



Mata Ajar

STRATEGI DAN IMPLEMENTASI E-COMMERCE

Topik Bahasan

STRATEGI PENGADAAN SOFTWARE ELECTRONIC COMMERCE

Versi

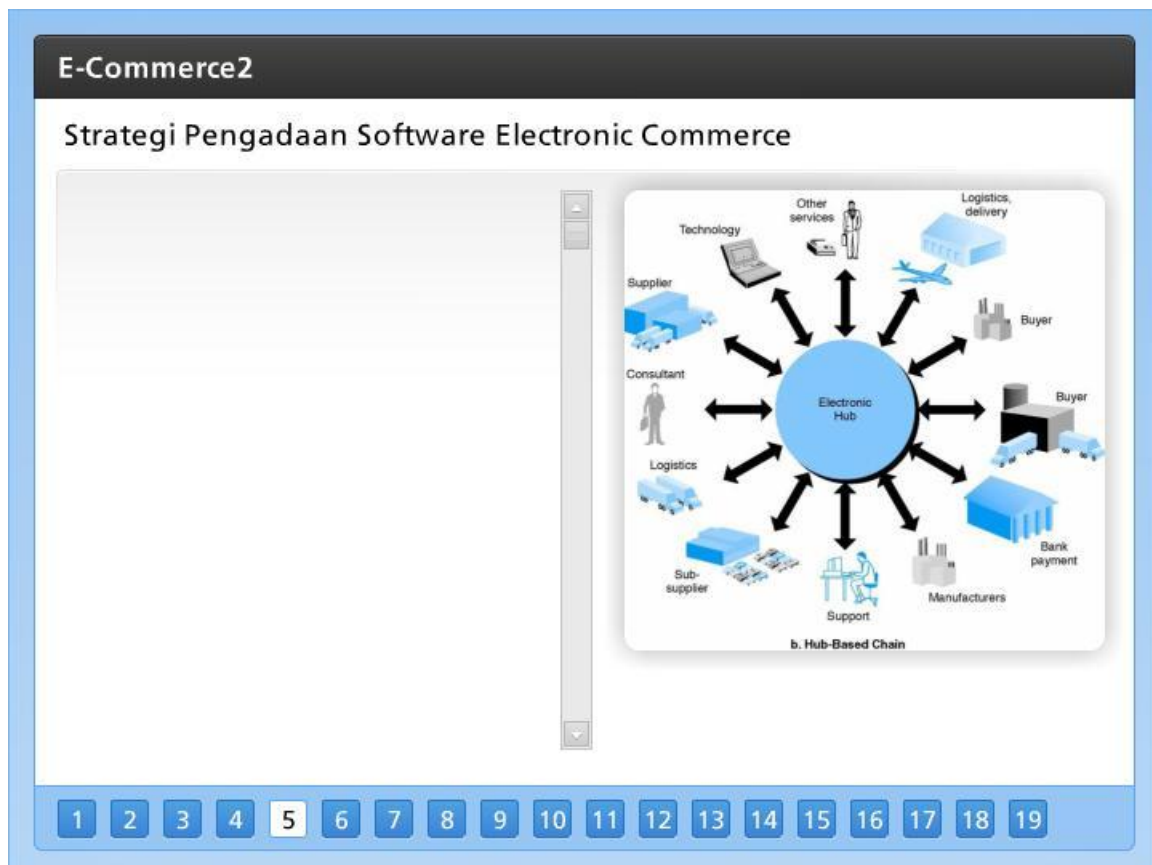
2013/1.0

Nama File

SDIE-9A-StrategiPengadaan.pdf

Referensi Pembelajaran

9-A



Bagi perusahaan yang ingin mengembangkan sistem perangkat lunak (software) dan teknologi informasinya agar dapat mengimplementasikan konsep E-Commerce, ada tiga pilihan pengadaan aplikasi yang dapat dilakukan (Fingar, 2000). Pilihan pertama adalah membeli paket E-Commerce siap pakai (siap terap) yang telah tersedia di pasar; pilihan kedua adalah membuat sendiri aplikasi yang dibutuhkan (bekerja sama dengan Divisi Teknologi Informasi internal perusahaan); atau pilihan ketiga berupa pembelian komponen-komponen E-Commerce yang kemudian saling dikoneksikan dan diintegrasikan. Masing-masing skenario memiliki kelebihan dan kekurangannya, yang tentu saja merupakan tugas manajemen perusahaan untuk mengkaji dan menetapkan strategi yang ingin diterapkan. Berikut ini adalah gambaran ringkas mengenai seluk beluk dari masing-masing pendekatan.

