

Unidad II

Metodología de Solución de Problemas

2.1 Descripción del problema (enunciado).

Este aspecto nos indica describir de manera objetiva la realidad del problema que se está investigando. En la descripción se señalan todas las características de la problemática, los hechos y los acontecimientos que están en entorno social, al mismo tiempo se debe mencionar los antecedentes del problema.

- Antecedentes del estudio o problema de investigación.
- Las técnicas en las que se basó, las categorías de análisis o ejes centrales que permiten guiar el proceso de investigación.
- Los supuestos básicos en los que se apoya el enunciado del problema.

Un enunciado completo del problema incluye todos los hechos, relaciones y explicaciones que sean importantes en la investigación.

Hay que encuadrarlos en un enunciado descriptivo o en una pregunta que indique con claridad qué información ha de obtener el investigador para resolver el problema de investigación.

2.2 Definición de solución (especificaciones).

Diariamente es necesario enfrentar problemas y conflictos a los cuales se les deben encontrar soluciones aceptables de acuerdo al contexto. El proceso de solucionar problemas implica una serie de capacidades y habilidades del pensamiento que es importante desarrollar y evaluar en la preparación académica.

La resolución de problemas es una actividad cognitiva que consiste en proporcionar una respuesta-producto a partir de un objeto o de una situación.

Una de las capacidades más importantes en la resolución de problemas es la de hacer preguntas que permitan surgir de un conflicto y sortear la dificultad, algunas preguntas pueden servir para identificar el problema, otras para buscar alternativas, etc. Es posible preguntarse: ¿qué es lo que hace problemática esta situación? ¿qué me falta por saber? ¿cuántos problemas están involucrados? ¿cuál voy a intentar resolver? ¿qué es lo que no funciona? ¿cuáles son las alternativas que se pueden tomar? ¿qué conozco sobre este tema? ¿por dónde puedo empezar para que sea más fácil? etc.

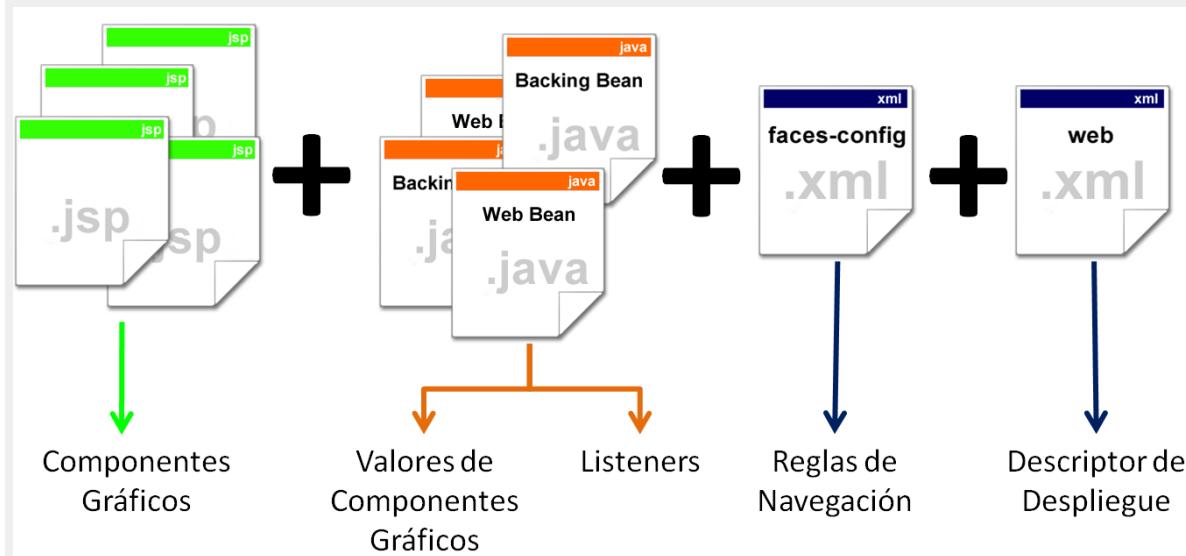
2.3 Diseño de la solución (modelado).

La solución que se plantea en el tutorial cubre los componentes web necesarios para el problema del Colegio.

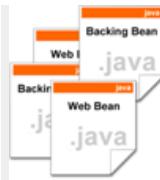
El modelo de control del marco de trabajo JSF se puede describir como un tipo de patrón MVC en donde existen controladores asociados a los elementos gráficos y éstos interactúan con el modelo de datos o del mundo para actualizar la presentación de dichos elementos.

Para el diseño de la capa web de una aplicación JSF debemos modelar los elementos que actuarán como controladores de las acciones (clases java que representan las entidades web), las páginas de presentación y las reglas de navegación del sistema (archivos de configuración que enlazan las capas).

Una aplicación JSF tiene como mínimo los siguientes componentes:



 Las páginas de interacción JSPs hacen parte de la tecnología Java Server Pages que permite utilizar HTML generado dinámicamente. Esta tecnología extiende el modelo de tecnología de Servlets. En el tutorial, al mostrar los resultados de las búsquedas, el tamaño de la tabla varía de acuerdo con el criterio de búsqueda de búsqueda de estudiantes. Es una alternativa para crear servlets de manera más sencilla. El lenguaje se basa en tags y existen librerías que proporcionan la implementación de páginas JSP incluyendo las librerías JSF para poder referenciar los elementos.



