

ULANGAN AKHIR SEMESTER
OBJECT ORIENTED PROGRAMMING I



Disusun oleh :

Nama : Nurul Chotimah

Kelas : 4C

NIM : 18090046

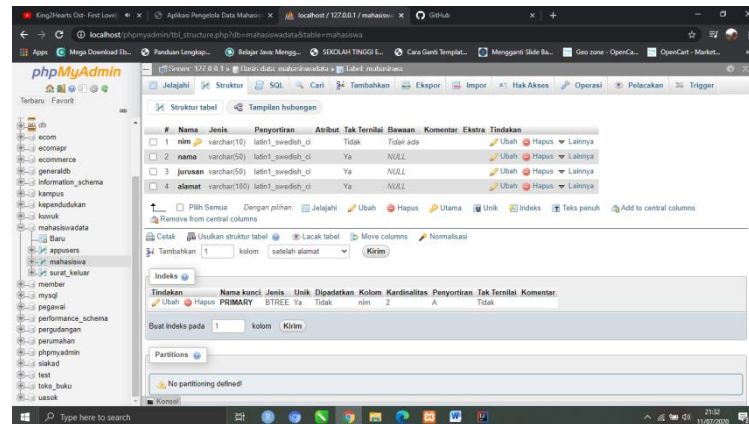
Dosen : Priyanto Tamami, S.Kom

DIV TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA TEGAL

2020

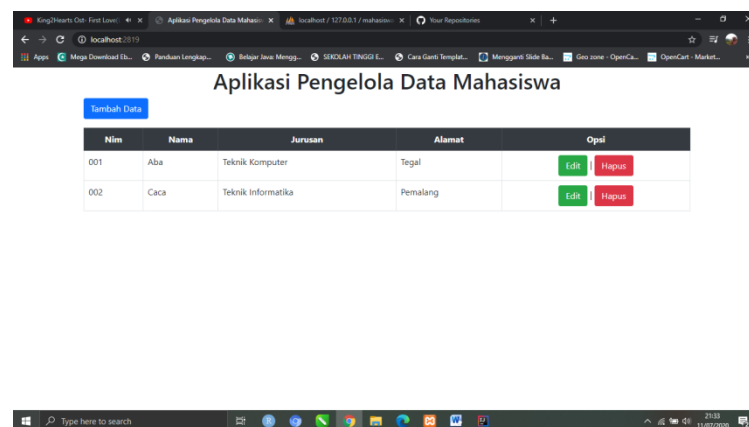
1. Pembentukan struktur tabel

Berikut adalah struktur table dari Data Mahasiswa



2. Desain aplikasi dan penjelasannya

Desain ini adalah halaman awal / utama, dimana data yang ada pada database ditampilkan. Disini berisi data – data dari mahasiswa dan berisi fitur diantaranya from tambah data, edit data, dan hapus data.



Fitur pertama tambah data, berikut ini adalah tampilan dari from tambah data dimana from ini berfungsi untuk menambahkan dari mahasiswa

Tambah Data Mahasiswa

[← Kembali](#)

Nim

Nama

Jurusan

Alamat

[Simpan](#)

Disini data dari mahasiswa diinputkan. Berikut adalah contohnya :

Tambah Data Mahasiswa

[← Kembali](#)

Nim

Nama

Jurusan

Alamat

[Simpan](#)

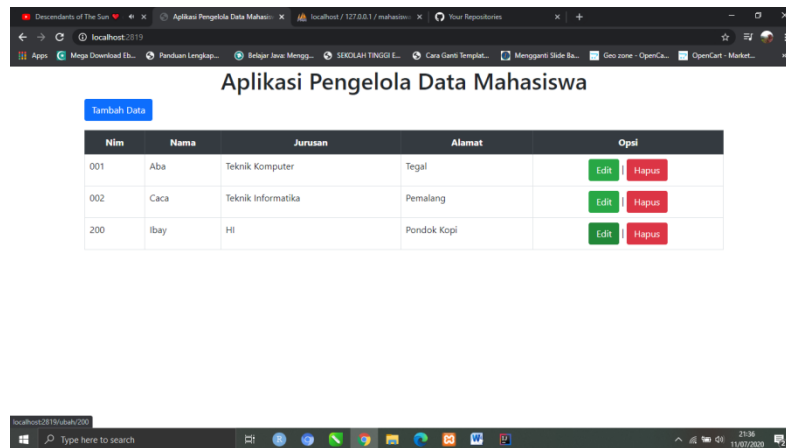
Setelah data berhasil diinputkan. Maka otomatis akan muncul di halaman utama

