazon, el cual cuando un cliente compra un producto desde nuestro enlace, recibimos una pequeña comisión. Siendo así el beneficio un poco menor pero reduciendo los gastos a 0. Tampoco tendremos en cuenta los libros electrónicos (e-books). Queremos fomentar mediante RadarBook el uso de libros físicos, por lo que nuestro servicio irá orientado solo a estos.

Existen ciertas características que queremos conservar para futuras versiones, pero que no se implementarán por motivos de planificación. En primer lugar, para mejorar la escalabilidad, sería conveniente añadir soporte multilenguaje con el fin de aumentar el público objetivo y expandirse a otros países. En segundo lugar, para mejorar la afiliación y el sentimiento de comunidad, el equipo ha pensado en añadir funcionalidades sociales como la implementación de foros y la creación de listas de libros recomendados basados en las búsquedas y comparaciones anteriormente realizadas por el cliente.

Presupuesto del Sistema

Presupuesto Desarrollo del Sistema

Hemos decidido que la duración del proyecto será de 4 meses, con 3 miembros. Debido a que el sueldo medio de un informático es de 36.200 euros brutos al año (1) , el sueldo bruto de cada paga, que son 14, será de 2.585,71 euros.

Duración del proyecto	4	meses									
Miembros del equipo	3	personas									
Esfuerzo requerido	**12,00**	personas·mes	1,09	personas·año							
Coste salarial	**4.343,34 €**	€/persona·mes									
Gastos de personal	**52.120,10 €**										
Sueldo bruto mensual	2.585,71 €	**Rellenar el sueldo bruto mensual deseado para obtener los costes salariales y el sueldo neto**									
Sueldo bruto anual	36.200,00 €	(en 14 pagas)									
COSTE SALARIAL ANUAL	47.776,76 €										
COSTE SALARIAL MENSUAL	**4.343,34 €**	(descontando el mes de vacaciones)									
				Cotizaciones	**Empresa**	**Empleado**		**IRPF**	**Hasta**	**€**	
Sueldo bruto	36.200,00 €			Contingencias comunes	23,60%	4,70%		19%	12.450,00 €	2.365,50 €	
+ Cotizaciones	11.576,76 €	31,98%		MEI	0,58%	0,12%		24%	20.200,00 €	1.860,00 €	
Coste salarial	**47.776,76 €**			Desempleo	5,50%	1,55%		30%	35.200,00 €	4.500,00 €	
				Formación profesional	0,60%	0,10%		37%	60.000,00 €	370,00 €	
Sueldo bruto mensual	2.585,71 €			AT y EP	1,50%			45%	300.000,00 €		
- Cotizaciones	167,30 €	6,47%		FOGASA	0,20%			47%			
- IRPF	649,68 €	25,13%		**Total**	**31,98%**	**6,47%**		**Total**		**9.095,50 €**	
Sueldo neto mensual	1.768,74 €								Tipo efectivo	25,13%	
- Sueldo neto anual	24.762,36 €			Base máxima	4.720,50 €	mensual					

Los gastos directos de ejecución serán explicados con profundidad en el apartado “In House”. Para los gastos complementarios:

- **Viajes, estancias y dietas**: promocionaremos el producto por 4 universidades, Madrid, Barcelona, Valencia y Granada. Aproximadamente, el coste del viaje en AVE a cada ciudad es de 60 euros, siendo el de Granada de 0 por tener nuestra oficina aquí. Además, pasaremos día y noche en cada ciudad, necesitando 20 euros por comida. En total, 300 euros.
- **Material de difusión y promoción**: promocionaremos el producto con folletos por todo el país. Estimamos 1000 euros para la promoción.
- **Inscripción en cursos y congresos**: ningún gasto para este apartado.
- **Otros gastos**: estimamos 150 euros para este apartado.

Proyecto	RadarBook		
Equipo	Adrián Martos Medina, José Miguel Aguado Coca, Alejandro Bolivar Corpas		
1.- GASTOS DE PERSONAL	**52.120,10 €**	**76,15%**	ver "Personal"
1.1. Personal de plantilla	52.120,10 €	76,15%	
1.2. Personal de nueva contratación	0,00 €	0,00%	
2.- GASTOS DIRECTOS DE EJECUCIÓN	**14.876,80 €**	**21,73%**	
Material inventariable: Adquisición de aparatos y equipos	7.485,49 €	10,94%	imputable 100%
Material inventariable: Amortización de aparatos y equipos	499,03 €	0,73%	ver "Amortizacio
Material inventariable: Alquiler de aparatos y equipos	0,00 €	0,00%	
Consultoría (asesoramiento técnico)	0,00 €	0,00%	
Software: adquisición, licencias y uso	542,28 €	0,79%	
Alquiler de instalaciones	6.000,00 €	8,77%	
Contratos de suministros	0,00 €	0,00%	
Adquisición de material fungible (p.ej. material de oficina)	350,00 €	0,51%	incluyendo mate
3.- GASTOS COMPLEMENTARIOS	**1.450,00 €**	**2,12%**	
Viajes, estancias y dietas	300,00 €	0,44%	
Material de difusión y promoción	1.000,00 €	1,46%	
Inscripción en cursos y congresos	0,00 €	0,00%	
Otros gastos	150,00 €	0,22%	especificar...
TOTAL (BASE IMPONIBLE, 1+2+3)	**68.446,90 €**	**100,00%**	
IVA (21%)	**14.373,85 €**		
IMPORTE (I.V.A. INCLUIDO)	**82.820,75 €**		

D	**Duración del proyecto**	**4**	meses
C	**Coste del equipo**	7.485,49 €	
A	**Período de amortización**	**60**	meses
	Coste amortizable en el proyecto	**499,03 €**	=C*D/A

Presupuesto Despliegue del Sistema

Versión “In House”

- **Material inventariable “Adquisición de aparatos y equipos”**: necesitaremos en nuestra oficina un ordenador por cada cliente, tres ordenadores que supondrán 5.187,27 euros (2). Además, hará falta un ordenador más potente en el que alojar el servidor, con un precio de 1.747,42 euros (3). Por último, por cada ordenador necesitaremos un teclado, un ratón, y un monitor, con el coste en total de 550,8 euros (4). En conclusión, para este apartado el coste total será de 7485,49. Respecto al material inmobiliario de la oficina, todo vendrá incluido en el local que vamos a alquilar.
- **Material inventariable “Amortización de aparatos y equipos”:** la amortización del equipo será de 5 años, por lo que el coste amortizable es de 499,03 euros.
- **Material inventariable “Alquiler de aparatos y equipos”:** no vamos a alquilar nada.
- **Consultoría “Asesoramiento técnico”:** no necesitaremos nada para asesoramiento, gasto de 0 euros.
- **Software “Adquisición, licencias y uso”:** la licencia de Microsoft Office tiene un precio de 11,70 por persona al mes (5), de 140,4 entre los 3 integrantes los 4 meses de trabajo. Además, la licencia de Adobe Photoshop tiene un precio de 33,49 por persona al mes (6), lo que da en los 4 meses 401,88 euros, además el primer pago de Rainforest API son 59 euros y el de Scrapper API 50€. En total, en licencias necesitaremos 651,28 euros.
- **Alquiler de instalaciones:** alquilaremos una oficina espaciosa en la que podamos trabajar los 3 integrantes del grupo. El precio será de 1.500 euros al mes (7), por lo que en 4 meses será de 6.000 euros.
- **Contratos de suministros:** todo este apartado estará incluido en el alquiler, por lo que el gasto será de 0 euros.
- **Adquisición de material fungible “p.ej. material de oficina”:** entre bolígrafos, paquetes de folios, material de trabajo… La estimación es de 350 euros.