Las clases de Java Beans, dentro del modelo MVC, son las clases que actuarán como controladores de las acciones. Podemos hacer una analogía entre el modelo MVC de swing y de JSF pues éstos controladores escuchan eventos de la vista e interactúan con el modelo para dar respuesta a las solicitudes. Cada uno de estos beans debe representar una entidad del mundo que quiere ser presentada de manera que el bean (o entidad web) contienen información que se puede presentar en las páginas jsp. En el tutorial se diseñaron dos beans, uno que representa un estudiante y otro que representa una lista de estudiantes.

Todos los beans que se asocien con elementos gráficos deben tener métodos get/set de sus atributos (utilizando los prefijos en inglés) pues se enlazan directamente con el nombre dado que el framework interpretará la página y buscará el método get/set correspondiente.



El archivo faces-config.xml define la navegación, los beans de java que utilizarán JSF y los archivos de propiedades para internacionalización. La navegación se define a partir de un punto de partida que corresponde a una acción y un punto de llegada que es una visualización del resultado de la acción y corresponde a una página jsp. La configuración de los beans establece el nombre con el que se invocan sus acciones en las páginas jsp y la ruta completa del paquete en el que se encuentra la clase que representa el backing bean.

JSF se encuentra desarrollado por arriba de los servetls, de manera que toda aplicación JSF es JSP/servlet donde cada página contiene componentes web JSF que encapsulan HTML. Es así como cada petición HTTP que requiere JSF pasa por un servlet de Faces, el cual hace parte del modelo de tecnología de este marco de trabajo.



El archivo web.xml es una modificación del web.xml estándar para indicar el tipo de solicitudes que serán procesadas por el servlet de JSF. El servlet hará la intercepción de todas las peticiones antes de invocar la página respectiva a la que le prepara un contexto en el servidor de aplicaciones para la información asociada a este.



Los archivos de propiedades para internacionalización contienen la información de configuración del texto y los mensajes en cada idioma. Las páginas se construyen utilizando etiquetas que hacen referencia a las llaves de estos archivos, una vez se procesan las solicitudes y se conoce el idioma en el que se quiere construir la respuesta, se reemplazan estas etiquetas por el valor obtenido del archivo correspondiente. Cambiando la configuración de idioma del explorador de internet, se utilizará un archivo o el otro.

De acuerdo con lo anterior, el diseño web del colegio consiste en el modelo de clases con los beans que enlazarán capas y el diseño de la navegación a través de diagramas de estado encontrados en la siguiente sección.

2.4 Desarrollo de la solución (codificación).

El aspecto más importante de cualquier propuesta es identificar y comprender el problema que el cliente busca resolver”.

Uno de los puntos del desarrollo de una propuesta de solución es presentar una noción propia del problema, así como la propuesta para resolverlo, con el fin de convencer al cliente de que tal propuesta es la mejor.

Para ello, se presentará lo que implica una descripción de los problemas:

- ▶ La naturaleza del problema.
- ▶ La historia del problema.
- ▶ Las características de la solución óptima.
- ▶ Las soluciones alternas consideradas.
- ▶ La solución o la técnica seleccionada.

2.5 Depuración y pruebas (pruebas).

La depuración es el proceso de encontrar los errores del programa y corregir o eliminar dichos errores.

Cuando se ejecuta un programa se pueden producir tres tipos de errores:

- ✖ Errores de compilación. Se producen normalmente por un uso incorrecto de las reglas del lenguaje de programación y suelen ser errores de sintaxis. Si existe un error de sintaxis, la computadora ni puede comprender la instrucción, no se obtendrá el programa objeto y el compilador imprimirá una lista de todos los errores encontrados durante la compilación.
- ✖ Errores de ejecución. Estos errores se producen por instrucciones que la computadora puede comprender pero no ejecutar. Ejemplos típicos son: división entre cero y raíces cuadradas de números negativos. En estos casos se detiene la ejecución del programa y se imprime un mensaje de error.
- ✖ Errores de lógica. Se producen en la lógica del programa y la fuente del error suele ser el diseño del algoritmo. Estos errores son los más difíciles de detectar, ya que el programa puede funcionar y no producir errores de compilación ni de ejecución, y sólo puede advertir el error por la obtención de resultados incorrectos. En este caso se debe volver a la fase de diseño del algoritmo, modificar el algoritmo, cambiar el programa fuente y compilar y ejecutar una vez más.

2.6 Documentación (manuales).

En sentido restringido, la **documentación** como ciencia documental se podría definir (a grandes rasgos) como la ciencia del procesamiento de la información, que proporciona información sobre algo con un fin determinado, de ámbito multidisciplinar o interdisciplinar.

Siguiendo a Fuentes y Pujol se puede señalar a la Documentación como una ciencia auxiliar e instrumental. También es una ciencia en si misma y una de las finalidades primordiales de la Documentación es informar.

A falta de un consenso, hay diversos autores, como Juan Ros García o José López Yepes, que la consideran una ciencia (documental), a la vez que una disciplina, no sólo una técnica. También pueden considerarse, en sentido general, las ciencias de la documentación y la documentación como sinónimos, si el contexto no perturba la intención del emisor, es decir, si no se distorsiona el mensaje del interlocutor porque no se dé ambigüedad semántica