

Tugas 1

Nama : Yosua Bisma Putrapratama

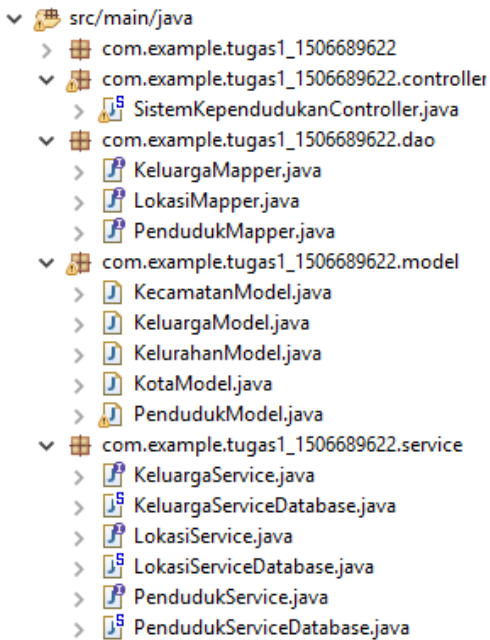
NPM : 1506689622

Kelas : APAP - B

SISTEM KEPENDUDUKAN ONLINE DKI JAKARTA

Pengembangan Fitur

Sebelum mengerjakan dan mengembangkan fitur-fitur, terlebih dahulu membuat dan mempersiapkan *package Model, Mapper, Service, dan Controller*.



- *Model* yang dibuat adalah PendudukModel.java, KeluargaModel.java, KelurahanModel.java, KecamatanModel.java, dan KotaModel.java
- *Mapper* yang dibuat adalah PendudukMapper.java, KeluargaMapper.java, dan LokasiMapper.java. Untuk tindakan yang dilakukan terhadap kelurahan, kecamatan, dan kota akan ditangani pada LokasiMapper.java
- *Service* terdiri dari *class interface* (PendudukService.java, KeluargaService.java, LokasiService.java) dan *interface* tersebut diimplementasikan di *class* PendudukServiceDatabase.java, dan LokasiServiceDatabase.java.
- *Controller* yang dibuat adalah di *class* SistemKependudukanController.java.
- Nantinya juga diperlukan membuat *view* berupa *file* HTML untuk menampilkan hasil.

Tugas 1

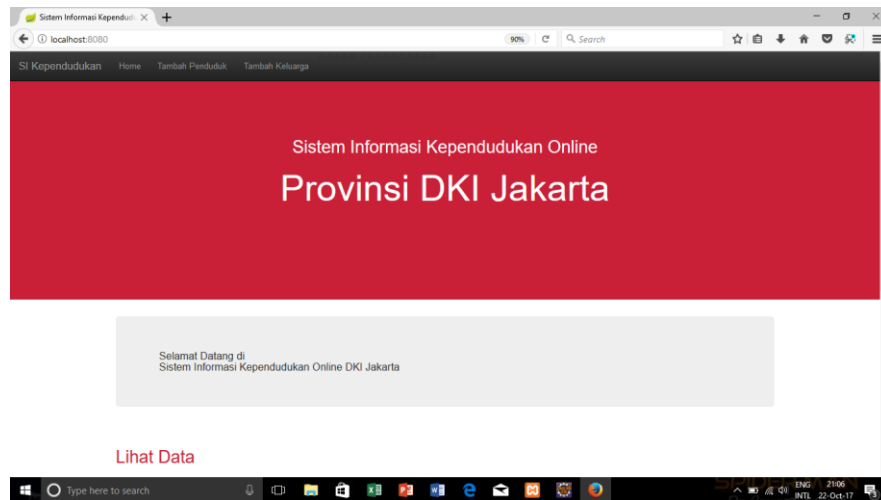
Nama : Yosua Bisma Putrapratama

NPM : 1506689622

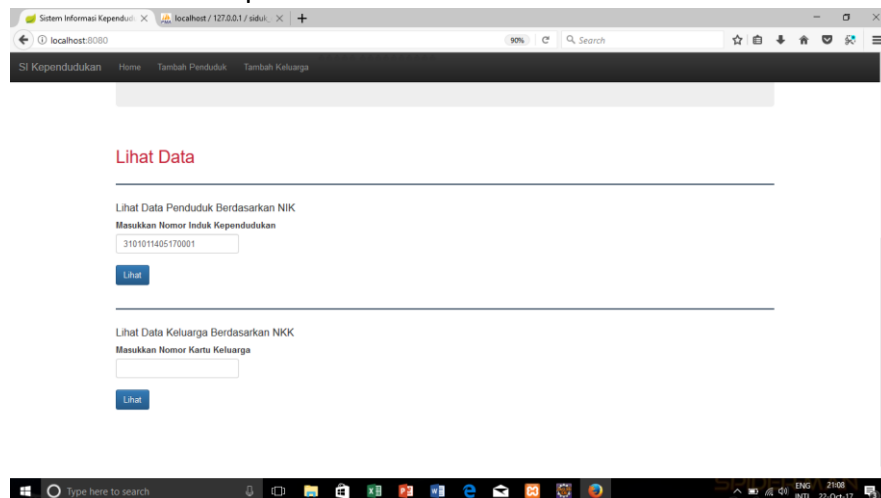
Kelas : APAP - B

1. Fitur Menampilkan Data Penduduk Berdasarkan NIK

Menampilkan data penduduk dengan memasukkan NIK ke dalam *form input*.



