



Universidad  
Nacional  
de Quilmes

# CONSTRUCCIÓN DE INTERFACES DE USUARIO

2do Cuatrimestre de 2019





## Equipo Docente

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



## Comunicación

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



# Qué veremos en la materia

- ▶ Qué son las Interfaces de Usuario
- ▶ Quién es el Usuario
- ▶ Qué es la Experiencia de Usuario (UX)
- ▶ Cómo mejorar la experiencia del usuario
- ▶ Distintos tipos de Aplicaciones:
  - ▷ Desktop
  - ▷ Web
  - ▷ Mobile



# Conceptos asociados a las UIs

- ▶ Mockups
- ▶ Dominio, Modelo y Vista de una App
- ▶ Patrones MVC/MVVM
- ▶ Binding
- ▶ Componentes
- ▶ Manejo de Eventos
- ▶ Validaciones y Presentación de Errores
- ▶ Manejo de Excepciones
- ▶ Aplicaciones Stateful/Stateless
- ▶ Server Side / Client Side



## Un poco de stack tecnológico...

- ▶ Git / GitHub
- ▶ Kotlin (como Java pero chévere)
- ▶ Arena MVVM Framework
- ▶ IntelliJ / Eclipse (IDEs)
- ▶ Javalin REST Framework
- ▶ 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://wiki.uqbar.org>
- ▶ En la documentación de cada tecnología
- ▶ En Google
- ▶ En Stack Overflow
- ▶ En Wikipedia (Inglés)



# Instancias de Evaluación

- ▶ 3 TPs (Grupos de 4)
  - ▷ TP1 : App Desktop
  - ▷ TP2 : API REST
  - ▷ TP3 : App Web
  - ▷ Taller Mobile (opcional)
- ▶ Parcial Teórico/Práctico (Individual)



# ¿Cómo se aprueba la materia?

- ▶ Promocionando los 3 TPs y el Parcial
- ▶ El Taller Mobile es opcional
  - ▷ Suma nota solo si se aprueba
- ▶ Todas las instancias  $\geq 7 \Rightarrow$  Promocionan
- ▶ Cada instancia tiene recuperatorio
- ▶ Si luego de recuperar tienen alguna nota
  - ▷ entre 4 y 6.99  $\Rightarrow$  van a Integrador
  - ▷ menor 4  $\Rightarrow$  desaprueban la materia





# Cuestiones organizativas

- ▶ Clases Teóricas
  - ▷ Martes de 18:30 a 21:30
- ▶ Clases Prácticas
  - ▷ C1 : Jueves de 16:00 a 19:00
  - ▷ C2 : Jueves de 19:00 a 22:00
- ▶ Grupos de 4 personas
- ▶ Las entregas de TPs son presenciales



## Sugerencias

- ▶ Si pueden traigan su Notebook a las prácticas para trabajar más cómodos
- ▶ Van a tener que *codear* mucho y van a 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



## ¿Qué onda los TPs?

- ▶ Dominio » Lo proveemos nosotros
- ▶ TP1 : Interfaz Desktop » Kotlin con Arena
- ▶ TP2 : API REST » Kotlin con Javalin
- ▶ TP3 : Interfaz Web » HTML/CSS/JS con ReactJS
- ▶ Taller Mobile » React Native



# Fechas Importantes

Si va todo bien:

- ▶ 29/08 Presentación TP1
- ▶ 19/09 Entrega TP1
- ▶ 26/09 Presentación TP2
- ▶ 10/10 Entrega TP2
- ▶ 17/10 Presentación TP3
- ▶ 14/11 Entrega TP3
- ▶ 26/11 Parcial

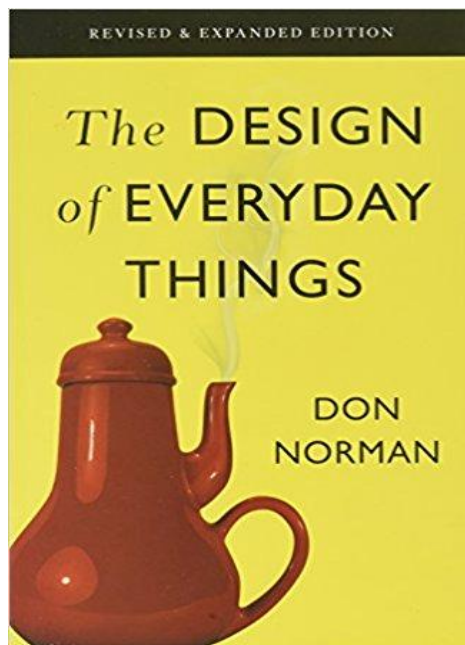
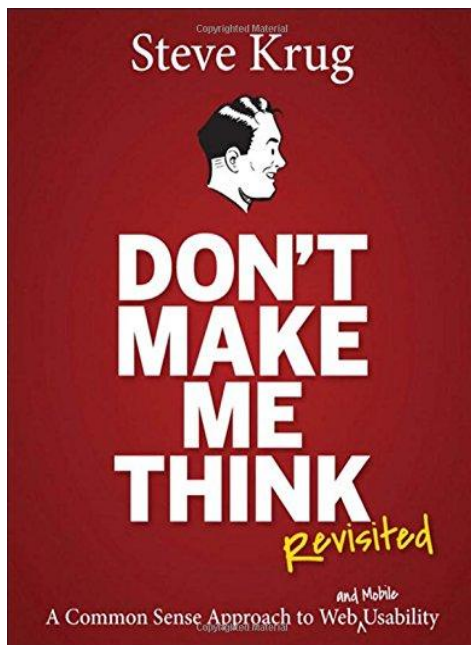


