

## Writeup Tugas 1 APAP

### Penjelasan

Pada Tugas ini hal pertama yang saya lakukan adalah membuat project Spring baru terlebih dahulu. Setelah itu saya membuat 4 package yaitu `com.example.model`, `com.example.controller`, `com.example.service`, dan `com.example.dao`.

Pada `com.example.model` saya membuat 5 class yaitu `PendudukModel`, `KeluargaModel`, `KelurahanModel`, `KecamatanModel`, dan `KotaModel`. Setiap kelas berisi atribut yang sesuai dengan tabel yang direpresentasikan, akan tetapi saya menambahkan atribut objek keluarga pada `PendudukModel`, objek kelurahan pada `KeluargaModel`, objek kecamatan pada `KelurahanModel`, dan objek kota pada `KecamatanModel`. Hal ini dilakukan agar ketika melakukan pencarian penduduk melalui NIK, alamat yang ditampilkan akan lebih lengkap.

Pada `com.example.service` saya membuat 1 interface yaitu `PendudukService` dan 1 class yaitu `PendudukServiceDatabase` yang mengimplementasi `PendudukService`. Interface dan class ini berfungsi sebagai DAO atau jembatan antara view dan database.

Pada `com.example.controller` saya membuat 1 class yaitu `PendudukController`. Class ini berfungsi memproses data yang diinput user dan menampilkan halaman. Proses seperti pembuatan NIK dan NKK dilakukan di `PendudukController`.

Pada `com.example.dao` saya membuat 1 interface yaitu `PendudukMapper`. Interface ini terhubung dengan class `PendudukServiceDatabase` dimana class inilah yang mengeksekusi setiap kode SQL yang diimplementasi pada interface.

Setelah itu saya menambahkan view index ke resources untuk memastikan apakah aplikasi dapat menampilkan view dengan wajar. Pada `PendudukController` saya membuat method dengan request mapping menuju index.

Percobaan pertama saya adalah membuat koneksi ke database dengan menjalankan program aplikasi. Hasilnya adalah gagal karena compiler tidak mengenali keberadaan package `com.example.dao`. Setelah diinvestigasi lagi, ternyata saya belum melakukan setup terhadap `pom.xml` karena belum ada MySQL dan mybatis di dalam dependencies. Saya mengikuti instruksi tutorial 4 dan memasukkan MySQL dan mybatis ke dalam dependencies. Setelah itu saya melakukan setup di `application.properties` untuk membuat koneksi ke database `siduk_dki` yang telah saya import ke XAMPP.

Saya menginstal `pom.xml` dengan `maven-install`, dan hasilnya berhasil menginstal. Setelah itu saya menyalakan Apache dan MySQL di XAMPP, kemudian menjalankan aplikasi untuk pertama kali. Hasilnya aplikasi berhasil di-compile dan dijalankan.

Fitur pertama yang saya kembangkan adalah menampilkan penduduk berdasarkan NIK dan menampilkan keluarga berdasarkan NKK. Tidak ada kendala selama mengerjakan fitur ini, meskipun awalnya saya ragu apakah penggunaan method SQL yang memanggil secara berantai dapat bekerja.

Fitur selanjutnya adalah penambahan keluarga. Kendala pada fitur ini ada pada proses pembuatan NKK. Awalnya saya mencoba mengembangkan pembuatan NKK di service, namun sepertinya merupakan bad practice jadi saya meletakkan pembuatan NKK di controller. Alhasil, interface

Sukrian Syahnel Putra

1306402066

APAP -B

PendudukService harus dikembangkan agar dapat memanggil method baru di MVC, seperti mengambil NKK dari keluarga lain untuk pengecekan.

Fitur selanjutnya adalah penambahan penduduk. Tidak ada kendala pada fitur ini karena kurang lebih sama dengan pengembangan fitur penambahan keluarga.

Fitur selanjutnya adalah melakukan update terhadap penduduk. Awalnya saya sempat berkeputus dengan proses generate NIK karena banyak tahap yang membingungkan, seperti mengecek kembali NIK karena saya sempat salah ketika melakukan pengecekan apakah NIK tersebut sudah ada atau tidak.

