

LAMPIRAN

Lampiran 1 : kuisioner Usability testing

Kuisioner Tentang Aplikasi Pencarian Dan Pemesanan Tempat Kost Berbasis Android Di Ponorogo.

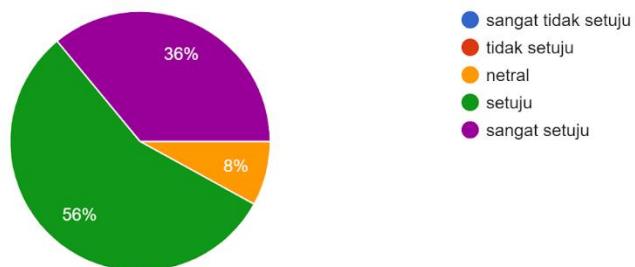
Keterangan :

(SS = Sangat Setuju, S = Setuju, N = Netral, TS = Tidak Setuju, STS = Sangat Tidak setuju)

<i>Aspek Learnability</i>						
Kode Soal	pertanyaan	SS	S	N	TS	STS
P1	Apakah aplikasi pencarian dan pemesanan tempat kost di ponorogo mudah di gunakan ?					
P2	Aplikasi pencarian dan pemesanan tempat kost di ponorogo sangat membantu ?					
P3	Apakah tombol dan icon di aplikasi pencarian dan pemesanan membantu dalam penggunaan ?					
P4	Apakah Jenis huruf yang digunakan pada aplikasi pencarian dan pemesanan tempat kost mudah dibaca ?					
P5	Apakah menu-menu yang ada cukup mudah untuk dipahami ?					
<i>Aspek Flexibility</i>						
Kode Soal	pertanyaan	SS	S	N	TS	STS
P6	Aplikasi pencarian dan pemesanan tempat kost dapat berjalan bersamaan dengan aplikasi lain ?					
P7	Apakah Fitur yang ada diaplikasi sudah sudah sesuai harapan pengguna ?					
<i>Aspek Effectiveness</i>						
Kode Soal	pertanyaan	SS	S	N	TS	STS
P8	Apakah Aplikasi pencarian dan pemesanan tempat kost menarik ?					
P9	Apakah Aplikasi pencarian dan pemesanan tempat kost menarik ?					
<i>Aspek Learnability</i>						
Kode Soal	pertanyaan	SS	S	N	TS	STS
P10	Aplikasi pencarian dan pemesanan tempat kost di ponorogo memberikan informasi yang lengkap tentang tempat kost					
P11	Apakah anda ingin memasang aplikasi pencarian dan pemesanan tempat kost di ponorogo untuk mencari tempat kost suatu saat ?					
P12	Apakah anda akan menyarankan orang lain untuk menginstal aplikasi pencarian dan pemesanan tempat kost di ponorogo ?					

Lampiran 2 : hasil kuisioner responden

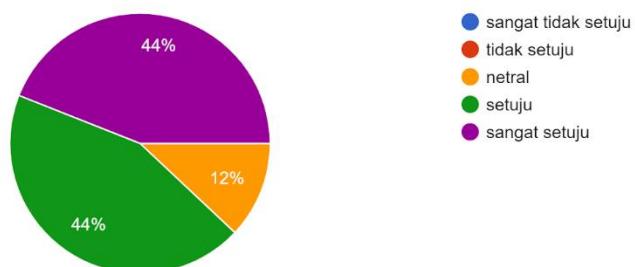
Apakah aplikasi pencarian dan pemesanan tempat kost di ponorogo mudah di gunakan ?
25 tanggapan



Keterangan :

Sangat setuju	= 9 responden
Setuju	= 14 responden
Netral	= 2 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

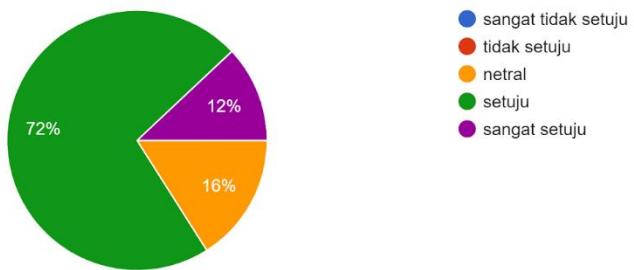
Aplikasi pencarian dan pemesanan tempat kost di ponorogo sangat membantu ?
25 tanggapan



