



# WebRatio: Low-Code para Aplicaciones Web

**José Alfonso Aguilar Calderón, Ph.D**

Profesor-Investigador  
Facultad de Informática Mazatlán  
Universidad Autónoma de Sinaloa



Prácticas modernas para crear software con calidad y sabor  
#SGVirtual

# Me presento

- Profesor-Investigador en **Facultad** de Informática **Mazatlán** de la Universidad **Autónoma** de Sinaloa hace 11 años.
- **Docente** de Licenciatura, Maestría y Doctorado.
- Sistema **Nacional** de Investigadores Nivel 1 en **Ingeniería** y Tecnología desde hace **10** años.
- Trabajé como **programador** de escritorio y web.
- **Aficionado** de la Usabilidad Web, Accesibilidad Web, Flutter, Ing. Requerimientos en Web y en **IoT**.
- **Participo** con amigos en proyectos de TI.

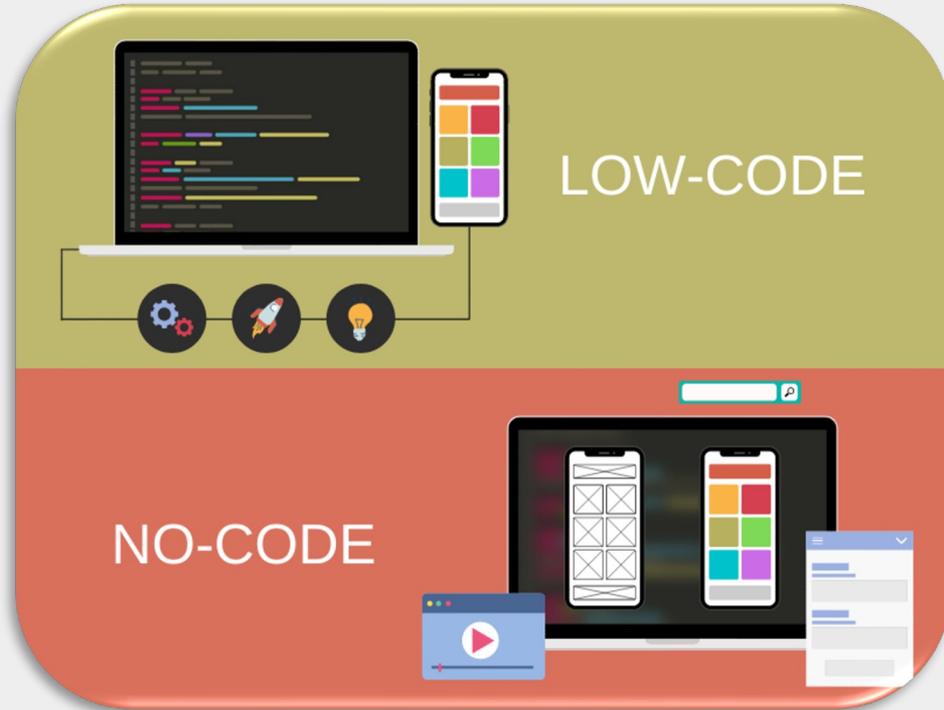
# Hablaremos de

- Low-Code (poco código) y No-Code (sin código).
- WebRatio: Low-Code Tool for Web Application Development.



# Low-Code VS No-Code

¿De donde provienen?



# Antecedentes del Low-Code y No-Code

- **Lenguajes** de Programación de Cuarta Generación.
  - SQL, ABAP, **Oracle** Reports.
  - **Décadas** de los 70s-90s aprox.
- Rapid Application **Development** (RAD).
  - Surge en los **80s**, pero en los 90s cobró fuerza.
  - Desarrollo de prototipos con herramientas **visuales**.
  - **Visual Basic**, Delphi y Oracle Forms.
- **Model-Driven** Development (MDD).
  - Surge en el **2000**.
  - Utiliza Model-Driven **Architecture** (MDA).
  - Propuesta por el Object Management Group (OMG).
  - **Crear** software a partir de **modelos** (gráficos, textuales).