# Por qué es importante UI





## 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)*

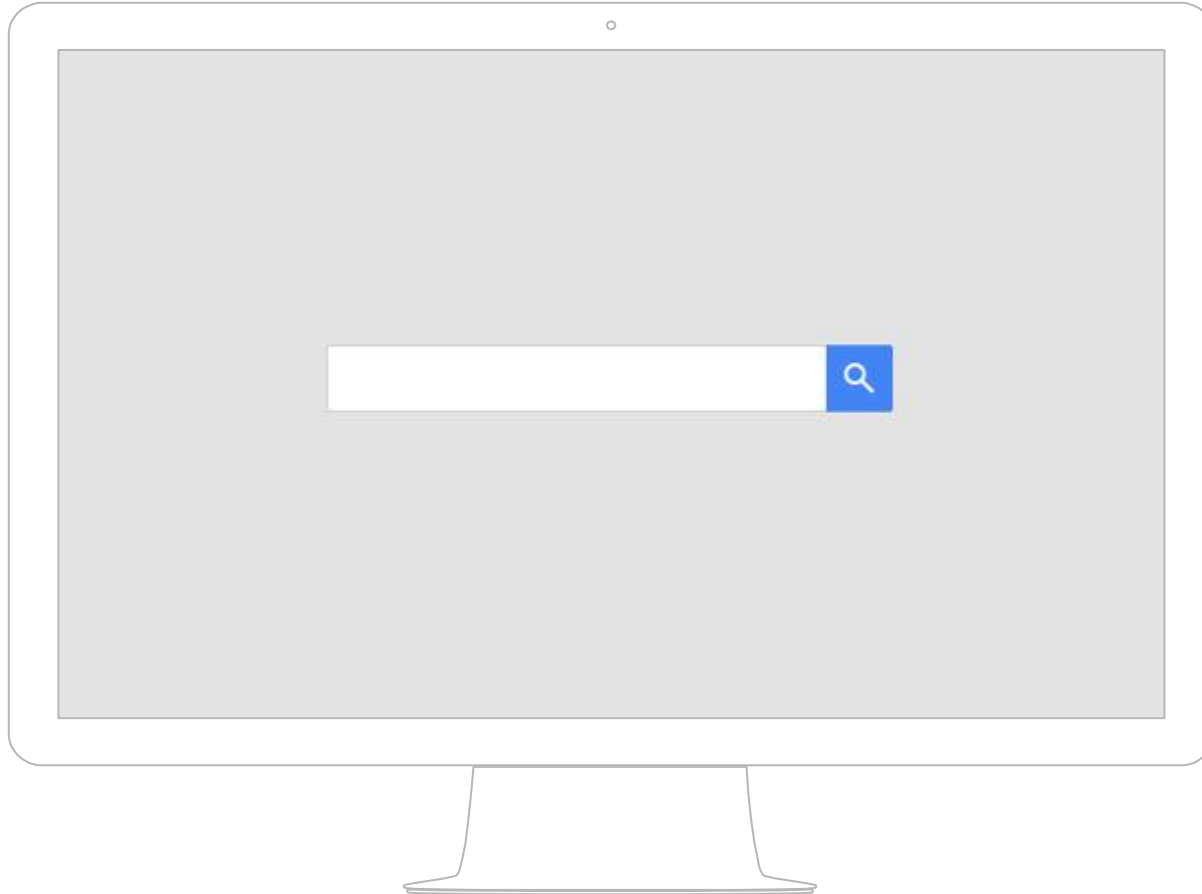
# Ejemplos

```
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
```

