TRABAJO PRÁCTICO Nº 1

Instalación y Características de los Sistemas Operativos

1) SISTEMA OPERATIVO: Es un programa que administra el Hardware de una computadora. Proporciona las bases para los programas de aplicación y actúa como un intermediario entre el usuario y el hardware.

Ejemplos:

Para PC:

-Solaris, Linux, Windows 7, Mac OS X, FreeBSD.

Para dispositivos móviles:

-Android, Windows pone, iOS, Blackberry.

2) HARDWARE: MÍNIMO Y RECOMENDADO

o Disco Duro: 195GB

o Tarjeta Gráfica: AMD Radeon HD 7290 Graphics

Con la configuración del hardware mínima el rendimiento del sistema suele ser pobre, ya que el SO sólo podrá ejecutar funciones básicas.

Con la configuración de hardware recomendada el SO funciona a una velocidad adecuada.

3) La BIOS es el software que localiza y reconoce los dispositivos necesarios para cargar el Sistema Operativo en la memoria RAM. La secuencia de arranque indica a la Bios que unidad debe buscar al arrancar el equipo.

4) Formatear un disco duro consiste en restablecer a este a su estado original para ser reutilizado o reescrito con una nueva información. Al realizar esta operación se elimina toda la información contenida en el. Antes de formatear hay que hacer backup de tus datos, así evitas perder información.

5) Una partición es la división de espacio que se le asigna a un disco duro. Un disco duro puede tener varias particiones, que son de cierto modo independientes entre si, dado que cada una de ellas tiene su propio sistema de archivos o formato, y el SO lo reconoce como un disco independiente (aunque físicamente sea uno).

Se puede formatear una partición sin necesidad de formatear todo el disco.

Hay dos tipos de particiones:

• Partición Primaria: Se encarga de iniciar la carga del SO

• Partición Extendida o Secundaria: Almacenan información

6) La instalación mínima de un programa consiste en instalar sólo ciertas funciones del programa, siguiendo los requisitos mínimos de la PC; la instalación personalizada se realiza en base a la elección de opciones que uno desea instalar; la instalación típica es aquella donde el programa instala por defecto todas las funciones.

7) Una máquina virtual se utiliza para emular el funcionamiento de un ordenador dentro de otro ordenador. Se puede crear para instalar otro sistema operativo eligiendo el disco duro, la memoria RAM, memoria gráfica, tarjeta de red o de sonido, etc, emulando todos los componentes de hardware. Esta emulación se ejecuta dentro de la máquina virtual, como si fuera un programa más y sin que afecte al ordenador que la ejecuta.

Este tipo de emulación se utiliza principalmente para probrar diferentes sistemas operativos, programas o configuraciones con total seguridad.

Ejs:

1)Requisitos Requisitos de Windows Requisitos de Linux

Procesador 233 MHz o superior 32bits a 1GHz

Memoria Ram Mínimo 64 MB de RAM 1Gb

Disco Duro Mínimo 1,5 GB 15Gb

Tarjeta Gráfica resolución Súper VGA (800x600) Tarjeta gráfica y monitor 800x600

o superior

2) ¿Cuáles son las características de la PC empleada?

o Procesador: AMD C-70 APU with Radeon(tm) HD Graphics 1.00 GHz

o Memoria Ram: 2,00 GB

3) Procesador: 2 GHz compatible con PAE, NX y SSE2.

RAM: 1 GB (32 bits) o 2 GB (64 bits).

Espacio en disco duro: 16 GB (32 bits) o 20 GB (64 bits).

Tarjeta gráfica: Dispositivo gráfico Microsoft DirectX 9 con controlador WDDM.

4)UEFI (Interfaz de firmware extensible unificada) es el código del firmware de un chip en la placa base que establece instrucciones para controlar las operaciones de los circuitos de un dispositivo. Este código se integra al hardware del dispositivo. Por otro lado, la función del BIOS es iniciar los componentes de hardware y el sistema operativo, entre otras.

El BIOS legacy

El BIOS

-ejecuta el código para verificar la integridad de los componentes del firmware antes de que se ejecute y lance el sistema operativo.