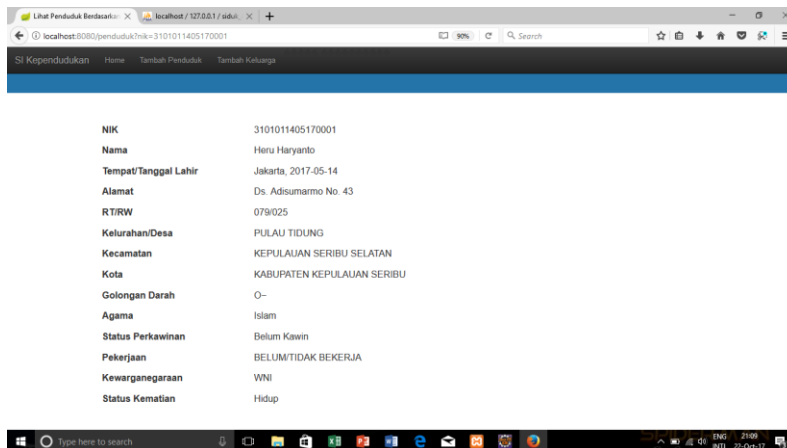
Memasukkan NIK penduduk



Tugas 1

Nama : Yosua Bisma Putrapratama
NPM : 1506689622
Kelas : APAP - B

Setelah di klik tombol “Lihat” akan ditampilkan tampilan seperti berikut :



Dalam membuat fitur pertama ini, dilakukan seperti berikut :

- Pada *class* PendudukMapper.java

```
@Select("select * from penduduk where nik = #{nik}")
@Results(value = {
    @Result(property="nik", column="nik"),
    @Result(property="nama", column="nama"),
    @Result(property="tempatLahir", column="tempat_lahir"),
    @Result(property="tanggalLahir", column="tanggal_lahir"),
    @Result(property="jenisKelamin", column="jenis_kelamin"),
    @Result(property="isWni", column="is_wni"),
    @Result(property="idKeluarga", column="id_keluarga"),
    @Result(property="agama", column="agama"),
    @Result(property="pekerjaan", column="pekerjaan"),
    @Result(property="statusPerkawinan", column="status_perkawinan"),
    @Result(property="statusDalamKeluarga", column="status_dalam_keluarga"),
    @Result(property="golonganDarah", column="golongan_darah"),
    @Result(property="isWafat", column="is_wafat")
})
PendudukModel selectPenduduk(@Param("nik") String nik);
```

- Pada *class* PendudukService.java
PendudukModel selectPenduduk(String nik);

- Pada *class* PendudukServiceDatabase.java

```
@Autowired
private PendudukMapper pendudukMapper;

@Override
public PendudukModel selectPenduduk(String nik) {
    log.info ("select penduduk with nik {}", nik);
    return pendudukMapper.selectPenduduk(nik);
}
```

Tugas 1

Nama : Yosua Bisma Putrapratama

NPM : 1506689622

Kelas : APAP - B

- Pada *class* SistemKependudukanController.java

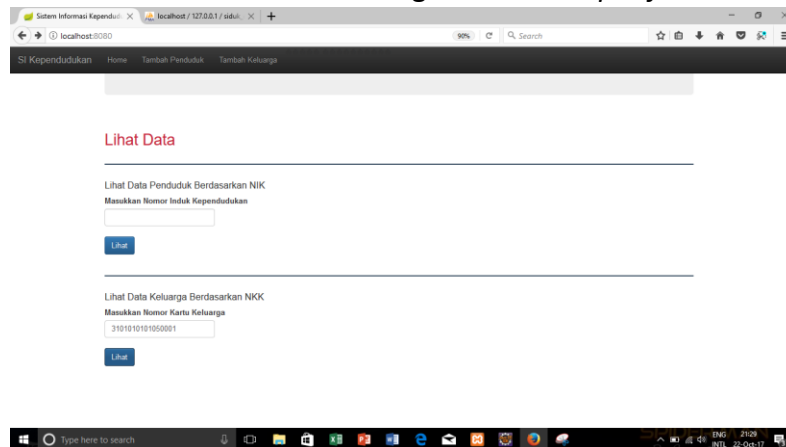
```
//Fitur Nomor 1
@RequestMapping("/penduduk")
public String view(Model model, @RequestParam(value = "nik", required=false) String nik) {
    PendudukModel penduduk = pendudukDAO.selectPenduduk(nik);
    int idKeluarga = penduduk.getIdKeluarga();
    KeluargaModel keluarga = keluargaDAO.selectKeluarga(idKeluarga);
    int idKelurahan = keluarga.getIdKelurahan();
    KelurahanModel kelurahan = lokasiDAO.selectKelurahan(idKelurahan);
    int idKecamatan = kelurahan.getIdKecamatan();
    KecamatanModel kecamatan = lokasiDAO.selectKecamatan(idKecamatan);
    int idKota = kecamatan.getIdKota();
    KotaModel kota = lokasiDAO.selectKota(idKota);
    if(penduduk == null) {
        //return "message-page";
    }
    model.addAttribute("penduduk", penduduk);
    model.addAttribute("keluarga", keluarga);
    model.addAttribute("kelurahan", kelurahan);
    model.addAttribute("kecamatan", kecamatan);
    model.addAttribute("kota", kota);
    return "view-penduduk";
}
```

Jadi *query* untuk fitur pertama diletakan di PendudukMapper yang nantinya akan dipanggil di PendudukServiceDatabase kemudian dijalankan dan dihubungkan ke *view* oleh SistemKependudukanController. *Query* tersebut digunakan untuk memilih penduduk berdasarkan NIK.

2. Fitur Menampilkan Data Keluarga Beserta Daftar Anggotanya Berdasarkan Nomor KK

Menampilkan data keluarga beserta anggota keluarga yang mempunyai idKeluarga yang sama. Id keluarga ini terdapat pada *class* PendudukModel yang nantinya akan dicari mana saja penduduk yang mempunyai id keluarga sesuai dengan nomor KK keluarga yang dimasukkan.

