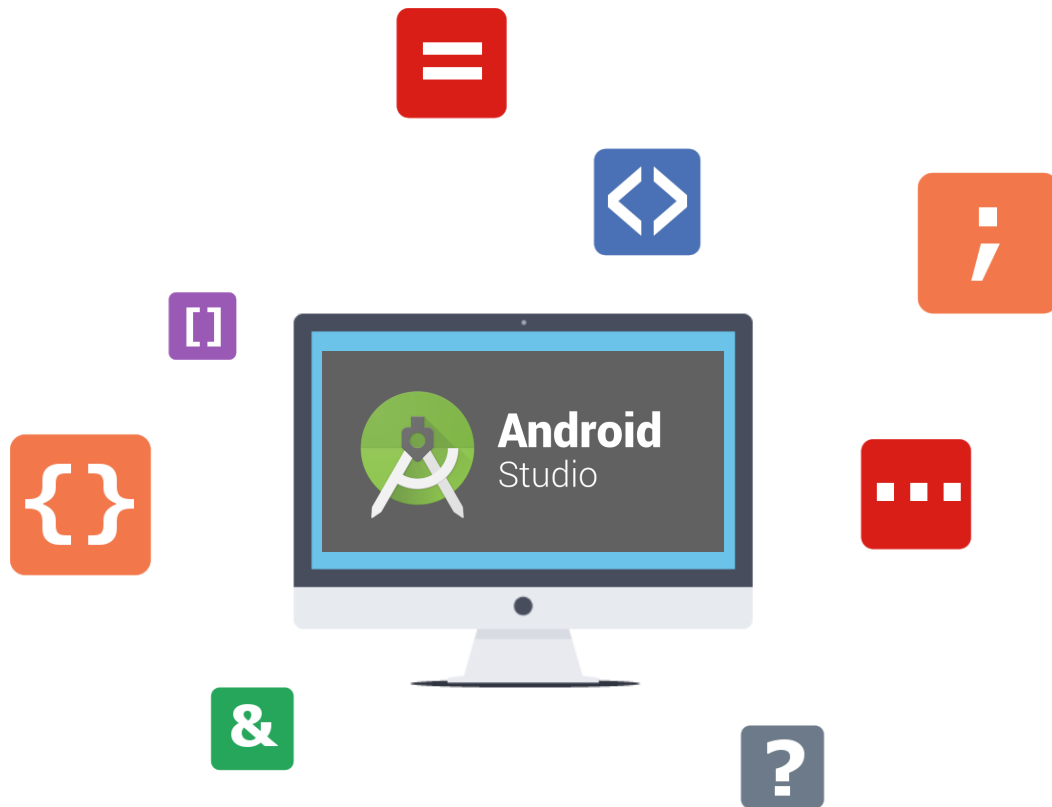




PELATIHAN ANDROID (*NOTIFICATION MESSAGE*)

SITKOM JAVA NETWORK

INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA



ZAYED ELFASA

2017

Tools yang dibutuhkan :

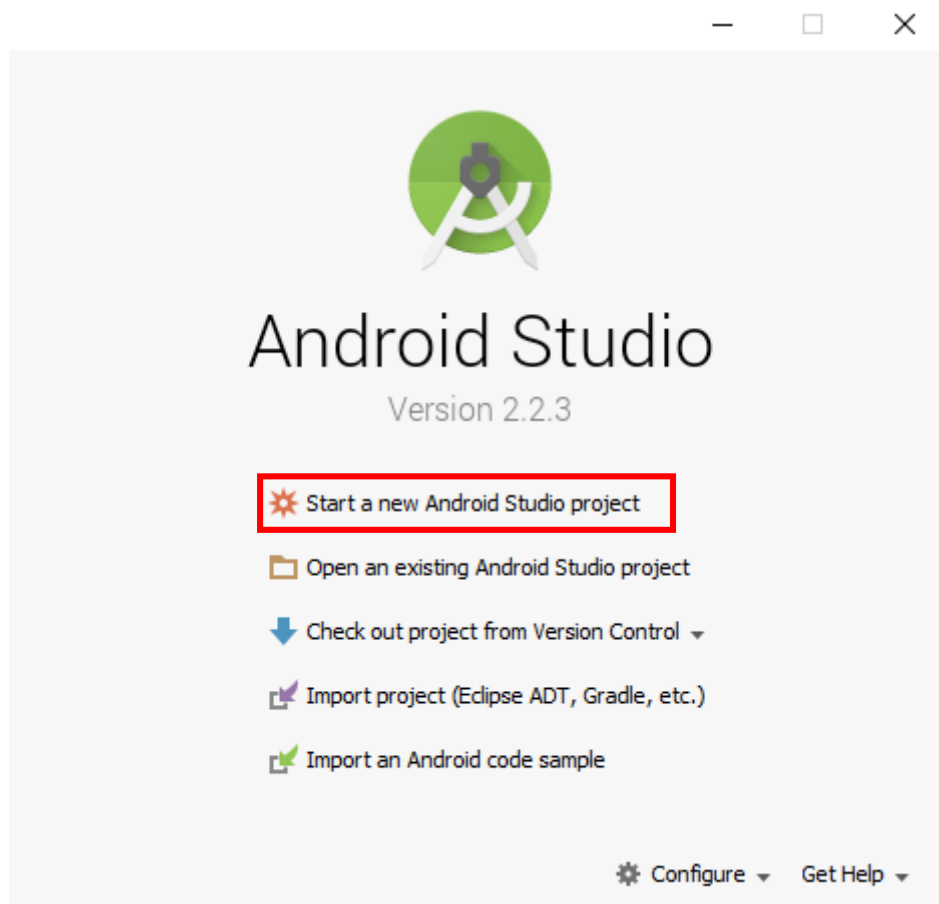
- 1. Android Studio**
- 2. Emulator Android / Smartphone Android**
- 3. XAMPP**
- 4. Editor Bahasa Pemrograman PHP seperti Netbeans, Sublime Text, atau ATOM.**
- 5. Internet**

BAGIAN 1

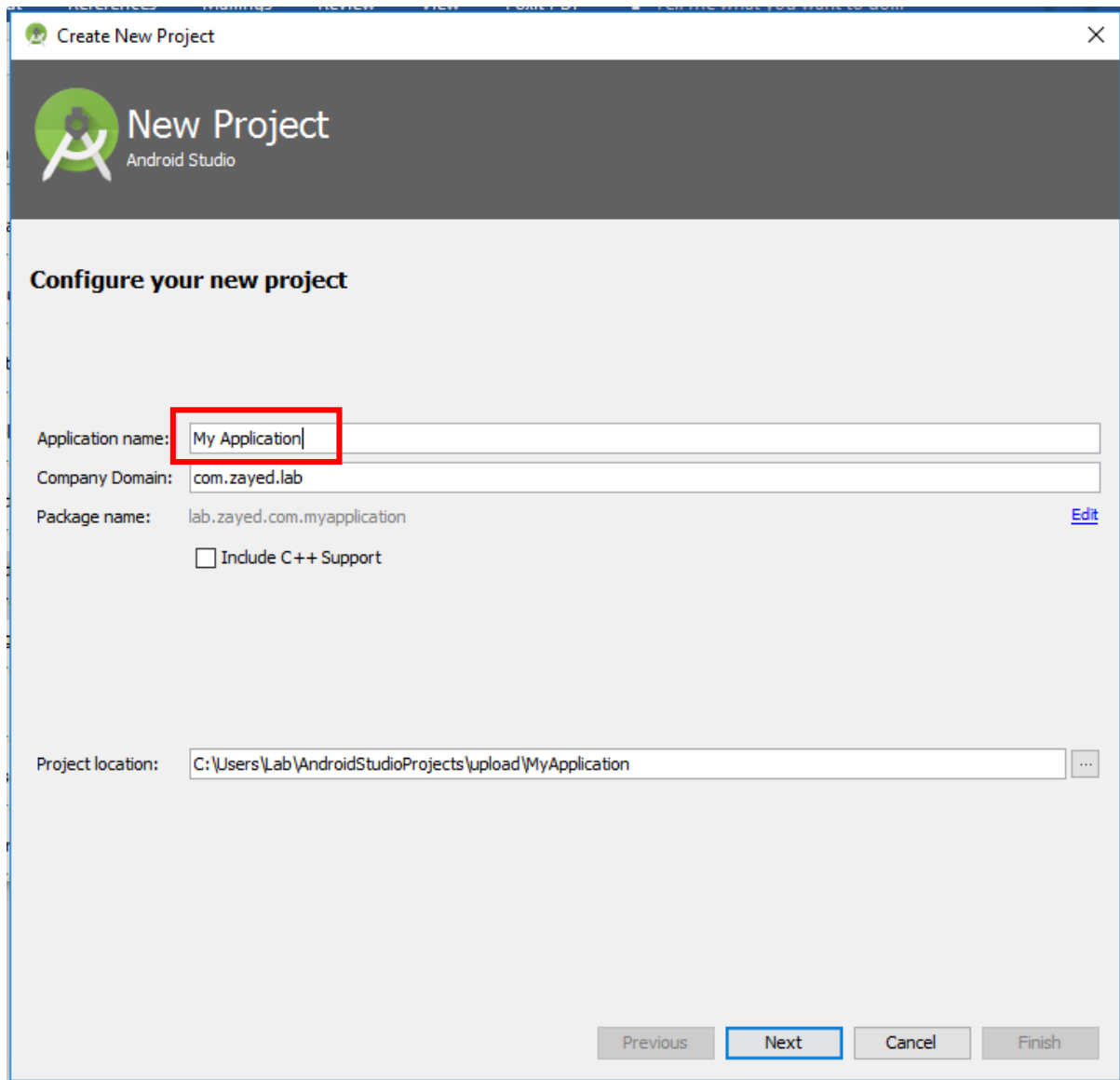
HAPPY CREATE NEW PROJECT!

Untuk membuat aplikasi, maka diperlukan untuk membuat project baru dengan cara sebagai berikut :

1. Buka Android Studio sampai menampilkan gambar seperti di bawah ini, kemudian pilih “Start a new project”.



Setelah itu akan tampil form seperti di bawah ini. Kemudian isikan “Application Name” dengan nama “BroadCast”. Untuk Bagian Company Domain dibiarkan saja. Project Location menunjukkan dimana tempat project Android disimpan. Untuk lokasi nya silahkan ditentukan masing-masing. Kemudian klik tombol “next”.



Create New Project

New Project
Android Studio

Configure your new project

Application name:

Company Domain:

Package name: [Edit](#)

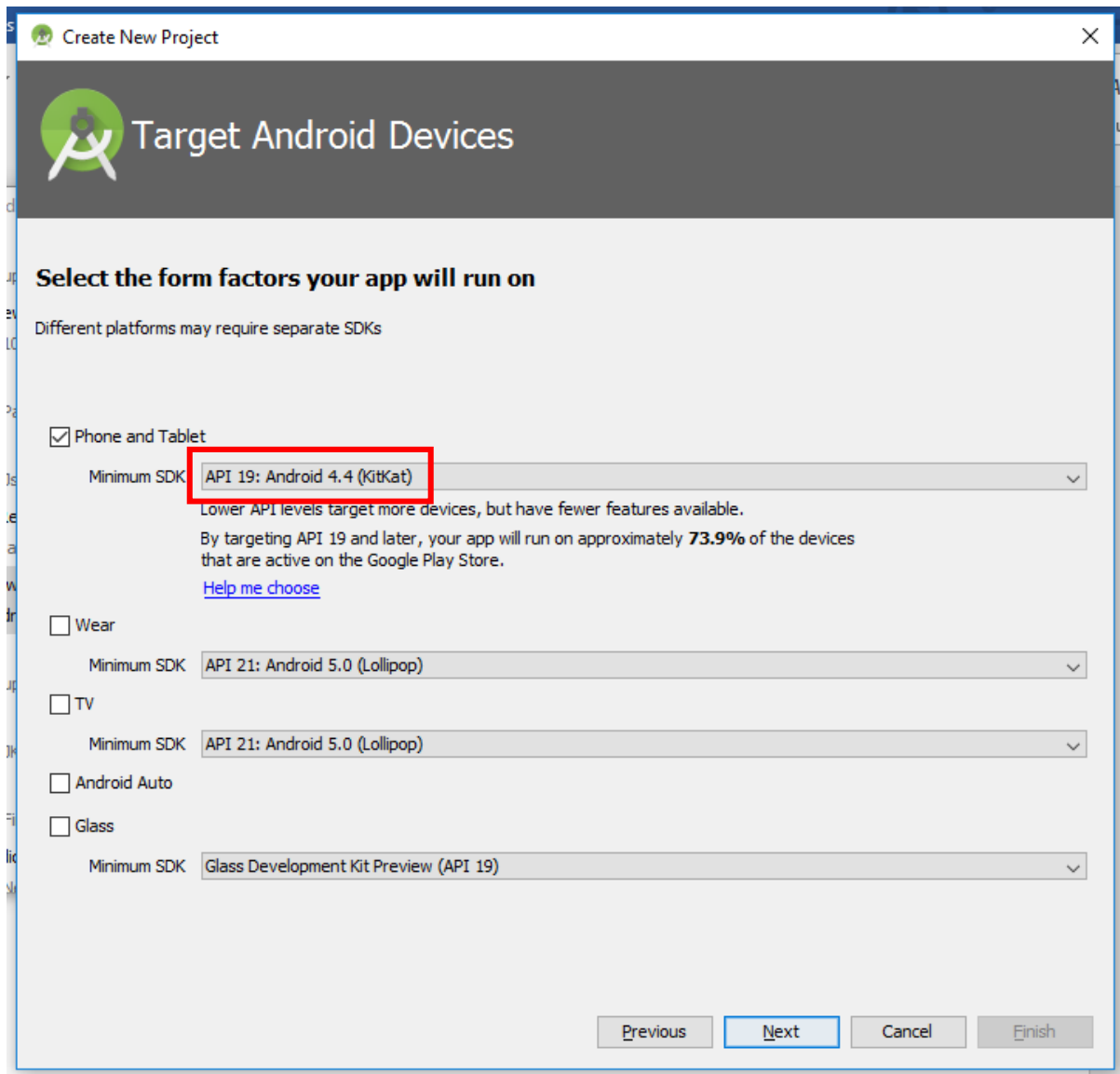
☐ Include C++ Support

Project location: ...

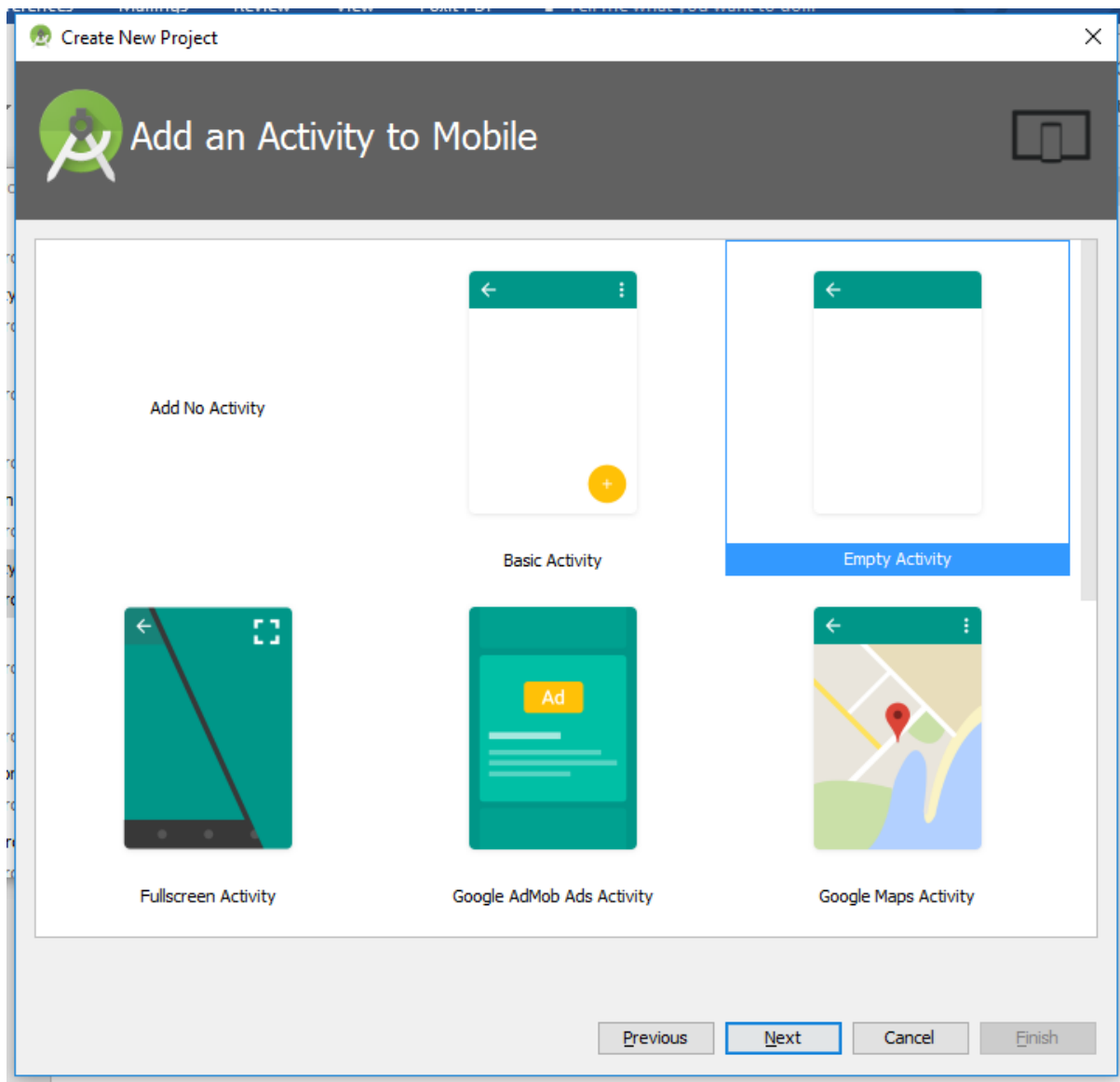
[Previous](#) [Next](#) [Cancel](#) [Finish](#)