Aplikasi Pengelola Data Mahasiswa

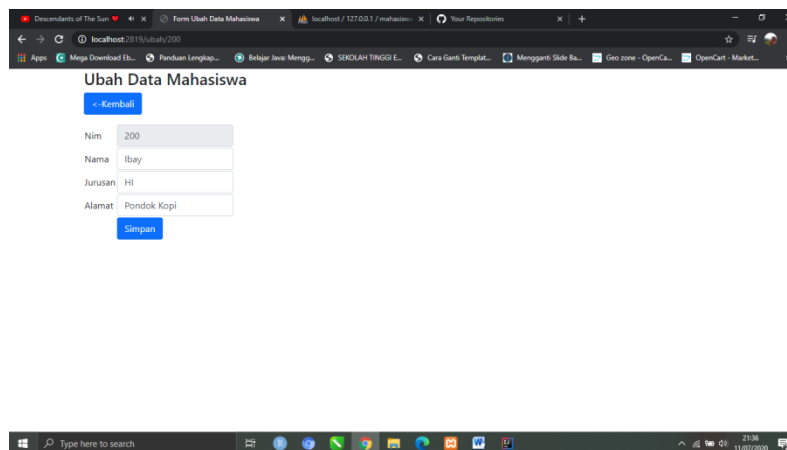
[Tambah Data](#)

Nim	Nama	Jurusan	Alamat	Opsl
001	Aba	Teknik Komputer	Tegal	Edit Hapus
002	Caca	Teknik Informatika	Pemalang	Edit Hapus
200	Ibay	HI	Pondok Kopi	Edit Hapus

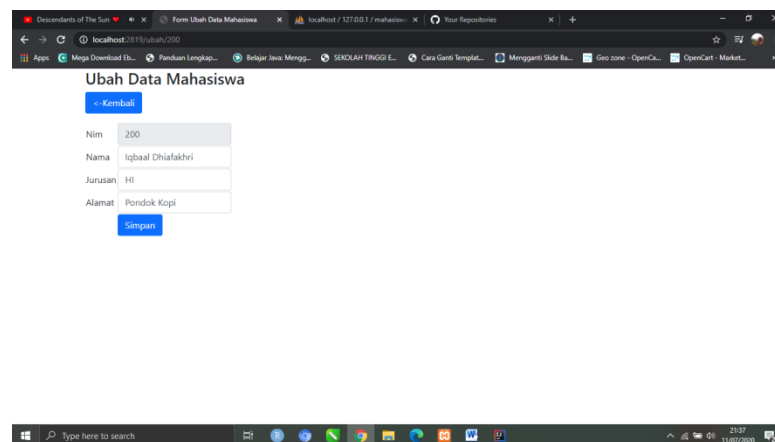
Fitur kedua adalah Edit data, kita tinggal mengklik Edit lalu akan membuka form edit data.



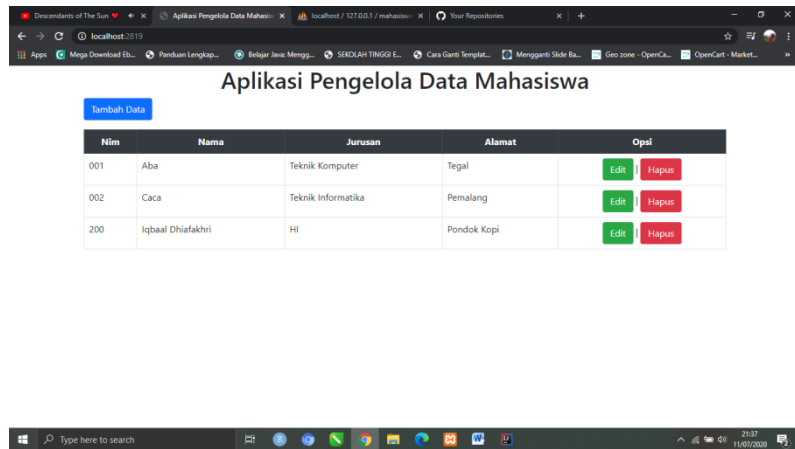
Ini adalah from dari Edit data, dimana edit data ini berfungsi untuk mengedit atau mengubah isi dari data mahasiswa