Memasukkan Nomor KK keluarga ke dalam *input form*



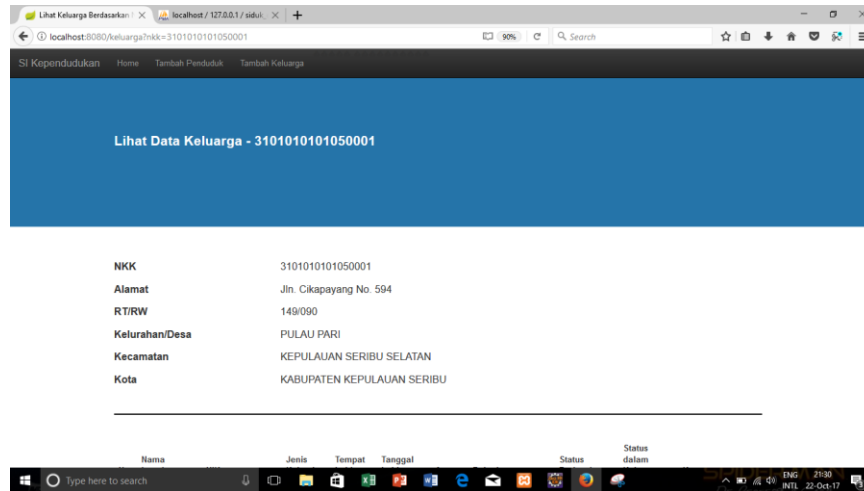
Tugas 1

Nama : Yosua Bisma Putrapratama

NPM : 1506689622

Kelas : APAP - B

Setelah klik tombol “Lihat” maka akan ditampilkan :



Penduduk yang mempunyai idKeluarga sesuai dengan Nomor KK yang dimasukkan akan ditampilkan dalam bentuk tabel

The screenshot shows a web browser window with the URL `localhost:8080/keluarga?nik=3101010101050001`. The page displays a table with the following data:

No.	Nama Lengkap	NIK	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Agama	Pekerjaan	Status Perkawinan	Status dalam Keluarga	Kewarganegaraan
1	Taswir Rajata	3101010303990001	Pria	Jakarta	1989-03-03	Islam	KARYAWAN HONORER	Kawin	Kepala Keluarga	WNI
2	Mulyono Hutagalung S.Gz	3101011408150001	Pria	Jakarta	2015-08-14	Islam	BELUM/TIDAK BEKERJA	Belum Kawin	Anak	WNI
3	Yuliana Wahyuni	3101016312160001	Wanita	Jakarta	2016-12-23	Islam	BELUM/TIDAK BEKERJA	Belum Kawin	Anak	WNI
4	Iffa Novitasari	3101014712830001	Wanita	Jakarta	1983-12-07	Islam	PETANI/PEKEBUN	Kawin	Istri	WNI
5	Arief Bawasadan	3101012362970001	Pria	Bandung	1997-02-23	Katolik	BELUM/TIDAK BEKERJA	Belum Kawin	Anak Ya	WNA
6	Bajuki Cahya Purnama	3101012362970002	Pria	Jakarta	1997-02-23	Kristen	BELUM/TIDAK BEKERJA	Kawin	Anak	WNI
7	Melody	3101016003970001	Wanita	Tokyo	1997-03-20	Kristen	DANCER AJA BUNG	Belum Kawin	Anak Mana Baru	WNI
8	Penduduk Baru Aje	3101016303970001	Wanita	Jakarta	1997-03-23	Buddha	BELUM/TIDAK BEKERJA	Kawin	Anak	WNI
9	Baru lagi 2	3101016303970002	Wanita	Bandung	1997-03-23	Islam	BELUM/TIDAK BEKERJA	Kawin	Anak	WNI

Tugas 1

Nama : Yosua Bisma Putrapratama

NPM : 1506689622

Kelas : APAP - B

Dalam membuat fitur kedua ini, dilakukan seperti berikut :

- Pada *class* KeluargaMapper.java

```
@Select("select * from keluarga where id = #{id}")
@Results(value = {
    @Result(property="nomorKk", column="nomor_kk"),
    @Result(property="alamat", column="alamat"),
    @Result(property="rt", column="rt"),
    @Result(property="rw", column="rw"),
    @Result(property="idKelurahan", column="id_kelurahan"),
    @Result(property="isTidakBerlaku", column="is_tidak_berlaku"),
    @Result(property="anggotaKeluarga", column="id",
        javaType = List.class,
        many=@Many(select="selectAnggotaKeluarga"))
})
KeluargaModel selectKeluarga(@Param("id") int id);

@Select("select id from keluarga where nomor_kk = #{nkk}")
int selectIdKeluarga(@Param("nkk") String nkk);

@Select("select * from penduduk where id_keluarga = #{id}")
@Results(value = {
    @Result(property="nik", column="nik"),
    @Result(property="nama", column="nama"),
    @Result(property="tempatLahir", column="tempat_lahir"),
    @Result(property="tanggalLahir", column="tanggal_lahir"),
    @Result(property="jenisKelamin", column="jenis_kelamin"),
    @Result(property="isWni", column="is_wni"),
    @Result(property="idKeluarga", column="id_keluarga"),
    @Result(property="agama", column="agama"),
    @Result(property="pekerjaan", column="pekerjaan"),
    @Result(property="statusPerkawinan", column="status_perkawinan"),
    @Result(property="statusDalamKeluarga", column="status_dalam_keluarga"),
    @Result(property="golonganDarah", column="golongan_darah"),
    @Result(property="isWafat", column="is_wafat")
})
List<PendudukModel> selectAnggotaKeluarga(@Param("id") int id);
```

- Pada *class* KeluargaService.java

```
KeluargaModel selectKeluarga(int id);

int selectIdKeluarga(String nkk);
```

- Pada *class* KeluargaServiceDatabase.java

```
@Autowired
private KeluargaMapper keluargaMapper;

@Override
public KeluargaModel selectKeluarga(int id) {
    Log.info ("select keluarga with id {}", id);
    return keluargaMapper.selectKeluarga(id);
}

@Override
public int selectIdKeluarga(String nkk) {
    Log.info ("select id keluarga with nkk {}", nkk);
    return keluargaMapper.selectIdKeluarga(nkk);
}
```

