

**Workshop Mobile Application**  
**Membuat Fragment Dengan Navigation**



**GOLONGAN A**

NAMA : RIFQI FEBRI VANDHIKA

NIM : E41200064

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**JURUSAN TEKNOLOGI INFORMASI**  
**POLITEKNIK NEGERI JEMBER**

**2021**

## DAFTAR ISI

<b>DAFTAR ISI</b> .....	2
<b>DAFTAR GAMBAR</b> .....	3
<b>BAB I PENDAHULUAN</b> .....	4
<b>1.1. Dasar Teori</b> .....	4
<b>1.2. Tujuan dan Manfaat</b> .....	4
<b>1.2.1 Mahasiswa mampu memahami konsep fragment.</b> .....	4
<b>1.2.2 Mahasiswa menerapkan fragment menggunakan navigasi</b> .....	4
<b>BAB II HASIL DAN PEMBAHASAN</b> .....	6
<b>2.1. ActivityMain.xml</b> .....	6
<b>2.2. Fragment.xml</b> .....	8
<b>2.3. Gambar_menu.xml</b> .....	10
<b>2.4. MainActivity.java</b> .....	12
<b>2.5. Fragment.java</b> .....	14
<b>2.6. Nav_header.xml</b> .....	15
<b>2.7. Strings, Styles, Styles(v21)</b> .....	16
<b>BAB III PENUTUP</b> .....	17
<b>3.1. Kesimpulan</b> .....	17
<b>DAFTAR PUSTAKA</b> .....	18

## DAFTAR GAMBAR

Gambar 1 ActivityMain 1.0 .....	6
Gambar 2 ActivityMain 1.1 .....	6
Gambar 3 Output ActivityMain .....	7
Gambar 4 FragmentTulisPesan .....	8
Gambar 5 Fragment Permainan .....	8
Gambar 6 Fragment Percakapan .....	8
Gambar 7 Output Fragment Permainan .....	9
Gambar 8 gambar_menu .....	10
Gambar 9 Outpu Gambar Menu.....	11
Gambar 10 MainActivity.java 1.0.....	12
Gambar 11 MainActivity.java 1.1.....	12
Gambar 12 Fragment TulisPesan .....	14
Gambar 13 Fragment Percakapan .....	14
Gambar 14 Fragment Permainan .....	14
Gambar 15 Nav_header.....	15
Gambar 16 Strings .....	16
Gambar 17 Styles .....	16
Gambar 18 Styles(v21) .....	16

## **BAB I PENDAHULUAN**

### **1.1. Dasar Teori**

Penggunaan Komponen View dan Logic Berulang Kali - Fragment dapat dipakai untuk menampilkan data atau melakukan event tertentu di beberapa activity berbeda. Dukungan Untuk Tablet - Di beberapa aplikasi, versi tablet dari sebuah activity memberikan layout yang berbeda dari versi handphone yang juga berbeda dari versi TV. Fragment memungkinkan activity untuk menggunakan fragment dalam membuat antarmuka sesuai dengan perangkat yang membukanya.

Orientasi Layar - Seringkali dalam aplikasi, versi portrait dan landscape sebuah aplikasi memiliki layout yang berbeda. Fragment memungkinkan kedua orientasi tersebut untuk menggunakan tampilan yang berbeda menggunakan elemen yang sama.

Fragment digunakan agar komponen tampilan aplikasi menjadi fleksibel dan dapat digunakan kembali (reusable). Satu activity bisa memiliki lebih dari satu fragment. Tidak seperti activity, fragment tidak perlu didaftarkan ke dalam file AndroidManifest.xml. Satu kelas Java dinyatakan sebagai sebuah fragment ketika kelas tersebut meng- extends (inherit) kelas fragment. Melalui Android Support Library, fragment bersifat kompatibel sampai Android api level 10 Gingerbread. Analogi yang mendekati fragment pada platform lain adalah penggunaan komponen iframe pada aplikasi berbasis web.

