

## LEMBAR REVISI TUGAS AKHIR

NAMA MAHASISWA : Gloriyano Cristho Daniel Pepuho  
NOMOR POKOK : 5025201121  
JUDUL TUGAS AKHIR : PENGELOLAAN PENGGUNAAN INFRASTRUKTUR GPU UNTUK  
PENGGUNA BERBASIS DOCKER CONTAINER MENGGUNAKAN  
JUPYTERLAB  
RMK : Komputasi Berbasis Jaringan  
DOSEN PEMBIMBING : 1.Ir. Ary Mazharuddin Shiddiqi, S.Kom., M.Comp.Sc., Ph.D.  
2. Royyana Muslim Ijtihadie, S.Kom., M.Kom., Ph.D.

NO	REVISI	Lokasi
1	point revisi: - Menghapus kata “Departement” yang ganda - Menghapus kata “Engineering” - Mengubah “Sepuluh Nopember Institute of Technology”  hasil revisi: - Department of Informatics - Institut Teknologi Sepuluh Nopember Surabaya	Cover Bahasa Inggris
2	point revisi: Melakukan revisi keterangan gambar di halaman 59 pada DAFTAR GAMBAR  hasil revisi: - Gambar 4.15: User yang terdaftar di halaman admin JupyterHub - Gambar 4.16: Total container yang aktif di node rpl .	Halaman ix
3	point revisi: Menghapus whitespace	Halaman 23, 26, 28, 29, 32, 35, 37, 38, 52, 53

Catatan :

- Tambahkan point revisi jika diperlukan
- Berikan keterangan penempatan revisi dengan jelas, sehingga memudahkan penguji untuk memeriksa. Contoh
  - o buku TA Subbab 2.2 halaman 30
  - o fitur program ditambahkan di menu abc
- Dosen penguji memberikan tanda tangan setelah mahasiswa ybs mengerjakan revisi

Surabaya, 28 Juli 2025  
Dosen Penguji,

Prof. Tohari Ahmad, S.Kom., M.IT., Ph.D

## LEMBAR REVISI TUGAS AKHIR

NAMA MAHASISWA : Gloriyano Cristho Daniel Pepuho  
NOMOR POKOK : 5025201121  
JUDUL TUGAS AKHIR : PENGELOLAAN PENGGUNAAN INFRASTRUKTUR GPU UNTUK  
PENGGUNA BERBASIS DOCKER CONTAINER MENGGUNAKAN  
JUPYTERLAB  
RMK : Komputasi Berbasis Jaringan  
DOSEN PEMBIMBING : 1.Ir. Ary Mazharuddin Shiddiqi, S.Kom., M.Comp.Sc., Ph.D.  
2. Royyana Muslim Ijtihadie, S.Kom., M.Kom., Ph.D.

NO	REVISI	Lokasi
1	<p>point revisi: Revisi Abstrak, menghapus kata “berarti”.</p> <p>Pengujian dilakukan dengan mensimulasikan pengguna secara bersamaan menggunakan. Hasil uji coba awal dengan 10 pengguna menunjukkan bahwa sistem mampu menangani permintaan secara serentak tanpa kendala <b>berarti</b>. Namun, pengujian lanjutan dalam skala besar masih diperlukan untuk mengevaluasi ketahanan sistem secara menyeluruh.</p> <p><i>hasil revisi:</i></p> <p>Pengujian dilakukan dengan menyimulasikan pengguna secara bersamaan menggunakan alat uji beban Locust. Hasil uji coba dengan 10 hingga 20 pengguna simultan menunjukkan bahwa sistem mampu menangani permintaan secara paralel tanpa kendala signifikan. Sistem berhasil meluncurkan seluruh server JupyterLab dengan distribusi node yang efisien. Hal ini menunjukkan bahwa arsitektur yang dikembangkan cukup andal untuk skenario multi-pengguna. Pengujian skala lebih besar tetap diperlukan untuk mengevaluasi batas maksimal ketahanan dan skalabilitas sistem.</p>	Halaman i
2	<p>point revisi: Revisi Kesimpulan, menghapus kata “berarti”.</p> <p>1. Pengujian awal dengan 10 pengguna secara bersamaan menunjukan bahwa sistem mampu mendistribusikan <i>container</i> tanpa kendala <b>berarti</b>.</p>	Halaman 67

