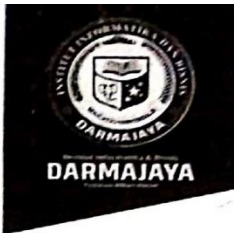


Lampiran



Bandar Lampung, 22 Januari 2021

Nomor : Penelitian.012/DMJ/DEKAN/BAAK/I-21
Lampiran : -
Perihal : Permohonan Izin Penelitian

Kepada Yth,
Kepala UMKM Saung Bambu Wisata Tani Edukasi
Di-
Desa Rejo Sari, Kec. Seputih Raman, Lampung Tengah

Dengan hormat,

Sehubungan dengan peraturan Akademik Institut Bisnis dan Informatika (IBI) bahwa mahasiswa/i Strata Satu (S1) yang akan menyelesaikan studinya diwajibkan untuk memiliki pengalaman kerja dengan melaksanakan Penelitian dan membuat laporan yang waktunya disesuaikan dengan kalender Institut Bisnis dan Informatika (IBI) Darmajaya.

Untuk itu kami mohon kerja sama Bapak/Ibu agar kiranya dapat menerima mahasiswa/i untuk melakukan Penelitian, yang pelaksanaannya dimulai dari tanggal 25 Januari 2021 s.d 28 Februari 2021 (selama satu bulan tiga hari)

Adapun mahasiswa/i tersebut adalah :

Nama : Rian Frinando
NPM : 1711010028
Jurusan : S1 Teknik Informatika
Jenjang : Strata Satu (S1)

Demikian permohonan ini dibuat, atas perhatian dan kerjasama yang baik kami ucapkan terimakasih.

Dean Fakultas Ilmu Komputer,

Zahid Jamal, S.T., M.Eng
NIDN 00590203

Tembusan:

1. Jurusan S1 Teknik Informatika
2. Arsip.



**PEMERINTAH KABUPATEN LAMPUNG TENGAH
KECAMATAN SEPUTIH RAMAN**

Jl. Raya Seputih Raman No. 132 Kode Pos 34155 Telp. (0725) 49100

SURAT IZIN PENELITIAN / SURVEY / PENGABDIAN / KKN / KKL

NOMOR : 070/43 /Kc.a.VIII.19/2021

- MEMBACA** : Surat Dekan Universitas Darmajaya Nomor : Penelitian.012/DMJ/DEKAN/BAAK/I-21 tanggal 22 Januari 2021 tentang Permohonan Izin Penelitian.
- MENINGAT** : 1. Keputusan Direktur Jenderal Sosial Politik Departemen Dalam Negeri No. 14 tahun 1981 tanggal 31 Juni 1981 tentang Surat Pemberitahuan Penelitian.
2. Peraturan Daerah Propinsi Lampung Nomor 12 tahun 2007 tentang Pembentukan Organisasi dan Tata Kerja Lembaga Tehnis Daerah Propinsi Lampung.
3. Surat Gubernur KDH Tingkat I Lampung Nomor OP.030/461/G. Sospol/1985 tanggal 5 Februari 1985, tentang Permohonan Izin Penelitian/Survey bagi Dinas/Instansi dan Mahasiswa.

DENGAN INI DIBERIKAN IZIN KEPADA :

- Nama / NPM** : **RIAN FRINANDO / 1711010028**
- Pekerjaan** : Mahasiswa Universitas Darmajaya
- Lokasi Penelitian** : Kampung Rejo Asri, Kecamatan Seputih Raman, Kabupaten Lampung Tengah
- T u j u a n** : Mengadakan Penelitian dalam Penyusunan Skripsi
- Judul penelitian** : **“PERANGKAT LUNAK MARKET PLACE UMKM BERBASIS ANDROID STUDY KASUS DI DESA REJO ASRI KECAMATAN SEPUTIH RAMAN”**
- CATATAN** : Setelah selesai melaksanakan kegiatan berdasarkan Surat Izin ini agar melaporkan hasilnya secara tertulis Kepada Bupati Lampung Tengah Cq. Kepala Kantor Kesatuan Bangsa dan Politik Kabupaten Lampung Tengah.

