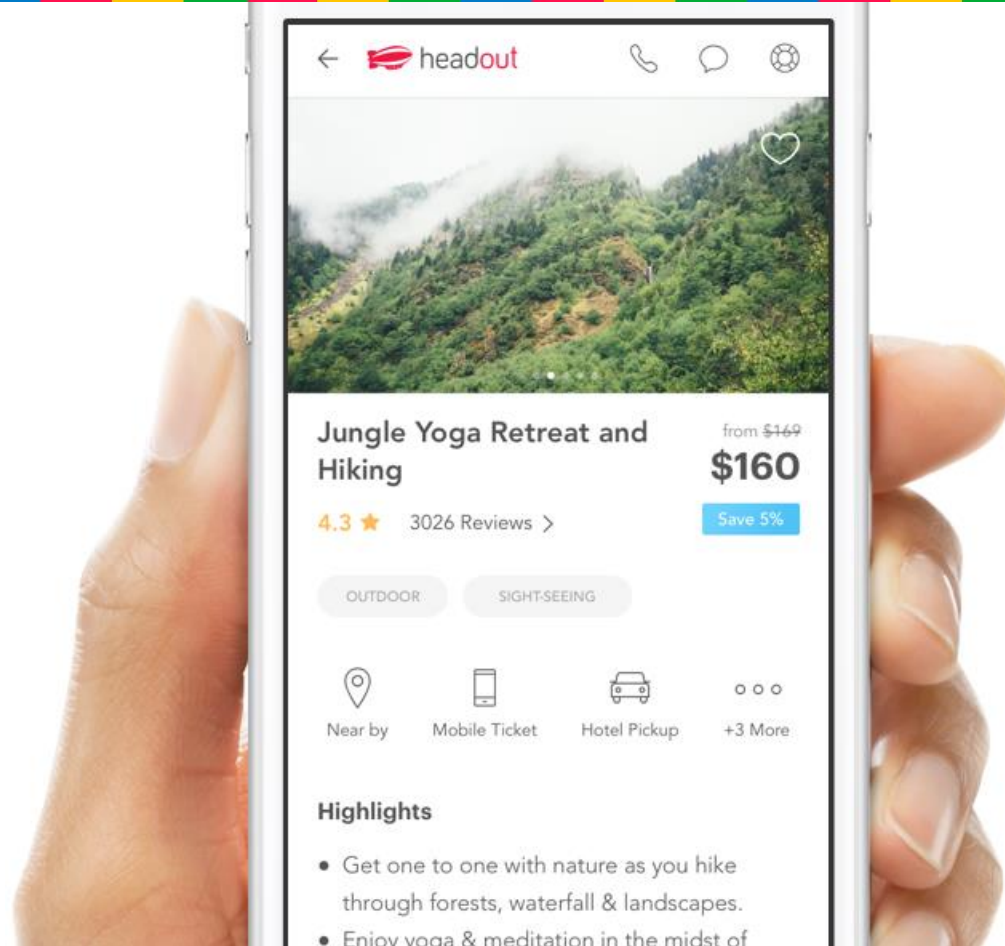


# Modul 2: Application UI dan UX

- Action Bar & Nav Drawer
- Localization

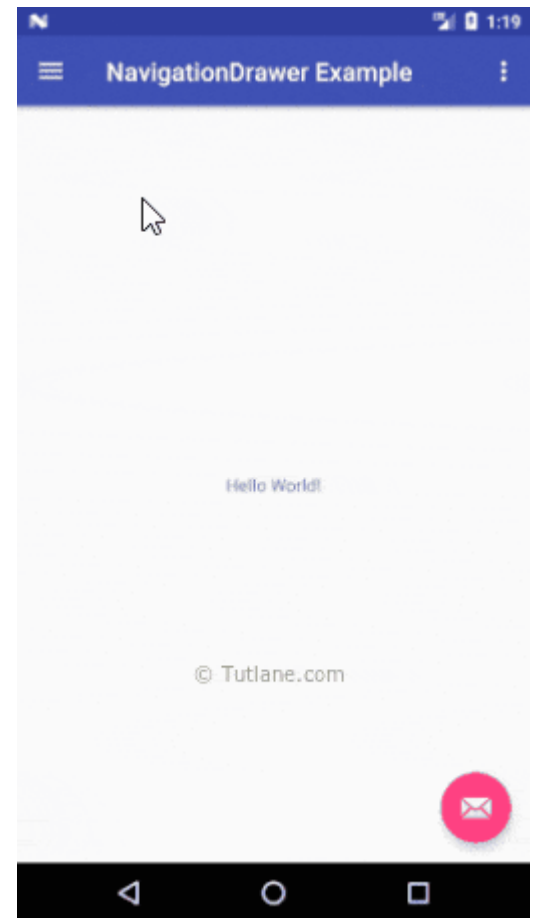


Jurusan Ilmu Komputer  
Fakultas MIPA  
Universitas Lampung

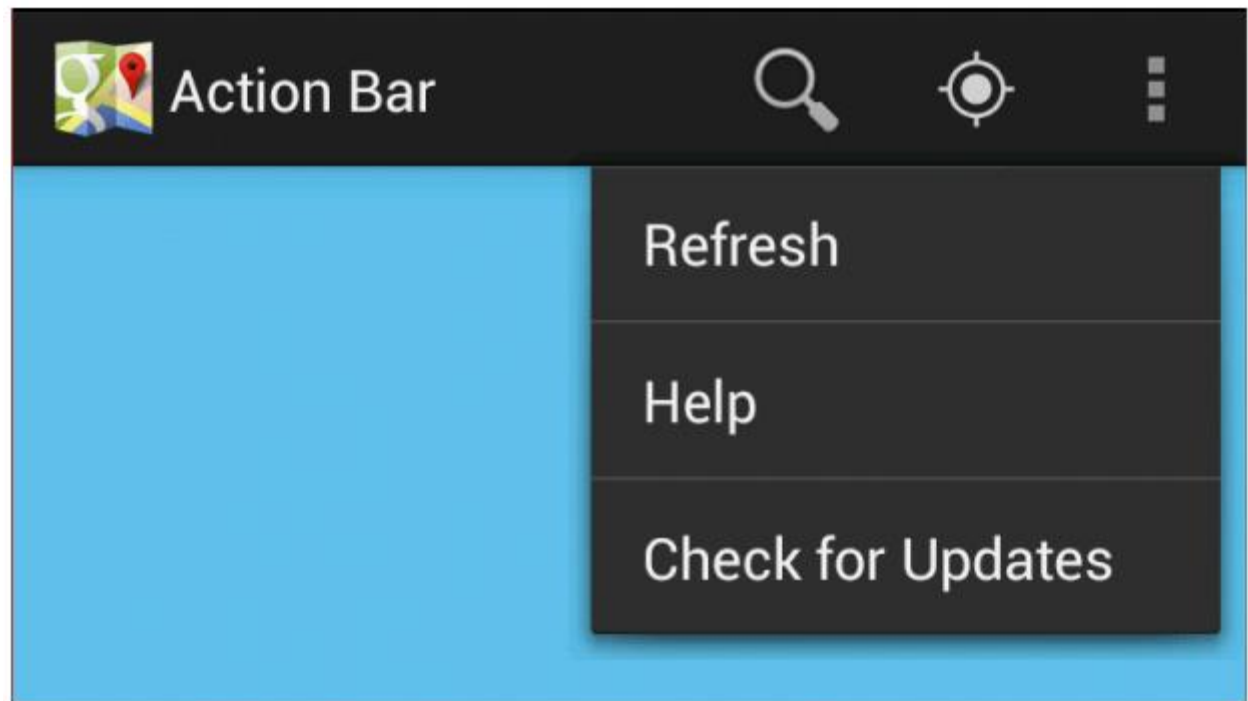


Google Developers  
Authorized Training Partner

# Action Bar & Navigation Drawer



# Action Bar





# App Bar

- Appbar atau action bar adalah salah satu komponen penting di dalam struktur sebuah aplikasi Android.
- Appbar merupakan rancangan yang dianjurkan dalam membuat aplikasi.
- Bila kita menggunakannya dengan baik, maka kita dapat mempertahankan konsistensi aplikasi.
- Tentunya ini akan berimbas positif pada *user experience*.





# App Bar

Beberapa fungsi penting dari appbar adalah :

- Memberikan ruangan kepada identitas app dan juga mengindikasikan lokasi pengguna pada saat menjalankan aplikasi.
- Akses ke beberapa *action* penting dengan lebih mudah seperti *action* pencarian.
- Sebagai *navigation* dan *switch* antar view (tampilan).



# App Bar



AppBar atau actionBar dibedakan menjadi 4 berdasarkan fungsinya masing-masing:

## 1. **AppIcon**

Ikon menandakan identitas dari sebuah aplikasi. Dapat juga menandakan lokasi pengguna. Pada actionBar ini terdapat tombol *up* untuk kembali ke *parent* jika ia tidak berada di *top-level screen*.



# App Bar



## 2. ViewControl

Ketika aplikasi menampilkan data dalam beberapa macam view, maka view control dapat digunakan sebagai alat kontrol dari *switch* view.

*Switch* antar view biasanya menggunakan spinner atau tabs.

Jika aplikasi tidak menampilkan beberapa views, maka view control dapat digunakan sebagai *title* (judul) dari sebuah aplikasi.



# App Bar



## 3. ActionButtons

Tempat untuk meletakkan beberapa *action* penting dalam sebuah aplikasi.