-Probar los componentes clave de hardware en la computadora para garantizar que toda la información cargue correctamente y no genere problemas sobre la información.

-Controla módulos adicionales como la tarjeta de vídeo o la tarjeta de red de área local, entre otros dispositivos.

-Selecciona el dispositivo de arranque que puede ser el disco duro, una unidad de CD o un dispositivo USB.

Una de las diferencias es que UEFI arranca ejecutando un código en 32 o 64-bit de modo protegido en la CPU, no en modo de 16 bits como suele ser el caso de BIOS.

Dentro de las características adicionales de UEFI está la reducción en el tiempo de inicio y reanudación, y cuenta con un proceso que ayuda a prevenir de ataques del tipo bootkit y utilizar el modo Secure Boot. Estas son algunas de las razones por las cuales UEFI podría reemplazar a BIOS en el sistema de arranque de las computadoras.

UEFI BIOS admite dos modos de inicio: Legacy BIOS y UEFI. Algunos dispositivos y sistemas operativos aún no admiten el BIOS basado en UEFI y solo se inician desde Legacy BIOS.

5)1-En el programa de Oracle VM VirtualBox crear nueva máquina virtual y elegimos la instalación de Windows en 64 bits.

2-Elegimos la cantidad mínima de memoria RAM. La de defecto es 1 GB.

3-Luego elegimos el tipo de disco duro, en la cual usaremos la opción que el programa marca por defecto: Virtual Disk Image o VDI.

4-Elegimos el almacenamietno en unidad de disco duro dinámico (usará espacio en su disco físico a medida que se llena hasta un tamaño fijo, sin embargo no se reducirá de nuevo cuando el espacio en el se libere).

5-En el siguiente paso tendremos que establecer la ubicación del disco duro y elegir el tamaño con el que empezará a "funcionar". Podemos introducirla a mano o mediante el icono que queda a la derecha del campo de introducción de texto. Cuando lo tengamos listo, pulsamos "Crear". Ahora la máquina virtual ya está creada y configurada. Después de esto, sólo nos queda instalar el sistema operativo.

6-Al iniciar la VM se abrirá una especie de segunda pantalla dentro de nuestro monitor, y es en este momento donde tendremos que decirle a la máquina virtual dónde hemos guardado la imagen ISO. Cuando la hayamos ubicado, pulsamos en "Iniciar".

7-Comienza la instalación de Windows con la configuración de idioma, formato de hora y teclado, y la colocación de la clave de producto del SO (en este último podemos colocar 'omitir').

8-Luego, se puede hacer una personalización de la configuración, con respecto por ejemplo a permitir el envío de información sobre contactos, calendarios, aplicaciones, ubicación y errores y diagnóstico a Microsoft. También sobre la conexión automática a redes.

6-Se puede iniciar el SO asociado a una cuenta de Microsoft. Para esto se debe colocar el correo electrónico y la contraseña de la cuenta (se puede omitir este paso).

7-Continuará la instalación de todos los drivers y archivos necesarios, hasta que aparezca un cartel de 'Bienvenido' y la conocida imagen del escritorio de Windows.

6) Procesador: x86 a 1 GHz.

Memoria: 1 GB de RAM.

Espacio en disco duro: 15 GB.

Tarjeta gráfica: capaz de soportar una resolución de 800 x 600.

7)1-En el programa de Oracle VM VirtualBox crear nueva máquina virtual y elegimos la instalación de Ubuntu en 64 bits.

2-Elegimos la cantidad mínima de memoria RAM. La de defecto es 1 GB.

3-Luego elegimos el tipo de disco duro, en la cual usaremos la opción que el programa marca por defecto: Virtual Disk Image o VDI.

4-Elegimos el almacenamietno en unidad de disco duro dinámico (usará espacio en su disco físico a medida que se llena hasta un tamaño fijo, sin embargo no se reducirá de nuevo cuando el espacio en el se libere).

