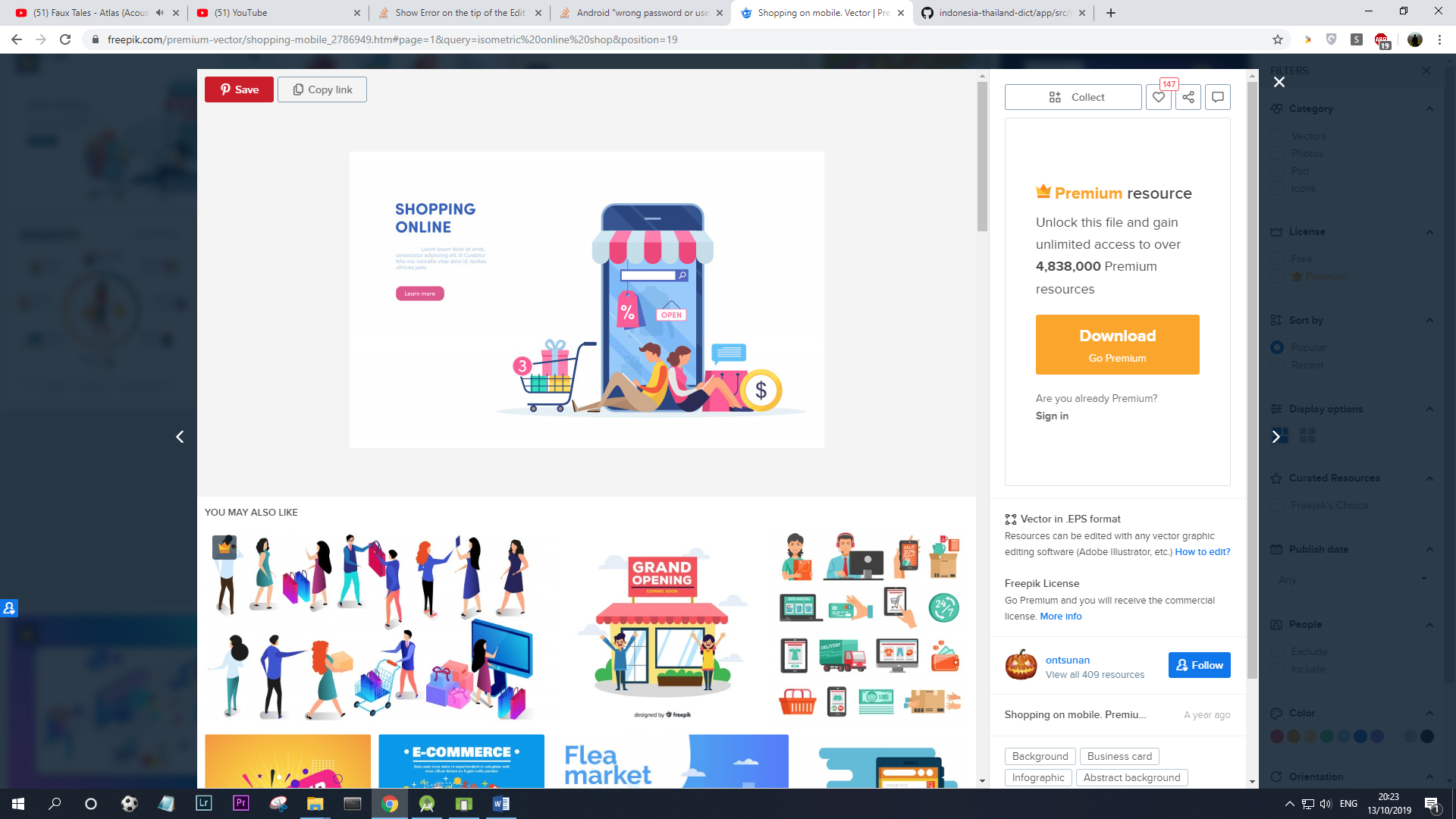
**Prototyping Online Shop with Android Native Programming**

