



Universidad  
Nacional  
de Quilmes

# CONSTRUCCIÓN DE INTERFACES DE USUARIO

1er Cuatrimestre de 2019



## Equipo Docente

- ▶ Leandro Di Lorenzo
- ▶ Juan Pablo Mottes
- ▶ Facundo Polo
- ▶ Álvaro Piorno

## Comunicación

- ▶ tpi-doc-uis@listas.unq.edu.ar
  - ▷ Sólo Docentes
- ▶ tpi-est-uis@listas.unq.edu.ar
  - ▷ Alumnos y Docentes



# ¿Qué vamos a ver en la materia?

- ▶ Qué son las Interfaces de Usuario
- ▶ Quién es el Usuario
- ▶ Cómo mejorar la experiencia del Usuario (UX)
- ▶ Diferencias entre vista y modelo
- ▶ Conceptos y patrones
  - ▷ Mockups
  - ▷ Eventos, Binding, Layouts, Validaciones
  - ▷ Excepciones y presentación de Errores
  - ▷ MVC, MVVM
  - ▷ Stateful / Stateless
  - ▷ Server Side / Client Side
  - ▷ Desktop / Web (y algo de Mobile)



# Un poco de stack tecnológico...

- ▶ Git / GitHub
- ▶ Kotlin (como Java pero chévere)
- ▶ Arena MVVM Framework
- ▶ IntelliJ
- ▶ XTRest
- ▶ Postman
- ▶ JSON
- ▶ HTML / CSS
- ▶ Bootstrap
- ▶ Javascript
- ▶ ReactJS
- ▶ React Native





# Dónde conseguir información...

- ▶ <http://interfaces.web.unq.edu.ar/>
- ▶ <https://github.com/unq-ui/material>
- ▶ <http://algo3.uqbar-project.org/> (cuatris anteriores)
- ▶ <http://wiki.uqbar.org/>
- ▶ En Google
- ▶ En Stack Overflow
- ▶ En Wikipedia (Inglés)
- ▶ En los libros



# Va a haber que codear mucho

- ▶ Guías Prácticas
- ▶ TP Desktop
- ▶ TP Web
- ▶ Taller Mobile
- ▶ Parcial



# ¿Cómo se aprueba la materia?

- ▶ Aprobando los 2 TPs y el Parcial
- ▶ Los TPs son en grupo (2 o 3), las notas... depende
- ▶ El Parcial es individual y se toma al final de la cursada
- ▶ El Taller Mobile (grupal) suma nota pero no resta
- ▶ Nota  $\geq 7$  en cada instancia  $\Rightarrow$  Promocionan
- ▶ Sino se recupera lo que se necesite
- ▶ Si luego de recuperar tiene alguna nota
  - ▷ entre 4 y 6.99  $\Rightarrow$  van a Integrador
  - ▷ menor 4  $\Rightarrow$  desaprueban la materia



# Cuestiones organizativas

- ▶ En general
  - ▷ 16 a 18 trabajo en el aula / repaso / consultas
  - ▷ 18 a 21 clase teórico/práctica
  - ▷ 21 a 22 consultas
- ▶ Todo aquel que pueda traer Notebook, hágalo
- ▶ Van a tener que codear mucho y aprender muchas tecnologías nuevas, no se dejen estar
- ▶ ¡Que no estemos en objetos no significa que no haya que usar patrones de diseño!
- ▶ Y también nos importa (mucho) el estilo del código

"Always code as if the guy who ends up maintaining your code will be a violent psychopath who knows where you live." - Martin Golding





# ¿Qué onda los trabajos, el taller, las prácticas y el parcial?

- ▶ TP1
  - ▷ Dominio » Kotlin
  - ▷ Interfaz Desktop » Arena + Kotlin
- ▶ TP2
  - ▷ Servicio REST » XTRest + Kotlin
  - ▷ Interfaz Web » Bootstrap + ReactJS
- ▶ Taller Mobile » React Native
- ▶ Parcial » Temas teóricos/prácticos de UI/UX



# Fechas Importantes

Si va todo bien:

- ▶ 28/03 Presentación TP1
- ▶ 25/04 Entrega TP1
- ▶ 02/05 Presentación TP2
- ▶ 06/06 Entrega TP2
- ▶ 13/06 Taller Mobile
- ▶ 27/06 Entrega (opcional) Taller Mobile
- ▶ 04/07 Parcial
- ▶ 18/07 Recuperatorio Parcial + TPs
- ▶ 25/07 Integrador