Tugas 1

Nama : Yosua Bisma Putrapratama

NPM : 1506689622

Kelas : APAP - B

- Pada *class* SistemKependudukanController.java

```
//Fitur Nomor 2
@RequestMapping("/keluarga")
public String viewKeluarga(Model model, @RequestParam(value = "nkk", required=false) String nkk) {
    int idkeluarga = keluargaDAO.selectIdKeluarga(nkk);
    KeluargaModel keluarga = keluargaDAO.selectKeluarga(idKeluarga);
    int idKelurahan = keluarga.getIdKelurahan();
    KelurahanModel kelurahan = lokasiDAO.selectKelurahan(idKelurahan);
    int idKecamatan = kelurahan.getIdKecamatan();
    KecamatanModel kecamatan = lokasiDAO.selectKecamatan(idKecamatan);
    int idKota = kecamatan.getIdKota();
    KotaModel kota = lokasiDAO.selectKota(idKota);
    if(keluarga == null) {
        //return "message-page";
    }
    model.addAttribute("keluarga", keluarga);
    model.addAttribute("kelurahan", kelurahan);
    model.addAttribute("kecamatan", kecamatan);
    model.addAttribute("kota", kota);
    return "view-keluarga";
}
```

Di fitur kedua terlebih dahulu mencari id keluarga dari nkk, kemudian mencari keluarga yang sesuai dengan nkk yang diinput. Setelah itu didapatkan id kelurahan dari KeluargaModel yang didapat, idKelurahan tersebut digunakan untuk mencari KelurahanModel. Nantinya di KelurahanModel ini akan didapatkan idKecamatan untuk mencari KecamatanModel. Dari KecamatanModel akan didapat idKota untuk mencari KotaModel. KelurahanModel, KecamatanModel, dan KotaModel akan digunakan dan diambil namanya untuk ditampilkan di *view*.

3. Fitur Menambahkan Penduduk Baru Sebagai Anggota Keluarga Baru

Klik “Tambah Penduduk” pada *navigation bar*

Tambah Penduduk

Masukkan Nama

Tempat Lahir

Tanggal Lahir
 Format YYYY-MM-DD, Contoh: 1945-08-17 Untuk Kelahiran 17 Agustus 1945

Jenis Kelamin

Golongan Darah

Agama

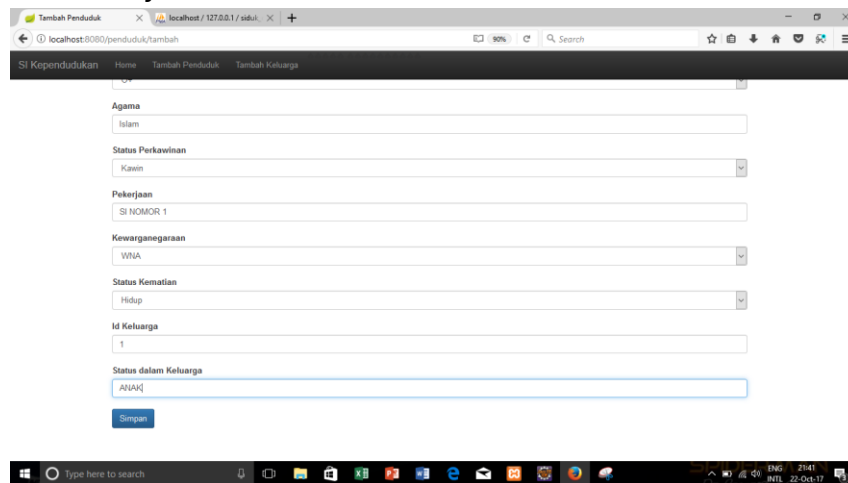
Status Perkawinan

Pekerjaan

Tugas 1

Nama : Yosua Bisma Putrapratama
NPM : 1506689622
Kelas : APAP - B

Kemudian *form* tersebut diisi.

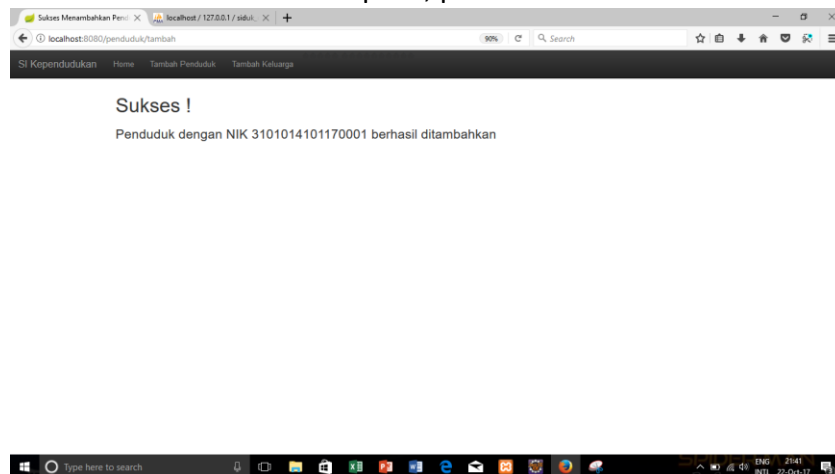


The screenshot shows a web browser window with the URL `localhost:8080/penduduk/tambah`. The page has a navigation bar with links: `SI Kependudukan`, `Home`, `Tambah Penduduk`, and `Tambah Keluarga`. The form contains the following fields:

- Agama:
- Status Perkawinan:
- Pekerjaan:
- Kewarganegaraan:
- Status Kematian:
- Id Keluarga:
- Status dalam Keluarga:

A `Simpan` button is located at the bottom of the form.