Setelah itu Android Studio ini akan menampilkan “Target Android Devices”. Pada field “Phone And Tablet” untuk kali ini pilihlah Minimal SDK yaitu “API 19: Android 4.4 (Kitkat)”

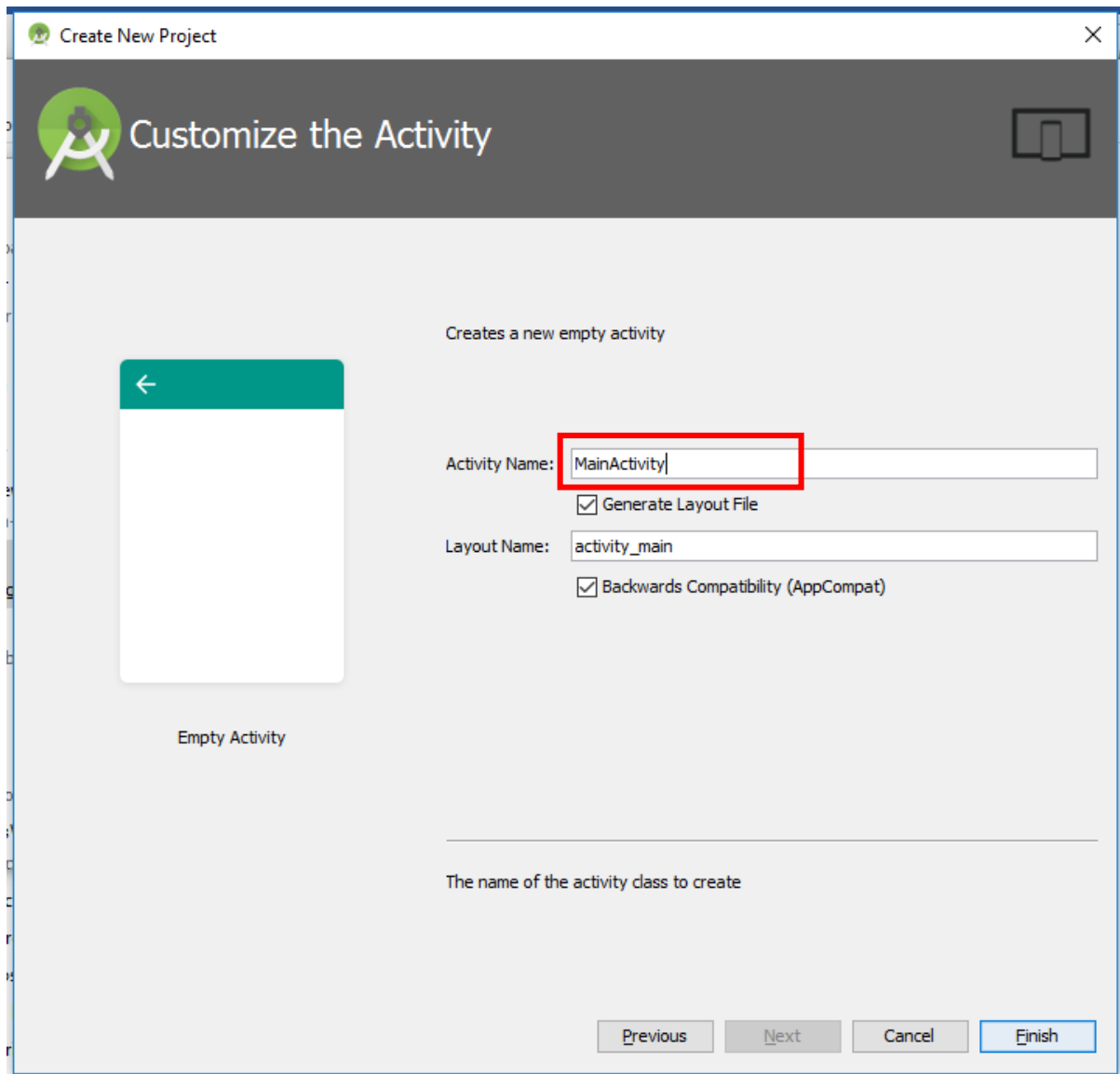
Untuk field yang lainnya tidak perlu ditandai. Kemudian klik tombol “next”.



Setelah itu Android Studio akan menampilkan Activity utama yang akan digunakan ketika pertama kali activity tampil. Untuk saat ini pilih “Empty Activity” agar layout yang akan dituliskan kode nantinya kosongan. Setelah itu langsung menuju tombol “Next” kemudian klik untuk menuju form selanjutnya.



Form terakhir sebelum Android melakukan Build Project yaitu memberikan nama Activity yang pertama kali dijalankan. Biasanya dari Android Studio sudah memberikan nilai default "MainActivity" untuk Activity-nya dan memberikan nilai default "activity_main" pada layout-nya. Dua tempat inilah nantinya akan difokuskan untuk melakukan pengembangan aplikasi broadcast ini. "MainActivity" digunakan untuk menuliskan program dalam bentuk Bahasa java dan "activity_main" digunakan untuk menuliskan program dalam bentuk tag XML yang disesuaikan dengan standar android studio. Setelah itu klik tombol "finish".



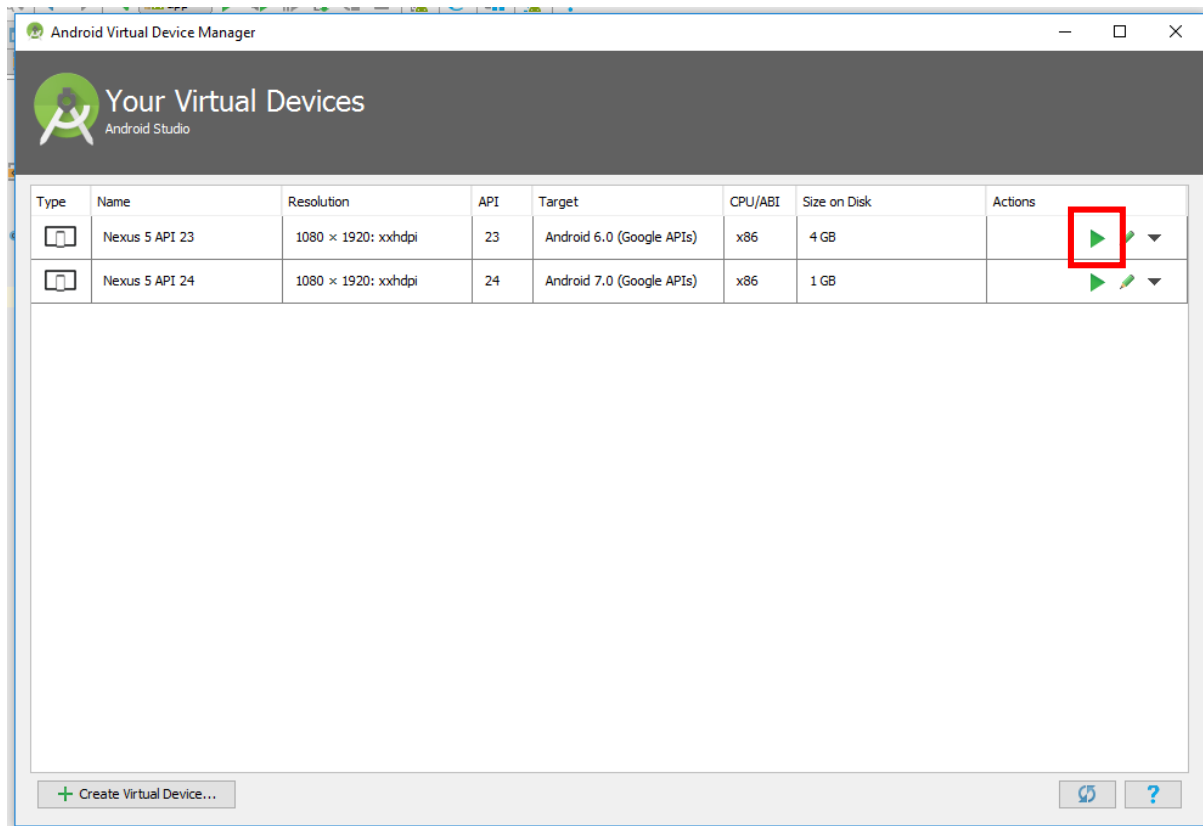
HAY.. I AM ANDROID STUDIO 😊

Sebelum memulai membuat program maka harus kenal fitur bantuan untuk membuka emulator dan compile agar kita dapat menjalankan programnya.

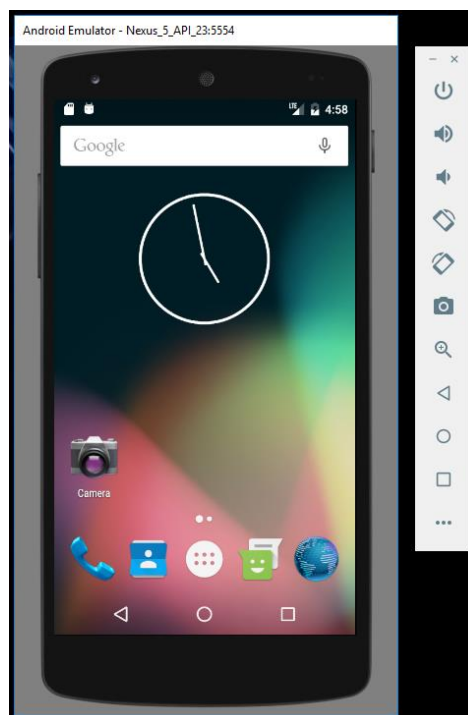
Cara membuka emulator



Untuk membuka emulator pilihlah “AVD Manager” seperti contoh pada gambar di atas dengan cara klik nanti akan menampilkan form seperti di bawah ini :



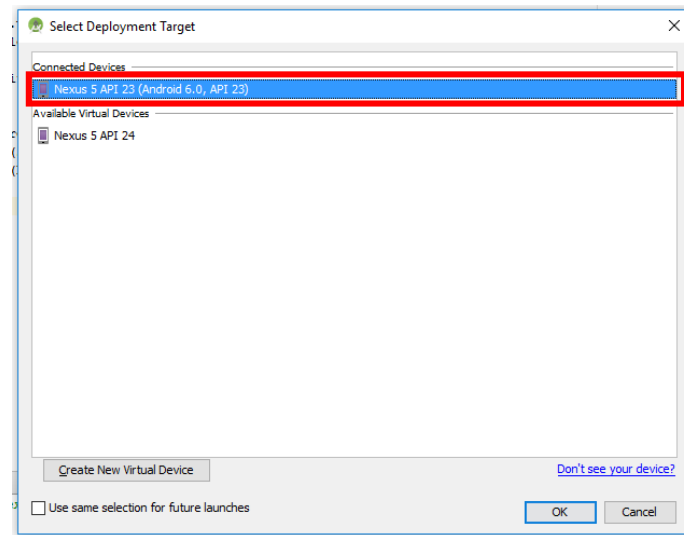
Setelah tampil seperti form di atas, kemudian pilihlah satu dari emulator yang telah dibuat dengan menekan tombol segitiga seperti pada gambar di atas. Setelah itu ditunggu dan emulator akan langsung hidup seperti gambar di bawah ini :



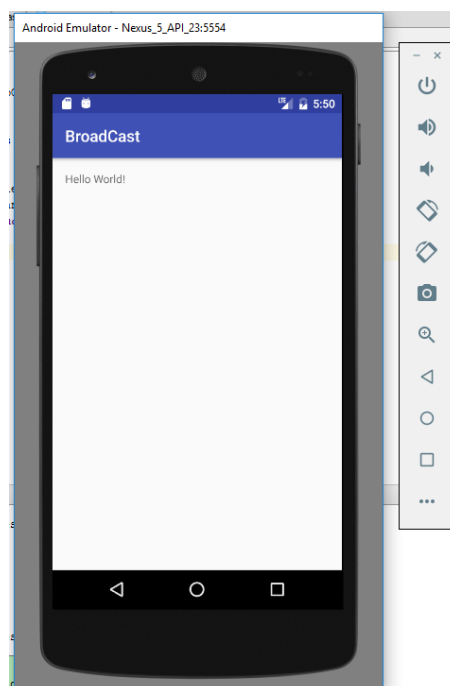
Untuk melakukan compiling dapat sebuah program ada dua cara yaitu :



1. Compile dari awal dengan cara klik tobol segitiga seperti pada gambar di atas. Setelah itu akan muncul form seperti gambar di bawah ini untuk memilih device yang terkoneksi :



2. Setelah itu klik tombol "Ok". Setelah itu emulator langsung akan menampilkan hasil dari program yang telah dibuat.



BAGIAN 2

HAPPY LAYOUT-ING ☺

a. Layout activity_main.xml

Ketika membuat project baru pada Android Studio, maka langsung disuguhkan dengan sebuah activity_main. Activity pada activity_main digunakan untuk menampilkan login yang terdiri dari tampilan field untuk mengisi nama pengguna dan tampilan field untuk mengisi kata sandi. Isilah tag layout seperti di bawah ini :

```
<?xml version="1.0" encoding="utf-8" ?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="lab.zayed.com.broadcast.MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical" >
        <EditText
            android:id="@+id/editUserName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="NAMA PENGGUNA"/>

        <EditText
            android:id="@+id/editPassword"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="KATA SANDI"/>

        <Button
            android:text="LOGIN"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/btnLogin"/>
    </LinearLayout>
</RelativeLayout>
```

b. Layout activity_send_broad_cast.xml

Activity_send_broad_cast ini digunakan untuk mengisi pesan apa yang ingin disampaikan kepada pengguna yang lainnya (*broadcast*). Pesannya terdiri dari judul

pesan dan isi pesan yang akan disampaikan. Untuk melihat syntax layout-nya dapat dilihat di bawah ini.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_send_broad_cast"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="lab.zayed.com.broadcast.SendBroadCastActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical" >

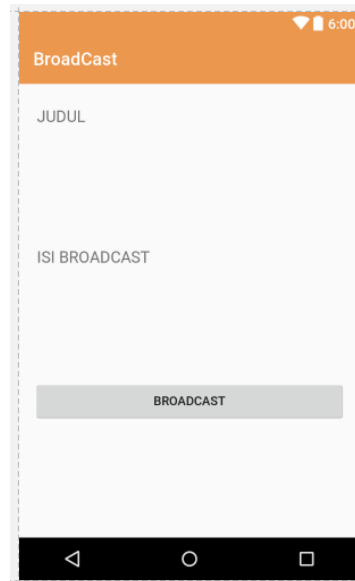
        <EditText
            android:id="@+id/editJudul"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="JUDUL"/>

        <EditText
            android:id="@+id/editIsiBroadCast"
            android:layout_width="match_parent"
            android:layout_height="550px"
            android:hint="ISI BROADCAST"/>

        <Button
            android:text="BROADCAST"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/btnBroadCast"/>
    </LinearLayout>

</RelativeLayout>
```

Setelah mengikuti syntax diatas maka akan menampilkan layout seperti di bawah ini :

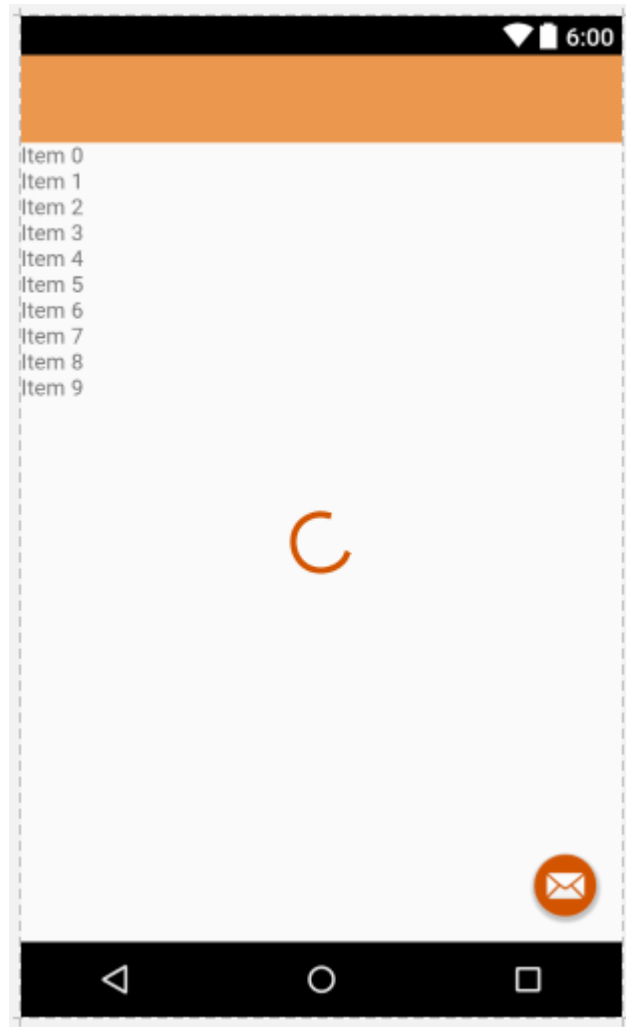


c. Layout content_list_broad_cast

Layout ini digunakan untuk menampilkan daftar pengumuman yang telah dikirim kepada pengguna lainnya. Daftar tampilan dalam bentuk list seperti yang ada pada *google play store*. Untuk menampilkan daftar informasi perlu tag RecyclerView dan ProgressBar untuk melakukan loading jika data belum siap. Untuk lebih jelasnya mari kita buat tampilannya dulu. Ikuti *syntax* layout di bawah ini :

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/content_list_broad_cast"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context="lab.zayed.com.broadcast.ListBroadcastActivity"
    tools:showIn="@layout/activity_list_broad_cast2">
    <ProgressBar
        android:id="@+id/progresBar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"/>
    <android.support.v7.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
    </android.support.v7.widget.RecyclerView>
</RelativeLayout>
```