## 4. Overflow

Tempat untuk meletakkan *action* yang jarang digunakan.







# Toolbar

- Fitur terbaru material design yang berhubungan dengan appbar adalah toolbar.
- Toolbar merupakan pengganti dari action bar yang mempunyai kontrol dan fleksibilitas lebih.
- Developer dapat dengan mudah mengubah tampilan toolbar.
- Toolbar juga mendukung perangkat Android lama.
- Untuk mengubah action bar menjadi toolbar, Anda dapat menggunakan metode `setSupportActionBar`.





# Toolbar

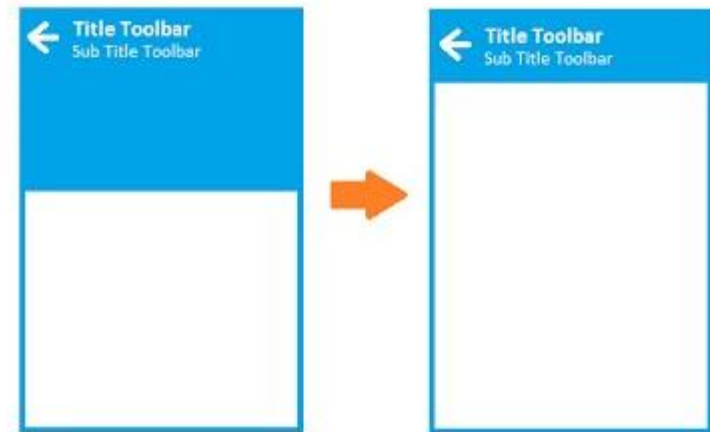
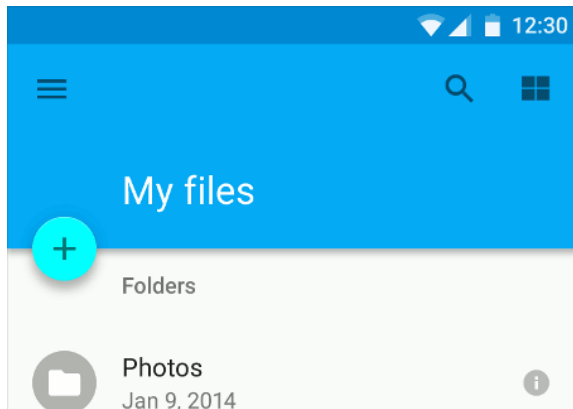
Beberapa fitur yang didukung oleh toolbar yaitu :

- **Navigation Button** : Dapat berupa tombol *up*, *navigation menu toggle*, *close*, *collapse*, *done*, atau *glyph* lainnya.
- **Logo Aplikasi** : Dapat dimodifikasi tinggi dan panjangnya.
- **Title & Subtitle** : *Title* bisa menunjukkan di mana aplikasi sedang berada. *Subtitle* merupakan tambahan informasi tentang konten yang sedang diakses.



# Toolbar

- **More Custom Views** : Toolbar dapat ditambah dengan beberapa *child views* di dalamnya.
- **Action Menu** : Akan berada pada sebelah kanan toolbar sesuai dengan tingkatannya yaitu *frequent*, *important* atau *typical*.



