

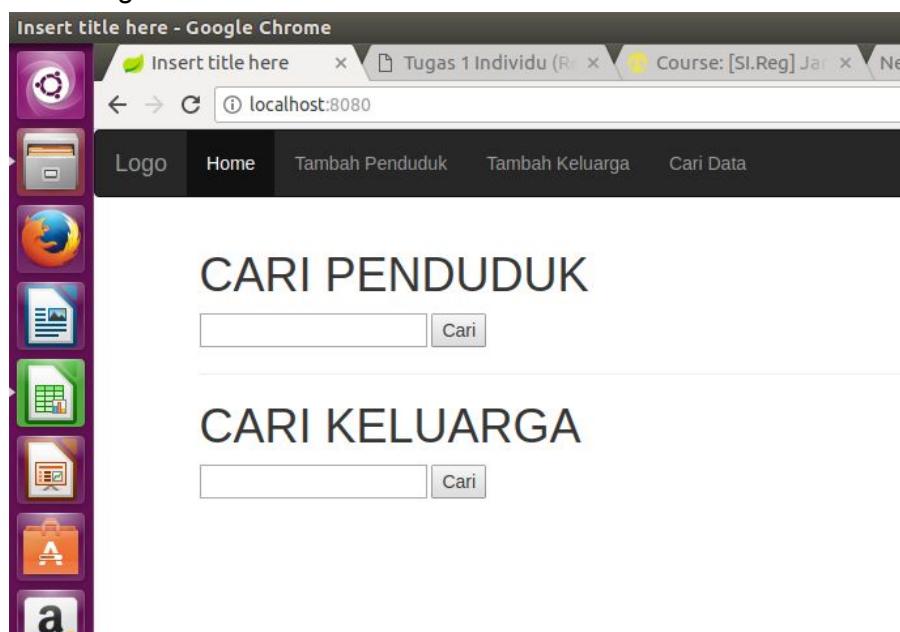
Enrico Jano Fiandi
1506689490
WRITE UP TUGAS 1 ADPAP

IMPORTANT

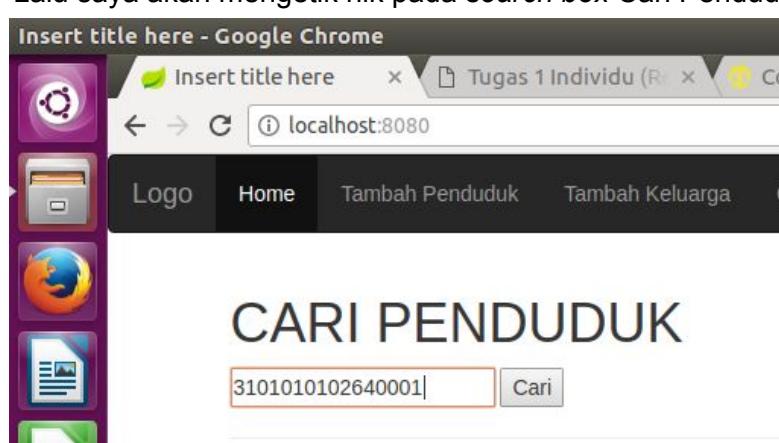
1. Sebelum melakukan testing terhadap aplikasi yang saya buat, diperlukan penambahan **AUTO INCREMENT** Pada field id table **penduduk** dan field id table **keluarga** dan meng-set auto increment tersebut dengan value id + 1 pada record terakhir yang diinput kedalam masing-masing table.
2. **Proses Validasi Form** dan **penambahan Error Page** untuk **nilai bonus** akan dijelaskan pada setiap fitur yang saya rasa membutuhkan hal-hal tersebut.

Fitur 1 - Tampilkan Data Penduduk Berdasarkan NIK

Proses menampilkan data penduduk berdasarkan nik dimulai dengan membuka halaman utama lagi website



Lalu saya akan mengetik nik pada *search box* Cari Penduduk dan mengklik *button Cari*



Dan jika NIK ditemukan maka akan menampilkan halaman Lihat Data Penduduk berdasarkan NIK tersebut.

Insert title here - Google Chrome
Tugas 1 Individu (R... x) tutorial-06-thymeleaf x Tugas ADPAP N R
Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Lihat Data Penduduk - 3101010102640001
NIK : 3101010102640001
Nama : Aris Edward Gunawan S.Sos
Tempat/Tanggal Lahir : Jakarta, 1964-02-01
Alamat : 3101010102640001
RT/RW : 057/052
Kelurahan/Desa : PULAU TIDUNG
Kecamatan : KEPULAUAN SERIBU SELATAN
Kota : KABUPATEN KEPULAUAN SERIBU
Golongan Darah : O+
Agama : Islam
Status Perkawinan : Kawin
Pekerjaan : PEMBANTU RUMAH TANGGA
Kewarganegaraan : WNI
Status Kematian : hidup

Nonaktifkan Edit

Validasi Jika Parameter NIK tidak terdapat didalam url

NOT FOUND - Google Chrome
NOT FOUND x Tugas 1 Individu (R... x) Course: [SI.Reg] Jar x New Tab
Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Halaman yang anda cari tidak ditemukan

Validasi jika Parameter NIK ada, namun tidak ada penduduk yang memiliki NIK tersebut di dalam database (Contoh, nik = 0)

NOT FOUND - Google Chrome
NOT FOUND x Tugas 1 Individu (R... x) Course: [SI.Reg] Jar x New Tab
Files Home Tambah Penduduk Tambah Keluarga Cari Data

Halaman yang anda cari tidak ditemukan

Penjelasan Kode

```

    @RequestMapping("/penduduk")
    public String penduduk(@RequestParam(value = "nik", required = false) String nik, Model model){
        if(nik == null){
            return "not-found";
        }
        Penduduk penduduk = pendudukDAO.selectPenduduk(nik);
        if(penduduk == null){
            return "not-found";
        }

        Keluarga keluarga = keluargaDAO.selectKeluarga(penduduk.getId_keluarga());
        Kelurahan kelurahan = lokasiDAO.selectKelurahan(keluarga.getId_kelurahan());
        Kecamatan kecamatan = lokasiDAO.selectKecamatan(kelurahan.getId_kecamatan());
        Kota kota = lokasiDAO.selectKota(kecamatan.getId_kota());

        model.addAttribute("penduduk", penduduk);
        model.addAttribute("keluarga", keluarga);
        model.addAttribute("kelurahan", kelurahan);
        model.addAttribute("kecamatan", kecamatan);
        model.addAttribute("kota", kota);
        model.addAttribute("isChanged", false);
        return "penduduk";
    }
}

```

Method tersebut akan dieksekusi ketika *end point* “/penduduk” dipanggil. Jika parameter NIK bernilai null yang dimana artinya tidak terdapat pada url maka akan mengembalikan halaman *not-found*. Jika parameter NIK ada namun NIK tersebut tidak terdapat didalam database maka akan mengembalikan halaman *not-found* juga. Berikut adalah query untuk mengambil objek penduduk di dalam database dengan menggunakan method selectPenduduk()

```

@Select("Select * from penduduk where nik = #{nik}")
Penduduk selectPenduduk(@Param("nik") String nik);

```

Jika parameter valid dan terdapat dalam database maka akan dijalankan 4 query yaitu query *select keluarga*, *select kelurahan*, *select kecamatan* dan *select kota*.

```

@Select("Select * from keluarga where id = #{id_keluarga}")
Keluarga selectKeluarga(@Param("id_keluarga") int id_keluarga);

@Select("Select * from kelurahan where id = #{id_kelurahan}")
Kelurahan selectKelurahan(@Param("id_kelurahan") int id_kelurahan);

@Select("Select * from kecamatan where id = #{id_kecamatan}")
Kecamatan selectKecamatan(@Param("id_kecamatan") int id_kecamatan);

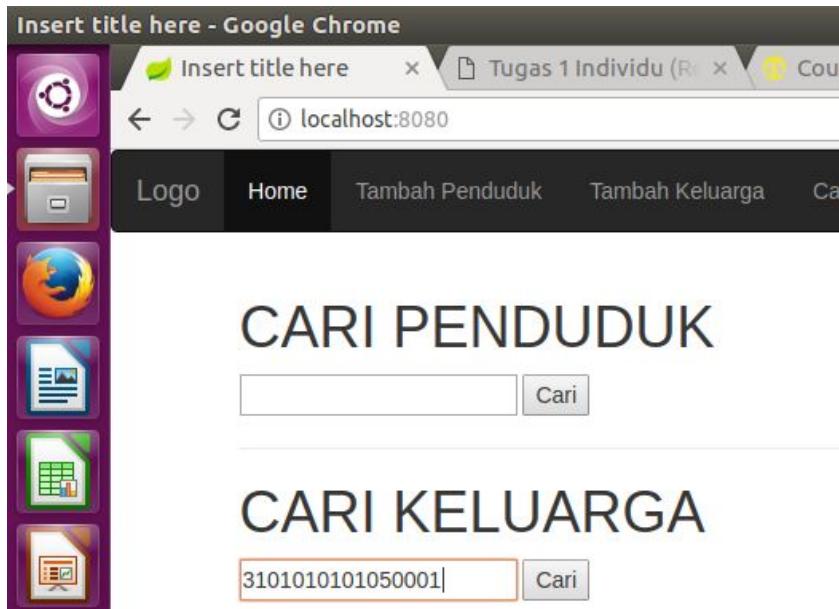
--- -----
@Select("Select * from kota where id = #{id_kota}")
Kota selectKota(@Param("id_kota") int id_kota);

```

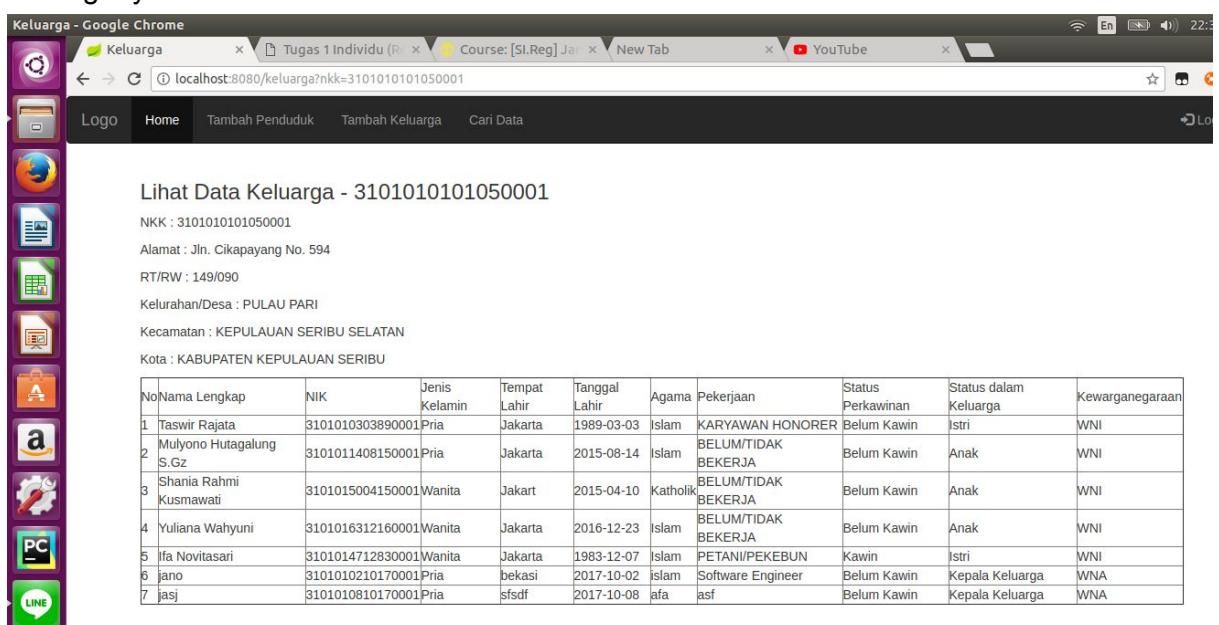
Saya akan mendapatkan detail keluarga, dan domisili dari penduduk dengan menjalankan query tersebut. Jika query sudah dieksekusi maka object penduduk, keluarga, kelurahan, kecamatan, dan kota akan dikirimkan ke view penduduk.html. *isChanged* nanti akan dijelaskan pada fitur mengubah status kematian penduduk.

Fitur 2 - Tampilkan Data Keluarga Beserta Daftar Anggotanya Berdasarkan Nomor KK

Proses menampilkan Data Keluarga Beserta Daftar Anggotanya berdasarkan Nomor KK dimulai dengan membuka halaman utama dan memasukan Nomor KK pada search box Cari Keluarga dan mengklik button Cari



Dan jika Nomor KK ditemukan maka akan menampilkan detail keluarga beserta anggota keluarganya

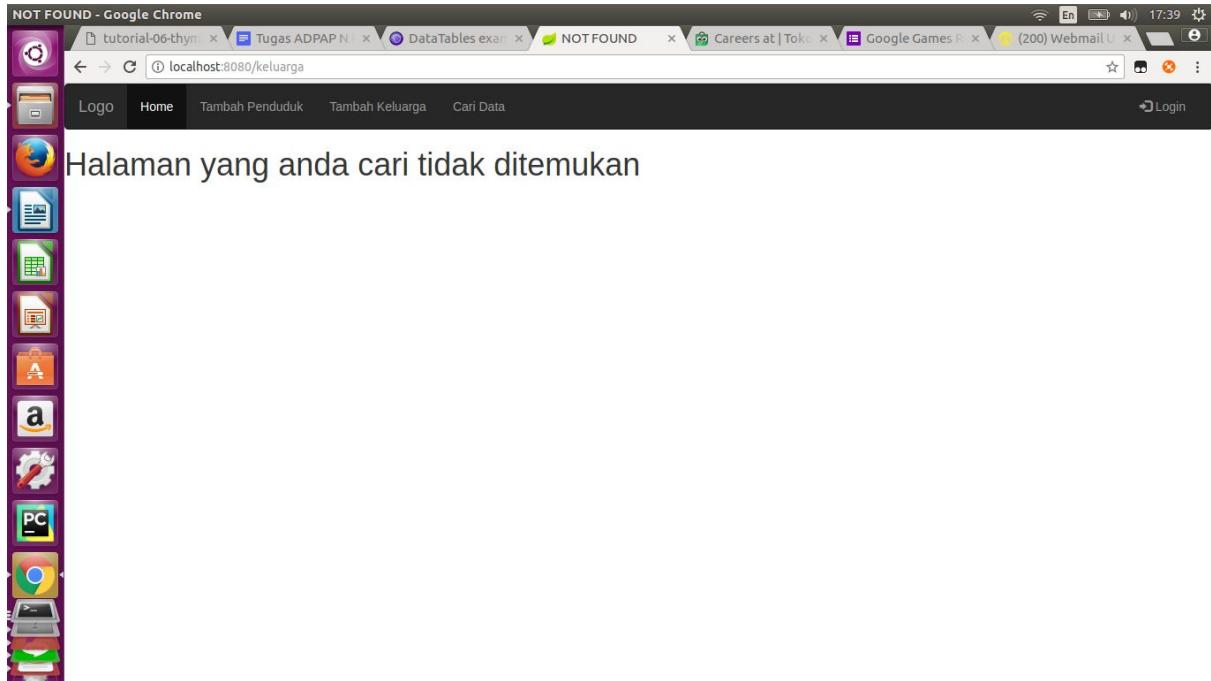


Lihat Data Keluarga - 3101010101050001

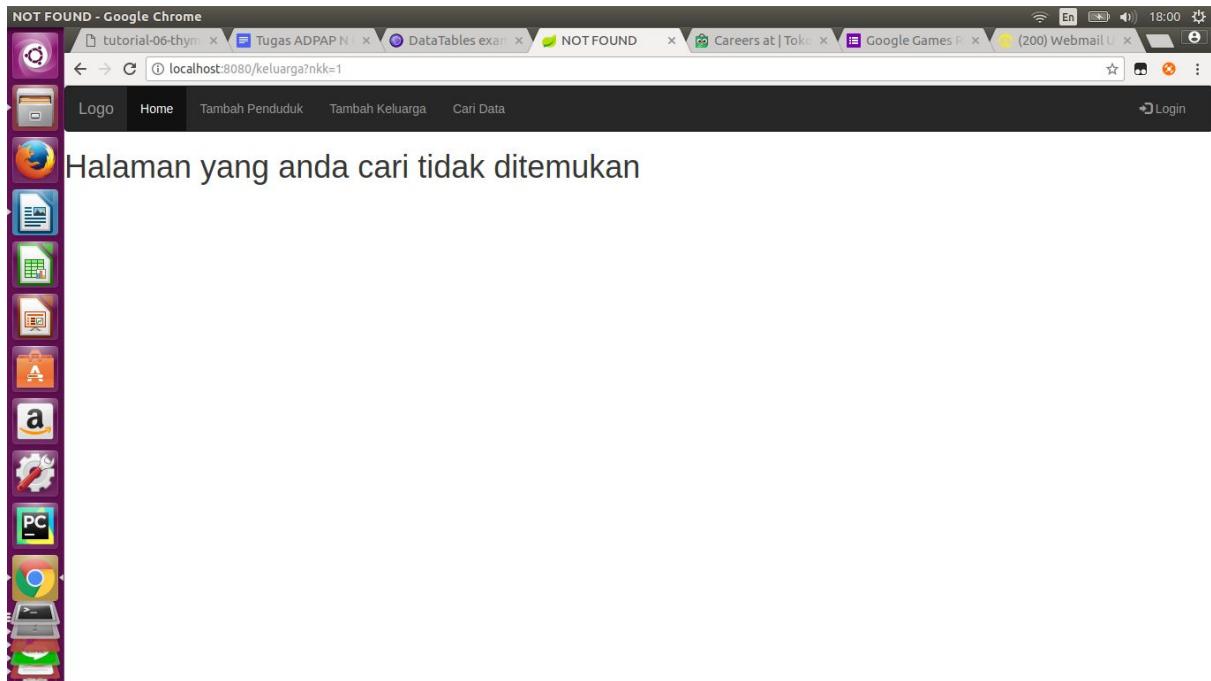
NKK : 3101010101050001
Alamat : Jln. Cikapayang No. 594
RT/RW : 149/090
Kelurahan/Desa : PULAU PARI
Kecamatan : KEPULAUAN SERIBU SELATAN
Kota : KABUPATEN KEPULAUAN SERIBU

