

BAB 2

LANDASAN TEORI

2.1 PT Divistant Teknologi Indonesia

Divistant adalah penyedia terkemuka solusi *cloud* dan *DevOps*, yang mengkhususkan diri dalam membantu bisnis mempercepat inovasi, menyederhanakan operasi, dan mencapai transformasi digital yang *seamless*. Layanan Divistant secara *end-to-end* memberdayakan perusahaan untuk memanfaatkan potensi penuh teknologi *cloud* modern dan metodologi *DevOps*, memungkinkan untuk melacak dengan efisien, meningkatkan fleksibilitas, dan menjaga keunggulan kompetitif dalam lanskap digital yang terus berkembang saat ini. Adapun *core value* dari Divistant adalah sebagai berikut.

1. *Curiosity, drives our growth.*
2. *Impact, creates our happiness.*
3. *Respect, makes us outstanding.*
4. *Courage, is our strength.*
5. *Commitment, defines our success.*
6. *Adaptability, fuels our innovation*

2.1.1 Logo PT Divistant Teknologi Indonesia

Simbol atau logo adalah representasi visual dari identitas merek dalam bentuk ideogram, simbol, lambang, ikon, atau tanda. Komponen fisik utama dari sebuah merek adalah logonya. Logo yang baik harus dapat menyampaikan semua kualitas tidak berwujud yang membentuk esensi perusahaan (visi, misi, nilai, budaya) [11]. Logo PT Divistant Teknologi Indonesia dapat dilihat pada Gambar 2.1.



Gambar 2.1 Logo PT Divistant Teknologi Indonesia

2.1.2 Visi dan Misi PT Divistant Teknologi Indonesia

Visi merupakan gambaran atau cita-cita yang ingin dicapai oleh sebuah organisasi di masa yang akan datang. Visi dihadirkan untuk bisa menghadirkan motivasi dan inspirasi pada anggota organisasi untuk mencapai tujuan yang diinginkan dalam jangka panjang. Sedangkan Misi merupakan tujuan spesifik yang ingin dicapai bersifat pernyataan pada sebuah organisasi. Berbeda dengan visi, misi bersifat lebih spesifik dan dihadirkan untuk menggambarkan sasaran, keunggulan, alasan adanya organisasi, dan target yang akan dicapai.

Adapun visi dari PT Divistant Teknologi Indonesia adalah “Berkembang secara konsisten untuk menjadi *leading performer* dengan produk dan solusi yang berkualitas tinggi di pasar global yang kompetitif.” Adapun misi dari PT Divistant Teknologi Indonesia adalah berikut:

1. Merancang produk teknologi masa depan untuk memberdayakan transformasi digital.
2. Membangun ekosistem digital untuk mendukung percepatan transformasi digital secara *end to end*.

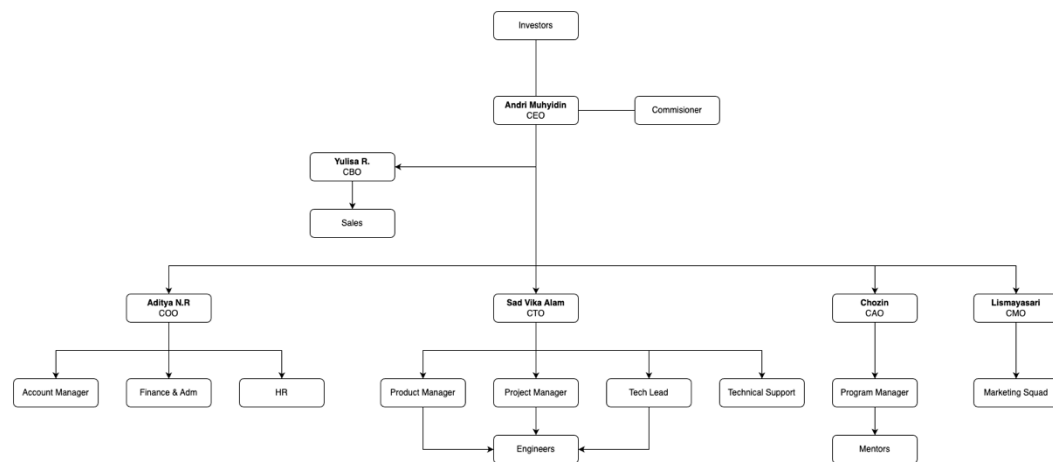
2.1.3 Struktur Organisasi PT Divistant Teknologi Indonesia

Struktur merupakan suatu susunan yang sistematis dan terorganisir yang mana struktur dapat merujuk pada berbagai salah satu di antaranya struktur organisasi. Hadirnya struktur yang baik bisa menghasilkan efisiensi, kemudahan koordinasi, hingga mencapai tujuan. Organisasi merupakan kumpulan orang yang terdiri dari dua orang atau lebih di mana orang-orang tersebut bekerja secara Bersama untuk mencapai tujuan tertentu. Pada umumnya organisasi memiliki struktur yang jelas di mana struktur tersebut sering kali dibagi menjadi beberapa divisi yang memiliki fungsi tersendiri.