Keterangan :

Sangat setuju	= 11 responden
Setuju	= 11 responden
Netral	= 3 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

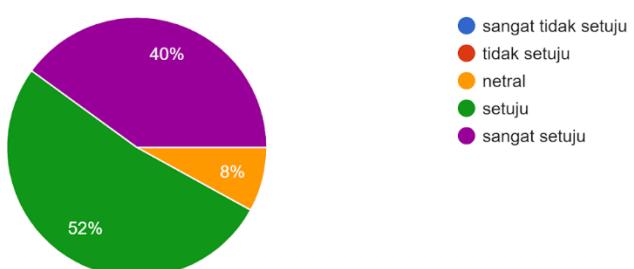
Apakah tombol dan icon di aplikasi pencarian dan pemesanan membantu dalam penggunaan ?
25 tanggapan



Keterangan :

Sangat setuju	= 3 responden
Setuju	= 18 responden
Netral	= 4 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

Apakah Jenis huruf yang digunakan pada aplikasi pencarian dan pemesanan tempat kost mudah dibaca ?
25 tanggapan

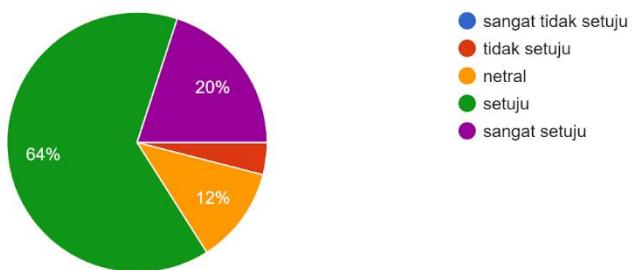


Keterangan :

Sangat setuju	= 10 responden
Setuju	= 13 responden
Netral	= 2 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

Apakah menu-menu yang ada cukup mudah untuk dipahami ?

25 tanggapan

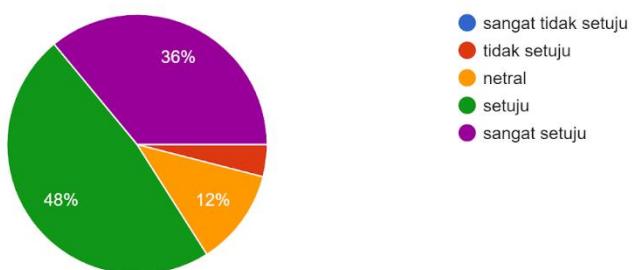


Keterangan :

Sangat setuju	= 5 responden
Setuju	= 16 responden
Netral	= 3 responden
Tidak setuju	= 1 responden
Sangat tidak setuju	= 0 responden

Aplikasi pencarian dan pemesanan tempat kost dapat berjalan bersamaan dengan aplikasi lain

25 tanggapan

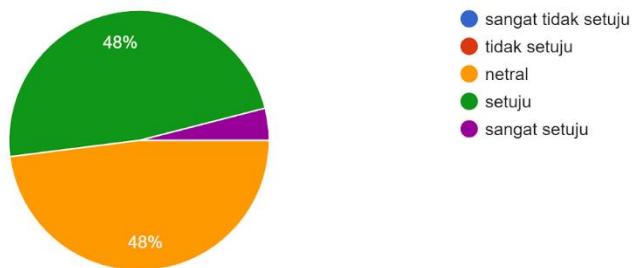


Keterangan :

Sangat setuju	= 9 responden
Setuju	= 12 responden
Netral	= 3 responden
Tidak setuju	= 1 responden
Sangat tidak setuju	= 0 responden

Apakah Fitur yang ada diaplikasi sudah sesuai harapan pengguna ?