5-En el siguiente paso tendremos que establecer la ubicación del disco duro y elegir el tamaño con el que empezará a "funcionar". Podemos introducirla a mano o mediante el icono que queda a la derecha del campo de introducción de texto. Cuando lo tengamos listo, pulsamos "Crear". Ahora la máquina virtual ya está creada y configurada. Después de esto, sólo nos queda instalar el sistema operativo.

6-Ahora se abrirá una especie de segunda pantalla dentro de nuestro monitor, y es en este momento donde tendremos que decirle a la máquina virtual dónde hemos guardado la imagen ISO. Cuando la hayamos ubicado, pulsamos en "Iniciar".

7-La máquina iniciará el proceso de arranque y se iniciará el proceso de instalación. En la primera pantalla que aparezca presionamos en 'Instalar Ubuntu' (se puede elegir 'Probar Ubuntu' sin instalarlo).

8-En la siguiente pantalla seleccionamos "Instalar software de terceros" para poder usar Flash o reproducir archivos MP3

9-Luego elegiremos la opción 'Borrar disco e instalar Ubuntu'.

10-Luego seleccionamos la zona horaria y la distribución del teclado, y configuraremos el usuario y la contraseña para acceder al sistema operativo y realizar tareas administrativas.

11-A partir de aquí, sólo tenemos que dejar que el programa de instalación trabaje, hasta que nos pida reiniciar la máquina para acceder al sistema operativo. Cuando lo hagamos, llegaremos hasta la pantalla de inicio de sesión. Introducimos nuestra contraseña y nos encontraremos con el escritorio de Ubuntu.

TRABAJO PRÁCTIVO N°2

Drivers

1) Es un programa que permite al SO interaccionar con un periférico, haciendo una abstracción del hardware y proporcionando una interfaz para utilizar el dispositivo.

Le indica al SO como controlar y comunicarse con un dispositivo particular.

2) La actualización de los controladores le garantiza el óptimo funcionamiento de su hardware. Aunque las ventajas dependen del controlador en cuestión que quiera actualizar, las actualizaciones de los controladores ofrecen por lo general uno o varios de los siguientes beneficios:

Mejor compatibilidad del hardware y menos conflictos del sistema

Un mejor rendimiento del dispositivo

Eliminación de los errores de programación y de los dispositivos

Nuevas funciones del dispositivo o funciones sin bloquear

Un mejor rendimiento del equipo y mayor estabilidad

3) Plug-and-play se refiere a la capacidad de un sistema informático de configurar automáticamente los dispositivos al conectarlos. Permite poder enchufar un dispositivo y utilizarlo inmediatamente, sin preocuparte de la configuración.

El Plug-and-play es un estándar para añadir hardware que requiere que un dispositivo se identifique a sí mismo al conectarse.

4) a)El Administrador de dispositivos es una utilidad de Windows a través de la cual podemos gestionar todo el Hardware instalado en el equipo. b)Para abrirlo, abrimos el Panel de control>Hardware y Sonido>Dispositivos e Impresoras>Administrador de dispositivos.

c)Los drivers se distribuyen según el hardware al que permite el funcionamiento. Por ejemplo, existe clasificación de adaptadores de pantalla, adaptadores de red, baterías, bluetooth, controladoras de almacenamiento, etc.

5)a)En información del sistema veo qué sistema operativo es el que trabaja, el núcleo de linux, el procesador, memoria, disco duro y tarjeta gráfica, y todos sus datos.

b) Se denomina kernel.

TRABAJO PRÁCTICO Nº 3

Prototipado

1)Una herramienta de desarrollo de software es un programa informático que usa un programador para crear, depurar, gestionar o mantener un programa . Existen herramientas con fines muy diversos; por ejemplo generadores de código, conversores de código, compiladores, depuradores, generadores de estilo, etc.

2)El ambiente de desarrollo integrado o IDE es un software que proporciona servicios integrales para facilitarle al programador el desarrollo de software.

Normalmente, un IDE consiste de:

-Editor de código fuente, Herramientas de construcción automáticas, Depurador, Compilador, Intérprete.