No	Nama Lengkap	NIK	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Agama	Pekerjaan	Status Perkawinan	Status dalam Keluarga	Kewarganegaraan
1.	Taswir Rajata	3101010303890001	Pria	Jakarta	1989-03-03	Islam	KARYAWAN HONORER	Belum Kawin	Istri	WNI
2.	Mulyono Hutagalung S.Gz	3101011408150001	Pria	Jakarta	2015-08-14	Islam	BELUM/TIDAK BEKERJA	Belum Kawin	Anak	WNI
3.	Shania Rahmi Kusmawati	3101015004150001	Wanita	Jakart	2015-04-10	Katholik	BELUM/TIDAK BEKERJA	Belum Kawin	Anak	WNI
4.	Yuliana Wahyuni	3101016312160001	Wanita	Jakarta	2016-12-23	Islam	BELUM/TIDAK BEKERJA	Belum Kawin	Anak	WNI
5.	Ifa Novitasari	3101014712830001	Wanita	Jakarta	1983-12-07	Islam	PETANI/PEKEBUN	Kawin	Istri	WNI
6.	jano	3101010210170001	Pria	bekasi	2017-10-02	islam	Software Engineer	Belum Kawin	Kepala Keluarga	WNA
7.	jasj	3101010810170001	Pria	sfsdf	2017-10-08	afa	asd	Belum Kawin	Kepala Keluarga	WNA

Validasi Jika parameter Nomor KK tidak terdapat pada url maka akan menampilkan halaman *not-found*



Validasi Jika parameter Nomor KK tedapat pada url namun tidak terdapat keluarga yang memiliki Nomor KK tersebut di dalam database (contoh nkk = 1)



Penjelasan Kode

```
@RequestMapping("/keluarga")
public String keluarga(@RequestParam(value = "nkk", required = false) String nkk, Model model){
    if(nkk == null){
        return "not-found";
    }

    Keluarga keluarga = keluargaDAO.selectKeluargaNkk(nkk);
    if(keluarga == null){
        return "not-found";
    }
    Kelurahan kelurahan = lokasiDAO.selectKelurahan(keluarga.getId_kelurahan());
    Kecamatan kecamatan = lokasiDAO.selectKecamatan(kelurahan.getId_kecamatan());
    Kota kota = lokasiDAO.selectKota(kecamatan.getId_kota());

    model.addAttribute("keluarga", keluarga);
    model.addAttribute("kelurahan", kelurahan);
    model.addAttribute("kecamatan", kecamatan);
    model.addAttribute("kota", kota);

    return "keluarga";
}
```

Method tersebut akan dieksekusi ketika endpoint "/keluarga" dipanggil. Jika nkk bernilai null berarti nkk tidak terdapat di dalam parameter maka akan menampilkan halaman not found. Jika ada maka objek keluarga dengan nkk tersebut akan di-select dari database dengan menggunakan method selectKeluargaNkk(), berikut adalah query untuk mengambil objek keluarga di dalam database

```
@Select("Select * from keluarga where id = #{id_keluarga}")
Keluarga selectKeluarga(@Param("id_keluarga") int id_keluarga);
```

Jika objek keluarga bernilai null maka keluarga dengan nkk tersebut tidak ditemukan di dalam database sehingga halaman not found akan ditampilkan. Jika keluarga dengan nkk tersebut ada, maka akan diambil detail kelurahan, kecamatan, dan kota seperti yang sudah dijelaskan di **penjelasan kode fitur 1** untuk mengambil detail domisili dari keluarga tersebut. Lalu saya akan mengirimkan objek keluarga, kelurahan, kecamatna, dan kota ke view keluarga.html menggunakan class Model.

Fitur 3 - Menambahkan Penduduk Baru Sebagai Anggota Keluarga

Untuk menambahkan penduduk baru sebagai anggota keluarga, langkah pertama yang perlu dilakukan adalah mengakses *end point* “/penduduk/tambah” yang nantinya akan menampilkan form untuk menambahkan penduduk baru seperti screenshot dibawah ini

Insert title here - Google Chrome

Course: [SI.Reg] Ars ... tutorial-06-thymele ... Tugas ADPAP N Ric ... DataTable

localhost:8080/penduduk/tambah

Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Nama

Tempat Lahir

Tanggal Lahir mm/dd/yyyy

Jenis Kelamin

Golongan Darah

Agama

Status Perkawinan

Pekerjaan

Kewarganegaraan

Status Kematian

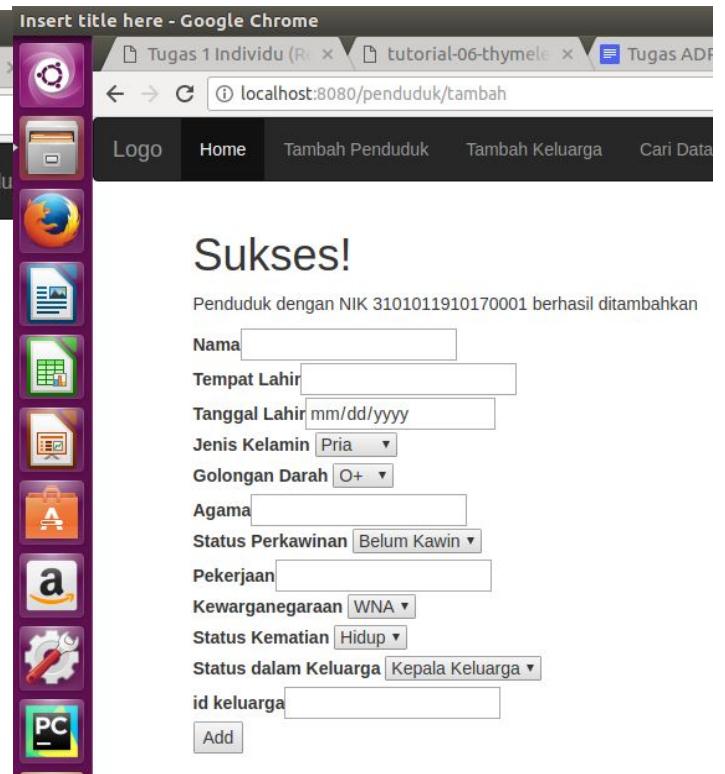
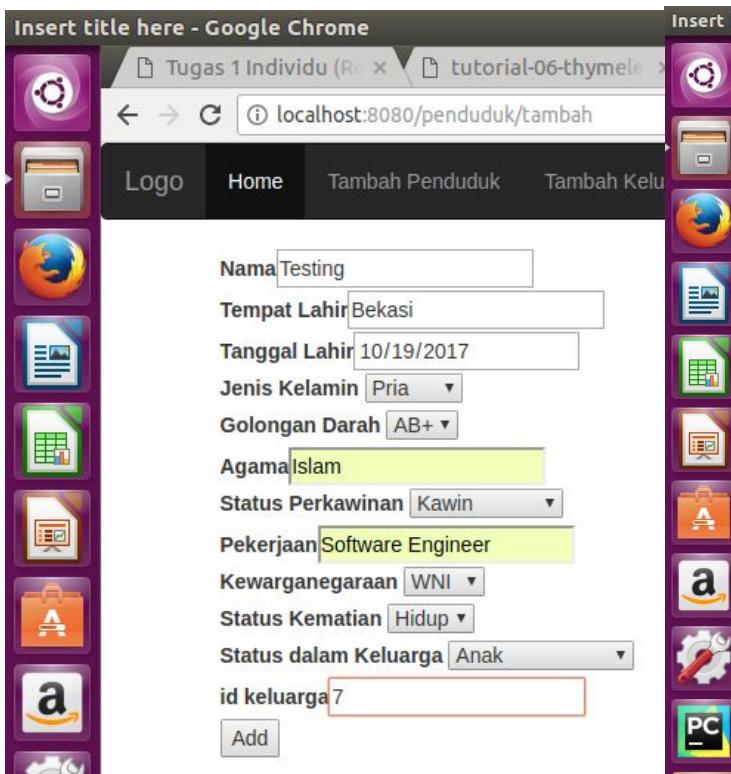
Status dalam Keluarga

id keluarga

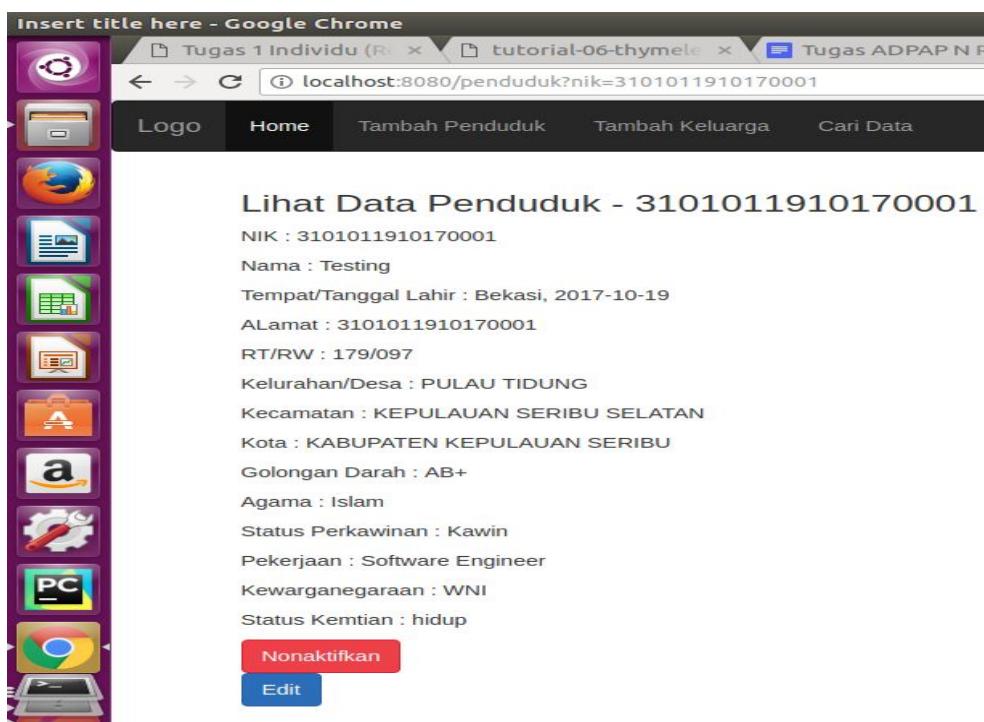
Add

Lalu proses input penduduk berdasarkan beberapa kasus dapat dilihat sebagai berikut

1. Kasus penduduk pria



Penduduk dengan NIK **3101011910170001** berhasil diinput, NIK tersebut telah memenuhi format yang diminta, dapat dilihat nomor urutnya adalah 0001 karena tidak ada penduduk pria lain yang memiliki kombinasi domisili dan tanggal lahir yang sama dengan penduduk tersebut.



2. Kasus penduduk wanita

Insert title here - Google Chrome

Tugas 1 Individu (R) tutorial-06-thymeleaf Tugas AD

localhost:8080/penduduk/tambah

Logo Home Tambah Penduduk Tambah Keluarga

Nama: Testing Dua
Tempat Lahir: Jakarta
Tanggal Lahir: 10/20/2017
Jenis Kelamin: Wanita
Golongan Darah: B+
Agama: Kristen
Status Perkawinan: Belum Kawin
Pekerjaan: System Analyst
Kewarganegaraan: WNA
Status Kematian: Hidup
Status dalam Keluarga: Istri
id keluarga: 7
Add



Insert title here - Google Chrome

Tugas 1 Individu (R) tutorial-06-thymeleaf Tugas AD

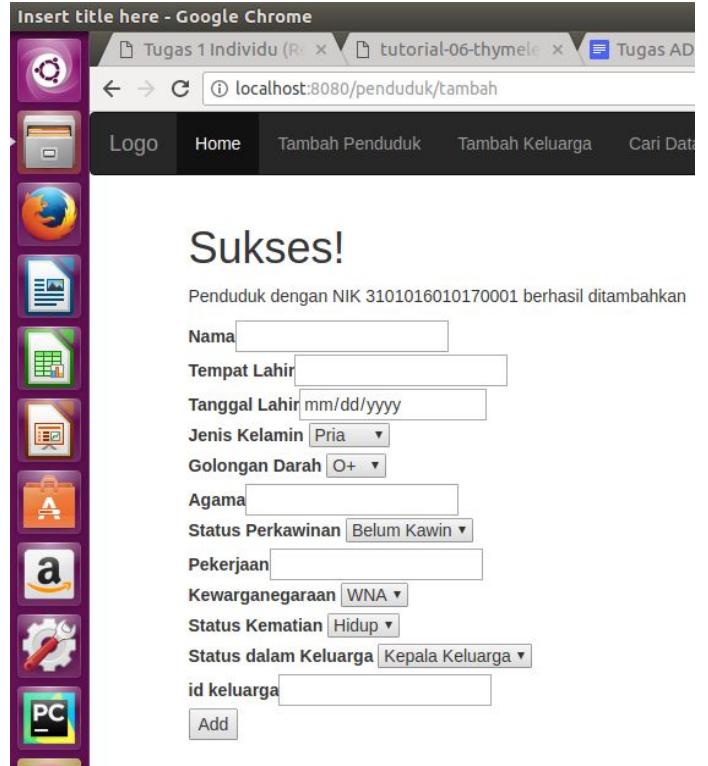
localhost:8080/penduduk/tambah

Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Sukses!

Penduduk dengan NIK 3101016010170001 berhasil ditambahkan

Nama:
Tempat Lahir:
Tanggal Lahir: mm/dd/yyyy
Jenis Kelamin: Pria
Golongan Darah: O+
Agama:
Status Perkawinan: Belum Kawin
Pekerjaan:
Kewarganegaraan: WNA
Status Kematian: Hidup
Status dalam Keluarga: Kepala Keluarga
id keluarga:
Add



Penduduk dengan NIK **3101016010170001** berhasil diinput ke dalam database. Data dilihat tanggal lahir yang awalnya 20 ditambah 40 menjadi 60 karena penduduk tersebut mememiliki jenis kelamin wanita, dan nomor urutnya 00001 karena tidak ada penduduk wanita lain yang memiliki kombinasi domisili dan tanggal lahir yang sama dengan penduduk tersebut

Insert title here - Google Chrome

Tugas 1 Individu (R) tutorial-06-thymeleaf Tugas ADPAP N RI

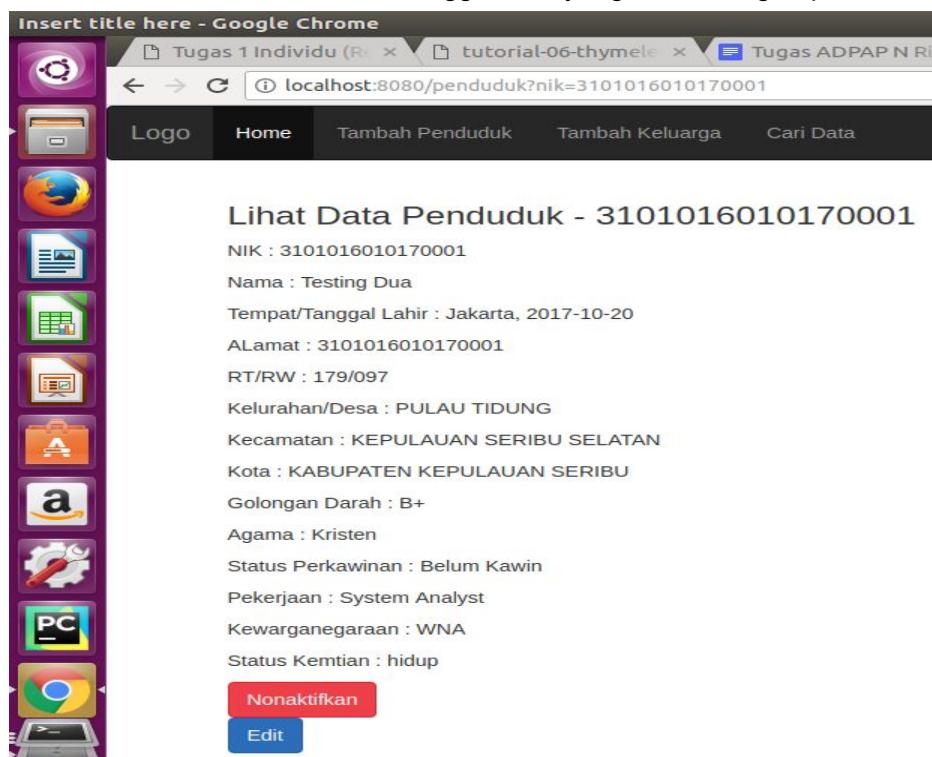
localhost:8080/penduduk?nik=3101016010170001

Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Lihat Data Penduduk - 3101016010170001

NIK : 3101016010170001
Nama : Testing Dua
Tempat/Tanggal Lahir : Jakarta, 2017-10-20
Alamat : 3101016010170001
RT/RW : 179/097
Kelurahan/Desa : PULAU TIDUNG
Kecamatan : KEPULAUAN SERIBU SELATAN
Kota : KABUPATEN KEPULAUAN SERIBU
Golongan Darah : B+
Agama : Kristen
Status Perkawinan : Belum Kawin
Pekerjaan : System Analyst
Kewarganegaraan : WNA
Status Kematian : hidup

Nonaktifkan Edit



3. Kasus sudah ada penduduk yang memiliki kombinasi domisili dan tanggal lahir yang sama

Saya akan menggunakan domisili dan tanggal lahir yang sama dengan nik penduduk pada input penduduk kasus pertama (**3101011910170001**) yaitu dengan jenis kelamin pria, id_keluarga 7 dan tanggal lahir 19-10-2017

The screenshot displays two instances of a web application. The left instance shows a form for adding a new resident. The right instance shows a success message after the addition.

