

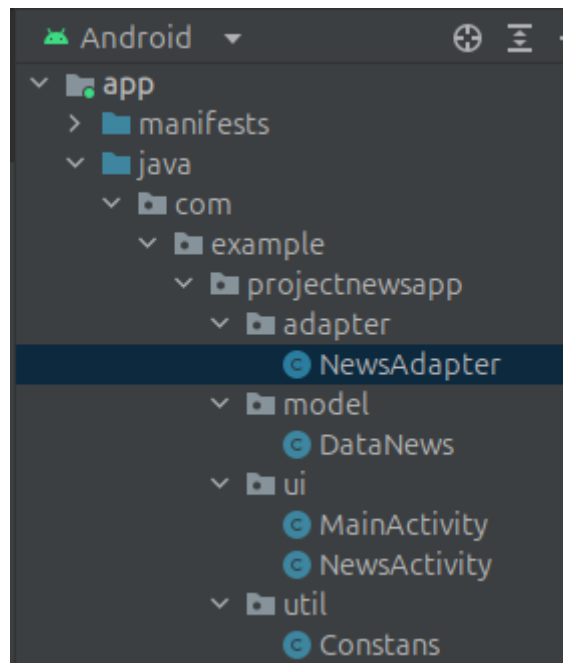
## Project II - News App

2x Pertemuan

### A. Deskripsi

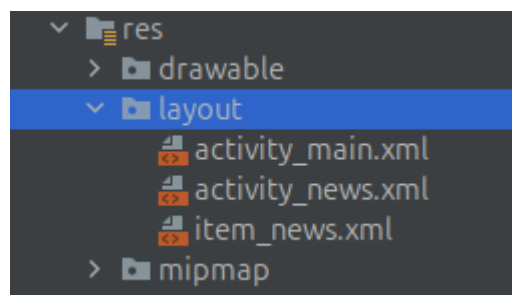
Pada Project II bertujuan untuk mengenal komponen baru yaitu Json, webview dan RecyclerView. Komponen tersebut untuk merancang dan membuat aplikasi Berita/Panduan dengan spesifikasi:

1. MainActivity.java
2. NewsActivity.java
3. Constans.java
4. DataNews.java
5. NewsAdapter.java



Gambare 1.0 Struktur java pada Project News App

6. activity\_main.xml
7. activity\_news.xml
8. item\_news.xml



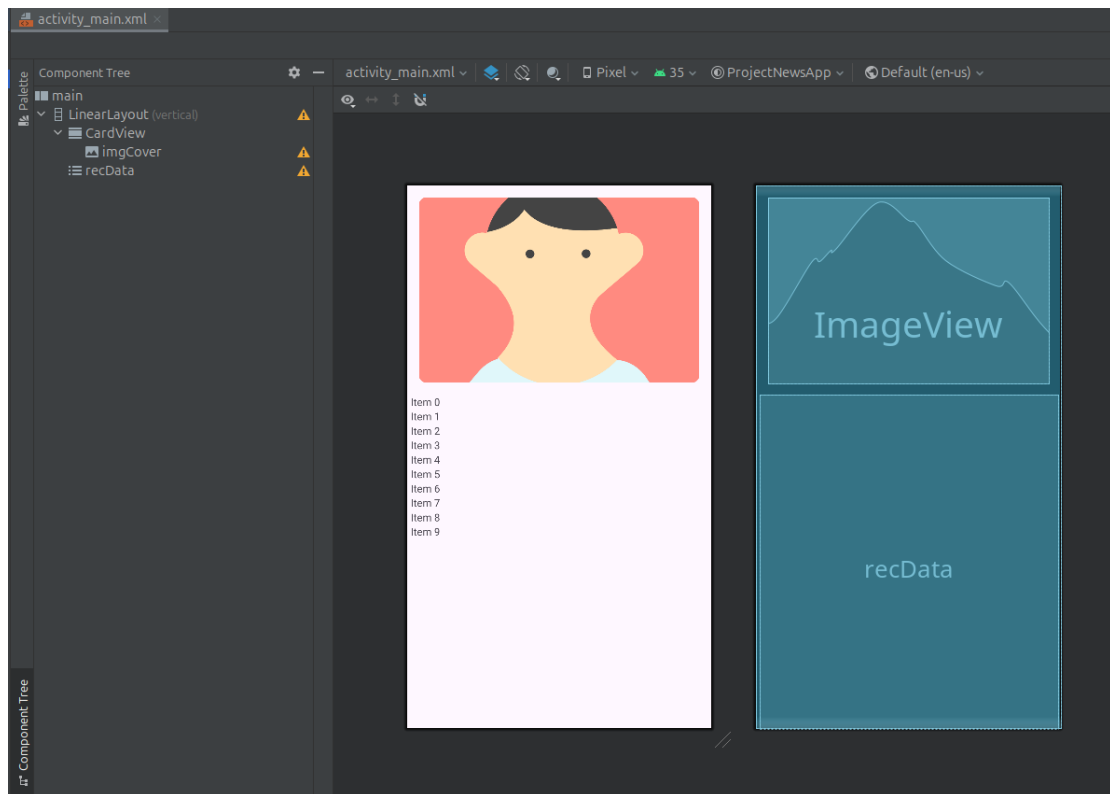
Gambar 1.1 Struktur xml pada Project News App

