

# INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

## ***¿Qué es el diseño de una interface de usuario (UI)?***

- Basado en el diseño de computadoras, aplicaciones, máquinas, dispositivos móviles, aplicaciones de software y sitios web enfocados en la experiencia del usuario y la interacción.
- Es una actividad multidisciplinaria que involucra varias ramas de diseño como diseño gráfico, industrial, web, de software y ergonomía; dicha actividad se encuentra implicada en un amplio rango de proyectos, desde sistemas para computadoras, vehículos y hasta aviones comerciales.

## ***¿Cuál es el objetivo de la UI?***

- El objetivo de la UI es mantener la interacción con los destinatarios (usuarios) de una forma más atractiva, centrando el diseño en ellos.
- El diseño gráfico y diseño industrial basan sus conocimientos para que los usuarios aprendan lo más rápido posible el funcionamiento del software que van a usar.
- Las herramientas principales que utilizan son recursos como la gráfica, pictogramas, estética, simbología, sin afectar el funcionamiento técnico eficiente.
  - Los ingenieros de software tienen que estar del lado técnico del software.

## ***UI: Conceptos iniciales***

- La UI es la categoría de diseño que crea un medio de comunicación entre el hombre y la máquina.
  - ➔ Con un conjunto de principios se crea un formato de pantalla.
  - ➔ Vemos necesario estudiar las preferencias de las personas para producir tecnología que se adapte al ser humano.
  - ➔ Si bien se ve necesario lo anterior, en la actualidad tendemos a estudiar sólo a la tecnología. El resultado de ello es que se termina exigiendo a las personas que se adapten a la tecnología.
- Una interfaz difícil de utilizar provoca errores de parte de los usuarios o que los mismos rechacen usar el sistema.
- Personas diferentes pueden tener distintos estilos de percepción, comprensión y trabajo. Se genera una diversidad.
- La interfaz debe brindar un rápido acceso al contenido y conocimiento del sistema, de forma que no se pierda comprensión mientras el usuario se va desplazando a través de la información.

# INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

## ***A tener en cuenta***

- La variedad existente de tecnologías que se deben adaptar al usuario:
  - Hipertextos, sonidos, presentaciones en 3D, videos, realidad virtual, etc.
- Configuraciones de hardware:
  - Teclado, mouse, dispositivos de presentación gráfica, lápices, anteojos de realidad virtual, reconocimiento de voz, etc.
- Variedad de dispositivos:
  - PC, equipos específicos, celulares, televisiones, etc.