Struktur organisasi adalah padanan kata yang terdiri dari struktur dan organisasi di mana telah disebutkan sebelumnya padanan dua kata tersebut memiliki arti sebagai suatu kerangka atau susunan yang sistematis dalam mewujudkan pola tetap pada hubungan bidang kerja maupun orang-orang yang

terlibat di dalamnya. Struktur organisasi memiliki fungsi untuk melakukan *monitoring* terhadap elemen kekuasaan, pekerjaan, tanggung jawab dan orang-orang yang berhubungan sehingga dapat diidentifikasi kedudukannya, tugasnya, tanggung jawabnya, kewajibannya, dan fungsinya.

Dalam menjalankan kegiatannya untuk melakukan bisnis, PT Divistant Teknologi Indonesia memiliki struktur organisasi yang memiliki tugas dan kewajiban tersendiri pada setiap divisi yang dijalankan. Berikut adalah gambar struktur organisasi PT Divistant Teknologi Indonesia.



Gambar 2.2 Struktur Organisasi PT Divistant Teknologi Indonesia

2.2 CRM (*Customer Relationship Management*)

CRM merupakan salah satu sarana untuk menjalin hubungan yang berkelanjutan antara perusahaan dengan para *stakeholder* maupun *shareholder*-nya. Saat ini banyak perusahaan yang memanfaatkan CRM untuk menjalin hubungan erat dengan pelanggan, dengan memanfaatkan CRM, perusahaan akan mengetahui apa yang diharapkan dan diperlukan pelanggannya sehingga akan tercipta ikatan emosional yang mampu menciptakan hubungan bisnis yang erat dan terbuka serta komunikasi dua arah atau timbal balik di antara mereka, dengan demikian kesetiaan pelanggan dapat dipertahankan dan tidak mudah berpindah ke lain produk dan merek, terlebih produk dan merek perusahaan pesaing [1].

CRM memiliki keuntungan dan manfaat [1]. Adapun keuntungan dan manfaat CRM adalah sebagai berikut.

1. Meningkatkan Loyalitas Pelanggan

Dengan menggunakan CRM memungkinkan perusahaan untuk mendapatkan informasi secara keseluruhan dari berbagai sumber yang berhubungan langsung dengan pelanggan, baik melalui *call center*, *website*, staf pemasaran dan pelayanan di lapangan.

2. Mengurangi Biaya

Dengan menggunakan CRM memungkinkan setiap kegiatan penjualan dan pelayanan menjadi lebih murah karena pemasarannya lebih terfokus pada pelanggan yang tepat serta waktu yang tepat pula.

3. Kegiatan Operasional Menjadi Lebih Efisien

Ketika proses penjualan dan pelayanan menjadi semakin mudah maka hal ini dapat mengurangi dari risiko penurunan kualitas pelayanan ke pelanggan dan beban arus kas. Misalnya saja dengan pemakaian teknologi *call center* serta *website* hal ini dapat mengurangi hambatan proses administratif, biaya, maupun birokrasi yang mungkin saja suatu saat bisa muncul.

4. *Time to Market* Semakin Meningkat

Dengan CRM, perusahaan akan mendapatkan informasi dari pelanggan berupa data tentang produk apa saja yang menjadi *trend* pembelian saat ini. Hal ini akan mempercepat produk masuk ke pasar.

5. Meningkatnya Pendapatan

Kumpulan informasi yang diperoleh dari aplikasi CRM ini akan memudahkan perusahaan dalam meningkatkan pelayanan dan keuntungan perusahaan.

2.3 Cloud Computing

Cloud Computing adalah Sebuah model *client server* dengan sumber daya seperti server, *storage*, *network*, dan *software* dapat dipandang sebagai layanan yang dapat di akses oleh pengguna secara *remote* dan virtual melalui internet di mana penggunaan internet dan konektivitas memudahkan dalam penggunaan sebuah layanan meliputi interaksi satu *library* dengan *library* lainnya [12]. Di samping hal tersebut *cloud computing* merupakan gaya baru komputasi untuk skala dinamis dan sumber daya secara virtualisasi maka daripada itu *cloud computing* merupakan paradigma baru mengenai komputasi dengan teknologi yang dapat

mengelola dan mengalokasikan anggaran secara efektif dan efisien dengan pengurangan biaya komputasi serta mempercepat skalabilitas [13].

2.3.1 Karakteristik Cloud Computing

Selain daripada manfaat yang telah disebutkan sebelumnya, *cloud computing* juga mempunyai beberapa karakteristik penting yang mana karakteristik tersebut meliputi *On Demand*, *Broad Network*, *Resource Pooling*, *Rapid Elasticity*, *Measured Service* dan *Scalable* [4].

1. *On Demand*, dimana pengguna dapat memesan mengelola layanan tanpa interaksi langsung dengan penyedia layanan.
2. *Broad Network*, kemampuan ketersediaan melalui jaringan dengan mengenalkan pengguna sebagai platform.
3. *Resource Pooling*, melakukan perpaduan berdasarkan sumber komputasi pengguna untuk dapat melayani konsumen.
4. *Rapid Elasticity*, memiliki kemampuan cepat dan elastisitas.
5. *Measured Service*, melakukan pengawasan secara otomatis dan mengoptimalkan sumber daya yang digunakan berdasarkan pengukuran yang sesuai.
6. *Scalable*, melakukan penyesuaian kapasitas maupun *traffic*.