DIKELUARKAN DI : SEPUTIH RAMAN
PADA TANGGAL : 17 MARET 2021



EKO PRANARTO, S.IP

Camat Tk. I, IV/b

NIP. 19650414 198603 1 009

- Tembusan** : Disampaikan kepada Yth,
1. Bupati Lampung Tengah (sebagai laporan)
2. Peringgal



DARMAJAYA

Yayasan Alifan Husin

Jl. Zuhri Abdin Pagar Alam No. 93 Bandar Lampung 35142 Telp. 787214 Fax. 700261 <http://darmajaya.ac.id>

FORMULIR

BIRO ADMINISTRASI AKADEMIK KEMAHASISWAAN (BAAK)

FORM KONSULTASI/BIMBINGAN SKRIPSI/TUGAS AKHIR *)

NAMA : Rian Frinardo
 NPM : 1711010028
 PEMBIMBING I : ketut Artaya, s.kom., M.T.I.
 PEMBIMBING II :
 JUDUL LAPORAN : Perangkat Lunak Marketplace UMKM berbasis android...
 studi kasus desa Rejo #sr, kecamatan seputih raman, lamteng
 TANGGAL SK : s.d (6+2 bulan)

No	HARI/TANGGAL	HASIL KONSULTASI	PARAF
1	Senin, 28 Desember 2020	Perbaiki BAB 1, ruang lingkup ditambahkan batasan masalah, pempat penelitian, waktu penelitian	
2	Selasa 5 Januari 2021	perbaikan BAB 1, BAB 3.	
3	Kamis 21 Januari 2021	BAB 1 latar belakang di perbaiki penulisan-ca	
4	3 Januari 3 Februari 2021	BAB 2, metode penelitian menurut ahli, BAB 3 perbaiki subbab tidak boleh pakai titik	
5	16 Februari 2021	perbaikan BAB 3 analisis -rang berjalan	
6	1 Maret 2021	perbaikan BAB 3 penulisan kalimat	
7	5 Maret 2021	perbaikan BAB IV gambar diperkecil, dan BAB V kesimpulan ditambahkan poin lagi.	
8	7 Maret 2021	Perbaikan BAB 3 Desain database	
9	9 Maret 2021	Perbaiki BAB I, II, III, IV, V, perbaiki PPT.	
10			

*) Coret yang tidak perlu

Bandar Lampung,
Ketua Jurusan

(Dr. Chairani, S.kom., M.Eng....)
NIK. 00480802

Sourcode membuat URL

```
package com.rian.ecommerce_v1.server;

public class BaseURL {

    //public static String baseUrl = "http://192.168.18.7:5050/";
    //public static String baseUrl = "http://192.168.18.5:5051/";
    //public static String baseUrl = "http://192.168.43.81:5051/";
    public static String baseUrl = "http://studio81.co.id:5051/";

    public static String login = baseUrl + "users/signin";
    public static String register = baseUrl + "users/signup";
    public static String updateProfile = baseUrl + "users/updateprofile/";

    public static String checkToko = baseUrl + "toko/cektoko/";
    public static String daftarToko = baseUrl + "toko/daftar";
    public static String getDataToko = baseUrl + "toko/getall";

    public static String checkBarang = baseUrl + "barang/cekbarang/";
    public static String addBarang = baseUrl + "barang/addbarang";
    public static String getDataBarang = baseUrl + "barang/getbarangbyid/";
    public static String deleteBarang = baseUrl + "barang/deletebyid/";
    public static String updateBarang = baseUrl + "barang/update/";

    public static String addPesanan = baseUrl + "transaksi/addkeranjang";
    public static String getDataPesanan = baseUrl + "transaksi/cart/";
    public static String getDataPengiriman = baseUrl + "transaksi/toko/";
    public static String deleteDataPesanan = baseUrl + "transaksi/cart/";
    public static String checkoutDataPesanan = baseUrl + "transaksi/cart/";
    public static String pembayaranPesanan = baseUrl +
"transaksi/buktibayar";

    public static String getDataUser = baseUrl + "access/getdataUser";
    public static String completeUser = baseUrl + "access/completeData/";
    public static String updateUser = baseUrl + "access/updateUser/";

}
```

