

LAMPIRAN

Source Code Class Login

```
package com.tellindo.ppb;

/**
 * Created by aryatama on 10/01/18.
 */

import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.net.ConnectivityManager;
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 com.tellindo.ppb.app.AppControl;

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

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

public class Login extends AppCompatActivity {
    ProgressDialog pDialog;
    Button btn_login;
    EditText txt_kode, txt_password;
    //Intent intent;

    int success;
    ConnectivityManager conMgr;

    private String url = Server.URL + "login.php";
```

```

private static final String TAG = Login.class.getSimpleName();

private static final String TAG_SUCCESS = "success";
private static final String TAG_MESSAGE = "message";

public final static String TAG_USERNAME = "username";
public final static String TAG_KODE = "kd";

String tag_json_obj = "json_obj_req";

SharedPreferences sharedPreferences;
Boolean session = false;
String username, kd;
public static final String my_shared_preferences = "my_shared_preferences";
public static final String session_status = "session_status";

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

    conMgr = (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);
    {
        if (conMgr.getActiveNetworkInfo() != null
            && conMgr.getActiveNetworkInfo().isAvailable()
            && conMgr.getActiveNetworkInfo().isConnected()) {
        } else {
            Toast.makeText(getApplicationContext(), "No Internet Connection",
                Toast.LENGTH_LONG).show();
        }
    }

    btn_login = findViewById(R.id.btn_login);
    txt_kode = findViewById(R.id.txt_kode);
    txt_password = findViewById(R.id.txt_password);

    // Cek session login jika TRUE maka langsung buka MainActivity
    sharedPreferences = getSharedPreferences(my_shared_preferences,
Context.MODE_PRIVATE);
    session = sharedPreferences.getBoolean(session_status, false);
    kd = sharedPreferences.getString(TAG_KODE, null);
    username = sharedPreferences.getString(TAG_USERNAME, null);

    if (session) {
        Intent intent = new Intent(Login.this, Menu.class);

```

```

        intent.putExtra(TAG_KODE, kd);
        intent.putExtra(TAG_USERNAME, username);
        finish();
        startActivity(intent);
    }

    btn_login.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {
            // TODO Auto-generated method stub
            String kode = txt_kode.getText().toString();
            String pass = txt_password.getText().toString();

            // Enkripsi
            MCrypt mcrypt = new MCrypt();
            String password = "";
            try {
                password = MCrypt.bytesToHex(mcrypt.encrypt(pass));
            } catch (Exception e) {
                e.printStackTrace();
            }

            // mengecek kolom yang kosong
            if (kode.trim().length() > 0 && password.trim().length() > 0) {
                if (conMgr.getActiveNetworkInfo() != null
                    && conMgr.getActiveNetworkInfo().isAvailable()
                    && conMgr.getActiveNetworkInfo().isConnected()) {
                    checkLogin(kode, password);
                } else {
                    Toast.makeText(getApplicationContext(), "No Internet
Connection", Toast.LENGTH_LONG).show();
                }
            } else {
                // Prompt user to enter credentials
                Toast.makeText(getApplicationContext(), "Kolom tidak boleh
kosong", Toast.LENGTH_LONG).show();
            }
        }
    });

    private void checkLogin(final String kode, final String password) {
        pDialog = new ProgressDialog(this);
    }

```

```

pDialog.setCancelable(false);
pDialog.setMessage("Logging in ...");
showDialog();

StringRequest strReq = new StringRequest(Request.Method.POST, url, new
Response.Listener<String>() {

    @Override
    public void onResponse(String response) {
        Log.e(TAG, "Login Response: " + response.toString());
        hideDialog();

        try {
            JSONObject jsonObj = new JSONObject(response);
            success = jsonObj.getInt(TAG_SUCCESS);

            // Check for error node in json

            if (success == 1) {
                String username = jsonObj.getString(TAG_USERNAME);
                String kd = jsonObj.getString(TAG_KODE);

                Log.e("Successfully Login!", jsonObj.toString());

                Toast.makeText(getApplicationContext(),
jObj.getString(TAG_MESSAGE), Toast.LENGTH_LONG).show();

                // menyimpan login ke session
                SharedPreferences.Editor editor = sharedPreferences.edit();
                editor.putBoolean(session_status, true);
                editor.putString(TAG_KODE, kd);
                editor.putString(TAG_USERNAME, username);
                editor.commit();

                // Memanggil main activity
                Intent intent = new Intent(Login.this, Menu.class);
                intent.putExtra(TAG_KODE, kd);
                intent.putExtra(TAG_USERNAME, username);
                finish();
                startActivity(intent);
            } else {
                Toast.makeText(getApplicationContext(),
jObj.getString(TAG_MESSAGE),
Toast.LENGTH_LONG).show();
            }
        }
    }
}