Left Window (Form):

- Nama: Testing duplikat
- Tempat Lahir: London
- Tanggal Lahir: 10/19/2017
- Jenis Kelamin: Pria
- Golongan Darah: O+
- Agama: Islam
- Status Perkawinan: Kawin
- Pekerjaan: CTO of Google
- Kewarganegaraan: WNA
- Status Kematian: Hidup
- Status dalam Keluarga: Kepala Keluarga
- id keluarga: 7

Right Window (Success Message):

Sukses!

Penduduk dengan NIK 3101011910170002 berhasil ditambahkan

- Nama: [empty]
- Tempat Lahir: [empty]
- Tanggal Lahir: mm/dd/yyyy
- Jenis Kelamin: Pria
- Golongan Darah: O+
- Agama: [empty]
- Status Perkawinan: Belum Kawin
- Pekerjaan: [empty]
- Kewarganegaraan: WNA
- Status Kematian: Hidup
- Status dalam Keluarga: Kepala Keluarga
- id keluarga: [empty]

Penduduk dengan NIK **3101011910170002** berhasil diinput ke database, dapat dilihat nomor urutnya adalah 0002 karena sebelumnya sudah ada penduduk yang memiliki kombinasi domisili dan tanggal lahir serupa yang memiliki nomor urut 0001 sehingga penduduk yang baru diinput nomor urutnya perlu diincrement.

The screenshot shows a detailed view of a resident's information. The resident has NIK 3101011910170002.

Lihat Data Penduduk - 3101011910170002

NIK : 3101011910170002
Nama : Testing duplikat
Tempat/Tanggal Lahir : London, 2017-10-19
Alamat : 3101011910170002
RT/RW : 179/097
Kelurahan/Desa : PULAU TIDUNG
Kecamatan : KEPULAUAN KEPULAUAN SERIBU SELATAN
Kota : KABUPATEN KEPULAUAN SERIBU
Golongan Darah : O+
Agama : Islam
Status Perkawinan : Kawin
Pekerjaan : CTO of Google
Kewarganegaraan : WNA
Status Kematian : hidup

Buttons:

- Nonaktifkan
- Edit

Validasi jika input tidak lengkap (kasus ini nama saya kosongkan)

Insert title here - Google Chrome

Tugas 1 Individu (R... x) tutorial-06-thymeleaf x

localhost:8080/penduduk/tambah

Logo Home Tambah Penduduk Tambah Keluarga

Nama

Tempat Lahir

Tanggal Lahir

Jenis Kelamin

Golongan Darah

Agama

Status Perkawinan

Pekerjaan

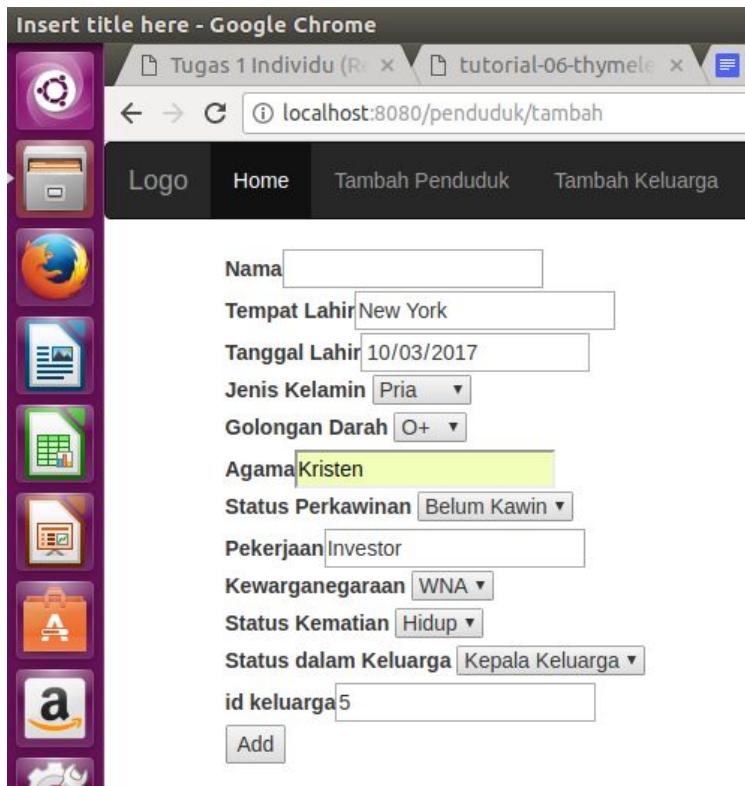
Kewarganegaraan

Status Kematian

Status dalam Keluarga

id keluarga

Add



Akan menampilkan halaman pesan error

Insert title here - Google Chrome

tutorial-06-thymeleaf x Tugas ADPAP N R... x

localhost:8080/penduduk/tambah

Logo Home Tambah Penduduk Tambah Keluarga

ERROR!

Lengkapi seluruh data terlebih dahulu dengan valid

Nama

Tempat Lahir

Tanggal Lahir

Jenis Kelamin

Golongan Darah

Agama

Status Perkawinan

Pekerjaan

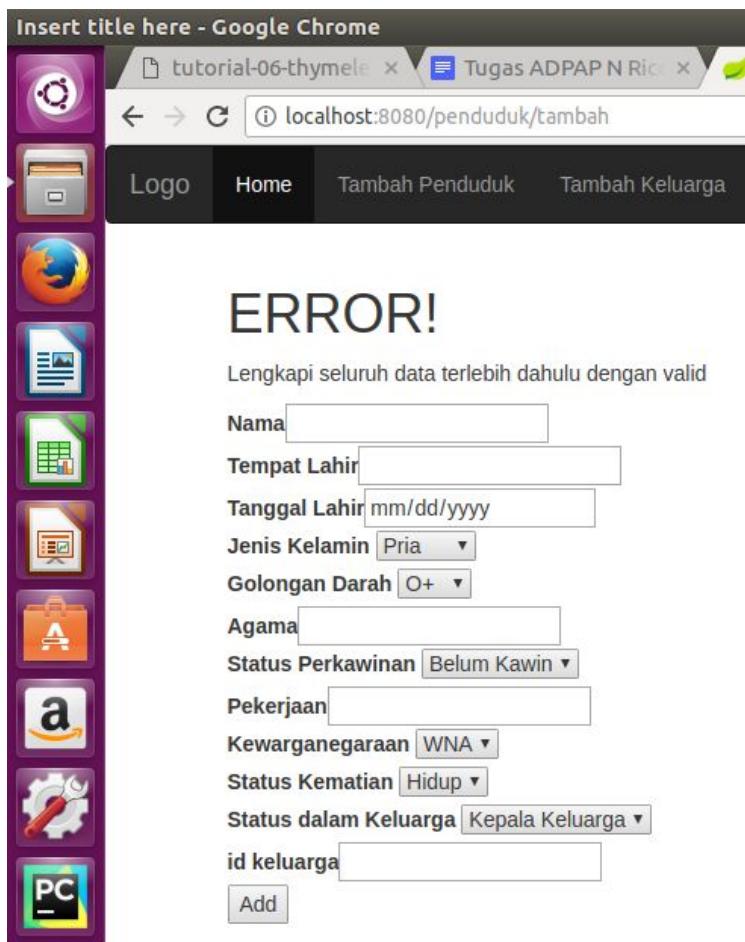
Kewarganegaraan

Status Kematian

Status dalam Keluarga

id keluarga

Add



Validasi jika id keluarga tidak valid (kasus ini saya masukan id keluarga -1)

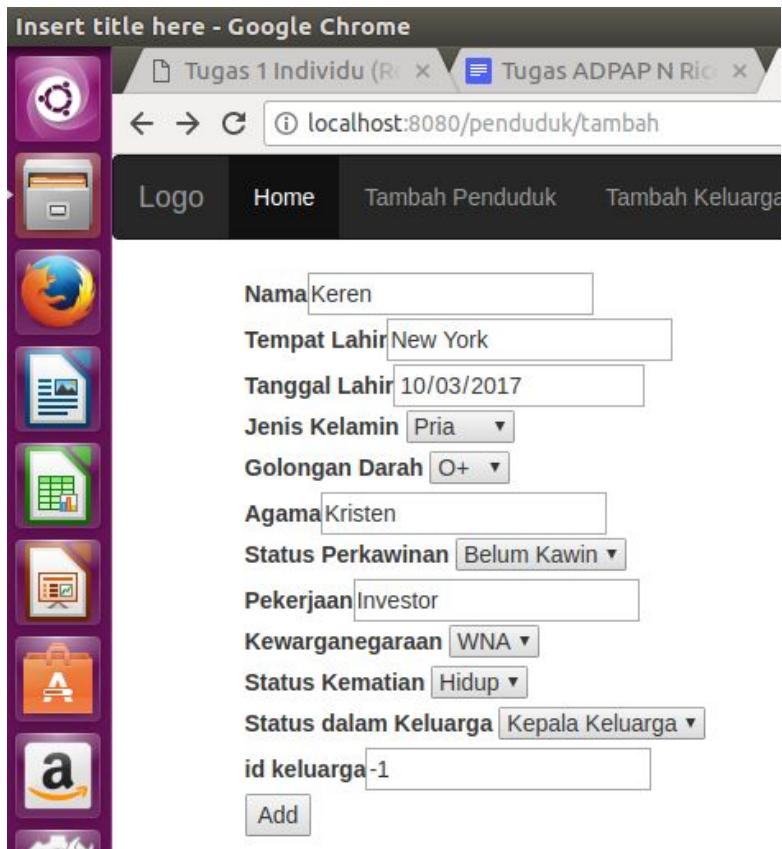
Insert title here - Google Chrome

Tugas 1 Individu (R... Tugas ADPAP N Ric...

localhost:8080/penduduk/tambah

Logo Home Tambah Penduduk Tambah Keluarga

Nama Keren
Tempat Lahir New York
Tanggal Lahir 10/03/2017
Jenis Kelamin Pria ▾
Golongan Darah O+ ▾
Agama Kristen
Status Perkawinan Belum Kawin ▾
Pekerjaan Investor
Kewarganegaraan WNA ▾
Status Kematian Hidup ▾
Status dalam Keluarga Kepala Keluarga ▾
id keluarga -1
Add



Akan menampilkan pesan **error**

Insert title here - Google Chrome

Tugas 1 Individu (R... Tugas ADPAP N Ric...

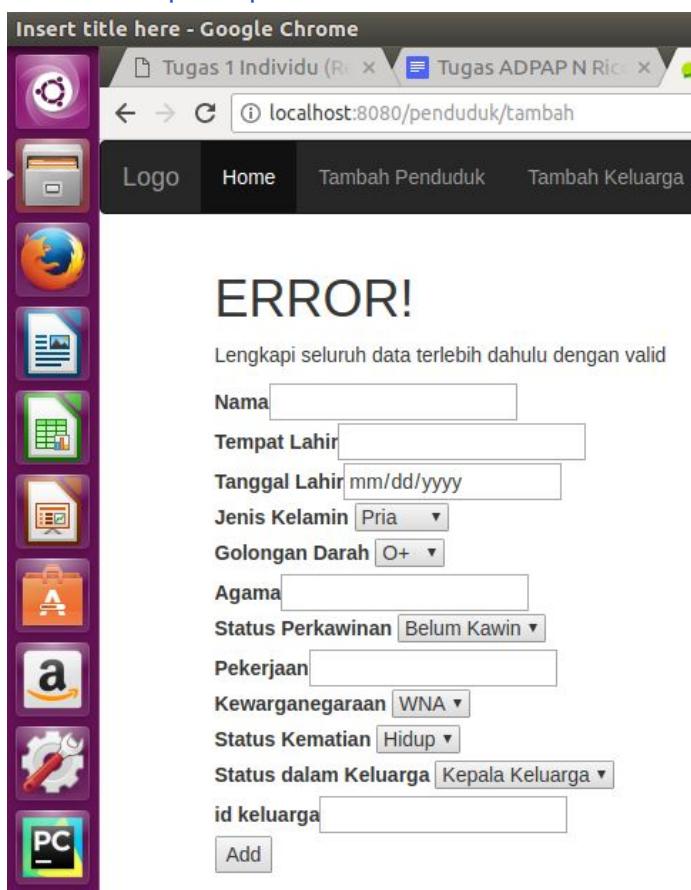
localhost:8080/penduduk/tambah

Logo Home Tambah Penduduk Tambah Keluarga

ERROR!

Lengkapi seluruh data terlebih dahulu dengan valid

Nama
Tempat Lahir
Tanggal Lahir mm/dd/yyyy
Jenis Kelamin Pria ▾
Golongan Darah O+ ▾
Agama
Status Perkawinan Belum Kawin ▾
Pekerjaan
Kewarganegaraan WNA ▾
Status Kematian Hidup ▾
Status dalam Keluarga Kepala Keluarga ▾
id keluarga
Add



Penjelasan Kode

```
④ @RequestMapping("/penduduk/tambah")
public String formAddPenduduk(Model model){
    model.addAttribute("penduduk", new Penduduk());
    model.addAttribute("isAdded", false);
    return "form-add-penduduk";
}
```

Method tersebut akan dieksekusi ketika *end point* “/penduduk/tambah” dipanggil di url. Method tersebut akan menampilkan view form-add-penduduk dan sebelum ditampilkan, akan dikirimkan object penduduk kosong dan boolean isAdded untuk menandakan apakah sedang terjadi proses penambahan atau belum. Pada saat form ditampilkan nilai isAdded false karena belum terjadi proses insert

```
⑤ @RequestMapping(value = "/penduduk/tambah", method = RequestMethod.POST)
public String addPenduduk(@Valid Penduduk penduduk, BindingResult bindingResult,
    Model model){

    if(bindingResult.hasErrors()){
        model.addAttribute("penduduk", new Penduduk());
        model.addAttribute("isAdded", false);
        model.addAttribute("isError", true);
        return "form-add-penduduk";
    }
    Keluarga keluarga = keluargaDAO.selectKeluarga(penduduk.getId_keluarga());
    if(keluarga == null){
        model.addAttribute("penduduk", new Penduduk());
        model.addAttribute("isAdded", false);
        model.addAttribute("isError", true);
        return "form-add-penduduk";
    }
    Kelurahan kelurahan = lokasiDAO.selectKelurahan(keluarga.getId_kelurahan());
    Kecamatan kecamatan = lokasiDAO.selectKecamatan(kelurahan.getId_kecamatan());
    String[] birth = penduduk.getTanggal_lahir().split("-");
    String year = birth[0].substring(2);
    String month = birth[1];
    int tempDay = Integer.parseInt(birth[2]);
    tempDay = penduduk.getJenis_kelamin() == 0 ? tempDay : tempDay + 40;
    String day = tempDay < 10 ? "0" + tempDay : "" + tempDay;

    String nik = kecamatan.getKode_kecamatan().substring(0, kecamatan.getKode_kecamatan().length() - 1);
    nik += day + month + year;

    List<Penduduk> similar = pendudukDAO.selectSimilarNik(nik);
    String nomor_urut = "0001";
    if(similar.size() > 0){
        nomor_urut = "" + (Integer.parseInt(similar.get(similar.size() - 1).getNik().substring(12)) + 1);
        for(int i = nomor_urut.length(); i < 4; i++){
            nomor_urut = "0" + nomor_urut;
        }
    }
    nik += nomor_urut;
    penduduk.setNik(nik);
    pendudukDAO.addPenduduk(penduduk);
    model.addAttribute("nik", nik);
    model.addAttribute("isAdded", true);
    model.addAttribute("penduduk", new Penduduk());
    return "form-add-penduduk";
}
```

Method tersebut akan dieksekusi ketika *end point* “/penduduk/tambah” diakses dengan request POST. Terjadi proses *form validation* sebelum masuk ke method ini yaitu dengan menggunakan anotasi @Valid yang diletakan di object parameter dan anotasi @NotNull dan @NotEmpty yang di letakan di properties modal.

```
package com.example.tugas1.model;

import javax.validation.constraints.NotNull;
@Data
@AllArgsConstructor
@NoArgsConstructor
public class Penduduk {
    private int id;

    private String nik;
    @NotEmpty
    private String nama;
    @NotEmpty
    private String tempat_lahir;
    @NotEmpty
    private String tanggal_lahir;
    @NotNull
    private int jenis_kelamin;
    @NotNull
    private int is_wni;
    @NotNull
    private int id_keluarga;
    @NotEmpty
    private String agama;
    @NotEmpty
    private String status_perkawinan;
    @NotEmpty
    private String status_dalam_keluarga;
    @NotEmpty
    private String golongan_darah;
    @NotNull
    private int is_wafat;
    @NotEmpty
    private String pekerjaan;

    private Keluarga keluarga;
    private Kelurahan kelurahan;
    private Kecamatan kecamatan;
    private Kota kota;
```

Jika ada kondisi yang tidak memenuhi @NotNull atau @NotEmpty maka class BindingResult akan mengembalikan nilai true ketika method hasErrors() di panggil, lalu akan ditampilkan pesan error di form add penduduk. Jika tidak maka id keluarga dari penduduk akan dicari di database, jika tidak ada maka object keluarga akan bernilai null dan akan mengembalikan pesan error. Jika ada, maka kelurahan dan kecamatan dari penduduk tersebut akan dicari dan disimpan dalam objek. Setelah itu akan ada proses pembuatan NIK dengan menggunakan tanggal lahir dan domisili, jika sudah ada penduduk dengan kombinasi tersebut maka nomor urut terakhir akan di tambah 1 (misal 0001 menjadi 0002). Jika sudah maka penduduk akan diinsert kedalam database dan akan menampilkan pesan sukses pada halaman form add penduduk.

```
@Insert("Insert into penduduk(nik, nama, tempat_lahir, tanggal_lahir, jenis_kelamin, is_wni, "
        + "id_keluarga, agama, pekerjaan, status_perkawinan, status_dalam_keluarga, golongan_darah, is_wafat)"
        + "VALUES(#{nik}, #{nama}, #{tempat_lahir}, #{tanggal_lahir}, #{jenis_kelamin}, #{is_wni}, "
        + " #{id_keluarga}, #{agama}, #{pekerjaan}, #{status_perkawinan}, #{status_dalam_keluarga}, #{golongan_darah}, #{is_wafat})")
void addPenduduk(Penduduk penduduk);
```

isAdded dan isError adalah suatu boolean yang menandakan apakah perlu menampilkan pesan sukses atau pesan error di halaman form add penduduk

Fitur 4 - Menambahkan Keluarga Baru

Langkah pertama untuk menginput keluarga adalah dengan membuka halaman Tambah Keluarga

The screenshot shows a web browser window titled "Add Keluarga - Google Chrome". The address bar displays "localhost:8080/keluarga/tambah". The main content area contains several input fields and dropdown menus. At the top are fields for "Alamat", "RT", and "RW". Below these are three dropdown menus labeled "Pilih Kota", "Pilih Kecamatan", and "Pilih Kelurahan". At the bottom of the form is a single "Add" button.

