



REKTOR UNIVERSITAS GADJAH MADA
KEPUTUSAN REKTOR UNIVERSITAS GADJAH MADA
NOMOR 1197/UN1.P5/KPT/HUKOR/2025

TENTANG
PEDOMAN PENGELOLAAN PEMISAHAN TUGAS
PADA SISTEM FINANCE SIMASTER UNIVERSITAS GADJAH MADA

REKTOR UNIVERSITAS GADJAH MADA,

Menimbang : bahwa untuk mendukung tata kelola sistem informasi yang baik dan menjamin layanan teknologi informasi yang handal dan menjamin keamanan data yang memerlukan kebijakan formal mengenai pedoman pengembangan dan pengelolaan Sistem Finance Simaster Universitas Gadjah Mada, perlu menetapkan Keputusan Rektor Universitas Gadjah Mada tentang Pedoman Pengelolaan Pemisahan Tugas pada Sistem Finance Simaster Universitas Gadjah Mada;

Mengingat :
1. Peraturan Pemerintah Nomor 67 Tahun 2013 tentang Statuta Universitas Gadjah Mada (Lembaran Negara Republik Indonesia Tahun 2013 Nomor 165, Tambahan Lembaran Negara Republik Indonesia Nomor 5454);
2. Peraturan Majelis Wali Amanat Universitas Gadjah Mada Nomor 4/SK/MWA/2014 tentang Organisasi dan Tata Kelola (*Governance*) Universitas Gadjah Mada, sebagaimana telah diubah terakhir dengan Peraturan Majelis Wali Amanat Universitas Gadjah Mada Nomor 5 Tahun 2023 tentang Perubahan Kedelapan atas Peraturan Majelis Wali Amanat Universitas Gadjah Mada Nomor 4/SK/MWA/2014 tentang Organisasi dan Tata Kelola (*Governance*) Universitas Gadjah Mada;
3. Keputusan Rektor Universitas Gadjah Mada Nomor 743/UN1.P/KPT/HUKOR/2022 tentang Wakil Rektor Bidang Perencanaan, Aset, dan Sistem Informasi Universitas Gadjah Mada Periode Tahun 2022-2027;

MEMUTUSKAN:

Menetapkan : KEPUTUSAN REKTOR UNIVERSITAS GADJAH MADA TENTANG PEDOMAN PENGELOLAAN PEMISAHAN TUGAS PADA SISTEM FINANCE SIMASTER UNIVERSITAS GADJAH MADA.

KESATU : Pedoman Pengelolaan Pemisahan Tugas pada Sistem Finance Simaster Universitas Gadjah Mada sebagaimana tercantum dalam Lampiran Keputusan ini.

KEDUA : Pelaksanaan pengembangan Sistem Finance Simaster harus dilakukan berdasarkan Pedoman Pengelolaan Pemisahan Tugas pada Sistem Finance Simaster Universitas Gadjah Mada.



KETIGA : Keputusan ini mulai berlaku pada tanggal ditetapkan.

Ditetapkan di Yogyakarta
pada tanggal 1 Oktober 2025
a.n. REKTOR
WAKIL REKTOR BIDANG PERENCANAAN,
ASET DAN SISTEM INFORMASI,

ditandatangani secara elektronik

ARIEF SETIAWAN BUDI NUGROHO

Tembusan:

1. Sekretaris Universitas
2. Direktur Teknologi Informasi
di Universitas Gadjah Mada



LAMPIRAN KEPUTUSAN REKTOR UNIVERSITAS GADJAH MADA
NOMOR : 1197/UN1.P5/KPT/HUKOR/2025
TANGGAL : 1 OKTOBER 2025
TENTANG : PEDOMAN PENGELOLAAN PEMISAHAN TUGAS PADA SISTEM
FINANCE SIMASTER UNIVERSITAS GADJAH MADA

A. Istilah dan Pengertian

1. *Development Environment* atau yang selanjutnya disebut *development* adalah lingkungan khusus untuk pengembangan fitur dan perbaikan *bug*. Lingkungan ini ada pada *workstation* masing-masing pengembang.
2. *Developer* adalah individu atau tim teknis yang bertugas untuk merancang, membangun, memodifikasi, dan memelihara kode sistem informasi sesuai dengan kebutuhan yang disampaikan oleh *user*.
3. *Lead Developer* adalah peran dalam tim pengembangan sistem informasi yang memiliki kewenangan untuk melakukan reviu kualitas kode dan persetujuan perubahan dan penyatuan kode (*merge*) ke lingkungan produksi yang diajukan oleh *developer*.
4. *Production Environment* adalah lingkungan sistem yang digunakan oleh pengguna akhir dengan data asli.
5. *Quality Assurance (QA)* adalah suatu proses yang bertujuan untuk menjamin bahwa perangkat lunak atau sistem dikembangkan, diuji, dan dikelola sesuai dengan kebutuhan pengguna, standar teknis, serta peraturan yang berlaku.
6. *Staging Environment* adalah lingkungan uji coba untuk *Quality Assurance (QA)* dan *User Acceptance Test (UAT)*.
7. *User* adalah pihak atau individu yang menggunakan dan memiliki akses sistem informasi untuk mendukung proses bisnis universitas.