## B. Panduan

### 1. MainActivity

Halaman MainActivity merupakan activity untuk menampilkan list/daftar berita sebagai berikut:

- a. MainActivity.java
- b. activity\_main.xml
- c. item\_news.xml



Gambar 2.0 Desain dan Blueprint activity\_main.xml

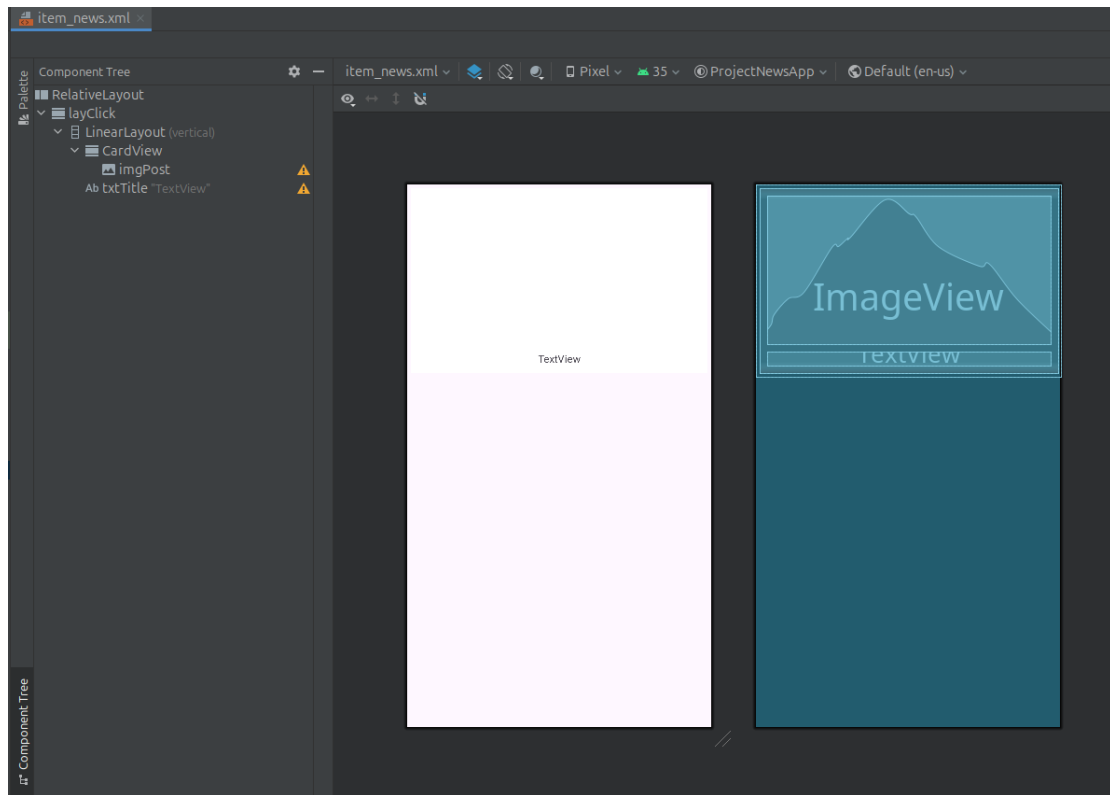
Pada desain activity\_utama.xml terdapat beberapa component yang digunakan, sebagai berikut:

- a. RelativeLayout
- b. LinerLayout Vertikal
- c. ImageView
- d. CardView
- e. Recyclerview

item\_news.xml merupakan desain untuk menampung data list/daftar berita yang akan ditampilkan di MainActivity.java, terdapat beberapa component yang digunakan, sebagai berikut:

- a. RelativeLayout
- b. LinerLayout Vertikal

- c. ImageView
- d. CardView
- e. TextView



Gambar 2.1 Desain dan Blueprint item\_news.xml

#### Code MainActivity.java

```
public class MainActivity extends AppCompatActivity {
    RecyclerView recData;
    NewsAdapter postAdapter;
    ArrayList<DataNews> newsList;
    ImageView imgCover;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        imgCover = findViewById(R.id.imgCover);
        recData = findViewById(R.id.recData);
        recData.setHasFixedSize(true);
        RecyclerView.LayoutManager mLayoutManager = new GridLayoutManager(MainActivity.this, 2);
        recData.setLayoutManager(mLayoutManager);
        newsList = new ArrayList<>();
        getDataNews();
    }
    public String loadJSONFromAsset() {
        String json = null;
        try {
            InputStream is = getAssets().open("news.json");
            int size = is.available();
            byte[] buffer = new byte[size];
            is.read(buffer);
```

```
is.close();
json = new String(buffer, StandardCharsets.UTF_8);
} catch (IOException ex) {
    ex.printStackTrace();
    return null;
}
return json;
}
}

public void getDataNews() {
    try {
        JSONObject jsonObject = new JSONObject(loadJSONFromAsset());
        JSONArray jsonArray = jsonObject.getJSONArray("DATA_NEWS");
        for (int i = 0; i < jsonArray.length(); i++) {
            JSONObject jo = jsonArray.getJSONObject(i);
            DataNews developers = new DataNews();
            developers.title = jo.getString("title_news");
            developers.link_image = jo.getString("image_link");
            developers.description = jo.getString("link_news");
            newsList.add(developers);
        }

        Random r = new Random();
        int numberCover = r.nextInt(newsList.size());
        Picasso.get().load(newsList.get(numberCover).getLink_image()).into(imgCover);
        imgCover.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Constans.TITLE = newsList.get(numberCover).getTitle();
                Constans.IMAGE = newsList.get(numberCover).getLink_image();
                Constans.DESCRPTION = newsList.get(numberCover).getDescription();
                Intent intent = new Intent(MainActivity.this, NewsActivity.class);
                startActivity(intent);
            }
        });

        postAdapter = new NewsAdapter( newsList, MainActivity.this);
        recData.setAdapter(postAdapter);

    } catch (JSONException e) {
        Toast.makeText(MainActivity.this, e.toString(), Toast.LENGTH_LONG).show();
    }
}
}
```

Code DataNews.java

```
public class DataNews {
    public String title;
    public String description;
    public String link_image;

    public void setTitle(String title) {
        this.title = title;
    }
    public String getTitle() {
        return title;
    }
}
```

```
public String getDescription() {  
    return description;  
}  
public String getLink_image() {  
    return link_image;  
}  
}
```

#### Code Constans.java

```
public class Constans {  
    public static String TITLE ;  
    public static String IMAGE;  
    public static String DESCRIPTION;  
}
```