Setelah menuliskan kode seperti di bawah ini maka akan menampilkan layout seperti di bawah ini :



d. Layout rv_informasi

Layout ini adalah layout bantuan untuk menampilkan informasi dari broadcast yang telah disampaikan. Nantinya layout ini akan disambungkan pada metode RecyclerView sebagai layout bantuannya. Untuk syntaksnya dapat dilihat di bawah ini :

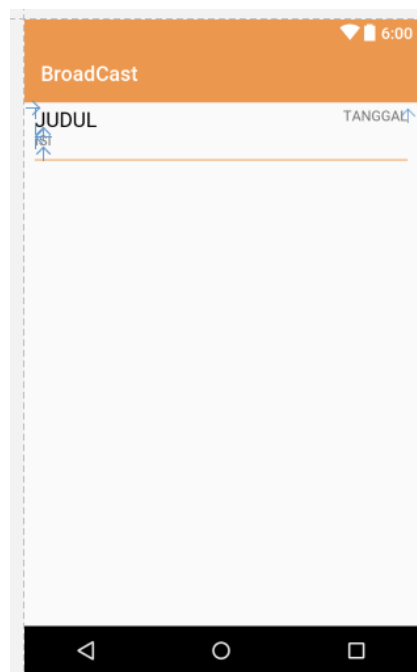
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="10dp"
    android:layout_marginRight="10dp"
    android:layout_marginTop="5dp">
    <TextView
        android:id="@+id/textID"
        android:text="ID"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:visibility="gone"/>
    <TextView
        android:id="@+id/textJUDUL"
        android:text="JUDUL"
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginRight="70dp"
        android:textSize="20dp"
        android:textColor="#000000"/>
<TextView
    android:id="@+id/textTANGGAL"
    android:text="TANGGAL"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"/>
<TextView
    android:id="@+id/textISI"
    android:text="ISI"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textJUDUL"
    android:layout_toEndOf="@+id/textID" />
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_gravity="center"
    android:layout_marginBottom="5dp"
    android:layout_marginTop="10dp"
    android:background="@color/colorPrimaryDark"
    android:layout_below="@id/textISI"/>
</RelativeLayout>

```

Setelah menuliskan layout di atas, maka hasilnya seperti di bawah ini :

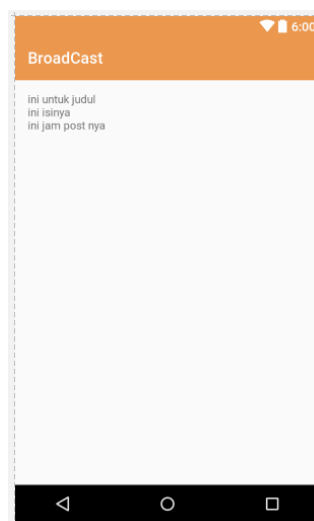


e. Layout activity_detil_informasi

Layout ini digunakan untuk menampilkan detail dari sebuah informasi dari *broadcast*. Deil informasi ini akan tampil apabila pengguna memilih salah satu daftar dari informasi. Syntax pembuatan layout untuk activity_detil_informasi dapat dilihat di bawah ini :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_detil_informasi"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="lab.zayed.com.broadcast.DetilInformasiActivity">
    <TextView
        android:text="ini untuk judul"
        android:id="@+id/textJUDUL"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
    <TextView
        android:text="ini isinya"
        android:id="@+id/textISI"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
    <TextView
        android:text="ini jam post nya"
        android:id="@+id/textJAMPOST"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
</LinearLayout>
```

Setelah menuliskan syntax di atas, maka akan tampil layout seperti di bawah ini :



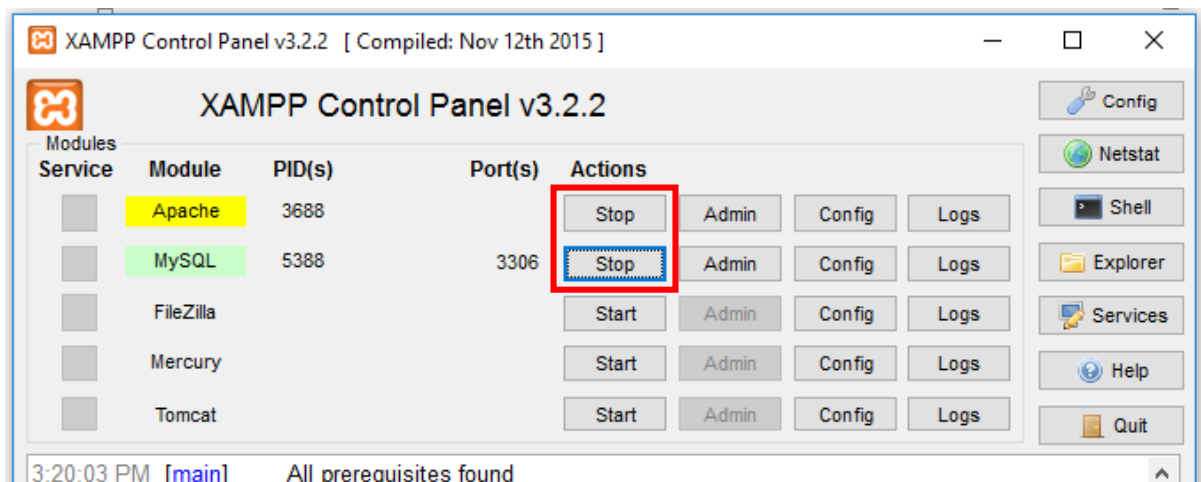
BAGIAN 3

LETS HAPPY PHP CODE! (SERVER SIDE) ☺

Sebelum memulai membuat program di server side menggunakan PHP, mari kita buat dulu database nya menggunakan MySQL.

Caranya sebagai berikut :

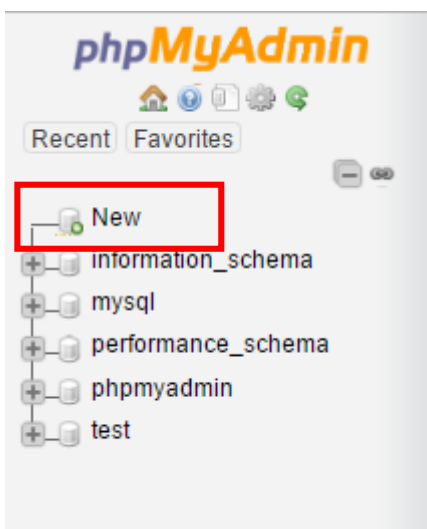
Bukalah XAMPP kemudian nyalakan Apache dan MySQL nya :



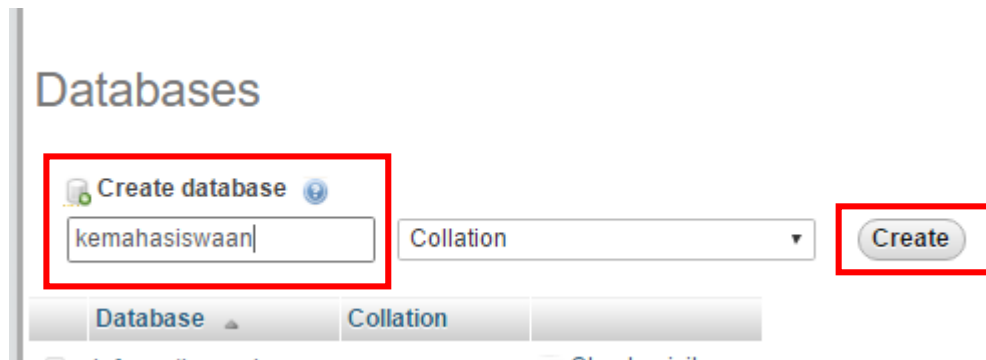
Setelah itu bukalah phpmyadmin kemudian buat database :

<http://localhost/phpmyadmin>

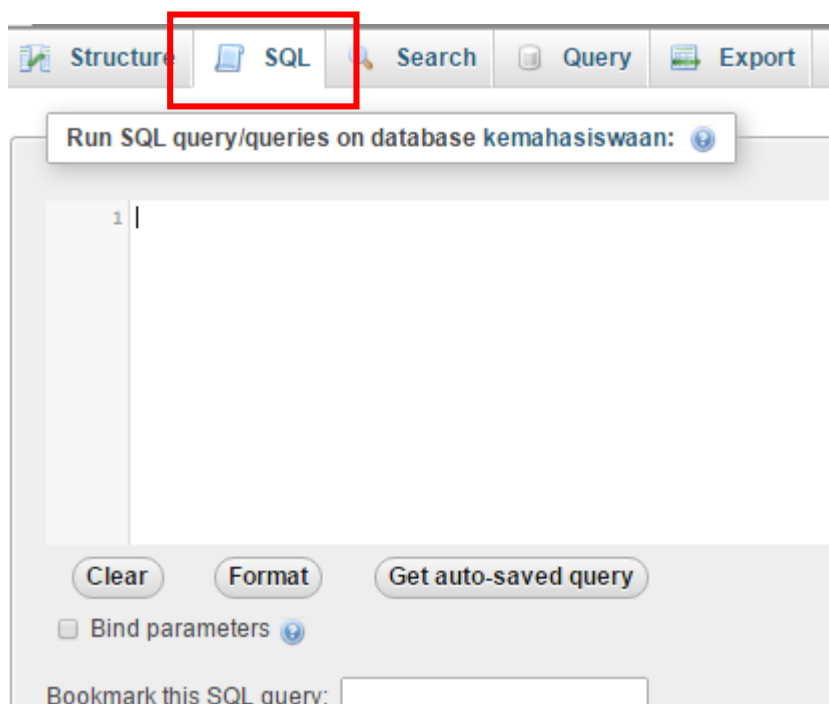
Buat database dengan nama “kemahasiswaan” dengan cara klik “new” pada tab pojok kiri phpmyadmin.



Setelah itu isilah field pada “Create Database” dengan nama “kemahasiswaan” kemudian klik tombol “create”



Setelah database “kemahasiswaan” telah selesai dibuat, Anda harus mengisi database dalam bentuk tabel “informasi” dan tabel “fcmregid” pada tab “SQL”.



Setelah itu isilah query create database dengan syntax di bawah ini :

```
/*
SQLyog Ultimate v11.11 (64 bit)
MySQL - 5.5.5-10.1.13-MariaDB : Database - kemahasiswaan
*****

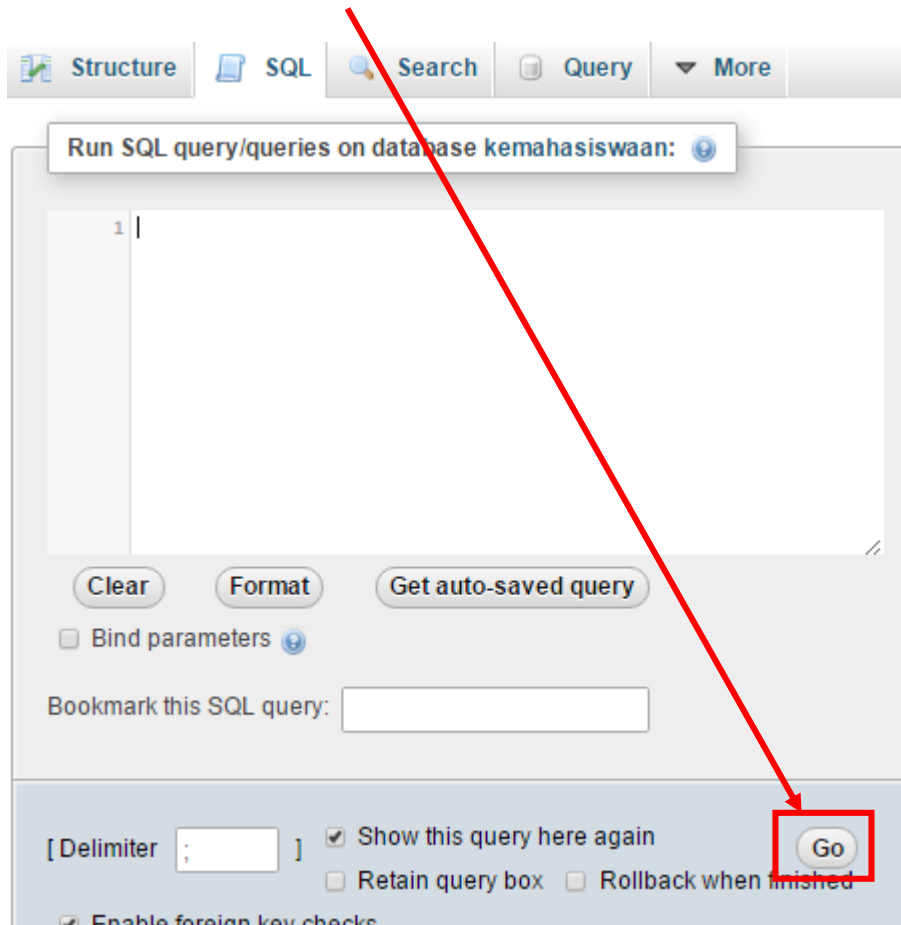
*/
/*!40101 SET NAMES utf8 */;
/*!40101 SET SQL_MODE=''*/;
```

```

/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
CREATE DATABASE /*!32312 IF NOT EXISTS*/`kemahasiswaan` /*!40100 DEFAULT
CHARACTER SET latin1 */;
USE `kemahasiswaan`;
/*Table structure for table `fcmregid` */
DROP TABLE IF EXISTS `fcmregid`;
CREATE TABLE `fcmregid` (
  `id` int(2) NOT NULL AUTO_INCREMENT,
  `fcmid` varchar(200) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
/*Data for the table `fcmregid` */
/*Table structure for table `informasi` */
DROP TABLE IF EXISTS `informasi`;
CREATE TABLE `informasi` (
  `id` int(2) NOT NULL AUTO_INCREMENT,
  `judul` varchar(150) NOT NULL,
  `isi` varchar(1000) NOT NULL,
  `jampost` varchar(15) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;
/*Data for the table `informasi` */
insert into `informasi`(`id`,`judul`,`isi`,`jampost`) values (1,'Penyerahan
Penghargaan kepada UKM LUG','Penyerahan Penghargaan kepada UKM LUG karena
telah memenangkan bermacam-macam lomba yang di adakan di daerah surabaya
ataupun di luar surabaya','16:40:56'),(2,'Penyerahan Penghargaan kepada UKM
SJN','Penyerahan Penghargaan kepada UKM LUG karena telah berdedikasi terhadap
Institut bisnis dan informatik stikom surabaya','18:46:32');
/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

```

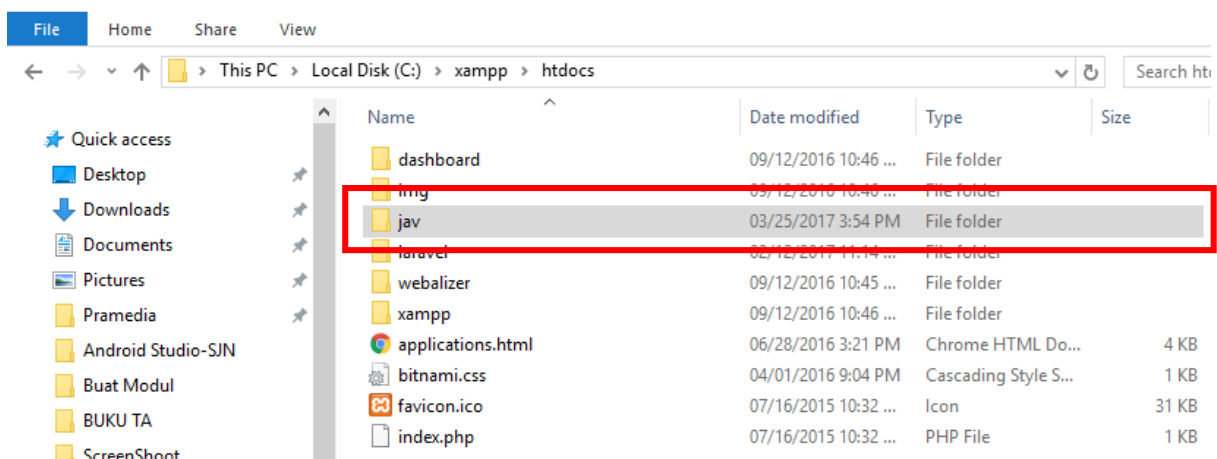
Setelah itu klik tombol “go” yang berada pada sebelah kanan bawah.



Setelah itu mari kita lanjut membuat code untuk server side menggunakan bahasa PHP.