## ***6 principios para el diseño de UI***

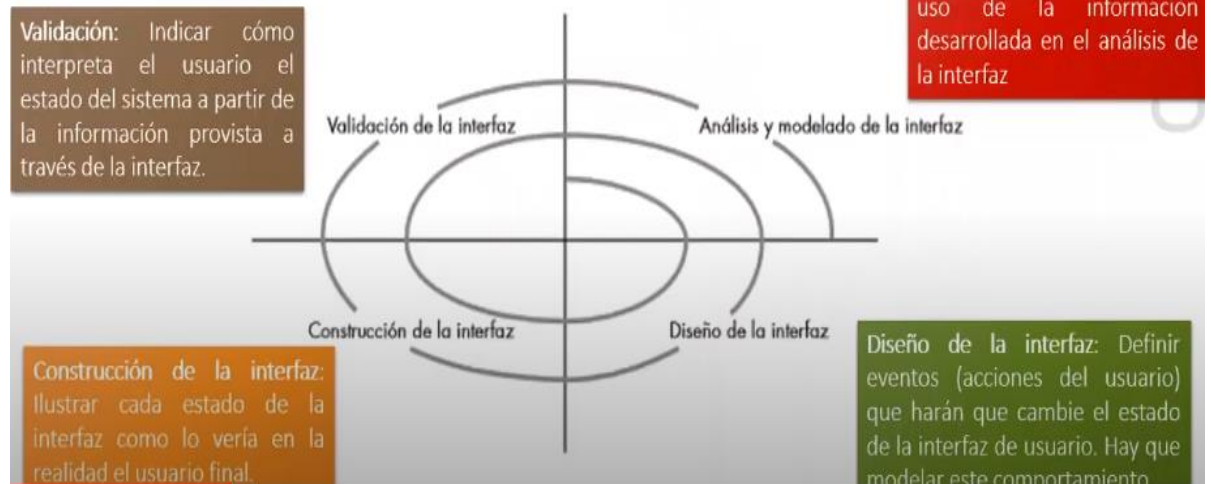


- Analizar y comprender que hace el usuario:
  - Para tener una idea de a donde apuntar el diseño.
- Prototipo en papel:
  - Realizamos un bosquejo en papel de como se vería nuestro software visualmente.
- Evaluación de diseño del prototipo:
  - Esto nos dirá si el prototipo inicial en papel es el correcto. Si es el correcto puedo pasar a diseño de prototipo.
- Realizar el diseño dinámico del prototipo:
  - Hacer la aplicación incompleta con la parte gráfica.
- Evaluación de diseño dinámico:
  - Se evalúa con usuarios finales y estos nos darán el feedback.
- Se implementa la interfaz definitiva:
  - Creando la aplicación más completa y agregándole base de datos.

# INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

## Diseño de UI: proceso

- El proceso de análisis y diseño de interfaz es iterativo.



## Diseño de experiencias de Usuario (Udx) -> UX

- El diseño de experiencias de usuario es un conjunto de métodos que se aplican al proceso de diseño y dichos métodos buscan satisfacer necesidades del cliente de forma que se proporcione una buena experiencia para los usuarios finales.



# INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

## *Diseño de interfaces UI y UX – Fuentes de información*

Tomamos información de diferentes fuentes que permiten estudiar al usuario:

- a. Encuestas.
- b. Información de ventas.
- c. Información de mercado.
- d. Información de charlas de apoyo al usuario.

Tomamos en cuenta

- » Franja de edad
- » Etnia
- » Género
- » Experiencia
- » Nivel de ingresos
- » Idioma
- » Nivel de Estudios
- » Localización
- » Ocupación o profesión
- » Religión

- Mientras más información, mejor nuestro diseño.

## **Contexto y ambiente de trabajo**

- Se estudian las partes implicadas en el sistema.
- Se revisan las competencias del producto.
- Recorridos del usuario dentro del sistema físico o virtual.
- ¿Qué trabajo realizará el usuario en circunstancias específicas?
- ¿Cuáles tareas y subtareas se efectuarán cuando el usuario haga su trabajo?
- ¿Qué dominio de problema específico manipulará el usuario al realizar su labor?
- ¿Cuál es la secuencia de tareas? ¿Cuál es la jerarquía de las tareas?
- **La etapa de revisión del contexto y ambiente se realiza en paralelo mientras se genera la especificación de requerimientos.**

# INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

---

## *Aspectos de diseño de interfaz*

### *Reglas básicas de diseño (son DORADAS)*

- Dar control al usuario.
- Reducir la carga de memoria del usuario.
- Lograr una interfaz consistente.
- Sumamos factores humanos

### *Dar control al usuario*

El usuario busca que el sistema reaccione ante sus necesidades y lo ayude a hacer sus tareas. Como logramos esto:

- ➔ Definir modos de interaccionar para que el usuario no haga acciones innecesarias.
- ➔ Proporcionar interacción flexible.
- ➔ Incluir operaciones de interrumpir y deshacer.
- ➔ Depurar la interacción a medida que aumenta la destreza del usuario.
- ➔ Ocultar al usuario ocasional los elementos técnicos internos.
- ➔ Diseñar interacción directa con los objetos que aparecen en pantalla.

### *Reducir la carga de memoria (literal, memoria cerebral) del usuario*

- ➔ Reducir la demanda a corto plazo.
- ➔ Definir valores por defecto que tengan un significado.
- ➔ Definir accesos directos que sean intuitivos.
- ➔ El formato visual de la interfaz debe basarse en una metáfora de la realidad.
- ➔ Desglosar información de forma progresiva.

### *Lograr interfaz consistente*

- ➔ Utilizar mismas reglas de diseños y estilos para las mismas interacciones.
- ➔ Mantener modelos que sean prácticos para el usuario, a menos que dichos modelos sean imprescindibles cambiarlos.
- ➔ Mantener una consistencia en toda la familia de aplicaciones.
- ➔ Permitir que el usuario incluya una tarea actual en un contexto que posea algún significado.
- ➔ El usuario debe tener la capacidad de determinar de donde viene y hacia donde puede ir.

# INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

## **Factores humanos**

- Percepción visual/auditiva/táctil.
- Memoria humana.
- Razonamiento.
- Capacitación.
- Comportamiento/habilidades personales.
- Diversidad de usuarios.
  - Usuarios casuales: necesitan interfaces que los guíen.
  - Usuarios con experiencia: requieren interfaces ágiles.

## **Usabilidad – concepto**

- La usabilidad no proviene de la estética, de mecanismos de interacción o de interfaces inteligentes. **La usabilidad es obtenida cuando la arquitectura de la interfaz se ajusta a las necesidades de las personas que la usarán.**
- No se llega a una definición formal de usabilidad. Es parte de lo semántico del software y las necesidades de las personas.

»Donahue la define: “La usabilidad es una medida de cuán bien un sistema de cómputo [...] facilita el aprendizaje, ayuda a quienes lo emplean a recordar lo aprendido, reduce la probabilidad de cometer errores, les permite ser eficientes y los deja satisfechos con el sistema.”

## **¿Cuándo existe la usabilidad?**

En base a las respuestas que den los usuarios que usen el sistema sobre las siguientes preguntas:

- » ¿El sistema es utilizable sin ayuda o enseñanza continua?
- » ¿Las reglas de interacción ayudan a un usuario preparado a trabajar con eficiencia?
- » ¿Los mecanismos de interacción se hacen más flexibles a medida que los usuarios conocen más?
- » ¿Se ha adaptado el sistema al ambiente físico y social en el que se usará?
- » ¿El usuario está al tanto del estado del sistema? ¿Sabe en todo momento dónde está?
- » ¿La interfaz está estructurada de manera lógica y consistente?



## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

- » ¿Los mecanismos, iconos y procedimientos de interacción son consistentes en toda la interfaz?
- » ¿La interacción prevé errores y ayuda al usuario a corregirlos?
- » ¿La interfaz es tolerante a los errores que se cometen?
- » ¿Es sencilla la interacción?

Si cada una de estas preguntas obtiene un “SI” como respuesta, es probable que se haya logrado la usabilidad

### ***Principios de usabilidad de Jacob Nielsen***

- Los principios de diseño de Nielsen enuncian el diálogo correcto que debe proveer una interfaz de usuario.
- Estos principios fueron desarrollados por Jacob Nielsen y se utilizan para el diseño de interfaces y, como métricas de evaluación de interfaces que ya han sido desarrolladas.

1. Dialogo simple y natural
2. Lenguaje del usuario
3. Minimizar el uso de la memoria del usuario
4. Consistencia
5. Feedback
6. Salidas evidentes
7. Mensajes de error
8. Prevención de errores
9. Atajos
10. Ayudas

### ***Dialogo simple y natural***

Forma en que la interacción con el usuario debe llevarse a cabo.

## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

- » Realizar una escritura correcta, sin errores de tipeo
- » No mezclar información importante con la irrelevante
- » Distribución adecuada de la información
- » Prompts lógicamente bien diseñados
- » Evitar el uso excesivo de mayúsculas y de abreviaturas
- » Unificar el empleo de las funciones predefinidas

### ***Lenguaje del usuario***

Emplear en el sistema un lenguaje familiar para el usuario, usar el lenguaje del usuario.

- » No utilizar palabras técnicas, ni extranjeras
- » Evitar el truncamiento excesivo de palabras
- » Diseñar correctamente las entradas de datos
- » Emplear un grado adecuado de información (ni excesivo ni escaso)

### ***Minimizar el uso de la memoria del usuario:***

Evitar que el usuario esfuerce su memoria para interactuar con el sistema.

- » Brindar Información de contexto
- » Brindar información de la navegación y sesión actual
- » Visualización de rangos de entrada admisibles, ejemplos, formatos

### ***Consistencia***

Que no hayan ambigüedades en el aspecto visual ni tecnológico en el diálogo o en el comportamiento del sistema.

- » La consistencia es un punto clave para ofrecer confiabilidad y seguridad al sistema
- » Debe existir una consistencia terminológica y visual

### ***Feedback***



## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

Respuesta gráfica o textual en la pantalla, frente a la acción del usuario. El sistema debe mantener al usuario informado de lo que está ocurriendo.

- » Brindar información de los estados de los procesos
- » Brindar información del estado del sistema y del usuario
- » Utilización de mensajes de aclaración, validaciones, confirmación y cierre
- » Realizar validaciones de los datos ingresados por el usuario

### ***Salidas evidentes***

Que el usuario tenga a su alcance de forma identificable y accesible una opción de salida.

