Using the IGxDialog object

```
Private Sub BtnAdd_Click(ByVal sender As System.Object, ByVal e _
                        As System.EventArgs) Handles BtnAdd.Click

      'Declare ArcObjects outside of Try/Catch so we can dispose of them in
      Finally
      Dim bObjectSelected As Boolean
      Dim pGxDialog As IGxDialog = New GxDialog
      'Only show shapefiles
      Dim pFilter As IGxObjectFilter = New GxFilterShapefiles
      'Collection of layers to be assigned here
      Dim pGxObjects As IEnumGxObject = Nothing
      Dim pGxDataset As IGxDataset
      Dim pFLayer As IFeatureLayer

      Try
      'initialize and open mini browser
      With pGxDialog
            .AllowMultiSelect = False
            .ButtonCaption = "Select"
            .Title = "Select layer"
            'using shapefile filter
            .ObjectFilter = pFilter
            'open dialog passing handle to Application from AddIn
             bObjectSelected = .DoModalOpen(My.ArcMap.Application.hWnd,
      pGxObjects)
      End With

      'no file selected; exit
      If bObjectSelected = Nothing Then Exit Sub

      'get first dataset
      pGxDataset = pGxObjects.Next
      'no dataset has been selected
      If pGxDataset Is Nothing Then Exit Sub

      'iterate through remaining datasets
      Do Until pGxDataset Is Nothing
            pFLayer = New FeatureLayer
            'assign dataset to featureLayer
            pFLayer.FeatureClass = pGxDataset.Dataset
            'assign dataset name to featureLayer
            pFLayer.Name = pGxDataset.Dataset.Name
            ' Do something with the featureLayer
            ' Your code here
            ' Go to next dataset
            pGxDataset = pGxObjects.Next
      Loop
      Catch ex As Exception
            MessageBox.Show("BtnAdd_Click Exception: " & ex.Message)
      Finally
      ' Release ArcObjects objects
            pGxObjects = Nothing
            pFilter = Nothing
            pGxDialog = Nothing
            pGxDataset = Nothing
            pFilter = Nothing
            pFLayer = Nothing
            ' Call garbage collection
            GC.Collect()
            GC.WaitForPendingFinalizers()
      End Try

End Sub
```

Accessing the map document

This is an enhanced version of the sample code from the ArcGIS Resource Center.
http://help.arcgis.com/en/sdk/10.0/arcobjects_net/conceptualhelp/index.html#/How_to_access_maps_and_layers_via_the_MapDocument/0001000001n3000000/

```
'Dim ArcObjects objects outside of Try/Catch
Dim pMxDocument As IMxDocument
Dim pMaps As IMaps
Dim pMap As IMap

Try
      ' Access to current Document from addIn
      pMxDocument = My.ArcMap.Document
      ' Maps collection from current Document
      pMaps = pMxDocument.Maps
      ' Initialize the counter
      Dim i As Integer
      ' Loop through each map in the document
      For i = 0 To pMaps.Count - 1 Step i + 1
            pMap = pMaps.Item(i)
            ' Write map name to log
            Debug.WriteLine(pMap.Name)
            ' Layers collection from current map
            Dim pEnumLayer As IEnumLayer = pMap.Layers(Nothing, True)
            ' Move to beginning of layers collection
            pEnumLayer.Reset()
            ' Retrieve first layer
            Dim pLayer As ILayer = pEnumLayer.Next()
                  While Not pLayer Is Nothing
                        MsgBox(pLayer.Name)
                        ' Move to next layer
                        pLayer = pEnumLayer.Next()
                  End While
      Next
Catch ex As Exception
      MsgBox("onClick() Exception: " + ex.Message)
Finally
      ' Release ArcObjects resources
      pMap = Nothing
      pMaps = Nothing
      pMxDocument = Nothing
      ' Call garbage collection
      GC.Collect()
      GC.WaitForPendingFinalizers()
End Try`
```