Kemudian klik tombol “Simpan”, penduduk berhasil ditambahkan



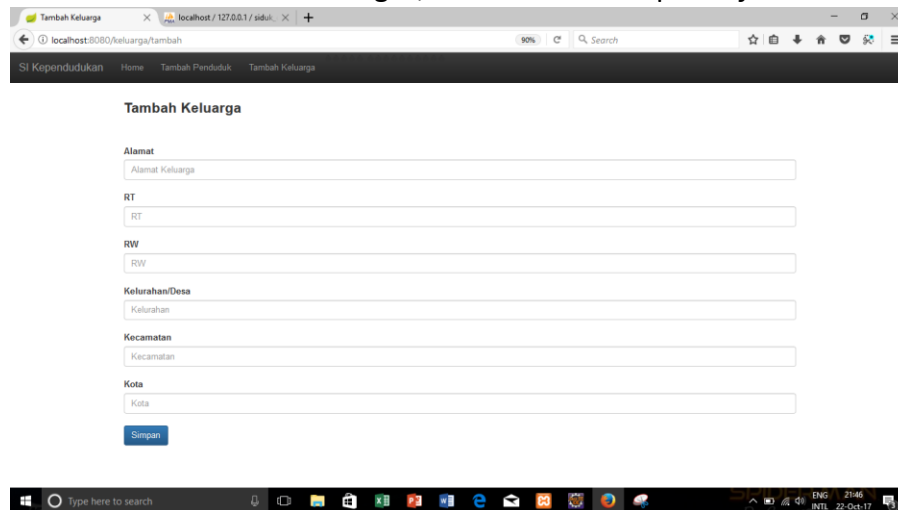
Pada pengerjaan fitur ketiga ini, dibuat terlebih dahulu object `PendudukModel` di *controller* setelah itu untuk mendapatkan dan *generate* NIK dibutuhkan informasi dari lokasi dan tanggal lahir. Dalam hal ini saya mempartisi menjadi dua method. Method pertama untuk mendapatkan 12 angka didepan kemudian method final NIK nya dengan memperhatikan jumlah penduduk yang sudah ada untuk menghasilkan 4 angka dibelakangnya. Jika dalam database *string* 12 digit pertama sudah ada maka otomatis untuk menghitung 4 digit terakhir ditambahkan 1 dari jumlah yang sudah ada.

Tugas 1

Nama : Yosua Bisma Putrapratama
NPM : 1506689622
Kelas : APAP - B

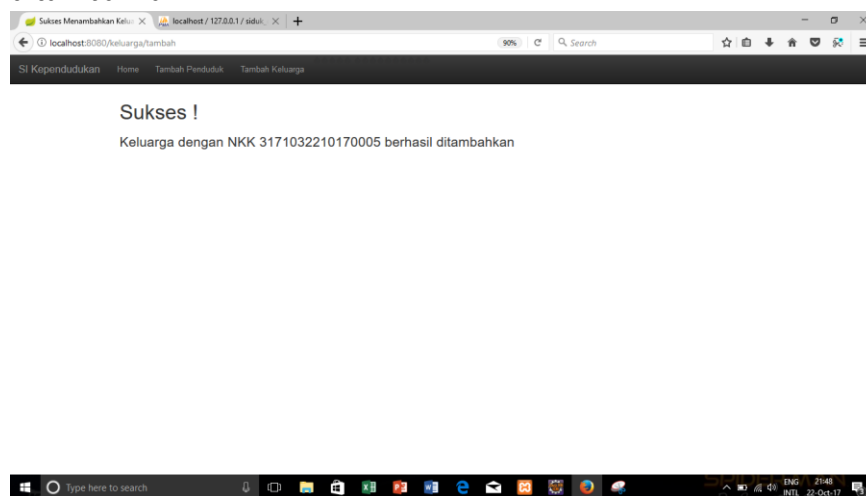
4. Fitur Menambahkan Keluarga Baru

Klik tombol “Tambah Keluarga”, maka akan ditampilkan *form*



The screenshot shows a web browser window with the title 'Tambah Keluarga'. The address bar shows 'localhost:8080/keluarga/tambah'. The page has a navigation bar with 'SI Kependudukan', 'Home', 'Tambah Penduduk', and 'Tambah Keluarga'. The main content area is titled 'Tambah Keluarga' and contains a form with the following fields: 'Alamat Keluarga', 'RT', 'RW', 'Kelurahan/Desa', 'Kecamatan', and 'Kota'. Each field has a small dropdown arrow on the right. Below the fields is a blue button labeled 'Simpan'.

Setelah klik tombol “Simpan” maka akan menampilkan informasi keluarga sukses ditambahkan



The screenshot shows the same web browser window after clicking the 'Simpan' button. The page now displays a success message: 'Sukses !' followed by 'Keluarga dengan NKK 3171032210170005 berhasil ditambahkan'.

Jika *input* masih bernilai *null* maka *view* form-add-keluarga akan ditampilkan. Setelah itu membuat *object* KeluargaModel. Informasi yang dibutuhkan untuk *generate* NKK dapat dicari dari 6 digit pertama yang dimiliki oleh *nomor kelurahan* dari KelurahanModel yang dipilih dengan informasi nama kelurahan. 6 digit tersebut kemudian dimasukkan kedalam *method* untuk *generate* NKK sementara. NKK sementara akan menambahkan 6 digit tersebut dengan tanggal pembuatan NKK. Nantinya dari NKK sementara akan dicari ada berapa keluarga yang mempunyai nomor KK sejenis dengan NKK sementara yang berhasil kita *generate* dengan memanfaatkan like 'NkkSementara%' pada *query*.

Tugas 1

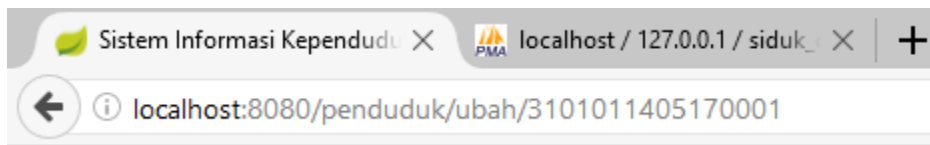
Nama : Yosua Bisma Putrapratama
NPM : 1506689622
Kelas : APAP - B

Nantinya kita dapat mengisi 4 digit dibelakangnya dengan jumlah yang didapatkan dan ditambahkan satu. Selanjutnya kita dapat menambahkan *object* KeluargaModel ke dalam *database*.