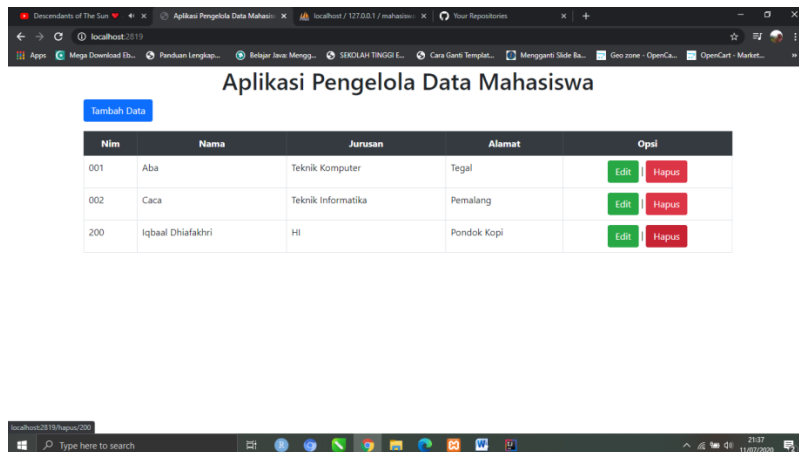
Missalnya data dari mahasiswa bernama Ibay akan di ubah menjadi Iqbaal Dhiafakhri.



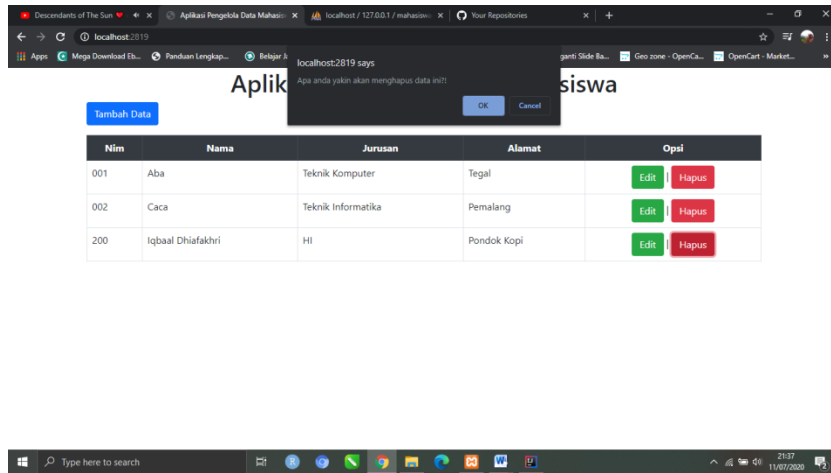
Setelah mengklik simpan maka akan data otomatis akan di perbarui. Yang sebelumnya Ibay menjadi Iqbaal Dhiafakhri.



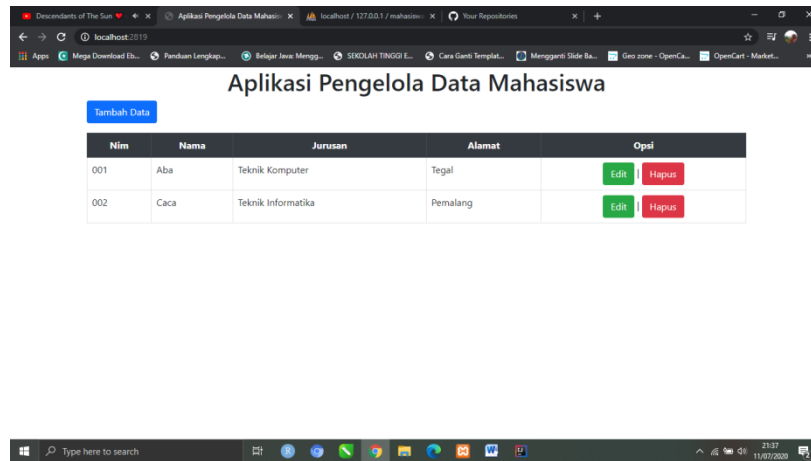
Fitur selanjutnya yaitu Hapus data. Dimana fitur ini berfungsi untuk menghapus data dari mahasiswa. Cara penggunaannya cukup dengan mengklik tombol hapus.