En la  
página  
de la  
materia

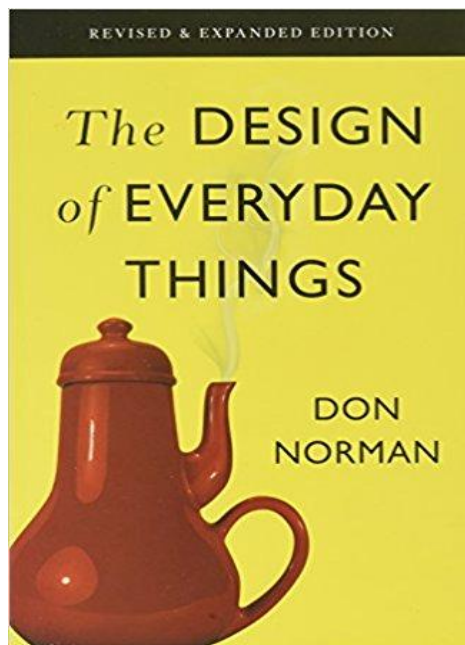
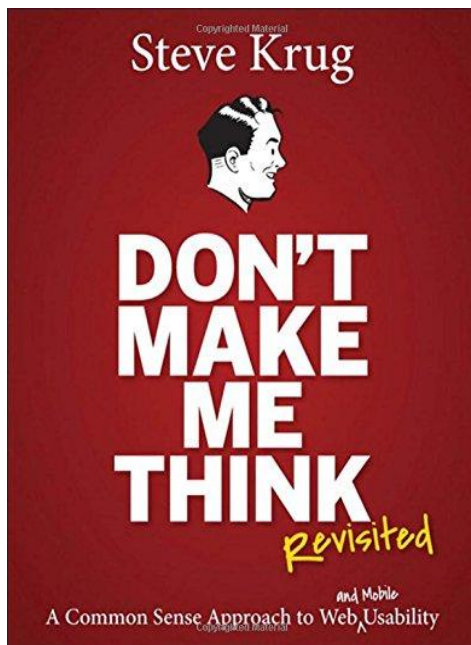


# ¿Por qué es importante esta materia?





# Bibliografía súper recomendada



# ¿Preguntas hasta acá?



# 1.

## INTRODUCCIÓN A LAS INTERFACES DE USUARIO

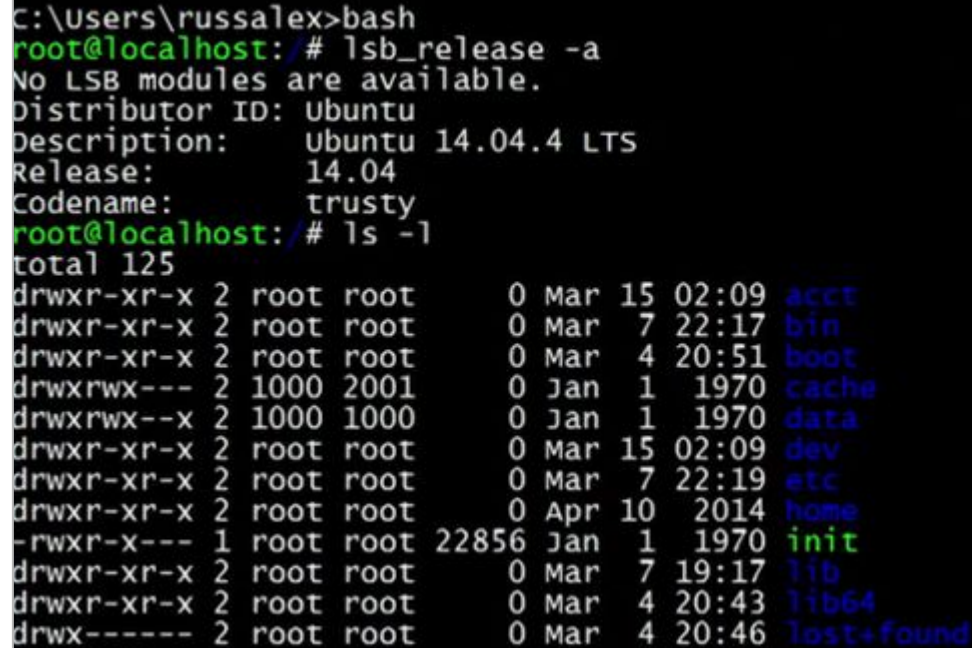


# ¿Qué es una Interfaz de Usuario?

“**La interfaz de usuario** (IU), en el campo del diseño industrial de la interacción humano-computadora, **es el espacio donde ocurren las interacciones entre humanos y máquinas.**”

*[https://en.wikipedia.org/wiki/User\\_interface](https://en.wikipedia.org/wiki/User_interface)*