2.- GASTOS DIRECTOS DE EJECUCIÓN	14.869,31 €	21,47%
Material inventariable: Adquisición de aparatos y equipos	7.485,49 €	10,81%
Material inventariable: Amortización de aparatos y equipos	382,54 €	0,55%
Material inventariable: Alquiler de aparatos y equipos	0,00 €	0,00%
Consultoría (asesoramiento técnico)	0,00 €	0,00%
Software: adquisición, licencias y uso	651,28 €	0,94%
Alquiler de instalaciones	6.000,00 €	8,67%
Contratos de suministros	0,00 €	0,00%
Adquisición de material fungible (p.ej. material de oficina)	350,00 €	0,51%

El coste inicial de la puesta en marcha, será de 14.869,31 euros, es decir, lo que nos costará desplegar la parte hardware y software.
Tendremos 10000 consultas. El coste inicial por usuario será de 1,49 euros.
El coste mensual será aproximadamente de 4.000 euros.

El coste mensual por usuario será de 0,4 euros.

Versión “En la Nube”

Nos hemos decantado por Google Cloud(8). En su calculadora, hemos elegido App Engine, con zona local en Reino Unido (Londres), sistema operativo Windows Server, uso constante de cargas de trabajo, 730 horas al mes, máquina C2D con 20GB de boot y 2 SSD de 375 de local. El coste será de 637,84 euros al mes, 2551,36 euros en total.

En la adquisición de aparatos y equipos, ya no será necesario sumar el precio del ordenador servidor, porque no lo necesitamos.

Proyecto	**RadarBook**		
Equipo	Adrián Martos Medina, José Miguel Aguado Coca, Alejandro Bolivar Corpas		
1.- GASTOS DE PERSONAL	**52.120,10 €**	**75,27%**	ver "Personal"
1.1. Personal de plantilla	52.120,10 €	75,27%	
1.2. Personal de nueva contratación	0,00 €	0,00%	
2.- GASTOS DIRECTOS DE EJECUCIÓN	**15.673,25 €**	**22,64%**	
Material inventariable: Adquisición de aparatos y equipos	5.738,07 €	8,29%	imputable 100% al proyecto
Material inventariable: Amortización de aparatos y equipos	382,54 €	0,55%	ver "Amortizaciones"
Material inventariable: Alquiler de aparatos y equipos	2.551,36 €	3,68%	
Consultoría (asesoramiento técnico)	0,00 €	0,00%	
Software: adquisición, licencias y uso	651,28 €	0,94%	
Alquiler de instalaciones	6.000,00 €	8,67%	
Contratos de suministros	0,00 €	0,00%	
Adquisición de material fungible (p.ej. material de oficina)	350,00 €	0,51%	incluyendo material informático <60€
3.- GASTOS COMPLEMENTARIOS	**1.450,00 €**	**2,09%**	
Viajes, estancias y dietas	300,00 €	0,43%	
Material de difusión y promoción	1.000,00 €	1,44%	
Inscripción en cursos y congresos	0,00 €	0,00%	
Otros gastos	150,00 €	0,22%	especificar...
TOTAL (BASE IMPONIBLE, 1+2+3)	**69.243,35 €**	**100,00%**	
IVA (21%)	**14.541,10 €**		
IMPORTE (I.V.A. INCLUIDO)	**83.784,45 €**		

El coste inicial, y el coste inicial por usuario es de 0 euros. El coste al mes será de 746,84 euros, 10000 consultas, y el precio por consulta es de 0,07 euros.

Diseño Arquitectónico

Decisión de Arquitectura

Para nuestro proyecto, entre las diversas arquitecturas existentes en el mundo del desarrollo, nos hemos decantado por una Arquitectura Modelo Vista Controlador (MVC).

Esto nos va a permitir separar la aplicación en partes. El cliente, accede a la página principal, el index.html. Al realizar la búsqueda (por nombre o por ISBN), se redirige la información a través de un dispatcher al controlador, que será el servlet.

Este controlador hace tres llamadas al Modelo(el modelo encapsula las distintas clases implementadas para las fuentes de datos y para almacenar la información de los libros, la lógica), una por cada fuente de datos. Cada fuente va a devolver un libro distinto. El modelo devuelve estos tres libros al controlador, que pasa estos tres libros a la vista (es un JSP) que va a proceder a mostrar la información de cada libro. En la vista hay también un error JSP, el cuál aparece si ocurre algún error a la hora de buscar el libro.

Al usar este tipo de arquitectura vamos a tener distintos beneficios:

1) Podemos reutilizar código.
2) Es fácil de hacer que crezca el sistema. Al añadir nuevas funcionalidades, como todo está separado, la implementación va a ser sencilla, al igual que el mantenimiento.
3) Trabajar con compañeros también va a ser sencillo, unos se pueden encargar del modelo, otros de la vista, y otros del controlador, al mismo tiempo.
4) El código va a estar bien organizado.

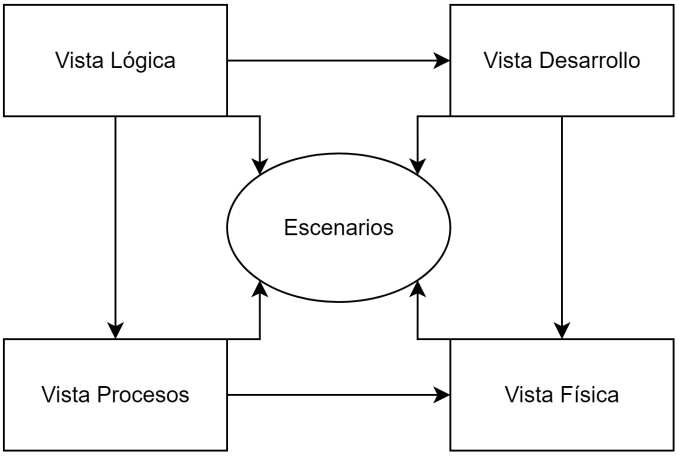
Alternativas a nuestro Diseño

Como alternativas, nos hemos encontrado principalmente con tres:

1) Arquitectura Monolítica: es más sencilla de implementar, pero descartada porque conforme crezca la aplicación, van a aparecer más problemas.
2) Arquitectura Cliente servidor: los clientes se comunican con el servidor, y el servidor responde. Descartada por posibles fallos.
3) Hemos descartado usar base de datos junto a nuestro proyecto por su costosa implementación.