Membuat registrasi untuk penjual dan pembeli

```
package com.rian.ecommerce_v1.access;

import androidx.appcompat.app.AppCompatActivity;

import android.app.ProgressDialog;
import android.content.Intent;
import android.os.Bundle;
import android.text.InputType;
import android.view.View;
import android.view.WindowManager;
import android.widget.Button;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.VolleyLog;
import com.android.volley.toolbox.JsonObjectRequest;
import com.android.volley.toolbox.Volley;
import com.blogspot.atifsoftwares.animatoolib.Animatoo;
import com.rian.ecommerce_v1.R;
import com.rian.ecommerce_v1.server.BaseURL;
import com.rian.ecommerce_v1.utils.App;
import com.google.android.material.textfield.TextInputEditText;
import com.muddzdev.styleabletoastlibrary.StyleableToast;

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

import java.util.HashMap;

public class RegistSystem extends AppCompatActivity {

    private Button registPembeli, registPedagang;
    private TextInputEditText namaLengkap, username, password,
    nomorTelepon, emailAddress, alamatLengkap;
    private ProgressDialog progressDialog;

    private RequestQueue mRequestQueue;

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

        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
        WindowManager.LayoutParams.FLAG_FULLSCREEN);

        getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_INPUT_STATE_HI
        DDEN);

        mRequestQueue = Volley.newRequestQueue(this);
```

```

namaLengkap = (TextInputEditText) findViewById(R.id.fullName);
username = (TextInputEditText) findViewById(R.id.username);
password = (TextInputEditText) findViewById(R.id.password);
nomorTelepon = (TextInputEditText) findViewById(R.id.phone);
emailAddress = (TextInputEditText) findViewById(R.id.mail);
alamatLengkap = (TextInputEditText) findViewById(R.id.address);
registPembeli = (Button) findViewById(R.id.regist_pembeli);
registPedagang = (Button) findViewById(R.id.regist_pedagang);

namaLengkap.setInputType(InputType.TYPE_CLASS_TEXT |
InputType.TYPE_TEXT_FLAG_CAP_WORDS);
alamatLengkap.setInputType(InputType.TYPE_CLASS_TEXT |
InputType.TYPE_TEXT_FLAG_CAP_WORDS);

progressDialog = new ProgressDialog(this);
progressDialog.setCancelable(false);

registPembeli.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        App.getPref().clear();
        String dataNamaLengkap = namaLengkap.getText().toString();
        String dataUsername = username.getText().toString();
        String dataPassword = password.getText().toString();
        String dataNomorTelepon =
nomorTelepon.getText().toString();
        String dataEmailAddress =
emailAddress.getText().toString();
        String dataAlamatLengkap =
alamatLengkap.getText().toString();

        if (dataNamaLengkap.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data nama
lengkap tidak boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataUsername.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data
username tidak boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataPassword.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data
password tidak boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataNomorTelepon.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data nomor
telepon tidak boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataEmailAddress.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data alamat
email tidak boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataAlamatLengkap.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data alamat
lengkap tidak boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else {
            registPembeliFunction(dataNamaLengkap, dataUsername,
dataPassword, dataNomorTelepon, dataEmailAddress, dataAlamatLengkap);

```

```

    }
  }
});

registPedagang.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        App.getPref().clear();
        String dataNamaLengkap = namaLengkap.getText().toString();
        String dataUsername = username.getText().toString();
        String dataPassword = password.getText().toString();
        String dataNomorTelepon =
nomorTelepon.getText().toString();
        String dataEmailAddress =
emailAddress.getText().toString();
        String dataAlamatLengkap =
alamatLengkap.getText().toString();

        if (dataNamaLengkap.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data tidak
boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataUsername.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data tidak
boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataPassword.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data tidak
boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataNomorTelepon.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data tidak
boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataEmailAddress.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data tidak
boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else if (dataAlamatLengkap.isEmpty()){
            StyleableToast.makeText(RegistSystem.this, "Data tidak
boleh di kosongkan...", R.style.toastStyleWarning).show();
        }else {
            registPedagangFunction(dataNamaLengkap, dataUsername,
dataPassword, dataNomorTelepon, dataEmailAddress, dataAlamatLengkap);
        }
    }
});
}

private void registPembeliFunction(String dataNamaLengkap, String
dataUsername, String dataPassword, String dataNomorTelepon, String
dataEmailAddress, String dataAlamatLengkap) {
    HashMap<String, String> params = new HashMap<String, String>();

    params.put("nama", dataNamaLengkap);
    params.put("username", dataUsername);
    params.put("password", dataPassword);
}
}