### **1.2. Tujuan dan Manfaat**

#### **1.2.1 Mahasiswa mampu memahami konsep fragment.**

Dengan adanya praktikum ini mahasiswa dapat memahami apa itu fragment dan juga konsep – konsep dari fragment dalam android studio.

#### **1.2.2 Mahasiswa menerapkan fragment menggunakan navigasi**

Jadi mahasiswa bisa menerapkan pada android studio untuk membuat fragment menggunakan navigasi



## BAB II HASIL DAN PEMBAHASAN

### 2.1. ActivityMain.xml

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.drawerlayout.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:id="@+id/gambar_layout"
8     android:fitsSystemWindows="true"
9     tools:context=".MainActivity"
10    tools:openDrawer="start">
11    // tools:openDrawer="start" supaya nav_head dan gambar menu yang sudah dipanggil dalam
12    // com.google.android.material.navigation.NavigationView dapat berjalan
13
14    // selanjutnya LinearLayout untuk layoutnya
15    <LinearLayout
16        android:layout_width="match_parent"
17        android:layout_height="match_parent"
18        android:orientation="vertical">
19        // menggunakan toolbar, theme dan popupnya menggunakan style yang sudah dibuat dalam folder styles
20        <androidx.appcompat.widget.Toolbar
21            android:layout_width="match_parent"
22            android:layout_height="?attr/actionBarSize"
23            android:background="@color/design_default_color_primary"
24            android:id="@+id/toolbar"
25            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
26            app:popupTheme="@style/ThemeOverlay.AppCompat.Light"
27        />
28        // frame untuk kandungan fragmentnya
29        <FrameLayout
30            android:id="@+id/fragment_contain"
31            android:layout_width="match_parent"
32            android:layout_height="match_parent"/>
```

*Gambar 1 ActivityMain 1.0*

```
35 //navigasi dan menggunakan header layout dari nav_header dan menu dari gambar_menu
36 <com.google.android.material.navigation.NavigationView
37     android:layout_width="wrap_content"
38     android:layout_height="match_parent"
39     android:layout_gravity="start"
40     android:id="@+id/nav_view"
41     app:headerLayout="@layout/nav_header"
42     app:menu="@menu/gambar_menu"/>
43 </androidx.drawerlayout.widget.DrawerLayout>
```

*Gambar 2 ActivityMain 1.1*

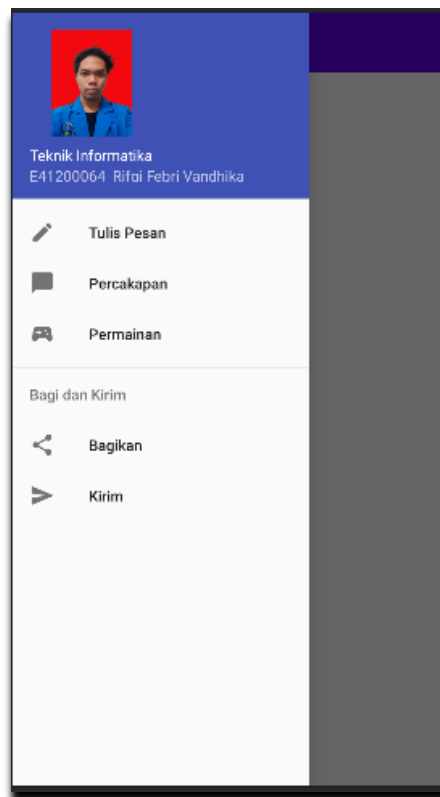
#### Penjelasan dan Pembahasan :

- Dalam gambar 1 dan 2 terdapat sourcecode ActivityMain yang merupakan dasar dari layout – layout lainnya atau bisa dibilang layout inti, jadi dalam layout ini saya memanggil dari xml lain yaitu nav\_header.xml dan gambar\_menu.xml nantinya digunakan untuk headerLayout menggunakan nav\_header.xml dan app:menu

menggunakan gambar\_menu.xml untuk memanggil di dalam navigation view karena ini digunakan untuk navigasi.

- DrawerLayout bertindak sebagai wadah tingkat atas untuk konten jendela yang memungkinkan tampilan "layout" interaktif ditarik keluar dari satu atau kedua tepi vertikal jendela.
- tools:openDrawer="start" digunakan untuk membuka atau memulai memunculkan navigasi yang sudah berhasil dimasukkan dalam navigationview yaitu ada nav\_header dan gambar\_menu.
- Selanjutnya menggunakan LinearLayout untuk toolbar dan framelayout. Toolbar sendiri adalah sekumpulan tombol navigasi dalam bentuk icon/tulisan yang terdapat pada bagian atas, bawah atau samping tampilan software, untuk memudahkan pengguna dalam menjalankan fitur – fitur yang ada pada sebuah software.

### Output ActivityMain



*Gambar 3 Output ActivityMain*

## 2.2. Fragment.xml

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:background="@android:color/holo_red_light">
6
7     //membuat layout untuk tulis pesan
8     <TextView
9         android:layout_width="wrap_content"
10        android:layout_height="wrap_content"
11        android:text="Tulis Pesan"
12        android:textSize="30sp"
13        android:layout_centerInParent="true"/>
14 </RelativeLayout>
```