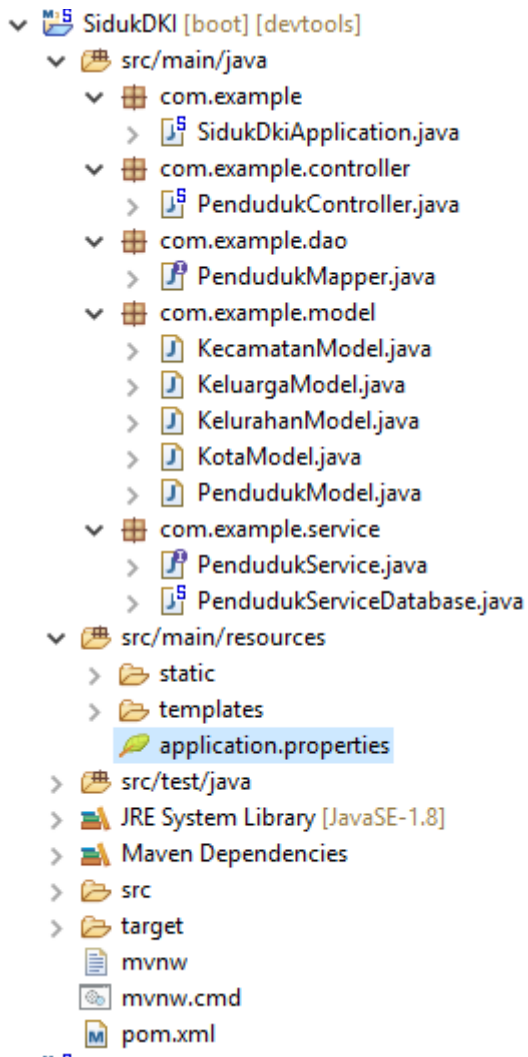
Fitur selanjutnya adalah melakukan update terhadap keluarga. Karena NIK mengalami update bersamaan dengan NKK, maka saya perlu mengupdate daftar anggota keluarga. Saya sempat salah ketika mencoba menggunakan for looping untuk list anggota keluarga, karena tidak terjadi update sama sekali pada saat itu. Untungnya saya bisa mengimplementasi fitur ini.

Fitur selanjutnya adalah melakukan update terhadap status kematian. Fitur ini sempat gagal bekerja karena ketika saya mencoba menonaktifkan penduduk di keluarga dengan satu orang, tidak terjadi perubahan terhadap status berlaku dari keluarga tersebut. Setelah menggunakan variabel tertentu akhirnya saya dapat mengimplementasi fitur ini meskipun sempat kebingungan saat memikirkan redirect yang ternyata cukup simple.

Fitur terakhir yang saya kembangkan adalah pencarian berdasarkan kota, kecamatan, dan kabupaten. Fitur ini dikembangkan dengan menggunakan tiga elemen Boolean untuk pengecekan template agar dapat memilih template yang aktif dan disabled. Selain itu, ketika saya mencoba menggunakan database Siduk DKI, pencarian memakan waktu yang cukup lama. Hal ini dikarenakan saya terlalu banyak memasukkan variabel ke dalam kode SQL yang ada di PendudukMapper. Untungnya fitur ini berhasil diimplementasi dengan baik meskipun memakan waktu cukup lama.

Sukrian Syahnel Putra  
1306402066  
APAP -B  
Struktur Project

Adapun struktur project dari tugas ini adalah sebagai berikut.
































