

## **DISEÑO**

El diseño no es tarea fácil. El fabricante quiere algo que se pueda producir económicamente. La tienda quiere algo que resulte atractivo para los clientes. El comprador tiene varias exigencias. En la tienda, el comprador se centra en el precio y el aspecto, y quizá en el valor de prestigio. En casa, esa misma persona prestará más atención a la funcionalidad y la capacidad de uso. Al servicio de reparaciones le preocupa la mantenibilidad: ¿hasta qué punto es fácil desmontar, diagnosticar y reparar el dispositivo? Las necesidades de quienes intervienen suelen ser diferentes y conflictivas. El diseño es un proceso inherentemente creativo e impredecible. Los diseñadores de sistemas interactivos deben combinar un completo conocimiento de factibilidad tecnológica con un sentido casi místico estético de lo que atrae a un usuario [Carrol & Rosson, 1985]. El diseño es un proceso, no es un estado y no puede representarse estáticamente, es un proceso no-jerárquico, no es estrictamente bottom-up ni top-down. El proceso es transformativo, incluye el desarrollo de soluciones parciales e incluye el descubrimiento de nuevas metas.

El diseño centrado en el usuario es un método iterativo donde se enfatiza el involucrar de forma activa al usuario, y el entendimiento claro de los requerimientos de la tarea y el usuario. De acuerdo con la norma ISO 13407, la incorporación del diseño centrado en el usuario se caracteriza por involucrar activamente a los usuarios y entender claramente los requerimientos del usuario y la tarea, una apropiada distribución de las funciones entre los usuarios y la tecnología, la iteración de las soluciones de diseño y el diseño multidisciplinario.

Es esencial involucrar al usuario para poder entender y definir el contexto de uso, las tareas y la forma en que van a trabajar los usuarios en el futuro con el producto o sistema. Los usuarios y desarrolladores deben interactuar a través del proceso de diseño; cuando se diseñan productos hechos a la medida del usuario, los usuarios reales pueden directamente involucrarse en el proceso de diseño, y para el diseño de productos de consumo se puede involucrar a representantes del grupo objetivo. El objetivo de la tecnología es asistir al usuario a desarrollar las tareas seleccionadas, por lo que el diseño debe identificar todas las tareas, la forma en que serán desarrolladas y cómo serán repartidas entre el usuario y la tecnología, decisión que no puede basarse únicamente en las capacidades de la tecnología y dejar las demás tareas al ser humano.

En el diseño interactivo, la retroalimentación de los usuarios es una importante fuente de información. Por lo general, no es posible definir exactamente las necesidades del usuario al inicio del proceso, ya que por una parte, el diseñador no tiene una clara idea de lo que el usuario pueda querer, y por la otra, el usuario no tiene una idea clara de lo que la tecnología pueda hacer.

El diseño centrado en el usuario requiere conjuntar una variedad de habilidades y conocimientos dependiendo de la naturaleza del sistema a desarrollar, por lo que el equipo multidisciplinario puede incluir usuarios finales, miembros de la gerencia, expertos en la aplicación, diseñadores del sistema, expertos en mercadotecnia, diseñadores gráficos, especialistas en factores humanos y personal de capacitación. Es posible que una misma persona represente varias de estas áreas, pero algo importante a considerar es que el diseñador nunca puede representar al usuario, a menos que el diseño sea desarrollado para su uso personal.

## **METÁFORAS**

El término metáfora está tradicionalmente asociado con el uso del lenguaje, se utiliza cuando se desea comunicar un concepto abstracto de una manera más familiar y accesible. En el diseño de las interfaces, las metáforas tienen un papel dominante, el uso de éstas puede asistir a los desarrolladores en conseguir maneras más eficientes y efectivas de desarrollar programas que permitan ser usados por comunidades de usuarios más diversos.