# Low-Code/No-Code

- **Acuñados** por Clay Richardson y John Rymer, analistas de Forrester, en un informe publicado en **2014** sobre herramientas para crear aplicaciones sin código.
- La idea detrás de Low-Code/No-Code es **democratizar** el acceso a la creación de **aplicaciones** y otros proyectos digitales que requieren **código** de por medio.
- No se trata de **eliminar** a los profesionales del código, solo **permitir** que cualquier profesional sea **capaz** de programar sin código o con el mínimo código posible.

# No-Code

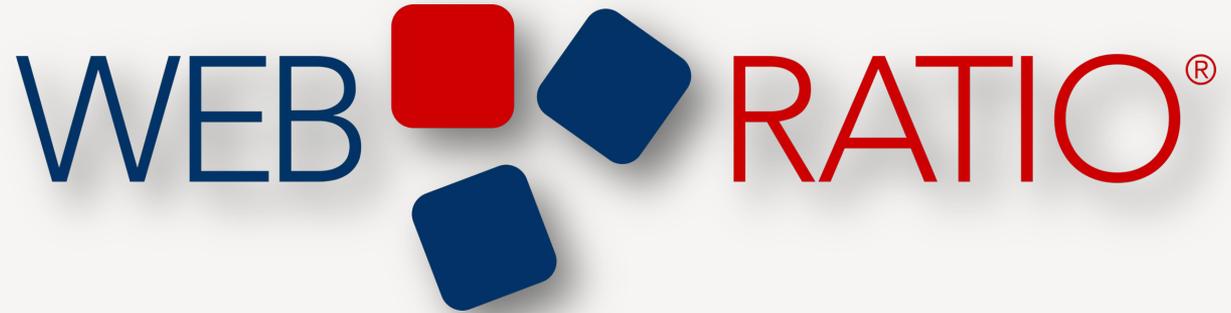
- **Nulo** Conocimiento de Programación.
- Drag **and** Drop.
- Aplicaciones **No-Robustas**.
- Son **útiles** para crear aplicaciones con un tiempo de vida **corto**.
- **Procesos** de negocio que no requieran **conexiones** a sistemas de **terceros**.



# Low-Code

- **Aplicaciones** que generan una cantidad considerable del Código Fuente del Proyecto.
- **Necesitan** ser complementadas por un ser humano para ser robustas.
- **Entorno** de desarrollo visual integrado, que permite arrastrar y soltar **componentes** de aplicaciones, conectarlos y crear una **aplicación** móvil o web.