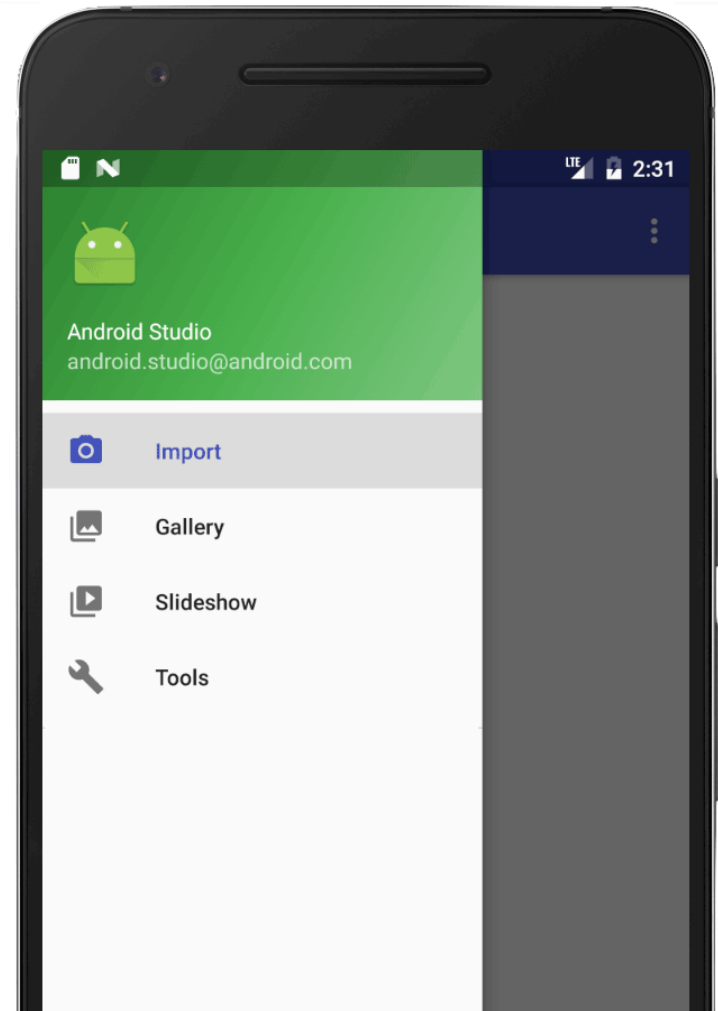


# Best Practice

- Apakah kita harus meninggalkan action bar bawaan dari appbar dan menggantinya dengan toolbar ?
- Jawabannya adalah tergantung dari kebutuhan Anda sebagai developer.
- Jika memang fitur dari aplikasi yang Anda buat tidak terpenuhi ketika menggunakan actionbar, maka toolbar adalah pilhan terbaik.



# Navigation Drawer





# Navigation Drawer

- Navigation drawer adalah sebuah panel yang berada di sisi kiri layar perangkat Android seperti sliding menu.
- Navigation drawer juga akan muncul ketika pengguna menekan ikon *hamburger* yang berada di sudut kiri panel action bar.





# Navigation Drawer

- Navigation drawer berisi menu-menu utama dalam sebuah aplikasi.
- Pada prinsipnya penggunaan navigation drawer sesuai dengan pendekatan yang terdapat pada material design, di mana navigasi utama aplikasi akan diletakkan.

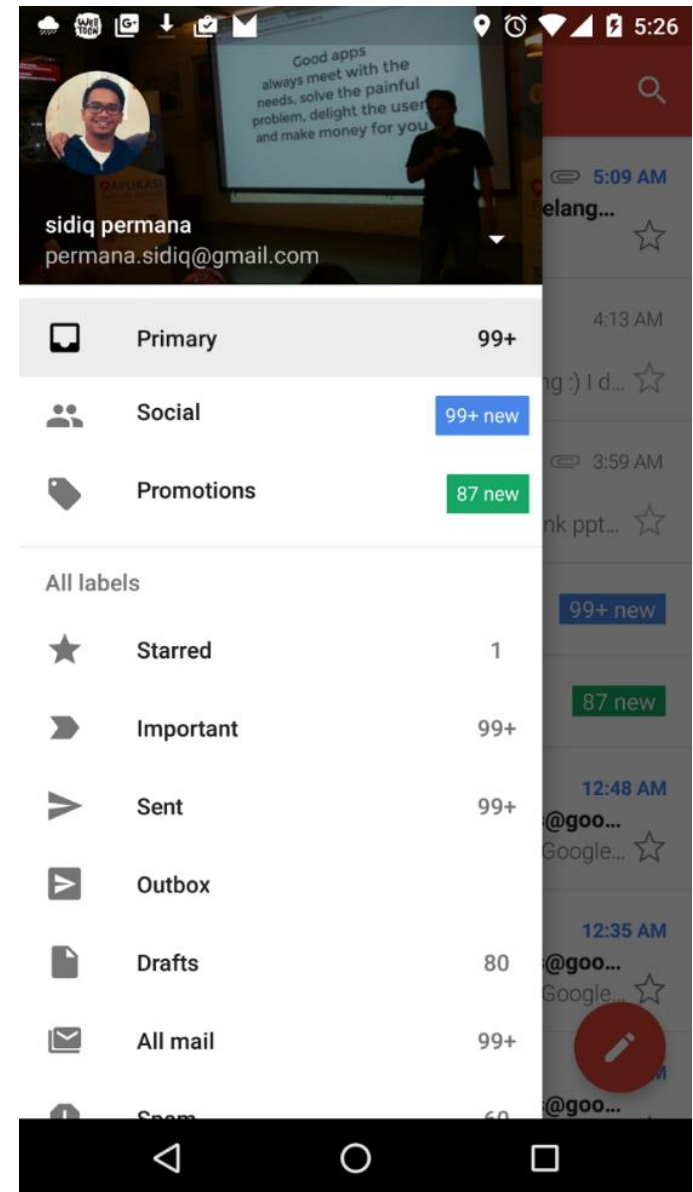




# Navigation Drawer

Navigation drawer digunakan bila:

- Aplikasi Anda memiliki *top level views*, dalam artian beberapa menu utama yang menjadi *parent* dari beberapa *submenu* di dalamnya.