Lalu proses input penduduk berdasarkan beberapa kasus dapat dilihat sebagai berikut

1. Kasus keluarga yang pertama kali diinput pada tanggal 22 Oktober 2017

The screenshot displays two side-by-side browser windows. The left window shows the "Add Keluarga" form with various input fields filled out. The right window shows a confirmation message "Sukses!" above a summary of the added family's details, including the NKK number "3101012210170001". Both windows have a similar interface with a sidebar containing icons for various applications.

Keluarga dengan NKK **3101012210170001** berhasil ditambahkan ke database, dapat dilihat nkk tersebut telah mengikuti format yang benar, dan nomor urutnya adalah 0001 karena merupakan keluarga pertama yang diinput pada tanggal 22 oktober 2017

Keluarga - Google Chrome

localhost:8080/keluarga?nkk=3101012210170001

Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Lihat Data Keluarga - 3101012210170001

NKK : 3101012210170001

Alamat : Jalan Testing

RT/RW : 01/01

Kelurahan/Desa : PULAU TIDUNG

Kecamatan : KEPULAUAN SERIBU SELATAN

Kota : KABUPATEN KEPULAUAN SERIBU

No	Nama Lengkap	NIK	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Agama	Pekerjaan	Status Perkawinan	Status dalam Keluarga	Kewarganegaraan
----	--------------	-----	---------------	--------------	---------------	-------	-----------	-------------------	-----------------------	-----------------

2. Kasus keluarga yang diinput kedua pada tanggal 22 oktober 2017

Add Keluarga - Google Chrome

localhost:8080/keluarga/tambah

Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Alamat: Jalan Testing 2
RT: 01
RW: 02
KABUPATEN KEPULAUAN SERIBU
KEPULAUAN SERIBU SELATAN
PULAU TIDUNG

Add

Sukses!

Keluarga dengan NKK 3101012210170002 berhasil ditambahkan

Alamat: Jalan Testing 2
RT: 01
RW: 02
Pilih Kota
Pilih Kecamatan
Pilih Kelurahan

Keluarga dengan nkk **3101012210170002** berhasil diinput ke dalam database, dapat dilihat nomor urut dari keluarga tersebut ada 0002 karena keluarga tersebut merupakan keluarga keedua yang iinput pada tanggal 22 oktober 2017 dan memiliki domisili yang sama dengan keluarga pada kasus 1

Validasi data keluarga tidak lengkap diinput (kasus ini alamat bhaksaya kosongkan)

The screenshot shows a web page titled "Add Keluarga - Google Chrome". The URL in the address bar is "localhost:8080/keluarga/tambah". The page has a navigation menu with "Logo", "Home", "Tambah Penduduk", and "Tambah Keluarga". Below the menu, there are input fields for "Alamat" (Address), "RT" (Rumah Tangga) containing "01", and "RW" (Ward) containing "02". A dropdown menu for "KABUPATEN KEPULAUAN SERIBU" is open, showing "KEPULAUAN SERIBU SELATAN" and "PULAU TIDUNG" as options. An "Add" button is at the bottom.

Akan menampilkan pesan error

The screenshot shows a web page titled "Add Keluarga - Google Chrome". The URL in the address bar is "localhost:8080/keluarga/tambah". The page has a navigation menu with "Logo", "Home", "Tambah Penduduk", and "Tambah Keluarga". The main content area displays a large "ERROR!" message. Below it, a message says "Lengkapi seluruh data terlebih dahulu dengan valid". There are input fields for "Alamat", "RT" (containing "RT"), and "RW" (containing "RW"). A dropdown menu for "Pilih Kota" is open, showing "Pilih Kecamatan" and "Pilih Kelurahan" as options. An "Add" button is at the bottom.

Penjelasan Code

```
@RequestMapping("/keluarga/tambah")
public String formAddKeluarga(Model model){
    model.addAttribute("keluarga", new Keluarga());
    model.addAttribute("isUpdated", false);
    return "form-add-keluarga";
}
```

Method tersebut akan dijalankan ketika *end point* "/keluarga/tambah" dan akan menampilkan halaman form add keluarga

```

@RequestMapping(value = "/keluarga/tambah", method = RequestMethod.POST)
public String addKeluarga(@Valid Keluarga keluarga, BindingResult bindingResult, Model model){

    if(bindingResult.hasErrors()){
        model.addAttribute("keluarga", new Keluarga());
        model.addAttribute("isUpdated", false);
        model.addAttribute("isError", true);
        return "form-add-keluarga";
    }

    Kelurahan kelurahan = lokasiDAO.selectKelurahan(keluarga.getId_kelurahan());

    DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy/MM/dd");
    LocalDate localDate = LocalDate.now();
    String date = dtf.format(localDate);
    String[] dateDetail = date.split("/");

    String nkk = kelurahan.getKode_kelurahan().substring(0,6) + dateDetail[2] + dateDetail[1] + dateDetail[0].substring(2);
    List<Keluarga> similar = keluargaDAO.selectSimilarNkk(nkk);
    String nomor_urut = "0001";
    if(similar.size() > 0){
        nomor_urut = "" + (Integer.parseInt(similar.get(similar.size() - 1).getNomor_kk().substring(12)) + 1);
        for(int i = nomor_urut.length(); i < 4; i++){
            nomor_urut = "0" + nomor_urut;
        }
    }
    nkk += nomor_urut;
    keluarga.setNomor_kk(nkk);

    keluargaDAO.addKeluarga(keluarga);
    model.addAttribute("nkk", nkk);
    model.addAttribute("isUpdated", true);
    return "form-add-keluarga";
}

```

Method tersebut akan dijalankan ketika form add keluarga disubmit, jika saat proses validasi dengan anotasi @Valid terjadi error (proses validasi dengan anotasi @Valid sudah dijelaskan pada fitur 3) maka akan menampilkan pesan error pada halaman form add keluarga, jika tidak maka objek kelurahan dari keluarga tersebut akan disimpan. Setelah itu NKK dari keluarga tersebut akan dibuat berdasarkan tanggal pembuatan dan domisili keluarga tersebut. Ketika NKK sudah dibuat maka keluarga akan ditambahkan ke dalam database dan pesan sukses akan ditampilkan pada form add keluarga.

Fitur 5 - Mengubah Data Penduduk

Langkah pertama untuk mengubah suatu penduduk adalah mengakses halaman detail penduduk

Lihat Data Penduduk - 3101011910170002

NIK : 3101011910170002

Nama : Testing duplikat

Tempat/Tanggal Lahir : London, 2017-10-19

Alamat : 3101011910170002

RT/RW : 179/097

Kelurahan/Desa : PULAU TIDUNG

Kecamatan : KEPULAUAN SERIBU SELATAN

Kota : KABUPATEN KEPULAUAN SERIBU

Golongan Darah : O+

Agama : Islam

Status Perkawinan : Kawin

Pekerjaan : CTO of Google

Kewarganegaraan : WNA

Status Kemitian : hidup

Nonaktifkan

Edit

Klik edit

Nama: Testing duplikat

Tempat Lahir: London

Tanggal Lahir: 10/19/2017

Jenis Kelamin: Pria

Golongan Darah: O+

Agama: Islam

Status Perkawinan: Kawin

Pekerjaan: CTO of Google

Kewarganegaraan: WNA

Status dalam Keluarga: Kepala Keluarga

id keluarga: 7

Update

Lalu ganti detail dari keluarga (pada kasus ini saya mengganti nama dan id keluarga agar nik terupdate).

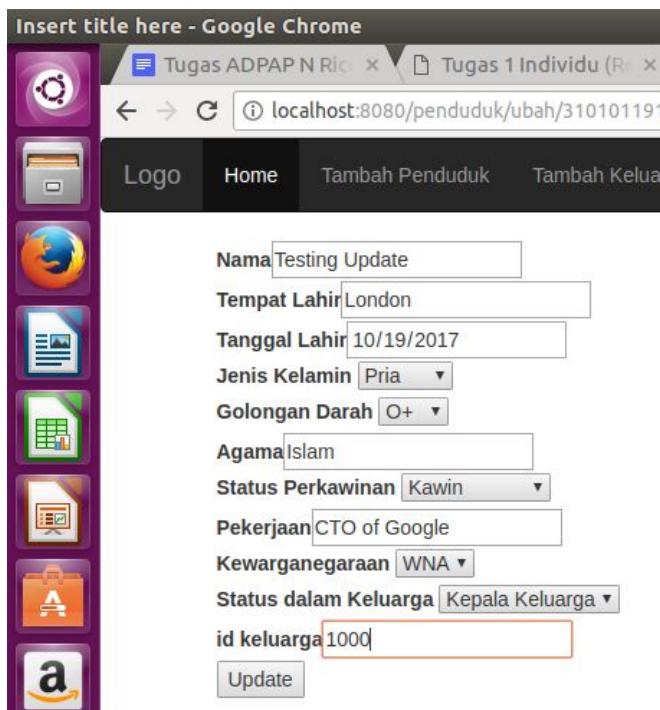
Insert title here - Google Chrome

Tugas ADPAP N Ric x Tugas 1 Individu (R x)

localhost:8080/penduduk/ubah/3101011910170002

Logo Home Tambah Penduduk Tambah Keluarga

Nama Testing Update
Tempat Lahir London
Tanggal Lahir 10/19/2017
Jenis Kelamin Pria ▾
Golongan Darah O+ ▾
Agama Islam
Status Perkawinan Kawin ▾
Pekerjaan CTO of Google
Kewarganegaraan WNA ▾
Status dalam Keluarga Kepala Keluarga ▾
id keluarga 1000
Update



Pesan sukses ditampilkan dan NIK yang lama tetap ditampilkan **3101011910170002**

Insert title here - Google Chrome

Tugas ADPAP N Ric x Tugas 1 Individu (R x) Inser

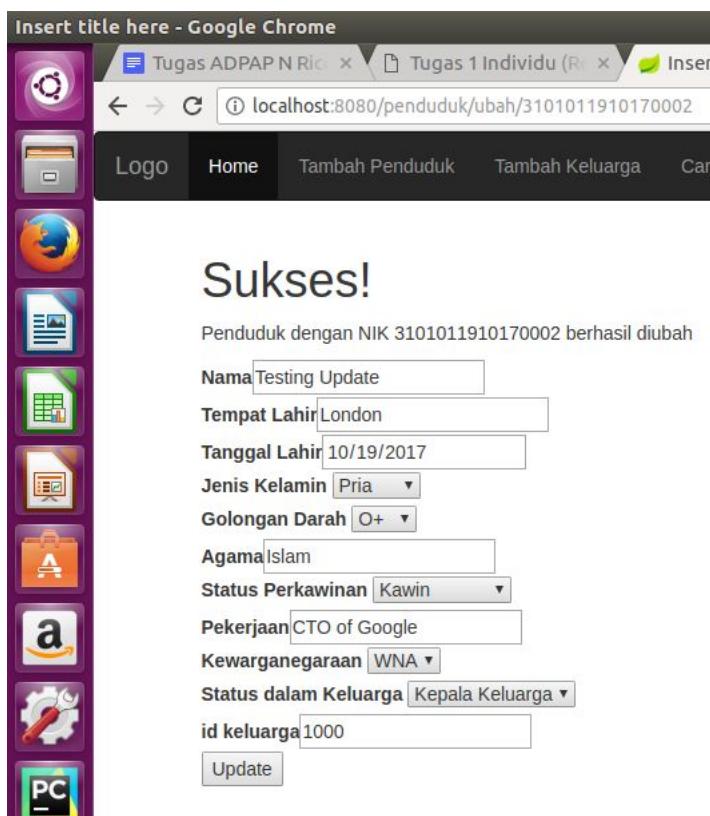
localhost:8080/penduduk/ubah/3101011910170002

Logo Home Tambah Penduduk Tambah Keluarga Car

Sukses!

Penduduk dengan NIK 3101011910170002 berhasil diubah

Nama Testing Update
Tempat Lahir London
Tanggal Lahir 10/19/2017
Jenis Kelamin Pria ▾
Golongan Darah O+ ▾
Agama Islam
Status Perkawinan Kawin ▾
Pekerjaan CTO of Google
Kewarganegaraan WNA ▾
Status dalam Keluarga Kepala Keluarga ▾
id keluarga 1000
Update



Saat diakses kembali halaman tersebut tidak akan ditemukan karena NIKnya sudah diganti akibat domisili dari penduduk tersebut sudah diubah (**Validasi NIK tidak ditemukan dalam database**)

NOT FOUND - Google Chrome
localhost:8080/penduduk/ubah/3101011910170002
Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Halaman yang anda cari tidak ditemukan

Nik tersebut berubah menjadi **3101021910170001**

Insert title here - Google Chrome
localhost:8080/penduduk?nik=3101021910170001
Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Lihat Data Penduduk - 3101021910170001

NIK : 3101021910170001
Nama : Testing Update
Tempat/Tanggal Lahir : London, 2017-10-19
Alamat : 3101021910170001
RT/RW : 081/079
Kelurahan/Desa : PULAU PANGGANG
Kecamatan : KEPULAUAN SERIBU UTARA
Kota : KABUPATEN KEPULAUAN SERIBU
Golongan Darah : O+
Agama : Islam
Status Perkawinan : Kawin
Pekerjaan : CTO of Google
Kewarganegaraan : WNA
Status Kematian : hidup

[Nonaktifkan](#) [Edit](#)

Validasi NIK tidak ditemukan didalam url



Validasi id keluarga tidak ditemukan (id keluarga diubah menjadi -1)

The image contains two side-by-side screenshots of a web application. Both screenshots show a form with the following fields:

- Nama: Testing Update
- Tempat Lahir: London
- Tanggal Lahir: 10/19/2017
- Jenis Kelamin: Pria
- Golongan Darah: O+
- Agama: Islam
- Status Perkawinan: Kawin
- Pekerjaan: CTO of Google
- Kewarganegaraan: WNA
- Status dalam Keluarga: Kepala Keluarga
- id keluarga: -1

The right screenshot includes an "Update" button below the form. Above the right screenshot, the word "ERROR!" is displayed in large capital letters, and a message below it says "Lengkapi seluruh data terlebih dahulu dengan valid".

Validasi jika data tidak lengkap (nama dikosongkan)

The screenshot displays two browser windows side-by-side. Both windows have a header bar with icons and a menu bar. The left window shows a form for updating a resident's information. The right window shows an error message.