```

```

        } catch (JSONException e) {
            // JSON error
            e.printStackTrace();
        }

    }

}, new Response.ErrorListener() {

    @Override
    public void onErrorResponse(VolleyError error) {
        Log.e(TAG, "Login Error: " + error.getMessage());
        Toast.makeText(getApplicationContext(),
            error.getMessage(), Toast.LENGTH_LONG).show();

        hideDialog();

    }
}) {

    @Override
    protected Map<String, String> getParams() {
        // Posting parameters to login url
        Map<String, String> params = new HashMap<String, String>();
        params.put("kode", kode);
        params.put("password", password);

        return params;
    }

};

// Adding request to request queue
AppControl.getInstance().addToRequestQueue(strReq, tag_json_obj);
}

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

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

```

Source Code Class Menu

```
package com.tellindo.ppb;

import android.app.ProgressDialog;
import android.net.ConnectivityManager;
import android.util.Log;
import android.view.View;
import android.content.Intent;
import android.content.Context;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.ListView;
import android.widget.TextView;
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 com.tellindo.ppb.app.AppControl;

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

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

/**
 * Created by aryatama on 26/12/17.
 */

public class Menu extends AppCompatActivity {

    ProgressDialog pDialog;
    Button btn_cpass, btn_logout, btn_history;
    TextView txt_id, txt_user;
    String username, kode;
    SharedPreferences sharedPreferences;
    ListView listview;
```

```

CustomListOrder adapter;
List<Order> pesanList = new ArrayList<>();

ConnectivityManager conMgr;

String tag_json_obj = "json_obj_req";

private String url = Server.URL + "list.php";

private static final String TAG = Menu.class.getSimpleName();

public static final String TAG_KODE = "kd";
public static final String TAG_USERNAME = "username";
public static final String TAG_IDPESANAN = "id_pesanan";

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

    btn_cpass = findViewById(R.id.btn_cpass);
    btn_logout = findViewById(R.id.btn_logout);
    btn_history = findViewById(R.id.btn_history);
    txt_id = findViewById(R.id.txt_id);
    txt_user = findViewById(R.id.txt_user);
    listview = findViewById(R.id.list_pesanan);
    adapter = new CustomListOrder(this, pesanList);
    listview.setAdapter(adapter);

    sharedPreferences = getSharedPreferences(Login.my_shared_preferences,
Context.MODE_PRIVATE);

    kode = getIntent().getStringExtra(TAG_KODE);
    username = getIntent().getStringExtra(TAG_USERNAME);

    txt_id.setText("ID : " + kode);
    txt_user.setText("USER : " + username);

    /*
                                     conMgr = (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);
    {
        if (conMgr.getActiveNetworkInfo() != null
            && conMgr.getActiveNetworkInfo().isAvailable()
            && conMgr.getActiveNetworkInfo().isConnected()) {
            daftarPesanan(kode);

```

```

    } else {
        Toast.makeText(getApplicationContext(), "No Internet Connection",
            Toast.LENGTH_LONG).show();
    }
}
*/

//Ganti Password
btn_cpass.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        Intent intent = new Intent(Menu.this, UPass.class);
        intent.putExtra(TAG_KODE, kode);

        startActivity(intent);
    }
});

btn_history.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        Intent intent = new Intent(Menu.this, History.class);
        intent.putExtra(TAG_KODE, kode);

        startActivity(intent);
    }
});

//----- klik daftar pesanan -----
listview.setOnItemClickListener(new AdapterView.OnItemClickListener()
{
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int
position, long i) {
        Intent intent = new Intent(Menu.this, Input.class);
        intent.putExtra(TAG_USERNAME, username);
        intent.putExtra(TAG_KODE, kode);

        String id_pesanan =
((TextView)view.findViewById(R.id.tvSpl)).getText().toString();
        intent.putExtra(TAG_IDPESANAN, id_pesanan);
        startActivity(intent);
    }
});

