

## **JAWABAN UJIAN AKHIR SEMESTER**

**Mata Kuliah : Cloud Computing**

**Nama : Gregorius Volansius Ragut**

**Semester/Kls : Ganjil/Q**

**NIM : 21101199**

**Waktu/ Sifat : Offline**

**Prog.Studi : Teknik Informasi**

---

1. Elastisitas adalah salah satu karakteristik utama dalam cloud computing karena memungkinkan perluasan atau pengurangan sumber daya komputer dengan cepat guna memenuhi permintaan yang terus berubah tanpa perlu investasi pembelian atau pemeliharaan sumber daya.

penting dalam pengelolaan sumber daya karena memungkinkan perusahaan untuk menghindari pembayaran untuk kapasitas yang tidak terpakai, serta untuk menyesuaikan jumlah sumber daya yang dialokasikan dengan jumlah sumber daya yang dibutuhkan. Dengan elastisitas, perusahaan dapat dengan mudah menyesuaikan kapasitas komputasi mereka, mendukung pertumbuhan, dan menghindari gangguan layanan.

2. Perbandingan antara Infrastructure as a Service (IaaS) dan Software as a Service (SaaS) adalah penting dalam memilih jenis layanan yang sesuai dengan kebutuhan bisnis dan organisasi.

Berikut adalah perbedaan utama antara keduanya:

- 1) Infrastructure as a Service (IaaS):

- IaaS merupakan layanan infrastruktur komputasi awan yang memungkinkan pengguna untuk mengirimkan file dan mengakses sumber daya komputer melalui konektivitas berbasis IP sebagai bagian dari layanan on-demand.
- Klien tidak perlu membeli perangkat lunak atau server, dan mereka bisa mengendalikan sumber daya komputasi secara independent.
- Contoh IaaS populer adalah Amazon EC2 dan Azure.

- 2) Software as a Service (SaaS):

- SaaS merupakan layanan penyedia software berbasis cloud yang disediakan kepada pengguna secara online
- Konsumen tidak perlu membeli atau menginstal perangkat lunak, karena semua aplikasi dan data disimpan dan dioperasikan oleh penyedia layanan di server eksternal mereka.
- SaaS cocok untuk organisasi yang memiliki kebutuhan aplikasi yang khusus dan ingin mengurangi biaya dan waktu penginstalan perangkat lunak.
- Contoh SaaS populer adalah Google Docs dan Salesforce.com

3. Konsep containerization menggunakan Docker

Docker memungkinkan membuat, menguji, dan menerapkan aplikasi dengan cepat dan efisien.

Docker mengemas perangkat lunak ke dalam unit standar yang disebut kontainer yang memiliki semua alat sistem, kode, dan waktu proses.

Hal ini memungkinkan untuk dengan cepat menerapkan dan menskalakan aplikasi ke lingkungan apa pun dan yakin bahwa mereka akan berjalan dengan baik dan menghemat uang dengan meningkatkan pemanfaatan sumber daya.

Kebiasaan kontainer dapat mempermudah proses pengembangan dan pengelolaan aplikasi karena beberapa alasan berikut:

- Isolasi: Kontainer memisahkan aplikasi dan sistem dari lingkungan host, menjadi lebih aman untuk mengembangkan dan mengelola aplikasi secara independent.
- Menghemat waktu penginstalan:  
Dengan menggunakan kontainer, dapat menginstal perangkat lunak dan mengatur lingkungan pengembangan dengan cepat dan efisien.
- Pemulihan dan pengujian:  
Kontainer memungkinkan pengembang untuk mengujicobak aplikasi dengan cepat dan efisien, menjadi lebih mudah untuk mengidentifikasi dan mengatasi masalah.
- Kolaborasi: Docker memungkinkan pengembang untuk bekerja sama dengan kolaborasi yang lebih baik, baik melalui Docker Compose maupun dengan ekosistem alat yang kuat

4. Redundansi dan ketersediaan tinggi menjadi faktor krusial dalam lingkungan cloud computing karena mereka memastikan kelangsungan operasional dan ketersediaan layanan. Redundansi, seperti penyimpanan data ganda di lokasi yang berbeda, memungkinkan sistem tetap beroperasi saat terjadi kegagalan. Tantangan dalam mencapai tingkat ketersediaan yang diinginkan termasuk manajemen kompleksitas infrastruktur yang redundant dan biaya tambahan yang terkait dengan menjaga tingkat ketersediaan yang tinggi oleh karena itu, sambil penting untuk memastikan ketersediaan tinggi, perlu diakui bahwa mencapai tingkat ketersediaan yang diinginkan dapat melibatkan investasi tambahan dan manajemen yang cermat.