Left Window (Form):

- Header: Insert title here - Google Chrome
- Address Bar: Tugas ADPAP N Ric x Tugas 1 Individu (R x) localhost:8080/penduduk/ubah/310102191015
- Menu Bar: Logo, Home, Tambah Penduduk, Tambah Keluarga
- Form Fields:
 - Nama: (empty input field)
 - Tempat Lahir: London
 - Tanggal Lahir: 10/19/2017
 - Jenis Kelamin: Pria
 - Golongan Darah: O+
 - Agama: Islam
 - Status Perkawinan: Kawin
 - Pekerjaan: CTO of Google
 - Kewarganegaraan: WNA
 - Status dalam Keluarga: Kepala Keluarga
 - id keluarga: 1000
- Buttons: Update

Right Window (Error Message):

- Header: Insert title here - Google Chrome
- Address Bar: Tugas ADPAP N Ric x Tugas 1 Individu (R x) localhost:8080/penduduk/ubah/310102191015
- Menu Bar: Logo, Home, Tambah Penduduk, Tambah Keluarga
- Error Message: ERROR!
- Text: Lengkapi seluruh data terlebih dahulu dengan valid
- Form Fields:
 - Nama: (empty input field)
 - Tempat Lahir: London
 - Tanggal Lahir: 10/19/2017
 - Jenis Kelamin: Pria
 - Golongan Darah: O+
 - Agama: Islam
 - Status Perkawinan: Kawin
 - Pekerjaan: CTO of Google
 - Kewarganegaraan: WNA
 - Status dalam Keluarga: Kepala Keluarga
 - id keluarga: 1000
- Buttons: Update

Penjelasan Kode

```
@RequestMapping(value={"/penduduk/ubah", "/penduduk/ubah/{nik}"})
public String formUpdatePenduduk(@PathVariable Optional<String> nik, Model model){
    if(!nik.isPresent()){
        return "not-found";
    }
    Penduduk penduduk = pendudukDAO.selectPenduduk(nik.get());
    if(penduduk == null){
        return "not-found";
    }
    model.addAttribute("penduduk", penduduk);
    model.addAttribute("isUpdated", false);
    return "form-update-penduduk";
}
```

Method tersebut akan dijalankan ketika *end point* “/penduduk/ubah” dipanggil di url, jika nik tidak terdapat pathvariable nik maka akan menampilkan halaman not found, jika nik ada namun tidak ditemukan pada database maka akan menampilkan halaman not found juga. Jika ada maka akan menampilkan form update penduduk.

```

    @RequestMapping(value="/penduduk/update/{nik}", method = RequestMethod.POST)
    public String updatePenduduk(@Valid Penduduk penduduk, BindingResult bindingResult, Model model){

        if(bindingResult.hasErrors()){
            model.addAttribute("penduduk", penduduk);
            model.addAttribute("isUpdated", false);
            model.addAttribute("isError", true);
            return "form-update-penduduk";
        }
        Keluarga keluarga = keluargaDAO.selectKeluarga(penduduk.getId_keluarga());
        if(keluarga == null){
            model.addAttribute("penduduk", penduduk);
            model.addAttribute("isAdded", false);
            model.addAttribute("isError", true);
            return "form-update-penduduk";
        }

        Kelurahan kelurahan = lokasiDAO.selectKelurahan(keluarga.getId_kelurahan());
        Kecamatan kecamatan = lokasiDAO.selectKecamatan(kelurahan.getId_kecamatan());
        String oldNik = penduduk.getNik();
        String[] birth = penduduk.getTanggal_lahir().split("-");
        String year = birth[0].substring(2);
        String month = birth[1];
        int tempDay = Integer.parseInt(birth[2]);
        tempDay = penduduk.getJenis_kelamin() == 0 ? tempDay : tempDay + 40;
        String day = tempDay < 10 ? "0" + tempDay : "" + tempDay;
        String nik = kecamatan.getKode_kecamatan().substring(0, kecamatan.getKode_kecamatan().length() - 1);
        nik += day + month + year;
        if(!nik.equals(penduduk.getNik().substring(0, 12))){
            List<Penduduk> similar = pendudukDAO.selectSimilarNik(nik);
            String nomor_urut = "0001";
            if(similar.size() > 0){
                nomor_urut = "" + (Integer.parseInt(similar.get(similar.size() - 1).getNik().substring(12)) + 1);
                for(int i = nomor_urut.length(); i < 4; i++){
                    nomor_urut = "0" + nomor_urut;
                }
            }
            nik += nomor_urut;
            penduduk.setNik(nik);
        }
        pendudukDAO.updatePenduduk(penduduk);
        model.addAttribute("nik", oldNik);
        model.addAttribute("isUpdated", true);
        return "form-update-penduduk";
    }
}

```

Method ini akan dijalankan ketika form update penduduk disubmit., akan terjadi fom validation menggunakan anotasi `@Valid` seperti yang sudah dijelaskan pada fitur sebelumnya. Jika error maka akan menampilkan pesan error pada halaman form update penduduk dan menampilkan pesan error, jika tidak ada error maka objek keluarga akan di cari di dalam database, jika tidak ada maka akan menampilkan pesan error di halaman form update penduduk, jika ada di dalam database maka akan dicek apakah terjadi perubahan domisili atau tanggal lahir. Jika iya maka nik akan digenerate seperti metode yang dijelaskan pada penduduk tambah , setelah itu data penduduk akan diupdate di dalam database dan menampilkan pesan sukses pada halaman form update penduduk.

Fitur 6 - Mengubah Data Keluarga

Langkah pertama yang perlu dilakukan adalah membuka detail keluarga

Lihat Data Keluarga - 3101010104070002

NKK : 3101010104070002

Alamat : Jln. Sutami No. 781

RT/RW : 137/033

Kelurahan/Desa : PULAU UNTUNG JAVA

Kecamatan : KEPULAUAN SERIBU SELATAN

Kota : KABUPATEN KEPULAUAN SERIBU

No	Nama Lengkap	NIK	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Agama	Pekerjaan	Status Perkawinan	Status dalam Keluarga	Kewarganegaraan
1	Kamila Yuliarti	3101016307830002	Wanita	Jakarta	1983-07-23	Katholik	TRANSPORTASI	Kawin	Istri	WNI
2	Jati Megantara	3101012005850002	Pria	Jakarta	1985-05-20	Katholik	PERANCANG BUSANA	Kawin	Kepala Keluarga	WNI
3	Narji Kusumo	3101012609870001	Pria	Jakarta	1987-09-26	Islam	ARSITEK	Kawin	Famil Lain	WNI
4	Testing Penduduk	3101012010170002	Pria	Bekasi	2017-10-20	Islam	Software Engineer	Belum Kawin	Anak	WNI

Lalu klik Edit

Update Keluarga - Google Chrome

localhost:8080/keluarga/ubah/3101010104070002

Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Alamat Jln. Sutami No. 781

RT 137

RW 033

KABUPATEN KEPULAUAN SERIBU ▾

KEPULAUAN SERIBU SELATAN ▾

PULAU UNTUNG JAVA ▾

Update

Lalu ubah data keluarga (pada kasus ini saya merubah alamat dan domisili agar nkk juga ikut berubah) nkk awal **3101010104070002**

Update Keluarga - Google Chrome

Tugas ADPAP N Ric... Tugas 1 Individu (R...
localhost:8080/keluarga/ubah/3101010104070002

Logo Home Tambah Penduduk Tambah Keluarga

Alamat: Jln. Testing Update
RT: 137
RW: 033
Pilih Kota: KOTA JAKARTA SELATAN
Pilih Kecamatan: PASAR MINGGU
Pilih Kelurahan: CILANDAK TIMUR
Update

Data keluarga sukses diupdate dan nkk berubah menjadi **3171020104070001**

Update Keluarga - Google Chrome

Tugas ADPAP N Ric... Tugas 1 Individu (R... Updat...
localhost:8080/keluarga/ubah/3101010104070002

Logo Home Tambah Penduduk Tambah Keluarga Cari

Sukses!

Keluarga dengan NKK 3101010104070002 berhasil diubah

Alamat: Jln. Testing Update
RT: 137
RW: 033
Pilih Kota
Pilih Kecamatan
Pilih Kelurahan
Update

Lihat Data Keluarga - 3171020104070001

NKK : 3171020104070001
 Alamat : Jln. Testing Update
 RT/RW : 137/033
 Kelurahan/Desa : CILANDAK TIMUR
 Kecamatan : PASAR MINGGU
 Kota : KOTA JAKARTA SELATAN

No	Nama Lengkap	NIK	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Agama	Pekerjaan	Status Perkawinan	Status dalam Keluarga	Kewarganegaraan
1	Kamila Yuliarti	3171026307830002	Wanita	Jakarta	1983-07-23	Katholik	TRANSPORTASI	Kawin	Istri	WNI
2	Jati Megantara	3171022005850001	Pria	Jakarta	1985-05-20	Katholik	PERANCANG BUSANA	Kawin	Kepala Keluarga	WNI
3	Narji Kusumo	3171022609870001	Pria	Jakarta	1987-09-26	Islam	ARSITEK	Kawin	Famili Lain	WNI
4	Testing Penduduk	3171022010170001	Pria	Bekasi	2017-10-20	Islam	Software Engineer	Belum Kawin	Anak	WNI

Karena Domisili keluarga berubah maka nik setiap anggota akan berubah juga mengikuti domisili tersebut

Kota : KOTA JAKARTA SELATAN

Edit

No	Nama Lengkap	NIK	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Agama	Pekerjaan	Status Perkawinan	Status dalam Keluarga	Kewarganegaraan
1	Kamila Yuliarti	3171026307830002	Wanita	Jakarta	1983-07-23	Katholik	TRANSPORTASI	Kawin	Istri	WNI
2	Jati Megantara	3171022005850001	Pria	Jakarta	1985-05-20	Katholik	PERANCANG BUSANA	Kawin	Kepala Keluarga	WNI
3	Narji Kusumo	3171022609870001	Pria	Jakarta	1987-09-26	Islam	ARSITEK	Kawin	Famili Lain	WNI
4	Testing Penduduk	3171022010170001	Pria	Bekasi	2017-10-20	Islam	Software Engineer	Belum Kawin	Anak	WNI

Validasi jika form tidak lengkap

Update Keluarga - Google Chrome

localhost:8080/keluarga/ubah/3171020102170002

Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Alamat: (highlighted in red)

RT: 073

RW: 025

Kota: KOTA JAKARTA SELATAN

Pilih Kelurahan/Desa: PASAR MINGGU

Pilih Kelurahan/Desa: CILANDAK TIMUR

Update

Update Keluarga - Google Chrome

localhost:8080/keluarga/ubah/3171020102170002

Logo Home Tambah Penduduk Tambah Keluarga Cari Data

Alamat: Dk. Barang iang Idh No. (highlighted in red)

RT: 073

RW: 025

Kota: KOTA JAKARTA SELATAN

Pilih Kelurahan/Desa: PASAR MINGGU

Pilih Kelurahan/Desa: CILANDAK TIMUR

Update

ERROR!

Lengkapi seluruh data terlebih dahulu dengan valid

Penjelasan Kode

```
@RequestMapping(value={"/keluarga/ubah", "/keluarga/ubah/{nkk}"})
public String formUpdateKeluarga(@PathVariable Optional<String> nkk ,Model model){
    if(!nkk.isPresent()){
        return "not-found";
    }
    Keluarga keluarga = keluargaDAO.selectKeluargaNkk(nkk.get());
    if(keluarga == null){
        return "not-found";
    }
    model.addAttribute("keluarga", keluarga);
    Kelurahan kelurahan = lokasiDAO.selectKelurahan(keluarga.getId_kelurahan());
    Kecamatan kecamatan = lokasiDAO.selectKecamatan(kelurahan.getId_kecamatan());

    model.addAttribute("id_kelurahan", keluarga.getId_kelurahan());
    model.addAttribute("id_kecamatan", kelurahan.getId_kecamatan());
    model.addAttribute("id_kota", kecamatan.getId_kota());

    model.addAttribute("isUpdated", false);
    return "form-update-keluarga";
}
```

Method tersebut akan dijalankan jika *end point* “/keluarga/ubah” dipanggil di url. Jika nkk tidak ada di dalam path variable maka akan menampilkan not found, jika ada maka nkk tersebut akan dicari di database jika tidak ada maka akan menampilkan not found. Jika ada maka akan menampilkan form update keluarga.

```

@RequestMapping(value = "/keluarga/ubah/{nkk}", method = RequestMethod.POST)
public String updateKeluarga(@Valid Keluarga keluarga, BindingResult bindingResult, Model model){

    if(bindingResult.hasErrors()){
        Keluarga temp = keluargaDAO.selectKeluargaNkk(keluarga.getNomor_kk());
        model.addAttribute("keluarga", temp);
        Kelurahan kelurahan = lokasiDAO.selectKelurahan(keluarga.getId_kelurahan());
        Kecamatan kecamatan = lokasiDAO.selectKecamatan(kelurahan.getId_kecamatan());

        model.addAttribute("id_kelurahan", keluarga.getId_kelurahan());
        model.addAttribute("id_kecamatan", kelurahan.getId_kecamatan());
        model.addAttribute("id_kota", kecamatan.getId_kota());

        model.addAttribute("isUpdated", false);
        return "form-update-keluarga";
    }

    Kelurahan kelurahan = lokasiDAO.selectKelurahan(keluarga.getId_kelurahan());

    DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy/MM/dd");

    String date = keluarga.getNomor_kk().substring(6,12);

    String nkk = kelurahan.getKode_kelurahan().substring(0,6) + date;
    if(!nkk.equals(keluarga.getNomor_kk())){

        List<Keluarga> similar = keluargaDAO.selectSimilarNkk(nkk);
        String nomor_urut = "0001";
        if(similar.size() > 0){
            nomor_urut = "" + (Integer.parseInt(similar.get(similar.size() - 1).getNomor_kk().substring(6,12)) + 1);
            for(int i = nomor_urut.length(); i < 4; i++){
                nomor_urut = "0" + nomor_urut;
            }
        }

        nkk += nomor_urut;
        keluarga.setNomor_kk(nkk);
        List<Penduduk> anggotaKeluarga = pendudukDAO.selectAnggotaKeluarga(keluarga.getId());

        Kecamatan kecamatan = lokasiDAO.selectKecamatan(kelurahan.getId_kecamatan());
        for(Penduduk penduduk : anggotaKeluarga){
            String[] birth = penduduk.getTanggal_lahir().split("-");
            String year = birth[0].substring(2);

            if(similar_penduduk.size() > 0){
                nomor_urut_penduduk = "" + (Integer.parseInt(similar_penduduk.get(similar_penduduk.size() - 1).getNomor_kk().substring(6,12)) + 1);
                for(int i = nomor_urut_penduduk.length(); i < 4; i++){
                    nomor_urut_penduduk = "0" + nomor_urut_penduduk;
                }
            }
            nik += nomor_urut_penduduk;
            penduduk.setNik(nik);
            pendudukDAO.updatePenduduk(penduduk);
        }
    }

    keluargaDAO.updateKeluarga(keluarga);
    model.addAttribute("nkk", keluarga.getNomor_kk());
    model.addAttribute("isUpdated", true);
    return "form-update-keluarga";
}

```

Method ini akan diakses ketika form method update disubmit , akan dilakukan form validations seperti yang dijelaskan sebelumnya. Lalu data keluarga akan diupdate beserta nkknya jika terjadi perubahan pada domisili, jika terjadi perubahan pada domisili maka nik anggota keluarga juga akan diubah, lalu akan menampilkan pesan sukses ke form update keluarga.

Fitur 7 - Mengubah Status Kematian Penduduk

Langkah pertama yang perlu dilakukan adalah mengkases detail penduduk

A screenshot of a web browser window titled "Insert title here - Google Chrome". The address bar shows "localhost:8080/penduduk?nik=3101010101690001". The main content displays the following information:

Lihat Data Penduduk - 3101010101690001

NIK : 3101010101690001
Nama : Ridwan Hasim Marpaung
Tempat/Tanggal Lahir : Atambua, 1969-01-01
ALamat : 3101010101690001
RT/RW : 177/056
Kelurahan/Desa : PULAU UNTUNG JAWA
Kecamatan : KEPULAUAN SERIBU SELATAN
Kota : KABUPATEN KEPULAUAN SERIBU
Golongan Darah : B-
Agama : Islam
Status Perkawinan : Kawin
Pekerjaan : PEMBANTU RUMAH TANGGA
Kewarganegaraan : WNI
Status Kematian : hidup

Nonaktifkan Edit

The sidebar on the left contains icons for various applications like file manager, browser, and system settings.

Klik button nonaktifkan dan akan menampilkan pesan sukses

A screenshot of a web browser window titled "Insert title here - Google Chrome". The address bar shows "localhost:8080/penduduk". The main content displays the following message:

Sukses !

Penduduk dengan NIK 3101010101690001 sudah tidak aktif

Lihat Data Penduduk - 3101010101690001

NIK : 3101010101690001
Nama : Ridwan Hasim Marpaung
Tempat/Tanggal Lahir : Atambua, 1969-01-01
ALamat : 3101010101690001
RT/RW : 177/056
Kelurahan/Desa : PULAU UNTUNG JAWA
Kecamatan : KEPULAUAN SERIBU SELATAN
Kota : KABUPATEN KEPULAUAN SERIBU
Golongan Darah : B-
Agama : Islam
Status Perkawinan : Kawin
Pekerjaan : PEMBANTU RUMAH TANGGA
Kewarganegaraan : WNI
Status Kematian : wafat

Edit

The sidebar on the left contains icons for various applications like file manager, browser, and system settings.

