

## **Prinsip-Prinsip & Etika di Software Engineering**

Bidang rekayasa perangkat lunak telah sangat matang sejak dimulai lebih dari 40 tahun yang lalu. Selama ini para praktisi telah belajar pelajaran berharga yang berkontribusi pada praktik terbaik saat ini. Beberapa telah menjadi usang, tetapi banyak yang masih sangat relevan dan diimplementasikan secara luas saat ini.

Kualitas adalah Nomor 1

Tidak ada yang lebih penting daripada memberikan produk berkualitas kepada pelanggan — mereka tidak akan memberikan toleransi apabila ada sesuatu yang kurang. Namun, setiap orang pasti memiliki pemikiran atau definisi tentang kualitas software yang berbeda dan oleh sebab itu perlu untuk menentukan dan mengukur kualitas tersebut. Contoh-contoh kualitas yang baik adalah seberapa dekat software memenuhi permintaan pelanggan, berapa banyak (atau sedikit) cacat yang dimilikinya, dan berapa biaya yang dibutuhkan untuk melakukan produksi. Ukuran kualitas perlu ditentukan sebelumnya untuk memastikan target yang benar sedang dikejar dan dipenuhi.

Perangkat Lunak Berkualitas Tinggi adalah Mungkin

Meskipun mungkin sulit untuk menghasilkan perangkat lunak berkualitas tinggi, berdasarkan metode dan teknik rekayasa perangkat lunak yang telah terbukti memenuhi sasaran kualitas. Contohnya termasuk melibatkan pelanggan, membuat prototipe, melakukan inspeksi, dan menggunakan proses perangkat lunak tambahan.

Berikan Produk kepada Pelanggan Sejak Dini

Banyak proyek perangkat lunak yang gagal dikarenakan pelanggan mendapatkan sebuah pandangan pertama pada perangkat lunak mereka yang terlambat dalam siklus pengembangannya. Ini adalah motivasi utama untuk pengenalan metode Agile. Hampir tidak mungkin untuk mengetahui semua kebutuhan pelanggan di awal, dan melibatkan pelanggan sedini mungkin sangat penting untuk menjaring seluruh kebutuhan yang tepat.

Keterlibatan sejak dini pelanggan sangat membantu dalam menentukan kebutuhan. Pelanggan mungkin berpikir mereka menginginkan sebuah fitur tertentu, atau berpikir mereka ingin antarmuka pengguna tertentu, tetapi tunggu sampai mereka mendapatkan versi demo

perangkat lunak. Sehingga mereka dapat terbantu dalam mengambil keputusan dalam merumuskan kebutuhan yang sebenarnya.

### Gunakan Proses Pengembangan Perangkat Lunak yang Tepat

Ada banyak model proses perangkat lunak, dan tidak ada satu pun yang benar-benar sesuai untuk setiap jenis proyek. Untuk contoh, proses Waterfall bekerja dengan baik untuk proyek-proyek di mana semua persyaratan sudah diketahui di depan. Sebaliknya, proses Agile dan Iteratif lainnya dipanggil ketika hanya beberapa persyaratan diketahui sebelumnya. Seorang Software Engineer dan seorang pemimpin proyek yang baik mampu akan meluangkan waktu untuk memahami jenis proyek yang sedang dikerjakan dan menggunakan model yang sesuai.

### Minimalkan Jarak Intelektual

Prinsip ini mengatakan bahwa untuk setiap solusi perangkat lunak dalam masalah dunia nyata, struktur dari kedua perangkat lunak solusi dan masalah dunia nyata harus sedekat mungkin. Semakin dekat struktur satu sama lain, semakin mudah mengembangkan dan memelihara perangkat lunak karena memiliki kesinambungan dengan kehidupan nyata.

### Inspect Code

Ini harus diperluas untuk dibaca “Inspect All Artifacts” di mana artefak didefinisikan sebagai produk apa pun dari proses pengembangan perangkat lunak termasuk spesifikasi teknis, rencana pengujian, dokumentasi, dan kode. Inspeksi telah terbukti mampu menemukan kesalahan sedini mungkin, meningkatkan kualitas, dan menurunkan biaya proyek secara signifikan.

### Orang adalah Kunci Sukses

Orang-orang yang sangat terampil dan termotivasi merupakan faktor paling penting yang berkontribusi terhadap keberhasilan proyek perangkat lunak. Orang-orang yang tepat ini dapat menembus berbagai rintangan walaupun dengan peralatan yang tidak memadai dan berbagai masalah yang tak terduga. Orang-orang ini akan mencari jalan untuk mengatasi rintangan dan membuat proyek itu sukses. Mempekerjakan dan mempertahankan orang-orang terbaik sangat penting untuk menghasilkan perangkat lunak berkualitas tinggi dan sukses.

Kepercayaan pada etika Software Engineering kini telah menjadi bagian penting dari budaya kontemporer. Untuk mengambil contoh misalnya, ahli radiologi untuk memverifikasi kebenaran kode yang mengendalikan mesin X-ray yang menunjuk ke arahnya kepala. Pada

titik tertentu, tidak ada pilihan selain mengasumsikan bahwa perangkat lunak dibuat dan diinstal di berbagai sistem yang diterapkan tidak memiliki jaminan bahwa telah terimplementasikan dengan benar, dan dengan cara yang konsisten dengan kepentingan publik.

Seorang Software Engineer harus berkomitmen untuk membuat analisis, spesifikasi, desain, pengembangan, pengujian dan pemeliharaan perangkat lunak profesi yang bermanfaat dan dihormati. Seorang Software Engineer harus mematuhi Delapan Prinsip berikut:

**Publik**— Seorang Software Engineer harus bertindak konsisten dengan kepentingan publik.

**Client dan Pegawai** — Seorang Software Engineer harus mampu memberikan tindakan yang terbaik untuk kepentingan Client ataupun Atasan mereka secara konsisten dengan kepentingan publik.

**Produk** — Seorang Software Engineer harus memastikan bahwa produk mereka dan modifikasi terkait, memenuhi standar profesional setinggi mungkin.

**Judgment** — Seorang Software Engineer harus menjaga integritas dan kemandirian dalam penilaian profesional mereka.

**Manajemen** — Manajer dan pemimpin pengembangan perangkat lunak harus bisa membagikan dan memsosialkan suatu pendekatan etis untuk manajemen sebuah pengembangan perangkat lunak dan pemeliharaannya.

**Profesi** — Seorang Software Engineer harus meningkatkan integritas dan reputasi profesi yang konsisten dengan kepentingan publik.

**Kolega** — Seorang Software Engineer harus adil dan mendukung rekan kerja mereka.

**Diri Sendiri** — Seorang Software Engineer harus mau untuk terus belajar sepanjang hayat yang berkaitan dengan praktik profesinya dan harus mempromosikan pendekatan etika terhadap praktik profesi.

|             |                                                 |
|-------------|-------------------------------------------------|
| Judul Tugas | BPJS & Digital Devide                           |
| MK          | Ethical Issues in Electronic Information System |
| Mahasiswa   | M. Riski Qisthiano (182420040)                  |
| Tanggal     | 05-01-2020                                      |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>1. Dampak Teknologi Informasi terhadap biaya layanan kesehatan di Indonesia</p> <ul style="list-style-type: none"> <li>• Meningkatkan kualitas pelayanan kesehatan</li> <li>• Mencegah kesalahan medis</li> <li>• Mengurangi biaya perawatan kesehatan</li> <li>• Meningkatkan efisiensi administrasi</li> <li>• Menurunkan dokumen</li> <li>• Memperluas akses jangkauan perawatan</li> </ul>                                                                                                                                                                                                                                                                             |
| <p>2. studi kasus BPJS, dan problema etika yang mungkin muncul</p> <ul style="list-style-type: none"> <li>• Kemajuan teknologi informasi dibidang kesehatan memang dapat memberikan banyak manfaat,terutama dalam pemerataan akses dan informasi terhadap kesehatan, namun ada pula dampak buruk yang dapat ditimbulkan dari hal ini. Contohnya adalah berkembangnya teknologi tentang penyedia informasi kesehatan atau alat diagnosa kesehatan yang dapat digunakan sendiri mampu membawa kekhawatiran terhadap eksistensi profesi dokter dan tenaga kesehatan lainnya di tengah-tengah masyarakat. Bagaimanapun, teknologi tetaplah sebuah alat untuk kehidupan</li> </ul> |

# **Etika dan Desain Web Tokopedia berdasarkan BBB online**



**M APRILIANSYAH R**

NIM 182420064

**MEGITER TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BINA DARMA  
PALEMBANG  
2019**

# **Etika dan Desain Web Tokopedia berdasarkan BBB online**

## **Abstrak**

Tujuan dari penulisan ini untuk mengetahui apakah desain web tokopedia memasukkan unsure etis dalam situs webnya. Dalam tulisan ini desain web tokopedia dievaluasi berdasarkan prinsip-prinsip yang ada pada BBBonline. Lima prinsip yang digunakan sebagai instrument evaluasi adalah: (1) Komunikasi yang jujur dan akurat, (2) Transparansi informasi, (3) Penerapan keamanan informasi, (4) Kepuasanpelanggan, dan (5) Melindungianak anak. Temuan penelitian menunjukkan bahwa beberapa prinsip dipatuhi dengan baik, sementara yang lain membutuhkan lebih banyak perhatian.

Kata kunci: etika, *e-commerce*, analisis web, kepercayaan.

## **1. Perkenalan**

Kemunculan e-commerce disebabkan oleh keberadaan Internet dan teknologi komputer. Daya dorong teknologi ini mengakibatkane-commerce menyebar seluruh dunia.Pada saat yang sama, selama lima tahunterakhir, telah terjadi peningkatan signifikan dalam tingkat risiko bisnis baru yang memengaruhi produk, penyedia jasa dan pelanggan. Risiko-risiko ini muncul dari meningkatnya penggunaan Internet dan kekhawatiran pelanggan kepada keamanan, privasi dan pemenuhan layanan saat membeli online. Kekhawatiran inimenjadi masalah serius dan risiko bisnis untuk keberhasilan setiap perusahaan, terutama yang menggunakan teknologi internet. Untuk mengembangkan, memperkuat,mempercepat dan mempertahankan kepercayaan pelanggan dalam *e-commerce* sangat penting bagi perusahaan untuk mengidentifikasi dan mengelola risiko bisnis baru ini secara proaktif.

Makalah ini menyajikan temuan studi tentang desain web tokopedia. Satu dari tujuan dari penelitian ini adalah untuk mengetahui sejauh mana tokopedia menggabungkan etika dan elemen yang dapat dipercaya di situs web mereka. Tinjauan pustaka menyoroti masalah etika dan masalah kepercayaan konsumen dalam kaitannya dengane-commerce, dan tinjauan umum *e-commerce* di Indonesia. Pemilihan situs web dan bagaimana ulasannya yang dilakukan dibahas pada bagian berikut.Akhirnya, temuan penelitian disajikan dan dibahas.

## 2. Tinjauan Pustaka

### 2.1. *E-commerce* dan masalah etis

Secara singkat *e-commerce* didefinisikan sebagai proses jual beli melalui jaringan computer. Kemudian istilah baru e-bisnis muncul, yang membawa definisi yang lebih luas dari *e-commerce*. e-bisnis bukan hanya jual beli barang dan jasa, tetapi juga melayani pelanggan, berkolaborasi dengan mitra bisnis dan melakukan transaksi elektronik dalam suatu organisasi. Kekuatan pendorong di balik berkembangnya *e-commerce* adalah teknologi internet, khususnya *World Wide Web (WWW)*. Namun, menurut Zwass, itu akan sepenuhnya salah menafsirkan *e-commerce* sebagai bagian besar pengembangan teknologi. Cara melakukan bisnis seperti ini dapat dipahami sebagai penyebaran teknologi informasi bersamaan dengan kemajuan organisasi dan manajemen yang memicu teknologi ini berkembang. Munculnya *e-commerce* membawa sejumlah masalah etika seputar privasi pelanggan, risiko, keamanan, dan keandalan. Masalah privasi menjadi banyak perhatian, khususnya di negara berkembang. Teknologi telah berkembang ke titik yang rinci, profil individu yang komprehensif dapat dibuat dan disebarluaskan kepada orang-orang yang belum pernah terjadi sebelumnya. Teknologi telah secara substansial mengubah hubungan antara pelanggan dan pedagang,

Selain masalah privasi, masalah keamanan dikenal sebagai penghalang utama belanja online. Salah satu aspek keamanan terkait potensi kerusakan sumber teknologi dan celah keamanan yang mengatur transaksi *e-commerce*. Proses transaksi online yang lancar dan aman tergantung pada fungsi perangkat keras dan perangkat lunak, serta protokol kriptografi yang digunakan. Aspek lain dari masalah keamanan adalah keamanan informasi, pencurian data dan data yang hilang, dan akses ke data pribadi dan informasi yang rahasia. Menurut Aljifri et al., dalam konteks negara berkembang, keamanan menjadi masalah paling penting dalam pertumbuhan *e-commerce*.

Banyak orang khawatir untuk membeli secara online karena kurangnya kepercayaan. Lohse et al. menemukan bahwa kekhawatiran tentang privasi dan kekurangan kepercayaan online adalah salah satu faktor utama yang menyebabkan pelanggan keluar dari bisnis online. Menurut Chadwick, ada dua tindakan yang dapat diambil oleh perusahaan *e-commerce* untuk mendapatkan kepercayaan konsumen yaitu: desain web dan tindakan membangun kepercayaan. Di bagian depan desain web, Kim dan Moon menemukan bahwa persepsi kepercayaan dapat dimanipulasi dengan memvariasikan desain halaman web, khususnya penggunaan grafis dan warna.

Melihat secara khusus perilaku membangun kepercayaan, sebagian besar analisis membagi isu kepercayaan *e-commerce* menjadi masalah privasi, risiko, keamanan, dan keandalan. Chadwick mengkritik perspektif tersebut, menganggap pembagian itu sebagai sesuatu yang hanya berasal dari aspek *technise-commerce* dan mengabaikan faktor manusia dalam *e-commerce*. Dia menekankan pentingnya mengakui kepercayaan sebagai komponen dari hubungan antara konsumen dan perusahaan *e-commerce*. Ini akan menjadi pertimbangan bagaimana informasi itu diproses dan dijaga agar tetap aman, dengan cara yang memungkinkan pesan perdagangan dibuat dan digunakan untuk mempengaruhi kepercayaan.

## **2.2. BBBonline**

BBB Online adalah sebuah pendekatan tata kelola mandiri yang bekerja untuk mempercepat pertumbuhan industri internet. BBB Online mendorong secara etis praktik bisnis online melalui privasi program dan program keandalan. BBB Online Segel Privasi mengkonfirmasi bahwa perusahaan berdiri di belakangnya kebijakan privasi online dan telah memenuhi program persyaratan tentang penanganan pribadi informasi yang disediakan melalui situs webnya. Itu Segel dalam mengkonfirmasi bahwa perusahaan telah ditinjau untuk memenuhi kebenaran dalam pedoman iklan dan mengikuti praktik layanan pelanggan yang baik. Saat ini, lebih dari 9800 bisnis telah memenuhi syarat untuk menampilkan segel dalam BBB Online; 822 situs web telah mendapatkan segel privasi BBB Online.

Di bawah program keandalan, BBB Online telah membuat kode praktik bisnis online yang bertindak sebagai panduan etis untuk "bisnis kekonsumen" dalam *e-commerce*. BBB Online mengklaim bahwa kode berisi pedoman praktis berbasis kinerja, dari pada mendikte metode untuk mencapai tujuan yang dapat mengganggu model bisnis tertentu. Kode ini dirancang untuk memungkinkan bisnis online mengambil keuntungan dari teknologi yang berkembang dan untuk mendorong inovasi sambil tetap berpegang pada bisnis berprinsip praktik yang memberikan informasi yang benar dan akurat untuk pelanggan online. Studi ini telah mengadopsi kode ini sebagai kerangka kerja untuk mengevaluasi situs web *took* pada perilaku *e-commerce*.

## **3. Metodologi Penelitian**

Ada banyak cara untuk mengevaluasi kualitas situs web. Namun, sebagian besar penelitian terdahulu berfokus pada kualitas umum situs web yang mendorong transaksi dari calon pelanggan. Karena yang utama tujuan dari penelitian ini adalah untuk melihat kualitas situs web dari dimensi etis, kemudian diputuskan berdasarkan kode BBB Online untuk praktik

bisnis online sebagai kerangka kerja untuk menganalisis situs web tokopedia. Penelitian ini melakukan *review* situs web tokopedia. Kriteria utama yang digunakan dalam memilih perusahaan adalah bahwa situs web harus menyediakan mekanisme pembayaran online, atau dengan kata lain, fitur *e-commerce* tersedia. Secara khusus proses *review* dilakukan mengikuti langkah-langkah berikut:

1. Lima prinsip kode praktik BBB Online di analisis dan indikator spesifik diidentifikasi untuk setiap prinsip.
2. Indikator-indikator ini disusun dalam bentuk template.
3. Template digunakan untuk meninjau situs web, mengidentifikasi indikator mana yang tersedia di situs web tersebut.
4. Data yang dikumpulkan dari situs web dikompilasi, dan kemudian dikumpulkan untuk di analisis.

Kode praktik bisnis BBB Online didasarkan pada prinsip-prinsip berikut:

- 1. Komunikasi yang Jujur dan Akurat:** Ini Prinsipnya menyatakan bahwa bisnis online tidak boleh terlibat dalam praktik menipu atau menyesatkan dengan Berkenaan dengan segala aspek perdagangan elektronik, termasuk iklan, pemasaran, atau dalam penggunaannya teknologi. Indikator diidentifikasi untuk ini prinsip, yaitu:
  - Materi iklan yang jujur.
  - Tidak ada penyalahgunaan hyperlink.
  - Penggunaan istilah pencarian yang tepat.
  - Penggunaan segel pihak ketiga fungsional.
  - Non-gangguan browser pelanggan.
- 2. Transparansi:** Menurut BBB Online, prinsip ini berarti bahwa pedagang online harus mengungkapkannya kepada mereka informasi pelanggan dan calon pelanggan tentang bisnis, barang atau layanan yang tersedia untuk pembelian online, dan transaksi itu sendiri. Terkait dengan informasi bisnis, perusahaan harus memberikan rincian seperti nama resmi perusahaan, nama mana di mana ia menjalankan bisnis, perusahaan alamat fisik, metode kontak online, titik kontak yang bertanggung jawab untuk pelanggan pertanyaan, dan nomor telepon atau faksimili. Informasi tentang produk dan layanan harus mencakup semua informasi yang diperlukan agar pelanggan untuk membuat pilihan informasi tentang apakah akan membeli produk atau layanan tersebut. Terkait dengan informasi tentang online transaksi itu sendiri, kode menyatakan online itu

pedagang harus memberikan informasi yang cukup demikian. Bahwa pelanggan dapat membuat pilihan berdasarkan informasi apakah akan terlibat dalam transaksi atau tidak. Itu pedagang juga harus menyediakan pelanggan dengan kesempatan untuk meninjau dan menyetujui transaksi dan menerima konfirmasi transaksi. Untuk ini Prinsipnya, total 25 indikator diidentifikasi yang mencerminkan diskusi di atas.

**3. Praktik dan Keamanan Informasi:** Prinsip ini mengharuskan pedagang online untuk mengadopsi informasi praktik yang memperlakukan informasi pribadi pelanggan dengan hati-hati. Mereka harus memposting dan mematuhi privasi kebijakan berdasarkan prinsip informasi yang adil, ambil langkah-langkah yang tepat untuk memberikan keamanan yang memadai, dan hormati preferensi konsumen mengenai email yang tidak diminta. Tujuh indikator diidentifikasi untuk mewakili prinsip ini, yaitu:

- Aksesibilitas informasi pribadi dikumpulkan.
- Pilihan pelanggan untuk penggunaan bisnis pribadi informasi.
- Tindakan pengamanan yang diambil untuk melindungi informasi.
- Kebijakan pelanggaran.
- Penggunaan standar enkripsi.
- Penggunaan standar keamanan.
- Praktik surat yang tidak diminta.

**4. Kepuasan Pelanggan:**

Prinsip ini ditekankan pada pentingnya memastikan kepuasan pelanggan. Bisnis harus berusaha memastikan pelanggan mereka puas dengan menghormati representasi mereka, menjawab pertanyaan, dan menyelesaikan masalah pelanggan keluhan dan perselisihan secara tepat waktu dan responsif cara. Enam indikator diidentifikasi untuk mencerminkan hal ini prinsip, yaitu:

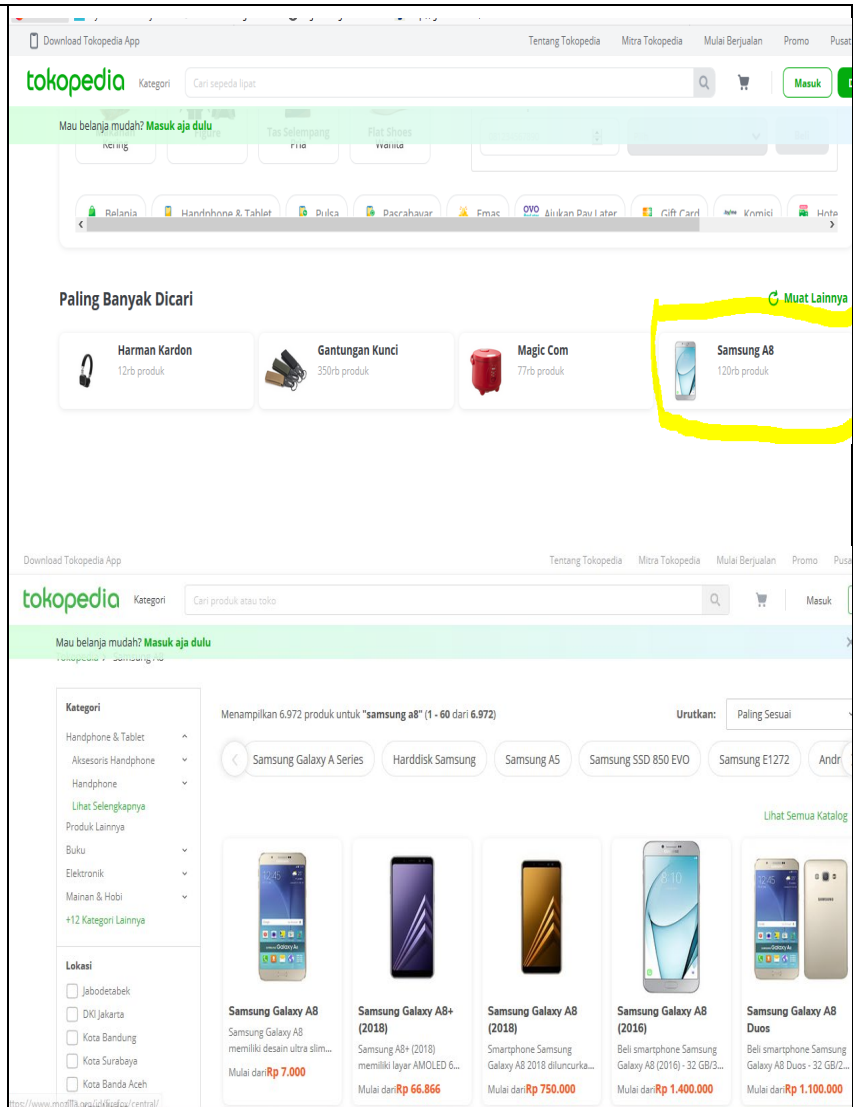
- Deklarasi hukum tentang pemenuhan janji.
- Respons terhadap pertanyaan pelanggan melalui Tanya Jawab.
- Respons terhadap pertanyaan pelanggan melalui email.
- Kontak untuk penyelesaian sengketa.
- Mekanisme internal untuk menyelesaikan perselisihan.
- Resolusi perselisihan pihak ketiga.

**5. Melindungi Anak-Anak:** Prinsip terakhir menyatakan bahwa jika pengiklan online menargetkan anak-anak di bawah umur 13, mereka harus berhati-hati untuk

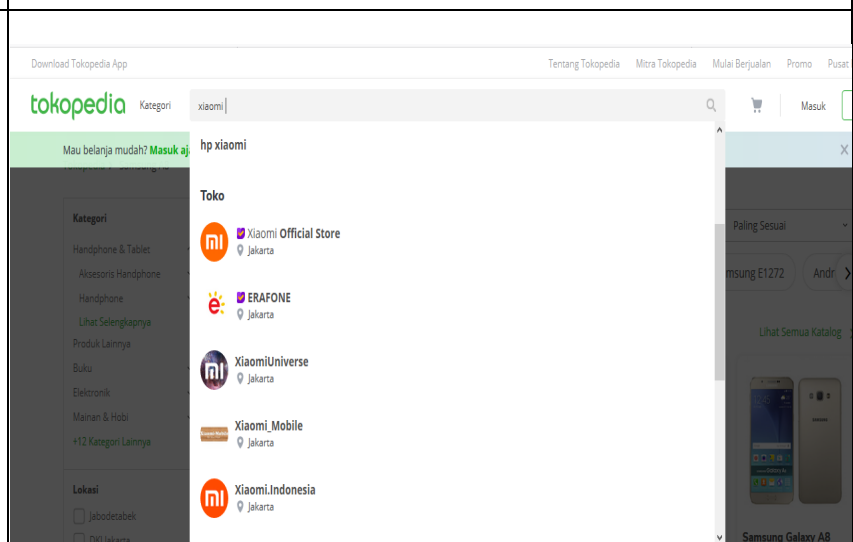
#### 4. Hasil evaluasi web

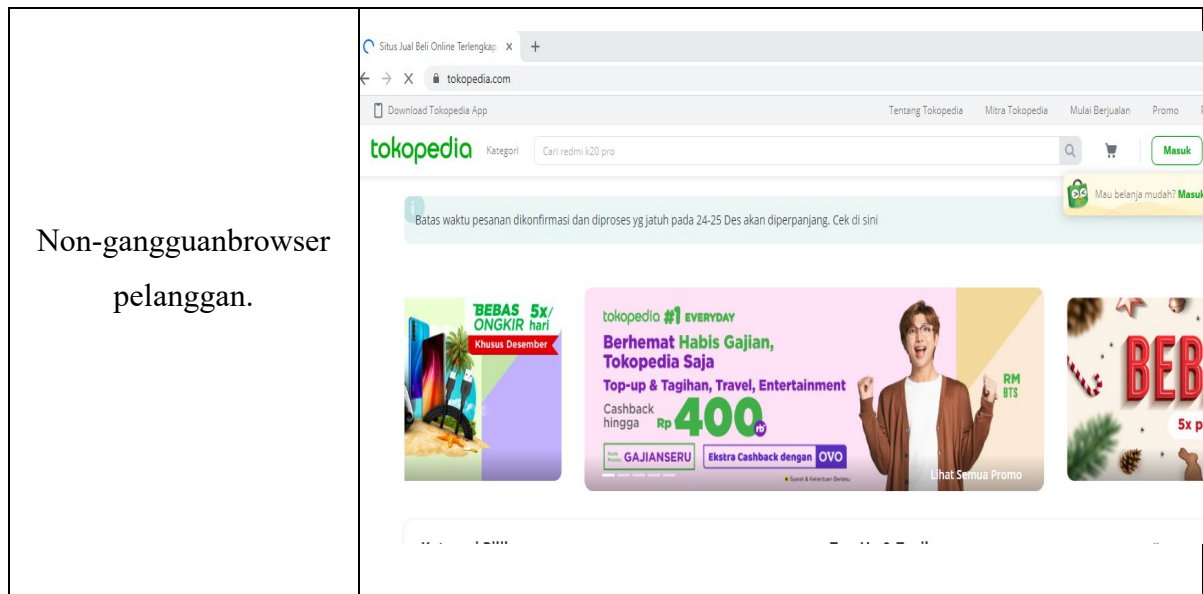
| Prinsip 1                       | Keterangan                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Materi iklan yang jujur.</p> |  <p>The image shows a promotional banner for Tokopedia's 'GAJIAN EKSTRA' campaign. The banner features a man in a brown jacket and white shirt, with the text 'Banyak Tabungan Emas Sekarang' and 'Cashback hingga Rp150rb'. A sidebar on the right contains 'Info Promo' details: 'Periode Promo 25 - 31 Des 2019', 'Minimum Transaksi Rp300.000', and a 'Cek Sekarang' button.</p> <p><b>SYARAT DAN KETENTUAN</b></p> <ol style="list-style-type: none"> <li>Promo berlaku selama periode mulai dari tanggal <b>25 Desember</b> hingga <b>31 Desember 2019</b> (pukul 00.00 – 23.59 WIB)</li> <li>Promo berlaku untuk <b>180 pengguna</b> pertama per harinya. Pengguna akan mendapatkan cashback ke OVO Points sebesar <b>2% hingga Rp150.000</b> dengan melakukan transaksi di Tokopedia Emas dan menggunakan kode promo <b>GAJIANEMAS</b>.</li> <li>Untuk mendapatkan potongan, minimal transaksi adalah <b>Rp300.000</b>.</li> <li>Satu akun Tokopedia dan/atau satu nomor handphone, satu identitas resmi/KTP hanya bisa mengikuti promo ini sebanyak <b>satu kali</b> untuk produk emas dan <b>satu kali</b> untuk produk reksa dana selama masa promo.</li> </ol> <p><a href="#">Read More</a> ✓</p> |

Tidak ada penyalahgunaan  
hyperlink



Penggunaan istilah pencarian yang tepat.





Berdasarkan tabel 1. Tokopedia berupaya menampilkan materi iklan yang benar. Ini sulit untuk memastikan niat pengiklan dalam mendesain iklan, bagaimanapun, selama materi representasi bebas dari penipuan, maka itu dianggap jujur. Tabel 1 menunjukkan secara umum, tokopedia memanfaatkan hyperlink dengan benar. Istilah pencarian terkait dengan konten situs web.

**Tabel 2. Transparansi**

| Prinsip 2                     | Keterangan |           |
|-------------------------------|------------|-----------|
|                               | ada        | tidak ada |
| Nama dan URL perusahaan       | ✓          |           |
| Alamat surat                  | ✓          |           |
| Alamat email                  | ✓          |           |
| Telpon dan No. Faks           | ✓          |           |
| Mekanisme pembayaran          | ✓          |           |
| Prosedur Pengembalian barang  | ✓          |           |
| Biaya pengiriman              | ✓          |           |
| Informasi Pajak               | ✓          |           |
| Informasi Tagihan             | ✓          |           |
| Klaim pengiriman              | ✓          |           |
| Informasi Ketersediaan barang | ✓          |           |
| Ulasan transaksi              | ✓          |           |

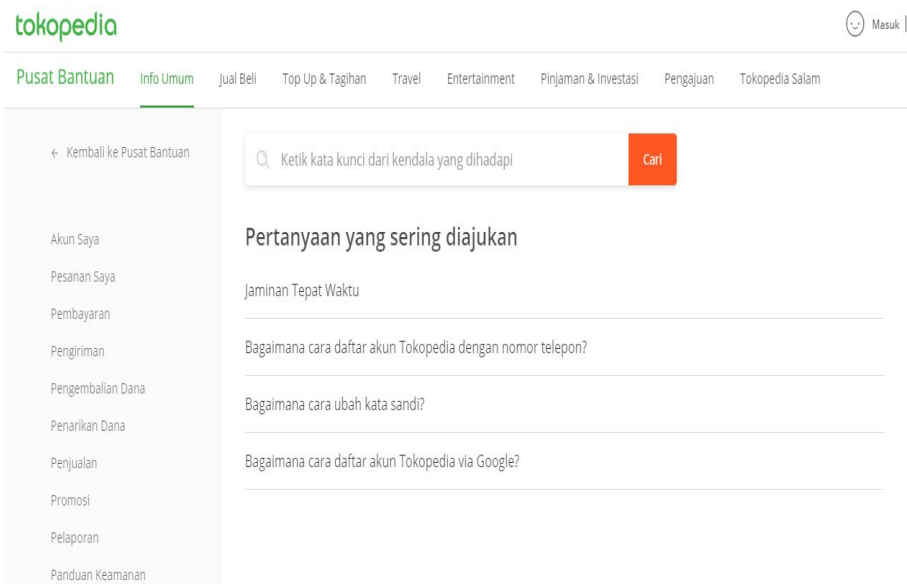
Dari table 2. Terlihat jelas bahwa tokopedia menampilkan semua informasi bisnis.

**Tabel 3. Praktik dan penerapan keamanan informasi**

| Prinsip 3                    | Keterangan                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Penerapan keamanan informasi |  <p>PCI compliant by ControlCase</p> <p>CLIENT NAME: PT Tokopedia<br/>         LEVEL OF CERTIFICATION: Level 1<br/>         PCI CERTIFICATION DATE: Apr-30-2019<br/>         PCI EXPIRATION DATE: Apr-29-2020<br/>         DESCRIPTION: PT Tokopedia has been validated for Payment Card Industry Data Security Standard (PCI DSS) 3.2.1 compliance by ControlCase on the date mentioned in this certificate.</p> <p>DISCLAIMER: ControlCase makes no representation or warranty as to whether PT Tokopedia is secure from either an internal or external attack or whether cardholder data is at risk of being compromised. ControlCase makes no representations company's business activities or operations.</p> |

Untuk system keamanan tokopedia menggunakan PCI compliant dari ControlCase. Control case merupakan perusahaan penyedia jasa untuk keamanan transaksi online yang telah melayani lebih dari 400 perusahaan di 40 negara dunia.

**Tabel 4. Kepuasan pelanggan**

| Prinsip 4   | Keterangan                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Halaman FAQ |  <p>tokopedia</p> <p>Pusat Bantuan Info Umum Jual Beli Top Up &amp; Tagihan Travel Entertainment Pinjaman &amp; Investasi Pengajuan Tokopedia Salam</p> <p>Kembali ke Pusat Bantuan</p> <p>Akun Saya Pesanan Saya Pembayaran Pengiriman Pengembalian Dana Penarikan Dana Penjualan Promosi Pelaporan Panduan Keamanan</p> <p>Pertanyaan yang sering diajukan</p> <p>Jaminan Tepat Waktu</p> <p>Bagaimana cara daftar akun Tokopedia dengan nomor telepon?</p> <p>Bagaimana cara ubah kata sandi?</p> <p>Bagaimana cara daftar akun Tokopedia via Google?</p> |



|  |  |
|--|--|
|  |  |
|--|--|

Dari table diatas tookpedia telah menyediahalaman FAQ, aplikasi chat antar penjual-pembeli, halaman diskusi produk. Tetapi tokopedia tidak menampilkan pedoman perlindungan anak

## 5. Kesimpulan

Tulisan ini mengkaji situs web tokopedia menggunakan kode pedoman praktik BBB Online dan, telah memperoleh informasi yang berguna tentang praktik desain web tokopedia. Dalam tulisan ini ditemukan bahwa beberapa prinsip telah dipenuhi dengan baik sementara yang lain membutuhkan lebih banyak perhatian. Prinsipnya yang mana yang paling ditaati adalah prinsip 2 yang berhubungan dengan transparansi informasi, diikuti oleh prinsip 1 yang ada di Informasi yang benar dan akurat. Itu bukti menunjukkan bahwa prinsip perlindungan anak adalah yang paling tidak ditaati. Meskipun ada beberapa keterbatasan, seperti, hanya menggunakan satu *website* untuk diulas, temuan penelitian ini menjelaskan masalah *e-commerce* dan kesadaran etika serta penerapannya di negara berkembang.

## Daftar Pustaka

[www.tokopedia.com](http://www.tokopedia.com)

BBBOnline, [www.bbbonline.org](http://www.bbbonline.org),

Aljifri H. A., Pons A., and Collins D., "Global E-Commerce: A Framework for Understanding and Overcoming the Trust Barrier," *Information Management and Computer Security*, vol. 11, no.3, pp. 130-138, 2003

Grabner-Kraeuter S., "The Role of Consumers' Trust in Online Shopping," *Journal of Business Ethics*, vol. 39, no. 1 / 2, pp. 43-50, August 2002

OECD, *The Economic and Social Impact of E-Commerce: Preliminary Findings and Research Agenda*, Paris, 1999

Brendon C., "In E-commerce, Customer Trust is No Longer an Option: It is a Requirement for Success," *Annual Quality Congress Proceedings*, pp. 355, 2002

Chadwick S. A., "Communicating Trust in E-Commerce Interactions," *Management Communication Quarterly*, vol. 14, no. 4, pp. 653-658, 2001

Lohse J. L., Bellman S., and Johnson E. J., "Consumer Buying Behavior on the Internet: Findings from Past Data," *Journal of Interaction Marketing*, vol. 14, pp. 15-29, 2000

Scheffelman G. W. and Vinsonhaler J. F., "A Synthesis of Research on the Properties of Effective Internet Commerce Web Sites," *Journal of Computer Information Systems*, pp. 23-49, 2002-2003

Sinnappan S., Carlson J., and Sukunesan B. P., "Website Quality in the Australian Electronic Marketplace: Application of the WebQual™ Instrument," in *Proceedings of the 3<sup>rd</sup> International Conference on Information Technology in Asia*, Sarawak, Malaysia, July 2003

**ETIKA DAN TANGGUNG JAWAB PROFESIONAL DALAM  
PENGEMBAHAN PERAN KETUKAN DI UNIVERSITAS  
INA INSAN**



**oleh:**

**Muhammad Irvai**

**124203**

**Dosen Pengampu: M. Iman Erdiansyah M.M. Ph.D.**

**PROGRAM STUDI MAJISTER TEKNIK INFORMATIKA**

**UNIVERSITAS INADARMA**

**TANGGAL AKADEMIK 2022**

# **ETIKA DAN TANGGUNG JAWAB PROFESIONAL DALAM PEMILIHAN PERAN KETIDAKDIKURSIFAN INSAN**

## **ABSTRAK**

Secara umum para profesional yang bergerak dalam bidang apapun memiliki kode etik dan tanggungjawab dalam menjalankan aktivitas atau pekerjaannya masing-masing. Setiap bidang tentu saja memiliki keunikan sesuai dengan karakteristik pekerjaannya. Hal ini juga berlaku pula dalam dunia teknologi informasi khususnya dalam dunia komputasi, rekayasa perangkat lunak maupun sistem informasi. Seorang Profesional komputasi pada prinsipnya memiliki kewajiban etis bagi klien, atasan, profesional lainnya, dan masyarakat dalam memenuhi tanggung jawab profesional. Kewajiban-kewajiban tersebut dituangkan dalam kode etik, yang dapat digunakan untuk membuat keputusan tentang masalah etika dalam dunia komputasi atau teknologi informasi terkhusus di lingkungan Universitas Bina Insan Lubuklinggau

Kata kunci: *Kode etik, Tanggungjawab profesional*

## **1. PENDAHULUAN**

### **1.1 Latar Belakang**

Dalam periode global saat ini ekonomi, ilmu pengetahuan, teknologi dan informasi memainkan peranan yang sangat penting bagi umat manusia. Kemajuan berbagai teknologi baru yang belum pernah terjadi sebelumnya memberikan pengaruh yang tidak kecil pada pekerjaan dan kehidupan sehari-hari. Dalam berbagai proses yang terjadi dalam era globalisasi saat ini banyak yang menjadi tidak terkendali atau dapat di sebut sebagai periode turbulensi atau periode ketidakpastian. Dalam lingkungan yang penuh dengan kompetisi, pengembangan ilmu pengetahuan dan teknologi untuk kepentingan bisnis juga tak luput dari kondisi ketidakpercayaan para individu yang terlibat didalamnya.

Yang terjadi saat ini dapat digambarkan bagaimana tuntutan banyak perusahaan maupun institusi memiliki pekerja yang profesional dan berkualifikasi tinggi, bagaimana seorang manajer atau teknisi sepatutnya berperilaku ketika dihadapkan dalam situasi yang sulit? Bagaimana seorang karyawan yang senantiasa dituntut untuk dapat mengembangkan produk baru, teknologi, dan bekerja dengan penuh inisiatif dalam kaitannya membuka pasar baru? Bagaimana pula keputusan yang diambil mampu mempengaruhi lingkungan, kinerja dan konsumen. Secara normatif tentu saja, karyawan atau pelaku bisnis bertindak dalam batas-batas hubungan kerja dan berperilaku sesuai dengan gambaran kerja unit dalam organisasi atau bahkan instruksi dari atasan sebagai personal yang memegang tanggung jawab terakhir. Namun dengan kondisi seperti itu, tidak berarti bahwa pegawai atau karyawan dapat terbebas dari tanggung jawab individu atau kolektif.

Prinsip *Responsibility*, dimana setiap individu diharapkan untuk bertindak secara bertanggung jawab terhadap kepentingan manusia. Hal ini tentu saja menimbulkan pertanyaan mengenai perilaku sosial, etika serta tanggung jawab manusia yang terlibat khususnya dalam pengembangan teknologi informasi. Pertanyaan terbesar adalah bagaimana kaitannya etika dan profesionalisme dalam pengembangan teknologi Informasi khususnya dalam bidang rekayasa perangkat lunak (*software engineering*) [4].

Sebagaimana *Engineer* yang lain, seorang *Software Engineer* harus meyakini bahwa pekerjaannya melibatkan tanggungjawab yang lebih luas dari hanya sekedar keterampilan teknis. Sebagaimana diketahui bahwa ranah pengembangan perangkat lunak secara jelas dibatasi oleh hukum lokal, nasional maupun internasional terkhusus di Universitas Bina Insan. Seorang *software engineer* harus memiliki perilaku etis dan moral secara bertanggung jawab jika ingin dihargai sebagai profesional. Dengan demikian, sebagai profesional harus menjunjung tinggi standar normal kejujuran dan integritas. Namun, terdapat batasan dimana standar perilaku seorang *software engineer* yang dapat diterima diluar aturan hukum yang berlaku. Beberapa hal tersebut menyangkut: [1]

1. Kerahasiaan, seorang engineer harus menghormati kerahasiaan atasan maupun klien
2. Seorang engineer menerima pekerjaan sesuai kompetensinya.
3. Seorang engineer harus menyadari hukum setempat yang mengatur penggunaan kekayaan intelektual seperti paten, hak cipta, dll dalam hal ini harus berhati-hati memastikan bahwa properti intelektual pengusaha dan klien dilindungi.
4. Seorang *engineer* Perangkat Lunak sebaiknya tidak menggunakan kemampuannya untuk menyalahgunakan komputer orang lain.

## **1.2 Rumusan Masalah**

Adapun rumusan permasalahan yang dilakukan dalam penelitian ini adalah bagaimana etika dan professional dalam pengembangan perangkat lunak di lingkungan Universitas Bina Insan Lubuklinggau

## **1.3 Tujuan Penelitian**

Tujuan dilakukan proses penelitian ini adalah untuk mengetahui etika dan professional seorang *engineer* dalam pengembangan perangkat lunak di lingkungan Universitas Bina Insan

## **2. Tinjauan Pustaka**

### ***2.1 Profesional***

Profesional berasal dari kata profesi yaitu suatu yang melekat pada seseorang yang memberikan jasa atau keterampilan yang dimilikinya bagi orang lain, maka profesional merupakan sebutan pelakunya[8]. Dalam hal kaitannya dengan Profesional komputasi termasuk didalamnya *desainer hardware, software engineer, database administrator*, analis sistem, dan ilmuwan komputer. Dengan cara bagaimana profesi komputasi menyerupai pekerjaan yang tidak tradisional dianggap sebagai sebuah profesi, seperti tukang pipa, model fashion, dan pramuniaga? Sebuah aktivitas dianggap sebagai sebuah profesi jika memiliki karakteristik khususnya dalam

penguasaan pengetahuan khusus dan keterampilan, sering disebut "body knowledge" yang diperoleh melalui pendidikan formal atau pengalaman praktis.

## **2.2 Etika dan Kode Etik**

Kode etik adalah sistem norma, nilai dan aturan profesional tertulis yang secara tegas menyatakan apa yang benar dan baik dan apa yang tidak benar dan tidak baik bagi profesional. Kode etik menyatakan perbuatan apa yang benar atau salah, perbuatan apa yang harus dilakukan dan apa yang harus dihindari[1]. Tujuan kode etik agar profesional memberikan jasa sebaik-baiknya kepada pemakai atau kliennya. Adanya kode etik akan melindungi perbuatan yang tidak profesional.

Ketaatan tenaga profesional terhadap kode etik merupakan ketaatan naluriah yang telah bersatu dengan pikiran, jiwa dan perilaku tenaga profesional. Jadi ketaatan itu terbentuk dari masing-masing orang bukan karena paksaan. Dengan demikian tenaga profesional merasa bila dia melanggar kode etiknya sendiri maka profesinya akan rusak dan yang rugi adalah dia sendiri. Kode etik disusun oleh organisasi profesi sehingga masing-masing profesi memiliki kode etik tersendiri.

## **2.3 Rekayasa Perangkat Lunak**

Perangkat lunak menurut Roger S. Pressman(1997) adalah [5]

- (1) Perintah (program komputer) yang bila dieksekusi memberikan fungsi dan unjuk kerja seperti yang diinginkan
- (2) Struktur data yang memungkinkan program memanipulasi informasi secara proporsional
- (3) Dan, dokumen yang menggambarkan operasi dan kegunaan program

Berdasarkan hal tersebut dapat dijelaskan bahwa pada prinsipnya perangkat lunak merupakan kumpulan perintah yang dapat digunakan sesuai keinginan dalam hal memanipulasi informasi secara proporsional.