Berikut merupakan file yang ada di src/main/resources/static



















Sukrian Syahnel Putra

1306402066

APAP -B

- ▼  src/main/resources
  - ▼  static
    - ▼  css
      -  bootstrap-grid.css
      -  bootstrap-grid.css.map
      -  bootstrap-grid.min.css
      -  bootstrap-grid.min.css.map
      -  bootstrap-reboot.css
      -  bootstrap-reboot.css.map
      -  bootstrap-reboot.min.css
      -  bootstrap-reboot.min.css.map
      -  bootstrap.css
      -  bootstrap.css.map
      -  bootstrap.min.css
      -  bootstrap.min.css.map
    - ▼  DataTables
      - ▼  DataTables-1.10.16
        - >  css
        - >  images
        - >  js
        -  datatables.css
        -  datatables.js
        -  datatables.min.css
        -  datatables.min.js
      - ▼  js
        -  bootstrap.js
        -  bootstrap.min.js
        -  jquery-3.2.1.min.js
        -  jquery-3.2.1.slim.min.js

Berikut merupakan file di resources/templates

- ▼  templates
  - ▼  fragments
    -  fragment.html
  -  cari-penduduk.html
  -  form-add-keluarga.html
  -  form-add-penduduk.html
  -  form-update-keluarga.html
  -  form-update-penduduk.html
  -  index.html
  -  not-found-keluarga.html
  -  not-found-penduduk.html
  -  success-add-keluarga.html
  -  success-add-penduduk.html
  -  success-update-keluarga.html
  -  success-update-kematian.html
  -  success-update-penduduk.html
  -  view-keluarga.html
  -  view-penduduk.html

Sukrian Syahnel Putra  
1306402066  
APAP -B  
Optimisasi Database

Optimisasi yang saya lakukan terhadap database adalah sebagai berikut:

Melakukan indexing pada tabel keluarga berdasarkan nomor KK, di samping menjadikan id sebagai primary key.

Melakukan indexing pada tabel penduduk berdasarkan NIK, di samping menjadikan id sebagai primary key.

Melakukan indexing pada tabel kelurahan berdasarkan kode kelurahan.

Melakukan indexing pada tabel kecamatan berdasarkan kode kecamatan.

Melakukan indexing pada tabel kota berdasarkan kode kota.

### Stress Testing

Awalnya saya mencoba melakukan testing untuk membuka fitur pencarian berdasarkan kota, kecamatan, dan kelurahan, dengan 50 user dan 10 sekon ramp-up time. Akan tetapi, karena keterbatasan laptop yang saya gunakan yang menyebabkan program berjalan sangat lambat, saya menurunkan jumlah user menjadi 10 dengan ramp-up time yang sama. Fitur yang dibuka pun bukanlah fitur yang cukup berat, yaitu mencari penduduk berdasarkan NIK. Stress Testing diujikan sebanyak 5 kali dengan beban yang sama.

Hasil Stress Testing adalah sebagai berikut.

TutorialAPAP.jmx (C:\Users\User\Desktop\Temporary Recycle Bin\apache-jmeter-3.2\bin\TutorialAPAP.jmx) - Apache JMeter (3.2.1 r1790748)

File Edit Search Run Options Help

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:  Browse... Log/Display Only: ☐ Errors ☐ Successes

Sample #	Start Time	Thread Name	Label	Sample Time	Status	Bytes	Sent Bytes	Latency	Connect Time
1	20:57:45.568	Thread Group 1-5	HTTP Request	10717	Success	6807	147	10702	1
2	20:57:42.574	Thread Group 1-2	HTTP Request	13710	Success	6807	147	13695	1
3	20:57:43.566	Thread Group 1-3	HTTP Request	12717	Success	6807	147	12703	1
4	20:57:48.569	Thread Group 1-6	HTTP Request	9716	Success	6807	147	9701	1
5	20:57:41.569	Thread Group 1-1	HTTP Request	14714	Success	6807	147	14699	2
6	20:57:44.567	Thread Group 1-4	HTTP Request	11717	Success	6807	147	11701	2
7	20:57:48.566	Thread Group 1-8	HTTP Request	23033	Success	6807	147	23032	2
8	20:57:50.567	Thread Group 1-10	HTTP Request	22575	Success	6807	147	22574	1
9	20:57:49.568	Thread Group 1-9	HTTP Request	23580	Success	6807	147	23576	1
10	20:57:47.567	Thread Group 1-7	HTTP Request	25620	Success	6807	147	25620	1