Penjelasan Kode

```
@RequestMapping(value = "/penduduk", method = RequestMethod.POST)
public String ubahStatus(@RequestParam(value="nik", required = false) String nik, @RequestParam(valu
    if(nik == null){
        return "not-found";
    }

    pendudukDAO.updateStatusKematian(nik);

    List<Penduduk> anggotaKeluarga = pendudukDAO.selectAnggotaKeluarga(id_keluarga);
    boolean isTidakBerlaku = true;
    for(Penduduk penduduk : anggotaKeluarga){
        if(penduduk.getIs_wafat() == 0){
            isTidakBerlaku = false;
            break;
        }
    }

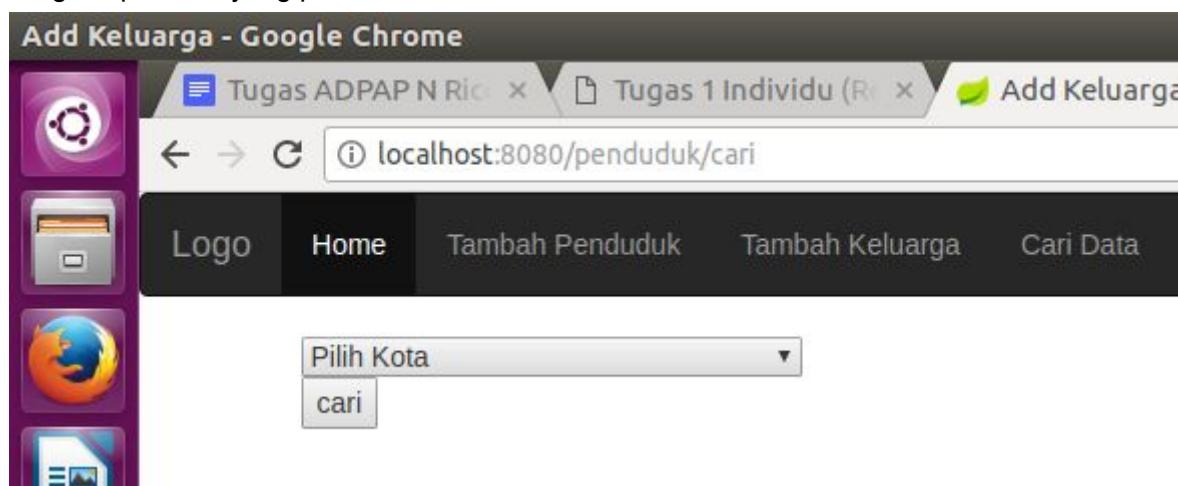
    if(isTidakBerlaku){
        keluargaDAO.updateStatusKeluarga(id_keluarga);
    }
    Penduduk penduduk = pendudukDAO.selectPenduduk(nik);
    Keluarga keluarga = keluargaDAO.selectKeluarga(penduduk.getId_keluarga());
    Kelurahan kelurahan = lokasiDAO.selectKelurahan(keluarga.getId_kelurahan());
    Kecamatan kecamatan = lokasiDAO.selectKecamatan(kelurahan.getId_kecamatan());
    Kota kota = lokasiDAO.selectKota(kecamatan.getId_kota());

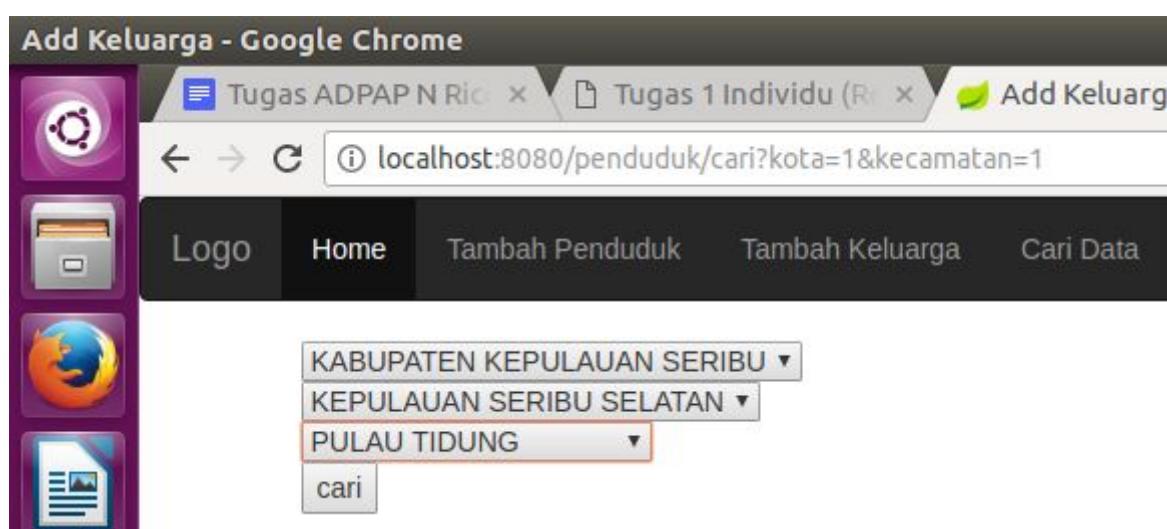
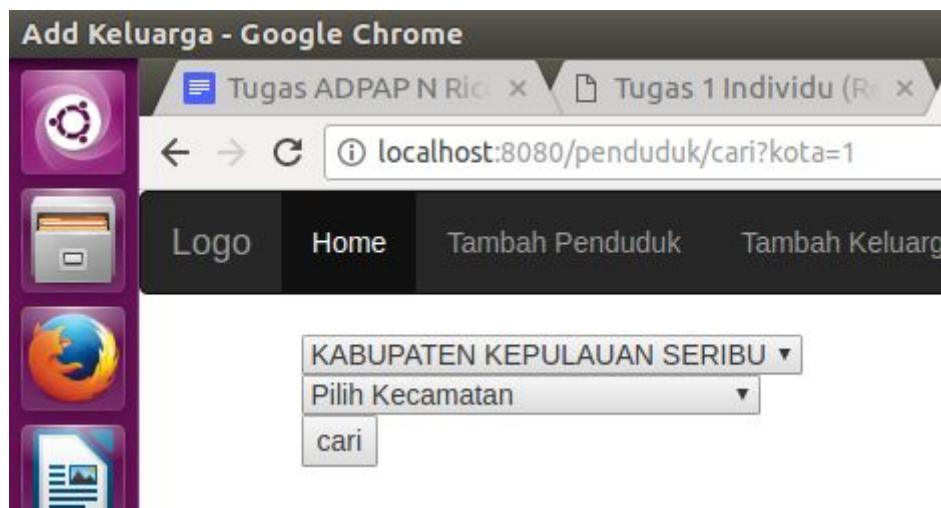
    model.addAttribute("keluarga", keluarga);
    model.addAttribute("kelurahan", kelurahan);
    model.addAttribute("kecamatan", kecamatan);
    model.addAttribute("kota", kota);
    model.addAttribute("penduduk", penduduk);
    model.addAttribute("isChanged", true);
    return "penduduk";
}
```

Method ini akan dijalankan ketika status kematian penduduk diubah, jika penduduk tersebut merupakan anggota keluarga terakhir yang masih hidup maka status keluarga dinonaktifkan.

Fitur 8 & bonus - Tampilkan Data Penduduk Berdasarkan Kota/Kabupaten, Kecamatan, dan Kelurahan Tertentu & Menampilkan Penduduk Paling Muda dan Paling Tua di Suatu Kelurahan

Langkah pertama yang perlu dilakukan adalah membuka halaman cari data dan memilih domisili





Hasil dari proses pencarian

The screenshot shows the search results for "PULAU TIDUNG". The main content area displays two tables of data: "Penduduk dengan usia termuda" and "Penduduk dengan usia tertua". Below these are two tables of individual records. At the bottom, there is a table listing multiple individuals with columns for Nik, Nama, and Kelamin (Gender). The table has a header row and several data rows.

Nik	Nama	Kelamin
3101014101550001	Cintia Usamah	Pria
3101017108170001	Ami Uyainah S.I.P	Pria
3101010104890001	Taswir Hutasoit	Pria
3101010104910001	Mahfud Setiawan M.TI.	Pria
3101010105890001	Radika Dabukke	Pria
3101010108140001	Dalimin Ihsan Manullang S.Kom	Pria

Penjelasan kode

```
@RequestMapping("/penduduk/cari")
public String cariPenduduk(@RequestParam(value = "kota", required = false) String id_kota,
|   @RequestParam(value = "kecamatan", required = false) String id_kecamatan,
|   @RequestParam(value = "kelurahan", required = false) String id_kelurahan, Model model){

    model.addAttribute("kota", lokasiDAO.selectAllKota());
    if(id_kota != null && id_kecamatan != null && id_kelurahan != null){
        List<Penduduk> list_penduduk = pendudukDAO.selectPendudukBylocation(Integer.parseInt(id_kelurahan));
        Kelurahan kelurahan = lokasiDAO.selectKelurahan(Integer.parseInt(id_kelurahan));
        Kecamatan kecamatan = lokasiDAO.selectKecamatan(kelurahan.getId_kecamatan());
        Kota kota = lokasiDAO.selectKota(kecamatan.getId_kota());

        model.addAttribute("list_penduduk", list_penduduk);
        model.addAttribute("kelurahan", kelurahan);
        model.addAttribute("kecamatan", kecamatan);
        model.addAttribute("kota", kota);
        model.addAttribute("isResult", false);
        return "list-penduduk";
    }
    else if(id_kota != null){
        int temp_kota = Integer.parseInt(id_kota);
        model.addAttribute("kecamatan", lokasiDAO.selectKecamatanByKota(temp_kota));
        model.addAttribute("selected_id_kota", temp_kota);
        model.addAttribute("isKecamatan", true);
        if(id_kecamatan != null){
            int temp_kecamatan = Integer.parseInt(id_kecamatan);
            model.addAttribute("kelurahan", lokasiDAO.selectKelurahanByKecamatan(temp_kecamatan));
            model.addAttribute("selected_id_kota", temp_kota);
            model.addAttribute("selected_id_kecamatan", temp_kecamatan);
            model.addAttribute("isKelurahan", true);
        }
    }
    model.addAttribute("isResult", true);
    return "list-penduduk";
}
```

Method tersebut akan dijalankan ketika tombol cari di klik pada halaman cari data. Jika kota kecamatan dan kelurahan sudah terisi maka akan dilakukan proses pencarian dan menampilkan hasil tersebut dalam bentuk datatable. Jika tidak maka akan dicari kecamatan berdasarkan kota, dan mencari kelurahan berdasarkan kecamatan.

```
var list_penduduk = [[${list_penduduk}]];
if(list_penduduk != null){
    var pendudukMax = list_penduduk[0];
    var pendudukMin = list_penduduk[0];
    for(var i = 1 ; i < list_penduduk.length; i++){
        var penduduk = list_penduduk[i];
        var tempDate = penduduk['tanggal_lahir'].split("-");
        var tempDateMax = pendudukMax['tanggal_lahir'].split("-");
        var tempDateMin = pendudukMin['tanggal_lahir'].split("-");
        var date = new Date(tempDate[2], tempDate[0] - 1, tempDate[1]);
        var dateMax = new Date(tempDateMax[2], tempDateMax[0] - 1, tempDateMax[1]);
        var dateMin = new Date(tempDateMin[2], tempDateMin[0] - 1, tempDateMin[1]);

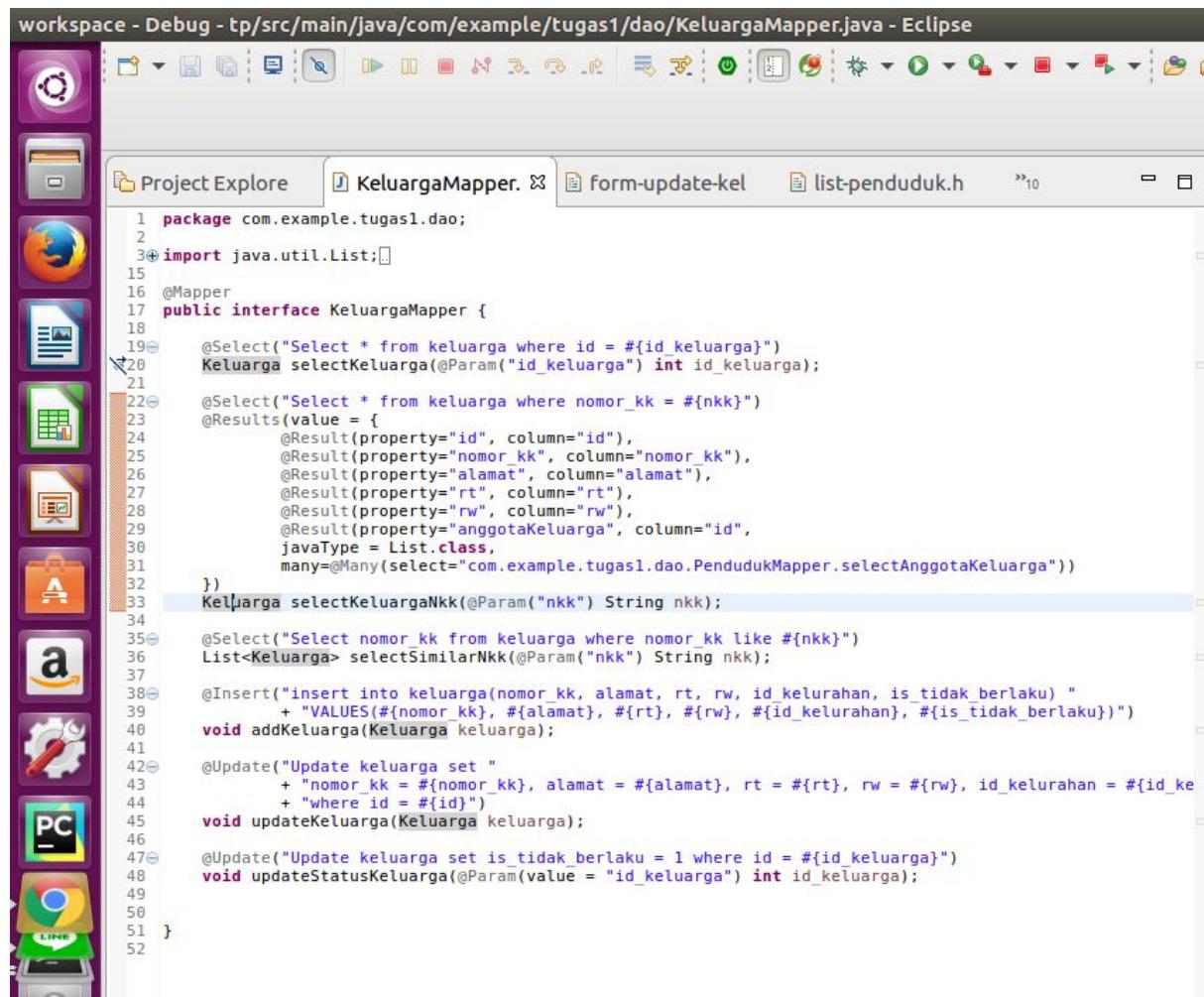
        if(date > dateMax){
            pendudukMax = penduduk;
        }
        if(date < dateMin){
            pendudukMin = penduduk;
        }
    }
    penduduk['jenis_kelamin'] = penduduk['jenis_kelamin'] == 0 ? 'Pria' : 'Wanita';
}
```

Diatas merupakan algoritma untuk mencari penduduk termuda dan tertua menggunakan javascript

Optimisasi database

Untuk mempercepat proses eksekusi query, saya menambahkan primary key pada kolom id pada setiap table yang ada di dalam database, lalu saya menambahkan index pada kolom nik pada table penduduk dan kolom nomor_kk pada table keluarga, hal ini dikarenakan kedua kolom tersebut sering digunakan sebagai condition pada query-query yang ada di dalam program ini. Saya juga menambahkan index pada kolom id_keluarga, id_kelurahan, id_kota, dan id_kecamatan karena digunakan dalam proses join agar proses joinnya tidak mencapai kompleksitas N^2 .