*Gambar 4 FragmentTulisPesan*

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:background="@android:color/holo_green_light">
6
7     //membuat layout untuk permainan
8     <TextView
9         android:layout_width="wrap_content"
10        android:layout_height="wrap_content"
11        android:text="Permainan"
12        android:textSize="30sp"
13        android:layout_centerInParent="true"/>
14 </RelativeLayout>
```

*Gambar 5 Fragment Permainan*

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:background="@android:color/holo_purple">
6
7     //membuat layout untuk percakapan
8     <TextView
9         android:layout_width="wrap_content"
10        android:layout_height="wrap_content"
11        android:text="Percakapan"
12        android:textSize="30sp"
13        android:layout_centerInParent="true"/>
14 </RelativeLayout>
```

*Gambar 6 Fragment Percakapan*



### **Penjelasan dan Pembahasan :**

- Disini terdapat fragment\_tulispesan.xml, fragment\_permainan.xml, dan fragment\_percakapan.xml fragment disini adalah untuk layout ketika user menekan atau klik navigasi yang tersedia tetapi disini navigasi yang saya beri fragment hanya navigasi tulis pesan, percakapan dan permainan.
- Disini saya setting menggunakan warna yang berbeda beda dan untuk text yang tampil pada layout juga berbeda.
- Fragment merupakan salah satu komponen pada Android Studio dengan fungsi yang hampir sama seperti activity tetapi memiliki “ lifecycle ” yang berbeda. Fragment merupakan bagian dari sebuah activity yang mana sebuah fragment tidak akan ada bila tidak ada sebuah activity karena fragment membutuhkan akses dari activity untuk dapat dijalankan.