Membeli Paket Siap Terap

Ditinjau dari jenisnya, paling tidak ada dua karakteristik paket yang beredar secara luas di pasaran. Jenis paket pertama adalah aplikasi siap terap yang

standar data dan fasilitasnya telah baku, sehingga perusahaan tidak dapat mengadakan perubahan terhadap alur kerja program maupun jenis datanya. Aplikasi ini biasa diistilahkan sebagai software yang bersifat statis. Jenis paket kedua adalah aplikasi siap terap dimana susunan data dan alur kerja program dapat sedikit banyak dirubah atau diganti sesuai dengan kebutuhan perusahaan (customisation). Aplikasi dengan kemampuan demikian dikatakan sebagai software yang bersifat dinamis.

Kelebihan dari aplikasi pertama tentu saja harganya yang relatif murah, karena prinsip bisnis dari sang pembuat software adalah "mass product", membuat software statis sebanyak-banyaknya untuk dipasarkan secara retail ke perusahaan-perusahaan. Jenis aplikasi ini baik dipergunakan untuk perusahaan-perusahaan kecil yang tidak perlu membangun sistem E-Commerce besar dan kompleks. Kelebihan kedua adalah bahwa biasanya software jenis ini mudah dipelajari dan cepat diimplementasikan karena sederhana dan lugas. Dengan berbekal buku panduan (manual), maka perusahaan dapat dengan mudah mempelajarinya. Keuntungan lainnya adalah tersedianya cukup dukungan supports dan services dari berbagai kalangan karena sifatnya yang umum dan mudah dimengerti. Kekurangannya jelas terletak pada keberadaannya sebagai "black box", dalam arti kata tidak dapat diubah-ubah alur programnya sesuai dengan keinginan perusahaan karena pabrik pembuat software tersebut tidak memberikan "source code" programnya. Hal ini menyebabkan rendahnya tingkat integritas perangkat lunak jika ingin dihubungkan dengan perangkat lunak lain, terutama yang telah dimiliki oleh perusahaan saat ini. Tidak jarang terjadi kasus dimana terpaksa harus diadakan redundansi dalam proses pemasukan data (data entry), karena paket siap terap yang dibeli tidak dapat secara otomatis dihubungkan dengan aplikasi lain yang dimiliki perusahaan.

Software jenis kedua yang bersifat dinamis, secara prinsip lebih baik dan lebih fleksibel dibandingkan dengan yang berjenis statis. Tentu saja konsekuensinya adalah lebih mahalnya harga jual dari software terkait. Namun tingkat fleksibilitas yang ada biasanya sangat terbatas, sehingga tidak dapat secara utuh (seratus persen) memenuhi kebutuhan unik perusahaan. Karena adanya proses

kustomisasi dari perusahaan, maka jelas implementasinya agak lebih lambat dibandingkan dengan software jenis statis, karena alasan adanya proses-proses instalasi yang harus dilakukan terlebih dahulu. Perusahaan juga harus menyusun manualnya diluar manual yang ada agar para pengguna dapat mengetahui fasilitas-fasilitas yang ada setelah proses kustomisasi dilakukan. Software jenis ini secara prinsip tetap merupakan "black box", sehingga sulit diintegrasikan dengan aplikasi-aplikasi lain. Jika dapat sekalipun, harus diadakan ekstra usaha untuk membuat program koneksi (interface) yang tentu saja memakan waktu dan biaya yang tidak sedikit.

Secara umum, membeli paket statis bukanlah merupakan pilihan terbaik di saat lingkungan bisnis secara dinamis berganti dari waktu ke waktu. Selain karena tidak fleksibel, perusahaan tidak akan dapat memperoleh keuntungan yang signifikan darinya karena tidak secara utuh mensimulasikan kebutuhan perusahaan sesungguhnya. Membeli paket dinamis merupakan pilihan yang cukup baik sejauh perusahaan yang bersangkutan memiliki alokasi finansial yang cukup karena biasanya semakin tinggi tingkat fleksibilitas software, akan semakin mahal pula harganya. Hal lain yang perlu diingat adalah akan tergantungnya perusahaan terkait dengan vendor, konsultan, atau pihak ketiga lain yang mengerti benar mengenai seluk beluk software tersebut, sehingga perusahaan harus memiliki hubungan kerja yang terkoordinasi dengan baik.