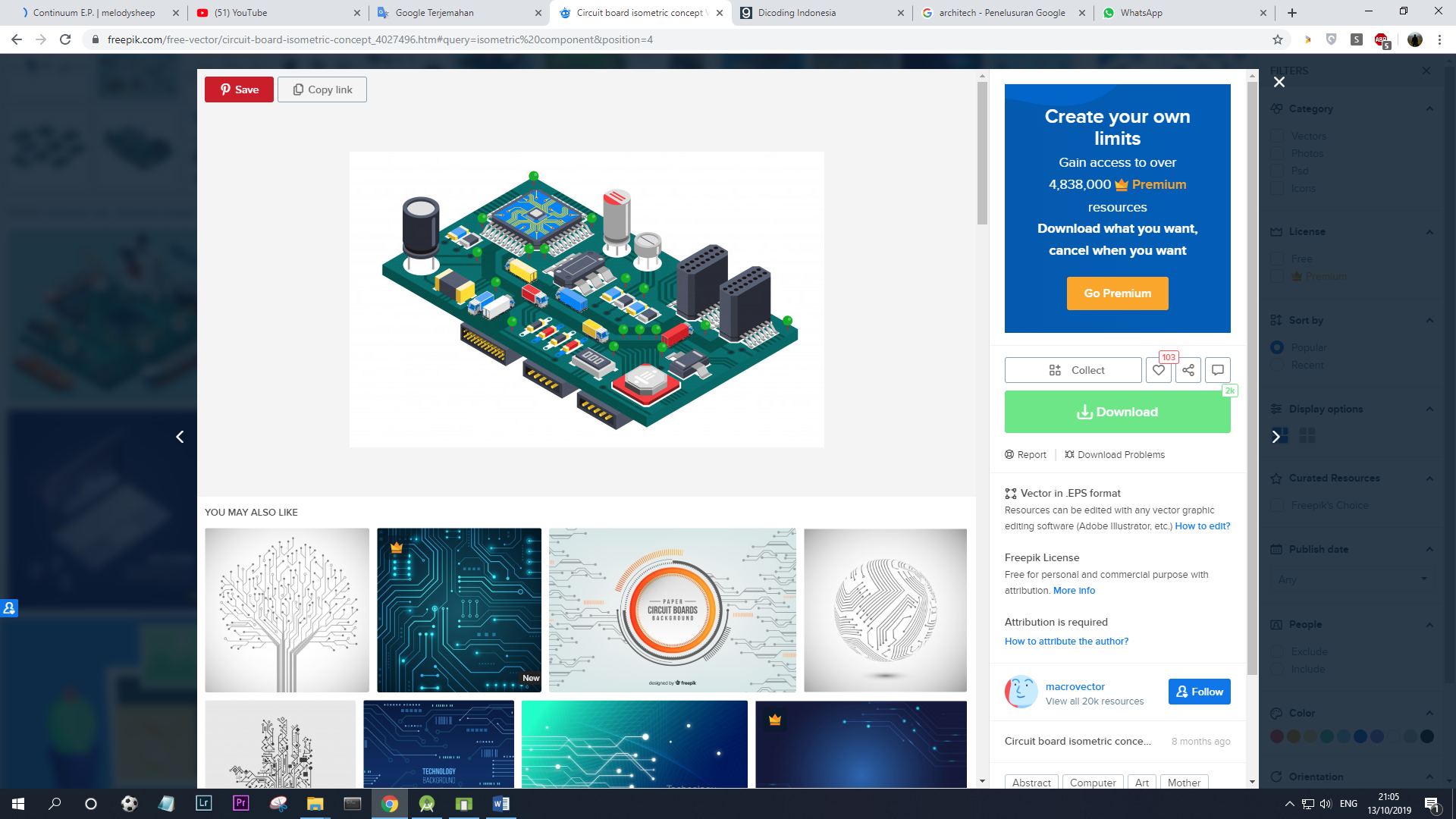


Ditulis Oleh:

 Kharisma Muzaki Ghufron

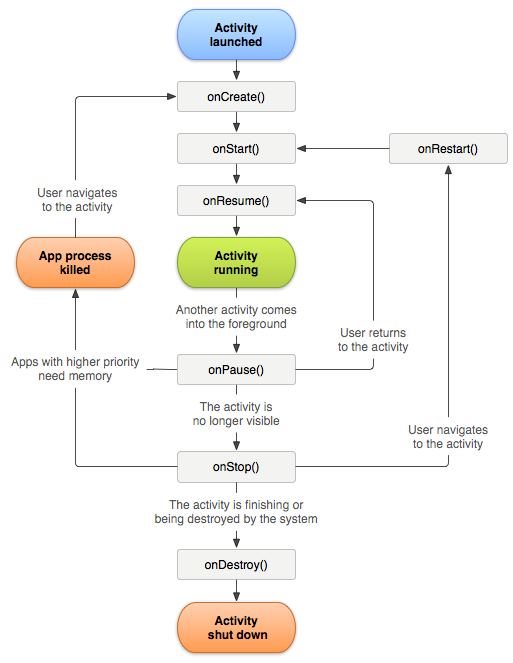
<http://www.github.com/forderation>

**Arsitektur Utama:**



Di bagian ini anda akan belajar teori arsitektur umum android yang sering digunakan dalam pembuatan aplikasi diantaranya yaitu Activity, Fragment, Context dan Intent. Perlu diketahui sebelum mengimplementasikan pengerjaan dalam aplikasi android anda perlu beberapa teori dan arsitektur dalam sebuah komponen.