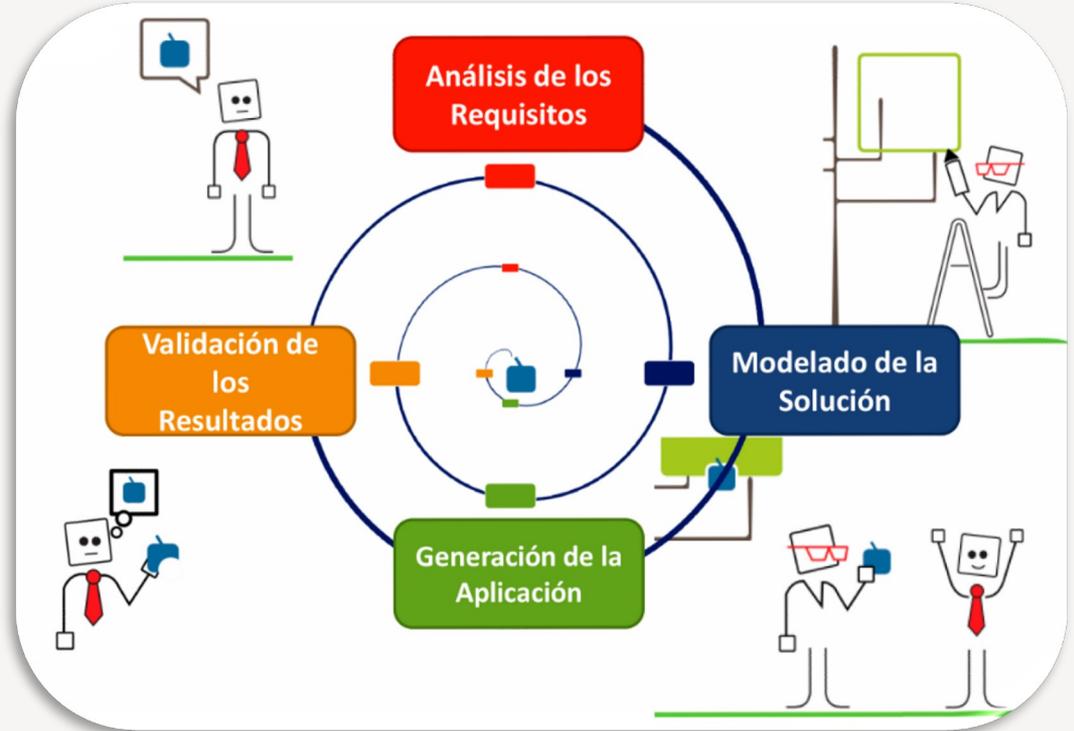
**WebRatio, ¿Qué es?**

# WebRatio, ¿Qué es?

- WebRatio (empresa) fue **fundada** en Milán Italia, como una **spin off** del Politécnico di Milano.
- WebRatio es un **entorno** de desarrollo ágil que te permite **desarrollar** aplicaciones personalizadas.
- Es un entorno de desarrollo Model-Driven (**Low-Code**), que permite producir **aplicaciones** web Java ejecutables en un entorno Web.
- **Captura** los requerimientos de negocio en **modelos** independientes de a tecnología de implementación a utilizar.

# ¿Cómo Funciona WebRatio?

- WebRatio tiene un ciclo de vida basado en el **prototipado** de la aplicación Web a desarrollar.
- **Durante** el ciclo de vida, el stakeholder (en este caso el usuario-directivo de la empresa) está **vinculado** durante el desarrollo del proyecto.



# Proceso de Desarrollo con WebRatio para Aplicaciones Web



- 1 MODELADO
- 2 DEFINICIÓN DEL LAYOUT
- 3 GENERACIÓN AUTOMÁTICA
- 4 ENTREGA MULTIDISPOSITIVO

# Proceso de Desarrollo con WebRatio para Aplicaciones Web

## 1. MODELADO

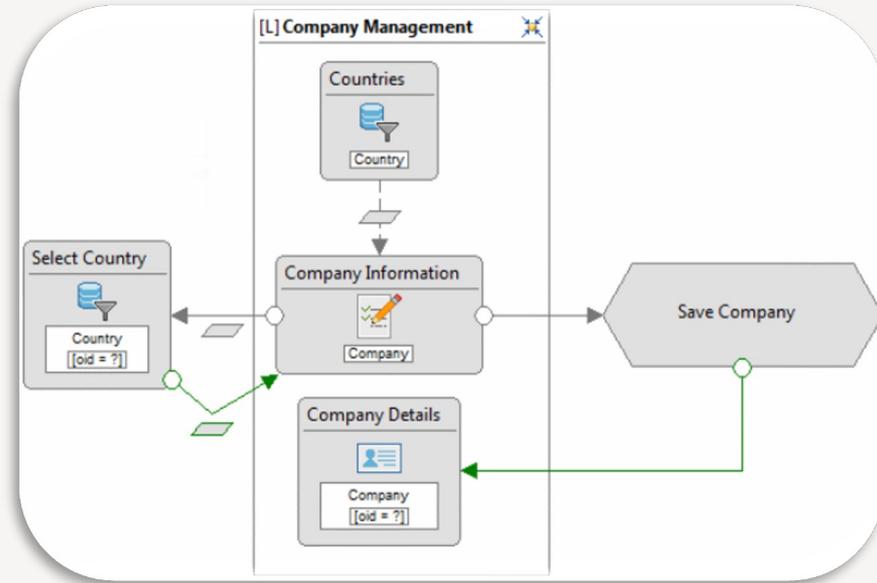
- Modelo de Flujos de Interacción
- Modelo de Dominio
- Modelo de Flujos de Integración
- Modelo de Procesos de Negocios