5. Dalam pandangan saya, keamanan antara cloud public dan private memiliki perbedaan yang signifikan. Cloud private menawarkan tingkat kontrol dan privasi yang lebih tinggi karena dioperasikan secara khusus untuk satu organisasi dan berada di belakang firewall. Di sisi lain, cloud public dikelola oleh pihak ketiga dan digunakan oleh beberapa perusahaan, sehingga pengguna memiliki kendali terbatas atas lingkungan mereka.

Organisasi seharusnya memilih cloud public ketika mereka membutuhkan skalabilitas dan elastisitas yang tinggi, serta ketika mereka ingin menghindari biaya modal yang tinggi karena model biaya berlangganan yang fleksibel.

Di sisi lain, cloud private lebih cocok digunakan ketika organisasi membutuhkan tingkat kontrol dan keamanan yang tinggi atas data sensitif dan aplikasi kritis mereka.

Dengan demikian, pemilihan antara cloud public dan private bergantung pada kebutuhan spesifik dan tingkat kontrol serta keamanan yang diinginkan oleh organisasi tersebut.

6. Docker adalah alat kontenerisasi yang memungkinkan pengembang untuk mengemas aplikasi dan dependensi mereka ke dalam satu paket yang dapat dijalankan di berbagai lingkungan, seperti komputer pengembangan, server produksi, atau lingkungan cloud.

Berikut adalah beberapa kelebihan yang signifikan dalam menggunakan Docker dan kontainer dalam konteks pengembangan dan implementasi aplikasi:

- **Portabilitas:**  
Kontainer dapat dijalankan di berbagai platform dan lingkungan, sehingga memudahkan pengembangan dan implementasi aplikasi.
- **Isolasi:**  
Kontainer menyediakan isolasi yang kuat antara aplikasi yang berjalan di dalamnya, sehingga jika satu container mengalami masalah atau kegagalan, itu tidak akan memengaruhi container lainnya
- **Efisiensi Sumber Daya:**  
Kontainer sangat efisien dalam penggunaan sumber daya, karena mereka memerlukan lebih sedikit sumber daya dibandingkan dengan virtual machine (VM) tradisional
- **Pengembangan dan Penyebaran:**  
Docker memungkinkan pengembang untuk mengemas aplikasi beserta dependensinya ke dalam satu paket yang dapat dikelola dengan mudah. Hal ini mempercepat siklus pengembangan dan memudahkan proses penyebaran aplikasi
- **Integrasi dengan Teknologi Modern:**  
Kontainer dapat diintegrasikan dengan teknologi modern seperti Kubernetes, yang membantu mengelola kluster kontainer di beberapa host.

Dalam konteks pengembangan dan implementasi aplikasi, Docker dan kontainer memberikan banyak keuntungan, seperti portabilitas yang tinggi, isolasi yang kuat, efisiensi sumber daya, dan pengembangan yang lebih cepat. Oleh karena itu, penggunaan Docker dan kontainer menjadi solusi yang populer untuk mengelola dependensi dan memastikan konsistensi aplikasi di berbagai lingkungan.

7. **Skalabilitas horizontal dalam arsitektur cloud** memungkinkan penambahan atau pengurangan instance virtual machine atau container sesuai dengan kebutuhan. Dalam konteks pengembangan dan implementasi aplikasi, skalabilitas horizontal memungkinkan sistem untuk menangani lonjakan lalu lintas dengan menambahkan lebih banyak instans, sehingga memastikan ketersediaan layanan dan kelangsungan operasional. Keuntungan utama dari skalabilitas horizontal adalah kemampuan untuk meningkatkan kapasitas sistem dengan cepat dan efisien, serta menghindari biaya modal yang tinggi.