25 tanggapan

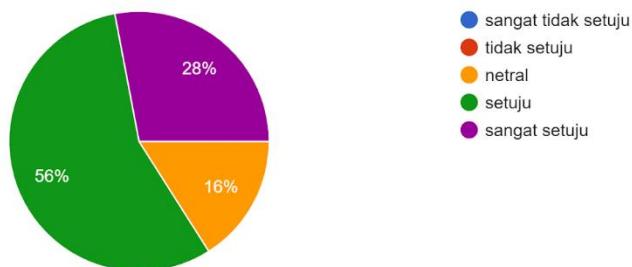


Keterangan :

Sangat setuju	= 1 responden
Setuju	= 12 responden
Netral	= 12 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

Apakah Aplikasi pencarian dan pemesanan tempat kost menarik ?

25 tanggapan

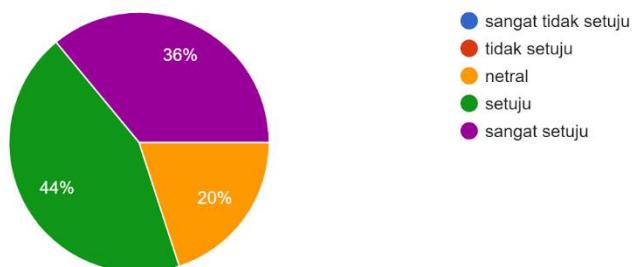


Keterangan :

Sangat setuju	= 7 responden
Setuju	= 14 responden
Netral	= 4 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

Apakah tidak membutuhkan waktu lama untuk mendapatkan informasi tempat kost ?

25 tanggapan

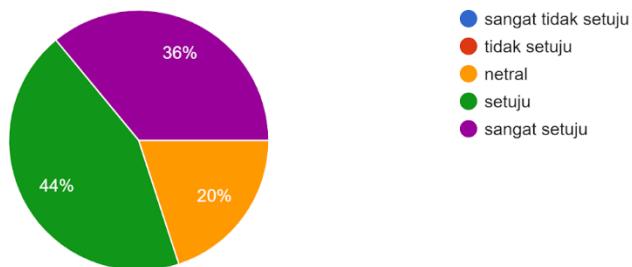


Keterangan :

Sangat setuju	= 9 responden
Setuju	= 11 responden
Netral	= 5 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

Aplikasi pencarian dan pemesanan tempat kost di ponorogo memberikan informasi yang lengkap tentang tempat kost

25 tanggapan

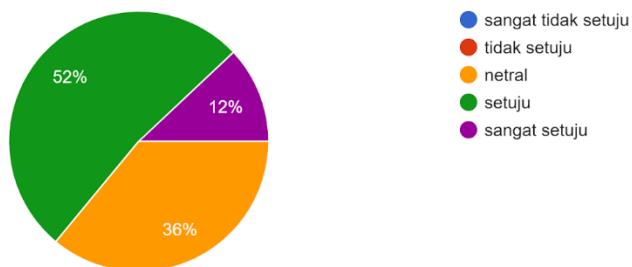


Keterangan :

Sangat setuju	= 9 responden
Setuju	= 11 responden
Netral	= 5 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

Apakah anda ingin memasang aplikasi pencarian dan pemesanan tempat kost di ponorogo untuk mencari tempat kost suatu saat ?

25 tanggapan

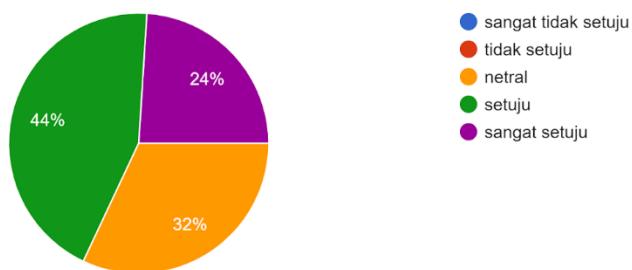


Keterangan :

Sangat setuju	= 3 responden
Setuju	= 13 responden
Netral	= 9 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

Apakah anda akan menyarankan orang lain untuk menginstal aplikasi pencarian dan pemesanan tempat kost di ponorogo ?

