BAB II
LANDASAN TEORI

II.1 Cloud Computing

Cloud computing mengacu pada aplikasi dan layanan yang berjalan pada jaringan terdistribusi menggunakan sumber daya virtual dan diakses oleh protokol internet umum dan standar jaringan. Hal ini dibedakan dengan gagasan bahwa *resource* adalah virtual dan tak terbatas dan itu merupakan rincian dari *physical system* dimana *software* berjalan diabstraksikan dari pengguna.

Dalam upaya untuk lebih menggambarkan *cloud computing*, sejumlah jenis *cloud* telah didefinisikan. Terdapat dua kelas yang berbeda dari *cloud* diantaranya adalah yang didasarkan pada *deployment model*, dan berdasarkan *service model*. *Deployment model* memberitahu dimana *cloud* berada dan untuk tujuan apa. *Public*, *private*, *community*, dan *hybrid clouds* merupakan *deployment models*.

Service model menggambarkan jenis layanan yang penyedia layanan tawarkan. *Service model* yang paling dikenal diantaranya adalah *software as a service*, *platform as a service*, dan *infrastructure as a service* atau dikenal sebagai SPI model. *Service model* saling membangun satu sama lain dan menentukan apa yang *vendor* harus kelola, dan apa yang merupakan tanggung jawab klien.

Cloud computing merupakan pergeseran paradigma nyata dalam cara dimana sistem dikerahkan. Skala besar dari sistem *cloud computing* telah diaktifkan seiring popularitas internet, dan pertumbuhan beberapa perusahaan jasa besar. *Cloud computing* membuat impian akan *utility computing* menjadi lebih mungkin, jauh lebih *scalable*, dan sistem yang tersedia secara universal.

Tidak semua aplikasi diuntungkan dengan *deployment* di *cloud*. Masalah dengan *latency*, kontrol transaksi, dan dalam keamanan tertentu, serta kepatuhan terhadap peraturan patut menjadi perhatian khusus. [2]

II.2 Microsoft Azure

Microsoft Azure adalah platform *cloud computing Microsoft*, yang menyediakan berbagai macam layanan yang dapat digunakan tanpa membeli. *Azure* memungkinkan pesatnya pengembangan yan memberikan solusi, dan menyediakan sumber daya untuk menyelesaikan tugas-tugas yang mungkin tidak sesuai dalam lingkungan lokal. Komputasi, *storage*, jaringan, dan layanan aplikasi *Azure* berfokus pada membangun solusi besar tanpa perlu khawatir tentang bagaimana pemasangan *physical infrastructure*.

Microsoft menyediakan support untuk *public*, *private*, dan *hybrid cloud*. *Windows Azure Pack* adalah *add-on* gratis untuk *Microsoft System Center* yang memungkinkan kita untuk meng-*host* banyak layanan inti *Azure* di data *center* kita sendiri, dan memberikan pengalaman *self-service* portal ke pengguna. Kita dapat mengintegrasikan itu semua ke dalam *hybrid cloud* melalui penggunaan *virtual private network.* [3]

II.3 Windows Phone

Windos phone diperkenalkan pada tanggal 15 Februari 2010 pada pameran MWC, Barcelona. *Microsoft* memperkenalkan sebuah sistem operasi yang dianggap sebagai *a revolutionary new platfrom*.

Windows phone tercipta hasil kerja sama antara CEO *Microsoft* Steve Ballmer dengan CEO *Nokia* Stephen Elop yang dimana sistem operasi ini diterapkan pada ponsel genggam dengan merk *Nokia*. Windows phone dihadirkan dengan antarmuka pengguna yang tampak lebih bersih dan segar. Filosofi desain dari sistem informasi ini sendiri dinamakan *metro style* sehingga menunjukan ciri yang jelas, informasi yang mudah diperoleh dan menggunakan simbol-simbol yang mudah dipahami. [4]

II.4 sAlgoritma *Joint Photograpic Experts*

Joint Photograpic Experts (JPEG, dibaca jay-peg) di rancang untuk kompresi beberapa *full-color* atau *gray-scale* dari suatu gambar yang asli, seperti pemandangan asli di dunia ini. JPEGs bekerja dengan baik pada continous tone images seperi *photographs* atau semua pekerjaan seni yang menginginkan yang nyata; tetapi tidak terlalu bagus pada ketajaman gambar dan seni pewarnaan seperti

penulisan, kartun yang sederhana atau gambar yang mengunakan banyak garis. JPEG sudah mendukung untuk 24-bit color *depth* atau sama dengan 16,7 juta warna (224 = 16.777.216 warna). *Progressive* JPEGs (p-JPEGs) adalah tipe dari beberapa persen lebih kecil dibandingkan baseline JPEGs: tetapi keuntungan dari JPEG dan tipetipenya telihat pada langkah-langkahnya sama seperti *iinterlaced GIFs*.