### **Metáforas verbales**

Las metáforas son una parte integrante del lenguaje y del pensamiento, aparecen en las conversaciones cotidianas. Cuando alguien se encuentra frente a una nueva herramienta tecnológica (como una computadora) se tiende a compararlo con alguna cosa conocida. Un ejemplo clásico es cuando se utiliza una computadora por primera vez (en edad adulta), se compara inmediatamente con una máquina de escribir de una manera metafórica. Estos enlaces metafóricos proporcionan los fundamentos por los cuales el usuario desarrolla su modelo propio de computadora. Los conocimientos de los elementos y sus relaciones en un dominio familiar se traspasan a los elementos y sus relaciones en un dominio no familiar. Análogamente, se pueden aplicar otras metáforas, por ejemplo, explicar el funcionamiento de los archivos

con la metáfora del funcionamiento de un archivo físico. En general, las metáforas verbales son herramientas útiles para ayudar a los usuarios a iniciarse en el aprendizaje de un nuevo sistema.

### **Metáforas visuales**

Xerox fue una de las primeras empresas que se dio cuenta de la importancia de diseñar interfaces simulando el mundo físico concreto que nos es familiar. En vez de pensar, en interfaces verbales, como medio de hacer entender a los usuarios la interfaz, fue un poco más lejos y desarrolló una interfaz visual basada en la oficina física, esta interfaz fue desarrollada para el sistema *Star*, la base de la metáfora consistió en crear objetos electrónicos simulando los objetos físicos en una oficina. Ésta incluía papel, carpetas, bandejas y archivadores, la metáfora de organización global presentada por pantalla fue la sobremesa y el área de trabajo se parecía a una típica mesa de oficina. Los archivos fueron transformados en representaciones pictóricas en sustitución de entidades abstractas con nombres arbitrarios. Esta metáfora fue aplicada a la computadora *Lisa* de Apple y más tarde al Macintosh, posteriormente fue aplicada a las computadoras personales.

La metáfora visual normalmente es una imagen que permite representar alguna cosa, en la cual el usuario puede reconocer lo que representa y por extensión puede comprender el significado que recubre. Las metáforas pueden variar de pequeñas imágenes puestas en botones de barras hasta pantallas completas en algunos programas. Las personas entienden las metáforas por intuición, se comprende el significado de los controles que se encuentran en la interfaz, porque se conecta mentalmente con otros procesos previamente aprendidos. Las metáforas se basan en asociaciones percibidas de manera similar por el diseñador y el usuario, si el usuario no tiene la misma base cultural que el desarrollador, es fácil que la metáfora falle, incluso teniendo la misma base cultural puede haber faltas de comprensión importantes.

### **Metáfora global**

La metáfora global es una metáfora que proporciona el marco para todas las otras metáforas de sistema. Por ejemplo, la metáfora del escritorio, desarrollada originalmente para la computadora Xerox *Star* y generalizada para la computadora Apple de Macintosh se puede considerar como un primer ejemplo de amplia difusión de metáfora global. La metáfora del escritorio (mencionada arriba) ha sido combinada con otras metáforas para permitir que los usuarios puedan ejecutar un amplio abanico de tareas en la computadora. Un ejemplo, es la barra de desplazamiento, ésta es una metáfora basada en la idea del rollo, es decir, como un papiro que se desplaza para leer. Otros ejemplos de metáforas son: el menú, las ventanas, el cortar y pegar, etc. Desde el punto de vista cognitivo, uno de los problemas que se puede presentar es cómo interpretar estas metáforas compuestas, se ha comprobado que el usuario desarrolla modelos mentales múltiples de la interfaz.

### **Metodología de creación de metáforas**

Las metáforas visuales son un aspecto importante del diseño de un sistema interactivo con interfaz visual y evidentemente ha de formar parte del diseño, por lo tanto aunque a continuación se describe la metodología de diseño de metáforas, siempre se ha de tener en cuenta su relación con el diseño global.

**Definición funcional.** En esta primera fase se parte del trabajo realizado en la recogida de requisitos, lo cual permitirá disponer de los primeros datos para establecer las primeras metáforas o del análisis de tareas en que se puedan precisar las funcionalidades del sistema.