# Ejemplos



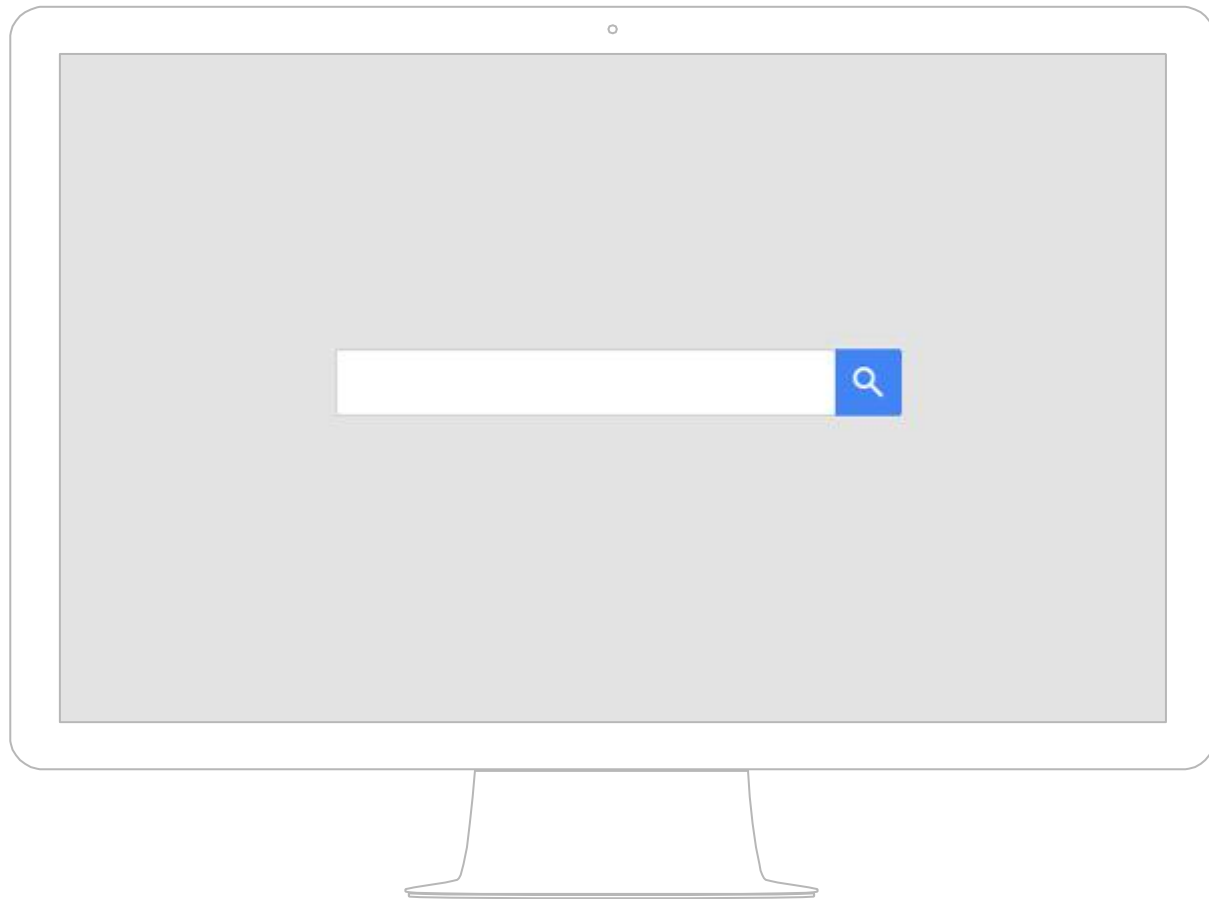
```
C:\Users\russalex>bash
root@localhost:~# lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 14.04.4 LTS
Release:        14.04
Codename:       trusty
root@localhost:~# ls -l
total 125
drwxr-xr-x  2 root root    0 Mar 15 02:09 acct
drwxr-xr-x  2 root root    0 Mar  7 22:17 bin
drwxr-xr-x  2 root root    0 Mar  4 20:51 boot
drwxrwx---  2 1000 2001    0 Jan  1  1970 cache
drwxrwx--x  2 1000 1000    0 Jan  1  1970 data
drwxr-xr-x  2 root root    0 Mar 15 02:09 dev
drwxr-xr-x  2 root root    0 Mar  7 22:19 etc
drwxr-xr-x  2 root root    0 Apr 10  2014 home
-rwxr-x---  1 root root 22856 Jan  1  1970 init
drwxr-xr-x  2 root root    0 Mar  7 19:17 lib
drwxr-xr-x  2 root root    0 Mar  4 20:43 lib64
drwx-----  2 root root    0 Mar  4 20:46 lost+found
```



