

apuntes-ipc-estesi.pdf



LauuraBM



Interfaces persona computador



2º Grado en Ingeniería Informática



**Escuela Técnica Superior de Ingeniería Informática
Universidad Politécnica de Valencia**

2021

INTERFACES PERSONA-COMPUTADOR

UNIVERSIDAD POLITÉCNICA DE VALENCIA

LAURA BM

WUOLAH

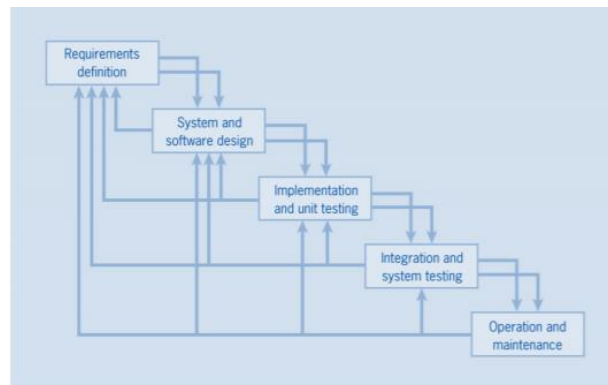
TEMA 1: INTRODUCCIÓN A LAS INTERFACES PERSONA-COMPUTADOR

La interacción persona-computador es una disciplina que aplica técnicas de la Psicología experimental a la Informática

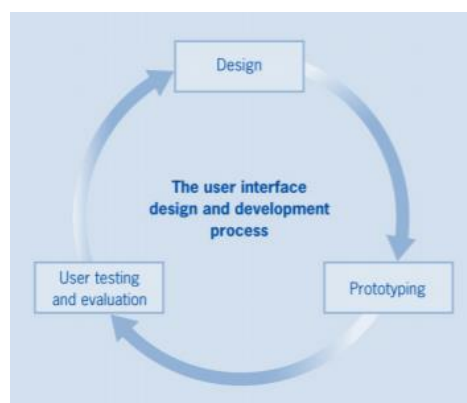
- Usuario: Persona que interacciona con un sistema informático
- Interacción: Todos los intercambios que suceden entre el usuario y el ordenador
- IPO (Interacción Persona-Ordenador): Disciplina relacionada con el diseño, implementación y evaluación de sistemas informáticos interactivos para su uso por seres humanos

Las interfaces de usuario efectivas tienen un gran impacto en la sociedad, pero, por otro lado, interfaces deficientes provocan frustración, miedo o errores.

➤ Ciclo de vida clásico para el desarrollo de programas:



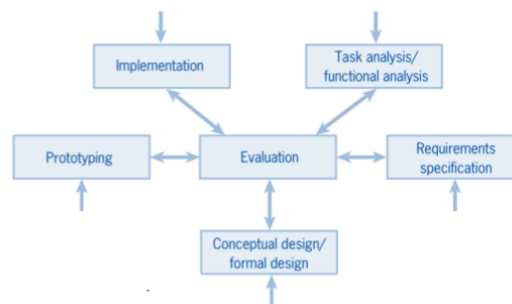
➤ Iteraciones para el desarrollo de la interfaz de usuario:



TEMA 2: ANÁLISIS DE REQUISITOS

DISEÑO CENTRADO EN EL USUARIO

- **Involucra a los usuarios durante el proceso de diseño y desarrollo**
- **Se centra en entender:**
 - usuarios
 - tareas que realizan con el sistema
 - entornos en los que se usará el sistema
- **Usuarios:**
 - Clientes, otras personas en la organización, usuarios finales, etc. (son *stakeholders*)
 - Tipos de usuarios: primarios (por ej. Trabajadores) y secundarios (por ej. Clientes)
- **Principios del DCU:**
 - Que los usuarios se impliquen activamente
 - Asignar bien las funciones entre usuario y sistema
 - Hacer buenas soluciones de diseño
 - Crear equipos de diseño con distintas disciplinas
- **Actividades en el DCU:**
 - Entender y especificar el contexto de uso
 - Especificar los requisitos del usuario y de la organización
 - Producir soluciones (prototipos)
 - Evaluar diseños con los usuarios a partir de los requisitos
- **El ciclo de vida del DCU:**



- **¿Cuándo y cómo implicar a los usuarios?:**
 - Durante la especificación de requisitos
 - Durante el prototipado
 - Justo antes de publicar el sistema
 - Después de publicar el sistema

(Cuanto más tarde impliquemos al usuario, menos cambios se podrán realizar)

ANÁLISIS DE REQUISITOS

- **Requisito:**
 - “Norma” sobre un futuro producto que dice qué debería hacer o cómo debería comportarse (pero ojo, NO dice cómo el sistema debe hacer las cosas, si no qué debe hacer)
 - Los requisitos se pueden definir con distintos niveles de abstracción
 - Se obtienen mediante observación, entrevistas o encuestas (sobre todo entrevistas)
- **2 tipos de requisitos:**
 - Funcionales: qué debería hacer el sistema
 - Ej: dar de alta productos, hacer listados de ellos, etc.
 - NO funcionales: restricciones sobre el sistema
 - Ej: debería ejecutarse en Linux y no en Windows, debe seguir la norma x de calidad, debe ejecutarse en 5 seg
 - Son funcionalidades del sistema, NO del usuario.
 - Están más asociados a la fiabilidad

DESCRIBIENDO A LOS USUARIOS

ESTUDIOS ETNOGRÁFICOS

- **Tipos de observación:**
 - Directa:
 - Estudios de campo: se toman notas sobre comportamientos en el lugar de trabajo o casa
 - Estudios controlados: el usuario interacciona con el sistema en un entorno controlado
 - Ventajas: fácil, resultados interesantes
 - Desventajas: difícil capturar toda la info, lo que no se anota se pierde, y al observar al usuario se puede alterar su comportamiento, es más subjetivo
 - Indirecta:
 - Grabar en vídeo, captura de pulsaciones de teclado/ratón, etc.
 - Ventajas: se captura todo, es más objetivo
 - Desventajas: análisis más costoso, usuarios intimidados

ESTUDIOS DE USUARIOS

- **Estudio de usuarios y dominio:**
 - Describe y clasifica a los usuarios dependiendo de sus características
 - Entrevista a usuarios reales o si no, a expertos del dominio
- **Extraer grupos de usuarios, y centrar el diseño en esos grupos:**
 - Una persona en un grupo interactuará con el sistema de manera diferente

➤ **Tipos de usuarios:**

- Usuarios noveles
- Usuarios ocasionales expertos
- Usuarios frecuentes expertos

Dependiendo del tipo de usuario ofreceremos unas interfaces u otras.

Por ej, al experto le va a frustrar no poder cambiar a interfaces más expertas, entonces hay que poner varias opciones: más fáciles para inexpertos y más complejas para expertos. Los expertos tienden a buscar shortcuts o teclas rápidas, mientras que los inexpertos como no saben cómo hacerlo rápido tiran pa lo largo pero fácil.

➤ **Entrevistas:**

Podemos hacer entrevistas para ver cuáles son los requisitos de los usuarios dentro del sistema que estamos creando. Hay que poder identificar el tipo de usuario que tenemos, saber qué preguntar y qué no.

- Muy importante: planificar bien la entrevista
 - Entrevistas estructuradas: conjunto predefinido de preguntas
 - Entrevistas flexibles: temas predefinidos, pero sin una secuencia
- Qué se debe hacer:
 - El entrevistado se debe sentir cómodo, no se debe sentir evaluado, debe sentir que sus opiniones se tienen en cuenta y que necesitas su sinceridad.
 - Si se entrevista a un grupo de personas, estos deben estar en el mismo nivel: jefes con jefes, empleados con empleados, pq si no, pueden mentir pa quedar bien
 - Realizar un pequeño estudio piloto, y luego grabar las entrevistas
 - Empezar la entrevista con una pregunta abierta
- Qué NO se debe hacer:
 - Preguntas dirigidas, en ellas se puede inferir la respuesta, manipular
 - Escenarios hipotéticos
 - Con qué frecuencia se hace algo
 - Escalas absolutas
 - Preguntas de sí o no

Hay que tener en cuenta que el entrevistado quiere quedar bien, y que no es consciente de su comportamiento.

➤ **Encuestas:**

- Son un conjunto definido de preguntas para recoger información más precisa
- Cerradas:
 - Sí/no/nsnc
 - Escala numérica entre 2 cosas opuestas
- Abiertas:
 - Sin respuesta predefinida
 - Da más info, pero es más difícil de analizar

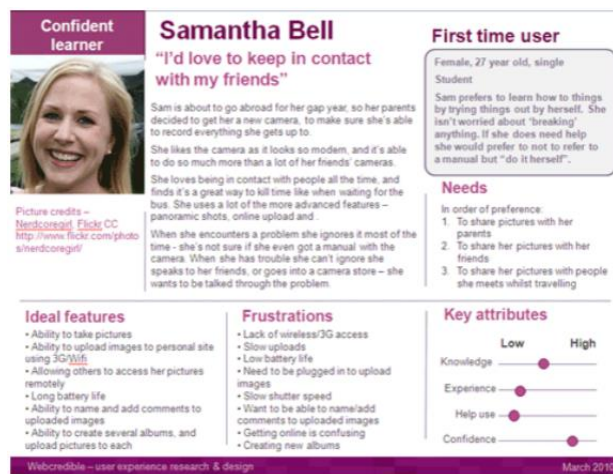
- **Un análisis competitivo es un camino fácil y rápido para establecer un punto de partida en el diseño**

Si vamos a hacer algún tipo de app, es imprescindible que como mínimo hay que buscar en Google que apps hacen lo mismo o algo parecido a la que vamos a hacer, para por lo menos, cumplir con esos mínimos.

PERSONAS

- **Modelo de una persona**

- Motivación, gustos, intenciones, comportamiento, objetivos
- Actitud frente a tecnologías y cómo usa el sistema
- Dibujo o foto de la persona para contar su “historia”
- Se construye a partir de los resultados de la entrevista con usuarios reales
- Ejemplo:



Con esto intentamos crear un modelo del tipo de usuario que tenemos, con las características del sistema que sean ideales para él.

DESCRIBIENDO TAREAS

ANÁLISIS DE TAREA

Después de identificar a los usuarios, debemos entender los objetivos que tienen al usar el sistema. El análisis de tareas estudia la funcionalidad que debe ofrecer el sistema a los usuarios para que alcancen sus objetivos.