Lalu akan muncul notifikasi hapus. Klik sok maka data akan terhapus.

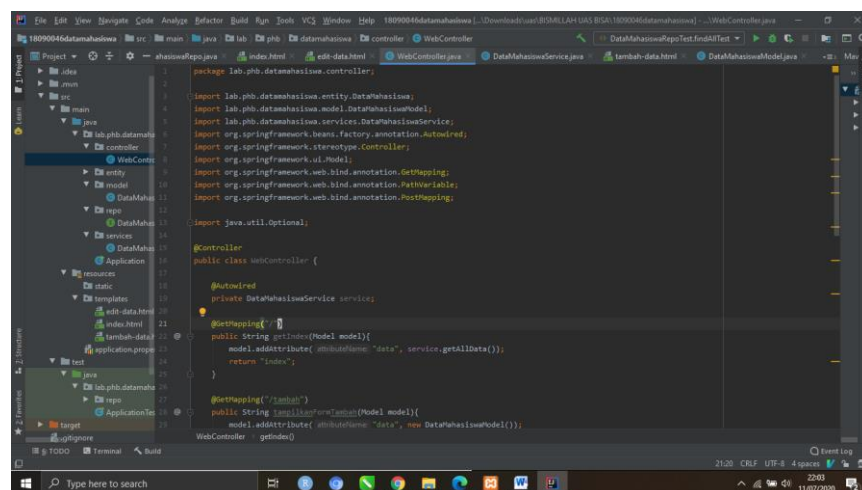


Ini adalah tampilan ketika data sudah terhapus yang sebelumnya ada 3 data menjadi 2 data.



3. Kode program

a. Web Controller



```

import javax.servlet.http.*;
import org.springframework.stereotype.*;
import org.springframework.ui.*;

@Controller
public class WebController {

    @RequestMapping("/simpan")
    public String simpan(DataMahasiswaModel data) {
        service.save(service.convertToEntity(data));
        return "redirect:/";
    }

    @RequestMapping("/ubah/{nia}")
    public String ubah(@PathVariable("nia") String nia, Model model) {
        Optional<DataMahasiswa> result = service.findById(nia);
        if(result.isPresent()) {
            DataMahasiswa data1 = result.get();
            model.addAttribute("data", data1);
            return "edit-data";
        } else {
            return "redirect:/";
        }
    }

    @RequestMapping("/hapus/{nia}")
    public String hapus(@PathVariable("nia") String nia) {
        service.removeId(nia);
        return "redirect:/";
    }
}

```

b. Entity

```

package lab.phb.datamahasiswa.entity;

import javax.persistence.*;
import org.hibernate.annotations.*;
import org.springframework.data.jpa.repository.*;

@Entity
@Table(name = "mahasiswa")
@Data
public class DataMahasiswa {

    @Id
    @Column(name = "nia")
    private String nia;

    @Column(name = "nama")
    private String nama;

    @Column(name = "jurusan")
    private String jurusan;

    @Column(name = "alamat")
    private String alamat;

    public String getNia() { return nia; }

    public void setNia(String nia) { this.nia = nia; }
}