Decisión Conjunto de Vistas

Para el desarrollo de nuestro proyecto, nos hemos decantado por un modelo de vistas 4+1 de Kruchten. Gracias a este modelo, podremos representar nuestro sistema desde 4 vistas, la lógica, la de desarrollo, la de proceso y la física. El “+1” hace referencia a una vista adicional, llamada escenarios, la cual reúne información de todas las 4 vistas.
Para el desarrollo de esta práctica, indagaremos más a fondo en cada apartado, dando explicaciones acerca de nuestras decisiones



Requisitos Funcionales para nuestro diseño

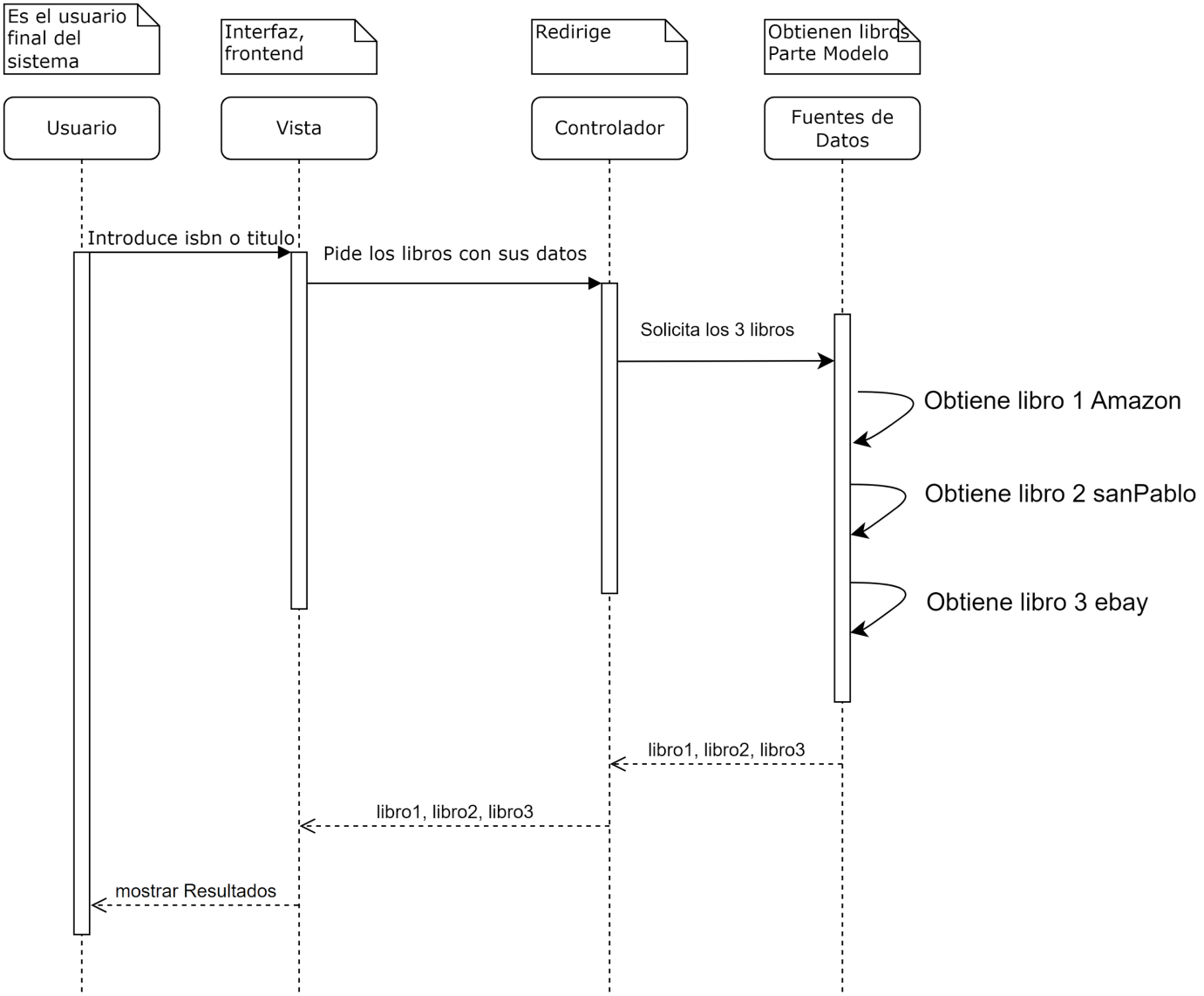
En nuestra aplicación, debemos cumplir una serie de requisitos funcionales para su correcto desarrollo:

RF1) Búsqueda de libros por ISBN: el sistema permitirá a los usuarios buscar el libro deseado mediante el código ISBN. Se mostrará el resultado correspondiente.
RF 2) Búsqueda de libros por título: el sistema permitirá a los usuarios buscar el libro deseado mediante el título. Se mostrará el resultado correspondiente.
RF 3) Integración con las API: el sistema se integrará con las API para obtener información acerca de los libros buscados por el usuario.
RF 4) Mostrar Resultados: el sistema debe mostrar los resultados del libro buscado, incluyendo el título, el precio, las valoraciones proporcionadas por las APIs , el enlace para realizar la compra, imagen, fecha, páginas, autores e isbn.
RF 5) Interfaz de Usuario: la interfaz del sistema debe ser sencilla para proporcionar una buena experiencia de uso al usuario.
RF 6) Incidencias y Errores: el sistema mostrará cualquier error que haya con un mensaje.
RF 7) Mostrar reseñas: al hacer la búsqueda se mostrarán las reseñas proporcionadas por Amazon.
RF 8) Proporcionar valoración de uno de los libros: el libro proporcionado por Amazon mostrará la valoración que tiene para que el usuario se haga una idea de si le interesa comprarlo o no.