Ejemplos de IDE's para desarrollo Web:

Eclipse, Net Beans, JetBrains Web Storm, Aptana, Dreamweaber, Microsoft Visual Studio Express for Web, Sublime Text, NotePad ++

3) **Tipos de herramientas que tienen los entornos de programación:**

IDE: software que proporciona servicios integrales para facilitarle al programador el desarrollo del software.

Sistema de Control de Versiones: permite gestionar archivos de un proyecto (y sus versiones) y que sus integrantes puedan acceder remotamente a ellos.

Bug Tracker: realiza seguimiento de defectos o Bugs que surgen.

Pruebas de desempeño o Load Test: simula la realidad a la que está expuesto el sistema en producción.

Pruebas de estrés: simula mayor carga que la esperada.

Pruebas de resistencia: evalúa el desempeño del sistema luego de una carga duradera por un largo período de tiempo.

Documentación: sirve para entender lo que se hace y por qué, y permite mantener el código.

4) GESTIÓN DE VERSIONES O VCS

Permiten gestionarlos archivos de un proyecto (y sus versiones) y que sus integrantes puedan acceder remotamente a ellos.

Los usuarios pueden: Descargar los archivos, modificar archivos, publicar cambios, volver a versiones anteriores, ramificar a partir de una versión, aplicar cambios de distintas versiones.

VENTAJAS DE VCS:

-Permite llevar cuenta de los cambios de un conjunto de archivos digitales en el tiempo

-Cada versión registrada en el tiempo es una revisión

-Permite tanto a un desarrollador como a un grupo de desarrolladores, gestionar el código del proyecto.

HERRAMIENTAS DE VCS:

-Git

-SVN

-Mercurial

5) GIT: Es un sistema distribuido de control de versiones. Es libre. Crea branches y hace merge rapido y fácil. Cada "Clón" es un repositorio completo.

Servidores que pueden utilizarlo: Github y Bitbucket.

6) La documentación de Código es importante porque permite entender que se está haciendo y porqué. Mantener el código. A tener en cuenta:

No uses solo una linea, divídela en párrafos para que sea más legible

Tabular los comentarios de lineas consecutivas

No comentar obviedades

Se profesional (No insultes o coloques frases fuera de contexto)

Revisa la ortografía

No comentes si no es necesario

Comentarios simple y directo

Documenta mientras desarrollas

Mantén los comentarios actualizados

7)Doxygen es un generador de documentación para C++, C, Java, Objective-C, Python, IDL (versiones Corba y Microsoft), VHDL y en cierta medida para PHP, C# y D. Dado que es fácilmente adaptable, funciona en la mayoría de sistemas Unix así como en Windows y Mac OS X.

8) Líneas de comando de GIT para:

