

3 Diseño.

3.1 Diseño.

El sistema de clasificaciones personales, SOPHIA, tiene lo siguiente como objetivos generales:

1. Permitir al usuario poder clasificar, bajo su criterio, los diferentes tipos de material contenidos en las distintas bibliotecas digitales.
2. Almacenar la clasificación hecha por cada usuario.
3. Crear una interfaz que para el usuario sea fácil de manejar.

Además los aspectos generales que se consideraron para SOPHIA son:

1. Debe ser general para aceptar cualquier acervo de archivos.
2. Debe ser independiente de plataforma.
3. Debe estar disponible por medio de un navegador web.

Debido a que este sistema es un proyecto de la biblioteca de la UDLA-P pertenece entonces al entorno de U-DL-A, la figura 8 mostrará en donde se encuentra ubicado.

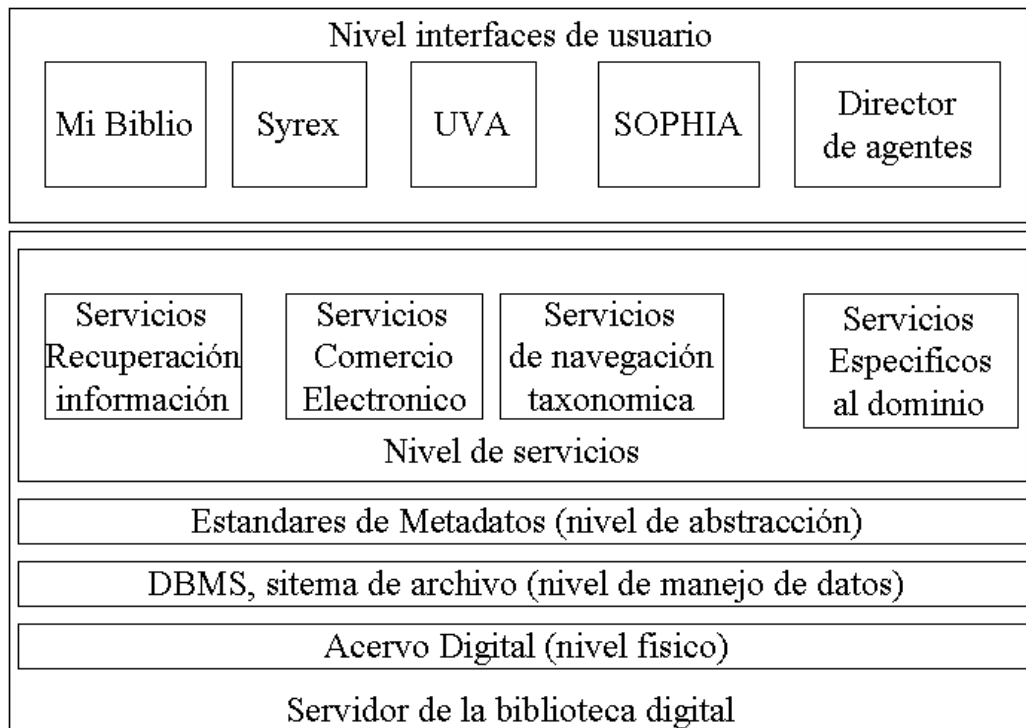


figura 8 Arquitectura U-DL-A

SOPHIA se encuentra en la parte de espacios personales en el nivel de interfaces del usuario.

3.2-Diseño del sistema.

3.2.1-Diseño de la interfaz.

Si bien la interfaz no es lo único que importa de un sistema si es uno de los aspectos más importantes, porque finalmente es con lo que el usuario va a interactuar. Además en

años recientes se ha demostrado que el 48% del código de un sistema está destinado para la interfaz del mismo [Nielsen 1993]. La implementación de una interfaz es un proceso parecido a la construcción de una casa, si no se tienen los cimientos adecuados después por más que se quiera adornar no se podrá mejorar [Raskin 2001].