- **Terminología:**

- Objetivo: resultado final a obtener
- Tarea: conjunto estructurado de actividades con orden
- Acción: operación o paso de una tarea

➤ **Relación entre objetivo, tarea y acción:**



➤ **Características de las tareas:**

- Cuánto cambian las tareas de una ocasión a otra
- Frecuencia
- El conocimiento necesario para realizarla
- Cómo le afectan los cambios en el entorno a la tarea
- Si tiene restricciones temporales
- Si hay peligros de seguridad asociados a la tarea
- Es individual o colectiva
- El usuario estará centrado en la tarea o estará cambiando entre tareas
- OJO: la secuencia de tareas para alcanzar un objetivo puede cambiar de usuario a usuario

ESCENARIOS DE TAREA

- **Escenario de tarea:** una descripción narrativa de una tarea, tal y como se realiza en la actualidad
- Historia sobre el uso del sistema
 - Personalizadas: describen una instancia y situación específicas
 - Detallada: describe paso a paso el procedimiento seguido por el usuario para realizar la acción, y las características y comportamiento del sistema
 - Incluye problemas y dificultades
 - Se deberían evaluar por los usuarios para comprobar que describen la tarea adecuadamente
- **Escenario de uso:** describen el uso previsto del sistema

CASOS DE USO

- **Casos de uso:** describen el uso previsto del sistema
 - Los actores son los usuarios y otros sistemas que interaccionan con el sistema que se está describiendo
 - Concretos:
 - Son parecidos a los escenarios de tarea, pero no están personalizados (son un poco más genéricos)
 - 2 columnas, una para las acciones del usuario y otra para las respuestas del sistema

ANÁLISIS JERÁRQUICO DE TAREAS

- **Análisis jerárquico de tareas:**
 - Consiste en dividir las tareas en subtareas, que luego se subdividen en sub-subtareas, etc.
 - Las subtareas se agrupan como planes que especifican cómo se realiza cierta tarea en una situación real
 - Se centra en las acciones físicas y observables realizadas
 - El punto de partida es un objetivo del usuario
- **Entorno de las tareas:**
 - Físico
 - De seguridad
 - Social
 - De la organización
 - De soporte de usuario

STORYBOARDS

Los storyboards son una secuencia de ilustraciones de eventos que explican una escena compleja, con sus personajes y acciones. Se usan para describir tareas, muestran el flujo de la interacción, y muestran las tareas que debe hacer la persona para conseguir su objetivo.

- **Los storyboards tienen que transmitir:**
 - Situación:
 - Gente involucrada
 - En qué entorno se encuentran
 - Qué tarea van a realizar
 - Secuencia:
 - Qué pasos se van a seguir
 - Qué hace que se vaya a utilizar la aplicación
 - Satisfacción:
 - Qué motiva a la gente a usar el sistema
 - Qué permite que la gente consiga su objetivo
 - Qué necesidad resuelve la aplicación

TEMA 3: DISEÑO CONCEPTUAL

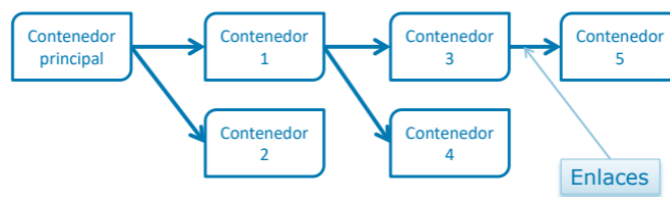
INTRODUCCIÓN

Este tema describe técnicas para pasar del análisis de requisitos al diseño físico de una interfaz de usuario

- **Reingeniería del trabajo:**
 - Desarrollar una nueva versión de un sistema existente
 - Quizás los usuarios tengan que empezar a trabajar de forma diferente
- **Objetivos:**
 - Concienciarse de la potencia y eficiencia que hace posible la automatización
 - Rediseñar el trabajo para dar un apoyo más efectivo a los objetivos del negocio
 - Aprovechar el conocimiento actual del usuario y tener en cuenta las limitaciones cognitivas y las capacidades del usuario al diseñar nuevas tareas

DISEÑO CONCEPTUAL: DIAGRAMA DE CONTENIDOS

- **Diseño conceptual:** proceso de establecer la organización y estructura de la interfaz de usuario
- **Diagrama de contenidos:** prototipo de baja fidelidad que representa la organización de una interfaz desde el punto de vista del diseñador



- Contenedor: parte del trabajo del usuario y las funciones que lo soportan
- Enlaces: navegación del usuario entre las distintas áreas funcionales dentro de la interfaz
- **Posibles relaciones entre el diagrama de contenidos y la interfaz final:**
 - Cuando se diseñan el contenido y la estructura de la interfaz el diagrama de contenidos debe servir como una guía
- **El diagrama de contenidos se crea a partir de:** información obtenida durante la recopilación de requisitos y a partir de los casos de uso concretos. Será probablemente incompleto, pero es útil para identificar las áreas funcionales y sus relaciones





DISEÑO CONCEPTUAL: PROCEDIMIENTO

- **Pasos para crear un diagrama de contenidos:**
 1. Identificar los objetos de tarea primarios, los atributos y las acciones
 2. Identificar los contenedores y los objetos de tarea que van en cada uno de ellos
 3. Conectar los contenedores para mostrar el flujo de la navegación

➤ **Paso 1: Identificar los objetos de tarea primarios, los atributos y las acciones**

- Se usan para decidir qué va en cada contenedor y para establecer los enlaces entre contenedores
- Similar al diseño orientado a objetos:
 - Objetos de tarea primarios: entidades con información o datos con las que interactúa el usuario para llevar a cabo sus tareas. Se deben buscar en la documentación de los requisitos y en los casos de uso concretos
 - Atributos: las propiedades de los objetos de tarea o enlaces a otros objetos. Hay 2 tipos:
 - *Propiedades*: datos que pertenecen exclusivamente al objeto
 - *Objetos hijo*: son objetos de tarea por sí mismos, pero a su vez atributos de otro u otros objetos
 - Acciones: funciones que puede realizar el usuario sobre los objetos de tarea. Además de las acciones específicas sobre los objetos, también se deben considerar acciones estándar como ver, crear, borrar, copiar, guardar, editar, imprimir, etc.
- Cómo usar casos de uso concretos para identificar los objetos de tarea y sus atributos
 - Subrayado simple para nombres de objetos de tarea
 - Subrayado doble para los atributos de esos objetos de tarea
- Tras identificar los objetos de tarea y los atributos, se agrupan en una tabla por objeto de tarea
- Como es difícil capturar todos los objetos de tarea, atributos y acciones de los casos de uso concretos, se debe utilizar una estrategia iterativa

➤ **Paso 2: Identifica los contenedores y los objetos de tarea que van en cada uno de ellos**

- Cada contenedor ayuda al usuario a realizar un cierto trabajo agrupando las funciones y los objetos de tarea necesarios. En el paso 3 conectaremos los contenedores con enlaces
- Elementos en un contenedor:
 - Nombre: el nombre del contenedor
 - Propósito: una frase indicando cómo va a soportar la tarea del usuario
 - Funciones:  invocadas por el usuario,  invocadas por el sistema
 - Enlaces: nombres de contenedores con los que está conectado:
 -  el nuevo contenedor sustituye al actual
 -  los dos contenedores trabajan en paralelo
 - Objetos: los objetos de tarea cuyos atributos y acciones son necesarios para el contenedor
 - Restricciones: cualquier restricción para el contenedor (velocidad, fiabilidad, disponibilidad)

- Plantilla para contenedores:

Nombre
Propósito
Funciones
● {realizadas por el usuario}
■ {realizadas por el sistema}
Enlaces
▶ {enlace sencillo}
▶▶ {enlace doble}
Objetos
Restricciones

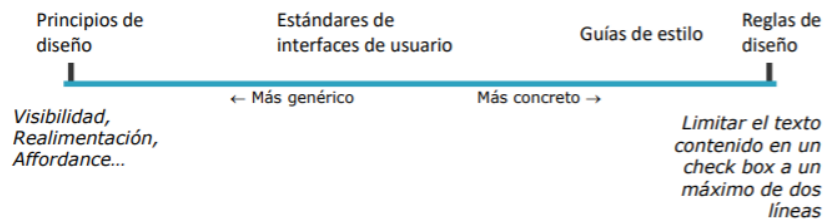
- El contenedor principal representa la primera pantalla que se encuentra el usuario. Tendrá enlaces a:
 - Tareas vitales: el usuario tendrá que realizar estas tareas rápidamente, aun en condiciones de estrés
 - Tareas frecuentes: tareas en las que el usuario emplea la mayor parte del tiempo: debe ser rápido acceder a ellas
 - Ayudas de navegación: facilitan al usuario encontrar aquello que es capaz de hacer la aplicación
- Otros contenedores derivan de los casos de uso concretos (cada caso de uso muestra la secuencia de pasos necesarios para realizar una tarea particular)

➤ **Paso 3: Conectar los contenedores para mostrar el flujo de la navegación**

- Los enlaces reflejan el orden de las acciones que sigue el usuario para realizar una tarea determinada
- Normalmente la identificación y los enlaces de los contenedores se hacen en paralelo
- Los enlaces se pueden etiquetar mediante condiciones en que el flujo puede atravesar el enlace (de interacción)
- Se deben hacer varias iteraciones sobre los prototipos correspondientes
- Para evaluar el diagrama de contenidos, se deben hacer varias pasadas por los casos de uso concretos
- Un diagrama de contenidos puede ocupar varias pantallas o varios contenedores pueden combinarse en una sola pantalla.

TEMA 4: DISEÑO FÍSICO

Diseñar una interfaz de usuario consiste en tomar muchas decisiones de diseño que se pueden basar en muchas fuentes:



PRINCIPIOS DE DISEÑO

➤ **Los principios de diseño son:**

- Generales, aplicables y duraderos
- Requieren aclaración

➤ **Principios:**

- Visibilidad: los controles deberían ser fáciles de encontrar
 - Para hacer que el uso del sistema sea más fácil, se puede incrementar la visibilidad de la interfaz (por ejemplo, poniendo accesos directos para las tareas más utilizadas, pero OJO puede hacer que parezca más complejo)
- Affordance: debería ser obvio cómo utilizar la interfaz
- Realimentación: el sistema debería decirnos qué está haciendo en cada momento, pero no poner mucho texto
- Simplicidad:
 - La IU debe ser tan sencilla como sea posible
 - Debería utilizar el mismo lenguaje que los usuarios
 - Utilizar acciones, iconos, palabras y controles que sean naturales para los usuarios
 - Descomponer tareas complejas en subtareas más sencillas
- Estructura:
 - Organizar la IU de forma que tenga significado y sea útil
 - Las cosas que los usuarios piensan que están relacionadas deberían aparecer juntas
 - Utilizar metáforas para proporcionar una estructura reconocible
- Consistencia:
 - Uniformidad en la apariencia, colocación y comportamiento, para construir interfaces que sean fáciles de aprender y recordar
 - Diferentes tipos de consistencia: acciones, objetos, color, composición, iconos, fuentes, etc.
 - OJO: a veces interesa ser inconsistente

- Tolerancia:
 - Prevenir que el usuario cometa errores y facilitar la recuperación del sistema
 - Deshabilitar los botones u opciones de menú que no estén disponibles, mostrar el formato de entrada requerido...
 - *Recuperabilidad:* cómo de fácil es para los usuarios corregir un error
 - *Recuperación hacia delante:* Aceptar el error del usuario y ayudarlo a corregirlo
 - *Recuperación hacia atrás:* Permitir a los usuarios deshacer sus acciones
 - Los buenos mensajes de error ayudan a los usuarios a corregir sus errores
 - Guías para crear mensajes de error:
 - Explicar los errores para ayudar al usuario a corregirlos
 - Proporcionar información adicional si el usuario lo solicita
 - Utilizar un lenguaje que entienda el usuario
 - Utilizar un lenguaje positivo, no amenazante
 - Utilizar términos específicos y constructivos
 - Asegurarse de que el sistema asuma la culpabilidad de los errores

➤ **Heurísticas de usabilidad de Jakob Nielsen para el diseño de la interacción:**

- 1. Visibilidad del estado del sistema
 - El sistema debería mantener siempre informado al usuario sobre lo que está pasando, mediante una realimentación apropiada.
- 2. Relacionar el sistema y el mundo real
 - El sistema debería hablar el lenguaje del usuario, en vez de usar términos tecnológicos. Seguir las convenciones del mundo real, haciendo que la información aparezca de una forma natural y lógica.
- 3. Control y libertad del usuario
 - A menudo los usuarios seleccionan funciones del sistema por error y necesitan una “salida de emergencia” para abandonar el estado actual sin tener que realizar muchos pasos. Implementar las opciones de deshacer y rehacer.
- 4. Consistencia y estándares
 - Los usuarios NO deberían tener que preguntarse si diferentes palabras, situaciones o acciones son lo mismo. Sigue las convenciones de la plataforma.
- 5. Prevención de errores
 - Hacer un diseño cuidadoso que previene los problemas. Eliminar las condiciones propensas a producir errores o pedir al usuario una confirmación antes de realizar la acción (eso solo en cosas importantes).
- 6. Reconocer en vez de recordar
 - Minimizar la necesidad de que el usuario recuerde cómo realizar tareas haciendo que los objetos, acciones y opciones sean visibles.