2.3.2 Layanan Cloud Computing

Cloud computing menyediakan berbagai layanan di mana layanan tersebut dapat digunakan oleh pengguna berdasarkan kebutuhan dari pengguna terhadap sistem yang akan dibangun layanan tersebut meliputi :

1. *IaaS (Infrastructure as a Service)*

IaaS (Infrastructure as a Service) yaitu layanan *cloud computing* dengan sistem operasi dijadikan menjadi virtual, di mana pengguna dapat melakukan komputasi hingga pada level sistem operasi. Dalam kata lain layanan ini memberikan kemampuan kepada konsumen atau pengguna untuk memproses, menyimpan, berjaringan, dan komputasi sumberdaya lain yang penting, di mana konsumen dapat menyebarkan dan menjalankan perangkat lunak secara bebas secara virtual [4].

2. *PaaS (Platform as a Service)*

PaaS (Platform as a Services) yaitu memberikan layanan *cloud computing* dengan layanan pada level aplikasi seperti penyediaan *platform database*, penyediaan web server hingga aplikasi lainnya. Dalam kata lain layanan ini memberikan kemampuan kepada konsumen atau pengguna untuk menyebarkan aplikasi yang dibuat konsumen atau diperoleh ke infrastruktur komputasi awan menggunakan bahasa pemrograman dan peralatan yang didukung oleh *provider* dengan kata lain konsumen tidak mengelola atau mengendalikan infrastruktur *cloud* yang akan digunakan [4].

3. *SaaS (Software as a Service)*

SaaS (Software as a Service) yaitu layanan *cloud computing* dengan layanan berbentuk aplikasi yang sudah jadi kepada para developer. Dalam kata lain layanan ini memberikan kemampuan kepada konsumen atau pengguna untuk menggunakan aplikasi penyedia dapat beroperasi pada infrastruktur awan. Dalam kata lain konsumen tidak mengelola atau mengendalikan infrastruktur *cloud* yang mendasari [4].

Berbagai jenis layanan *cloud computing* dapat diakses melalui beberapa penyedia layanan *cloud computing* beberapa diantaranya adalah Amazon Web Service, Microsoft Azure, Google Cloud Platform, dan Alibaba Cloud.

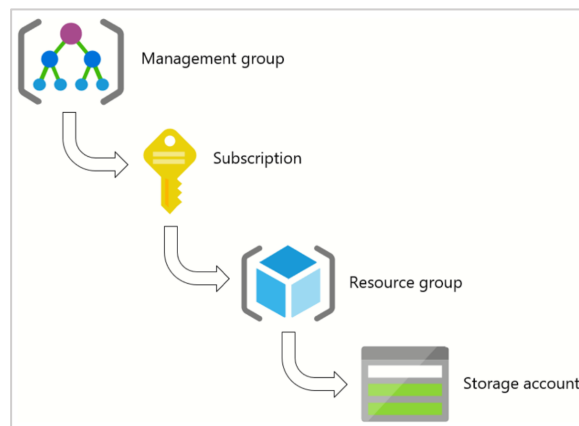
2.4 Microsoft Azure

Microsoft Azure adalah salah satu penyedia layanan *cloud computing* terkemuka. Memiliki sejumlah layanan *cloud* untuk mengelola, membangun, dan mendeploy aplikasi di jaringan global di mana dan siapa pun dapat mengaksesnya kapan saja. Di portal Microsoft Azure, terdapat tab-tab seperti dashboard, unggah, unduh, edit, bagikan, hapus, dan klon yang membantu pengguna menavigasi berbagai layanan yang disediakan oleh Microsoft Azure dengan mudah. Microsoft Azure terdiri dari berbagai fitur seperti keamanan, integrasi, analitik, dan machine learning, DevOps, platform yang konsisten, skalabilitas dan fleksibilitas. Berkat siklus pengembangan keamanan Microsoft Azure, platform ini menjadi lebih kuat dalam hal keamanan data dan kerja. Microsoft Azure memungkinkan pembayaran

berbasis penggunaan (*pay-as-you-go*). Juga mendukung integrasi .NET dan manajemen API, pengembangan yang mudah, layanan kontainer, dan virtualisasi [14].

2.4.1 Azure Management Scope

Dalam sistem *file* komputer, terdapat sebuah konsep yang dinamakan hierarki (*volume*, *folder*, *subfolder*, dan *file*). Izin yang ditetapkan pada cakupan yang lebih tinggi mengalir berdasarkan warisan ke cakupan yang lebih rendah. Contohnya memberikan akses *read-only* kepada pengguna pada drive E, mengakibatkan pengguna tersebut mewarisi akses *read-only* ke semua *folder*, *subfolder*, dan *file* yang ada pada drive E tersebut [15].