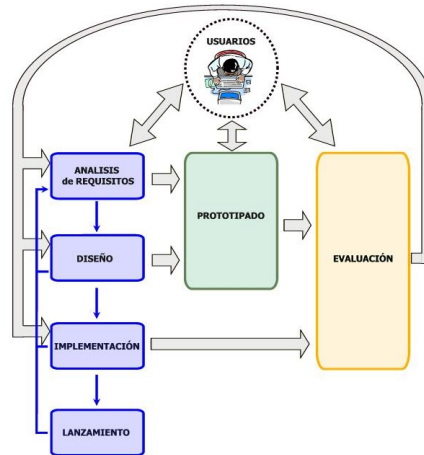
**Identificación de los problemas del usuario.** En esta etapa se hace un estudio de los usuarios para ver en qué tienen problemas y qué aspectos de la funcionalidad les implican. La mejor manera es ver a los usuarios utilizando funcionalidades similares y ver qué problemas tienen. Explicar lo que quieren hacer y ver si lo entienden, enseñándoles el prototipo y observando la forma en que lo utilizan.

**Generación de la metáfora.** Una buena forma de iniciar esta tarea es hacer un examen cuidadoso de la manera tradicional de realizar la tarea. Hay que disponer de tiempo para observar los problemas que los usuarios tienen así como las herramientas que utilizan para resolverlos en su trabajo diario. Una vez identificados los problemas y las herramientas que utilizan, ver cuáles de ellas envuelven algunas características que los usuarios encuentran difíciles de comprender, estas son buenas candidatas para nuevas metáforas.

**Evaluación de la metáfora.** Una vez que se han generado varias metáforas, es el momento de evaluar cuál escoger para expresar la nueva funcionalidad. Lo que se debe tomar en cuenta es: Representatividad, es decir, ¿es la metáfora fácil de representar?, las metáforas ideales tienen representación visual, auditiva y palabras asociadas. Adaptabilidad a la audiencia, los usuarios tienen que entender la metáfora, porque aunque cumpla los otros criterios no nos sirve. Extensibilidad, ¿qué más pueden hacer las metáforas?, una metáfora puede tener otras partes de la estructura que pueden ser útiles más adelante.

## INGENIERÍA DE LA INTERFAZ

En el desarrollo de sistemas interactivos se pueden aplicar técnicas de Ingeniería del Software, pero modificando algunos aspectos de los métodos de diseño clásico para adaptarlos a estos sistemas, algunos de los aspectos a considerar son: Captura de requisitos de interacción, análisis de tareas, realización de prototipos así como evaluación.



Un diseño centrado en el usuario requiere de una continua evaluación del producto a desarrollar. Por este motivo, cobran gran importancia los siguientes aspectos:

**Métodos formales:** Permiten una especificación precisa y sin ambigüedad del diseño a generar. Permite una verificación formal de propiedades y en algunos casos se puede generar la implementación automáticamente.

**Prototipado:** Los prototipos son documentos, diseños o sistemas que simulan o tienen implementadas partes del sistema final. El prototipo es una herramienta muy útil para hacer participar al usuario en el desarrollo y poder evaluar el producto en las primeras fases del diseño (modelo del ciclo de vida basado en prototipos). Un prototipo puede ser cualquier cosa que represente una idea básica del sistema o aparato que deseamos. Puede ser desde un guión o diagrama en papel de una pequeña pieza de un software complejo, una maqueta de cartón hasta un moldeado o prensado de la pieza de metal del aparato. Prototipos de baja-fidelidad, son aquellos cuya finalidad es de exploración. No se parecen al producto final. Son baratos, fáciles de hacer y rápidos de obtener.

### **Prototipos**

Los prototipos son documentos, diseños o sistemas que simulan o tienen implementadas partes del sistema final. El prototipo es una herramienta muy útil para hacer participar al usuario en el desarrollo y poder evaluar el producto en las primeras fases del desarrollo. La razón principal del uso de los prototipos es la reducción en coste y tiempo que supone su uso en la implementación del futuro sistema, esta reducción se puede conseguir o bien reduciendo el número de características o bien reduciendo el nivel de implementación de las funcionalidades de las características

### **Escenarios**

Los escenarios son historias sobre personas y sus actividades. Los escenarios destacan objetivos sugeridos por la apariencia y comportamiento del sistema; qué es lo que las personas quieren hacer con el sistema; qué procedimientos se usan, cuáles no se usan, se realizan o no satisfactoriamente y qué interpretaciones hacen de lo que les sucede. Los escenarios tienen un diagrama que incluye secuencias de acciones y eventos, cosas que hacen los actores, cosas que les suceden, cambios en las circunstancias de la configuración y otras. Las representaciones de escenarios pueden ser elaboradas como prototipos, a través del uso de storyboards, videos y herramientas de prototipado rápido. Los escenarios son un poderoso mecanismo para la comunicación entre los analistas del sistema y los usuarios. Los escenarios pueden ser usados para explicar las situaciones de trabajo, pueden ser utilizados para formular propuestas o imaginar situaciones que ayuden en el diseño conceptual.