- 7. Flexibilidad y eficiencia de uso
 - Los aceleradores (shortcuts o teclas rápidas) permiten a los usuarios expertos trabajar más rápido, por lo que el mismo sistema se adapta a ambos tipos de usuario. Permitir a los usuarios personalizar las acciones frecuentes.
- 8. Estética y diseño minimalista
 - Los diálogos no deberían contener información irrelevante o que se necesita raramente.
- 9. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de un error
 - Los mensajes de error en lenguaje sencillo (sin códigos), indicando de forma precisa el problema, y sugiriendo de forma constructiva una solución.
- 10. Ayuda y documentación
 - Debería ser sencillo buscar dicha información, debería estar centrada en las tareas del usuario, enumerar los pasos concretos a seguir, y no ser demasiado larga (aunque es mejor que el sistema se pueda usar sin documentación).

ESTÁNDARES DE INTERFACES DE USUARIO

- **Estándares para HCI y usabilidad:**
 - **ISO 9241:** Ergonomic requirements for office work with visual display terminals (VDTs)
 - **ISO 14915:** Software ergonomics for multimedia user interfaces
 - **ISO 13407:** Human-centered design processes for interactive systems
 - **ISO/CD 20282:** Ease of operation of everyday products

GUÍAS DE ESTILO

Los principios son abstractos y se tienen que interpretar. Las reglas de diseño son más específicas, y están diseñadas específicamente para una plataforma.

- **Guías de estilo:** convenciones básicas para productos específicos o para una familia de productos. Normalmente incluyen:
 - Una descripción e ilustración de los estilos de interacción necesarios y de los controles de la interfaz de usuario
 - Guía de cuándo y cómo utilizarlos
 - Plantillas que muestran cómo debería ser el aspecto de las pantallas
- **2 tipos de guías de estilo:**
 - Comerciales:
 - Publicadas por una compañía
 - Compuestas por reglas de diseño muy específicas
 - Sólo aplicables a una plataforma concreta
 - Incluye principios y directrices
 - Personalizadas:
 - Específica para un proyecto
 - Se define en los primeros momentos del proceso de desarrollo y puede ayudar durante la captura de requisitos y la toma de decisiones
 - Promueven la consistencia a través de la interfaz de usuario
 - Ayuda a crear una imagen corporativa
 - Se pueden hacer a medida dependiendo de las circunstancias del proyecto

PRINCIPIOS DE UNA BUENA CONFIGURACIÓN

- **1. Crear grupos naturales**
 - Tener en cuenta la estructura de la información, crear grupos lógicos
 - Diferentes colores de fondo, líneas de separación, espacio en blanco, fuentes diferentes...
 - En una IGU, agrupar los controles relacionados
- **2. Separar los componentes de la actividad activa actual**
 - Enfatizar lo que el usuario está haciendo en este momento
- **3. Resaltar los componentes importantes**
 - Subrayar los componentes más importantes, pero nada más
 - ¿Cómo? Color, tipo, tamaño, posición en la pantalla, animación...
 - Combinar efectos refuerza el resultado
- **4. Utilizar el espacio en blanco de forma efectiva**
 - Con frecuencia el espacio en blanco es más efectivo que las líneas
 - Esto puede implicar que la información se divida en varias pantallas

- **5. Hacer que los controles sean visibles**
 - Los controles en pantalla deberían **sugerir cuáles son sus funciones**, aprovechando el conocimiento que tienen los usuarios de otros IU y del mundo en general
- **6. Equilibrar la estética y la usabilidad**
 - Mantener un **equilibrio entre los diseños atractivos y la usabilidad.**

DISEÑANDO UNA INTERFAZ GRÁFICA DE USUARIO

Hay que traducir el diagrama de contenidos construido durante la fase de diseño conceptual (y los otros elementos recopilados durante la fase de análisis de requisitos) a una interfaz. Pasos:

- **1. Elegir los controles para estructurar la interacción**
 - La mayoría de los IGU se organizan utilizando un contenedor de alto nivel, como las ventanas, cuadros de diálogo o pestañas
 - **Ventana Principal:**
 - Contiene un **marco**, una **barra de título**, menús, barras de scroll, etc.
 - Generalmente representan **objetos principales de tarea** (el documento en un procesador de texto, o la fotografía que se está editando)
 - Normalmente sólo hay unas **pocas ventanas principales** a las que vuelve el usuario constantemente
 - En ocasiones hay una **ventana principal que actúa como lanzadera**
 - **Ventanas secundarias:** Proporciona **funcionalidad adicional** y apoyo al usuario
 - **Cuadros de mensaje**
 - Muestran mensajes, generalmente sobre algún problema que el usuario tiene que solucionar antes de seguir trabajando
 - Los cuadros de mensaje normalmente son modales (se bloquea la interacción con el resto de la aplicación hasta que se cierran)
 - Los cuadros de mensaje no modales permiten al usuario interactuar con otras ventanas de la aplicación
 - **Cuadros de diálogo**
 - Son **invocados por el usuario**
 - Se suelen utilizar para pedir información adicional
 - Pueden ser pantallas complejas con campos de texto, botones, etc.
 - Un asistente es una serie de cuadros de diálogo en un orden determinado que guía al usuario en la realización de tareas complejas
 - **Pestañas**
 - Son útiles para clasificar las propiedades de los objetos de tarea representados en una ventana
 - La información en cada pestaña debería ser independiente
 - Retos: Utilizar demasiadas pestañas. Los usuarios pueden olvidar completar o pasar por alto la información en alguna de ellas

- Controles para controlar la interacción
 - Menús
 - Barras de herramientas
 - Complementan la jerarquía de menús
 - Contienen una serie de comandos usados frecuentemente, representados por iconos
 - La utilidad de un botón se explica en un tooltip
 - Hay varias barras de herramientas clasificadas en grupos lógicos
 - Puede ser complicado seleccionar los iconos apropiados
 - Las cintas combinan menús y barras de herramientas
 - Botones
 - Normalmente usados para controlar la operación de los cuadros de diálogo
 - Es importante usar etiquetas comprensibles
 - Tamaño: para agrupar visualmente los botones deberían tener la misma forma y tamaño. Botones en una fila pueden tener diferentes anchuras

➤ 1. Elegir los controles para estructurar la interacción

- Radio buttons and check boxes
- List Boxes
 - Permiten al usuario elegir entre un gran número de opciones
 - Selección simple/múltiple
 - Desplegable/permanente, dependiendo del espacio disponible
 - Son más flexibles que check boxes/radio buttons
 - Se deberían utilizar valores por defecto adecuados para acelerar el uso del programa
 - Se pueden combinar con cajas de texto, que pueden actuar como filtro (combo box)
- Cajas de texto
 - Es el control más flexible para introducir información
 - No es adecuado para introducir información con formato
 - Pautas:
 - El tamaño de la caja de texto debería indicar la cantidad de información esperada
 - Si el usuario puede introducir una gran cantidad de texto, utilizar una caja de texto multilínea con barras de scroll

TEMA 5: ASPECTOS HUMANOS

INTRODUCCIÓN

- Sabiendo sobre **percepción humana** podemos crear **interfaces mucho mejores**

4 PRINCIPIOS PSICOLÓGICOS

1 – LOS USUARIOS VEN LO QUE ESPERAN VER

- “Esperaba que el botón dijera Aceptar, así que lo pulsé (realmente decía Borrar)”
- Nuestro conocimiento previo nos permite dar sentido al mundo, pero nos hace manejar las situaciones inesperadas torpemente
- Desde el punto de vista del diseñador:
 - **Principio de consistencia.** Mantén en **toda la interfaz el mismo esquema de color, orden de los botones, nombres de los conceptos**, etc.
 - **Principio de aprovechamiento del conocimiento previo.** Usa **conceptos familiares para los usuarios**

2 – LOS USUARIOS TIENEN DIFICULTADES PARA CENTRARSE EN MÁS DE UNA ACTIVIDAD A LA VEZ

- “He olvidado lo que se supone que tenía que hacer a continuación”
- Los usuarios de ordenadores a menudo tienen que dividir su atención entre varias actividades simultáneas
- La **interfaz debería recordarles el siguiente paso** de lo que tienen que hacer
- El efecto de la **“Cocktail party”**
- Para centrar la atención del usuario, puedes seguir estos principios:
 - **Principio de la organización perceptual:** agrupa cosas que van juntas
 - **Principio de importancia:** la información **más importante** debería situarse en un lugar prominente, o ponerlo **más grande que el resto**, pero sin pasarse

3 – ES MÁS FÁCIL RECONOCER ALGO QUE RECORDARLO

- “No puedo recordar el nombre del comando”
- Combinamos conocimiento que está en nuestra cabeza con conocimiento que está en el mundo
- La **interfaz debería dar la información necesaria, y no confiar en la memoria del usuario**
 - Implementar menús, iconos y metáforas en pantalla en vez de una interfaz por la línea de comandos o una combinación de teclas
 - Pero tener en cuenta a usuarios avanzados

4 – ES MÁS FÁCIL PERCIBIR UN DISEÑO ESTRUCTURADO

- “¿Dónde puedo hacer clic en esta página?”
- **Principio de la organización perceptual:** agrupa cosas que van juntas

LEYES DE GESTALT

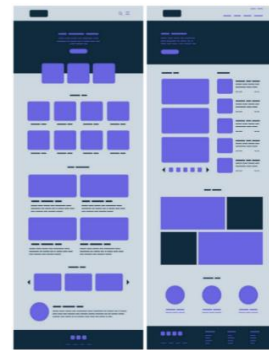
- La **psicología Gestalt** explica la **percepción humana** mediante una serie de leyes
- Somos capaces de **darle sentido al mundo** a partir de cómo se organizan sus componentes
- El cerebro es capaz de adquirir y mantener percepciones con significado a partir de un mundo aparentemente caótico, de manera que **escenas complejas se reducen en simples**

PROXIMIDAD

- **Objetos cercanos parece que forman grupos**, en vez de una colección aleatoria de individuos



- Aplicada a IU:
 - Agrupa información similar y organiza el contenido
 - El espacio en blanco crea contraste y guía al usuario en la dirección deseada, e impulsa la jerarquía visual
 - Ayuda al usuario a conseguir sus objetivos más rápidamente y profundizar en el contenido