# Ejemplos



# Ejemplos



# Ejemplos



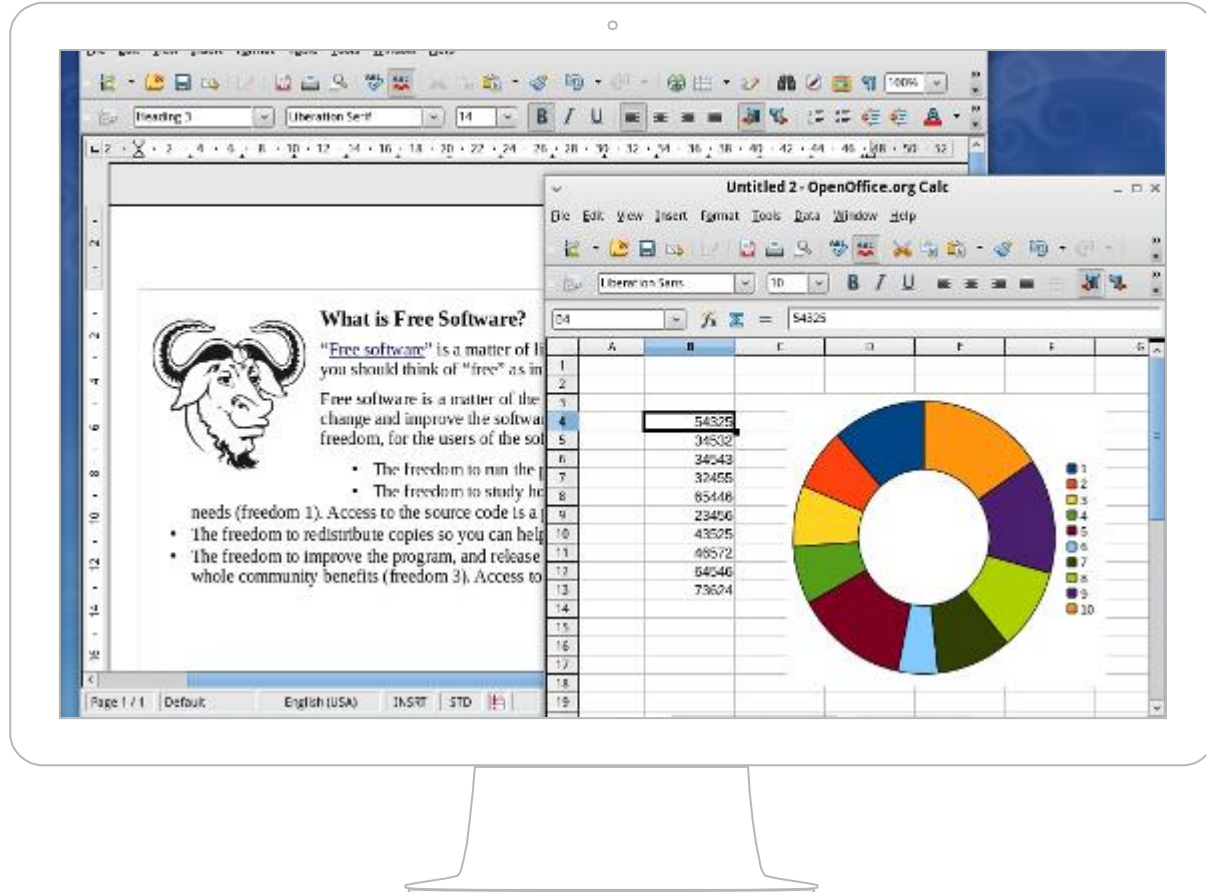
Why is the keypad arrangement different for a telephone and a calculator?