```

```

package lab.phb.datamahasiswa.entity;

import javax.persistence.*;
import org.hibernate.annotations.*;
import org.springframework.data.jpa.repository.*;

@Entity
@Table(name = "mahasiswa")
@Data
public class DataMahasiswa {

    @Id
    @Column(name = "nia")
    private String nia;

    @Column(name = "nama")
    private String nama;

    @Column(name = "jurusan")
    private String jurusan;

    @Column(name = "alamat")
    private String alamat;

    public String getNia() { return nia; }

    public void setNia(String nia) { this.nia = nia; }

    public String getNama() { return nama; }

    public void setNama(String nama) { this.nama = nama; }

    public String getJurusan() { return jurusan; }

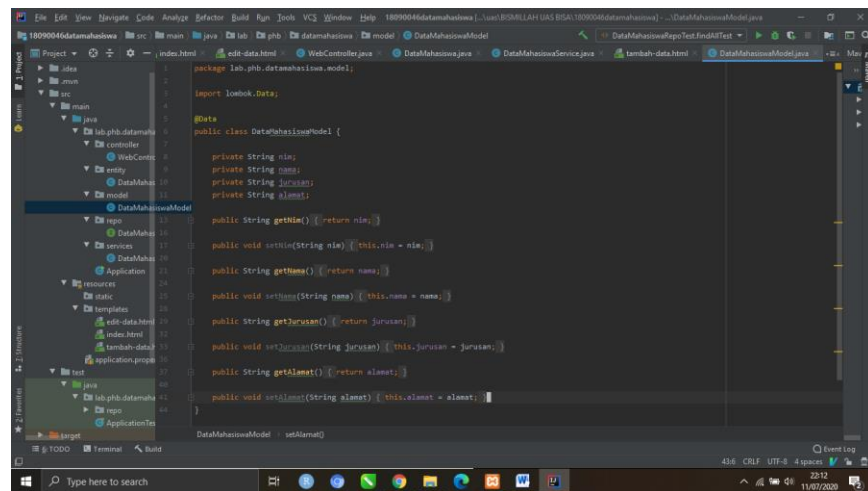
    public void setJurusan(String jurusan) { this.jurusan = jurusan; }

    public String getAlamat() { return alamat; }

    public void setAlamat(String alamat) { this.alamat = alamat; }
}

```

c. Model



```
package lab.phb.datamahasiswa.model;

import lombok.Data;

@Data
public class DataMahasiswaModel {

    private String nisn;
    private String nama;
    private String jurusan;
    private String alamat;

    public String getNisn() { return nisn; }
    public void setNisn(String nisn) { this.nisn = nisn; }

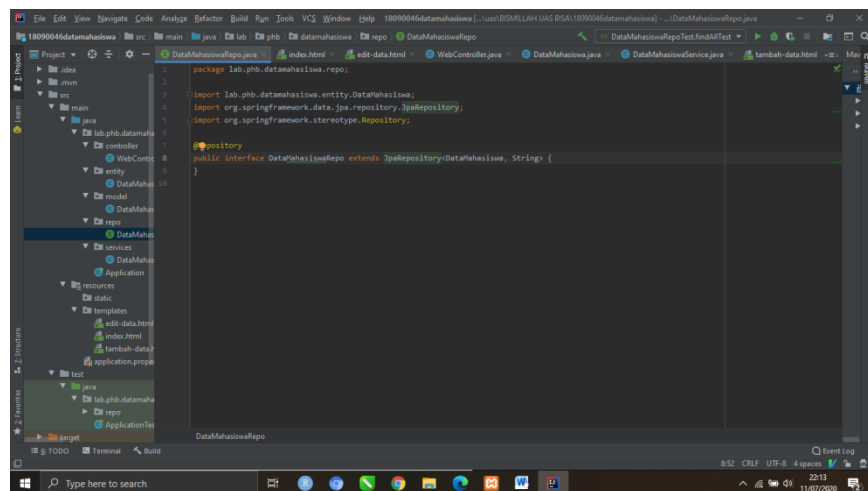
    public String getNama() { return nama; }
    public void setNama(String nama) { this.nama = nama; }

    public String getJurusan() { return jurusan; }
    public void setJurusan(String jurusan) { this.jurusan = jurusan; }

    public String getAlamat() { return alamat; }
    public void setAlamat(String alamat) { this.alamat = alamat; }

    DataMahasiswaModel() { setAlamat(); }
}
```

