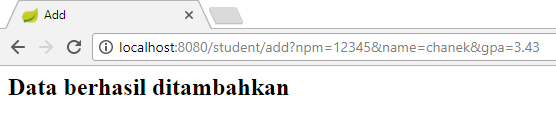
**a.**

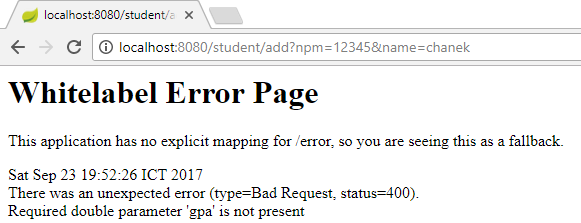
Tutorial 3 merupakan kelanjutan dari Tutorial 2 dengan tingkat kompleksitas yang lebih tinggi. Perbedaan utama Tutorial 3 dari Tutorial 2 adalah Tutorial 3 menggunakan service sehingga object yang dibuat dapat melakukan berbagai macam behaviour. Behaviour yang dimaksud adalah add, select, selectAll, dan delete.

**b.**

**Pertanyaan 1 :**

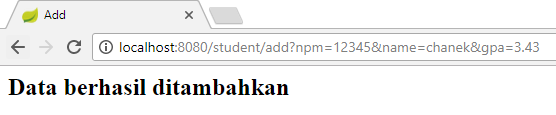


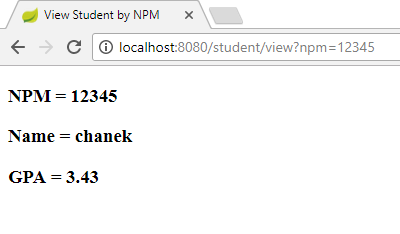
**Pertanyaan 2 :**



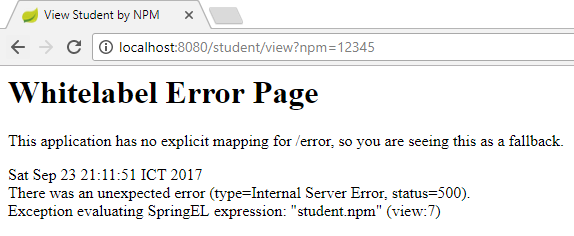
Error, karena tidak terdapat parameter gpa sedangkan parameter gpa harus ada.

**Pertanyaan 3 :**



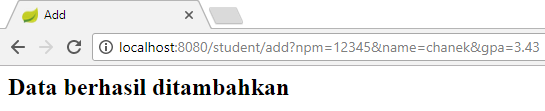


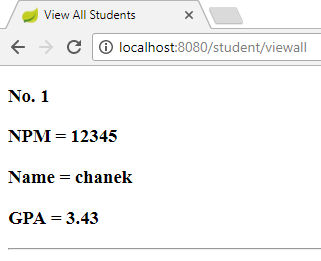
**Pertanyaan 4 :**

****

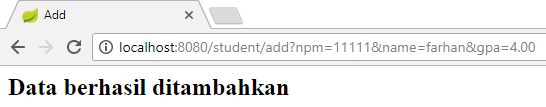
Tidak, karena data-data yang sudah diadd sebelumnya tidak tersimpan. Hal itu terjadi karena website tidak terhubing dengan database.

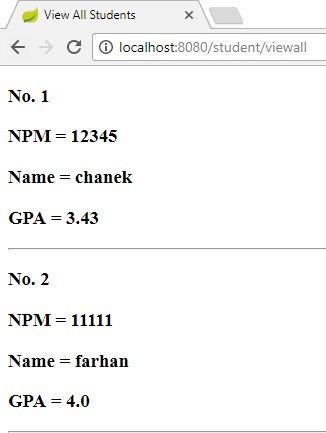
**Pertanyaan 5 :**

****

****

**Pertanyaan 6 :**

****

****

**C.**

@Override

**public** StudentModel selectStudent(String npm) {

StudentModel hasil = **null**;

**for**(**int** i=0;i<*studentList*.size();i++){

StudentModel siswa = *studentList*.get(i);

**if**(siswa.getNpm().equals(npm)) {

hasil = siswa;

}

}

**return** hasil;

}

**D.**

Fitur delete dilakukan dengan membuat fungsi deletePath pada StudentController. Fungsi ini serupa dengan fitur viewPath dimana dilakukan dengan masukan npm. Jika npm tidak diisi maka akan dilempar ke page deletenull. Apabila npm yang dimasukkan tidak ditemukan pada list maka akan dilempar ke page deletenotfound. Dan apabila ditemukan maka akan digunakan service selectstudent untuk mengambil object student dengan npm yang dimaksud. Kemudian dipanggil service deletestudent yang akan menghapus student dengan npm yang dimaksud dari ArrayList dan akan dilempar ke page delete.