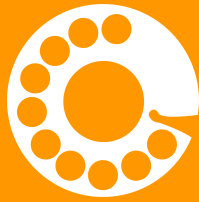
# Ejemplos



# Ejemplos



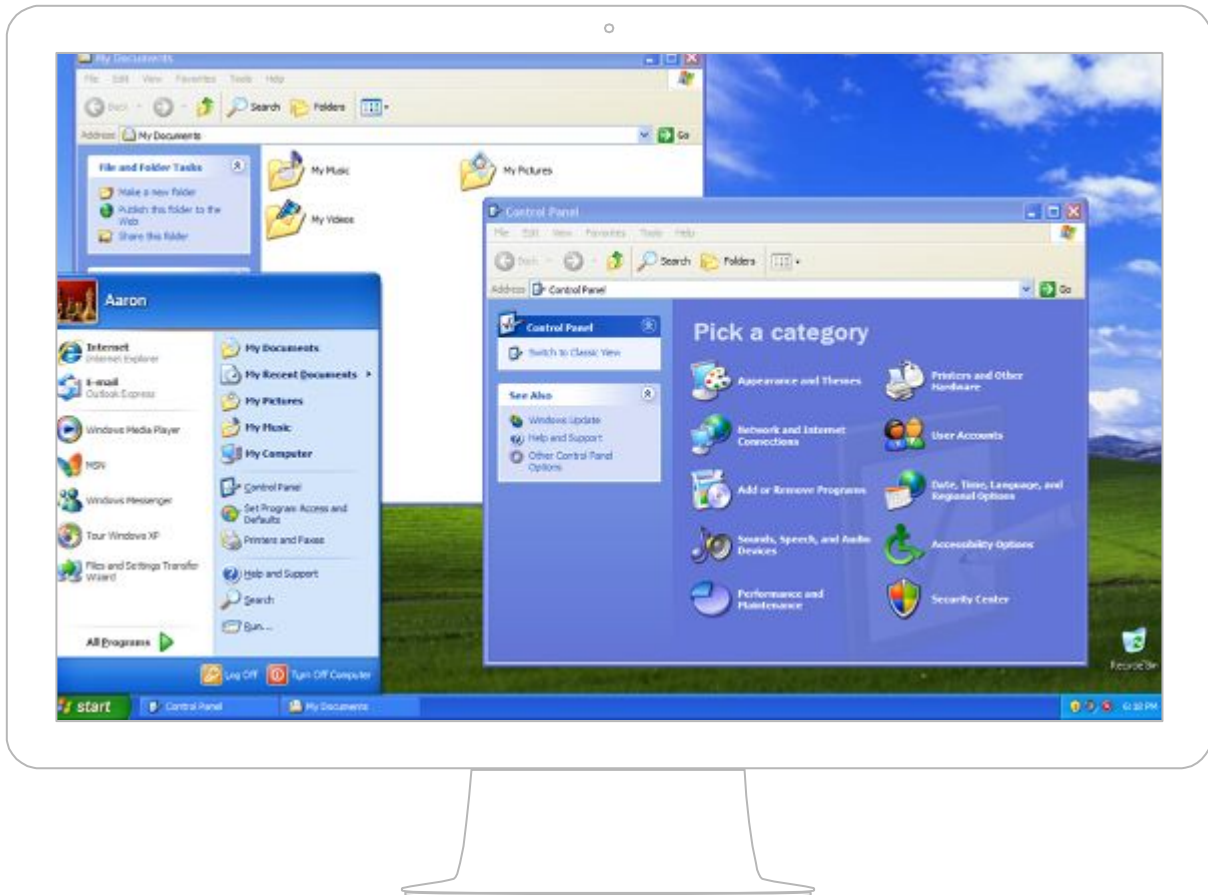
# Ejemplos



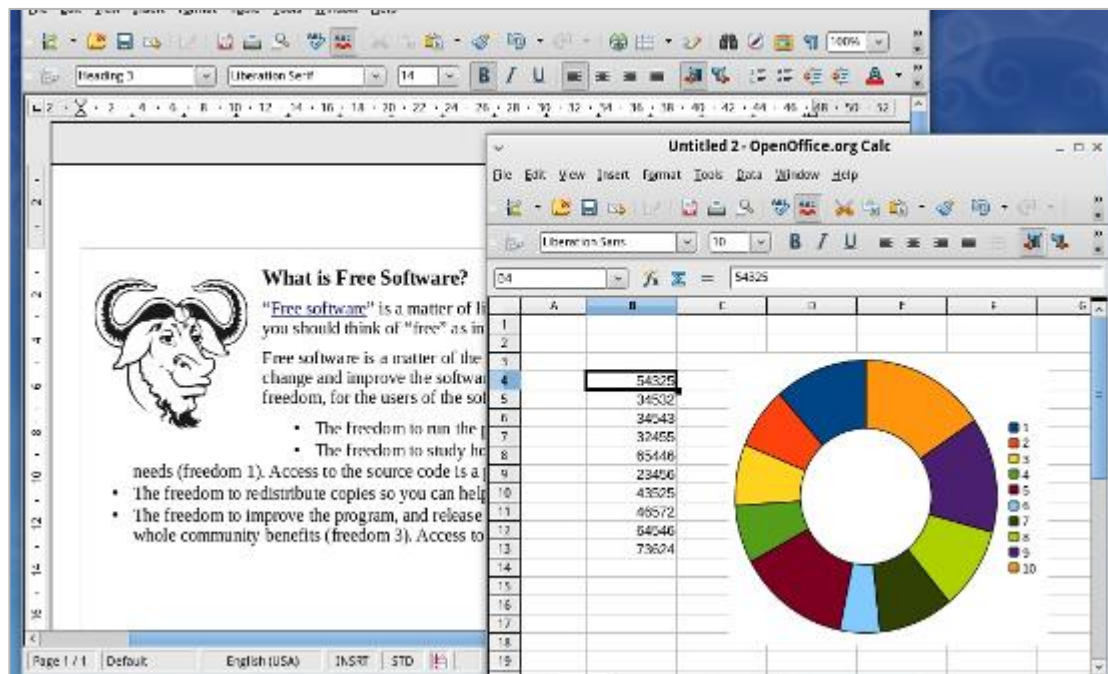
Why is the keypad arrangement different for a telephone and a calculator?