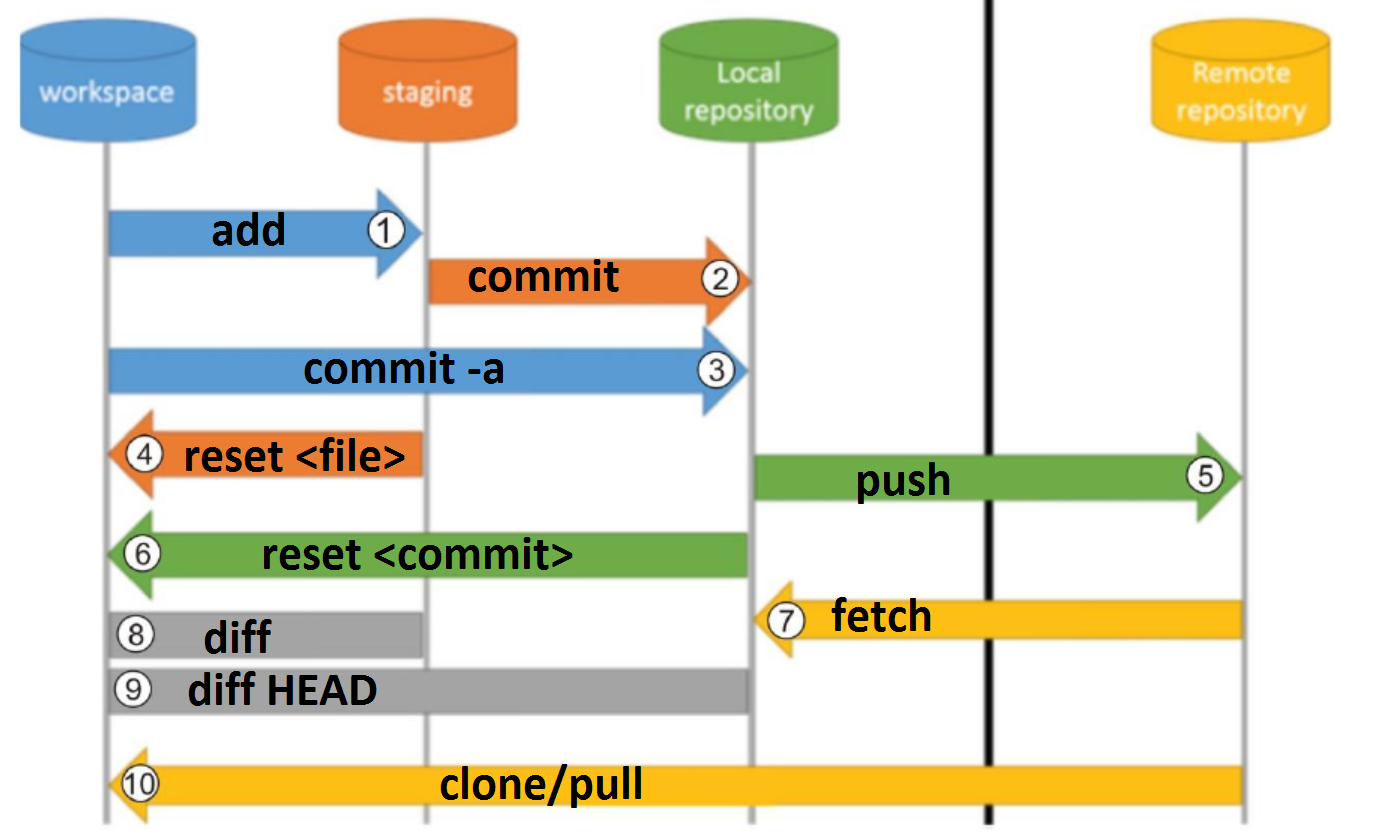
a) Clonar un repositorio : git clone

b) Actualizar tu repositorio local al commit más nuevo : git fetch

c) Registrar cambios (añadirlos al Index) : git add "archivo.html"

d) Hacer un commit : git commit -m "mensaje"

e) Enviar cambios a repositorio remoto : git push

9)

10) Los Bug Trackers realizan un seguimiento de defectos o Bugs que surgen en el funcionamiento del sistema. Esta aplicación se utiliza para probar soluciones automatizadas, llevando un registro histórico de las alteraciones y gestionando equipos de trabajo de forma remota. Un ejemplo muy utilizado es el *Mantis Bug Tracker.*

11) Pruebas de desempeño o Load Test: simula la realidad a la que está expuesto el sistema en producción.

Pruebas de estrés: simula mayor carga que la esperada.

Pruebas de resistencia: evalúa el desempeño del sistema luego de una carga duradera por un largo período de tiempo.

12) Consiste en hacer integraciones automáticas de un proyecto lo más a menudo posible para así poder detectar fallos cuanto antes. Entendemos por integración la compilación y ejecución de pruebas de todo un proyecto.

El proceso suele ser: cada cierto tiempo (horas), descargarse las fuentes desde el control de versiones (por ejemplo CVS, Git, Subversion, Mercurial o Microsoft Visual SourceSafe) compilarlo, ejecutar pruebas y generar informes.

Para esto suelen utilizarce aplicaciones como Solano CI,Bamboo,Pipeline, Apache Continuum, Hudson, Jenkins, GoCD, CruiseControl o Anthill (para proyectos Java) o CruiseControl.Net, Team Foundation Build para .Net, que se encargan de controlar las ejecuciones, apoyadas en otras herramientas como Ant o Maven (también para proyectos Java), o Nant o MSBUILD (para .Net) que se encargan de realizar las compilaciones, ejecutar las pruebas y realizar los informes.