B. Prosedur

1. Pemisahan Lingkungan
 - a) pengembangan Finance Simaster wajib memiliki minimal 3 (tiga) lingkungan yaitu *development*, *staging* dan *production*;
 - b) *Database production* tidak boleh digunakan untuk *development* ataupun *staging*;
 - c) dalam hal lingkungan *development* atau *staging* memerlukan *raw* data *production*, maka sebagian data *production* dapat di-*copy* dan kemudian di-*masking* dari berbagai data yang menunjukkan privasi sebelum dipergunakan di lingkungan *development* atau *staging*; dan
 - d) Aturan penggunaan data adalah sebagai berikut:
 - 1) *Production*: semua data nyata yang berkualitas dan sesuai dengan yang dibutuhkan pengguna;
 - 2) *Staging*: sebagian atau beberapa data nyata yang diambil dari produksi yang sudah dilakukan *masking* untuk menghilangkan privasi dan identifikasi unik data; dan
 - 3) *Development*: tidak ada data yang diambil *production* atau *staging*.
2. Hak Akses
 - a) setiap akses ke sistem finance simaster wajib menggunakan akun SSO personal bukan akun SSO institusi/akun unit kerja;
 - b) Direktur Teknologi Informasi menetapkan personil yang diberikan hak akses ke lingkungan *production*; dan
 - c) penetapan personil tersebut dituangkan ke dalam surat tugas dan dievaluasi secara berkala maksimum 2 (dua) tahun sekali.
3. *Change Management*
 - a) unit kerja mengajukan permohonan pengembangan/perubahan/perbaikan fitur melalui admin *roadmap unit* di laman admin.simaster.ugm.ac.id;



- b) pimpinan Direktorat Teknologi Informasi (DTI) melakukan reviu permintaan fitur baru dan/atau perubahan dan membentuk tim kajian;
 - c) tim kajian melakukan reviu dan membuat rekomendasi kelayakan permohonan;
 - d) apabila hasil kajian merekomendasikan bahwa permohonan pengembangan/perubahan/perbaikan diproses, pimpinan DTI menunjuk *project manager*.
 - e) *project manager* kemudian membuat daftar tugas dan menugaskan *developer* untuk melakukan penulisan kode (*coding*).
 - f) proses selanjutnya mengikuti siklus dan alur kerja *development* dan *deployment* yang selaras dengan prinsip UGM Lean.
4. Monitoring dan Evaluasi
- a) aktivitas akses di lingkungan *production* dicatat dalam berkas *log*; dan
 - b) berkas *log* disimpan minimal 1 (satu) tahun dan diperiksa secara berkala.

C. Alur Kerja *Development* dan *Deployment*

1. *developer* mengembangkan/memperbaiki fitur di lingkungan *development* dan melakukan unit testing pada fitur yang dikerjakan;
2. setelah melakukan *unit testing*, *developer* melakukan *commit* perubahan dan melakukan *submit* ke *server staging/testing*;
3. QA atau *user* untuk melakukan pengujian fungsional, dan integrasi pada lingkungan *staging*;
4. apabila ditemukan *bug*, maka QA/*User* menginformasikan ke *developer* dan dilakukan kembali proses pengembangan/perbaikan fitur;
5. QA/*User* mengisi form *User Acceptance Test* (UAT) setelah melakukan pengujian fitur di lingkungan *staging*;
6. setelah UAT ditandatangani oleh QA/*User*, *lead developer* memproses *merge request* ke *branch* utama; dan
7. *DevOps Engineer* melakukan *merge* dan melakukan *deploy* ke *production*.

D. Peran dan Tanggung Jawab

1. *DevOps Engineer*, memiliki tugas dan tanggung jawab:
 - a) mengelola *pipeline* CI/CD untuk *build*, *test*, dan *deployment* otomatis;
 - b) menjalankan *deployment* ke *staging* dan *production* sesuai persetujuan;
 - c) menjaga konsistensi *environment* (Dev, Test, Prod);
 - d) memastikan keamanan, *logging*, dan *monitoring* pasca *deployment*; dan
 - e) menyusun mekanisme *rollback* bila *deployment* gagal.
2. *Developer*, memiliki tugas dan tanggung jawab:
 - a) menganalisis kebutuhan *user* dan membuat desain teknis;
 - b) mengembangkan kode program sesuai standar;
 - c) melakukan *unit testing* dan dokumentasi kode;
 - d) mengajukan *merge request* untuk *review Lead Developer*; dan
 - e) melakukan perbaikan (*bug fixing*) sesuai hasil QA/UAT.
3. *Lead Developer*, memiliki tugas dan tanggung jawab:
 - a) memberikan *approval*/penolakan *merge request* sebelum integrasi ke *branch* utama;
 - b) memberikan arahan standar *coding*, arsitektur, dan keamanan sistem;
 - c) membimbing *developer* lain dalam aspek teknis dan *best practice*; dan
 - d) berkoordinasi dengan QA dan *DevOps* untuk memastikan kesiapan sebelum *deployment*.
4. *Quality Assurance*, memiliki tugas dan tanggung jawab:
 - a) menyusun rencana uji (*test plan*) dan skenario uji (*test case*);
 - b) melakukan *functional testing*, *integration testing*, dan *regression testing*;
 - c) berkoordinasi dengan *user* untuk *User Acceptance Test* (UAT);



- d) memberikan rekomendasi *go/no-go* sebelum *deployment* ke *production*; dan
- e) mencatat temuan *bug/defect* dan memastikan perbaikannya.

a.n. REKTOR
WAKIL REKTOR BIDANG PERENCANAAN,
ASET DAN SISTEM INFORMASI,

ditandatangani secara elektronik

ARIEF SETIAWAN BUDI NUGROHO