# Ejemplos



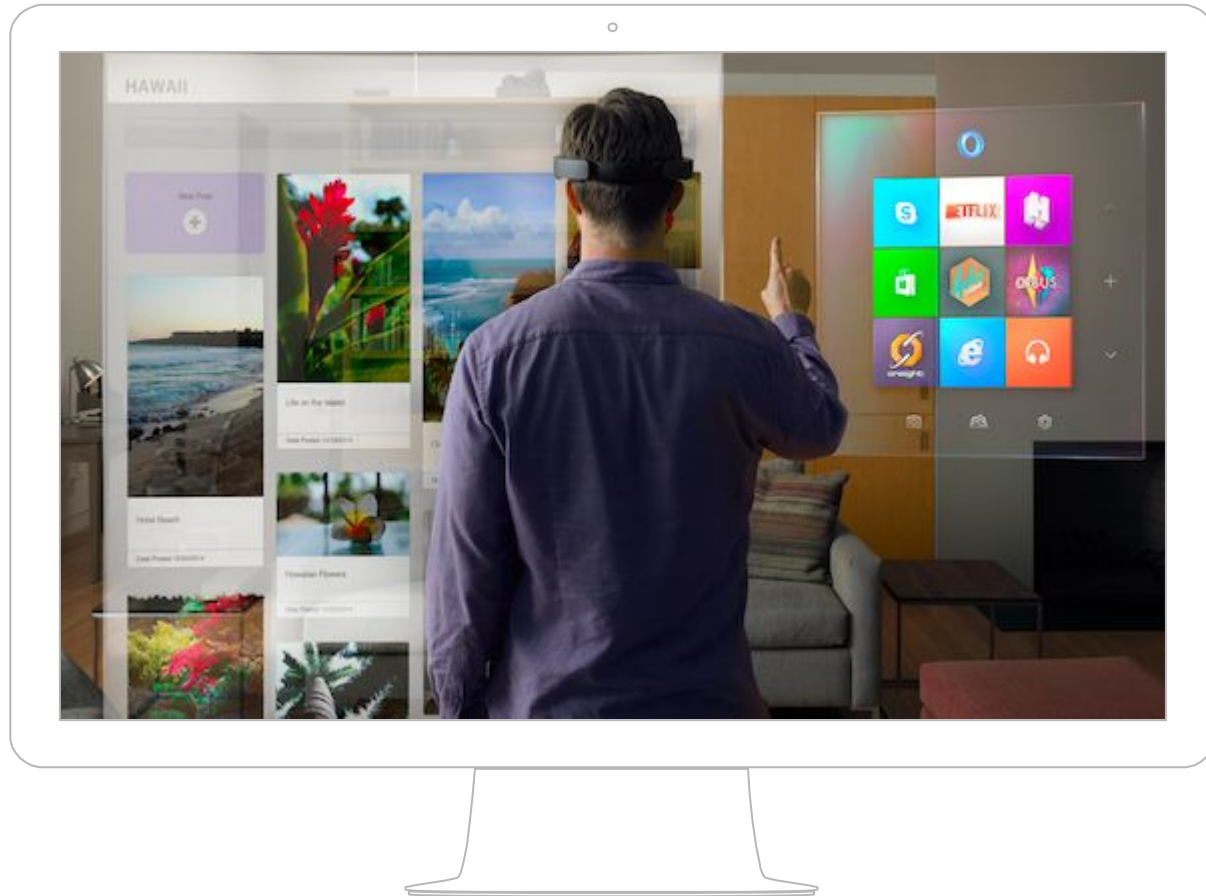
# Ejemplos



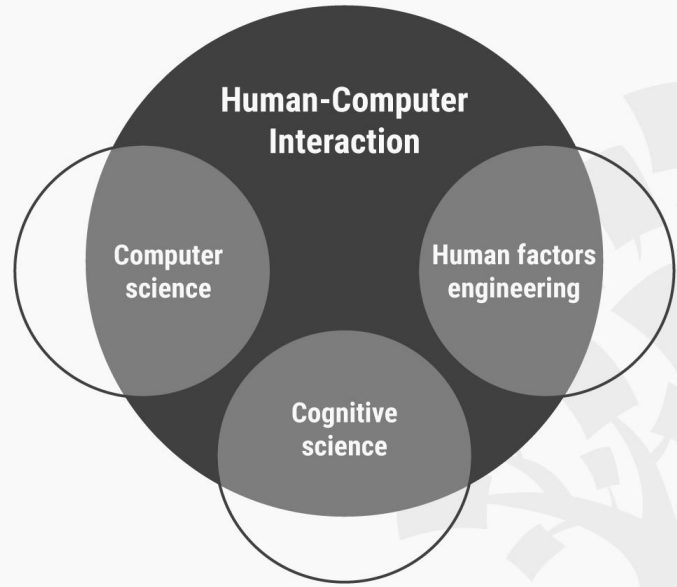
# Ejemplos



# Ejemplos



## The Multidisciplinary Field of HCI



## The Evolution of UX Design

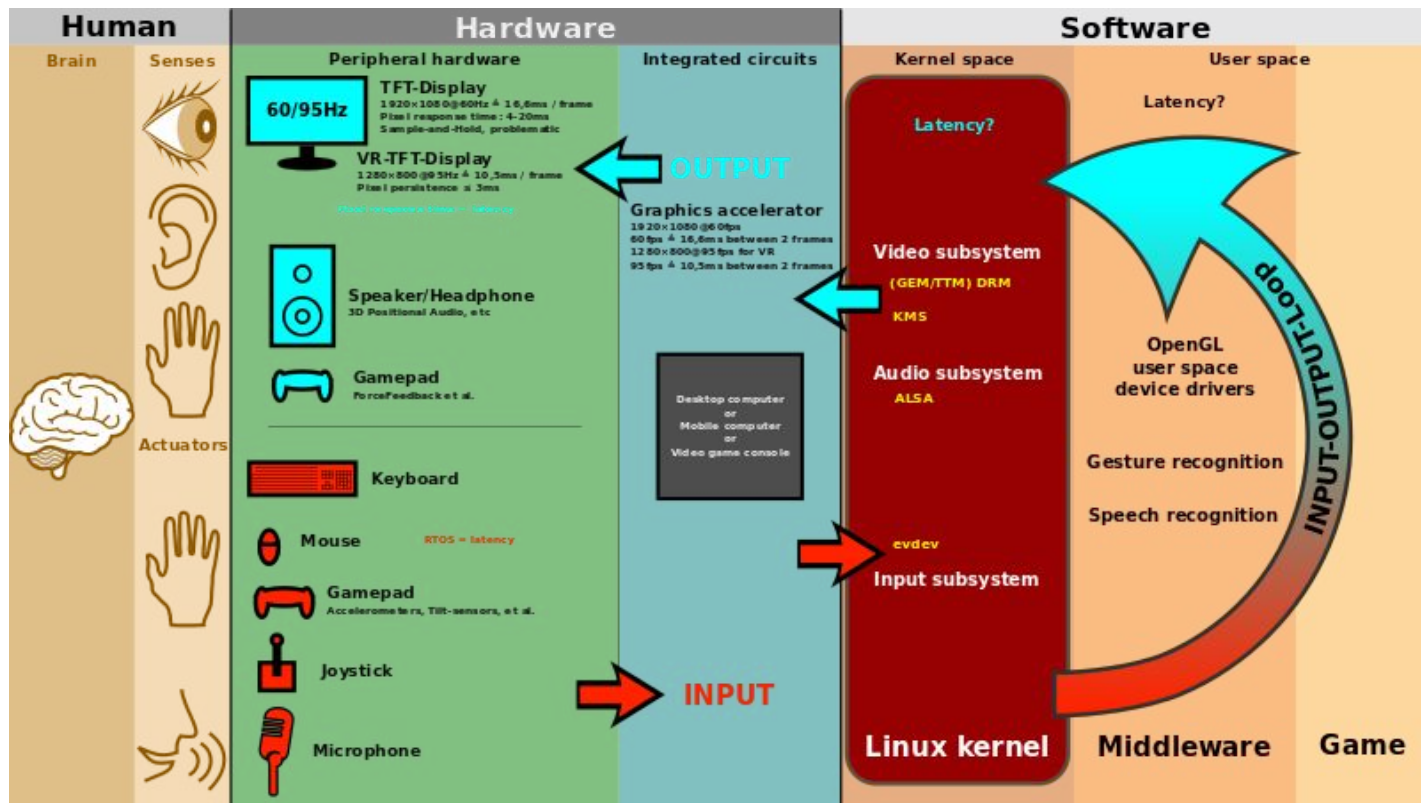


# Interfaces de Usuario

- ▶ Las Interfaces de Usuario comenzaron a estudiarse a finales de 1970
- ▶ Se investigaba cómo las personas interactuaban con las computadoras.
- ▶ Equipos multidisciplinares
  - ▷ Ciencias de la Computación
  - ▷ Ciencia Cognitiva
  - ▷ Ingenierías
- ▶ Parte evolucionó en lo que hoy conocemos como UX (User Experience)



# ¿Cómo se da la interacción en la Computadora?



# ¿Cuál es la diferencia entre Interfaz de Usuario (UI) y Experiencia de Usuario (UX)?

- ▶ La **Interfaz de Usuario** (UI) es una serie de pantallas, páginas y elementos visuales —como botones e íconos— que son utilizados para interactuar con un dispositivo
- ▶ La **Experiencia del Usuario** (UX), por otro lado, es la experiencia interna que una persona tiene mientras interactúa con todos los aspectos de un producto o servicio (alta subjetividad)