Definisi ini agak sedikit berbeda dengan definisi menurut Ian Sommerville (2004),

Perangkat Lunak adalah[6]

*Computer programs and associated documentation such as requirements, design models and user manuals. Software products may be developed for a particular customer or may be developed for a general market. Software products may be*

- *Generic - developed to be sold to a range of different customers e.g. PC software such as Excel or Word.*
- *Bespoke (custom) - developed for a single customer according to their specification.*

Definisi di atas menjelaskan bahwa perangkat lunak dalam pembuatan dan pengembangannya tergantung pada *customer*-nya dalam hal ini penggunaanya secara umum maupun pengguna khusus. Selain harus memahami definisi dari perangkat lunak, juga harus mengenal karakteristik perangkat lunak yang berbeda dari berbagai hal lain yang dibuat oleh manusia. Manusia merupakan makhluk yang dinamis sehingga perangkat lunak yang dibuat antara manusia yang satu dengan yang lainnya tentu saja akan berbeda sesuai dengan level kreativitas dan inovasinya. Menurut Roger S. Pressman (1997) perangkat lunak lebih merupakan elemen logika dan bukan merupakan elemen fisik. Dengan demikian, perangkat lunak memiliki ciri yang berbeda dari perangkat keras yaitu[5]

- (1) Perangkat lunak dibangun dan dikembangkan, tidak dibuat dalam bentuk klasik
- (2) Perangkat lunak tidak pernah usang
- (3) Sebagian besar perangkat lunak dibuat secara custom-built, serta tidak dapat dirakit dari komponen yang sudah ada

### 3. Pembahasan

Pada implementasinya terdapat banyak peluang bahwa dalam pengembangan perangkat lunak dapat digunakan untuk penyalahgunaan fungsi khususnya dalam dunia kejahatan di lingkungan Universitas Bina Insan Lubuklinggau. Untuk memastikan bahwa perangkat lunak digunakan dengan baik untuk kemaslahatan umat manusia maka ada upaya agar para *software engineer* harus berkomitmen untuk menjadikan profesi ini menguntungkan dan dihormati sehingga diperlukan harus mematuhi standar Kode Etik dan Profesional.

Software engineer berkomitmen untuk membuat analisis, spesifikasi, desain, pengembangan, pengujian dan pemeliharaan perangkat lunak yang menguntungkan dan dihargai sebagai sebuah profesi. Secara umum software engineer harus mematuhi 8(delapan) prinsip berikut[3]:

1. PUBLIC – *Software engineers shall act consistently with the public interest.* Pernyataan ini berarti seorang software engineer harus bertindak secara konsisten dengan kepentingan publik.
2. CLIENT AND EMPLOYER – *Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.* Seorang software engineer harus bertindak sesuai dengan kepentingan terbaik klien dan atasan yang konsisten dengan kepentingan publik.
3. PRODUCT – *Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.* Seorang Software engineer harus memastikan bahwa produk dan modifikasi yang terkait dengan memenuhi standar profesional setinggi mungkin.
4. JUDGMENT – *Software engineers shall maintain integrity and independence in their professional judgment.* Seorang Software engineer harus mempertahankan integritas dan kemandirian dalam penilaian profesional yang dimiliki.
5. MANAGEMENT – *Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software*

*development and maintenance.* Seorang pimpinan dan manajer Software engineering melakukan pendekatan etis kepada manajemen pengembangan perangkat lunak dan pemeliharaan.

6. PROFESSION – *Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.* Seorang Software engineer harus memajukan integritas dan reputasi profesi yang konsisten dengan kepentingan publik.
7. COLLEAGUES – *Software engineers shall be fair to and supportive of their colleagues.* seorang Software engineer harus bersikap adil dan mendukung rekan-rekan atau kolega dalam pekerjaan.
8. SELF – *Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.* Seorang Software engineer harus berpartisipasi dalam pembelajaran seumur hidup tentang praktek profesi yang dimiliki dan akan mempromosikan pendekatan etis untuk profesinya

#### **4. Kesimpulan**

Berdasarkan apa yang telah dijelaskan di atas bahwa fungsi engineer tidak hanya merancang, membuat, merawat dan mengembangkan teknologi yang dikembangkan tetapi pada implementasinya tindakan yang dilakukan dalam hal ini perilaku yang dibangun pun harus mengikuti kode etik yang telah ditetapkan oleh organisasi profesional sesuai dengan bidang yang diembannya. Selain aturan hukum dan kode etik yang harus ditaati disebabkan karena fungsi profesional adalah mengutamakan klien, atasan dan ranah publik yang menyangkut profesi engineer yang bersangkutan

## **. Daftar Pustaka**

- [1] Donald Gotterbarn, 1997 The Professionalization of Software Engineering and its significance for Ethics Education East Tennessee State University Johnson City, Tennessee
- [2] Kamus Besar Bahasa Indonesia tahun 2008 Pusat Bahasa Departemen Pendidikan Nasional Indonesia
- [3] The Institute of Electrical and Electronics Engineers, Inc. And the Association for Computing Machinery, Inc. 1999
- [4] Gotterbarn, D and Miller, K and Rogerson, S "Software Engineering Code of Ethics," SIGCAS Newsletter July 1997.
- [5] Presman, Roger S., (1997) Software Engineering: A Practitioner's Approach. McGraw Hill Book Co.
- [6] Ian Sommerville, (2004).Software Engineering 7th Edition", Addison-Wesley
- [7] [http://en.wikipedia.org/wiki/Professional\\_responsibility](http://en.wikipedia.org/wiki/Professional_responsibility)
- [8] Michael C. Loui, Keith W. Miller, Ethics and Professional Responsibility in Computing.Encyclopedia Of Computer Science

# Etics in Software Development

## 1. Pendahuluan

Seorang *Software Development* adalah orang yang menerapkan prinsip-prinsip rekayasa perangkat lunak dalam mendesain, pengembangan, pengujian, dan evaluasi perangkat lunak dan sistem yang membuat komputer atau apapun yang berisi perangkat lunak. Sebelum pertengahan tahun 1990-an, sebagian besar praktisi perangkat lunak menyebut dirinya programmer atau software developer, tanpa memandang pekerjaan yang sebenarnya. Banyak orang lebih suka menyebut dirinya pemrogrammer dan software developer, karena dapat diterima secara luas, sementara istilah *Software Development* masih dalam perdebatan.

Istilah programmer sering digunakan sebagai istilah yang merujuk kepada mereka yang tidak memiliki tools, keterampilan, pendidikan, atau etika untuk membangun perangkat lunak yang berkualitas baik. Akibatnya, banyak praktisi menyebut diri sendiri sebagai *Software Development* untuk melepaskan diri dari stigma yang melekat pada kata programmer. Di banyak perusahaan, untuk berbagai kategori programmer, nama jabatan programmer atau software developer telah diubah menjadi Software Development. Istilah tersebut menimbulkan kebingungan, karena ada beberapa penolakan, dengan argumentasi bahwa semua orang pada dasarnya melakukan hal yang sama dengan perangkat lunak, sedangkan yang lain menggunakan istilah untuk membuat sebuah perbedaan, dengan argumentasi bahwa pekerjaan tersebut benar-benar berbeda.

Seorang Software Engineer dapat dikatakan sebagai sebuah profesi karena seseorang yang bekerja sebagai *Software Development* haruslah berpengetahuan dan memiliki pengalaman kerja di bidangnya. Julius Hermawan (2003), mencatat dua karakteristik yang dimiliki oleh *Software Development* sehingga pekerjaan tersebut layak disebut sebuah profesi, yaitu :

- Kompetensi

Kompetensi yang dimaksud yaitu sifat yang selalu menuntut professional *Software Development* untuk memperdalam dan memperbaharui pengetahuan dan keterampilannya sesuai tuntutan profesinya.

- Tanggung jawab pribadi

Yang dimaksud yaitu kesadaran untuk membebaskan hasil pekerjaannya sebagai tanggung jawab pribadi.

## **2. Kelompok Bidang**

Agar dapat melaksanakan tugas dan tanggung jawabnya secara baik dan benar, maka seorang *Software Development* perlu terus mengembangkan bidang ilmu dalam pengembangan perangkat lunak seperti misalnya :

- Bidang ilmu metodologi pengembangan perangkat lunak  
Bidang ilmu tersebut mencakup teknik analisa masalah, desain atau perancangan sistem yang ada dan yang akan dibangun, serta implementasi pemrograman dari desain menjadi perangkat lunak siap pakai.
- Manajemen sumber daya  
Bidang ilmu tentang bagaimana merencanakan, mengadakan, mengawasi, dan mengoptimalkan pemanfaatan sumber daya untuk keperluan pengembangan perangkat lunak yang dibangunnya.
- Mengelola kelompok kerja  
Merupakan bidang ilmu manajemen dan organisasi tentang bagaimana melakukan sinergi antar komponen dalam sebuah kelompok kerja untuk mencapai tujuan tertentu.
- Komunikasi  
Merupakan bidang ilmu yang mempelajari teknik komunikasi dan interaksi dengan manusia lain.

Bidang kerja *Software Development* meliputi bidang kerja System Analyst dan Programmer yaitu mencakup semua fase dalam *Software Development* life cycle : analisa sistem, analisa kebutuhan perangkat lunak, perancangan sistem, pembuatan dan perawatan perangkat lunak, ditambah dengan tugas dari manager proyek yaitu perencanaan dan pengelolaan proyek.

## **3. Kode Etik**

Selain kemampuan yang harus dimiliki seorang Software Development, seorang *Software Development* juga harus mempunyai kode etik, dan amat disayangkan di negara kita Indonesia, sejauh ini belum ada kode etik khusus yang ditujukan bagi profesi *Software Development* dan IT (Information Technology) . Sebagai salah satu bidang profesi, *Software Development* dan IT Profesional

bukan pengecualian, diperlukan rambu-rambu tersebut yang mengatur bagaimana para IT profesional ini melakukan kegiatannya. Maka dari itu, Joint team IEEE Computer Society dan ACM telah mengeluarkan kode etik dalam bidang software Development, antara lain harus mengikuti Delapan Prinsip berikut :

- Masyarakat, perekayasa perangkat lunak akan bertindak secara konsisten sesuai dengan kepentingan masyarakat.
- Klien dan Atasan, Perekayasa perangkat lunak akan melakukan yang terbaik bagi klien dan atasan mereka, konsisten dengan kepentingan masyarakat.
- Produk, Perekayasa perangkat lunak akan mejamin bahwa produk mereka dan modifikasi yang mereka lakukan terhadapnya memenuhi standar profesional yang setinggi-tingginya.
- Penilaian, Perekayasa perangkat lunak akan mempertahankan integritas dan independensi penilaian profesional mereka.
- Manajemen, manajer dan pemimpin rekayasa perangkat lunak akan mengikuti dan mempromosikan pendekatan etis terhadap manajemen pengembangan dan pemeliharaan perangkat lunak.
- Profesi, Perekayasa perangkat lunak akan mempertinggi integritas dan reputasi profesinya konsisten dengan kepentingan masyarakat.
- Kolega, Perekayasa perangkat lunak akan bersifat adil dan mendukung terhadap koleganya.
- Diri Sendiri, Perekayasa perangkat lunak akan berpartisipasi dalam pembelajaran seumur hidup mengenai praktek profesi mereka dan akan mempromosikan pendekatan etis terhadap praktek profesi tersebut.

Kode Etik ini dibuat terkait dengan perilaku dan keputusan yang dibuat oleh para Software Development Profesional yang mencakup profesi praktisi, pendidik, manajer, supervisor, pembuat kebijakan dan termasuk trainee dan mahasiswa profesi *Software Development*.

Perekayasa perangkat lunak harus memiliki tanggung jawab etis dan moral jika ingin dihormati sebagai profesional. Terdapat beberapa standar dan kode etik yang harus dipertimbangkan, yaitu :

a. Konfidensialitas

Harus menghormati konfidensialitas atasan dan kliennya walaupun tidak ada persetujuan yang ditanda tangani secara formal.

Kompetensi Tidak boleh menyalahi tingkat kompetensinya (melebihi atau menyimpang)

b. Hak Properti Intelektual

Menyadari terhadap hukum yang mengatur penggunaan properti intelektual, seperti paten, hak cipta dan lain sebagainya.

c. Penyalahgunaan Komputer

Tidak boleh dengan sengaja menyalah gunakan komputer yang nantinya berakibat merugikan orang lain, seperti penyebaran virus, penyadapan dan lain sebagainya.

#### **4. Faktor Yang Mempengaruhi**

Faktor-faktor yang mempengaruhi profesionalisme Software Development, diantaranya :

- Pesatnya perkembangan dan kemajuan teknologi, sehingga seorang Software Developmenting harus profesional jika ingin bertahan di dunia Developmenting dan harus memiliki pengetahuan dan pengalaman di bidangnya .
- Cepatnya kehadiran software-software terbaru, sehingga seorang Software Developmenting harus menciptakan software baru dan berkualitas agar pengguna software merasa puas dengan hasil kinerjanya.
- Meningkatnya kebutuhan terhadap teknologi yang semakin canggih, untuk dapat berpartisipasi dalam kemajuan teknologi seorang Software Developmenting harus berperan aktif atau mengembangkan apa yang menjadi skillnya.
- Aktivitas *Software Development* yang membuat pilihan praktis lain sehingga mempengaruhi orang lain secara signifikan.
- Profesional komputasi sejak dari desain pembangunan dan memainkan alat-alat komputerisasi.
- Komputerisasi yang berpengaruh pada orang lain.
- Pengaruh aturan yang di buat oleh IEEE, ACM, BCS, dan lain-lain.

#### **5. Keterampilan Yang Harus Dimiliki**

Seorang *Software Development* harus mempunyai kemampuan dan keterampilan, diantaranya :

- Menguasai keterampilan dan pengetahuan rekayasa perangkat lunak.
- Mampu bekerja sebagai perorangan dan sebagai bagian dari suatu tim untuk mengembangkan suatu perangkat lunak yang berkualitas.
- Mampu merancang suatu perangkat lunak sesuai dengan sasaran proyek. Batasan biaya, waktu, pengetahuan, sistem yang berjalan dan organisasi.
- Mampu memberikan desain solusi sesuai dengan aplikasi yang dibangun dan mampu mengintegrasikan dengan pendekatan etis, sosial, hukum dan ekonomi.
- Mampu mendemonstrasikan suatu pemahaman akan teori, model, teknik dalam hal identifikasi masalah dan analisa, desain perangkat lunak, pengembangan, implementasi, verifikasi dan dokumentasi.
- Mampu dan paham akan pentingnya negosiasi, kebiasaan bekerja efektif, leadership dan komunikasi baik dengan stakeholders maupun dengan rekan tim pada saat pengembangan software.
- Selalu mempelajari model baru, teknik, teknologi yang muncul serta mengembangkan profesional yang berkelanjutan.

## **6. Kewajiban**

Software Developments memiliki kewajiban untuk mempertimbangkan keselamatan, kesehatan, dan kesejahteraan pengguna mereka. Pengembangan Software adalah proses sosial dan *Software Development* memiliki dua kewajiban untuk berhasil :

- Berdasarkan pada standar teknis
- Kewajiban tanggung jawab sosial kepada mereka yang akan bekerja sama produknya.

Adapun kewajiban lainnya :

- Konsisten pada kepentingan umum.
- Layak pemberi kerja.
- Harus dapat meyakinkan bahwa produk yang dihasilkan dalam kondisi sesuai dengan standart profesi.
- Dalam memberikan penilaian profesionalnya, tidak boleh memihak.

- Mengatur dan menjadi pemimpin yang menjadikan dan mempromosikan pendekatan etika pada manajemen pengembangan software dan pemeliharaannya.
- Adil dan membantu dalam kerja beretika
- Berpartisipasi dalam pembelajaran jangka panjang profesi mereka dan pendekatan etika dan keefisiennya.

## 7. Sikap

Sikap profesionalisme *Software Development*, diantaranya:

- Jujur dan adil.
- Memahami hukum yang terkait.
- Menghargai dan menghindari merugikan pihak lain.
- Menghargai hak milik.

## 8. Kesimpulan

Berdasarkan apa yang telah dijelaskan di atas bahwa fungsi engineer tidak hanya merancang, membuat, merawat dan mengembangkan teknologi yang dikembangkan tetapi pada implementasinya tindakan yang dilakukan dalam hal ini perilaku yang dibangun pun harus mengikuti kode etik yang telah ditetapkan oleh organisasi profesional sesuai dengan bidang yang diembannya. Selain aturan hukum dan kode etik yang harus ditaati disebabkan karena fungsi profesional adalah mengutamakan klien, atasan dan ranah publik yang menyangkut profesi engineer yang bersangkutan.

## 9. Daftar Pustaka

<https://www.coursehero.com/file/p3bqoiuv/Perekayasa-perangkat-lunak-harus-memiliki-tanggung-jawab-etis-dan-moral-jika/>

## ETICS: the international software engineering service for the grid

To cite this article: A D Meglio *et al* 2008 *J. Phys.: Conf. Ser.* **119** 042010

View the [article online](#) for updates and enhancements.

### Related content

- [Multi-platform Automated Software Building and Packaging](#)  
A Abad Rodriguez, V E Gomes Gouveia, D Meneses *et al*.
- [Article I. Multi-platform Automated Software Building and Packaging](#)  
A Abad Rodriguez, V E Gomes Gouveia, D Meneses *et al*.
- [Interoperability framework for communication between processes running on different mobile operating systems](#)  
A Gal, I Filip and F Dragan

### Recent citations

- [Experiences with moving to open source standards for building and packaging](#)  
D H van Dok *et al*
- [Vaidas Giedrimas](#)
- [Testing Component Compatibility in Evolving Configurations](#)  
Ilchul Yoon *et al*



**IOP | ebooks™**

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

# ETICS: the International Software Engineering Service for the Grid

Alberto Di Meglio<sup>1</sup>, Marc-Elia Bégin<sup>1</sup>, Peter Couvares<sup>2</sup>, Elisabetta Ronchieri<sup>3</sup>,  
Eva Takacs<sup>4</sup>

<sup>1</sup>CERN (Switzerland), <sup>2</sup>University of Wisconsin-Madison (USA), <sup>3</sup>INFN CNAF (Italy), <sup>4</sup>4D SOFT Ltd (Hungary)

Corresponding author e-mail: alberto.di.meglio@cern.ch

**Abstract.** The ETICS system is a distributed software configuration, build and test system designed to fulfil the needs of improving the quality, reliability and interoperability of distributed software in general and grid software in particular. The ETICS project is a consortium of five partners (CERN, INFN, Engineering Ingegneria Informatica, 4D Soft and the University of Wisconsin-Madison). The ETICS service consists of a build and test job execution system based on the Metronome software and an integrated set of web services and software engineering tools to design, maintain and control build and test scenarios. The ETICS system allows taking into account complex dependencies among applications and middleware components and provides a rich environment to perform static and dynamic analysis of the software and execute deployment, system and interoperability tests. This paper gives an overview of the system architecture and functionality set and then describes how the EC-funded EGEE, DILIGENT and OMII-Europe projects are using the software engineering services to build, validate and distribute their software. Finally a number of significant use and test cases will be described to show how ETICS can be used in particular to perform interoperability tests of grid middleware using the grid itself.

## 1. Introduction

Several large-scale open-source software projects have to deal with the need to organize complex software life cycle management infrastructures and processes in order to guarantee required levels of quality, interoperability and maintainability. Often these projects have to face resource, skill, time and budget constraints that may lead to the risk of releasing software difficult to deploy, maintain, understand and integrate with other applications. Research projects with limited duration have to focus on developing software of increasing functionality through their lifetime, but cannot always guarantee that the software will still be accessible, maintainable and documented after the conclusion of their mandate. If the projects are additionally geographically and administratively distributed across several organizations, ensuring that components developed by different developers, in different languages, on different platforms and with non homogeneous tools and processes is often a daunting challenge that may lead to software difficult to manage and maintain. Under the pressure of short deadlines and large requirement sets, project managers may have to face the decision of cutting testing and quality assurance verifications, which can ultimately cause delays in the releases or diminished usability of the software due to the excessive number of undetected problems.

Even when functional tests are performed, the nature itself of complex middleware, such as that developed for the grid, the provision of adequate hardware and network resources is a cost that no project but the largest ones can easily afford. When middleware and applications are deployed on tests

or certification testbeds a lot of time is usually spent trying to make middleware suites and applications to interoperate due to the different configuration assumptions and different versions of common libraries.

In this work, we present ETICS [1], an integrated infrastructure for the automated build, configuration, integration and testing (BCIT) of software, discussing its architecture and presenting a number of examples of usage by large grid projects funded under the FP6 EC framework. ETICS aims to support research and development initiatives producing distributed and grid software by integrating existing procedures, tools and resources in a coherent infrastructure and providing an intuitive access point through a web portal and a professionally-managed, multi-platform resource facility based on Grid technologies.

ETICS doesn't replace configuration management tools like CVS [2] or SVN [3] or compilation tools like make [4] or ant [5]. Rather, ETICS provides a continuous integration and testing framework where such tools can be used. The advantage of using ETICS rather than the individual tools is that the specific syntax and usage requirements of each tool are abstracted by the ETICS data model allowing the software developers and maintainers to manage large, complex and distributed code bases with a unique, consistent set of commands and definitions. Other continuous integration and testing tools exists already (notable example are CruiseControl [6] and Maven [7]). However, most existing tools are dedicated to Java developers only and do not support execution of builds and tests 'jobs' on remote resources and local operations only are supported.

This paper is organized as follows. Section 2 describes the system architecture; Section 3 documents the data model, explaining how ETICS represents and uses information. Section 4 describes the different ETICS web applications, while Section 5 describes the command line tools and the APIs; Section 6 presents the Metronome job management system used by ETICS as execution engine; Section 7 outlines the Grid Quality Certification Model, a proposal to standardize the quality assessment of grid and distributed software projects; finally Sections 8 and 9 presents a number of real use cases and the future work foreseen.

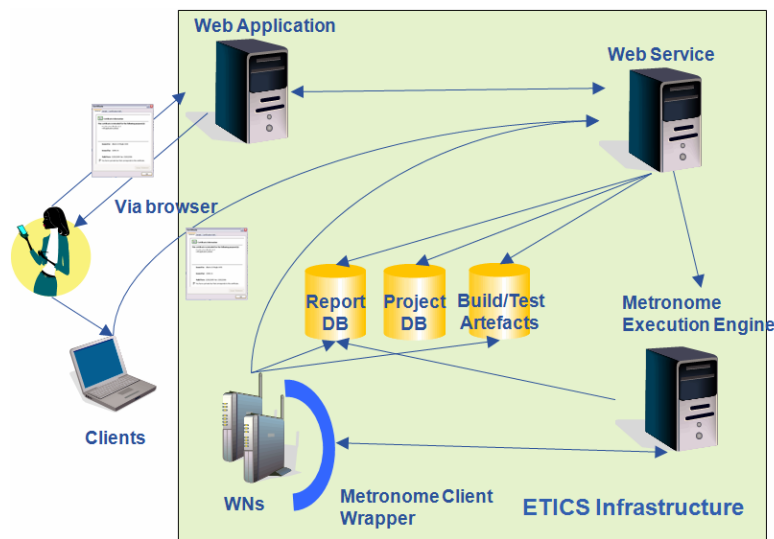
## **2. ETICS Overview and System Architecture**

The ETICS system is being developed by a project partially funded by the European Commission in the context of the FP6 program. It is composed of five partners, CERN (the project coordinator), INFN, Engineering SpA, 4D SOFT Ltd and the University of Wisconsin – Madison. The foreseen duration of the project is two years, from 1<sup>st</sup> January 2006 to 31<sup>st</sup> December 2007. An extension for two more years has been requested as part of the FP7 program.

ETICS aims to establish a distributed and managed infrastructure providing common software engineering tools and processes. Integrated pools of resources and easily accessible interfaces to the build and test tools have therefore to be provided and maintained for running automated builds and test suites. In addition all build and tests operations that can be executed on the remote worker nodes have to be reproducible locally on user computers to allow users to design and verify their testsuites and build configurations.

A centre of exchange of software configuration information and documentation is required in order to allow projects to organize and store metadata about their software and access information about other existing projects. A repository of standard tools, packages, documentation and interoperability information is required so that new and existing projects can easily and rapidly create new build configurations and testsuites.

Taking into account the above mentioned requirements, the ETICS system architecture is split into several entities, as shown on Figure 1: the ETICS Web Service; a number of web applications; a set of command line interfaces and APIs; the data model and its persistent storage implementation; and the job execution engine.



**Figure 1: The ETICS System Architecture**

### 2.1. System Features

The ETICS Build and Tests system comes with a default set of features to build software and test packages and services. The initial set of features can additionally be augmented by adding plugins [8] to the system providing specialized services. ETICS uses a plugin framework with a published interface that allows users or service providers to customize the system behavior as necessary.

The initial set of features comprises a job management engine to automate the execution of build and test jobs with support for a wide range of operating systems, CPU architectures and compilers. The internal data model allows specifying dependencies between packages or tests, which are automatically managed by the build and test engine and compiled or deployed as required. The basic set of functional plugins allows to trigger triggers coding convention checks, unit test execution, documentation generation and to collecting and publish the build and test results to the ETICS online repository.

The ETICS system supports several Version Control Systems (like CVS and SVN) and can be extended to additional systems. It has an internal packaging system that allows developers to generate automatically packages in various formats (e.g., tarballs, RPMS, debs, MSIs) and for different platforms by specifying simple sets of properties. The current version of the ETICS system (v. 2.0.3) supports several operating systems such as Scientific Linux (SLC3, SLC4, SL5), Ubuntu 7, Debian 4, various versions of Red Hat Linux, CentOS, MacOSX and Windows (on a smaller set of features) on both 32 and 64 bit (ia64 and x86\_64) CPUs.

The build engine can be instructed to build large package sets entirely from source or as a combination of source and precompiled binaries and supports strict release processes by providing volatile package repository for the development phases and permanent, non modifiable repositories for release distributions, ensuring the reproducibility of the entire build and test process over time. The compilation architecture is normally automatically detected and the appropriate binaries are downloaded from the repository. However, it is also possible to override the automatic detection specifying an alternative platform in order to perform cross-platform builds. This can be useful for example to run 32-bit builds on 64-bit (x86\_64) platforms (this requires some support from the modules to be built in order to accept the necessary compilation flags).

### 2.2. Authentication and Authorization

The underlying security infrastructure is based on digital certificates [1]. The web applications and the command line client authenticate themselves using standard x.509 [9] certificates. Users are modeled as fully qualified x.509 principal names as they appear in standard x.509-compliant certificates. The ETICS Web Service verifies the user certificate Distinguished Name in the database of existing users

involved in a project, and it allows or denies the operation based on the roles assigned to the users. From that point onwards, the Web Service uses a service certificate to interact with other internal services. The access control list on the persisted data is enforced by the Web Service. The following roles are defined in the system:

| Role                 | Description                                                                            |
|----------------------|----------------------------------------------------------------------------------------|
| Administrator        | A super user enabled to perform all the operations allowed in the ETICS infrastructure |
| Module Administrator | Responsible for handling individual projects                                           |
| Developer            | Works on the implementation of the software and can submit remote builds               |
| Integrator           | Runs remote builds and registers packages in the package repository                    |
| Tester               | Runs remote tests stores test results in the test repository                           |
| Release Manager      | Responsible for defining project level configurations and releases for a project       |
| Guest                | Read-only access to publicly accessible information                                    |

### 3. The Data Model

The data model and the storage are designed based on the principles of industry standard information models like CIM [10] and introduce formal entities to organize the software projects structure, the build configuration, the security information, the build and job result set and so on. The data model describes explicitly the objects and the relationships between objects (as in the CIM model). In addition, the model allows representing the results of running a build and test job in a way that can be consumed by the web application to generate reports.

A software project structure is represented using three main objects: the Project, the Subsystem and the Component. A Project can contain Subsystems or Components and a Subsystem can contain Components. Although the model also allows nested Subsystem, this is not currently implemented. The three structural objects (generically referred as Modules) represent respectively the abstract identity of a project, a subsystem or a test node and a physical package to be built or a test to be executed, but no version information. The Configuration object is instead used to represent the version information of a Module. Every module can have more than one Configuration. A build or test list is composed of the project structure and an associated list of configurations. Not all structural modules in a build or test list needs having an associated configuration, which is equivalent to building or testing subsets of the full structure.

Every Module can be associated to sets of Version Control System Commands, Build Commands, Test Commands, Properties, Environment Variables and Dependencies. The sets are associated to the modules via an intermediate Platform object, which allows specifying separate sets per platform. The Platform object normally represents a particular combination of Operating System, CPU architecture and compilers or interpreters.

Dependencies are used to specify that a configuration has to be built or tested before another one. Dependencies can be static if they constrain the relationship between two named Configurations or dynamic if they specify a relationship between a Configuration and a Module. In the latter case, the Configuration of the Module to be used during a build or test run is resolved at run-time using project-wide information. This method is used to control the dependencies at the project level and enforce consistency across a project.

Following the typical CIM format, the relationships between the various objects are described by relationship objects. For example to associate a set of versions and dependencies to a module ETICS uses:

- One 'component' object containing generic properties of the module (name, license, repository root URL, etc) and a Globally Unique ID (GUID)
- One or more 'configuration' objects containing the version information, version control tags, path on the repository, etc) and a Globally Unique ID (GUID)

- One 'componentConfiguration' object for each 'configuration' object associating the GUID of the module with each GUID of the configurations
- One 'vcsCommand', one 'buildCommand' and one 'testCommand' object per configuration and per platform containing the checkout commands, the build commands (e.g. init, configure, compile, package, publish, etc) and the test commands (e.g. init, configure, test, publish, etc) and the GUIDs
- One or more relationship object to associate each configuration with its commands
- If required, one or more 'property', 'environment' and 'dependency' objects per configuration and per platform (each one with its own GUID)
- One or more relationship object to associate each configuration with its properties, environment variables and dependencies

In addition, the dependencies can be declared 'static' or 'dynamic'. A static dependency is a relationship between two named configurations (e.g. 'webservice v. 1.0.0' depends on 'tomcat v. 5.0.24'). A dynamic dependency is a relationship between a named configuration and a module (e.g. 'webservice v. 1.0.0' depends on 'tomcat'). In the case of dynamic dependencies, the exact configuration to be used during the build is resolved by ETICS during the execution of the build using global project information (e.g. a project mandates that 'tomcat v. 5.0.24' must be used for all components depending on 'tomcat') or using version constraints (e.g. 'tomcat >= 5.0' is specified and the highest available version in the ETICS database satisfying the constraint is used).

The data storage back-end holds the persisted data model in the form of a relational DB implementation based on MySQL [11]. It supports different deployment models. For example, the Web Service, the Web Applications and the DB can be hosted on the same node, but as the number of requests increases with more users using the service, the database can be hosted on a separate node.

## 4. The Web Service and the Web Applications

### 4.1. The ETICS Web Service

The Web Service is the entity providing business logic for the entire system and it is used by the command line client and by all web applications. An important goal of the web service is to abstract the data storage backend, which holds the persisted version of the ETICS data model. For simplicity and better scalability, the web service is stateless. This means that it does not use a stateful web service paradigm, such as Web Services Resource Framework [12], which still has to prove itself in high-availability applications. The ETICS Web Service is developed in Java and runs as a service in a Tomcat [13] container. It provides a set of high level methods to read and write objects from and into the database and a set of specialized methods to generate object factories, manipulate objects or get information about the operations performed by users and the system for auditing and logging purposes. Additionally, the Web Service abstracts the access to the underlying job execution engine presenting a common build and test job interface layer for remote submission.

### 4.2. The Build and Test Web Application

The Build and Test Web Application<sup>1</sup> is responsible for providing a presentation layer to allow the user to browse, create and modify Modules and all associated objects. In addition it provides data forms to submit remote build and test jobs to be executed in the ETICS resource pools. It is a stateful application in order to maintain the security credentials and session information. It is developed in Java and runs as a service in a Tomcat container. The current version uses an open source, but proprietary framework called jduck [14] (developed by one of the ETICS partners, Engineering SpA), but a new version using the more standard Google Web Toolkit [15] is under preparation.

### 4.3. The ETICS Repository

---

<sup>1</sup> <https://etics.cern.ch/eticsPortal>

During the build and test process, both locally on user computers and remotely on the ETICS build nodes, the ETICS client assembles a report including information generated during the build or test such as the commands used to execute the job, high-level information about the module and configuration being built or tested, the execution logs generated during the process and customized reports generated by the metrics collectors or by tools activated during the execution (such as static metrics generators, unit tests engines, compliance checkers, etc). The build and test reports generated during builds and tests performed on the ETICS resource pools are stored in the centralized ETICS Repository and are accessible via the Repository web application<sup>2</sup>. Also in this case the application is developed as a Java web service to be run in a Tomcat.

The ETICS software repository is also the standard location where all the software artifacts are stored (e.g. packages, build and test reports). The repository is composed of two main parts: the volatile storage area and a permanent storage area.

The Volatile storage area contains temporary user-defined storage directories that users can allocate on the repository for their private use. Packages and documents in the Volatile storage areas are always overwritten with the most recent files produced at the end of a build or test run. Files in the volatile areas are deleted after 40 days to make space for new ones.

The permanent (i.e. Registered) storage area is the official location where to store officially released and registered packages and documents. A package is registered when it can be made public (e.g. final version of a configuration). Files in the permanent area are never deleted and never overwritten with newer versions, therefore only the first ever produced instance of a specific package or document is stored.

The ETICS repository is accessible using a Java web application that gives access to both the Volatile and the Registered storage areas of the ETICS repository. The current implementation provides read-only access to the repository and allows downloading packages and documents. A full-featured web application based on digital library concepts is under development and will provide secure read and write access and a set APIs to implement high-level repository solutions and to federate multiple repositories together.

#### *4.4. The ETICS User Administration Application*

Users and roles in ETICS can be managed using the ETICS Administration application<sup>3</sup>, a Java web application for Tomcat. The application can be used to manage user certificate registration and to assign to each user (that is each certificate) one or more access roles. Roles are assigned per project, per module or per configuration and allow managing access to the ETICS system and operation with a very fine grained control.

### **5. The CLIs and APIs**

The Command Line Interfaces (the ETICS Client) provide a similar functionality as the web application and makes use of the same web service interface for simplicity and symmetry. The client is developed in Python and can be used directly by the user on local resources (for example a developer machine) to execute the same build and test operations that are executed by the system in the internal, controlled build nodes. Reproducibility of the results is guaranteed by executing each build or test in independent, isolated workspace where the environment is fully managed by ETICS Client using the information stored in the central database and package repositories.

The client also provides functions to create and manage build and test configurations in the local workspace without using the information in the central database [16]. This functionality allows developers to model their software and try it in different conditions before committing the information in the central database.

### **6. The Metronome Execution Engine**

<sup>2</sup> <https://etics.cern.ch/eticsPortal/#repository>

<sup>3</sup> <https://etics.cern.ch/eticsPortal/#legAdmin> (requires access privileges)

The Metronome Job Execution Engine [17] developed by the University of Wisconsin-Madison allows the ETICS system to fully automate the submission of builds and tests on distributed resource pools. Build and test jobs can be executed on a regular schedule, on a large set of different combinations of operating systems, compilers and CPU architectures. The Metronome engine relies on Condor, a specialized workload management system for computing intensive jobs.

Metronome is in practice used to schedule the deployment of an ETICS CLI client and a set of build or test instructions for the ETICS client to execute. The actual execution of the build or test on the grid worker node is performed by the ETICS client. All information about configurations and dependencies are therefore taken at run-time from the central ETICS database allowing the execution of distributed builds and tests/

The resource pools are currently managed by CERN, INFN and the University of Wisconsin-Madison and collectively provide for public use more than 120 physical build nodes and more than 30 distinct combinations of operating systems, compilers and CPU architectures. Individual projects using ETICS can additionally use dedicated private resources where the build and test jobs can be executed under strict control.

### *6.1. The Co-Scheduling Engine*

The ETICS/Metronome system implements a mechanism to allocate multiple nodes to a test job so that different components of the test (services, clients, test suites, etc) can be executed on separate nodes [8]. This functionality, known as C-Scheduling, allows deploying complex multi-node tests and simulate the interaction of clients, services and applications in realistic situations.

A complex test suite can be modeled in ETICS as a set of nodes and each node can be assigned one more components from the ETICS repository. The components can be middleware services or applications, clients, test scripts and so on. Helper deployment modules are also available to deploy automatically standard services like Tomcat or MySQL.

When a co-scheduled test is submitted to the ETICS system for execution, the test structure as modeled in the ETICS database is analyzed and a job submission file is automatically generated and used to instruct Metronome to pre-allocate the required number of nodes. Once all nodes have been allocated, the test components are deployed on each node using the standard ETICS Client. The different services are started in random order and are then synchronized by using the internal information system. The ETICS client provides a set of APIs to advertise and retrieve properties among the deployed services. The properties can be used inside the deployment scripts or the test scripts as semaphores to synchronize the execution of each service. For example, Service A can initialize and then advertise its IP address, while Service B depending on the Service A starts its initialization and the waits (with a configurable timeout) for the IP address to be advertise before proceeding.

The different test components signal to the system their status and in particular when they have finished executing and are ready for clean up. The system keeps track of the status of each allocated node and once all components have finished executing, the results are sent back to the Metronome submission server and the nodes are cleaned restoring them to the initial conditions for accepting new jobs.

## **7. Examples of usage of ETICS**

All the entities described in the data model can be edited by using the CLIs and the Web Application. In particular the CLIs allow users to script the entire build or test process using simple sets of commands. As an example we consider the use case of a developer who needs to create a new configuration (a version) of a module as part of a larger project, register the new configuration in the ETICS database, build it, edit the configuration to modify information like dependency types or versions and rebuild it. The following set of commands can be used to perform the describe operations:

|                                                                                                        |                                                                                                                       |
|--------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| Set the project to work with and get the project structure                                             | <i>etics-get-project &lt;project&gt;</i>                                                                              |
| Checkout an existing configuration of a module                                                         | <i>etics-checkout -c &lt;configuration&gt; &lt;module&gt;</i>                                                         |
| Build the configuration                                                                                | <i>etics-build &lt;module&gt;</i>                                                                                     |
| Use the existing configuration to create a new configuration                                           | <i>etics-configuration prepare -o configuration.ini &lt;module&gt;</i>                                                |
| Edit the configuration file changing version number, cvs tag, dependencies, etc as needed              | <i>Use any text editor</i>                                                                                            |
| Register the new configuration                                                                         | <i>etics-configuration modify -i configuration.ini</i>                                                                |
| Checkout the new configuration to get any new dependency (use -merge to preserve existing information) | <i>etics-checkout -merge -c &lt;configuration&gt; &lt;module&gt;</i>                                                  |
| Rebuild                                                                                                | <i>etics-build -c &lt;newconfiguration&gt; &lt;module&gt;</i>                                                         |
| Submit a remote build on two different platforms and register the binaries in the ETICS repository     | <i>etics-submit build -register -platforms &lt;platform1,platform2&gt; -c &lt;newconfiguration&gt; &lt;module&gt;</i> |

The configuration.ini file uses a specific format based on the standard INI format to describe all module and configuration properties, the dependencies, the checkout, build and test commands and so on.

## 8. The Grid Quality Certification Model

One of the major goals of the ETICS project is to define a Quality Assurance certification model suitable for assessing and monitoring software quality for grid projects. During its first 18 months of operations, a model has been defined in collaboration with the grid projects using ETICS and proposed for feedback at a number of international events (recently at the EGEE 07 conference in Budapest). The model, which is currently called Grid Quality Certification Model (GQCM) for its main application area, is however suitable for assessing in general the quality of distributed software development projects [18].

The GQCM model position itself between standard QA guidelines like CMMi [19] or ISO/IEC 25000:2005<sup>4</sup> and individual software metrics definitions. The standard models normally define qualitative guidelines for an organization, from where the expected quality of the software product can be inferred, while individual metrics provide quantitative ways of assessing specific, but normally disconnected properties, which have to be considered in specific context to be given meaningful interpretations.

We believe that there is a currently a gap between the standards and the commonly used software metrics, which Grid Quality Certification proposes to fill by defining a practical implementation of the QA guidelines using specific set of metrics and assessment perspectives. In this respect, GQCM is closer to and compatible with ISO 9126<sup>5</sup>, but provides the “how”, not only the “what”. GQCM specifies in particular predefined and coherent sets of metrics and thresholds and ways of interpreting and consolidating them, whereas ISO 9126 leaves to the software development organization to define them.

GQCM and its implementation in the ETICS system provide therefore homogeneous, not subjective and comparable results and a repeatable workflow, extremely valuable especially in contexts where the QA culture or expertise is not strong or too costly to acquire.

<sup>4</sup> ‘Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE’, [http://webstore.iec.ch/preview/info\\_isoiec25000%7Bed1.0%7Den.pdf](http://webstore.iec.ch/preview/info_isoiec25000%7Bed1.0%7Den.pdf)

<sup>5</sup> ISO (1991). *International Standard ISO/IEC 9126. Information technology -- Software product evaluation -- Quality characteristics and guidelines for their use*, International Organization for Standardization, International Electrotechnical Commission, Geneva.

GQCM is structured in families or Evaluation Modules (EM). Each EM groups a set of QA evaluation techniques in a specific QA area: Coding style, Static analysis, Structural testing, Functional testing and Standards compliance. Each EM is associated to a one or more metrics within the QA area, which QA systems implementing the model must collect using the tools of their choice. Additionally, the metrics results can be grouped in the three points of view or 'perspectives', Quality, Platforms and Standards.

The perspectives are independent, contrary to other waterfall models and provide a partial score that contributes to produce the overall QA score of the software under assessment. Individual projects are also free to focus the Quality perspective alone or in combination with another perspective in case for example multi-platform support or compliance to standards is not applicable or desired. The overall score allows monitoring the progress of the software over time, taking corrective actions in deficient areas and comparing the status of different software systems.

## 9. ETICS Build and Test System Application Examples

### 9.1. EGEE/gLite

The EGEE project<sup>6</sup>, co-funded funded by the European Commission and coordinated by CERN, brings together experts from over 32 countries with the common aim of building on recent advances in Grid technology and developing a service Grid infrastructure which is available to scientists 24 hours-a-day. The project provides researchers in academia and industry with access to a production level Grid infrastructure, independent of their geographic location.

The EGEE project also produces a grid middleware distribution called gLite, which is born from the collaborative efforts of more than 80 people in 12 different academic and industrial research centers. gLite provides a bleeding-edge, best-of-breed framework for building grid applications.

The gLite development process has been standardized on the ETICS system for building the software and testing the deployment of the service metapackages (such as the User Interface, the Worker Node, the WMS Server and the CREAM Computing Element Server). The entire gLite distribution is composed of about 580 distinct packages and almost 3000 configurations for a total of more than 1.6 million lines of code in C/C++, Java, Python and other languages.

| gLite       |     |                |      |              |       |
|-------------|-----|----------------|------|--------------|-------|
| Modules     | 579 | Configurations | 2932 | Dependencies | 11179 |
| - Project   | 1   | - Project      | 8    | - Internals  | 6414  |
| - Subsystem | 39  | - Subsystem    | 320  | - Externals  | 4764  |
| - Component | 539 | - Component    | 2604 | - Others     | 1     |

Before moving to the ETICS system, the gLite distribution was only developed on a single platform (CERN Scientific Linux 3 32bit), while it is now automatically built on many different platforms several times per day (CERN Scientific Linux 3 32bit, CERN Scientific Linux 4 32bit and 64bit, Scientific Linux 5, Red Hat Enterprise Server 4, Debian 4, Ubuntu 7, CentOS 4). Some experimental builds on subsets of the gLite components have been performed on MacOSX (Leopard). More recently a new project called Grid4Win (coordinated by INFN in Italy and sponsored by Microsoft) has adopted ETICS as build system in the effort of porting some of the gLite components to Windows using the Unix Services for Windows on Windows Server 2008.

### 9.2. DILIGENT

The main objective of DILIGENT is to create an advanced testbed that will allow members of dynamic virtual e-Science organizations to access shared knowledge and to collaborate in a secure, coordinated, dynamic and cost-effective way [20]. This testbed has integrated Grid and Digital Library (DL) technologies, where regression test activity is performed. It has been validated by two

<sup>6</sup> EGEE Middleware Architecture, August 2004, <https://edms.cern.ch/file7476451/1.0/Architecture.pdf>

complementary real-life application scenarios: the former comes from the culture heritage domain, the latter from the environmental e-Science domain.