☐ Scroll automatically? ☐ Child samples? No of Samples 10 Latest Sample 25620 Average 16809 Deviation 5827

Hasil Tes Pertama

Sukrian Syahnel Putra

1306402066

APAP -B

TutorialAPAP.jmx (C:\Users\User\Desktop\Temporary Recycle Bin\apache-jmeter-3.2\bin\TutorialAPAP.jmx) - Apache JMeter (3.2.r1790748)

File Edit Search Run Options Help

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:  Browse... Log/Display Only: ☐ Errors ☐ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	21:02:58.977	Thread Group 1-1	HTTP Request	2415	✓	6807	147	2415	2
2	21:02:59.989	Thread Group 1-2	HTTP Request	3695	✓	6807	147	3695	1
3	21:03:00.984	Thread Group 1-3	HTTP Request	4221	✓	6807	147	4221	1
4	21:03:01.984	Thread Group 1-4	HTTP Request	4810	✓	6807	147	4810	1
5	21:03:03.984	Thread Group 1-6	HTTP Request	4665	✓	6807	147	4664	2
6	21:03:02.984	Thread Group 1-5	HTTP Request	5666	✓	6807	147	5664	1
7	21:03:04.969	Thread Group 1-7	HTTP Request	3832	✓	6807	147	3832	1
8	21:03:05.985	Thread Group 1-8	HTTP Request	3860	✓	6807	147	3860	1
9	21:03:06.985	Thread Group 1-9	HTTP Request	3870	✓	6807	147	3870	1
10	21:03:07.986	Thread Group 1-10	HTTP Request	3204	✓	6807	147	3204	1

☐ Scroll automatically? ☐ Child samples? No of Samples: 10 Latest Sample: 3204 Average: 4023 Deviation: 849

21:03 22/10/2017

Hasil Tes Kedua

TutorialAPAP.jmx (C:\Users\User\Desktop\Temporary Recycle Bin\apache-jmeter-3.2\bin\TutorialAPAP.jmx) - Apache JMeter (3.2.r1790748)

File Edit Search Run Options Help

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:  Browse... Log/Display Only: ☐ Errors ☐ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	21:03:40.430	Thread Group 1-1	HTTP Request	2856	✓	6807	147	2855	2
2	21:03:41.453	Thread Group 1-2	HTTP Request	3170	✓	6807	147	3170	1
3	21:03:42.452	Thread Group 1-3	HTTP Request	3505	✓	6807	147	3505	1
4	21:03:43.453	Thread Group 1-4	HTTP Request	3400	✓	6807	147	3400	0
5	21:03:44.452	Thread Group 1-5	HTTP Request	3599	✓	6807	147	3599	1
6	21:03:45.450	Thread Group 1-6	HTTP Request	3214	✓	6807	147	3213	1
7	21:03:46.452	Thread Group 1-7	HTTP Request	3330	✓	6807	147	3329	1
8	21:03:47.451	Thread Group 1-8	HTTP Request	3643	✓	6807	147	3642	2
9	21:03:48.452	Thread Group 1-9	HTTP Request	2984	✓	6807	147	2984	1
10	21:03:49.452	Thread Group 1-10	HTTP Request	3761	✓	6807	147	3761	1

☐ Scroll automatically? ☐ Child samples? No of Samples: 10 Latest Sample: 3761 Average: 3346 Deviation: 277

21:03 22/10/2017

Hasil Tes Ketiga

Sukrian Syahnel Putra

1306402066

APAP -B

The screenshot shows the Apache JMeter 3.2.1 interface. The left sidebar displays a test plan with a thread group containing an HTTP request and a 'View Results in Table' listener. The main window shows the 'View Results in Table' view for the listener named 'View Results in Table'. It includes a table with 10 samples of HTTP requests, their start times, thread names, labels, sample times, statuses, bytes, sent bytes, latencies, and connect times. The status for all samples is 'Success' (green checkmark). The summary statistics at the bottom indicate 10 samples, the latest sample is 4721, the average is 4105, and the deviation is 721.

