



UINSSC

UNIVERSITAS ISLAM NEGERI SIBER
SYEKH NURJATI CIREBON

**KEBIJAKAN/PROSEDUR
(SOP)**

PERILISAN PRODUK IT

PUSAT TEKNOLOGI INFORMASI DAN KOMUNIKASI

**UNIVERSITAS ISLAM NEGERI SIBER
SYEKH NURJATI CIREBON**

SOP PERILISAN PRODUK IT

1. PENDAHULUAN

SOP ini merupakan panduan terstruktur bagi seluruh tim terkait dalam proses perilisan produk IT. Tujuannya adalah untuk menjamin kualitas, stabilitas, dan keamanan produk sebelum diluncurkan, serta memastikan komunikasi dan koordinasi yang efektif di antara tim.

2. TUJUAN

- Menjamin kualitas dan stabilitas produk sebelum dirilis.
- Mengelola risiko yang mungkin timbul saat perilisan.
- Memastikan semua pihak terkait memahami dan menjalankan perannya.
- Menyediakan panduan yang jelas dan terstandarisasi untuk setiap perilisan.

3. PIHAK TERLIBAT DAN TANGGUNG JAWAB

<i>Pihak Terlibat</i>	<i>Tanggung Jawab Utama</i>
<i>Product Owner (PO) / Manajer Proyek</i>	Memimpin dan menyetujui jadwal rilis, berkomunikasi dengan stakeholder, dan memberikan persetujuan akhir.
<i>Tim Pengembang (Developer)</i>	Mengembangkan fitur, melakukan unit testing, dan menyiapkan kode untuk deployment.
<i>Tim Quality Assurance (QA)</i>	Melakukan pengujian menyeluruh (fungsional, regresi, performa, keamanan).
<i>Tim Infrastruktur (Ops)</i>	Menyiapkan server, melakukan deployment, dan memantau sistem pasca-rilis.
<i>Tim Komunikasi</i>	Menyiapkan dan mendistribusikan informasi rilis kepada pengguna dan stakeholder.

4. TAHAPAN SOP PERILISAN PRODUK IT

4.1 Perencanaan Pra-Rilis

1. Kick-off Meeting: PO/Manajer Proyek memimpin pertemuan untuk menentukan jadwal rilis, fitur yang akan dirilis, dan mengidentifikasi potensi risiko.
 2. Penyusunan Rencana Rilis: Dokumen rencana rilis dibuat, mencakup:
 - Daftar fitur atau perbaikan.
 - Tanggal dan waktu rilis yang disepakati.
 - Daftar tugas dan penanggung jawab (Tim Pengembang, QA, Ops).
 - Rencana rollback jika terjadi kegagalan.
 3. Pengembangan dan Uji Unit: Tim Pengembang menyelesaikan kode dan melakukan uji unit secara mandiri. Kode yang dianggap stabil kemudian dipindahkan ke lingkungan pengujian (staging environment).
-

4.2 Pengujian dan Persiapan

1. Pengujian QA: Tim QA melakukan pengujian menyeluruh pada lingkungan staging. Pengujian ini mencakup:
 - Uji Fungsional: Memastikan fitur berjalan sesuai spesifikasi.
 - Uji Regresi: Memastikan fitur baru tidak merusak fitur yang sudah ada.
 - Uji Performa: Mengukur kinerja sistem.
 - Uji Keamanan: Mengidentifikasi celah keamanan.
 2. Persetujuan Rilis: Setelah semua pengujian berhasil dan tidak ada bug krusial, Tim QA memberikan persetujuan kepada PO/Manajer Proyek.
 3. Finalisasi Kode: PO/Manajer Proyek memberikan persetujuan final untuk perilisan. Tim Pengembang melakukan code freeze (penghentian pengembangan fitur baru) dan mempersiapkan kode untuk deployment.
 4. Persiapan Infrastruktur: Tim Infrastruktur menyiapkan server produksi, memastikan kapasitas dan konfigurasi sudah sesuai.
-

4.3 Perilisan (Deployment)

1. Pemberitahuan Perilisan: Tim Komunikasi memberitahukan kepada stakeholder dan pengguna mengenai jadwal rilis dan kemungkinan downtime (jika ada).
 2. Deployment: Tim Infrastruktur melakukan deployment kode dari lingkungan staging ke lingkungan produksi. Proses ini harus didokumentasikan.
 3. Smoke Test: Segera setelah deployment, Tim Pengembang dan QA melakukan uji cepat (smoke test) pada lingkungan produksi untuk memastikan sistem berjalan dengan baik.
 4. Verifikasi Rilis: PO/Manajer Proyek memverifikasi bahwa produk atau fitur baru telah berhasil dirilis dan berfungsi dengan benar.
-

4.4 Pasca-Rilis

1. Pemantauan Sistem: Tim Infrastruktur memantau performa, error log, dan metrik penting lainnya secara intensif selama 24-48 jam pertama setelah rilis.
2. Penanganan Masalah (Rollback Plan): Jika ditemukan bug kritis atau masalah serius, segera aktifkan Rencana Rollback. Tim Infrastruktur mengembalikan sistem ke versi sebelumnya yang stabil.
3. Sosialisasi dan Dokumentasi: Tim Komunikasi melakukan sosialisasi fitur baru kepada pengguna. Tim Pengembang memperbarui dokumentasi teknis dan panduan pengguna.
4. Retrospeksi: Setelah rilis selesai, semua tim terkait mengadakan pertemuan retrospeksi untuk mengevaluasi proses, mengidentifikasi tantangan, dan menentukan area perbaikan untuk rilis selanjutnya.

5. RENCANA ROLLBACK

- **Identifikasi Masalah:** Segera setelah masalah kritis terdeteksi, tim harus berkoordinasi untuk mengidentifikasi penyebab dan dampaknya.
- **Keputusan Rollback:** PO/Manajer Proyek berhak memutuskan untuk melakukan rollback jika masalah tidak dapat diperbaiki dengan cepat.
- **Proses Rollback:** Tim Infrastruktur mengembalikan sistem ke versi stabil sebelumnya, biasanya dengan memulihkan cadangan (backup) atau men-deploy kembali versi kode yang lama.
- **Pemberitahuan:** Informasikan kepada pengguna bahwa sistem telah dikembalikan ke versi sebelumnya untuk perbaikan.