

Membuat Flyout Menu pada Android App menggunakan Xamarin

Junindar, ST, MCPD, MOS, MCT, MVP .NET

junindar@gmail.com

<http://junindar.blogspot.com>

Lisensi Dokumen:

Copyright © 2003 IlmuKomputer.Com

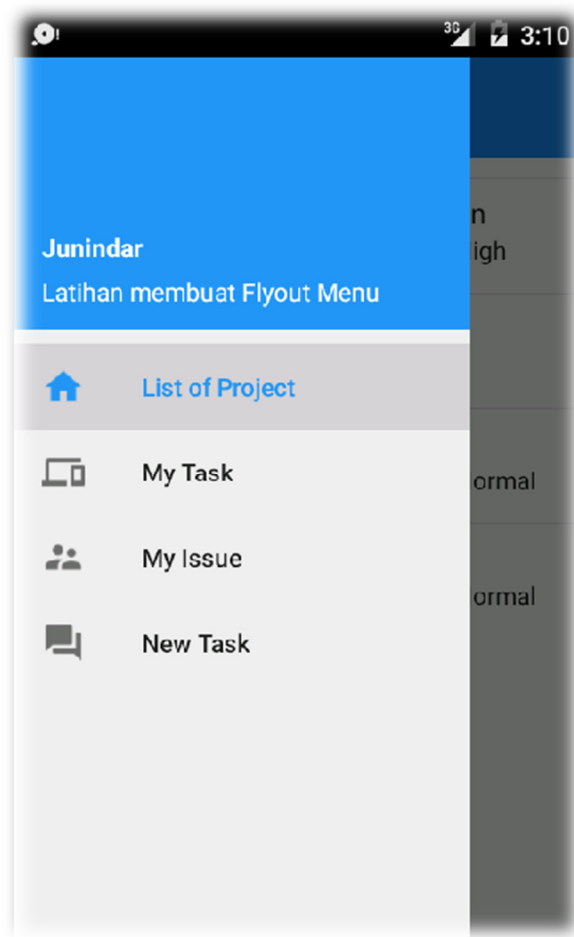
Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Abstrak

Android adalah salah satu *mobile oprating system* yang saat banyak digunakan di berbagai *smartphone*, baik yang berspesifikasi rendah maupun mumpuni. Dikarenakan android saat ini merupakan salah satu mobile OS paling digemari, maka tidak dipungkiri sangat banyak sekali aplikasi-aplikasi mobile untuk android. Untuk membuat aplikasi mobile ini ada beberapa tools yang bisa digunakan dan salah satunya adalah dengan menggunakan Xamarin.

Pendahuluan

Pada artikel ini akan dijelaskan bagaimana cara untuk membuat flyout menu pada android app dengan menggunakan Xamarin. Seperti telah dijelaskan pada artikel sebelumnya (<http://junindar.blogspot.my/2016/02/menampilkan-data-sharepoint-list-pada.html>), Xamarin adalah salah tools untuk membuat aplikasi mobile dimana bahasa pemograman yang digunakan adalah C#. Untuk memudahkan dalam memahami artikel ini, disarankan untuk menyelesaikan latihan pada artikel sebelumnya pada link diatas.



Flyout menu saat ini sering digunakan pada aplikasi-aplikasi android, sebut aja aplikasi Gmail, dimana menu ditampilkan dengan melakukan slide kekiri atau kekanan pada screen untuk menampilkan menu tersebut.