5. Fitur Mengubah Data Penduduk

Diakses pada *browser* dengan `/penduduk/ubah/{nik}`.

Misal kita ingin mengubah data Penduduk dengan NIK = 3101011405170001, maka yang dituliskan pada *address bar* di *browser* adalah



Kemudian akan muncul tampilan informasi penduduk tersebut

A screenshot of a web application interface for updating a resident's information. The form is titled 'Ubah Penduduk' and contains several input fields and dropdown menus. The fields are labeled: 'Masukkan Nama' (with the value 'Heru Haryanto'), 'Tempat Lahir' (with the value 'Jakarta'), 'Tanggal Lahir' (with the value '2017-05-14'), 'Jenis Kelamin' (with the value 'Pria'), 'Golongan Darah' (with the value 'A+'), 'Agama' (with the value 'Islam'), 'Status Perkawinan' (with the value 'Belum Kawin'), and 'Pekerjaan'. The form is displayed on a web page with a navigation bar at the top containing 'SI Kependudukan', 'Home', 'Tambah Penduduk', and 'Tambah Keluarga'. The browser's address bar shows the URL 'localhost:8080/penduduk/ubah/3101011405170001'.

Jika dilakukan pengubahan terhadap tanggal lahir atau id keluarga maka bisa berpengaruh terhadap NIK penduduk tersebut.

Untuk melakukan pengubahan terhadap penduduk, akan berpengaruh juga nantinya pada perubahan NIK penduduk dari lama ke NIK yang baru sesuai informasi baru yang dimasukkan. Jadi, perlu didapatkan dulu NIK lama nantinya akan dicek apakah NIK lama sama dengan yang baru. Jika tidak sama maka akan dicari pula jumlah penduduk yang sudah memiliki 12 digit pertama nik yang baru. Kemudian jumlah tersebut akan digunakan untuk mengisi 4 digit terakhir dari NIK yang baru dengan menambahkan 1, karena datanya bertambah 1.

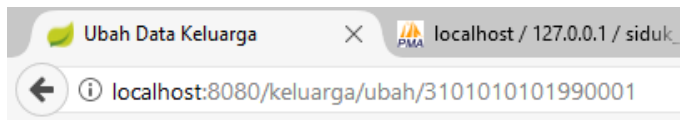
Tugas 1

Nama : Yosua Bisma Putrapratama
NPM : 1506689622
Kelas : APAP - B

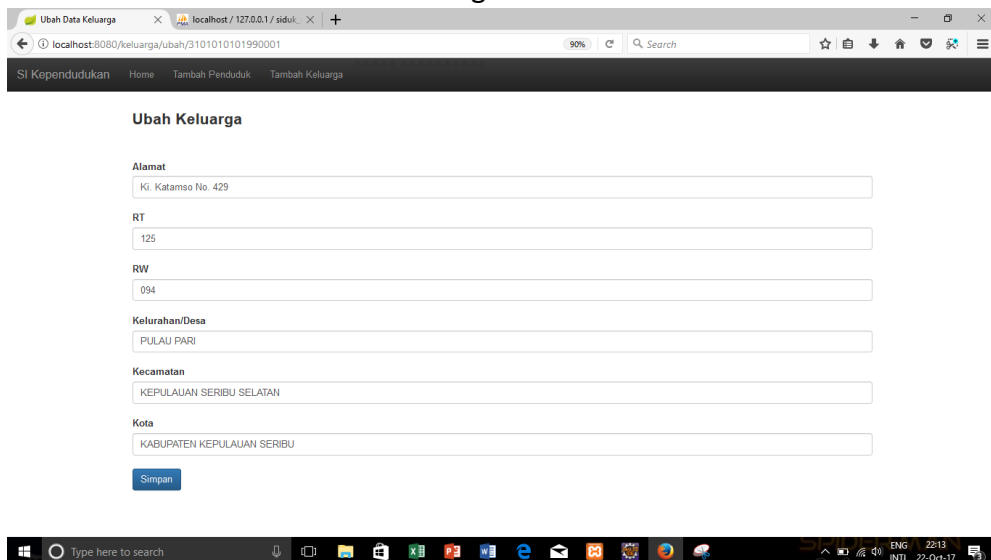
6. Fitur Mengubah Data Keluarga

Diakses pada *browser* dengan `/keluarga/ubah/{nkk}`.

Misal kita ingin mengubah data Keluarga dengan NKK = 3101010101990001, maka yang dituliskan pada *address bar* di *browser* adalah



Maka akan muncul informasi keluarga tersebut



Sama halnya dengan Penduduk, Nomor KK pada keluarga juga dapat berpengaruh dari informasi yang diubah. Tentunya nanti perlu dibandingkan antara Nomor KK lama dengan Nomor KK baru yang di-*generate*. Perbandingan dilakukan hanya terhadap 12 digit pertama karena berkaitan dengan domisili dan tanggal pembuatan KK. Nantinya jika ada keluarga yang memiliki 12 digit nkk yang sama akan dihitung jumlahnya untuk mengisi 4 digit terakhir pada NKK yang baru.

