

Smart Devices

GeneXus™ 15

Navigation

Navigation with CallOptions

```

Navigation
├── Properties
│   ├── ShowTarget      None
│   ├── TargetName     Character(40)
│   └── HideTarget      None
├── Methods
│   ├── ShowTarget      None
│   ├── TargetName     Character(40)
│   └── HideTarget      None
└── TargetName         Character(40)
  
```

```

Event ClientStart
Composite
  PanelChatTips.CallOptions.Target = !"Right"
  PanelChatTips()
EndComposite
EndEvent

Event 'DisplayChatTips'
  Navigation.ShowTarget(!"Right")
EndEvent
  
```

Muchas apps utilizan el patrón de Slide navigation para desplegar un menú haciendo tap sobre un ícono en la parte superior izquierda de la pantalla (familiarmente conocido como “ícono de hamburguesa”).

Esquemáticamente, el panel principal (nuestro menú) es cargado en el costado izquierdo de la app y permanece escondido hasta que el usuario hace tap sobre el ícono. Esta manera de mostrar contenido ha sido expandida a otras secciones, no solo a esa izquierda. Por esta razón GeneXus 15 incorpora el right navigation style utilizando el atributo **CallOptions** de un panel (para iOS también existe el bottom navigation style).

En el ejemplo, observemos que el ícono de hamburguesa tiene el comportamiento conocido, pero hemos incluido un nuevo botón en el lado derecho de la Application Bar: un help icon. Su propósito es desplegar un panel de help contextual una vez que el usuario final hace tap en el ícono.

Para implementar esto, primero que nada observemos que aparece el Nuevo objeto externo Navigation.

Y luego lo primero que tenemos que hacer es lograr que el estilo de navegación de la app sea Slide. Esto ya sabemos hacerlo.

Luego:

Indicar a través del Target attribute de las **CallOptions** del panel que queremos invocar a la derecha... ¡que queremos invocarlo a la derecha, justamente!

Y realizar luego la invocación al objeto. Por defecto esta acción simplemente carga el panel en el sector derecho pero no lo muestra.

Para mostrarlo, debemos utilizar el objeto externo **Navigation**.

Esto es análogo a lo que vimos en Web.

Define your own Search