d. Repo



```
package lab.phb.datamahasiswa.repo;

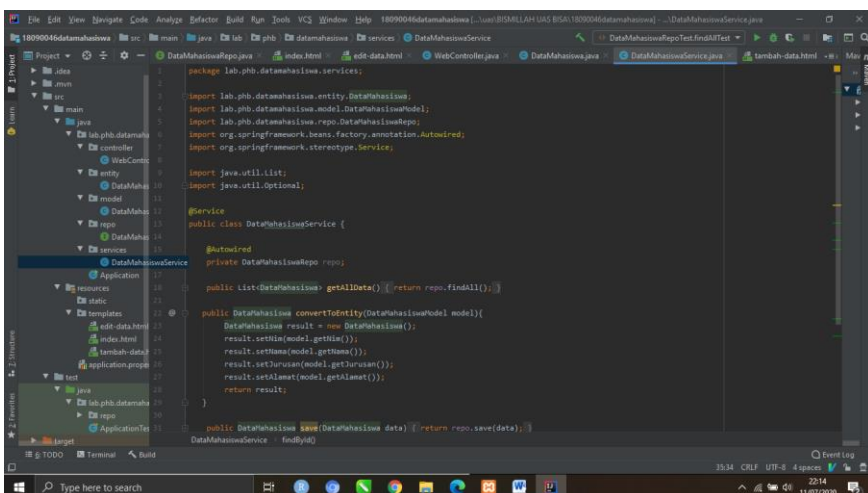
import lab.phb.datamahasiswa.entity.DataMahasiswa;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

@Repository
public interface DataMahasiswaRepo extends JpaRepository<DataMahasiswa, String> {

}

DataMahasiswaRepo
```

e. Services



```
package lab.phb.datamahasiswa.services;

import lab.phb.datamahasiswa.entity.DataMahasiswa;
import lab.phb.datamahasiswa.model.DataMahasiswaModel;
import lab.phb.datamahasiswa.repo.DataMahasiswaRepo;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.List;
import java.util.Optional;

@Service
public class DataMahasiswaService {

    @Autowired
    private DataMahasiswaRepo repo;

    public List<DataMahasiswa> getAllData() { return repo.findAll(); }

    public DataMahasiswa convertToEntity(DataMahasiswaModel model) {
        DataMahasiswa result = new DataMahasiswa();
        result.setNisn(model.getNisn());
        result.setNama(model.getNama());
        result.setJurusan(model.getJurusan());
        result.setAlamat(model.getAlamat());
        return result;
    }

    public DataMahasiswa save(DataMahasiswa data) { return repo.save(data); }

    DataMahasiswaService() { findById(); }
}
```



```

<tr>
<td>nama</td>
<td><input class="form-control" type="text" value="" id="nama"></td>
</tr>
<tr>
<td>jurusan</td>
<td><input class="form-control" type="text" value="" id="jurusan"></td>
</tr>
<tr>
<td>alamat</td>
<td><input class="form-control" type="text" value="" id="alamat"></td>
</tr>
<tr>
<td></td>
<td><input class="btn btn-primary" type="submit" value="simpan"></td>
</tr>
</table>
</div>
</div>
<!-- Popper.js first, then Bootstrap 5 -->
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E97vFbigrT7KFLnXBFI悦" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/5.0.0-alpha1/js/bootstrap.min.js" integrity="sha384-qj1Xv6qdsiguweoo4ClQ9eNf3SG58ttx1wzwgoS" crossorigin="anonymous"></script>
</body>
</html>