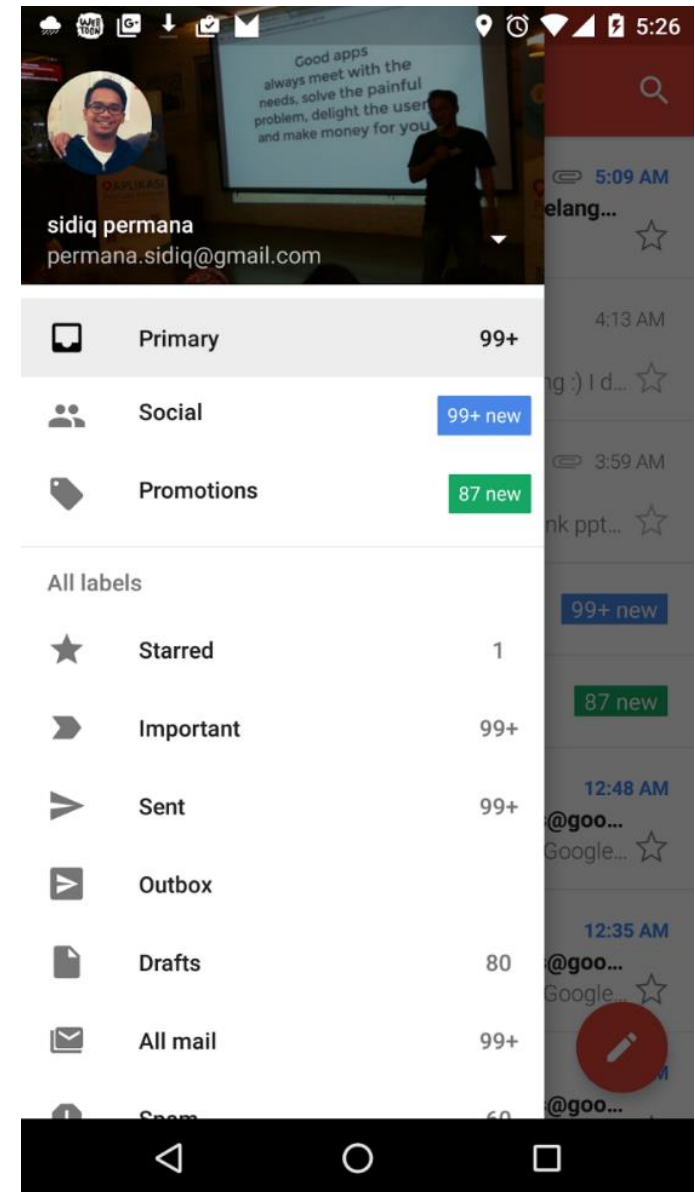




# Navigation Drawer

Navigation drawer digunakan bila:

- Jika kamu ingin memberikan kecepatan kepada pengguna untuk mengakses menu yang tidak terhubung secara langsung.

