

TUGAS AKHIR

**Pembangunan *Backend* Sistem Informasi *KRS Online*
Berbasis *Microservices***



Disusun Oleh

Danang Kusuma Yudha

14 07 07818

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ATMA JAYA YOGYAKARTA

2019

HALAMAN PENGESAHAN

Tugas Akhir Berjudul
Pembangunan *Backend* Sistem Informasi KRS *Online* Berbasis *Microservices*

Disusun oleh:

Danang Kusuma Yudha
(NIM: 140707818)

Dinyatakan telah memenuhi syarat
Pada Tanggal: Januari 2019

Pembimbing I,



(Irya Wisnubhadra, S.T., M.T.)

Pembimbing II,



(Y. Sigit Purnomo W.P., S.T., M.Kom.)

Tim Penguji:

Penguji I



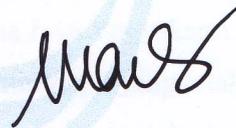
(Irya Wisnubhadra, S.T., M.T.)

Penguji II,



(Patricia Ardanari, S.Si, M.T.)

Penguji III,



(Martinus Maslim, S.T., M.T.)

Yogyakarta, Januari 2019

Universitas Atma Jaya Yogyakarta

Fakultas Teknologi Industri

Dekan,



(Dr. A. Teguh Siswanto, M. Sc.)

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena dengan rahmad dan karunia nya saya dapat menyelesaikan penulisan tugas yang merupakan syarat untuk meraih gelar sarjana teknik di Universitas Atma Jaya Yogyakarta.

Saya menyadari bahwa sesungguhnya apa yang telah dicapai tak lepas dari bantuan orang-orang terdekat maupun dosen dan karyawan di Universitas Atma Jaya Yogyakarta. Oleh karena itu dengan sepenuh ketulusan hati saya ucapkan terimakasih atas dukungan yang diberikan baik itu moral maupun materi yang telah membuat penulis dapat menyelesaikan penulisan Tugas Akhir ini, Oleh sebab itu penulis ingin mempersembahkan rasa terima kasih kepada:

1. Allah Subhanahu Wa Ta'ala yang ajaran nya telah menjadi pendoman hidup saya dalam menghadapi masa-masa sulit dan keputusan.
2. Bapak Dr. A. Teguh Siswanto selaku Dekan Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta
3. Martinus Maslim, S.T., M.T., selaku Ketua Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta.
4. Bapak Irya Wisnubhadra, S.T., M.T., selaku Dosen Pembimbing I yang telah meluangkan waktu dan pikiran untuk memberi bimbingan, petunjuk dan pengarahan kepada penulis sehingga skripsi ini dapat diselesaikan dengan baik.
5. Y. Sigit Purnomo WP., S.T., M.Kom., selaku Dosen Pembimbing II yang telah meluangkan waktu dan pikiran untuk memberi bimbingan, petunjuk dan pengarahan kepada penulis sehingga skripsi ini dapat diselesaikan dengan baik.
6. Seluruh Dosen dan Staf Pengajar Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta yang telah membantu penulis selama masa kuliah di Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta.

7. Orang tua yang selalu mendoakan serta memberikan dorongan, semangat, dan motivasi kepada penulis agar dapat menyelesaikan kuliah dengan baik dan tepat waktu.
8. Seluruh keluarga besar tercinta yang selalu memberikan dorongan dan motivasi untuk penulis selama masa kuliah.
9. Semua teman dan sahabat angkatan 2014 Teknik Informatika Atma Jaya Yogyakarta yang tidak bisa disebutkan satu persatu.
10. Semua orang yang secara tidak langsung memberikan dukungan dan semangat.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna karena keterbatasan waktu dan pengetahuan yang dimiliki penulis. Oleh karena itu segala kritik dan saran yang bersifat membangun sangat diharapkan. Akhir kata, semoga skripsi ini dapat berguna dan bermanfaat bagi semua pihak.

Yogyakarta, ____ Januari 2019

Penulis

DAFTAR ISI

HALAMAN PENGESAHAN.....	i
KATA PENGANTAR	ii
DAFTAR ISI.....	iv
DAFTAR GAMBAR	vii
DAFTAR TABEL.....	x
INTISARI.....	Error! Bookmark not defined.
BAB I PENDAHULUAN	2
1.1 Latar Belakang.....	2
1.2 Rumusan Masalah.....	4
1.3 Tujuan.....	4
1.4 Batasan Masalah	5
1.5 Metodologi Penelitian.....	5
1.6 Sistematika Penulisan Tugas Akhir.....	6
BAB II TINJAUAN PUSTAKA.....	9
BAB III LANDASAN TEORI.....	12
3.1 Konsep Dasar Sistem Informasi	12
3.2 Monolitik	12
3.3 <i>Microservices</i>	14
3.4 Docker.....	16
3.4.1 <i>Docker File</i>	16
3.4.2 <i>Image</i>	16
3.4.3 <i>Container</i>	17
3.4.3 <i>Docker Swarm</i>	17