Caranya adalah Anda masuk ke dalam c:\xampp\htdocs kemudian buat folder baru bernama “java” (optional).

Nanti hasilnya seperti berikut ini :



Setelah itu mari kita membuat program menggunakan PHP!

*Semua file yang ada di bawah ini ada di dalam folder “jav”.

1. Buat file database.php

File ini digunakan untuk menghubungkan dengan database yang telah disiapkan di atas.

Syntax nya sebagai berikut :

```

1  <?php
2  /**
3   * create by zayed elfasa
4   */
5  class database
6  {
7      private static $dbName = 'kemahasiswaan';
8      private static $dbHost = 'localhost';
9      private static $dbUsername = 'root';
10     private static $dbUserPassword = '';
11
12     private static $scont = null;
13
14     public function __construct() {
15         die('Init function is not allowed');
16     }
17
18     public static function connect()
19     {
20         // One connection through whole application
21         if ( null == self::$scont )
22         {
23             try
24             {
25                 self::$scont = new PDO( "mysql:host=".self::$dbHost.";". "dbname=".self::$dbName, self::$dbUsername, self::$dbUserPassword);
26             }
27             catch(PDOException $e)
28             {
29                 die($e->getMessage());
30                 echo "Connection failed: " . $e->getMessage();
31             }
32         }
33         return self::$scont;
34     }
35
36     public static function disconnect()
37     {
38         self::$scont = null;
39     }
40 }
```

2. Buat file modelInformasi.php

File ini merupakan file untuk penyimpanan sementara atribut atribut yang berkaitan dengan satu informasi. modelInformasi.php merupakan file objek yang berikan method-method setter untuk set nilai dari atribut informasi dan method-method getter untuk get (mengambil) nilai dari atribut informasi. Syntax-nya adalah sebagai berikut :

```
<?php
2 class modelInformasi {
3     var $id;
4     var $judul;
5     var $isi;
6     var $jamPost;
7     public function __construct() {}
8     public function __construct1($id, $judul, $isi, $jamPost) {
9         $this->id = $id;
10        $this->judul = $judul;
11        $this->isi = $isi;
12        $this->jamPost = $jamPost;
13    }
14    public function setId($id) {
15        $this->id = $id;
16    }
17    public function setJudul($judul) {
18        $this->judul = $judul;
19    }
20    public function setIsi($isi) {
21        $this->isi = $isi;
22    }
23    public function setJamPost($jamPost) {
24        $this->jamPost = $jamPost;
25    }
26    public function getId() {
27        return $this->id;
28    }
29    public function getJudul() {
30        return $this->judul;
31    }
32    public function getIsi() {
33        return $this->isi;
34    }
35    public function getJamPost() {
36        return $this->jamPost;
37    }
38 }
```

3. Buat file controllerInformasi.php

File controllerInformasi.php adalah file controller yang bertujuan untuk menghubungkan antara model dengan file index php untuk melempar data ke aplikasi Android. File controllerInformasi.php berisikan method untuk set informasi yang akan di broadcast dan method get untuk mengambil data dari database. Syntaxnya sebagai berikut :

```
<?php class controllerInformasi{
2   function getInformasi() {
3       $pdo = database::connect();
4       $sql = "select id, judul, isi, jampost from informasi";
5       $arr = [];
6       $inc = 0;
7       foreach ($pdo->query($sql) as $row) {
8           $modelInformasi = new modelInformasi();
9           $modelInformasi->setId($row['id']);
10          $modelInformasi->setJudul($row['judul']);
11          $modelInformasi->setIsi($row['isi']);
12          $modelInformasi->setJamPost($row['jampost']);
13          $jsonArrayObject = (array("id" => $modelInformasi->getId(), "judul" => $modelInformasi->getJudul(), "isi" =>
14              $modelInformasi->getIsi(), "jampost" => $modelInformasi->getJamPost()));
15          $arr[$inc] = $jsonArrayObject;
16          $inc++;
17      }
18      echo json_encode($arr, JSON_PRETTY_PRINT);
19      database::disconnect();
20  }
21  function setInformasi($judul, $isi, $jamPost) {
22      try {
23          $pdo = database::connect();
24          $sql = "INSERT INTO informasi (judul, isi, jampost) VALUES ('".$judul."', '".$isi."', '".$jamPost."')";
25          // use exec() because no results are returned
26          $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
27          $pdo->exec($sql);
28          $returnArray = array("status" => "1", "message" => "New record created successfully");
29          echo json_encode($returnArray, JSON_PRETTY_PRINT);
30          database::disconnect();
31      } catch(PDOException $e) {
32          // echo $sql . "<br>" . $e->getMessage();
33          $returnArray = array("status" => "0", "message" => $e->getMessage());
34          echo json_encode($returnArray, JSON_PRETTY_PRINT);
35      }
36  }
37 }
```

4. Buat file index.php

File index.php ini digunakan untuk memanggil method getInformasi() yang ada pada controllerInformasi.php. Setelah dipanggil akan kemudian link untuk file index.php akan menjadi endpoint untuk mendapatkan data informasi yang disebarkan dalam format JSON. Syntax pada file index.php ada pada di bawah ini :

```
<?php
2   include 'controllerInformasi.php';
3   $controllerInformasi = new controllerInformasi();
4   $controllerInformasi->getInformasi();
```

5. Buat file insert.php

File insert.php digunakan untuk menginputkan data-data informasi yang akan disebar (broadcast). File insert.php ini akan menjadi endpoint untuk menginputkan data informasi yang dilakukan pada aplikasi Android. Syntax pada file insert.php ada di bawah ini :

```
1 <?php
2 include 'controllerInformasi.php';
3 $controllerInformasi = new controllerInformasi();
4
5 $judul = $_POST['judul'];
6 $isi = $_POST['isi'];
7 $jampost = $_POST['jampost'];
8 $controllerInformasi->setInformasi($judul, $isi, $jampost);
```

BAGIAN 4

LETS HAPPY JAVA CODE!

build.gradle

Tambahkan compile pada file build.gradle pada Gradle Scripts -> build.gradle(module.app)

Tambahkan di dalamnya dependencies

```
compile 'com.android.volley:volley:1.0.0'
```

Contohnya seperti syntax di bawah ini :

```
dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {
        exclude group: 'com.android.support', module: 'support-annotations'
    })
    compile 'com.android.support:appcompat-v7:25.1.0'
    compile 'com.android.support:design:25.1.0'
    compile 'com.android.volley:volley:1.0.0'
    testCompile 'junit:junit:4.12'

    compile 'com.google.firebase:firebase-messaging:9.6.0'
}
```

Kemudian tambahkan permission untuk internet pada manifest.xml yang terletak pada folder “app” -> “manifests” -> “AndroidManifest.xml” :

```
<uses-permission android:name="android.permission.INTERNET" />
```

AppController.java

Kemudian buatlah file pada folder java -> nama.project.com dengan cara klik kanan pada mouse kemudian pilih new->java class kemudian simpan nama file java dengan nama AppController.java. File ini digunakan untuk menghubungkan antara client dan server.

Isi dari AppController.java sebagai berikut :

```
package lab.zayed.com.broadcast;

import android.app.Application;
import android.text.TextUtils;

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.toolbox.Volley;
```



```

/**
 * Created by lab on 12/8/16.
 */

public class ApplicationController extends Application {

    public static final String TAG = ApplicationController.class.getSimpleName();

    private RequestQueue mRequestQueue;

    private static ApplicationController mInstance;

    @Override
    public void onCreate() {
        super.onCreate();
        mInstance = this;
    }

    public static synchronized ApplicationController getInstance() {
        return mInstance;
    }

    public RequestQueue getRequestQueue() {
        if (mRequestQueue == null) {
            mRequestQueue =
Volley.newRequestQueue(getApplicationContext());
        }

        return mRequestQueue;
    }

    public <T> void addToRequestQueue(Request<T> req, String tag) {
        req.setTag(TextUtils.isEmpty(tag) ? TAG : tag);
        getRequestQueue().add(req);
    }

    public <T> void addToRequestQueue(Request<T> req) {
        req.setTag(TAG);
        getRequestQueue().add(req);
    }

    public void cancelPendingRequests(Object tag) {
        if (mRequestQueue != null) {
            mRequestQueue.cancelAll(tag);
        }
    }
}

```

Informasi.java

Setelah itu buatlah file Informasi.java. File ini dibuat untuk setter dan getter dari atribut informasi yang akan ditampilkan.

Syntaxnya sebagai berikut :

```
package lab.zayed.com.broadcast;
```

```

import android.os.Parcel;
import android.os.Parcelable;

/**
 * Created by Lab on 14/03/2017.
 */

public class Informasi implements Parcelable {
    private String id, judul, isi, jampost;

    public Informasi() {
        // none
    }

    public Informasi(String id, String judul, String isi, String jampost) {
        this.id = id;
        this.judul = judul;
        this.isi = isi;
        this.jampost = jampost;
    }

    public String getId() {
        return id;
    }

    public void setId(String id) {
        this.id = id;
    }

    public String getJudul() {
        return judul;
    }

    public void setJudul(String judul) {
        this.judul = judul;
    }

    public String getIsi() {
        return isi;
    }

    public void setIsi(String isi) {
        this.isi = isi;
    }

    public String getJampost() {
        return jampost;
    }

    public void setJampost(String jampost) {
        this.jampost = jampost;
    }

    private Informasi(Parcel in) {
        this.id = in.readString();
        this.judul = in.readString();
        this.isi = in.readString();
        this.jampost = in.readString();
    }

    @Override

```

```

    public int describeContents() {
        return 0;
    }

    @Override
    public void writeToParcel(Parcel dest, int flags) {

    }

    public static final Parcelable.Creator<Informasi> CREATOR = new
    Parcelable.Creator<Informasi>() {
        @Override
        public Informasi createFromParcel(Parcel source) {
            return new Informasi(source);
        }

        @Override
        public Informasi[] newArray(int size) {
            return new Informasi[0];
        }
    };
}

```

DetilInformasiActivity.java

Setelah menuliskan kode program pada AppController.java, kemudian buat file java lagi untuk DetilInformasiActivity.java. File ini digunakan untuk melihat detil sebuah informasi. Syntax nya sebagai berikut :

```

package lab.zayed.com.broadcast;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;

public class DetilInformasiActivity extends AppCompatActivity {

    TextView textJUDUL , textISI ,textJAMPOST;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detil_informasi);

        textJUDUL = (TextView) findViewById(R.id.textJUDUL);
        textISI = (TextView) findViewById(R.id.textISI);
        textJAMPOST = (TextView) findViewById(R.id.textJAMPOST);

        Intent in = getIntent();
        textJUDUL.setText(in.getStringExtra("judul"));
        textISI.setText(in.getStringExtra("isi"));
        textJAMPOST.setText(in.getStringExtra("jampost"));
    }
}

```

ListBroadCastActivity.java

Setelah membuat file DetilInformasiActivity.java buatlah kembali file ListBroadCastActivity.java. File ini digunakan untuk menampilkan daftar dari seluruh informasi. Syntax nya sebagai berikut :

```
package lab.zayed.com.broadcast;

import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.Handler;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.DefaultItemAnimator;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.util.Log;
import android.view.GestureDetector;
import android.view.MotionEvent;
import android.view.View;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.VolleyLog;
import com.android.volley.toolbox.JsonArrayRequest;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.List;

public class ListBroadCastActivity extends AppCompatActivity {

    //private ProgressDialog pDialog;
    private List<Informasi> informasiList;
    private String urlJsonArray = "http://172.25.87.79:81/jav/";
    private static String TAG = MainActivity.class.getSimpleName();
    private Informasi infor;
    private ProgressBar progressBar;
    private RecyclerView rv;
    private recyclerInformasi ri;
    private LinearLayoutManager linearLayoutManager;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_list_broad_cast2);
        progressBar = (ProgressBar) findViewById(R.id.progresBar);
        rv = (RecyclerView) findViewById(R.id.recyclerView);
```

```

        FloatingActionButton fab = (FloatingActionButton)
findViewById(R.id.fab);
        informasiList = new ArrayList<>();

        linearLayoutManager = new LinearLayoutManager(this);
        ri = new RecyclerView(informasiList);

        rv.setLayoutManager(linearLayoutManager);
        rv.setAdapter(ri);

        makeJsonArrayRequest();
        loadData();

        /*pDialog = new ProgressDialog(this);
        pDialog.setMessage("Please wait...");
        pDialog.setCancelable(false);
        hideDialog();*/

        Intent intent = getIntent();
        if(intent.getStringExtra("username").equals("user")) {
            fab.setVisibility(View.GONE);
        }

        fab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                /*Snackbar.make(view, "Replace with your own action",
                Snackbar.LENGTH_LONG)
                .setAction("Action", null).show();*/
                //startActivity(new Intent(ListBroadCastActivity.this,
                SendBroadCastActivity.class));
                //informasiList.clear();
                //ri.notifyItemRangeRemoved(0, informasiList.size());
                startActivityForResult(new
                Intent(getApplicationContext(), SendBroadCastActivity.class
                ), 1);
            }
        });
    }

    private void makeJsonArrayRequest() {
        //showDialog();
        JsonRequest req = new JsonRequest(urlJsonArray,
        new Response.Listener<JSONArray>() {
            @Override
            public void onResponse(JSONArray response) {
                Log.d(TAG, "Response : " + response.toString());
                try {
                    // Parsing json array response
                    // loop through each json object
                    for (int i = 0; i < response.length(); i++) {
                        JSONObject jsonInformasi = (JSONObject)
response.get(i);

                        infor = new
                Informasi(jsonInformasi.getString("id"), jsonInformasi.getString("judul"),
                jsonInformasi.getString("isi"), jsonInformasi.getString("jampost"));
                        informasiList.add(infor);
                    }
                    //ri.addAll(informasiList);
                    //ri.setLoading(false);
                } catch (JSONException e) {

```

```

        e.printStackTrace();
        Toast.makeText(getApplicationContext(), "Error: " + e.getMessage(), Toast.LENGTH_LONG).show();
    }
    //hideDialog();
    progressBar.setVisibility(View.GONE);
}
}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        VolleyLog.d(TAG, "Error: " + error.getMessage());
        Toast.makeText(getApplicationContext(), error.getMessage(),
Toast.LENGTH_SHORT).show();
        progressBar.setVisibility(View.GONE);
    }
});

// Adding request to request queue
AppController.getInstance().addToRequestQueue(req);
}

/*private void showDialog() {
    if(!pDialog.isShowing()) {
        pDialog.show();
    }
}

private void hideDialog() {
    if (pDialog.isShowing()) {
        pDialog.dismiss();
    }
}*/

@Override
protected void onResume() {
    super.onResume();
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent
data) {
    //super.onActivityResult(requestCode, resultCode, data);
    if(resultCode == RESULT_OK){
        //Update List
        //Toast.makeText(this, "MASUK RESULT OK",
Toast.LENGTH_SHORT).show();
        progressBar.setVisibility(View.VISIBLE);

        //makeJsonArrayRequest();
    }

    /*else if (resultCode == RESULT_CANCELED) {
        //Do nothing?
        //Toast.makeText(this, "MASUK RESULT CANCELED",
Toast.LENGTH_SHORT).show();
    }*/
}

private void loadData() {
    ri = new recyclerInformasi(informasiList);
    rv.setHasFixedSize(true);
}

```

```

        RecyclerView.LayoutManager mLayoutManager = new
LinearLayoutManager(getApplicationContext());
        rv.setLayoutManager(mLayoutManager);
        //mRecyclerView.addItemDecoration(new DividerItemDecoration(this,
LinearLayoutManager.VERTICAL));
        rv.setItemAnimator(new DefaultItemAnimator());
        rv.setAdapter(ri);

        rv.setOnItemClickListener(new
RecyclerViewTouchListener(getApplicationContext(), rv, new
ClickListener() {
            @Override
            public void onClick(View view, int position) {
                Informasi infor = informasiList.get(position);
                //Toast.makeText(ListBroadCastActivity.this,
infor.getIsi(), Toast.LENGTH_SHORT).show();

                Intent in = new Intent(ListBroadCastActivity.this,
DetailInformasiActivity.class);
                in.putExtra("judul", infor.getJudul());
                in.putExtra("isi", infor.getIsi());
                in.putExtra("jampost", infor.getJampost());
                startActivity(in);
            }

            @Override
            public void onLongClick(View view, int position) {

            }
        }));

        ri.notifyDataSetChanged();
    }

    public interface ClickListener {
        void onClick(View view, int position);
        void onLongClick(View view, int position);
    }

    public static class RecyclerViewTouchListener implements
RecyclerView.OnItemTouchListener {

        private GestureDetector gestureDetector;
        private ListBroadCastActivity.ClickListener clickListener;

        public RecyclerViewTouchListener(Context context, final RecyclerView
recyclerView, final ListBroadCastActivity.ClickListener clickListener) {
            this.clickListener = clickListener;
            gestureDetector = new GestureDetector(context, new
GestureDetector.SimpleOnGestureListener() {
                @Override
                public boolean onSingleTapUp(MotionEvent e) {
                    return true;
                }

                @Override
                public void onLongPress(MotionEvent e) {
                    View child = recyclerView.findViewById(e.getX(),
e.getY());

                    if(child != null && clickListener != null) {
                        clickListener.onLongClick(child,