Dalam hal perbedaan antara virtualisasi dan kontainerisasi, virtualisasi memungkinkan beberapa sistem operasi dan aplikasi berjalan pada satu mesin fisik, sedangkan kontainerisasi memungkinkan beberapa aplikasi berjalan pada satu sistem operasi.

Kontainerisasi lebih ringan dan efisien dalam penggunaan sumber daya, sehingga memungkinkan pengembang untuk mengemas aplikasi dan dependensi mereka ke dalam satu paket yang dapat dijalankan di berbagai lingkungan.

Oleh karena itu, kontainerisasi dan Docker memberikan manfaat signifikan dalam pengembangan dan implementasi aplikasi, seperti portabilitas yang tinggi, isolasi yang kuat, efisiensi sumber daya, dan pengembangan yang lebih cepat.

8. SaaS (Software as a Service) dan FaaS (Function as a Service) adalah dua model layanan cloud computing yang berbeda. SaaS menyediakan aplikasi yang dapat diakses melalui internet, sedangkan FaaS menyediakan lingkungan untuk menjalankan kode fungsional tanpa harus mengelola infrastruktur.

Kapan lebih masuk akal menggunakan FaaS daripada SaaS atau sebaliknya tergantung pada kebutuhan spesifik organisasi. Jika organisasi membutuhkan aplikasi yang dapat diakses melalui internet, seperti aplikasi email atau manajemen proyek, maka SaaS mungkin lebih cocok. Namun, jika organisasi membutuhkan lingkungan untuk menjalankan kode fungsional, seperti pengolahan data atau analisis, maka FaaS mungkin lebih cocok.

Dalam konteks pengembangan aplikasi, FaaS memberikan manfaat signifikan dalam hal skalabilitas dan efisiensi sumber daya. FaaS memungkinkan pengembang untuk menjalankan kode fungsional tanpa harus mengelola infrastruktur, sehingga memungkinkan pengembang untuk fokus pada pengembangan aplikasi dan meningkatkan efisiensi sumber daya.

Oleh karena itu, pemilihan antara SaaS dan FaaS tergantung pada kebutuhan spesifik organisasi dan tujuan penggunaan layanan cloud computing.

9. Docker Hub adalah repositori SaaS yang memfasilitasi manajemen kontainer dengan menyediakan tempat untuk berbagi, menyimpan, dan mengelola gambar kontainer. Docker Hub memungkinkan pengguna untuk dengan mudah menemukan kontainer yang ada, berbagi kontainer mereka, dan mengotomatiskan alur kerja pengembangan perangkat lunak. Pengguna juga dapat mengakses gambar kontainer resmi dari perangkat lunak open source populer dan mengintegrasikan alur kerja Docker mereka dengan alat pengujian dan penyebaran yang ada. Dengan menggunakan repositori publik seperti Docker Hub, pengembang dapat memanfaatkan komunitas yang besar dan beragam untuk berbagi dan berkolaborasi dalam pengembangan perangkat lunak berbasis kontainer. Oleh karena itu, ada keuntungan signifikan dalam menggunakan repositori publik seperti Docker Hub, termasuk akses ke gambar kontainer yang ada, kolaborasi dengan komunitas, dan integrasi dengan alat pengujian dan penyebaran yang ada.
10. Beberapa langkah konkret yang dapat diambil untuk meningkatkan keamanan data dalam lingkungan cloud computing meliputi:
  - Enkripsi Data: Melakukan enkripsi data saat istirahat dan saat disimpan di cloud untuk melindungi kerahasiaan informasi sensitive.
  - Manajemen Akses: Memastikan bahwa hanya pengguna yang sah yang memiliki akses ke data yang tepat, dan menggunakan otentikasi multi-faktor untuk lapisan keamanan tambahan.
  - Pemantauan dan Audit: Melakukan pemantauan terus-menerus terhadap aktivitas data dan melakukan audit secara berkala untuk mendeteksi dan mencegah ancaman keamanan.
  - Keamanan Jaringan: Menggunakan firewall, deteksi intrusi, dan teknologi keamanan jaringan lainnya untuk melindungi data dari serangan luar.