```



```

    }
});

//Logout
btn_logout.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        // update login session ke FALSE dan mengosongkan nilai id dan
username
        SharedPreferences.Editor editor = sharedPreferences.edit();
        editor.putBoolean(Login.session_status, false);
        editor.putString(TAG_KODE, null);
        editor.putString(TAG_USERNAME, null);
        editor.commit();

        Intent intent = new Intent(Menu.this, Login.class);
        finish();
        startActivity(intent);
    }
});
}

//ListView Pesanan
private void daftarPesanan(final String kode){
    pDialog = new ProgressDialog(this);
    pDialog.setCancelable(false);
    pDialog.setMessage("Loading ...");
    showDialog();

    StringRequest arrReq = new StringRequest(Request.Method.POST,url, new
Response.Listener<String>() {

        @Override
        public void onResponse(String response) {
            Log.e(TAG, response.toString());
            hideDialog();

            try {
                JSONArray j = new JSONArray(response);
                for (int i = 0; i < j.length(); i++) {
                    try {
                        JSONObject jsonObj = j.getJSONObject(i);
                        Order o = new Order();
                        o.setNospl(jsonObj.getString("no_spl"));

```

```

        o.setNamalang(jObj.getString("namalang"));
        o.setAlamat(jObj.getString("alamat"));
        o.setTel(jObj.getString("tel"));
        o.setHp(jObj.getString("hp"));
        o.setTglo(jObj.getString("tglo"));
        o.setTgloa(jObj.getString("tgloa"));

        pesanList.add(o);
    } catch (JSONException e) {
        // JSON error
        e.printStackTrace();
    }
}
adapter.notifyDataSetChanged();
} catch (JSONException e) {
    e.printStackTrace();
}
}
}, new Response.ErrorListener() {

    @Override
    public void onErrorResponse(VolleyError error) {
        Log.e(TAG, "Error: " + error.getMessage());
        Toast.makeText(getApplicationContext(),
            error.getMessage(), Toast.LENGTH_LONG).show();

        hideDialog();

    }
}) {

    @Override
    protected Map<String, String> getParams() {
        // Posting parameters to login url
        Map<String, String> params = new HashMap<String, String>();
        params.put("kode", kode);

        return params;
    }
};

// Adding request to request queue
AppControl.getInstance().addToRequestQueue(arrReq, tag_json_obj);
}

```

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

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

protected void onResume() {
    super.onResume();
    pesanList.clear();

    conMgr = (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);
    {
        if (conMgr.getActiveNetworkInfo() != null
            && conMgr.getActiveNetworkInfo().isAvailable()
            && conMgr.getActiveNetworkInfo().isConnected()) {
            daftarPesanan(kode);
        } else {
            Toast.makeText(getApplicationContext(), "No Internet Connection",
                Toast.LENGTH_LONG).show();
        }
    }
}
}
```

Source Code Class Input

```
package com.tellindo.ppb;

import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.net.ConnectivityManager;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.Editable;
import android.text.TextWatcher;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextClock;
import android.widget.TextView;
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 com.tellindo.ppb.app.AppControl;

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

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

public class Input extends AppCompatActivity {

    TextView txt_user,no_spl,no_sp,nosbg,tglrsg,cicil,gol,lak;
    TextView txt1,txt2;
    EditText nometer,kdmerk,spk1,spk2,ket;
    String username,nospl,kode;
    TextClock tglrsg;
    RadioGroup tipe;
    Button input;
    ProgressDialog pDialog;
```

```

ConnectivityManager conMgr;
String tag_json_obj = "json_obj_req";
private String urls = Server.URL + "select.php";
private String urli = Server.URL + "indate.php";
private static final String TAG = Menu.class.getSimpleName();

public static final String TAG_KODE = "kd";
public static final String TAG_USERNAME = "username";
public static final String TAG_IDPESANAN = "id_pesanan";