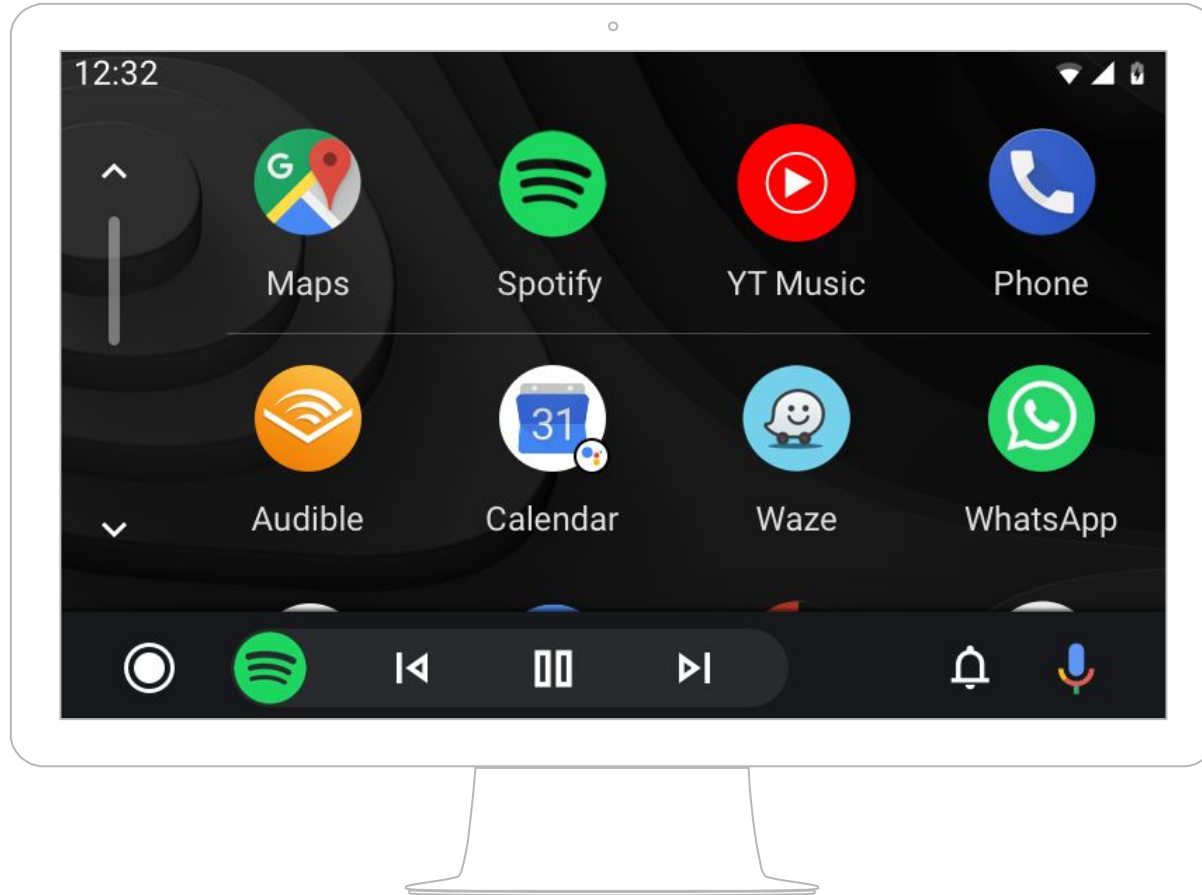
# Ejemplos



# Ejemplos

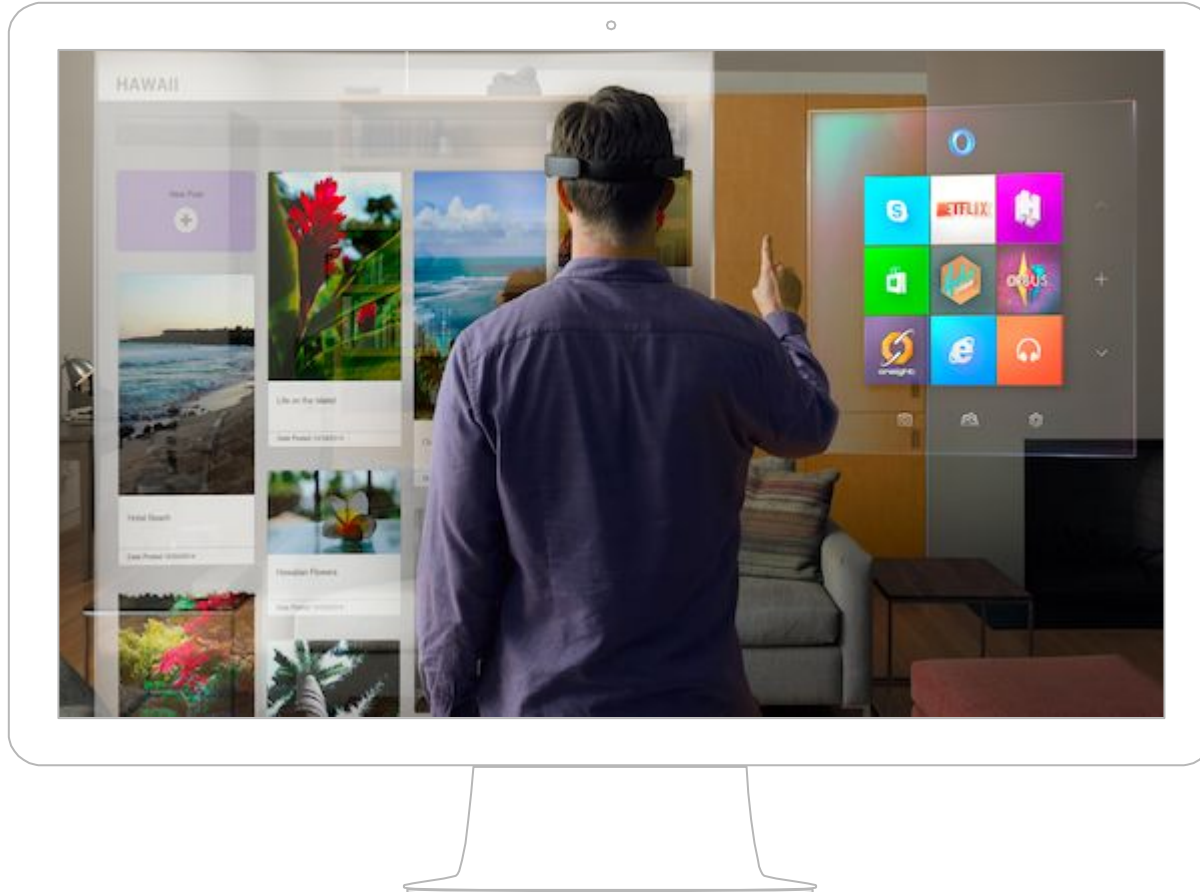


# Ejemplos

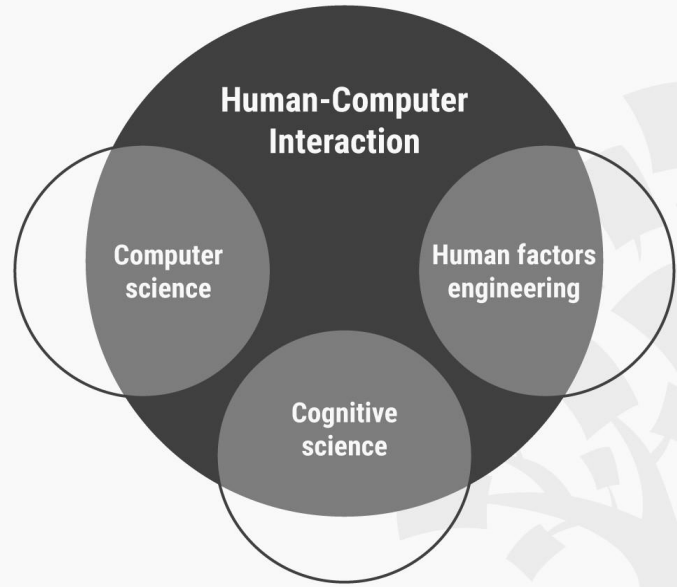




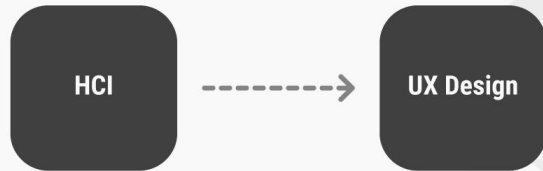