Por lo tanto para el desarrollo de la interfaz de SOPHIA se tomaron algunas métricas importantes propuestas por expertos en la materia, estos aspectos son:

1. La interfaz debe tener textos propios para los usuarios, es decir evitar poner términos técnicos.
2. Reducir el gasto de memoria de los usuarios, es decir no poner pantallas donde el usuario tenga que memorizar o aprenderse algo para después utilizarlo.
3. Evitar en lo posible que el sistema tenga errores, si los llegara a tener que sean explícitos para el usuario por medio de mensajes de error.
4. Que el usuario tenga retroalimentación, es decir que el usuario sepa cuando el sistema está ocupado realizando una tarea y cuando esta libre para atenderlo.

Aparte de haber seguido con las medidas anteriores se realizaron pruebas(sección 3.2.1.1), con el sistema UVA, lo cual resultó de gran ayuda puesto que se pudieron observar desde un principio los alcances y limitaciones de la representación de información

por medio de un árbol de jerarquías. Por lo tanto en el desarrollo del sistema se tomaron las ventajas y se trataron de pulir las desventajas vistas por medio de las pruebas.

3.2.1.1 Pruebas con UVA

En el principio de todo proyecto de tesis se deben de tomar bases sólidas tanto para el desarrollo del documento como para el desarrollo del sistema. Esto se logra a través de investigación en libros, revistas, artículos y en software ya implementado. Para la creación de SOPHIA se tomó como plataforma de inicio a UVA, debido a la similitud de los objetivos que se querían alcanzar en SOPHIA con los resultados producidos con UVA.

UVA entonces fue el prototipo inicial del sistema. Con él se realizaron pruebas a cinco usuarios, entre ellos profesores y estudiantes de la UDLA. La prueba consistió de seis preguntas que se debían responder en un tiempo de doce minutos. A los usuarios del sistema se les dio una explicación de cinco minutos acerca de cómo manejar a UVA.

En el tiempo que los usuarios empezaban con sus tareas, se iba tomando nota de cada uno de los aspectos de su comportamiento; los usuarios novatos tardaban algún tiempo para poder manejar la interfaz y los expertos les bastó con escuchar las instrucciones.

Entre los resultados arrojados se encontró que:

?? El aspecto visualización en tres dimensiones dejaba a los individuos satisfechos, pues todos coincidieron en que es una buena forma de representar el contenido de la biblioteca.

?? En cambio surgió un pequeño malestar debido al tiempo de respuesta del sistema, pues este carga toda la rama de un árbol de una sola vez, es decir si el usuario explotaba el nodo filosofía el sistema traía todos los documentos bajo esta categoría.

Por las razones anteriores expuestas dentro de las secciones anteriores la interfaz en forma de “árbol” continuó dentro de SOPHIA, aunque con alguna variante, y también se agregó la opción de que el usuario escoja cuantos nodos quiere por rama. Finalmente se hace notoria la presencia de las pruebas realizadas en UVA, ya que como se mencionó se tomaron las ventajas y se trataron de suprimir las desventajas.

3.2.1.2 Por qué la interfaz en 2D.

Uno de los aspectos que se tienen que cuidar en la implementación de la interfaz es que sea entendible para la mayoría de los usuarios. Por lo tanto es indispensable basarse en pruebas y artículos realizados por expertos sobre el tema.

Una de estas prueba, realizadas por [Sebrechts et al 1999],consistió en manejar un sistema por medio de 3 interfaces diferentes, sólo texto, en 2D y 3D, los usuarios fueron divididos en dos grupos; novatos y expertos. Los resultados arrojaron las ventajas y desventajas de cada tipo de interfaz.

La interfaz de sólo texto fue la más fácil de comprender. El tiempo de ejecución y terminación promedio de cada una de las tareas fue el más bajo de las tres interfaces. La

verdadera desventaja se presenta cuando el número de documentos a buscar crece. En esta prueba no se dio el caso debido a que sólo se trabajó con 100 documentos.