JPEG adalah algoritma kompresi secara *lossy*. JPEG bekerja dengan merubah gambar spasial dan merepresentasikan ke dalam pemetaan frekuensi. *Discrete Cosine Transform* (DCT) dengan memisahkan antara informasi frekuensi yang rendah dan tinggi dari sebuah gambar. Informasi frekuensi yang tinggi akan diseleksi untuk dihilangkan yang terikat pada penganturan kualitas yang digunakan. Kompresi dengan tingkatan yang lebih baik, tingkatan yang lebih baik dari informasi yang dihilangkan. Waktu Kompresi dan dekompresi dilaksanakan dengan simetris. [5]

II.5 API

API (Application Programming Interface) adalah kumpulan fungsi-fungsi atau perintah-perintah untuk menggantikan bahasa yang digunakan dalam system call dengan bahasa yang lebih terstruktur dan mudah dimengerti.

Keuntungan memprogram dengan menggunakan API adalah:

1. Portabilitas. Programmer yang menggunakan API dapat menjalankan programnya dalam sistem operasi mana saja asalkan sudah terinstall API tersebut. Sedangkan system call berbeda antar sistem operasi, dengan catatan dalam implementasinya mungkin saja berbeda.
2. Lebih Mudah Dimengerti. API menggunakan bahasa yang lebih terstruktur dan mudah dimengerti daripada bahasa system call. Hal ini sangat penting dalam hal editing dan pengembangan.

System call interface ini berfungsi sebagai penghubung antara API dan system call yang dimengerti oleh sistem operasi. System call interface ini akan menerjemahkan perintah dalam API dan kemudian akan memanggil system calls yang diperlukan.

Untuk membuka suatu file tersebut user menggunakan program yang telah dibuat dengan menggunakan bantuan API, maka perintah dari user tersebut

diterjemahkan dulu oleh program menjadi perintah open(). Perintah open() ini merupakan perintah dari API dan bukan perintah yang langsung dimengerti oleh kernel sistem operasi. Oleh karena itu, agar keinginan user dapat dimengerti oleh sistem operasi, maka perintah open() tadi diterjemahkan ke dalam bentuk system call oleh system call interface. Implementasi perintah open() tadi bisa bermacam-macam tergantung dari sistem operasi yang kita gunakan. [6]

II.6 JSON

JSON (JavaScript Object Notation) adalah format data interchange-ringan. Sangat mudah dibaca dan ditulis oleh manusia. Sangat mudah untuk mesin untuk parse, dan generate. Hal ini didasarkan pada subset dari bahasa pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang benar-benar bahasa independen tetapi menggunakan konvensi yang akrab bagi programmer dari C, termasuk C , C ++, C #, Java, JavaScript, Perl, Python, dan banyak lainnya. Properti ini membuat JSON menjadi bahasa data interchange yang ideal.

JSON dibentuk oleh dua struktur, yaitu collection dari nama atau value pairs. Dalam berbagai bahasa, ini direalisasikan object, record, struct, dictionary, hash table, keyed list, atau associative array. Selain itu ordered list dari value. Dalam kebanyakan bahasa, ini direalisasikan sebagai array, vector, list, atau sequence. [7]

II.7 Cloud Computing Security

Cloud computing merupakan sebuah istilah untuk kegiatan menyelesaikan suatu proses atau perhitungan melalui media internet dengan pemanfaat segala sumber daya yang dimiliki oleh sekumpulan komputer yang saling terhubung pada suatu jaringan yang sama. Teknologi ini menggunakan internet dan server pusat yang biasanya tidak dalam satu tempat yang sama untuk menjaga atau mengelola data dan aplikasi. Dibalik itu semua aspek keamanan merupakan hal yang paling penting pada sebuah sistem di Cloud. Berikut merupakan aspek berbahaya yang dapat timbul dari penggunaan teknologi cloud computing antara lain (Setiawan, 2010) :

a. Disrupts Services

Gangguan terhadap layanan biasanya terjadi karena faktor alam. Cuacana yang kurang baik sehingga koneksi tidak dapat berjalan dengan semestinya atau terjadi bencana alam seperti gempa bumi dan sebagainya merupakan ancaman yang dapat membuat server penyedia layanan bermasalah dan tidak dapat berjalan.