# Ejemplos



## The Multidisciplinary Field of HCI



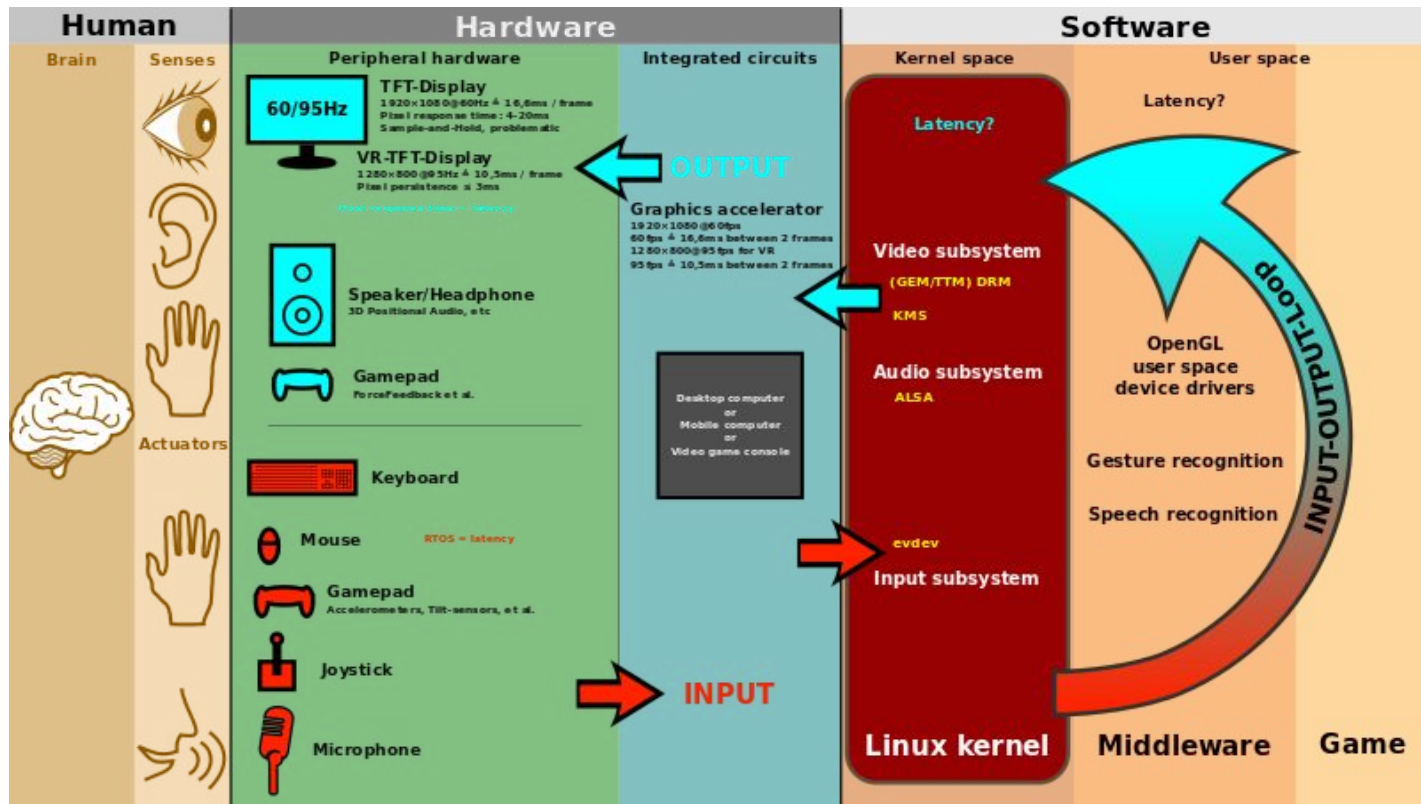
## The Evolution of UX Design



# Interfaces de Usuario

- ▶ Se comenzó a estudiar las UI a finales de 1970
- ▶ La idea era investigar cómo las personas interactuaban con las computadoras
- ▶ Equipos multidisciplinarios
  - ▷ Ciencias de la Computación
  - ▷ Ciencia Cognitiva
  - ▷ Ingenierías
- ▶ Parte evolucionó en lo que hoy conocemos como UX (User Experience)