```

```

recyclerView.getChildAdapterPosition(child));
        }
    });
}

@Override
public boolean onInterceptTouchEvent(RecyclerView rv, MotionEvent
e) {
    View child = rv.findChildViewUnder(e.getX(), e.getY());
    if(child != null && clickListener != null &&
gestureDetector.onTouchEvent(e)) {
        //clickListener.onClick(child,
rv.getChildAdapterPosition(child));
        clickListener.onClick(child, rv.getChildPosition(child));
    }
    return false;
}

@Override
public void onTouchEvent(RecyclerView rv, MotionEvent e) {

}

@Override
public void onRequestDisallowInterceptTouchEvent(boolean
disallowIntercept) {

}
}
}

```

MainActivity.java

Pada MainActivity.java digunakan untuk melakukan pengecekan login. Syntaxnya sebagai berikut :

```

package lab.zayed.com.broadcast;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements
View.OnClickListener {

    EditText editTextUsername;
    EditText editTextPassword;
    Button btnLogin;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}

```



```

        editTextUsername = (EditText) findViewById(R.id.editUserName);
        editTextPassword = (EditText) findViewById(R.id.editPassword);
        btnLogin = (Button) findViewById(R.id.btnLogin);
        btnLogin.setOnClickListener(this);
    }

    @Override
    public void onClick(View v) {
        switch (v.getId()) {
            case R.id.btnLogin:
                Intent intent = new Intent(MainActivity.this,
ListBroadcastActivity.class);

                if(editTextUsername.getText().toString().trim().equals("mhs") &&
editTextPassword.getText().toString().trim().equals("12345")) {
                    intent.putExtra("username",
editTextUsername.getText().toString().trim());
                    intent.putExtra("password",
editTextPassword.getText().toString().trim());
                    startActivity(intent);
                }

                else if
(editTextUsername.getText().toString().trim().equals("user") &&
editTextPassword.getText().toString().trim().equals("biasa")) {
                    intent.putExtra("username",
editTextUsername.getText().toString().trim());
                    intent.putExtra("password",
editTextPassword.getText().toString().trim());
                    startActivity(intent);
                }

                else {
                    Toast.makeText(this, "GAGAL LOGIN",
Toast.LENGTH_SHORT).show();
                }

                break;
            default:
                break;
        }
    }
}

```

recyclerInformasi.java

Setelah itu buatlah File recyclerInformasi.java yang digunakan untuk memberikan id yang diperlukan pada list informasi. Syntax nya sebagai berikut :

```

package lab.zayed.com.broadcast;

import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

```

```

import java.util.List;

/**
 * Created by Lab on 14/03/2017.
 */

public class recyclerInformasi extends
RecyclerView.Adapter<recyclerInformasi.myViewHolder> {

    private List<Informasi> informasiList;

    @Override
    public myViewHolder onCreateViewHolder(ViewGroup parent, int viewType)
    {
        View itemView =
LayoutInflater.from(parent.getContext()).inflate(R.layout.rv_informasi,
parent, false);
        return new myViewHolder(itemView);
    }

    @Override
    public void onBindViewHolder(myViewHolder holder, int position) {
        Informasi m = informasiList.get(position);
        holder.textViewID.setText(m.getId());
        holder.textViewJudul.setText(m.getJudul());
        holder.textViewIsi.setText(m.getIsi());
        holder.textViewJamPost.setText(m.getJampost());
    }

    @Override
    public int getItemCount() {
        return informasiList.size();
    }

    public recyclerInformasi(List<Informasi> informasiList) {
        this.informasiList = informasiList;
    }

    public class myViewHolder extends RecyclerView.ViewHolder {
        public TextView textViewID, textViewJudul, textViewIsi,
textViewJamPost;

        public myViewHolder(View itemView) {
            super(itemView);

            textViewID = (TextView) itemView.findViewById(R.id.textID);
            textViewJudul = (TextView)
itemView.findViewById(R.id.textJUDUL);
            textViewIsi = (TextView) itemView.findViewById(R.id.textISI);
            textViewJamPost = (TextView)
itemView.findViewById(R.id.textTANGGAL);
        }
    }
}

```

SendBroadCastActivity.java

Setelah itu buatlah sebuah file dengan nama SendBroadCastActivity.java yang digunakan untuk mengirim pesan yang akan di-broadcast. Syntaxnya sebagai berikut :

```

package lab.zayed.com.broadcast;

import android.app.ProgressDialog;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.android.volley.Request;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;

import java.util.Calendar;
import java.util.HashMap;
import java.util.Map;

public class SendBroadCastActivity extends AppCompatActivity implements
View.OnClickListener{

    EditText editTextJudul;
    EditText editTextISI;
    Button buttonSendMessage;
    String LINK_SEND_BROADCAST = "http://172.25.87.79:81/jav/insert.php";
    ProgressDialog pDialog;
    String TAG = getClass().getSimpleName();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_send_broad_cast);

        editTextJudul = (EditText) findViewById(R.id.editJudul);
        editTextISI = (EditText) findViewById(R.id.editIsiBroadCast);
        buttonSendMessage = (Button) findViewById(R.id.btnBroadCast);
        buttonSendMessage.setOnClickListener(this);

        pDialog = new ProgressDialog(this);
        pDialog.setCancelable(true);
    }

    @Override
    public void onClick(View v) {
        switch(v.getId()) {
            case R.id.btnBroadCast:
                Toast.makeText(this, "TELAH DI-BROADCAST",
                    Toast.LENGTH_SHORT).show();
                showDialog();
                StringRequest strSendBroadCast = new
                StringRequest(Request.Method.POST, LINK_SEND_BROADCAST, new
                Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {
                        Log.d(TAG, "Response: " + new
                        StringBuilder(response));
                    }
                }, new Response.ErrorListener() {

```

```

        @Override
        public void onErrorResponse(VolleyError error) {
            Log.e(TAG, "Error: " + error.getMessage());
        }
    }) {
        @Override
        protected Map<String, String> getParams() {
            // Posting parameters to login url
            Calendar c = Calendar.getInstance();
            Map<String, String> params = new HashMap<>();
            params.put("judul",
editTextJudul.getText().toString());
            params.put("isi",
editTextISI.getText().toString());
            params.put("jampost", new
StringBuilder(c.get(Calendar.HOUR_OF_DAY)+":"+c.get(Calendar.MINUTE)+":"+c.
get(Calendar.SECOND)).toString());
            return params;
        }
    };

AppController.getInstance().addToRequestQueue(strSendBroadCast);
hideDialog();
Intent returnIntent = new Intent();
setResult(RESULT_OK, returnIntent);
finish();
break;

default:
break;
}

}

private void showDialog() {
    if (!pDialog.isShowing()) pDialog.show();
}

private void hideDialog() {
    if (pDialog.isShowing()) pDialog.dismiss();
}
}

```

BAGIAN 5

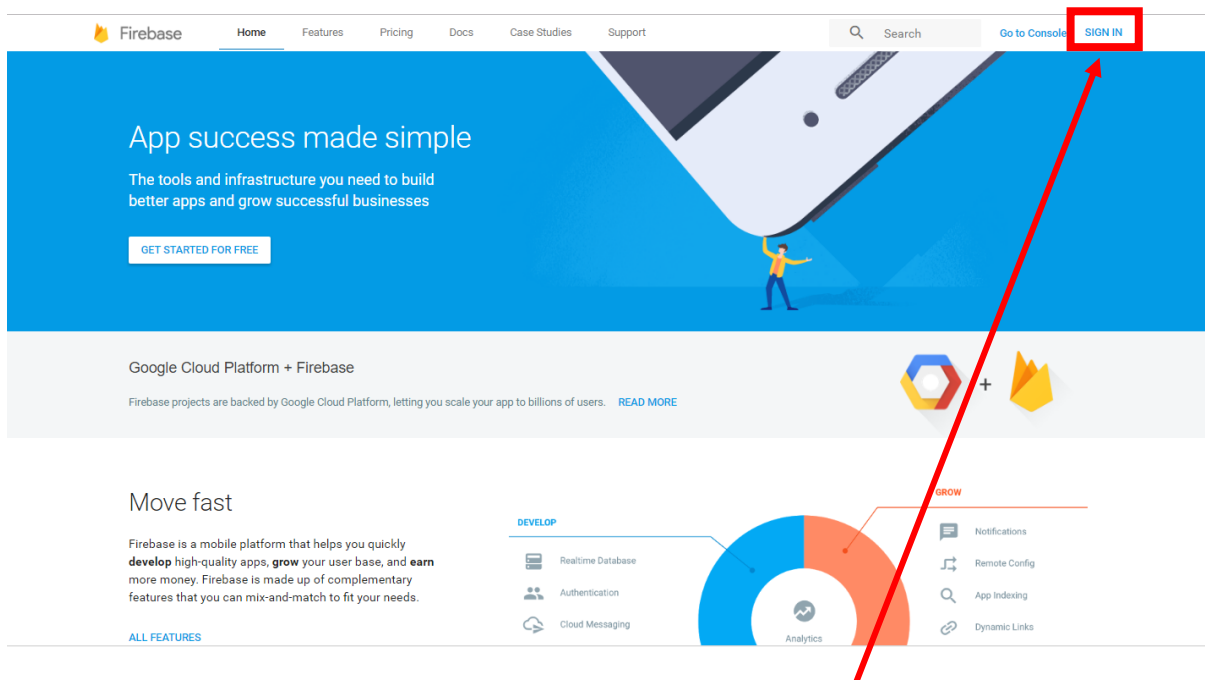
HEY.. I GET YOUR NOTIFICATION ON YOUR APPS!

Sebelum masuk ke dalam dunia coding lagi untuk membuat notifikasi pada Android, Anda harus mendaftarkan ID dari sebuah devices Android. Setiap Android memiliki ID yang berfungsi untuk mengenali devices. ID tersebut salah satunya dipakai untuk mengirimkan notifikasi agar tidak salah alamat.

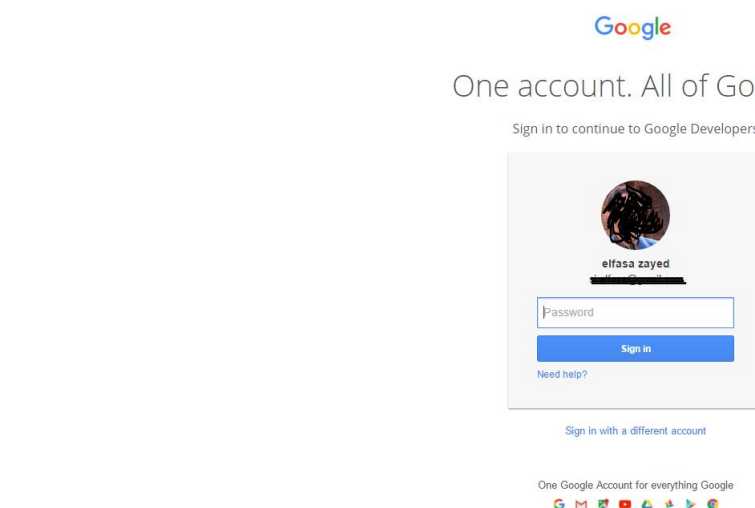
Cara untuk mendaftarkan ID tersebut di dalam database, terlebih dahulu anda harus daftarkan aplikasi anda di dalam “**Firebase**” pada link <https://firebase.google.com/>

Silahkan ikuti cara di bawah ini :

Silahkan menuju <https://firebase.google.com/> seperti di bawah ini :

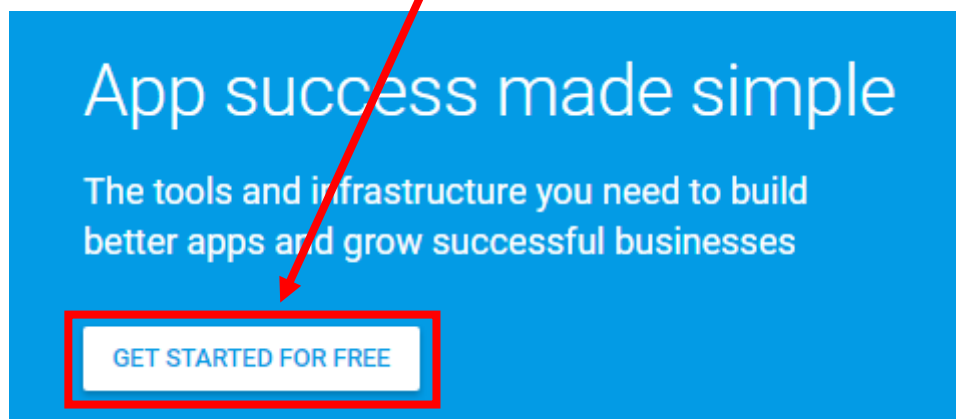


Setelah website tersebut selesai loading, kemudian cari tombol SIGN IN pada pojok kanan atas kemudian klik dan website tersebut akan menuju login akun Google Anda.



Setelah itu masukan email dan password anda masing-masing untuk login pada firebase. Setelah berhasil login halaman langsung re-direct pada halaman utama firebase, tetapi status Anda telah login (lihat halaman web pada pojok kanan atas).

Setelah itu Anda menekan tombol "Get Started For Free" untuk masuk mendaftarkan project Anda pada firebase.

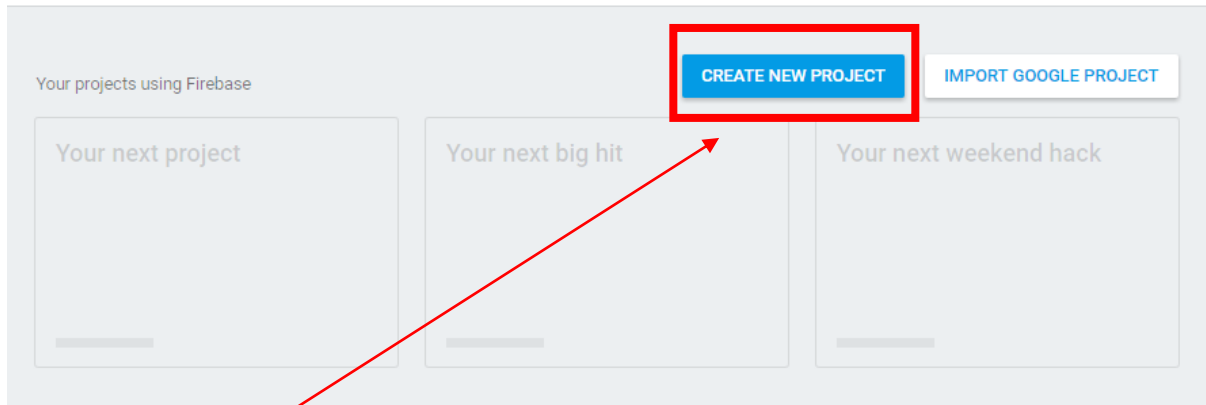
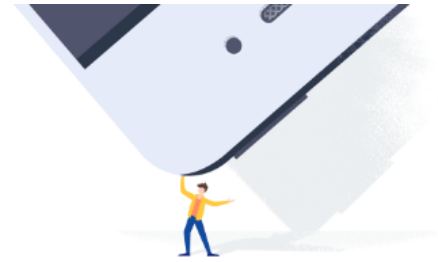


Setelah itu Anda akan dibawa pada halaman seperti di bawah ini :

Welcome to Firebase

Tools from Google for developing great apps, engaging with your users, and earning more through mobile ads. [Learn more](#)

[Documentation](#) <> [Sample code](#) ≡ [API reference](#) [Support](#)

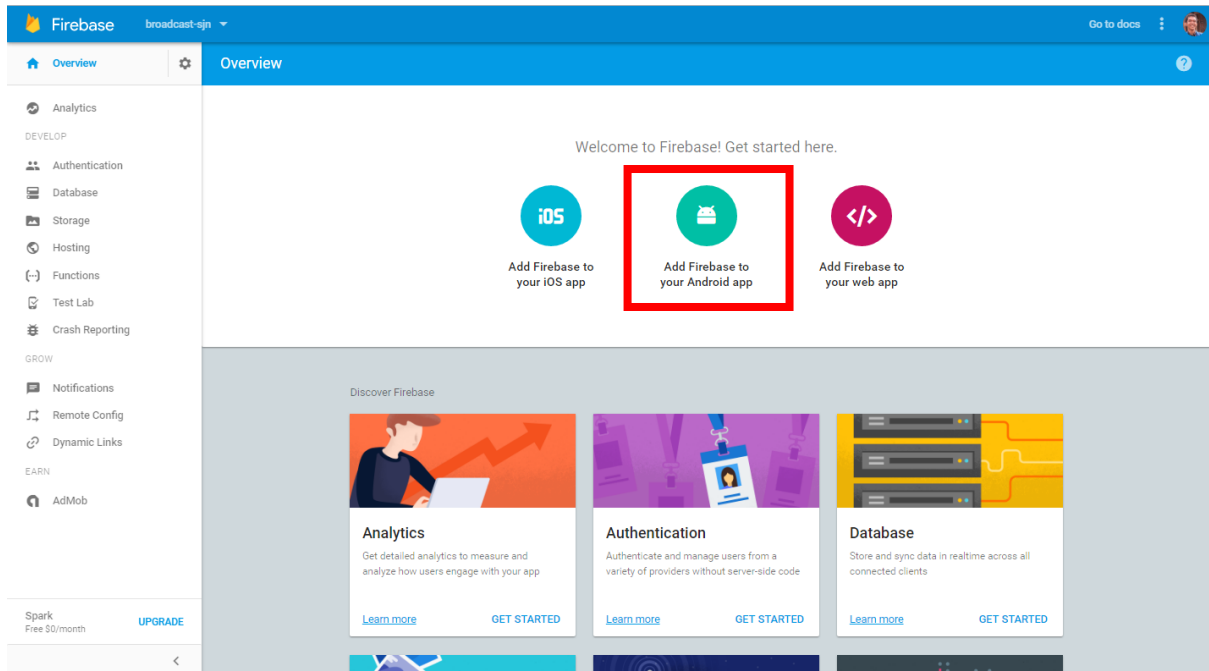


Pada halaman ini anda daftarkan project yang telah Anda buat dengan cara menekan tombol “Create New Project” kemudian akan menampilkan Form seperti di bawah ini :