25 tanggapan



Keterangan :

Sangat setuju	= 6 responden
Setuju	= 11 responden
Netral	= 8 responden
Tidak setuju	= 0 responden
Sangat tidak setuju	= 0 responden

Lampiran 3 : Source Code Data Kamar

```

<?php
error_reporting(0);
include "koneksi.php";

$idcat=$_GET['idcat'];

$cari=$_GET['cari'];
$idkategori=$_GET['idkategori'];
$iduser=$_GET['iduser'];

// if ($cari=="'||$cari=="null") {
//     $cari="";
// }elseif ($idkategori=="'||$idkategori=="null") {
//     $idkategori="";
// }

$response = array();

if ( $idkategori!="null") {
    $q="SELECT a.*,b.*,c.Kategori,b.id as idkosan from user a
join kosan b on b.iduser=a.id
left join kategori c on b.idkategori=c.id where c.id='$idkategori' ";
}elseif ($cari=="null" ||$cari=="") {
    $q=" SELECT a.*,b.*,c.Kategori,b.id as idkosan from user a
join kosan b on b.iduser=a.id
left join kategori c on b.idkategori=c.id where  a.nama like '%$cari%' ";
}else{
    $q="SELECT a.*,b.*,c.Kategori,b.id as idkosan from user a
join kosan b on b.iduser=a.id
left join kategori c on b.idkategori=c.id ";
}

$result = mysql_query("$q");
$cekdata=mysql_num_rows($result);

if ($cekdata <1) {

    $response["success"] = 0;
    $response["message"] = "kosong";
    die(json_encode($response));
}
else{
    while($row = mysql_fetch_array($result)){
        // temporary array to create single category
        $tmp = array();
        $tmp["id_barang"] = $row["idkosan"];
        $tmp["nama"] = $row["namakosan"];
        $tmp["harga"] = rupiah($row["harga"]);
        $tmp["gambar"] = $row["gambar1"];
        $tmp["gambar2"] = $row["gambar2"];
        $tmp["gambar3"] = $row["gambar3"];
        $tmp["ambildeskripsi"] = $row["fasilitas"];

        $tmp["phone"] = $row["phone"];
        $tmp["jenis"] = $row["jeniskosan"];
        $tmp["lat"] = $row["Longitude"];
        $tmp["lng"] = $row["Latitude"];
        $tmp["message"] = "ada";

        array_push($response, $tmp);
    }
    array_push($response, $tmp);
}

$n++;
}

header('Content-Type: application/json');
echo json_encode($response);

```

Lampiran 4 : Detail Transaksi

```
<?php
error_reporting(0);
include "koneksi.php";

$iduser=$_GET['iduser'];
$idjual=$_GET['idjual'];
$response = array();

    | | $result = mysql_query("SELECT a.* ,sum(b.qty*c.harga)as totalharga,d.nama as nmkosan,c.gambar1,c.gambar
left join transaksi_jual_rinci b on a.id=b.id_jual
left join kosan c on b.id_kos=c.id
left join user d on c.iduser=d.id where a.id_member='".$iduser' and a.id='".$idjual' group by b.id_kos");

$count=mysql_num_rows($result);

$total=mysql_fetch_assoc(mysql_query("SELECT a.* ,sum(b.qty*c.harga)as totalharga,d.nama as nmkosan,c.g
left join transaksi_jual_rinci b on a.id=b.id_jual
left join kosan c on b.id_kos=c.id
left join user d on c.iduser=d.id where a.id_member='".$iduser' and a.id='".$idjual' group by b.id_jual"));

$alamat=mysql_fetch_assoc(mysql_query("SELECT * from user where id='".$iduser'"));

cekdata=mysql_num_rows($result);

if ($cekdata <1) {
    $response["success"] = 0;
    $response["message"] = "GAGAL!";
    die(json_encode($response));
}
else{
    while($row = mysql_fetch_array($result)){
// temporary array to create single category
$tmp = array();

    $totalperstok=$row["qty"]*$row["hargajual"];

    $tmp["nama"] = $row["nmkosan"];
    $tmp["harga"] = rupiah($row["harga"]);
    $tmp["gambar"] = $row["gambar1"];
    $tmp["struk"] = $row["struk"];
    $tmp["alamat"]=$alamat[alamat];

    $jumlahsemua=$total["totalharga"];
    $tmp["qty"] = $row["qty"];

    $tmp["total"] = rupiah($total["totalharga"]);
    if ($row[struk]== "") {
        | | $tmp["status"] = "Belum dibayar";
    }else{
        | | $tmp["status"] = "Sudah dibayar";
    }

    $tmp["count"]=$count;
}

array_push($response, $tmp);
}
```