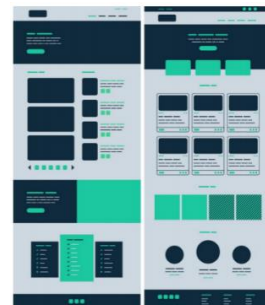


SIMILITUD

- **Elementos del mismo color o forma parecen formar grupos**



- Aplicada a IU:
 - Ayuda a organizar y clasificar objetos y asociarlos con un significado o función específicos
 - La similitud puede ser por color, forma, tamaño, textura...
 - Se puede utilizar para enfatizar un objeto haciéndolo diferente del resto (contraste) y dirigir la atención del usuario a éste



CIERRE

- **Rellenamos los huecos y áreas cerradas parece que forman un objeto**

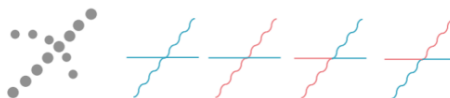


- Aplicada a IU:
 - Disminuye el número de elementos para comunicar información
 - Reduce la complejidad y hace los diseños más atractivos

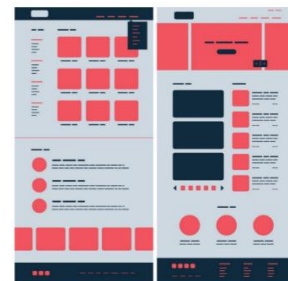


CONTINUIDAD

- **Elementos alineados parecen formar líneas**



- Aplicada a IU:
 - Los elementos que siguen una línea continua se perciben agrupados
 - La alineación de elementos nos puede ayudar a movernos a través de una página haciéndolo más legible
 - Crea orden y guía al usuario



SIMETRÍA

- **Elementos simétricos se perciben como pertenecientes a lo mismo, dándonos sensación de orden y solidez**



- Aplicada a IU:
 - Da sensación de orden y estabilidad
 - Es útil para mostrar la información de forma rápida y eficiente

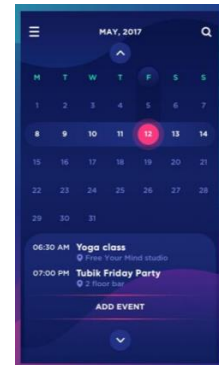


SEPARACIÓN FIGURA-FONDO

- Cuando hay bordes o diferencias en el color, textura o brillo, separamos un objeto y su fondo

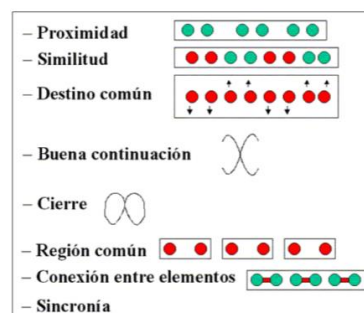


- Aplicada a IU:
 - Ciertos elementos se ven como figuras y otros como fondo
 - Hay figuras que, aunque ocupen el mismo porcentaje de lugar dentro de una imagen, se perciben como figura y no como fondo
 - Los seres humanos tenemos tendencia a separar las figuras del fondo



ORGANIZACIÓN DE OBJETOS

- **Procesos de división:** diferencian las diversas partes de una figura
- **Procesos de agrupación:** hacen que varios elementos de la escena se perciban conjuntamente
- Principios de agrupación:



PERCEPCIÓN VISUAL

COLOR

- **Evitar** la combinación de **colores oponentes**: rojo-verde, amarillo-azul, rojo-azul
- Usar **altos contrastes** de color entre la letra y el fondo
- Limitar el **número de colores** (4 para novatos, 7 para expertos)
- Usar **azul claro** sólo para las áreas de fondo
- Usar el **blanco** para la información periférica
- La **luminosidad** disminuye en el orden blanco, amarillo, cian, verde, magenta, rojo y azul
- Usar blanco, cian o verde sobre **fondos oscuros**
- **Evitar** colores muy **saturados**
- **No solo usar el color** como medio de codificación: existen múltiples problemas de visión

CONTRASTE

- Podemos **aprovechar las leyes Gestalt** de separación de un objeto y su fondo para dirigir la atención del usuario a algún punto de la interfaz
- Se **disminuye** el efecto si hay **demasiados elementos resaltados**
- El **contraste** se debería utilizar con cuidado

BRILLO

- **Luminancia**: Reacción a la cantidad de luz emitida por un objeto
 - A mayor luminancia, mayor agudeza visual
 - Una luminancia excesiva incrementa el parpadeo
 - Debe ser inversamente proporcional a la duración del estímulo
- El **ambiente luminoso** del entorno de trabajo influye en cómo vemos la pantalla:
 - Alinear las pantallas en relación correcta con las fuentes de luz
 - Nunca colocar una pantalla contra una pared o un espejo
 - Usar modo oscuro para minimizar los destellos

ACTIVIDADES AUTOMÁTICAS Y ACTIVIDADES CONTROLADAS

- **Actividades automáticas**: se pueden realizar en **paralelo con otras**
- **Actividades controladas**: necesitan **atención completa**

TEMA 6: ESTILOS DE INTERACCIÓN

INTRODUCCIÓN

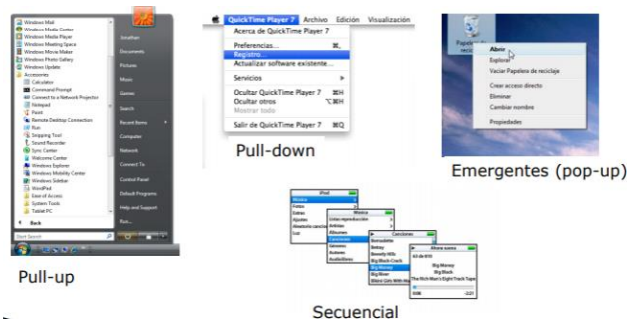
- Los **estilos de interacción** son formas en las que un usuario y un ordenador se pueden comunicar
 - Son controles de interfaces de usuario y la definición de su comportamiento
 - Proporciona una vista y un comportamiento a los componentes de la interfaz

INTERFACES DE LÍNEA DE COMANDO

- **Ventajas:**
 - Permiten dar instrucciones directas al sistema
 - Son flexibles: permiten que las órdenes tengan opciones
 - Se pueden aplicar a múltiples argumentos a la vez
 - Los usuarios expertos sienten que tienen el control, y pueden expresar acciones complejas rápida y concisamente
- **Desventajas:**
 - Son difíciles de aprender
 - Se deben recordar los comandos, puesto que las órdenes disponibles no están visibles
 - No muestran ninguna ayuda sobre cuál puede ser la siguiente acción a realizar
 - Las órdenes pueden tener una sintaxis compleja
 - Varían de un sistema a otro
 - Tienen poca tolerancia a los errores, alta tasa de errores
 - La retención a largo plazo es baja
 - Es difícil dar mensajes de error y ayuda online específicos
 - (Estos problemas se pueden reducir usando nombres de órdenes y abreviaturas consistentes)
- Los **lenguajes de órdenes** aparecieron como **medios de comunicación con el sistema operativo**
- Hoy las **interfaces de línea de comando se siguen utilizando para:**
 - Interacción con el sistema operativo
 - Administración de sistemas, lenguajes de script
 - Aplicaciones de consola
- **Guías de diseño para interfaces de línea de comando**
 - Elegir nombres de órdenes significativos y representativos
 - Seguir una sintaxis consistente
 - Usar reglas consistentes para abreviar los comandos y respuestas
 - Usar nombres de comando cortos
 - Si se pueden abreviar los comandos o las respuestas a comandos, usar abreviaturas comunes
 - Limitar el número de maneras de realizar una tarea
 - Permitir que los usuarios definan macros

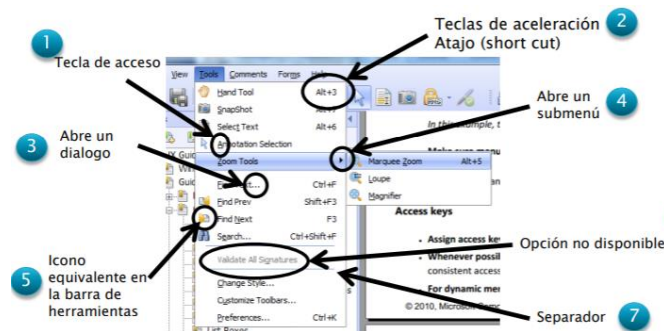
SELECCIÓN DE MENÚS

- Un **menú** es un **conjunto de opciones** entre las que tiene que elegir el usuario
- El uso de menús **evita muchos de los problemas de las interfaces por la línea de comando**
- Los **elementos** del menú deben ser **auto-explicativos y distinguibles**
- Los menús están indicados **para usuarios no expertos**: con poco entrenamiento, usuarios intermitentes, usuarios que no conocen la terminología, o que necesitan ayuda en la toma de decisiones



MENÚS AISLADOS

- **Menú binario**: si es repetitivo, ofrecer atajos y valores por defecto
- **Radio buttons**: opciones excluyentes
- **Check boxes**: múltiples opciones binarias
- **Menús desplegables**: mostrar teclas de aceleración
- **Menús emergentes**: suelen depender del contexto (**posición del ratón**)
- **En ambos casos**:
 - Posible organización de submenús en cascada
 - Opciones desactivadas en color gris
 - Opciones con teclas de acceso



OTROS MENÚS

- **Barras de herramientas**: usan **iconos** en vez de texto
- **Cintas**
- **Menús de tarta**

MENÚS CON MUCHOS ELEMENTOS

- Menús con scroll
- Combo boxes
- Menús de ojo de pez
- Barras de desplazamiento
- Menús bidimensionales

MENÚS EMBEBIDOS

- Alternativa a los menús explícitos
- Permiten ver opciones dentro de su contexto
- Precusores de hiperenlaces en www

COMBINACIÓN DE MENÚS

INTRODUCCIÓN

- **Técnicas para organizar elementos en menús**
 - Secuencias de menú
 - Menús simultáneos
 - Menús con estructura de árbol
 - Mapas de menús
 - Redes cíclicas y acíclicas de menú

SECUENCIAS DE MENÚS

- El usuario hace **una elección cada vez**
- **Facilitan la selección:** para usuarios **noveles** y tareas sencillas
- Ejemplo: **asistentes de instalación, venta de pizzas online...**

MENÚS SIMULTÁNEOS

- Presentan a la vez **varios menús en pantalla**
- Los usuarios realizan la **selección en cualquier orden**
- Para **usuarios experimentados**

MENÚS CON ESTRUCTURA DE ÁRBOL

- **Agrupar opciones en categorías** elegidas de forma natural y fácil de entender
- Usan **terminología del dominio** de la aplicación
- Recomendaciones: **4-8 elementos por menú, 3-4 niveles**

MAPAS DE MENÚS

- Se usan **cuando un menú con estructura de árbol es demasiado grande**
- Permiten una **visión global**, manteniendo el sentido de la posición
- Ejemplos: Site maps
- Otro ejemplo: mapa de atajos de programas

REDES CÍCLICAS Y ACÍCLICAS DE MENÚS

- Permiten a los usuarios **llegar a un elemento desde distintos puntos de inicio**
- Se utilizan para:
 - Relaciones sociales
 - **Rutas de transporte**
 - www

ORGANIZACIÓN DEL CONTENIDO

AGRUPACIÓN POR TAREAS EN MENÚS CON ESTRUCTURA DE ÁRBOL

- Crear **grupos de opciones similares**
- Formar **grupos que cubran todas las posibilidades**
- Asegurarse de que las **opciones no se superponen**
- Usar **terminología familiar**, pero que se diferencien los elementos