# Ejemplo de UI vs UX (I)

UI



UX



# Ejemplo de UI vs UX (II)



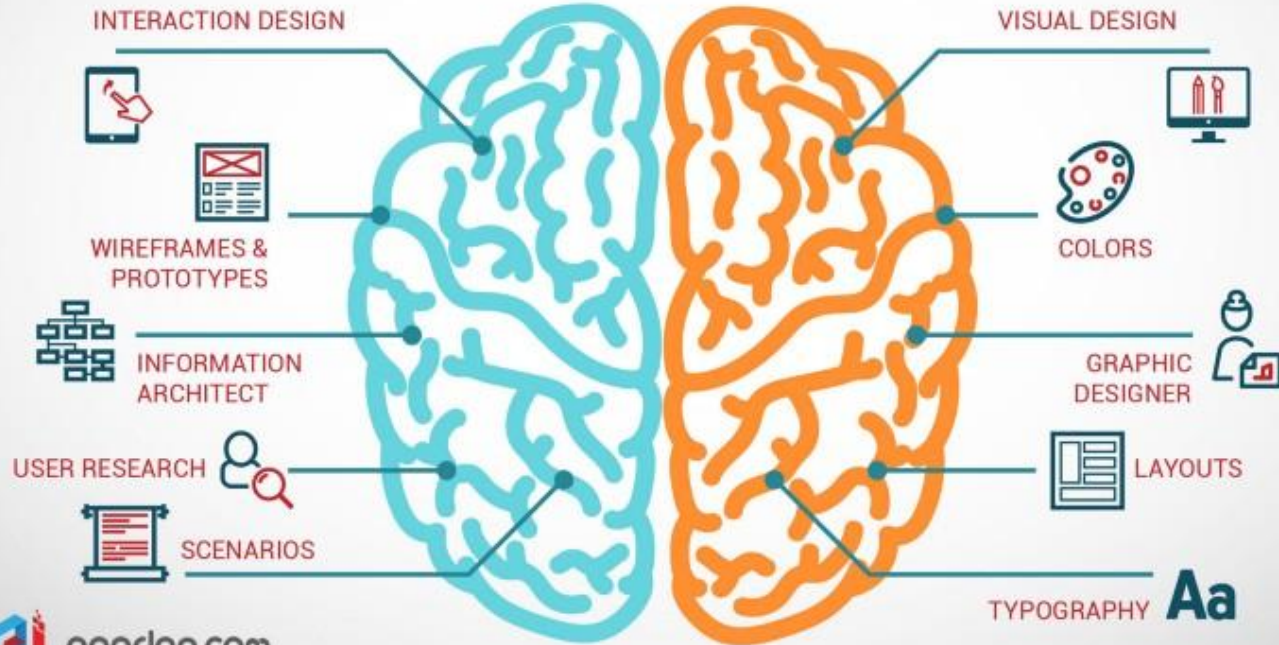
KNOWING THE DIFFERENCE BETWEEN

UX

&

UI

DESIGN



# Principios Fundamentales de la Interacción

Cuando interactuamos con un producto, debemos descubrir cómo hacerlo. Esto significa descubrir qué hace, cómo funciona y qué operaciones son posibles: “descubribilidad” (discoverability). La “descubribilidad” es el resultado de la aplicación apropiada de seis conceptos psicológicos:

- ▶ Affordances (Ofrecimientos)
- ▶ Signifiers (Significantes)
- ▶ Constraints (Restricciones)
- ▶ Mappings (Mapeos)
- ▶ Feedback (Retroalimentación)
- ▶ Conceptual Model (Modelo Conceptual)



# Affordances (Ofrecimientos)



- ▶ Relación entre las propiedades del objeto y las capacidades del usuario de comprender cómo es posible usarlo.
- ▶ No es una propiedad, es una relación.
- ▶ Depende tanto de las propiedades del objeto como de la comprensión del usuario.

# Signifiers (Significantes)



- ▶ Los ofrecimientos determinan qué acciones son posibles.
- ▶ Los significantes, dónde esas acciones se llevan a cabo.
- ▶ Los usuarios suelen buscar pistas que brinden sentido.
- ▶ Los diseñadores deben ser capaces de proveerlas.
- ▶ Lo que los usuarios necesitan y los diseñadores proveen, esos son significantes



# Constraints (Restricciones)



- ▶ Con el uso apropiado de las restricciones se limita a un número de acciones posibles (o al menos deseadas).
- ▶ Las restricciones deben ser
  - ▷ Útiles
  - ▷ Fáciles de ver e interpretar
  - ▷ Permitir hacer algo que antes no se podía
- ▶ También previenen de errores o acciones no deseadas



# Mappings (Mapeos)

- ▶ Los mejores mapeos son aquellos en donde la relación entre los controles y el objeto a ser controlado se vuelve obvia.
- ▶ Idealmente los mapeos deberían ser claros y sin ambigüedades

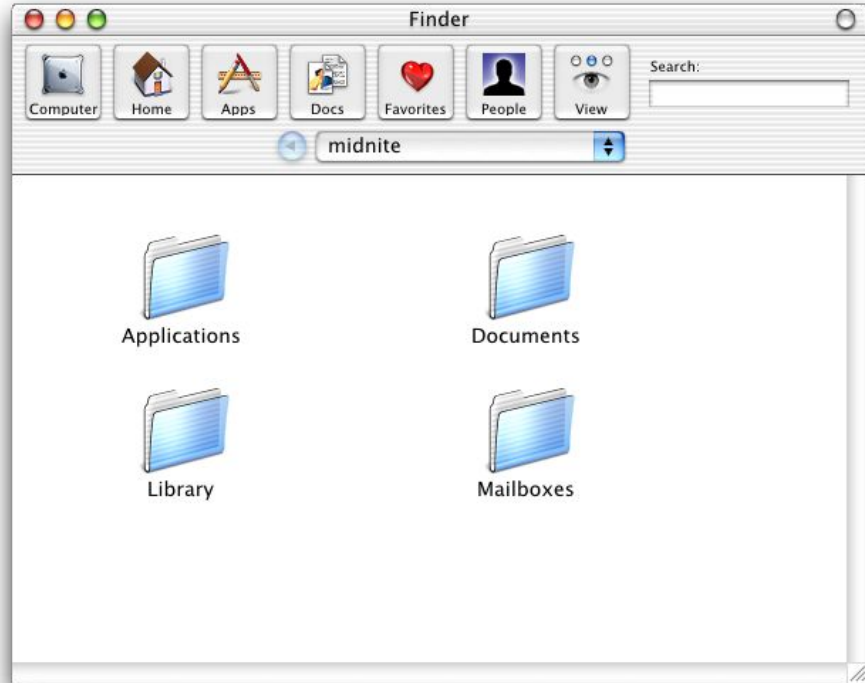


# Feedback (Retroalimentación)



- ▶ El feedback es la comunicación del resultado de una acción.
- ▶ Debe ser inmediato.
- ▶ Aún un pequeño delay puede ser desconcertante.

# Conceptual Model (Modelo Conceptual)



- ▶ Un modelo conceptual es una representación (usualmente muy simplificada) del objeto que buscamos explicar.
- ▶ Los documentos, las carpetas, los íconos de un SO ayudan a crear un modelo conceptual de cómo se organizan los archivos y directorios dentro de la computadora.