3.5	SignalR	17
3.6	InfluxDB	17
3.7	.Net Core.....	18
BAB IV ANALISIS DAN PERANCANGAN SISTEM		19
4.1	Analisis Arsitektur Microservices	19
4.2	<i>Orchestration</i>	23
4.3	Perancangan <i>Service</i>	28
4.3.1	Struktur <i>Project</i>	28
4.3.2	Perancangan <i>Service Gateway</i>	29
4.3.3	Perancangan <i>Service Kelas, Mahasiswa dan Pendjadwalan</i>	29
4.3.4	Perancangan <i>Service Autentikasi</i>	34
4.3.4	Perancangan <i>Services Tampilan Pengguna dan Tampilan .. Admin</i>	36
BAB V IMPLEMENTASI DAN PENGUJIAN MICROSERVICES		37
5.1	Pembuatan Docker File.....	37
5.1.1	<i>Service Gateway</i>	37
5.1.2	<i>Service Autentikasi</i>	38
5.1.3	<i>Service Penjadwalan</i>	39
5.1.4	<i>Service Mahasiswa</i>	40
5.1.5	<i>Service Kelas</i>	41
5.1.6	<i>Service Tampilan Admin</i>	42
5.1.5	<i>Service Tampilan Pengguna</i>	42
5.2	Pembuatan <i>Node</i>	43
5.3	Inialisasi <i>Swarm</i>	43
5.4	<i>Join Worker</i> Kedalam <i>Swarm</i>	44
5.5	Pembuatan <i>Compose File</i>	45



5.6	<i>Deployment Service</i>	53
5.7	Tampilan Admin	54
5.8	Tampilan Pengguna	55
5.9	Virtualizer	56
5.10	Portainer	57
5.11	Grafana	58
5.12	Pengujian <i>Microservices</i>	59
BAB VI		84
KESIMPULAN DAN SARAN		84
6.1	Kesimpulan	84
6.2	Saran	84
DAFTAR PUSTAKA		85

DAFTAR GAMBAR

Gambar 3. 1. Arsitektur Monolitik Taksi <i>Online</i>	13
Gambar 3. 2. Heterogenitas Pada <i>Microservices</i>	14
Gambar 3. 3. Penskalaan Pada <i>Microservices</i>	15
Gambar 4. 1. Arsitektur Monolitik Sistem KRS UAJY.....	19
Gambar 4. 2. Pembagian Sistem Menjadi Bagian <i>Frontend</i> dan <i>Backend</i>	20
Gambar 4. 3. Arsitektur <i>Microservices</i> KRS	21
Gambar 4. 4. <i>Swarm Cluster</i>	23
Gambar 4. 5. Alokasi <i>Service</i> Baru	25
Gambar 4. 6. <i>Container Down</i>	26
Gambar 4. 7. <i>Node Down</i>	27
Gambar 4. 8. Struktur <i>Project</i>	28
Gambar 4. 9. Penggunaan <i>Entity Framework</i>	30
Gambar 4. 10. Autentikasi <i>Controller</i>	30
Gambar 4. 11. Konfigurasi <i>Token</i>	34
Gambar 5. 1. <i>Docker File Service Gateway</i>	37
Gambar 5. 2. <i>Docker File Service</i> Autentikasi.....	38
Gambar 5. 3. <i>Docker File Service</i> Penjadwalan.....	39
Gambar 5. 4. <i>Docker File Service</i> Mahasiswa	40
Gambar 5. 5. <i>Docker File Service</i> Kelas	41
Gambar 5. 6. <i>Docker File Service</i> Tampilan Admin.....	42
Gambar 5. 7. <i>Docker File Service</i> Tampilan Pengguna	42
Gambar 5. 8. Pembuatan <i>Node</i>	43
Gambar 5. 9. Inisialisasi <i>Swarm</i>	43
Gambar 5. 10. Join <i>Worker1</i> Kedalam <i>Swarm</i>	44
Gambar 5. 11. Join <i>Worker2</i> Kedalam <i>Swarm</i>	44
Gambar 5. 12. Join <i>Worker3</i> Kedalam <i>Swarm</i>	44
Gambar 5. 13. Daftar <i>Node</i>	45
Gambar 5. 14. <i>Compose File</i>	50
Gambar 5. 15. <i>Deployment Services</i>	53