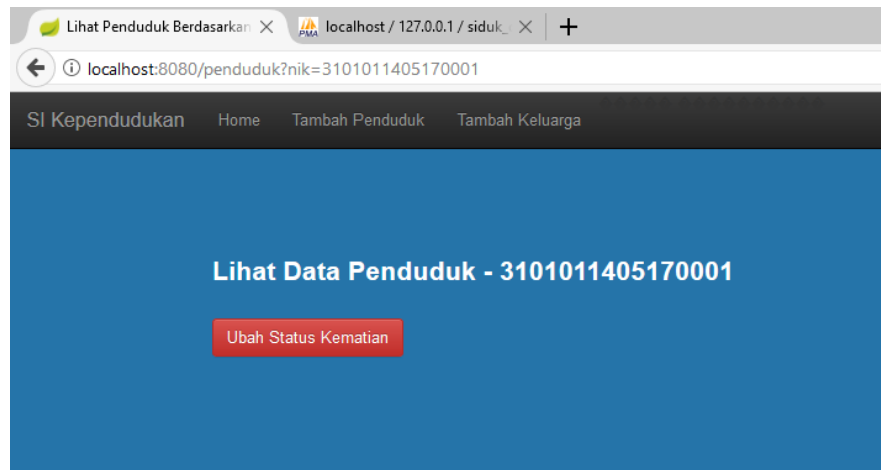
Tugas 1

Nama : Yosua Bisma Putrapratama
NPM : 1506689622
Kelas : APAP - B

7. Fitur Mengubah Status Kematian Penduduk

Ketika kita melakukan *view* terhadap penduduk dengan NIK tertentu, maka terdapat tombol untuk mengubah status kematian penduduk tersebut.

Contoh mengakses penduduk dengan NIK = 3101011405170001



Ketika tombol “Ubah Status Kematian” diklik, maka kolom *is_wafat* pada *database* menjadi 1.

Prosesnya ketika didapatkan *object* penduduk berdasarkan NIK yang diinput, maka menggunakan *method setter* untuk mengganti atribut *isWafat* menjadi 1. Kemudian *object* penduduk tersebut dimasukkan ke dalam *update*. Pada proses ini tidak berpengaruh terhadap NIK penduduk tersebut seperti pada fitur ubah, tetapi hanya mengubah *value* pada *attribute is_Wafat* di *database*.

8. Fitur Menampilkan Data Penduduk Berdasarkan Kota/Kabupaten, Kecamatan, dan Kelurahan Tertentu

Dalam fitur kedelapan ini, kita dapat memilih kriteria tertentu untuk menampilkan penduduk berdasarkan domisili tertentu berkaitan dengan Kota, Kecamatan, dan Kelurahan. Pilihan kecamatan yang ditampilkan dipengaruhi oleh kota yang dipilih sebelumnya, kemudian hasil kecamatan yang dipilih akan berpengaruh terhadap pilihan kelurahan yang ditampilkan. Nantinya informasi / *value* ini dapat digunakan untuk mencari data penduduk.

Tugas 1

Nama : Yosua Bisma Putrapratama

NPM : 1506689622

Kelas : APAP - B

Optimisasi Database

Optimisasi database dilakukan dengan *create index* untuk Penduduk, Keluarga, Kelurahan, Kecamatan, dan Kota yang ada di *database*.

- Berikut ini optimisasi pada penduduk dengan melakukan *indexing* terhadap kolom yang terkait dan banyak digunakan yaitu *id* dan *nik*
`CREATE INDEX idx_penduduk`
`ON penduduk (id, nik);`
- Berikut ini optimisasi pada keluarga dengan melakukan *indexing* terhadap kolom yang terkait dan banyak digunakan yaitu *id* dan *nomor_kk*.
`CREATE INDEX idx_keluarga`
`ON keluarga (id, nomor_kk);`
- Berikut ini optimisasi pada kelurahan dengan melakukan *indexing* terhadap kolom yang terkait dan banyak digunakan yaitu *id*.
`CREATE INDEX idx_kelurahan`
`ON kelurahan (id);`
- Berikut ini optimisasi pada kecamatan dengan melakukan *indexing* terhadap kolom yang terkait dan banyak digunakan yaitu *id*.
`CREATE INDEX idx_kecamatan`
`ON kecamatan (id);`
- Berikut ini optimisasi pada kota dengan melakukan *indexing* terhadap kolom yang terkait dan banyak digunakan yaitu *id*.
`CREATE INDEX idx_kota`
`ON kota (id);`

Dengan CREATE INDEX, proses pencarian data pada query SELECT menjadi jauh lebih cepat. Pembuatan INDEX bertujuan untuk mempercepat proses pencarian data pada kolom di tabel tertentu. Pemanfaatan INDEX ini mampu mempercepat sample time (selisih waktu mengirim request sampai mendapatkan respons), dari rata-rata sample time mencapai ribuan milisecond menjadi hanya belasan milisecond.

Tugas 1

Nama : Yosua Bisma Putrapratama

NPM : 1506689622

Kelas : APAP - B

Stress Testing

Dengan melakukan *stress testing* untuk aplikasi yang dibuat diperoleh hasil seperti berikut :

Sebelum optimisasi Sample Time bernilai lebih besar dibandingkan dengan Sample Time yang sudah dilakukan optimisasi dengan *indexing*.

Berikut contoh untuk *path* /penduduk?nik=3101011405170001

Thread Group

Name: Thread Group

Comments:
Action to be taken after a Sampler error

☒ Continue ☐ Start Next Thread Loop ☐ Stop Thread ☐ Stop Test ☐ Stop Test Now

Thread Properties

Number of Threads (users): 100

Ramp-Up Period (in seconds): 10

Loop Count: ☐ Forever 1

☐ Delay Thread creation until needed

☐ Scheduler

Berikut ini hasil sebelum optimisasi

Apache JMeter (3.2-r1792140)