Gambar 2.3 Azure Management Scope

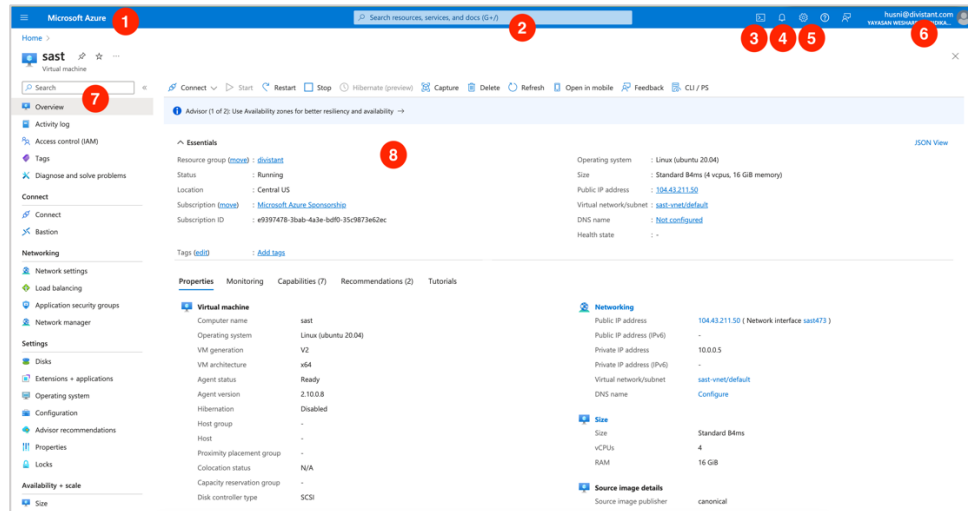
Scope inheritance dan *multilayer* bekerja dengan cara yang hampir sama pada Azure. Semua *resource group* terdapat dalam *azure subscription*, dan kita dapat menggabungkan satu atau beberapa *subscription* ke dalam sebuah *management group*. Mengacu pada Gambar 2.3, bila *user* atau *service account* diberikan hak akses pada level *subscription*, maka ia dapat mengelola setiap *resource* yang ada pada semua *resource group* yang ada pada *subscription* tersebut.

2.4.2 Azure Management Tool

Beberapa *tools* yang dapat digunakan untuk mengatur Azure [15] adalah sebagai berikut.

1. Azure Portal

Azure portal adalah aplikasi web responsif yang dapat digunakan untuk melakukan administrasi grafis pada Azure. Portal Azure dapat diakses pada alamat <https://portal.azure.com>. Tampilan Portal Azure dapat dilihat pada Gambar 2.4.



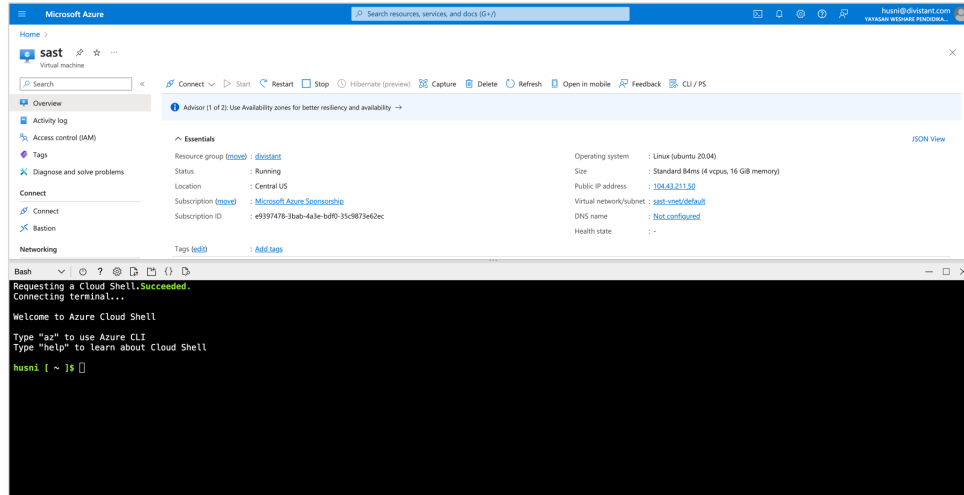
Gambar 2.4 Portal Azure

Berdasarkan Gambar 2.4, berikut adalah penjelasan setiap komponen pada Portal Azure.

- 1) *Page header*, adalah *top navigation bar* yang sering disebut *global navigation* karena terdapat di semua halaman portal azure.
- 2) *Global search*, dapat mencari semua *resource* pada Azure. Hasil pencarian juga termasuk tautan dokumentasi.
- 3) *Cloud shell*, untuk membuka *Azure Cloud Shell* untuk mengeksekusi *command-line*.
- 4) *Notifications*, untuk melihat proses *deployment* ataupun *deployment* yang telah lalu.
- 5) *Global subscription filter*, menampilkan *Azure subscription* untuk membuat portal lebih mudah untuk dijelajahi.
- 6) *Your account*, untuk melakukan *log out*, berpindah *directory*, dan mengubah profil akun.
- 7) *Resources pane*, daftar *resource* yang sedang dikonfigurasi.
- 8) *Configuration blade*, layar detail setiap komponen pada *resource* azure.

2. Azure CLI dan Azure Cloud Shell

Azure command-line interface (CLI) adalah *cross-platform command-line* untuk Azure. Azure CLI dapat dijalankan di dalam browser yakni di dalam Portal Azure ataupun dapat digunakan pada desktop terminal dengan cara melakukan install Azure *CLI* pada terminal desktop. Tampilan Azure CLI pada Azure *Cloud Shell* dapat dilihat pada Gambar 2.5.



Gambar 2.5 Azure Cloud Shell

3. Azure SDK

Bagi software developer, sebuah *software development kit* (SDK) merepresentasikan *API abstraction layer*. Azure SDK menyediakan *template* proyek dan *library code* yang dibutuhkan untuk berinteraksi dengan Azure. Azure SDK tersedia untuk berbagai bahasa pemrograman seperti go, java, .NET, node.js, PHP, python, ruby dan swift ataupun frameworks seperti android dan xamarin.