El tiempo de respuesta con la interfaz en 2D fue constante y se mantuvo en promedio a la mitad de las tres. Los usuarios novatos no tuvieron mucho problema en adaptarse y los expertos la manejaban con facilidad. Tal vez la verdadera desventaja es que cuando se conocía mejor la interfaz el tiempo de terminación de las tareas no mejoró en mucho.

Con la última interfaz los problemas fueron para los usuarios novatos pues no se acostumbraban fácilmente al entorno, el problema con los usuarios expertos fue realmente el uso de la interfaz por medio del ratón, que según el mismo artículo se hubiera podido evitar utilizando un dispositivo llamado "spaceball". Otra desventaja fue que los novatos se perdían fácilmente en este ambiente y tenían que comenzar la tarea de nuevo. La mayor ventaja se presentó cuando los grupos aprendieron a manejar la interfaz, pues el tiempo de cumplimiento de las tareas bajó dramáticamente, pero aún así era el mayor de los tres.

Tomando los resultados de este artículo podemos ver que la interfaz que representa menos problemas para los dos grupos es la interfaz en 2D. Debido a que su manejo y comprensión es fácil para los dos grupos; novatos y expertos.

Algunos expertos mencionan que las interfaces en 3D se deben utilizar sólo en los campos que lo requieran, sería una buena opción por ejemplo usarlo en el mundo de la medicina o la química donde se puede ver un tumor o simular la consistencia de una

partícula. Si la representación en 3D no es la adecuada lo único que se logra es añadir más carga cognitiva al usuario [Nielsen 1998].

Para el buen manejo de una interfaz en 3D se necesitan aditamentos especiales como el mencionado anteriormente, “spaceball”, además de una buena resolución en pantalla, pues los objetos se encuentran en perspectiva y los que se encuentren en la parte de atrás no se podrán distinguir si la resolución no es la adecuada.

Finalmente la decisión de utilizar una interfaz en 2D (figura 9) se basó en los siguientes puntos:

1. La representación de grandes volúmenes de información es mejor que en interfaces de sólo texto.
2. El tiempo de aprendizaje por parte de los usuarios es mínimo. Siempre lo han manejado por ejemplo la interfaz en 2D del explorador de archivos en los diferentes sistemas operativos.
3. Relativamente es fácil de implementar.
4. Es menos complejo que las interfaces en 3D.
5. Y el tiempo de despliegado en pantalla es menor, debido a que los objetos son menos pesados.

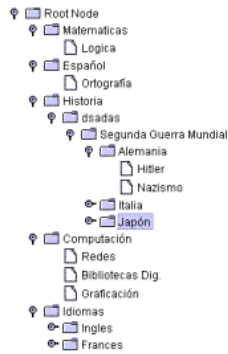


figura 9 Árbol en 2D

Debido a que en este proyecto debemos tener la facultad de ver los materiales incluidos en los diferentes repositorios y al mismo tiempo observar nuestra clasificación de acervo, se tomó la idea de que la interfaz consista de tres secciones (figura 10): una sección donde el usuario pueda observar el material que existe en los diferentes acervos, es decir, un árbol o lista de material; otra donde pueda almacenar sus clasificaciones; árbol de usuario y una final donde pueda modificar la estructura de su árbol de clasificación.

Dentro de las interfaces se implementan herramientas para el manejo de la misma estas herramientas pueden ser botones, barras de desplazamiento, cajas de herramientas y demás. Una de estas resultó ser interesante para el funcionamiento del sistema, ya que provee una forma intuitiva de manejo para los usuarios. Además de poder realizar, por medio de esta herramienta, trabajo de fondo en el sistema, por ejemplo almacenamiento en

la base datos, sin que el usuario gaste tiempo en realizar esta misma acción de otra manera. La comparación entre herramientas y la elección de una en particular se describirán en la sección 3.2.1.3

