

Investigación

Operativa



Escuela de Educación Secundaria Técnica N°1
"Juan Bautista Alberdi"
Conesa - San Nicolás

6to año

Fecha de entrega 4/9

Prototipos

Es frecuente que los clientes no sepan lo que quieren, pero cuando ven algo y utilizan prototipos, pronto saben lo que no quieren.

Los prototipos son una representación limitada de un producto, permite a las partes probarlo en situaciones reales o explorar su uso, creando así un proceso de diseño de iteración que genera calidad.

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

¿Por qué un prototipo?

Porque son útiles para comunicar, discutir y definir ideas entre los diseñadores y las partes responsables.

Los prototipos apoyan la evaluación de productos, clarifican requisitos de usuario y definen alternativas.

Prototipos de baja fidelidad

Utilizan materiales distintos al del producto final, son baratos, simples y fáciles de producir.

Son particularmente útiles en las fases iniciales del desarrollo, durante el diseño conceptual.

Prototipo de alta fidelidad

Son aquellos que se parecen al producto final y utiliza sus mismos materiales.

Marc Retting (1994) desaconseja el uso de prototipos de alta fidelidad porque:

- Necesitan mucho tiempo para crearse.
- Las pruebas tienden a centrarse en aspectos superficiales.
- Los desarrolladores se resisten a cambiar algo que les ha llevado horas crear.
- Crea excesiva expectación.

Diseño de iteración

Refinamiento con retroalimentación cíclica

Diseño de iteración es simplemente pasar de una versión de diseño a otra mejor, y en cada paso, se realiza una evaluación de usabilidad (test o heurístico) para certificar la mejora. Cada iteración conlleva la elección de un pequeño conjunto de requisitos que permite rápidamente diseñar, implementar y probar.

Beneficios

Es un error separar el diseño de un producto de su proceso productivo. El proceso del diseño iterativo ya era definido en 1993 por Jakob Nielsen en *Iterative User Interface Design* como «el más sencillo y barato». Además:

- Reduce los riesgos de forma temprana.
- El progreso es visible desde las primeras etapas.
- Refina el sistema a las necesidades reales de los usuarios.
- Gestiona la complejidad: el equipo no se ve abrumado por la «parálisis del análisis».
- El conocimiento adquirido en un ciclo se puede utilizar metódicamente para mejorar el propio proceso de desarrollo.

Herramientas para hacer prototipos de software

1. **Pencil project**

2. **INVISION**

3. **Axure RP**

4. **Justinmind**

5. **BALSAMIQ MOCKUPS**

6. **MOCKINGBIRD**

7. **MOCKUP BUILDER**

8. **MOCKFLOW**

9. **HOTGLOO**