```

h. Index

```

<DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/5.0.0-alpha1/css/bootstrap.min.css" integrity="sha384-Q6E97vFbigrT7KFLnXBFI悦" crossorigin="anonymous">
<title>Aplikasi Pengelolaan Data Mahasiswa</title>
</head>
<body>
<div class="text-center">
<h1>Aplikasi Pengelolaan Data Mahasiswa</h1>
</div>
<div class="container">
<div class="text-center">
<button class="btn btn-primary mb-3">Tambah Data</button>
</div>
<div class="table">
<table class="table table-bordered">
<thead>
<tr>
<th class="text-center">Nama</th>
<th class="text-center">Jurusan</th>
<th class="text-center">Alamat</th>
</tr>
</thead>
<tbody>
<tr>
<td class="text-center">{{data.nama}}</td>
<td class="text-center">{{data.jurusan}}</td>
<td class="text-center">{{data.alamat}}</td>
</tr>
</tbody>
</table>
</div>
</div>

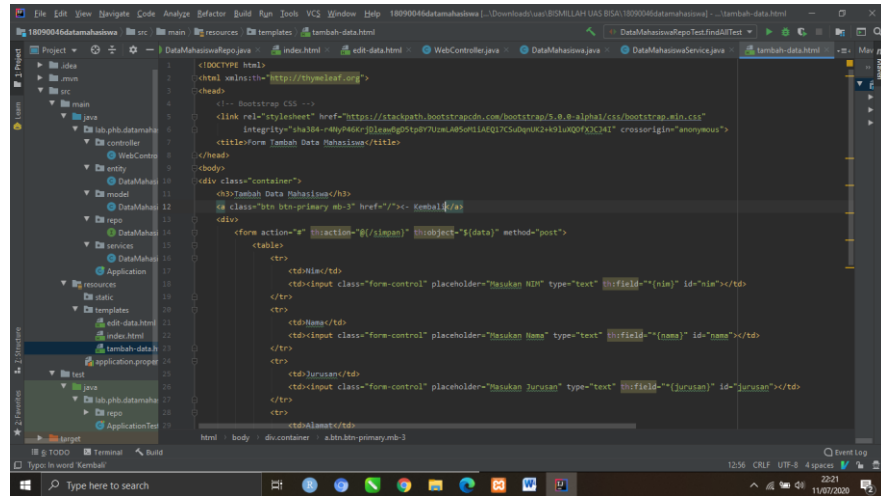
```

```

<DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/5.0.0-alpha1/css/bootstrap.min.css" integrity="sha384-Q6E97vFbigrT7KFLnXBFI悦" crossorigin="anonymous">
<title>Aplikasi Pengelolaan Data Mahasiswa</title>
</head>
<body>
<div class="text-center">
<h1>Aplikasi Pengelolaan Data Mahasiswa</h1>
</div>
<div class="container">
<div class="text-center">
<button class="btn btn-primary mb-3">Tambah Data</button>
</div>
<div class="table">
<table class="table table-bordered">
<thead>
<tr>
<th class="text-center">Nama</th>
<th class="text-center">Jurusan</th>
<th class="text-center">Alamat</th>
</tr>
</thead>
<tbody>
<tr>
<td class="text-center">{{data.nama}}</td>
<td class="text-center">{{data.jurusan}}</td>
<td class="text-center">{{data.alamat}}</td>
</tr>
</tbody>
</table>
</div>
</div>

```

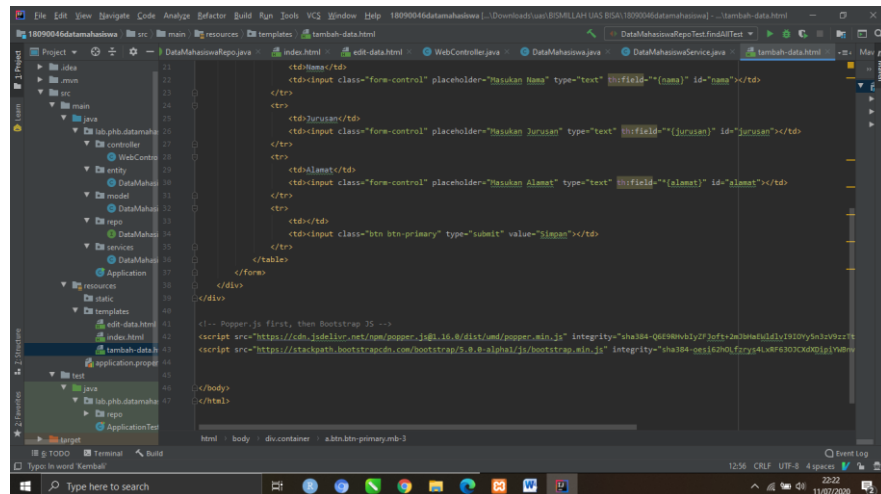
i. Tambah_data



```

1 <!DOCTYPE html>
2 <html xmlns:th="http://thymeleaf.org">
3 <head>
4 <!-- Bootstrap CSS -->
5 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/5.0.0-alpha1/css/bootstrap.min.css"
6 integrity="sha384-rh4p46rJ5owf8y80U9wL8q3R3611L9Q7S50qK2k49L0Q9X3C341" crossorigin="anonymous">
7 <title>Form Tambah Data Mahasiswa</title>
8 </head>
9 <body>
10 <div class="container">
11 <h3>Tambah Data Mahasiswa</h3>
12 <div class="btn btn-primary mb-3" href="#">Kembali</div>
13 <div>
14 <form action="#">
15 <table>
16 <tr>
17 <td>Nim</td>
18 <td><input class="form-control" placeholder="Masukan Nim" type="text" th:field="nim" id="nim"/></td>
19 </tr>
20 <tr>
21 <td>Nama</td>
22 <td><input class="form-control" placeholder="Masukan Nama" type="text" th:field="nama" id="nama"/></td>
23 </tr>
24 <tr>
25 <td>Jurusan</td>
26 <td><input class="form-control" placeholder="Masukan Jurusan" type="text" th:field="jurusan" id="jurusan"/></td>
27 </tr>
28 <tr>
29 <td colspan="2"><input class="btn btn-primary" type="submit" value="Simpan"></td>
30 </tr>
31 </table>
32 </div>
33 </div>
34 </body>
35 </html>