- » Brindar salidas de cada pantalla.
- » Salidas para cada contexto.
- » Salidas para cada acción, tarea o transacción.
- » Brindar salidas en cada estado.
- » Visualización de Opciones de Cancelación, Salidas, de Suspende, de Deshacer y Modificación.

### ***Mensajes de error***

Información que brinda el sistema ante la ocurrencia de errores (es un feedback). De que forma se ayuda al usuario para que salga de la situación en la que se encuentra.

- » Deben existir mensajes de error para ser usados en los momentos que corresponda.
- » Brindar Información del error, explicar el error y dar alternativas a seguir.
- » Se deben categorizar los diferentes tipos de mensajes.
- » No deben existir mensajes de error intimidatorios.
- » Manejar adecuadamente la forma de aparición de los mensajes.

### ***Prevención de errores***

Evitar que el usuario llegue a un error.

## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

- » Brindar rangos de entradas posibles para que el usuario seleccione y no tenga que tipear (escribir)
- » Mostrar ejemplos, valores por defecto y formatos de entrada admisibles
- » Brindar mecanismos de corrección automática en el ingreso de los datos
- » Flexibilidad en las entradas de los usuarios

### **Atajos**

La interfaz debe proveer alternativas de manejo para que resulte cómodo y amigable tanto para novatos como para experimentados.

- » Brindar mecanismos alternativos para acelerar la interacción con el sistema
- » Brindar la posibilidad de reorganizar barras de herramientas, menús, de acuerdo a la necesidad del usuario
- » Brindar mecanismos de Macros, atajos, definición de teclas de función

### **Ayudas**

Componentes que asisten al usuario. Un mal diseño de ayudas puede llegar a entorpecer y dificultar la usabilidad.

- » Deben existir las ayudas
- » Se deben brindar diferentes tipos de ayuda : generales, contextuales, específicas, en línea
- » Las ayudas deben proveer diferentes formas de lectura
- » Se deben brindar diferentes mecanismos de asistencia como búsquedas, soporte en línea, e-mail del soporte técnico, acceso a las preguntas frecuente

---

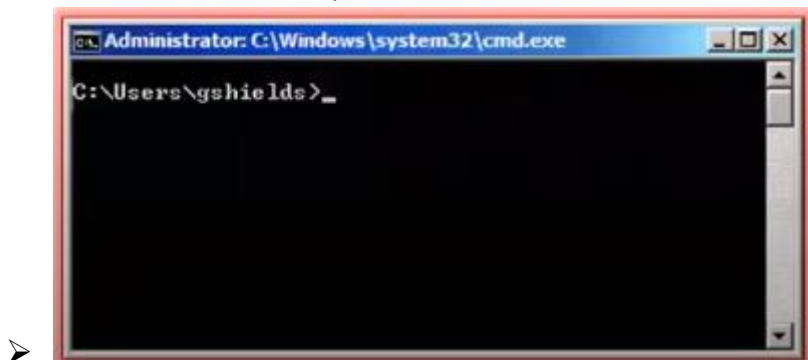
### ***Estilos de interfaces***

# INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

- » Interfaz de comandos
- » Interfaz de selección de menú
- » GUI: Interfaz gráfica de usuarios
- » Interfaz de relleno de formularios
- » Manipulación directa: Interacción directa con los objetos de la pantalla: rápida, intuitiva y fácil de aprender.
- » Interfaz de reconocimiento de voz
- » Interfaz Inteligente

