



# Reacciona!

Extensiones de Reactive y ReactiveUI

# Sponsors

Sin ellos no sería posible el evento!



# MC \_18

# Ciani Afonso



He estado programando con las tecnologías de Microsoft durante los últimos diez años, he desarrollado complejos sistemas de rastreo GPS, software de seguros y aplicaciones móviles con Xamarin durante los 4 últimos años

# ¿Qué es la programación reactiva?

La Programación reactiva, es un estilo de **microarquitectura y paradigma** que implica el enrutamiento inteligente y el consumo de eventos, para cambiar el comportamiento de una aplicación.

“La programación reactiva es una programación con flujos de datos asíncronos”

# Acerca de las extensiones de Reactive

En 2010 Microsoft publica RX.net

Es un conjunto de librerías diseñadas para ayudar con la creación de aplicaciones asíncronas y basada en eventos.

Es open source

<http://reactivex.io/>



[ReactiveX](#) is a collection of open source projects. The content of this page is licensed under Creative Commons Attribution 3.0 License, and code samples are licensed under the BSD License.

devs  dna



# Acerca de las extensiones de Reactive

Lenguajes soportados

- Java: [RxJava](#)
- JavaScript: [RxJS](#)
- C#: [Rx.NET](#)
- C#(Unity): [UniRx](#)
- Scala: [RxScala](#)
- Clojure: [RxClojure](#)
- C++: [RxCpp](#)
- Lua: [RxLua](#)
- Ruby: [Rx.rb](#)
- Python: [RxPY](#)
- Go: [RxGo](#)
- Groovy: [RxGroovy](#)
- JRuby: [RxJRuby](#)
- Kotlin: [RxKotlin](#)
- Swift: [RxSwift](#)
- PHP: [RxPHP](#)
- Elixir: [reaxive](#)
- Dart: [RxDart](#)

**C#**

**JAVA**

**C++**

**JavaScript**

**Python**

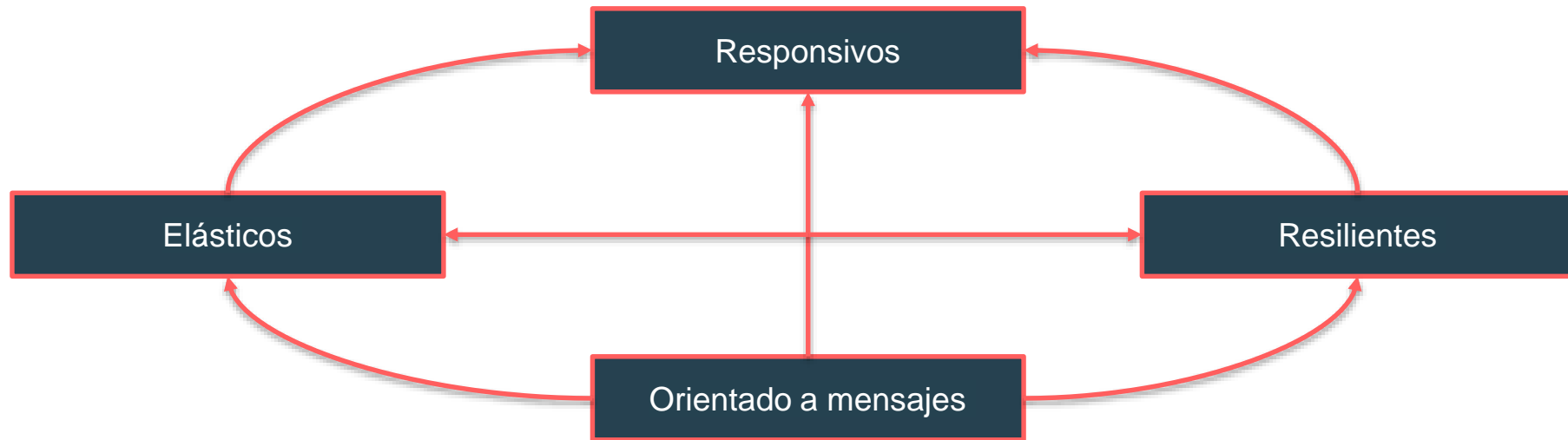
**Otros**



devs  dna



# En 2014 se publica el Manifiesto Reactivo



# Acerca de las extensiones de Reactive

¿Quién lo usa?