2.4.3 Azure Virtual Machine

Sebuah VM adalah representasi perangkat lunak dari sebuah komputer. Setiap wilayah Azure terdiri dari beberapa pusat data; setiap pusat data terdiri dari ribuan perangkat keras server *blade*. VM berjalan dalam struktur komputasi masif ini. Pada tingkat abstraksi yang lebih tinggi, sebuah mesin virtual Azure adalah sumber daya *cloud* yang terdiri dari tiga subsistem berikut [15].

1. Compute

Anda dikenakan biaya untuk setiap menit yang dialokasikan dan berjalan pada *virtual machine* Azure. Dialokasikan berarti Microsoft menyediakan

daya komputasi perangkat keras — terutama sumber daya CPU dan RAM — untuk *virtual machine*.

2. *Storage*

Azure menggunakan fitur yang disebut *Managed Disk* untuk menyimpan VHD OS dan data Anda. Dengan *Managed Disk*, Anda tidak perlu mengelola infrastruktur penyimpanan. Level penyimpanan mana yang cocok untuk beban kerja *virtual machine* tergantung pada kebutuhan akan kecepatan dan ketersediaan yang tinggi [15].

3. *Network*

Konfigurasi jaringan *virtual machine* independen dari sumber daya *virtual machine* itu sendiri. Jaringan pada *virtual machine* terdiri dari komponen-komponen berikut [15]:

- a. Sumber daya *virtual machine*
- b. Sumber daya antarmuka jaringan virtual
- c. Sumber daya alamat IP publik (opsional)
- d. Sumber daya grup keamanan jaringan (opsional)

2.4.4 Azure Storage

Storage adalah contoh layanan universal di Microsoft Azure. Terlepas dari jenis beban kerja yang dijalankan (*virtual machine*, layanan aplikasi, *function*, *machine learning*, atau apapun itu) diperlukan penyimpanan objek yang persisten.

Azure storage mewakili sumber daya yang menyediakan penyimpanan yang sangat tersedia dan persisten untuk tiga tipe data utama: *unstructured*, *semisctructured*, dan *structured* [15].

1. *Unstructured data*

Unstructured data adalah data yang tidak memiliki skema yang mengharuskannya. Contohnya objek biner (*blob*) seperti *file*, *hard drive* virtual, *file* dokumen, dan *file* media. Semua objek ini adalah objek penyimpanan file dengan format dan konten tertentu. Sebuah objek blob adalah file dengan ukuran apa pun, apakah itu *file* teks 1K atau *file hard disk virtual* 120GB.

2. *Semisstructured data*

Semisstructured data tidak memiliki susunan kolom-baris seperti data relasional, tetapi tidak sebebaskan data biner murni yang tidak terstruktur. Layanan tabel dalam Azure bersifat setengah terstruktur dalam bentuk pasangan data kunci/nilai. *Database NoSQL* adalah contoh data *semistructured*. Cosmos DB adalah produk *database NoSQL* utama dari Azure.

3. *Structured data*

Jika Anda pernah bekerja dengan sistem basis data relasional mana pun, Anda memahami data terstruktur — data yang diuraikan menjadi satu atau lebih tabel di mana tipe data tertentu mengikat setiap kolom. Contoh basis data terstruktur dalam ekosistem Azure termasuk *Azure SQL Database* dan *Azure Database for MySQL Servers*.

2.5 Web Hosting

Web hosting adalah jasa penyewaan tempat penyimpanan data di internet yang diperlukan oleh sebuah *website*. *Web hosting* adalah salah satu syarat mutlak agar *website* bisa *online* dan dapat diakses internet dari seluruh dunia. Apabila *website* diibaratkan sebagai rumah, maka *web hosting* adalah lahan untuk membangun rumah tersebut. Secara fisik, *web hosting* berupa komputer dan perangkat-perangkatnya yang juga dapat berfungsi sebagai *server* internet [16].

Berdasarkan layanan yang diberikan, *web hosting* dibagi menjadi beberapa tipe, yaitu *shared hosting*, *VPS hosting*, *dedicated server hosting* [16].

1. *Shared hosting* adalah tipe *web hosting* yang menggunakan komputer *server* yang sama untuk beberapa pengguna *hosting* sekaligus. Keunggulan dari *shared hosting* adalah harganya yang relatif murah. Namun apabila terjadi gangguan pada salah satu pengguna *web hosting* akan berakibat pada semua pengguna *web hosting*. Walaupun demikian *shared hosting* adalah tipe *web hosting* yang paling populer. Perusahaan jasa penyedia *web hosting* banyak menawarkan paket yang menarik untuk tipe *web hosting* ini.

2. *VPS (Virtual Private Server) hosting* adalah tipe *web hosting* yang menggunakan *virtual machine*. Setiap pengguna *web hosting* seolah-olah mempunyai komputer sendiri. Gangguan pada salah satu pengguna *hosting* tidak akan mengganggu pengguna yang lain. *VPS hosting* lebih baik apabila dibandingkan dengan *shared hosting*. Konsekuensinya harga *VPS hosting* lebih mahal daripada *shared hosting*.
3. *Dedicated server hosting* adalah tipe *web hosting* yang menggunakan satu komputer *server* untuk satu pengguna *hosting*. Pengguna *dedicated server* biasanya perusahaan-perusahaan besar yang membutuhkan sumber daya *web hosting* yang besar.