DILIGENT is a Java based system that uses the Web Service Core framework of the Globus Toolkit comprised of approximately 90 components, which are Web Services, libraries, port-lets, and wrappers of gLite components, developed by several teams across Europe.

| Diligent    |     |                |      |              |      |
|-------------|-----|----------------|------|--------------|------|
| Modules     | 581 | Configurations | 1688 | Dependencies | 5326 |
| - Project   | 1   | - Project      | 14   | - Internals  | 5024 |
| - Subsystem | 27  | - Subsystem    | 103  | - Externals  | 302  |
| - Component | 553 | - Component    | 1571 | - Others     | 0    |

The DILIGENT project has started using ETICS for building and testing remotely through Metronome. The current testing environment of the DILIGENT system consists of six worker nodes (i.e., virtual machines). ETICS local and remote testing facilities assist the DILIGENT testing process by executing remote regression tests and applying the static analysis and metric calculation utilities - being included as 'plugins' - in the Test and Metrics Framework of ETICS in an automated and straightforward way.

### 9.3. OMII-Europe

The OMII-Europe project<sup>7</sup> aims to bring together the best technologies from Europe and elsewhere and make them available in a usable and supported form to scientists. It delivers production quality middleware relevant to large-scale and smaller collaborative Grids by federating development activities across Europe. It also maximizes benefits of global collaborations with USA and China. Specific actions performed by this project covers for example the assessment of the QA process, the re-engineering of the adaptations to standards, the setting up of a central repository for software, documentation and training.

OMII-Europe uses ETICS to build and test its re-engineered middleware components on different platforms, the same supported by the original providers, and to perform deployment tests of its services. In addition, this project has to check interoperability between job submission engines by validating their conformance to the OGSA-BES recommendation. In order to do so, test suites have been developed in order to interpret and parse the recommendation into specific service calls, together with independent clients to perform tests. ETICS has therefore been adopted to perform this test automatically as part of the QA practices. OMII-Europe has a dedicated ETICS installation at the University of Southampton and also uses the public distributed testbed.

### 9.4. EUChinaGrid/EGEE/ETICS IPv6 Collaboration

EUChina<sup>8</sup>, EGEE and ETICS are collaborating on the testing of IPv6 compliance of a number of component in the EGEE/gLite distribution. Each partner in this collaboration focuses on a specific task: EUChina has defined IPv6 code compliance tools and metrics to perform static compliance analysis; EGEE has worked on the porting of some gLite components to IPv6 in both pure IPv6 and dual stack configurations; finally ETICS has contributed providing the tools and resources to automated the IPv6 tests across three different sites (CERN in Switzerland, GARR in Italy and IN2P3 in Paris) and to produce full compliance reports of all software registered in the system [8].

## 10. Conclusions and Future Work

<sup>7</sup> OMII-Europe, <http://omii-europe.org/OMII-Europe>

<sup>8</sup> EUChina Grid, <http://roc.euchinagrid.org/home.php>

In this paper, we presented ETICS, a system that provides services and tools for automatic building, configuration and testing of distributed and grid software. Then, we provided an overview of the ETICS architecture split into several entities like the ETICS Web Service going in details for each of them. We also detailed one of the major goals of the ETICS project that consists of defining a Quality Assurance certification model for grid and distributed software development projects. Finally, we described examples of ETICS build and test applications, such as EGEE, DILIGENT, OMII-Europe and EUChinaGrid, also involved during the system development.

We believe that the ETICS system has potentialities in solving many of the problems faced by open-source software projects during their life cycle management processes, in particular guaranteeing high levels of quality, interoperability and maintainability. Therefore, we consider this project only the beginning of a large one. We intend to add extra plugins to the ETICS system in order not only to evaluate quality and interoperability metrics but also to perform complex tests like the installation of services. We also plan to improve the current ETICS performances considering users' feedback and to add further features accordingly to the needs of the ETICS community.

## References

- [1] Bégin M-E, Diez-Andino Sancho G, Di Meglio A, Ferro E, Ronchieri E, Selmi M and Zurek M 2007 *Springer Verlag Lecture Notes in Computer Science (LNCS) Series, LNCS 4401* pp 81-97
- [2] Purdy G N 2000 *CVS Pocket Reference* (O'Reilly)
- [3] Mason M 2005 *Pragmatic Version Control Using Subversion [Illustrated] (Paperback)* (Pragmatic Bookshelf)
- [4] Talbott S 1991 *Managing Projects with Make* (2nd Ed. O'Reilly & Associates, Inc.)
- [5] Hatcher E and Loughran S 2002 *Java Development with Ant [Illustrated] (Paperback)* (Manning Publications)
- [6] Trueman T 2004 *Cruise Control (Hardcover)* (HarperTeen)
- [7] Massol V and O'Brien T 2005 *Maven: A Developer's Notebook (Developer's Notebooks) [ILLUSTRATED] (Paperback)* (O'Reilly Media, Inc.)
- [8] Bégin M-E, Couvares P, Diez-Andino Sancho G, Da Ronco S, Di Meglio A, Dini L, Fabriani P, Gietz B, Pavlos A, Ronchieri E, Selmi M, Takacs E and Zurek M 2007 *Tenth World Conference on Integrated Design & Process Technology* (Antalya, Turkey)
- [9] Gutmann P 2000 *X.509 Style Guide*  
(<http://www.cs.auckland.ac.nz/~pgut001/pubs/x509guide.txt>)
- [10] McMorran A, Ault G, Morgan C, Elders I and McDonald J 2006 *Power Engineering Society General Meeting 2006* (IEEE)
- [11] DuBois P 2005 *MySQL (Third Edition)* (Sams Developer's Library)
- [12] Foster I, Frey J, Graham S, Tuecke S, Czajkowski K, Ferguson D, Leymann F, Nally M, Storey T, and Weerawaranna S 2004 *Globus Alliance*
- [13] Brittain J and Darwin I F 2003 *Tomcat: The Definitive Guide [ILLUSTRATED] (Paperback)* (O'Reilly Media, Inc.)
- [14] Godfrey M and Grossman D 1999 *Thirtieth SIGCSE technical symposium on Computer science education*
- [15] Chaganti P 2007 *Google Web Toolkit: GWT Java Ajax Programming (Paperback)* (Packt Publishing)
- [16] Bégin M-E, Da Ronco S, Diez-Andino Sancho G, Gentilini M, Ronchieri E and Selmi M 2007 *International Conference on Computing in High Energy and Nuclear Physics(CHEP)*
- [17] Pavlo A, Couvares P, Gietzel R, Karp A, Alderman I D, Livny M and Bacon C 2006 *LISA06: Twentieth Systems Administration Conference* (Washington DC, USA) pp 263-273
- [18] Rippa A, Bégin M-E, Di Meglio A and Manieri A 2007 *Third Conference of the EELA Project*
- [19] Chrissis M B, Konrad M and Shrum S 2003 *CMMI – Guidelines for Process Integration and Product Improvement (Hardcover)* (Addison-Wesley Professional)
- [20] Castelli D, Candela L, Pagano P and Simi M 2005 *2<sup>nd</sup> IEEE – CS International Symposium Global Data Interoperability* (IEEE Computer Society) pp 56-99

## **Tugas Kelompok Etics in Software Development**

**Nama :**

**1. Indri Endang Lestari**

**2. Rifan Fadilah**

**3. Sulistiyani**

**4. Zulius Akbar Amin**

**Kelas : MTI 19AR2**

### **Etika Dalam Pengembangan Perangkat Lunak**

Kepercayaan pada etika Software Engineering kini telah menjadi bagian penting dari budaya kontemporer. Untuk mengambil contoh misalnya, ahli radiologi untuk memverifikasi kebenaran kode yang mengendalikan mesin X-ray yang menunjuk ke arahnya kepala. Pada titik tertentu, tidak ada pilihan selain mengasumsikan bahwa perangkat lunak dibuat dan diinstal di berbagai sistem yang diterapkan tidak memiliki jaminan bahwa telah terimplementasikan dengan benar, dan dengan cara yang konsisten dengan kepentingan publik. Seorang Software Engineer harus berkomitmen untuk membuat analisis, spesifikasi, desain, pengembangan, pengujian dan pemeliharaan perangkat lunak profesi yang bermanfaat dan dihormati. Seorang Software Engineer harus mematuhi Delapan Prinsip berikut:

**1. Publik**

Seorang Software Engineer harus bertindak konsisten dengan kepentingan publik.

**2. Client dan Pegawai**

Seorang Software Engineer harus mampu memberikan tindakan yang terbaik untuk kepentingan Client ataupun Atasan mereka secara konsisten dengan kepentingan publik.

**3. Produk**

Seorang Software Engineer harus memastikan bahwa produk mereka dan modifikasi terkait, memenuhi standar profesional setinggi mungkin.

**4. Judgment**

Seorang Software Engineer harus menjaga integritas dan kemandirian dalam penilaian profesional mereka.

## 5. Manajemen

Manajer dan pemimpin pengembangan perangkat lunak harus bisa membagikan dan memsosialkan suatu pendekatan etis untuk manajemen sebuah pengembangan perangkat lunak dan pemeliharaannya.

## 6. Profesi

Seorang Software Engineer harus meningkatkan integritas dan reputasi profesi yang konsisten dengan kepentingan publik.

## 7. Kolega

Seorang Software Engineer harus adil dan mendukung rekan kerja mereka.

## 8. Diri Sendiri

Seorang Software Engineer harus mau untuk terus belajar sepanjang hayat yang berkaitan dengan praktik profesinya dan harus mempromosikan pendekatan etika terhadap praktik profesi.

### **1. Etika umum**

Etika umum merupakan etika yang bersifat general dan berlaku bagi semua orang, berbeda dengan etika khusus yang hanya berlaku pada satu perusahaan atau kelompok, etika umum mengikat semua orang baik yang ada dikelompok A maupun di kelompok B. Etika umum seorang software engineer merupakan aturan dasar atau pegangan tentang bagaimana seorang software engineer sebaiknya bertindak atau mengambil keputusan.

Adapula kode etik yang harus diikuti seorang software engineer yaitu :

1. Seorang software engineer harus memperhatikan kepentingan umum dalam pembuatan software.
2. Bisa mempertanggung jawabkan secara penuh apapun yang mereka kerjakan atau mereka buat.
3. Software yang dibuat harus berguna dan bermanfaat bagi masyarakat dan tidak memiliki dampak negatif yang dapat merugikan.
4. Adil dan tidak boleh memalsukan informasi mengenai program aplikasi yang dibuatnya.
5. Memberikan yang terbaik bagi client atau orang yang memberikan pekerjaan tetapi harus tetap menjaga dan memperhatikan kepentingan umum.

6. Tidak boleh menggunakan perangkat lunak secara ilegal atau tanpa izin dari pembuat atau orang yang memiliki hak cipta terhadap perangkat lunak tersebut.
7. Hanya menerima pekerjaan yang sesuai dengan bidang yang dikuasainya.
8. Dapat menjaga kerahasiaan informasi apapun yang didapatkan dari perangkat lunak yang dibuatnya.
9. Melaporkan masalah yang dapat merugikan masyarakat umum yang terdapat dalam perangkat lunak atau dokumen tentang perangkat lunak tersebut pada klien atau yang memberikan pekerjaan
10. Harus memperhatikan perangkat lunak yang dibuatnya, apakah sudah memenuhi standar atau belum.
11. Tidak membuat bug atau kegagalan program secara sengaja agar mendapatkan keuntungan lebih
12. Tidak sebaiknya menggunakan keterampilan yang dimiliki untuk melakukan cracking/hacking atau apapun yang dapat merugikan orang lain.
13. Tidak boleh membuat dokumentasi mengenai perangkat lunak dengan asal asal dan tidak akurat.

## **2. Etika khusus**

Selain etika ĩ etika umum yang berlaku bagi seorang software engineer, Google sebagai perusahaan berbasis dunia juga memiliki etika khusus yang berlaku bagi para pegawainya. Etika ini beragam mulai dari etika yang berhubungan dengan kepentingan google sebagai perusahaan, hingga etika yang berhubungan dengan pelanggan. Berikut merupakan etika yang perlu diikuti para pegawai google

### **1. Melayani pengguna**

google merupakan perusahaan yang memberikan banyak layanan berbasis data, dari e-mail, penyimpanan file, search engine dll, sehingga hal yang paling berharga bagi google merupakan kepercayaan pelanggan, oleh sebab itu, pegawai google diharapkan selalu mengambil tindakan yang menjaga dan meningkatkan kepercayaan pelanggan pada google. Hal ini dilakukan dengan sebisa mungkin menjaga privasi dan kebebasan berpendapat pengguna serta keamanan data dari pihak tak ber-hak, sekali pun dihadapan intervensi pemerintah.

Pegawai google juga diharapkan untuk selalu berusaha meningkatkan daya guna google dan kepuasan pengguna dengan bersikap responsive terhadap tanggapan pengguna baik pertanyaan maupun masalah,serta memberikan ide- ide untuk meningkatkan kualitas layanan google.

2. Saling menopang satu sama lain

Pegawai diterima dengan dasar pencapaian dan kompetensi professional, sehingga dilarang keras melakukan diskriminasi terhadap ras, warna, agama, kenegaraan, keturunan, jenis kelamin dan karakteristik lain yang dilindungi hukum.

Pegawai diharapkan untuk menjaga kesehatan dan keamanan selama bekerja, sehingga dilarang menggunakan obat obatan terlarang, selain itu, pegawai juga tak diperbolehkan melakukan kekerasan atau pun membawa senjata di tempat kerja.

3. Menghindari konflik kepentingan

Pegawai google diharapkan untuk menghindari keadaan yang dapat menyebabkan konflik kepentingan dengan google, seperti berinvestasi dan memiliki bagian pada perusahaan kompetitor google, menduduki jabatan anggota dewan pada perusahaan lain, mengambil kesempatan bisnis yang didapatkan dari google tanpa izin, atau pun menerima hadiah dengan nilai signifikan dari perusahaan kompetitor.

4. Data rahasia

Google sering kali membocorkan informasi mengenai inovasi terbaru dalam konferensi pers, namun membocorkan informasi tertentu perusahaan dapat menyebabkan kerugian perusahaan pada peluncuran produk, atau pun menghilangkan keunggulan kompetitif, sehingga pegawai diharapkan untuk mengenali, menjaga, maupun menghancurkan (saat sewajarnya) informasi sensitive google maupun perusahaan partner.

Google menghormati competitor dan ingin bersaing secara adil, sehingga seperti google tak menginginkan mantan pegawai google menyebarkan informasi sensitive pada perusahaan saingan, google juga mengharapakan pegawai google tak memanfaatkan informasi sensitive kompetitor.

5. Melindungi aset google

Merek dagang, logo, hak cipta dan rahasia dagang merupakan salah satu aset google yang paling berharga. Menggunakannya secara tak sah dapat menyebabkan kerugian besar, sehingga karyawan tak diperbolehkan untuk menggunakan logo, merek dagang,

informasi dan properti google tanpa persetujuan terlebih dahulu dari tim marketing. Selain itu pegawai juga tak diperbolehkan menggunakan properti intelektual perusahaan lain tanpa izin.

6. Menggunakan uang google

Uang yang digunakan untuk kegiatan operational google merupakan milik google dan pemegang saham, oleh sebab itu, pegawai diharapkan menggunakan uang dengan bijaksana dengan jumlah yang masuk akal. Pegawai juga diharapkan untuk mencatat dengan jelas semua pengeluaran yang ditanggung oleh google.

Daftar Pustaka

<https://abc.xyz/investor/other/google-code-of-conduct.html>

<https://sites.google.com/a/email.vccs.edu/bus100mawan/google-code-of-ethics>

<https://www.computer.org/web/education/code-of-ethics>

# Etics in Software Development

Kelompok 1 : Indri Endang Lestari, Sulistyawati, Rifan Fadilah, Zulus Akbar Amin

## 1. Pendahuluan

Seorang *Software Development* adalah orang yang menerapkan prinsip-prinsip rekayasa perangkat lunak dalam mendesain, pengembangan, pengujian, dan evaluasi perangkat lunak dan sistem yang membuat komputer atau apapun yang berisi perangkat lunak. Sebelum pertengahan tahun 1990-an, sebagian besar praktisi perangkat lunak menyebut dirinya programmer atau software developer, tanpa memandang pekerjaan yang sebenarnya. Banyak orang lebih suka menyebut dirinya pemrogrammer dan software developer, karena dapat diterima secara luas, sementara istilah *Software Development* masih dalam perdebatan.

Istilah programmer sering digunakan sebagai istilah yang merujuk kepada mereka yang tidak memiliki tools, keterampilan, pendidikan, atau etika untuk membangun perangkat lunak yang berkualitas baik. Akibatnya, banyak praktisi menyebut diri sendiri sebagai *Software Development* untuk melepaskan diri dari stigma yang melekat pada kata programmer. Di banyak perusahaan, untuk berbagai kategori programmer, nama jabatan programmer atau software developer telah diubah menjadi Software Development. Istilah tersebut menimbulkan kebingungan, karena ada beberapa penolakan, dengan argumentasi bahwa semua orang pada dasarnya melakukan hal yang sama dengan perangkat lunak, sedangkan yang lain menggunakan istilah untuk membuat sebuah perbedaan, dengan argumentasi bahwa pekerjaan tersebut benar-benar berbeda.

Seorang Software Enginer dapat dikatakan sebagai sebuah profesi karena seseorang yang bekerja sebagai *Software Development* haruslah berpengetahuan dan memiliki pengalaman kerja di bidangnya. Julius Hermawan (2003), mencatat dua karakteristik yang dimiliki oleh *Software Development* sehingga pekerjaan tersebut layak disebut sebuah profesi, yaitu :

- Kompetensi

Kompetensi yang dimaksud yaitu sifat yang selalu menuntut professional *Software Development* untuk memperdalam dan memperbaharui pengetahuan dan keterampilannya sesuai tuntutan profesinya.

- Tanggung jawab pribadi  
Yang dimaksud yaitu kesadaran untuk membebankan hasil pekerjaannya sebagai tanggung jawab pribadi.

## 2. Kelompok Bidang

Agar dapat melaksanakan tugas dan tanggung jawabnya secara baik dan benar, maka seorang *Software Development* perlu terus mengembangkan bidang ilmu dalam pengembangan perangkat lunak seperti misalnya :

- Bidang ilmu metodologi pengembangan perangkat lunak  
Bidang ilmu tersebut mencakup teknik analisa masalah, desain atau perancangan sistem yang ada dan yang akan dibangun, serta implementasi pemrograman dari disain menjadi perangkat lunak siap pakai.
- Manajemen sumber daya  
Bidang ilmu tentang bagaimana merencanakan, mengadakan, mengawasi, dan mengoptimalkan pemanfaatan sumber daya untuk keperluan pengembangan perangkat lunak yang dibangunnya.
- Mengelola kelompok kerja  
Merupakan bidang ilmu manajemen dan organisasi tentang bagaimana melakukan sinergi antar komponen dalam sebuah kelompok kerja untuk mencapai tujuan tertentu.
- Komunikasi  
Merupakan bidang ilmu yang mempelajari teknik komunikasi dan interaksi dengan manusia lain.

Bidang kerja *Software Development* meliputi bidang kerja System Analyst dan Programmer yaitu mencakup semua fase dalam *Software Development* life cycle : analisa sistem, analisa kebutuhan perangkat lunak, perancangan sistem, pembuatan dan perawatan perangkat lunak, ditambah dengan tugas dari manager proyek yaitu perencanaan dan pengelolaan proyek.

## 3. Kode Etik

Selain kemampuan yang harus dimiliki seorang *Software Development*, seorang *Software Development* juga harus mempunyai kode etik, dan amat disayangkan di negara kita Indonesia, sejauh ini belum ada kode etik khusus yang ditujukan bagi profesi *Software Development* dan IT (Information Technology) .

Sebagai salah satu bidang profesi, *Software Development* dan IT Profesional bukan pengecualian, diperlukan rambu-rambu tersebut yang mengatur bagaimana para IT profesional ini melakukan kegiatannya. Maka dari itu, Joint team IEEE Computer Society dan ACM telah mengeluarkan kode etik dalam bidang software Development, antara lain harus mengikuti Delapan Prinsip berikut :

- Masyarakat, perekayasa perangkat lunak akan bertindak secara konsisten sesuai dengan kepentingan masyarakat.
- Klien dan Atasan, Perekayasa perangkat lunak akan melakukan yang terbaik bagi klien dan atasan mereka, konsisten dengan kepentingan masyarakat.
- Produk, Perekayasa perangkat lunak akan mejamin bahwa produk mereka dan modifikasi yang mereka lakukan terhadapnya memenuhi standar profesional yang setinggi-tingginya.
- Penilaian, Perekayasa perangkat lunak akan mempertahankan integritas dan independensi penilaian profesional mereka.
- Manajemen, manajer dan pemimpin rekayasa perangkat lunak akan mengikuti dan mempromosikan pendekatan etis terhadap manajemen pengembangan dan pemeliharaan perangkat lunak.
- Profesi, Perekayasa perangkat lunak akan mempertinggi integritas dan reputasi profesinya konsisten dengan kepentingan masyarakat.
- Kolega, Perekayasa perangkat lunak akan bersifat adil dan mendukung terhadap koleganya.
- Diri Sendiri, Perekayasa perangkat lunak akan berpartisipasi dalam pembelajaran seumur hidup mengenai praktek profesi mereka dan akan mempromosikan pendekatan etis terhadap praktek profesi tersebut.

Kode Etik ini dibuat terkait dengan perilaku dan keputusan yang dibuat oleh para Software Development Profesional yang mencakup profesi praktisi, pendidik, manajer, supervisor, pembuat kebijakan dan termasuk trainee dan mahasiswa profesi *Software Development*.

Perekayasa perangkat lunak harus memiliki tanggung jawab etis dan moral jika ingin dihormati sebagai profesional. Terdapat beberapa standar dan kode etik yang harus dipertimbangkan, yaitu :

a. Konfidensialitas

Harus menghormati konfidensialitas atasan dan kliennya walaupun tidak ada persetujuan yang ditanda tangani secara formal.

Kompetensi Tidak boleh menyalahi tingkat kompetensinya (melebihi atau menyimpang)

b. Hak Properti Intelektual

Menyadari terhadap hukum yang mengatur penggunaan properti intelektual, seperti paten, hak cipta dan lain sebagainya.

c. Penyalahgunaan Komputer

Tidak boleh dengan sengaja menyalah gunakan komputer yang nantinya berakibat merugikan orang lain, seperti penyebaran virus, penyadapan dan lain sebagainya.

#### 4. Faktor Yang Mempengaruhi

Faktor-faktor yang mempengaruhi profesionalisme Software Development, diantaranya :

- Pesatnya perkembangan dan kemajuan teknologi, sehingga seorang Software Developmenting harus profesional jika ingin bertahan di dunia Developmenting dan harus memiliki pengetahuan dan pengalaman di bidangnya .
- Cepatnya kehadiran software-software terbaru, sehingga seorang Software Developmenting harus menciptakan software baru dan berkualitas agar pengguna software merasa puas dengan hasil kinerjanya.
- Meningkatnya kebutuhan terhadap teknologi yang semakin canggih, untuk dapat berpartisipasi dalam kemajuan teknologi seorang Software Developmenting harus berperan aktif atau mengembangkan apa yang menjadi skillnya.
- Aktivitas *Software Development* yang membuat pilihan praktis lain sehingga mempengaruhi orang lain secara signifikan.
- Profesional komputasi sejak dari desain pembangunan dan memainkan alat-alat komputerisasi.
- Komputerisasi yang berpengaruh pada orang lain.
- Pengaruh aturan yang di buat oleh IEEE, ACM, BCS, dan lain-lain.

## 5. Keterampilan Yang Harus Dimiliki

Seorang *Software Development* harus mempunyai kemampuan dan keterampilan, diantaranya :

- Menguasai keterampilan dan pengetahuan rekayasa perangkat lunak.
- Mampu bekerja sebagai perorangan dan sebagai bagian dari suatu tim untuk mengembangkan suatu perangkat lunak yang berkualitas.
- Mampu merancang suatu perangkat lunak sesuai dengan sasaran proyek. Batasan biaya, waktu, pengetahuan, sistem yang berjalan dan organisasi.
- Mampu memberikan desain solusi sesuai dengan aplikasi yang dibangun dan mampu mengintegrasikan dengan pendekatan etis, sosial, hukum dan ekonomi.
- Mampu mendemonstrasikan suatu pemahaman akan teori, model, teknik dalam hal identifikasi masalah dan analisa, desain perangkat lunak, pengembangan, implementasi, verifikasi dan dokumentasi.
- Mampu dan paham akan pentingnya negosiasi, kebiasaan bekerja efektif, leadership dan komunikasi baik dengan stakeholders maupun dengan rekan tim pada saat pengembangan software.
- Selalu mempelajari model baru, teknik, teknologi yang muncul serta mengembangkan profesional yang berkelanjutan.

## 6. Kewajiban

Software Developments memiliki kewajiban untuk mempertimbangkan keselamatan, kesehatan, dan kesejahteraan pengguna mereka. Pengembangan Software adalah proses sosial dan *Software Development* memiliki dua kewajiban untuk berhasil :

- Berdasarkan pada standar teknis
- Kewajiban tanggung jawab sosial kepada mereka yang akan bekerja sama produknya.

Adapun kewajiban lainnya :

- Konsisten pada kepentingan umum.
- Layak pemberi kerja.
- Harus dapat meyakinkan bahwa produk yang dihasilkan dalam kondisi sesuai dengan standart profesi.
- Dalam memberikan penilaian profesionalnya, tidak boleh memihak.

- Mengatur dan menjadi pemimpin yang menjadikan dan mempromosikan pendekatan etika pada manajemen pengembangan software dan pemeliharannya.
- Adil dan membantu dalam kerja beretika
- Berpartisipasi dalam pembelajaran jangka panjang profesi mereka dan pendekatan etika dan keefisiennya.

## 7. Sikap

Sikap profesionalisme *Software Development*, diantaranya:

- Jujur dan adil.
- Memahami hukum yang terkait.
- Menghargai dan menghindari merugikan pihak lain.
- Menghargai hak milik.

## 8. Kesimpulan

Berdasarkan apa yang telah dijelaskan di atas bahwa fungsi engineer tidak hanya merancang, membuat, merawat dan mengembangkan teknologi yang dikembangkan tetapi pada implementasinya tindakan yang dilakukan dalam hal ini perilaku yang dibangun pun harus mengikuti kode etik yang telah ditetapkan oleh organisasi profesional sesuai dengan bidang yang diembannya. Selain aturan hukum dan kode etik yang harus ditaati disebabkan karena fungsi profesional adalah mengutamakan klien, atasan dan ranah publik yang menyangkut profesi engineer yang bersangkutan.

## 9. Daftar Pustaka

Prasetyo, Hanung Nindito. 2013. Etika dan Tanggungjawab Profesional Dalam Ranah Rekayasa Perangkat Lunak. JURNAL LPKIA Vol. 2 No. 2, Juni 2013.

<https://www.coursehero.com/file/p3bqoiuv/Perekayasa-perangkat-lunak-harus-memiliki-tanggung-jawab-etis-dan-moral-jika/>

**PERAN SOFTWARE ENGINEER DAN ETIKA DALAM SEWA SOFTWARE /  
PERANGKAT DI PERUSAHAAN**

Safta Hastini

(182420084)

safta.hastini@gmail.com

Magister Teknik Informatika, Universitas Bina Darma Palembang

**ABSTRAK**

Dalam perkembangan teknologi informasi, pelaku industri di Indonesia membutuhkan banyak perangkat dan software untuk mendukung kinerja dalam mencapai target perusahaan. Seorang TIM IT khususnya Software Engineer dan IT Support memikirkan langkah cepat untuk membuat kinerja perusahaan agar tetap berjalan dengan cara melakukan sewa software / perangkat dengan vendor yang terpercaya. Diluar sana banyak sekali vendor software yang menawarkan bermacam-macam harga dan software berbagai macam sistem. Oleh karena itu, perlu adanya kode etik dari pihak penyewa dalam penggunaan perangkat dan software yang disewa tersebut.

**Kata kunci:** kode etik, software engineer, IT Support

**I. PENDAHULUAN**

Suatu industri membutuhkan banyak perangkat dan software untuk mendukung kinerja dalam mencapai target perusahaan untuk berkembang dengan demikian perusahaan tentunya seorang TIM IT khususnya Software Engineer dan IT Support memikirkan langkah cepat untuk membuat kinerja perusahaan agar tetap berjalan dengan cara melakukan sewa software / perangkat dengan vendor yang terpercaya. Diluar sana banyak sekali vendor software yang menawarkan bermacam-macam harga dan software berbagai macam sistem. Tentunya perusahaan mengambil yang terbaik sistem

sewa dengan maintenance dengan menandatangani kontrak demikian kita harus memahami kode etik pada sewa software tersebut.

Kode etik adalah pernyataan cita-cita dan peraturan pelaksanaan pekerjaan (yang membedakannya dari murni pribadi) dan merupakan pedoman yang dilaksanakan anggota kelompok organisasi profesi. Kode etik dapat dikatakan sebagai ciri utama keberadaan sebuah profesi. Sifat dan orientasi kode etik hendaknya singkat; sederhana, jelas dan konsisten; masuk akal, dapat diterima, praktis dan dapat dilaksanakan; komprehensif dan lengkap; dan positif dalam formulasinya. Oleh

karena itu pada jurnal ini akan dibahas kode etik pada profesi dalam bidang IT terhadap perangkat sewa di perusahaan.

## II. TINJAUAN PUSTAKA

IT Support adalah seorang teknisi pada sebuah organisasi/perusahaan yang memiliki tanggungjawab untuk instalasi, evaluasi dan peningkatan terhadap tiga objek utama yaitu komputer, software, dan pengembangan sistem jaringan (network system).

Software Engineer adalah seorang profesional yang bertanggung jawab untuk memproduksi produk yang cocok untuk digunakan oleh klien. Perangkat lunak adalah istilah khusus untuk data yang diformat, dan disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca, dan ditulis oleh komputer. Dengan kata lain, bagian sistem komputer yang tidak berwujud.

Kode etik dapat berfungsi sebagai penyeimbang atas sisi negatif yang mungkin timbul dari suatu profesi, menjadi kompas penunjuk arah moral dan sekaligus penjamin mutu moral profesi itu di mata masyarakat.

Etika profesi adalah sikap hidup berupa keadilan untuk memberikan pelayanan profesional terhadap masyarakat dengan ketertiban penuh dan keahlian sebagai pelayanan dalam rangka

melaksanakan tugas berupa kewajiban terhadap masyarakat. (Suhrawardi Lubis, 1994: 6-7)

## III. PEMBAHASAN

### 3.1 IT Support

Pada bagian definisi telah kami jelaskan inti dari tugas seorang Support di bidang IT tersebut, namun kali ini kami akan mencoba rincinya agar Anda lebih mendapat gambaran. Berikut adalah rincian kerjanya.

1. Anda harus memastikan komputer yang digunakan oleh user dapat berfungsi normal/berjalan seperti seharusnya
2. Anda harus memastikan komputer yang digunakan oleh si user tersebut terhubung ke jaringan dan dapat berkomunikasi seperti seharusnya
3. Anda harus memastikan kalau aplikasi-aplikasi yang dipakai oleh si user berfungsi seperti yang seharusnya
4. Anda juga perlu memastikan kalau semua data yang user punya dalam komputer itu tidak dapat dibaca atau diambil oleh pihak yang tidak berkepentingan tanpa seizin user
5. Anda juga harus cek dan update setiap pembaharuan sistem operasi maupun aplikasi yang dijalankan oleh user
6. Anda bertugas melakukan backup terhadap semua data agar aman dan jika sistem mengalami gangguan, Anda

perlu restore data dari backup system yang ada itu

7. Anda pun bertanggungjawab terhadap mesin pendukung semacam printer, scanner dan lain sebagainya
8. Anda merupakan pihak yang turut menjaga keamanan sistem lewat berbagai tools seperti scanning dengan antivirus dan lain sebagainya
9. Anda melakukan pengaturan terhadap browser
10. Anda bertugas juga untuk cek dan perbaiki bila sewaktu-waktu ada masalah pada jaringan komputer user

### **Etika Profesi IT Support**

IT Support merupakan salah satu pekerjaan IT yang mengharuskan seseorang bisa mengatasi masalah umum yang terjadi pada komputer seperti install software, perbaikan hardware dan membuat jaringan komputer. Profesi ini cukup gampang dilakukan karena dapat dilakukan secara otodidak tanpa memerlukan pendidikan khusus.

### **3.2 Software Engineer**

**Software Engineer** adalah sebuah profesi atau pekerjaan di dunia IT yang deskripsi kerjanya melakukan aktivitas engineering (analisa, rekayasa, spesifikasi, implementasi, dan validasi) untuk menghasilkan produk berupa perangkat lunak yang digunakan untuk memecahkan

masalah pada berbagai bidang. Tugas dari seorang software engineer mirip dengan pekerjaan seorang programmer dan system analyst, maka dari itu software engineer diwajibkan paham mengenai keduanya. Yang membedakan software engineer dengan kedua profesi tersebut adalah seorang software engineer harus mendalami ilmu mengenai Software Development Life Cycle (SDLC) sebagai modal untuk mengembangkan software, mulai dari requirement hingga maintenance. Profesi ini dibutuhkan oleh perusahaan yang memang bergerak di bidang IT, khususnya bidang pengembangan software.

### **Etika Profesionalisme Software Engineer**

Kerahasiaan, seorang engineer harus menghormati kerahasiaan atasan maupun klien. Seorang engineer menerima pekerjaan sesuai kompetensinya. Seorang engineer harus menyadari hukum setempat yang mengatur penggunaan kekayaan intelektual seperti paten, hak cipta, dll dalam hal ini harus berhati-hati memastikan bahwa properti intelektual pengusaha dan klien dilindungi. Seorang engineer Perangkat Lunak sebaiknya tidak menggunakan kemampuannya untuk menyalahgunakan komputer orang lain atau menyalahgunakan software atau hardware yang disewa perusahaan untuk

## TUGAS ETHICT SOFTWARE DEVELOPMENT

memenuhi kebutuhan Teknologi informasi di perusahaan tempat mereka bekerja seperti mencuri perangkat, menyalahgunakan hak paten dari penyedia layanan atau vendor. Secara garis besar kelebihan kekurangan software beli dan sewa adalah:

| Software Sekali Beli                                                                 | Software Sewa                                                                                      |
|--------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| Tentunya bayar cuma sekali Anda akan merasa tenang karena datanya ada di tempat Anda | Harga lebih murah, resiko lebih kecil Softwarena selalu diupdate Ada bug, perbaikannya lebih cepat |

### Kekurangan:

| Software Jadi                                                                                       | Software Sewa                                                                                                     |
|-----------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| Harus mengeluarkan budget yg besar untuk membelinya Biasanya perlu instal di masing-masing komputer | Masih ada resiko terhadap keamanan data, meskipun prosentasenya kecil Jika dihitung, cost jangka panjangnya lebih |

## IV. KESIMPULAN

Berdasarkan apa yang telah dijelaskan di atas bahwa fungsi engineer tidak hanya merancang, membuat, merawat dan mengembangkan teknologi yang dikembangkan tetapi pada implementasinya tindakan yang dilakukan dalam hal ini perilaku yang dibangun pun

harus mengikuti kode etik yang telah ditetapkan oleh organisasi profesional sesuai dengan bidang yang diembannya. Selain aturan hukum dan kode etik yang harus ditaati disebabkan karena fungsi profesional adalah mengutamakan klien, atasan dan ranah publik yang menyangkut profesi engineer yang bersangkutan.

## DAFTAR PUSTAKA

- [1] Arya, Dewa. Etika Profesi di bidang teknologi informasi. Diakses 12 Desember 2019 <https://dewaarya.wordpress.com/2013/05/01/etika-profesi-di-bidang-teknologi-informasi/>
- [2] <http://yogapw.wordpress.com/2015/01/17/c-etika-profesi-dalam-dunia-teknologi-informasi/>
- [3] <http://chaniagorandy.blogspot.com/2015/01/17/profesi-programmer.html>
- [4] <http://missyuntary.files.wordpress.com/2015/01/17/etika-programmer.pdf>
- [5] [http://www.academia.edu/2015/01/16Blog\\_ini\\_dibuat\\_untuk\\_memenuhi\\_tugas\\_mata\\_kuliah\\_etika\\_Profesi\\_IT\\_ETIKA\\_PROFESI\\_Y\\_P\\_D\\_F\\_Transfo\\_r\\_m\\_e\\_r\\_2\\_.0](http://www.academia.edu/2015/01/16Blog_ini_dibuat_untuk_memenuhi_tugas_mata_kuliah_etika_Profesi_IT_ETIKA_PROFESI_Y_P_D_F_Transfo_r_m_e_r_2_.0)
- [6] [http://titasinsi.blogspot.com/2015/01/17/pengertian-etika-dan-profesi-dalam\\_8662.html](http://titasinsi.blogspot.com/2015/01/17/pengertian-etika-dan-profesi-dalam_8662.html)

## TUGAS ETHICT SOFTWARE DEVELOPMENT

- [7] <https://girlycious09.wordpress.com/2015/01/17/profesi-yang-ada-di-bidang-teknologi-informasi/>
- [8] <https://shandrakatherine.wordpress.com/2015/01/17/makalah-etika-profesi/>
- [9] <http://www.duniaremaja.net/pdf/tinjauan-profesi-dibidang-teknologi-informasi.html>

## **Tugas Kelompok Etics in Software Development**

**Nama :**

**1. Indri Endang Lestari**

**2. Rifan Fadilah**

**3. Sulistiyani**

**4. Zulius Akbar Amin**

**Kelas : MTI 19AR2**

### **Etika Dalam Pengembangan Perangkat Lunak**

Kepercayaan pada etika Software Engineering kini telah menjadi bagian penting dari budaya kontemporer. Untuk mengambil contoh misalnya, ahli radiologi untuk memverifikasi kebenaran kode yang mengendalikan mesin X-ray yang menunjuk ke arahnya kepala. Pada titik tertentu, tidak ada pilihan selain mengasumsikan bahwa perangkat lunak dibuat dan diinstal di berbagai sistem yang diterapkan tidak memiliki jaminan bahwa telah terimplementasikan dengan benar, dan dengan cara yang konsisten dengan kepentingan publik. Seorang Software Engineer harus berkomitmen untuk membuat analisis, spesifikasi, desain, pengembangan, pengujian dan pemeliharaan perangkat lunak profesi yang bermanfaat dan dihormati. Seorang Software Engineer harus mematuhi Delapan Prinsip berikut:

**1. Publik**

Seorang Software Engineer harus bertindak konsisten dengan kepentingan publik.

**2. Client dan Pegawai**

Seorang Software Engineer harus mampu memberikan tindakan yang terbaik untuk kepentingan Client ataupun Atasan mereka secara konsisten dengan kepentingan publik.

**3. Produk**

Seorang Software Engineer harus memastikan bahwa produk mereka dan modifikasi terkait, memenuhi standar profesional setinggi mungkin.

**4. Judgment**

Seorang Software Engineer harus menjaga integritas dan kemandirian dalam penilaian profesional mereka.

## 5. Manajemen

Manajer dan pemimpin pengembangan perangkat lunak harus bisa membagikan dan memsosialkan suatu pendekatan etis untuk manajemen sebuah pengembangan perangkat lunak dan pemeliharaannya.

## 6. Profesi

Seorang Software Engineer harus meningkatkan integritas dan reputasi profesi yang konsisten dengan kepentingan publik.

## 7. Kolega

Seorang Software Engineer harus adil dan mendukung rekan kerja mereka.

## 8. Diri Sendiri

Seorang Software Engineer harus mau untuk terus belajar sepanjang hayat yang berkaitan dengan praktik profesinya dan harus mempromosikan pendekatan etika terhadap praktik profesi.

### **1. Etika umum**

Etika umum merupakan etika yang bersifat general dan berlaku bagi semua orang, berbeda dengan etika khusus yang hanya berlaku pada satu perusahaan atau kelompok, etika umum mengikat semua orang baik yang ada dikelompok A maupun di kelompok B. Etika umum seorang software engineer merupakan aturan dasar atau pegangan tentang bagaimana seorang software engineer sebaiknya bertindak atau mengambil keputusan.

Adapun kode etik yang harus diikuti seorang software engineer yaitu :

1. Seorang software engineer harus memperhatikan kepentingan umum dalam pembuatan software.
2. Bisa mempertanggung jawabkan secara penuh apapun yang mereka kerjakan atau mereka buat.
3. Software yang dibuat harus berguna dan bermanfaat bagi masyarakat dan tidak memiliki dampak negatif yang dapat merugikan.
4. Adil dan tidak boleh memalsukan informasi mengenai program aplikasi yang dibuatnya.
5. Memberikan yang terbaik bagi client atau orang yang memberikan pekerjaan tetapi harus tetap menjaga dan memperhatikan kepentingan umum.

6. Tidak boleh menggunakan perangkat lunak secara ilegal atau tanpa izin dari pembuat atau orang yang memiliki hak cipta terhadap perangkat lunak tersebut.
7. Hanya menerima pekerjaan yang sesuai dengan bidang yang dikuasainya.
8. Dapat menjaga kerahasiaan informasi apapun yang didapatkan dari perangkat lunak yang dibuatnya.
9. Melaporkan masalah yang dapat merugikan masyarakat umum yang terdapat dalam perangkat lunak atau dokumen tentang perangkat lunak tersebut pada klien atau yang memberikan pekerjaan
10. Harus memperhatikan perangkat lunak yang dibuatnya, apakah sudah memenuhi standar atau belum.
11. Tidak membuat bug atau kegagalan program secara sengaja agar mendapatkan keuntungan lebih
12. Tidak sebaiknya menggunakan keterampilan yang dimiliki untuk melakukan cracking/hacking atau apapun yang dapat merugikan orang lain.
13. Tidak boleh membuat dokumentasi mengenai perangkat lunak dengan asal asal dan tidak akurat.

## **2. Etika khusus**

Selain etika ĩ etika umum yang berlaku bagi seorang software engineer, Google sebagai perusahaan berbasis dunia juga memiliki etika khusus yang berlaku bagi para pegawainya. Etika ini beragam mulai dari etika yang berhubungan dengan kepentingan google sebagai perusahaan, hingga etika yang berhubungan dengan pelanggan. Berikut merupakan etika yang perlu diikuti para pegawai google

### **1. Melayani pengguna**

google merupakan perusahaan yang memberikan banyak layanan berbasis data, dari e-mail, penyimpanan file, search engine dll, sehingga hal yang paling berharga bagi google merupakan kepercayaan pelanggan, oleh sebab itu, pegawai google diharapkan selalu mengambil tindakan yang menjaga dan meningkatkan kepercayaan pelanggan pada google. Hal ini dilakukan dengan sebisa mungkin menjaga privasi dan kebebasan berpendapat pengguna serta keamanan data dari pihak tak ber-hak, sekali pun dihadapan intervensi pemerintah.

Pegawai google juga diharapkan untuk selalu berusaha meningkatkan daya guna google dan kepuasan pengguna dengan bersikap responsive terhadap tanggapan pengguna baik pertanyaan maupun masalah,serta memberikan ide- ide untuk meningkatkan kualitas layanan google.

2. Saling menopang satu sama lain

Pegawai diterima dengan dasar pencapaian dan kompetensi professional, sehingga dilarang keras melakukan diskriminasi terhadap ras, warna, agama, kenegaraan, keturunan, jenis kelamin dan karakteristik lain yang dilindungi hukum.

Pegawai diharapkan untuk menjaga kesehatan dan keamanan selama bekerja, sehingga dilarang menggunakan obat obatan terlarang, selain itu, pegawai juga tak diperbolehkan melakukan kekerasan atau pun membawa senjata di tempat kerja.

3. Menghindari konflik kepentingan

Pegawai google diharapkan untuk menghindari keadaan yang dapat menyebabkan konflik kepentingan dengan google, seperti berinvestasi dan memiliki bagian pada perusahaan kompetitor google, menduduki jabatan anggota dewan pada perusahaan lain, mengambil kesempatan bisnis yang didapatkan dari google tanpa izin, atau pun menerima hadiah dengan nilai signifikan dari perusahaan kompetitor.

4. Data rahasia

Google sering kali membocorkan informasi mengenai inovasi terbaru dalam konferensi pers, namun membocorkan informasi tertentu perusahaan dapat menyebabkan kerugian perusahaan pada peluncuran produk, atau pun menghilangkan keunggulan kompetitif, sehingga pegawai diharapkan untuk mengenali, menjaga, maupun menghancurkan (saat sewajarnya) informasi sensitive google maupun perusahaan partner.

Google menghormati competitor dan ingin bersaing secara adil, sehingga seperti google tak menginginkan mantan pegawai google menyebarkan informasi sensitive pada perusahaan saingan, google juga mengharapakan pegawai google tak memanfaatkan informasi sensitive kompetitor.

5. Melindungi aset google

Merek dagang, logo, hak cipta dan rahasia dagang merupakan salah satu aset google yang paling berharga. Menggunakannya secara tak sah dapat menyebabkan kerugian besar, sehingga karyawan tak diperbolehkan untuk menggunakan logo, merek dagang,

informasi dan properti google tanpa persetujuan terlebih dahulu dari tim marketing. Selain itu pegawai juga tak diperbolehkan menggunakan properti intelektual perusahaan lain tanpa izin.

6. Menggunakan uang google

Uang yang digunakan untuk kegiatan operational google merupakan milik google dan pemegang saham, oleh sebab itu, pegawai diharapkan menggunakan uang dengan bijaksana dengan jumlah yang masuk akal. Pegawai juga diharapkan untuk mencatat dengan jelas semua pengeluaran yang ditanggung oleh google.

