

## INSERT TO MySQL SERVER

**Modul ini merupakan kelanjutan dari Modul-modul sebelumnya.**

Studi kasus yang diambil pada modul ini adalah melakukan insert berupa text dan Foto yang diambil dari Galery atau diambil dari Camera. Metode pengiriman data ke server menggunakan “voley” dan Foto yang dikirim akan diubah kedalam bentuk string terlebih dahulu menggunakan base64 encoding sehingga dapat dikirimkan sebagai text.

Tambahkan dependencies pada gradle:

```
dependencies {  
    ...  
  
    implementation 'com.android.volley:volley:1.1.1'  
    implementation 'com.star_zero:sse:1.2.0'  
    implementation 'com.karumi:dexter:5.0.0'  
}
```

Tambahkan Permission pada Manifest

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

Buat Activity baru untuk registrasi user, tambahkan button diatas Recyclerview pada layout activity\_main.xml.

```
<?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:layout_width="match_parent"  
    android:layout_height="match_parent"  
    tools:context=".MainActivity">  
  
    <LinearLayout  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:orientation="vertical">  
        <Button  
            android:id="@+id/btnreguser"  
            android:layout_width="wrap_content"  
            android:layout_height="wrap_content"  
            android:text="Add New User"  
            tools:ignore="InvalidId" />  
  
        <android.support.v7.widget.RecyclerView  
            android:id="@+id/recyclerView"  
            android:layout_width="match_parent"  
            android:layout_height="match_parent"  
            android:padding="4dp"  
            android:scrollbars="vertical"  
            android:background="@android:color/darker_gray"/>  
  
    </LinearLayout>
```

```
</RelativeLayout>
```

Beri onclickListener pada Button tersebut yang mengarah ke activity registrasi user di MainActivity.java

```
...
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    mList = new ArrayList<>();

    Button regBtn = (Button) findViewById(R.id.btnreguser);
    regBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent intent = new Intent(v.getContext(), RegUser.class);
            v.getContext().startActivity(intent);
        }
    });

    mRecyclerView = findViewById(R.id.recyclerView);
    mRecyclerView.setHasFixedSize(true);
    mLayoutManager = new LinearLayoutManager(this);
    mAdapter = new ExampleAdapter(mList);

    mRecyclerView.setLayoutManager(mLayoutManager);
    mRecyclerView.setAdapter(mAdapter);
    sse();
}
....
```

Buat layout baru dan beri nama (sebagai contoh saja) registeruser.xml

```
<?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="match_parent">

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

        <ImageView
            android:id="@+id/imageuser"
            android:layout_width="129dp"
            android:layout_height="109dp"
            android:layout_gravity="center"
            android:src="@drawable/ic_local_see_black_24dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Username" />

        <EditText
```

```

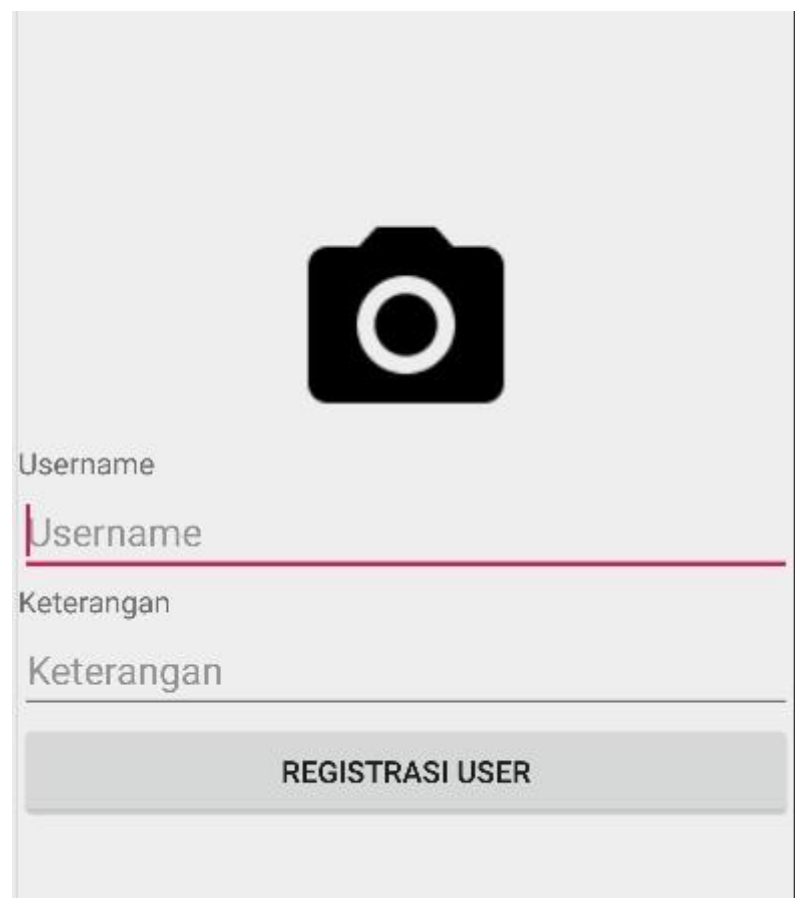
        android:id="@+id/editTextUsername"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Username" />
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Keterangan" />

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

    <Button
        android:id="@+id/btnreguser"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Registrasi User"/>
</LinearLayout>
</RelativeLayout>