```


Membuat login penjual dan pembeli

```
package com.rian.ecommerce_v1.access;

import androidx.appcompat.app.AppCompatActivity;

import android.app.ProgressDialog;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.LinearLayout;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.VolleyLog;
import com.android.volley.toolbox.JsonObjectRequest;
import com.android.volley.toolbox.Volley;
import com.blogspot.atifsoftwares.animatoolib.Animatoo;
import com.rian.ecommerce_v1.pedagang.MainPedagang;
import com.rian.ecommerce_v1.pembeli.MainPembeli;
import com.rian.ecommerce_v1.R;
import com.rian.ecommerce_v1.models.ModelUsers;
import com.rian.ecommerce_v1.server.BaseURL;
import com.rian.ecommerce_v1.utils.App;
import com.rian.ecommerce_v1.utils.GsonHelper;
import com.rian.ecommerce_v1.utils.Prefs;
import com.rian.ecommerce_v1.utils.Utils;
import com.google.android.material.textfield.TextInputEditText;
import com.muddzdev.styleabletoastlibrary.StyleableToast;

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

import java.util.HashMap;

public class LoginSystem extends AppCompatActivity {

    private Button loginToSystem;
    private LinearLayout registToSystem;
    private TextInputEditText username, password;
    private ProgressDialog progressDialog;
    private RequestQueue mRequestQueue;

    ModelUsers modelUsers;

    boolean BackPress = false;

    @Override
```

```

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

    getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
WindowManager.LayoutParams.FLAG_FULLSCREEN);

getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_INPUT_STATE
_HIDDEN);

    mRequestQueue = Volley.newRequestQueue(this);

    username = (TextInputEditText) findViewById(R.id.username);
    password = (TextInputEditText) findViewById(R.id.password);
    loginToSystem = (Button) findViewById(R.id.login_to_system);
    registToSystem = (LinearLayout) findViewById(R.id.regist);

    modelUsers = (ModelUsers)
GsonHelper.parseGson(App.getPref().getString(Prefs.PREF_STORE_PROFILE,
""), new ModelUsers());

    if(Utils.isLoggedIn()){
        int dataRole = Integer.parseInt(modelUsers.getRole());
        if (dataRole == 1){
            Intent i = new Intent(this , MainPembeli.class);
            startActivity(i);
            finish();
        }else if(dataRole == 2){
            Intent i = new Intent(this , MainPedagang.class);
            startActivity(i);
            finish();
        }
    }

    loginToSystem.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String dataUsername = username.getText().toString();
            String dataPassword = password.getText().toString();

            if (dataUsername.isEmpty()){
                StyleableToast.makeText(LoginSystem.this, "Data
username tidak boleh di kosongkan...",
R.style.toastStyleWarning).show();
            }else if (dataPassword.isEmpty()){
                StyleableToast.makeText(LoginSystem.this, "Data
password tidak boleh di kosongkan...",
R.style.toastStyleWarning).show();
            }else {
                loginFunction(dataUsername, dataPassword);
            }
        }
    });
}

```

```

        registToSystem.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(LoginSystem.this,
                RegistSystem.class));
                Animatoo.animateSlideDown(LoginSystem.this);
            }
        });

        progressDialog = new ProgressDialog(this);
        progressDialog.setCancelable(false);
    }

    private void loginFunction(String dataUsername, String dataPassword)
    {
        HashMap<String, String> params = new HashMap<String, String>();

        params.put("username", dataUsername);
        params.put("password", dataPassword);

        progressDialog.setTitle("Mohon tunggu sebentar...");
        showDialog();

        final JsonObjectRequest req = new
        JsonObjectRequest(BaseURL.Login, new JSONObject(params),
            new Response.Listener<JSONObject>() {
                @Override
                public void onResponse(JSONObject response) {
                    hideDialog();
                    try {
                        String strMsg =
                response.getString("message");
                        boolean statusMsg =
                response.getBoolean("status");

                            if (statusMsg == true) {

                StyleableToast.makeText(LoginSystem.this, strMsg,
                R.style.toastStyleSuccess).show();
            }
        }
    }
    }