Modelo 4+1

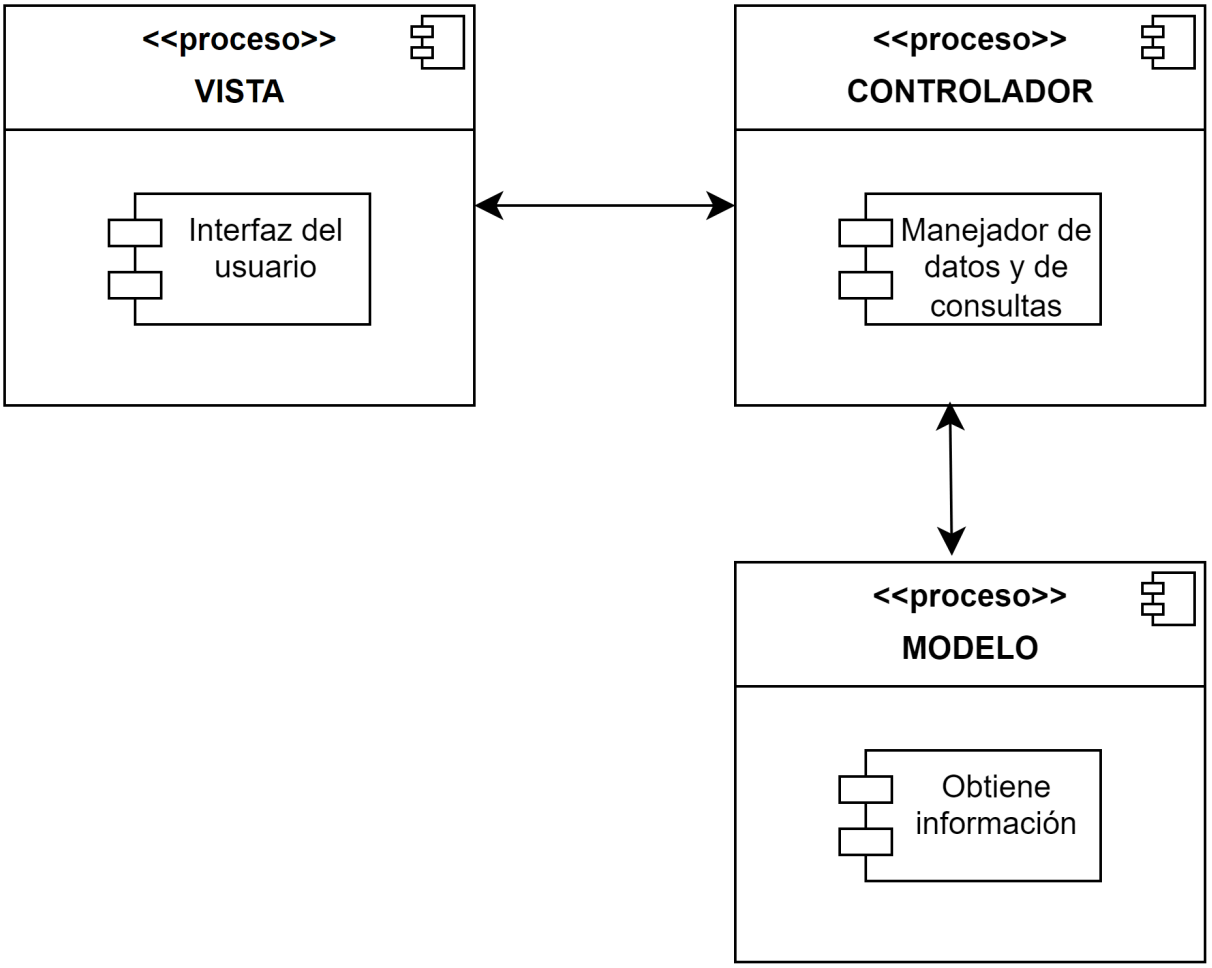
Vista Lógica

Esta vista debe de mostrar la funcionalidad que se le debe ofrecer al usuario final.



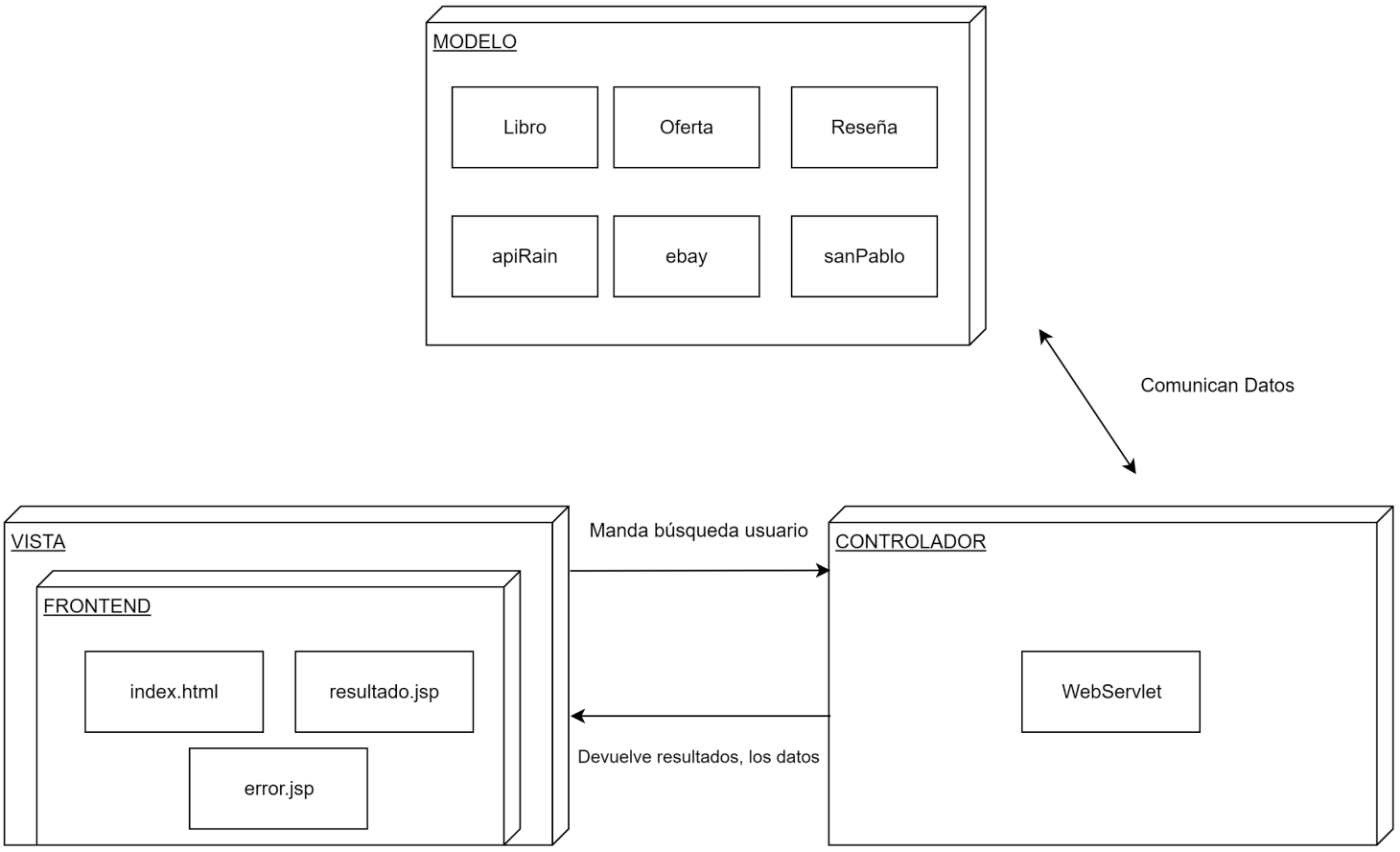
Vista de Procesos

La vista de procesos dentro del modelo 4+1, se basa en describir cómo el sistema se comporta en tiempo de ejecución. Ésta se enfoca en los procesos que se ejecutan, la concurrencia, su comunicación y cómo lleva a cabo el procesamiento de datos a través del sistema. En este sistema hemos podido identificar vamos a tener un proceso en la vista (la interfaz de usuario), un proceso en el controlador (manejador de datos y consultas) y otro proceso en el modelo (obtener datos).



Vista de Desarrollo

Esta vista se centra en la perspectiva del programador y se enfoca en mostrar los distintos componentes del sistema. Puesto que vamos a usar una arquitectura modelo vista controlador, haremos la división del sistema en tres partes. La parte de la vista contiene el frontend (index.html, resultado.jsp, error.jsp). La parte del controlador recibe las solicitudes de la vista, obtiene las respuestas comunicándose con el modelo, y genera las respuestas de nuevo hacia la vista. El modelo contiene las clases que interaccionan con las fuentes de datos y las clases dónde se almacenan estos datos.

