



Herramienta gráfica para el diseño de componentes simples y estados estáticos de una interfaz gráfica de usuario.



CONTEXTUALIZACIÓN





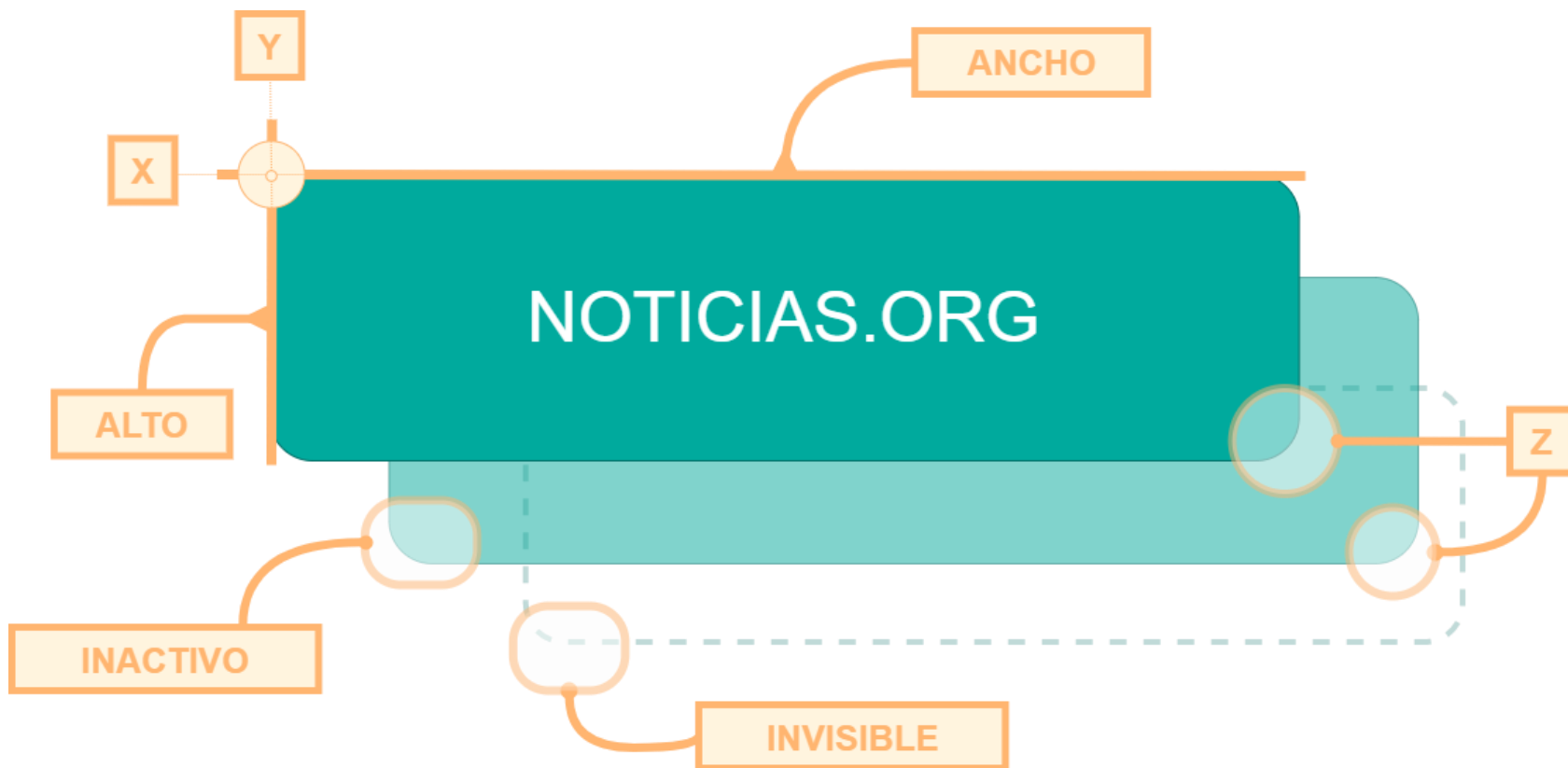
CONTEXTUALIZACIÓN

**COMPONENTE
SIMPLE**

Nombre
Fichero (Ruta)
Activo
Visible
Alto
Ancho
Posición Eje X
Posición Eje Y
Posición Eje Z



CONTEXTUALIZACIÓN





CONTEXTUALIZACIÓN

ESTADO

Número

*Componente
Simple*

*,
Componente
Simple*

*,
Componente
Simple*

...