```

Menu pedagang

```
package com.rian.ecommerce_v1.pedagang;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;

import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.widget.TextView;

import com.airbnb.lottie.LottieAnimationView;
import com.blogspot.atifsoftwares.animatoolib.Animatoo;
import com.rian.ecommerce_v1.R;
import com.rian.ecommerce_v1.access.LoginSystem;
import com.rian.ecommerce_v1.models.ModelUsers;
import com.rian.ecommerce_v1.server.BaseURL;
import com.rian.ecommerce_v1.utils.App;
import com.rian.ecommerce_v1.utils.GsonHelper;
import com.rian.ecommerce_v1.utils.Prefs;
import com.google.android.material.navigation.NavigationView;
import com.squareup.picasso.Picasso;

import de.hdodenhof.circleimageview.CircleImageView;

public class MainPedagang extends AppCompatActivity {

    DrawerLayout drawerLayout;
    ActionBarDrawerToggle toggle;
    Toolbar toolbar;
    NavigationView navigationView;
    CircleImageView photoProfile;
    TextView fullname, email;
    ModelUsers modelUsers;
    LottieAnimationView defaultPhoto;

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

        drawerLayout = (DrawerLayout) findViewById(R.id.drawer_layout);
        toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
```

```

        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setTitle("Ecommerce UMKM");
        toggle = new ActionBarDrawerToggle(this, drawerLayout, toolbar,
R.string.open, R.string.close);
        drawerLayout.addDrawerListener(toggle);
        toggle.syncState();
        navigationView = (NavigationView) findViewById(R.id.nav_view);

        loadFragment(new DashboardFragment());
        navigationView.setNavigationItemSelectedListener(new
NavigationView.OnNavigationItemSelectedListener() {
            @Override
            public boolean onNavigationItemSelectedListener(@NonNull MenuItem item)
        {

            int id = item.getItemId();
            switch (id) {
                case R.id.dashboard:
                    loadFragment(new DashboardFragment());
                    break;
                case R.id.me:
                    loadFragment(new AccountFragment());
                    break;
                case R.id.addToko:
                    loadFragment(new TokoFragment());
                    break;
                case R.id.addItem:
                    loadFragment(new ManageItemFragment());
                    break;
                case R.id.sendItem:
                    loadFragment(new SendItemFragment());
                    break;
                case R.id.Logout:
                    App.getPref().clear();
                    startActivity(new Intent(MainPedagang.this,
LoginSystem.class));
                    Animatoo.animateSlideUp(MainPedagang.this);
                    break;
            }
            return true;
        }
    });

    View headerItem = navigationView.getHeaderView(0);

    photoProfile = (CircleImageView)
headerItem.findViewById(R.id.photo_profile);
    fullname = (TextView) headerItem.findViewById(R.id.full_name);
    email = (TextView) headerItem.findViewById(R.id.email_address);
    defaultPhoto = (LottieAnimationView)
headerItem.findViewById(R.id.images);

    modelUsers = (ModelUsers) GsonHelper.parseGson(

```

```

        App.getPref().getString(Prefs.PREF_STORE_PROFILE, ""),
        new ModelUsers()
    );

    String dProfilePhoto = modelUsers.getFotoProfile();

    if (dProfilePhoto == null){
        photoProfile.setVisibility(View.GONE);
        defaultPhoto.setVisibility(View.VISIBLE);
    } else{
        defaultPhoto.cancelAnimation();
        defaultPhoto.setVisibility(View.GONE);
        photoProfile.setVisibility(View.VISIBLE);
        Picasso.get().load(BaseURL.baseUrl +
dProfilePhoto).into(photoProfile);
    }

    fullname.setText(modelUsers.getNama());
    email.setText(modelUsers.getEmail());
}

private void loadFragment(Fragment fragment) {
    FragmentManager f = getSupportFragmentManager();
    FragmentTransaction t = f.beginTransaction();
    t.replace(R.id.frame, fragment).commit();
    drawerLayout.closeDrawer(GravityCompat.START);
    t.addToBackStack(null);
}
}

```

Menu toko

```

private void checkDataBarang() {
    HashMap<String, String> params = new HashMap<String, String>();
    params.put("_id", _idPedagang);
    final JSONObjectRequest req = new JSONObjectRequest(Request.Method.GET,
BaseURL.checkBarang + _idPedagang, new JSONObject(params),
    new Response.Listener<JSONObject>() {
        @Override
        public void onResponse(JSONObject response) {
            System.out.println("response = " + response);
            hideDialog();
            try {
                boolean statusMsg = response.getBoolean("status");
                if (statusMsg == true) {
                    String toko = response.getString("result");
                    JSONArray jsonArray = new JSONArray(toko);
                    for (int i = 0; i < jsonArray.length(); i++) {
                        JSONObject jsonObject =
jsonArray.getJSONObject(i);
                        final ModelBarangs itemBarang = new
ModelBarangs();