# Cómo funciona la interacción con 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)





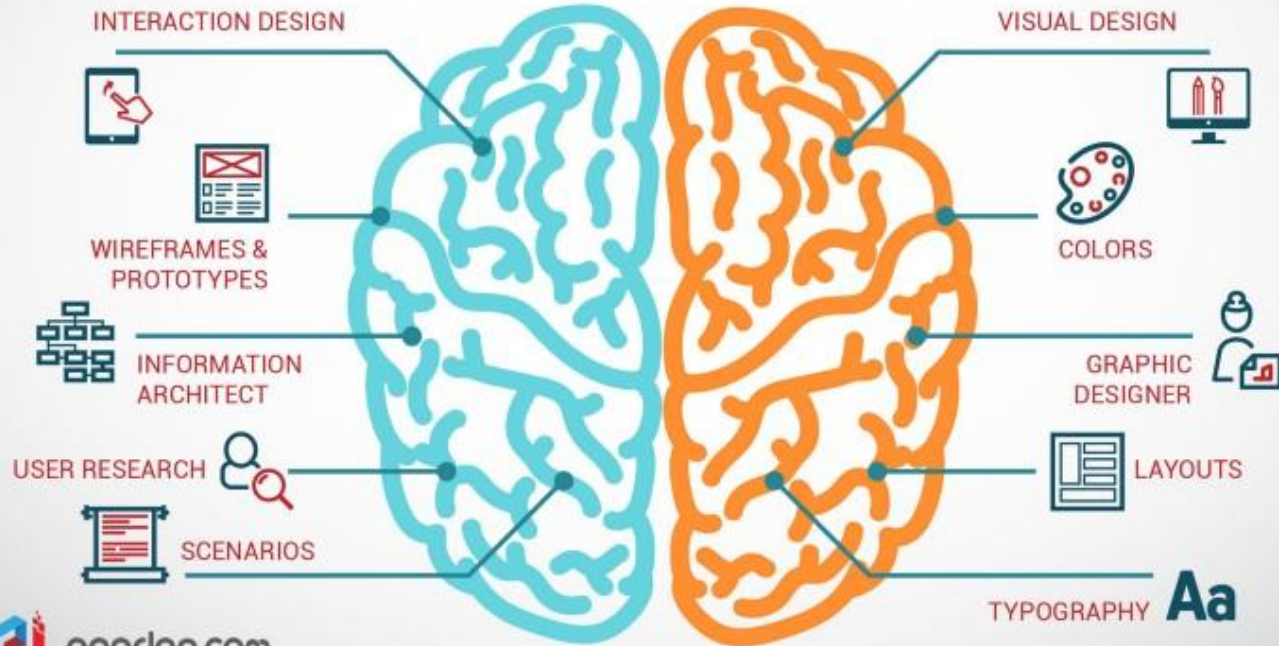
## KNOWING THE DIFFERENCE BETWEEN

# 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

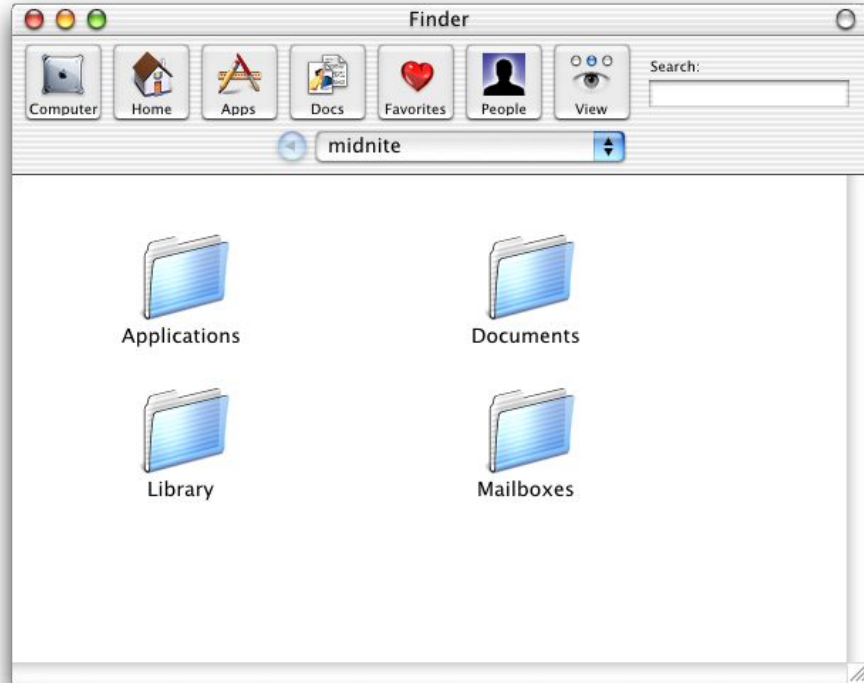


# Feedback (Retroalimentación)



- ▶ 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
- ▶ Entender quién es el usuario objetivo