## ***Interfaz de comandos***

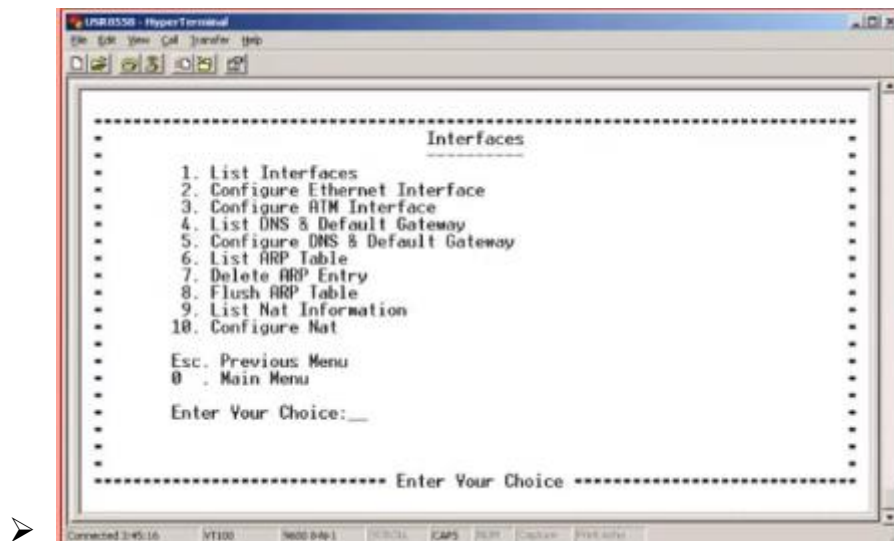
- Interfaz más elemental y básica.
  - Solo interactuamos con texto.
- Generalmente se interactúa desde una línea de comando de una consola de una aplicación en particular.
  - Usando el teclado.
- Características:
  - Poderoso y flexible.
  - Administración de errores pobre.
  - Es difícil de aprender el como usar una de estas interfaces.



## ***Interfaz de selección de menú***

- Se presenta un conjunto de opciones, que se pueden seleccionar por el usuario.
- Solo se interactúa con caracteres indicados.
- Características:
  - Se evitan errores de usuario.
  - Lento para usuarios con experiencia.

## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO



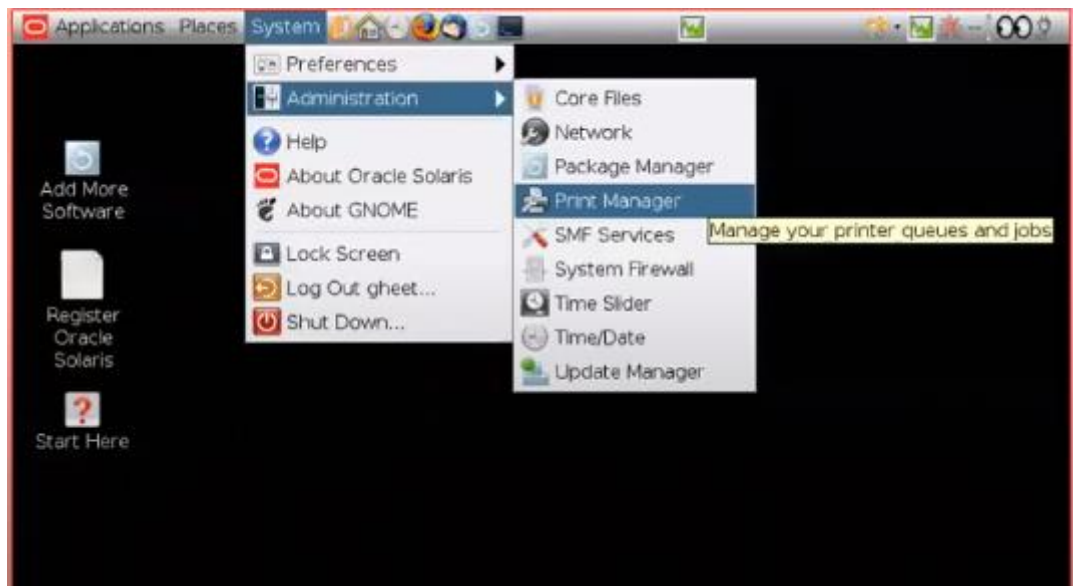
### ➤ *Interfaz gráfica de usuario*

#### ➤ Ventanas:



#### ➤ Iconos y menús:

## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO



### *Interfaz de llenado de formulario*

- Se introducen datos de forma sencilla en campos de un formulario.
- Fácil de aprender pero ocupa mucho espacio en pantalla.

A screenshot of the 'Buzón de atención' (Attention Mailbox) web form on the administración.gob.es website. The form is titled 'Buzón de atención' and includes a search bar at the top right. On the left, there is a sidebar with links to 'Contacto', 'Teléfono 960', 'Buzón de atención', 'Chat 960', and 'Twitter @960gobes'. The main content area contains a 'Gestión Mensajes' section with a 'Mensaje' field, a 'Texto (Máximo 1000 caracteres)' field, and an 'Adjuntar Fichero' section with a file selection button. Below this is an 'Identificación' section with fields for 'Nombre', 'Apellidos', 'Correo electrónico de respuesta', and 'Resposta correo electrónico de respuesta'. At the bottom right, there are 'Enviar' and 'Cancelar' buttons.