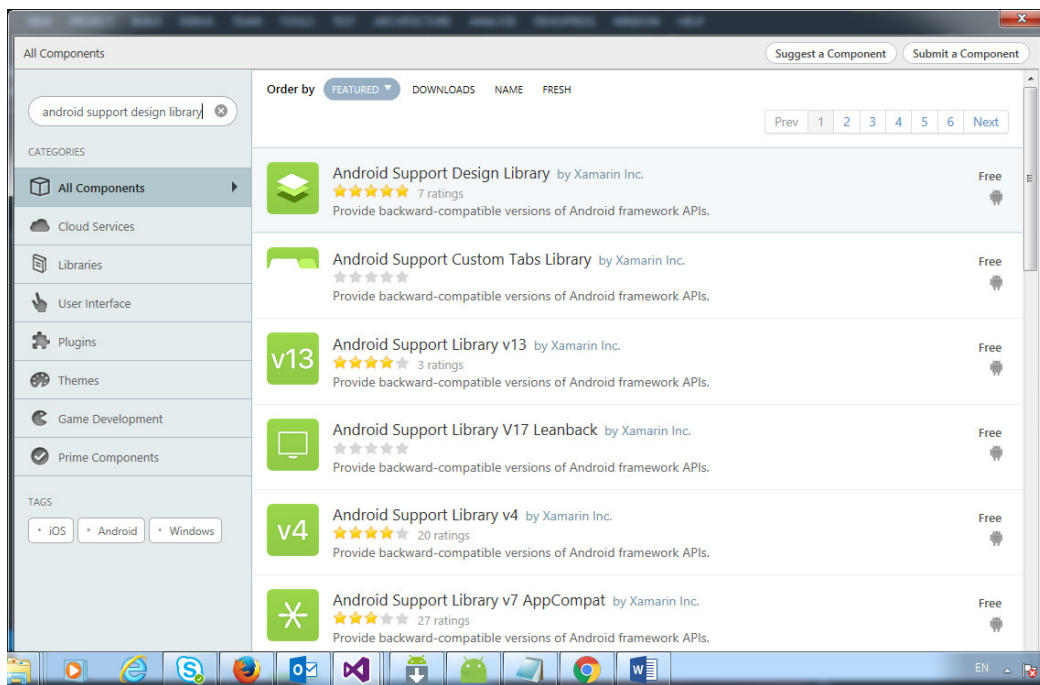
Isi

Untuk memudahkan memahami isi dari artikel ini, kita akan membuat sebuah project latihan, Untuk memudahkan gunakan project xamarin pada artikel sebelumnya. Dan buka project tersebut menggunakan Visual Studio.

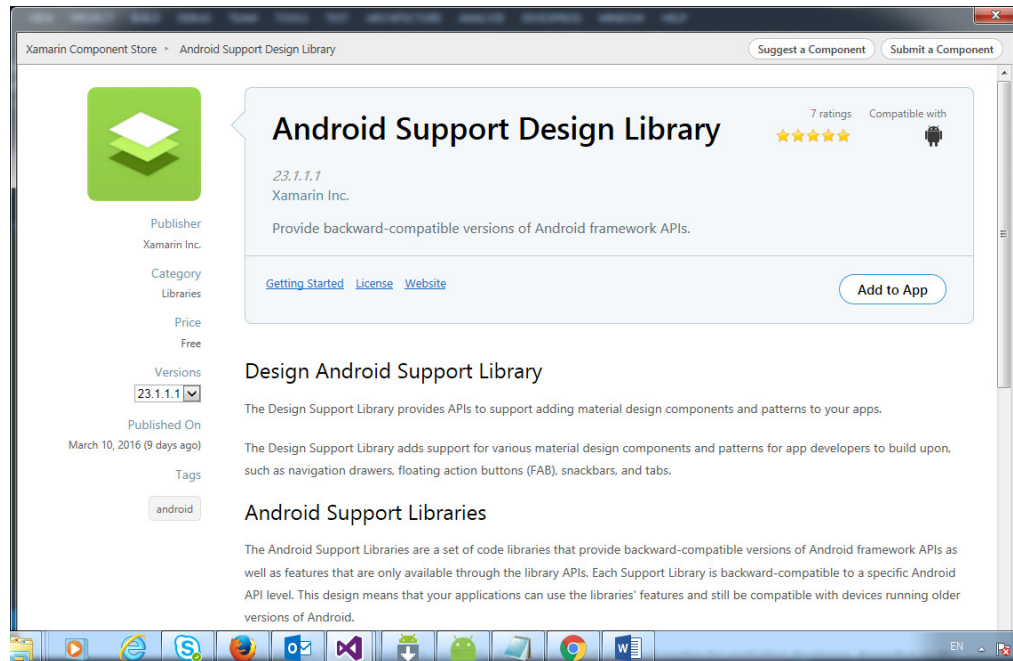
Menambahkan Component Pada Project

Sebelum masuk ke sintaks terlebih dahulu kita tambahkan beberapa component yang diperlukan untuk proses pembuatan flyout menu ini, Component-component ini adalah *Android Support Design Library* dan *Android Support Library v7 AppCompat*.