Lampiran 5 : Source Code List kost Harvensine

```
<?php
error_reporting(0);
include "koneksi.php";

$idcat=$_GET['idcat'];

$cari=$_GET['cari'];
$idkategori=$_GET['idkategori'];
$iduser=$_GET['iduser'];

$lat=$_GET[lat];
$lng=$_GET[long];

$response = array();

$result = mysql_query("SELECT a.*,b.id as idkosan, round((6371 * ACOS(SIN(RADIANS(latitude)) * SIN(
*cos(RADIANS(latitude)) * COS(RADIANS($lat)))),2) AS jarak FROM user a left join kosan b on a.id=b.iduser HAVING
$cekdata=mysql_num_rows($result);

if ($cekdata <1) {
    $response["success"] = 0;
    $response["message"] = "kosong";
    die(json_encode($response));
}
else{
    while($row = mysql_fetch_array($result)){
        $tmp = array();
        $tmp["id_barang"] = $row["idkosan"];
        $tmp["nama"] = $row["namakosan"];
        $tmp["harga"] = rupiah($row["harga"]);
        $tmp["gambar"] = $row["gambar1"];
        $tmp["gambar2"] = $row["gambar2"];
        $tmp["gambar3"] = $row["gambar3"];
        $tmp["ambildeskripsi"] = $row["fasilitas"];

        $tmp["phone"] = $row["phone"];
        $tmp["jenis"] = $row["jeniskosan"];
        $tmp["lat"] = $row["Longitude"];
        $tmp["lng"] = $row["Latitude"];
        $tmp["message"] = "ada";

        array_push($response, $tmp);
    }
    $n++;
}

header('Content-Type: application/json');
echo json_encode($response);
```

Lampiran 6 : Source Code Kategori Kost

```
<?php
error_reporting(0);
include "koneksi.php";

$response = array();

$result = mysql_query("SELECT * from kategori");

$cekdata=mysql_num_rows($result);

if ($cekdata <1) {
    $response["success"] = 0;
    $response["message"] = "GAGAL!";
    die(json_encode($response));
}
else{
    while($row = mysql_fetch_array($result)){
        // temporary array to create single category
        $tmp = array();
        $tmp["id"] = $row["id"];
        $tmp["kategori"] = $row["Kategori"];
        array_push($response, $tmp);
    }
}
$n++;
}
```



Lampiran 7 : Source Code Pencarian Kost terdekat

```
private void OnGPS() {
    final AlertDialog.Builder builder = new AlertDialog.Builder( context: this );
    builder.setMessage("Enable GPS").setCancelable(false).setPositiveButton( text: "Yes", (dialog, which) + {
        startActivityForResult(new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
    }).setNegativeButton( text: "No", (dialog, which) + { dialog.cancel(); });
    final AlertDialog alertDialog = builder.create();
    alertDialog.show();
}
private void getLocation() {
    if (ActivityCompat.checkSelfPermission(
        context: this, Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(
        context: this, Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions( activity: this, new String[]{Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_LOCATION);
    } else {
        Location locationGPS = locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
        if (locationGPS != null) {
            double lat = locationGPS.getLatitude();
            double longi = locationGPS.getLongitude();
            ambillat = String.valueOf(lat);
            ambillong = String.valueOf(longi);

            showLocation.setText("Your Location: " + "\n" + "Latitude: " + latitude + "\n" + "Longitude: " + longitude);
        } else {
            Toast.makeText( context: this, text: "Unable to find location.", Toast.LENGTH_SHORT).show();
        }
    }
}

@Override
public void getItemOffsets(Rect outRect, View view, RecyclerView parent, RecyclerView.State state) {
    int position = parent.getChildAdapterPosition(view); // item position
    int column = position % spanCount; // item column

    if (includeEdge) {
        outRect.left = spacing - column * spacing / spanCount; // spacing - column * ((1f / spanCount) * spacing)
        outRect.right = (column + 1) * spacing / spanCount; // (column + 1) * ((1f / spanCount) * spacing)

        if (position < spanCount) { // top edge
            outRect.top = spacing;
        }
        outRect.bottom = spacing; // item bottom
    } else {
        outRect.left = column * spacing / spanCount; // column * ((1f / spanCount) * spacing)
        outRect.right = spacing - (column + 1) * spacing / spanCount; // spacing - (column + 1) * ((1f / spanCount) * spacing)
        if (position >= spanCount) {
            outRect.top = spacing; // item top
        }
    }
}
```