1. Activity:



* Activity merupakan sebuah komponen di Android yang berfungsi untuk menampilkan user*interface* ke layar handset Android pengguna. Ini seperti pada saat Anda melihat daftar percakapan pada aplikasi *chat* atau daftar *email* pada aplikasi Gmail di ponsel Android Anda.
* Umumnya dalam sebuah aplikasi terdapat lebih dari satu activityyang saling terhubung dengan tugas yang berbeda-beda.
* Activity merupakan salah satu komponen penting Android yang memiliki daur hidup (*life cycle*) dalam sebuah *stack* pada *virtual sandbox*yang disiapkan oleh *Dalvik Virtual Machine* (DVM) atau *Android Runtime* (ART) yang bersifat *last in first out*.
* Pada implementasinya, activity selalu memiliki satu layout *user interface*dalam bentuk ***berkas xml***.
* Suatu aplikasi Android bisa memiliki lebih dari satu activitydan harus terdaftar di berkas ***AndroidManifest.xml***sebagai sub aplikasi.
* Sebuah kelas Java dinyatakan sebuah activityjika mewarisi (*extends*) *superclass*Activity atau turunannya seperti AppCompatActivity dan FragmentActivity.

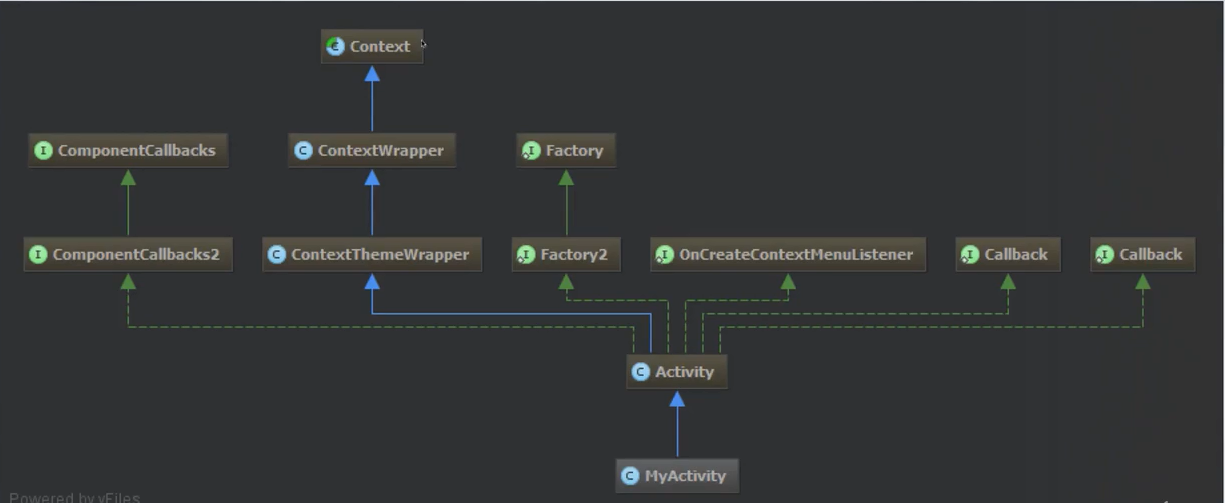
1. Fragment:



Ada beberapa *state* yang perlu anda ketahui sebelum menggunakan fragment.

* **Resumed**  
  Fragment bisa dilihat ketika activity sedang berjalan.
* **Paused**  
  Ketika ada activity lain yang menutupi sebagian dari activity dimana fragment ditambahkan. Yang dimaksud menutupi sebagian adalah ketika activity-nya tidak tertutup sepenuhnya oleh activity lain. Jadi masih ada bagian dari activity yang masih bisa bisa dilihat di layar.
* **Stopped**  
  Ketika fragment tidak kelihatan di layar. Bisa jadi karena activity dimana fragment itu ditambahkan berhenti atau bahkan fragment itu sendiri sudah dihapus dari activity. Pada kondisi ini fragment masih hidup dengan semua informasinya. Akan tetapi sudah tidak kelihatan di layar dan akan dihancurkan.

1. Context:



* Context merupakan representasi untuk mendapatakan suatu data dalam ruang lingkup
* Context diasumsikan seperti remote TV & channel merupakan resource, service, dan lain sebagainya.
* Context dideklarasikan sebagai abstract, dengan begitu implementasi akan disediakan otomatis oleh system operasi android.