2.6 Linux

GNU/Linux adalah sistem operasi yang dibuat oleh Linus Benedict Torvalds dan disebarakan secara bebas di internet di mana orang lain bisa mengembangkan dan menggunakan untuk keperluannya sendiri. Namun, perlu dijelaskan bahwa GNU/Linux di sini bisa bermakna ganda. Pertama, GNU/Linux berarti *kernel* Linux. Pengertian kedua berarti sebuah sistem yang di dalamnya sudah terdapat *kernel*, *shell*, dan program pendukung lain yang siap didistribusikan dan dipakai. GNU/Linux dalam penelitian ini mengacu pada pengertian yang kedua. GNU/Linux adalah sistem operasi yang bebas dipakai, didistribusikan, dan dikembangkan kembali. Oleh karena itu, GNU/Linux mempunyai banyak varian yang lebih dikenal dengan istilah distro [17].

2.6.1 Ubuntu

Meskipun sudah ada banyak sistem operasi berbasis Linux, tidak ada yang bisa digunakan dengan mudah oleh masyarakat umum. Mark Shuttleworth, seorang pengusaha asal Afrika Selatan, datang dengan ide Ubuntu Linux. Microsoft Windows terkenal sebagai sistem operasi yang mudah digunakan dan menjadi populer di kalangan pengguna komputer. Tujuan Mark Shuttleworth adalah mengembangkan sistem operasi berbasis Linux yang sederhana untuk semua kalangan masyarakat. Dia mendirikan sebuah perusahaan bernama Canonical. Awalnya, sistem operasi Ubuntu dimulai sebagai proyek oleh perusahaan

Canonical. Sistem operasi Ubuntu adalah proyek perangkat lunak yang hampir gratis. Ubuntu telah mendapatkan popularitas besar karena kemudahannya dibandingkan dengan sistem operasi berbasis Linux lainnya. Keterlibatan komunitas ada di semua tahap pengembangan Ubuntu. Istilah Ubuntu dipinjam dari pemikiran filosofis Afrika. Gagasan dasar dari Ubuntu adalah 'Kemanusiaan terhadap sesama'. Ubuntu Linux pertama kali dirilis pada 20 Oktober 2004. Salinan Ubuntu Linux tersedia di Internet untuk diunduh [18].

Ubuntu berbasis pada Debian Linux. Debian adalah proyek Linux besar dengan partisipasi aktif dari komunitas. Ian Murdock, seorang ahli komputer muda, memulai proyek Debian Linux pada tahun 1993. Debian adalah sistem operasi berbasis Linux yang populer di kalangan programmer dan administrator sistem. Ubuntu telah membuat beberapa perubahan pada Debian Linux agar lebih bermanfaat bagi masyarakat umum. Statistik menunjukkan bahwa Ubuntu adalah sistem operasi berbasis Linux yang paling populer [18].

2.7 Apache Web Server

Web server adalah *software* yang membuat *website* dapat diakses. Web server dapat melakukan itu dengan cara *listening* pada *port* 80 dan 443 dan menyediakan *file* pada *directory* tertentu sebagai *respon* atas *request* yang diterima pada *port* tersebut. Port 80 adalah *default port* untuk HTTP dan *port* 443 adalah untuk jenis yang terenkripsi (HTTPS). Salah satu web server yang paling banyak digunakan adalah Apache. Apache telah banyak digunakan dan seperti menjadi standar umum untuk waktu yang cukup lama. Seorang admin web server hampir tidak ada yang tidak pernah menggunakan Apache, apalagi untuk perusahaan yang sudah lama ada. Nama sebenarnya untuk web server ini adalah Apache HTTP server. HTTP server adalah salah satu dari lebih dari 200 proyek yang di-*manage* oleh Apache Software Foundation [19].

2.8 PHP

PHP adalah bahasa pemrograman *script server-side* yang didesain untuk pengembangan web. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum. PHP diciptakan oleh Rasmus Lerdorf pertama kali tahun

1994. Saat ini PHP adalah singkatan dari *PHP: Hypertext Preprocessor*, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri. PHP dapat digunakan dengan gratis (*free*) dan bersifat *open source*. PHP dirilis dalam lisensi *PHP License*, sedikit berbeda dengan lisensi *GNU General Public License (GPL)* yang biasa digunakan untuk proyek *open source* [20].

2.9 MYSQL

MySQL merupakan *tools Database Management System (DBMS) open source* yang mendukung *multiuser, multithreaded*, populer, dan *free*. SQL adalah bahasa permintaan *database* tertentu di mana sub bahasa dapat membuat dan memanipulasi data di dalam *database*. SQL digunakan untuk melakukan tugas-tugas seperti melakukan *update* terhadap *database*, yang merujuk pada konsep *Relational Database Management System (RDBMS)* [20].

2.10 cPanel

cPanel adalah perangkat lunak *control panel online* untuk melakukan pengaturan *website* dalam sebuah *web hosting*. cPanel adalah tool utama untuk pengguna *web hosting*. Beberapa fungsi utama cPanel antara lain: unggah data website, pengaturan data *website*, instalasi *content management system*, dan lain-lain. Data-data *website* dapat diunggah menggunakan fitur yang ada di cPanel. Kita dapat mengunggah data *website* dalam bentuk *archive* dan mengekstraknya menggunakan fitur yang ada di cPanel. Melalui cPanel juga kita bisa mengatur data-data *website* seperti menambah, menghapus atau memodifikasi data *website* [16].