Lampiran 8 : Source Code List Kost Android

```
package com.example.pc_render.carikosan.listkosan;

import ...

public class ListKosan extends AppCompatActivity {

    RecyclerView mRecyclerview;
    RecyclerView.Adapter mAdapter;
    List<ModelData> mItems;
    Button btnInsert, btnDelete;
    private Button tampil;
    ProgressDialog pd;
    TextView kosong;

    String idcat,idkategori;
    SharedPreferences iduser;
    String ambiliduser;
    EditText pencarian;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.tampil_list_kosan);
        kosong=(TextView)findViewById(R.id.kosong);

        mRecyclerview = (RecyclerView) findViewById(R.id.recyclerviewTemp);
        pd = new ProgressDialog( context: ListKosan.this);
        mItems = new ArrayList<>();

        RecyclerView.LayoutManager mLayoutManager = new GridLayoutManager( context: this, spanCount: 2);
        mRecyclerview.setLayoutManager(mLayoutManager);
        mRecyclerview.addItemDecoration(new GridSpacingItemDecoration( spanCount: 2, dpToPx(10), includeEdge: true));
        mRecyclerview.setItemAnimator(new DefaultItemAnimator());
        mAdapter = new AdapterlistKosan( context: ListKosan.this, mItems);
        mRecyclerview.setAdapter(mAdapter);

        iduser = getSharedPreferences(
            Login.IDUSER,
            mode: Context.MODE_PRIVATE +
            Context.MODE_PRIVATE | Context.MODE_PRIVATE);
        ambiliduser = (iduser.getString(
            Login.KEY_IDUSER, defValue: "NA"));

        Intent data = getIntent();
        final int update = data.getIntExtra( name: "update", defaultValue: 0);
        idcat = data.getStringExtra( name: "idkategori");

        pencarian=(EditText)findViewById(R.id.pencarian);
        pencarian.setOnEditorActionListener((v, actionId, event) -> {
            if (actionId == EditorInfo.IME_ACTION_SEARCH) {
                mItems.clear();
                MengambildataPencarian();
                return true;
            }
        });
    }
}
```

```
        }
        return false;
    });

    if (update==200) {
        pencarian.setText(data.getStringExtra( name: "pencarian"));
        Mengambildatapencarian();

    }else if (update==5){
        idkategori=data.getStringExtra( name: "idkategori");
        MengambilData();
    }else{
        MengambilData();
    }
}

public static class GridSpacingItemDecoration extends RecyclerView.ItemDecoration {

    private int spanCount;
    private int spacing;
    private boolean includeEdge;

    public GridSpacingItemDecoration(int spanCount, int spacing, boolean includeEdge) {
        this.spanCount = spanCount;
        this.spacing = spacing;
        this.includeEdge = includeEdge;
    }

    @Override
    public void getItemOffsets(Rect outRect, View view, RecyclerView parent, RecyclerView.State state) {
        int position = parent.getChildAdapterPosition(view); // item position
        int column = position % spanCount; // item column

        if (includeEdge) {
            outRect.left = spacing - column * spacing / spanCount; // spacing - column * ((1f / spanCount) * spacing)
            outRect.right = (column + 1) * spacing / spanCount; // (column + 1) * ((1f / spanCount) * spacing)

            if (position < spanCount) { // top edge
                outRect.top = spacing;
            }
            outRect.bottom = spacing; // item bottom
        } else {
            outRect.left = column * spacing / spanCount; // column * ((1f / spanCount) * spacing)
            outRect.right = spacing - (column + 1) * spacing / spanCount; // spacing - (column + 1) * ((1f / spanCount) * spacing)
            if (position >= spanCount) {
                outRect.top = spacing; // item top
            }
        }
    }

    private int dpToPx(int dp) {
        Resources r = getResources();
        return Math.round(TypedValue.applyDimension(TypedValue.COMPLEX_UNIT_DIP, dp, r.getDisplayMetrics()));
    }
    private void MengambilData() {
```

```

private void MengambilData() {
    pd.setMessage("Please Wait...");
    pd.setCancelable(true);
    pd.show();
    System.out.println(Server.URL + "android/list_kosan.php?idcat="+idcat+"&idkategori="+idkategori+"&iduser="+ambiliduser);
    JsonArrayRequest reqData = new JsonArrayRequest(Request.Method.GET,
        url: Server.URL + "android/list_kosan.php?idcat="+idcat+"&idkategori="+idkategori+"&iduser="+ambiliduser, jsonRequest: null,
        response) {
        for (int i = 0; i < response.length(); i++) {
            try {
                JSONObject data = response.getJSONObject(i);
                ModelData md = new ModelData();
                md.setGambar(data.getString( name: "gambar"));
                md.setGambar2(data.getString( name: "gambar2"));
                md.setGambar3(data.getString( name: "gambar3"));
                md.setNama(data.getString( name: "nama"));
                md.setHarga(data.getString( name: "harga"));
                md.setIdbarang(data.getString( name: "id_barang"));
                md.setAmbildeskripsi(data.getString( name: "ambildeskripsi"));
                md.setLat(data.getString( name: "lat"));
                md.setLng(data.getString( name: "lng"));
                md.setPhone(data.getString( name: "phone"));
                md.setJenis(data.getString( name: "jenis"));

                if (data.getString( name: "message")=="kosong"){
                    kosong.setVisibility(View.VISIBLE);
                }else {
                    kosong.setVisibility(View.GONE);
                }
                mItems.add(md);
            } catch (JSONException e) {
                e.printStackTrace();
                System.out.println("data kosong");
            }
        }
        mAdapter.notifyDataSetChanged();
        pd.cancel();
    },
    (error) {
        pd.cancel();
        Toast.makeText(getApplicationContext(), "Ada Kesalahan Mohon Periksa Kembali", Toast.LENGTH_LONG).show();
    });
}

Config_app.getInstance().addToRequestQueue(reqData);
}

private void Mengambildatapencarian() {
    pd.setMessage("Please Wait...");
    pd.setCancelable(true);
    pd.show();
}

String input = pencarian.getText().toString();

System.out.println(Server.URL + "android/listpencarian.php?cari="+pencarian.getText().toString()+"&idkategori="+idkategori);