```

```

        final String _id =
jsonObject.getString("idUser");
        final String namaBarang =
jsonObject.getString("namaBarang");
        final String deskripsiBarang =
jsonObject.getString("deskripsiBarang");
        final String hargaBarang =
jsonObject.getString("hargaBarang");
        final String stokBarang =
jsonObject.getString("stokBarang");
        final String fotoBarang =
jsonObject.getString("fotoBarang");
        itemBarang.setNamaBarang(namaBarang);

itemBarang.setDeksripsiBarang(deskripsiBarang);
        itemBarang.setHargBarang(hargaBarang);
        itemBarang.setStokBarang(stokBarang);
        itemBarang.setIdUser(_id);
        itemBarang.setFotobarang(fotoBarang);
        listItemPedagang.add(itemBarang);

recyclerItemPedagang.setAdapter(recycleViewAdapterItemPedagang);
    }
    noItemData.setVisibility(View.GONE);
    itemData.setVisibility(View.VISIBLE);
    } else {
        noItemData.setVisibility(View.VISIBLE);
        itemData.setVisibility(View.GONE);
    }
    } catch (JSONException e) {
        e.printStackTrace();
    }
    }
    }, new Response.ErrorListener() {
@Override
public void onErrorResponse(VolleyError error) {
    VolleyLog.e("Error: ", error.getMessage());
    hideDialog();
}
});
mRequestQueue.add(req);

```


Menu pembeli

```
package com.rian.ecommerce_v1.pembeli;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;

import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.widget.TextView;

import com.airbnb.lottie.LottieAnimationView;
import com.blogspot.atifsoftwares.animatoolib.Animatoo;
import com.rian.ecommerce_v1.R;
import com.rian.ecommerce_v1.access.LoginSystem;
import com.rian.ecommerce_v1.models.ModelUsers;
import com.rian.ecommerce_v1.server.BaseURL;
import com.rian.ecommerce_v1.utils.App;
import com.rian.ecommerce_v1.utils.GsonHelper;
import com.rian.ecommerce_v1.utils.Prefs;
import com.google.android.material.navigation.NavigationView;
import com.squareup.picasso.Picasso;

import de.hdodenhof.circleimageview.CircleImageView;

public class MainPembeli extends AppCompatActivity {

    DrawerLayout drawerLayout;
    ActionBarDrawerToggle toggle;
    Toolbar toolbar;
    NavigationView navigationView;
    CircleImageView photoProfile;
    TextView fullname, email;
    ModelUsers modelUsers;
    LottieAnimationView defaultPhoto;

    int id;

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

        drawerLayout = (DrawerLayout) findViewById(R.id.drawer_layout);
```

```

        toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setTitle("Ecommerce UMKM");
        toggle = new ActionBarDrawerToggle(this, drawerLayout, toolbar,
R.string.open, R.string.close);
        drawerLayout.addDrawerListener(toggle);
        toggle.syncState();
        navigationView = (NavigationView) findViewById(R.id.nav_view);

        loadFragment(new DashboardPembeliFragment());
        navigationView.setNavigationItemSelectedListener(new
NavigationView.OnNavigationItemSelectedListener() {
            @Override
            public boolean onNavigationItemSelectedListener(@NonNull MenuItem item)
{
                id = item.getItemId();
                switch (id) {
                    case R.id.dashboard_pembeli:
                        loadFragment(new DashboardPembeliFragment());
                        break;
                    case R.id.keranjang_pembeli:
                        loadFragment(new KeranjangPembeliFragment());
                        break;
                    case R.id.pembayaran_pembeli:
                        loadFragment(new PembayaranPembeliFragment());
                        break;
                    case R.id.pengiriman_pembeli:
                        loadFragment(new PengirimanSelesaiFragment());
                        break;
                    case R.id.pembelian_pembeli:
                        loadFragment(new LogPembelianFragment());
                        break;
                    case R.id.me_pembeli:
                        loadFragment(new AccountPembeliFragment());
                        break;
                    case R.id.logout_pembeli:
                        App.getPref().clear();
                        startActivity(new Intent(MainPembeli.this,
LoginSystem.class));
                        Animatoo.animateSlideUp(MainPembeli.this);
                        break;
                }
                return true;
            }
        });

        View headerItem = navigationView.getHeaderView(0);

        photoProfile = (CircleImageView)
headerItem.findViewById(R.id.photo_profile);
        fullname = (TextView) headerItem.findViewById(R.id.fullname);
        email = (TextView) headerItem.findViewById(R.id.email_address);