Daftar Pustaka

<https://abc.xyz/investor/other/google-code-of-conduct.html>

<https://sites.google.com/a/email.vccs.edu/bus100mawan/google-code-of-ethics>

<https://www.computer.org/web/education/code-of-ethics>

**NIM : 182420086**

**NAMA : TRI AKHYARI ROMADHON**

Hal-hal inilah yang kemudian memunculkan unsur etika sebagai faktor yang sangat penting kaitannya dengan penggunaan sistem informasi berbasis komputer, mengingat salah satu penyebab pentingnya etika adalah karena etika melingkupi wilayah – wilayah yang belum tercakup dalam wilayah hukum. Faktor etika disini menyangkut identifikasi dan penghindaran terhadap *unethical behavior* dalam penggunaan sistem informasi berbasis komputer

### **1. Perilaku Moral , Konsep Etika dan Hukum**

Dalam suatu masyarakat yang memiliki kesadaran sosial, tentunya setiap orang diharapkan dapat melakukan apa yang benar secara moral, etis dan mengikuti ketentuan hukum yang berlaku.. Moral adalah tradisi kepercayaan mengenai perilaku benar dan salah. Moral dipelajari setiap orang sejak kecil sewaktu yang bersangkutan masih anak-anak. Sejak kecil , anak-anak sudah diperkenalkan perilaku moral untuk membedakan mana yang baik dan buruk, mana yang boleh dan tidak, atau mana tindakan yang terpuji dan tercela.

Sebagai contoh: anak-anak diminta berlaku sopan terhadap orang tua, menghormati guru, atau tidak menyakiti teman-temannya. Pada saat anak-anak telah dewasa, dia akan mempelajari berbagai peraturan yang berlaku di masyarakat dan diharapkan untuk diikuti. Peraturan-peraturan tingkah laku ini adalah perilaku moral yang diharapkan dimiliki setiap individu..

Walau berbagai masyarakat tidak mengikuti satu set moral yang sama, terdapat keseragaman kuat yang mendasar. “Melakukan apa yang benar secara moral” merupakan landasan perilaku sosial kita. Tindakan kita juga diarahkan oleh etika (ethics). Kata *ethics* berakar dari bahasa Yunani *ethos*, yang berarti karakter. Etika adalah satu set kepercayaan, standar, atau pemikiran yang mengisi suatu individu, kelompok atau masyarakat. Semua individu bertanggung jawab pada masyarakat atas perilaku mereka. Masyarakat dapat berupa suatu kota, negara, atau profesi.

Hukum adalah peraturan perilaku formal yang dipaksakan oleh otoritas berdaulat, seperti pemerintah, pada rakyat atau warga negaranya. Hingga kini sangat sedikit hukum

yang mengatur penggunaan komputer. Hal ini karena komputer merupakan penemuan baru dan sistem hukum kesulitan mengikutinya.

Kita dapat melihat bahwa penggunaan komputer dalam bisnis diarahkan oleh nilai-nilai moral dan etika dari para manajer, spesialis informasi dan pemakai, dan juga hukum yang berlaku. Hukum paling mudah diinterpretasikan karena berbentuk tertulis. Di pihak lain, etika tidak didefinisikan secara persis dan tidak disepakati oleh semua anggota masyarakat. Bidang yang sukar dari etika komputer inilah yang sedang memperoleh banyak perhatian.

Program etika adalah suatu sistem yang terdiri dari berbagai aktivitas yang dirancang untuk mengarahkan pegawai dalam melaksanakan pernyataan komitmen. Suatu aktivitas yang umum adalah pertemuan orientasi yang dilaksanakan bagi pegawai baru. Selama pertemuan ini, subyek etika mendapat cukup perhatian. Contoh lain dari program etika adalah audit etika. Dalam audit etika, seseorang auditor internal mengadakan pertemuan dengan seorang manajer selama beberapa jam untuk mempelajari bagaimana unit manajer tersebut melaksanakan pernyataan komitmen. Kode etik khusus instansi, Banyak instansi telah merancang kode etika mereka sendiri. Kadang-kadang kode ini diadaptasi dari kode etik dari organisasi sejenis.

## **2. Perlunya Etika Dalam Pemanfaatan Teknologi Informasi**

Perlindungan atas hak individu di internet dan membangun hak informasi merupakan sebagian dari permasalahan etika dan sosial dengan penggunaan sistem informasi yang berkembang luas. Permasalahan etika dan sosial lainnya, di antaranya adalah: perlindungan hak kepemilikan intelektual, membangun akuntabilitas sebagai dampak pemanfaatan sistem informasi, menetapkan standar untuk pengamanan kualitas sistem informasi yang mampu melindungi keselamatan individu dan masyarakat, mempertahankan nilai yang dipertimbangkan sangat penting untuk kualitas hidup di dalam suatu masyarakat informasi.

Dari berbagai permasalahan etika dan sosial yang berkembang berkaitan dengan pemanfaatan sistem informasi, dua hal penting yang menjadi tantangan manajemen untuk dihadapi, yaitu:

- a. Memahami risiko-risiko moral dari teknologi baru. Perubahan teknologi yang cepat mengandung arti bahwa pilihan yang dihadapi setiap individu juga berubah dengan cepat begitu pula keseimbangan antara risiko dan hasil serta kekhawatiran

kemungkinan terjadinya tindakan yang tidak benar. Perlindungan atas hak privasi individu telah menjadi permasalahan etika yang serius dewasa ini. Di samping itu, penting bagi manajemen untuk melakukan analisis mengenai dampak etika dan sosial dari perubahan teknologi. Mungkin tidak ada jawaban yang selalu tepat untuk bagaimana seharusnya perilaku, tetapi paling tidak ada perhatian atau manajemen tahu mengenai risiko-risiko moral dari teknologi baru.

- b. Membangun kebijakan etika organisasi yang mencakup permasalahan etika dan sosial atas sistem informasi. Manajemen bertanggung jawab untuk mengembangkan, melaksanakan, dan menjelaskan kebijakan etika organisasi. Kebijakan etika organisasi berkaitan dengan sistem informasi meliputi, antara lain: privasi, kepemilikan, akuntabilitas, kualitas sistem, dan kualitas hidupnya. Hal yang menjadi tantangan adalah bagaimana memberikan program pendidikan atau pelatihan, termasuk penerapan permasalahan kebijakan etika yang dibutuhkan.

Etika merupakan prinsip-prinsip mengenai suatu yang benar dan salah yang dilakukan setiap orang dalam menentukan pilihan sebagai pedoman perilaku mereka. Perkembangan teknologi dan sistem informasi menimbulkan pertanyaan baik untuk individu maupun masyarakat pengguna karena perkembangan ini menciptakan peluang untuk adanya perubahan sosial yang hebat dan mengancam adanya distribusi kekuatan, uang, hak, dan kewajiban.

Seperti teknologi – teknologi lainnya, teknologi informasi dapat dipakai untuk mencapai kemajuan masyarakat, namun dapat juga digunakan untuk perbuatan kriminal dan mengancam nilai-nilai masyarakat yang dihargai. Bagaimanapun, perkembangan teknologi informasi akan menghasilkan manfaat-manfaat untuk berbagai pihak dan kemungkinan biaya bagi pihak-pihak lainnya.

Dengan menggunakan sistem informasi, penting untuk dipertanyakan, bagaimana tanggung jawab secara etis dan sosial dapat ditempatkan dengan memadai dalam pemanfaatan sistem informasi. Etika, sosial, dan politik merupakan tiga hal yang berhubungan dekat sekali. Permasalahan etika yang dihadapi dalam perkembangan sistem informasi manajemen umumnya tercermin di dalam lingkungan sosial dan politik.

Untuk dapat memahami lebih baik hubungan ketiga hal tersebut di dalam pemanfaatan sistem informasi, diidentifikasi lima dimensi moral dari era informasi yang sedang berkembang ini, yaitu:

- a. Hak dan kewajiban informasi; apa hak informasi yang dimiliki oleh seorang individu atau organisasi atas informasi? Apa yang dapat mereka lindungi? Kewajiban apa yang dibebankan kepada setiap individu dan organisasi berkenaan dengan informasi?
- b. Hak milik dan kewajiban; bagaimana hak milik intelektual dilindungi di dalam suatu masyarakat digital di mana sulit sekali untuk masalah kepemilikan ini ditrasir dan ditetapkan akuntabilitasnya, dan begitu mudahnya hak milik untuk diabaikan?
- c. Akuntabilitas dan pengendalian; siapa bertanggung jawab terhadap kemungkinan adanya gangguan-gangguan yang dialami individu, informasi, dan hak kepemilikan?
- d. Kualitas sistem; standar data dan kualitas sistem apa yang diinginkan untuk melindungi hak individu dan keselamatan masyarakat?
- e. Kualitas hidup; nilai apa yang harus dipertahankan di dalam suatu informasi dan masyarakat berbasis pengetahuan? Lembaga apa yang harus ada untuk melindungi dari kemungkinan terjadinya pelanggaran informasi? Nilai budaya dan praktik-praktik apa yang diperlukan di dalam era teknologi informasi yang baru?

Perkembangan teknologi dan sistem informasi banyak membawa perubahan pada berbagai aspek kehidupan, khususnya yang mempengaruhi etika dan sosial masyarakat. Beberapa organisasi telah mengembangkan kode etik sistem informasi. Namun demikian, tetap ada perdebatan berkaitan dengan kode etik yang dapat diterima secara umum dengan kode etik sistem informasi yang dibuat secara spesifik. Sebagai manajer maupun pengguna sistem informasi, kita didorong untuk mengembangkan seperangkat standar etika untuk pengembangan kode etika sistem informasi, yaitu yang berbasiskan pada lima dimensi moral yang telah disampaikan di awal, yaitu:

- a. Hak dan kewajiban informasi; Kode etik sistem informasi harus mencakup topik-topik, seperti: privasi e-mail setiap karyawan, pemantauan tempat kerja, perlakuan informasi organisasi, dan kebijakan informasi untuk pengguna.
- b. Hak milik dan kewajiban; Kode etik sistem informasi harus mencakup topik-topik, seperti: lisensi penggunaan perangkat lunak, kepemilikan data dan fasilitas organisasi, kepemilikan perangkat lunak yang buat oleh pegawai pada perangkat keras organisasi, masalah *copyrights* perangkat lunak. Pedoman tertentu untuk hubungan kontraktual dengan pihak ketiga juga harus menjadi bagian dari topik di sini.
- c. Akuntabilitas dan pengendalian; Kode etik harus menyebutkan individu yang bertanggung jawab untuk seluruh sistem informasi dan menggaris bawahi bahwa

individu-individu inilah yang bertanggung jawab terhadap hak individu, perlindungan terhadap hak kepemilikan, kualitas sistem dan kualitas hidup.

- d. Kualitas sistem; Kode etik sistem informasi harus menggambarkan tingkatan yang umum dari kualitas data dan kesalahan sistem yang dapat ditoleransi. Kode etik juga harus dapat mensyaratkan bahwa semua sistem berusaha mengestimasi kualitas data dan kemungkinan kesalahan sistem.
- e. Kualitas hidup; Kode etik sistem informasi juga harus dapat menyatakan bahwa tujuan dari sistem adalah meningkatkan kualitas hidup dari pelanggan dan karyawan dengan cara mencapai tingkatan yang tinggi dari kualitas produk, pelayanan pelanggan, dan kepuasan karyawan.

Sebuah organisasi pengguna komputer profesional tertua di dunia *Association for computing machinery* (ACM) yang didirikan pada tahun 1947 telah menyusun kode etik dan perilaku profesional ( *Code of ethics and professional practice*) yang diharapkan diikuti oleh anggotanya. Selain itu kode etik dan praktik profesional rekayasa software (*software engineering code of ethics and professional*) dibuat dengan tujuan agar bertindak sebagai panduan untuk mengajarkan dan mempraktikkan rekayasa software yaitu penggunaan prinsip-prinsip perancangan dalam pengembangan software.

Kode etik dan perilaku profesional ACM, berisikan keharusan yang merupakan pernyataan tanggung jawab pribadi. Kode ini di bagi menjadi empat bagian :

- a. **Keharusan Moral Umum.** Keharusan ini berkaitan dengan perilaku moral ( memberi kontribusi pada masyarakat, menghindari bahaya, berlaku jujur, dapat dipercaya dan adil) dan isu – isu yang pada saat ini mendapatkan perhatian hukum (hak milik, hak cipta, privasi, kerahasiaan )
- b. **Tanggung jawab profesional yang lebih spesifik.** Hal ini berkenaan dengan dimensi kinerja profesional. Isu moral, seperti berlaku jujur dalam melakukan evaluasi dan menghargai komitmen. Isu hukum dan tanggung jawab sosial untuk berkontribusi terhadap pemahaman umum mengenai komputer.
- c. **Keharusan Kepemimpinan Organisasi.** Sebagai pemimpin anggota ACM memiliki tanggung jawab untuk mendukung penggunaan sah sumber daya komputer, menstimulasi orang lain di organisasi untuk memenuhi tanggung jawab

sosial, memungkinkan pihak lain di dalam organisasi mendapatkan manfaat dari komputer, serta melindungi kepentingan para pengguna.

- d. **Kepatuhan kepada kode.** Di sini anggota ACM harus mengindikasikan dukungan untuk kode etik.

Kode ACM membahas lima dimensi utama pekerjaan yang berkaitan dengan komputer / moral, hukum, kinerja profesional, tanggung jawab sosial, dukungan internal. Tabel 12.1 mengilustrasikan bagaimana lima wilayah ini dibahas oleh tiga bagian utama. Meskipun kode ACM ditujukan untuk para anggota ACM, kode ini memberi panduan yang baik untuk semua profesional komputer.

Tabel 12.1

Topik yang tercakup dalam kode etik dan perilaku profesional ACM

|                                          | Perilaku Moral | Tanggung Jawab Hukum | Kinerja profesional | Tanggung jawab sosial | Dukungan internal |
|------------------------------------------|----------------|----------------------|---------------------|-----------------------|-------------------|
| Keharusan Moral Umum                     | √              | √                    |                     | √                     |                   |
| Tanggung Jawab Profesional yang spesifik | √              | √                    | √                   | √                     |                   |
| Keharusan Kepemimpinan Organisasi        |                | √                    |                     |                       | √                 |

Etika komputer (*computer ethics*) dimaknai sebagai analisis mengenai sifat dan dampak sosial teknologi komputer, serta informasi dan justifikasi kebijakan untuk menggunakan teknologi tersebut secara etis. Karena itu, etika komputer terdiri dari dua aktivitas utama.

- a. Waspada dan sadar bagaimana komputer mempengaruhi masyarakat.

- b. Karena itu harus berbuat sesuatu dengan memformulasikan kebijakan – kebijakan yang memastikan bahwa teknologi tersebut digunakan secara tepat dan etis.

Namun ada satu hal yang sangat penting: "bukan hanya manajemen - pengelola informasi sendiri yang bertanggung jawab atas etika komputer". Para manajer puncak lain juga bertanggung jawab. Keterlibatan seluruh instansi merupakan keharusan mutlak dalam dunia *end-user computing* saat ini, semua manajer di semua area bertanggung jawab atas penggunaan komputer yang etis di area mereka. Dan selain manajer, setiap pegawai bertanggung jawab atas aktivitas mereka yang berhubungan dengan komputer.

Ada tiga alasan utama minat masyarakat yang tinggi pada etika komputer, yaitu : Kelenturan logika (*Logical malleability*), transformasi, dan faktor tak kasat mata (*invisibility factors*).

- a. **Kelenturan logika** (*Logical malleability*). Yang dimaksud dengan kelenturan logika (*logical malleability*) adalah kemampuan memprogram komputer untuk melakukan apa pun yang kita inginkan. Komputer bekerja tepat seperti yang diinstruksikan oleh programernya. Kelenturan logika inilah yang menakutkan masyarakat. Tetapi masyarakat sebenarnya tidak takut terhadap komputer. Sebaliknya masyarakat takut terhadap orang-orang yang memberi perintah di belakang komputer.
- b. **Faktor transformasi**. Alasan kepedulian pada etika komputer ini didasarkan pada fakta bahwa komputer dapat mengubah secara drastis cara kita melakukan sesuatu. Kita dapat melihat transformasi tugas yang sama pada semua jenis organisasi. Contoh dalam hal penggunaan surat elektronik (*e-mail*). *Email* tidak hanya memberikan cara bertelepon yang lain, tetapi memberikan cara komunikasi yang sama sekali baru. Transformasi serupa dapat dilihat pada cara manajer mengadakan rapat. Dulu para manajer harus berkumpul secara fisik di satu lokasi, sekarang mereka dapat bertemu dalam bentuk konferensi video.
- c. **Faktor tak kasat mata** (*invisibility factors*). Alasan ketiga minat masyarakat pada etika komputer adalah karena semua operasi internal komputer tersembunyi dari penglihatan. Operasi internal yang tidak nampak ini membuka peluang pada nilai – nilai pemrograman yang tidak terlihat, perhitungan rumit yang tidak terlihat dan penyalahgunaan yang tidak terlihat. Nilai-nilai pemrograman yang tidak terlihat adalah perintah – perintah yang programmer kodekan menjadi program yang mungkin dapat atau tidak menghasilkan pemrosesan yang diinginkan pemakai. Selama penulisan program, programmer harus membuat

serangkaian pertimbangan nilai seperti bagaimana program mencapai tujuannya. Ini bukan suatu tindakan jahat dari pihak programmer, tetapi lebih merupakan kurangnya pemahaman. Ketidak mampuan komputer memberikan apa yang diinginkan pemakainya disebabkan oleh faktor tak kasat mata ini.

Perhitungan rumit yang tidak terlihat berbentuk program-program yang demikian rumit sehingga tidak dimengerti oleh pemakai. Manajer menggunakan tanpa mengetahui sama sekali bagaimana program tersebut melaksanakan perhitungan.

Penyalahgunaan yang tidak terlihat meliputi tindakan yang sengaja melanggar batasan hukum dan etika. Semua tindakan kejahatan komputer termasuk kategori ini, demikian pula tindakan tidak etis seperti mengganggu hak privasi individual, dan memata-matai.

### **3. Kode Etik Penggunaan komputer**

Pada tahun 1992, koalisi etika komputer yang tergabung dalam lembaga etika komputer (CEI) memfokuskan pada kemajuan teknologi informasi, etika dan perusahaan serta kebijakan publik. CEI mengalamatkannya pada kebijakan organisasi, publik, industrial, dan akademis. Lembaga ini memperhatikan perlunya isu mengenai etika berkaitan dengan kemajuan teknologi informasi dalam masyarakat dan telah menciptakan sepuluh perintah etika penggunaan komputer adalah

- a. Tidak menggunakan komputer untuk merugikan orang lain.
- b. Tidak mengganggu pekerjaan komputer orang lain.
- c. Tidak memata-matai file komputer orang lain.
- d. Tidak menggunakan komputer untuk mencuri.
- e. Tidak menggunakan komputer untuk bersaksi palsu.
- f. Tidak menyalin atau menggunakan kepemilikan perangkat lunak dimana anda belum membayarnya.
- g. Tidak menggunakan sumber daya komputer orang lain tanpa otorisasi atau kompensasi yang sesuai.
- h. Tidak mengambil untuk diri sendiri karya intelektual orang lain.
- i. Harus memikirkan tentang konsekuensi sosial program yang anda tulis bagi sistem yang anda desain.
- j. Harus menggunakan komputer yang menjamin pertimbangan dan bagi sesama manusia.

Persoalan etis khusus penggunaan komputer berasal dari karakteristik unik komputer dan peran yang dimainkannya. Komputer saat ini merupakan aset yang dapat dinegosiasikan, komputer juga melayani sebagai instrumen tindakan, sehingga tingkatan dimana pemberi layanan komputer dan user harus bertanggung jawab terhadap integritas output komputer menjadi sebuah persoalan yang penting untuk diperhatikan.

Kebutuhan terhadap profesionalisme dalam hal pemberi layanan dalam industri komputer, sebagai mana sistem personil yang mendukung dan memelihara teknologi komputer. Kode Etik adalah konsekuensi logis bagi realisasi komitmen dalam penggunaan teknologi komputer secara aman baik dalam sektor publik maupun swasta.

Ada kebutuhan bagi profesionalisme pada wilayah pengguna sistem komputer, dalam hal tanggung jawab mereka untuk beroperasi secara legal dalam respek penuh menurut urutan yang benar. User harus dibuat sadar terhadap risiko operasi ketika sistem sedang digunakan dan diinstall; mereka memiliki tanggung jawab untuk mengidentifikasi penyelewengan keamanan. Ini menunjukkan kode etik dalam komunitas user.

Pendidikan dapat memainkan peran yang sangat penting dalam pengembangan standar etis dalam layanan komputer dan komunitas user. Pembukaan komputer terjadi pada masa paling awal dalam banyak negara, seringkali pada level sekolah dasar. Ini menghadirkan kesempatan untuk mengenalkan standar etis yang dapat diperluas sebagaimana anak kecil berubah melalui sekolah sampai memasuki dunia kerja. Universitas dan institut yang lebih tinggi levelnya juga memasukkan etika komputer dalam kurikulum sejak persoalan etis muncul dan memiliki konsekuensi pada semua area lingkungan penggunaan komputer.

#### **4. Daftar istilah Penting**

- 1) *ethics* (etika) adalah satu set kepercayaan, standar, atau pemikiran yang mengisi suatu individu, kelompok atau masyarakat.
- 2) *computer ethics* (Etika komputer) dimaknai sebagai analisis mengenai sifat dan dampak sosial teknologi komputer, serta informasi dan justifikasi kebijakan untuk menggunakan teknologi tersebut secara etis.

- 3) Program etika adalah suatu sistem yang terdiri dari berbagai aktivitas yang dirancang untuk mengarahkan pegawai dalam melaksanakan pernyataan komitmen.
- 4) *Association for computing machinery* (ACM) asosiasi pengguna mesin komputer.
- 5) *Code of ethics and professional practice* (kode etik dan perilaku profesional )
- 6) *Software engineering code of ethics and professional* ( kode etik dan praktik profesional rekayasa software )

## 5. Rangkuman

- 1) Terdapat lima dimensi moral dari era informasi yaitu: Hak dan kewajiban informasi; Hak milik dan kewajiban; Akuntabilitas dan pengendalian; Kualitas sistem; Kualitas hidup; dari Nilai budaya dan praktik-praktik apa yang diperlukan di dalam era teknologi informasi.
- 2) Penggunaan komputer dalam bisnis haruslah diarahkan oleh nilai-nilai moral dan etika dari para manajer, spesialis informasi dan pemakai, dan juga hukum yang berlaku. Hukum paling mudah diinterpretasikan karena berbentuk tertulis. Di pihak lain, etika tidak didefinisikan secara persis dan tidak disepakati oleh semua anggota masyarakat. Bidang yang sukar dari etika komputer inilah yang sedang memperoleh banyak perhatian.
- 3) Etika merupakan prinsip-prinsip mengenai suatu yang benar dan salah yang dilakukan setiap orang dalam menentukan pilihan sebagai pedoman perilaku mereka. Perkembangan teknologi dan sistem informasi menimbulkan pertanyaan baik untuk individu maupun masyarakat pengguna karena perkembangan ini menciptakan peluang untuk adanya perubahan sosial dan mengancam adanya distribusi kekuatan, uang, hak, dan kewajiban.
- 4) Pendidikan dapat memainkan peran yang sangat penting dalam pengembangan standar etis dalam layanan komputer dan komunitas user. Ini menghadirkan kesempatan untuk mengenalkan standar etis penggunaan teknologi informasi yang dapat diperluas mulai dari pendidikan dasar sampai perguruan tinggi. Universitas dan institut yang lebih tinggi levelnya perlu memasukkan etika komputer dalam kurikulum.

## **6. latihan soal**

- 1) Jelaskan mengapa persoalan etika menjadi hal yang penting dalam Pemanfaatan Teknologi Informasi.
- 2) Jelaskan lima dimensi moral pada era informasi yang sedang berkembang saat ini
- 3) Jelaskan kode etik dalam sistem informasi harus mencakup topik-topik apa saja.
- 4) Kode etik dan perilaku profesional ACM, berisikan keharusan yang merupakan pernyataan tanggung jawab pribadi. Sebutkan Kode etik dari ACM tersebut.
- 5) Sebutkan dan jelaskan bahwa etika komputer terdiri dari dua aktivitas utama.
- 6) Sebutkan dan jelaskan etika penggunaan komputer menurut asosiasi yang tergabung dalam lembaga etika komputer (CEI)

## ETIKA & TANGGUNG JAWAB PROFESIONAL DALAM REKAYASA PERANGKAT LUNAK

Tri Susanti  
182420071  
MTI19A

### Abstrak

Secara umum para profesional yang bergerak dalam bidang apapun memiliki kode etik dan tanggungjawab dalam menjalankan aktivitas atau pekerjaannya masing-masing. Setiap bidang tentu saja memiliki keunikan sesuai dengan karakteristik pekerjaannya. Hal ini juga berlaku pula dalam dunia teknologi informasi khususnya dalam dunia komputasi, rekayasa perangkat lunak maupun sistem informasi. Seorang Profesional komputasi pada prinsipnya memiliki kewajiban etis bagi klien, atasan, profesional lainnya, dan masyarakat dalam memenuhi tanggung jawab profesional. Kewajiban-kewajiban tersebut dituangkan dalam kode etik, yang dapat digunakan untuk membuat keputusan tentang masalah etika dalam dunia komputasi atau teknologi informasi.

**Keyword : Etika, Tanggung Jawab Profesional**

### I. PENDAHULUAN

Dalam periode global saat ini ekonomi, ilmu pengetahuan, teknologi dan informasi memainkan peran yang sangat penting bagi umat manusia. Kemajuan berbagai teknologi baru yang belum pernah terjadi sebelumnya memberikan pengaruh yang tidak kecil pada pekerjaan dan kehidupan sehari-hari. Dalam berbagai proses yang terjadi dalam era globalisasi saat ini banyak yang menjadi tidak terkendali atau dapat disebut sebagai periode turbulensi atau periode ketidakpastian. Dalam lingkungan yang penuh dengan kompetisi, pengembangan ilmu pengetahuan dan teknologi untuk kepentingan bisnis juga tak luput dari kondisi ketidakpercayaan para individu yang terlibat didalamnya.

Yang terjadi saat ini dapat digambarkan bagaimana tuntutan perusahaan memiliki pekerja yang profesional dan berkualifikasi tinggi, bagaimana seorang manajer atau teknisi sepatutnya berperilaku ketika dihadapkan dalam situasi yang sulit? Bagaimana seorang karyawan yang senantiasa dituntut untuk dapat mengembangkan produk baru, teknologi, dan bekerja dengan penuh inisiatif dalam kaitannya membuka pasar baru? Bagaimana pula keputusan yang diambil mampu mempengaruhi lingkungan, kinerja dan konsumen. Secara normatif tentu saja, karyawan atau pelaku bisnis bertindak dalam batas-batas hubungan kerja dan berperilaku sesuai dengan gambaran kerja unit dalam organisasi atau bahkan instruksi dari atasan sebagai personal yang memegang tanggung jawab terakhir. Namun dengan kondisi seperti itu, tidak berarti bahwa pegawai atau karyawan dapat terbebas dari tanggung jawab individu atau kolektif. Prinsip *Responsibility*, dimana setiap individu diharapkan untuk bertindak secara bertanggung jawab terhadap kepentingan manusia. Hal ini tentu saja menimbulkan pertanyaan mengenai perilaku sosial, etika serta tanggung jawab manusia yang terlibat khususnya dalam pengembangan teknologi informasi. Pertanyaan terbesar adalah bagaimana kaitannya etika dan profesionalisme dalam pengembangan teknologi Informasi khususnya dalam bidang rekayasa perangkat lunak (*software engineering*).

Seorang *software engineer* harus memiliki perilaku etis dan moral secara bertanggung jawab jika ingin dihargai sebagai profesional. Dengan demikian, sebagai profesional harus menjunjung tinggi standar normal kejujuran dan integritas. Namun, terdapat batasan dimana

standar perilaku seorang *software engineer* yang dapat diterima diluar aturan hukum yang berlaku. Beberapa hal tersebut menyangkut

1. Kerahasiaan, seorang engineer harus menghormati kerahasiaan atasan maupun klien
2. Seorang engineer menerima pekerjaan sesuai kompetensinya.
3. Seorang engineer harus menyadari hukum setempat yang mengatur penggunaan kekayaan intelektual seperti paten, hak cipta, dll dalam hal ini harus berhati-hati memastikan bahwa properti intelektual pengusaha dan klien dilindungi.
4. Seorang engineer Perangkat Lunak sebaiknya tidak menggunakan kemampuannya untuk menyalahgunakan komputer orang lain.

## **II. TINJAUAN PUSTAKA**

### **2.1. Profesional**

Profesional berasal dari kata profesi yaitu suatu yang melekat pada seseorang yang memberikan jasa atau keterampilan yang dimilikinya bagi orang lain, maka profesional merupakan sebutan pelakunya. Dalam hal kaitannya dengan Profesional komputasi termasuk didalamnya *desainer hardware, software engineer, database administrator*, analis sistem, dan ilmuwan komputer. Karena profesional memahami bahwa pengetahuannya selalu mengalami kemajuan, profesional juga harus mampu melanjutkan pendidikannya dengan membaca berbagai publikasi. Prinsip yang paling penting menjadi seorang profesional adalah wajib untuk menyimpan informasi spesifik tentang rahasia klien. Mengapa demikian? Karena Profesional pada umumnya cenderung memiliki klien, bukan pelanggan. Hal ini dapat di analogikan seperti penjual yang harus selalu mencoba untuk mampu memuaskan keinginan setiap pelanggannya, maka profesional harus mampu untuk memenuhi kebutuhan kliennya.

### **2.2. Responsibility dan Profesional Responsibility**

Responsibility dalam hal ini adalah tanggungjawab. Tanggungjawab menurut Kamus Besar Bahasa Indonesia (2008) adalah keadaan wajib menanggung segala sesuatunya. Tanggungjawab adalah kesadaran manusia akan tingkah lakunya yang disengaja maupun tidak disengaja. Apabila dikaitkan dengan ranah profesional menurut Wikipedia (2011) dalam hal ini sebut saja tanggungjawab profesional merupakan tindakan secara profesional, mematuhi hukum, menghindari konflik kepentingan, dan menempatkan kepentingan klien di atas kepentingan sendiri.

### **2.3. Kode Etik**

Kode etik adalah sistem norma, nilai dan aturan profesional tertulis yang secara tegas menyatakan apa yang benar dan baik dan apa yang tidak benar dan tidak baik bagi profesional. Kode etik menyatakan perbuatan apa yang benar atau salah, perbuatan apa yang harus dilakukan dan apa yang harus dihindari. Tujuan kode etik agar profesional memberikan jasa sebaik-baiknya kepada pemakai atau kliennya. Adanya kode etik akan melindungi perbuatan yang tidak profesional. Ketaatan tenaga profesional terhadap kode etik merupakan ketaatan naluriyah yang telah bersatu dengan pikiran, jiwa dan perilaku tenaga profesional. Jadi ketaatan itu terbentuk dari masing-masing orang bukan karena paksaan. Dengan demikian tenaga profesional merasa bila dia melanggar kode etiknya sendiri maka profesinya akan rusak dan yang rugi adalah dia sendiri. Kode etik disusun oleh organisasi profesi sehingga masing-masing profesi memiliki kode etik tersendiri.

### **2.4. Rekayasa Perangkat Lunak**

#### **2.4.1. Rekayasa**

Istilah rekayasa sering ditemui dalam kehidupan sehari-hari khususnya dalam bidang teknik, ada istilah rekayasa perangkat lunak (software engineering), rekayasa sistem (system engineering) sampai pada istilah rekayasa jembatan. Dalam banyak kajian istilah rekayasa mengacu kepada aktivitas analisis, desain, konstruksi, dan proses manajemen suatu entitas (objek kajian tertentu). Dalam kamus bahasa Indonesia, istilah rekayasa justru telah menjadi kerdil sebab diartikan sebagian sebagai ‘usaha atau aktivitas menipu’. Namun dalam konteks ini tidak dapat menyalahkan kamus besar bahasa Indonesia, karena suatu istilah dalam bahasa suatu bangsa mengacu kepada fungsi istilah tersebut digunakan. Dalam aspek global, rekayasa dalam hal teknologi telah ‘mengubah dunia’ dengan begitu dahsyat. Berbagai ‘perubahan’ yang terjadi saat ini menurut sebagian besar ahli ditentukan oleh teknologi dan istilah teknologi sangat erat berkaitan dengan istilah rekayasa.

#### **2.4.2. Sejarah & Esensi Rekayasa**

Telaah implikasi rekayasa bagi umum baru dimulai pada tahun 1970-an. Pada awal penggunaan istilah rekayasa, masyarakat menganggap bahwa rekayasawan sebagai alat produksi saja, bukan sebagai seorang pengambil keputusan yang bertanggung jawab. Namun seiring dengan perkembangan zaman, saat ini sebagian masyarakat telah memahami dan menyadari bahwa proses dan produk ke rekayasa (teknologi) merupakan hasil dari kreativitas personal. Mengapa ditulis sebagai sebagian masyarakat? Karena memang masih sebagian saja masyarakat yang menyadari bahwa dalam rekayasa nilai moral, perilaku dan kemampuan sang rekayasawan akan sangat mempengaruhi nilai kreasinya; semakin baik nilai moral seorang rekayasawan, biasanya semakin tinggi nilai keselamatan penggunaan hasil rekayasanya. *Engineer* adalah seorang profesional yang bertanggung jawab untuk memproduksi produk yang cocok untuk digunakan oleh klien. Untuk memastikan bahwa suatu produk layak digunakan, membutuhkan pemahaman tentang lingkungan di mana produk itu digunakan.

#### **2.6. Rekayasa Perangkat Lunak**

Seiring dengan perkembangan teknologi informasi yang sedemikian pesatnya. Perkembangan komputer yang di awalnya ditujukan sebagai alat bantu hitung manusia telah memiliki kemampuan ribuan kali atau mungkin saja telah jutaan kali lipat sejak awal kemunculannya. Perkembangan Perangkat keras (*hardware*) yang awalnya sangat menentukan dalam perkembangan teknologi informasi saat ini bahkan telah mampu dilampaui oleh perangkat lunak (software) dalam memberikan pengaruhnya dalam era industri. Pada masa awal perhitungannya, sistem berbasis komputer dikembangkan dengan menggunakan manajemen yang berorientasi pada perangkat keras. Para manajer proyek memfokuskan diri kepada perangkat keras karena pengembangan sistem membutuhkan biaya yang sangat besar dan menyita waktu. Untuk mengembangkan perangkat lunak pada waktu itu para pemrogram hanya sekedar coba-coba dan sangat berhati-hati. Sekarang walaupun distribusi biaya untuk pengembangan sistem berbasis komputer telah berubah sangat drastis. Dan telah dapat diprediksi bahwa perangkat lunak merupakan salah satu bentuk tunggal yang memiliki biaya sangat tinggi namun dirasakan oleh berbagai enterprise khususnya perusahaan merupakan investasi yang sangat berpengaruh terhadap maju atau mundur bahkan matinya sebuah perusahaan. Untuk memahami sejauh mana manusia membutuhkan perangkat lunak maka harus memahami apa dan bagaimana perangkat lunak tersebut. Sampai saat ini banyak pakar yang telah mendefinisikan perangkat lunak, namun dalam tulisan ini hanya akan diperkenalkan definisi yang sangat terkenal dari perangkat lunak. Perangkat lunak menurut Roger S. Pressman (1997) adalah

- (1) Perintah (program komputer) yang bila dieksekusi memberikan fungsi dan unjuk kerja seperti yang diinginkan

- (2) Struktur data yang memungkinkan program memanipulasi informasi secara proporsional
- (3) Dan, dokumen yang menggambarkan operasi dan kegunaan program

Berdasarkan hal tersebut dapat dijelaskan bahwa pada prinsipnya perangkat lunak merupakan kumpulan perintah yang dapat digunakan sesuai keinginan dalam hal memanipulasi informasi secara proporsional.

Menurut Roger S. Pressman (1997) perangkat lunak lebih merupakan elemen logika dan bukan merupakan elemen fisik. Dengan demikian, perangkat lunak memiliki ciri yang berbeda dari perangkat keras yaitu

- (1) Perangkat lunak dibangun dan dikembangkan, tidak dibuat dalam bentuk klasik
- (2) Perangkat lunak tidak pernah usang
- (3) Sebagian besar perangkat lunak dibuat secara custom-built, serta tidak dapat dirakit dari komponen yang sudah ada

### **2.5. International standard for profesional software development & Ethical responsibility**

Sebagaimana penjelasan sebelumnya bahwa profesional harus memiliki kode etik yang disusun oleh organisasi maka ACM(Association for Computing Machinery) dan IEEE(Institute Of Electrical & Electronics Engineers) telah bekerja sama untuk menghasilkan kode etik dan profesional bersama dalam ranah *software engineer* yang menjelaskan secara singkat substansinya dan menyatakan :

*Computers have a central and growing role in commerce, industry, government, medicine, education, entertainment and society at large. Software engineers are those who contribute by direct participation or by teaching, to the analysis, specification, design, development, certification, maintenance and testing of software systems. Because of their roles in developing software systems, software engineers have significant opportunities to do good or cause harm, to enable others to do good or cause harm, or to influence others to do good or cause harm. To ensure, as much as possible, that their efforts will be used for good, software engineers must commit themselves to making software engineering a beneficial and respected profession. In accordance with that commitment, software engineers shall adhere to the following Code of Ethics and Professional Practice.*

Secara singkat pernyataan diatas menjelaskan bahwa dengan perkembangan dunia komputer yang begitu pesat maka para software engineer secara etika harus memiliki kontribusi dengan berpartisipasi langsung atau dengan proses pengajaran, untuk proses analisis, spesifikasi, desain, pengembangan, sertifikasi, pemeliharaan dan pengujian perangkat lunak.

## **III. PEMBAHASAN**

Pada implementasinya terdapat banyak peluang bahwa rekayasa perangkat lunak dapat digunakan untuk penyalahgunaan fungsi khususnya dalam dunia kejahatan. Untuk memastikan bahwa rekayasa perangkat lunak digunakan dengan baik untuk kemaslahatan umat manusia maka ada upaya agar para software engineer harus berkomitmen untuk menjadikan profesi ini menguntungkan dan dihormati sehingga diperlukan harus mematuhi standar Kode Etik dan Profesional. *Software engineer* berkomitmen untuk membuat analisis, spesifikasi, desain, pengembangan, pengujian dan pemeliharaan perangkat lunak yang menguntungkan dan dihargai sebagai sebuah profesi. Secara umum *software engineer* harus mematuhi 8(delapan) prinsip berikut :

1. *PUBLIC* – *Software engineers shall act consistently with the public interest.* Pernyataan ini berarti seorang software engineer harus bertindak secara konsisten dengan kepentingan publik.
2. *CLIENT AND EMPLOYER* – *Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.* Seorang software

- engineer harus bertindak sesuai dengan kepentingan terbaik klien dan atasan yang konsisten dengan kepentingan publik.
3. *PRODUCT* – *Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.* Seorang Software engineer harus memastikan bahwa produk dan modifikasi yang terkait dengan memenuhi standar profesional setinggi mungkin.
  4. *JUDGMENT* – *Software engineers shall maintain integrity and independence in their professional judgment.* Seorang Software engineer harus mempertahankan integritas dan kemandirian dalam penilaian profesional yang *the Association for Computing* dimiliki. *Machinery, Inc. 1999*
  5. *MANAGEMENT* – *Software engineering Gotterbarn, D and Miller, K managers and leaders shall subscribe to and and Rogerson, S "Software promote an ethical approach to the Engineering Code of Ethics," management of software development and SIGCAS Newsletter July 1997. maintenance.* Seorang pimpinan dan manajer *Presman, Roger S., (1997)* Software engineering melakukan pendekatan *Software Engineering: A etis* kepada manajemen pengembangan *Practitioner's Approach.* perangkat lunak dan pemeliharaan. *McGraw Hill Book Co.*
  6. *PROFESSION* – *Software engineers shall Ian Sommerville, advance the integrity and reputation of the (2004).Software Engineering profession consistent with the public interest. 7th Edition", Addison-Wesley*  
Seorang Software engineer harus memajukan [http://en.wikipedia.org/wiki/Professional\\_responsibility](http://en.wikipedia.org/wiki/Professional_responsibility) integritas dan reputasi profesi yang konsisten dengan kepentingan publik. *Michael C. Loui, Keith W.*
  7. *COLLEAGUES* – *Software engineers shall be Miller, Ethics and Professional fair to and supportive of their colleagues. Responsibility in* seorang Software engineer harus bersikap adil *Computing.Encyclopedia Of* dan mendukung rekan-rekan atau kolega dalam *Computer Science* pekerjaan.
  8. *SELF* – *Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.* Seorang Software engineer harus berpartisipasi dalam pembelajaran seumur hidup tentang praktek profesi yang dimiliki dan akan mempromosikan pendekatan etis untuk profesinya.

#### IV. KESIMPULAN

Berdasarkan apa yang telah dijelaskan di atas bahwa fungsi *engineer* tidak hanya merancang, membuat, merawat dan mengembangkan teknologi yang dikembangkan tetapi pada implementasinya tindakan yang dilakukan dalam hal ini perilaku yang dibangun pun harus mengikuti kode etik yang telah ditetapkan oleh organisasi profesional sesuai dengan bidang yang diembannya. Selain aturan hukum dan kode etik yang harus ditaati disebabkan karena fungsi profesional adalah mengutamakan klien, atasan dan ranah publik yang menyangkut profesi *engineer* yang bersangkutan.

|             |                                                                                                                                                |
|-------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| Judul Tugas | Kode Etik Profesi IT (Programmer)                                                                                                              |
| MK          | Ethical Issues in Electronic Information System                                                                                                |
| Mahasiswa   | 1. Ahmad Roni Yosef (182420043)<br>2. M. Riski Qisthiano (182420040)<br>3. Nyoman Gede Sujana (182420047)<br>4. Yayan Indra Irawan (182420038) |
| Tanggal     | 05-12-2019                                                                                                                                     |
| Kelompok    | 2                                                                                                                                              |

### **Pendahuluan**

Programmer adalah orang yang menggunakan kode program dengan menggunakan bahasa pemrograman tertentu, untuk menghasilkan sistem/aplikasi yang diinginkan. Programmer juga dapat dikatakan sebagai sebuah pekerjaan, pekerjaan yang belum umum ditemui pada saat seorang Guru bertanya cita-cita pada Muridnya didalam kelas. Artinya Murid yang berada dikelas belum mengetahui, apa itu Programmer dan pekerjaan yang ditangani olehnya.[1]

Programmer merupakan sebuah profesi yang bertanggung jawab terhadap pengembangan aplikasi atau sistem informasi. Aplikasi-aplikasi yang dibuat tentu memiliki banyak manfaat terutama terhadap user. Saat ini, ada banyak sekali bahasa pemrograman yang umum digunakan oleh para programmer. PHP, C#, VB. NET, dan Java merupakan beberapa contoh bahasa pemrograman yang saat ini banyak digunakan oleh para programmer. Oleh karena itu seorang programmer tentu harus bisa menguasai banyak bahasa pemrograman agar dapat menunjang suatu pekerjaan.[2]

### **Kelompok Bidang**

Profesi seorang programmer dapat dikelompokkan menjadi beberapa bidang sebagai berikut :

#### 1. Programmer

Programmer adalah individu yang bertugas dalam hal rincian implementasi, pengemasan, dan modifikasi algoritma serta struktur data, dituliskan dalam sebuah bahasa pemrograman tertentu. Deskripsi Pekerjaan :

- a. Menulis program (coding) dengan menggunakan pemrograman tertentu ( VB, VB.NET, Java ).
- b. Memahami konsep basis data.
- c. Mampu mengoperasikan aplikasi basis data.

- d. Melakukan pengujian terhadap aplikasi program.
- e. Melakukan analisis terhadap aplikasi program.
- f. Melakukan riset , desain, dokumentasi dan modifikasi aplikasi software.
- g. Melakukan analisis dan memperbaiki kerusakan (error ) pada software dengan tepat dan cara yang akurat.
- h. Menyediakan status laporan aplikasi yang diperlukan.

## 2. Database Programmer

Programmer yang menguasai perancangan dan pemrograman database menggunakan Aplikasi Ms. Access dan Ms. Sql Server ATAU PL/SQL dan Oracle Form Developer 10g.

## 3. Web Programmer

Programmer web yang menguasai pengembangan aplikasi web berbasis HTML dan ASP.Net ATAU aplikasi web berbasis HTML dan J2EE serta Struts Framework ATAU aplikasi web berbasis MySQL dan PHP.

## 4. Multimedia Programmer

Programmer multimedia yang menguasai penggunaan teknologi dan pengembangan aplikasi berbasis multimedia.

## 5. Embedded Programmer

Programmer yang menguasai arsitektur sistem mikroprosesor, interfacing dan pemrograman embedded.