## Storyboard

Un storyboard es una narración gráfica de una historia en cuadros consecutivos, se puede utilizar este concepto que se utiliza en el diseño cinematográfico, teatro, etc. para la realización de un escenario de interacción que puede ser evaluado con diferentes técnicas. Una de las opciones que se tienen en un storyboard para una aplicación es que se pueden indicar los enlaces a diferentes páginas del storyboard a partir de los resultados de las interacciones del usuario.

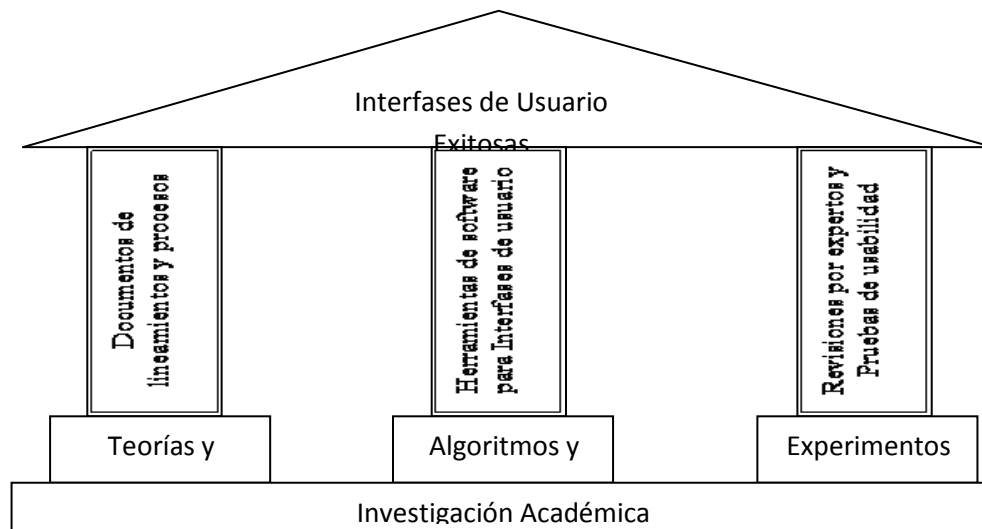
## Prototipo de papel

Este tipo de prototipo se basa en la utilización de papel, tijeras, lápiz o instrumentos que se puedan utilizar para describir un diseño en un papel. Este sistema nos permite una gran velocidad y flexibilidad. Para poder simular las diferentes interacciones que se van a realizar con el sistema, se realiza una hoja para cada uno de los diferentes escenarios que se vayan a tener, apilando dichas hojas que permitirán simular la aplicación.

Para utilizar el prototipo de papel se sitúa un escenario de uso de futuro en el que el diseñador actúa como coordinador. El prototipo será analizado por un posible usuario e intentará realizar algunas de las tareas que se pretende diseñar. En voz alta se irán realizando las interacciones y se irán cambiando las hojas de papel en función de las interacciones que se vayan realizando. El coste es muy reducido, necesitando únicamente los recursos humanos dedicados a la realización del prototipo. Los cambios se pueden realizar muy rápidamente y sobre la marcha. Si el diseño no funciona se pueden reescribir las hojas erróneas o rediseñarlas y volver a probar la tarea a realizar. Los usuarios o los actores se sienten más cómodos para poder realizar críticas al diseño debido a la sencillez del mismo por lo que no se sienten cohibidos a dar sus opiniones.

Una buena interfaz se vuelve invisible al momento de interactuar con ella. Nos damos cuenta de su existencia cuando los errores se hacen presentes al sentimos perdidos, ingenuos y con la situación fuera de control.

Para lograr una buena interfaz, existen:



Para un diseño exitoso se pueden encontrar:

**Teorías de alto nivel:** un marco de referencia y un lenguaje para discutir aspectos independientes de la aplicación, explicar el comportamiento de una interfaz o predecirlo, entre ellos tenemos: Modelo conceptual -semántico- sintáctico- léxico, GOMS, modelo de etapas de acción, consistencia a través de gramáticas, modelo OAI.