2.11 Performance Test

Performance test dapat didefinisikan sebagai investigasi teknis yang dilakukan untuk menentukan atau memvalidasi kecepatan, skalabilitas, atau stabilitas dari produk yang sedang diuji. Aktivitas terkait performance test seperti *testing* dan *tuning*, memiliki fokus untuk mencapai response time, throughput, dan resource utilization pada titik di mana ia masuk ke dalam performance objective

untuk aplikasi yang sedang dites [21]. Performance test biasanya dilakukan untuk mencapai beberapa hal seperti:

- a) *Assesment* kesiapan *production*
- b) Mengevaluasi sistem pada kriteria *performance* tertentu
- c) Membandingkan karakteristik *performance* dari beberapa sistem
- d) Mencari sumber masalah pada *performance*
- e) Mendukung *system tuning*
- f) Mencari level *throughput*

Pada penelitian ini, *performance test* dilakukan seperti pada poin c, yakni untuk membandingkan *performance* dari beberapa sistem. Tes pada sistem berjalan dan tes pada sistem usulan. Terdapat berbagai *tools* untuk melakukan *performance test* dan salah satunya adalah k6.

2.11.1 K6

K6 adalah *tools* untuk melakukan *load testing* dan verifikasi performa. Dengan menggunakan k6, *developer* dan *engineer* dapat melakukan tes pada aplikasi web dan layanan *network* dan mengevaluasi *performance* dengan *load* yang berbeda. Beberapa keuntungan menggunakan k6 adalah kemudahan penggunaan *tool* untuk *load testing*, mendukung *scripting* javascript, dan menyediakan laporan yang mendetail [22].

2.11.2 Performance Metrics

Metrics digunakan untuk mengukur bagaimana suatu sistem bekerja di bawah kondisi pengujian. Berdasarkan tipenya, *metrics* terbagi ke dalam empat jenis [23].

1. *Counter*, adalah metrik yang menjumlahkan nilai-nilai yang ditambahkan secara kumulatif.
2. *Gauge*, adalah metrik yang menyimpan nilai *min* (minimum), *max* (maksimum), dan *last values* (nilai terakhir yang ditambahkan).
3. *Rate*, adalah metrik yang melacak persentase nilai-nilai yang ditambahkan yang tidak bernilai nol.

4. *Trend*, adalah metrik yang menghitung statistik dari nilai-nilai yang ditambahkan (*min*, *max*, *average*, and *percentiles*).

Pada k6 terdapat *metrics* bawaan dan juga *metrics* kustomisasi dapat dibuat jika diperlukan. Adapun *standard built-in metrics*, k6 akan selalu mengumpulkan *metrics* berikut, tanpa mengidahkan protokol apa yang digunakan [24]. Beberapa *standard built-in metrics* dan penjelasannya dapat dilihat pada Tabel 2.1.

Tabel 2.1 K6 standar built-in metrics

Nama metrics	Tipe	Deskripsi
<i>vus</i>	<i>Gauge</i>	Jumlah <i>virtual user</i> aktif saat ini.
<i>vus_max</i>	<i>Gauge</i>	Jumlah maksimum <i>virtual users</i> yang <i>possible</i> .
<i>iterations</i>	<i>Counter</i>	Agregasi berapa kali <i>virtual user</i> mengeksekusi <i>js script (default function)</i>
<i>iteration_duration</i>	<i>Trend</i>	Waktu yang dibutuhkan untuk menyelesaikan iterasi.
<i>group_duration</i>	<i>Trend</i>	<i>Duration</i> dari group pada <i>script</i> k6. Pada k6, kita dapat membuat <i>group</i> untuk mengelompokkan <i>http request</i> . Pada <i>test</i> ini, semua <i>http request</i> dibuat menjadi satu <i>group</i> , sehingga <i>duration</i> akan hampir sama dengan <i>iteration duration</i> .
<i>data_received</i>	<i>Counter</i>	Akumulasi data yang diterima.
<i>data_sent</i>	<i>Counter</i>	Akumulasi data yang dikirim.

Selain itu, pada K6 terdapat beberapa *metrics* yang dikhususkan untuk protokol yang digunakan untuk *test*. Contohnya untuk protokol HTTP, pada k6 terdapat *HTTP-specific built-in metrics*. *Metrics* tersebut hanya dihasilkan saat *test* membuat *request* HTTP [24]. *HTTP-specific built-in metrics* dan penjelasannya dapat dilihat pada Tabel 2.2.

Tabel 2.2 HTTP-specific built-in metrics

Nama metrics	Tipe	Deskripsi
<i>http_reqs</i>	<i>Gauge</i>	Jumlah total HTTP <i>request</i> yang di-generate oleh k6.
<i>http_req_blocked</i>	<i>Trend</i>	Waktu yang dihabiskan terblokir (menunggu <i>slot TCP connection</i>) sebelum inisialisasi <i>request</i> .