## Kode Etik

Setiap profesi biasanya mempunyai kode etik nya masing masing. Merujuk pada wikipedia, kode etik merupakan tatanan etika yang disepakati oleh suatu kelompok masyarakat tertentu. Kode etik bisa kita sebut sebagai pedoman untuk para professional dalam bekerja. Dengan begitu harapan idealnya client/perusahaan yang memperkerjakan atau menggunakan jasa dapat terhindar dari perilaku tidak professional.

Berikut adalah Kode Etik dari Programmer

- **Berkontribusi untuk kehidupan masyarakat yang baik.** Programmer harus mengembangkan sistem komputer yang dapat

mengurangi dampak negatif terhadap masyarakat seperti ancaman sosial dan keamanan, dan dapat membuat aktifitas dan pekerjaan yang lebih mudah. Programmer sebaiknya membangun sesuatu dengan standar yang tinggi.

- **Menghindari hal-hal yang dapat membahayakan orang lain.** Sistem komputer memiliki dampak tidak langsung kepada pihak ketiga. Sistem dapat menyebabkan kehilangan informasi dan sumber daya, dan itu berbahaya untuk pengguna, masyarakat, atau pekerja. Oleh karena itu software developer harus meminimalisir resiko tersebut dengan mengikuti desain standar dan testing yang baik.
- **Jujur dan dapat dipercaya.** Prinsip ini mendorong programmer untuk lebih jujur serta sadar akan keterbatasan pengetahuan mereka saat menuliskan sistem komputer. Juga, jika programmer mengetahui ada kesalahan dalam sistem, dia dapat melaporkan segera untuk menghindari hal-hal yang tidak diinginkan.
- **Memberikan penghargaan untuk aset intelektual.** Software developer dilarang keras untuk mengakui hasil karya orang lain, bahkan juga ketika program tersebut tidak terlindungi oleh "copyright" atau "patent". Mereka harus mengenali dan mengakui pekerjaan/karya orang lain, dan mereka harus menggunakan ide mereka sendiri untuk mengembangkan software.
- **Menghormati privasi orang lain.** Sistem komputer bisa saja disalahgunakan oleh beberapa orang dalam pelanggaran privasi orang lain. Software developer harus menuliskan program yang dapat melindungi informasi pengguna yang dapat menangkal orang tidak dikenal (tidak berizin) mengakses informasi tersebut.
- **Menghormati Kerahasiaan.** Software developer harus bersedia menjaga rahasia informasi terkait pekerjaannya dan segala informasi terkait proyek yang sedang dikerjakannya jika client atau perusahaan menginginkan hal tersebut.

### **Faktor Yang Mempengaruhi**

1. **Komunikasi team**  
Meningkatnya ukuran produk yang dihasilkan akan menurunkan produktivitas programmer akibat meningkatnya kerumitan antara komponen-komponen program dan akibat meningkatnya komunikasi yang perlu dilakukan antara programmer, manajer, dan pelanggan.
2. **Kerumitan produk**  
Tiga level kerumitan produk : program aplikasi, program utility, program level sistem.
3. **Kendali perubahan**  
Perubahan terhadap produk harus tetap meminta persetujuan manajer sebagai penanggung jawab proyek. Dampak perubahan harus dapat ditelusuri, diuji, dan didokumentasikan.
4. **Tingkat keandalan**  
Setiap produk harus mempunyai keandalan standar. Peningkatan keandalan dihasilkan melalui perhatian yang sangat besar pada tahap analisa. Peningkatan keandalan akan menurunkan produktivitas.
5. **Pemahaman Keterampilan**  
Pelanggan adalah penyumbang utama terhadap kegagalan dalam memahami masalah adalah : tidak memahami permasalahan perusahaannya, mengerti kemampuan dan keterbatasan komputer, tidak mempunyai pengetahuan dasar tentang logika dan algoritma, software engineer tidak memahami lapangan aplikasi, gagal mendapatkan informasi kebutuhan pelanggan karena pelanggan bukan seorang end user.
6. **Persyaratan Keterampilan**  
Berbagai keterampilan harus ada dalam sebuah proyek perangkat lunak, misalnya: keterampilan berkomunikasi dengan pelanggan untuk memastikan keinginannya dengan jelas-jelasnya, kemampuan dalam pendefinisian masalah dan perancangan, kemampuan implementasi dengan penulisan program yang benar, kemampuan debugging secara deduktif dengan kerangka 'what if', dokumentasi, kemampuan bekerja dengan pelanggan, semua keterampilan tersebut harus senantiasa dilatih.
7. **Fasilitas dan Sumber Daya**  
Fasilitas non teknis yang tetap perlu diperhatikan yang berkaitan dengan motivasi programmer misalnya : mesin yang baik, serta tempat yang tenang, atau ruang kerjanya dapat ditata secara pribadi
8. **Pelatihan yang cukup**

Banyak programmer yang dilatih dalam bidang-bidang : ilmu komputer, teknik elektro, akuntansi, matematika, tetapi jarang yang mendapat pelatihan dalam bidang teknik perangkat lunak.

### **Keterampilan**

- Keterampilan berkomunikasi dengan pelanggan untuk memastikan keinginannya dengan jelas-jelasnya
  - Kemampuan dalam pendefinisian masalah dan perancangan
  - Kemampuan implementasi dengan penulisan program yang benar
  - Kemampuan debugging secara deduktif dengan kerangka 'what if'
  - Dokumentasi
- Kemampuan bekerja dengan pelanggan, semua keterampilan tersebut harus senantiasa dilatih.

### **Kewajiban**

1. Memahami konsep dasar sistem operasi. Memahami dasar sistem operasi itu penting supaya dalam menjalankan atau membuat suatu program aplikasi mudah karena sudah memahami cara kerja sistem operasi yang kita gunakan.
2. Memahami konsep dasar jaringan. Sebuah aplikasi tidak dapat berjalan sendiri. Aplikasi tersebut pasti harus berhubungan dengan internet, melayani banyak pengguna
3. Memahami konsep dasar relational database. Setiap aplikasi pasti memiliki sebuah database dalam penyimpanan datanya untuk itu programmer khususnya Database Programmer ditekankan menguasai relational database.
4. Karena sekarang jaman internet, maka wajib memahami protokol HTTP, FTP, POP3, SMTP, SSH. Protokol HTTP sekarang adalah protokol yang paling banyak digunakan di internet.
5. Karena sekarang jaman globalisasi, maka wajib memahami Unicode. Unicode itu penting supaya aplikasi kita tetap bisa diinstal di komputer mana saja.
6. Memahami lebih dari satu bahasa pemrograman. Pemahaman lebih dari satu bahasa itu penting agar wawasan programmer lebih terbuka. Bahwa tidak ada bahasa yang one-for-all.
7. Cara menggunakan Version Control. Dalam dunia kerja, penggunaan version control adalah wajib. Ini standar (de facto) internasional. Jika mempunyai project opensource, baik di Sourceforge, Apache, Codehaus, dan semua hosting project opensource, pasti programmer akan diberikan version control.

### **SIKAP**

1. Mempunyai sikap dan kepribadian baik, komunikatif, mudah beradaptasi dengan lingkungan kerja, cekatan dan fleksibel.
2. Mampu bekerja berorientasi jadwal, mengatur pekerjaan multiple project dan bekerja sama dalam team.
3. Membuat kontrak kerja dengan klien.
4. Menyukai dan mengerti dasar – dasar pemrograman.

### **DAFTAR PUSTAKA**

- [1] A. Dan, P. Dasar, and B. Android, “Aplikasi programmer pintar algoritma dan pemrograman dasar berbasis android,” pp. 322–328, 2018.
- [2] A. Supriyatna, “Sistem Informasi Forum Diskusi Programmer Berbasis Web Menggunakan Rapid Application Development,” vol. 7, no. November, pp. 138–147, 2018.
- <http://etikaprogrammer2013.blogspot.com/2013/05/kelompok-bidang-programmer.html>
  - [https://en.wikipedia.org/wiki/Programming\\_ethics](https://en.wikipedia.org/wiki/Programming_ethics)
  - <https://www.ieee.org/index.html>
  - <https://andrewaprianto.wordpress.com/2017/03/31/etika-profesionalisme-seorang-programmer/>
  - <http://www.mampirlah.com/teknik-informatika/makalah-seorang-programmer.html>

**UJIAN AKHIR SEMESTER**  
**ETHICAL ISSUES IN ELECTRONIC INFORMATION SYSTEM**  
**DOSEN PENGASUH: M. IZMAN HERDIANSYAH, PhD**  
**KELAS MTI 19 Reguler B; 11 Desember 2019**

**YUNIARTI DENITA SARI**  
**182420087**

**Penyelesaian Soal**

1. Dalam proses administratif universitas dosen luar biasa mempunyai tugas pokok menjalankan Tridinanti Perguruan Tinggi, yaitu memberi layanan pendidikan-pengajaran, penelitian, dan pengabdian pada masyarakat. Dalam menjalankan hal-hal tersebut dosen luar biasa berperan sebagai :
  - a. Fasilitator dan narasumber pembelajaran mahasiswa.
  - b. Peneliti dan pakar bidang ilmunya masing-masing, untuk mengembangkan ilmu, teknologi, kebudayaan dan seni.
  - c. Pengabdian masyarakat dengan cara menerapkan keahliannya bagi kesejahteraan masyarakat dan kemajuan kemanusiaan.

Dosen luar biasa juga mempunyai fungsi dalam mengembangkan akademik dan profesi, serta berpartisipasi dalam tata pamong institusi. Mengingat tentang banyaknya proses administratif universitas yang harus dipenuhi oleh dosen luar biasa, hal ini menimbulkan tantangan tersendiri di universitas. Universitas harus mampu menghasilkan dosen yang kompeten yang mampu membuat perancangan perkuliahan sesuai dengan ketentuan kerja yang telah ditentukan oleh universitas dengan melihat metode-metode dan teknik pembelajaran yang digunakan oleh dosen luar biasa tersebut. Selain perancangan kuliah, kegiatan tri Dharma Perguruan Tinggi juga harus diperhatikan dan berkordinasi dengan semua unit kerja demi menghasilkan dan melaksanakan program pembelajaran yang inovatif mengikuti perkembangan Ilmu Teknologi saat ini.

Perekrutan dosen luar biasa sangat baik karena dapat menghasilkan dosen-dosen baru yang kompeten. Dengan perekrutan dosen yang kemudian dibudidayakan dengan standar operasional universitas, hal ini dapat membuat universitas memiliki tenaga kerja dengan standar yang diinginkan. Mengingat

saat ini ilmu pengetahuan terus berkembang, hal ini tidak menutup kemungkinan bahwa tenaga pengajar seperti dosen juga terus berkembang. Dengan adanya perekrutan dosen luar biasa maka universitas dapat mengenal banyak dosen dan kemudian menemukan dosen yang cocok, yang apabila sudah benar-benar memenuhi standar universitas barulah dapat diangkat menjadi dosen tetap.

Kegiatan seperti ini dianggap optimal karena dapat mengurangi perputaran karyawan/dosen yang bekerja sehingga proses pergantian karyawan atau dosen yang berhenti bekerja dapat dikurangi. Jika universitas terlalu terburu-buru menerima dosen tetap tanpa melihat kompetensi yang ada dalam dosen yang kemudian berujung pada pemecatan, hal ini tentunya tidak baik dalam proses administrasi dalam universitas. Selain mengganggu proses administrasi yang ada, kegiatan Tri Dinasti Perguruan Tinggi tentunya akan terganggu dimana juga dosen yang bersangkutan tidak dapat lagi mengajar dan harus digantikan oleh dosen pengganti, tentunya hal ini akan menyulitkan mahasiswa karena proses pengajaran setiap dosen berbeda.

2. Saat ini semakin banyak perusahaan yang dalam perekrutan karyawan atau pegawainya menggunakan status *outsourcing* atau alih daya. Jenis pekerjaan yang sekarang di *outsourcing*-kan juga terus bertambah. Jika dulu jenis pekerjaan yang di-*outsourcing*-kan hanya sebatas pekerjaan *supporting* atau pekerja pendukung saja, sekarang sudah semakin banyak pekerja *back-office* yang di *outsourcing*-kan.

Dalam manajemen SDM sendiri dapat dijabarkan dalam aspek manajerial yang meliputi perencanaan, pengorganisasian, pengarahan dan pengawasan, serta aspek operasional yang meliputi pengadaan, kompensasi, pengembangan dan integrasi serta pemeliharaan dalam kerja. Jika dilihat dari aspek-aspek yang ada maka *outsourcing* baiknya dilakukan pada aspek operasional, sedangkan aspek yang tetap *in-house* yaitu aspek manajerial. Aspek manajerial ini biasanya dilakukan oleh pemegang keputusan seperti manajer untuk melakukan *outsourcing* di aspek operasional. Dimana jabatan yang biasanya di-*outsourcing*-kan seperti satpam, *office boy*, operator, *customer service* yang kemudian

menjalankan aspek operasional seperti pengadaan karyawan, pemberian kompensasi, pengembangan atau pelatihan sampai pemeliharaan karyawan *outsourcing*.

3. Kegiatan riset dan pengembangan dalam perusahaan farmasi sangat penting dan bersifat rahasia. Namun, tak jarang prospek operasional inipun dijalakan oleh pegawai/karyawan yang berstatus *outsourcing*. Hal ini tentunya dilakukan dengan banyak pertimbangan. Dalam kegiatannya proses *outsourcing* juga dilindungi secara hukum dari kedua belah pihak yang dituangkan dalam perjanjian kerja yang dibuat sebelum kerjasama dimulai. Dengan adanya perjanjian kerja ini maka pihak perusahaan tidak perlu cemas akan data penting dan rahasia perusahaan dapat tersebar keluar karena dalam prosesnya telah dilindungi secara hukum. Bukan hanya itu, ada beberapa alasan dan pertimbangan yang membuat proses *outsourcing* dapat dilakukan, diantaranya :

- a. Meningkatkan fokus perusahaan

Dengan melakukan *outsourcing*, perusahaan dapat memusatkan diri pada masalah dan strategi utama dan umum, sementara pelaksanaan tugas sehari-hari yang kecil-kecil diserahkan kepada pihak ketiga.

- b. Memanfaatkan kemampuan kelas dunia

Spesialisasi pekerjaan seperti yang dimiliki dan dikembangkan oleh para kontraktor (*outsourcing provider*) mengakibatkan kontraktor tersebut memiliki keunggulan kelas dunia dalam bidangnya. Tentu saja disini diasumsikan bahwa *outsourcing* diberikan betul-betul kepada kontraktor yang unggul dibidang pekerjaan yang dikontrakkan.

- c. Mempercepat keuntungan yang diperoleh dari *re-engineering*

*Outsourcing* adalah produk samping dan salah satu manajemen *tool* lagi yang sangat unggul, yaitu *business process reengineering*. *Reengineering* adalah pemikiran kembali secara fundamental mengenai proses bisnis, dengan tujuan untuk melakukan perbaikan secara dramatis tentang ukuran-ukuran keberhasilan yang sangat kritis bagi perusahaan, yaitu biaya, mutu, jasa, dan kecepatan.

d. Membagi resiko

Apabila semua aktivitas dilakukan oleh perusahaan sendiri, semua investasi yang diperlukan untuk setiap aktivitas tersebut harus dilakukan oleh perusahaan sendiri pula. Perlu diingat bahwa semua bentuk investasi menanggung resiko tertentu. Apabila semua investasi dilakukan sendiri maka seluruh resiko juga ditanggung sendiri. Apabila beberapa aktivitas perusahaan dikontrakkan kepada pihak ketiga maka resiko akan ditanggung bersama pula.

e. Sumber daya sendiri dapat digunakan untuk kebutuhan-kebutuhan lain

Setiap perusahaan tentu mempunyai ketebatasan dalam kepemilikan sumber daya. Tantangan yang terus-menerus harus dihadapi adalah bahwa sumber daya tersebut harus selalu dimanfaatkan untuk memanfaatkan bidang-bidang yang paling menguntungkan. *Outsourcing* memungkinkan perusahaan untuk menggunakan sumber daya yang dimiliki secara terbatas tersebut untuk bidang-bidang kegiatan utama, yaitu hal yang paling dibutuhkannya.

f. Memungkinkan tersedianya dana kapital

*Outsourcing* juga bermamfaat untuk mengurangi investasi dana kapital pada kegiatan *non-core*. Sebagai ganti dari melakukan investasi dibidang kegiatan tersebut, lebih baik mengkontrakkan sesuai dengan kebutuhan yang dibiayai dengan dana operasi, bukan dana investasi. Dengan demikian, dana kapital dapat digunakan pada aktivitas yang lebih bersifat utama.

g. Menciptakan data segar

*Outsourcing*, sering kali dapat dilakukan tidak hanya mengontrakkan aktivitas tertentu pada pihak ketiga, tetapi juga disertai dengan penyerahan penjualan/penyewaan aset yang digunakan untuk melakukan aktivitas tertentu tersebut.

h. Mengurangi dan mengendalikan biaya operasional

Keuntungan yang sangat taktis dari *outsourcing* adalah memungkinkan untuk mengurangi dan mengendalikan biaya operasi. Pengurangan biaya ini dapat dan dimungkinkan diperoleh dari mitra *outsource* melalui berbagai hal, misalnya spesialisasi, struktur pembiayaan yang lebih rendah, ekonomi skala besar (*economics of scale*).Pengurangan ini tidak mungkin dapat diperoleh

apabila aktivitas yang bersangkutan dilakukan sendiri karena tidak mempunyai kemudahan seperti yang dimiliki oleh mitra *outsource* diatas.

i. Memperoleh sumber daya yang tidak dimiliki sendiri

Perusahaan melakukan *outsourcing* untuk suatu aktivitas tertentu karena perusahaan tidak memiliki sumber daya yang dibutuhkan untuk melakukan aktivitas tersebut secara baik dan memadai. Misalnya dalam hal aktivitas logistik, untuk memperoleh biaya logistik yang optimal diperlukan suatu model analitis yang canggih. Banyak perusahaan tidak mempunyai ahli yang cukup dan cakap untuk mengembangkan model-model ini. Oleh karena itu, jalan satu-satunya adalah melakukan *outsourcing*.

j. Memecahkan masalah yang sulit dikendalikan atau dikelola

*Outsourcing* dapat juga digunakan untuk mengatasi pengelolaan hal atau mengawasi fungsi yang sulit dikendalikan. Fungsi yang sulit dikelola dan dikendalikan ini, misalnya birokrasi ekstern yang sangat berbelit yang harus ditaati oleh perusahaan yang dimiliki negara dalam menjalankan fungsi pembelian barang dan jasa, yang sulit ditembus dengan cara-cara biasa. Hal ini mungkin dapat dipecahkan dengan mengkontrakkan saja seluruh pekerjaan tersebut pada pihak ketiga yang berbentuk swasta, yang tidak terikat pada birokrasi tertentu. Contoh adalah mengontrakkan pemeliharaan peralatan karena setelah dilakukan usaha terus-menerus untuk memperbaiki sistem dan kinerja fungsi pemeliharaan, tidak juga dapat diperbaiki secara cukup signifikan.

### **Penyelesaian Mini Kasus**

1. Kode etik profesi itu merupakan sarana untuk membantu para pelaksana sebagai seseorang yang professional supaya tidak dapat merusak etika profesi. Dalam perusahaan, kode etik profesi sendiri memiliki peran yang penting untuk melaksanakan kegiatan operasional perusahaan. Kode etik profesi memberikan pedoman bagi setiap anggota profesi tentang prinsip profesionalitas yang digariskan. Maksudnya bahwa dengan kode etik profesi, pelaksana profesi mampu mengetahui suatu hal yang boleh dilakukan dan yang tidak boleh dilakukan. Kode etik profesi merupakan sarana kontrol sosial bagi masyarakat atas profesi yang bersangkutan. Maksudnya bahwa etika profesi dapat

memberikan suatu pengetahuan kepada masyarakat agar juga dapat memahami arti pentingnya suatu profesi, sehingga memungkinkan pengontrolan terhadap para pelaksana di lapangan kerja (kalangan sosial). Kode etik profesi mencegah campur tangan pihak di luar organisasi profesi tentang hubungan etika dalam keanggotaan profesi. Arti tersebut dapat dijelaskan bahwa para pelaksana profesi pada suatu instansi atau perusahaan yang lain tidak boleh mencampuri pelaksanaan profesi di lain instansi atau perusahaan.

Dalam lingkup TI, kode etik profesinya memuat kajian ilmiah mengenai prinsip atau norma-norma dalam kaitan dengan hubungan antara professional atau developer TI dengan klien, antara para profesional sendiri, antara organisasi profesi serta organisasi profesi dengan pemerintah. Salah satu bentuk hubungan seorang profesional dengan klien (pengguna jasa) misalnya pembuatan sebuah program aplikasi.

Dalam operasional tentu banyak tantangan dan hambatan yang ditemui dilapangan dimana setiap keputusan yang diambil harus sesuai dengan kode etik yang ada. Seorang profesional tidak dapat membuat program semaunya, ada beberapa hal yang harus ia perhatikan seperti untuk apa program tersebut nantinya digunakan oleh kliennya atau *user*, ia dapat menjamin keamanan (*security*) sistem kerja program aplikasi tersebut dari pihak-pihak yang dapat mengacaukan sistem kerjanya (misalnya: *hacker*, *cracker*, dll). Kode etik profesi informatikawan merupakan bagian dari etika profesi. Jika para profesional TI melanggar kode etik, mereka dikenakan sanksi moral, sanksisosial, dijauhi, *banned* dari pekerjaannya, bahkan mungkin dicopot dari jabatannya.

2. Teknologi semakin maju dan berkembang bersamaan dengan perkembangan zaman, teknologi telah menjadi bagian dalam kehidupan sehari-hari dimana teknologi berperan besar dalam berbagai kegiatan yang dikerjakan sehari-hari, oleh karena itu teknologi saat ini berkembang menjadi lebih canggih dan berdasarkan perkembangan teknologi, penggunaan energi yang dibutuhkan untuk menjalankan fungsi dari teknologi tersebut juga sangat besar. Salah satu perangkat teknologi yang sangat berperan dalam kehidupan sehari-hari yaitu

komputer. Komputer merupakan salah satu perangkat teknologi yang penggunaan energinya sangat besar sehingga dapat membuat pihak pengguna merasa terbebani dalam faktor penggunaan biaya. Oleh karena itu pada zaman moderen ini banyak *user* yang mulai memikirkan cara dalam penghematan energi pada suatu teknologi. *Green Computing* adalah suatu metode dimana para *user* menggunakan suatu perangkat komputerisasi dengan memanfaatkan sumber daya energi sebaik mungkin sehingga menjadi lebih efisien dan ramah lingkungan. Biasanya *Green Computing* diterapkan pada lingkungan kerja yang lebih banyak menggunakan energi dalam pengoperasian peralatan elektronik seperti komputer dan peralatan elektronik lainnya yang semakin lama semakin lebih canggih. Dengan demikian perusahaan dapat melakukan penghematan energi dan biaya, karena dengan melakukan penghematan biaya maka dapat membantu kegiatan kerja menjadilebih efisien.

Dengan digunakannya program ini maka akan dirasakan dampak etis yang positif dalam kegiatan operasional perusahaan. Penggunaan energi berkurang, dari teknik komputasi hijau diterjemahkan ke dalam emisi karbondioksida yang lebih rendah, yang berasal dari pengurangan bahan bakar fosil yang digunakan dalam pembangkit listrik dan transportasi. Konservasi sumber daya juga lebih efisien karena lebih sedikit energi yang dibutuhkan untuk memproduksi, menggunakan, dan membuang produk. Menghemat energi dan sumber daya yang dilakukan ini juga berarti menghemat uang. Komputasi hijau bahkan termasuk mengubah kebijakan pemerintah untuk mendorong daur ulang dan menurunkan penggunaan energi oleh individu dan bisnis. Dampak positif lainnya yaitu mengurangi resiko yang ada dalam laptop seperti kimia diketahui menyebabkan kanker, kerusakan saraf dan reaksi kekebalan tubuh pada manusia.

Dalam penerapannya sendiri terdapat beberapa hal yang harus diperhatikan dan dipenuhi oleh perusahaan untuk melakukan pendekatan *green computing*, diantaranya:

a. Efisiensi Algoritma

Efisiensi algoritma adalah bagaimana menggunakan suatu sumber daya yang efisien dalam menjalankan sebuah perintah atau algoritma. *Green Computing* menekankan efisiensi algoritma untuk pembuatan program, agar sumber

daya yang dipergunakan menjadi lebih kecil dan berujung pada penghematan sumber daya.

Terdapat 2 pendekatan mengenai efisiensi algoritma, yaitu:

- Kecepatan dalam menjalankan suatu perintah atau algoritma dengan benar. Hal ini dipengaruhi oleh : Banyaknya langkah pemrograman, Besar dan jenis input data, Jenis operasi, Komputer dan kompilator yang ada.
- Berapa besarnya memori yang digunakan untuk menjalankan algoritma tersebut. Banyaknya langkah yang digunakan dan jenis variabel data yang dipakai dalam suatu algoritma akan sangat mempengaruhi penggunaan memori. Dalam hal ini, diharapkan dapat memperkirakan seberapa banyak kebutuhan memori yang diperlukan selama proses berlangsung hingga proses selesai dikerjakan. Dengan demikian, dapat disiapkan *storage* yang memadai agar proses suatu algoritma berjalan tanpa ada hambatan atau kekurangan memori.

Cara untuk mencegah penggunaan algoritma yang tidak baik adalah sebagai berikut:

- Menyesuaikan bahasa pemrograman, tipe, dan alat tempat program berada dengan kebutuhan yang ada.
- Mengurangi langkah langkah dan operator yang tidak perlu dalam pembuatan program.
- Menggunakan Teknik Umum untuk meningkatkan efisiensi pengerjaan algoritma, seperti menggunakan *Indexed array* atau *binary search*.
- Penggunaan *Dependency Tree* dan *Spreadsheet*.
- Menggunakan cara yang lebih baik dalam mencari *String*. Misalnya dengan menggunakan *Declarative Notation*.
- Menggunakan *Hot Spot Analyzer* untuk mengetahui tempat-tempat dengan *performance* yang kurang baik (memakan waktu lama untuk dikerjakan) dengan tujuan memperbaiki titik-titik lemah tersebut.
- Melakukan *Benchmarking* untuk membandingkan performa program dengan program yang terdahulu atau program yang lebih baik.
- Menggunakan *Compiled language* daripada *Interpreted Language*.

- Melakukan Optimasi pada *Compiler* atau menggunakan *Just In Time Compiler*.

b. Virtualisasi

Merupakan penggabungan beberapa (dua atau lebih) *physical system* kedalam sebuah *physical system* yang lebih besar dan lebih cepat. Setiap *image system* berisikan sistem operasi dan aplikasi pendukungnya, dan setiap *image* memiliki sistem operasi yang sama atau sistem operasi yang berbeda. Mesin virtual sebenarnya bukan merupakan hal yang baru dalam dunia komputer. *Virtual machine* biasanya digunakan untuk pembagian *hardware* yang sama yang diakses banyak program atau untuk memungkinkan perangkat lunak agar lebih portabel diantara berbagai jenis sistem operasi.

c. Terminal Server

Terminal Server merupakan sebuah teknologi yang memungkinkan beberapa komputer untuk mengakses server secara langsung, dimana semua proses terjadi di dalam server sedangkan computer yang terkoneksi bisa menikmati *operating system*, *storage*, bahkan *device* yang ada pada server tersebut. Jika konsep ini digabungkan dengan menggunakan *thin client* (yang memiliki penggunaan energi 8 kali lebih kecil), maka efisiensi energi akan dapat ditingkatkan.

d. Power Management

Untuk menurunkan pemakaian energi pada computer, terdapat fasilitas *Power Management* yang memungkinkan *operating system* untuk mengkases aspek-aspek yang berhubungan dengan *powersaving* dari *hardware* yang ada di komputer tersebut. Hal ini memungkinkan sistem untuk mematikan komponen secara otomatis, seperti monitor dan *harddrive* setelah waktu tertentu. Contoh yang paling bisa kita lihat adalah *hibernate*, dimana sebagian besar komponen (*Processor* dan *RAM*) dimatikan. Komponen yang dapat ditekan penggunaan energinya adalah

- *Power Supply*. *Power Supply* biasanya mempunyai nilai efisiensi antara 70-75%. Untuk menghemat energi, *Energy Star* (standard efisiensi) menstandarkan *Power Supply* agar mempunyai efisiensi minimal 80%.
- *Storage*. *Solid State Drive* mempunyai tingkat konsumsi yang rendah ketimbang *hard disk*. Mengurangi konsumsi listrik untuk *storage* yang

besar sembari tetap membuat penyimpanannya bersifat *online* adalah sasaran dari penelitian para ahli saat ini.

- *Video Card*. *Video Card* dengan kemampuan yang tinggi adalah komponen yang paling banyak memakai energi listrik pada sebuah komputer. Untuk itu, gunakanlah VGA *on-board* dan VGA yang mempunyai standar GPU untuk *performance per watt*.
- *Display* (monitor). Untuk monitor, pakailah monitor LCD yang memiliki penggunaan listrik lebih rendah daripada monitor CRT atau sekalian memakai monitor LED.
- *Operating System*. *Operating System* dapat memberikan akses kepada *power management*. Microsoft (contohnya), memberikan akses *power management* kepada *user*. Bahkan dalam *Windows Vista*, *power management* dapat diatur secara sentral oleh sistem administrator.

e. Produk dan Material

Dengan meningkatnya kesadaran akan *Green Computing*, sebuah standar diperlukan untuk memandu perusahaan-perusahaan hingga perorangan dalam memilih perangkat komputer. Untuk itu, dibentuklah EPEAT (*Electronic Product Environmental Assessment Tool*) untuk membandingkan produk-produk digital berdasarkan dampaknya terhadap lingkungan. Selain itu, EPEAT juga menjadi panduan yang jelas dan konsisten bagi produsen digital. Standar ini membagi produk menjadi tiga kategori, *bronze* untuk produk yang mencapai semua standar utama, *silver* untuk produk yang mencapai semua standar utama dan 50% standar pilihan, serta *gold* untuk produk yang mencapai semua standar utama dan 75% standar pilihan. Standar yang diatur adalah

- Standar Materi. Sebuah produk digital harus meminimalisir penggunaan materi yang dapat merusak lingkungan. Materi ini termasuk *cadmium*, merkuri, timah, *hexavalent chromium*, dan materi penghambat api yang mengandung *bromine*. Baterai harus bebas dari timah, *cadmium*, dan merkuri. Produk juga tidak diperbolehkan mengandung *polyvinyl chloride* (PVC) kecuali kabel-kabel dan sambungan-sambungan dengan berat kurang dari 25 gram.

- Standar Daur Ulang. Sebuah produk minimal harus dapat didaur ulang sebesar 65%. Produsen harus memberikan petunjuk bagi pengguna tentang materi-materi yang membutuhkan penanganan daur ulang khusus. Materi plastik pada produk digital harus dapat didaur ulang sebanyak 5 – 25% kecuali panel sirkuit dan kemasan produk, materi plastik ini juga harus ditandai dengan identifikasi standar ISO. Materi logam harus dapat dipisahkan dengan mudah dari materi plastik. Produk tidak boleh mengandung pelapis atau cat yang tidak dapat didaur atau dipakai ulang.
- Standar Usia. Produk harus memiliki setidaknya tiga tahun garansi atau layanan servis. Semua komputer pribadi dan laptop harus dapat di *upgrade* dengan produk-produk yang mudah dicari masyarakat, seperti *memory drive*, *chip*, dan *card* harus dapat diganti atau diperluas. Hal ini juga berarti produk komputer pribadi dan laptop harus memiliki desain yang memungkinkan *upgrade* komponen-komponen utama. *Sparepart* harus tersedia selama lima tahun dari masa pembelian dan pembeli harus diberi tahu bagaimana caranya mendapatkan *sparepart* tersebut.
- Konservasi Energi. Produk yang sesuai harus memenuhi standar terbaru U.S ENERGY STAR. Produk ini juga harus dapat dipasangkan setidaknya satu jenis aksesoris penghemat energi.
- Pernyataan Publik. Produsen harus mendemonstrasikan pada publik bahwa produknya ramah lingkungan sesuai dengan standar ISO 14001, dan akan lebih baik lagi bila perusahaannya juga dapat memenuhi dan mendemonstrasikan salah satu dari kriteria ISO 14001, European EMAS atau U.S. EPA Performance Track. Produsen juga diharuskan untuk membuat laporan yang memenuhi tiga standar U.S EPA Performance Track atau Global Reporting Initiative (GRI) Sustainability Reporting Guidelines (2002).
- Kemasan. Material yang digunakan untuk kemasan harus ditulis. Kemasan tidak boleh mengandung logam berat, kecuali untuk keperluan mendaur ulang isi. Materi yang dapat didaur ulang harus dicantumkan pada kemasan, berikut pula persentasinya jika ada. Materi yang tidak dapat didaur ulang, harus dapat dipisahkan dengan mudah dari kemasan.

Dianjurkan setidaknya 90% dari kemasan harus dapat didaur ulang atau dijadikan pupuk. Kemasan sebaiknya mengikuti anjuran dari U.S. EPA Comprehensive Procurement. Selain itu juga dianjurkan agar perusahaan menyediakan jasa gratis yang menerima kemasan untuk didaur ulang atau ditukar dengan suatu produk hasil daur ulang.

3. Saat ini banyak perusahaan yang mulai memperhatikan media sosial yang rata-rata digunakan oleh masyarakat dari semua kalangan. Banyak juga perusahaan yang menanyakan akun social media yang digunakan oleh calon pelamar sebagai salah satu pertimbangan dalam wawancara. Namun untuk melihat kredibilitas akun tersebut tidak memerlukan username dan password untuk diakses secara langsung, maka dari itu saya akan menolak permintaan untuk memberikan username dan password yang diminta. Jika pewawancara ingin mengetahui lebih dalam pribadi pelamar melalui social media yang digunakannya, hal ini dapat dilakukan cukup dengan melihat profil akun atau informasi lain yang ada di sosial media tersebut. Biasanya pemilik akun akan memberikan informasi yang sederhana seperti tanggal lahir, hobby, kesukaan sampai kegiatan yang sering dilakukan dalam postingan-postingan yang di upload. Sehingga, pemberian username dan password sebenarnya tidak diperlukan.

UJIAN AKHIR SEMESTER  
ETHICAL ISSUES IN ELECTRONIC INFORMATION SYSTEM  
NAMA : ZENA LUSI  
DOSEN PENGASUH: M. IZMAN HERDIANSYAH, PhD  
KELAS : MTI19 B SEMESTER 3

### **Penyelesaian Soal**

1. Dalam proses administratif universitas dosen luar biasa mempunyai tugas pokok menjalankan Tri Dharma Perguruan Tinggi, yaitu memberi layanan pendidikan-pengajaran, penelitian, dan pengabdian pada masyarakat. Dalam menjalankan hal-hal tersebut dosen luar biasa berperan sebagai :
  - a. Fasilitator dan narasumber pembelajaran mahasiswa.
  - b. Peneliti dan pakar bidang ilmunya masing-masing, untuk mengembangkan ilmu, teknologi, kebudayaan dan seni.
  - c. Pengabdian masyarakat dengan cara menerapkan keahliannya bagi kesejahteraan masyarakat dan kemajuan kemanusiaan.

Dosen luar biasa juga mempunyai fungsi dalam mengembangkan akademik dan profesi, serta berpartisipasi dalam tata pamong institusi. Mengingat tentang banyaknya proses administratif universitas yang harus dipenuhi oleh dosen luar biasa, hal ini menimbulkan tantangan tersendiri di universitas. Universitas harus mampu menghasilkan dosen yang kompeten yang mampu membuat perancangan perkuliahan sesuai dengan ketentuan kerja yang telah ditentukan oleh universitas dengan melihat metode-metode dan teknik pembelajaran yang digunakan oleh dosen luar biasa tersebut. Selain perancangan kuliah, kegiatan tri Dharma Perguruan Tinggi juga harus diperhatikan dan berkordinasi dengan semua unit kerja demi menghasilkan dan melaksanakan program pembelajaran yang inovatif mengikuti perkembangan Ilmu Teknologi saat ini.

Perekrutan dosen luar biasa sangat baik karena dapat menghasilkan dosen-dosen baru yang kompeten. Dengan perekrutan dosen yang kemudian dibudidayakan dengan standar operasional universitas, hal ini dapat membuat universitas memiliki tenaga kerja dengan standar yang diinginkan. Mengingat saat ini ilmu pengetahuan terus berkembang, hal ini tidak menutup kemungkinan bahwa tenaga pengajar seperti dosen juga terus berkembang. Dengan adanya perekrutan dosen luar biasa maka universitas dapat mengenal banyak dosen dan kemudian menemukan dosen yang cocok, yang apabila sudah benar-benar memenuhi standar universitas barulah dapat diangkat menjadi dosen tetap.

Kegiatan seperti ini dianggap optimal karena dapat mengurangi perputaran karyawan/dosen yang bekerja sehingga proses pergantian karyawan atau dosen yang berhenti bekerja dapat dikurangi. Jika universitas terlalu terburu-buru menerima dosen tetap tanpa melihat kompetensi yang ada dalam dosen yang kemudian berujung pada pemecatan, hal ini tentunya tidak baik dalam proses administrasi dalam universitas. Selain mengganggu proses administrasi yang ada, kegiatan Tri Dharma Perguruan Tinggi tentunya akan terganggu dimana juga dosen yang bersangkutan tidak dapat lagi mengajar dan harus digantikan oleh dosen pengganti, tentunya hal ini akan menyulitkan mahasiswa karena proses pengajaran setiap dosen berbeda.

2. Saat ini semakin banyak perusahaan yang dalam perekrutan karyawan atau pegawainya menggunakan status *outsourcing* atau alih daya. Jenis pekerjaan yang sekarang di *outsourcing*-kan juga terus bertambah. Jika dulu jenis pekerjaan yang di-*outsourcing*-kan hanya sebatas pekerjaan *supporting* atau pekerja pendukung saja, sekarang sudah semakin banyak pekerja *back-office* yang di *outsourcing*-kan.

Dalam manajemen SDM sendiri dapat dijabarkan dalam aspek manajerial yang meliputi perencanaan, pengorganisasian, pengarahan dan pengawasan, serta aspek operasional yang meliputi pengadaan, kompensasi, pengembangan dan integrasi serta pemeliharaan dalam kerja. Jika dilihat dari aspek-aspek yang ada maka *outsourcing* baiknya dilakukan pada aspek operasional, sedangkan aspek yang tetap *in-house* yaitu aspek manajerial. Aspek manajerial ini biasanya dilakukan oleh pemegang keputusan seperti manajer untuk melakukan *outsourcing* di aspek operasional. Dimana jabatan yang biasanya di-*outsourcing*-kan seperti satpam, *office boy*, operator, *customer service* yang kemudian menjalankan aspek operasional seperti pengadaan karyawan, pemberian kompensasi, pengembangan atau pelatihan sampai pemeliharaan karyawan *outsourcing*.

3. Kegiatan riset dan pengembangan dalam perusahaan farmasi sangat penting dan bersifat rahasia. Namun, tak jarang prospek operasional inipun dijalankan oleh pegawai/karyawan yang berstatus *outsourcing*. Hal ini tentunya dilakukan dengan banyak pertimbangan. Dalam kegiatannya proses *outsourcing* juga dilindungi secara hukum dari kedua belah pihak yang dituangkan dalam perjanjian kerja yang dibuat sebelum kerjasama dimulai. Dengan adanya perjanjian kerja ini maka pihak perusahaan tidak perlu cemas akan data penting dan rahasia perusahaan dapat tersebar keluar karena dalam prosesnya telah dilindungi secara hukum. Bukan hanya itu, ada beberapa alasan dan pertimbangan yang membuat proses *outsourcing* dapat dilakukan, diantaranya :

a. Meningkatkan fokus perusahaan

Dengan melakukan *outsourcing*, perusahaan dapat memusatkan diri pada masalah dan strategi utama dan umum, sementara pelaksanaan tugas sehari-hari yang kecil-kecil diserahkan kepada pihak ketiga.

b. Memanfaatkan kemampuan kelas dunia

Spesialisasi pekerjaan seperti yang dimiliki dan dikembangkan oleh para kontraktor (*outsourcing provider*) mengakibatkan kontraktor tersebut memiliki keunggulan kelas dunia dalam bidangnya. Tentu saja disini diasumsikan bahwa *outsourcing* diberikan betul-betul kepada kontraktor yang unggul dibidang pekerjaan yang dikontrakkan.

c. Mempercepat keuntungan yang diperoleh dari *re-engineering*

*Outsourcing* adalah produk samping dan salah satu manajemen *tool* lagi yang sangat unggul, yaitu *business process reengineering*. *Reengineering* adalah pemikiran kembali secara fundamental mengenai proses bisnis, dengan tujuan untuk melakukan perbaikan secara dramatis tentang ukuran-ukuran keberhasilan yang sangat kritis bagi perusahaan, yaitu biaya, mutu, jasa, dan kecepatan.

d. Membagi resiko

Apabila semua aktivitas dilakukan oleh perusahaan sendiri, semua investasi yang diperlukan untuk setiap aktivitas tersebut harus dilakukan oleh perusahaan sendiri pula. Perlu diingat bahwa semua bentuk investasi menanggung resiko tertentu. Apabila semua investasi dilakukan sendiri maka seluruh resiko juga ditanggung sendiri. Apabila beberapa aktivitas perusahaan dikontrakkan kepada pihak ketiga maka resiko akan ditanggung bersama pula.

e. Sumber daya sendiri dapat digunakan untuk kebutuhan-kebutuhan lain

Setiap perusahaan tentu mempunyai keterbatasan dalam kepemilikan sumber daya. Tantangan yang terus-menerus harus dihadapi adalah bahwa sumber daya tersebut harus selalu dimanfaatkan untuk memanfaatkan bidang-bidang yang paling menguntungkan. *Outsourcing* memungkinkan perusahaan untuk menggunakan sumber daya yang dimiliki secara terbatas tersebut untuk bidang-bidang kegiatan utama, yaitu hal yang paling dibutuhkannya.

f. Memungkinkan tersedianya dana kapital

*Outsourcing* juga bermamfaat untuk mengurangi investasi dana kapital pada kegiatan *non-core*. Sebagai ganti dari melakukan investasi dibidang kegiatan tersebut, lebih baik mengkontrakkan sesuai dengan kebutuhan yang dibiayai dengan dana operasi, bukan dana investasi. Dengan demikian, dana kapital dapat digunakan pada aktivitas yang lebih bersifat utama.

g. Menciptakan data segar

*Outsourcing*, sering kali dapat dilakukan tidak hanya mengontrakkan aktivitas tertentu pada pihak ketiga, tetapi juga disertai dengan penyerahan penjualan/penyewaan aset yang digunakan untuk melakukan aktivitas tertentu tersebut.

h. Mengurangi dan mengendalikan biaya operasional

Keuntungan yang sangat taktis dari *outsourcing* adalah memungkinkan untuk mengurangi dan mengendalikan biaya operasi. Pengurangan biaya ini dapat dan dimungkinkan diperoleh dari mitra *outsource* melalui berbagai hal, misalnya spesialisasi, struktur pembiayaan yang lebih rendah, ekonomi skala besar (*economics of scale*). Pengurangan ini tidak mungkin dapat diperoleh apabila aktivitas yang bersangkutan dilakukan sendiri karena tidak mempunyai kemudahan seperti yang dimiliki oleh mitra *outsource* diatas.

i. Memperoleh sumber daya yang tidak dimiliki sendiri

Perusahaan melakukan *outsourcing* untuk suatu aktivitas tertentu karena perusahaan tidak memiliki sumber daya yang dibutuhkan untuk melakukan aktivitas tersebut secara baik dan memadai. Misalnya dalam hal aktivitas logistik, untuk memperoleh biaya logistik yang optimal diperlukan suatu model analitis yang canggih. Banyak perusahaan tidak mempunyai ahli yang cukup dan cakap untuk mengembangkan model-model ini. Oleh karena itu, jalan satu-satunya adalah melakukan *outsourcing*.

j. Memecahkan masalah yang sulit dikendalikan atau dikelola

*Outsourcing* dapat juga digunakan untuk mengatasi pengelolaan hal atau mengawasi fungsi yang sulit dikendalikan. Fungsi yang sulit dikelola dan dikendalikan ini, misalnya birokrasi ekstern yang sangat berbelit yang harus ditaati oleh perusahaan yang dimiliki negara dalam menjalankan fungsi pembelian barang dan jasa, yang sulit ditembus dengan cara-cara biasa. Hal ini mungkin dapat dipecahkan dengan mengkontrakkan saja seluruh pekerjaan tersebut pada pihak ketiga yang berbentuk swasta, yang tidak terikat pada birokrasi tertentu. Contoh adalah mengkontrakkan pemeliharaan peralatan karena setelah dilakukan usaha terus-menerus untuk memperbaiki sistem dan kinerja fungsi pemeliharaan, tidak juga dapat diperbaiki secara cukup signifikan.