CONTEXTUALIZACIÓN

NOTICIAS.ORG

RECIENTES

ANTIGUAS

OTRAS



1

NOTICIAS.ORG

RECIENTES

ANTIGUAS

OTRAS



2

NOTICIAS.ORG

RECIENTES

ANTIGUAS

OTRAS



3



PLANIFICACIÓN

Framework GUI:

Electron, Gtk, Qt

LANG:

Js, Python, C++

CREAR y VALIDAR

XML

CAPTURAR y RESPONDER

EVENTO DE USUARIO



PLANIFICACIÓN

Framework GUI:
Electron, Gtk, Qt



Qt

LANG:
Js, Python, C++



Python

CREAR y VALIDAR
XML



Lxml

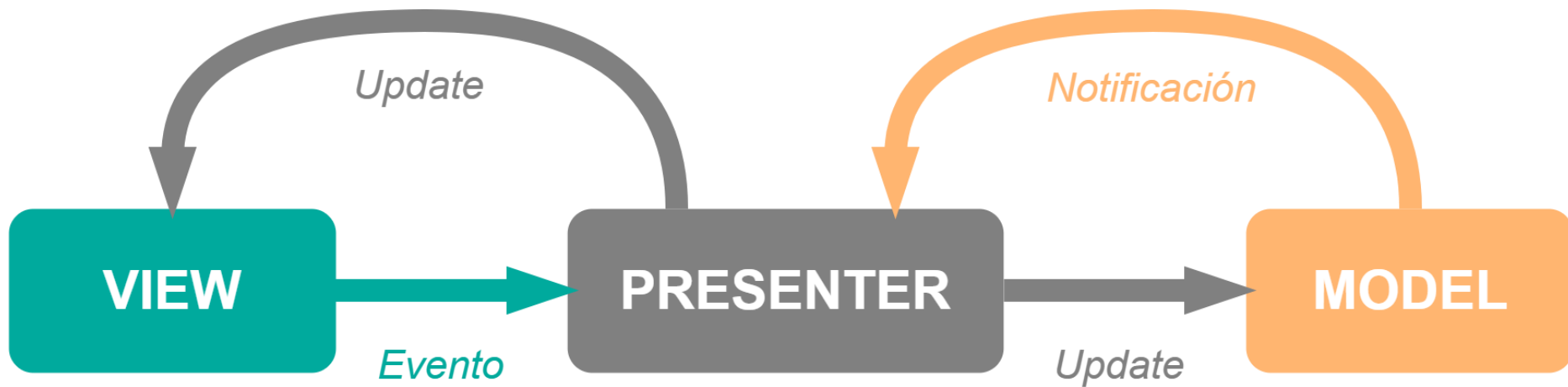
CAPTURAR y RESPONDER
EVENTO DE USUARIO



Qt



PLANIFICACIÓN



MVP *Vista Pasiva*



PLANIFICACIÓN

Investigación	7 días
Framework para interfaces gráficas	2 días
Componentes necesario del framework	2 días
Librería para procesamiento XML	1 día
Librería para soporte multi-idioma	1 día
Arquitecturas habituales en aplicaciones de escritorio	1 día
Interfaz	13 días
Ventana principal y diseño de la clase	2 días
Barra de menús, herramientas y estado	2 días
Area de trabajo	3 días
Listas de información visual	2 días
Menus contextuales	1 día
Acciones de usuario	2 días
Alertas	1 día

Componente Simple	9 días
Diseño de la clase	4 días
Añadir y eliminar	1 día
Profundidad + / -	1 día
Visible y Activo	1 día
Escala + / -	1 día
Desplazar	1 día
Estado	6 días
Diseño de la clase	3 días
Clonar	1 día
Eliminar	1 día
(Re)Ordenar	1 día
Persistencia	9 días
Diseño de la clase	2 días
Histórico de acciones	3 días
Servicio de Logs	1 día
Conversion a XML	1 día
Conversion desde XML	1 día
Validacion del DTD	1 día
Experiencia de usuario	6 días
Soporte multi-idioma	2 días
Atajos habituales	1 día
Zoom al cursor	1 día
Ocultar interfaz	1 día
Mensajes en barra de estado	1 día