Modelado

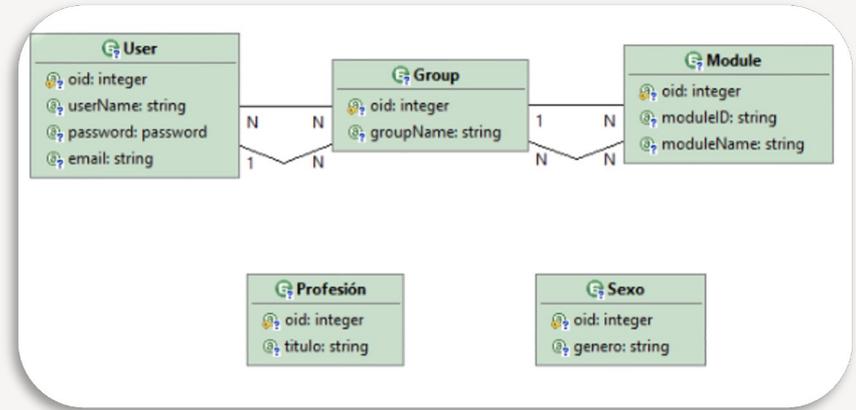
# Modelado de los Flujos de Interacción

- **Describe** la aplicación en su totalidad cómo los usuarios **interactúan** con la aplicación, la estructura de navegación, el contenido proporcionado y las acciones disponibles.
- Utiliza la notación **IFML** (Interaction Flow Modeling Language), el cual es un lenguaje estándar de la OMG.
- En un contexto Model-Driven, este sería nuestro modelo a convertir a código fuente.



# Modelo de Dominio

- Describe la organización conceptual de los datos de la aplicación, almacenando la información ingresada por el usuario y guardada por la aplicación.
- El modelo de dominio puede estar representado en formato ER o UML.



# Modelo de Flujos de Integración

- Describe la secuencia de las operaciones necesarias a fin de cumplir la lógica del negocio una vez que los usuarios ejecutan un evento.
- El componente Action de IFML maneja esta situación en la versión actual de WebRatio.

# Proceso de Desarrollo con WebRatio

## 2. LAYOUT



- Permite crear estilos para aplicaciones web o móviles, utiliza lenguajes de diseño estándar como HTML, CSS y Javascript.
- Utiliza estilos predeterminados o crea uno propio: utiliza templates predefinidos o crea uno a medida.

# Proceso de Desarrollo con WebRatio

## 3. GENERACIÓN AUTOMÁTICA

- Genera automáticamente código enterprise-class.
- Aplicación Web Java personalizada.
- Estándar y sin runtime propietario.



# 3. Generación Automática



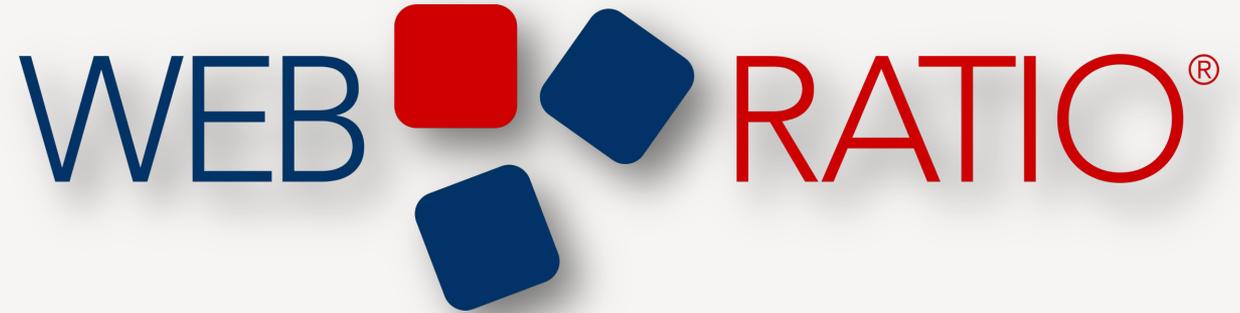
- **Aplicaciones** generadas y empaquetadas.
  - Genera código Java estándar y altamente optimizado que asegura la escalabilidad.
- Generación de código estándar.
  - Despliega las aplicaciones con 1 click, utiliza el poder expresivo de SQL y el amplio repositorio de JQuery enriquecido con UI plugins, fragmentos de JavaScript y templates HTML5 y CSS.
- Soporte la construcción e integración continua.
  - El código generado puede ser invocado por una serie de procedimientos.