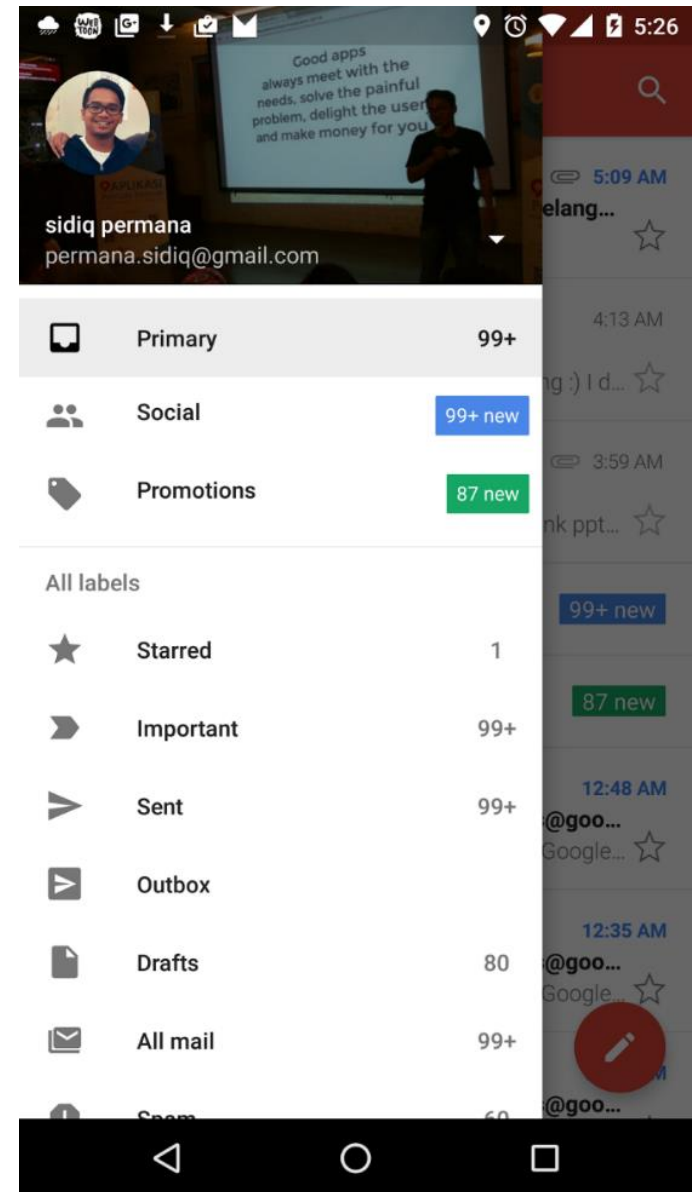




# Navigation Drawer

Navigation drawer digunakan bila:

- Bila Anda ingin kembali ke halaman utama dengan cepat.

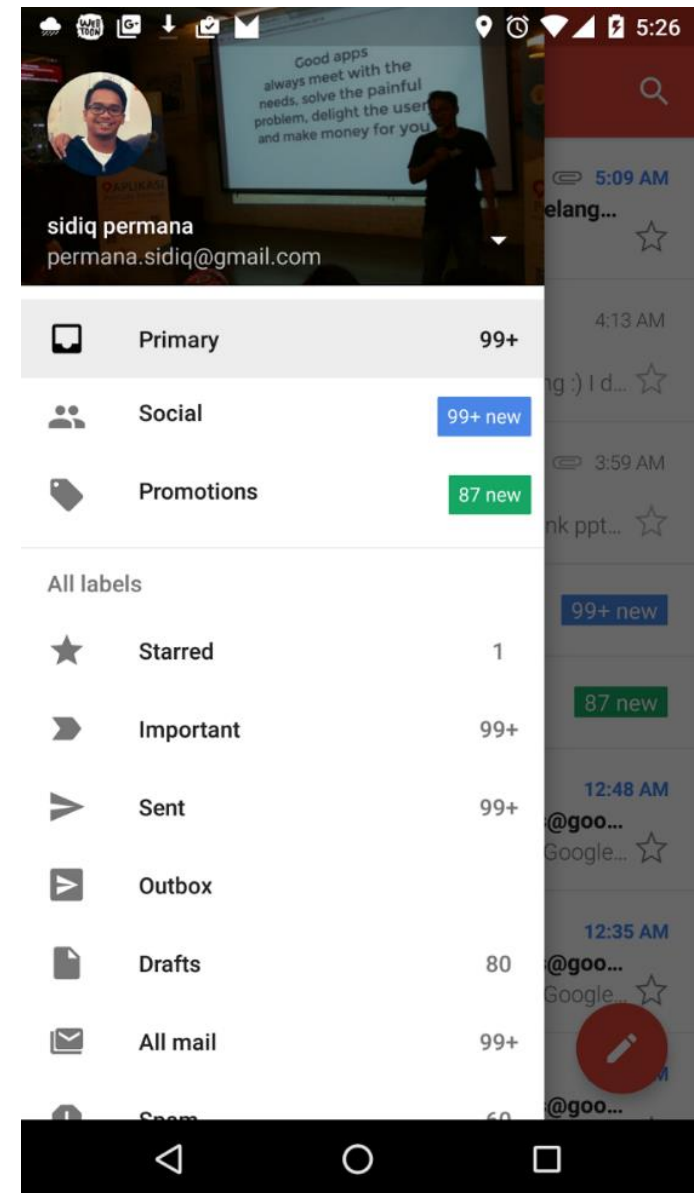




# Navigation Drawer

Navigation drawer digunakan bila:

- Jika Anda ingin menempatkan menu yang jarang diakses. Contohnya, *term and conditions*, *about app*, legal dan lain sebagainya.





*CodeLab*

# Localization





# Introduction

- Android merupakan sebuah platform dengan jumlah basis pengguna terbesar pada pangsa pasar perangkat *smartphone* saat ini.
- Jumlah ini terus meningkat.
- Perangkat Android pun semakin hari kian terjangkau.
- Tak hanya itu, di banyak negara kualitas jaringan internet pun terus membaik.





# Introduction

- Kondisi di atas adalah peluang bagi pengembang aplikasi untuk menciptakan aplikasi kelas dunia.
- Agar aplikasinya dapat diterima di beragam tempat, maka pengembang perlu menerapkan localization.





# Localiztion (Multi Language)

- Localization adalah upaya melokalkan konten dan *resource* (sumber daya) pada sebuah aplikasi agar dapat beradaptasi dengan kebutuhan budaya dan perkembangan di banyak negara yang menjadi target pasar aplikasi.
- Meskipun secara *default* aplikasi berjalan menggunakan bahasa Inggris, namun localization dapat meningkatkan pengalaman pengguna (UX) dalam mengoperasikan aplikasi.