- ▶ 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...

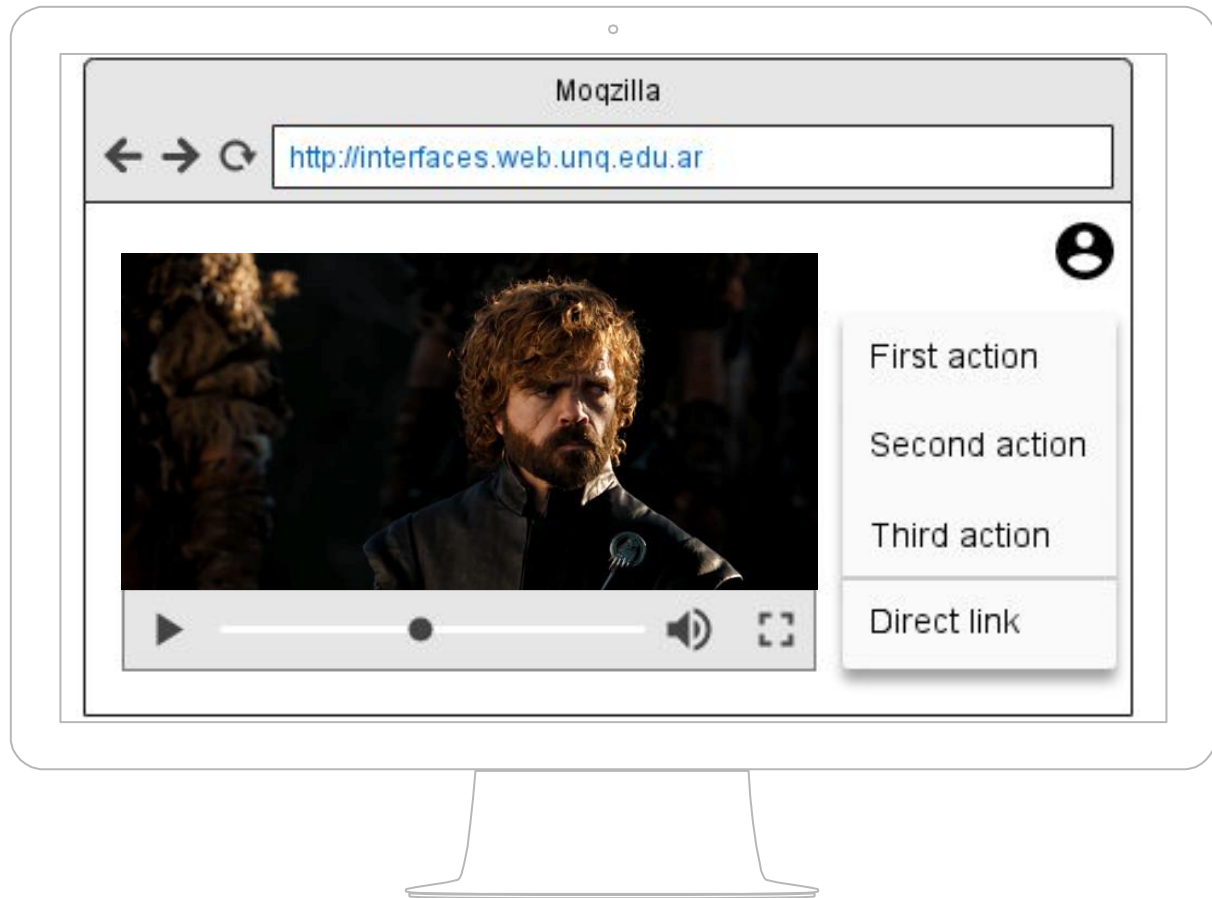
## Mockups

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 de Tarea

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

Suban las capturas a <https://bit.ly/33IiZ8s> en

- ▶ C1 - 16 a 19
- ▶ C2 - 19 a 22

Los vamos a discutir de forma anónima en la Práctica.

# ¿Preguntas?