**TRABAJO PRÁCTICO Nº 4**

**Herramientas de desarrollo**

1. INTERFAZ DE USUARIO

Es el medio con que el usuario puede comunicarse con una máquina, equipo, computadora o dispositivo, y comprende todos los puntos de contacto entre el usuario y el equipo.

Incluyen elementos como menús, ventanas, contenido gráfico, cursor, etc.

PROCESO DE DISEÑO DE INTERFAZ DE USUARIO

Proceso iterativo donde los usuarios interactúan con los diseñadores y prototipados de la interfaz para decidir las características, organización, apariencia y funcionamiento de la interfaz de usuario del sistema.

Factores a tener en cuenta cuando se diseña:

-Analizar y comprender las actividades del usuario

-Realizar el diseño del prototipo en papel

-Evaluar el diseño con los usuarios finales

-Realizar el diseño dinamico del prototipo

-Prototipo ejecutable

-Evaluar el diseño con los usuarios finales

-Implementar la interfaz de usuario definitiva

1. Pasos del proceso de diseño de interfaz de usuario:

**PASO 1:**

El diseño de la interfaz de usuario comienza con la identificación de los requerimientos del usuario, la tarea y el ambiente.

**PASO 2:**

Una vez identificadas las tareas del usuario, se crean y analizan los escenarios para éste y se define un conjunto de objetos y acciones de la interfaz.

**PASO 3:**

Esto forma la base para crear una plantilla o prototipo que ilustra el diseño gráfico y la colocación de los iconos, la definición de textos descriptivos, la

especificación y títulos de las ventanas, y la especificación de aspectos mayores y menores del menú.

**PASO 4:**

Con el empleo de herramientas, se hace el prototipo, se implementa en definitiva el modelo del diseño y se evalúa la calidad del resultado.

1. Se debe considerar las capacidades físicas y mentales de las personas que utilizarán el software ya que:

-Las personas tienen una memoria limitada a corto plazo.

-Cuando el sistema falla, y emite avisos y mensajes genera estrés en el usuario.

-Las personas poseen un amplio rango de capacidades físicas.

-Las personas poseen diferentes preferencias de interacción.

1. Los principios del diseño son:

**Familiaridad del usuario:** Utilizar términos que el usuario conozca **Unifomidad:** Operaciones comparables se deben activar de la misma forma

**Mínimo Sorpresa:** El comportamiento del sistema debe ser esperable

**Recuperabulidad:** Mecanismos de recuración de errores

**Guía de Usuario:** Retroalimentación significativa frente a errores. Asesoramiento y características.

**Diversidad de Usuarios:** Interacción para diferentes tipos de usuario. (Casuales, Potenciales)

1. Según Shneiderman existen 5 estilos de interacción:
2. Manipulación Directa
3. Selección de menús
4. Rellenado de Formularios
5. Lenguaje de Comandos
6. Lenguaje Natural

**ESTILOS DE INTERACCIÓN**

| **Estilo de Interacción** | **Ventajas** | **Desventajas** | **Aplicación** |
| --- | --- | --- | --- |
| Manipulación Directa | Interacción rápida  Intuitiva  Fácil de Aprender | Difícil de Implementar  Sólo adecuada para sistemas visuales | Videojuegos  Sistemas CAD |
| Selección de menús | Evita errores de usuario  Requiere teclear poco | Lenta para usuarios experimentados  Compleja si hay muchas opciones | Sistemas de propósito general |
| Rellenado de Formularios | Introducción de datos sencilla | Ocupa espacio en pantalla  Opciones del usuario no se ajustan a los campos | Control de Stock  Préstamos personales |
| Lenguaje de Comandos | Poderoso y Flexible | Difícil de aprender  Gestión pobre de errores | Sistemas operativos  Sistemas de comandos y control |
| Lenguaje Natural | Accesible a usuarios casuales  Fácil de ampliar | Requiere más tecleo  Los sistemas de comprensión de LN no son fiables | Sistemas de recuperación de información |

**MODELO DE SEEHEIM**

1. El modelo plantea la comunicación entre el usuario y la aplicación estructurada en tres niveles:

1) El nivel de presentación. La parte estática y visible de la interfaz que se comunica con el usuario y se construye sobre sistemas de ventanas y cajas de herramientas. Supondría el léxico de la interfaz.