### **Penyelesaian Mini Kasus**

1. Kode etik profesi itu merupakan sarana untuk membantu para pelaksana sebagai seseorang yang profesional supaya tidak dapat merusak etika profesi. Dalam perusahaan, kode etik profesi sendiri memiliki peran yang penting untuk melaksanakan kegiatan operasional perusahaan. Kode etik profesi memberikan pedoman bagi setiap anggota profesi tentang prinsip profesionalitas yang digariskan. Maksudnya bahwa dengan kode etik profesi, pelaksana profesi mampu mengetahui suatu hal yang boleh dilakukan dan yang tidak boleh dilakukan. Kode etik profesi merupakan sarana kontrol sosial bagi masyarakat atas profesi

yang bersangkutan. Maksudnya bahwa etika profesi dapat memberikan suatu pengetahuan kepada masyarakat agar juga dapat memahami arti pentingnya suatu profesi, sehingga memungkinkan pengontrolan terhadap para pelaksana di lapangan kerja (kalangan sosial). Kode etik profesi mencegah campur tangan pihak di luar organisasi profesi tentang hubungan etika dalam keanggotaan profesi. Arti tersebut dapat dijelaskan bahwa para pelaksana profesi pada suatu instansi atau perusahaan yang lain tidak boleh mencampuri pelaksanaan profesi di lain instansi atau perusahaan.

Dalam lingkup TI, kode etik profesinya memuat kajian ilmiah mengenai prinsip atau norma-norma dalam kaitan dengan hubungan antara professional atau developer TI dengan klien, antara para profesional sendiri, antara organisasi profesi serta organisasi profesi dengan pemerintah. Salah satu bentuk hubungan seorang profesional dengan klien (pengguna jasa) misalnya pembuatan sebuah program aplikasi.

Dalam operasional tentu banyak tantangan dan hambatan yang ditemui dilapangan dimana setiap keputusan yang diambil harus sesuai dengan kode etik yang ada. Seorang profesional tidak dapat membuat program semaunya, ada beberapa hal yang harus ia perhatikan seperti untuk apa program tersebut nantinya digunakan oleh kliennya atau *user*, ia dapat menjamin keamanan (*security*) sistem kerja program aplikasi tersebut dari pihak-pihak yang dapat mengacaukan sistem kerjanya (misalnya: *hacker*, *cracker*, dll). Kode etik profesi informatikawan merupakan bagian dari etika profesi. Jika para profesional TI melanggar kode etik, mereka dikenakan sanksi moral, sanksisosial, dijauhi, di-*banned* dari pekerjaannya, bahkan mungkin dicopot dari jabatannya.

2. Teknologi semakin maju dan berkembang bersamaan dengan perkembangan zaman, teknologi telah menjadi bagian dalam kehidupan sehari-hari dimana teknologi berperan besar dalam berbagai kegiatan yang dikerjakan sehari-hari, oleh karena itu teknologi saat ini berkembang menjadi lebih canggih dan berdasarkan perkembangan teknologi, penggunaan energi yang dibutuhkan untuk menjalankan fungsi dari teknologi tersebut juga sangat besar. Salah satu perangkat teknologi yang sangat berperan dalam kehidupan sehari-hari yaitu komputer. Komputer merupakan salah satu perangkat teknologi yang penggunaan energinya sangat besar sehingga dapat membuat pihak pengguna merasa terbebani dalam faktor penggunaan biaya. Oleh karena itu pada zaman moderen ini banyak *user* yang mulai memikirkan cara dalam penghematan energi pada suatu teknologi. *Green Computing* adalah suatu metode dimana para *user* menggunakan suatu perangkat komputerisasi dengan memanfaatkan sumber daya energi sebaik mungkin sehingga menjadi lebih efisien dan ramah lingkungan. Biasanya *Green Computing* diterapkan pada lingkungan kerja yang lebih banyak

menggunakan energi dalam pengoperasian peralatan elektronik seperti komputer dan peralatan elektronik lainnya yang semakin lama semakin lebih canggih. Dengan demikian perusahaan dapat melakukan penghematan energi dan biaya, karena dengan melakukan penghematan biaya maka dapat membantu kegiatan kerja menjadilebih efisien.

Dengan digunakannya program ini maka akan dirasakan dampak etis yang positif dalam kegiatan operasional perusahaan. Penggunaan energi berkurang, dari teknik komputasi hijau diterjemahkan ke dalam emisi karbondioksida yang lebih rendah, yang berasal dari pengurangan bahan bakar fosil yang digunakan dalam pembangkit listrik dan transportasi. Konservasi sumber daya juga lebih efisien karenalebih sedikit energi yang dibutuhkan untuk memproduksi, menggunakan, dan membuang produk. Menghemat energi dan sumber daya yang dilakukan ini juga berarti menghemat uang. Komputasi hijau bahkan termasuk mengubah kebijakan pemerintah untuk mendorong daur ulang dan menurunkan penggunaan energi oleh individu dan bisnis. Dampak positif lainnya yaitu mengurangi resiko yang ada dalam laptop seperti kimia diketahui menyebabkan kanker, kerusakan saraf dan reaksi kekebalan tubuh pada manusia.

Dalam penerapannya sendiri terdapat beberapa hal yang harus diperhatikan dan dipenuhi oleh perusahaan untuk melakukan pendekatan *green computing*, diantaranya:

a. Efisiensi Algoritma

Efisiensi algoritma adalah bagaimana menggunakan suatu sumber daya yang efisien dalam menjalankan sebuah perintah atau algoritma. *Green Computing* menekankan efisiensi algoritma untuk pembuatan program, agar sumber daya yang dipergunakan menjadi lebih kecil dan berujung pada penghematan sumber daya.

Terdapat 2 pendekatan mengenai efisiensi algoritma, yaitu:

- Kecepatan dalam menjalankan suatu perintah atau algoritma dengan benar. Hal ini dipengaruhi oleh : Banyaknya langkah pemrograman, Besar dan jenis input data, Jenis operasi, Komputer dan kompilator yang ada.
- Berapa besarnya memori yang digunakan untuk menjalankan algoritma tersebut. Banyaknya langkah yang digunakan dan jenis variabel data yang dipakai dalam suatu algoritma akan sangat mempengaruhi penggunaan memori. Dalam hal ini, diharapkan dapat memperkirakan seberapa banyak kebutuhan memori yang diperlukan selama proses berlangsung hingga proses selesai dikerjakan. Dengan demikian, dapat disiapkan *storage* yang memadai agar proses suatu algoritma berjalan tanpa ada hambatan atau kekurangan memori.

Cara untuk mencegah penggunaan algoritma yang tidak baik adalah sebagai berikut:

- Menyesuaikan bahasa pemrograman, tipe, dan alat tempat program berada dengan kebutuhan yang ada.

- Mengurangi langkah langkah dan operator yang tidak perlu dalam pembuatan program.
- Menggunakan Teknik Umum untuk meningkatkan efisiensi pengerjaan algoritma, seperti menggunakan *Indexed array* atau *binary search*.
- Penggunaan *Dependency Tree* dan *Spreadsheet*.
- Menggunakan cara yang lebih baik dalam mencari *String*. Misalnya dengan menggunakan *Declarative Notation*.
- Menggunakan *Hot Spot Analyzer* untuk mengetahui tempat-tempat dengan *performance* yang kurang baik (memakan waktu lama untuk dikerjakan) dengan tujuan memperbaiki titik-titik lemah tersebut.
- Melakukan *Benchmarking* untuk membandingkan performa program dengan program yang terdahulu atau program yang lebih baik.
- Menggunakan *Compiled language* daripada *Interpreted Language*.
- Melakukan Optimasi pada *Compiler* atau menggunakan *Just In Time Compiler*.

b. Virtualisasi

Merupakan penggabungan beberapa (dua atau lebih) *physical system* kedalam sebuah *physical system* yang lebih besar dan lebih cepat. Setiap *image system* berisikan sistem operasi dan aplikasi pendukungnya, dan setiap *image* memiliki sistem operasi yang sama atau sistem operasi yang berbeda. Mesin virtual sebenarnya bukan merupakan hal yang baru dalam dunia komputer. *Virtual machine* biasanya digunakan untuk pembagian *hardware* yang sama yang diakses banyak program atau untuk memungkinkan perangkat lunak agar lebih portabel diantara berbagai jenis sistem operasi.

c. Terminal Server

Terminal Server merupakan sebuah teknologi yang memungkinkan beberapa komputer untuk mengakses server secara langsung, dimana semua proses terjadi di dalam server sedangkan computer yang terkoneksi bisa menikmati *operating system*, *storage*, bahkan *device* yang ada pada server tersebut. Jika konsep ini digabungkan dengan menggunakan *thin client* (yang memiliki penggunaan energi 8 kali lebih kecil), maka efisiensi energi akan dapat ditingkatkan.

d. Power Management

Untuk menurunkan pemakaian energi pada computer, terdapat fasilitas *Power Management* yang memungkinkan *operating system* untuk mengakses aspek-aspek yang berhubungan dengan *powersaving* dari *hardware* yang ada di komputer tersebut. Hal ini memungkinkan sistem untuk mematikan komponen secara otomatis, seperti monitor dan *harddrive* setelah waktu tertentu. Contoh yang paling bisa kita lihat adalah *hibernate*,

dimana sebagian besar komponen (*Processor* dan *RAM*) dimatikan. Komponen yang dapat ditekan penggunaannya adalah

- *Power Supply*. *Power Supply* biasanya mempunyai nilai efisiensi antara 70-75%. Untuk menghemat energi, *Energy Star* (standar efisiensi) menstandarkan *Power Supply* agar mempunyai efisiensi minimal 80%.
- *Storage*. *Solid State Drive* mempunyai tingkat konsumsi yang rendah ketimbang *hard disk*. Mengurangi konsumsi listrik untuk *storage* yang besar sembari tetap membuat penyimpanannya bersifat *online* adalah sasaran dari penelitian para ahli saat ini.
- *Video Card*. *Video Card* dengan kemampuan yang tinggi adalah komponen yang paling banyak memakai energi listrik pada sebuah komputer. Untuk itu, gunakanlah *VGA on-board* dan *VGA* yang mempunyai standar GPU untuk *performance* per watt.
- *Display* (monitor). Untuk monitor, pakailah monitor LCD yang memiliki penggunaan listrik lebih rendah daripada monitor CRT atau sekalian memakai monitor LED.
- *Operating System*. *Operating System* dapat memberikan akses kepada *power management*. Microsoft (contohnya), memberikan akses *power management* kepada *user*. Bahkan dalam *Windows Vista*, *power management* dapat diatur secara sentral oleh sistem administrator.

e. Produk dan Material

Dengan meningkatnya kesadaran akan *Green Computing*, sebuah standar diperlukan untuk memandu perusahaan-perusahaan hingga perorangan dalam memilih perangkat komputer. Untuk itu, dibentuklah EPEAT (*Electronic Product Environmental Assessment Tool*) untuk membandingkan produk-produk digital berdasarkan dampaknya terhadap lingkungan. Selain itu, EPEAT juga menjadi panduan yang jelas dan konsisten bagi produsen digital. Standar ini membagi produk menjadi tiga kategori, *bronze* untuk produk yang mencapai semua standar utama, *silver* untuk produk yang mencapai semua standar utama dan 50% standar pilihan, serta *gold* untuk produk yang mencapai semua standar utama dan 75% standar pilihan. Standar yang diatur adalah

- Standar Materi. Sebuah produk digital harus meminimalisir penggunaan materi yang dapat merusak lingkungan. Materi ini termasuk *cadmium*, merkuri, timah, *hexavalentchromium*, dan materi penghambat api yang mengandung *bromin*. Baterai harus bebas dari timah, *cadmium*, dan merkuri. Produk juga tidak diperbolehkan mengandung *polyvinyl chloride* (PVC) kecuali kabel-kabel dan sambungan-sambungan dengan berat kurang dari 25 gram.
- Standar Daur Ulang. Sebuah produk minimal harus dapat didaur ulang sebesar 65%. Produsen harus memberikan petunjuk bagi pengguna tentang materi-materi yang membutuhkan penanganan daur ulang khusus. Materi plastik pada produk digital

harus dapat didaur ulang sebanyak 5 – 25% kecuali panel sirkuit dan kemasan produk, materi plastik ini juga harus ditandai dengan identifikasi standar ISO. Materi logam harus dapat dipisahkan dengan mudah dari materi plastik. Produk tidak boleh mengandung pelapis atau cat yang tidak dapat didaur atau dipakai ulang.

- Standar Usia. Produk harus memiliki setidaknya tiga tahun garansi atau layanan servis. Semua komputer pribadi dan laptop harus dapat di *upgrade* dengan produk-produk yang mudah dicari masyarakat, seperti *memory drive*, *chip*, dan *card* harus dapat diganti atau diperluas. Hal ini juga berarti produk komputer pribadi dan laptop harus memiliki desain yang memungkinkan *upgrade* komponen-komponen utama. *Sparepart* harus tersedia selama lima tahun dari masa pembelian dan pembeli harus diberi tahu bagaimana caranya mendapatkan *sparepart* tersebut.
  - Konservasi Energi. Produk yang sesuai harus memenuhi standar terbaru U.S ENERGY STAR. Produk ini juga harus dapat dipasangkan setidaknya satu jenis aksesoris penghemat energi.
  - Pernyataan Publik. Produsen harus mendemonstrasikan pada publik bahwa produknya ramah lingkungan sesuai dengan standar ISO 14001, dan akan lebih baik lagi bila perusahaannya juga dapat memenuhi dan mendemonstrasikan salah satu dari kriteria ISO 14001, European EMAS atau U.S. EPA Performance Track. Produsen juga diharuskan untuk membuat laporan yang memenuhi tiga standar U.S EPA Performance Track atau Global Reporting Initiative (GRI) Sustainability Reporting Guidelines (2002).
  - Kemasan. Material yang digunakan untuk kemasan harus ditulis. Kemasan tidak boleh mengandung logam berat, kecuali untuk keperluan mendaur ulang isi. Materi yang dapat didaur ulang harus dicantumkan pada kemasan, berikut pula persentasinya jika ada. Materi yang tidak dapat didaur ulang, harus dapat dipisahkan dengan mudah dari kemasan. Dianjurkan setidaknya 90% dari kemasan harus dapat didaur ulang atau dijadikan pupuk. Kemasan sebaiknya mengikuti anjuran dari U.S. EPA Comprehensive Procurement. Selain itu juga dianjurkan agar perusahaan menyediakan jasa gratis yang menerima kemasan untuk didaur ulang atau ditukar dengan suatu produk hasil daur ulang.
3. Saat ini banyak perusahaan yang mulai memperhatikan media sosial yang rata-rata digunakan oleh masyarakat dari semua kalangan. Banyak juga perusahaan yang menanyakan akun social media yang digunakan oleh calon pelamar sebagai salah satu pertimbangan dalam wawancara. Namun untuk melihat kredibilitas akun tersebut tidak memerlukan username dan password untuk diakses secara langsung, maka dari itu saya akan menolak permintaan

untuk memberikan username dan password yang diminta. Jika pewawancara ingin mengetahui lebih dalam pribadi pelamar melalui social media yang digunakannya, hal ini dapat dilakukan cukup dengan melihat profil akun atau informasi lain yang ada di sosial media tersebut. Biasanya pemilik akun akan memberikan informasi yang sederhana seperti tanggal lahir, hobby, kesukaan sampai kegiatan yang sering dilakukan dalam postingan-postingan yang di upload. Sehingga, pemberian username dan password sebenarnya tidak diperlukan.

# **ETICS IN SOFTWARE DEVELOPMENT**



**OLEH**

**AHKMAD IPANDY**

**182420088**

**PROGRAM PASCASARJANA  
MAGISTER TEKNIK INFORMATIKA  
UNIVERSITAS BINA DARMA PALEMBANG  
TAHUN 2019**

# **BAB I**

## **PENDAHULUAN**

### **Latar Belakang**

Saat ini perkembangan zaman yang diiringi kemajuan teknologi mendorong setiap orang untuk meningkatkan kemampuan dalam hal penguasaan teknologi informasi sehingga dalam penguasaan teknologi tersebut harus memperhatikan kode etik. Kode etik merupakan sistem norma, nilai dan aturan profesional tertulis yang secara tegas menyatakan apa yang baik, benar, apa yang tidak baik atau benar bagi profesional. Kode etik juga menerangkan perbuatan yang salah atau benar, perbuatan yang harus dihindari ataupun dilakukan. Tujuan kode etik adalah agar seorang profesional nantinya dapat memberikan jasa sebaik mungkin kepada konsumen atau pemakainya. Kode etik tersebut diharapkan akan ditaati oleh setiap orang bukan karena paksaan. Dengan adanya kode etik tersebut maka akan melindungi perbuatan yang tidak profesional. Dalam pemahaman kode etik sangat erat hubungannya dengan etika profesi. Etika profesi sangatlah dibutuhkan dalam berbagai bidang teknologi informasi.

Kode etik profesi merupakan bagian dari norma yang lebih umum yang ada dalam etika profesi. Kode etik ini memperjelas dan merinci norma-norma ke bentuk yang lebih sempurna walaupun sebenarnya norma-norma tersebut sudah tersirat dalam etika profesi.

Kata etika berasal dari bahasa Yunani yaitu “Ethos” yang berarti karakter, watak, kesusilaan atau adat kebiasaan. Etika mempersoalkan bagaimana manusia harus bertindak. Menurut Kamus Besar Bahasa Indonesia, etika adalah ilmu tentang apa yang baik dan yang buruk, tentang hak dan kewajiban moral, kumpulan asas atau nilai yang berkenaan dengan akhlak, nilai mengenai benar dan salah yang dianut masyarakat. Menurut Martin (1993), etika didefinisikan sebagai “the discipline which can act as the performance index or reference for our control system” yang artinya disiplin yang dapat bertindak sebagai acuan atau indeks capaian untuk sistem kendali kita.

### Rumusan Masalah

1. Pengertian Etika Profesi
2. Pengertian Networking Engineering
3. Kode Etik Network Engineering
4. Hal-hal yang dianjurkan
5. Hal-hal yang tidak dianjurkan

### Tujuan

1. Kita dapat mengetahui pengertian Etika Profesi
2. Kita dapat mengetahui apa itu Networking Engineering
3. Kita dapat mengetahui kode etik seorang Network Engineering
4. Kita dapat mengetahui apa yang dianjurkan dan tidak dianjurkan pada profesi Network Engineering

## **BAB II**

### **PEMBAHASAN**

#### **Pengertian Etika**

Etika (Etimologi), berasal dari bahasa Yunani adalah “Ethos”, yang berarti watak kesusilaan atau adat kebiasaan (custom). Etika biasanya berkaitan erat dengan perkataan moral yang merupakan istilah dari bahasa Latin, yaitu “Mos” dan dalam bentuk jamaknya “Mores”, yang berarti juga adat kebiasaan atau cara hidup seseorang dengan melakukan perbuatan yang baik (kesusilaan), dan menghindari hal-hal tindakan yang buruk. Etika dan moral dalam kegiatan sehari-hari terdapat perbedaan, yaitu moral atau moralitas untuk penilaian perbuatan yang dilakukan, sedangkan etika adalah untuk pengkajian sistem nilai-nilai yang berlaku. Ada dua macam etika dalam menentukan baik dan buruknya perilaku manusia:

1. Etika Deskriptif, yaitu etika yang berusaha meneropong secara kritis dan rasional sikap dan perilaku manusia dan apa yang dikejar oleh manusia dalam hidup ini sebagai sesuatu yang bernilai. Etika deskriptif memberikan fakta sebagai dasar untuk mengambil keputusan tentang perilaku atau sikap yang mau diambil.
2. Etika Normatif, yaitu etika yang berusaha menetapkan berbagai sikap dan pola perilaku ideal yang seharusnya dimiliki oleh manusia dalam hidup ini sebagai sesuatu yang bernilai. Etika normatif memberi penilaian sekaligus memberi norma sebagai dasar dan kerangka tindakan yang akan diputuskan.

#### **Pengertian Profesi**

Profesi merupakan suatu jabatan atau pekerjaan yang menuntut keahlian atau keterampilan dari pelakunya. Biasanya sebutan “profesi” selalu dikaitkan dengan pekerjaan atau jabatan yang dipegang oleh seseorang, akan tetapi tidak semua pekerjaan atau jabatan dapat disebut profesi karena profesi menuntut keahlian para pemangkuinya. Hal ini mengandung arti bahwa suatu pekerjaan atau jabatan yang disebut profesi tidak dapat dipegang oleh sembarang orang, akan tetapi memerlukan suatu persiapan melalui pendidikan dan pelatihan yang dikembangkan khusus untuk itu. Pekerjaan tidak sama dengan profesi.

Istilah yang mudah dimengerti oleh masyarakat awam adalah sebuah profesi sudah pasti menjadi sebuah pekerjaan, namun sebuah pekerjaan belum tentu menjadi sebuah profesi. Profesi memiliki mekanisme serta aturan yang harus dipenuhi sebagai suatu ketentuan, sedangkan kebalikannya, pekerjaan tidak memiliki aturan yang rumit seperti itu. Hal inilah yang harus diluruskan di masyarakat, karena hampir semua orang menganggap bahwa pekerjaan dan profesi adalah sama. Secara umum ada beberapa ciri atau sifat pada profesi, yaitu:

1. Adanya pengetahuan khusus, yang biasanya keahlian dan keterampilan ini dimiliki berkat pendidikan, pelatihan dan pengalaman yang bertahun-tahun.
2. Adanya kaidah dan standar moral yang sangat tinggi. Hal ini biasanya setiap pelaku profesi mendasarkan kegiatannya pada kode etik profesi.
3. Mengabdikan pada kepentingan masyarakat, artinya setiap pelaksana profesi harus meletakkan kepentingan pribadi di bawah kepentingan masyarakat.

### **Pengertian Etika Profesi**

Etika profesi menurut keiser dalam (Suhrawardi Lubis, 1994:6-7) adalah sikap hidup berupa keadilan untuk memberikan pelayanan profesional terhadap masyarakat dengan penuh ketertiban dan keahlian sebagai pelayanan dalam rangka melaksanakan tugas berupa kewajiban terhadap masyarakat. Kode etik profesi adalah sistem norma, nilai dan aturan profesional tertulis yang secara tegas menyatakan apa yang benar dan baik, dan apa yang tidak benar dan tidak baik bagi profesional. Kode etik menyatakan perbuatan apa yang benar atau salah, perbuatan apa yang harus dilakukan dan apa yang harus dihindari. Tujuan kode etik yaitu agar profesional memberikan jasa sebaik-baiknya kepada pemakai atau nasabahnya. Dengan adanya kode etik akan melindungi perbuatan yang tidak profesional. Prinsip dasar di dalam etika profesi yaitu:

1. Tanggung jawab.
  - a. Terhadap pelaksanaan pekerjaan itu dan terhadap hasilnya.
  - b. Terhadap dampak dari profesi itu untuk kehidupan orang lain atau masyarakat pada umumnya.
2. Keadilan.

3. Prinsip ini menuntut untuk memberikan kepada siapa saja apa yang menjadi haknya.
4. Prinsip Kompetensi, melaksanakan pekerjaan sesuai jasa profesionalnya, kompetensi dan ketekunan.
5. Prinsip Prilaku Profesional, berperilaku konsisten dengan reputasi profesi.
6. Prinsip Kerahasiaan, menghormati kerahasiaan informasi.

### **Etika Profesi Seorang Engineer**

Etika Profesi Engineer (insinyur) untuk membantu pelaksana sebagai seseorang yang profesional dibidang keteknikan supaya tidak dapat merusak etika profesi diperlukan sarana untuk mengatur profesi sebagai seorang profesional dibidangnya berupa kode etik profesi. Ada tiga hal pokok yang merupakan fungsi dari kode etik profesi tersebut, yaitu:

1. Kode etik profesi memberikan pedoman bagi setiap anggota profesi tentang prinsip profesionalitas yang digariskan. Maksudnya bahwa dengan kode etik profesi, pelaksana profesi mampu mengetahui suatu hal yang boleh dia lakukan dan yang tidak boleh dilakukan
2. Kode etik profesi merupakan sarana kontrol sosial bagi masyarakat atas profesi yang bersangkutan. Maksudnya bahwa etika profesi dapat memberikan suatu pengetahuan kepada masyarakat agar juga dapat memahami arti pentingnya suatu profesi, sehingga memungkinkan pengontrolan terhadap para pelaksana di lapangan kerja (kalangan social).
3. Kode etik profesi mencegah campur tangan pihak diluar organisasi profesi tentang hubungan etika dalam keanggotaan profesi. Arti tersebut dapat dijelaskan bahwa para pelaksana profesi pada suatu instansi atau perusahaan yang lain tidak boleh mencampuri pelaksanaan profesi di lain instansi atau perusahaan.

Tanggung jawab profesi yang lebih spesifik seorang profesional diantaranya:

- a. Mencapai kualitas yang tinggi dan efektifitas baik dalam proses maupun produk hasil kerja profesional.
- b. Menjaga kompetensi sebagai profesional.
- c. Mengetahui dan menghormati adanya hukum yang berhubungan dengan kerja yang profesional.

- d. Menghormati perjanjian, persetujuan, dan menunjukkan tanggung jawab.

### **Profesi sebagai Network Engineering**

Kemajuan teknologi informasi dan komputer merupakan hal yang tak bisa dielakkan dan akan terus-menerus terjadi dengan berbagai macam teknologi baru yang bermunculan dan bertujuan mengatasi beragam masalah atau kebutuhan yang ada. Salah satu teknologi yang semakin banyak dipakai oleh banyak perusahaan baik besar maupun kecil adalah jaringan komputer atau biasa disebut networking.

Seorang network engineering mempunyai tugas utama untuk mengatur jaringan komputer baik jaringan skala kecil hingga skala besar, yang nantinya akan digunakan untuk pertukaran data komputer pada sebuah perusahaan, lembaga dan sebagainya. Kehadiran serta tanggung jawab teknisi jaringan menjadi sangat penting guna kelancaran usaha terlebih lagi bagi perusahaan besar. Adapun pekerjaan seorang network engineer biasanya adalah seputar network administrator, design jaringan, instalasi dan maintenance komunikasi antar computer yang ada di dalam sebuah perusahaan atau organisasi.

Tugas seorang network engineering:

1. Melakukan monitoring traffic user.
2. Melakukan troubleshooting.
3. Membuat planning & topologi jaringan supaya mudah dalam troubleshooting. misal alokasi IP address, VLAN, routing, dll.
4. Mengaplikasikan keamanan jaringan. misal membuat aturan firewall (memblok akses ke network tertentu, website tertentu, dll).
5. Membuat bandwidth management.
6. Remote support atau on-site support.

Umumnya, seseorang yang belajar networking pasti membidik sertifikat guna masa depan yang lebih baik, dimana sertifikat tersebut dapat digunakan untuk berkarir di perusahaan. Adapun sertifikasi dibagi menjadi beberapa peringkat, yaitu : lokal, nasional dan internasional. Dari segi biaya, sertifikasi internasional akan lebih mahal daripada lokal. Namun, selain lebih “bergengsi” sertifikasi ini lebih diakui kredibilitasnya oleh berbagai perusahaan

terutama perusahaan asing sehingga kemampuan pemegang sertifikasi tersebut tak diragukan lagi. Salah satu contoh sertifikasi internasional pada bidang networking adalah CCNA (Cisco Certified Networking Associate) yang merupakan satu dari sekian banyak sertifikasi networking yang ditawarkan oleh Cisco.

Seorang network engineers harus dapat bekerja pada 4 sistem, yaitu:

- Local Area Networks(LANs)
- Metropolitan Area Networks(MANs)
- Wide Area Networks(WANs)
- Global Area Networks(GANs)

Biasanya beberapa kebutuhan umum yang dicari oleh sebuah perusahaan dalam mencari network engineering adalah sebagai berikut:

- Implementation and possible design of converged networks
- Implementation and possible design of local and remote systems
- Development and monitoring of network dial-up or VPN Connection
- Analyzing corporate network applications and client-server environments

Untuk dapat memenuhi kebutuhan sebuah perusahaan dalam bidang jaringan komputer, kita harus memiliki beberapa keahlian sebagai berikut:

1. Menguasai konsep dasar mengenai jaringan seperti topologi, protokol-protokol komunikasi, standar-standar networking, media komunikasi data dan keamanan jaringan baik LAN maupun WAN.
2. Menguasai konsep dan desain infrastruktur jaringan dan troubleshooting-nya.
3. Menguasai desain, instalasi dan terminasi media jaringan seperti kabel tembaga/UTP, fiber optic, Wireless communication dll.
4. Menguasai setting, pemanfaatan dan troubleshooting perangkat jaringan seperti router, switch, firewall, proxy, modem dll.

5. Memahami instalasi dan setting PC dan server yang bisa digunakan dalam infrastruktur jaringan seperti domain controller, proxy, firewall, mailserver dll.
6. Menguasai secara teknis dan praktis mengenai keamanan jaringan / sistem

Aktivitas standar yang dilakukan oleh orang-orang berprofesi dibidang Network Engineering adalah:

- Memasang, mendukung, memelihara server hardware dan infrastruktur software baru.
- Mengatur e-mail, anti spam, dan virus protection.
- Melakukan setting user account, izin dan password.
- Memonitor penggunaan jaringan.
- Memastikan cost-effective dan efisiensi penggunaan server.
- Mengusulkan dan menyediakan solusi IT untuk masalah bisnis dan manajemen.
- Memastikan semua peralatan IT memenuhi standar industri.
- Menganalisa dan menyelesaikan kesalahan, mulai dari major system crash sampai kelupaan password.
- Mengerjakan rutin preventative measures dan mengimplementasikan dan memonitor keamanan jaringan, jika jaringan terkoneksi ke internet.
- Menyediakan pelatihan dan dukungan teknis untuk user dengan bermacam tingkat pengetahuan IT dan kompetensi.
- Mengawasi staff lain, seperti help desk technician.
- Bekerja dekat dengan departemen/organisasi lain dan berkolaborasi dengan staff IT lain.
- Merencanakan dan mengimplementasikan pengembangan IT untuk masa mendatang dan menjalankan kerja proyek.
- Mengelola website dan memelihara jaringan internal.
- Memonitor penggunaan web oleh para pekerja.

Sebagai pendukung dan nilai plus, seorang network engineers juga harus memiliki skill programming, tentu saja programming yang berhubungan dengan jaringan computer. Beberapa bahasa pemrograman yang banyak digunakan sebagai bahasa pemrograman jaringan atau client-server yaitu c/c++, perl, bash, dan assembly. Sekarang banyak sekali lembaga-lembaga pendidikan baik yang dikelola pemerintah maupun swasta, yang menyediakan pelatihan jaringan computer, seperti cisco, juniper, compTIA, dll.

Network engineer bekerja secara internal sebagai bagian dari tim pendukung IT di organisasi atau secara eksternal sebagai bagian dari perusahaan konsultasi networking dengan beberapa client.

## **Tokoh Network Engineering**

### **Leonard Bosack ( CEO CISCO )**

Leonard Bosack (lahir tahun 1952) bersama dengan mantan istrinya, Sandy Lerner, jabatannya co-founder dari Cisco Systems, sebuah perusahaan multinasional american-based yang mendesain dan menjual barang elektronik, dan jaringan teknologi komunikasi. Dia juga telah mendapatkan penghargaan the Computer Entrepreneur Award in 2009 untuk co-founding cisco system untuk sistem dan perintis juga memajukan komersialisasi teknologi routing dan memberikan dampak perubahan yang besar dalam industri komputer.

Dia sebagian besar bertanggung jawab untuk merintis komersialisasi luas teknologi Local Area Network (LAN) untuk menghubungkan komputer secara geografis yang berbeda atas sistem multiprotocol router. Pada tahun 1990, manajemen Cisco memecat istrinya Sandy Lerner, dan Bosack mengundurkan diri. Bosack saat ini menjadi CEO XKL LLC, sebuah perusahaan rekayasa yang didanai swasta yang membahas dan mengembangkan jaringan optik untuk komunikasi data.

Lahir di Pennsylvania pada tahun 1952 dari keluarga Katolik Polandia, Leonard Bosack lulus dari Perguruan Tinggi La Salle pada tahun 1969. Pada tahun 1973, Bosack lulus dari Universitas Wharton School di Pennsylvania, dan bergabung dengan DEC sebagai Hardware Engineer. Pada tahun 1979, ia diterima di Stanford University dan mulai belajar ilmu komputer. Selama waktunya di Stanford, ia ditawarkan untuk menjadi support engineer untuk

menyelesaikan proyek 1981 yang menghubungkan semua Stanford's mainframes, minis, mesin LISP dan Altos. Kemudian bertemu istrinya Sandra Lerner di Stanford, di mana ia adalah manajer laboratorium Business School, dan pasangan ini menikah pada tahun 1980. Bekerja sama pada tahun 1984 mereka mulai mendirikan Cisco di Menlo Park.

Pada tahun 1984, Leonard Bosack mendirikan Cisco dengan tujuan mengkomersilkan Advanced Gateway Server. Advanced Gateway Server adalah versi revisi dari router Stanford dibangun oleh William Yeager dan Andy Bechtolsheim. Bosack dan Lerner merancang dan membuat router di rumah mereka dan bereksperimen menggunakan jaringan Stanford. Awalnya, Bosack dan Lerner pergi ke Stanford dengan proposisi untuk mulai membangun dan menjual router, tapi sekolah menolaknya. Saat itulah mereka mendirikan perusahaan mereka sendiri dan menamakannya "Cisco", diambil dari nama kota di utara. Produk Cisco dikembangkan di garasi mereka menjual awal tahun 1986 dari mulut ke mulut.

Pada bulan pertama, Cisco mampu mendarat kontrak senilai lebih dari \$200.000. Perusahaan ini memproduksi teknologi revolusioner seperti multiport router-specific line cards dan protokol routing yang canggih, dan mereka mendominasi pasar saat itu. Cisco go public pada tahun 1990, tahun yang sama bahwa Bosack mengundurkan diri. Bosack dan Lerner berjalan menjauh dari Cisco dengan membawa \$170.000.000 setelah dipaksa keluar oleh manajer profesional pemodal ventura perusahaan yang membawahnya. Pada tahun 1996, pendapatan Cisco sebesar \$5400000000, sehingga salah satu kisah sukses terbesar di Silicon Valley. Pada tahun 1998, perusahaan ini bernilai lebih dari \$6 miliar dan dikendalikan lebih dari tiga perempat dari bisnis router.

Seiring dengan co-founding Cisco Systems, Bosack sebagian besar bertanggung jawab untuk pertama merintis komersialisasi luas jaringan area lokal (LAN). Dia dan sesama anggota staf di Stanford yang berhasil menghubungkan 5.000 komputer di universitas pada 16 mil persegi (41 km<sup>2</sup>) area kampus. Kontribusi ini sangat penting dalam konteksnya, karena pada waktu itu, teknologi seperti itu yang menggunakan LAN tidak pernah terdengar. Tantangan mereka telah mengatasi masalah ketidakcocokan, dalam rangka menciptakan sistem pertama LAN yang benar.

Leonard Bosack juga memegang peran kepemimpinan teknis yang signifikan di AT & T Bell Labs dan Digital Equipment Corporation. Setelah mendapatkan gelar Master dalam ilmu komputer dari Stanford University, ia menjadi Direktur Fasilitas Komputer untuk Departemen universitas jurusan Ilmu Komputer. Ia menjadi kontributor kunci untuk teknologi jaringan yang muncul didorong oleh Departemen Pertahanan Amerika Serikat (ARPAnet), yang merupakan awal dari Internet hari ini.

Kemajuan teknologi Bosack yang terbaru termasuk ciptaannya sistem baru in-line amplifikasi serat optik yang mampu mencapai transmisi data kecepatan latency belum pernah terjadi sebelumnya dari 6,071 milidetik (serat ditambah peralatan latency) lebih 1.231 kilometer dari serat, yang kira-kira jarak antara Chicago dan New York City.

Bosack terinspirasi oleh keyakinannya bahwa dengan memanfaatkan ilmu fisika, komponen serat optik dengan kecepatan transmisi data dapat ditingkatkan dengan perangkat yang menggunakan daya yang lebih kecil, sedikit ruang dan membutuhkan pendingin.

### **Hal-hal yang tidak dianjurkan profesi sebagai Network Engineering**

- Umum
  1. Menghindari konflik interest nyata atau yang diperkirakan sedapat mungkin, dan membukakannya pada para pihak yang terpengaruh ketika muncul.
  2. Menolak sogokan dalam segala bentuknya.
  3. Berupaya menghindari kecelakaan pada orang lain, milik, reputasi, atau pekerjaan dengan tindakan salah atau maksud jahat.
- Khusus
  1. Tidak boleh mengubah konfigurasi di dalam jaringan tanpa sepengetahuan perusahaan.
  2. Tidak boleh membiarkan data-data perusahaan disabotase.

### **Hal-hal yang dianjurkan profesi sebagai Network Engineering**

- Umum

1. Mampu menjalin hubungan kerja yang harmonis dengan rekan kerja yang lain.
2. Selalu memperhatikan keselamatan kerja sesuai dengan standar yang berlaku.
3. Selalu mendayagunakan disiplin.
4. Mampu melaksanakan & menjaga kredibilitas ketika dilimpahkan wewenang oleh atasannya.
5. Berusaha untuk memperluas pengetahuan publik, dan untuk mencegah kesalahpahaman prestasi teknik.

- Khusus

1. Harus dapat mengamankan jaringan yang telah dibentuk untuk melayani user.
2. Menambahkan software dan hardware yang hanya dibutuhkan.
3. Mencatat dan melaporkan permasalahan di dalam komputer user di dalam jaringan.
4. Memiliki sikap yang disiplin dan tetap pada tugas yang sudah dibuat

## ETICS: the international software engineering service for the grid

To cite this article: A D Meglio *et al* 2008 *J. Phys.: Conf. Ser.* **119** 042010

View the [article online](#) for updates and enhancements.

### Related content

- [Multi-platform Automated Software Building and Packaging](#)  
A Abad Rodriguez, V E Gomes Gouveia, D Meneses *et al*.
- [Article I. Multi-platform Automated Software Building and Packaging](#)  
A Abad Rodriguez, V E Gomes Gouveia, D Meneses *et al*.
- [Interoperability framework for communication between processes running on different mobile operating systems](#)  
A Gal, I Filip and F Dragan

### Recent citations

- [Experiences with moving to open source standards for building and packaging](#)  
D H van Dok *et al*
- [Vaidas Giedrimas](#)
- [Testing Component Compatibility in Evolving Configurations](#)  
Ilchul Yoon *et al*



**IOP | ebooks™**

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

# ETICS: the International Software Engineering Service for the Grid

Alberto Di Meglio<sup>1</sup>, Marc-Eliañ Bégin<sup>1</sup>, Peter Couvares<sup>2</sup>, Elisabetta Ronchieri<sup>3</sup>,  
Eva Takacs<sup>4</sup>

<sup>1</sup>CERN (Switzerland), <sup>2</sup>University of Wisconsin-Madison (USA), <sup>3</sup>INFN CNAF (Italy), <sup>4</sup>4D SOFT Ltd (Hungary)

Corresponding author e-mail: alberto.di.meglio@cern.ch

**Abstract.** The ETICS system is a distributed software configuration, build and test system designed to fulfil the needs of improving the quality, reliability and interoperability of distributed software in general and grid software in particular. The ETICS project is a consortium of five partners (CERN, INFN, Engineering Ingegneria Informatica, 4D Soft and the University of Wisconsin-Madison). The ETICS service consists of a build and test job execution system based on the Metronome software and an integrated set of web services and software engineering tools to design, maintain and control build and test scenarios. The ETICS system allows taking into account complex dependencies among applications and middleware components and provides a rich environment to perform static and dynamic analysis of the software and execute deployment, system and interoperability tests. This paper gives an overview of the system architecture and functionality set and then describes how the EC-funded EGEE, DILIGENT and OMII-Europe projects are using the software engineering services to build, validate and distribute their software. Finally a number of significant use and test cases will be described to show how ETICS can be used in particular to perform interoperability tests of grid middleware using the grid itself.

## 1. Introduction

Several large-scale open-source software projects have to deal with the need to organize complex software life cycle management infrastructures and processes in order to guarantee required levels of quality, interoperability and maintainability. Often these projects have to face resource, skill, time and budget constraints that may lead to the risk of releasing software difficult to deploy, maintain, understand and integrate with other applications. Research projects with limited duration have to focus on developing software of increasing functionality through their lifetime, but cannot always guarantee that the software will still be accessible, maintainable and documented after the conclusion of their mandate. If the projects are additionally geographically and administratively distributed across several organizations, ensuring that components developed by different developers, in different languages, on different platforms and with non homogeneous tools and processes is often a daunting challenge that may lead to software difficult to manage and maintain. Under the pressure of short deadlines and large requirement sets, project managers may have to face the decision of cutting testing and quality assurance verifications, which can ultimately cause delays in the releases or diminished usability of the software due to the excessive number of undetected problems.

Even when functional tests are performed, the nature itself of complex middleware, such as that developed for the grid, the provision of adequate hardware and network resources is a cost that no project but the largest ones can easily afford. When middleware and applications are deployed on tests

or certification testbeds a lot of time is usually spent trying to make middleware suites and applications to interoperate due to the different configuration assumptions and different versions of common libraries.

In this work, we present ETICS [1], an integrated infrastructure for the automated build, configuration, integration and testing (BCIT) of software, discussing its architecture and presenting a number of examples of usage by large grid projects funded under the FP6 EC framework. ETICS aims to support research and development initiatives producing distributed and grid software by integrating existing procedures, tools and resources in a coherent infrastructure and providing an intuitive access point through a web portal and a professionally-managed, multi-platform resource facility based on Grid technologies.

ETICS doesn't replace configuration management tools like CVS [2] or SVN [3] or compilation tools like make [4] or ant [5]. Rather, ETICS provides a continuous integration and testing framework where such tools can be used. The advantage of using ETICS rather than the individual tools is that the specific syntax and usage requirements of each tool are abstracted by the ETICS data model allowing the software developers and maintainers to manage large, complex and distributed code bases with a unique, consistent set of commands and definitions. Other continuous integration and testing tools exists already (notable example are CruiseControl [6] and Maven [7]). However, most existing tools are dedicated to Java developers only and do not support execution of builds and tests 'jobs' on remote resources and local operations only are supported.

This paper is organized as follows. Section 2 describes the system architecture; Section 3 documents the data model, explaining how ETICS represents and uses information. Section 4 describes the different ETICS web applications, while Section 5 describes the command line tools and the APIs; Section 6 presents the Metronome job management system used by ETICS as execution engine; Section 7 outlines the Grid Quality Certification Model, a proposal to standardize the quality assessment of grid and distributed software projects; finally Sections 8 and 9 presents a number of real use cases and the future work foreseen.

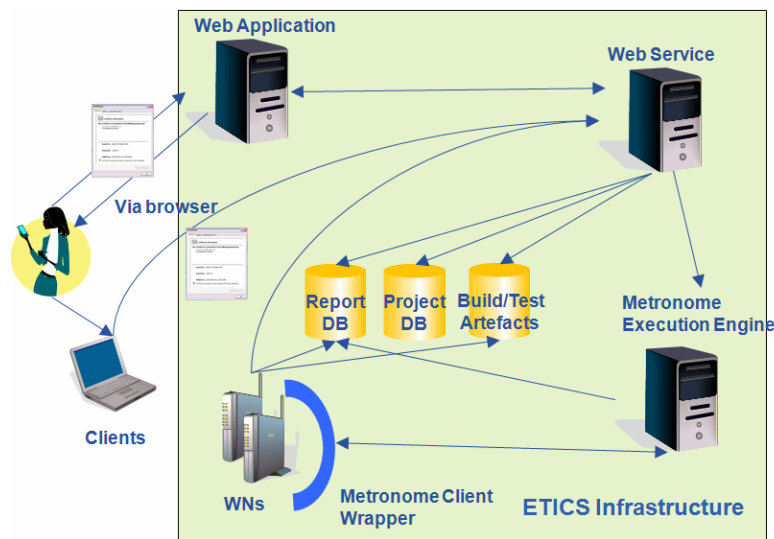
## **2. ETICS Overview and System Architecture**

The ETICS system is being developed by a project partially funded by the European Commission in the context of the FP6 program. It is composed of five partners, CERN (the project coordinator), INFN, Engineering SpA, 4D SOFT Ltd and the University of Wisconsin – Madison. The foreseen duration of the project is two years, from 1<sup>st</sup> January 2006 to 31<sup>st</sup> December 2007. An extension for two more years has been requested as part of the FP7 program.

ETICS aims to establish a distributed and managed infrastructure providing common software engineering tools and processes. Integrated pools of resources and easily accessible interfaces to the build and test tools have therefore to be provided and maintained for running automated builds and test suites. In addition all build and tests operations that can be executed on the remote worker nodes have to be reproducible locally on user computers to allow users to design and verify their testsuites and build configurations.

A centre of exchange of software configuration information and documentation is required in order to allow projects to organize and store metadata about their software and access information about other existing projects. A repository of standard tools, packages, documentation and interoperability information is required so that new and existing projects can easily and rapidly create new build configurations and testsuites.

Taking into account the above mentioned requirements, the ETICS system architecture is split into several entities, as shown on Figure 1: the ETICS Web Service; a number of web applications; a set of command line interfaces and APIs; the data model and its persistent storage implementation; and the job execution engine.