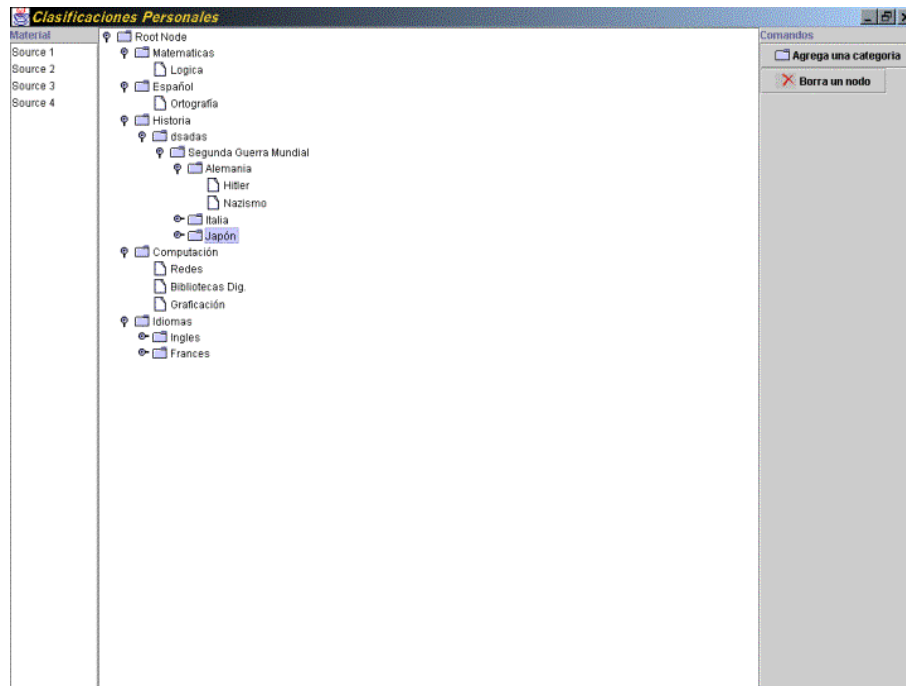


Figura 10 Propuesta inicial de Interfaz SOPHIA.

3.2.1.3 “Drag and Drop” (D&D).

D&D consiste en seleccionar un elemento y mantener oprimido el botón del ratón, “arrastrando” el objeto hasta soltar el botón “tirando” a este en alguna otra parte deseada de la pantalla [Rekimoto 1997].

Para que un usuario aprenda como debe de actuar ante cierta interfaz depende principalmente de la complejidad de sus herramientas. Diversos estudios sean realizado sobre el tema, aportando detalles importantes.

Una de estas comparaciones es realizada en [Beaudouin 2000]. Se midieron diferentes aspectos propuestos por el autor para demostrar que herramientas tienen ventajas sobre otras y en qué sentido.

Las características medidas fueron:

- ?? Grado de indirección. Es la medida dada en dos dimensiones en el campo espacio-temporal. Entendiéndose por espacio la distancia, en pantalla, entre la herramienta utilizada y el objeto sobre el cual va actuar y por temporal el tiempo que tarda la herramienta en realizar la acción sobre el objeto.
- ?? Grado de Integración. Es la medida de libertad de movimiento que se da entre la herramienta y el dispositivo de entrada. Por ejemplo una barra de desplazamiento es una herramienta en 1D controlada por un dispositivo 2D, como lo es el ratón, entonces el grado de integración es de $1 / 2$.
- ?? Grado de compatibilidad. Mide la similitud de las acciones físicas del usuario sobre la herramienta y la respuesta de esta sobre la interfaz. Por ejemplo la similitud del movimiento de un objeto cuando se arrastra y el del ratón en la mano del usuario.

Los instrumentos medidos fueron entre otros: barras de desplazamiento, menús, barras de herramientas, teclas llave, d&d y algunos más. Los resultados que nos interesan por supuesto son los de “*drag and drop*”, los cuales salieron positivos, debido a que las medidas del D&D salieron mejor en la mayoría de las comparaciones.