**Principios:** Son objetivos generales que pueden ser útiles para organizar el diseño, aconsejan al diseñador cómo debe proceder, sin embargo, no se especifican métodos para obtener esos objetivos, y está limitado al uso práctico (por ejemplo: conocer al usuario, minimizar el esfuerzo para realizar una tarea, mantener la consistencia, etc.).

**Guías prácticas:** reglas y recordatorios útiles en el proceso de diseño de interfaces.

## Modelo conceptual semántico-sintáctico-léxico

El nivel conceptual es el modelo mental del sistema interactivo; el nivel semántico corresponde al significado de la entrada por el usuario y la salida de la computadora; el nivel sintáctico es la forma en que las unidades semánticas se agrupan en instrucciones para la computadora y finalmente el nivel léxico son los mecanismos mediante los que el usuario especifica la sintaxis. Las ventajas de este modelo es que es comprensible, en forma top-down y describe un proceso de diseño.

## Metas, operadores, métodos y selección de reglas (GOMS)

Los usuarios formulan metas y sub-metas, cada una de las cuales se logra usando métodos y procedimientos. Los operadores son “actos perceptuales, motores o cognitivos elementales cuya ejecución es necesaria para cambiar algún aspecto del estado mental del usuario o afectar al entorno de la tarea”. Las reglas de selección son las estructuras de control para elegir entre los diversos métodos disponibles para completar un objetivo.

**KEYSTROKE- LEVEL MODEL:** tiempos de realización para tareas expertas libres de errores sumando el tiempo de pulsación, apuntamiento, localización, dibujado, razonamiento y espera por la respuesta de la máquina. Intenta predecir el tiempo requerido por un usuario para realizar una tarea usando una interfaz dada, tiene variantes para usuarios expertos, curvas de aprendizaje, diferentes márgenes de error, etc. Sirve como apoyo en estudios de usuarios para creación de manuales y ayuda en línea.

Ejemplo: Un usuario requiere con frecuencia convertir mediciones de temperatura expresadas en grados centígrados a Fahrenheit y viceversa, para ello se cuenta con la siguiente interfaz:

Se realiza una tabla donde se colocan las acciones o métodos necesarios para llevar a cabo la tarea, se estima el tiempo para cada acción (esto se hace eligiendo usuarios de todos los tipos y promediando el tiempo que tardan en realizar “x” acción), se omiten aquellas acciones que no dependen del usuario. A continuación se presenta la tabla para el ejemplo que se está explicando.

ACCIÓN	SÍMBOLO	TIEMPO
Un teclazo	K	0.2 seg.
Apuntar con el cursor	P	1.1. seg.
Mover la mano del teclado al ratón, joystick, etc. (homing)	H	0.4 seg.
Preparación mental para el siguiente paso.	M	1.35 seg.
Espera por respuesta de la máquina.	R	variable

Las suposiciones que se hacen son las siguientes: el promedio de caracteres es de 4 incluyendo signo y punto decimal, se requieren ambas conversiones ( $^{\circ}\text{C}$  a  $^{\circ}\text{F}$  ó  $^{\circ}\text{F}$  a  $^{\circ}\text{C}$ ) con la misma frecuencia, además se da por hecho que no se cometen errores. En el caso de que no esté seleccionado el botón deseado se tomarán las siguientes acciones:

- Mover la mano al ratón: H
- Apuntar al botón deseado: H P
- Dar clic al botón: H P K
- Regresar la mano al teclado: H P K H
- Teclear la temperatura: H P K H K K K K
- Teclear “enter”: H P K H K K K K K

Las heurísticas que se toman en cuenta para eliminar M's son:

1. Inserción inicial de una M antes de cada K y antes de cada P que seleccione un comando, pero no antes de P's que apunten a argumentos de un comando
  2. Eliminación de M's antes de una acción que resulta de otra (ej. El cursor se mueve antes de oprimir una tecla, por tanto P M K se vuelve P K)
  3. Eliminación de M's dentro de unidades cognitivas.
  4. Eliminación de M's antes de delimitadores consecutivos (ej. fin de comando e inicio de argumento).
  5. Eliminación de M's para secuencias habituales que se vuelven unidades cognitivas (argumentos fijos de comandos); para secuencias variables, la M permanece.
  6. Eliminación de M's que se traslapan con tiempos de espera R.
- Ahora, se aplican dichas heurísticas al ejemplo dado líneas arriba.