ORDENACIÓN DE LOS ELEMENTOS

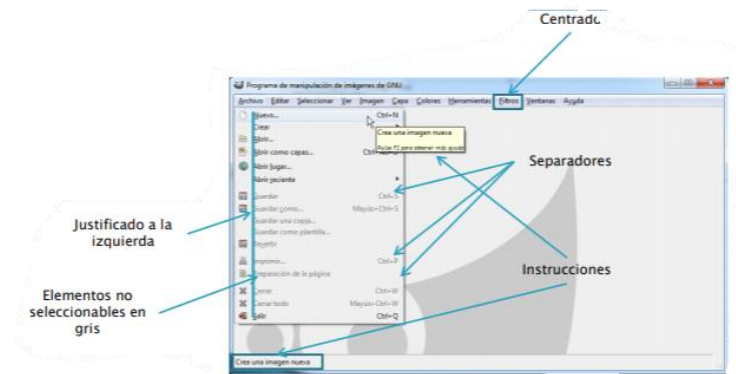
- Cómo **ordenar los elementos** de cada menú
 - Orden natural
 - Orden estándar
- Los elementos se pueden ordenar de **forma adaptativa**

DISEÑO DE MENÚS

- **Títulos:** deben ser sencillos y descriptivos
- **Nombres y formato:**
 - Usar términos familiares y consistentes
 - Asegurarse de que los términos son distintos unos de otros
 - Usar frases cortas y concisas
 - Empezar cada opción por una palabra clave

DISEÑO GRÁFICO DE MENÚS

- Definir **guías consistentes** para el diseño de menús:
 - Títulos (centrados o justificados a la izquierda)
 - Colocación de elementos (justificados a la izquierda, líneas en blanco)
 - Instrucciones
 - Mensajes de error



USO RÁPIDO DE MENÚS

- **Atajos de teclado**
 - Para usuarios expertos
 - Pueden cambiar en la traducción de la aplicación
- **Menús de tarta**
 - Si el usuario recuerda la posición del elemento a seleccionar, puede adelantarse
- **Menús Tear-off**

MENÚS DE AUDIO

- Se usan **cuando las manos y los ojos están ocupados**
- Proponen las opciones y **preguntan al usuario**
- El usuario **responde por teclado o por voz**
- Interfaz **no persistente**: el usuario debe memorizar las opciones disponibles
- El sistema debe dar **realimentación auditiva**
- Recomendaciones:
 - En una interfaz telefónica, se deberían dar como mucho 3 o 4 opciones
 - El sistema debería permitir al usuario seleccionar la opción antes de acabar la presentación del resto

MENÚS PARA PANTALLAS PEQUEÑAS

- **Aplicaciones:** entretenimiento, teléfonos móviles, PDA, etc.
- **Objetivos de diseño**
 - Aprendizaje: deben ser fáciles de aprender
 - Deben ser apropiados al dominio de la aplicación
 - Organizar opciones por importancia y frecuencia de uso
 - Simples: concentrarse en funciones importantes
 - Proporcionar realimentación: planificación frente a las interrupciones y proporcionar realimentación continua
- **Elementos**
 - Botones hardware para opciones especiales
 - Pantallas táctiles para la interacción basada en gestos
 - Iconos grandes, sencillos y en color

MENÚS EN INTERFACES DE LÍNEA DE COMANDO

- **Menús de órdenes:**
 - Muestran las opciones disponibles
 - Útiles para usuarios novatos e intermedios
 - Muestran las teclas de aceleración

FORMULARIOS

- Se usan cuando hay que introducir muchos datos de distintos tipos
- **Permiten a los usuarios**
 - moverse fácilmente entre los campos
 - dejar algunos campos en blanco
 - corregir campos ya rellenados
 - introducir información textual y numérica
- **Guías de diseño**
 - Título significativo (fácil)
 - Instrucciones comprensibles
 - Agrupación y secuencia de campos con sentido
 - Aspecto atractivo para el usuario
 - Las etiquetas de campo justificadas a la derecha, cajas de entrada alineadas verticalmente
 - Nombres familiares para los campos
 - Terminología consistente y abreviaturas
 - Espacio visible y límites en los campos de entrada
 - Movimiento del cursor adecuado
 - Corrección de errores de caracteres individuales y campos enteros
 - Prevención de errores cuando sea posible
 - Mensajes de error para valores incorrectos
 - Proporcionar realimentación inmediata
 - Resaltar claramente los campos obligatorios

- Incluir mensajes explicativos para los campos
- Indicar que se ha completado el formulario para facilitar el control del usuario

Ejemplo de formulario MAL HECHO

- Este formulario no sigue las guías:
 - Todos los campos tienen el mismo tamaño
 - Las etiquetas están lejos del campo
 - Se debe utilizar el cierre de manera no consistente
 - El botón "Delete extension" no está en una posición consistente, debería de estar a la derecha

CAMPOS CON FORMATO ESPECÍFICO

- Algunos campos requieren un **formato específico**
 - Texto (justificado a la izquierda)
 - Números (justificados a la izquierda en la entrada, justificados a la derecha en pantalla, alineación de la coma decimal)
 - Números de teléfono
 - Números de identificación, fechas, horas, monedas

CUADROS DE DIÁLOGO

- Combinan **menús y formularios**
 - Usar un título adecuado y un estilo consistente
 - Recorrido de arriba-izquierda / abajo-derecha
 - Opciones agrupadas por tema y énfasis
 - Diseño consistente
 - Terminología consistente, fuentes, mayúsculas y justificación
 - Botones estándar
- **Relación con otros elementos**
 - Deben aparecer y desaparecer suavemente
 - Tendrán un marco diferenciable pero no muy ancho
 - Serán lo bastante pequeñas para evitar tapar otros elementos
 - Mostrar junto a elementos relacionados
 - No deben ocultar otros elementos necesarios
 - Será fácil hacerlas desaparecer
 - Estará claro como completarlos o cancelarlos

MANIPULACIÓN DIRECTA

- La **manipulación directa** permite al **usuario interactuar directamente** con los objetos de la interfaz
 - Se utiliza un dispositivo de entrada continuo, como un ratón, un puntero, un joystick o una pantalla táctil
 - Ejemplo: arrastrar un fichero de una carpeta a otra
- **Características**
 - **Representaciones visuales continuas** de los objetos y acciones de interés
 - Los **objetos** de la tarea **se manipulan directamente**
 - **Acciones rápidas, incrementales y reversibles** cuyos **efectos** en los objetos de interés son inmediatamente **visibles**
 - El **usuario siente que está manipulando los objetos del dominio** y no los de la interfaz
 - Los **usuarios noveles** pueden **aprender rápido** la funcionalidad básica
- **Ventajas de los procesadores de texto WYSIWYG:**
 - Muestran una **página completa de texto**
 - **Muestran el documento en el formato que aparecerá cuando se imprima**
 - Muestran **claramente el punto de edición** mediante el cursor
 - Control natural del movimiento del cursor
 - Uso de **iconos en pantalla para las acciones más comunes**
 - Muestran el resultado de una acción de forma inmediata
 - **Respuesta rápida**
 - Ofrecen acciones fácilmente reversibles
 - Fácil de aprender a usar
- Los **videojuegos** siempre han utilizado la **manipulación directa**
- **Computer Aided Design (CAD), Manufacturing (CAM), control de procesos**

PROBLEMAS DE LA MANIPULACIÓN DIRECTA

- Es un **problema para las personas con problemas de visión**
- Las **representaciones visuales** son generalmente grandes **ocupando mucho espacio en pantalla**
- Los **usuarios deben aprender** el significado de las representaciones visuales
- La representación visual **puede ser engañosa**
- **Escribir comandos puede ser más rápido** para los usuarios expertos
- **Problema con las pantallas pequeñas**
- **Difícil** en algunas aplicaciones

INTERFACES AVANZADAS

INTERFACES POR LENGUAJE NATURAL

- El usuario interactúa con el ordenador por medio de un lenguaje natural y familiar para dar instrucciones y recibir respuestas
- El usuario puede escribir o hablar
- No hay sintaxis u ordenes que aprender
- No está claro cuando es deseable
 - Los ordenadores pueden mostrar información 1000x más rápido que el usuario introduciendo ordenes
 - Los usuarios principiantes e intermedios prefieren elegir entre un conjunto visible de opciones
 - Los expertos prefieren un lenguaje de ordenes preciso y conciso
 - Falta de contexto
 - Requiere aclaraciones que ralentizan la interacción

REALIDAD VIRTUAL

- Sistema interactivo que ofrece una percepción sensorial de un mundo sintético que sustituye completamente al real en el canal sensorial elegido
- Rompe las limitaciones físicas del espacio y permite a los usuarios actuar como si estuvieran en otro lugar

REALIDAD AUMENTADA

- Una combinación de una escena real y elementos virtuales
- Los elementos virtuales se generan por ordenador e incluyen en la escena real información adicional
- Los movimientos del usuario deben capturarse reflejarse en la nueva información a visualizar

REALIDAD AUMENTADA Y VIRTUAL

- El éxito de entornos virtuales y aumentados depende de la integración de:
 - Pantallas
 - Localización de la cabeza
 - Localización de las manos
 - Realimentación de fuerzas
 - Entrada y salida de sonido
 - Otras sensaciones
 - Realidad virtual y aumentada cooperativa y competitiva

TELEOPERACIÓN

- Dos “padres”: manipulación directa en ordenadores personales y control de procesos en entornos complejos
- La operación física es remota
- Factores que complican la teleoperación:
 - Retrasos
 - En la transmisión
 - En la operación
 - Realimentación incompleta
 - Interferencias imprevistas
 - Recuperación de errores complejos

COMPARACIÓN DE ESTILOS DE INTERACCIÓN

ESTILO DE INTERACCIÓN	MÁS	MENOS
Lenguaje de órdenes	<ul style="list-style-type: none">• Flexible• Llama a los usuarios “potentes”• Apoya la iniciativa del usuario permitiendo la creación de macros	<ul style="list-style-type: none">• Pobre manejo de errores• Requiere entrenamiento importante y memorización
Selección de menús	<ul style="list-style-type: none">• Fácil de aprender• Necesita menos pulsaciones de teclas que los lenguajes de órdenes• Facilita la toma de decisiones• Bueno para principiantes y usuarios esporádicos	<ul style="list-style-type: none">• Peligro de muchos menús• Puede retrasar a usuarios frecuentes• Consume espacio de pantalla
Formularios	<ul style="list-style-type: none">• Simplifica la entrada de datos• Requiere un entrenamiento modesto• Ayuda a los usuarios con valores por defecto	<ul style="list-style-type: none">• Consume espacio de pantalla
Manipulación directa	<ul style="list-style-type: none">• Presenta conceptos visualmente• Facilita el aprendizaje• Facilita la retención• Permite evitar errores• Anima a la exploración	<ul style="list-style-type: none">• Necesita pantallas gráficas y dispositivos de entrada continua• Los iconos y metáforas pueden tener distintos significados para diferentes usuarios
Interfaces por lenguaje natural	<ul style="list-style-type: none">• Alivia la carga del aprendizaje de la sintaxis	<ul style="list-style-type: none">• Puede ser impredecible• Difícil de implementar