**Figure 1: The ETICS System Architecture**

### 2.1. System Features

The ETICS Build and Tests system comes with a default set of features to build software and test packages and services. The initial set of features can additionally be augmented by adding plugins [8] to the system providing specialized services. ETICS uses a plugin framework with a published interface that allows users or service providers to customize the system behavior as necessary.

The initial set of features comprises a job management engine to automate the execution of build and test jobs with support for a wide range of operating systems, CPU architectures and compilers. The internal data model allows specifying dependencies between packages or tests, which are automatically managed by the build and test engine and compiled or deployed as required. The basic set of functional plugins allows to trigger triggers coding convention checks, unit test execution, documentation generation and to collecting and publish the build and test results to the ETICS online repository.

The ETICS system supports several Version Control Systems (like CVS and SVN) and can be extended to additional systems. It has an internal packaging system that allows developers to generate automatically packages in various formats (e.g., tarballs, RPMS, debs, MSIs) and for different platforms by specifying simple sets of properties. The current version of the ETICS system (v. 2.0.3) supports several operating systems such as Scientific Linux (SLC3, SLC4, SL5), Ubuntu 7, Debian 4, various versions of Red Hat Linux, CentOS, MacOSX and Windows (on a smaller set of features) on both 32 and 64 bit (ia64 and x86\_64) CPUs.

The build engine can be instructed to build large package sets entirely from source or as a combination of source and precompiled binaries and supports strict release processes by providing volatile package repository for the development phases and permanent, non modifiable repositories for release distributions, ensuring the reproducibility of the entire build and test process over time. The compilation architecture is normally automatically detected and the appropriate binaries are downloaded from the repository. However, it is also possible to override the automatic detection specifying an alternative platform in order to perform cross-platform builds. This can be useful for example to run 32-bit builds on 64-bit (x86\_64) platforms (this requires some support from the modules to be built in order to accept the necessary compilation flags).

### 2.2. Authentication and Authorization

The underlying security infrastructure is based on digital certificates [1]. The web applications and the command line client authenticate themselves using standard x.509 [9] certificates. Users are modeled as fully qualified x.509 principal names as they appear in standard x.509-compliant certificates. The ETICS Web Service verifies the user certificate Distinguished Name in the database of existing users

involved in a project, and it allows or denies the operation based on the roles assigned to the users. From that point onwards, the Web Service uses a service certificate to interact with other internal services. The access control list on the persisted data is enforced by the Web Service. The following roles are defined in the system:

| Role                 | Description                                                                            |
|----------------------|----------------------------------------------------------------------------------------|
| Administrator        | A super user enabled to perform all the operations allowed in the ETICS infrastructure |
| Module Administrator | Responsible for handling individual projects                                           |
| Developer            | Works on the implementation of the software and can submit remote builds               |
| Integrator           | Runs remote builds and registers packages in the package repository                    |
| Tester               | Runs remote tests stores test results in the test repository                           |
| Release Manager      | Responsible for defining project level configurations and releases for a project       |
| Guest                | Read-only access to publicly accessible information                                    |

### 3. The Data Model

The data model and the storage are designed based on the principles of industry standard information models like CIM [10] and introduce formal entities to organize the software projects structure, the build configuration, the security information, the build and job result set and so on. The data model describes explicitly the objects and the relationships between objects (as in the CIM model). In addition, the model allows representing the results of running a build and test job in a way that can be consumed by the web application to generate reports.

A software project structure is represented using three main objects: the Project, the Subsystem and the Component. A Project can contain Subsystems or Components and a Subsystem can contain Components. Although the model also allows nested Subsystem, this is not currently implemented. The three structural objects (generically referred as Modules) represent respectively the abstract identity of a project, a subsystem or a test node and a physical package to be built or a test to be executed, but no version information. The Configuration object is instead used to represent the version information of a Module. Every module can have more than one Configuration. A build or test list is composed of the project structure and an associated list of configurations. Not all structural modules in a build or test list needs having an associated configuration, which is equivalent to building or testing subsets of the full structure.

Every Module can be associated to sets of Version Control System Commands, Build Commands, Test Commands, Properties, Environment Variables and Dependencies. The sets are associated to the modules via an intermediate Platform object, which allows specifying separate sets per platform. The Platform object normally represents a particular combination of Operating System, CPU architecture and compilers or interpreters.

Dependencies are used to specify that a configuration has to be built or tested before another one. Dependencies can be static if they constrain the relationship between two named Configurations or dynamic if they specify a relationship between a Configuration and a Module. In the latter case, the Configuration of the Module to be used during a build or test run is resolved at run-time using project-wide information. This method is used to control the dependencies at the project level and enforce consistency across a project.

Following the typical CIM format, the relationships between the various objects are described by relationship objects. For example to associate a set of versions and dependencies to a module ETICS uses:

- One 'component' object containing generic properties of the module (name, license, repository root URL, etc) and a Globally Unique ID (GUID)
- One or more 'configuration' objects containing the version information, version control tags, path on the repository, etc) and a Globally Unique ID (GUID)

- One 'componentConfiguration' object for each 'configuration' object associating the GUID of the module with each GUID of the configurations
- One 'vcsCommand', one 'buildCommand' and one 'testCommand' object per configuration and per platform containing the checkout commands, the build commands (e.g. init, configure, compile, package, publish, etc) and the test commands (e.g. init, configure, test, publish, etc) and the GUIDs
- One or more relationship object to associate each configuration with its commands
- If required, one or more 'property', 'environment' and 'dependency' objects per configuration and per platform (each one with its own GUID)
- One or more relationship object to associate each configuration with its properties, environment variables and dependencies

In addition, the dependencies can be declared 'static' or 'dynamic'. A static dependency is a relationship between two named configurations (e.g. 'webservice v. 1.0.0' depends on 'tomcat v. 5.0.24'). A dynamic dependency is a relationship between a named configuration and a module (e.g. 'webservice v. 1.0.0' depends on 'tomcat'). In the case of dynamic dependencies, the exact configuration to be used during the build is resolved by ETICS during the execution of the build using global project information (e.g. a project mandates that 'tomcat v. 5.0.24' must be used for all components depending on 'tomcat') or using version constraints (e.g. 'tomcat >= 5.0' is specified and the highest available version in the ETICS database satisfying the constraint is used).

The data storage back-end holds the persisted data model in the form of a relational DB implementation based on MySQL [11]. It supports different deployment models. For example, the Web Service, the Web Applications and the DB can be hosted on the same node, but as the number of requests increases with more users using the service, the database can be hosted on a separate node.

## 4. The Web Service and the Web Applications

### 4.1. The ETICS Web Service

The Web Service is the entity providing business logic for the entire system and it is used by the command line client and by all web applications. An important goal of the web service is to abstract the data storage backend, which holds the persisted version of the ETICS data model. For simplicity and better scalability, the web service is stateless. This means that it does not use a stateful web service paradigm, such as Web Services Resource Framework [12], which still has to prove itself in high-availability applications. The ETICS Web Service is developed in Java and runs as a service in a Tomcat [13] container. It provides a set of high level methods to read and write objects from and into the database and a set of specialized methods to generate object factories, manipulate objects or get information about the operations performed by users and the system for auditing and logging purposes. Additionally, the Web Service abstracts the access to the underlying job execution engine presenting a common build and test job interface layer for remote submission.

### 4.2. The Build and Test Web Application

The Build and Test Web Application<sup>1</sup> is responsible for providing a presentation layer to allow the user to browse, create and modify Modules and all associated objects. In addition it provides data forms to submit remote build and test jobs to be executed in the ETICS resource pools. It is a stateful application in order to maintain the security credentials and session information. It is developed in Java and runs as a service in a Tomcat container. The current version uses an open source, but proprietary framework called jduck [14] (developed by one of the ETICS partners, Engineering SpA), but a new version using the more standard Google Web Toolkit [15] is under preparation.

### 4.3. The ETICS Repository

---

<sup>1</sup> <https://etics.cern.ch/eticsPortal>

During the build and test process, both locally on user computers and remotely on the ETICS build nodes, the ETICS client assembles a report including information generated during the build or test such as the commands used to execute the job, high-level information about the module and configuration being built or tested, the execution logs generated during the process and customized reports generated by the metrics collectors or by tools activated during the execution (such as static metrics generators, unit tests engines, compliance checkers, etc). The build and test reports generated during builds and tests performed on the ETICS resource pools are stored in the centralized ETICS Repository and are accessible via the Repository web application<sup>2</sup>. Also in this case the application is developed as a Java web service to be run in a Tomcat.

The ETICS software repository is also the standard location where all the software artifacts are stored (e.g. packages, build and test reports). The repository is composed of two main parts: the volatile storage area and a permanent storage area.

The Volatile storage area contains temporary user-defined storage directories that users can allocate on the repository for their private use. Packages and documents in the Volatile storage areas are always overwritten with the most recent files produced at the end of a build or test run. Files in the volatile areas are deleted after 40 days to make space for new ones.

The permanent (i.e. Registered) storage area is the official location where to store officially released and registered packages and documents. A package is registered when it can be made public (e.g. final version of a configuration). Files in the permanent area are never deleted and never overwritten with newer versions, therefore only the first ever produced instance of a specific package or document is stored.

The ETICS repository is accessible using a Java web application that gives access to both the Volatile and the Registered storage areas of the ETICS repository. The current implementation provides read-only access to the repository and allows downloading packages and documents. A full-featured web application based on digital library concepts is under development and will provide secure read and write access and a set APIs to implement high-level repository solutions and to federate multiple repositories together.

#### *4.4. The ETICS User Administration Application*

Users and roles in ETICS can be managed using the ETICS Administration application<sup>3</sup>, a Java web application for Tomcat. The application can be used to manage user certificate registration and to assign to each user (that is each certificate) one or more access roles. Roles are assigned per project, per module or per configuration and allow managing access to the ETICS system and operation with a very fine grained control.

### **5. The CLIs and APIs**

The Command Line Interfaces (the ETICS Client) provide a similar functionality as the web application and makes use of the same web service interface for simplicity and symmetry. The client is developed in Python and can be used directly by the user on local resources (for example a developer machine) to execute the same build and test operations that are executed by the system in the internal, controlled build nodes. Reproducibility of the results is guaranteed by executing each build or test in independent, isolated workspace where the environment is fully managed by ETICS Client using the information stored in the central database and package repositories.

The client also provides functions to create and manage build and test configurations in the local workspace without using the information in the central database [16]. This functionality allows developers to model their software and try it in different conditions before committing the information in the central database.

### **6. The Metronome Execution Engine**

<sup>2</sup> <https://etics.cern.ch/eticsPortal/#repository>

<sup>3</sup> <https://etics.cern.ch/eticsPortal/#legAdmin> (requires access privileges)

The Metronome Job Execution Engine [17] developed by the University of Wisconsin-Madison allows the ETICS system to fully automate the submission of builds and tests on distributed resource pools. Build and test jobs can be executed on a regular schedule, on a large set of different combinations of operating systems, compilers and CPU architectures. The Metronome engine relies on Condor, a specialized workload management system for computing intensive jobs.

Metronome is in practice used to schedule the deployment of an ETICS CLI client and a set of build or test instructions for the ETICS client to execute. The actual execution of the build or test on the grid worker node is performed by the ETICS client. All information about configurations and dependencies are therefore taken at run-time from the central ETICS database allowing the execution of distributed builds and tests/

The resource pools are currently managed by CERN, INFN and the University of Wisconsin-Madison and collectively provide for public use more than 120 physical build nodes and more than 30 distinct combinations of operating systems, compilers and CPU architectures. Individual projects using ETICS can additionally use dedicated private resources where the build and test jobs can be executed under strict control.

### *6.1. The Co-Scheduling Engine*

The ETICS/Metronome system implements a mechanism to allocate multiple nodes to a test job so that different components of the test (services, clients, test suites, etc) can be executed on separate nodes [8]. This functionality, known as C-Scheduling, allows deploying complex multi-node tests and simulate the interaction of clients, services and applications in realistic situations.

A complex test suite can be modeled in ETICS as a set of nodes and each node can be assigned one more components from the ETICS repository. The components can be middleware services or applications, clients, test scripts and so on. Helper deployment modules are also available to deploy automatically standard services like Tomcat or MySQL.

When a co-scheduled test is submitted to the ETICS system for execution, the test structure as modeled in the ETICS database is analyzed and a job submission file is automatically generated and used to instruct Metronome to pre-allocate the required number of nodes. Once all nodes have been allocated, the test components are deployed on each node using the standard ETICS Client. The different services are started in random order and are then synchronized by using the internal information system. The ETICS client provides a set of APIs to advertise and retrieve properties among the deployed services. The properties can be used inside the deployment scripts or the test scripts as semaphores to synchronize the execution of each service. For example, Service A can initialize and then advertise its IP address, while Service B depending on the Service A starts its initialization and the waits (with a configurable timeout) for the IP address to be advertise before proceeding.

The different test components signal to the system their status and in particular when they have finished executing and are ready for clean up. The system keeps track of the status of each allocated node and once all components have finished executing, the results are sent back to the Metronome submission server and the nodes are cleaned restoring them to the initial conditions for accepting new jobs.

## **7. Examples of usage of ETICS**

All the entities described in the data model can be edited by using the CLIs and the Web Application. In particular the CLIs allow users to script the entire build or test process using simple sets of commands. As an example we consider the use case of a developer who needs to create a new configuration (a version) of a module as part of a larger project, register the new configuration in the ETICS database, build it, edit the configuration to modify information like dependency types or versions and rebuild it. The following set of commands can be used to perform the describe operations:

|                                                                                                        |                                                                                                                       |
|--------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| Set the project to work with and get the project structure                                             | <i>etics-get-project &lt;project&gt;</i>                                                                              |
| Checkout an existing configuration of a module                                                         | <i>etics-checkout -c &lt;configuration&gt; &lt;module&gt;</i>                                                         |
| Build the configuration                                                                                | <i>etics-build &lt;module&gt;</i>                                                                                     |
| Use the existing configuration to create a new configuration                                           | <i>etics-configuration prepare -o configuration.ini &lt;module&gt;</i>                                                |
| Edit the configuration file changing version number, cvs tag, dependencies, etc as needed              | <i>Use any text editor</i>                                                                                            |
| Register the new configuration                                                                         | <i>etics-configuration modify -i configuration.ini</i>                                                                |
| Checkout the new configuration to get any new dependency (use -merge to preserve existing information) | <i>etics-checkout -merge -c &lt;configuration&gt; &lt;module&gt;</i>                                                  |
| Rebuild                                                                                                | <i>etics-build -c &lt;newconfiguration&gt; &lt;module&gt;</i>                                                         |
| Submit a remote build on two different platforms and register the binaries in the ETICS repository     | <i>etics-submit build -register -platforms &lt;platform1,platform2&gt; -c &lt;newconfiguration&gt; &lt;module&gt;</i> |

The configuration.ini file uses a specific format based on the standard INI format to describe all module and configuration properties, the dependencies, the checkout, build and test commands and so on.

## 8. The Grid Quality Certification Model

One of the major goals of the ETICS project is to define a Quality Assurance certification model suitable for assessing and monitoring software quality for grid projects. During its first 18 months of operations, a model has been defined in collaboration with the grid projects using ETICS and proposed for feedback at a number of international events (recently at the EGEE 07 conference in Budapest). The model, which is currently called Grid Quality Certification Model (GQCM) for its main application area, is however suitable for assessing in general the quality of distributed software development projects [18].

The GQCM model position itself between standard QA guidelines like CMMi [19] or ISO/IEC 25000:2005<sup>4</sup> and individual software metrics definitions. The standard models normally define qualitative guidelines for an organization, from where the expected quality of the software product can be inferred, while individual metrics provide quantitative ways of assessing specific, but normally disconnected properties, which have to be considered in specific context to be given meaningful interpretations.

We believe that there is a currently a gap between the standards and the commonly used software metrics, which Grid Quality Certification proposes to fill by defining a practical implementation of the QA guidelines using specific set of metrics and assessment perspectives. In this respect, GQCM is closer to and compatible with ISO 9126<sup>5</sup>, but provides the “how”, not only the “what”. GQCM specifies in particular predefined and coherent sets of metrics and thresholds and ways of interpreting and consolidating them, whereas ISO 9126 leaves to the software development organization to define them.

GQCM and its implementation in the ETICS system provide therefore homogeneous, not subjective and comparable results and a repeatable workflow, extremely valuable especially in contexts where the QA culture or expertise is not strong or too costly to acquire.

<sup>4</sup> ‘Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE’, [http://webstore.iec.ch/preview/info\\_isoiec25000%7Bed1.0%7Den.pdf](http://webstore.iec.ch/preview/info_isoiec25000%7Bed1.0%7Den.pdf)

<sup>5</sup> ISO (1991). *International Standard ISO/IEC 9126. Information technology -- Software product evaluation -- Quality characteristics and guidelines for their use*, International Organization for Standardization, International Electrotechnical Commission, Geneva.

GQCM is structured in families or Evaluation Modules (EM). Each EM groups a set of QA evaluation techniques in a specific QA area: Coding style, Static analysis, Structural testing, Functional testing and Standards compliance. Each EM is associated to a one or more metrics within the QA area, which QA systems implementing the model must collect using the tools of their choice. Additionally, the metrics results can be grouped in the three points of view or 'perspectives', Quality, Platforms and Standards.

The perspectives are independent, contrary to other waterfall models and provide a partial score that contributes to produce the overall QA score of the software under assessment. Individual projects are also free to focus the Quality perspective alone or in combination with another perspective in case for example multi-platform support or compliance to standards is not applicable or desired. The overall score allows monitoring the progress of the software over time, taking corrective actions in deficient areas and comparing the status of different software systems.

## 9. ETICS Build and Test System Application Examples

### 9.1. EGEE/gLite

The EGEE project<sup>6</sup>, co-funded funded by the European Commission and coordinated by CERN, brings together experts from over 32 countries with the common aim of building on recent advances in Grid technology and developing a service Grid infrastructure which is available to scientists 24 hours-a-day. The project provides researchers in academia and industry with access to a production level Grid infrastructure, independent of their geographic location.

The EGEE project also produces a grid middleware distribution called gLite, which is born from the collaborative efforts of more than 80 people in 12 different academic and industrial research centers. gLite provides a bleeding-edge, best-of-breed framework for building grid applications.

The gLite development process has been standardized on the ETICS system for building the software and testing the deployment of the service metapackages (such as the User Interface, the Worker Node, the WMS Server and the CREAM Computing Element Server). The entire gLite distribution is composed of about 580 distinct packages and almost 3000 configurations for a total of more than 1.6 million lines of code in C/C++, Java, Python and other languages.

| gLite       |     |                |      |              |       |
|-------------|-----|----------------|------|--------------|-------|
| Modules     | 579 | Configurations | 2932 | Dependencies | 11179 |
| - Project   | 1   | - Project      | 8    | - Internals  | 6414  |
| - Subsystem | 39  | - Subsystem    | 320  | - Externals  | 4764  |
| - Component | 539 | - Component    | 2604 | - Others     | 1     |

Before moving to the ETICS system, the gLite distribution was only developed on a single platform (CERN Scientific Linux 3 32bit), while it is now automatically built on many different platforms several times per day (CERN Scientific Linux 3 32bit, CERN Scientific Linux 4 32bit and 64bit, Scientific Linux 5, Red Hat Enterprise Server 4, Debian 4, Ubuntu 7, CentOS 4). Some experimental builds on subsets of the gLite components have been performed on MacOSX (Leopard). More recently a new project called Grid4Win (coordinated by INFN in Italy and sponsored by Microsoft) has adopted ETICS as build system in the effort of porting some of the gLite components to Windows using the Unix Services for Windows on Windows Server 2008.

### 9.2. DILIGENT

The main objective of DILIGENT is to create an advanced testbed that will allow members of dynamic virtual e-Science organizations to access shared knowledge and to collaborate in a secure, coordinated, dynamic and cost-effective way [20]. This testbed has integrated Grid and Digital Library (DL) technologies, where regression test activity is performed. It has been validated by two

<sup>6</sup> EGEE Middleware Architecture, August 2004, <https://edms.cern.ch/file7476451/1.0/Architecture.pdf>

complementary real-life application scenarios: the former comes from the culture heritage domain, the latter from the environmental e-Science domain.

DILIGENT is a Java based system that uses the Web Service Core framework of the Globus Toolkit comprised of approximately 90 components, which are Web Services, libraries, port-lets, and wrappers of gLite components, developed by several teams across Europe.

| Diligent    |     |                |      |              |      |
|-------------|-----|----------------|------|--------------|------|
| Modules     | 581 | Configurations | 1688 | Dependencies | 5326 |
| - Project   | 1   | - Project      | 14   | - Internals  | 5024 |
| - Subsystem | 27  | - Subsystem    | 103  | - Externals  | 302  |
| - Component | 553 | - Component    | 1571 | - Others     | 0    |

The DILIGENT project has started using ETICS for building and testing remotely through Metronome. The current testing environment of the DILIGENT system consists of six worker nodes (i.e., virtual machines). ETICS local and remote testing facilities assist the DILIGENT testing process by executing remote regression tests and applying the static analysis and metric calculation utilities - being included as 'plugins' - in the Test and Metrics Framework of ETICS in an automated and straightforward way.

### 9.3. OMII-Europe

The OMII-Europe project<sup>7</sup> aims to bring together the best technologies from Europe and elsewhere and make them available in a usable and supported form to scientists. It delivers production quality middleware relevant to large-scale and smaller collaborative Grids by federating development activities across Europe. It also maximizes benefits of global collaborations with USA and China. Specific actions performed by this project covers for example the assessment of the QA process, the re-engineering of the adaptations to standards, the setting up of a central repository for software, documentation and training.

OMII-Europe uses ETICS to build and test its re-engineered middleware components on different platforms, the same supported by the original providers, and to perform deployment tests of its services. In addition, this project has to check interoperability between job submission engines by validating their conformance to the OGSA-BES recommendation. In order to do so, test suites have been developed in order to interpret and parse the recommendation into specific service calls, together with independent clients to perform tests. ETICS has therefore been adopted to perform this test automatically as part of the QA practices. OMII-Europe has a dedicated ETICS installation at the University of Southampton and also uses the public distributed testbed.

### 9.4. EUChinaGrid/EGEE/ETICS IPv6 Collaboration

EUChina<sup>8</sup>, EGEE and ETICS are collaborating on the testing of IPv6 compliance of a number of component in the EGEE/gLite distribution. Each partner in this collaboration focuses on a specific task: EUChina has defined IPv6 code compliance tools and metrics to perform static compliance analysis; EGEE has worked on the porting of some gLite components to IPv6 in both pure IPv6 and dual stack configurations; finally ETICS has contributed providing the tools and resources to automated the IPv6 tests across three different sites (CERN in Switzerland, GARR in Italy and IN2P3 in Paris) and to produce full compliance reports of all software registered in the system [8].

## 10. Conclusions and Future Work

<sup>7</sup> OMII-Europe, <http://omii-europe.org/OMII-Europe>

<sup>8</sup> EUChina Grid, <http://roc.euchinagrid.org/home.php>

In this paper, we presented ETICS, a system that provides services and tools for automatic building, configuration and testing of distributed and grid software. Then, we provided an overview of the ETICS architecture split into several entities like the ETICS Web Service going in details for each of them. We also detailed one of the major goals of the ETICS project that consists of defining a Quality Assurance certification model for grid and distributed software development projects. Finally, we described examples of ETICS build and test applications, such as EGEE, DILIGENT, OMII-Europe and EUChinaGrid, also involved during the system development.

We believe that the ETICS system has potentialities in solving many of the problems faced by open-source software projects during their life cycle management processes, in particular guaranteeing high levels of quality, interoperability and maintainability. Therefore, we consider this project only the beginning of a large one. We intend to add extra plugins to the ETICS system in order not only to evaluate quality and interoperability metrics but also to perform complex tests like the installation of services. We also plan to improve the current ETICS performances considering users' feedback and to add further features accordingly to the needs of the ETICS community.

## References

- [1] Bégin M-E, Diez-Andino Sancho G, Di Meglio A, Ferro E, Ronchieri E, Selmi M and Zurek M 2007 *Springer Verlag Lecture Notes in Computer Science (LNCS) Series, LNCS 4401* pp 81-97
- [2] Purdy G N 2000 *CVS Pocket Reference* (O'Reilly)
- [3] Mason M 2005 *Pragmatic Version Control Using Subversion [Illustrated] (Paperback)* (Pragmatic Bookshelf)
- [4] Talbott S 1991 *Managing Projects with Make* (2nd Ed. O'Reilly & Associates, Inc.)
- [5] Hatcher E and Loughran S 2002 *Java Development with Ant [Illustrated] (Paperback)* (Manning Publications)
- [6] Trueman T 2004 *Cruise Control (Hardcover)* (HarperTeen)
- [7] Massol V and O'Brien T 2005 *Maven: A Developer's Notebook (Developer's Notebooks) [ILLUSTRATED] (Paperback)* (O'Reilly Media, Inc.)
- [8] Bégin M-E, Couvares P, Diez-Andino Sancho G, Da Ronco S, Di Meglio A, Dini L, Fabriani P, Gietz B, Pavlos A, Ronchieri E, Selmi M, Takacs E and Zurek M 2007 *Tenth World Conference on Integrated Design & Process Technology* (Antalya, Turkey)
- [9] Gutmann P 2000 *X.509 Style Guide* (<http://www.cs.auckland.ac.nz/~pgut001/pubs/x509guide.txt>)
- [10] McMorran A, Ault G, Morgan C, Elders I and McDonald J 2006 *Power Engineering Society General Meeting 2006* (IEEE)
- [11] DuBois P 2005 *MySQL (Third Edition)* (Sams Developer's Library)
- [12] Foster I, Frey J, Graham S, Tuecke S, Czajkowski K, Ferguson D, Leymann F, Nally M, Storey T, and Weerawaranna S 2004 *Globus Alliance*
- [13] Brittain J and Darwin I F 2003 *Tomcat: The Definitive Guide [ILLUSTRATED] (Paperback)* (O'Reilly Media, Inc.)
- [14] Godfrey M and Grossman D 1999 *Thirtieth SIGCSE technical symposium on Computer science education*
- [15] Chaganti P 2007 *Google Web Toolkit: GWT Java Ajax Programming (Paperback)* (Packt Publishing)
- [16] Bégin M-E, Da Ronco S, Diez-Andino Sancho G, Gentilini M, Ronchieri E and Selmi M 2007 *International Conference on Computing in High Energy and Nuclear Physics(CHEP)*
- [17] Pavlo A, Couvares P, Gietzel R, Karp A, Alderman I D, Livny M and Bacon C 2006 *LISA06: Twentieth Systems Administration Conference* (Washington DC, USA) pp 263-273
- [18] Rippa A, Bégin M-E, Di Meglio A and Manieri A 2007 *Third Conference of the EELA Project*
- [19] Chrissis M B, Konrad M and Shrum S 2003 *CMMI – Guidelines for Process Integration and Product Improvement (Hardcover)* (Addison-Wesley Professional)
- [20] Castelli D, Candela L, Pagano P and Simi M 2005 *2<sup>nd</sup> IEEE – CS International Symposium Global Data Interoperability* (IEEE Computer Society) pp 56-99

## ETIKA & TANGGUNGJAWAB PROFESIONAL DALAM REKAYASA PERANGKAT LUNAK

### *Abstrak*

*Secara umum para profesional yang bergerak dalam bidang apapun memiliki kode etik dan tanggungjawab dalam menjalankan aktivitas atau pekerjaannya masing-masing. Setiap bidang tentu saja memiliki keunikan sesuai dengan karakteristik pekerjaannya. Hal ini juga berlaku pula dalam dunia teknologi informasi khususnya dalam dunia komputasi, rekayasa perangkat lunak maupun sistem informasi. Seorang Profesional komputasi pada prinsipnya memiliki kewajiban etis bagi klien, atasan, profesional lainnya, dan masyarakat dalam memenuhi tanggung jawab profesiona. Kewajiban-kewajiban tersebut dituangkan dalam kode etik, yang dapat digunakan untuk membuat keputusan tentang masalah etika dalam dunia komputasi atau teknologi informasi.*

**Keyword : Kode etik, Tanggung Jawab Profesional**

### I. PENDAHULUAN

Dalam periode global saat ini ekonomi, ilmu pengetahuan, teknologi dan informasi memainkan peran yang sangat penting bagi umat manusia. Kemajuan berbagai teknologi baru yang belum pernah terjadi sebelumnya memberikan pengaruh yang tidak kecil pada pekerjaan dan kehidupan sehari-hari. Dalam berbagai proses yang terjadi dalam era globalisasi saat ini banyak yang menjadi tidak terkendali atau dapat di sebut sebagai periode turbulensi atau periode ketidakpastian. Dalam lingkungan yang penuh dengan kompetisi, pengembangan ilmu pengetahuan dan teknologi untuk kepentingan bisnis juga tak luput dari kondisi ketidakpercayaan para individu yang terlibat didalamnya.

Yang terjadi saat ini dapat digambarkan bagaimana tuntutan perusahaan memiliki pekerja yang profesional dan berkualifikasi tinggi, bagaimana seorang manajer atau teknisi sepatutnya berperilaku ketika dihadapkan dalam situasi yang sulit? Bagaimana seorang karyawan yang senantiasa dituntut untuk dapat mengembangkan produk baru, teknologi, dan bekerja dengan penuh inisiatif dalam kaitannya membuka pasar baru? Bagaimana pula keputusan yang diambil mampu mempengaruhi lingkungan, kinerja dan konsumen. Secara normatif tentu saja, karyawan atau pelaku bisnis bertindak dalam batas-batas hubungan kerja dan berperilaku sesuai dengan gambaran kerja unit dalam organisasi atau bahkan instruksi dari atasan sebagai personal yang memegang tanggung jawab terakhir. Namun dengan kondisi seperti itu, tidak berarti bahwa pegawai atau karyawan dapat terbebas dari tanggung jawab individu atau kolektif. Prinsip *Responsibility*, dimana setiap individu diharapkan untuk bertindak secara bertanggung jawab terhadap kepentingan manusia. Hal ini tentu saja menimbulkan pertanyaan mengenai perilaku sosial, etika serta tanggung jawab manusia yang terlibat khususnya dalam pengembangan teknologi informasi. Pertanyaan terbesar adalah bagaimana kaitannya etika dan profesionalisme dalam pengembangan teknologi Informasi khususnya dalam bidang rekayasa perangkat lunak (*software engineering*).

Seorang *software engineer* harus memiliki perilaku etis dan moral secara bertanggung jawab jika ingin dihargai sebagai profesional. Dengan demikian, sebagai profesional harus menjunjung tinggi standar normal kejujuran dan integritas. Namun, terdapat batasan dimana standar perilaku seorang *software engineer* yang dapat diterima diluar aturan hukum yang berlaku. Beberapa hal tersebut menyangkut

1. Kerahasiaan, seorang engineer harus menghormati kerahasiaan atasan maupun klien
2. Seorang engineer menerima pekerjaan sesuai kompetensinya.

3. Seorang engineer harus menyadari hukum setempat yang mengatur penggunaan kekayaan intelektual seperti paten, hak cipta, dll dalam hal ini harus berhati-hati memastikan bahwa properti intelektual pengusaha dan klien dilindungi.
4. Seorang engineer Perangkat Lunak sebaiknya tidak menggunakan kemampuannya untuk menyalahgunakan komputer orang lain.

## **II. TINJAUAN PUSTAKA**

### **2.1. Profesional**

Profesional berasal dari kata profesi yaitu suatu yang melekat pada seseorang yang memberikan jasa atau keterampilan yang dimilikinya bagi orang lain, maka profesional merupakan sebutan pelakunya. Dalam hal kaitannya dengan Profesional komputasi termasuk didalamnya *desainer hardware, software engineer, database administrator*, analis sistem, dan ilmuwan komputer. Karena profesional memahami bahwa pengetahuannya selalu mengalami kemajuan, profesional juga harus mampu melanjutkan pendidikannya dengan membaca berbagai publikasi. Prinsip yang paling penting menjadi seorang profesional adalah wajib untuk menyimpan informasi spesifik tentang rahasia klien. Mengapa demikian? Karena Profesional pada umumnya cenderung memiliki klien, bukan pelanggan. Hal ini dapat di analogikan seperti penjual yang harus selalu mencoba untuk mampu memuaskan keinginan setiap pelanggannya, maka profesional harus mampu untuk memenuhi kebutuhan kliennya.

### **2.2. Responsibility dan Profesional Responsibility**

Responsibility dalam hal ini adalah tanggungjawab. Tanggungjawab menurut Kamus Besar Bahasa Indonesia (2008) adalah keadaan wajib menanggung segala sesuatunya. Tanggungjawab adalah kesadaran manusia akan tingkah lakunya yang disengaja maupun tidak disengaja. Apabila dikaitkan dengan ranah profesional menurut Wikipedia (2011) dalam hal ini sebut saja tanggungjawab profesional merupakan tindakan secara profesional, mematuhi hukum, menghindari konflik kepentingan, dan menempatkan kepentingan klien di atas kepentingan sendiri.

### **2.3. Kode Etik**

Kode etik adalah sistem norma, nilai dan aturan profesional tertulis yang secara tegas menyatakan apa yang benar dan baik dan apa yang tidak benar dan tidak baik bagi profesional. Kode etik menyatakan perbuatan apa yang benar atau salah, perbuatan apa yang harus dilakukan dan apa yang harus dihindari. Tujuan kode etik agar profesional memberikan jasa sebaik-baiknya kepada pemakai atau kliennya. Adanya kode etik akan melindungi perbuatan yang tidak profesional. Ketaatan tenaga profesional terhadap kode etik merupakan ketaatan naluriyah yang telah bersatu dengan pikiran, jiwa dan perilaku tenaga profesional. Jadi ketaatan itu terbentuk dari masing-masing orang bukan karena paksaan. Dengan demikian tenaga profesional merasa bila dia melanggar kode etiknya sendiri maka profesinya akan rusak dan yang rugi adalah dia sendiri. Kode etik disusun oleh organisasi profesi sehingga masing-masing profesi memiliki kode etik tersendiri.

### **2.4. Rekayasa Perangkat Lunak**

#### **2.4.1. Rekayasa**

Istilah rekayasa sering ditemui dalam kehidupan sehari-hari khususnya dalam bidang teknik, ada istilah rekayasa perangkat lunak (*software engineering*), rekayasa sistem (*system engineering*) sampai pada istilah rekayasa jembatan. Dalam banyak kajian istilah rekayasa mengacu kepada aktivitas analisis, desain, konstruksi, dan proses manajemen suatu entitas

(objek kajian tertentu). Dalam kamus bahasa Indonesia, istilah rekayasa justru telah menjadi kerdil sebab diartikan sebagian sebagai ‘usaha atau aktivitas menipu’. Namun dalam konteks ini tidak dapat menyalahkan kamus besar bahasa Indonesia, karena suatu istilah dalam bahasa suatu bangsa mengacu kepada fungsi istilah tersebut digunakan. Dalam aspek global, rekayasa dalam hal teknologi telah ‘mengubah dunia’ dengan begitu dahsyat. Berbagai ‘perubahan’ yang terjadi saat ini menurut sebagian besar ahli ditentukan oleh teknologi dan istilah teknologi sangat erat berkaitan dengan istilah rekayasa.

#### **2.4.2. Sejarah & Esensi Rekayasa**

Telaah implikasi rekayasa bagi umum baru dimulai pada tahun 1970-an. Pada awal penggunaan istilah rekayasa, masyarakat menganggap bahwa rekayasawan sebagai alat produksi saja, bukan sebagai seorang pengambil keputusan yang bertanggung jawab. Namun seiring dengan perkembangan zaman, saat ini sebagian masyarakat telah memahami dan menyadari bahwa proses dan produk ke rekayasa (teknologi) merupakan hasil dari kreativitas personal. Mengapa ditulis sebagai sebagian masyarakat? Karena memang masih sebagian saja masyarakat yang menyadari bahwa dalam rekayasa nilai moral, perilaku dan kemampuan sang rekayasawan akan sangat mempengaruhi nilai kreasinya; semakin baik nilai moral seorang rekayasawan, biasanya semakin tinggi nilai keselamatan penggunaan hasil rekayasanya. *Engineer* adalah seorang profesional yang bertanggung jawab untuk memproduksi produk yang cocok untuk digunakan oleh klien. Untuk memastikan bahwa suatu produk layak digunakan, membutuhkan pemahaman tentang lingkungan di mana produk itu digunakan.

#### **2.6. Rekayasa Perangkat Lunak**

Seiring dengan perkembangan teknologi informasi yang sedemikian pesatnya. Perkembangan komputer yang di awalnya ditujukan sebagai alat bantu hitung manusia telah memiliki kemampuan ribuan kali atau mungkin saja telah jutaan kali lipat sejak awal kemunculannya. Perkembangan Perangkat keras (*hardware*) yang awalnya sangat menentukan dalam perkembangan teknologi informasi saat ini bahkan telah mampu dilampaui oleh perangkat lunak (*software*) dalam memberikan pengaruhnya dalam era industri. Pada masa awal perhitungannya, sistem berbasis komputer dikembangkan dengan menggunakan manajemen yang berorientasi pada perangkat keras. Para manajer proyek memfokuskan diri kepada perangkat keras karena pengembangan sistem membutuhkan biaya yang sangat besar dan menyita waktu. Untuk mengembangkan perangkat lunak pada waktu itu para pemrogram hanya sekedar coba-coba dan sangat berhati-hati. Sekarang walaupun distribusi biaya untuk pengembangan sistem berbasis komputer telah berubah sangat drastis. Dan telah dapat diprediksi bahwa perangkat lunak merupakan salah satu bentuk tunggal yang memiliki biaya sangat tinggi namun dirasakan oleh berbagai enterprise khususnya perusahaan merupakan investasi yang sangat berpengaruh terhadap maju atau mundur bahkan matinya sebuah perusahaan. Untuk memahami sejauh mana manusia membutuhkan perangkat lunak maka harus memahami apa dan bagaimana perangkat lunak tersebut. Sampai saat ini banyak pakar yang telah mendefinisikan perangkat lunak, namun dalam tulisan ini hanya akan diperkenalkan definisi yang sangat terkenal dari perangkat lunak. Perangkat lunak menurut Roger S. Pressman(1997) adalah

- (1) Perintah (program komputer) yang bila dieksekusi memberikan fungsi dan unjuk kerja seperti yang diinginkan
- (2) Struktur data yang memungkinkan program memanipulasi informasi secara proporsional
- (3) Dan, dokumen yang menggambarkan operasi dan kegunaan program

Berdasarkan hal tersebut dapat dijelaskan bahwa pada prinsipnya perangkat lunak merupakan kumpulan perintah yang dapat digunakan sesuai keinginan dalam hal memanipulasi informasi secara proporsional.

Menurut Roger S. Pressman (1997) perangkat lunak lebih merupakan elemen logika dan bukan merupakan elemen fisik. Dengan demikian, perangkat lunak memiliki ciri yang berbeda dari perangkat keras yaitu

- (1) Perangkat lunak dibangun dan dikembangkan, tidak dibuat dalam bentuk klasik
- (2) Perangkat lunak tidak pernah usang
- (3) Sebagian besar perangkat lunak dibuat secara custom-built, serta tidak dapat dirakit dari komponen yang sudah ada

### **2.5. International standard for profesional software development & Ethical responsibility**

Sebagaimana penjelasan sebelumnya bahwa profesional harus memiliki kode etik yang disusun oleh organisasi maka ACM(*Association for Computing Machinery*) dan IEEE(*Institute Of Electrical & Electronics Engineers*) telah bekerja sama untuk menghasilkan kode etik dan profesional bersama dalam ranah *software engineer* yang menjelaskan secara singkat substansinya dan menyatakan :

*Computers have a central and growing role in commerce, industry, government, medicine, education, entertainment and society at large. Software engineers are those who contribute by direct participation or by teaching, to the analysis, specification, design, development, certification, maintenance and testing of software systems. Because of their roles in developing software systems, software engineers have significant opportunities to do good or cause harm, to enable others to do good or cause harm, or to influence others to do good or cause harm. To ensure, as much as possible, that their efforts will be used for good, software engineers must commit themselves to making software engineering a beneficial and respected profession. In accordance with that commitment, software engineers shall adhere to the following Code of Ethics and Professional Practice.*

Secara singkat pernyataan diatas menjelaskan bahwa dengan perkembangan dunia komputer yang begitu pesat maka para software engineer secara etika harus memiliki kontribusi dengan berpartisipasi langsung atau dengan proses pengajaran, untuk proses analisis, spesifikasi, desain, pengembangan, sertifikasi, pemeliharaan dan pengujian perangkat lunak.

## **III. PEMBAHASAN**

Pada implementasinya terdapat banyak peluang bahwa rekayasa perangkat lunak dapat digunakan untuk penyalahgunaan fungsi khususnya dalam dunia kejahatan. Untuk memastikan bahwa rekayasa perangkat lunak digunakan dengan baik untuk kemaslahatan umat manusia maka ada upaya agar para software engineer harus berkomitmen untuk menjadikan profesi ini menguntungkan dan dihormati sehingga diperlukan harus mematuhi standar Kode Etik dan Profesional. *Software engineer* berkomitmen untuk membuat analisis, spesifikasi, desain, pengembangan, pengujian dan pemeliharaan perangkat lunak yang menguntungkan dan dihargai sebagai sebuah profesi. Secara umum *software engineer* harus mematuhi 8(delapan) prinsip berikut :

1. *PUBLIC – Software engineers shall act consistently with the public interest.* Pernyataan ini berarti seorang software engineer harus bertindak secara konsisten dengan kepentingan publik.
2. *CLIENT AND EMPLOYER – Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.* Seorang software engineer harus bertindak sesuai dengan kepentingan terbaik klien dan atasan yang konsisten dengan kepentingan publik.
3. *PRODUCT – Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.* Seorang Software engineer harus

memastikan bahwa produk dan modifikasi yang terkait dengan memenuhi standar profesional setinggi mungkin.

4. *JUDGMENT – Software engineers shall maintain integrity and independence in their professional judgment.* Seorang Software engineer harus mempertahankan integritas dan kemandirian dalam penilaian profesional yang *the Association for Computing Machinery, Inc. 1999*
5. *MANAGEMENT – Software engineering Gotterbarn, D and Miller, K managers and leaders shall subscribe to and and Rogerson, S "Software promote an ethical approach to the Engineering Code of Ethics," management of software development and SIGCAS Newsletter July 1997. maintenance.* Seorang pimpinan dan manajer *Presman, Roger S., (1997) Software engineering melakukan pendekatan Software Engineering: A etis kepada manajemen pengembangan Practitioner's Approach.* perangkat lunak dan pemeliharaan. *McGraw Hill Book Co.*
6. *PROFESSION – Software engineers shall Ian Sommerville, advance the integrity and reputation of the (2004).Software Engineering profession consistent with the public interest. 7th Edition", Addison-Wesley*  
Seorang Software engineer harus memajukan [http://en.wikipedia.org/wiki/Professional\\_responsibility](http://en.wikipedia.org/wiki/Professional_responsibility) dengan integritas dan reputasi profesi yang konsisten dengan kepentingan publik. *Michael C. Loui, Keith W.*
7. *COLLEAGUES – Software engineers shall be Miller, Ethics and Professional fair to and supportive of their colleagues. Responsibility in* seorang Software engineer harus bersikap adil *Computing.Encyclopedia Of* dan mendukung rekan-rekan tua kolega dalam *Computer Science* pekerjaan.
8. *SELF – Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.* Seorang Software engineer harus berpartisipasi dalam pembelajaran seumur hidup tentang praktek profesi yang dimiliki dan akan mempromosikan pendekatan etis untuk profesinya.

#### IV. KESIMPULAN

Berdasarkan apa yang telah dijelaskan di atas bahwa fungsi *engineer* tidak hanya merancang, membuat, merawat dan mengembangkan teknologi yang dikembangkan tetapi pada implementasinya tindakan yang dilakukan dalam hal ini perilaku yang dibangun pun harus mengikuti kode etik yang telah ditetapkan oleh organisasi profesional sesuai dengan bidang yang diembannya. Selain aturan hukum dan kode etik yang harus ditaati disebabkan karena fungsi profesional adalah mengutamakan klien, atasan dan ranah publik yang menyangkut profesi *engineer* yang bersangkutan.

## ⇒ **Etika dalam Pengembangan Perangkat Lunak**

Kepercayaan pada etika Software Engineering kini telah menjadi bagian penting dari budaya kontemporer. Untuk mengambil contoh misalnya, ahli radiologi untuk memverifikasi kebenaran kode yang mengendalikan mesin X-ray yang menunjuk ke arahnya kepala. Pada titik tertentu, tidak ada pilihan selain mengasumsikan bahwa perangkat lunak dibuat dan diinstal di berbagai sistem yang diterapkan tidak memiliki jaminan bahwa telah terimplementasikan dengan benar, dan dengan cara yang konsisten dengan kepentingan publik.