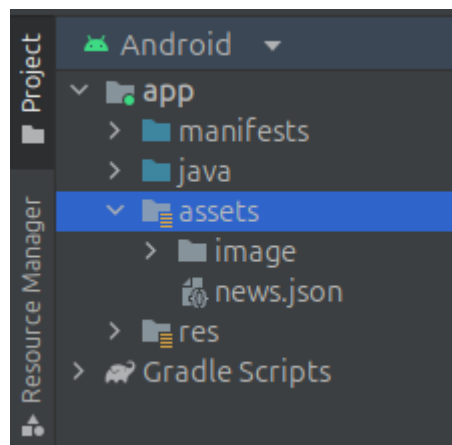
#### Code NewsAdapter.java

```
public class NewsAdapter extends RecyclerView.Adapter {  
    public static ArrayList<DataNews> pdflist;  
    public static ArrayList<DataNews> mFilteredList;  
    public Context context;  
  
    public NewsAdapter(ArrayList<DataNews> pdflist, Context context) {  
        this.pdflist = pdflist;  
        this.context = context;  
        mFilteredList = pdflist;  
    }  
  
    @Override  
    public RecyclerView.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  
        View v = LayoutInflater.from(parent.getContext())  
            .inflate(R.layout.item_news, parent, false);  
        return new ViewHolder(v);  
    }  
  
    @Override  
    public void onBindViewHolder(@NonNull final RecyclerView.ViewHolder holder,  
    @SuppressWarnings("RecyclerView") final int position) {  
  
        if (holder instanceof ViewHolder) {  
            final DataNews webList = mFilteredList.get(position);  
            ((ViewHolder) holder).txtTitle.setText(webList.getTitle());  
            Picasso.get()  
                .load(webList.getLink_image())  
                .into(((ViewHolder) holder).imgPost);  
  
            ((ViewHolder) holder).layClick.setOnClickListener(new View.OnClickListener() {  
                @Override  
                public void onClick(View v) {  
                    Constans.TITLE = webList.getTitle();  
                    Constans.IMAGE = webList.getLink_image();  
                    Constans.DESCRIPTION = webList.getDescription();  
                    Intent intent = new Intent(context, NewsActivity.class);  
                    context.startActivity(intent);  
                }  
            });  
        }  
    }  
}
```

```
    }  
    });  
}  
}  
  
public int getItemCount() {  
    return mFilteredList == null ? 0 : mFilteredList.size();  
}  
  
public class ViewHolder extends RecyclerView.ViewHolder {  
    public TextView txtTitle, txtShort, txtStart, txtDiscount, txtPrice;  
    public ImageView imgPost, imgBookmark;  
    public CardView layClick;  
  
    public ViewHolder(View itemView) {  
        super(itemView);  
        imgPost = itemView.findViewById(R.id.imgPost);  
        layClick = itemView.findViewById(R.id.layClick);  
        txtTitle = itemView.findViewById(R.id.txtTitle);  
    }  
}  
  
}
```

## 2. JSON File

Json merupakan format data yang digunakan untuk menyimpan dan menukar data secara terbuka yang bisa digunakan untuk berbagai bahasa pemrograman, Json tersimpan di folder Asset.



Gambar 2.1 news.json pada folder Asset

### Struktur Json

```
{  
  "DATA_NEWS": [  
    {  
      "title_news": "",  
      "link_news": "",  
      "image_link": ""  
    }  
  ]  
}
```