Query keluarga

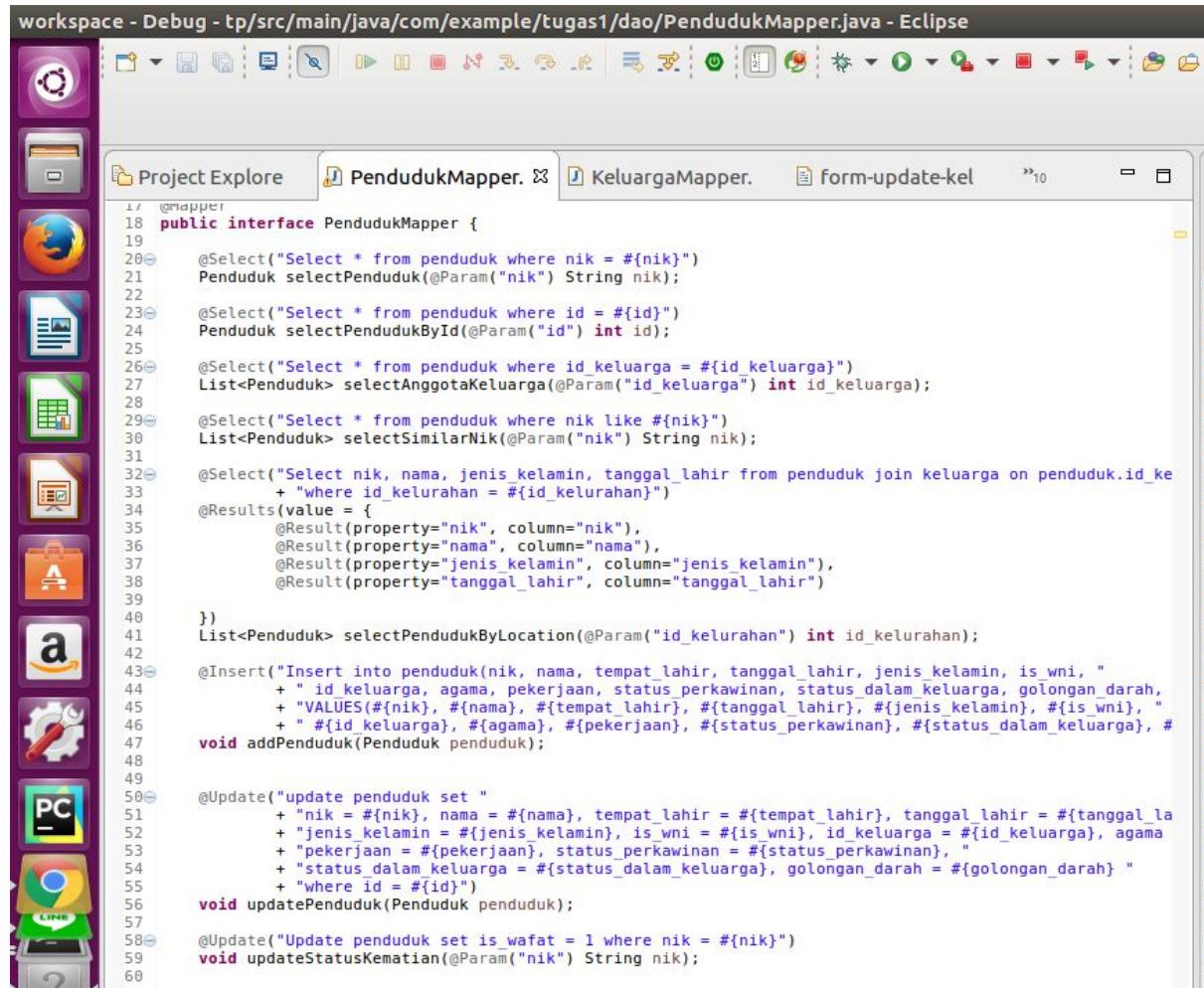


The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** workspace - Debug - tp/src/main/java/com/example/tugas1/dao/KeluargaMapper.java - Eclipse
- Toolbar:** Standard Eclipse toolbar with various icons for file operations, search, and run.
- Project Explorer:** Shows the project structure with "KeluargaMapper" selected.
- Code Editor:** Displays the Java code for the KeluargaMapper interface. The code includes annotations for MyBatis (e.g., @Select, @Insert, @Update) and Java code for managing a list of household members.

```
1 package com.example.tugas1.dao;
2
3 import java.util.List;
4
5 @Mapper
6 public interface KeluargaMapper {
7
8     @Select("Select * from keluarga where id = #{id_keluarga}")
9     Keluarga selectKeluarga(@Param("id_keluarga") int id_keluarga);
10
11    @Select("Select * from keluarga where nomor_kk = #{nkk}")
12    @Results(value = {
13        @Result(property="id", column="id"),
14        @Result(property="nomor_kk", column="nomor_kk"),
15        @Result(property="alamat", column="alamat"),
16        @Result(property="rt", column="rt"),
17        @Result(property="rw", column="rw"),
18        @Result(property="anggotaKeluarga", column="id",
19            javaType = List.class,
20            many=@Many(select="com.example.tugas1.dao.PendudukMapper.selectAnggotaKeluarga"))
21    })
22    Keluarga selectKeluargaNkk(@Param("nkk") String nkk);
23
24    @Select("Select nomor_kk from keluarga where nomor_kk like #{nkk}")
25    List<Keluarga> selectSimilarNkk(@Param("nkk") String nkk);
26
27    @Insert("insert into keluarga(nomor_kk, alamat, rt, rw, id_kelurahan, is_tidak_berlaku) "
28        + "VALUES(#{nomor_kk}, #{alamat}, #{rt}, #{rw}, #{id_kelurahan}, #{is_tidak_berlaku})")
29    void addKeluarga(Keluarga keluarga);
30
31    @Update("Update keluarga set "
32        + "nomor_kk = #{nomor_kk}, alamat = #{alamat}, rt = #{rt}, rw = #{rw}, id_kelurahan = #{id_ke
33        + "where id = #{id}"})
34    void updateKeluarga(Keluarga keluarga);
35
36    @Update("Update keluarga set is_tidak_berlaku = 1 where id = #{id_keluarga}")
37    void updateStatusKeluarga(@Param(value = "id_keluarga") int id_keluarga);
38
39}
40
41}
```

Query penduduk

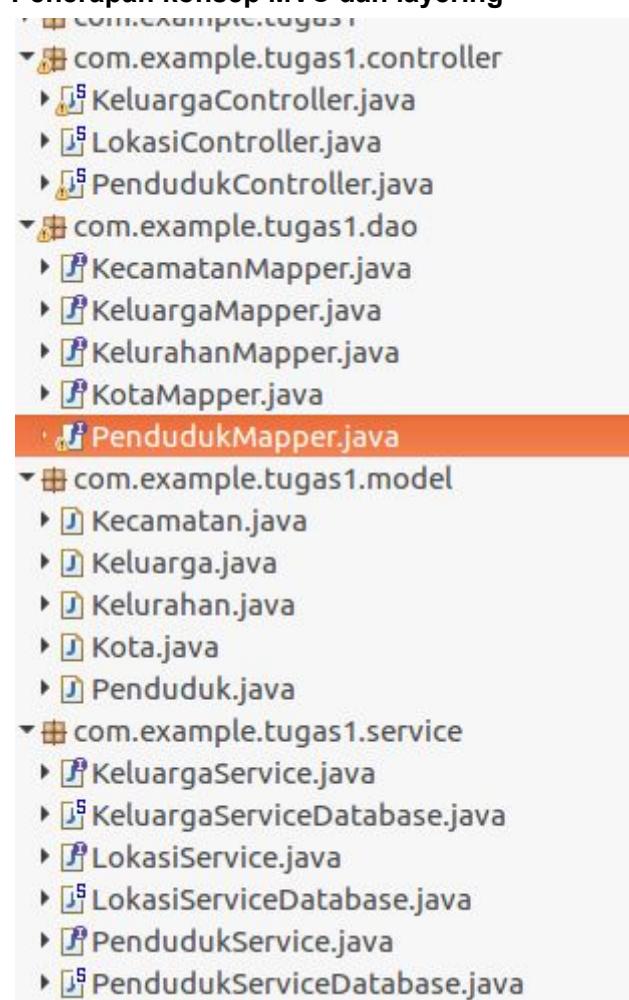


The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** workspace - Debug - tp/src/main/java/com/example/tugas1/dao/PendudukMapper.java - Eclipse
- Project Explorer:** Shows the file `PendudukMapper.java` selected.
- Code Editor:** Displays the Java code for the `PendudukMapper` interface. The code includes methods for selecting, inserting, updating, and deleting data from the `Penduduk` and `Keluarga` tables.

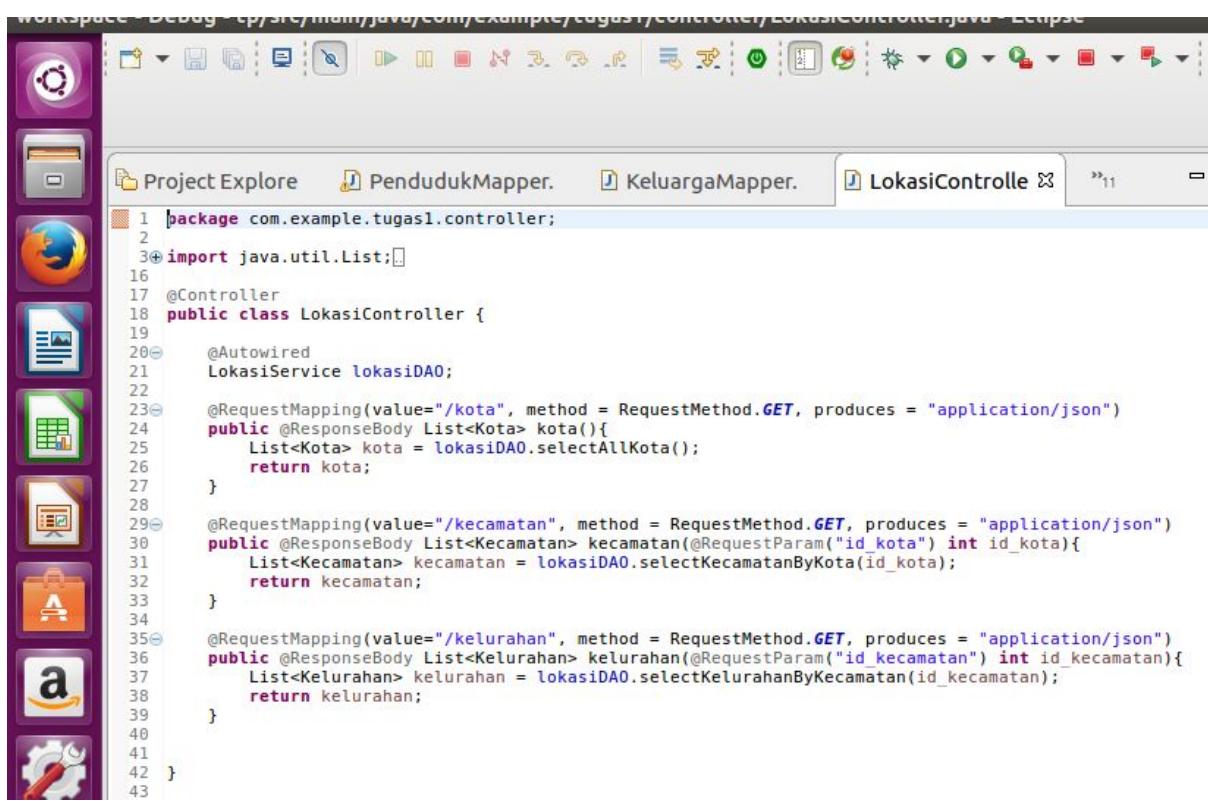
```
1  @Mapper
2  public interface PendudukMapper {
3
4      @Select("Select * from penduduk where nik = #{nik}")
5      Penduduk selectPenduduk(@Param("nik") String nik);
6
7      @Select("Select * from penduduk where id = #{id}")
8      Penduduk selectPendudukById(@Param("id") int id);
9
10     @Select("Select * from penduduk where id_keluarga = #{id_keluarga}")
11     List<Penduduk> selectAnggotaKeluarga(@Param("id_keluarga") int id_keluarga);
12
13     @Select("Select * from penduduk where nik like #{nik}")
14     List<Penduduk> selectSimilarNik(@Param("nik") String nik);
15
16     @Select("Select nik, nama, jenis_kelamin, tanggal_lahir from penduduk join keluarga on penduduk.id_ke
17             + "luruhan = #{id_kelurahan}"
18     @Results(value = {
19         @Result(property="nik", column="nik"),
20         @Result(property="nama", column="nama"),
21         @Result(property="jenis_kelamin", column="jenis_kelamin"),
22         @Result(property="tanggal_lahir", column="tanggal_lahir")
23     })
24     List<Penduduk> selectPendudukByLocation(@Param("id_kelurahan") int id_kelurahan);
25
26     @Insert("Insert into penduduk(nik, nama, tempat_lahir, tanggal_lahir, jenis_kelamin, is_wni, "
27             + "id_keluarga, agama, pekerjaan, status_perkawinan, status_dalam_keluarga, golongan_darah,
28             + "VALUES/#{nik}, #{nama}, #{tempat_lahir}, #{tanggal_lahir}, #{jenis_kelamin}, #{is_wni},
29             + #{id_keluarga}, #{agama}, #{pekerjaan}, #{status_perkawinan}, #{status_dalam_keluarga}, #"
30     void addPenduduk(Penduduk penduduk);
31
32
33     @Update("update penduduk set "
34             + "nik = #{nik}, nama = #{nama}, tempat_lahir = #{tempat_lahir}, tanggal_lahir = #{tanggal_la
35             + "jenis_kelamin = #{jenis_kelamin}, is_wni = #{is_wni}, id_keluarga = #{id_keluarga}, agama
36             + "pekerjaan = #{pekerjaan}, status_perkawinan = #{status_perkawinan}, "
37             + "status_dalam_keluarga = #{status_dalam_keluarga}, golongan_darah = #{golongan_darah} "
38             + "where id = #{id}")
39     void updatePenduduk(Penduduk penduduk);
40
41     @Update("Update penduduk set is_wafat = 1 where nik = #{nik}")
42     void updateStatusKematian(@Param("nik") String nik);
43
44 }
```

Penerapan konsep MVC dan layering



Pada project ini saya memiliki 4 package yaitu controller, dao, model, dan service Controller :

Pada controller saya membagi menjadi 3 file yaitu PendudukController, KeluargaController, dan LokasiController. PendudukController berfungsi untuk mengatur aktifitas yang berhubungan dengan penduduk seperti tambah, update, dll. KeluargaController berfungsi untuk mengatur aktifitas yang berhubungan dengan keluarga seperti tambah, update, dll. LokasiController berfungsi untuk mengirimkan data kota, kecamatan, dan kelurahan dalam bentuk JSON, yang akan digunakan pada fitur cari penduduk. Mengapa kota, kecamatan, kelurahan saya jadikan satu controller? Hal ini dikarenakan hanya terdapat 1 operasi dari masing-masing objek yaitu operasi selection sehingga lebih efisien jika dibuatkan 1 controller saja



```
1 package com.example.tugas1.controller;
2
3+ import java.util.List;
4
5+ @Controller
6 public class LokasiController {
7
8     @Autowired
9     LokasiService lokasiDAO;
10
11     @RequestMapping(value="/kota", method = RequestMethod.GET, produces = "application/json")
12     public @ResponseBody List<Kota> kota(){
13         List<Kota> kota = lokasiDAO.selectAllKota();
14         return kota;
15     }
16
17     @RequestMapping(value="/kecamatan", method = RequestMethod.GET, produces = "application/json")
18     public @ResponseBody List<Kecamatan> kecamatan(@RequestParam("id_kota") int id_kota){
19         List<Kecamatan> kecamatan = lokasiDAO.selectKecamatanByKota(id_kota);
20         return kecamatan;
21     }
22
23     @RequestMapping(value="/kelurahan", method = RequestMethod.GET, produces = "application/json")
24     public @ResponseBody List<Kelurahan> kelurahan(@RequestParam("id_kecamatan") int id_kecamatan){
25         List<Kelurahan> kelurahan = lokasiDAO.selectKelurahanByKecamatan(id_kecamatan);
26         return kelurahan;
27     }
28
29 }
30
31 }
```

Dao:

Package ini berfungsi untuk menyimpan file mapper yang berisi query-query yang digunakan dalam program ini, ada 5 macam file yaitu PendudukMapper, KeluargaMapper, KotaMapper, KecamatanMapper, dan KelurahanMapper.

Model:

Package ini berfungsi untuk menyimpan representasi data dari table yang nantinya akan disimpan dalam bentuk objek. Terdapat 5 model di dalam project ini yaitu Penduduk, Keluarga, kota, Kecamatan, dan Kelurahan.

Service

Package ini berfungsi sebagai *middle-layer* antara controller dengan database. Terdapat 6 file yaitu PendudukService, PendudukServiceDatabase, KeluargaService, KeluargaServiceDatabase, LokasiService, LokasiServiceDatabase. File Service adalah sebuah interface yang akan diimplementasikan oleh file ServiceDatabase. Setiap method yang terdapat dalam file ServiceDatabase akan memanggil method query di dalam file-fie mapper yang terdapat di dalam package dao.

Stress testing

Number of Threads sama dengan 100 dan Ramp-up period 10 detik

Url : <http://localhost:8080/penduduk?nik=3101010101690001>

View Results in Table										
Name: <input type="text" value="View Results in Table"/>										
Comments:										
Write results to file / Read from file										
Sample #	Start Time	Thread Name	Label	Sample Time...	Status	Bytes	Sent Bytes	Latency	Connect Tim...	
1	19:02:29.063	Thread Grou...	HTTP Request	17	✓	2269	147	17	1	
2	19:02:29.163	Thread Grou...	HTTP Request	7	✓	2269	147	7	1	
3	19:02:29.264	Thread Grou...	HTTP Request	18	✓	2269	147	18	0	
4	19:02:29.364	Thread Grou...	HTTP Request	7	✓	2269	147	7	1	
5	19:02:29.464	Thread Grou...	HTTP Request	6	✓	2269	147	6	1	
6	19:02:29.565	Thread Grou...	HTTP Request	6	✓	2269	147	6	0	
7	19:02:29.666	Thread Grou...	HTTP Request	9	✓	2269	147	9	1	
8	19:02:29.765	Thread Grou...	HTTP Request	6	✓	2269	147	5	0	
9	19:02:29.865	Thread Grou...	HTTP Request	5	✓	2269	147	5	0	
10	19:02:29.965	Thread Grou...	HTTP Request	5	✓	2269	147	5	0	
11	19:02:30.065	Thread Grou...	HTTP Request	7	✓	2269	147	7	0	
12	19:02:30.165	Thread Grou...	HTTP Request	15	✓	2269	147	15	1	
13	19:02:30.265	Thread Grou...	HTTP Request	11	✓	2269	147	11	1	
14	19:02:30.366	Thread Grou...	HTTP Request	7	✓	2269	147	7	0	
15	19:02:30.466	Thread Grou...	HTTP Request	5	✓	2269	147	5	0	
16	19:02:30.566	Thread Grou...	HTTP Request	20	✓	2269	147	20	0	
17	19:02:30.669	Thread Grou...	HTTP Request	8	✓	2269	147	8	1	
18	19:02:30.770	Thread Grou...	HTTP Request	5	✓	2269	147	5	0	
19	19:02:30.870	Thread Grou...	HTTP Request	9	✓	2269	147	9	1	
20	19:02:30.970	Thread Grou...	HTTP Request	6	✓	2269	147	6	1	
21	19:02:31.071	Thread Grou...	HTTP Request	9	✓	2269	147	9	0	
22	19:02:31.171	Thread Grou...	HTTP Request	7	✓	2269	147	7	0	
23	19:02:31.271	Thread Grou...	HTTP Request	12	✓	2269	147	11	0	
24	19:02:31.371	Thread Grou...	HTTP Request	11	✓	2269	147	11	1	
25	19:02:31.471	Thread Grou...	HTTP Request	12	✓	2269	147	12	1	
26	19:02:31.577	Thread Grou...	HTTP Request	7	✓	2269	147	7	0	
27	19:02:31.677	Thread Grou...	HTTP Request	10	✓	2269	147	10	1	
28	19:02:31.780	Thread Grou...	HTTP Request	8	✓	2269	147	8	1	
29	19:02:31.880	Thread Grou...	HTTP Request	5	✓	2269	147	5	0	
30	19:02:31.980	Thread Grou...	HTTP Request	6	✓	2269	147	6	0	

Berdasarkan result tersebut, waktu rata-rata yang dibutuhkan untuk mengakses url tersebut adalah 8 yang dimana terbilang cukup cepat, hal ini dikarenakan terdapat index pada kolom nik untuk mencari data penduduk dalam database sehingga operasi searchingnya ada Log N.