Gambar 5. 16. Daftar <i>Services</i>	53
Gambar 5. 17. Tampilan Admin	54
Gambar 5. 18. Tampilan Pengguna	55
Gambar 5. 19. Tampilan Virtualizer	56
Gambar 5. 20. Tampilan Portainer	57
Gambar 5. 21. Tampilan Grafana	58
Gambar 5. 22. <i>Thread Group</i> KRS Pengguna	59
Gambar 5. 23. <i>Thread Group</i> KRS Admin	59
Gambar 5. 24. <i>Field Request</i> Autentikasi	61
Gambar 5. 25. <i>Response Data</i> Autentikasi	61
Gambar 5. 26. <i>Field</i> Kelas KRS	62
Gambar 5. 27. <i>Response</i> Kelas KRS	62
Gambar 5. 28. <i>Field</i> Get Data Mahasiswa	63
Gambar 5. 29. <i>Response</i> Get Data Mahasiswa	63
Gambar 5. 30. <i>Field</i> Get Jadwal Mahasiswa	64
Gambar 5. 31. <i>Response</i> Get Jadwal Mahasiswa	64
Gambar 5. 32. <i>Field</i> Get Detil Kelas	65
Gambar 5. 33. <i>Response</i> Get Detil Kelas	65
Gambar 5. 34. <i>Field</i> Ambil KRS	66
Gambar 5. 35. <i>Field</i> Batal KRS	66
Gambar 5. 36. <i>Field</i> Get Daftar Fakultas	67
Gambar 5. 37. <i>Response</i> Get Daftar Fakultas	67
Gambar 5. 38. <i>Field</i> Get Jadwal KRS	68
Gambar 5. 39. <i>Response</i> Get Jadwal KRS	68
Gambar 5. 40. <i>Field</i> Get Kelas KRS	69
Gambar 5. 41. <i>Response</i> Get Kelas KRS	69
Gambar 5. 42. <i>Field</i> Update Kapasitas	70
Gambar 5. 43. <i>Field</i> Add Jadwal	70
Gambar 5. 44. <i>Field</i> Update Jadwal	71
Gambar 5. 45. <i>Field</i> Update Jadwal	71
Gambar 5. 46. Hasil Pengujian KRS Pengguna	72
Gambar 5. 47. Hasil Pengujian KRS Admin	72

Gambar 5. 48. Visualisasi <i>Swarm Normal</i>	76
Gambar 5. 49. Visualisasi <i>Swarm Worker Down</i>	77
Gambar 5. 50. Penggunaan RAM Pada <i>Service Autentikasi</i>	78
Gambar 5. 51. Penggunaan CPU Pada <i>Service Autentikasi</i>	78
Gambar 5. 52. Penggunaan RAM Pada <i>Service Gateway</i>	80
Gambar 5. 53. Penggunaan RAM Pada <i>Service Gateway</i>	80
Gambar 5. 54. Penggunaan RAM Pada <i>Service Kelas</i>	81
Gambar 5. 55. Penggunaan CPU Pada <i>Service Kelas</i>	81
Gambar 5. 56. Penggunaan RAM Pada <i>Service Mahasiswa</i>	82
Gambar 5. 57. Penggunaan CPU Pada <i>Service Mahasiswa</i>	82
Gambar 5. 58. Penggunaan RAM Pada <i>Service Penjadwalan</i>	83
Gambar 5. 59. Penggunaan CPU Pada <i>Service Penjadwalan</i>	83

DAFTAR TABEL

Tabel 2. 1. Tabel Perbandingan dengan Penelitian Terdahulu.....	10
Tabel 4. 1. Tabel Daftar <i>End Point Service</i> Mahasiswa, Kelas dan Schedule.....	31
Tabel 4. 2. Tabel Daftar <i>End Point</i> Autentikasi.....	35



INTISARI

Danang Kusuma Yudha (14 07 07818)

KRS atau kartu rencana studi adalah kartu yang berisi daftar mata kuliah yang akan diikuti oleh setiap mahasiswa dalam satu semester. Pengisian kartu KRS pada Universitas Atma Jaya Yogyakarta sudah dilakukan secara *online*. Namun Pengisian KRS yang dilakukan secara online memiliki kendala pada saat jumlah mahasiswa yang melakukan pengisian KRS berjumlah banyak maka situs yang digunakan untuk melakukan pengisian KRS secara *online* mengalami peningkatan penggunaan CPU dan RAM pada *server* sehingga menyebabkan *server* yang digunakan menjadi *down*.

Tujuan dari penelitian ini adalah untuk mengubah arsitektur pada sistem KRS Online di Universitas Atma Jaya Yogyakarta yang berbentuk monolitik menjadi *microservices*. Tujuan dari penggunaan arsitektur *microservices* dikarenakan dalam arsitektur ini sistem yang dibuat dalam bentuk monolitik dapat dipecah berdasarkan *service* domain nya dan dijalankan dalam bentuk *container* sehingga kita dapat melakukan *monitoring* dan pembatasan sumber daya pada masing-masing *service*.

Docker digunakan sebagai *engine* untuk menjalankan *container* karena memiliki fitur *orchestration* yang dapat melakukan *load balancing* dan *scheduling* terhadap *container* sehingga membuat sistem KRS Online yang dibangun akan menjadi lebih *reliable* dan *fault tolerance*.

Kata kunci: *Docker, Container, Microservices, Monolitik, KRS, Sistem Informasi.*