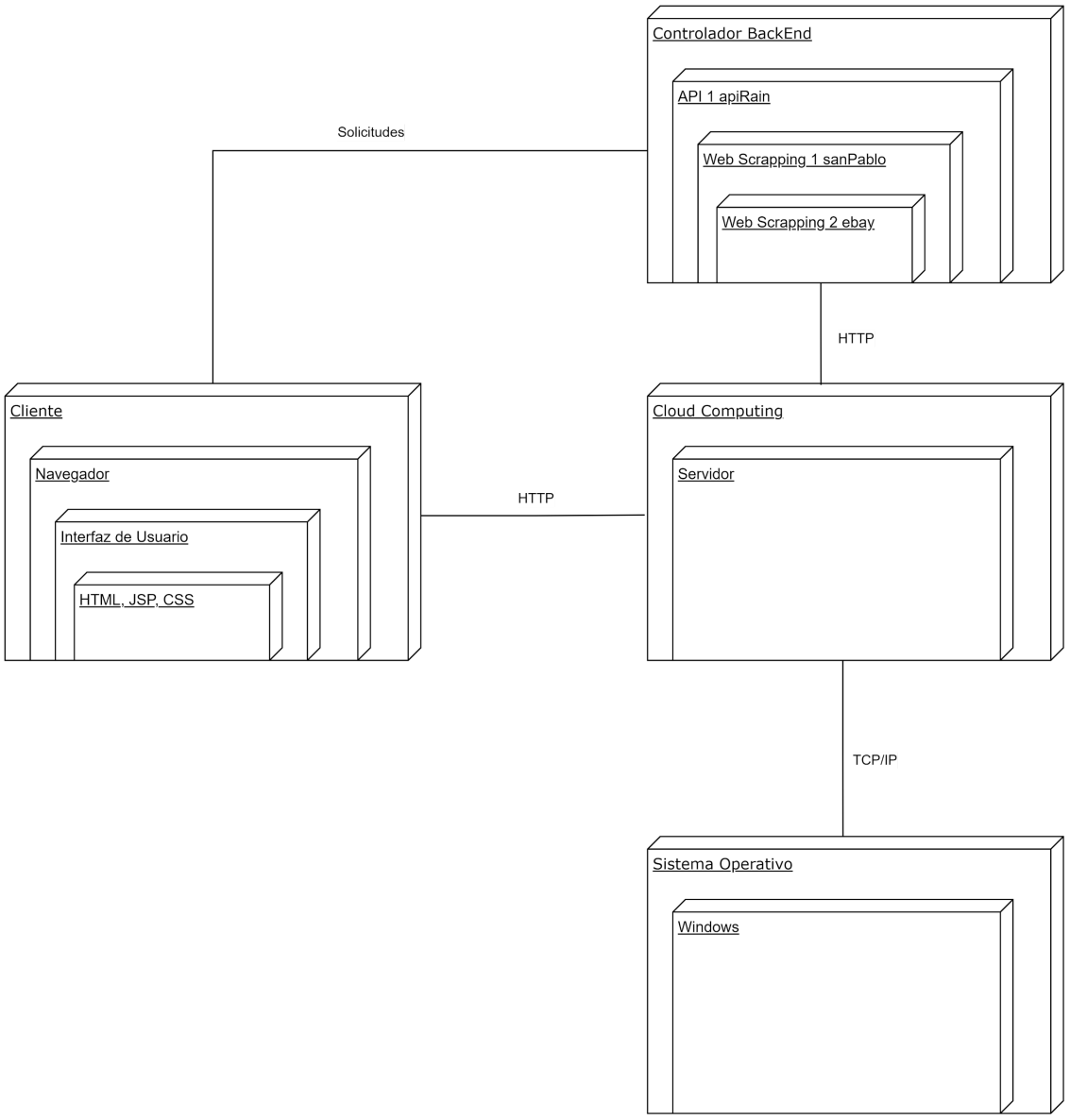


Vista Física

Esta vista representa el despliegue y la ejecución de nuestro sistema entrando en detalle con los componentes físicos, base de datos y sistema. Es decir, componentes hardware y software para nuestra aplicación.

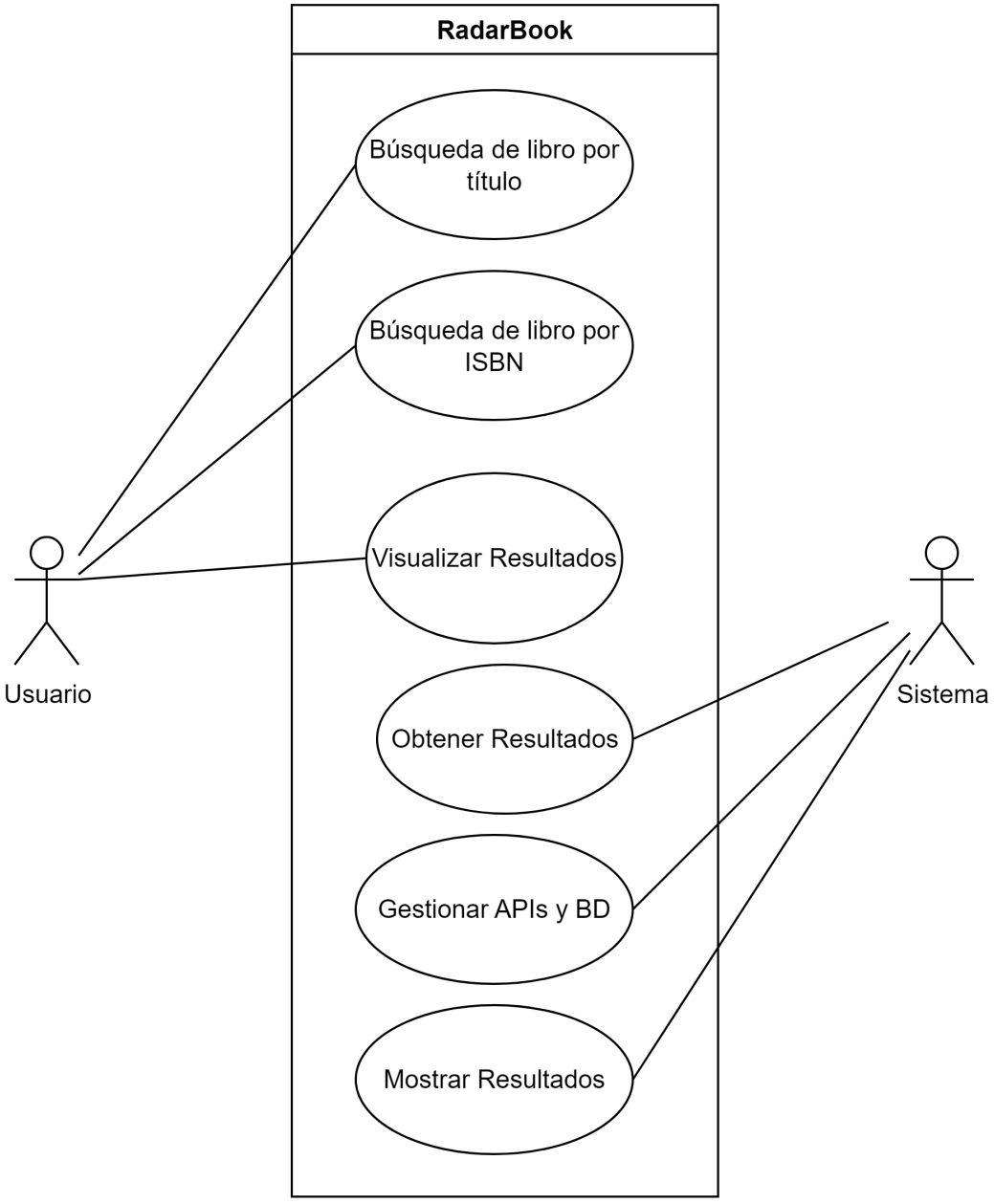
Para esta vista, tenemos en cuenta lo siguiente:

- Servidor: haremos uso de un servidor de cloud computing.
- Sistema Operativo: haremos uso de Windows.