PLANIFICACIÓN

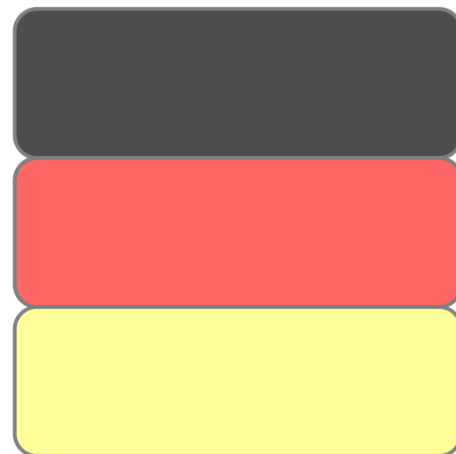
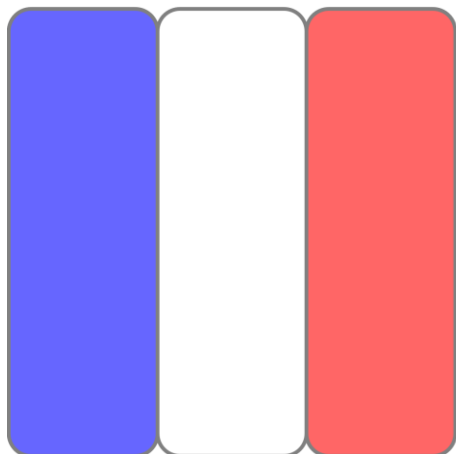
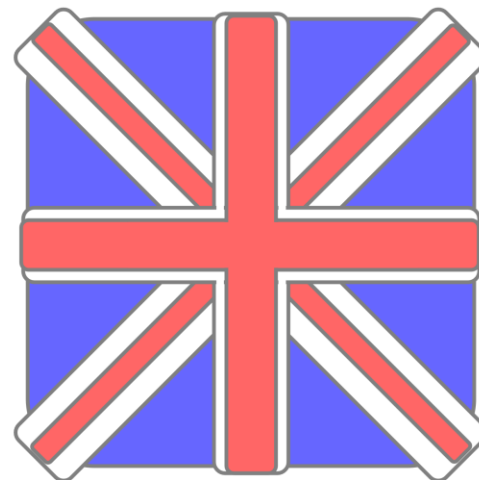
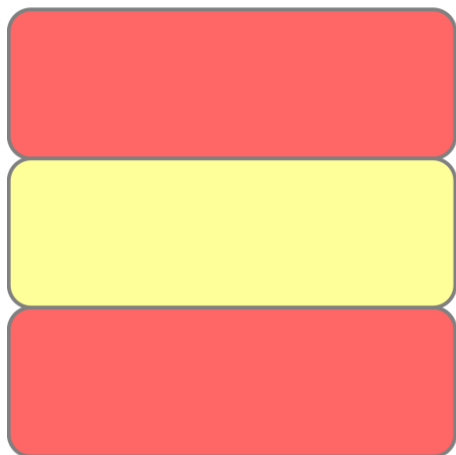
Investigación	42 h
Interfaz	78 h
Componente	54 h
Estado	36 h
Persistencia	54 h
Exp. Usuario	36 h
TOTAL	300 h

EJECUCIÓN

Investigación	42 h
Interfaz	81 h
Componente	54 h
Estado	36 h
Persistencia	54 h
Exp. Usuario	33 h
TOTAL	300 h