b. Theft of Information

Pencurian data merupakan hal yang sangat rentan terjadi pada aplikasi yang menggunakan teknologi Cloud Computing. Hal ini dikarenakan data yang disimpan pada server yang berada di internet sehingga sangat rentan untuk terjadi penyadapan data.

c. Loss of Privacy

Terjadi kebocoran data pribadi milik pengguna adalah hal yang berbahaya dan rentan terjadi. Hal ini dikarenakan penyedia layanan dapat mengambil data pribadi milik pengguna sehingga data pribadi milik pengguna menjadi tidak terjamin lagi kerahasiaannya.

d. Damage information

Data yang rusak akibat jaringan yang kurang baik sehingga menghasilkan data yang corrupt juga merupakan hal yang berbahaya bagi pengguna teknologi Cloud. Hal tersebut akan mengganggu pengguna teknologi ini jika terjadi kerusakan data yang cukup banyak dan tidak memiliki Backup data.

Berdasarkan beberapa masalah yang telah dipaparkan sebelumnya maka dibutuhkan sebuah sistem keamanan yang mampu mengatasi permasalahan tersebut. Penelitian ini menggunakan Azure Cloud Service dimana setiap keamanan pada sistem aplikasi Solidare diamankan secara otomatis oleh pihak penyedia layanan. Beberapa fitur yang ditawari oleh Azure Cloud Service yang dapat mengamankan sistem *cloud* adalah sebagai berikut :

a. Layanan Infrastruktur

Windows Azure menyediakan infrastruktur dengan skala yang sesuai dengan kebutuhan. Baik dalam membuat aplikasi baru atau menjalankan aplikasi yang telah disediakan.

b. Aplikasi Mobile

Windows Azure memberikan kemudahan dalam pembuatan aplikasi mobile. Aplikasi yang telah dibuat dan dapat langsung dimasukan ke penyimpanan komputasi awan.

c. Aplikasi Web

Windows Azure menawarkan keamanan dan fleksibilitas pengembangan, penyebaran, dan pilihan skala untuk berbagai macam ukuran aplikasi web.

d. Penyimpanan, Pencadangan, dan Pemulihan

Windows Azure menyediakan penyimpanan, pencadangan, dan solusi pemulihan data apapun.

e. Identitas dan Manajemen Akses

Windows Azure Active Directory memberikan layanan pengamanan pada identitas perusahaan. Serta melakukan manajemen pada banyak pengguna di sebuah perusahaan.

f. Integrasi

Windows Azure memperbolehkan pengguna untuk membawa seluruh aplikasi, data, perangkat, mitra ke perangkat lokal dan ke awan.

g. Manajemen Data

Windows Azure menyediakan solusi yang tepat untuk kebutuhan data pengguna. [3]

II.8 UML

UML (*Unifield Modeling Language*) adalah sebuah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan dan membangun sistem perangkat lunak berbasis Objek (*Object Oritend*). UML mulsi diperkenalkan oleh Object Management Group, sebuah organisasi yang telah mengembangkan model teknologi dan standar OOP sejak tahun 1980an. Berikut empat macam diagram UML yang sering digunakan untuk pembangunan perangakat lunak berorientasi objek. [8]

1. *Use case diagram*

Use case diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case diagram* terdiri atas diagram untuk *use case* dan a*ctor*. *Acto*r merepresentasikan orang yang akan mengoperasikan atau orang

yang berinteraksi dengan sistem aplikasi.Use case merepresentasikan operasi-operasi yang dilakukan oleh actor. *Use case* digambarkan berbentuk elips dengan nama operasi dituliskan didalamnnya. Actor yang melakukan operasi dihubungkan dengan garis lurus ke use case.

2. *Sequence Diagram*

Sequence diagram menjelaskan secara detail urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari *use case*: interaksi yang terjadi antar *class*, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi

3. *Collaboration Diagram*

Collaboration diagram dipakai untuk memodelkan interaksi antar objek di dalam sistem. Berbeda dengan *sequence diagram* yang lebih menonjolkan kronologis dari operasi-operasi yang dilakukan, *collaboration diagram* lebih fokus pada pemahaman atas keseluruhan operasi yang dilakukan oleh objek.

4. *Class Diagram*

Class diagram merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. *Class* diagram menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan. [8]