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

    /* test */
    txt1 = findViewById(R.id.textView);
    txt2 = findViewById(R.id.textView2);
    /* test */

    tglrsbg = findViewById(R.id.tglrsbg);
    tipe = findViewById(R.id.tipe);
    txt_user = findViewById(R.id.txt_user);
    no_spl = findViewById(R.id.no_spl);
    no_sp = findViewById(R.id.no_sp);
    nosbg = findViewById(R.id.nosbg);
    tglsgb = findViewById(R.id.tglsgb);
    cicil = findViewById(R.id.cicil);
    gol = findViewById(R.id.gol);
    lak = findViewById(R.id.lak);
    input = findViewById(R.id.btn_input);

    nometer = findViewById(R.id.nometer);
    kdmerk = findViewById(R.id.kdmerk);
    spk1 = findViewById(R.id.spk1);
    spk2 = findViewById(R.id.spk2);
    ket = findViewById(R.id.ket);

//----- STRING
-----//
    kode = getIntent().getStringExtra(TAG_KODE);
    username = getIntent().getStringExtra(TAG_USERNAME);
    nospl = getIntent().getStringExtra(TAG_IDPESANAN);

//----- kolom nomor spk
-----//

```

```

    spk1.addTextChangedListener(new TextWatcher() {
        public void onTextChanged(CharSequence s, int start,int before, int
count)
        {
            // TODO Auto-generated method stub
            if(spk1.getText().toString().length()==5)    //size as per your
requirement
            {
                spk2.requestFocus();
            }
        }
        public void beforeTextChanged(CharSequence s, int start,
            int count, int after) {
            // TODO Auto-generated method stub

        }

        public void afterTextChanged(Editable s) {
            // TODO Auto-generated method stub
        }

    });

//-----          select          sambungan
-----//
                                conMgr    =    (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);
    {
        if (conMgr.getActiveNetworkInfo() != null
            && conMgr.getActiveNetworkInfo().isAvailable()
            && conMgr.getActiveNetworkInfo().isConnected()) {
            select(kode,nospl);
        } else {
            Toast.makeText(getApplicationContext(), "No Internet Connection",
                Toast.LENGTH_LONG).show();
        }
    }

//-----          TOMBOL          INPUT
-----//
    input.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {
            // TODO Auto-generated method stub

```

```

        if (tipe.getCheckedRadioButtonId() != -1) {
            String tgl = tglrsbg.getText().toString();
            String nmtr = nometer.getText().toString();
            String kdmrk = kdmerk.getText().toString();
            String s1 = spk1.getText().toString();
            String s2 = spk2.getText().toString();
            String kt = ket.getText().toString();

            String tp = ((RadioButton)
findViewById(tipe.getCheckedRadioButtonId())).getText().toString();

            if (nmtr.trim().length() > 0 && kdmrk.trim().length() > 0 &&
s1.trim().length() > 0 && s2.trim().length() > 0 && kt.trim().length() > 0) {
                if (s2.trim().length() > 3) {
                    if (conMgr.getActiveNetworkInfo() != null &&
conMgr.getActiveNetworkInfo().isAvailable() &&
conMgr.getActiveNetworkInfo().isConnected()) {
                        indate(tgl, nmtr, kdmrk, s1, s2, kt, tp, kode, nospl);
                    } else {
                        Toast.makeText(getApplicationContext(), "No Internet
Connection", Toast.LENGTH_LONG).show();
                    }
                } else {
                    Toast.makeText(getApplicationContext(), "lengkapi spk",
Toast.LENGTH_LONG).show();
                }
            } else {
                Toast.makeText(getApplicationContext(), "Kolom tidak boleh
kosong", Toast.LENGTH_LONG).show();
            }
        } else {
            Toast.makeText(getApplicationContext(), "Mohon tentukan TIPE",
Toast.LENGTH_LONG).show();
        }
    }
});
}

//----- SELECT SAMBUNGAN
-----//

private void select(final String kode, final String nospl) {
    pDialog = new ProgressDialog(this);
    pDialog.setCancelable(false);
    pDialog.setMessage("Loading ...");
    showDialog();
}

```