TEMA 7: USABILIDAD

INTRODUCCIÓN

- Las **buenas interfaces:**
 - refuerzan la confianza de los usuarios
 - permiten que el usuario tenga claro el estado del sistema y que sepan el resultado de sus acciones
 - Permiten que el usuario se concentre en su trabajo
- **Hay otras partes del sistema que están ocultas**, como la base de datos
- **Propiedades de un buen sistema informático**
 - Exactitud, disponibilidad, eficiencia, seguridad, facilidad de uso, mantenibilidad
 - Pero normalmente los desarrolladores se centran en la funcionalidad

USABILIDAD

FACTORES

- **Propiedades de una interfaz usable:**
 - Adecuación al uso. El sistema permite realizar las tareas del usuario
 - Facilidad de aprendizaje (para distintos usuarios)
 - Eficiencia en la tarea. Para usuarios frecuentes
 - Facilidad de recordar
 - Satisfacción subjetiva
 - Reducción errores
- Es difícil diseñar un sistema que sea excepcional en todos los aspectos, por lo que es importante **decidir qué aspectos son los más importantes para nuestro sistema**

OBJETIVOS Y MEDIDAS

- **Estándar ISO 9241: Ergonomics of HumanSystem Interaction. Parte 11: Guidance on usability**
 - usabilidad: grado en que un producto puede ser utilizado por usuarios especificados para lograr objetivos concretos con eficacia, eficiencia y satisfacción, en un determinado contexto de uso
 - eficacia: precisión y completitud
 - eficiencia: recursos consumidos
 - satisfacción: ausencia de incomodidad, actitud positiva del usuario

➤ **ISO 9241: medidas para otros objetivos de usabilidad**

OBJETIVOS	MEDIDAS DE EFECTIVIDAD	MEDIDAS DE EFICIENCIA	MEDIDAS DE SATISFACCIÓN
	Porcentaje de objetivos alcanzados	Tiempo para completar una tarea	Escala de evaluación de la satisfacción
	Porcentaje de usuarios que completan una tarea con éxito	Tareas completadas por unidad de tiempo	Frecuencia de uso discrecional
	Precisión media de las tareas completadas	Coste económico de completar una tarea	Frecuencia de quejas
Satisface las necesidades de los usuarios entrenados	Número de tareas avanzadas realizadas; Porcentaje de funciones apropiadas utilizadas	Eficiencia relativa en comparación con un usuario experto	Escala de evaluación de la satisfacción en relación a las funciones avanzadas
Satisface las necesidades de usuarios no entrenados	Porcentaje de tareas terminadas con éxito al primer intento	Tiempo empleado en el primer intento*; Eficiencia relativa en el primer intento	Tasa de uso no obligatorio
Satisface las necesidades de uso poco frecuente o intermitente	Porcentaje de tareas terminadas con éxito tras un periodo de no utilización	Tiempo empleado en volver a aprender las funciones*; Número de errores recurrentes	Frecuencia de reutilización
Minimización del uso de los elementos de ayuda	Número de consultas a la documentación; Número de llamadas a soporte técnico; Número de accesos a la ayuda	Tiempo productivo*; Tiempo de aprendizaje de una tarea*	Escala de evaluación de la satisfacción en relación a los elementos de ayuda
Facilidad de aprendizaje	Número de funciones aprendidas; Porcentaje de usuarios que consiguen el criterio de competencia	Tiempo de aprendizaje para alcanzar el criterio de competencia* Tiempo de reaprendizaje*; Eficiencia relativa durante el aprendizaje	Escala de evaluación de la facilidad de aprendizaje
Tolerancia a errores	Porcentaje de errores corregidos o registrados por el sistema; Número de errores de uso tolerado	Tiempo empleado en corregir los errores	Escala de evaluación del tratamiento de errores
Legibilidad	Porcentaje de palabras leídas correctamente a la distancia normal de visión	Tiempo necesario para leer correctamente un número de caracteres dado	Escala de evaluación de las molestias visuales

- **Compromisos:**
 - Se puede aumentar el tiempo de aprendizaje para incrementar la rapidez de ejecución
 - Si hay que reducir la tasa de errores, la velocidad de ejecución se puede ver reducida
- Hay que **especificar los objetivos del proyecto**, para poder explicar las decisiones tomadas
- **Es más fácil medir la usabilidad del sistema una vez que se ha construido, pero puede ser demasiado tarde**
- Los usuarios y los diseñadores pueden **probar la interfaz mediante prototipos**
- Los **prototipos** pueden ser:
 - de baja fidelidad: mock-ups en papel
 - de alta fidelidad: prototipos interactivos
- **1- Se escribe la documentación y la ayuda online antes de construir la interfaz**, para refinar el diseño
- **2- Se implementa la aplicación** con las herramientas adecuadas
- **3- Las pruebas de validación certifican que el sistema cumple** con los requisitos

ÁREAS DE APLICACIÓN

- Todo sistema informático se beneficia de un interfaz usable, pero en las siguientes áreas es más importante:
 - **Sistemas críticos**
 - **Usos industriales y comerciales**
 - **Aplicaciones de entretenimiento**
 - **Interfaces exploratorios, creativos y colaborativos**
 - **Sistemas socio-técnicos**

EJEMPLO DE ESTUDIO DE USABILIDAD

- **Medidas de usabilidad:**
 - **Tiempo de aprendizaje**
 - **Rapidez de ejecución**
 - **Tasa de errores de los usuarios**
 - **Retención a lo largo del tiempo**
 - **Satisfacción subjetiva**

SISTEMAS CRÍTICOS

Medidas de usabilidad	Importancia
Tiempo de aprendizaje	POCA
Rapidez de ejecución	MUCHA
Tasa de errores	MUCHA
Retención a lo largo del tiempo	POR REPETICIÓN
Satisfacción subjetiva	POCA



USOS INDUSTRIALES Y COMERCIALES

Medidas de usabilidad	Importancia
Tiempo de aprendizaje	MUCHA
Rapidez de ejecución	MUCHA
Tasa de errores	MODERADA
Retención a lo largo del tiempo	POR REPETICIÓN
Satisfacción subjetiva	MODESTA



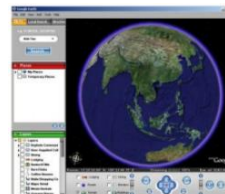
APLICACIONES DE ENTRETENIMIENTO

Medidas de usabilidad	Importancia
Tiempo de aprendizaje	MUCHA
Rapidez de ejecución	MODERADA
Tasa de errores	MUCHA
Retención a lo largo del tiempo	MODERADA
Satisfacción subjetiva	MUCHA



INTERFACES EXPLORATORIAS, CREATIVAS Y VOLABORATIVAS

Medidas de usabilidad	Importancia
Tiempo de aprendizaje	MODERADA
Rapidez de ejecución	MODERADA
Tasa de errores	MODERADA
Retención a lo largo del tiempo	MODERADA
Satisfacción subjetiva	MUCHA



SISTEMAS SOCIO-TÉCNICOS

Medidas de usabilidad	Importancia
Tiempo de aprendizaje	MUCHA
Rapidez de ejecución	MUCHA
Tasa de errores	MUCHA
Retención a lo largo del tiempo	POCA
Satisfacción subjetiva	POCA



USABILIDAD UNIVERSAL

- Crear **productos que sean usables** por gente con diferentes capacidades, que pueda usarse en muchas situaciones, siendo comercialmente práctico
- Los diseñadores de interfaces deben **tener en cuenta:**
 - Habilidades físicas y entornos de trabajo
 - Diferentes personalidades, culturas
 - Discapacidad
 - Mayores
 - Niños
 - Diversidad de hardware y software

HABILIDADES FÍSICAS Y ENTORNOS DE TRABAJO

- La ergonomía se encarga de **acomodar los entornos de trabajo a las personas**
- **Principios de diseño ergonómico (UNE-EN-614-1)**
 - a) La **altura de utilización, u otras dimensiones funcionales** de la máquina, deben estar **adaptadas** al operador y al tipo de trabajo a realizar
 - b) el **tipo, la situación y las posibilidades de regulación de los asientos** deben ser los **adecuados** para las dimensiones corporales del operador y para las tareas
 - c) **debe preverse espacio suficiente** para todas las partes del cuerpo, de forma que se pueda realizar la tarea con buenas posturas y movimientos de trabajo y se facilite el acceso y los cambios de postura
 - d) **las empuñaduras y pedales de las máquinas deben estar adaptados** a la anatomía funcional de la mano o del pie y a las dimensiones de la población de operadores
 - e) **los mandos, empuñaduras y pedales de uso frecuente, estarán situados al alcance** inmediato de las manos o de los pies del operador
- **La antropometría es la ciencia que trata de medir las dimensiones del cuerpo humano**
 - **Antropometría estática:** dan **medidas estándar** de personas de una población, típicamente de pie o sentada
 - **Antropometría dinámica:** describe **rangos de movimientos**, alcances, trayectorias
- Hay que **tener en cuenta dichas medidas para obtener interfaces usables**
 - Siempre que sea posible, permitir adaptar el interfaz. Si no, realizar distintas versiones
- **La norma UNE EN 614-1 establece:**
 - Cuando se trate de establecer espacios libres, se empleará el **valor del percentil 95 u otro mayor**
 - Para alcances, se usará el valor correspondiente al percentil 5 u otro menor

DIFERENTES PERSONALIDADES

- Hay una **gran variedad de preferencias personales**
- Diferencias entre personas

DIVERSIDAD CULTURAL E INTERNACIONAL

- **Dependiendo del tipo de cultura**, un **usuario** puede preferir **interfaces estáticas**, mientras que otro puede preferir **interfaces dinámicas**
- Además, **los gustos cambian** rápidamente
- La **necesidad de adaptar las aplicaciones a distintas lenguas** hace que las arquitecturas software deban ser flexibles

ACCESIBILIDAD

- **Distintas capacidades:**
 - visión reducida
 - audición reducida
 - movilidad reducida
- **Principios de diseño universal:**
 - Uso equitativo
 - Flexibilidad de uso
 - Uso sencillo e intuitivo
 - Información perceptible
 - Tolerancia a los errores
 - Bajo esfuerzo físico
 - Tamaño y espacio para aprox. y uso
- **Herramientas de accesibilidad:**
 - Lupas de pantalla
 - Texto a voz
 - Reconocimiento de voz
 - Advertencias visuales
 - Punteros controlados con los ojos
- Obligación de que los servicios públicos ofrezcan servicios informáticos accesibles
- **Tener en cuenta las necesidades de estos usuarios**
 - no aumenta demasiado el coste
 - aumenta la usabilidad para todos los usuarios
- **Usuarios sin discapacidad también se pueden beneficiar**

MAYORES

- La **edad** tiene **efectos negativos** sobre las **capacidades físicas y cognitivas** de las personas
- La **informática** puede proporcionar **nuevas oportunidades** de interacción social
- La **sociedad puede beneficiarse** de un acceso rápido y eficaz a la experiencia de los mayores
- **Adaptar el tamaño de las fuentes, el contraste** de las pantallas y el **volumen de los sonidos**
- También pueden **usar punteros más fáciles** de usar, **rutras de navegación más claras**, **organización consistente** de los interfaces y **comandos más sencillos**