En el caso de grado de indirección, las ventajas que ofrece D&D, en contra de una barra de herramientas, son grandes, pues el objeto arrastrado seguirá el mismo movimiento del ratón en la mano del usuario y la acción que se realice de fondo será casi inmediata. En cambio el uso de la barra de herramientas supone que el objeto se encuentre a cierta distancia de esta, además las acciones no siempre son inmediatas.

El grado de integración en D&D es uno de los mejores debido a que la herramienta tiene movimiento en 2D y el ratón también tiene movimiento en ese plano, es decir el grado de integración es igual a 1, mucho mejor que $1 / 2$ que como ya se mencionó es la medida de una barra de desplazamiento.

En el nivel compatibilidad se vuelve a encontrar al *drag and drop* entre las herramientas con mejor calificación debido a que el movimiento del dispositivo de entrada es bastante similar al movimiento del objeto en pantalla.

Por último la elección de la función de D&D, se debió a las razones expuestas anteriormente, además de ser considerado una herramienta familiar para los usuarios tanto expertos como novatos, esto se debe a que en la actualidad la mayoría de los paquetes para cualquier plataforma lo utiliza.

3.2.1.4 Retroalimentación

Todos los sistemas consumen tiempo para realizar los procesos que demandan los usuarios. El tiempo que se tarde en responder el programa no debe sobre pasar ciertos limites, puesto implicaría desesperación y muy posiblemente el abandono del sistema por parte del usuario.

Para evitar que el usuario pierda el interés en manipular un sistema, a causa del tiempo de espera, se le tiene que ofrecer retro-alimentación. El sistema tiene que avisarle a la persona cuando esta libre para atenderla y cuando se encuentra ocupado realizando algún proceso.

Los rangos estimados van desde 0.1 hasta 10 segundos [Nielsen 1993], en los umbrales de respuesta de 0.1 hasta 1 segundo la retro-alimentación no es necesaria debido a que el sistema responderá lo suficientemente rápido. Entre 1 y 10 segundos la retro-alimentación se vuelve indispensable, debido a que el usuario tiene que estar informado de que es lo que sucede en ese tiempo, sino sucediera así la persona podría dejar de utilizar el programa.

3.3-Almacenamiento de datos

Debido a que las clasificaciones de los usuarios, deben poder ser recuperadas se tiene que desarrollar un método que lo realice. Existen diversas formas que van desde guardar los datos en archivos hasta utilizar una base de datos. Dentro de SOPHIA se decidió que el almacenamiento de los datos fuese a través de una base, debido a que presenta las siguientes ventajas[Hogan 1990]:

1. Soporta relaciones complejas de datos.
2. Provee de mecanismos seguros para acceder a los datos.
3. Provee mecanismos sofisticados para recuperación y respaldo.
4. Tiene la capacidad de poder ser consultada en línea.

Para la buena implementación de una base de datos dentro de un sistema, se debe diseñar un modelo de datos llamado entidad-relación. El modelo de datos para la base de este sistema tiene características de lo desarrollado en Mutant [Flores, 1997]. Este modelo permite guardar de una manera eficiente las clasificaciones hechas por los usuarios, es decir se almacena el mínimo de información y se le da el máximo rendimiento al usuario. El esquema entidad-relación utilizado en SOPHIA es mostrado en la figura 11.

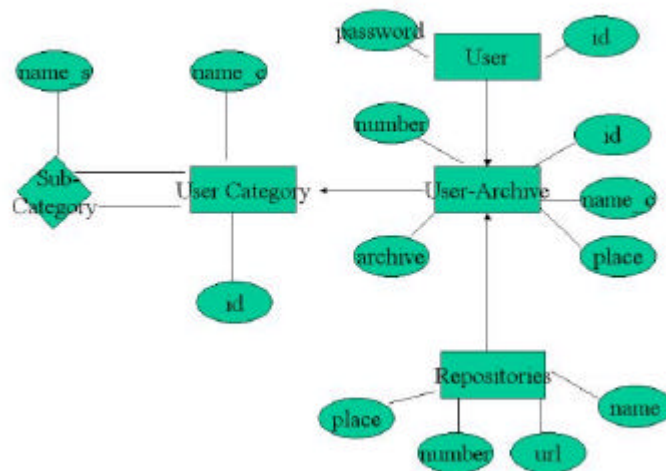


figura 11 Esquema entidad relación de SOPHIA.