# Localiztion (Multi Language)

- Pada dasarnya semua *resource* yang dimiliki oleh aplikasi dapat dilokalkan.
- Mulai dari aset gambar atau ikon, teks, audio, mata uang, dan dimensi.
- Di antara semua tipe tersebut, teks merupakan yang paling krusial.
- Jika aplikasi kita mendukung banyak bahasa, lebih banyak pengguna dari beragam negara di dunia yang akan memahami aplikasi yang kita buat.





# Android Localization of Strings

- Generally, android consider **English** is a default language and it loads the string resources from **/res/values/strings.xml**.
- In case, if we want to add a support for other languages, we need to create a **values** folder by appending the Hyphen and ISO language code.





# Android Localization of Strings

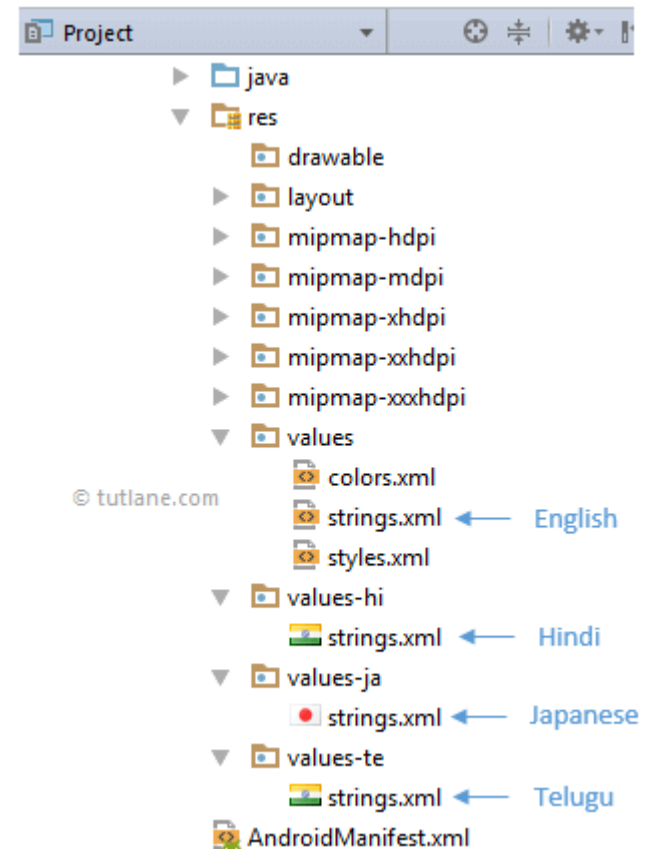
- For example, if we want to add support for **Japanese**, then we need to create a **values** folder named **values-ja** under **res** folder and add a **strings.xml** file in it with all the strings that needs to translate into **Japanese** Language.





# Android Localization of Strings

- Following is the example of defining a multiple **values** folder in **/res** directory to support localization of strings in multiple languages such as Telugu, Hindi and Japanese.





# Android Localization of Strings

- Once we create a required files and change the device language through **Settings à Language & Input a Select Language (Japanese)**, android OS will check for the appropriate language resources available in the app.





# Android Localization of Strings

- In case, if the app supports a selected language, then android will look for the string resources in **values-(ISO language code)** folder of the project.
- For example, the selected language is **Japanese**, then it will load the string values from **values-ja/strings.xml** file.





# Android Localization Example

## values/strings.xml

```
<resources>  
  <string name="app_name">Localization Example</string>  
  <string name="hello">Hello World</string>  
  <string name="tutlane">Tutlane</string>  
  <string name="android">Android Tutorial</string>  
  <string name="reigster">Register</string>  
</resources>
```



# Android Localization Example

## values-hi/strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="app_name">स्थानीयकरणउदाहरण</string>
  <string name="hello">नमस्तेदुनिया</string>
  <string name="tutlane">ट्यूटोरियलगली</string>
  <string name="android">एंड्रॉयडट्यूटोरियल</string>
  <string name="reigster">रजिस्टर</string>
</resources>
```







# Android Localization Example

## values-te/strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="app_name">స్థానికీకరణఉదాహరణ</string>
  <string name="hello">హలోవరల్డ్</string>
  <string name="tutlane">ట్యుటోరియల్స్</string>
  <string name="android">ఆండ్రాయిడ్ట్యుటోరియల్</string>
  <string name="reigster">నమోదు</string>
</resources>
```