NIÑOS

- **Su motivación principal es el entretenimiento y la educación**
 - juguetes controlados por ordenador, instrumentos musicales, herramientas artísticas
 - una vez aprenden a leer, pueden usar el teclado y usar aplicaciones de escritorio, web o móviles
 - en la adolescencia, pueden ser usuarios avanzados
- **Los objetivos** de sistemas orientados a los niños son:
 - **acelerar el proceso educativo**
 - **facilitar la socialización** con otros niños
 - **y reforzar la confianza** en uno mismo
- Los **niños buscan aplicaciones interactivas** y que le ayuden a interactuar con otros niños
- **En general,**
 - **aceptan algo de frustración o miedo, siempre que puedan volver a empezar sin mucha penalización**
 - **no aceptan ironía o humor inapropiado**
 - le gustan los **entornos que pueden explorar, con personajes conocidos y les gusta la repetición**
- En sistemas dirigidos a niños pequeños, **hay que tener en cuenta sus limitaciones y evitar**
 - **doble clic**, arrastre del ratón o botones muy pequeños
 - **textos complejos**
 - **secuencias de comandos complejas**
- También hay que **tener en cuenta su poca capacidad de concentración y poca habilidad de trabajo** con conceptos simultáneos
- En sistemas para niños conectados a Internet hay que **asegurar la seguridad**

DIVERSIDAD DE HARDWARE Y SOFTWARE

- Los sistemas se deben **adaptar a un rango grande de hardware**
 - **hardware antiguo**, con p.e., baja resolución y poca memoria
 - **hardware moderno**, con gran capacidad de almacenamiento y procesamiento
 - **dispositivos móviles**, con pantallas reducidas y memoria limitada
- **...y de software**
 - **nuevas versiones del S.O.**, navegadores web y otros programas

TEMA 8: PROTOTIPADO

INTRODUCCIÓN

- Un **prototipo** es un ejemplo que se usa como **modelo de lo que vendrá más tarde**

MOTIVACIÓN

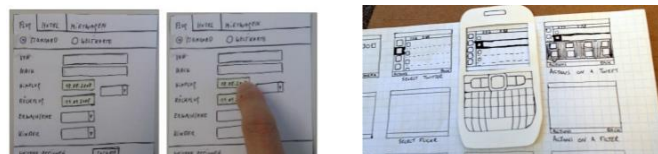
- El prototipado **permite a los diseñadores avanzar, demostrar y evaluar diseños**
- Ayuda a **concentrarse en el diseño y no en los detalles** de implementación
- Son casi siempre **incompletos, fáciles de cambiar y se descartan cuando ya no se necesitan**
- Hay **distintos tipos** de prototipos que proporcionan **distintos niveles de fidelidad**
- **De baja a alta fidelidad:**
 - **Storyboards, prototipos en papel, maquetas (mockups) digitales, prototipos dinámicos**

PROTOTIPOS DE BAJA FIDELIDAD

- **Storyboards**
 - **Se concentran en las tareas**, en lugar del interfaz
 - Se utilizan durante la **etapa de análisis de requisitos**



- **Los prototipos en papel** se utilizan en la **etapa de análisis de requisitos** y también en la parte de **diseño visual**
 - Se empieza diseñando la interfaz de usuario, dibujándola en papel
 - Los usuarios se pueden involucrar en la evaluación y la evolución del diseño



- **Directrices para los prototipos en papel:**
 - **Mantener todo el material**
 - **Trabajar rápido y hacer componentes reutilizables**
 - Si hay algo **difícil de simular** describir la **interacción de palabra**
 - **Posters grandes pueden ser útiles** para dibujar el prototipo y proporcionar contexto al usuario
 - Mezclar **hardware y software** y diferentes fidelidades
 - Si es apropiado, **añadir contexto** incluyendo elementos familiares del sistema operativo

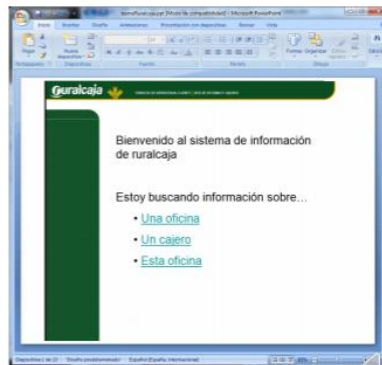
PROTOTIPOS DE BAJA A MEDIA FIDELIDAD

- **Wireframe** (o esquema de páginas o plano de pantallas)
 - Usado típicamente en diseño web, pero también en diseño móvil y de escritorio
 - Muestra la organización de pantalla
 - Se concentra en la funcionalidad, el comportamiento y la prioridad de contenidos
 - Conecta la estructura conceptual con el diseño visual



PROTOTIPOS DE ALTA FIDELIDAD

- **Maquetas (mockups) digitales**
 - Aproximaciones más detalladas al diseño final
 - Permiten una evaluación más formal



PROTOTIPOS EN VÍDEO

- Es una **excelente herramienta de comunicación**
- **Conecta decisiones del interfaz con tareas**
 - Ayuda a orientar las elecciones del interfaz, se asegura de que pienses en una interfaz completa y ayuda a detectar elementos innecesarios
- **¿Qué debe mostrar el vídeo?**
 - Toda la tarea, incluida la motivación y el éxito
 - Ilustrar las tareas más importantes
- **¿Cómo se debe hacer el vídeo?**
 - Empezar con un esquema
 - Se puede empezar a grabar para ver qué pasa
 - La cámara no es importante
 - Encontrar gente y una localización realista
 - Recuerda que lo importante es el mensaje
- **Consideraciones**
 - Usar audio o una película muda con subtítulos
 - La interfaz puede ser de papel, una maqueta, código o invisible
 - Puede mostrar tanto el éxito como el fracaso
 - Edítalo lo menos posible

EVALUACIÓN DE PROTOTIPOS

MAGO DE OZ

- **Problema: para recibir realimentación de los usuarios de una aplicación interactiva, necesitamos un prototipo que funcione**
- **La técnica del mago de Oz nos permite evaluar la interacción del usuario con los primeros prototipos**
- Hay un **operador humano** que **mueve los elementos interactivos como si fuera la aplicación**
- Permite **probar la interacción del usuario con una tecnología que todavía no existe**
- **Tiene sentido si es más rápido que desarrollar la aplicación real**
- **Retos:**
 - Interfaces de alta fidelidad que “funcionan” pueden hacer pensar al usuario que el desarrollo está hecho
 - Las interfaces de alta fidelidad dificultan que el usuario haga fuertes críticas al sistema
 - Es fácil prototipar algo que no se puede
 - Si quieres que tus usuarios crean que es un sistema real, tendrás que construir algún tipo de control remoto, que permita al mago estar escondido

TEMA 9: EVALUACIÓN DE LA INTERFAZ

INTRODUCCIÓN

- Realizar una **evaluación externa** de nuestros interfaces es imprescindible
- Dependiendo del proyecto, se debería **reservar una cantidad adecuada de recursos (5-20%)**
- Se deben **realizar pruebas de usabilidad** a lo largo del proyecto
- En sistemas de alta disponibilidad, se deben **probar situaciones de alta carga de trabajo**, e incluso de **fallos parciales**
- Existen **múltiples metodologías de evaluación**
- Recuerda que **la usabilidad se puede medir y evaluar**
- **La evaluación debería medir el nivel en el que un producto se ajusta a los requisitos de usabilidad**
 - Requisitos de usabilidad **cualitativos**
 - Requisitos de usabilidad **cuantitativos/Métricas de usabilidad**
- **Niveles para evaluar las métricas de usabilidad**
 - Actual
 - Caso mejor
 - Planeado
 - Caso peor

TIPOS DE EVALUACIÓN

- **Qué:**
 - Encontrar **tantos problemas** de usabilidad como sea posible: **evaluación diagnóstica**
 - Medir el nivel de cumplimiento de los requisitos: **evaluación por medidas**
- **Cuándo:**
 - Durante el desarrollo: **evaluación formativa**
 - Al final: **evaluación sumativa**
- **Cómo:**
 - Temprana, con **prototipos de baja fidelidad**, informal: **evaluación exploratoria**
 - Al final, verificando el cumplimiento de los requisitos, **experimento formal**: **evaluación de validación**
 - Elegir una entre varias opciones, **análisis estadístico**: **evaluación competitiva**
- **Quién:**
 - Usuarios: **estudios con usuarios**
 - Expertos: **evaluación experta**
 - Otros

ESTUDIOS CON USUARIOS

- **Selección de usuarios**
 - Normalmente 5 son suficientes
 - Idealmente usuarios reales
- **Dependiendo de dónde** se realiza el estudio:
 - En el entorno del usuario: estudios de campo
 - En otros entornos: estudios controlados

LABORATORIOS DE USABILIDAD

- **Laboratorio típico**: espacio de los observadores

TIPOS DE PRUEBAS DE USABILIDAD

- **Pruebas de campo y laboratorios portátiles**
 - Se lleva el laboratorio de usabilidad al lugar donde se usará el sistema final
 - Se deberá intentar capturar la mayor cantidad de información de cada ejecución
 - Otra opción es liberar versiones beta a un gran número de usuarios
- **Pruebas de usabilidad remotas**
 - Un gran número de usuarios prueban el sistema desde el lugar donde usarán la aplicación
 - Se puede buscar a los participantes en las bases de datos de clientes o en foros on-line
 - Las pruebas pueden ser síncronas o asíncronas
 - **Ventajas**: se puede acceder a un gran número de participantes, es barato, se prueban en el hardware del usuario
 - **Desventajas**: el control sobre el usuario es menor y se pierden sus reacciones
 - Algunos estudios dicen que este tipo de pruebas encuentran más problemas que las técnicas tradicionales
- **Prueba-a-romperlo**
 - Originario de las pruebas de videojuegos, se le pide al usuario que intente “romper” el sistema
 - Este tipo de pruebas de estrés permite robustecer las aplicaciones
- **Pruebas de usabilidad competitivas**
 - Compara la nueva interfaz con una versión anterior, o con una interfaz de la competencia
 - Se compara el tiempo en realizar una tarea o la tasa de fallos en un sistema y en otro



PREPARANDO EL EXPERIMENTO

- **Realizar un plan detallado del experimento, incluyendo:**
 - Qué se desea medir
 - Cantidad, tipos y fuentes de los participantes
 - Duración (30-90 min) y tareas que deben realizar los participantes
 - El contenido de los cuestionarios y entrevistas que se realizarán a los participantes
- **Realizar una prueba piloto con un número reducido de participantes (1-3)**
 - Nos aseguramos de que todos los detalles de la evaluación funcionan
 - Realizar la prueba en el mismo lugar y tan parecido a como se realizará el estudio real como sea posible
 - Analizar e interpretar los datos, para asegurarnos de que se ha recogido toda la información necesaria

ESTRUCTURA DE UNA SESIÓN

- **Bienvenida**
 - Los participantes deben ser tratados con respeto, y hacerles saber que no se les está evaluando
 - También deben ser informados de:
 - El objetivo del estudio
 - El procedimiento a seguir y la duración. Cómo pedir una pausa
 - Quién revisará las grabaciones y su uso después del estudio
 - Un compromiso de confidencialidad y anonimato
 - Un aviso de los peligros de la prueba
 - La posibilidad de abandonar la prueba sin penalización
 - Un modo de contacto posterior
- **Filtro de selección: nos aseguramos de que el participante se ajusta al perfil buscado**
 - su experiencia con los ordenadores, o con la tarea a realizar, motivación, nivel educativo, capacidad de comprensión
 - su agudeza visual, su mano dominante, edad, género
- **Firma del consentimiento y quizá un acuerdo de confidencialidad**
- **Encuesta filtro de selección**
- **Usar los escenarios de tarea para describir los pasos que debe seguir el usuario**
 - Seleccionar los más importantes
- **Entrevista después de la sesión**
 - Revisar las grabaciones y preguntar al usuario sobre sus intenciones y acciones
 - Encuesta
- **Incentivo**