```

StringRequest arrReq = new StringRequest(Request.Method.POST, urls,
new Response.Listener<String>() {

    @Override
    public void onResponse(String response) {
        Log.e(TAG, response.toString());
        hideDialog();

        try {
            JSONObject jsonObj = new JSONObject(response);
            int success = jsonObj.getInt("success");

            // Check for error node in json

            if (success == 1) {

                Toast.makeText(getApplicationContext(),
                    jsonObj.getString("message"), Toast.LENGTH_LONG).show();

                no_spl.setText(nospl);
                no_sp.setText(jsonObj.getString("no_sp"));
                nosbg.setText(jsonObj.getString("nosbg"));
                tglsbg.setText(jsonObj.getString("tglsbg"));
                cicil.setText(jsonObj.getString("cicil"));
                gol.setText(jsonObj.getString("gol"));
                nometer.setText(jsonObj.getString("nometer"));
                kdmerk.setText(jsonObj.getString("kdmerk"));
                spk1.setText(jsonObj.getString("spk1"));
                spk2.setText(jsonObj.getString("spk2"));
                ket.setText(jsonObj.getString("ket"));
                lak.setText(kode);
            } else {
                //gagal kembali ke menu
                Toast.makeText(getApplicationContext(),
                    jsonObj.getString("message"), Toast.LENGTH_LONG).show();
                finish();
            }

        } catch (JSONException e) {
            e.printStackTrace();
        }
    }
}, new Response.ErrorListener() {

    @Override
    public void onErrorResponse(VolleyError error) {

```



```

        Log.e(TAG, "Error: " + error.getMessage());
        Toast.makeText(getApplicationContext(),
            error.getMessage(), Toast.LENGTH_LONG).show();

        hideDialog();

    }
}) {

    @Override
    protected Map<String, String> getParams() {
        // Posting parameters to login url
        Map<String, String> params = new HashMap<String, String>();
        params.put("kode", kode);
        params.put("nospl", nospl);

        return params;
    }

};

// Adding request to request queue
AppControl.getInstance().addToRequestQueue(arrReq, tag_json_obj);
}

//----- INPUT UPDATE SAMBUNGAN -----//
private void indate(final String tgl, final String nmtr, final String kdmrk, final
String s1, final String s2, final String kt, final String tp, final String kd, final
String nspl){
    pDialog = new ProgressDialog(this);
    pDialog.setCancelable(false);
    pDialog.setMessage("Loading ...");
    showDialog();

    StringRequest arrReq = new StringRequest(Request.Method.POST,urli,
new Response.Listener<String>() {

        @Override
        public void onResponse(String response) {
            Log.e(TAG, response.toString());
            hideDialog();

            try {
                JSONObject jsonObj = new JSONObject(response);
                int success = jsonObj.getInt("success");

```

```

        // Check for error node in json

        if (success == 1) {

            Toast.makeText(getApplicationContext(),
                jsonObj.getString("message"), Toast.LENGTH_LONG).show();

            // biar refresh activity menu ?

            finish();

        } else {
            //gagal
            Toast.makeText(getApplicationContext(),
                jsonObj.getString("message"), Toast.LENGTH_LONG).show();

        }
    } catch (JSONException e) {
        e.printStackTrace();
    }
}
}, new Response.ErrorListener() {

    @Override
    public void onErrorResponse(VolleyError error) {
        Log.e(TAG, "Error: " + error.getMessage());
        Toast.makeText(getApplicationContext(),
            error.getMessage(), Toast.LENGTH_LONG).show();

        hideDialog();

    }
}) {

    @Override
    protected Map<String, String> getParams() {
        // Posting parameters to login url
        Map<String, String> params = new HashMap<String, String>();
        params.put("tgl", tgl);
        params.put("nmtr", nmtr);
        params.put("kdmrk", kdmrk);
        params.put("s1", s1);
        params.put("s2", s2);
        params.put("kt", kt);
        params.put("tp", tp);
    }
}

```

```
        params.put("kd", kd);
        params.put("nspl", nspl);

        return params;
    }

};

// Adding request to request queue
AppControl.getInstance().addToRequestQueue(arrReq, tag_json_obj);
}

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

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