Escenarios

Con esta vista se representan los distintos casos de uso que tendrá nuestro sistema, y los actores que participan, en nuestro caso, usuario y sistema. Haciendo uso de un diagrama de casos de uso representaremos esta vista.



URL Radarbook

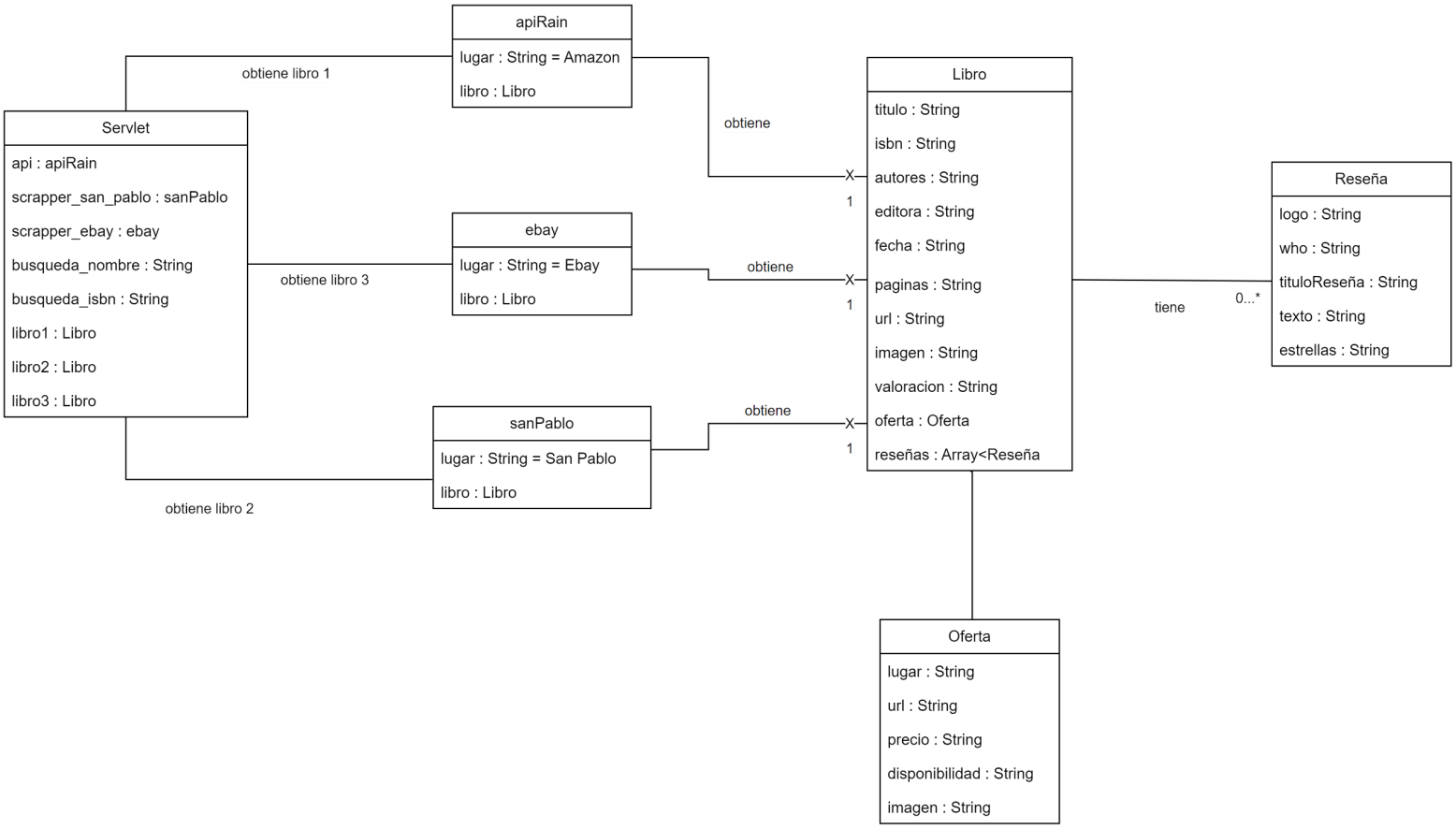
Url para acceder

Para acceder al comparador de precios se necesita la siguiente URL
https://radarbook.appspot.com/

Quedan aproximadamente 300 consultas hasta terminar el plan de 500 consultas.