1. Pada project klik kanan Node Components lalu pilih Get More Components
2. Lalu pada form component serach dua component diatas.

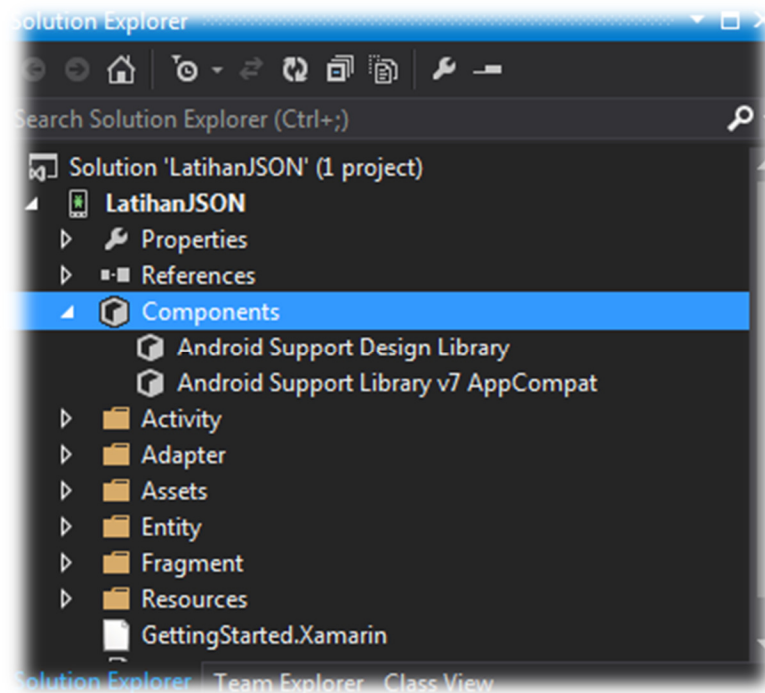


3. Klik component tersebut dan klik button “Add to App”



4. Lakukan langkah-langkah diatas untuk component *Android Support Library v7 AppCompat*.

Setelah kita melakukan langkah diatas, maka kita akan mendapatkan dua component pada project kita, seperti gambar dibawah.

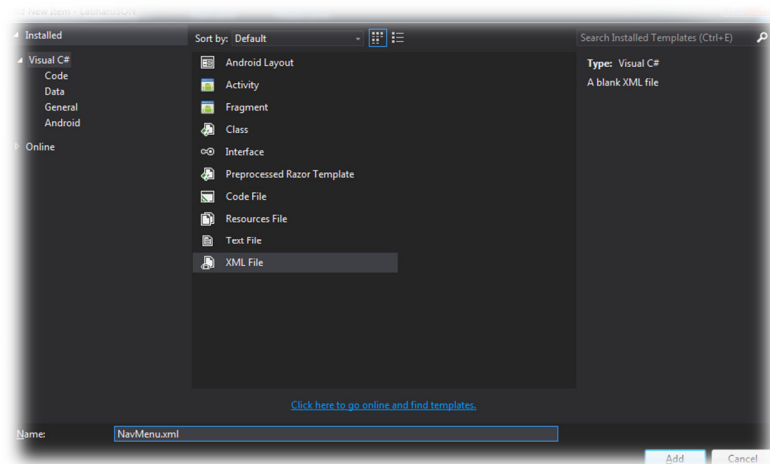


Buka Strings.xml pada folder Resources>Values dan tambahkan dua string berikut kedalam tag <resources>.

```
<string name="open_drawer">Open drawer</string>  
<string name="close_drawer">Close drawer</string>
```

Membuat Text Menu pada XML

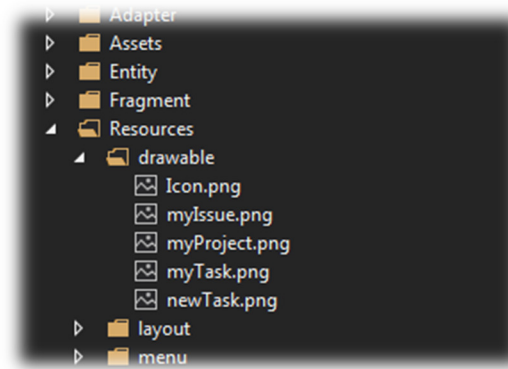
1. Tambahkan sebuah folder Menu didalam Resources folder.
2. Pada folder menu tambahkan xml file dengan nama "NavMenu".