```

Anda boleh mengganti gambar default pada <Imageview\> id/imageuser dengan gambar yang lain.



Username

Username

Keterangan

Keterangan

REGISTRASI USER

Buat Class baru dalam file RegUser.java

```
public class RegUser extends AppCompatActivity {

    private ImageView imageView;
    private EditText username, keterangan;
    private static final String IMAGE_DIRECTORY = "/user";
    private int GALLERY = 1, CAMERA = 2;

    private ProgressDialog progressDialog;
    private String API = "http://192.168.0.101/android/androidAPI.php";

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

        imageView = (ImageView) findViewById(R.id.imageuser);
        username = (EditText) findViewById(R.id.editTextUsername);
        keterangan = (EditText) findViewById(R.id.editTextKeterangan);

        imageView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                pictureDialog();
            }
        });

        Button regUser = (Button) findViewById(R.id.btnreguser);
        regUser.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                registerUser();
            }
        });
    }

    private void pictureDialog()
    {
        AlertDialog.Builder pictureDialog = new AlertDialog.Builder(this);
        pictureDialog.setTitle("Select Action");
        String[] dialogItem = {
            "Select Photo from Gallery",
            "Capture Photo from Camera",
            "Cancel"
        };

        pictureDialog.setItems(dialogItem,
            new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which) {
                    switch (which)
                    {
                        case 0:
                            fromGallery();
                            break;
                        case 1:
                            fromCamera();
                            break;
                        default:
                            dialog.dismiss();
                    }
                }
            }
        );
        pictureDialog.show();
    }

    private void fromGallery()
    {
        Intent galleryIntent = new Intent(Intent.ACTION_PICK,
            MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
        startActivityForResult(galleryIntent, GALLERY);
    }
}
```

```

private void fromCamera()
{
    Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    startActivityForResult(intent,CAMERA);
}

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data)
{
    super.onActivityResult(requestCode, resultCode, data);
    if(resultCode == this.RESULT_CANCELED)
        return;

    if(requestCode==GALLERY)
    {
        if(data != null)
        {
            Uri contentURI = data.getData();
            Bitmap bitmap = null;
            try {
                bitmap = MediaStore.Images.Media.getBitmap(this.getContentResolver(),
                    contentURI);
                imageView.setImageBitmap(bitmap);
            } catch (IOException e) {
                e.printStackTrace();
                Toast.makeText(getApplicationContext(), "Failed",
                    Toast.LENGTH_LONG).show();
            }
        }
    }else if(requestCode == CAMERA)
    {
        Bitmap bitmap = (Bitmap) data.getExtras().get("data");
        imageView.setImageBitmap(bitmap);
    }
}

private void registerUser()
{
    Bitmap bitmap = ((BitmapDrawable) imageView.getDrawable()).getBitmap();
    ByteArrayOutputStream bytes = new ByteArrayOutputStream();
    //convert bitmap menjadi format data JPEG dengan kualitas 90% dan simpan hasil
    //didalam bytes
    bitmap.compress(Bitmap.CompressFormat.JPEG, 90, bytes);
    byte[] picdat = bytes.toByteArray();
    final String mypic = Base64.encodeToString(picdat,0);

    final String uname = username.getText().toString();
    final String ket = keterangan.getText().toString();

    StringRequest stringRequest = new StringRequest(Request.Method.POST,
        API,
        new Response.Listener<String>() {
            @Override
            public void onResponse(String response) {

                try {
                    Log.d("ResponseWEB: ", response);
                    JSONObject jsonObject = new JSONObject(response);
                    Toast.makeText(getApplicationContext(),
                        jsonObject.getString("message"),Toast.LENGTH_LONG).show();

                    Intent intent = new Intent(getApplicationContext(),
                        MainActivity.class);
                    getApplicationContext().startActivity(intent);

                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        },