### *Manipulación directa*

- Interfaces de manipulación directa



## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO



Hardware Específico



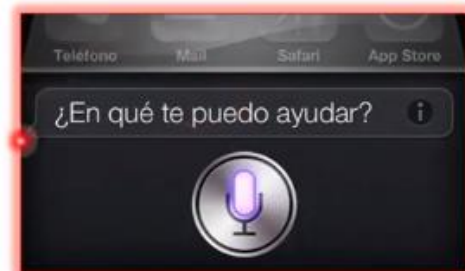
Hardware Específico y evolución a la pantalla táctil

- Interfaces de manipulación directa táctil



### **Reconocimiento de voz**

- Comunicación con dispositivos a través de la voz.





# INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

## *Interfaces inteligentes*

- Aquellas interfaces tan inteligentes que usan ejemplos para crecer y ofrecer servicios.
  - Página WorkFromAlpha.

## *Interfaces para diferentes dispositivos*



---

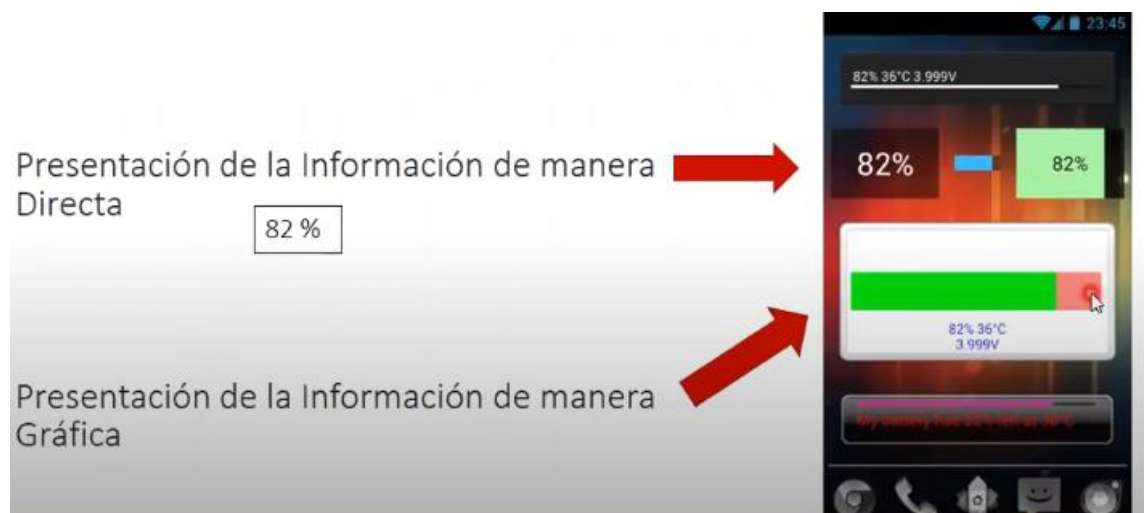
## *Presentación de información en pantalla*

### *Presentación de la información*

- » Se deben conocer los usuarios y como utilizarán el sistema.
- » ¿Información precisa o relación entre los valores?
- » ¿Es necesario presentar inmediatamente los cambios?
- » ¿El usuario realiza acciones en función de los cambios?
- » ¿Información textual o numérica?
- » ¿Información estática o dinámica?

- Mantener separada la lógica del software de la presentación y la información misma ( enfoque MVC )