```

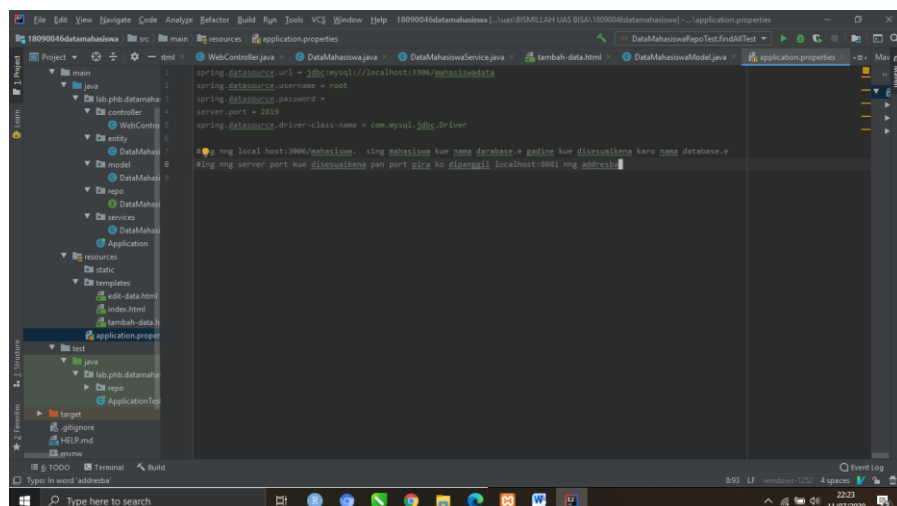


```

1 <!DOCTYPE html>
2 <html xmlns:th="http://thymeleaf.org">
3 <head>
4 <!-- Bootstrap CSS -->
5 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/5.0.0-alpha1/css/bootstrap.min.css"
6 integrity="sha384-rh4p46rJ5owf8y80U9wL8q3R3611L9Q7S50qK2k49L0Q9X3C341" crossorigin="anonymous">
7 <title>Form Tambah Data Mahasiswa</title>
8 </head>
9 <body>
10 <div class="container">
11 <h3>Tambah Data Mahasiswa</h3>
12 <div class="btn btn-primary mb-3" href="#">Kembali</div>
13 <div>
14 <form action="#">
15 <table>
16 <tr>
17 <td>Nim</td>
18 <td><input class="form-control" placeholder="Masukan Nim" type="text" th:field="nim" id="nim"/></td>
19 </tr>
20 <tr>
21 <td>Nama</td>
22 <td><input class="form-control" placeholder="Masukan Nama" type="text" th:field="nama" id="nama"/></td>
23 </tr>
24 <tr>
25 <td>Jurusan</td>
26 <td><input class="form-control" placeholder="Masukan Jurusan" type="text" th:field="jurusan" id="jurusan"/></td>
27 </tr>
28 <tr>
29 <td colspan="2"><input class="btn btn-primary" type="submit" value="Simpan"></td>
30 </tr>
31 </table>
32 </div>
33 </div>
34 </body>
35 </html>

```

j. Application_propertis



```

1 spring.datasource.url=jdbc:mysql://localhost:3306/mahasiswa
2 spring.datasource.username=root
3 spring.datasource.password=
4 spring.datasource.driver-class-name=com.mysql.jdbc.Driver
5 server.port=8080
6
7 #Mng local host:8080/mahasiswa, mng mahasiswa kue pama database.c gading kue disusulting kare pama database.c
8 #Mng mng server port kue disusulting pan port pira ke disusulting localhost:8080 mng address

```

k. Pom.xml