*Regla 0:* agregar M's para cada K y P:

H M P M K H M K M K M K M K M K

*Regla 1:* cambiar P M K a P K:

H M P K H M K M K M K M K M K

*Regla 2:* eliminar M's dentro de unidades cognitivas (en este caso, la temperatura).

H M P K H M K K K K M K

*Regla 4:* La M antes de la última K permanece

Reglas 3 y 5 no se aplican

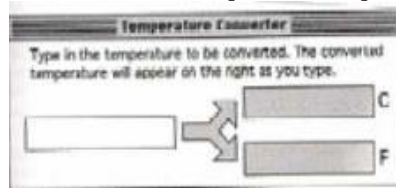
Tiempo estimado:  $0.4 + 1.35 + 1.1 + 0.2 + 0.4 + 1.35 + 4*0.2 + 1.35 + 0.2 = 7.15$  seg.

En caso de que el botón deseado ya esté seleccionado, el método es: M K K K K M K

El tiempo estimado es  $1.35 + 4*0.2 + 1.35 + .2 = 3.7$  seg.

Como las opciones se requieren con la misma frecuencia, el tiempo promedio es  $(7.15 + 3.7) / 2 = 5.4$  seg.

Después de realizar éste prueba se cambia el diseño que se tenía quedando el siguiente:



### Modelo de etapas de acción

Este modelo, considera siete etapas para describir la interacción humano-computadora:

1. Formación de una intención.
2. Formación de una meta.
3. Especificación de una acción.
4. Ejecución de la acción.
5. Percepción del estado del sistema.
6. Interpretación del estado del sistema.
7. Evaluación del resultado.

Debido a eso, el modelo identifica brechas de ejecución (entre las intenciones del usuario y las acciones disponibles) y de evaluación (entre la representación de la computadora y las expectativas del usuario), además origina principios útiles como la visibilidad de estado así como las alternativas de acción.

### Consistencia a través de gramáticas

Utiliza conjuntos de tareas válidas y conjuntos de acciones válidas como notación para especificar la interacción humano-computadora. Ejemplo de acciones válidas: adelantar-cursor-caracter [dirección = adelante, unidad = caracter], atrasar-cursor-palabra [dirección = atrás, unidad = palabra]. Esta notación se utiliza con frecuencia para verificar consistencia y completitud.

## Modelo de Interfaz Objeto-Acción (OAI)

Están basados en la representación visual de objetos y acciones de la tarea del usuario, se deben seleccionar los objetos de la interfaz, en este modelo hacer un diseño-acción comienza con la comprensión de la tarea, dicha tarea incluye el universo de objetos del mundo real con que trabajan los usuarios para llevar a cabo sus intuiciones y las acciones que aplican a esos objetos. Los objetos son generalmente sustantivos. Se consideran solamente aquellos objetos que los usuarios necesitan ver o a través de los cuales se efectúan acciones en la interfaz, pueden existir otros objetos importantes para los desarrolladores del software, pero no para los diseñadores de la interfaz. Pueden existir relaciones “objeto”-”sub-objeto” Ejemplo: registro de cuenta bancaria / entrada en el registro de cuenta bancaria.

*Escenario de caso de uso:* Pago de Facturas con cheques impresos

1. Ingreso de facturas (nombre del beneficiario y fecha de pago) en una planilla.
2. Ver el nuevo saldo (saldo actual menos cada monto a pagar).
3. Decidir si existen suficientes fondos para pagar las facturas.
  - a. Si existen (80% de las veces), ir al paso 4.
  - b. Si no existen (20%), marcar las facturas que no se pagarán (80%) o cambiar los montos de pago (20%), luego ir al paso 4.
4. Indicar al sistema que pague las facturas.
5. Colocar las opciones de impresión para los cheques.
6. Colocar el papel de impresión de cheques en la impresora.
7. Imprimir los cheques.
8. Observar los cheques.
9. Reimprimir los cheques si existe un problema (30% de las veces).
10. Observar el registro actualizado de la cuenta corriente bancaria (es observado 50% de las veces).
11. Generar reportes (50% de las veces).