3. Dan tambahkan sintaks berikut pada file NavMenu.

```
<?xml version="1.0" encoding="UTF-8" ?>  
<menu xmlns:android="http://schemas.android.com/apk/res/android">  
  <group android:checkableBehavior="single">  
    <item  
      android:id="@+id/nav_listproject"  
      android:icon="@drawable/myProject"  
      android:title="List of Project" />  
    <item  
      android:id="@+id/nav_mytask"  
      android:icon="@drawable/myTask"  
      android:title="My Task" />  
    <item  
      android:id="@+id/nav_myissue"  
      android:icon="@drawable/myIssue"  
      android:title="My Issue" />  
    <item  
      android:id="@+id/nav_newtask"  
      android:icon="@drawable/newTask"  
      android:title="New Task" />  
  </group>  
</menu>
```

Pada file xml diatas, kita telah membuat 4 item pada menu. Masing-masing item terdapat id, icon dan title dari menu. Seandainya kita tidak menginginkan icon pada menu, hapus baris “`android:icon="@drawable/myIssue"`”. myIssue merupakan nama icon yang akan digunakan. Dan langkah selanjutnya adalah memasukkan icon menu pada project pada folder Resources > drawable, seperti pada gambar dibawah ini.



Setting Themes untuk Flyout Menu

1. Pada folder “values” tambahkan sebuah xml file dengan nama “Styles”.
2. Dan ketikkan sintaks seperti dibawah ini.

```
<?xml version="1.0" encoding="UTF-8" ?>
<resources>
    <style name="MyTheme.Base" parent="Theme.AppCompat.Light.DarkActionBar">
        <item name="windowNoTitle">true</item>
        <item name="windowActionBar">false</item>
        <item name="colorPrimary">#2196F3</item>
        <item name="colorPrimaryDark">#1976D2</item>
        <item name="colorAccent">#FF4081</item>
    </style>

    <style name="MyTheme" parent="MyTheme.Base">
        <item name="android:windowContentTransitions">true</item>
        <item name="android:windowAllowEnterTransitionOverlap">true</item>
        <item name="android:windowAllowReturnTransitionOverlap">true</item>
        <item
name="android:windowSharedElementEnterTransition">@android:transition/move
</item>
        <item
name="android:windowSharedElementExitTransition">@android:transition/move<
/ item>
    </style>
</resources>
```

Membuat Header untuk Flyout Menu

1. Tambahkan sebuah “Android Layout” pada folder layout, dengan nama “Header.xml”.
2. Dan ketikkan sintaks seperti dibawah ini.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="178dp"
    android:background="?attr/colorPrimary"
    android:orientation="vertical"
    android:weightSum="1">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:orientation="vertical"
        android:layout_alignParentBottom="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="16dp"
            android:textColor="#ffffff"
            android:text="Junindar"
            android:textSize="14sp"
            android:textStyle="bold" />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textColor="#ffffff"
            android:layout_marginLeft="16dp"
            android:layout_marginTop="5dp"
            android:text="Latihan membuat Flyout Menu"
            android:textSize="14sp"
            android:textStyle="normal" />
    </LinearLayout>
</RelativeLayout>
```

Pada layout diatas, kita menambahkan dua buah TextView (Junindar dan Latihan membuat Flyout Menu).

Mengganti Main Layout

Selanjutnya kita akan mengubah pada Layout Main.xml.,

1. Buka Layout Main.xml
2. Lalu ubah sintaks nya seperti dibawah.

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v4.widget.DrawerLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:id="@+id/drawer_layout"
android:fitsSystemWindows="true">
<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">
<!-- The main content view -->
    <FrameLayout
        android:id="@+id/content_frame"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@android:color/white" />
<!-- The navigation drawer -->
    <android.support.v7.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:minHeight="?attr/actionBarSize"
        android:background="?attr/colorPrimary"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
        app:popupTheme="@style/ThemeOverlay.AppCompat.Light" />
</RelativeLayout>
<android.support.design.widget.NavigationView
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_gravity="start"
    android:id="@+id/nav_view"
    app:menu="@menu/navmenu"
    app:headerLayout="@layout/header" />
</android.support.v4.widget.DrawerLayout>
```