Sample #	Start Time	Thread Name	Label	Sample Time...	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	21:04:13.007	Thread Group 1-1	HTTP Request	3368	Success	6807	147	3368	2
2	21:04:14.008	Thread Group 1-2	HTTP Request	3580	Success	6807	147	3580	1
3	21:04:15.009	Thread Group 1-3	HTTP Request	3016	Success	6807	147	3016	1
4	21:04:16.008	Thread Group 1-4	HTTP Request	3473	Success	6807	147	3473	1
5	21:04:17.013	Thread Group 1-5	HTTP Request	5132	Success	6807	147	5132	1
6	21:04:18.029	Thread Group 1-6	HTTP Request	4441	Success	6807	147	4441	1
7	21:04:19.013	Thread Group 1-7	HTTP Request	4538	Success	6807	147	4538	1
8	21:04:20.016	Thread Group 1-8	HTTP Request	5065	Success	6807	147	5065	1
9	21:04:22.008	Thread Group 1-10	HTTP Request	3725	Success	6807	147	3724	1
10	21:04:21.012	Thread Group 1-9	HTTP Request	4721	Success	6807	147	4721	1

## Hasil Tes Keempat

The screenshot shows the Apache JMeter 3.2.1 interface. The left sidebar displays a test plan with a thread group containing an HTTP request and a 'View Results in Table' listener. The main window shows the 'View Results in Table' view for the listener named 'View Results in Table'. It includes a table with 10 samples of HTTP requests, their start times, thread names, labels, sample times, statuses, bytes, sent bytes, latencies, and connect times. The status for all samples is 'Success' (green checkmark). The summary statistics at the bottom indicate 10 samples, the latest sample is 618, the average is 2857, and the deviation is 1960.

Sample #	Start Time	Thread Name	Label	Sample Time...	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	21:04:55.340	Thread Group 1-1	HTTP Request	654	Success	6807	147	654	2
2	21:04:56.346	Thread Group 1-2	HTTP Request	2757	Success	6807	147	2757	1
3	21:04:59.330	Thread Group 1-5	HTTP Request	4434	Success	6807	147	4434	1
4	21:05:02.331	Thread Group 1-8	HTTP Request	1433	Success	6807	147	1433	1
5	21:05:00.330	Thread Group 1-6	HTTP Request	3437	Success	6807	147	3437	1
6	21:04:57.330	Thread Group 1-3	HTTP Request	6449	Success	6807	147	6447	1
7	21:04:58.334	Thread Group 1-4	HTTP Request	5451	Success	6807	147	5450	1
8	21:05:01.331	Thread Group 1-7	HTTP Request	2456	Success	6807	147	2455	1
9	21:05:03.332	Thread Group 1-9	HTTP Request	883	Success	6807	147	883	1
10	21:05:04.331	Thread Group 1-10	HTTP Request	618	Success	6807	147	618	1

## Hasil Tes Kelima

Berdasarkan lima hasil tes ini dan tes pertama yang gagal karena keterbatasan laptop, saya menyimpulkan bahwa aplikasi yang saya kembangkan lambat ketika melakukan load awal, akan tetapi menjadi lebih baik dengan pengetesan selanjutnya. Dugaan saya adalah Spring telah mengeset cache aktif secara default sehingga ketika mengetes secara 5 kali performanya menjadi lebih baik. Akan tetapi, bukan berarti aplikasi yang saya kembangkan dapat dibilang reliable karena pada tes pertama yang menggunakan fitur cari penduduk berdasarkan kelurahan, kecamatan, dan kota, tes tersebut mengalami kegagalan karena keterbatasan perangkat. Selain itu, masih banyak kode-kode yang

Sukrian Syahnel Putra

1306402066

APAP -B

menumpuk dan pemanggilan SQL yang tidak perlu di dalam aplikasi saya sehingga dapat saya bilang aplikasi ini tidaklah sempurna meskipun dapat menjalankan setiap fitur yang diminta.