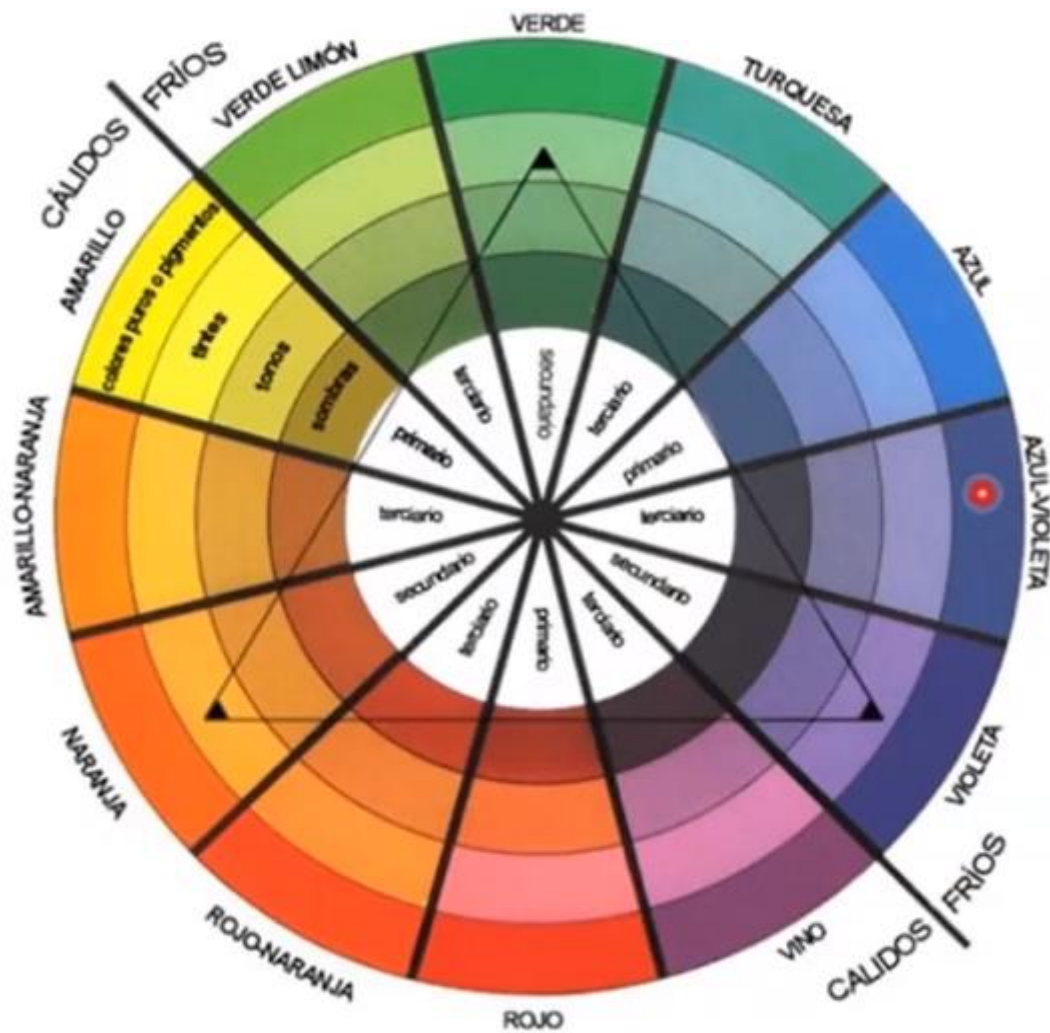
## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO



### Colores

- No utilizar más de 4 o 5 colores en una misma ventana, y no más de 7 en la interfaz total del sistema.
- Utilizar el código de colores para apoyar la tarea que los usuarios tratan de llevar a cabo.
- Ser cuidadoso el usar grupo de colores.
- Si se utilizan muchos colores o sin son muy brillantes, **el despliegue puede ser confuso**
- Dos opciones:
  - Si se elige un color y quiero dos colores, se elige el opuesto (ya que lo complementa), ejemplo si elijo amarillo-naranja, elijo azul-violeta.
  - Usar la regla del triángulo para tres colores: si elijo verde, debo elegir naranja y violeta. Si elijo turquesa, elijo amarillo-naranja y vino.

## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO

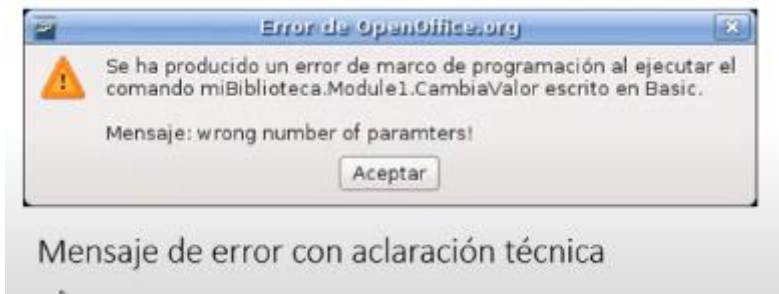


- Limitar el número de colores utilizados.
- No asociar solamente colores a significados.
  - 10% de humanos no perciben el color, acompañarlos de algún otro tipo de identificación.
- Usar los colores de forma consistente y combinarlos de forma cuidadosa.
- Usar cambio de color para mostrar cambios en el estado del sistema.