# Proceso de Desarrollo con WebRatio

## 4. ENTREGA MULTIDISPOSITIVO



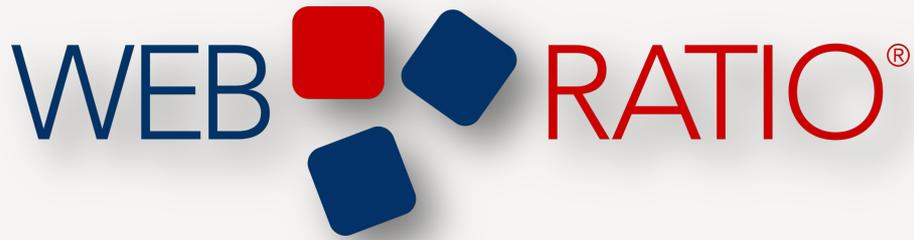
- WebRatio permite el desarrollo de aplicaciones accesibles desde la Web y Móviles.
- Uso de HTML5 y CSS para crear interfaces web responsivas.
- IFML para definir un flujo de interfaz de usuario diferente para cada dispositivo.
- Crea una UI personalizada capaz de darle al usuario una experiencia diferente en cada canal.



## Ventajas de Utilizar WebRatio

# Ventajas de Utilizar WebRatio

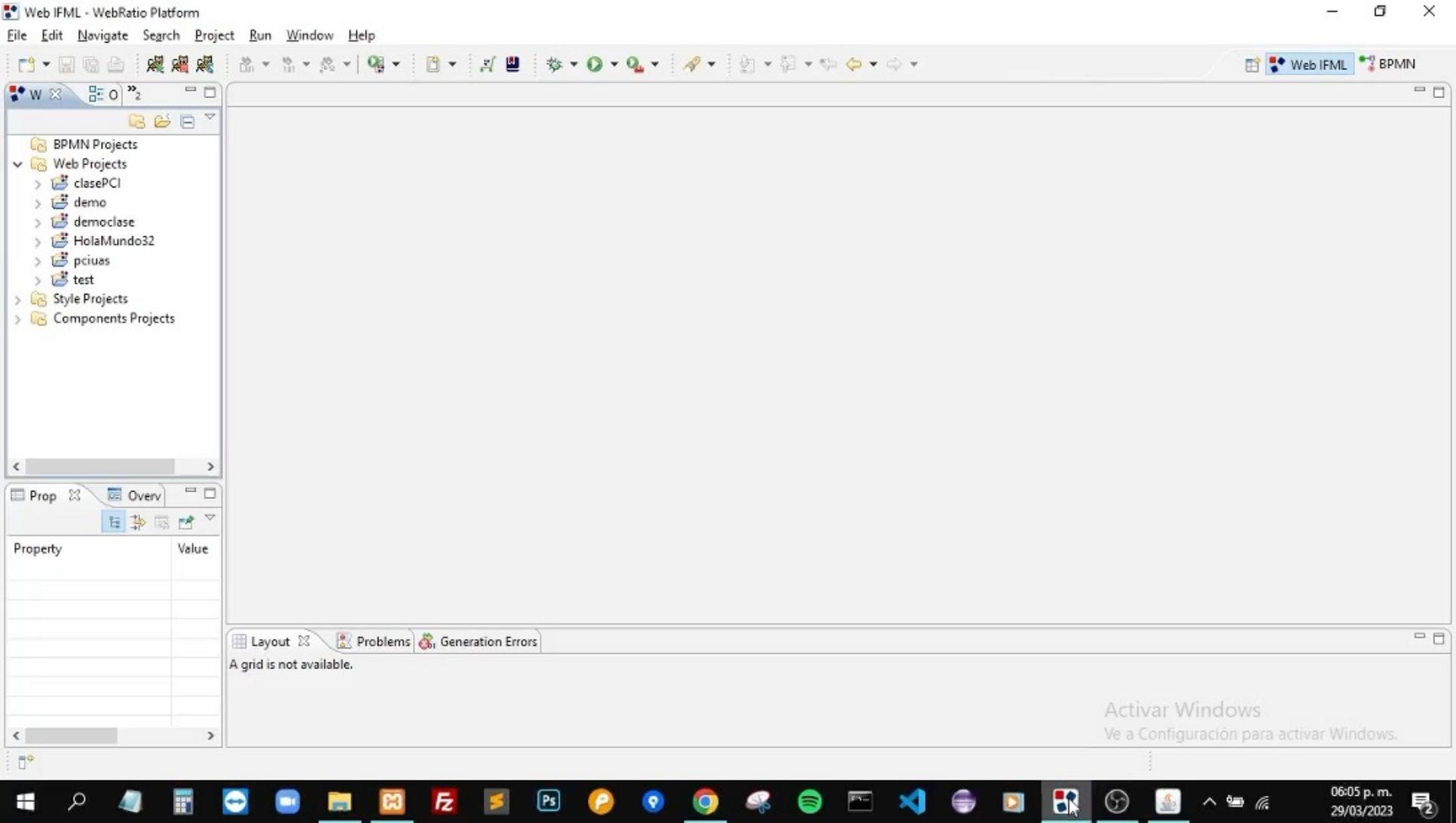
- **Sincronía** entre la aplicación Web y los objetivos de negocio.
- **Desarrollo** rápido, acorta el tiempo de desarrollo por lo que aumenta la productividad.
- **Reutilización** del software a nivel de componentes.
- **Uniformidad** en el ambiente de desarrollo empresarial.
- **No** se necesita ningún control o componente bajo licencia comercial.
- **Corta curva** de aprendizaje.



# Demo

**Web App** para registro de  
conferencistas de SG Virtual  
**Conference**

- Crear base de datos (MySQL).
- Crear conexión.
- Crear Tablas.
- Modelar la Funcionalidad de la Web App.
- Generar Código Fuente.



- BPMN Projects
- Web Projects
  - classPCI
  - demo
  - democlase
  - HolaMundo32
  - pcius
  - test
- Style Projects
- Components Projects

Prop Overv

Property	Value

A grid is not available.

Activar Windows  
Ve a Configuración para activar Windows.

# Finalmente

- El uso de un lenguaje de modelado visual y abstracto, no te hará perder tiempo en tareas repetitivas sino que te concentrarás en el valor agregado de las actividades, incrementando tu la productividad.
- Un modelado visual fácil de entender y aprender, ayuda a una mejor interacción entre los negocios y las TICs.
- Al construir aplicaciones con IFML, se ofrece mayor simplicidad sobre lenguajes de programación JAVA.
- El estándar IFML es rápido de aprender para los inexpertos y mucho más rápido en el caso de los desarrolladores con experiencia.
- Curva de aprendizaje muy corta.



# ¡Gracias!

¿Preguntas?

**Facebook:** ProfJosé Alfonso Aguilar

**Twitter:** @Mode2001

**YouTube:** @ANOVALABMX33

**LinkedIn:** José Alfonso Aguilar

**Correo E.:** ja.aguilar@uas.edu.mx