2) El nivel de diálogo. La parte dinámica que maneja los eventos o mensajes que se producen como consecuencia de las acciones del usuario sobre la interfaz, y establece la comunicación entre el nivel de presentación y el nivel de aplicación, es decir, establece la relación entre los eventos y su correspondiente respuesta por parte del nivel de aplicación. Se podría equiparar a la sintaxis de la comunicación entre los otros dos niveles.

3) Interfaz de aplicación. La parte de la aplicación que el usuario controla a través de la interfaz y que es visible para éste último. Equivaldría a la semántica de la aplicación. La división en capas facilita el tratamiento de cada una por separado y además promueve la reutilización y la portabilidad, con lo que se posibilita el desarrollo rá- pido de prototipos, tarea fundamental para un diseñador.

1. Scheiderman determinó 14 pautas clave para la utilización efectiva del color en las interfaces del usuario. EL color puede mejorar las interfaces de usuario ayudando a los usuarios a comprender y manejar la complejidad.
2. Las 5 pautas más importantes de Shneiderman son:

- Limitar el número de colores utilizados y ser conservador en la forma de utilizarlos

- Utilizar un cambio de color para mostrar un cambio en el estado del sistema

- Utilizar el código de colores para apoyar la tarea que los usuarios están tratando de llevar a cabo

- Utilizar el código de colores de una forma consciente y uniforme

- Ser cuidadoso al utilizar pares de colores

9) La usabilidad es la medición cualitativa de la facilidad y eficiencia con la que un humano emplea las funciones y características que ofrece el producto de alta tecnología. Sus atributos son:

 **Aprendizaje:** Tiempo que tarda un usuario nuevo en ser productivo con el sistema

 **Velocidad de Funcionamiento:** Respuesta del sistema a las operaciones de trabajo del usuario

 **Robustez:** Tolerancia del sistema a errores de usuario.

 **Recuperación:** Cómo se recupera el sistema a los errores del usuario

 **Adaptación:** El sistema se adapta a único modelo de trabajo

10) TÉCNICAS DE EVALUACIÓN DE LA INTERFAZ

* Cuestionarios de la opinión de los usuarios sobre la interfaz
* Observación de los usuarios cuando usan el sistema
* Videos del uso típico del sistema
* Incluir código en el software que recopila información de los recursos utilizados y errores comunes

11) Los prototipos son una representación limitada de un producto, permite a las partes probarlo en situaciones reales o explorar su uso, creando así un [proceso de diseño de iteración](http://albertolacalle.com/diseno-iteracion.htm) que genera calidad.

Un prototipo puede ser cualquier cosa, desde un trozo de papel con sencillos dibujos a un complejo software.

**Ventajas:**

Porque son útiles para comunicar, discutir y definir ideas entre los [diseñadores](http://albertolacalle.com/diseno.htm) y las partes responsables.

Los prototipos apoyan la [evaluación de productos](http://albertolacalle.com/hci-test-usabilidad.htm), clarifican [requisitos de usuario](http://albertolacalle.com/hci/requisitos-usuario.htm) y definen alternativas.

Un sketch es un bosquejo rápido que se realiza a mano (con lápiz o bolígrafo) sobre papel para reflejar la primera idea o concepto, constituyendo así el punto de partida de lo que será el producto final. Es un bosquejo estático de baja calidad.

Un wireframe se emplea para definir cuál será el contenido y dónde se ubicarán los distintos elementos. Es también una representación de baja calidad, y suele entregarse como un archivo de imagen en lugar de en papel como en el caso anterior.

Un mockup permite representar estáticamente de forma más visual el proyecto. Su calidad es media alta y también se entrega en formato de imagen.

12) Proceso de evaluar la forma en que se utiliza una interfaz y verificar que cumple los requerimientos del usuario.   
  
Es parte del proceso de verificación y validación de los sistemas software.

**TRABAJO PRÁCTICO Nº 6**

**Aplicaciones web**

1. **Aplicación web**

Software almacenado en una computadora o servidor web, que los usuarios pueden utilizar a través de Internet o de una Intranet, con un navegador web, para obtener los servicios que ofrezca.

Ejemplos:

* Gestores de correo
* Web mails
* Wikis
* Blogs
* Tiendas en Línea

2) **Ventajas:**

* Facilidad de acceso (sólo se necesita navegador web)
* Independencia de sistema operativo
* Facilidad de actualización y mantenimiento (sin redistribuir o instalar software)

Según el tipo de acceso se clasifican en:

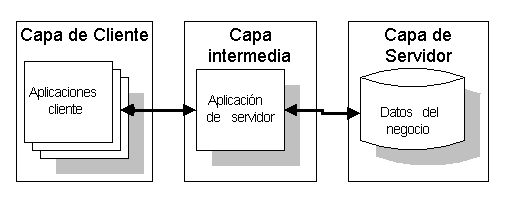
Públicas: Tiendas virtuales, diarios digitales, portales de internet, etc.

Restringidas: Intranets, que ofrecen servicios para mejorar gestiones internas de una empresa (control hs de personal, gestión ed proyectos o tareas), etc

3) Estructura de la página Web:

Normalmente posee 3 capas: 

* Capa del Cliente
* Capa Intermedia
* Capa de Servidor



4) Una **página web** es un documento estático al que se accede a través de un navegador (la información sólo se puede leer, no interactuar con ella).

Una **aplicación web** es una página web especial, que tiene información sobre la que se puede interactuar e incluso cambiar.

**5) Web Estática:** Muestra información al navegante, pero no le permite interactuar con la página web visitada.

**Web Dinámica:** Contiene elementos que permiten la comunicación activa entre el usuario y la aplicación, y cuyo contenido se genera a partir de lo que el usuario introduce.

6) Un servidor de aplicaciones es Software que ayuda al servidor Web a procesar las páginas que contienen scripts o etiquetas del lado del servidor. Cuando se solicita al servidor una página de este tipo, el servidor Web pasa la página al servidor de aplicaciones para su procesamiento antes de enviarla al navegador.

7)**WEB 2.0**

Representa la evolución de las aplicaciones tradicionales hacia aplicaciones web, enfocadas en el usuario.

Se trata de aplicaciones que generan colaboración y de servicios que reemplacen las aplicaciones de escritorio (permite que la gente escriba en internet, como por ejemplo, blogs o wikis) y comparta sus producciones (fotos, videos, etc).

8)**Lenguajes utilizados en la programación web:**

-HTML (HyperText Markup Language)

-JavaScript

-Python

-PHP (Hypertext Pre-processor)

-ASP (Active Server Pages)

-ASP.Net

-JSP (Java Server Pages)

-Ruby

9) **“HTML”** se define como (HyperText Markup Language) y HTML5 es la última evolución de la norma que define HTML. Se trata de una nueva versión de HTML, con nuevos elementos, atributos y comportamientos. Contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance

10) **CSS** son las siglas de Cascading Style Sheets - Hojas de Estilo en Cascada - que es un lenguaje que describe la presentación de los documentos estructurados en hojas de estilo para diferentes métodos de interpretación.

11) Para agregar estilos se puede:

 Estilo en Linea: Se usa el atributo "style" para definir como la etiqueta debe mostrarse. ej <h1 style=....>

 Incrustando una Hoja de Estilo: Se incorpora en <head>

 Enlazando una hoja de Estilo Externa

12) Buenas Prácticas:

* Comprobar el diseño en varios navegadores
* Depuración (Ej. firebug)
* Comentar el código
* Identar y hacer el código fácil de leer
* Usar sistema común de nombrado
* Evitar tamaños absolutos en fuentes o elementos
* Utilizar notación de colores en hexadecimal
* Ordenar los elementos según pertenezcan a cabecera, contenido principal o pie de página