Pada Layout Main.xml ini, kita akan memanggil Layout Header (`app:headerLayout="@layout/header"`) yang telah kita buat sebelumnya dan menu item pada file navmenu.xml (`app:menu="@menu/navmenu"`)

Mengganti Sintaks pada MainActivity.cs

Selanjutnya kita akan mengganti sintaks pada MainActivity.cs

1. Pada folder Activity, buka file MainActivity.cs
2. Tambahkan namespace seperti dibawah dibawah ini.


```
using Android.App;
using Android.Widget;
using Android.OS;
using Android.Support.V4.Widget;
using Android.Support.V7.App;
```

Selanjutnya tambahkan dua buah method seperti dibawah.

```
private void RefreshForm(int position)
{
    var fragment = new ListProjectFragment();
    var arguments = new Bundle();
    arguments.PutInt(ListProjectFragment.ArgMenuNumber, position);
    fragment.Arguments = arguments;

    FragmentManager.BeginTransaction()
        .Replace(Resource.Id.content_frame, fragment)
        .Commit();
}

void NavigationView_NavigationItemSelected(object sender,
    NavigationView.NavigationItemSelectedEventArgs e)
{
    Toast.MakeText(this, e.MenuItem.TitleFormatted +
        " clicked.", ToastLength.Short).Show();
    drawerLayout.CloseDrawers();
}
```

Method pertama (RefreshForm) adalah untuk menampilkan data pada sharepoint yang telah dijelaskan pada artikel sebelumnya. Sedangkan yang kedua, berfungsi menampilkan nama menu item pada saat di klik. Selanjutnya pada OnCreate ganti sintaksnya seperti dibawah ini.

```
base.OnCreate(bundle);

SetContentView(Resource.Layout.Main);
drawerLayout = FindViewById<DrawerLayout>(Resource.Id.drawer_layout);

        // Init toolbar
var toolbar =
FindViewById<Android.Support.V7.Widget.Toolbar>(Resource.Id.toolbar);
SetSupportActionBar(toolbar);

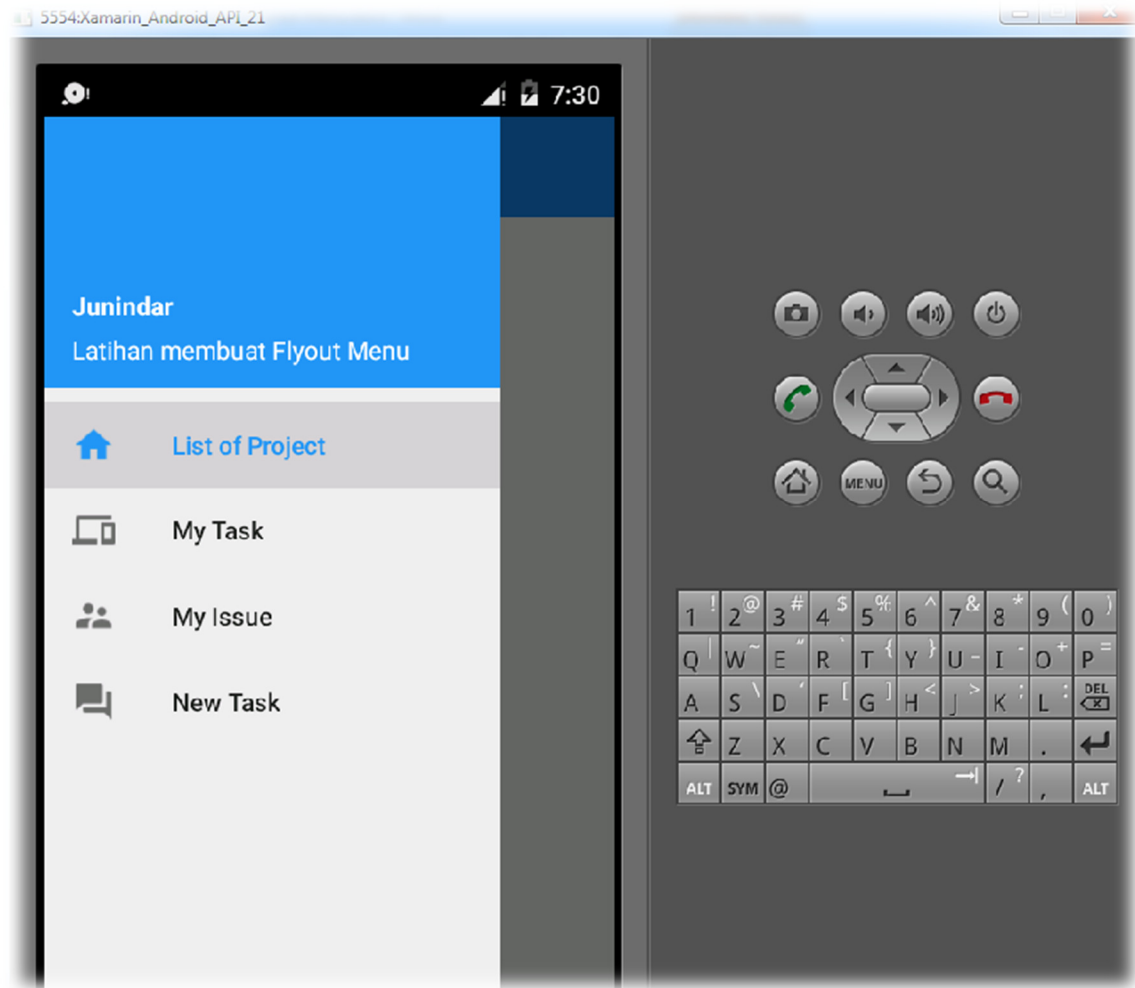
        // Attach item selected handler to navigation view
var navigationView = FindViewById<NavigationView>(Resource.Id.nav_view);
navigationView.NavigationItemSelected += NavigationView_NavigationItemSelected;

        // Create ActionBarDrawerToggle button and add it to the toolbar
var drawerToggle = new ActionBarDrawerToggle(this, drawerLayout, toolbar,
        Resource.String.open_drawer, Resource.String.close_drawer);

drawerLayout.SetDrawerListener(drawerToggle);
drawerToggle.SyncState();

navigationView.Menu.FindItem(Resource.Id.nav_listproject).SetChecked(true);
RefreshForm(0);
```

