

BAB VI

WORKMANAGER

6.1. Tujuan Pembelajaran

- a. Mahasiswa dapat mengerti apa yang dimaksud *Background Proses*
- b. Mahasiswa dapat mengerti apa yang dimaksud *WorkManager*
- c. Mahasiswa dapat menerapkan *WorkManager* kedalam Aplikasi

6.2. Software yang dibutuhkan

- a. Java JDK
- b. Android Studio 4.2
- c. SDK API 26

6.3. WorkManager

WorkManager merupakan bagian dari Android Jetpack dan merupakan sebuah Architecture Component untuk melakukan pekerjaan pada background yang memerlukan kombinasi dari eksekusi yang oportunistik dan terjamin. Eksekusi oportunistik berarti WorkManager akan melakukan pekerjaan pada background segera saat memungkinkan. Sedangkan eksekusi terjamin berarti WorkManager akan mengurus logic untuk memulai pekerjaan kalian dibawah berbagai macam situasi bahkan saat anda keluar dari aplikasi anda. Beberapa keuntungan WorkManager memiliki berbagai macam keuntungan.

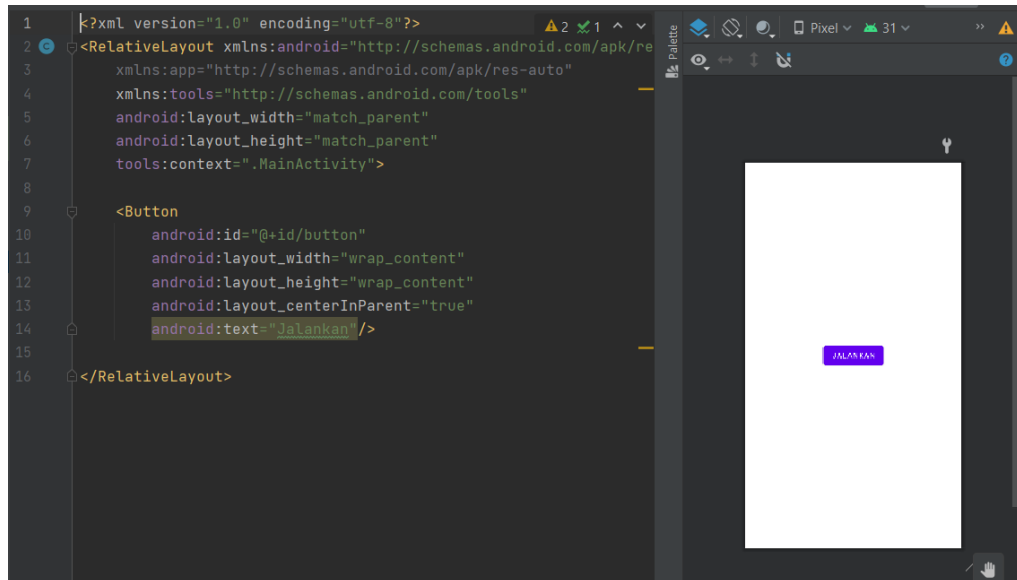
- a. Mendukung asynchronous one-off dan periodic tasks
- b. Mendukung constraint seperti kondisi jaringan, kapasitas penyimpanan dan status pengisian daya
- c. Mengikat permintaan pekerjaan yang kompleks, termasuk menjalankan pekerjaan secara paralel
- d. Mengoutputkan dari suatu permintaan pekerjaan sebagai input untuk pekerjaan selanjutnya
- e. Dapat bekerja baik dengan ataupun tanpa Google Play Services

6.4. Langkah – langkah Praktikum

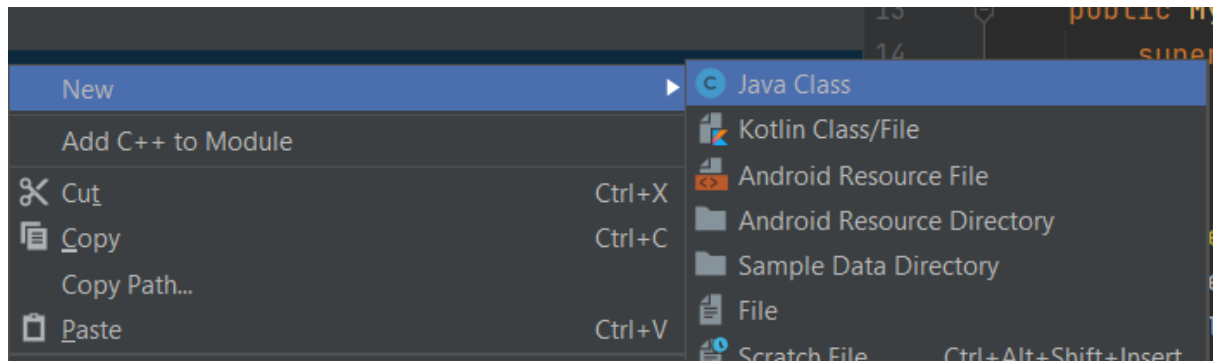
1. Buatlah project baru dengan “Empty Activity” sebagai starter seperti pada praktikum sebelumnya
2. Tambahkan library worker pada gradle, jika sudah bisa sync project

```
implementation "androidx.work:work-runtime:2.7.0"
```

3. Buka “activity_main.xml” lalu buatlah sebuah button seperti gambar dibawah



4. Buatlah class dengan nama “MyWorker” pada bagian folder package dengan cara klik kanan folder >> New >> Java Class\



5. Buka class MyWorker lalu extends dengan Packager Worker Secara Manual. Setelah itu isi source code seperti dibawah :

```
public class MyWorker extends Worker{
    public MyWorker(@NonNull Context context, @NonNull
    WorkerParameters workerParams) {
        super(context, workerParams);
    }

    @NonNull
    @Override
    public Result doWork() {
        displayNotification("Ini adalah Task Kamu", "Task kamu
```

```

sudah selesai");
        return Result.success();
    }

    public void displayNotification(String task, String desc){
        NotificationManager manager =
            (NotificationManager)
            getApplicationContext().getSystemService(Context.NOTIFICATION_SERVICE);

        NotificationChannel channel = new
            NotificationChannel("terserah",
                "test123", NotificationManager.IMPORTANCE_HIGH);
        manager.createNotificationChannel(channel);

        NotificationCompat.Builder builder = new
            NotificationCompat.Builder(getApplicationContext(),
                "terserah")
            .setContentTitle(task)
            .setContentText(desc)
            .setSmallIcon(R.mipmap.ic_launcher)
            .setPriority(NotificationCompat.PRIORITY_HIGH)

        .setCategory(NotificationCompat.CATEGORY_MESSAGE);

        manager.notify(1, builder.build());
    }
}

```

6. Buka class MainActivity.java lalu isi source code seperti dibawah :

```

public class MainActivity extends AppCompatActivity {

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

        final OneTimeWorkRequest request = new
            OneTimeWorkRequest.Builder(MyWorker.class).build();

        findViewById(R.id.button).setOnClickListener(new
            View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    WorkManager.getInstance().enqueueUniqueWork(
                        "Notifikasi",
                        ExistingWorkPolicy.REPLACE,
                        request);
                }
            });
    }
}

```

7. Build dan jalankan program kalian

6.5. Tugas Rumah

Buatlah proyek sederhana dengan mengimplementasikan workmanager didalamnya selain seperti proyek praktikum (onetime / periodic)