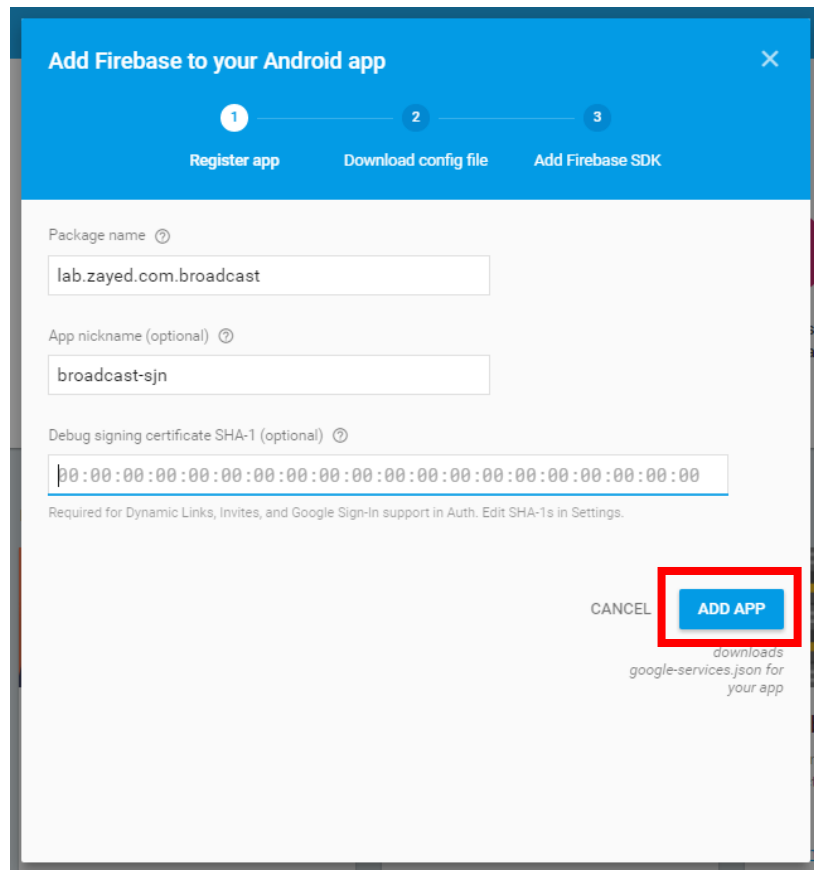
Isilah field yang pada form “Create a project” :

1. “Project Name” disesuaikan dengan nama project ketika Anda pertama kali membuat project.
2. “Country/region” isilah negara Anda masing-masing.

Setelah selesai semuanya diisi, kemudian klik tombol “Create Project” kemudian ditunggu sampai pembuatan project baru selesai. Setelah selesai akan menampilkan tampilan seperti di bawah ini :



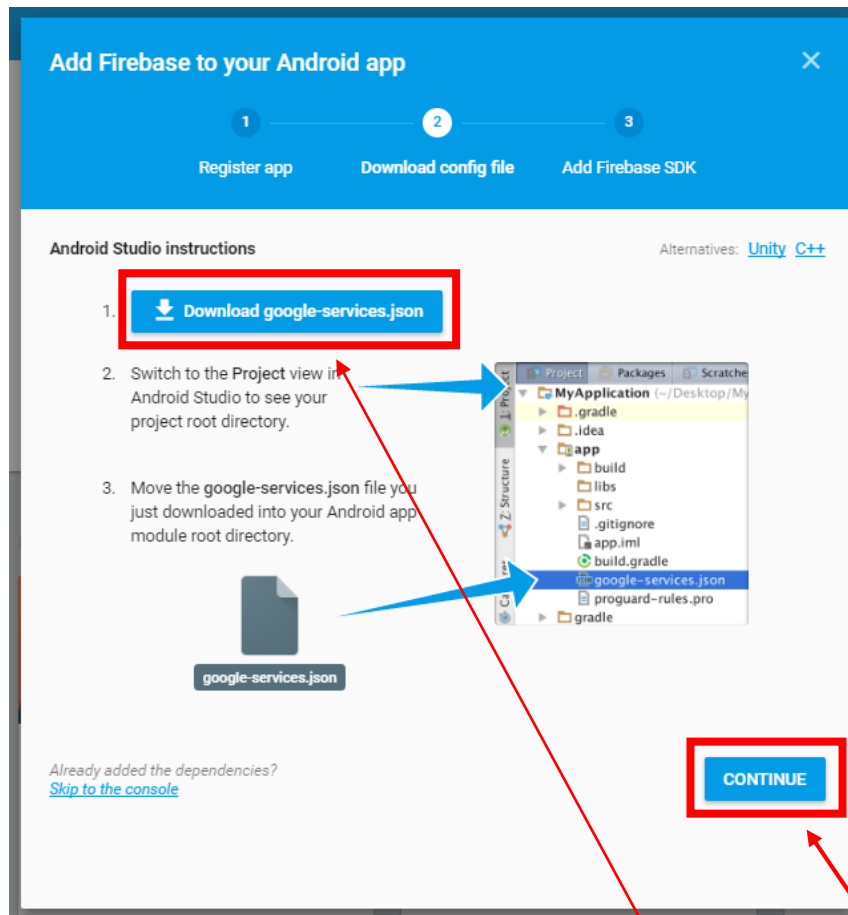
Setelah itu pillihlah “Add Firebase to your Android app” kemudian akan menampilkan form seperti di bawah ini :



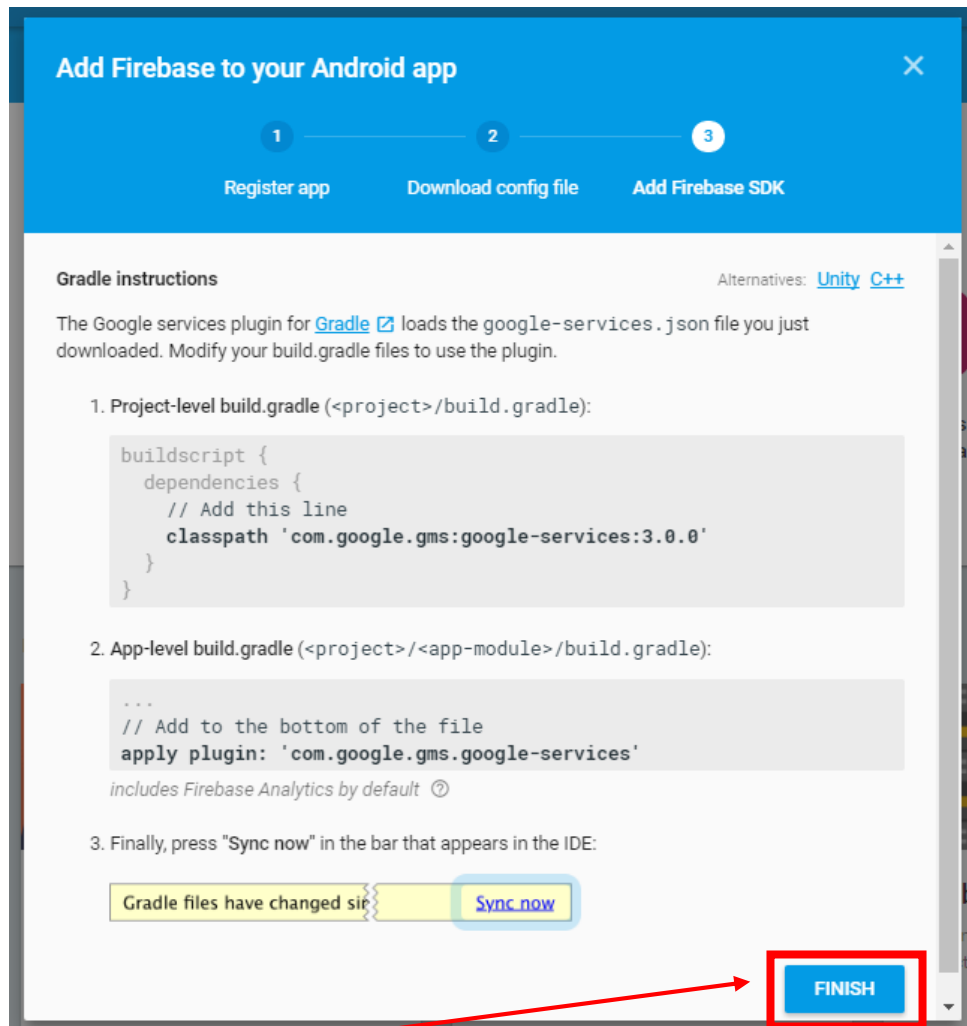
Kemudian isilah field yang kosong. Beberapa ada yang optional sehingga dapat dikosongkan.

1. Package Name diisi dengan nama “company domain”.
2. App Nickname dapat diisi dengan nama project Anda. Tetapi untuk App nickname boleh tidak diisi karena sifatnya “optional”.
3. Debug signing certificate SHA-1 boleh tidak diisin karena sifatnya optional.

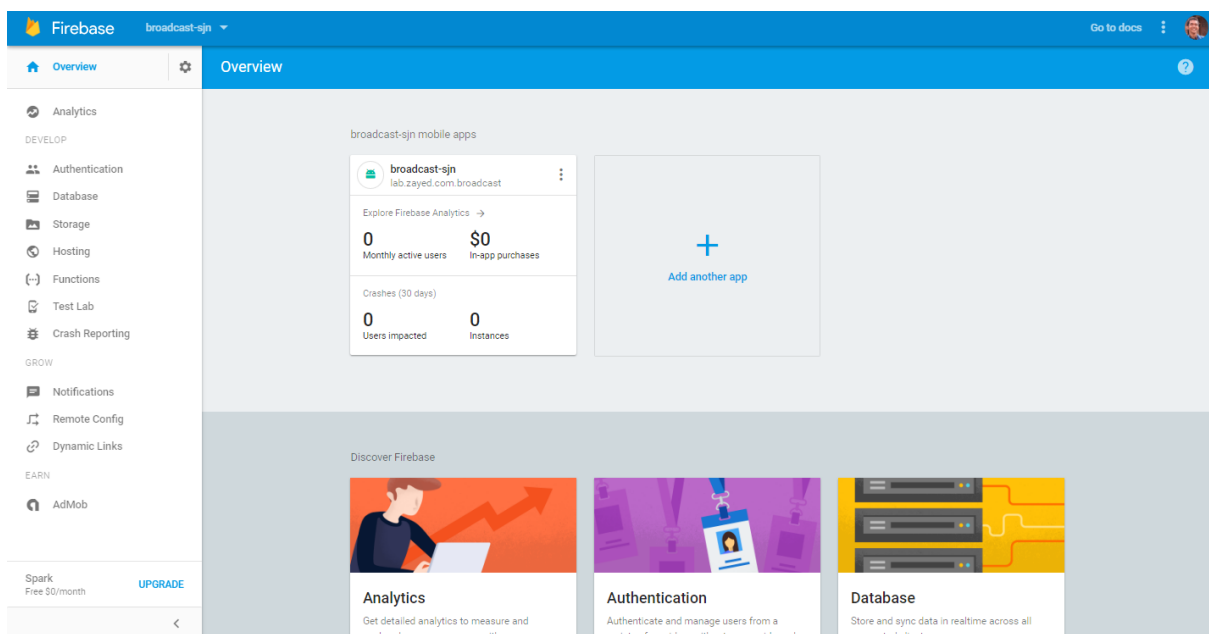
Setelah semua field terlewati kemudian klik tombol “ADD APP” kemudian Anda langsung diarahkan pada form berikutnya seperti form di bawah ini :



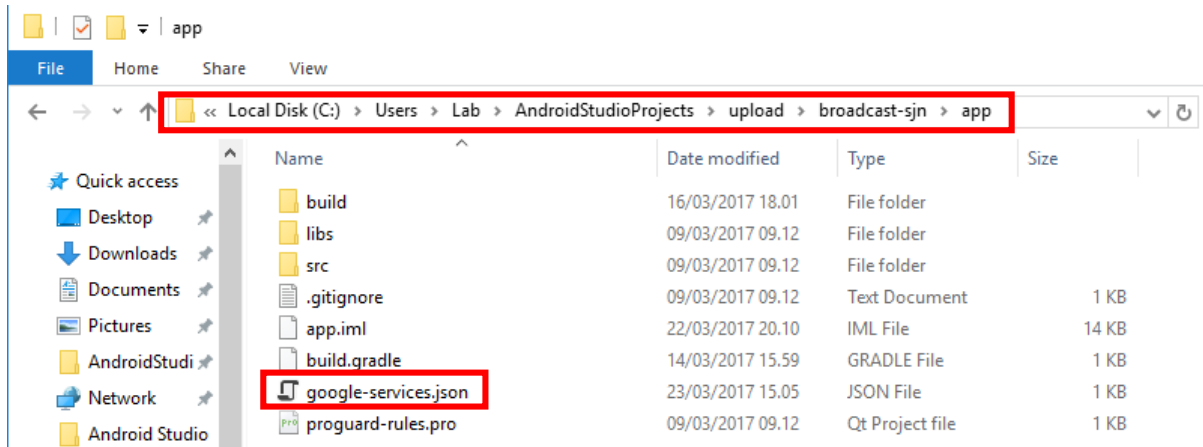
Setelah tampil seperti form di atas, kemudian pilihlah tombol “Download google-service-json” yang nantinya langsung akan unduh file berformat JSON. Setelah itu klik tombol “CONTINUE”. Setelah itu langsung menuju form selanjutnya seperti di bawah ini :



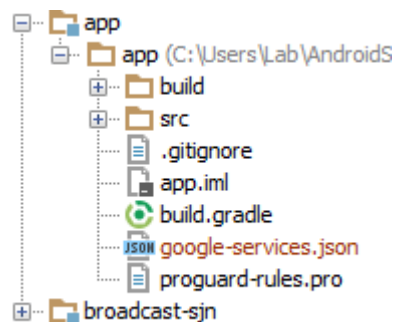
Setelah itu klik "FINISH". Jika telah selesai mendaftarkan aplikasi buatan Anda pada google firebase, maka akan menampilkan web seperti ini :



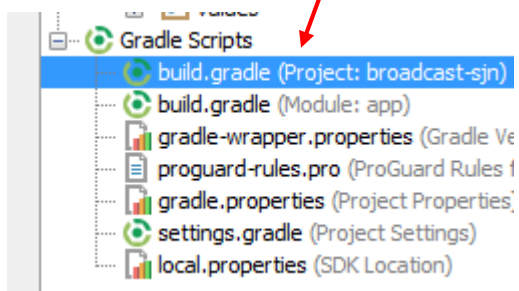
File yang telah di download yang berformat JSON bernama “google-services.json” tadi akan dimasukkan ke dalam project yang telah Anda buat. Masukkan ke dalam foder “app”. Folder “app” sendiri ada pada “project-anda -> app”.



Jika telah menyalin file “google-services.json” pada structure file project Android studio seperti di bawah ini :



Setelah google-services.json telah dimasukkan ke dalam project, mari kita menuju build.gradle(project:nama_project_anda). Buka lah file tersebut dengan cara double klik pada file tersebut.



Sehingga akan menampilkan sebuah dependencies.

Setelah itu tambahkan syntax berikut ini :

```
classpath 'com.google.gms:google-services:3.0.0'
```

Pada dependencies. Contohnya seperti di bawah ini :

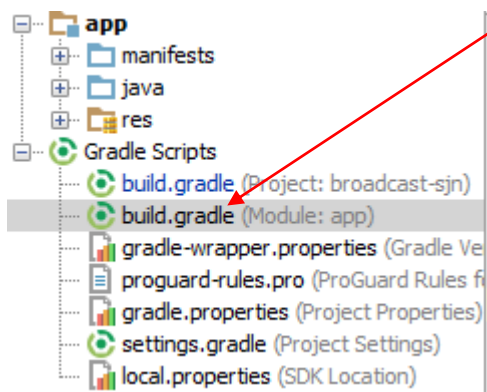
```
buildscript {
    repositories {
        jcenter()
    }
    dependencies {
        classpath 'com.android.tools.build:gradle:2.2.3'
        classpath 'com.google.gms:google-services:3.0.0'
        // NOTE: Do not place your application dependencies here; they belong
        // in the individual module build.gradle files
    }
}

allprojects {
    repositories {
        jcenter()
    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}
```

Setelah itu klik “sync now” yang berada pada pojok atas dengan tujuan agar library diunduh melalui internet.