Apa yang bisa dilakukan dengan context?:

* Memuat resource atau sumber daya
* Memuat activity baru
* Membuat views
* Memperoleh system service

Cara memanggil context:

* getApplicationContext()
* getContext()
* getBaseContext()

1. Intent

Intent adalah mekanisme untuk melakukan sebuah action dan komunikasi antar komponen aplikasi misal activity. Contoh sering yang dilakukan adalah: memindahkan satu activity ke activity lain dengan atau tidak membawa data.

**Arsitektur Basis Data:**



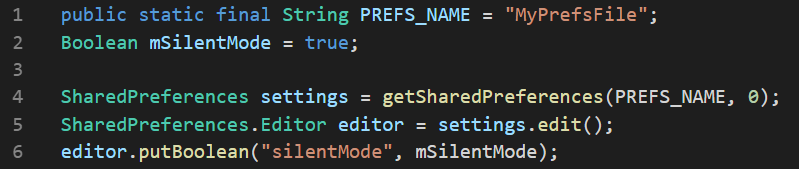
Sebuah aplikasi umumnya tidak lepas dengan penyimpanan baik itu bersifat permanen maupun tidak, di dalam pemrograman android sendiri menawarkan layanan basis data salah satunya yaitu SQLite, sebelum data dimasukkan diperlukan sebuah pemanggilan layanan untuk memasukkan perintah sql agar data dapat disimpan. Selain itu pengambilan data perlu dimanipulasi agar data dapat yang di proses dapat terorientasikan salah satu konsep yang diperlukan yaitu POJO dalam bahasa Java.

1. Class (Plain Old Java Object) dan Parcelable

* Class adalah sebuah kelas yang memiliki atribut, method tempat anda membuat rancagan sebuah struktur data dalam bentuk object oriented. Class ini berfungsi untuk menyimpan model data suatu obyek. Pada Java yang memiliki properti/variabel dan metode setter-getter
* Parcelable adalah suatu interface yang memungkinkan anda melakukan pengiriman objek dari suatu activity ke activity lain. Obyek yang di implementasikan dengan parcelable akan memudahkan Anda dalam mengiriman data dari satu activity ke activity lainnya.

1. Shared Preferences

Media penyimpanan jenis ini akan memudahkan untuk menyimpan data dengan tipe data primitif seperti boolean, int, long, atau string yang tidak berelasi satu sama lain. Konsep yang digunakan pada jenis ini adalah pasangan antara key–value. Ini seperti implementasi cookies pada pengembangan aplikasi berbasis web. Contohnya pada sebuah kelas activity sebagai berikut:



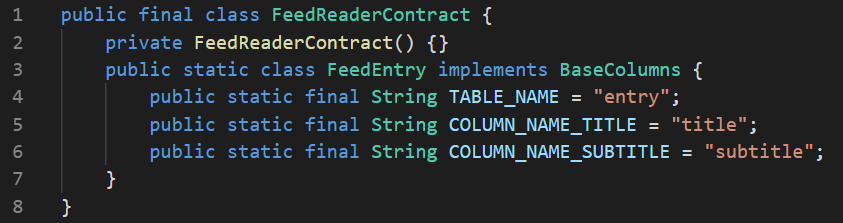
Pada implementasi diatas adalah aplikasi menyimpan sebuah keadaan dengan instruksi menyimpan nilai Boolean pada key: silentMode

1. SQlite Database

SQLite merupakan database yang bersifat open source yang mendukung operasi relasi standar yang umum terdapat pada engine database seperti sintaks SQL dan operasi transaksi. Meskipun berjalan seperti database, pada umumnya sqlite berukuran kecil dan mampu berjalan pada peranti dengan memori terbatas seperti smartphone.

SQLite hanya mendukung beberapa tipe data seperti text untuk penyimpanan data dalam bentuk string, int untuk menyimpan data dalam bentuk bilang bulat, real untuk penyimpan data dalam bentuk bilangan pecahan/bilangan presisi. Jadi apabila ingin menyimpan data yang tidak didukung oleh SQLite maka diharuskan dilakukan proses konversi tipe data yang sesuai dengan tipe data yang didukung sebelum melakukan penyimpanan data.

Langkah pertama dalam pembuatan dan pembaruan database SQLite oleh aplikasi adalah mendefinisikan skema data yang akan diimplementasikan ke dalam database. Semua skema yang ada, dimasukkan pada satu file Java yang berisi nama table, nama kolom, bahkan query yang akan dijalankan.



1. Content Value

Content Value memungkinkan anda memasukkan informasi ke dalam objek dalam bentuk pasangan key-value yang direpresentasikan sebagai kolom dan nilainya. Objek kemudian dapat diteruskan ke method insert () dari kelas SQLiteDatabase untuk menyisipkan atau memperbarui pada method WritableDatabase.

Contoh penggunaan dari content value dalah sebagai berikut:

