



BAB II

LANDASAN TEORI



Hak cipta dilindungi Undang-Undang
© Hak cipta milik IBI KKG (Institut Bisnis dan Informatika Kwik Kian Gie)

E. Message Queue

1. Pengertian Message Queue

Menurut *Wojciech Krzemien* dalam publikasinya yang berjudul *Addressing Scalability with Message Queues: Architecture and Use Cases for DIRAC Interware*, beliau mendefinisikan *message queue* sebagai sebuah arsitektur dalam skema komunikasi *asynchronous* yang menyajikan solusi dalam masalah-masalah yang ditimbulkan dalam model komputasi yang terdistribusi. Dengan adanya komponen *message queue* sebagai intermediasi antara dua proses yang saling berinteraksi dapat mempermudah proses *decoupling* antara dua titik komunikasi yang membuat sistem tersebut lebih fleksibel dan menyajikan tingkat skalabilitas dan redundansi yang tinggi.

2. Arsitektur Message Queue

Dalam buku *“Web Scalability for Startup Engineers : Tips & Techniques for Scaling Your Web Application”* yang ditulis oleh *Artur Ejsmont*, implementasi *message queue* melibatkan 3 komponen:

a. Message Producer

Producer, yang juga sering disebut sebagai *publisher* atau pengirim, berperan sebagai objek atau pihak yang mengirimkan pesan ke dalam komponen *message queue*. Mereka adalah bagian dari sistem yang memulai proses *asynchronous*.

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



b. Message Queue

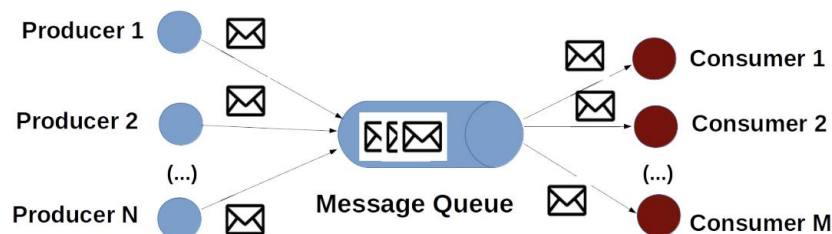
C *Message queue*, yang juga sering disebut sebagai *message broker*, *message oriented middleware (MOM)*, atau *enterprise service bus (ESB)*, adalah komponen utama dari proses *asynchronous* berbasis *message queue*. Komponen ini berperan sebagai *buffer* (penyangga) pesan-pesan yang diterima dari *producers* yang nantinya akan diproses oleh *consumers*.

c. Message Consumer

Message consumer, yang juga sering disebut sebagai *subscriber*, berperan sebagai objek atau pihak yang menerima dan memproses pesan dari komponen *message queue* dan bereaksi terhadap pesan yang didapatkan.

Gambar 2.1

Komunikasi *message queue* antara *producer* dan *consumer*



Sumber: (Ejsmont, A., 2015)

3. Keuntungan Message Queue

Komunikasi *message queue* didasari oleh sebuah gagasan untuk membuat komponen intermediasi (antrian) antara entitas yang saling berinteraksi antara satu dengan yang lainnya. Dimana entitas tersebut disebut sebagai *consumer* dan *producer*. Separasi ini membuat komunikasi antara entitas menjadi *asynchronous* karena *consumer* dan *producer* tidak perlu



berinteraksi dalam waktu yang sama. Pendekatan ini menciptakan beberapa keuntungan (Reagan, 2018):

a. Skalabilitas

Dalam model *message queue*, jumlah *consumer* yang memproses sebuah *message* bisa lebih dari satu. Dengan demikian, komponen *message queue* bisa berperan sebagai *load balancer* untuk sebuah sistem sehingga akan menjadi sangat mudah bagi pengembang sistem untuk meningkatkan jumlah *consumer* (sebagai pemroses). Implementasi yang demikian menyebabkan pekerjaan dapat dilakukan secara bersamaan (*concurrent*) dan tidak bergantung dengan pekerjaan sebelumnya (*asynchronous*).

b. Fleksibilitas

Komponen *message queue* dapat menahan ledakan beban kerja dan menjaga aplikasi agar tetap berjalan dengan normal ketika ada lonjakan akses terhadap aplikasi.

c. Temporal Decoupling

Karena *producer* dan *consumer* hanya bisa berkomunikasi melewati komponen *message queue*, dan karena *message* yang ada di dalam komponen *message queue* akan tetap ada di dalam komponen tersebut hingga *message* tersebut diterima oleh *consumer*, maka bisa dibuat sistem dimana *consumer* yang akan memproses pekerjaan yang ada di *message queue* hanya diaktifkan ketika ada pekerjaan di *message queue*, dan berada dalam kondisi non-aktif jika tidak ada pekerjaan.

d. Resiliensi

Hak Cipta Dilindungi Undang-Undang

Hak Cipta milik IBI KKG (Institut Bisnis dan Informatika Kwik Kian Gie)

Institut Bisnis dan Informatika Kwik Kian Gie

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



Resiliensi dapat diciptakan melalui *temporal decoupling*. Ketika salah satu **C** consumer gagal dalam memproses suatu *message* di dalam *message queue*, *message* tersebut tidak akan hilang dan dapat diproses ulang oleh consumer yang sama atau consumer lainnya.

4. Jenis Implementasi Message Queue

Metode implementasi *message queue* dalam sebuah sistem terbagi menjadi 3 macam, yaitu:

a. Metode Direct Worker Queue

Dalam metode ini, *consumer* dan *producer* hanya perlu mengetahui nama dari sebuah *queue* yang diinginkan. Setiap pesan yang dibuat oleh *producer* ditambahkan ke dalam sebuah *queue*. *Queue* ini diidentifikasi berdasarkan namanya, dan beberapa *producer* dapat mempublikasikan pesan ke *queue* yang sama kapan saja. Setiap pesan yang masuk ke dalam *message queue* ini akan diarahkan hanya ke satu *consumer*. Dengan demikian, jika dalam sebuah sistem *message queue* terdapat jumlah *consumer* yang lebih dari satu, setiap *consumer* hanya mendapatkan sebagian dari seluruh pesan yang dipublikasikan ke *queue* tersebut dan setiap pesan hanya akan diterima oleh satu *consumer*.

Hak Cipta Dilindungi Undang-Undang

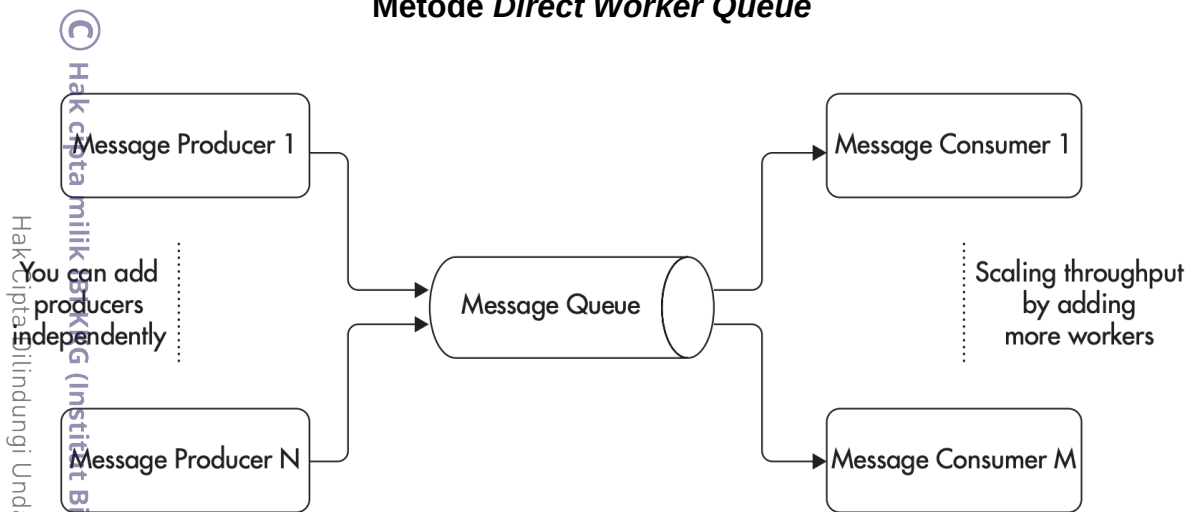
C Hak cipta milik IBIKKG (Institut Bisnis dan Informatika Kwik Kian Gie)

Institut Bisnis dan Informatika Kwik Kian Gie

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



Gambar 2.2
Metode *Direct Worker Queue*



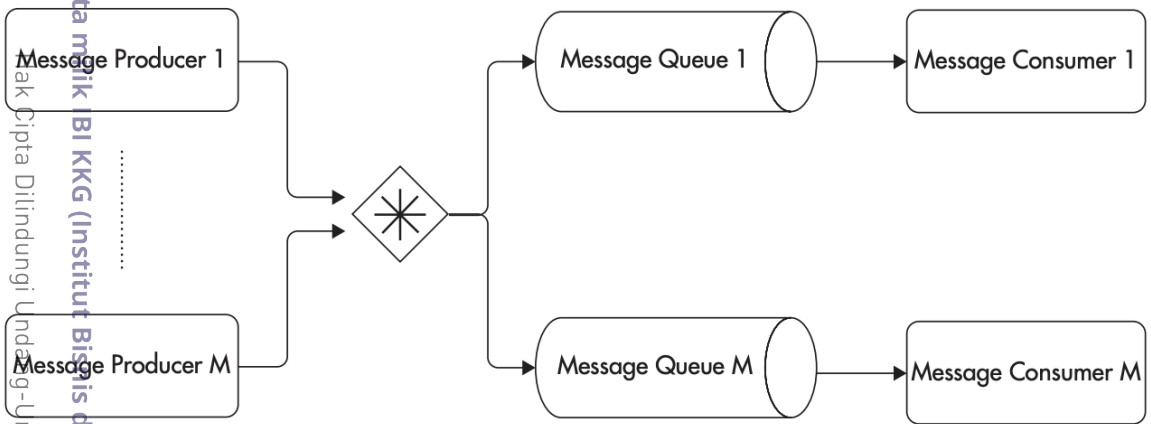
Sumber: (Ejsmont, A., 2015)

b. Metode Publish/Subscribe

Dalam model *publish/subscribe*, pesan dapat diterima oleh lebih dari satu *consumer* (berbeda dengan metode *direct worker queue* dimana satu pesan hanya dapat diterima dan diproses oleh satu *consumer*). Dalam model ini, *publisher* mempublikasikan pesan ke dalam sebuah *topic*, bukan sebuah *queue*. Setiap pesan yang masuk ke dalam sebuah *topic* akan diduplikasi untuk setiap *consumer* yang menunggu pesan dari *topic* yang sama.



Gambar 2.3
Metode Publish/Subscribe



Sumber: (Ejsmont, A., 2015)

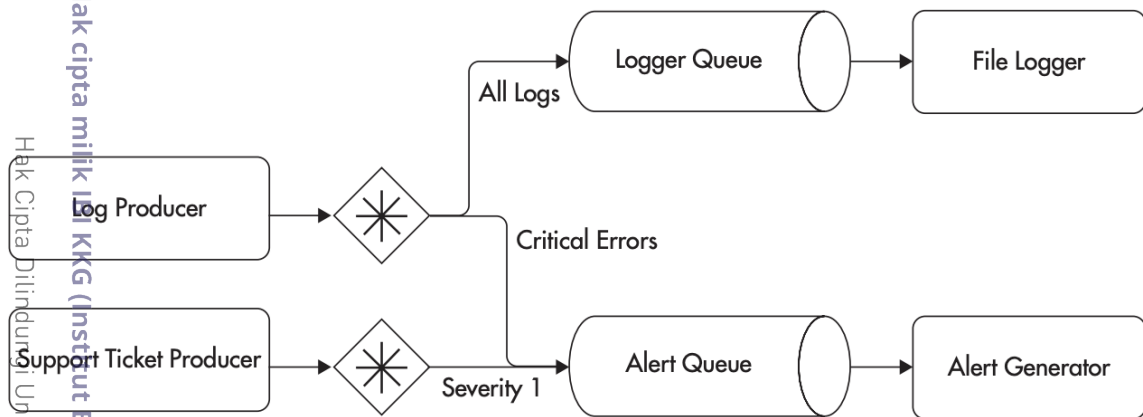
c. Metode Custom Routing Rules

Dalam metode ini, *consumer* dapat menentukan secara fleksibel pesan apa saja yang harus diarahkan kepada dirinya. *Consumer* dapat membuat aturan rute dengan komponen *message queue* untuk menciptakan perutean fleksibel dan spesifik.

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



Gambar 2.4
Contoh Metode Custom Routing Rules



Sumber: (Ejsmont, A., 2015)

A. Asynchronous Programming

Definisi *asynchronous programming* dari Mark Williams (2019) dalam bukunya yang berjudul *Expert Twisted: Event-Driven and Asynchronous Programming with Python* adalah pelaksanaan eksekusi program yang tidak terjadi secara berurutan dengan waktu dimana program dapat terus berjalan ketika suatu titik dalam eksekusi program sedang menunggu selesainya suatu perintah. Berbeda dengan *synchronous programming* dimana segala perintah dalam alur program akan menyebabkan seluruh eksekusi program akan berhenti berjalan dan menunggu akan selesainya perintah yang dilaksanakan secara berurutan.

Harsh Chawla (2019) dalam bukunya yang berjudul *Building Microservices Applications on Microsoft Azure: Designing, Developing, Deploying, and Monitoring* menjelaskan keuntungan sistem yang bersifat



asynchronous dibanding dengan sistem yang *synchronous*. Menurut beliau, komunikasi antara aplikasi yang *asynchronous* mengurangi ketergantungan antar program. Sistem yang bersifat *asynchronous* juga lebih responsif dan fleksibel terhadap perubahan.

B. Web

1. Pengertian Web

World Wide Web (WWW) yang biasa disebut sebagai web, adalah bidang informasi dimana dokumen dan media lainnya diidentifikasi melalui *Unified Resource Locators (URL)*, yang bisa diakses melalui jaringan internet. Web membuka jalan bagi komputer personal untuk dapat melaksanakan aplikasi berbasis web dan menemukan dokumen-dokumen multimedia dari mana saja yang ada di internet.

Web adalah penemuan yang cukup baru. Pada tahun 1989, *Tim Berners-Lee* dari *CERN* mulai mengembangkan sebuah teknologi untuk berbagi informasi melalui dokumen teks yang dinamakan *HyperText Markup Language (HTML)*. Beliau juga menulis sebuah protokol untuk sistem informasi yang baru ini, yang ia namakan sebagai *World Wide Web*. Untuk lebih spesifik, ia menamakan protokol ini sebagai *HyperText Transfer Protocol (HTTP)* sebagai protokol komunikasi dalam jaringan web (Deitel, 2011).

2. Mekanisme Web

Dalam ruang lingkup web, *resource* adalah segala sesuatu dalam bentuk dokumen teks (*HTML*) atau multimedia yang ada diatas sebuah *server* dan bisa diakses melalui web. Dokumen *HTML* biasanya mengandung

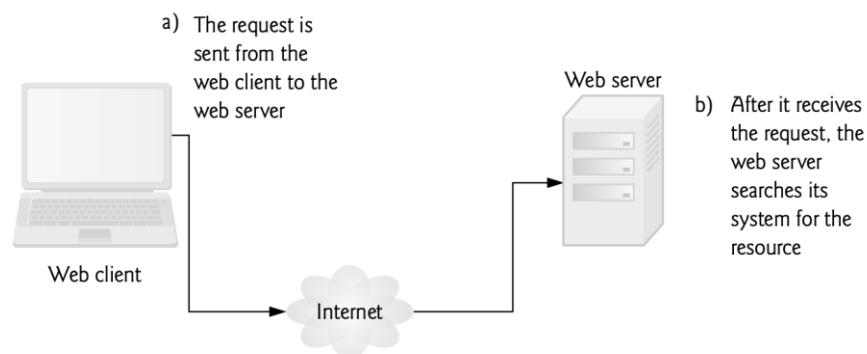


1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.