Url : <http://localhost:8080/keluarga?nkk=3101011010170001>

View Results in Table										
<input type="checkbox"/> Name: View Results in Table <input type="checkbox"/> Comments: <input type="checkbox"/> Write results to file / Read from file Filename <input type="text"/> <input type="button" value="Browse..."/> Log/Display Only: <input type="checkbox"/> Errors <input type="checkbox"/> Successes <input type="button" value="Configure"/>										
Sample #	Start Time	Thread Name	Label	Sample Time...	Status	Bytes	Sent Bytes	Latency	Connect Tim...	
1	19:03:23.577	Thread Grou...	HTTP Request	12	✓	2343	150	12	3	▲
2	19:03:23.677	Thread Grou...	HTTP Request	24	✓	2343	150	24	1	=
3	19:03:23.778	Thread Grou...	HTTP Request	6	✓	2343	150	6	0	=
4	19:03:23.878	Thread Grou...	HTTP Request	6	✓	2343	150	6	1	=
5	19:03:23.981	Thread Grou...	HTTP Request	5	✓	2343	150	5	0	=
6	19:03:24.081	Thread Grou...	HTTP Request	8	✓	2343	150	8	1	=
7	19:03:24.181	Thread Grou...	HTTP Request	7	✓	2343	150	7	1	=
8	19:03:24.281	Thread Grou...	HTTP Request	9	✓	2343	150	9	1	=
9	19:03:24.381	Thread Grou...	HTTP Request	7	✓	2343	150	7	1	=
10	19:03:24.482	Thread Grou...	HTTP Request	6	✓	2343	150	6	0	=
11	19:03:24.583	Thread Grou...	HTTP Request	6	✓	2343	150	6	0	=
12	19:03:24.683	Thread Grou...	HTTP Request	10	✓	2343	150	10	0	=
13	19:03:24.783	Thread Grou...	HTTP Request	7	✓	2343	150	7	0	=
14	19:03:24.883	Thread Grou...	HTTP Request	9	✓	2343	150	9	1	=
15	19:03:24.983	Thread Grou...	HTTP Request	7	✓	2343	150	7	0	=
16	19:03:25.085	Thread Grou...	HTTP Request	6	✓	2343	150	6	0	=
17	19:03:25.185	Thread Grou...	HTTP Request	5	✓	2343	150	5	0	=
18	19:03:25.285	Thread Grou...	HTTP Request	7	✓	2343	150	7	1	=
19	19:03:25.385	Thread Grou...	HTTP Request	7	✓	2343	150	7	1	=
20	19:03:25.485	Thread Grou...	HTTP Request	7	✓	2343	150	7	1	=
21	19:03:25.586	Thread Grou...	HTTP Request	6	✓	2343	150	6	0	=
22	19:03:25.686	Thread Grou...	HTTP Request	6	✓	2343	150	6	0	=
23	19:03:25.786	Thread Grou...	HTTP Request	6	✓	2343	150	6	1	=
24	19:03:25.885	Thread Grou...	HTTP Request	6	✓	2343	150	6	1	=
25	19:03:25.987	Thread Grou...	HTTP Request	5	✓	2343	150	5	0	=
26	19:03:26.087	Thread Grou...	HTTP Request	5	✓	2343	150	5	0	=
27	19:03:26.187	Thread Grou...	HTTP Request	5	✓	2343	150	5	0	=
28	19:03:26.287	Thread Grou...	HTTP Request	6	✓	2343	150	6	1	=
29	19:03:26.388	Thread Grou...	HTTP Request	8	✓	2343	150	8	0	=
30	19:03:26.486	Thread Grou...	HTTP Request	21	✓	2343	150	19	2	▼
<input type="checkbox"/> Scroll automatically?	<input type="checkbox"/> Child samples?	No of Samples	98		Latest Sample	6	Average	7	Deviation	4

Berdasarkan result tersebut, waktu rata-rata yang dibutuhkan untuk mengakses url tersebut adalah 7. Walaupun terdapat operasi join *many-to-many* saat akan mengambil list anggota keluarga query tersebut tetap berjalan dengan cepat karena sudah diberikan index pada atribut id_keluarga pada table penduduk dan nomor_kk pada table keluarga.

<http://localhost:8080/penduduk/cari?kota=1&kecamatan=1&kelurahan=1>

View Results in Table										
<input checked="" type="checkbox"/> Name: View Results in Table <input type="checkbox"/> Comments: <input type="checkbox"/> Write results to file / Read from file Filename <input type="text"/> <input type="button" value="Browse..."/> Log/Display Only: <input type="checkbox"/> Errors <input type="checkbox"/> Successes <input type="button" value="Configure"/>										
Sample #	Start Time	Thread Name	Label	Sample Time...	Status	Bytes	Sent Bytes	Latency	Connect Tim...	
1	18:59:05.922	Thread Grou...	HTTP Request	82	✓	390435	162	26	1	▲
2	18:59:06.004	Thread Grou...	HTTP Request	32	✓	390435	162	27	0	○
3	18:59:06.113	Thread Grou...	HTTP Request	43	✓	390435	162	27	0	○
4	18:59:06.213	Thread Grou...	HTTP Request	43	✓	390435	162	25	1	○
5	18:59:06.313	Thread Grou...	HTTP Request	64	✓	390435	162	44	1	○
6	18:59:06.418	Thread Grou...	HTTP Request	45	✓	390435	162	27	1	○
7	18:59:06.513	Thread Grou...	HTTP Request	26	✓	390435	162	23	0	○
8	18:59:06.613	Thread Grou...	HTTP Request	43	✓	390435	162	26	0	○
9	18:59:06.713	Thread Grou...	HTTP Request	42	✓	390435	162	25	0	○
10	18:59:06.813	Thread Grou...	HTTP Request	55	✓	390435	162	35	0	○
11	18:59:06.913	Thread Grou...	HTTP Request	28	✓	390435	162	26	1	○
12	18:59:07.013	Thread Grou...	HTTP Request	29	✓	390435	162	28	1	○
13	18:59:07.113	Thread Grou...	HTTP Request	24	✓	390435	162	22	0	○
14	18:59:07.213	Thread Grou...	HTTP Request	47	✓	390435	162	33	1	○
15	18:59:07.313	Thread Grou...	HTTP Request	101	✓	390435	162	35	0	○
16	18:59:07.413	Thread Grou...	HTTP Request	30	✓	390435	162	26	0	○
17	18:59:07.518	Thread Grou...	HTTP Request	38	✓	390435	162	36	1	○
18	18:59:07.613	Thread Grou...	HTTP Request	25	✓	390435	162	22	0	○
19	18:59:07.714	Thread Grou...	HTTP Request	58	✓	390435	162	36	14	○
20	18:59:07.813	Thread Grou...	HTTP Request	26	✓	390435	162	24	1	○
21	18:59:07.913	Thread Grou...	HTTP Request	28	✓	390435	162	25	1	○
22	18:59:08.012	Thread Grou...	HTTP Request	28	✓	390435	162	25	1	○
23	18:59:08.113	Thread Grou...	HTTP Request	55	✓	390435	162	33	1	○
24	18:59:08.212	Thread Grou...	HTTP Request	25	✓	390435	162	23	1	○
25	18:59:08.314	Thread Grou...	HTTP Request	31	✓	390435	162	29	1	○
26	18:59:08.414	Thread Grou...	HTTP Request	28	✓	390435	162	23	1	○
27	18:59:08.513	Thread Grou...	HTTP Request	25	✓	390435	162	23	0	○
28	18:59:08.614	Thread Grou...	HTTP Request	25	✓	390435	162	22	0	○
29	18:59:08.713	Thread Grou...	HTTP Request	27	✓	390435	162	25	0	○
30	18:59:08.813	Thread Grou...	HTTP Request	26	✓	390435	162	24	1	▼

Scroll automatically? Child samples? No of Samples 100 Latest Sample 25 Average 34 Deviation 16

Berdasarkan result tersebut, waktu rata-rata yang dibutuhkan untuk mengakses url tersebut adalah 34. Saat mengakses url tersebut kita perlu mengambil penduduk berdasarkan lokasi, namun query tersebut tetaplah cepat karena sudah diberikan index pada setiap atribut yang digunakan sebagai *join condition* yaitu atribut id_keluarga pada table penduduk, atribut id_kelurahan pada table keluarga, atribut id_kecamatan pada table kelurahan, dan atribut id_kota pada table kecamatan.

Url : <http://localhost:8080/keluarga/ubah/3101011010170001>

View Results in Table										
<input type="text" value="Name: View Results in Table"/> <input type="text" value="Comments:"/> <input type="checkbox"/> Write results to file / Read from file <input type="text" value="Filename"/> <input type="button" value="Browse..."/> Log/Display Only: <input type="checkbox"/> Errors <input type="checkbox"/> Successes <input type="button" value="Configure"/>										
Sample #	Start Time	Thread Name	Label	Sample Time...	Status	Bytes	Sent Bytes	Latency	Connect Tim...	
1	19:17:14.259	Thread Grou...	HTTP Request	11	✓	5770	148	10	2	▲
2	19:17:14.359	Thread Grou...	HTTP Request	7	✓	5770	148	7	1	
3	19:17:14.460	Thread Grou...	HTTP Request	20	✓	5770	148	20	0	
4	19:17:14.559	Thread Grou...	HTTP Request	10	✓	5770	148	10	1	
5	19:17:14.659	Thread Grou...	HTTP Request	8	✓	5770	148	7	1	
6	19:17:14.760	Thread Grou...	HTTP Request	10	✓	5770	148	10	0	
7	19:17:14.859	Thread Grou...	HTTP Request	7	✓	5770	148	7	1	
8	19:17:14.959	Thread Grou...	HTTP Request	8	✓	5770	148	8	1	
9	19:17:15.059	Thread Grou...	HTTP Request	7	✓	5770	148	6	1	
10	19:17:15.159	Thread Grou...	HTTP Request	8	✓	5770	148	8	1	
11	19:17:15.259	Thread Grou...	HTTP Request	7	✓	5770	148	7	1	
12	19:17:15.359	Thread Grou...	HTTP Request	7	✓	5770	148	7	1	
13	19:17:15.459	Thread Grou...	HTTP Request	6	✓	5770	148	6	0	
14	19:17:15.559	Thread Grou...	HTTP Request	7	✓	5770	148	7	1	
15	19:17:15.658	Thread Grou...	HTTP Request	8	✓	5770	148	8	0	
16	19:17:15.759	Thread Grou...	HTTP Request	9	✓	5770	148	8	1	
17	19:17:15.859	Thread Grou...	HTTP Request	8	✓	5770	148	8	1	
18	19:17:15.959	Thread Grou...	HTTP Request	8	✓	5770	148	8	1	
19	19:17:16.059	Thread Grou...	HTTP Request	8	✓	5770	148	8	1	
20	19:17:16.159	Thread Grou...	HTTP Request	6	✓	5770	148	6	0	
21	19:17:16.260	Thread Grou...	HTTP Request	19	✓	5770	148	19	0	
22	19:17:16.360	Thread Grou...	HTTP Request	8	✓	5770	148	8	2	
23	19:17:16.460	Thread Grou...	HTTP Request	12	✓	5770	148	12	2	
24	19:17:16.559	Thread Grou...	HTTP Request	17	✓	5770	148	17	1	
25	19:17:16.659	Thread Grou...	HTTP Request	8	✓	5770	148	8	1	
26	19:17:16.762	Thread Grou...	HTTP Request	13	✓	5770	148	13	1	
27	19:17:16.861	Thread Grou...	HTTP Request	12	✓	5770	148	11	1	
28	19:17:16.961	Thread Grou...	HTTP Request	10	✓	5770	148	10	1	
29	19:17:17.061	Thread Grou...	HTTP Request	13	✓	5770	148	13	1	
30	19:17:17.160	Thread Grou...	HTTP Request	10	✓	5770	148	10	2	▼

Scroll automatically? Child samples? No of Samples 100 Latest Sample 6 Average 8 Deviation 4

Berdasarkan result tersebut, waktu rata-rata yang dibutuhkan untuk mengakses url tersebut adalah 8. Lebih cepat dari mengakses detail keluarga, hal ini dikarenakan saat mengakses form update kita tidak perlu mengambil liat anggota keluarga

Url : <http://localhost:8080/penduduk/ubah/3101010101690001>

View Results in Table													
Name: View Results in Table													
Comments:													
Write results to file / Read from file													
Filename <input type="text"/>				<input type="button" value="Browse..."/>		Log/Display Only: <input type="checkbox"/> Errors <input type="checkbox"/> Successes <input type="button" value="Configure"/>							
Sample #	Start Time	Thread Name	Label	Sample Time...	Status	Bytes	Sent Bytes	Latency	Connect Tim...				
1	19:19:38.838	Thread Grou...	HTTP Request	17	✓	4277	148	17	0	▲			
2	19:19:38.938	Thread Grou...	HTTP Request	7	✓	4277	148	7	1	=			
3	19:19:39.038	Thread Grou...	HTTP Request	22	✓	4277	148	22	0	=			
4	19:19:39.139	Thread Grou...	HTTP Request	23	✓	4277	148	22	0	=			
5	19:19:39.239	Thread Grou...	HTTP Request	6	✓	4277	148	6	0	▼			
6	19:19:39.339	Thread Grou...	HTTP Request	9	✓	4277	148	8	0	=			
7	19:19:39.440	Thread Grou...	HTTP Request	6	✓	4277	148	6	0	=			
8	19:19:39.540	Thread Grou...	HTTP Request	8	✓	4277	148	8	1	=			
9	19:19:39.641	Thread Grou...	HTTP Request	6	✓	4277	148	6	0	=			
10	19:19:39.741	Thread Grou...	HTTP Request	7	✓	4277	148	7	0	=			
11	19:19:39.841	Thread Grou...	HTTP Request	6	✓	4277	148	6	0	=			
12	19:19:39.941	Thread Grou...	HTTP Request	6	✓	4277	148	6	0	=			
13	19:19:40.041	Thread Grou...	HTTP Request	8	✓	4277	148	8	0	=			
14	19:19:40.141	Thread Grou...	HTTP Request	10	✓	4277	148	10	1	=			
15	19:19:40.241	Thread Grou...	HTTP Request	8	✓	4277	148	8	1	=			
16	19:19:40.341	Thread Grou...	HTTP Request	6	✓	4277	148	6	1	=			
17	19:19:40.442	Thread Grou...	HTTP Request	19	✓	4277	148	18	1	=			
18	19:19:40.545	Thread Grou...	HTTP Request	6	✓	4277	148	6	0	=			
19	19:19:40.646	Thread Grou...	HTTP Request	15	✓	4277	148	15	1	=			
20	19:19:40.745	Thread Grou...	HTTP Request	9	✓	4277	148	9	1	=			
21	19:19:40.845	Thread Grou...	HTTP Request	8	✓	4277	148	8	0	=			
22	19:19:40.945	Thread Grou...	HTTP Request	6	✓	4277	148	6	0	=			
23	19:19:41.043	Thread Grou...	HTTP Request	6	✓	4277	148	6	1	=			
24	19:19:41.144	Thread Grou...	HTTP Request	11	✓	4277	148	11	1	=			
25	19:19:41.245	Thread Grou...	HTTP Request	10	✓	4277	148	10	1	=			
26	19:19:41.344	Thread Grou...	HTTP Request	10	✓	4277	148	10	1	=			
27	19:19:41.445	Thread Grou...	HTTP Request	8	✓	4277	148	8	0	=			
28	19:19:41.545	Thread Grou...	HTTP Request	7	✓	4277	148	7	0	=			
29	19:19:41.645	Thread Grou...	HTTP Request	5	✓	4277	148	5	0	=			
30	19:19:41.744	Thread Grou...	HTTP Request	6	✓	4277	148	6	0	▼			

Berdasarkan result tersebut, waktu rata-rata yang dibutuhkan untuk mengakses url tersebut adalah 7. Waktu tersebut sama dengan mengakses halaman detail penduduk karena query yang terdapat di dalam pengaksesan url tersebut sama yaitu mengkases data dari penduduk itu pada table penduduk beserta detail domisilinya.