# Resumen Detallado de Clases de Diseño de Interfaz, Experiencia de Usuario (UX) y UML

## 1. Introducción General

Durante las primeras clases se introdujeron los conceptos fundamentales sobre diseño de interfaz, experiencia de usuario y UML, que servirán de base para el desarrollo de ejercicios prácticos y del trabajo práctico grupal. El enfoque inicial estuvo centrado en: - **Diseño de interfaces de usuario.** - **User Experience (UX).** - **Lenguaje de modelado unificado (UML).** - **Uso de herramientas como StarUML.**

## 2. Diseño de Interfaz (según Pressman)

Basado principalmente en el libro de **Roger Pressman**, el diseño de interfaces constituye el vínculo entre el sistema y el ser humano. Tiene como objetivo facilitar la comunicación y la interacción, sentando las bases del prototipo de la solución.

### Principales ideas abordadas:

* La interfaz debe posibilitar la comunicación eficiente entre usuario y sistema.
* Permite identificar y dar forma a los requerimientos mediante interacción iterativa.
* Ayuda a obtener información clara del usuario final.

### Reglas Doradas del Diseño (Pressman):

1. **Ceder el control al usuario:** El usuario debe poder realizar acciones libremente, corregir errores, retroceder, y navegar sin restricciones.
2. **Reducir la carga de memoria del usuario:** Evitar que el usuario tenga que recordar pasos o estructuras para realizar tareas; la interfaz debe guiar.
3. **Ser consistente:** Uniformidad visual y funcional en botones, colores, estructuras y formas de navegación.

### Principios clásicos adicionales:

* Visibilidad de las acciones disponibles.
* Prevención de errores.
* Facilidad de aprendizaje.
* Flexibilidad y eficiencia.
* Uso de metáforas visuales conocidas.
* Seguimiento del estado de acciones.
* Simplicidad visual y comunicacional.

### Referencias adicionales:

* **Ley de Fitts:** fórmula matemática que permite medir la facilidad de interacción del usuario con elementos de la interfaz (por ejemplo, tamaño y distancia de botones).

### Pregunta típica de examen:

* *"Nombrar y explicar tres principios fundamentales del diseño de interfaz de usuario".*

## 3. Experiencia de Usuario (UX)

El diseño centrado en el usuario implica colocar al usuario en el centro del desarrollo. UX abarca desde la investigación inicial hasta el prototipado y validación.

### Etapas del proceso de UX:

1. **Investigación del usuario:**
   * Entrevistas y observación del usuario final.
   * Comprensión del perfil: edad, ocupación, nivel educativo y tecnológico.
   * Identificación de necesidades, frustraciones y dolencias ("pains").
2. **Prototipado de soluciones:**
   * Desarrollo de versiones preliminares de la interfaz basadas en la información relevada.
3. **Evaluación y pruebas de usabilidad:**
   * Pruebas controladas con usuarios reales.
   * Recolección de feedback.
   * Iteración y mejora del diseño.

### Modelo Conceptual CUE (Conceptual User Experience Model):

* **Capacidades:** Qué puede hacer el usuario dentro del sistema.
* **Usabilidad:** Nivel de facilidad, intuición y eficiencia en el uso.
* **Identidad emocional:** Qué tan identificado se siente el usuario.
* **Interacción:** Cómo navega y utiliza el sistema.

### Aclaraciones del docente:

* Importante diferenciar entre el usuario **pagador** (cliente) y el **usuario final** (quien realmente usará la herramienta).
* Incluir variables emocionales en el análisis (ej. frustración de vendedores, herramientas incómodas).

### Ejemplo mencionado:

* Vendedores de una farmacéutica usando un celular BlackBerry con teclas pequeñas. No era funcional por la ergonomía del usuario.

## 4. Trabajo Práctico: User Persona + Prototipos

### User Persona:

Representación semi-ficticia del usuario promedio del sistema.

**Debe incluir:** - Nombre. - Edad y ocupación. - Nivel de manejo tecnológico. - Objetivos laborales. - Frustraciones y necesidades reales.

### Herramientas sugeridas para prototipado:

* **Lucidchart:** Colaborativa, también permite diagramas UML.
* **Draftium:** Rápida y sencilla.
* **Figma:** Avanzada, profesional, colaborativa.
* **Balsamiq:** Baja fidelidad, exporta imágenes.
* **Excel:** Alternativa simple para representar flujo de pantallas.

### Notas importantes:

* El diseño es iterativo. No se espera que la primera versión sea definitiva.
* No se evalúa la estética, sino la funcionalidad y coherencia con el caso de uso.
* **La primera entrega es el 19 de abril.** No se admite postergación.

## 5. Fundamentos del Buen Diseño de Software

El diseño es parte esencial del desarrollo y debe pensarse desde el inicio. Se hace hincapié en que es **iterativo** e **incremental**.

### Características de un buen diseño:

* Cumple con los requerimientos funcionales.
* Es modular (dividido en componentes independientes).
* Legible, comprensible y mantenible.
* Preparado para cambios futuros.
* Representado con un lenguaje común: **UML**.

### Conceptos clave:

* **Separación de incumbencias:** Dividir grandes problemas en tareas más simples.
* **Ocultamiento de información:** Cada componente debe esconder sus detalles internos.
* **Agilidad al cambio:** El diseño debe permitir incorporar nuevas funcionalidades con el menor impacto posible.

### Importancia del mantenimiento:

* Aunque el desarrollo inicial dura meses, el mantenimiento puede durar años.
* Si el diseño es pobre, los cambios serán costosos y complejos.

### Discusión con el profesor:

* Se defendió el uso de "include" incluso si en el momento no se reutiliza, para pensar en extensibilidad futura.

## 6. UML (Lenguaje de Modelado Unificado)

### Características principales:

* Es un lenguaje estándar, utilizado globalmente.
* Facilita la comunicación en proyectos.
* Permite visualizar el sistema desde diferentes "vistas".

### Comparación:

* Como en arquitectura: una misma casa tiene planos estructurales, eléctricos, etc.

### Diagramas importantes:

* **Casos de uso:**
  + Relaciona actores con funcionalidades.
  + Da una visión completa de lo que el sistema debe hacer.

### Relaciones entre casos de uso:

* **Include:** La funcionalidad se reutiliza en múltiples casos.
* **Extend:** La funcionalidad ocurre bajo una condición especial.
* **Generalización:** Un actor o caso de uso hereda características de otro.

### Herramienta utilizada:

* **StarUML:** Se recomendó una versión antigua y estable para evitar licencias.

### Ejercicio práctico:

* Crear un diagrama de casos de uso con:
  + Flujo principal (ej. "Procesar venta").
  + Flujos alternativos (ej. "Pago con crédito", "Pago con débito").
  + Uso de relaciones como include y extend.

## Recomendaciones Finales del Curso

* No perder tiempo en la estética inicial de la interfaz.
* Priorizar la coherencia funcional.
* Trabajar activamente en grupo: el trabajo práctico es colaborativo.
* Estudiar los principios de diseño, UX y modelado UML.
* Ser conscientes de que el diseño es evolutivo: se mejora constantemente con el feedback del usuario.

**Recordatorio clave:** - El proceso de diseño es iterativo. - El primer entregable vence el **19 de abril**. - La funcionalidad es más importante que la apariencia. - Pensar el sistema con enfoque modular y extensible desde el inicio.