```

```

        defaultPhoto = (LottieAnimationView)
headerItem.findViewById(R.id.images);

        modelUsers = (ModelUsers) GsonHelper.parseGson(
            App.getPref().getString(Prefs.PREF_STORE_PROFILE, ""),
            new ModelUsers()
        );

        String dProfilePhoto = modelUsers.getFotoProfile();

        if (dProfilePhoto == null) {
            photoProfile.setVisibility(View.GONE);
            defaultPhoto.setVisibility(View.VISIBLE);
        } else {
            defaultPhoto.cancelAnimation();
            defaultPhoto.setVisibility(View.GONE);
            photoProfile.setVisibility(View.VISIBLE);
            Picasso.get().load(BaseURL.baseUrl +
dProfilePhoto).into(photoProfile);
        }

        fullname.setText(modelUsers.getNama());
        email.setText(modelUsers.getEmail());
    }

    private void loadFragment(Fragment fragment) {
        FragmentManager f = getSupportFragmentManager();
        FragmentTransaction t = f.beginTransaction();
        t.replace(R.id.frame, fragment).commit();
        drawerLayout.closeDrawer(GravityCompat.START);
        t.addToBackStack(null);
    }
}

```

keranjang pembeli

```

package com.rian.ecommerce_v1.pembeli;

import android.content.Intent;
import android.os.Bundle;

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.LinearLayout;

```

```

import android.widget.TextView;

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.VolleyLog;
import com.android.volley.toolbox.JsonObjectRequest;
import com.android.volley.toolbox.Volley;
import com.rian.ecommerce_v1.adapters.AdapterKeranjang;
import com.rian.ecommerce_v1.R;
import com.rian.ecommerce_v1.models.ModelKeranjang;
import com.rian.ecommerce_v1.models.ModelUsers;
import com.rian.ecommerce_v1.server.BaseURL;
import com.rian.ecommerce_v1.utils.App;
import com.rian.ecommerce_v1.utils.GsonHelper;
import com.rian.ecommerce_v1.utils.Prefs;
import com.rian.ecommerce_v1.utils.RupiahConvert;
import com.muddzdev.styleabletoastlibrary.StyleableToast;

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

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import java.util.HashMap;
import java.util.List;
import java.util.Locale;

public class KeranjangPembeliFragment extends Fragment {

    RecyclerView recyclerViewKeranjang;
    RecyclerView.Adapter recycleViewAdapterKeranjang;
    TextView dates, grandTotal, totalHargaBarang, biayaPengiriman;
    LinearLayout noItemData, availableData;
    Button clickItem;

    ModelUsers modelUsers;
    List<ModelKeranjang> listKeranjang;

    String _idPembeli, idPesanan;
    int biaya = 20000;
    int sum;
    private RequestQueue mRequestQueue;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
Bundle savedInstanceState) {
        View v = inflater.inflate(R.layout.fragment_keranjang_pembeli,
container, false);

```

```

        mRequestQueue = Volley.newRequestQueue(getActivity());

        noItemData = (LinearLayout) v.findViewById(R.id.no_item_keranjang);
        availableData = (LinearLayout)
v.findViewById(R.id.available_item_keranjang);
        clickItem = (Button) v.findViewById(R.id.btn_checkout);

        recyclerViewKeranjang = (RecyclerView)
v.findViewById(R.id.item_keranjang_pembeli);
        recyclerViewKeranjang.setHasFixedSize(true);
        recyclerViewKeranjang.setLayoutManager(new
LinearLayoutManager(getActivity()));
        listKeranjang = new ArrayList<>();
        recycleViewAdapterKeranjang = new AdapterKeranjang(getActivity(),
listKeranjang);

        modelUsers = (ModelUsers) GsonHelper.parseGson(
                App.getPref().getString(Prefs.PREF_STORE_PROFILE, ""),
                new ModelUsers()
        );

        _idPembeli = modelUsers.get_id();

        totalHargaBarang = (TextView) v.findViewById(R.id.total_harga);
        biayaPengiriman = (TextView) v.findViewById(R.id.biaya_pengiriman);

        biayaPengiriman.setText("Rp. 20.000,00");

        dates = (TextView) v.findViewById(R.id.date);
        grandTotal = (TextView) v.findViewById(R.id.grand_total);
        Date d = Calendar.getInstance().getTime();
        SimpleDateFormat dateFormat = new SimpleDateFormat("dd, MMMM yyyy",
Locale.getDefault());
        String formatDate = dateFormat.format(d);

        dates.setText(formatDate);

        clickItem.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                checkoutPesanan(_idPembeli);
            }
        });

        checkDataKeranjangPembeliFunction(_idPembeli);

        return v;
    }

    private void checkDataKeranjangPembeliFunction(String _idPembeli) {
        final JSONObjectRequest req = new
        JSONObjectRequest(Request.Method.GET, BaseURL.getDataPesanan + _idPembeli,