<i>http_req_connecting</i>	<i>Trend</i>	Waktu yang dihabiskan untuk <i>establish</i> koneksi TCP ke <i>remote host</i> .
<i>http_req_tls_handshaking</i>	<i>Trend</i>	Waktu yang dihabiskan untuk TLS <i>handshake</i> dengan <i>remote host</i> .
<i>http_req_sending</i>	<i>Trend</i>	Waktu yang dihabiskan untuk mengirim data ke <i>remote host</i> .
<i>http_req_waiting</i>	<i>Trend</i>	Waktu yang dihabiskan untuk menunggu respons dari <i>remote host</i> (“ <i>time to first byte</i> ” atau “ <i>TTFB</i> ”).
<i>http_req_receiving</i>	<i>Trend</i>	Waktu yang dihabiskan untuk menerima respons data dari <i>remote host</i> .
<i>http_req_duration</i>	<i>Trend</i>	Total waktu untuk sebuah request (<i>http_req_sending</i> <i>http_req_waiting</i> + <i>http_req_receiving</i>). Yakni berapa lama <i>remote server</i> memproses <i>request</i> dan merespon tanpa proses <i>DNS lookup</i> .
<i>http_req_failed</i>	<i>Rate</i>	Jumlah dari <i>request</i> yang gagal.

2.12 IP Address

Setiap mesin pada sebuah jaringan memiliki satu tanda pengenal. Seperti alamat saat kita ingin mengirim surat, komputer menggunakan tanda pengenal untuk mengirim data ke komputer tertentu pada sebuah jaringan. Sebagian besar jaringan saat ini, termasuk semua komputer yang ada di internet menggunakan protokol TCP/IP sebagai standar untuk berkomunikasi dalam jaringan. Pada protokol TCP/IP, penanda unik bagi sebuah komputer disebut *IP address*. IPv4 menggunakan 32 bit *binary* untuk membuat alamat unik pada jaringan. Sebuah alamat IPv4 direpresentasikan oleh empat nomor yang dipisahkan dengan titik. Setiap nomornya adalah berbasis desimal (base-10). Contohnya adalah 192.168.1.100 [25]. *IP address* terbagi menjadi dua, yakni *public* dan *private*.

2.12.1 IP Address Public

Sebuah komputer pada jaringan internet teridentifikasi oleh *IP address*. Untuk menghindari konflik, *IP address* didaftarkan ada *Network Information Centre (NIC)*. Sebuah *IP address public* diberikan kepada setiap komputer yang terhubung ke internet yang mana setiap IP itu unik. Dengan demikian, tidak mungkin terdapat dua komputer memiliki *IP address* yang sama di mana pun.

Skema pengalamatan ini, membuat setiap komputer dapat saling mencari satu sama lain secara *online* dan saling bertukar informasi. User tidak memiliki kontrol terhadap *IP address public* yang diberikan ke mereka. *IP address public* yang diberikan kepada komputer oleh *Internet Service Provider* segera setelah komputer terkoneksi ke *internet gateway* [25].

Sebuah *IP address public* dapat berupa *static* ataupun *dynamic*. *Static public IP address* tidak akan berubah dan digunakan untuk *hosting* halaman web atau sebuah layanan pada internet. Di sisi lain, *IP address public dynamic* dipilih dari kumpulan alamat yang tersedia dan berubah setiap kali seseorang terhubung ke internet. Sebagian besar pengguna internet hanya akan memiliki *IP dynamic* yang ditetapkan pada mereka dan akan hilang saat komputer tidak terhubung ke internet. Tetapi ketika tersambung kembali, akan mendapatkan *IP address* baru [25].

2.12.2 IP Address Private

Sebuah *IP address* dikatakan *private* jika *IP address* termasuk ke dalam *IP address range* yang direservasi untuk *private network* seperti *Local Area Network (LAN)*. *Internet Assigned Number Authority (IANA)* telah mereservasi tiga blok *IP address* untuk *private network* sebagai berikut [25]:

1. 10.0.0.0 – 10.255.255.255 (Total address: 16,777,216)
2. 172.16.0.0 – 172.31.255.255 (Total address: 1,048,576)
3. 192.168.0.0 – 192.168.255.255 (Total address: 65,536)

IP Address private digunakan untuk menandai komputer pada jaringan *private* termasuk rumah, sekolah, dan LAN bisnis pada bandara dan hotel yang mana membuat setiap komputer pada jaringan tersebut saling terkoneksi satu sama lain. Contohnya, jika sebuah *network* memiliki 10 komputer, masing-masing dapat diberikan *IP address* mulai dari 192.168.1.1 hingga 192.168.1.10. Tidak seperti *IP public*, seorang administrator dari *private network* dapat dengan leluasa memberikan *IP address* sesuai dengan keinginannya [25].

2.13 SSH

SSH (*Secure Shell*), adalah sebuah pendekatan populer, *powerful*, berbasis *software* untuk keamanan jaringan. Setiap kali data dikirim oleh komputer ke

jaringan, SSH secara otomatis mengenkripsinya. Ketika data mencapai penerima yang dituju, SSH secara otomatis men-*decrypt* (menguraikannya). Hasilnya adalah enkripsi transparan yang mana pengguna dapat bekerja secara normal, tanpa menyadari bahwa komunikasi mereka dienkripsi dengan aman dalam jaringan. Selain itu, SSH menggunakan algoritma enkripsi yang modern dan aman serta cukup efektif untuk ditemukan dalam aplikasi-aplikasi penting di perusahaan-perusahaan besar [26].

SSH menggunakan arsitektur *client-server*. Program server SSH, biasanya dipasang dan dijalankan oleh administrator sistem, menerima atau menolak koneksi masuk ke komputer *host*. Pengguna kemudian menjalankan program klien SSH, biasanya di komputer lain, untuk membuat permintaan ke server SSH. Semua komunikasi antara klien dan server dienkripsi dengan aman dan dilindungi dari modifikasi [26].