Setelah ditulis, mari kita masuk lagi ke dalam “build.gradle(module: app)” kemudian dibuka file tersebut dengan cara double klik :



Setelah itu tambahkan syntax di bawah ini di dalam dependencies :

```
compile 'com.google.firebase:firebase-messaging:9.6.0'
```

kemudian tambahkan syntax ini di bawahnya dependencies :

```
apply plugin: 'com.google.gms.google-services'
```

Contohnya sebagai berikut :

```
dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {
        exclude group: 'com.android.support', module: 'support-annotations'
    })
    compile 'com.android.support:appcompat-v7:25.1.0'
    compile 'com.android.support:design:25.1.0'
    compile 'com.android.volley:volley:1.0.0'
    testCompile 'junit:junit:4.12'

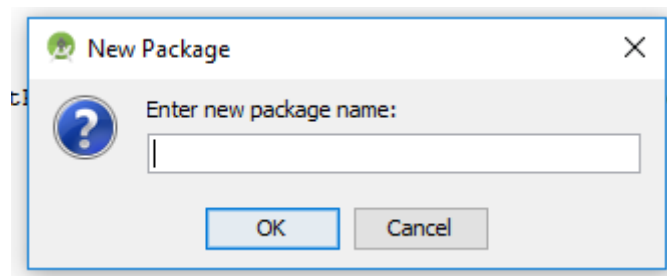
    compile 'com.google.firebase:firebase-messaging:9.6.0'

    apply plugin: 'com.google.gms.google-services'
}
```

Kemudian jangan lupa klik “*sync now*” pada pojok kanan atas *editor*.

Setelah semuanya selesai dilakukan di atas, buatlah packages pada folder java yang terdiri dari “app”, “service”, dan “util”. Caranya sebagai berikut :

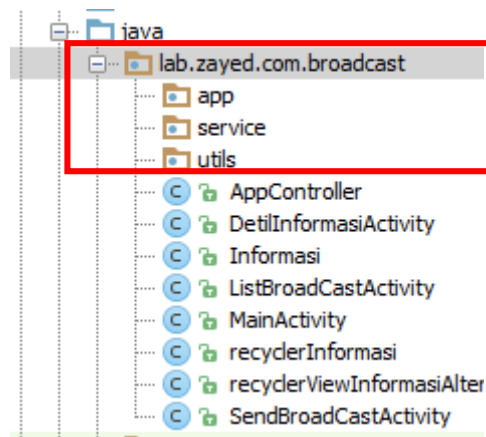
Klik kanan pada folder yang ada di dalam java misal nama packages nya “lab.zayed.com.broadcast” -> ini di klik kanan. kemudian pilih menu “new” kemudian pilih “package” setelah itu akan tampil seperti di bawah ini :



Isilah nama “package” sesuai dengan nama yang di atas yaitu :

1. App
2. Service
3. Utils

Sehingga tampil nama-nama package seperti di bawah ini :



Setelah itu, buatlah file java terbaru dengan nama “config.java” pada package “app”.

Isilah file java tersebut seperti di bawah ini :

```
public class config {

    // global topic to receive app wide push notifications
    public static final String TOPIC_GLOBAL = "global";

    // broadcast receiver intent filters
    public static final String REGISTRATION_COMPLETE =
"registrationComplete";
    public static final String PUSH_NOTIFICATION = "pushNotification";

    // id to handle the notification in the notification tray
    public static final int NOTIFICATION_ID = 100;
    public static final int NOTIFICATION_ID_BIG_IMAGE = 101;

    public static final String SHARED_PREF = "ah_firebase";
}
```

Setelah itu buatlah file java terbaru dengan nama “NotificationUtils.java” pada package “utils”. Isilah file java tersebut seperti di bawah ini :

```
package lab.zayed.com.broadcast.utils;

import android.app.ActivityManager;
import android.app.Notification;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.ComponentName;
import android.content.ContentResolver;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.media.Ringtone;
import android.media.RingtoneManager;
import android.net.Uri;
import android.os.Build;
import android.support.v4.app.NotificationCompat;
```

```

import android.text.Html;
import android.text.TextUtils;
import android.util.Patterns;

import java.io.IOException;
import java.io.InputStream;
import java.net.HttpURLConnection;
import java.net.URL;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.List;

import lab.zayed.com.broadcast.R;
import lab.zayed.com.broadcast.app.config;

/**
 * Created by Lab on 24/03/2017.
 */

public class NotificationUtils {
    private static String TAG = NotificationUtils.class.getSimpleName();

    private Context mContext;

    public NotificationUtils(Context mContext) {
        this.mContext = mContext;
    }

    public void showNotificationMessage(String title, String message,
String timeStamp, Intent intent) {
        showNotificationMessage(title, message, timeStamp, intent, null);
    }

    public void showNotificationMessage(final String title, final String
message, final String timeStamp, Intent intent, String imageUrl) {
        // Check for empty push message
        if (TextUtils.isEmpty(message))
            return;

        // notification icon
        final int icon = R.mipmap.ic_launcher;

        intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
Intent.FLAG_ACTIVITY_SINGLE_TOP);
        final PendingIntent resultPendingIntent =
            PendingIntent.getActivity(
                mContext,
                0,
                intent,
                PendingIntent.FLAG_CANCEL_CURRENT
            );

        final NotificationCompat.Builder mBuilder = new
NotificationCompat.Builder(
            mContext);

        final Uri alarmSound =
Uri.parse(ContentResolver.SCHEME_ANDROID_RESOURCE
+ "://" + mContext.getPackageName() + "/raw/notification");

```



```

        if (!TextUtils.isEmpty(imageUrl)) {

            if (imageUrl != null && imageUrl.length() > 4 &&
Patterns.WEB_URL.matcher(imageUrl).matches()) {

                Bitmap bitmap = getBitmapFromURL(imageUrl);

                if (bitmap != null) {
                    showBigNotification(bitmap, mBuilder, icon, title,
message, timeStamp, resultPendingIntent, alarmSound);
                } else {
                    showSmallNotification(mBuilder, icon, title, message,
timeStamp, resultPendingIntent, alarmSound);
                }
            } else {
                showSmallNotification(mBuilder, icon, title, message,
timeStamp, resultPendingIntent, alarmSound);
                playNotificationSound();
            }
        }

        private void showSmallNotification(NotificationCompat.Builder mBuilder,
int icon, String title, String message, String timeStamp, PendingIntent
resultPendingIntent, Uri alarmSound) {

            NotificationCompat.InboxStyle inboxStyle = new
NotificationCompat.InboxStyle();

            inboxStyle.addLine(message);

            Notification notification;
            notification =
mBuilder.setSmallIcon(icon).setTicker(title).setWhen(0)
                .setAutoCancel(true)
                .setContentTitle(title)
                .setContentIntent(resultPendingIntent)
                .setSound(alarmSound)
                .setStyle(inboxStyle)
                .setWhen(getTimeMillisec(timeStamp))
                .setSmallIcon(R.mipmap.ic_launcher)

.setLargeIcon(BitmapFactory.decodeResource(mContext.getResources(), icon))
                .setContentText(message)
                .build();

            NotificationManager notificationManager = (NotificationManager)
mContext.getSystemService(Context.NOTIFICATION_SERVICE);
            notificationManager.notify(config.NOTIFICATION_ID, notification);
        }

        private void showBigNotification(Bitmap bitmap,
NotificationCompat.Builder mBuilder, int icon, String title, String
message, String timeStamp, PendingIntent resultPendingIntent, Uri
alarmSound) {
            NotificationCompat.BigPictureStyle bigPictureStyle = new
NotificationCompat.BigPictureStyle();
            bigPictureStyle.setBigContentTitle(title);
            bigPictureStyle.setSummaryText(Html.fromHtml(message).toString());

```

```

        bigPictureStyle.bigPicture(bitmap);
        Notification notification;
        notification =
mBuilder.setSmallIcon(icon).setTicker(title).setWhen(0)
        .setAutoCancel(true)
        .setContentTitle(title)
        .setContentIntent(resultPendingIntent)
        .setSound(alarmSound)
        .setStyle(bigPictureStyle)
        .setWhen(getTimeMillisec(timestamp))
        .setSmallIcon(R.mipmap.ic_launcher)

.setLargeIcon(BitmapFactory.decodeResource(mContext.getResources(), icon))
        .setContentText(message)
        .build();

        NotificationManager notificationManager = (NotificationManager)
mContext.getSystemService(Context.NOTIFICATION_SERVICE);
        notificationManager.notify(config.NOTIFICATION_ID_BIG_IMAGE,
notification);
    }

    /**
     * Downloading push notification image before displaying it in
     * the notification tray
     */
    public Bitmap getBitmapFromURL(String strURL) {
        try {
            URL url = new URL(strURL);
            HttpURLConnection connection = (HttpURLConnection)
url.openConnection();
            connection.setDoInput(true);
            connection.connect();
            InputStream input = connection.getInputStream();
            Bitmap myBitmap = BitmapFactory.decodeStream(input);
            return myBitmap;
        } catch (IOException e) {
            e.printStackTrace();
            return null;
        }
    }

    // Playing notification sound
    public void playNotificationSound() {
        try {
            Uri alarmSound =
Uri.parse(ContentResolver.SCHEME_ANDROID_RESOURCE
                + "://" + mContext.getPackageName() +
"/raw/notification");
            Ringtone r = RingtoneManager.getRingtone(mContext, alarmSound);
            r.play();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    /**
     * Method checks if the app is in background or not
     */
    public static boolean isAppIsInBackground(Context context) {
        boolean isInBackground = true;

```

```

        ActivityManager am = (ActivityManager)
context.getSystemService(Context.ACTIVITY_SERVICE);
        if (Build.VERSION.SDK_INT > Build.VERSION_CODES.KITKAT_WATCH) {
            List<ActivityManager.RunningAppProcessInfo> runningProcesses =
am.getRunningAppProcesses();
            for (ActivityManager.RunningAppProcessInfo processInfo :
runningProcesses) {
                if (processInfo.importance ==
ActivityManager.RunningAppProcessInfo.IMPORTANCE_FOREGROUND) {
                    for (String activeProcess : processInfo.pkgList) {
                        if (activeProcess.equals(context.getPackageName()))
{
                            isInBackground = false;
                        }
                    }
                }
            } else {
                List<ActivityManager.RunningTaskInfo> taskInfo =
am.getRunningTasks(1);
                ComponentName componentInfo = taskInfo.get(0).topActivity;
                if
(componentInfo.getPackageName().equals(context.getPackageName())) {
                    isInBackground = false;
                }
            }

            return isInBackground;
        }

        // Clears notification tray messages
        public static void clearNotifications(Context context) {
            NotificationManager notificationManager = (NotificationManager)
context.getSystemService(Context.NOTIFICATION_SERVICE);
            notificationManager.cancelAll();
        }

        public static long getTimeMilliSec(String timeStamp) {
            SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd
HH:mm:ss");
            try {
                Date date = format.parse(timeStamp);
                return date.getTime();
            } catch (ParseException e) {
                e.printStackTrace();
            }
            return 0;
        }
    }
}

```

Setelah itu buatlah sebuah file dengan nama MyFirebaseInstanceIdService.java pada package "service". Isilah file tersebut seperti di bawah ini :

```

package lab.zayed.com.broadcast.service;

import android.content.Intent;
import android.content.SharedPreferences;
import android.support.v4.content.LocalBroadcastManager;

```

```

import android.util.Log;

import com.google.firebase.iid.FirebaseInstanceId;
import com.google.firebase.iid.FirebaseInstanceIdService;

import lab.zayed.com.broadcast.app.config;

/**
 * Created by Lab on 24/03/2017.
 */

public class MyFirebaseInstanceIdService extends FirebaseInstanceIdService
{
    private static final String TAG =
MyFirebaseInstanceIdService.class.getSimpleName();

    @Override
    public void onTokenRefresh() {
        super.onTokenRefresh();
        String refreshedToken =
FirebaseInstanceId.getInstance().getToken();

        // Saving reg id to shared preferences
        storeRegIdInPref(refreshedToken);

        // sending reg id to your server
        sendRegistrationToServer(refreshedToken);

        // Notify UI that registration has completed, so the progress
        indicator can be hidden.
        Intent registrationComplete = new
Intent(config.REGISTRATION_COMPLETE);
        registrationComplete.putExtra("token", refreshedToken);

        LocalBroadcastManager.getInstance(this).sendBroadcast(registrationComplete)
;
    }

    private void sendRegistrationToServer(final String token) {
        // sending gcm token to server
        Log.e(TAG, "sendRegistrationToServer: " + token);
    }

    private void storeRegIdInPref(String token) {
        SharedPreferences pref =
getApplicationContext().getSharedPreferences(config.SHARED_PREF, 0);
        SharedPreferences.Editor editor = pref.edit();
        editor.putString("regId", token);
        editor.commit();
    }
}

```

Setelah itu buatlah file “MyFirebaseMessagingService.java” pada package “service”. Isi file tersebut seperti di bawah ini :

```

package lab.zayed.com.broadcast.service;

import android.content.Context;

```

```

import android.content.Intent;
import android.support.v4.content.LocalBroadcastManager;
import android.text.TextUtils;
import android.util.Log;

import com.google.firebase.messaging.FirebaseMessagingService;
import com.google.firebase.messaging.RemoteMessage;

import org.json.JSONException;
import org.json.JSONObject;

import lab.zayed.com.broadcast.MainActivity;
import lab.zayed.com.broadcast.app.config;
import lab.zayed.com.broadcast.utils.NotificationUtils;

/**
 * Created by Lab on 24/03/2017.
 */

public class MyFirebaseMessagingService extends FirebaseMessagingService {
    private static final String TAG =
MyFirebaseMessagingService.class.getSimpleName();

    private NotificationUtils notificationUtils;

    @Override
    public void onMessageReceived(RemoteMessage remoteMessage) {
        Log.e(TAG, "From: " + remoteMessage.getFrom());

        if (remoteMessage == null)
            return;

        // Check if message contains a notification payload.
        if (remoteMessage.getNotification() != null) {
            Log.e(TAG, "Notification Body: " +
remoteMessage.getNotification().getBody());
            handleNotification(remoteMessage.getNotification().getBody());
        }

        // Check if message contains a data payload.
        if (remoteMessage.getData().size() > 0) {
            Log.e(TAG, "Data Payload: " +
remoteMessage.getData().toString());

            try {
                JSONObject json = new
JSONObject(remoteMessage.getData().toString());
                handleDataMessage(json);
            } catch (Exception e) {
                Log.e(TAG, "Exception: " + e.getMessage());
            }
        }

        private void handleNotification(String message) {
            if
(!NotificationUtils.isAppIsInBackground(getApplicationContext())) {
                // app is in foreground, broadcast the push message
                Intent pushNotification = new Intent(config.PUSH_NOTIFICATION);
                pushNotification.putExtra("message", message);
            }
        }
    }
}

```

```

LocalBroadcastManager.getInstance(this).sendBroadcast(pushNotification);

        // play notification sound
        NotificationUtils notificationUtils = new
NotificationUtils(getApplicationContext());
        notificationUtils.playNotificationSound();
    }else{
        // If the app is in background, firebase itself handles the
notification
    }
}

private void handleDataMessage(JSONObject json) {
    Log.e(TAG, "push json: " + json.toString());

    try {
        JSONObject data = json.getJSONObject("data");

        String title = data.getString("title");
        String message = data.getString("message");
        boolean isBackground = data.getBoolean("is_background");
        String imageUrl = data.getString("image");
        String timestamp = data.getString("timestamp");

        Log.e(TAG, "title: " + title);
        Log.e(TAG, "message: " + message);
        Log.e(TAG, "isBackground: " + isBackground);
        Log.e(TAG, "imageUrl: " + imageUrl);
        Log.e(TAG, "timestamp: " + timestamp);

        if
(!NotificationUtils.isAppIsInBackground(getApplicationContext())) {
            // app is in foreground, broadcast the push message
            Intent pushNotification = new
Intent(config.PUSH_NOTIFICATION);
            pushNotification.putExtra("message", message);

LocalBroadcastManager.getInstance(this).sendBroadcast(pushNotification);

            // play notification sound
            NotificationUtils notificationUtils = new
NotificationUtils(getApplicationContext());
            notificationUtils.playNotificationSound();
        } else {
            // app is in background, show the notification in
notification tray
            Intent resultIntent = new Intent(getApplicationContext(),
MainActivity.class);
            resultIntent.putExtra("message", message);

            // check for image attachment
            if (TextUtils.isEmpty(imageUrl)) {
                showNotificationMessage(getApplicationContext(), title,
message, timestamp, resultIntent);
            } else {
                // image is present, show notification with image

showNotificationMessageWithBigImage(getApplicationContext(), title,
message, timestamp, resultIntent, imageUrl);

```

```

    }
}
} catch (JSONException e) {
    Log.e(TAG, "Json Exception: " + e.getMessage());
} catch (Exception e) {
    Log.e(TAG, "Exception: " + e.getMessage());
}
}

/**
 * Showing notification with text only
 */
private void showNotificationMessage(Context context, String title,
String message, String timeStamp, Intent intent) {
    notificationUtils = new NotificationUtils(context);
    intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
    notificationUtils.showNotificationMessage(title, message,
timeStamp, intent);
}

/**
 * Showing notification with text and image
 */
private void showNotificationMessageWithBigImage(Context context,
String title, String message, String timeStamp, Intent intent, String
imageUrl) {
    notificationUtils = new NotificationUtils(context);
    intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
    notificationUtils.showNotificationMessage(title, message,
timeStamp, intent, imageUrl);
}
}

```

Setelah semua file telah ditulis seperti di atas, daftarkan service pada manifest.xml yang terletak pada folder “app” -> “manifest”-> “AndroidManifest.xml”. Cara daftarkan tuliskan syntaxnya seperti di bawah ini di dalam tag <application></application> :

```

<!-- Firebase Notifications -->
<service android:name=".service.MyFirebaseMessagingService">
    <intent-filter>
        <action android:name="com.google.firebase.MESSAGING_EVENT" />
    </intent-filter>
</service>

<service android:name=".service.MyFirebaseInstanceIdService">
    <intent-filter>
        <action android:name="com.google.firebase.INSTANCE_ID_EVENT" />
    </intent-filter>
</service>
<!-- ./Firebase Notifications -->