EXPERIENCIA DE USUARIO





EXPERIENCIA DE USUARIO

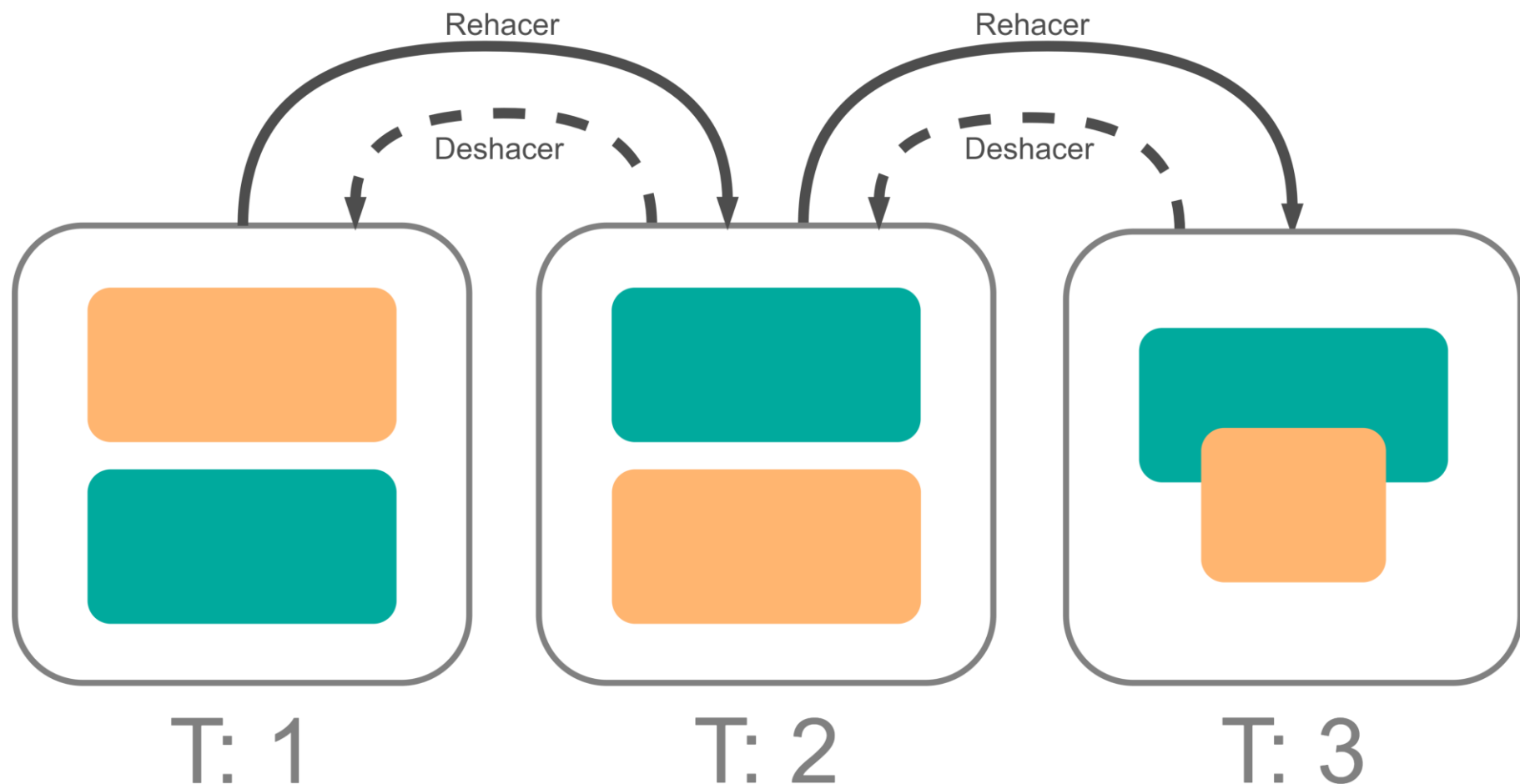
- Carga de imágenes
- Interacción area de trabajo
- Manejo de componentes
- Control de interfaz y ventana
- Menus contextuales
- Manejo de estados
- Guardado y Carga de XML
- Atajos de teclado ...

Interacciones
De Usuario





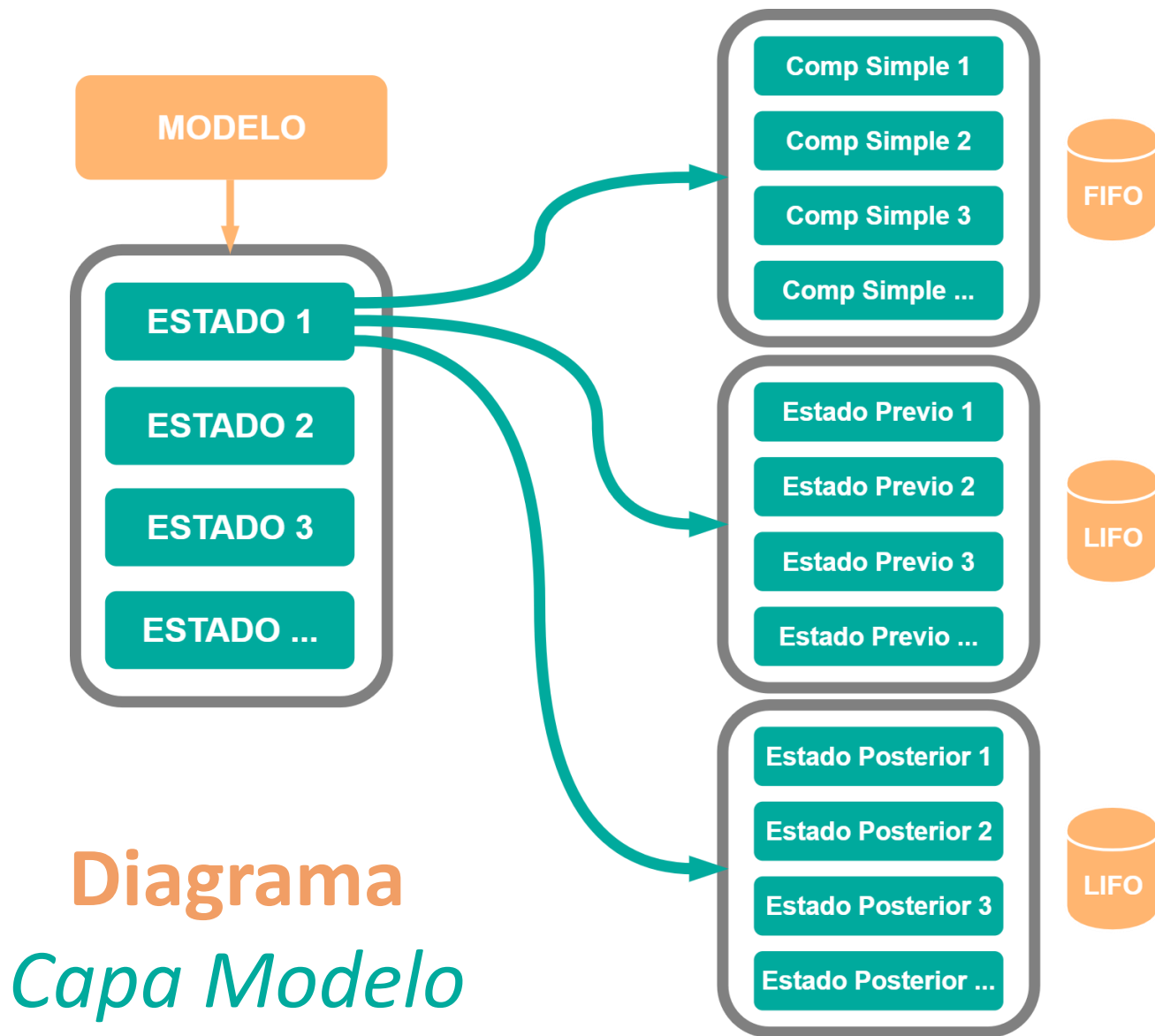
EXPERIENCIA DE USUARIO



Historial *de Acciones* (*UNDO*)



DETALLES IMPLEMENTACIÓN





OTRAS CARACTERÍSTICAS

```
(04:34:45.160110) Loading project from D:/devbox/py/prototypist/Demos/youtube.xml
(04:34:45.166654) XML validation against DTD: True
(04:34:49.847242) Decrementated viewport zoom to 62.09%
(04:34:52.398041) Increment viewport zoom to 68.3%
(04:35:05.986414) Moved item SC_x0v, 3.2210200000000253 pos on X, 0.0 pos on Y
(04:35:08.885463) Deleted a batch of components
(04:35:19.890949) Undone previous action
(04:35:21.042222) Redone subsequent action
(04:35:26.090219) State switched to: 1
(04:35:27.361843) Clonated state to the end
(04:35:27.366335) Context menu over states tree
(04:35:57.873281) Saving project on C:/Users/camba/Desktop/yt_xml/yt.xml
(04:36:04.466416) SESSION ENDED !
```