# Acerca de las extensiones de Reactive

¿Dónde deberíamos usarlo?



Datos en tiempo real



Streams de datos



Interfaz de usuario



devs dna



# Conceptos importantes



# Conceptos importantes

## IObservable<T>

- Envía los datos a los subscriptores
- Lo podemos ver como una cinta transportadora
- Puede tener o no un fin
- Los datos pueden ser procesados y filtrados



# IObservable<T>

## Ejemplos de IObservable<T>

```
Observable.Range(1,100);
```

```
Observable.Time(TimeSpan.FromSeconds(3));
```

```
Observable
```

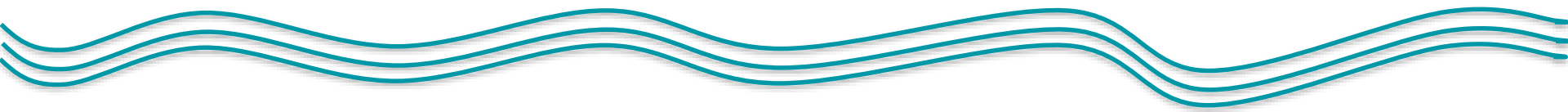
```
.FromEventPattern<EventHandler<TextChangedEventArgs>, TextChangedEventArgs>(
    x => this.textEntry.TextChanged += x,
    x => this.textEntry.TextChanged -= x)
```



devs dna



# IObservable<T>



devs dna



# Conceptos importantes

## Suscripciones

- Obtiene los datos de los **IObservable<T>**
- Cada valor se procesa con un **Action<T>**
- Podemos declarar varias suscripciones sobre un único **IObservable<T>**
- Nos podemos de-subscribir usando **IDisposable**



```
IObservable<T>()  
.Subscribe(obj => obj { $ Fish })
```



# Suscripciones

## Ejemplo de suscripciones

```
var fishSubscription =  
    fishObservable.Subscribe(fish => Debug.WriteLine($"{fish.Name}));
```

```
fishSubscription.Dispose();
```

```
var fishSubscription =  
    fishObservable  
        .Where(fish => !fish.IsTooSmall)  
        .Subscribe(fish => Debug.WriteLine($"{fish.Name}));
```

```
fishSubscription.Dispose();
```





# Conceptos importantes

## CompositeDisposable

- Un contenedor para objetos *disposeables*
- Nos permite *disposear* múltiples objetos a la vez
- Viene muy bien para *disposear*
  - Componentes de UI
  - Componentes nativos o de plataforma
  - Recursos no manejados



# CompositeDisposable

## Ejemplo de CompositeDisposable

```
CompositeDisposable disposables = new CompositeDisposable();
```

```
protected override void OnAppearing()  
{  
    base.OnAppearing();  
    myCompositeDisposable.Add(fishObservable  
                             .Where(fish => !fish.IsTooSmall)  
                             .Subscribe(fish => Debug.WriteLine($"{fish.Name})));  
}
```



# CompositeDisposable

## Ejemplo de CompositeDisposable

```
protected override void OnDisappearing()  
{  
    base.OnDisappearing();  
    myCompositeDisposable.Clear()  
}
```





Muéstrame  
el código



# Recursos

<http://reactivex.io/>

# Código

<https://github.com/ciani/Reactive-Examples>

fork

<https://github.com/TheEightBot/Reactive-Examples>

¿Quieres saber más sobre Xamarin? Mira nuestro canal de YouTube: <http://bit.ly/DevsDNA>



# Monkey Conf \_18

¡GRACIAS!