```

```

        new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                //progressDialog.hide();
                Toast.makeText(getApplicationContext(),
                    error.getMessage(), Toast.LENGTH_LONG).show();
            }
        }

        @Override
        protected Map<String, String> getParams() throws AuthFailureError {
            Map<String, String> params = new HashMap<>();
            params.put("username", uname);
            params.put("keterangan", ket);
            params.put("image", mypic);
            return params;
        }
    };

    RequestQueue requestQueue = Volley.newRequestQueue(this);
    requestQueue.add(stringRequest);
}

private void requestMultiplePermission(){
    Dexter.withActivity(this)
        .withPermissions(
            Manifest.permission.CAMERA,
            Manifest.permission.WRITE_EXTERNAL_STORAGE,
            Manifest.permission.READ_EXTERNAL_STORAGE)
        .withListener(new MultiplePermissionsListener() {
            @Override
            public void onPermissionsChecked(MultiplePermissionsReport report) {
                if(report.areAllPermissionsGranted()){
                    Toast.makeText(getApplicationContext(), "Semua OK",
                        Toast.LENGTH_LONG).show();
                }
                if(report.isAnyPermissionPermanentlyDenied()){
                }
            }

            @Override
            public void onPermissionRationaleShouldBeShown(
                List<PermissionRequest> permissions, PermissionToken token) {
                token.continuePermissionRequest();
            }
        })
        .withErrorListener(new PermissionRequestErrorListener() {
            @Override
            public void onError(DexterError error) {
                Toast.makeText(getApplicationContext(), "ERROR BRO!!",
                    Toast.LENGTH_LONG).show();
            }
        })
        .onSameThread()
        .check();
}
}

```

Tambahkan Activity baru ini pada manifest

```

<activity android:name=".RegUser"
    tools:ignore="DuplicateActivity"></activity>

```

#### Penjelasan:

Terdapat 2 ClickListener pada layout diatas; pada Gambar (ImageView) dan Pada Button. OnclickListener pada Gambar digunakan untuk mengambil gambar dengan 2 pilihan, yaitu dari Gallery atau Camera.

OnClickListener pada Gambar akan memanggil method `pictureDialog()` dimana method akan mentrigger sebuah diaolog box yang memberi pilihan antara ambil gambar dari Gallery (`fromGallery()`) atau dari Camera (`fromCamera()`). Masing-masing yang akan terpilih akan memanggil Intent yang akan di jalankan untuk mendapatkan hasil (`startActivityForResult()`).