LOG de Acciones



VALIDACIÓN MARCO TEÓRICO



```
def xml4dtd(xmlFile, dtdFile):  
  
    # Load DTD and check it syntax.  
    try:  
        dtd = etree.DTD(dtdFile)  
    except lxml.etree.DTDParseError:  
        print('WARNING: DTD had a bad syntax.')  
        sys.exit()  
  
    # Load XML and check it syntax  
    try:  
        rawFile = objectify.parse(xmlFile)  
    except lxml.etree.XMLSyntaxError:  
        print('WARNING: The file is empty or had a bad syntax.')  
        sys.exit()  
  
    # Objetify XML  
    oneLine = etree.tostring(rawFile)  
    rootNode = objectify.fromstring(oneLine)  
  
    # Validacion del XML cargado.  
    valid = dtd.validate(rootNode)  
    print('XML validation = {}'.format(valid))  
  
    # Shows the reasons why the DTD does not validate.  
    if dtd.error_log.filter_from_errors():  
        print('\nReasons:\n-----')  
        print(dtd.error_log.filter_from_errors()[0])  
  
    # Returns the node and her valid state.  
    return rootNode, valid
```

¿VÁLIDO?





DEMOSTRACIÓN



FUTURAS AMPLIACIONES



***PINCEL
DE TRAZOS***



***FIGURAS
GEOMÉTRICAS***



***COMPORTAMIENTO
DE COMPONENTES***



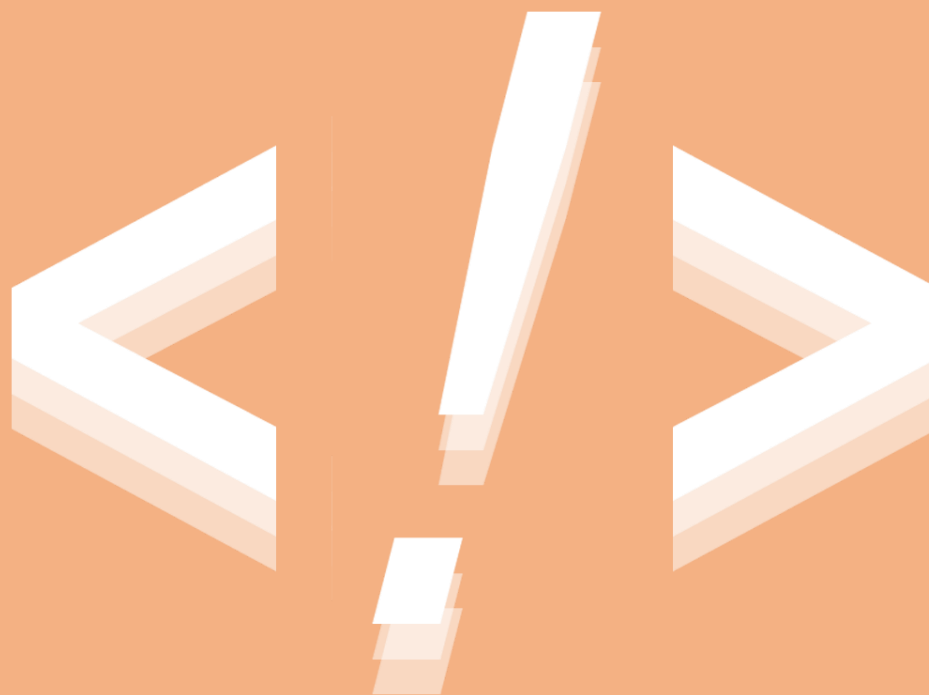
CONCLUSIONES

PROFESIONALES

- Acercamiento al area de informática gráfica
- Desarrollo de lógicas estandarizadas
- Continuación de trabajos de otros profesionales

PERSONALES

- Superar barrera de frontar nuevas tecnologías
- Satisfacción al superar determinados retos
- Desarrollo de un curso en UDEMY



GRACIAS
POR SU ATENCIÓN