Membangun Aplikasi Mandiri

Pilihan yang sering pula diterapkan oleh perusahaan adalah mencoba untuk membangun aplikasi E-Commerce-nya sendiri. Alasan utamanya adalah karena sulit ditemukan aplikasi yang secara utuh seratus persen dapat sesuai dengan kebutuhan unik perusahaan. Strategi ini biasa pula dilakukan oleh perusahaan yang memiliki Divisi Teknologi yang cukup kuat, dalam arti kata memiliki anggaran dana yang cukup, dan diisi oleh berbagai sumber daya manusia yang memiliki kompetensi dan keahlian memadai untuk membangun dan mengembangkan aplikasi bisnis.

Selain dapat memenuhi kebutuhan rinci dari perusahaan, kelebihan lain dari membangun aplikasi secara mandiri adalah tidak tergantungnya perusahaan terhadap pihak-pihak lain. Keuntungan selanjutnya adalah kemungkinan dirubahnya software terkait karena merupakan "white box", artinya perusahaan memiliki kode-kode program yang dibutuhkan. Dan keuntungan terakhir adalah memperkecil kemungkinan pesaing untuk memiliki aplikasi serupa karena sifatnya yang unik dan hanya diketahui oleh perusahaan pembuatnya.

Kekurangan dari strategi ini cukup banyak. Yang pertama adalah sangat menyita waktu, karena proses pembuatan yang cukup lama, terutama jika ingin mengikuti siklus pengembangan software dengan kualitas yang diinginkan. Durasi waktu ini tentu saja berbanding lurus dengan besarnya biaya pengembangan dan kerugian waktu (opportunity loss) yang harus ditanggung perusahaan. Kekurangan selanjutnya adalah kebutuhan perusahaan akan tenaga pembuat software yang handal dan berpengalaman agar kualitas software yang dihasilkan dapat dipertanggungjawabkan dengan baik. Harus diperhatikan bahwa dalam E-Commerce terdapat banyak sekali jenis komponen aplikasi yang sangat beragam fungsinya dan harus dihubungkan satu dengan lainnya. Tanpa adanya SDM yang demikian, tidak mustahil software yang dihasilkan akan berkualitas rendah yang dapat menjadi "bom waktu" bagi perusahaan di masa mendatang. Kekurangan selanjutnya adalah kesulitan perusahaan yang akan dialami jika terjadi perubahan kebutuhan bisnis yang secara langsung maupun tidak langsung berpengaruh terhadap spesifikasi software yang dibangun. Dengan kata lain, tingkat adaptasi software terhadap perubahan bisnis akan sangat rendah. Kekurangan lain adalah sulitnya membangun software ideal dalam arti kata yang dapat dengan mudah beradaptasi dengan pertumbuhan teknologi yang sedemikian cepat. Belum terhitung kemampuan software terkait untuk dihubungkan dengan perangkat lunak lain yang mungkin telah dimiliki dan akan dibeli perusahaan dikemudian hari.

Pada akhirnya, membangun aplikasi secara mandiri pada masa-masa ini bukanlah pilihan yang bijaksana. Alasan pertama adalah karena bagi mayoritas perusahaan, fungsi teknologi informasi bukanlah merupakan proses bisnis inti (core

processes), tapi merupakan fungsi penunjang. Dengan kata lain, keberadaan aktivitas yang berhubungan dengan pengembangan teknologi informasi oleh departemen terkait di perusahaan dimata manajemen merupakan suatu "cost center". Alasan kedua adalah karena kegiatan tersebut memakan waktu dan biaya yang sangat tinggi dengan tingkat kesuksesan yang masih harus dipertanyakan. Alasan ketiga adalah keberadaan proses ini justru akan memperlambat gerak perusahaan, terutama dalam bergerak cepat beradaptasi dengan setiap perubahan lingkungan bisnis yang terjadi.

Menyusun Komponen Aplikasi

Sejalan dengan perkembangan teknologi informasi terutama yang didasari pada paradigma "object oriented" (pembangunan sistem informasi berbasis objek), pemecahan terhadap berbagai permasalahan yang timbul pada dua pendekatan konvensional di atas mulai ditemukan oleh berbagai ahli manajemen dan praktisi teknologi informasi. Pada dasarnya, pendekatan yang dikembangkan adalah mencoba menggabungkan sifat-sifat positif dari kedua pendekatan terdahulu, sejalan dengan usaha untuk mengurangi sifat negatif dari keduanya.