# ¿Bueno pero.... Qué hago con todo esto?

Por el momento:

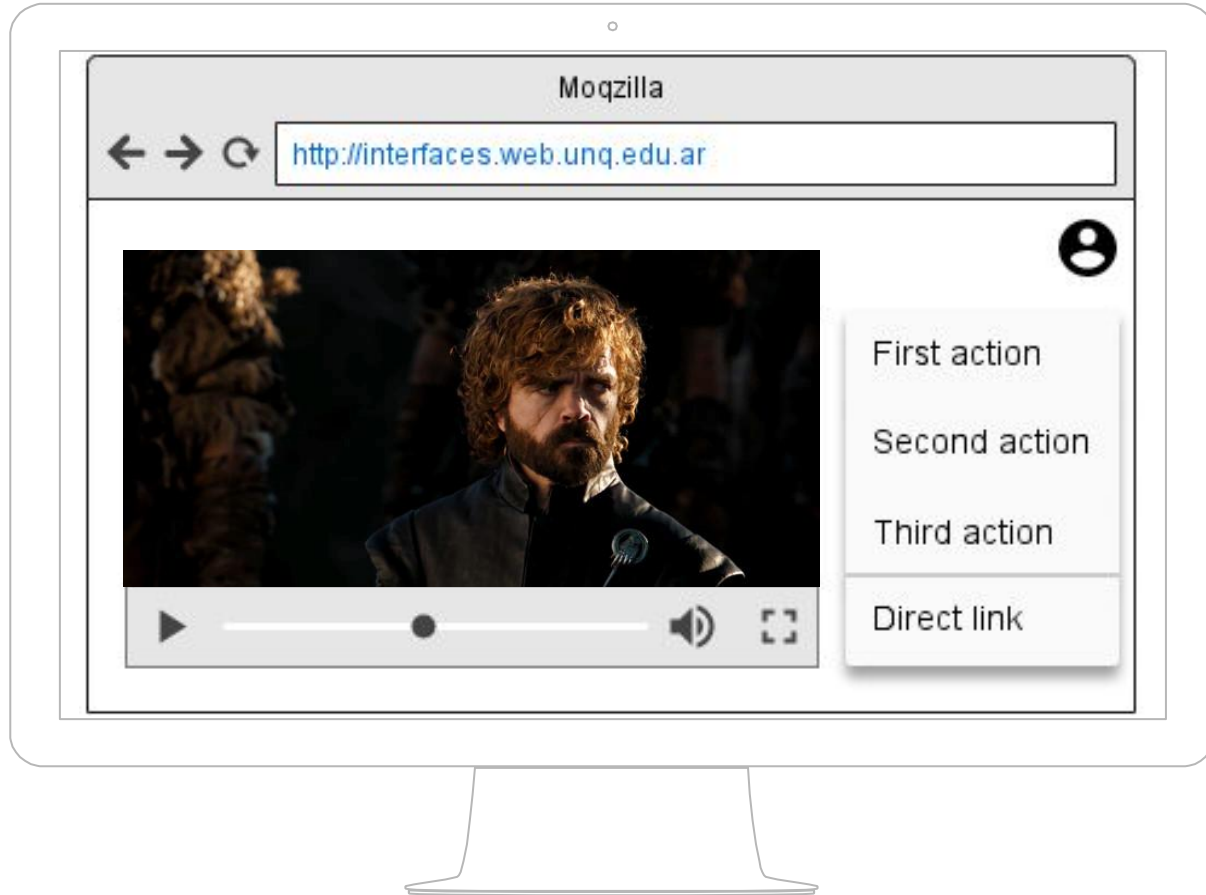
- ▶ Observar interfaces (todas las que detecten)
- ▶ Probarlas de distintas maneras
- ▶ Tratar de descubrirles “nuevas” funcionalidades
- ▶ Pensar qué mejoras se le podrían hacer
- ▶ Cuestionarlas
- ▶ Ver cuáles de los conceptos de discoverability cumplen y por qué no cumplen los otros
- ▶ Y sobre todo... “Sentir” la experiencia de usarlas

# Y para que puedan jugar... Mockups

Un **Mockup** es un diseño a escala, fundamentalmente sin interacción, que se utiliza para:

- ▶ Enseñar
- ▶ Hacer una demostración
- ▶ Evaluar una idea de diseño
- ▶ Promocionar un producto
- ▶ Etc...

# Ejemplo de Mockup



# Ejercicio

Ir al sitio <https://moqups.com/> y diseñar 2 mockups que representen

1. Un login de usuario en un sitio web
2. Una interfaz de un sitio web con contenido libre

Cuando terminen, mandenlos por mail a [leandro.dilorenzo@unq.edu.ar](mailto:leandro.dilorenzo@unq.edu.ar) (no tenemos lista docente todavía) para discutirlos entre todos (de forma anónima).

¿Preguntas?