### **Output Fragment**



*Gambar 7 Output Fragment Permainan*

### 2.3. Gambar\_menu.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <menu xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      tools:showIn="navigation_view">
5      <group android:checkableBehavior="single">
6          // membuat isi dari navigasi
7          <item
8              android:id="@+id/nav_tulispesan"
9              android:icon="@drawable/ic_tulispesan"
10             android:title="Tulis Pesan" />
11          <item
12              android:id="@+id/nav_percakapan"
13              android:icon="@drawable/ic_percakapan"
14              android:title="Percakapan" />
15          <item
16              android:id="@+id/nav_permainan"
17              android:icon="@drawable/ic_permainan"
18              android:title="Permainan" />
19      </group>
20
21      <item android:title="Bagi dan Kirim">
22          //membuat isi dari navigasi pada bagian Bagi dan Kirim
23          <menu>
24              <item
25                  android:id="@+id/nav_bagikan"
26                  android:icon="@drawable/ic_bagikan"
27                  android:title="Bagikan" />
28              <item
29                  android:id="@+id/nav_kirim"
30                  android:icon="@drawable/ic_kirim"
31                  android:title="Kirim" />
32          </menu>
33      </item>
34  </menu>
```

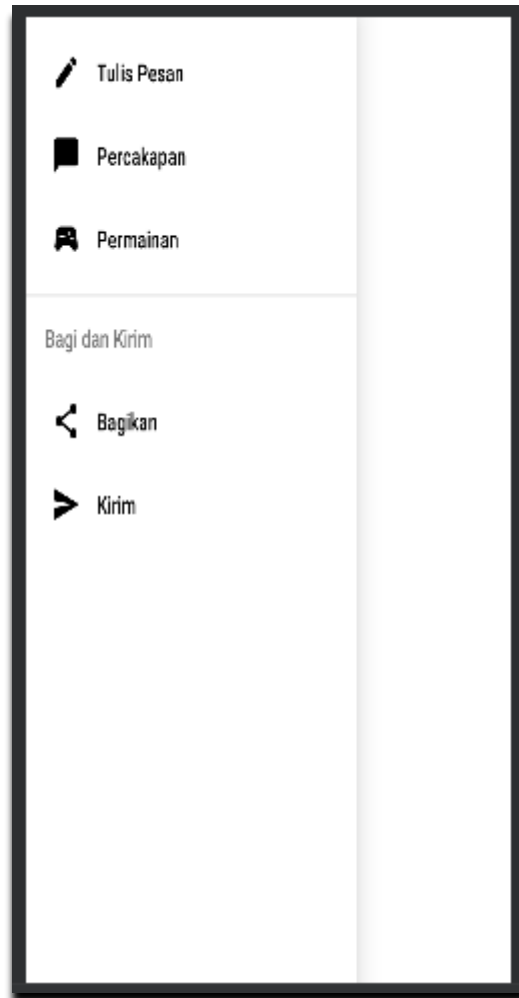
Gambar 8 gambar\_menu

#### Penjelasan dan Pembahasan :

- Pada gambar 8 terdapat source code untuk menu atau bisa dibilang untuk isi dari navigasi yang saya buat disini saya memberi item – item pada navigasi yang atas dengan logo dan tulisan, yaitu dengan tulispesan, percakapan, permainan dan untuk navigasi bawahnya ada untuk navigasi bagi dan kirim tetapi tidak saya beri fragment untuk kedua navigasi dibawah ini, hanya saya beri attention atau

alert tulisan jika user berhasil klik atau pencet item bagi atau kirim pada navigasi ini.

### Output Gambar Menu



*Gambar 9 Outpu Gambar Menu*

## 2.4. MainActivity.java

```
1 package com.example.fragment_bottomnavigation;
2
3 import ...
4
15 public class MainActivity extends AppCompatActivity implements NavigationView.OnNavigationItemSelectedListener {
16     private DrawerLayout drawer;
17     // program dengan bahasa java
18     @Override
19     protected void onCreate(Bundle savedInstanceState) {
20         // memanggil apa yang sudah dibuat dalam acvitivity_main.xml
21         super.onCreate(savedInstanceState);
22         setContentView(R.layout.activity_main);
23         // mengambil toolbar
24         Toolbar toolbar = findViewById(R.id.toolbar);
25         setSupportActionBar(toolbar);
26         // memanggil gambar layout yang sudah dibuat menggunakan navigation view
27         drawer = findViewById(R.id.gambar_layout);
28         NavigationView navigationView = findViewById(R.id.nav_view);
29         navigationView.setNavigationItemSelectedListener(this);
30         // membuka dan menutup navigasi dengan toggle
31         ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this, drawer, toolbar,
32             "Open Navigation Drawer", "Close Navigation Drawer");
33         drawer.addDrawerListener(toggle);
34         toggle.syncState();
35     }
36 }
```

*Gambar 10 MainActivity.java 1.0*

