

## LAPORAN

### A. Pendahuluan

Laporan ini berisi dokumentasi mengenai kode aplikasi sederhana dengan tema pendidikan menggunakan bahasa java dan menggunakan framework NetBeans. Aplikasi ini dirancang untuk mengelola data terkait mata pelajaran, ruang ujian, jadwal, dan penjadwalan ujian.

### B. Struktur Program

Aplikasi Terdiri dari beberapa kelas:

1. App\_pendidikan1
  - Berfungsi sebagai titik awal program
  - Membuat objek dari kelas lain dan menjalankan penjadwalan ujian
2. MataPelajaran
  - Menyimpan informasi tentang mata pelajaran, seperti nama dan kode
3. RuangUjian
  - Menyimpan informasi mengenai ruang ujian dan kapasitas jumlah peserta
4. Jadwal
  - Menyimpan informasi hari dan waktu ujian

### App\_pendidikan1

```
public class App_pendidikan1 {  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        // TODO code application logic here  
        // Buat objek dari kelas lainnya  
        MataPelajaran jarkom = new MataPelajaran ("Jaringan Komputer", "MP001");  
        MataPelajaran pbo = new MataPelajaran ("PBO", "MP002");  
        RuangUjian ruangjarkom = new RuangUjian ("R101", 40);  
        RuangUjian ruangpbo = new RuangUjian ("R102", 30);  
        Jadwal jadwaljarkom = new Jadwal ("Senin", "10:00 - 12:00");  
        Jadwal jadwalpbo = new Jadwal ("Selasa", "09:30 - 11:30");  
  
        //Buat Penjadwalan  
        Penjadwalan penjadwalan = new Penjadwalan (jarkom, pbo, ruangjarkom, ruangpbo, jadwaljarkom, jadwalpbo);  
  
        //Tampilan informasi penjadwalan  
        penjadwalan.tampilkanDetailPenjadwalan();  
    }  
}
```

```
public class App_pendidikan1  
public static void main(String[] args) {  
  
    MataPelajaran jarkom = new MataPelajaran ("Jaringan Komputer", "MP001");  
    MataPelajaran pbo = new MataPelajaran ("PBO", "MP002");  
    RuangUjian ruangjarkom = new RuangUjian ("R101", 40);  
    RuangUjian ruangpbo = new RuangUjian ("R102", 30);  
}
```

```

Jadwal jadwaljarkom = new Jadwal ("Senin","10:00 - 12:00");
Jadwal jadwalpbo = new Jadwal ("Selasa","09:30 - 11:30");

Penjadwalan penjadwalan = new Penjadwalan
(jarkom,pbo,ruangjarkom,ruangpbo,jadwaljarkom,jadwalpbo);

    penjadwalan.tampilkanDetailPenjadwalan();
}
}

```

## MataPelajaran

```

//MataPelajaran.java
class MataPelajaran {
    private String nama;
    private String kode;

    public MataPelajaran(String nama,String kode){
        this.nama = nama;
        this.kode = kode;
    }

    public String getNama(){
        return nama;
    }

    public String getKode(){
        return kode;
    }
}

```

```

class MataPelajaran {
    private String nama;
    private String kode;

    public MataPelajaran(String nama,String kode){
        this.nama = nama;
        this.kode = kode;
    }
}

```

```

    }

    public String getNama(){
        return nama;
    }
    public String getKode(){
        return kode;
    }
}

```

## RuangUji

```

//RuangUji.java
class RuangUjian {
    private String namaRuang;
    private int kapasitas;

    public RuangUjian(String namaRuang,int kapasitas){
        this.namaRuang = namaRuang;
        this.kapasitas = kapasitas;
    }

    public String getNamaRuang() {
        return namaRuang;
    }
    public int getKapasitas() {
        return kapasitas;
    }
}

```

```

class RuangUjian {
    private String namaRuang;
    private int kapasitas;

    public RuangUjian(String namaRuang,int kapasitas){
        this.namaRuang = namaRuang;
        this.kapasitas = kapasitas;
    }
}

```

```
public String getNamaRuang(){
    return namaRuang;
}
public int getKapasitas(){
    return kapasitas;
}
}
```

**Jadwal**

```
//Jadwal.java
class Jadwal{
    private String hari;
    private String waktu;

    public Jadwal(String hari,String waktu){
        this.hari = hari;
        this.waktu = waktu;
    }

    public String getHari(){
        return hari;
    }
    public String getWaktu(){
        return waktu;
    }
}

class Jadwal{
    private String hari;
    private String waktu;

    public Jadwal(String hari,String waktu){
        this.hari = hari;
        this.waktu = waktu;
    }

    public String getHari(){
```

```

        return hari;
    }
    public String getWaktu(){
        return waktu;
    }
}