Hyperlinks, yang saat diklik, akan membuka dokumen web lainnya. Bentuk yang dapat direferensi oleh *Hyperlink* dapat berupa tulisan, gambar, dan media lainnya. *URI (Uniform Resource Identifiers)* memberikan identitas terhadap dokumen-dokumen yang ada di internet. *URI* yang diawali dengan *http://* disebut sebagai *URL (Uniform Resource Locator)*. Pengguna dapat menggunakan aplikasi *web browser* sebagai *client* untuk mengakses dokumen-dokumen yang tersedia di *web* melalui alamat *URL* yang diberikan. Komputer yang menyediakan dokumen web yang bisa diakses melalui *URL* disebut sebagai *server*.

Ketika alamat *URL* dimasukan, sebuah aplikasi *web browser* melalui protokol *HTTP* akan melakukan *request* dokumen web yang ada di alamat tersebut. Gambar berikut akan memberikan gambaran proses ketika *client* (*web browser*) mengirimkan *request* ke sebuah *web server*.

Gambar 2.1
Proses request dari client ke server



Sumber: (Paul Deitel, H. D., 2011)

Pada gambar diatas, *client* dalam bentuk *web browser* melakukan *request* ke *server*. *Request* tersebut dalam bentuk yang paling sederhana



adalah: “*GET /books/downloads.html HTTP/1.1*”. Dalam *request* ini, kata *GET* adalah sebuah metode *HTTP* yang menandakan bahwa *client* meminta sebuah dokumen web dari server yang ada di lokasi “*/books/downloads.html*”. Server yang menerima *request* dalam bentuk *HTTP* mampu menterjemahkan *request* ini dan mengirimkan *response* balik ke *client*. *Response* ini berisi kode numerik (*response code*) yang menandakan status dari transaksi komunikasi antara *client* dan server (Deitel, 2011).

3. Metode HTTP

Metode *HTTP* (*HTTP Method*) adalah kode yang menunjukkan tindakan yang ingin dilakukan oleh *client*. Jenis *HTTP Method* yang paling umum digunakan ada 5, yaitu:

- a. GET, menandakan bahwa *client* ingin meminta resource yang ada di server di lokasi tertentu
- b. POST, menandakan bahwa *client* ingin menambahkan resource baru ke server di lokasi tertentu
- c. PUT, menandakan bahwa *client* ingin meletakkan sebuah resource tertentu ke server di lokasi tertentu
- d. DELETE, menandakan bahwa *client* ingin menghapus resource yang ada di server di lokasi tertentu
- e. PATCH, menandakan bahwa *client* ingin melakukan modifikasi parsial terhadap dokumen yang ada di server di lokasi tertentu

4. HTTP Status Code

Response code atau *status code* yang direpresentasikan dalam 3 digit angka numerik, menandakan status dari transaksi komunikasi antara *client*



dan server. Melalui *response code*, *client* dapat mengetahui status dari komunikasi antara *client* dan *server* yang sedang berlangsung. *Response code* dibagi menjadi 5 kategori:

- a. Informasional (1XX)
- b. Berhasil (2XX)
- c. Redireksi (3XX)
- d. Kesalahan *Client* (4XX)
- e. Kesalahan *Server* (5XX)

C Grafik Komputer

1. Pengenalan Grafik Komputer

Menurut *Steve Marschner (2016)* dalam bukunya yang berjudul "*Fundamentals of Computer Graphics*", beliau mendefinisikan grafik komputer sebagai segala bentuk penggunaan komputer untuk membuat dan mengubah media berbentuk grafis/gambar. Grafik ini dapat berupa dua dimensi atau tiga dimensi; gambar dapat berupa gambar sintesis yang dibuat oleh komputer atau gambar yang merupakan hasil dari pengubahan foto. Menurut beliau, sebagian besar dari praktisi grafik komputer setuju akan pembagian bidang dari grafik komputer:

- a. Modeling, bidang yang berinteraksi dengan spesifikasi matematis dari bentuk dan sifat-sifat visual yang bisa disimpan dalam komputer.
- b. Rendering, bidang yang mempelajari proses pengubahan model tiga dimensi dalam komputer menjadi gambar dua dimensi yang bisa ditampilkan ke perangkat keras output gambar seperti monitor.