```

```

null,
        new Response.Listener<JSONObject>() {
            @Override
            public void onResponse(JSONObject response) {
                System.out.println("response = " + response);
                try {
                    boolean statusMsg =
response.getBoolean("status");
                    if (statusMsg == true) {
                        String pesanan =
response.getString("result");

                        JSONArray jsonArray = new
JSONArray(pesanan);
                        for (int i = 0; i < jsonArray.length();
i++) {
                            JSONObject jsonObject =
jsonArray.getJSONObject(i);
                            final ModelKeranjang keranjangD = new
ModelKeranjang();
                            final String namaToko =
jsonObject.getJSONObject("toko").getString("namaToko");
                            final String namaPesanan =
jsonObject.getJSONObject("items").getString("namaBarang");
                            final String fotoBarang =
jsonObject.getJSONObject("items").getString("fotoBarang");
                            final String hargaBarang =
jsonObject.getJSONObject("items").getString("hargaBarang");
                            final String deskripsiBarang =
jsonObject.getJSONObject("items").getString("deskripsiBarang");
                            final String totalHarga =
jsonObject.getString("total");
                            final String _idPesanan =
jsonObject.getString("_id");
                            final String qty =
jsonObject.getString("qty");
                            final String status =
jsonObject.getString("status");
                            if (status.equals("0")) {

noItemData.setVisibility(View.GONE);
availableData.setVisibility(View.VISIBLE);
                                keranjangD.setNamaToko(namaToko);

                                keranjangD.setDeksripsiBarang(deskripsiBarang);

                                keranjangD.setNamaBarang(namaPesanan);

                                keranjangD.setFotobarang(fotoBarang);
                                    keranjangD.set_id(_idPesanan);
                                    keranjangD.setQty(qty);
                            }
                        }
                    }
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        }
    }
}

```

```

keranjangD.setHargaBarang(hargaBarang);

keranjangD.setTotalHarga(totalHarga);

        idPesanan = _idPesanan;

        int dataharga =

Integer.parseInt(totalHarga);

        sum += dataharga;

totalHargaBarang.setText(RupiahConvert.convertRupiah(Integer.toString(sum))
);

        int grandTotalHarga = sum + biaya;

grandTotal.setText(RupiahConvert.convertRupiah(Integer.toString(grandTotalH
arga)));

        listKeranjang.add(keranjangD);

recyclerViewKeranjang.setAdapter(recycleViewAdapterKeranjang);
        } else {

noItemData.setVisibility(View.VISIBLE);

availableData.setVisibility(View.GONE);
        }
        }
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }, new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
            VolleyLog.e("Error: ", error.getMessage());
        }
    });
    mRequestQueue.add(req);
}

private void checkoutPesanan(String _idPembeli) {
    System.out.println("DATA ID PESANAN = " + _idPembeli);
    HashMap<String, String> params = new HashMap<String, String>();
    params.put("key", "1");

    final JSONObjectRequest req = new
    JSONObjectRequest(Request.Method.PUT, BaseURL.checkoutDataPesanan +
    _idPembeli, new JSONObject(params),
        new Response.Listener<JSONObject>() {

```

```

        @Override
        public void onResponse(JSONObject response) {
            System.out.println("response = " + response);
            try {
                JSONObject jsonObject = new
JSONObject(response.toString());
                System.out.println("res = " +
jsonObject.toString());
                String strMsg =
jsonObject.getString("message");
                boolean status =
jsonObject.getBoolean("status");
                if (status == true) {
                    StyleableToast.makeText(getActivity(),
                    "Data berhasil checkout", R.style.toastStyleSuccess).show();
                    startActivity(new Intent(getActivity(),
                    MainPembeli.class));
                }
            } catch (JSONException e) {
                e.printStackTrace();
            }
        },
        new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                VolleyLog.e("Error: ", error.getMessage());
            }
        });
        mRequestQueue.add(req);
    }
}

```