### Contoh Penulisan Json

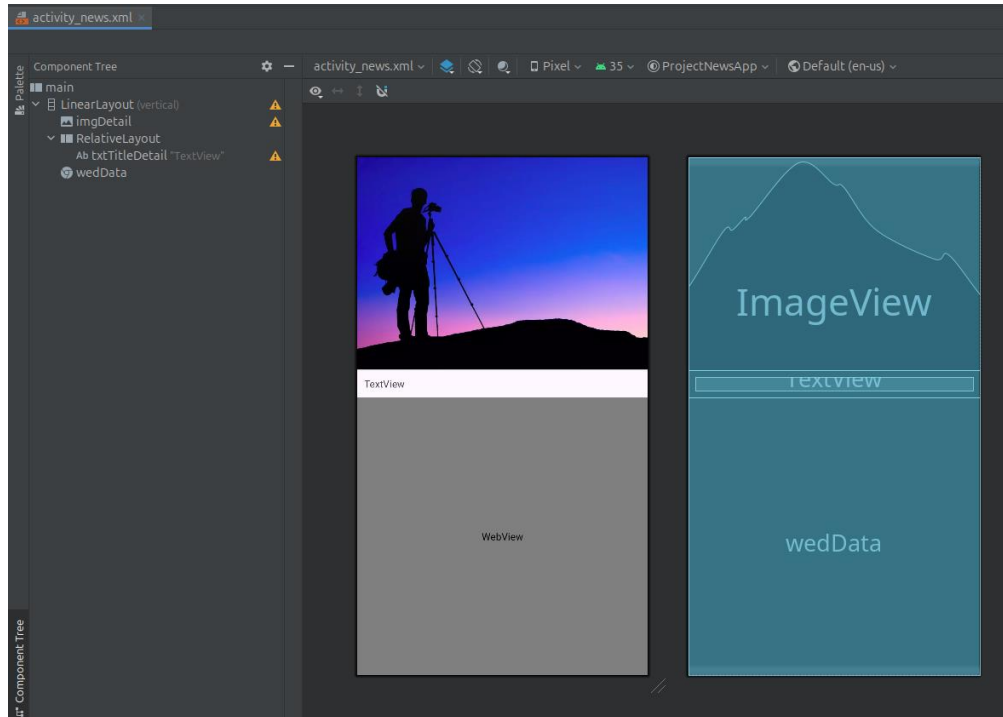
```
{
  "DATA_NEWS": [
    {
      "title_news": "Rice Cooker Cottage Cheese Cake",
      "link_news": "<p>Ini <span style=\"color: #ff0000;\">adalah</span> contoh <strong>penulisan</strong> berita pada aplikasi <span style=\"text-decoration: underline;\"><em><strong>Project News</strong></em></span></p>",
      "image_link": "file:///android_asset/image/cheese.webp"
    },
    {
      "title_news": "Easy butter croissants",
      "link_news": "<p>Ini <span style=\"color: #ff0000;\">adalah</span> contoh <strong>penulisan</strong> berita pada aplikasi <span style=\"text-decoration: underline;\"><em><strong>Project News</strong></em></span></p>",
      "image_link": "file:///android_asset/image/croissants.webp"
    },
    {
      "title_news": "Garlic butter steak bites",
      "link_news": "<p>Ini <span style=\"color: #ff0000;\">adalah</span> contoh <strong>penulisan</strong> berita pada aplikasi <span style=\"text-decoration: underline;\"><em><strong>Project News</strong></em></span></p>",
      "image_link": "file:///android_asset/image/steak.webp"
    },
    {
      "title_news": "Vegetable chow mein",
      "link_news": "<p>Ini <span style=\"color: #ff0000;\">adalah</span> contoh <strong>penulisan</strong> berita pada aplikasi <span style=\"text-decoration: underline;\"><em><strong>Project News</strong></em></span></p>",
      "image_link": "file:///android_asset/image/vegetable.webp"
    }
  ]
}
```

### 3. NewsActivity.java

Halaman NewsActivity merupakan activity untuk menampilkan detail news/panduan.

Komponen yang digunakan sebagai berikut:

- a. RelativeLayout
- b. LinerLayout Vertikal
- c. ImageView
- d. CardView
- e. TextView
- f. Webview



Gambar 3.0 Desain dan Blueprint activity\_news.xml

Code ActivityNews.java

```
public class NewsActivity extends AppCompatActivity {
    private WebView mWebView;
    private ImageView imgDetail;
    private TextView txtTitleDetail;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_news);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });
        imgDetail = findViewById(R.id.imgDetail);
        txtTitleDetail = findViewById(R.id.txtTitleDetail);
        mWebView = findViewById(R.id.wedData);

        //load title
        txtTitleDetail.setText(Constants.TITLE);

        //load image
        Picasso.get().load(Constants.IMAGE).into(imgDetail);

        //load description
        mWebView.loadDataWithBaseUrl(null, "<head><style>img{ max-width: 90%; width:auto; height:auto;}</style></head>" + Constants.DESCRPTION, "text/html", "UTF-8", null);
    }
}
```