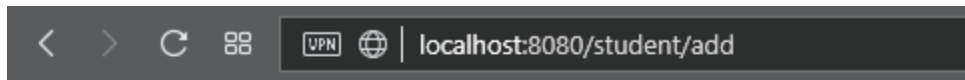


Ringkasan

Pada tutorial 3 ini saya memperdalam pengetahuan tentang MVC. Pengetahuan yang didapat adalah bagaimana membuat class model, service, controller dan cara untuk menghubungkan ketiganya. Selain itu saya juga mempelajari cara implementasi terhadap konsep *model* dan *service* dengan Spring Boot. Model adalah suatu objek yang menyimpan informasi terhadap sesuatu. Model digunakan untuk merepresentasikan segala macam atribut yang dimiliki oleh model. Service adalah suatu layer yang menjadi mediator atau perantara antara controller dan database. Terdapat service layer yang menjadi tempat menyimpan business logic yang digunakan untuk mengolah data dari database.

Tutorial

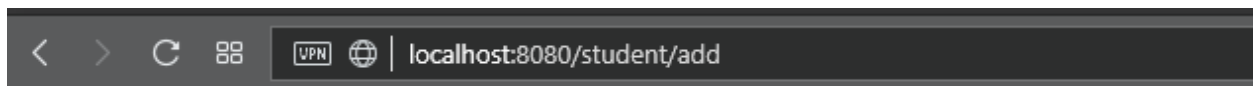
Pertanyaan 1:



Data berhasil ditambahkan

Tidak terjadi error karena parameter npm, nama, dan gpa terisi semua.

Pertanyaan 2:



Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

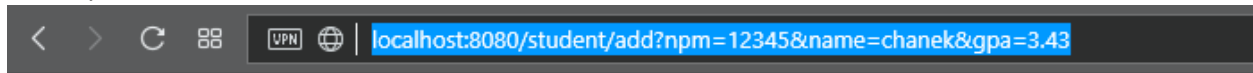
Sat Sep 23 20:27:30 ICT 2017

There was an unexpected error (type=Bad Request, status=400).

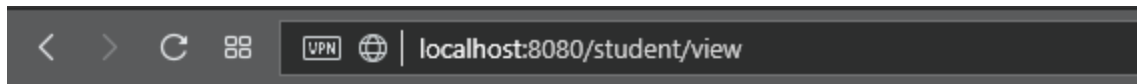
Required double parameter 'gpa' is not present

Terjadi error dikarenakan parameter gpa tidak ditemukan sehingga terjadi error ketika add student.

Pertanyaan 3:



Data berhasil ditambahkan



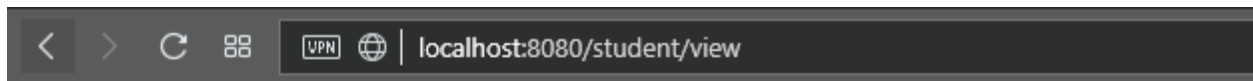
NPM = 12345

Name = chanek

GPA = 3.43

Muncul. Karena sudah memasukan data student kedalam list terlebih dahulu.

Pertanyaan 4:



Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

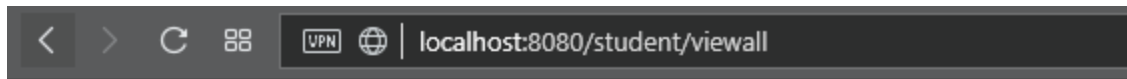
Sat Sep 23 20:52:52 ICT 2017

There was an unexpected error (type=Internal Server Error, status=500).

Exception evaluating SpringEL expression: "student.npm" (view:7)

Tidak muncul, karena spring bootnya di restart sehingga tidak ada isi didalam list nya.