DURANTE LA PRUEBA

- **Usa un cronómetro**
- **Acompaña cada comentario escrito durante la sesión con una marca de tiempo**
 - O usa un programa de captura de pulsaciones de teclado
 - O usa programas especializados en evaluaciones de usabilidad
- **Una técnica de prueba de usabilidad consiste en pedir al participante que exprese en voz alta su opinión**
 - **El personal debe facilitar esta comunicación y pedir pistas** sobre qué está pensando el participante. **No debe ayudarlo**
 - **Ventajas**
 - Realimentación inmediata de las opiniones de los participantes sobre la interfaz y sobre cualquier problema o sorpresa
 - Puede ayudar a los usuarios a concentrarse durante la sesión de evaluación
 - Útil para recoger datos cualitativos
 - **Desventajas**
 - Algunos participantes lo encuentran forzado y les distrae
 - Pensar en voz alta puede frenar los procesos mentales del participante, y reducir artificialmente su desempeño y tasas de error
 - Puede ser extenuante para el usuario si dura demasiado
- **Protocolo retrospectivo: preguntar a los usuarios sobre sus intenciones al realizar cada acción después de la prueba**
 - **Ventajas**
 - No interfiere con las medidas de desempeño
 - Útil para recoger datos cuantitativos
 - **Desventajas:**
 - El participante puede haber olvidado la razón por la que hizo cierta acción
 - Algunos participantes pueden sentirse intimidados por las cámaras

GRABACIONES

- Normalmente **se graba a los participantes**, para poder revisar posteriormente sus reacciones, errores, forma de trabajar, etc.
- Los métodos de **tracking visual** permiten calcular las zonas de la ventana que más atención atraen

ENCUESTAS

- Son un **método familiar y barato para capturar la opinión de un usuario o un experto**
- Permite **sondear la opinión de miles de usuarios** de una forma estructurada
- Las claves para realizar encuestas útiles es **tener claro sus objetivos** y diseñar **preguntas directas**
- Pueden ser **en papel u on-line**

➤ **Ventajas**

- Es más fácil no olvidar hacer alguna pregunta
- Es más fácil comparar las respuestas
- Se pueden recoger datos cuantitativos
- Se puede demostrar el progreso durante el desarrollo cuando el resultado de las encuestas mejora

➤ **Desventajas**

- Es difícil diseñar una buena encuesta
- Las preguntas cerradas son más fáciles de analizar, pero no dan la razón por la que el usuario eligió la respuesta

Tipos de preguntas en encuestas

- **Escala Likert.** El usuario debe mostrar su nivel de acuerdo con una afirmación: totalmente de acuerdo, de acuerdo, neutral, en desacuerdo, totalmente en desacuerdo
- **Encuesta binaria** sobre las reacciones del usuario al usar un procesador de texto (agradable o irritante)
- Otra opción, que gradúa las reacciones (Hostil 1234567 Amigable)
- **Cuestionarios tipo:**

- QUIS, Questionnaire for User Interaction Satisfaction:

PARTE 1: Experiencia con el sistema

• ¿Cuánto tiempo has trabajado con este sistema?

- ☐ menos de 1 hora
- ☐ de 1 hora a menos de 1 día
- ☐ de 1 día a menos de 1 semana
- ☐ de 1 semana a menos de 1 mes
- ☐ de 1 mes a menos de 6 meses
- ☐ de 6 meses a menos de 1 año
- ☐ de 1 año a menos de 2 años
- ☐ de 2 años a menos de 3 años
- ☐ 3 años o más

PARTE 6: Aprendizaje

6.1 Aprendiendo a usar el sistema

difícil 1 2 3 4 5 6 7 8 9 NA fácil

6.1.1 Empezando a usar el sistema

difícil 1 2 3 4 5 6 7 8 9 NA fácil

6.1.2 Aprendiendo características avanzadas

difícil 1 2 3 4 5 6 7 8 9 NA fácil

6.1.3 Velocidad de aprendizaje del sistema

lento 1 2 3 4 5 6 7 8 9 NA rápido

- System Usability Scale (SUS):

Strongly disagree 1 2 3 4 5 Strongly agree

1. I think that I would like to use this system frequently

2. I found the system unnecessarily complex

3. I thought the system was easy to use

4. I think that I would need the support of a technical person to be able to use this system

5. I found the various functions in this system were well integrated

6. I thought there was too much inconsistency in this system

7. I would imagine that most people would learn to use this system very quickly

8. I found the system very cumbersome to use

9. I felt very confident using the system

10. I needed to learn a lot of things before I could get going with this system

○ **Computer System Usability Questionnaire (CSUQ):**

1. Overall, I am satisfied with how easy it is to use this system.

STRONGLY AGREE	1	2	3	4	5	6	7	STRONGLY DISAGREE
COMMENTS:								

2. It is simple to use this system.

STRONGLY AGREE	1	2	3	4	5	6	7	STRONGLY DISAGREE
COMMENTS:								

3. I can effectively complete my work using this system.

STRONGLY AGREE	1	2	3	4	5	6	7	STRONGLY DISAGREE
COMMENTS:								

○ **Website Analysis and MeasureMent Inventory (WAMMI):**

What is your age?
choose ...

What is your gender?
☐ Male
☐ Female

Which of these browsers do you have experience with?(select all that apply)

☐ Safari
☐ Google Chrome
☐ Opera Browser
☐ Mozilla Firefox
☐ Internet Explorer
☐ something else

What do you normally visit the web site for?

Statements 1 - 10 of 20

	Strongly Agree	Strongly Disagree
This web site has much that is of interest to me.	<input type="radio"/>	<input type="radio"/>
It is difficult to move around this web site.	<input type="radio"/>	<input type="radio"/>

ANÁLISIS DE LOS RESULTADOS

- **Después de la sesión, habrás recogido:**
 - **Datos personales sobre los participantes**, notas de los observadores, grabaciones de audio o vídeo, formularios de captura de datos, datos cuantitativos sobre tiempos, errores y otras métricas de usabilidad, datos cuantitativos (y cualitativos) de las encuestas realizadas antes y después de la prueba, protocolos retrospectivos, una lista de problemas de usabilidad encontrados...
- **Un defecto de usabilidad es un problema en una interfaz de usuario que puede llevar al usuario a confusión, error, retraso o incapacidad de completar una tarea**

RESUMEN DE DATOS CUANTITATIVOS

- **Varias opciones:**
 - **Tablas, gráficas, y rankings**
 - **Estadísticas descriptivas** (media, mediana, moda)
 - **Estadísticas inferenciales** (pruebas de significancia estadística)

RECOMENDACIONES

- **Después de interpretar los resultados, se pueden realizar recomendaciones como:**
 - Puntos fuertes de la interfaz
 - Defectos a solucionar
 - Posibles defectos o puntos fuertes, pero sin pruebas suficientes
 - Áreas de la interfaz que no se han probado
 - Cambios a los requisitos de usabilidad u otros requisitos

EVALUACIÓN EXPERTA

- Normalmente **se realizan evaluaciones informales pidiendo opinión** a colegas o clientes
- Aunque una **técnica más efectiva** consiste en **consultar a evaluadores expertos**
 - internos o externos
- El **resultado** de una evaluación experta **puede ser:**
 - un informe formal con los problemas encontrados y recomendaciones
 - una presentación y discusión con los responsables o los diseñadores
- **Ventajas:**
 - Los resultados están disponibles más rápidamente
 - Más económicos que los estudios con usuarios
 - Los expertos pueden sugerir soluciones a los problemas encontrados
 - Ayudan a encontrar los errores obvios más fácilmente, para que no lleguen a las pruebas con usuarios
- **Desventajas:**
 - Los expertos no son usuarios reales, y sus predicciones sobre lo que harán los usuarios con la interfaz o sobre la importancia de un defecto pueden ser incorrectas
 - Los expertos tienen sus propias preferencias sobre interfaces, lo que puede sesgar la evaluación
 - La evaluación experta depende mucho de la experiencia del evaluador
- **El evaluador debería:**
 - Replicar las condiciones del usuario final
 - Tener tacto en sus recomendaciones
 - Comprender que es difícil en tan poco tiempo comprender todas las decisiones de diseño y la historia del desarrollo
 - Tener experiencia en el tipo de aplicación evaluada
 - Dejar el desarrollo de las soluciones a los diseñadores
 - Ser exhaustivo en el informe
 - Revisar la consistencia en todas las ventanas de la aplicación

TIPOS DE EVALUACIÓN

- **Evaluación heurística:** si la interfaz sigue la lista de heurísticas de diseño
- **Revisión de las guías de diseño:** revisa que se hayan seguido las indicaciones de las guías de diseño
- **Revisión de estándares:** comprueba que el diseño sigue un estándar
- **Inspección de consistencia:** comprueba la terminología, fuentes, paletas, formatos de E/S, etc.

INFORME

- **Con respecto al informe, se debería:**
 - Usar guías de diseño para estructurar el informe
 - Separar los problemas en función del tipo de usuario al que afecta
 - Ordenar las recomendaciones por importancia
 - Describir cada recomendación a nivel conceptual
 - Ser consciente de las restricciones técnicas o de negocio
 - Resolver el problema completo
 - Dar recomendaciones específicas y claras
 - Incluir pequeños detalles como faltas ortográficas, fallos de alineamiento de controles, inconsistencia en la posición de los controles, etc.

OTROS TIPOS DE EVALUACIÓN

- **Entrevistas de grupos** (focus groups)
 - Más o menos 8 personas dirigidas por un moderador discuten sobre algo
 - Útil cuando aún no existe la interfaz
- **Ordenación de tarjetas** (card sorting)
 - Se pide a los usuarios que agrupen elementos
 - Útil para identificar categorías y estructura
 - Estudiar los resultados, tanto:
 - donde hay acuerdo general
 - donde no hay acuerdo, y estudiar cómo resolverlo
- **Herramientas de validación automática**

PRUEBAS DE ACEPTACIÓN

- Son **pruebas que realiza un cliente sobre el sistema para comprobar que cumple los requisitos**
- Para **establecer requisitos de las interfaces de usuario**, hay que usar criterios medibles
 - **Tiempo necesario** para aprender a usar ciertas funciones
 - **Velocidad** de completado de una tarea
 - **Tasa de error** de los usuarios
 - **Retención** de las órdenes a lo largo del tiempo
 - **Satisfacción subjetiva** del usuario
- **Otros elementos medibles:** comprensión de la salida del sistema, tiempo de respuesta del sistema, procedimiento de instalación, documentación impresa, atractivo visual, etc.
- Se deberían **realizar por una parte neutral**
- Después de las pruebas de validación, aún se **deberán realizar pruebas de campo**



EVALUACIÓN DURANTE EL USO

- **Una vez que se ha publicado el sistema se debe hacer un seguimiento de este para mejorarlo**
- Hay que **minimizar los cambios** en la interfaz **para evitar molestar** a los usuarios
- **Herramientas:**
 - **Entrevistas y discusiones en grupo**
 - **Captura continua de datos de rendimiento** del usuario
 - **Consultas online o telefónicas**, buzones de sugerencias
 - **Grupos de discusión**, wikis y grupos de noticias
 - **Herramientas de evaluación automática**