La selección de objetos del usuario se hace de la siguiente manera:

1. Identificar los atributos de los objetos: es la información de los objetos que los usuarios necesitan ver cuando efectúan sus tareas.
2. Identificar las acciones del usuario en los objetos: Las acciones deben estar expresadas en la terminología utilizada por los usuarios (obtenida en el análisis).
3. Construir una tabla “Objetos-Acciones”

La tabla para el caso de uso es.

Objeto/Sub-objeto	Atributos	Acciones del usuario
Registro de Cuenta Bancaria	Nombre de Banco Número de cuenta Saldo actual Entradas en el registro	Ver Recorrer Agregar entrada Imprimir Ordenar
Entrada en Registro de Cuenta bancaria	Tipo(depósito, extracción,...) Monto Fecha Efectivizado Notas	Ver Editar Copiar y pegar Imprimir Anular Buscar Mover
Saldo actual	Valor	Ver Ajustar
Objeto/Sub-objeto	Atributos	Acciones del usuario
Cheque	Número de cuenta Monto Fecha Beneficiario	Ver Editar Imprimir Copiar Enviar/Transferir
Planilla de facturas	Fecha de creación	Crear Guardar Ver Recorrer Imprimir
Factura	Nombre beneficiario Monto total Monto a pagar	Ingresar datos Cambio de monto Borrar Pegar

## **PRINCIPIOS**

El principio más importante en el diseño es el conocimiento que se tenga sobre el usuario, es sumamente importante conocer el contexto social en el que se encuentran, esto es, conocer el nivel de acercamiento a la tecnología, su edad, su situación económica, la importancia que tiene y/o ejerce la tecnología sobre él, ¿qué tan frecuente la utiliza?, ¿le ayuda en su vida diaria?, ¿es molesta para el usuario tener que utilizarla?, etc. Todo esto es lo que motiva el uso de un software específico atendiendo a las demandas del usuario.

### **Clasificación de usuarios**

A continuación se presenta una clasificación de usuarios, ésta tiene que ver con el nivel de conocimiento de tecnología (software) del usuario.

**Usuarios novatos:** desconocen tareas e interfaces.

**Usuarios primerizos:** desconocen la interfaz.

Para los dos anteriores se recomienda restringir vocabulario y número de acciones disponibles, mantener una retroalimentación informativa continua, utilizar manuales en papel y en línea.

**Usuarios intermitentes con experiencia:** manejan la tarea y los conceptos de la interfaz, mas no su estructura y ubicación. Es recomendable utilizar secuencias consistentes de acciones, también guías a patrones frecuentes de uso

**Usuarios expertos:** Usuarios frecuentes con experiencia. Para este tipo de usuarios es importante proporcionarles tiempo de respuesta breve, retroalimentación concisa, atajos, aceleradores y macros, además de tutoriales.

## **DISEÑO UNIVERSAL**

Las diferencias individuales en un grupo de aproximadamente 30 personas pueden llegar a menudo a ser de un factor de 20 (Egan, 1988). Aunque muchos desarrolladores estarían de acuerdo en que el mayor número de usuarios sea capaz de usar sus productos, no todos ellos se sentirían tan dispuestos a realizar los esfuerzos necesarios para lograrlo. Microsoft (2000) estima que uno de cada cinco estadounidenses tiene algún tipo de discapacidad: 30 millones de personas en el mismo país pueden verse afectados por el diseño de su software.

Muchas empresas de gran tamaño e instituciones contratan como parte de su política de personal a un grupo fijo de personas calificadas legalmente como discapacitados. Muchos gobiernos han incluido reglamentos y leyes que especifican requisitos que deben cumplir los productos utilizados en la administración pública y en las organizaciones que dependen de ella. Las computadoras, y con ellas Internet, ofrecen la ocasión de romper las barreras físicas para estos individuos, abriéndoles una gran cantidad de oportunidades de relaciones sociales, opciones laborales y de todo tipo.