```

## Penjadwalan & Eksekusi

```

//Penjadwalan.java
class Penjadwalan{
    private MataPelajaran jarkom;
    private MataPelajaran pbo;
    private RuangUjian ruangjarkom;
    private RuangUjian ruangpbo;
    private Jadwal jadwaljarkom;
    private Jadwal jadwalpbo;

    public Penjadwalan(MataPelajaran jarkom, MataPelajaran pbo, RuangUjian ruangjarkom, RuangUjian
        this.jarkom = jarkom;
        this.pbo = pbo;
        this.ruangjarkom = ruangjarkom;
        this.ruangpbo = ruangpbo;
        this.jadwaljarkom = jadwaljarkom;
        this.jadwalpbo = jadwalpbo;
    }

    public void tampilkanDetailPenjadwalan(){
        System.out.println("---JADWAL UJIAN---");
        System.out.println("Mata Pelajaran: " + jarkom.getNama() + " (" + jarkom.getKode() + ")");
        System.out.println("Ruang: " + ruangjarkom.getNamaRuang() + " (Kapasitas: " + ruangjarkom.getKapasitas() + ")");
        System.out.println("Hari: " + jadwaljarkom.getHari());
        System.out.println("Waktu: " + jadwaljarkom.getWaktu());
        System.out.println("");
        System.out.println("");
        System.out.println("---JADWAL UJIAN---");
        System.out.println("Mata Pelajaran: " + pbo.getNama() + " (" + pbo.getKode() + ")");
        System.out.println("Ruang: " + ruangpbo.getNamaRuang() + " (Kapasitas: " + ruangpbo.getKapasitas() + ")");
        System.out.println("Hari: " + jadwalpbo.getHari());
        System.out.println("Waktu: " + jadwalpbo.getWaktu());
    }
}

```

```

class Penjadwalan{
    private MataPelajaran jarkom;
    private MataPelajaran pbo;
    private RuangUjian ruangjarkom;
    private RuangUjian ruangpbo;
    private Jadwal jadwaljarkom;
    private Jadwal jadwalpbo;
}

```

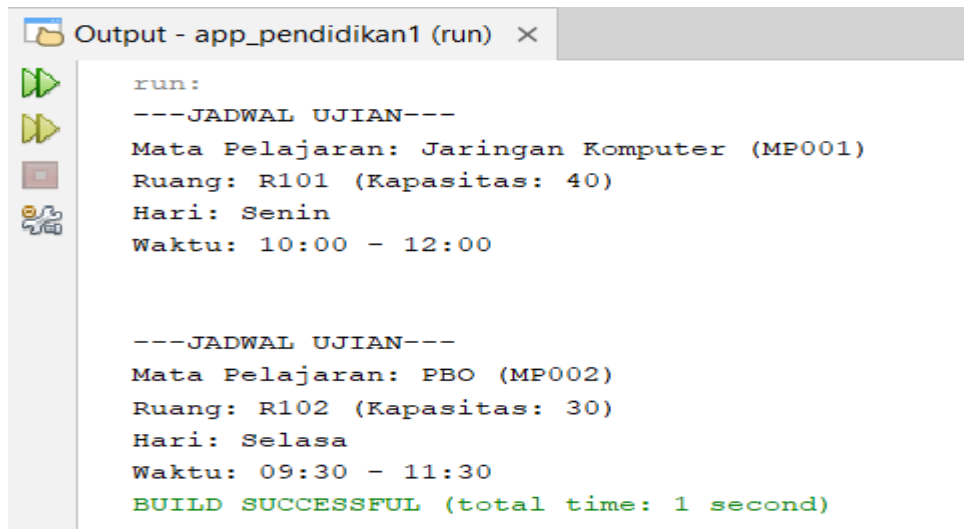
```

    public Penjadwalan(MataPelajaran jarkom, MataPelajaran pbo, RuangUjian
ruangjarkom, RuangUjian ruangpbo, Jadwal jadwaljarkom, Jadwal jadwalpbo){
        this.jarkom = jarkom;
        this.pbo = pbo;
        this.ruangjarkom = ruangjarkom;
        this.ruangpbo = ruangpbo;
        this.jadwaljarkom = jadwaljarkom;
        this.jadwalpbo = jadwalpbo;
    }

    public void tampilkanDetailPenjadwalan(){
        System.out.println("---JADWAL UJIAN---");
        System.out.println("Mata Pelajaran: " + jarkom.getNama() + " (" +
jarkom.getKode() + ")");
        System.out.println("Ruang: " + ruangjarkom.getNamaRuang() + " (Kapasitas:
" + ruangjarkom.getKapasitas() + ")");
        System.out.println("Hari: " + jadwaljarkom.getHari());
        System.out.println("Waktu: " + jadwaljarkom.getWaktu());
        System.out.println("");
        System.out.println("");
        System.out.println("---JADWAL UJIAN---");
        System.out.println("Mata Pelajaran: " + pbo.getNama() + " (" + pbo.getKode()
+ ")");
        System.out.println("Ruang: " + ruangpbo.getNamaRuang() + " (Kapasitas: " +
ruangpbo.getKapasitas() + ")");
        System.out.println("Hari: " + jadwalpbo.getHari());
        System.out.println("Waktu: " + jadwalpbo.getWaktu());
    }
}

```

## RUN



```
run:
---JADWAL UJIAN---
Mata Pelajaran: Jaringan Komputer (MP001)
Ruang: R101 (Kapasitas: 40)
Hari: Senin
Waktu: 10:00 - 12:00

---JADWAL UJIAN---
Mata Pelajaran: PBO (MP002)
Ruang: R102 (Kapasitas: 30)
Hari: Selasa
Waktu: 09:30 - 11:30
BUILD SUCCESSFUL (total time: 1 second)
```

## C. Penutup

Aplikasi yang saya bentuk dapat dikembangkan lebih lanjut untuk mendukung fitur tambahan, seperti integrasi dengan basis data, pengolahan lebih banyak mata pelajaran, serta antarmuka pengguna grafis. Dengan hasil konsep sekarang yang tergolong sangat dasar dan memiliki struktur kode yang jelas, pengembangan lanjutan diharapkan dapat dilakukan dengan mudah.