Pada pendekatan ketiga ini, perusahaan berusaha untuk membeli beragam komponen aplikasi berukuran kecil yang tersedia di pasaran, dan menghubung-hubungkannya secara mandiri agar tercipta sebuah aplikasi yang lebih besar sesuai dengan kebutuhan spesifik perusahaan. Komponen-komponen tersebut dewasa ini telah terjual di pasaran dengan harga yang sangat murah. Konsep pemasangannya kurang lebih dapat dianalogikan seperti mainan "lego" yang digemari oleh anak-anak. Dalam permainan tersebut, terdapat berbagai jenis warna dan ukuran komponen lego dengan sifat dan karakteristiknya masing-masing. Yang membuatnya menarik adalah kemampuan berbagai komponen lego tersebut untuk saling dihubung-hubungkan sehingga terbentuk suatu bangunan yang lebih besar. Konsep yang sama diterapkan pada aplikasi E-Commerce, dimana perusahaan mencoba untuk merencanakan dan mengembangkan sistem informasi E-Commerce-nya dengan cara membangunnya secara kontinyu dan berkesinambungan.

Kelebihan pertama dari pendekatan ini adalah biayanya yang relatif murah karena masing-masing komponen berukuran kecil dan diperjualbelikan secara eceran (retail). Kelebihan kedua adalah kemampuan perusahaan untuk membangun sistem dengan spesifikasi unik yang dibutuhkan, karena yang bersangkutan hanya tinggal menyambung-nyambungkan komponen yang ada. Kelebihan ketiga adalah mudahnya dilakukan perubahan secara cepat tanpa merusak kinerja total dari sistem secara signifikan, karena jika terjadi perubahan kebutuhan, hanya komponen terkait saja yang perlu diganti dengan komponen baru sesuai dengan keperluan. Kelebihan keempat adalah tingginya tingkat skalabilitas aplikasi karena sejalan dengan perkembangan perusahaan, sistem aplikasi E-Commerce yang ada dapat "tumbuh" menyesuaikan ukuran dan kebutuhan perusahaan yang lebih besar dan kompleks. Kelebihan kelima adalah rendahnya biaya pembangunan sistem dan cepatnya durasi implementasi karena proses pengembangannya tidak memakan waktu yang lama. Kelebihan keenam adalah sifat komponen yang siap terap (ready-to-use) dan mudah dipahami (user friendly), sehingga perusahaan tidak perlu membutuhkan sumber daya manusia dengan kemampuan dan keahlian khusus yang di atas rata-rata. Kelebihan lainnya adalah kemampuan komponen-komponen tersebut untuk berkomunikasi dengan aplikasi besar lainnya karena dibangun dengan berpedoman pada standar yang telah disepakati luas secara internasional. Ketergantungan kepada pihak ketiga pun menjadi berkurang karena pada dasarnya banyak sekali pihak-pihak yang menyediakan komponen-komponen serupa di pasaran, bahkan tidak berlebihan jika banyak diantaranya yang diberikan secara gratis (free shareware).

Kekurangan dari pendekatan ini pun ada, namun tidak dinilai sangat signifikan oleh perusahaan karena kelebihan-kelebihan yang ditawarkan. Pertama adalah adanya kemungkinan penurunan kinerja software (kurang optimum) sejalan dengan meningkatnya jumlah komponen yang dipergunakan (berbanding lurus dengan durasi total pengiriman data dari satu tempat ke tempat lainnya, atau sering dikatakan sebagai "throughput"). Kedua adalah bahwa perusahaan harus memiliki cetak biru (blue print) desain aplikasi E-Commerce yang akan dikembangkan agar tidak terjadi fenomena pembangunan sistem secara tambal sulam, yang dapat merendahkan tingkat realibilitas software. Hal terakhir adalah

diperlukannya manajemen pengelolaan yang baik, karena banyaknya komponen-komponen yang terlibat. Dokumentasi teknis secara teratur dan berkala harus direvisi dan dikelola dengan baik sehingga perusahaan dapat melacak sejarah dan rencana pengembangan software yang dilakukan.

Pada akhirnya, perusahaan-perusahaan modern lebih memilih untuk menggunakan pendekatan terakhir, yaitu membangun sistem E-Commerce melalui penggabungan komponen-komponen mandiri karena selain mudah dan murah, cara tersebut sejalan dengan kebutuhan bisnis yang berubah secara cepat dari waktu ke waktu.