Karena `startActivityForResult()` dijalankan, maka method `onActivityResult()` harus di overwrite. Syntax kedua method tersebut:

```
startActivityForResult(Intent intent, int requestCode)
```

```
onActivityResult(int requestCode, int resultCode, Intent data)
```

Karena `startActivityForResult()` bisa dipanggil untuk mengambil gambar dari Gallery atau Camera, maka didalam Method `onActivityResult()` pun disiapkan dua kondisi berdasarkan `requestCode` yang diberikan. Pada studi kasus ini digunakan sebuah variabel `GALLERY` (integer 1) dan `CAMERA` (integer 2). Code selanjutnya yang ada pada masing-masing pilihan adalah sebuah template cara ambil gambar dari Gallery atau Camera dan meletakkannya pada imageView yang disediakan.

Untuk argumen `resultCode` dihalaman web official android pun hanya dibahas terkait `RESULT_CANCELED`, yaitu jika user membatalkan untuk mengambil gambar.

Hal yang paling penting dijelaskan disini adalah code dalam method `registerUser()`. Gambar yang diambil dari Gallery ataupun Camera disimpan dalam bentuk Bitmap. Untuk dapat menyimpan gambar tersebut dalam database, kita harus merubah formatnya terlebih dahulu ke dalam format yang dikenali, pada studi kasus ini adalah JPEG. Langkah-langkahnya seperti yang terlihat pada code diatas dan variabel yang menyimpan image dalam bentuk JPEG dengan format yang dikenali oleh PC adalah `bytes[] picdat`. Selanjutnya data pada `picdat` harus diubah menjadi bentuk String agar dapat dikirim sebagai text menggunakan method HTTP POST. Base64 encoding digunakan untuk tujuan tersebut. Code selanjutnya adalah bentuk template dari Voley (AJAX jika di JS / JQUERY) dengan mengirimkan 3 data; username, keterangan dan image yang sudah diubah menjadi string.

Method `requestMultiplePermission()` tidak perlu dijelaskan panjang, code tersebut template untuk memastikan bahwa permission untuk mengakses telah diberikan. Jika belum, maka akan muncul permintaan untuk memberi akses.

## WEB SERVER

androidAPI.php

```
<?php
$response=array();

if($_SERVER['REQUEST_METHOD']=='POST'){

    if(isset($_POST['username']) &&
        isset($_POST['keterangan']) &&
        isset($_POST['image']))
    {
        //---- Put Image to Directory
        $img = base64_decode($_POST['image']);
        header('Content-Type: bitmap; charset=utf-8');
```

```

        $imgname = time().' .jpg';
        $file=fopen('image/' . $imgname, 'wb');
        fwrite($file,$img);
        fclose($file);
        //-----

        require_once "dboperations.php";
        $dbop = new dboperations();
        if($dbop->createUser($_POST['username'],$_POST['keterangan'],$imgname))
        {
            $response['error'] = false;
            $response['message']="Operation Succeeded";
        }
        else
        {
            $response['error'] = true;
            $response['message']="Failed Creating User";
        }
    }
    else
    {
        $response['error'] = true;
        $response['message']="Missing Values";
    }
}
else
{
    $response['error'] = true;
    $response['message']="Invalid Request";
}

echo json_encode($response);

```

?>

dboperations.php

```

<?php
class dboperations{
    private $con;

    function __construct(){
        require_once "db.php";
        $db = new DBconnect();
        $this->con = $db->connect();
    }
}

```



```

    }

    function createUser($username, $keterangan, $image)
    {
        $sql = "insert into usergambar
            values('','$username','$keterangan','$image')";

        return mysqli_query($this->con, $sql);
    }

    function getAllUsengambar()
    {
        $sql="select * from usergambar";
        $res=mysqli_query($this->con, $sql);
        $rows=array();
        $i=0;
        while($row = mysqli_fetch_assoc($res))
        {
            $rows[$i]=$row;
            $i++;
        }

        return $rows;
    }
}

?>

```

db.php

```

<?php
class DBconnect{
    private $con;
    function __construct()
    {
    }

    function connect()
    {
        $this->con = new mysqli("localhost","root","","android");

        if(mysqli_connect_errno())
        {
            echo "connection failed".mysqli_connect_err();
        }

        return $this->con;
    }
}

```

```
}  
?>
```