El diseño universal es el proceso de diseñar productos que sean usables por el rango más amplio de personas, funcionando en el rango más amplio de situaciones y que sean comercialmente practicables. El diseño de los productos y de entornos ha de ser usable por la mayor parte de la gente posible, sin necesidad de adaptación o de diseño especializado. Los principios a tener en cuenta son los siguientes:

**Uso equitativo.** El diseño ha de ser usable y de un precio razonable para personas con diferentes habilidades.

**Uso flexible.** El diseño se ha de acomodar a un rango amplio de personas con distintos gustos y habilidades.

**Uso simple e intuitivo.** El uso del diseño ha de ser fácil de entender, independientemente de la experiencia del usuario, conocimiento, habilidades del lenguaje y nivel de concentración en el momento de utilizar el artefacto.

**Información perceptible.** El diseño comunica la información necesaria de manera efectiva al usuario, independientemente de las condiciones ambientales.

**Tolerancia al error.** El diseño minimiza posibles incidentes por azar y las consecuencias adversas de acciones no previstas.

**Esfuerzo físico mínimo.** El diseño se ha de poder usar eficientemente y confortablemente con un mínimo de fatiga.

**Tamaño y espacio para poder aproximarse y usar el diseño.** El diseño ha de tener un espacio y un tamaño apropiado.

Existen diferentes discapacidades y también están presentes en grado diferente, a continuación se describen brevemente algunas de las categorías de discapacidades más importantes y las soluciones que se aplican en cada caso.



## **Deficiencias Visuales**

Las deficiencias visuales más comunes son las debidas a la incapacidad para captar correctamente los colores, los debidos a una visión reducida y finalmente, la ceguera, o falta de visión completa.

**Color.** El ojo humano contiene bastones y conos sensibles a la luz, los conos están especializados en el color, sin ellos veríamos en blanco y negro; hay conos para los colores rojo, verde y azul. A partir de su combinación se obtienen el resto de los colores. Los defectos de visualización del color provienen de una falta en alguno de los tres tipos de conos. La protanopia es la ausencia de color rojo, la ausencia de color verde se conoce como deuteranopia y tritanopia es la ausencia de color azul. La presencia de estos problemas afecta más a los hombres que a las mujeres ya que están relacionados con el cromosoma X, alrededor de un 8% de los hombres tienen algún problema de este tipo mientras que sólo un 0.5% de las mujeres lo padecerían. De los tres problemas, la falta de percepción del azul es el menos común pero el más grave de ellos. La lección para el diseñador de interfaces es sencilla: No codificar ninguna conducta importante únicamente mediante color, utilizar colores perfectamente distinguibles y comprobar visualización en distintas condiciones.

**Visión reducida.** Las discapacidades visuales van desde una falta de agudeza visual hasta la completa falta de visión. Una gran cantidad de los esfuerzos en interfaz actuales se apoyan en elementos gráficos, resulta lógico ofrecer a los usuarios con visión reducida la opción de utilizar esos elementos hasta el límite donde sea posible. Los ampliadores de pantalla son programas que permiten una ampliación de parte de la pantalla, esta visión ampliada facilita la lectura a los usuarios con dificultades visuales pero por otro lado introducen problemas de navegación y orientación dentro de los documentos de los usuarios. A menudo, estas utilidades funcionan ofreciendo dos visiones simultáneas de la información que se encuentran coordinadas entre sí. Por otro lado, muchas aplicaciones pueden ofrecer como parte de ellas mismas ayudas a los usuarios de una manera natural. Por ejemplo, los procesadores de textos pueden utilizar fuentes escalables que pueden ser ampliadas sin reducción de la calidad, y los programas de dibujo pueden aumentar partes de sus elementos con facilidad.

**Ceguera.** Cuando las deficiencias visuales llegan al límite en el que no es posible utilizar la información de las pantallas, la computadora necesitará cambiar el canal de comunicación y utilizar uno diferente. Los canales de salida más aprovechables en el momento actual son los de voz sintetizada y las tabletas de Braille actualizables. En ambos casos, toda la información pasaría a ser de tipo verbal y buena parte de la información gráfica necesitaría reconvertirse en descripciones textuales. Los elementos gráficos son descritos mediante etiquetas que los programadores insertan utilizando normas estandarizadas. Estas etiquetas son textos cortos o palabras que explican el significado de estos elementos gráficos. El teclado es el elemento de introducción de datos y navegación más importante para un usuario invidente ya que los instrumentos apuntadores pueden resultarle de poca utilidad.