```

Dan juga tambahkan permission baru GET_TASKS di atas ataupun di bawah permssionnya INTERNET.

```
<uses-permission android:name="android.permission.GET_TASKS"/>
```

Jika telah selesai nanti akan tampil seperti di bawah ini :

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="lab.zayed.com.broadcast">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.GET_TASKS"/>
```

Untuk membuat sebuah notifikasi pada Android, setiap devices Android harus sudah ter-install google-play-service. Biasanya ketika Anda membeli sebuah SmartPhone berbasis Android sudah langsung ter-install google-play-service. Tujuannya adalah agar SmartPhone memiliki ID untuk mendapatkan Notifikasi.

Untuk melihat ID notifikasi tersebut, mari kita sama-sama menuliskan syntax pada MainActivity.java. Ubahlah syntax tersebut menjadi seperti di bawah ini :

Tambahkan syntax String TAG dan BroadcastReceiver di bawah btnLogin :

```
private String TAG = MainActivity.class.getSimpleName();
private BroadcastReceiver mRegistrationBroadcastReceiver;
```

Jadi nantinya seperti di bawah ini :

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener {

    EditText editTextUsername;
    EditText editTextPassword;
    Button btnLogin;
    private String TAG = MainActivity.class.getSimpleName();
    private BroadcastReceiver mRegistrationBroadcastReceiver;
```

Setelah itu buatlah prosedur displayFirebaseRegId() diluar void onCreate tetapi masih di dalam class MainActivity.java

```
private void displayFirebaseRegId() {
    SharedPreferences pref =
    getApplicationContext().getSharedPreferences(config.SHARED_PREF, 0);
    String regId = pref.getString("regId", null);
    Log.e(TAG, "Firebase reg id: " + regId);
}
```

Setelah itu di dalam method onCreate tuliskan syntax code seperti di bawah ini :


```

mRegistrationBroadcastReceiver = new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {

        // checking for type intent filter
        if (intent.getAction().equals(config.REGISTRATION_COMPLETE)) {
            // gcm successfully registered
            // now subscribe to `global` topic to receive app wide notifications
            FirebaseMessaging.getInstance().subscribeToTopic(config.TOPIC_GLOBAL);
            displayFirebaseRegId();

        } else if (intent.getAction().equals(config.PUSH_NOTIFICATION)) {
            // new push notification is received
            String message = intent.getStringExtra("message");
            Toast.makeText(getApplicationContext(), "Push notification: " +
message, Toast.LENGTH_LONG).show();
        }
    }
};

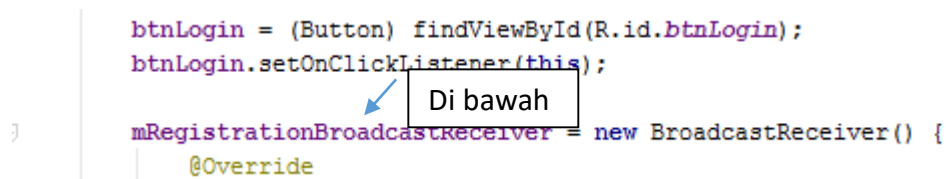
displayFirebaseRegId();

```

Tuliskan syntax tersebut di bawahnya

```
btnLogin.setOnClickListener(this);
```

Nanti hasilnya seperti di bawah ini :



```

    btnLogin = (Button) findViewById(R.id.btnLogin);
    btnLogin.setOnClickListener(this);

    mRegistrationBroadcastReceiver = new BroadcastReceiver() {
        @Override

```

Setelah itu tambahkan lagi method override onResume dan onPause di luarnya method onCreate tetapi masih di dalamnya class MainActivity.java :

```

@Override
protected void onResume() {
    super.onResume();

    // register GCM registration complete receiver
    LocalBroadcastManager.getInstance(this).registerReceiver(mRegistrationBroadcastReceiver,
        new IntentFilter(config.REGISTRATION_COMPLETE));

    // register new push message receiver
    // by doing this, the activity will be notified each time a new message arrives
    LocalBroadcastManager.getInstance(this).registerReceiver(mRegistrationBroadcastReceiver,
        new IntentFilter(config.PUSH_NOTIFICATION));

    // clear the notification area when the app is opened
    NotificationUtils.clearNotifications(getApplicationContext());
}

```

```

@Override
protected void onPause() {

LocalBroadcastManager.getInstance(this).unregisterReceiver(mRegistrationBroadcastReceiver);
    super.onPause();
}

```

Setelah itu cobalah compile aplikasi yang telah anda buat kemudian pada android studio tekan “alt + 6 “ untuk membuka android monitor yang bertujuan untuk melihat Firebase Reg ID yang telah dimiliki oleh device emulator masing-masing.

Nanti akan terlihat seperti di bawah ini :



Firebase “Reg ID” ini dimasukkan ke dalam database dengan cara menyalin ID tersebut ke dalam tabel “fcmregid”.

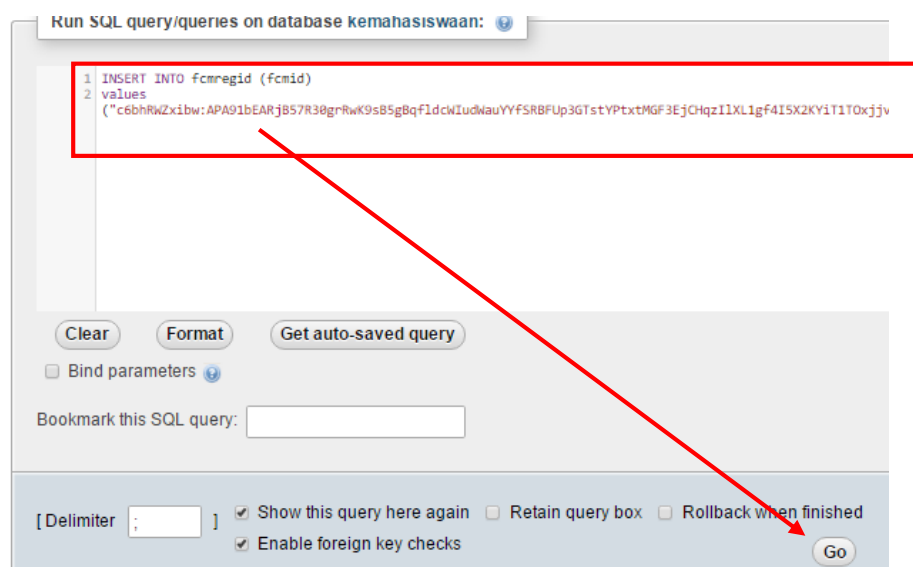
Silahkan ikuti syntax ini untuk memasukkan FCM REG ID :

```

INSERT INTO fcmregid (fcmid) values
("[TULISKAN_FCM_REG_ID_PADA_ANDROID_MONITOR_ANDA_MASING_MASING]");

```

Kemudian tulisan di atas masukkan ke dalam SQL box yang ada di phpmyadmin.



Kemudian Anda klik “Go”. Pastikan Memasukkan data ke dalam database berhasil.

Setelah itu mari menuju php kembali.

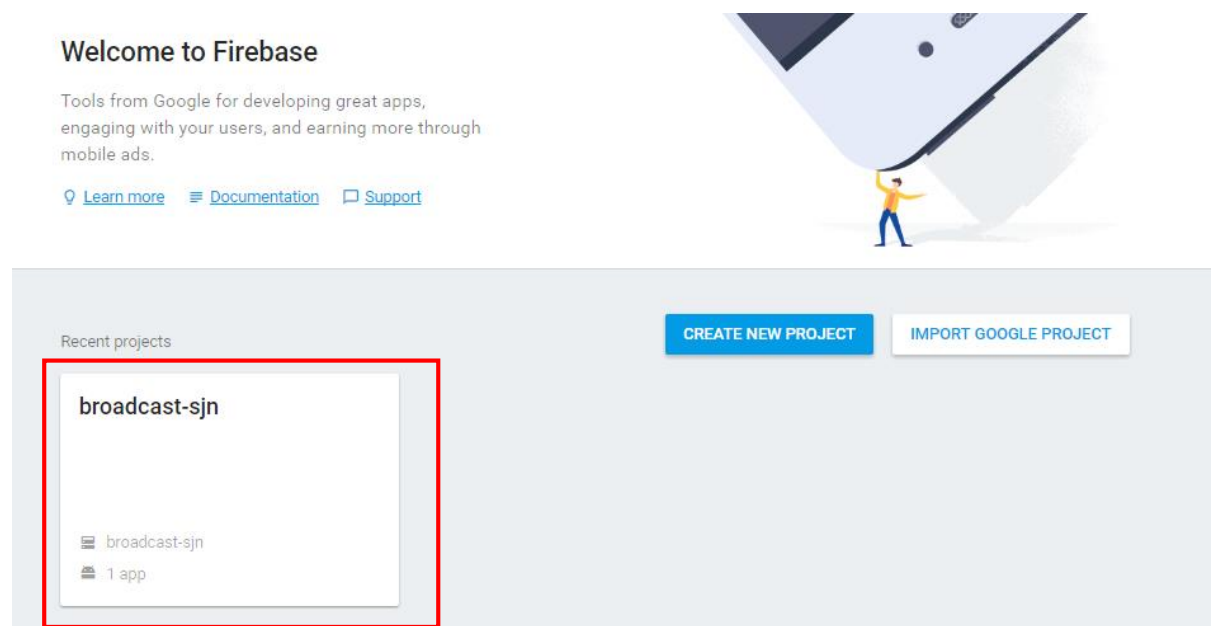
Pada folder “jav” yang ada di dalam folder “htdocs” buatlah file “config.php” yang isinya sebagai berikut :

```
<?php
// Firebase API Key
define('FIREBASE_API_KEY', '[MASUKKAN ID YANG ADA PADA google firebase]');
```

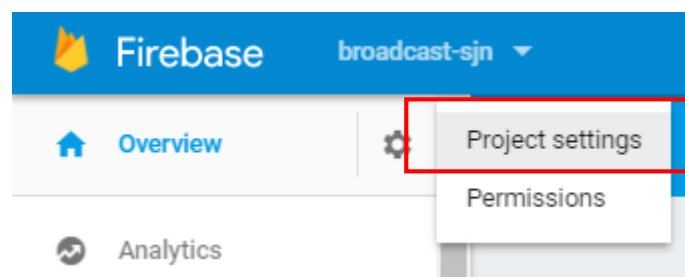
Untuk mendapatkan FIREBASE_API_KEY silahkan anda menuju

<https://console.firebase.google.com/>

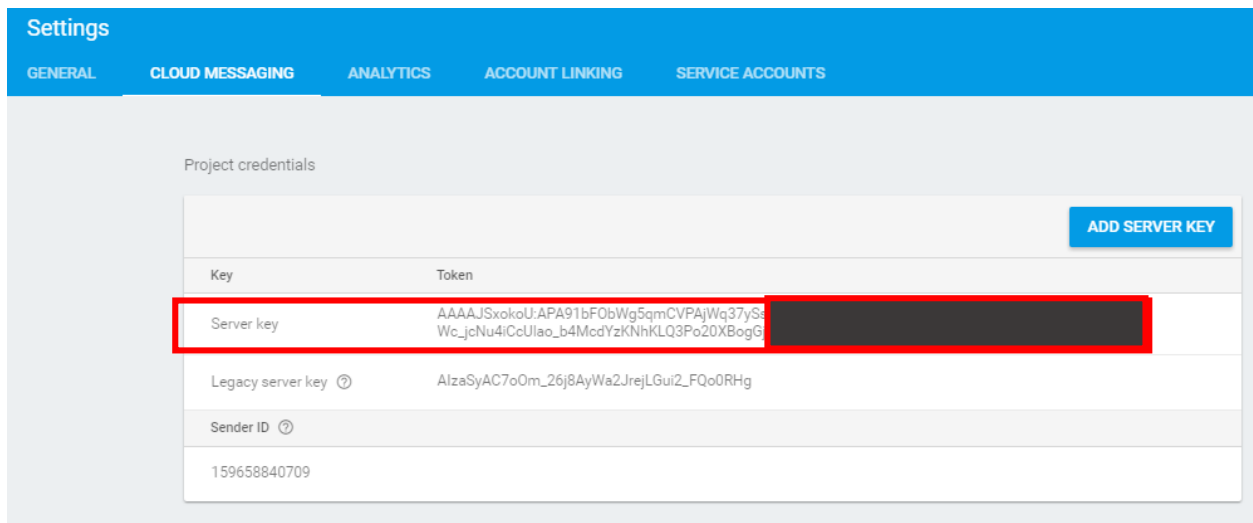
Kemudian pilih project yang telah didaftarkan :



Setelah masuk ke halaman berikutnya kemudian klik tombol gerigi, pilih yang “Project Setting”.



Kemudian pada halaman tersebut akan tampil seperti di bawah ini dan pilihlah tab “CLOUD MESSAGING”.



Salin Server key tersebut pada file “config.php” di atas.

Setelah itu buatlah file “Firebase.php” yang isinya sebagai berikut :

```
<?php
class Firebase {
    // sending push message to single user by firebase reg id
    public function send($to, $message) {
        $fields = array(
            'to' => $to,
            'data' => $message,
        );
        return $this->sendPushNotification($fields);
    }
    // Sending message to a topic by topic name
    public function sendToTopic($to, $message) {
        $fields = array(
            'to' => '/topics/' . $to,
            'data' => $message,
        );
        return $this->sendPushNotification($fields);
    }
    // sending push message to multiple users by firebase registration ids
    public function sendMultiple($registration_ids, $message) {
        $fields = array(
            'to' => $registration_ids,
            'data' => $message,
        );
        return $this->sendPushNotification($fields);
    }
    // function makes curl request to firebase servers
```

```

private function sendPushNotification($fields) {
    require_once __DIR__ . '/config.php';
    // Set POST variables
    $url = 'https://fcm.googleapis.com/fcm/send';
    $headers = array(
        'Authorization: key=' . FIREBASE_API_KEY,
        'Content-Type: application/json'
    );
    // Open connection
    $ch = curl_init();
    // Set the url, number of POST vars, POST data
    curl_setopt($ch, CURLOPT_URL, $url);
    curl_setopt($ch, CURLOPT_POST, true);
    curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    // Disabling SSL Certificate support temporarily
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
    curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($fields));
    // Execute post
    $result = curl_exec($ch);
    if ($result === FALSE) {
        die('Curl failed: ' . curl_error($ch));
    }
    // Close connection
    curl_close($ch);
    return $result;
}
}
?>

```

Setelah file “Firebase.php” dibuat, kemudian buatlah file baru bernama “Push.php” yang isinya sebagai berikut :

```

<?php
class Push {
    // push message title
    private $title;
    private $message;
    private $image;
    // push message payload
    private $data;
    // flag indicating whether to show the push
    // notification or not
    // this flag will be useful when perform some opertation
    // in background when push is received
    private $is_background;
    function __construct() {
    }
    public function setTitle($title) {

```

```

        $this->title = $title;
    }
    public function setMessage($message) {
        $this->message = $message;
    }
    public function setImage($imageUrl) {
        $this->image = $imageUrl;
    }
    public function setIsBackground($is_background) {
        $this->is_background = $is_background;
    }
    public function getPush() {
        $res = array();
        $res['data']['title'] = $this->title;
        $res['data']['is_background'] = $this->is_background;
        $res['data']['message'] = $this->message;
        $res['data']['image'] = $this->image;
        $res['data']['timestamp'] = date('Y-m-d G:i:s');
        return $res;
    }
}

```

Setelah itu anda perlu modifikasi file controller `controllerInformasi.php`. Ikuti syntax di bawah ini untuk method `setNotification` :

```

function setNotification($judul, $isi, $push_type) {
    $firebase = new Firebase();
    $push = new Push();
    $push->setTitle($judul);
    $push->setMessage($isi);
    $push->setImage('');
    $push->setIsBackground(FALSE);
    $json = '';
    $response = '';
    if ($push_type == 'topic') {
        $json = $push->getPush();
        $response = $firebase->sendToTopic('global', $json);
    } else if ($push_type == 'individual') {
        $json = $push->getPush();
        $pdo = database::connect();
        $sql = "select id, fcmid from fcmregid";
        foreach ($pdo->query($sql) as $row) {
            $regId = $row['fcmid'];
            $response = $firebase->send($regId, $json);
        }
        database::disconnect();
    }
}

```

Setelah itu tambahkan syntax

```
include 'firebase.php';  
include 'push.php';
```

Yang letaknya ada di bawah include “database.php”. Nanti terlihat seperti di bawah ini :

```
include 'modelInformasi.php';  
include 'database.php';  
include 'firebase.php';  
include 'push.php';
```

Setelah itu Anda modifikasi file “insert.php” dan tambahkan syntax di bawah ini :

```
$controllerInformasi->setNotification($judul, $isi, "individual");
```

Tuliskan syntax di atas di bawahnya syntax :

```
$controllerInformasi->setInformasi($judul, $isi, $jampost);
```

Nanti hasilnya seperti di bawah ini :

```
$controllerInformasi->setInformasi($judul, $isi, $jampost);  
$controllerInformasi->setNotification($judul, $isi, "individual");
```

Setelah itu mari kita COBA!

SELAMAT BAGI YANG BERHASIL 😊

Credits

Thanks to :

Ravi Tamada (Android Hive) for your tutorial.

Laboratorium Komputer Institut Bisnis dan Informatika STIKOM Surabaya.