PUNTO 2. DISEÑO DE INTERFACES DE USUARIO:

En muchas ocasiones va con ansiedad al manejo de una aplicación, ya que el ordenador es un elemento extraño para él. Para este tipo de usuarios novatos, el manejo de una aplicación puede ser frustrante, y por tanto ocasionar un rechazo de la misma. Surge la necesidad de estudiar cómo realizar una interfaz que evite esa frustración o rechazo. Entendemos por interfaz como, un envoltorio con el que se encuentra el usuario de nuestra aplicación que le permite manejarla, ese envoltorio es fundamental, para que nuestra aplicación sea aceptada, amigable, por cualquier tipo de usuario.

Existen una serie de factores que condicionan las interfaces, que son:

1. Tipo de usuario. Existen tres tipos de usuarios:
   1. Usuarios novicios (noob): Se caracterizan por no tener conocimientos de informática, no tienen conocimientos semánticos o sintácticos de la informática, llegan a la maquina con ansiedad. Para estas personas las aplicaciones tendrían que ser con, ayuda bien estructurada y bien desarrollada (activa y pasiva), restringir el vocabulario, pequeño número de opciones para hacer las cosas, conducir las tareas, evitar respuestas rápidas del sistema, mensajes de error constructivos, amigabilidad, mucha retroalimentación, confirmaciones…
   2. Usuarios intermedios u ocasionales: Son usuarios que tienen algunos conocimientos de informática, dominan algo la semántica y sintáctica, aunque la inmensa mayoría no conoce el desarrollo de la aplicación. Suelen aprender con cierta rapidez, y suelen recordar las acciones. La aplicación tiene que ser fluida, con respuestas más inmediatas, poca retroalimentación o confirmación, uso de macros para realizar mismas acciones, entornos no excesivamente cargados, ayuda más amplia, no es necesario manual de usuario sino ayuda mas resumida…
   3. Expertos: Son aquellos que tienen amplios conocimientos de informática, dominan la semántica y sintaxis, pueden aprender rápidamente, suelen ser profesionales por lo tanto el entorno tiene que ser profesional, las ayudas deben de ser completas, los datos de salida y entrada tienen que ser profesional, realizar acciones en pocos pasos, utilización de macros…
   4. Una aplicación de propósito general debe estar pensada, diseñada para un público en general, por lo que será necesario que el programador tome las decisiones para que su programa sea aceptado de forma general.
2. Área de trabajo: No todas las aplicaciones son iguales, y aparte del usuario que va dirigido y relacionado con él, es necesario que el programador tenga en cuenta en que área de trabajo va a ser explotada la aplicación y son:
   1. Sistemas críticos: Son aplicaciones pensadas para expertos y tienen un carácter de profesionalidad muy alto (control aéreo, bomberos, vigilancia, simulación…). También tienen un alto coste, con entrenamiento previo de usuarios que son expertos, requieren una gran motivación, teniendo un carácter muy profesional.
   2. Industrial y comercio: Se trata de aplicaciones no críticas que se basan en el manejo de BBDD, en el ámbito de los negocios. Están pensadas para usuarios intermedios y expertos, que han tenido que ser formados para su uso, destacando reserva de plazas de avión, aplicaciones bancarias, industriales (en cuanto a gestión), inventarios, etc. Deben ser: sin gran motivación, fáciles de aprender, entorno amigable, etc.
   3. Oficinas, hogares y entretenimiento: Área de trabajo que está enfocada a todos los tipos de usuarios. Procesadores de texto, juegos, contabilidad domestica. Son aplicaciones de no alto coste, fáciles de entender, gran fiabilidad, atractivos para la venta, no necesita formación. La aplicación debe ser de carácter general que valga para todo el mundo.
3. Factores humanos: El ser humano es limitado, en factores como el tiempo de concentración, percepción visual, memorizar acciones, sicología cognitiva de la lectura, etc.

Estos factores hay que tenerlos en cuenta cuando se hace el proyecto, para ello el programador debe ser capaz de utilizar una serie de mecanismos que los entornos visuales ponen a su disposición para evitar ese rechazo. Estos entornos destacan en aspectos como visuales (imágenes, tipo de letras…), auditivos (sonidos, música…), táctiles (manejo del ratón, teclado o pantallas táctiles) pueden hacer que los usuarios no sufran un cansancio y por tanto un rechazo. Sera necesario que el programador encuentre el punto en el que los factores humanos que evite que los usuarios se cansen de nuestra aplicación, sin abusar de sus contenidos.

Principales criterios, directrices para el diseño de interfaces:

Una aplicación está basada en tres categorías o vistas que el usuario va a trabajar cada una de ellas tiene unos criterios de desarrollo diferentes, estas vistas con:

1. Interacción general a nuestro proyecto: Suele ser la pantalla o el formulario más utilizado de la aplicación, es un interface que no se puede ignorar, lo que obliga a ser constantemente visitado. Tienen que ser consistente en cuanto a formatos, tamaños, colores, etc., debe tener un orden coherente, debe haber confirmaciones para realizar ciertas acciones, reducción de la cantidad de información a memorizar, es necesario perdonar los errores, tiene que existir de alguna forma cómo podemos ir atrás si se necesita, eficacia en el diálogo, necesaria organización de la geografía del formulario, utilización de verbos, adjetivos… cortas y significativas, no poner faltas de ortografía como tiene Microsoft, ayuda general
2. Interacción de entrada de datos: Esta influenciada por el manejo de teclado y ratón o pantallas táctiles, en cuanto al teclado existe la posibilidad de poder utilizar combinaciones de teclas, Ctrl-x (cualquier función)…, el ratón permite, a la hora de meter datos, jugar con menús emergentes, ambos usos se denominan eventos de teclado y ratón. Se pueden destacar las siguientes características:

Entrada de datos: Dirección del usuario, minimizar el número de acciones de entrada de datos, ser coherente en tamaños, facilitar la introducción de datos con objetos específicos, ser flexible en cuanto a la introducción de los datos, validar los campos, longitudes de campos con cajas de texto, coherencia con las acciones a realizar, eliminar las entradas innecesarias.

1. Interacción de salida de datos: Vuelven a intervenir cuestiones de color, tamaño, longitudes… a la hora de visualizar los datos. Cosas a tener en cuenta: Debe aparecer la información más significativa, no abrumar al usuario con datos, no abusar de siglas, acrónimos, palabras recortadas, buen uso de mayúsculas y minúsculas, utilizar formularios o ventanas auxiliares para presentar la información, adecuadas geografías de los formularios con la intención de hacerlo fácil al usuario.