### ***Soporte al usuario***

- Mensajes del sistema por acciones del usuario.
- Ayudas en línea.
- Documentación del sistema

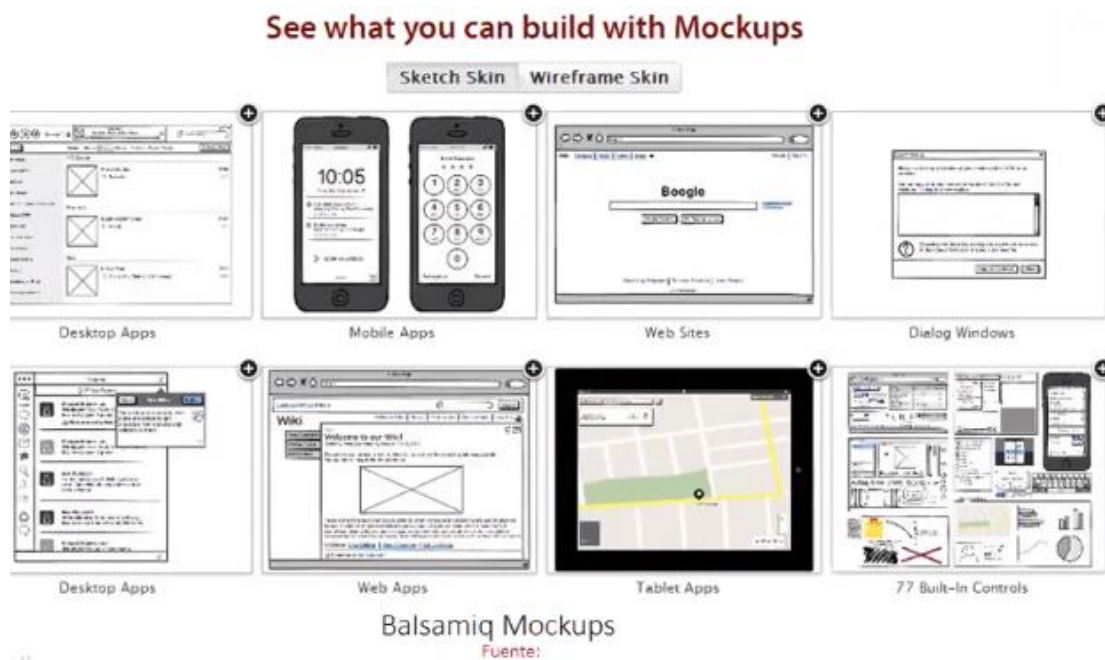
## INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO



### *Interfaces accesibles*

Son interfaces que respetan normas de diseño universal de forma que puedan ser accedidas por cualquier usuario independientemente de condiciones físicas y mentales.

### *Herramientas de prototipado*



# INGENIERÍA DE SOFTWARE 2 – CLASE 7: DISEÑO DE INTERFAZ DE USUARIO



## PENCIL PROJECT


An open-source GUI prototyping tool that's available for ALL platforms.

Pencil is built for the purpose of providing a free and open-source GUI prototyping tool that people can easily install and use to create mockups in popular desktop platforms.

The latest stable version of Pencil is **2.0.5** with **minor enhancements and bug-fixes**.

Download for Windows  
Version 2.0.5, .exe, ~22 MB

For other platforms?  
[See all downloads](#)

Mo.	ELEMENT	TYPE	DESCRIPTION
1	Logo/ search engine	Drop down list	Search by brand drop list
2	Navigation Buttons	Navigation	Direct links to relevant page
3	Navigation Buttons	Item Navigation	
4	Content	Link	Seasonal special offers
5	Special Offer	Text entry	
6	Content Link 1	Link	Link to New Release page
7	Content Link 1	Link	Link to Clearance page
8	Content Link 1	Link	Link to Bridal Registry page
9	Help	Link	
10	Shipping Information		
11	Footer		

Fuente: <http://www.smartdraw.com/>