```

```

JSONArrayRequest reqData = new JSONArrayRequest(Request.Method.GET,
    url: Server.URL + "android/listpencarian.php?cari="+pencarian.getText().toString()+"&idkategori="+idkategori+"&iduser="+ambilidu
    (response) -> {

        for (int i = 0; i < response.length(); i++) {
            try {
                JSONObject data = response.getJSONObject(i);
                ModelData md = new ModelData();
                md.setGambar(data.getString( name: "gambar"));
                md.setGambar2(data.getString( name: "gambar2"));
                md.setGambar3(data.getString( name: "gambar3"));
                md.setName(data.getString( name: "nama"));
                md.setHarga(data.getString( name: "harga"));
                md.setId_stok(data.getString( name: "id_stok"));
                md.setIdbarang(data.getString( name: "id_barang"));
                md.setCekdiskon(data.getString( name: "cekdiskon"));
                md.setLat(data.getString( name: "lat"));
                md.setLng(data.getString( name: "lng"));
                md.setPhone(data.getString( name: "phone"));
                md.setJenis(data.getString( name: "jenis"));

                md.setAmbildeskripsi(data.getString( name: "ambildeskripsi"));
                md.setSisa(data.getString( name: "sisastok"));
                if (data.getString( name: "message")=="kosong"){
                    kosong.setVisibility(View.VISIBLE);
                }else {
                    kosong.setVisibility(View.GONE);
                }
                mItems.add(md);
            } catch (JSONException e) {
                e.printStackTrace();
                System.out.println("data kosong");
            }
        }
        mAdapter.notifyDataSetChanged();
        pd.cancel();
    },
    (error) -> {
        pd.cancel();
        Toast.makeText(getApplicationContext(), "Ada Kesalahan Mohon Periksa Kembali", Toast.LENGTH_LONG).show();
    });
    Config_app.getInstance().addToRequestQueue(reqData);
}
}

```