File Edit Search Run Options Help

00:00:34 0 0 / 100

Test Plan
Thread Group
HTTP Request
View Results in Table
WorkBench

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename ☐ Log Display Only: ☐ Errors ☐ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time...
1	23.35.27.057	Thread Group 1-2	HTTP Request	10221	✓	4535	545	10221	1
2	23.35.28.981	Thread Group 1-3	HTTP Request	10138	✓	4535	545	10316	3
3	23.35.27.260	Thread Group 1-4	HTTP Request	28955	✓	4535	545	28955	1
4	23.35.28.053	Thread Group 1...	HTTP Request	26862	✓	4535	545	26862	2
5	23.35.28.253	Thread Group 1...	HTTP Request	26664	✓	4535	545	26663	0
6	23.35.27.964	Thread Group 1...	HTTP Request	27954	✓	4535	545	27953	1
7	23.35.27.710	Thread Group 1-9	HTTP Request	28168	✓	4535	545	28165	1
8	23.35.28.353	Thread Group 1...	HTTP Request	27574	✓	4535	545	27574	0
9	23.35.27.483	Thread Group 1-8	HTTP Request	28465	✓	4535	545	28465	1
10	23.35.28.653	Thread Group 1...	HTTP Request	27339	✓	4535	545	27339	0
11	23.35.28.053	Thread Group 1...	HTTP Request	27684	✓	4535	545	27683	1
12	23.35.28.653	Thread Group 1...	HTTP Request	27285	✓	4535	545	27285	1
13	23.35.28.453	Thread Group 1...	HTTP Request	27487	✓	4535	545	27487	0
14	23.35.27.166	Thread Group 1-3	HTTP Request	28777	✓	4535	545	28777	1
15	23.35.28.353	Thread Group 1...	HTTP Request	26593	✓	4535	545	26593	0
16	23.35.28.753	Thread Group 1...	HTTP Request	27184	✓	4535	545	27183	1
17	23.35.28.853	Thread Group 1...	HTTP Request	27095	✓	4535	545	27095	1
18	23.35.27.557	Thread Group 1-7	HTTP Request	28397	✓	4535	545	28397	1
19	23.35.27.864	Thread Group 1...	HTTP Request	28165	✓	4535	545	28165	2
20	23.35.27.667	Thread Group 1-8	HTTP Request	28292	✓	4535	545	28292	1
21	23.35.28.253	Thread Group 1...	HTTP Request	27798	✓	4535	545	27798	1
22	23.35.28.154	Thread Group 1...	HTTP Request	27607	✓	4535	545	27607	2
23	23.35.28.153	Thread Group 1...	HTTP Request	26814	✓	4535	545	26814	0
24	23.35.28.453	Thread Group 1...	HTTP Request	26517	✓	4535	545	26517	0
25	23.35.28.953	Thread Group 1...	HTTP Request	27018	✓	4535	545	27018	1
26	23.35.27.354	Thread Group 1-5	HTTP Request	29954	✓	4535	545	29954	0
27	23.35.28.653	Thread Group 1...	HTTP Request	26622	✓	4535	545	26622	1

☐ Scroll automatically? ☐ Child samples? No of Samples: 100 Latent Sample 24954 Average 27192 Deviation 2718

Tugas 1

Nama : Yosua Bisma Putrapratama

NPM : 1506689622

Kelas : APAP - B

Setelah itu dilakukan optimisasi hasilnya berikut ini dan *sample time* nya bisa lebih cepat

Apache JMeter (3.2 r1700748)

File Edit Search Run Options Help

Test Plan
Thread Group
HTTP Request
View Results in Table
Workbench

00:00:28 0 0 / 100

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: Browse Log Display Only: ☐ Errors ☐ Successes Configure

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time...
1	23:39:07.956	Thread Group 1-2	HTTP Request	5890	✓	4535	140	5890	1
2	23:39:07.963	Thread Group 1-1	HTTP Request	5883	✓	4535	140	5883	2
3	23:39:08.055	Thread Group 1-3	HTTP Request	5787	✓	4535	140	5787	1
4	23:39:08.863	Thread Group 1-	HTTP Request	18420	✓	4535	140	18420	0
5	23:39:09.851	Thread Group 1-	HTTP Request	18333	✓	4535	140	18332	0
6	23:39:09.451	Thread Group 1-	HTTP Request	17934	✓	4535	140	17934	1
7	23:39:09.950	Thread Group 1-	HTTP Request	17435	✓	4535	140	17435	1
8	23:39:09.951	Thread Group 1-	HTTP Request	18334	✓	4535	140	18334	1
9	23:39:09.951	Thread Group 1-	HTTP Request	17635	✓	4535	140	17635	1
10	23:39:09.253	Thread Group 1-5	HTTP Request	19135	✓	4535	140	19135	1
11	23:39:09.651	Thread Group 1-	HTTP Request	17537	✓	4535	140	17537	1
12	23:39:09.751	Thread Group 1-	HTTP Request	17637	✓	4535	140	17637	1
13	23:39:08.960	Thread Group 1-8	HTTP Request	18625	✓	4535	140	18624	1
14	23:39:09.351	Thread Group 1-	HTTP Request	22781	✓	4535	140	22781	1
15	23:39:09.351	Thread Group 1-	HTTP Request	22597	✓	4535	140	22596	1
16	23:39:10.650	Thread Group 1-	HTTP Request	21899	✓	4535	140	21899	1
17	23:39:08.753	Thread Group 1-	HTTP Request	23198	✓	4535	140	23198	1
18	23:39:09.651	Thread Group 1-	HTTP Request	22301	✓	4535	140	22301	1
19	23:39:08.650	Thread Group 1-9	HTTP Request	22595	✓	4535	140	22595	1
20	23:39:08.159	Thread Group 1-4	HTTP Request	23799	✓	4535	140	23799	1
21	23:39:08.260	Thread Group 1-6	HTTP Request	22595	✓	4535	140	22595	1
22	23:39:08.656	Thread Group 1-7	HTTP Request	22595	✓	4535	140	22595	1
23	23:39:09.251	Thread Group 1-	HTTP Request	22712	✓	4535	140	22712	1
24	23:39:10.467	Thread Group 1-	HTTP Request	22297	✓	4535	140	22297	0
25	23:39:11.467	Thread Group 1-	HTTP Request	21297	✓	4535	140	21297	1
26	23:39:10.767	Thread Group 1-	HTTP Request	22000	✓	4535	140	22000	0
27	15:16:11.506	Thread Group 1-	HTTP Request	20998	✓	4535	140	20998	2

☐ Scroll automatically? ☐ Child samples? No of Samples 100 Latest Sample 19242 Average 20048 Deviation 2959

1860 2040
NTL 22-Oct-19