Pertanyaan 5:



No. 1

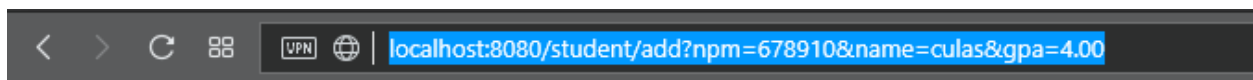
NPM = 12345

Name = chanek

GPA = 3.43

Muncul

Pertanyaan 6:



Data berhasil ditambahkan



No. 1

NPM = 12345

Name = chanek

GPA = 3.43

No. 2

NPM = 678910

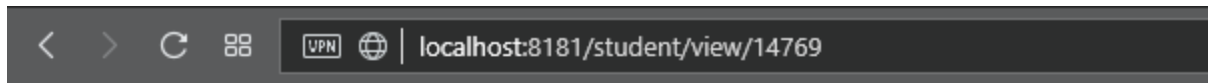
Name = culas

GPA = 4.0

Muncul. Karena setelah menambahkan data baru pada list dan dapat ditampilkan data tersebut oleh iterasi pada html sehingga dapat menampilkan seluruh isi list yang ada.

Latihan

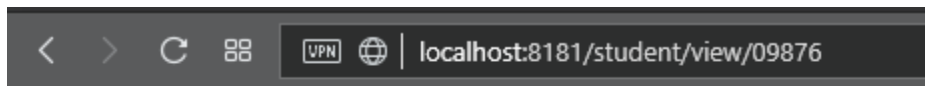
1.



NPM = 14769

Name = culas

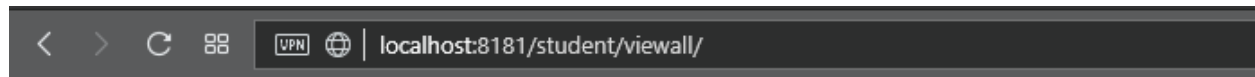
GPA = 4.0



NPM 09876 tidak ditemukan

Pada pembuatan fitur view dengan menggunakan path variable pertama kali saya membuat request mapping baru karena terdapat fitur view sebelumnya yang menggunakan request parameter supaya tidak ambigu dalam pemanggilannya. Kemudian saya memeriksa apakah parameter npm terisi, bila terisi kemudian di periksa kembali apakah npm tersebut ada didalam list. Bila ada maka akan mengambil data tersebut dari object student dan menampilkannya di browser. Bila tidak ditemukan maka akan menampilkan tampilan error yang menyatakan nomor npm tersebut tidak ditemukan.

2.



No. 1

NPM = 12345

Name = chanek

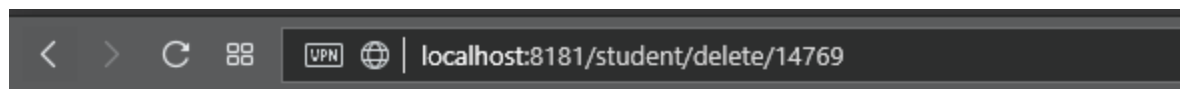
GPA = 3.43

No. 2

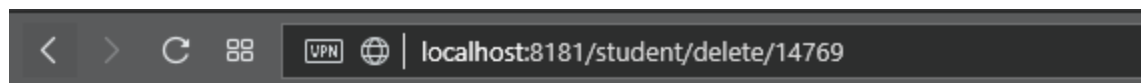
NPM = 14769

Name = culas

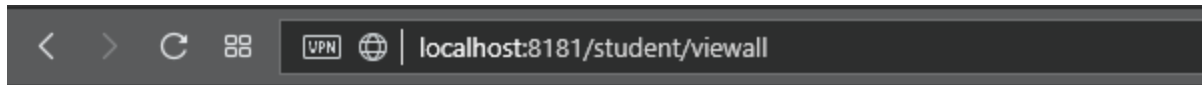
GPA = 4.0



data berhasil dihapus



NPM 14769 tidak ditemukan



No. 1

NPM = 12345

Name = chanek

GPA = 3.43

Pada pembuatan fitur delete ini sedikit mirip dengan fitur view, dimana pertama kali saya melakukan pemeriksaan terhadap parameter apakah terisi atau tidak. Bila terisi kemudian melakukan pencarian npm tersebut didalam list dengan menggunakan iterasi. Bila npm yang dicari cocok dengan hasil iterasi maka akan meremove data objek tersebut berdasarkan hasil iterasi. Bila tidak ditemukan akan menampilkan halaman error yang menyatakan bahwa npm tidak ditemukan.

METHOD selectStudent

```
@Override
public StudentModel selectStudent (String npm) {
    //implement

    for(int i = 0; i < studentList.size(); i++) {
        if(studentList.get(i).getNpm().equals(npm)) {
            return studentList.get(i);
        }
    }
    return null;
}
```