	<p>hasil revisi:</p> <p>1. Pengujian sistem dilakukan dengan menyimulasikan beban multi-user secara konkuren menggunakan alat uji Locust. Hasil pengujian dengan 10 hingga 20 pengguna simultan menunjukkan bahwa sistem mampu menangani permintaan secara paralel dengan stabil. Seluruh container JupyterLab berhasil dijalankan di node yang tersebar secara efisien, dan alokasi pengguna berpindah otomatis ke node lain saat beban tinggi terdeteksi.</p>	
<p>3</p>	<p>point revisi:</p> <p>Menambah beban komputasi, beban ini berupa proses komputasi intensif yang berjalan terus-menerus, sehingga memaksa node tersebut memiliki tingkat utilisasi CPU dan memori yang tinggi.</p> <p>Tujuan dari skenario ini adalah untuk memverifikasi apakah mekanisme pemilihan <i>node</i> oleh Discovery Service dapat secara otomatis mengalihkan proses <i>spawn</i> JupyterLab ke <i>node</i> lain yang memiliki beban lebih rendah</p> <p>hasil revisi:</p> <p>Telah ditambahkan penjelasan pada subbab <b>4.2.2.2</b> mengenai pengujian dengan menambahkan beban komputasi secara sengaja pada salah satu node. Pengujian ini bertujuan untuk memastikan bahwa sistem dapat menghindari node dengan beban tinggi dan secara otomatis mendistribusikan permintaan ke node lain yang memiliki sumber daya lebih ringan. Hasilnya menunjukkan bahwa distribusi beban berjalan sesuai harapan dan pemilihan node oleh Discovery API telah berfungsi secara adaptif.</p> <pre> [3]: import numpy as np import time import os  print("Starting computation") MATRIX_SIZE = 10000 A = np.random.rand(MATRIX_SIZE, MATRIX_SIZE) B = np.random.rand(MATRIX_SIZE, MATRIX_SIZE)  results = []  start = time.time() duration = 0 i = 0 while duration &lt; 60:     C = np.matmul(A, B)     results.append(C)     i += 1     duration = time.time() - start     print(f"Iteration {i} done, elapsed time: {duration:.2f}")  print(f"Total matrices stored: {len(results)}") print(f"Overload kernel hostname: {os.uname().nodename}")  Starting computation Iteration 1 done, elapsed time: 7.71 Iteration 2 done, elapsed time: 15.52 Iteration 3 done, elapsed time: 22.98 Iteration 4 done, elapsed time: 30.51 Iteration 5 done, elapsed time: 38.04 Iteration 6 done, elapsed time: 45.79 Iteration 7 done, elapsed time: 53.31 Iteration 8 done, elapsed time: 61.44 Total matrices stored: 8 Overload kernel hostname: rpl-1 </pre>	<p>Halaman 61</p>

```
[W 2025-07-23 05:43:56.741 EnterpriseGatewayApp] No channel specified, assuming shell: {'header': {'msg_id': '920c29ad-d81f-47c4-aa07-c9a80b4549f7', 'username': 'jovyan', 'session': '5291bd19-b3fc-4b27-bd3b-1282b5569987', 'msg_type': 'execute_request', 'version': '5.3'}, 'parent_header': {}, 'metadata': {}, 'content': {'code': '\nimport numpy as np\nimport time\nimport os\n\nprint("Starting computation")\nMATRIX_SIZE = 10000\nA = np.random.rand(MATRIX_SIZE, MATRIX_SIZE)\nB = np.random.rand(MATRIX_SIZE, MATRIX_SIZE)\n\nresults = []\nstart = time.time()\nduration = 0\ni = 0\nwhile duration < 60:\n    C = np.matmul(A, B)\n    results.append(C)\n    i += 1\n    duration = time.time() - start\n    print(f"Iteration {i} done, elapsed time: {duration:.2f}")\n\nprint(f"Total matrices stored: {len(results)}")\nprint(f"Overload kernel hostname: {os.uname().nodename}")\n', 'silent': False}}
```

```
[W 2025-07-23 05:43:56.742 EnterpriseGatewayApp] No channel specified, assuming shell: {'header': {'msg_id': 'f72a2b04-9558-405a-a513-85a000d7fa68', 'username': 'jovyan', 'session': '8356f093-c601-493f-b426-d44ebd9fb57c', 'msg_type': 'execute_request', 'version': '5.3'}, 'parent_header': {}, 'metadata': {}, 'content': {'code': '\nimport numpy as np\nimport time\nimport os\n\nprint("Starting computation")\nMATRIX_SIZE = 10000\nA = np.random.rand(MATRIX_SIZE, MATRIX_SIZE)\nB = np.random.rand(MATRIX_SIZE, MATRIX_SIZE)\n\nresults = []\nstart = time.time()\nduration = 0\ni = 0\nwhile duration < 60:\n    C = np.matmul(A, B)\n    results.append(C)\n    i += 1\n    duration = time.time() - start\n    print(f"Iteration {i} done, elapsed time: {duration:.2f}")\n\nprint(f"Total matrices stored: {len(results)}")\nprint(f"Overload kernel hostname: {os.uname().nodename}")\n', 'silent': False}}
```

```
[testuser7] Memulai virtual user...
[testuser7] Login berhasil (menerima 302 redirect).
[testuser7] Updated XSRF token and cookies after /hub/home GET.
[testuser7] Mulai proses pemilihan profil...
[testuser7] Profil terpilih: single-cpu
[testuser7] Node terpilih oleh Discovery Service: rpl-02
[testuser7] Mengirim permintaan spawn JupyterLab...
[testuser3] Memulai virtual user...
[testuser3] Login berhasil (menerima 302 redirect).
[testuser3] Updated XSRF token and cookies after /hub/home GET.
[testuser3] Mulai proses pemilihan profil...
[testuser3] Profil terpilih: single-cpu
[testuser3] Node terpilih oleh Discovery Service: rpl
[testuser3] Mengirim permintaan spawn JupyterLab...
[testuser6] Memulai virtual user...
[testuser6] Login berhasil (menerima 302 redirect).
[testuser6] Updated XSRF token and cookies after /hub/home GET.
[testuser6] Mulai proses pemilihan profil...
[testuser6] Profil terpilih: single-cpu
[testuser6] Node terpilih oleh Discovery Service: rpl
[testuser6] Mengirim permintaan spawn JupyterLab...
```

Catatan :

- Tambahkan point revisi jika diperlukan
- Berikan keterangan penempatan revisi dengan jelas, sehingga memudahkan penguji untuk memeriksa. Contoh
  - o buku TA Subbab 2.2 halaman 30
  - o fitur program ditambahkan di menu abc
- Dosen penguji memberikan tanda tangan setelah mahasiswa ybs mengerjakan revisi

Surabaya, 28 Juli 2025  
Dosen Penguji,

Baskoro Adi Pratomo, S.Kom.,  
M.Kom., Ph.D.