```
38 @Override
39 public boolean onNavigationItemSelected(@NonNull MenuItem item) {
40     // memberi aksi pilih untuk item pada navigasi
41     switch (item.getItemId()) {
42         case R.id.nav_tulispesan:
43             getSupportFragmentManager().beginTransaction().replace(R.id.fragment_contain,
44                 new Fragmenttulispesan()).commit();
45             break;
46         case R.id.nav_percakapan:
47             getSupportFragmentManager().beginTransaction().replace(R.id.fragment_contain,
48                 new Fragmentpercakapan()).commit();
49             break;
50         case R.id.nav_permainan:
51             getSupportFragmentManager().beginTransaction().replace(R.id.fragment_contain,
52                 new Fragmentpermainan()).commit();
53             break;
54         case R.id.nav_kirim:
55             Toast.makeText(this, "Kirim", Toast.LENGTH_SHORT).show();
56             break;
57         case R.id.nav_bagikan:
58             Toast.makeText(this, "Bagikan", Toast.LENGTH_SHORT).show();
59             break;
60     }
61     return false;
62 }
63 @Override
64 public void onBackPressed() {
65     if (drawer.isDrawerOpen(GravityCompat.START)) {
66         drawer.closeDrawer(GravityCompat.START);
67     } else {
68         super.onBackPressed();
69     }
70 }
71 }
```

*Gambar 11 MainActivity.java 1.1*

### **Penjelasan dan Pembahasan :**

- Disinilah tempat memberi aksi pada layout – layout yang sudah berhasil dibuat tadi yaitu dengan MainActivity.java pertama kita memanggil dulu apa yang sudah saya berhasil buat pada layout Activity\_main.xml yaitu dengan cara menggunakan sourcecode `super.onCreate(savedInstanceState);`  
`setContentView(R.layout.activity_main);`
- Setelah itu memanggil toolbar atau jendela utama pada app jadi `Toolbar toolbar = findViewById(R.id.toolbar);` adalah untuk menemukan atau memanggil toolbar pada app jadi ketika dijalankan app ini memiliki toolbar.
- Selanjutnya mengambil draw pada xml yang sudah berhasil dibuat tadi yang menggunakan id `@gambar_layout`, `gambar_layout` yaitu terletak pada `activity_main.xml`.
- Setelah itu ada toggle yang digunakan untuk membuka dan menutup navigasi disana tertulis `navigation_drawer_open` itu berasal dari string yang sudah saya buat pada `strings.xml`.
- `public boolean onNavigationItemSelected` digunakan untuk memilih item – item yang ada dalam navigasi disini menggunakan switch case dan break digunakan untuk memberhentikan jadi tidak terus – terusan berjalan jadi setelah klik bisa memilih item pada navigasi lain juga.

## 2.5. Fragment.java

```
1 package com.example.fragment_bottomnavigation;
2
3 import ...
4
11
12 public class Fragmenttulispesan extends Fragment {
13     // membuat class java fragmet tulispesan
14     @Nullable
15     @Override
16     public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
17         return inflater.inflate(R.layout.fragment_tulispesan, container, attachToRoot: false);
18     }
19 }
20
```

*Gambar 12 Fragment TulisPesan*

```
1 package com.example.fragment_bottomnavigation;
2
3 import ...
4
11
12 public class Fragmentpercakapan extends Fragment {
13     // membuat class java fragmet percakapan
14     @Nullable
15     @Override
16     public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
17         return inflater.inflate(R.layout.fragment_percakapan, container, attachToRoot: false);
18     }
19 }

```

*Gambar 13 Fragment Percakapan*