- c. Animation, bidang yang mempelajari pembuatan ilusi pergerakan melalui urutan dari gambar.

2. Penyuntingan Gambar

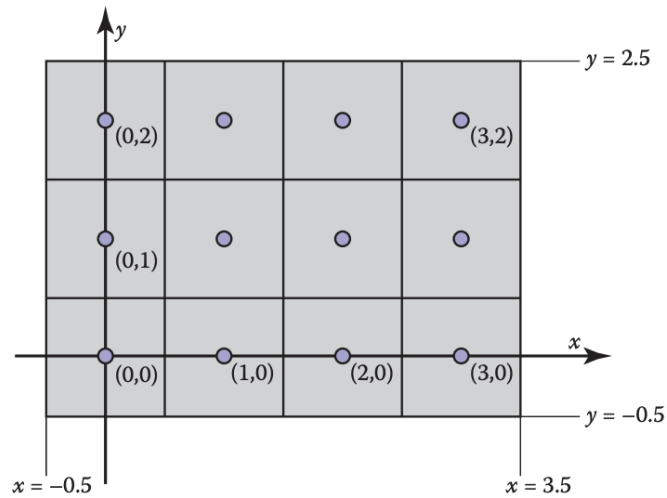
Steve Marschner (2016) mendefinisikan *image editing* atau penyuntingan gambar sebagai proses untuk mengubah atau memanipulasi sebuah gambar menggunakan komputer. Aplikasi komputer yang mampu melakukan manipulasi gambar disebut sebagai *image editor*. Dalam proses *image editing*, proses yang menambahkan sebuah gambar atau pola di atas gambar yang asli dengan tujuan untuk menandakan kepemilikan dari sebuah gambar disebut sebagai *watermarking*.

3. Raster Image

Steve Marschner (2016) menjelaskan bahwa dalam komputer, jenis grafik komputer dibagi menjadi dua jenis, yaitu raster image dan vector image. Raster Image atau *raster graphics* adalah sebuah struktur data dalam bentuk *dot matrix* yang merepresentasikan sebuah kotak yang terdiri dari *pixel* (titik warna). Struktur data ini juga sering disebut sebagai *bitmap* karena tiap warna *pixel* dispesifikasikan sebagai komposisi warna merah, hijau, dan biru (*RGB*, *Red Green Blue*), dengan angka *bits*. Setiap *pixel* yang membangun sebuah gambar dapat diidentifikasi berdasarkan lokasinya yang direpresentasikan dalam koordinat dua dimensi.



Gambar 2.5
Sistem koordinat *pixel*



Sumber: (Marschner, S., & Shirley, P., 2016)

Proses manipulasi gambar menggunakan komputer dalam bentuk *raster image* disebut sebagai *raster image editing*. Aplikasi komputer yang mampu melakukan proses tersebut disebut sebagai *raster image editor*. Dalam *raster image*, ukuran gambar adalah ukuran dari jumlah *pixel* yang merepresentasikan gambar tersebut (panjang gambar adalah jumlah *pixel* pada sumbu horizontal, dan lebar panjang adalah jumlah *pixel* pada sumbu vertikal). Sehingga proses yang mengubah ukuran ini disebut sebagai *image resizing*.

© Hak cipta milik IBI KKG (Institut Bisnis dan Informatika Kwik Kian Gie)

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



D. Batch



Hak Cipta Dilindungi Undang-Undang
Hal Cipta milik IBI KKG (Institut Bisnis dan Informatika Kwik Kian Gie)

Institut Bisnis dan Informatika Kwik Kian Gie

Dalam kamus bahasa Inggris *Cambridge*, *batch* artinya memperlakukan beberapa benda sekaligus agar mereka dapat diperlakukan secara bersamaan.

Dalam ilmu komputer, *batch system* adalah sistem yang memproses pekerjaan dalam jumlah besar dengan input yang sudah ditentukan sebelum sebuah operasi dijalankan (Silberschatz, 2018).

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.