Modelado de Integración de Datos

Esquema Integrado GCS

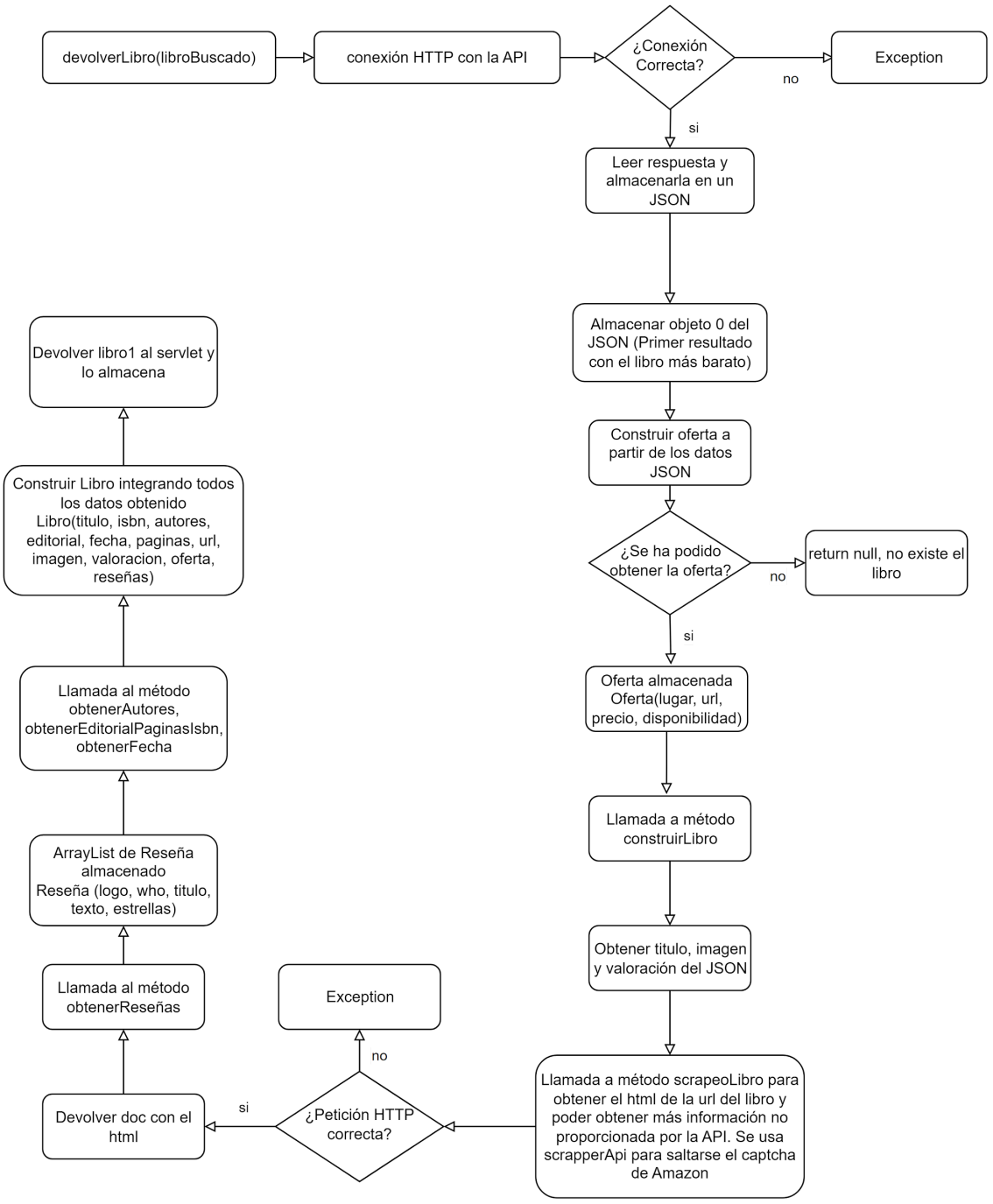


Tenemos una clase libro, de la cuál queremos obtener los atributos descritos en su clase. Cada libro va a contener una oferta, y el libro obtenido por la apiRain va a contener un array de Reseñas. apiRain, ebay y sanPablo realizan un método con el que obtienen el libro con toda su información. Cada libro es distinto. El servlet obtiene estos libros para redirigirlos a la vista.

Esquema de las fuentes de Datos LCS

apiRain

¿Cómo integra los datos la clase apiRain en un libro para mandárselo al servlet? A continuación lo explico con un diagrama de flujo:

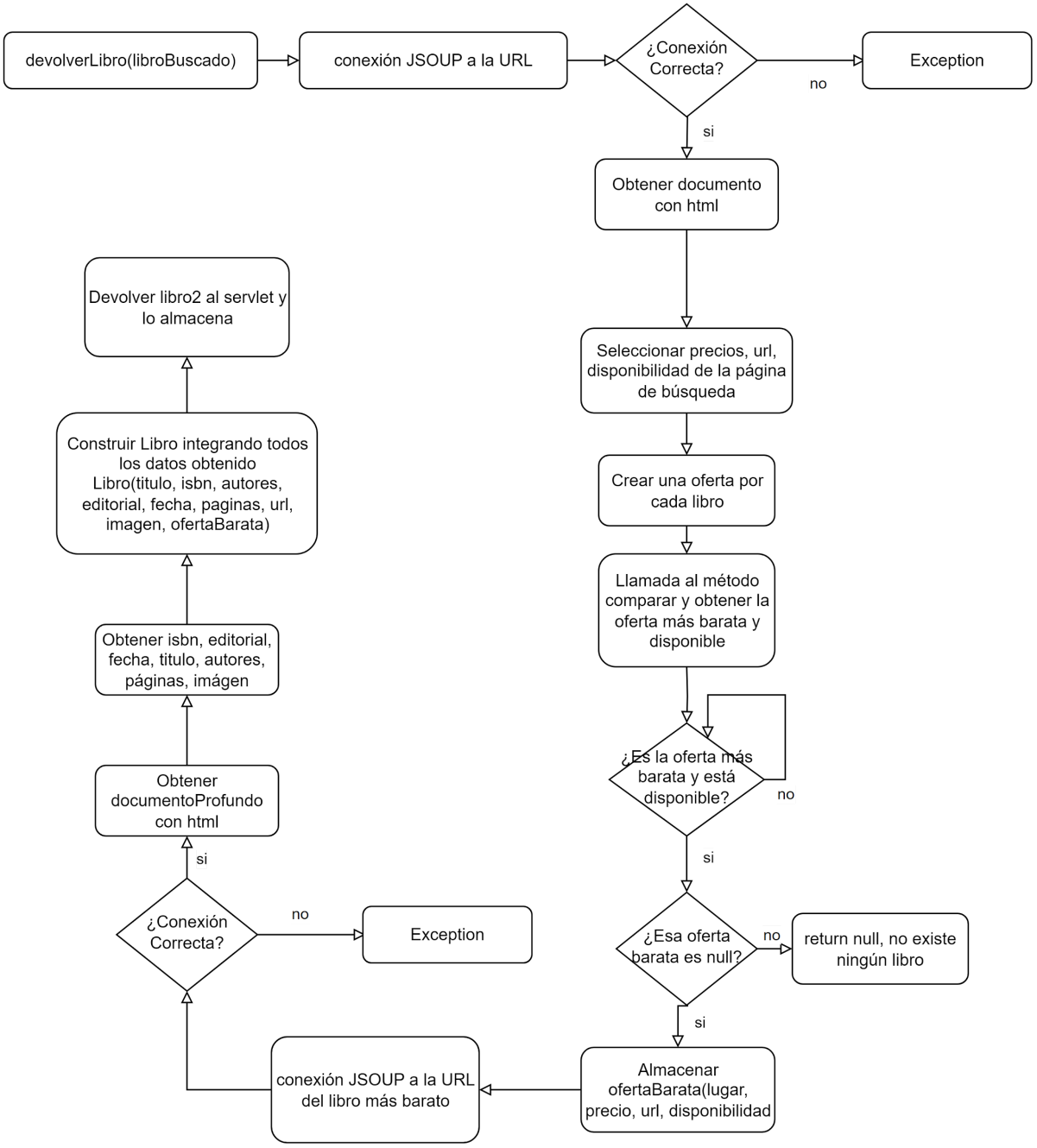


Por lo tanto, esta primera fuente de datos devuelve un
Libro libro1(titulo, isbn, autores, editorial, fecha, paginas, url, imagen, valoracion, oferta, reseñas), que contiene

1) Oferta oferta(lugar, url, precio, disponibilidad)
2) ArrayList<Reseña> reseñas(logo, who, titulo, texto, estrellas)

sanPablo

¿Cómo integra los datos la clase sanPablo en un libro para mandárselo al servlet? A continuación lo explico con un diagrama de flujo:

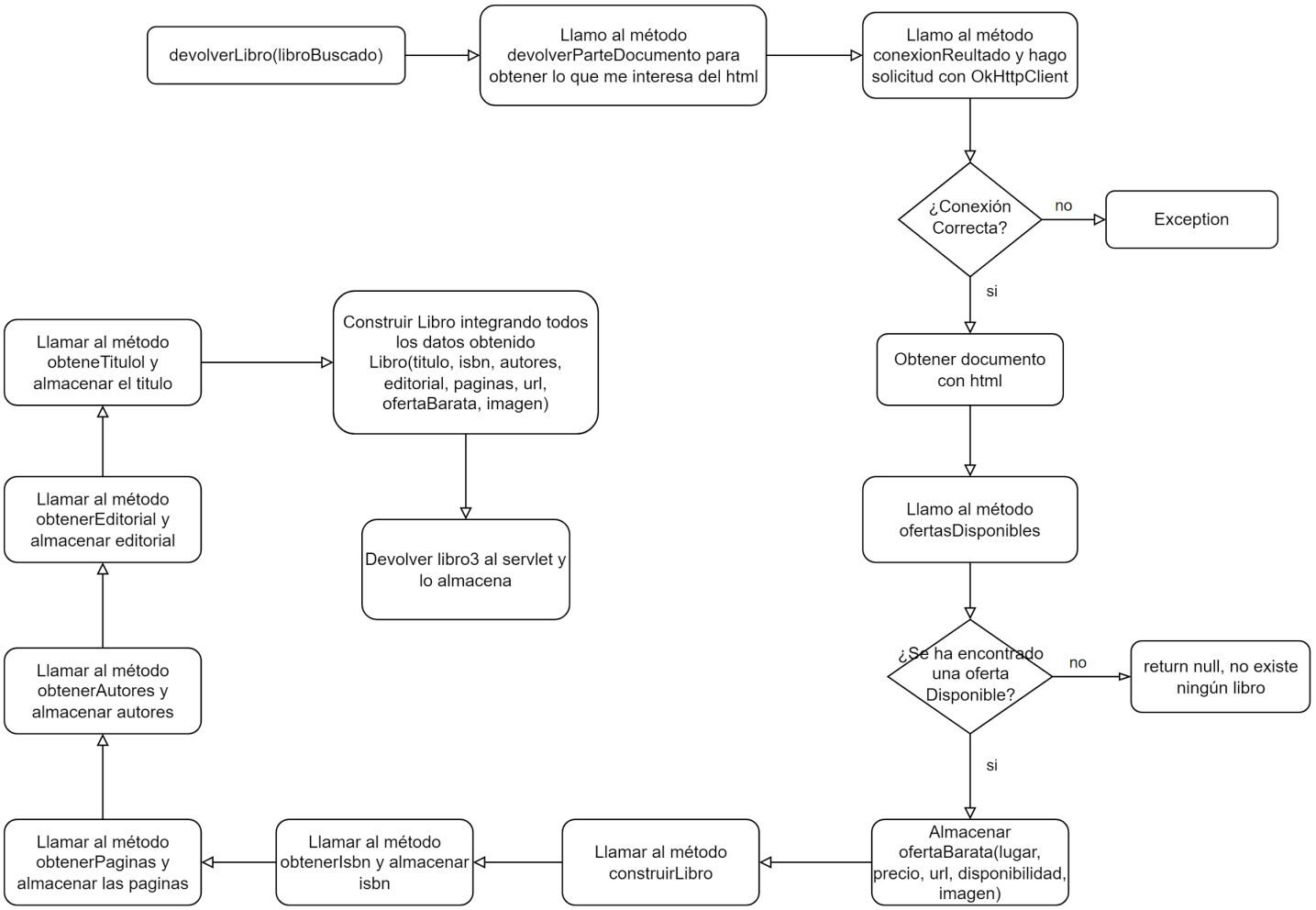


Por lo tanto, esta segunda fuente de datos devuelve un
Libro libro2(titulo, isbn, autores, editorial, fecha, paginas, url, imagen, oferta), que contiene

1) Oferta oferta(lugar, url, precio, disponibilidad)

ebay

¿Cómo integra los datos la clase ebay en un libro para mandárselo al servlet? A continuación lo explico con un diagrama de flujo:



Por lo tanto, esta segunda fuente de datos devuelve un
Libro libro3(titulo, isbn, autores, editorial, paginas, url, imagen, oferta), que contiene

1) Oferta oferta(lugar, url, precio, disponibilidad, imagen)

Correspondencias

- apiRain -> Libro, Oferta, Reseña

    - Libro libro1 = apiRain.devolverLibro(String aBuscar);
    - libro1.titulo -> Libro.titulo
    - libro1.isbn -> Libro.isbn
    - libro1.autores -> Libro.autores
    - libro1.editorial -> Libro.editorial
    - libro1.fecha -> Libro.fecha
    - libro1.paginas -> Libro.paginas
    - libro1.url -> Libro.url -> Oferta.url
    - libro1.imagen -> Libro.imagen
    - libro1.valoracion -> Libro.valoracion
    - libro1.oferta -> Libro.oferta
    - libro1.oferta.lugar -> Oferta.lugar
    - libro1.oferta.url -> Oferta.url
    - libro1.oferta.precio -> Oferta.precio
    - libro1.oferta.disponibilidad -> Oferta.disponibilidad
    - libro1.reseñas -> Libro.reseñas
    - libro1.reseñas.logo -> Reseñas.logo
    - libro1.reseñas.who -> Reseñas.who
    - libro1.reseñas.tituloReseña -> Reseñas.tituloReseña
    - libro1.reseñas.texto -> Reseñas.texto
    - libro1.reseñas.estrellas -> Reseñas.estrellas

- sanPablo-> Libro, Oferta

    - Libro libro2 = sanPablo.devolverLibro(String aBuscar);
    - libro2.titulo -> Libro.titulo
    - libro2.isbn -> Libro.isbn
    - libro2.autores -> Libro.autores
    - libro2.editorial -> Libro.editorial
    - libro2.fecha -> Libro.fecha
    - libro2.paginas -> Libro.paginas
    - libro2.url -> Libro.url -> Oferta.url
    - libro2.imagen -> Libro.imagen
    - libro2.oferta -> Libro.oferta
    - libro2.oferta.lugar -> Oferta.lugar
    - libro2.oferta.url -> Oferta.url
    - libro2.oferta.precio -> Oferta.precio
    - libro2.oferta.disponibilidad -> Oferta.disponibilidad

- ebay -> Libro, Oferta

    - Libro libro3 = ebay.devolverLibro(String aBuscar);
    - libro3.titulo -> Libro.titulo
    - libro3.isbn -> Libro.isbn
    - libro3.autores -> Libro.autores
    - libro3.editorial -> Libro.editorial
    - libro3.paginas -> Libro.paginas
    - libro3.url -> Libro.url -> Oferta.url
    - libro3.imagen -> Libro.imagen -> Oferta.imagen
    - libro3.oferta -> Libro.oferta
    - libro3.oferta.lugar -> Oferta.lugar
    - libro3.oferta.url -> Oferta.url
    - libro3.oferta.precio -> Oferta.precio
    - libro3.oferta.disponibilidad -> Oferta.disponibilidad
    - libro3.oferta.imagen -> Oferta.imagen