Seorang Software Engineer harus berkomitmen untuk membuat analisis, spesifikasi, desain, pengembangan, pengujian dan pemeliharaan perangkat lunak profesi yang bermanfaat dan dihormati. Seorang Software Engineer harus mematuhi Delapan Prinsip berikut:

1. Publik— Seorang Software Engineer harus bertindak konsisten dengan kepentingan publik.
2. Client dan Pegawai — Seorang Software Engineer harus mampu memberikan tindakan yang terbaik untuk kepentingan Client ataupun Atasan mereka secara konsisten dengan kepentingan publik.
3. Produk — Seorang Software Engineer harus memastikan bahwa produk mereka dan modifikasi terkait, memenuhi standar profesional setinggi mungkin.
4. Judgment — Seorang Software Engineer harus menjaga integritas dan kemandirian dalam penilaian profesional mereka.
5. Manajemen — Manajer dan pemimpin pengembangan perangkat lunak harus bisa membagikan dan memsialkan suatu pendekatan etis untuk manajemen sebuah pengembangan perangkat lunak dan pemeliharaannya.
6. Profesi — Seorang Software Engineer harus meningkatkan integritas dan reputasi profesi yang konsisten dengan kepentingan publik.
7. Kolega — Seorang Software Engineer harus adil dan mendukung rekan kerja mereka.
8. Diri Sendiri — Seorang Software Engineer harus mau untuk terus belajar sepanjang hayat yang berkaitan dengan praktik profesinya dan harus mempromosikan pendekatan etika terhadap praktik profesi.

## 1. Hubungan antara strategy bisnis, strategy IS dan strategy IT

Dalam mendapatkan keuntungan strategis dari Sistem Informasi / Teknologi Informasi (SI/TI), cara yang tepat adalah dengan memikirkan ulang bisnis dengan menganalisis permasalahan yang ada sekarang dan perubahan lingkungannya serta menggunakan IT sebagai penunjangnya. Strategy IS berdasarkan apa yang dibutuhkan perusahaan atau berdasarkan bisnis, berorientasi pada kebutuhan, dan focus pada pemenuhan aplikasi bisnis. Sedangkan strategy IT berdasarkan pada aktifitas, orientasi pemenuhan kebutuhan perusahaan dan berfokus pada teknologi seperti infrastruktur dan servis. Strategy IS/IT diperlukan untuk membuat Strategy bisnis yaitu membuat keputusan bisnis, tujuan dan arahan serta perubahan yang diambil perusahaan untuk menjalankan bisnisnya agar mendapatkan profit dan mengurangi resiko dalam bisnis perusahaan.

## 2. Evolusi Proses Strategi SI/TI

### Tahap 1

Departemen TI perlu untuk merencanakan antarmuka antara aplikasi dikembangkan secara terpisah, proyek demi proyek, agar dapat bekerja dengan efektif dan efisien, baik pada operasional bisnis dan di penggunaan teknologi. Mendapat pemahaman dari manajemen untuk meningkatkan dependensi bisnis terhadap sistemnya. Aplikasi pendukung dikembangkan dan manajemen mengartikan SI/TI dalam batasan tertentu, namun tingkat dependensinya perlahan meningkat

### Tahap 2

Manajemen menyadari adanya inisiasi review top-down dari aplikasi SI/TI terhadap dependensinya dengan bisnis. Penentuan prioritas disetujui berdasarkan kebutuhan bisnis yang berhubungan. Konsep Help desk diterapkan sebagai system pendukung

### Tahap 3

Fokus pada perencanaan SI/TI secara detail, untuk menetapkan cara terbaik dalam menerapkan aplikasi dan teknologi pendukung, atau untuk kasus tertentu, penerapan kembali sistem yang ada dengan lebih layak, terintegrasi dan lebih hemat biaya. Ada waktu untuk penerapan yang efektif, dan saat strategi berjalan, tidak ada kendala yang terjadi karena sumber daya TI digunakan sesuai budget.

### Tahap 4

User memegang kendali, tidak diarahkan oleh manajemen senior tetapi tidak juga diabaikan. Penggunaan SI/TI yang kreatif oleh user memperlihatkan adanya kesempatan baru, menggunakan informasi dengan cara baru untuk perluasan keunggulan kompetitif bisnis.

### Tahap 5

Merupakan tahap yang paling sulit untuk diraih. Menyelaraskan antara SI/TI dengan strategi bisnis menjadi tugas utama, dan hal ini membutuhkan perhatian dari eksekutif senior, manajemen dan spesialis TI (ketiga pihak harus bekerja bersama) Penerapan strategis diidentifikasi dan disetujui dalam konteks strategi bisnis

### 3. Pendekatan untuk pengembangan strategy SI/IT

#### Business-led

Administrasi dan Organisasi adalah pendekatan yang paling banyak diadopsi. Jika organisasi tersebut buruk dalam merumuskan strategi bisnis, itu akan mengalami kesulitan yang cukup besar dalam mengembangkan strategi SI / TI.

- Dilakukan oleh spesialis TI, berdasarkan strategi bisnis saat ini
- Tetapkan rencana investasi SI / TI

#### Didorong oleh Metode

Menggunakan teknik untuk mengidentifikasi kebutuhan SI dengan menganalisis proses bisnis

#### Teknologi

- Profesional TI menggunakan model analitis dan alat untuk menghasilkan rencana TI
- Kata 'arsitektur' dapat menggantikan 'rencana' atau 'strategi'

#### Administratif

- Tetapkan anggaran modal & pengeluaran TI untuk mencapai tujuan TI, berdasarkan daftar yang diprioritaskan
- Menganalisis rencana bisnis di tingkat fungsional

#### Organisasi

- Mengembangkan tema untuk investasi TI berdasarkan perjanjian bisnis tentang bagaimana TI dapat membantu mencapai tujuan bisnis
- Disetujui oleh tim manajemen senior

### 4. Kendala dalam Menjalankan proses Strategi SI/TI

- Gagal mendapatkan dukungan top management
- Tidak ada komunikasi dan komitmen untuk berubah dalam tubuh organisasi

- Tidak memperoleh cukup karyawan yang berkualifikasi untuk melakukan pekerjaan dengan benar
- Menyerahkan tanggung jawab kepada individu yang tidak berpengalaman, mempengaruhi waktu saat melakukan pekerjaan
- Tidak ada steering committee yang berkomitmen
- Bisnis strategi tidak dapat mengarahkan usaha strategi SI
- Gagal untuk mengantisipasi pengembangan baru dalam TI yang mungkin berdampak pada strategi
- Mengabaikan orang dan politik dalam formulasi dan perencanaan strategi
- Gagal untuk mengikutsertakan top manajemen yang mencukupi
- Mengabaikan tujuan bisnis
- Gagal untuk menerjemahkan tujuan dan strategi bisnis kedalam rencana aktivitas
- Gagal mengikut sertakan user
- Mengandalkan hanya pada permintaan aplikasi dari user
- Mengabaikan penilaian secara realistic kelemahan internal dari fungsi SI dalam memutuskan penerapan strategi
- Tidak melakukan analisa top-down untuk mengidentifikasi fungsi kritis yang harusnya didukung oleh strategi SI
- Gagal untuk mempertimbangkan strategi SI alternative untuk memberikan pilihan kepada top manajemen
- Gagal dalam mengevaluasi strategi Si bersama dengan seluruh manajer untuk memperoleh dukungan dan kerjasama pada implementasinya

## 5. Establishing an IS/IT process.

Sebuah Proses yang berkelanjutan

- Rencana muncul dari Strategi Si/TI yang perlu untuk diperbaharui atas permintaan, frekuensi ditentukan oleh laju perubahan
- Pengembangan dana kuisisi aplikasi merupakan respon terhadap prioritas permintaan, terkait erat dengan inisiasi bisnis yang lebih luas
- Dukungan terhadap infrastruktur TI, setelah didefinisikan untuk memenuhi kebutuhan strategi bisnis, harus dijaga dalam jangka waktu yang lama
- Mekanisme pengawasan internal dan eksternal terhadap bisnis dan SI/TI merupakan hal penting dalam proses manajemen strategis

## Proses Pembelajaran

Perencanaan Strategi SI/TI juga merupakan sebuah proses pembelajaran. Spesialis SI dan orang bisnis menjadi lebih sadar akan isu bisnis dan teknologi, kemudian belajar untuk mengidentifikasi dan mengembangkan kesempatan dalam lingkungan yang kompetitif

## Inisiasi Siklus Strategi

Pentingnya menyediakan waktu untuk merencanakan perencanaan, karena ini yang akan menentukan kesuksesan.

Strategi dapat maju tergantung pada kematangan prosesnya, pengalaman, starting poin, tujuan perencanaan serta target yang ingin dicapai.

## Kriteria Kesuksesan

Kriteria Kesuksesan dapat diidentifikasi sebagai berikut :

- Meningkatkan kontribusi SI/TI terhadap performa organisasi
- Tingkat keselarasan investasi TI dan strategi bisnis
- Memperoleh keunggulan kompetitif melalui penggunaan SI/TI
- Identifikasi aplikasi baru dan lebih baik
- Identifikasi aplikasi strategis
- Meningkatkan komitmen top manajemen
- Meningkatkan komunikasi antar user
- Peramalan yang lebih baik terhadap kebutuhan sumber daya TI
- Meningkatkan alokasi sumber daya TI
- Pengembangan arsitektur informasi
- Meningkatkan pengakuan SI/TI di dalam organisasi

## 6. IS/IT strategy framework.

### Framework strategy IS/IT

Initiate Strategy Process

Understand the Current Situation and Interpret Business Needs

Determine the business IS Strategy

Define Information and Systems Architecture

Formulate IT Supply Proposals

Prepare migration plans and business case

#### Initiate Strategy Process

- Maksud, tujuan, ruang lingkup dan hasil yang dikonfirmasi.
- Pendekatannya ditentukan
- Sumber daya yang diperoleh seperti alat otomatis
- Peserta bisnis diidentifikasi dan tim dikumpulkan dan, jika perlu, dilatih.
- Mekanisme kemudi dan manajemen untuk proses dibuat
- Merencanakan pekerjaan, tugas, waktu, peran dan tanggung jawab, dan pos pemeriksaan ditetapkan.

#### Understand the Current Situation & Interpret Business Needs

- Analisis strategi bisnis, tujuan, faktor penentu keberhasilan, masalah dan proses kritis
- Evaluasi operasi SI / TI saat ini, sistemnya, penyediaan informasi, sumber daya, organisasi, keterampilan dan layanan
- Analisis lingkungan bisnis eksternal dan internal

#### Determine the business IS Strategy

- Akumulasi permintaan bisnis IS diubah menjadi rekomendasi untuk penerapan SI / TI
- Sistem informasi konseptual dikonsolidasikan dan dipetakan ke dalam portofolio aplikasi

#### Define Information and Systems Architecture

- Langkah ini mengambil hasil analisis proses dan kebutuhan informasi
- Ini mewakili cita-cita masa depan dalam hal proses, informasi, dan sistem
- Perlu untuk merencanakan arah ketika mengembangkan rencana migrasi

#### Formulate IT Supply Proposals

Dalam praktiknya, pada titik ini strategi SI dan proposal penawaran TI dapat dimasukkan kembali ke dalam proses strategi bisnis untuk dipertimbangkan, dan akhirnya untuk konsolidasi. Manajemen bisnis senior dapat melakukannya memutuskan program investasi yang paling menguntungkan dan layak untuk bisnis. Rencana garis kemudian dapat dibangun untuk memetakan peta rute dan tonggak untuk inisiatif utama yang ditetapkan. Ini cenderung mencakup dekat kerjasama dengan area bisnis

untuk menyatukan SI / TI dan aspek bisnis dari pengembangan 'yang disetujui' untuk menghasilkan garis besar rencana migrasi dan kasus bisnis tingkat tinggi untuk masing-masing. Bisnis terperinci masih perlu disiapkan untuk setiap elemen program kapandana pembangunan sebenarnya diminta.

## 7. IS Strategy contents

Tujuan utama dari strategi TI adalah untuk menentukan bagaimana sumber daya dan teknologi akan diperoleh, dikelola dan dikembangkan untuk memenuhi strategi SI bisnis dalam kerangka kerja strategi manajemen. Strategi TI mengatasi faktor-faktor persediaan berikut:

- Manajemen portofolio aplikasi
- Organisasi SI / TI
- Mengelola pengembangan aplikasi
- Mengelola teknologi

# ETIKA & TANGGUNGJAWAB PROFESIONAL DALAM REKAYASA PERANGKAT LUNAK

Febriansyah

## Abstrak

Secara umum para profesional yang bergerak dalam bidang apapun memiliki kode etik dan tanggungjawab dalam menjalankan aktivitas atau pekerjaannya masing-masing. Setiap bidang tentu saja memiliki keunikan sesuai dengan karakteristik pekerjaannya. Hal ini juga berlaku pula dalam dunia teknologi informasi khususnya dalam dunia komputasi, rekayasa perangkat lunak maupun sistem informasi. Seorang Profesional komputasi pada prinsipnya memiliki kewajiban etis bagi klien, atasan, profesional lainnya, dan masyarakat dalam memenuhi tanggung jawab profesiona. Kewajiban-kewajiban tersebut dituangkan dalam kode etik, yang dapat digunakan untuk membuat keputusan tentang masalah etika dalam dunia komputasi atau teknologi informasi.

**Keyword : Etika, Tanggung Jawab Profesional**

## I. PENDAHULUAN

Dalam periode global saat ini ekonomi, ilmu pengetahuan, teknologi dan informasi memainkan peran yang sangat penting bagi umat manusia. Kemajuan berbagai teknologi baru yang belum pernah terjadi sebelumnya memberikan pengaruh yang tidak kecil pada pekerjaan dan kehidupan sehari-hari. Dalam berbagai proses yang terjadi dalam era globalisasi saat ini banyak yang menjadi tidak terkendali atau dapat disebut sebagai periode turbulensi atau periode ketidakpastian. Dalam lingkungan yang penuh dengan kompetisi, pengembangan ilmu pengetahuan dan teknologi untuk kepentingan bisnis juga tak luput dari kondisi ketidakpercayaan para individu yang terlibat didalamnya.

Yang terjadi saat ini dapat digambarkan bagaimana tuntutan perusahaan memiliki pekerja yang profesional dan berkualifikasi tinggi, bagaimana seorang manajer atau teknisi sepatutnya berperilaku ketika dihadapkan dalam situasi yang sulit? Bagaimana seorang karyawan yang senantiasa dituntut untuk dapat mengembangkan produk baru, teknologi, dan bekerja dengan penuh inisiatif dalam kaitannya membuka pasar baru? Bagaimana pula keputusan yang diambil mampu

mempengaruhi lingkungan, kinerja dan konsumen. Secara normatif tentu saja, karyawan atau pelaku bisnis bertindak dalam batas-batas hubungan kerja dan berperilaku sesuai dengan gambaran kerja unit dalam organisasi atau bahkan instruksi dari atasan sebagai personal yang memegang tanggung jawab terakhir. Namun dengan kondisi seperti itu, tidak berarti bahwa pegawai atau karyawan dapat terbebas dari tanggung jawab individu atau kolektif. Prinsip *Responsibility*, dimana setiap individu diharapkan untuk bertindak secara bertanggung jawab terhadap kepentingan manusia. Hal ini tentu saja menimbulkan pertanyaan mengenai perilaku sosial, etika serta tanggung jawab manusia yang terlibat khususnya dalam pengembangan teknologi informasi. Pertanyaan terbesar adalah bagaimana kaitannya etika dan profesionalisme dalam pengembangan teknologi Informasi khususnya dalam bidang rekayasa perangkat lunak (*software engineering*).

Seorang *software engineer* harus memiliki perilaku etis dan moral secara bertanggung jawab jika ingin dihargai sebagai profesional. Dengan demikian, sebagai profesional harus menjunjung tinggi standar normal kejujuran dan integritas.

Namun, terdapat batasan dimana standar perilaku seorang *software engineer* yang dapat diterima diluar aturan hukum yang berlaku. Beberapa hal tersebut menyangkut

1. Kerahasiaan, seorang engineer harus menghormati kerahasiaan atasan maupun klien
2. Seorang engineer menerima pekerjaan sesuai kompetensinya.
3. Seorang engineer harus menyadari hukum setempat yang mengatur

penggunaan kekayaan intelektual seperti paten, hak cipta, dll dalam hal ini harus berhati-hati memastikan bahwa properti intelektual pengusaha dan klien dilindungi.

4. Seorang engineer Perangkat Lunak sebaiknya tidak menggunakan kemampuannya untuk menyalahgunakan komputer orang lain.

## II. TINJAUAN PUSTAKA

### 2.1. Profesional

Profesional berasal dari kata profesi yaitu suatu yang melekat pada seseorang yang memberikan jasa atau keterampilan yang dimilikinya bagi orang lain, maka profesional merupakan sebutan pelakunya. Dalam hal kaitannya dengan Profesional komputasi termasuk didalamnya *desainer hardware, software engineer, database administrator*, analis sistem, dan ilmuwan komputer. Karena profesional memahami bahwa pengetahuannya selalu mengalami kemajuan, profesional juga harus mampu melanjutkan pendidikannya dengan membaca berbagai publikasi. Prinsip yang paling penting menjadi seorang profesional adalah wajib untuk menyimpan informasi spesifik tentang rahasia klien. Mengapa demikian? Karena Profesional pada umumnya cenderung memiliki klien, bukan pelanggan. Hal ini dapat di analogikan seperti penjual yang harus selalu mencoba untuk mampu memuaskan keinginan setiap pelanggannya, maka profesional harus mampu untuk memenuhi kebutuhan kliennya.

### 2.2. Responsibility dan Profesional Responsibility

Responsibility dalam hal ini adalah tanggungjawab. Tanggungjawab menurut Kamus Besar Bahasa Indonesia (2008) adalah keadaan wajib menanggung segala sesuatunya. Tanggungjawab adalah kesadaran manusia akan tingkah lakunya

yang disengaja maupun tidak disengaja. Apabila dikaitkan dengan ranah profesional menurut Wikipedia (2011) dalam hal ini sebut saja tanggungjawab profesional merupakan tindakan secara profesional, mematuhi hukum, menghindari konflik kepentingan, dan menempatkan kepentingan klien di atas kepentingan sendiri.

### 2.3. Kode Etik

Kode etik adalah sistem norma, nilai dan aturan profesional tertulis yang secara tegas menyatakan apa yang benar dan baik dan apa yang tidak benar dan tidak baik bagi profesional. Kode etik menyatakan perbuatan apa yang benar atau salah, perbuatan apa yang harus dilakukan dan apa yang harus dihindari. Tujuan kode etik agar profesional memberikan jasa sebaik-baiknya kepada pemakai atau kliennya. Adanya kode etik akan melindungi perbuatan yang tidak profesional. Ketaatan tenaga profesional terhadap kode etik merupakan ketaatan naluriah yang telah bersatu dengan pikiran, jiwa dan perilaku tenaga profesional. Jadi ketaatan itu terbentuk dari masing-masing orang bukan karena paksaan. Dengan demikian tenaga profesional merasa bila dia melanggar kode etiknya sendiri maka profesinya akan rusak dan yang rugi adalah dia sendiri. Kode etik disusun oleh organisasi profesi sehingga masing-masing profesi memiliki kode etik tersendiri.

## 2.4. Rekayasa Perangkat Lunak

### 2.4.1. Rekayasa

Istilah rekayasa sering ditemui dalam kehidupan sehari-hari khususnya dalam bidang teknik, ada istilah rekayasa perangkat lunak (software engineering), rekayasa sistem (system engineering) sampai pada istilah rekayasa jembatan. Dalam banyak kajian istilah rekayasa mengacu kepada aktivitas analisis, desain, konstruksi, dan proses manajemen suatu entitas (objek kajian tertentu). Dalam kamus bahasa Indonesia, istilah rekayasa justru telah menjadi kerdil sebab diartikan sebagian sebagai ‘usaha atau aktivitas menipu’. Namun dalam konteks ini tidak dapat menyalahkan kamus besar bahasa Indonesia, karena suatu istilah dalam bahasa suatu bangsa mengacu kepada fungsi istilah tersebut digunakan. Dalam aspek global, rekayasa dalam hal teknologi telah ‘mengubah dunia’ dengan begitu dahsyat. Berbagai ‘perubahan’ yang terjadi saat ini menurut sebagian besar ahli ditentukan oleh teknologi dan istilah teknologi sangat erat berkaitan dengan istilah rekayasa.

### 2.4.2. Sejarah & Esensi Rekayasa

Telaah implikasi rekayasa bagi umum baru dimulai pada tahun 1970-an. Pada awal penggunaan istilah rekayasa, masyarakat menganggap bahwa rekayasawan sebagai alat produksi saja, bukan sebagai seorang pengambil keputusan yang bertanggung jawab. Namun seiring dengan perkembangan zaman, saat ini sebagian masyarakat telah memahami dan menyadari bahwa proses dan produk ke rekayasa (teknologi) merupakan hasil dari kreativitas personal. Mengapa ditulis sebagai sebagian masyarakat? Karena memang masih sebagian saja masyarakat yang menyadari bahwa dalam rekayasa nilai moral, perilaku dan kemampuan sang rekayasawan akan sangat mempengaruhi nilai kreasinya; semakin baik nilai moral seorang rekayasawan, biasanya semakin tinggi nilai keselamatan penggunaan hasil rekayasanya. *Engineer* adalah seorang profesional yang bertanggung jawab

untuk memproduksi produk yang cocok untuk digunakan oleh klien. Untuk memastikan bahwa suatu produk layak digunakan, membutuhkan pemahaman tentang lingkungan di mana produk itu digunakan.

## 2.6. Rekayasa Perangkat Lunak

Seiring dengan perkembangan teknologi informasi yang sedemikian pesatnya. Perkembangan komputer yang diawalnya ditujukan sebagai alat bantu hitung manusia telah memiliki kemampuan ribuan kali atau mungkin saja telah jutaan kali lipat sejak awal kemunculannya. Perkembangan Perangkat keras (*hardware*) yang awalnya sangat menentukan dalam perkembangan teknologi informasi saat ini bahkan telah mampu dilampaui oleh perangkat lunak (*software*) dalam memberikan pengaruhnya dalam era industri. Pada masa awal perhitungannya, sistem berbasis komputer dikembangkan dengan menggunakan manajemen yang berorientasi pada perangkat keras. Para manajer proyek memfokuskan diri kepada perangkat keras karena pengembangan sistem membutuhkan biaya yang sangat besar dan menyita waktu. Untuk mengembangkan perangkat lunak pada waktu itu para pemrogram hanya sekedar coba-coba dan sangat berhati-hati. Sekarang walaupun distribusi biaya untuk pengembangan sistem berbasis komputer telah berubah sangat drastis. Dan telah dapat diprediksi bahwa perangkat lunak merupakan salah satu bentuk tunggal yang memiliki biaya sangat tinggi namun dirasakan oleh berbagai enterprise khususnya perusahaan merupakan investasi yang sangat berpengaruh terhadap maju atau mundur bahkan matinya sebuah perusahaan. Untuk memahami sejauh mana manusia membutuhkan perangkat lunak maka harus memahami apa dan bagaimana perangkat lunak tersebut. Sampai saat ini banyak pakar yang telah mendefinisikan perangkat lunak, namun dalam tulisan ini hanya akan diperkenalkan definisi yang sangat terkenal

dari perangkat lunak. Perangkat lunak menurut Roger S. Pressman(1997) adalah

- (1) Perintah (program komputer) yang bila dieksekusi memberikan fungsi dan unjuk kerja seperti yang diinginkan
- (2) Struktur data yang memungkinkan program memanipulasi informasi secara proporsional
- (3) Dan, dokumen yang menggambarkan operasi dan kegunaan program

Berdasarkan hal tersebut dapat dijelaskan bahwa pada prinsipnya perangkat lunak merupakan kumpulan perintah yang dapat digunakan sesuai keinginan dalam hal memanipulasi informasi secara proporsional.

Menurut Roger S. Pressman (1997) perangkat lunak lebih merupakan elemen logika dan bukan merupakan elemen fisik. Dengan demikian, perangkat lunak memiliki ciri yang berbeda dari perangkat keras yaitu

- (1) Perangkat lunak dibangun dan dikembangkan, tidak dibuat dalam bentuk klasik
- (2) Perangkat lunak tidak pernah usang
- (3) Sebagian besar perangkat lunak dibuat secara custom-built, serta tidak dapat dirakit dari komponen yang sudah ada

**2.5. International standard for profesional software development & Ethical reponsibility** Sebagaimana penjelasan sebelumnya bahwa profesional harus memiliki kode etik yang disusun oleh organisasi maka ACM(Association for Computing Machinery) dan IEEE(Institute Of Electrical & Electronics Engineers)

### III. PEMBAHASAN

Pada implementasinya terdapat banyak peluang bahwa rekayasa perangkat lunak dapat digunakan untuk penyalahgunaan fungsi khususnya dalam dunia kejahatan. Untuk memastikan bahwa rekayasa perangkat lunak digunakan dengan baik untuk kemaslahatan umat manusia maka ada upaya agar para software engineer

telah bekerja sama untuk menghasilkan kode etik dan profesional bersama dalam ranah *software engineer* yang menjelaskan secara singkat substansinya dan menyatakan :

*Computers have a central and growing role in commerce, industry, government, medicine, education, entertainment and society at large. Software engineers are those who contribute by direct participation or by teaching, to the analysis, specification, design, development, certification, maintenance and testing of software systems. Because of their roles in developing software systems, software engineers have significant opportunities to do good or cause harm, to enable others to do good or cause harm, or to influence others to do good or cause harm. To ensure, as much as possible, that their efforts will be used for good, software engineers must commit themselves to making software engineering a beneficial and respected profession. In accordance with that commitment, software engineers shall adhere to the following Code of Ethics and Professional Practice.*

Secara singkat pernyataan diatas menjelaskan bahwa dengan perkembangan dunia komputer yang begitu pesat maka para software engineer secara etika harus memiliki kontribusi dengan berpartisipasi langsung atau dengan proses pengajaran, untuk proses analisis, spesifikasi, desain, pengembangan, sertifikasi, pemeliharaan dan pengujian perangkat lunak.

harus berkomitmen untuk menjadikan profesi ini menguntungkan dan dihormati sehingga diperlukan harus mematuhi standar Kode Etik dan Profesional. *Software engineer* berkomitmen untuk membuat analisis, spesifikasi, desain, pengembangan, pengujian dan pemeliharaan perangkat lunak yang menguntungkan dan dihargai sebagai sebuah profesi. Secara

umum *software engineer* harus mematuhi 8(delapan) prinsip berikut :

1. *PUBLIC* – *Software engineers shall act consistently with the public interest.* Pernyataan ini berarti seorang *software engineer* harus bertindak secara konsisten dengan kepentingan publik.
2. *CLIENT AND EMPLOYER* – *Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.* Seorang *software engineer* harus bertindak sesuai dengan kepentingan terbaik klien dan atasan yang konsisten dengan kepentingan publik.
3. *PRODUCT* – *Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.* Seorang *Software engineer* harus memastikan bahwa produk dan modifikasi yang terkait dengan memenuhi standar profesional setinggi mungkin.
4. *JUDGMENT* – *Software engineers shall maintain integrity and independence in their professional judgment.* Seorang *Software engineer* harus mempertahankan integritas dan kemandirian dalam penilaian profesional yang *the Association for Computing Machinery, Inc. 1999*
5. *MANAGEMENT* – *Software engineering Gotterbarn, D and Miller, K managers and leaders shall subscribe to and and Rogerson, S "Software promote an ethical approach to the Engineering Code of*
- 9.

*Ethics," management of software development and SIGCAS Newsletter July 1997. maintenance.* Seorang pimpinan dan manajer *Presman, Roger S., (1997)* *Software engineering* melakukan pendekatan *Software Engineering: A* etis kepada manajemen pengembangan *Practitioner's Approach.* perangkat lunak dan pemeliharaan. *McGraw Hill Book Co.*

6. *PROFESSION* – *Software engineers shall Ian Sommerville, advance the integrity and reputation of the (2004).Software Engineering profession consistent with the public interest. 7th Edition", Addison-Wesley*

Seorang *Software engineer* harus memajukan

[http://en.wikipedia.org/wiki/Professional\\_responsibility](http://en.wikipedia.org/wiki/Professional_responsibility)

integritas dan reputasi profesi yang konsisten [fessional responsibility](http://en.wikipedia.org/wiki/Professional_responsibility) dengan kepentingan publik. *Michael C. Loui, Keith W.*

7. *COLLEAGUES* – *Software engineers shall be Miller, Ethics and Professional fair to and supportive of their colleagues. Responsibility in* seorang *Software engineer* harus bersikap adil *Computing.Encyclopedia Of* dan mendukung rekan-rekan taua kolega dalam *Computer Science* pekerjaan.
8. *SELF* – *Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.* Seorang *Software engineer* harus berpartisipasi dalam pembelajaran seumur hidup tentang praktek profesi yang dimiliki dan akan mempromosikan pendekatan etis untuk profesinya.

#### IV. KESIMPULAN

Berdasarkan apa yang telah dijelaskan di atas organisasi profesional sesuai dengan bidang bahwa fungsi *engineer* tidak hanya merancang yang diembannya. Selain aturan hukum dan membuat, merawat dan mengembangkan kode etik yang harus ditaati disebabkan karena teknologi yang dikembangkan tetapi pada fungsi profesional adalah mengutamakan klien, implementasinya tindakan yang dilakukan atasan dan ranah publik yang menyangkut dalam hal ini perilaku yang dibangun pun harus profesi *engineer* yang bersangkutan.

# ETIKA DAN TANGGUNGJAWAB DALAM PENGEMBANGAN PERANGKAT LUNAK

Fido Rizki  
(182420060)

[fidorizki@gmail.com](mailto:fidorizki@gmail.com)

Magister Teknik Informatika, Universitas Bina  
Darma Palembang

## Abstrak

Secara umum para profesional yang bergerak dalam bidang apapun memiliki kode etik dan tanggungjawab dalam menjalankan aktivitas atau pekerjaannya masing-masing. Setiap bidang tentu saja memiliki keunikan sesuai dengan karakteristik pekerjaannya. Hal ini juga berlaku pula dalam dunia teknologi informasi khususnya dalam dunia komputasi, rekayasa perangkat lunak maupun sistem informasi. Seorang Profesional komputasi pada prinsipnya memiliki kewajiban etis bagi klien, atasan, profesional lainnya, dan masyarakat dalam memenuhi tanggung jawab profesional. Kewajiban-kewajiban tersebut dituangkan dalam kode etik, yang dapat digunakan untuk membuat keputusan tentang masalah etika dalam dunia komputasi atau teknologi informasi.

**Keyword :** *Kode etik, Tanggungjawab Profesional*

## I PENDAHULUAN

Dalam periode global saat ini ekonomi, ilmu pengetahuan, teknologi dan informasi memainkan peran yang sangat penting bagi umat manusia. Kemajuan berbagai teknologi baru yang belum pernah terjadi sebelumnya memberikan pengaruh yang tidak kecil pada pekerjaan dan kehidupan sehari-hari. Dalam berbagai proses yang terjadi dalam era globalisasi saat ini banyak yang menjadi tidak terkendali atau dapat disebut sebagai periode turbulensi atau

periode ketidakpastian. Dalam lingkungan yang penuh dengan kompetisi, pengembangan ilmu pengetahuan dan teknologi untuk kepentingan bisnis juga tak luput dari kondisi ketidakpercayaan para individu yang terlibat didalamnya.

Yang terjadi saat ini dapat digambarkan bagaimana tuntutan perusahaan memiliki pekerja yang profesional dan berkualifikasi tinggi, bagaimana seorang manajer atau teknisi sepatutnya berperilaku ketika dihadapkan dalam situasi yang

sulit? Bagaimana seorang karyawan yang senantiasa dituntut untuk dapat mengembangkan produk baru, teknologi, dan bekerja dengan penuh inisiatif dalam kaitannya membuka pasar baru? Bagaimana pula keputusan yang diambil mampu mempengaruhi lingkungan, kinerja dan konsumen. Secara normatif tentu saja, karyawan atau pelaku bisnis bertindak dalam batas-batas hubungan kerja dan berperilaku sesuai dengan gambaran kerja unit dalam organisasi atau bahkan instruksi dari atasan sebagai personal yang memegang tanggung jawab terakhir. Namun dengan kondisi seperti itu, tidak berarti bahwa pegawai atau karyawan dapat bebas dari tanggung jawab individu atau kolektif. Prinsip *Responsibility*, dimana setiap individu diharapkan untuk bertindak secara bertanggung jawab terhadap kepentingan manusia. Hal ini tentu saja menimbulkan pertanyaan mengenai perilaku sosial, etika serta tanggung jawab manusia yang terlibat khususnya dalam pengembangan teknologi informasi. Pertanyaan terbesar adalah bagaimana kaitannya etika dan profesionalisme dalam pengembangan teknologi Informasi khususnya dalam bidang rekayasa perangkat lunak (*software engineering*)[4].

Sebagaimana engineer yang lain, seorang *Software Engineer* harus meyakini bahwa pekerjaannya melibatkan tanggungjawab yang lebih luas dari hanya sekedar keterampilan teknis. Sebagaimana diketahui bahwa ranah rekayasa perangkat lunak secara jelas dibatasi oleh hukum lokal, nasional maupun internasional. Seorang *software engineer* harus memiliki perilaku etis dan moral secara bertanggung jawab jika ingin dihargai sebagai profesional. Dengan demikian, sebagai profesional harus menjunjung tinggi standar normal kejujuran dan integritas. Namun, terdapat batasan dimana standar perilaku seorang *software engineer* yang dapat diterima diluar aturan hukum yang berlaku. Beberapa hal tersebut menyangkut[1]

1. Kerahasiaan, seorang engineer harus menghormati kerahasiaan atasan maupun klien
2. Seorang engineer menerima pekerjaan sesuai kompetensinya.
3. Seorang engineer harus menyadari hukum setempat yang mengatur penggunaan kekayaan intelektual seperti paten, hak cipta, dll dalam hal ini harus berhati-hati memastikan bahwa properti intelektual pengusaha dan klien dilindungi.

4. Seorang engineer Perangkat Lunak sebaiknya tidak menggunakan kemampuannya untuk menyalahgunakan komputer orang lain.

## **II TINJAUAN PUSTAKA**

### **2.1 Apa itu Profesional?**

Sebagaimana judul sub diatas, apa itu profesional? Profesional berasal dari kata profesi yaitu suatu yang melekat pada seseorang yang memberikan jasa atau keterampilan yang dimilikinya bagi orang lain, maka profesional merupakan sebutan pelakunya[8]. Dalam hal kaitannya dengan Profesional komputasi termasuk didalamnya *desainer hardware, software engineer, database administrator*, analis sistem, dan ilmuwan komputer. Dengan cara bagaimana profesi komputasi menyerupai pekerjaan yang tidak tradisional dianggap sebagai sebuah profesi, seperti tukang pipa, model fashion, dan pramuniaga? Sebuah aktivitas dianggap sebagai sebuah profesi jika memiliki karakteristik khususnya dalam penguasaan pengetahuan khusus dan keterampilan, sering disebut "body knowledge" yang diperoleh melalui pendidikan formal atau pengalaman praktis.

Karena profesional memahami bahwa pengetahuannya selalu mengalami kemajuan, profesional juga harus mampu melanjutkan pendidikannya dengan membaca berbagai publikasi. Prinsip yang paling penting menjadi seorang profesional adalah wajib untuk menyimpan informasi spesifik tentang rahasia klien. Mengapa demikian? Karena Profesional pada umumnya cenderung memiliki klien, bukan pelanggan. Hal ini dapat di analogikan seperti penjual yang harus selalu mencoba untuk mampu memuaskan keinginan setiap pelanggannya, maka profesional harus mampu untuk memenuhi kebutuhan kliennya.

### **2.2 Apa itu Responsibility dan Profesional responsibility?**

Responsibility dalam hal ini adalah tanggungjawab. Tanggungjawab menurut Kamus Besar Bahasa Indonesia (2008) adalah keadaan wajib menanggung segala sesuatunya[2]. Tanggungjawab adalah kesadaran manusia akan tingkah lakunya yang disengaja maupun tidak disengaja. Apabila dikaitkan dengan ranah profesional menurut Wikipedia (2011) dalam hal ini sebut saja tanggungjawab profesional merupakan tindakan secara profesional, mematuhi hukum,

menghindari konflik kepentingan, dan menempatkan kepentingan klien di atas kepentingan sendiri.

### 2.3 Apa itu etika & kode etik

Kode etik adalah sistem norma, nilai dan aturan profesional tertulis yang secara tegas menyatakan apa yang benar dan baik dan apa yang tidak benar dan tidak baik bagi profesional. Kode etik menyatakan perbuatan apa yang benar atau salah, perbuatan apa yang harus dilakukan dan apa yang harus dihindari[1]. Tujuan kode etik agar profesional memberikan jasa sebaik-baiknya kepada pemakai atau kliennya. Adanya kode etik akan melindungi perbuatan yang tidak profesional.

Ketaatan tenaga profesional terhadap kode etik merupakan ketaatan naluriah yang telah bersatu dengan pikiran, jiwa dan perilaku tenaga

profesional. Jadi ketaatan itu terbentuk dari masing-masing orang bukan karena paksaan. Dengan demikian tenaga profesional merasa bila dia melanggar kode etiknya sendiri maka profesinya akan rusak dan yang rugi adalah dia sendiri. Kode etik disusun oleh organisasi profesi sehingga masing-masing profesi memiliki kode etik tersendiri.

### 2.4 Rekayasa Perangkat Lunak

Istilah rekayasa sering ditemui dalam kehidupan sehari-hari khususnya dalam bidang teknik, ada istilah rekayasa perangkat lunak (software engineering), rekayasa sistem (system engineering) sampai pada istilah rekayasa jembatan. Dalam banyak kajian istilah rekayasa mengacu kepada aktivitas analisis, desain, konstruksi, dan proses manajemen suatu entitas (objek kajian tertentu). Dalam kamus bahasa Indonesia, istilah rekayasa justru telah menjadi kerdil sebab diartikan sebagian sebagai ‘usaha atau aktivitas menipu’. Namun dalam konteks ini tidak dapat menyalahkan kamus besar bahasa Indonesia, karena suatu istilah dalam bahasa suatu bangsa mengacu kepada fungsi istilah tersebut digunakan. Dalam aspek global, rekayasa dalam hal teknologi telah ‘mengubah dunia’ dengan begitu dahsyat. Berbagai ‘perubahan’ yang terjadi saat ini menurut sebagian besar ahli ditentukan oleh teknologi dan istilah teknologi sangat erat berkaitan dengan istilah rekayasa. Untuk menggali pemaknaan mengenai *engineering* (rekayasa ; kerekayasaan) sebaiknya diuraikan dari awal penggunaan istilah tersebut. Istilah *engineering* tentunya berangkat dari istilah dasar *teknik*. Arti kata teknik itu

sendiri adalah penerapan sains untuk kesejahteraan umat manusia (Zen, 1981). Istilah ini dipersempit oleh Martin & Schinzinger (1994) yaitu rekayasa merupakan penerapan ilmu pengetahuan dalam penggunaan sumber daya alam demi manfaat bagi masyarakat dan umat manusia, sedangkan rekayasa wan adalah seseorang yang menciptakan produk dan proses proses untuk memenuhi kebutuhan dasar manusia (pangan, papanda dan sandang), dengan akibat tambahan, meningkatkan kemudahan, kekuatan dan keindahan didalam kehidupan manusia sehari-hari.

Seiring dengan perkembangan teknologi informasi yang sedemikian pesatnya. Perkembangan komputer yang di awalnya ditujukan sebagai alat bantu hitung manusia telah memiliki kemampuan ribuan kali atau mungkin saja telah jutaan kali lipat sejak awal kemunculannya.

Perkembangan Perangkat keras (*hardware*) yang awalnya sangat menentukan dalam perkembangan teknologi informasi saat ini bahkan telah mampu dilampaui oleh perangkat lunak (*software*) dalam memberikan pengaruhnya dalam era industri. Pada masa awal perhitungannya, sistem berbasis komputer dikembangkan

dengan menggunakan manajemen yang berorientasi pada perangkat keras. Para manajer proyek memfokuskan diri kepada perangkat keras karena pengembangan sistem membutuhkan biaya yang sangat besar dan menyita waktu. Untuk mengembangkan perangkat lunak pada waktu itu para pemrogram hanya sekedar coba-coba dan sangat berhati-hati. Sekarang walaupun distribusi biaya untuk pengembangan sistem berbasis komputer telah berubah sangat drastis. Dan telah dapat diprediksi bahwa perangkat lunak merupakan salah satu bentuk tunggal yang memiliki biaya sangat tinggi namun dirasakan oleh berbagai enterprise khususnya perusahaan merupakan investasi yang sangat berpengaruh terhadap maju atau mundur bahkan matinya sebuah perusahaan. Untuk memahami sejauh mana manusia membutuhkan perangkat lunak maka harus memahami apa dan bagaimana perangkat lunak tersebut. Sampai saat ini banyak pakar yang telah mendefinisikan perangkat lunak, namun dalam tulisan ini hanya akan diperkenalkan definisi yang sangat terkenal dari perangkat lunak. Perangkat lunak menurut Roger S. Pressman(1997) adalah[5]

1. Perintah (program komputer) yang bila dieksekusi memberikan fungsi

dan unjuk kerja seperti yang diinginkan

2. Struktur data yang memungkinkan program memanipulasi informasi secara proporsional
3. Dan, dokumen yang menggambarkan operasi dan kegunaan program

Berdasarkan hal tersebut dapat dijelaskan bahwa pada prinsipnya perangkat lunak merupakan kumpulan perintah yang dapat digunakan sesuai keinginan dalam hal memanipulasi informasi secara proporsional. Definisi ini agak sedikit berbeda dengan definisi menurut Ian Sommerville (2004),

### III PEMBAHASAN

Pada implementasinya terdapat banyak peluang bahwa rekayasa perangkat lunak dapat digunakan untuk penyalahgunaan fungsi khususnya dalam dunia kejahatan. Untuk memastikan bahwa rekayasa perangkat lunak digunakan dengan baik untuk kemaslahatan umat manusia maka ada upaya agar para software engineer harus berkomitmen untuk menjadikan profesi ini menguntungkan dan dihormati sehingga diperlukan harus mematuhi standar Kode Etik dan Profesional. *Software engineer* berkomitmen untuk membuat analisis, spesifikasi, desain, pengembangan,

pengujian dan pemeliharaan perangkat lunak yang menguntungkan dan dihargai sebagai sebuah profesi. Secara umum *software engineer* harus mematuhi 8(delapan) prinsip berikut[3]:

1. *PUBLIC – Software engineers shall act consistently with the public interest.* Pernyataan ini berarti seorang software engineer harus bertindak secara konsisten dengan kepentingan publik.
2. *CLIENT AND EMPLOYER – Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.* Seorang software engineer harus bertindak sesuai dengan kepentingan terbaik klien dan atasan yang konsisten dengan kepentingan publik.
3. *PRODUCT – Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.* Seorang Software engineer harus memastikan bahwa produk dan modifikasi yang terkait dengan memenuhi

standar profesional setinggi mungkin.

4. *JUDGMENT – Software engineers shall maintain integrity and independence in their professional judgment.*

Seorang Software engineer harus mempertahankan integritas dan kemandirian dalam penilaian profesional yang dimiliki.

5. *MANAGEMENT – Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.*

Seorang pimpinan dan manajer Software engineering melakukan pendekatan etis kepada manajemen pengembangan perangkat lunak dan pemeliharaan.

6. *PROFESSION – Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.*

Seorang Software engineer harus memajukan integritas dan reputasi profesi yang konsisten dengan kepentingan publik.

7. *COLLEAGUES – Software engineers shall be fair to and*

*supportive of their colleagues.*

seorang Software engineer harus bersikap adil dan mendukung rekan-rekan atau kolega dalam pekerjaan.

8. *SELF – Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.*

Seorang Software engineer harus berpartisipasi dalam pembelajaran seumur hidup tentang praktek profesi yang dimiliki dan akan mempromosikan pendekatan etis untuk profesinya.

#### IV KESIMPULAN

Berdasarkan apa yang telah dijelaskan di atas bahwa fungsi *engineer* tidak hanya merancang, membuat, merawat dan mengembangkan teknologi yang dikembangkan tetapi pada implementasinya tindakan yang dilakukan dalam hal ini perilaku yang dibangun pun harus mengikuti kode etik yang telah ditetapkan oleh organisasi profesional sesuai dengan bidang yang diembannya. Selain aturan hukum dan kode etik yang harus ditaati disebabkan karena fungsi

profesional adalah mengutamakan klien, atasan dan ranah publik yang menyangkut profesi *engineer* yang bersangkutan

## V DAFTAR PUSTAKA

- [1] *Donald Gotterbarn, 1997 The Professionalization of Software Engineering and its significance for Ethics Education East Tennessee State University Johnson City, Tennessee*
- [2] *Kamus Besar Bahasa Indonesia tahun 2008 Pusat Bahasa Departemen Pendidikan Nasional Indonesia*
- [3] *The Institute of Electrical and Electronics Engineers, Inc. And The Institute of Electrical and Electronics Engineers, Inc. And and Rogerson, S "Software*
- [4] *Engineering Code of Ethics," SIGCAS Newsletter July 1997.*