```
1 package com.example.fragment_bottomnavigation;
2
3 import ...
4
11
12 public class Fragmentpermainan extends Fragment {
13     // membuat class java fragmet permainan
14     @Nullable
15     @Override
16     public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
17         return inflater.inflate(R.layout.fragment_permainan, container, attachToRoot: false);
18     }
19 }

```

*Gambar 14 Fragment Permainan*

### Penjelasan dan Pembahasan :

- Setelah membuat fragment.xml harus membuat fragment untuk java juga karena jika tidak membuat .java juga maka tidak bisa dijalankan hanya cuma layout saja..
- Jadi disini java digunakan untuk memberi aksi – aksi atau menjalankan dari layout menggunakan java selain java pada android studio juga bisa menggunakan kotlin.

## 2.6. Nav\_header.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="match_parent"
4      android:layout_height="176dp"
5      android:background="#3F51B5"
6      android:gravity="bottom"
7      android:orientation="vertical"
8      android:padding="16dp"
9      android:theme="@style/ThemeOverlay.AppCompat.Dark">
10     // membuat header untuk diatasnya navigasi
11     <ImageView
12         android:layout_width="117dp"
13         android:layout_height="101dp"
14         android:src="@drawable/saya" />
15     //membuat text keterangan
16     <TextView
17         android:layout_width="wrap_content"
18         android:layout_height="wrap_content"
19         android:paddingTop="8dp"
20         android:text="Teknik Informatika"
21         android:fontFamily="sans-serif"
22         android:textAppearance="@style/TextAppearance.AppCompat.Body1" />
23     <TextView
24         android:layout_width="wrap_content"
25         android:layout_height="wrap_content"
26         android:fontFamily="sans-serif"
27         android:text="E41200064_Rifqi Febri Vandhika" />
28 </LinearLayout>
```

*Gambar 15 Nav\_header*

### Penjelasan dan Pembahasan :

- Nav\_header.xml nama class yang saya gunakan untuk membuat heading pada navigasi saya supaya tidak terlihat polos.
- Disini saya menggunakan theme pada style yang sudah saya buat dengan saya namai ThemeOverlay.AppCompat.Dark maka pertama untuk memanggil nya dengan @style/.
- Pada header saya memasukkan photo dan text jadi disini menggunakan ImageView dan TextView.

## 2.7. Strings, Styles, Styles(v21)

```
1 <resources>
2   <string name="app_name">E41200064_fragmentbottomnavigation</string>
3   // membuat string name untuk membuka navigation dan menutup navigation
4   <string name="navigation_drawer_open">Open Navigation Drawer</string>
5   <string name="navigation_drawer_close">Close Navigation Drawer</string>
6 </resources>
```

Gambar 16 Strings

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3   // mengatur styles untuk membuat setting color primary primarydark dan accent
4   // memasukkan warna untuk mereka
5   <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
6     <item name="colorPrimary">@color/design_default_color_primary</item>
7     <item name="colorPrimaryDark">@color/design_default_color_primary_dark</item>
8     <item name="colorAccent">#E657C7</item>
9   </style>
10  //memberi nama tema dan keterangan false true untuk aksibar dan window tanpa title
11  <style name="AppTheme.NoActionBar">
12    <item name="windowActionBar">false</item>
13    <item name="windowNoTitle">true</item>
14  </style>
15 </resources>
```

Gambar 17 Styles

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3   //memberi nama tema dan keterangan false true untuk aksibar dan window tanpa title
4   //serta memberi warna transparan pada statusbar
5   <style name="AppTheme.NoActionBar">
6     <item name="windowActionBar">false</item>
7     <item name="windowNoTitle">true</item>
8     <item name="android:statusBarColor">@android:color/transparent</item>
9   </style>
10 </resources>
```

Gambar 18 Styles(v21)



## **BAB III PENUTUP**

### **3.1. Kesimpulan**

Navigation Drawer View adalah salah satu komponen navigasi menu ,atau biasa disebut sliding menu yang berfungsi untuk membungkus dan menavigasikan sebuah konten di dalam Activity atau Fragment, dari Aplikasi. Navigation drawer sekarang didukung oleh Material Design Library. yang termasuk ke dalam library Appcompat (v21).Navigation Drawer View sering digunakan pada Aplikasi- Aplikasi yang sudah populer seperti BBM. Untuk menggunakannya user(pengguna ) hanya perlu menggeser jempol atau jari kalian kekanan maka Navigation Drawer View akan tampil.

Mahasiswa memahami konsep dasar fragment dengan menerapkan contoh pada aplikasi android yang sering digunakan sehari-hari. Jadi mahasiswa berhasil mengimplementasikan fragment pada project individu serta memahami konsep implementasi fragment.

## **DAFTAR PUSTAKA**

Vandhika, R. F. (2021). Workshop Mobile Application. *Membuat Fragment Dengan Navigation*.