Como se observa el modelo de datos es simple, pero si pensamos en que el sistema se conecta con otras bases de datos o repositorios entonces este modelo se vuelve más complejo

3.4 Arquitectura de SOPHIA.

La arquitectura de SOPHIA es en cierto grado compleja pues el sistema interactúa no solo con la base de datos para almacenar la información también lo hace con los demás repositorios que se encuentran talvez fuera de nuestro país, como es el caso del repositorio de la universidad de Virginia o el repositorio de la NASA, los dos en Estados Unidos. La figura 12 muestra cual es la arquitectura del sistema.

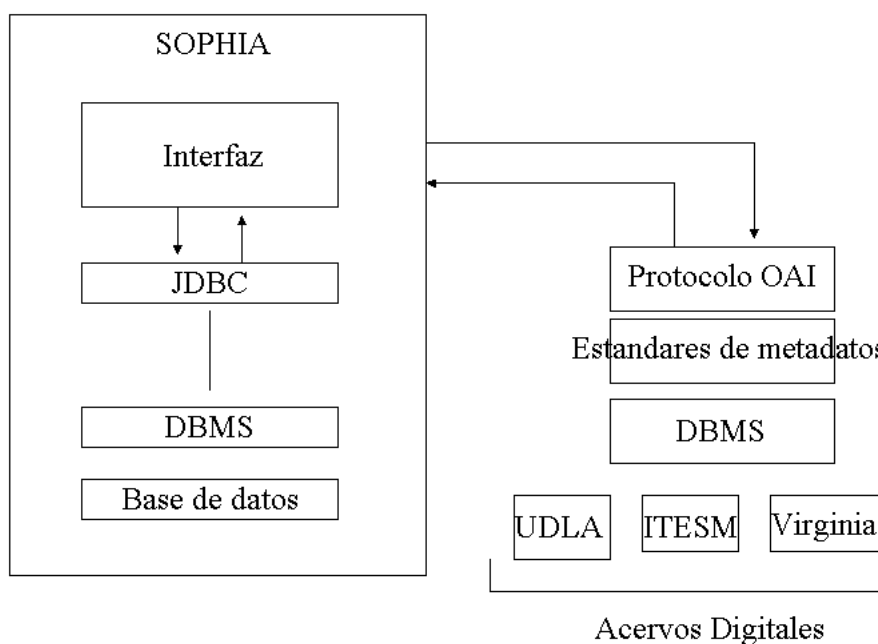


figura 12 Arquitectura de SOPHIA

SOPHIA almacena la información de los usuarios en la base de datos de una manera normal aprovechando las ventajas del JDBC, aquí es donde el sistema guarda las modificaciones de cada una de las clasificaciones. Como mencionamos lo hace utilizando el mínimo espacio posible.

Ahora bien el sistema hace peticiones a todos los repositorios de datos con el protocolo OAI, mencionado en el capítulo anterior. En este paso los estándares ya están predefinidos, por lo que se tomó como base lo desarrollado en MAIDL [Nava 2001], mencionado en el capítulo anterior, para lograr una comunicación entre los diferentes almacenes de datos y el sistema. Todos los archivos se encuentran bajo un mismo esquema

de metadatos, lo cual simplifica la tarea de encontrar datos comunes dentro de ellos. Por ejemplo en todos los documentos se encuentra al dato titulo bajo el metadato “title”.

Tomando los aspectos aquí mencionados como una base en el siguiente capítulo se explican cada uno de los detalles de la implementación del sistema.