# Android Localization Example

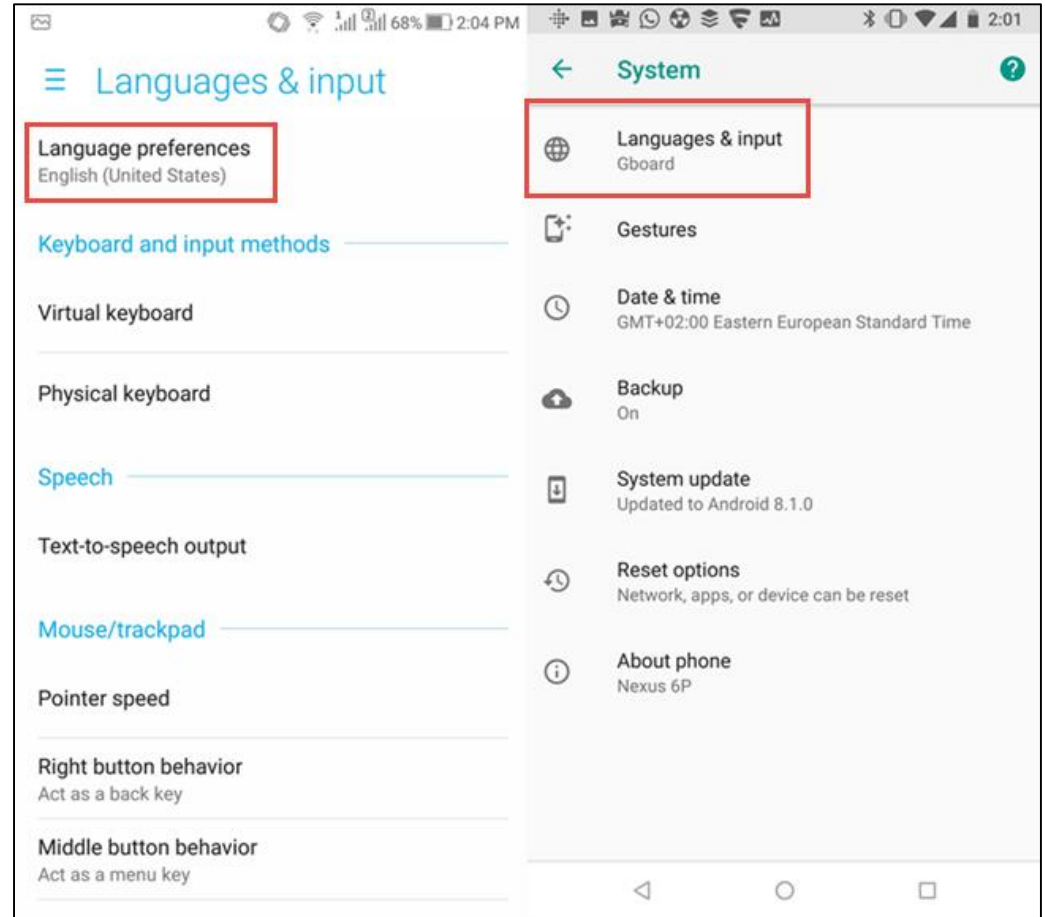
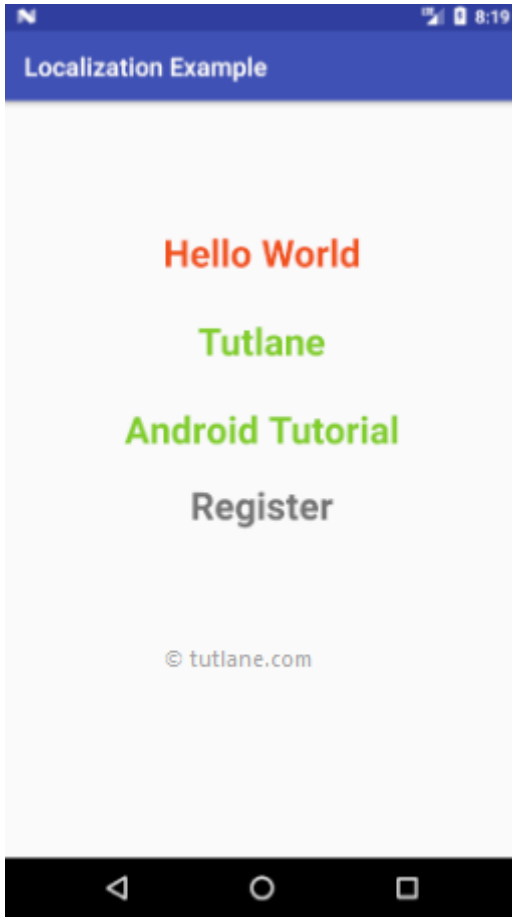
## values-ja/strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="app_name">ローカリゼーション例</string>
  <string name="hello">こんにちは世界</string>
  <string name="tutlane">ツタン</string>
  <string name="android">アンドロイドチュートリアル</string>
  <string name="reigster">登録</string>
</resources>
```





# Output





# Output





*CodeLab*