Jalankan program maka kita akan mendapatkan hasil seperti dibawah.



Penutup

Pada artikel ini telah dijelaskan bagaimana menggunakan Xamarin untuk membuat Flyout Menu pada Android App. Sedangkan untuk memudahkan dalam memahami isi artikel, maka penulis juga menyertakan dengan full source code project latihan ini, dan dapat di download disini.

<http://junindar.blogspot.co.id/2016/04/membuat-flyout-menu-pada-android-app.html>

Referensi

1. www.msdn.microsoft.com
2. www.planetsourcecode.com
3. www.codeproject.com
4. www.aspnet.com

Masih banyak lagi referensi yang ada di Internet. Anda tinggal cari di www.Google.com. Dengan kata kunci “**tutorial VB.Net**”

Biografi Penulis.



Junindar Lahir di Tanjung Pinang, 21 Juni 1982. Menyelesaikan Program S1 pada jurusan Teknik Informatika di Sekolah Tinggi Sains dan Teknologi Indonesia (ST-INTEN-Bandung). Junindar mendapatkan Award Microsoft MVP VB pertanggal 1 oktober 2009 hingga saat ini. Senang mengutak-atik computer yang berkaitan dengan bahasa pemrograman. Keahlian, sedikit mengerti beberapa bahasa pemrograman seperti : VB.Net, C#, SharePoint, ASP.NET, VBA. Reporting: Crystal Report dan Report Builder. Database: MS Access, MY SQL dan SQL Server. Simulation / Modeling Packages: Visio Enterprise, Rational Rose dan Power Designer. Dan senang bermain gitar, karena untuk bisa menjadi pemain gitar dan seorang programmer sama-sama membutuhkan seni. Pada saat ini bekerja di salah satu Perusahaan Consulting dan Project Management di Malaysia sebagai Senior Consultant. Memiliki beberapa sertifikasi dari Microsoft yaitu Microsoft Certified Professional Developer (MCPD – SharePoint 2010), MOS (Microsoft Office Specialist) dan MCT (Microsoft Certified Trainer) Mempunyai moto hidup: “**Jauh lebih baik menjadi Orang Bodoh yang giat belajar, dari pada orang Pintar yang tidak pernah mengimplementasikan ilmunya**”.