BAB II
LANDASAN TEORI

II.1. Portal Berita

Portal berita adalah suatu halaman *website* yang menyediakan informasi tentang suatu kejadian terbaru atau kejadian yang sudah terjadi. Informasi yang disampaikan tidak jauh berbeda dengan informasi yang ada pada media cetak, hanya saja pada media internet atau disebut juga sebagai media online. Media online adalah media massa yang tersaji secara online pada *website* [8]. Secara *content* penulisan berita, berita yang ditulis dalam *website* biasanya memiliki kaidah penulisan yang sama dengan tata cara penulisan berita pada TV, radio, atau koran. Portal berita ini mudah dalam diakses karena disimpan dalam jaringan internet, sehingga siapapun dapat dengan mudah mengaksesnya.Fitur fitur yang terdapat pada portal berita diantaranya adalah:

1. *Headline*

   Headline atau judul dari berita tersebut yang bertujuan agar memudahkan pembaca untuk mengetahui peristiwa yang sedangg terjadi dan menampilkan berita yang sedang menjadi popular.

2. *Lead*

   Lead atau inti dari berita tersebut sehingga lead adalah unsur yang terpenting karena dapat sangat menentukan apakah berita tersebut menarik untuk di bacca atau tidak.

3. *Body*

   Isi berita yang menceritakan informasi tersebut secara singkat, padat dan jelas.

II.2. *Web scrapping*

Web scrapping merupakan suatu teknik untuk mengutip data ataupun informasi dari suatu web atau blog menggunakan perangkat lunak dengan metode tertentu. Biasanya perangkat lunak tersebut ensimulasikan aktifitas manusia terhadap suatu web atau blog dengan menggunakan low - level HTTP atau

menggunakan web browser [9].*Web scraping* mempunyai banyak kegunaan dan sangat membantu untuk pengambilan dokumen, salah satunya yaitu untuk konten berita dimana isi kontenya langsung diambil dari situs yang dijadikan target. Secara umum dalam mengimplementasikan teknik *web scrapping* dibutuhkan beberapa tahap yaitu

1. Memanggil/*Request* url target.

   Sistem akan memanggil target yang dalam hal ini adalah alamat *url http* dari *web* yang dijadikan target, contohnya adalah www.kompas.com

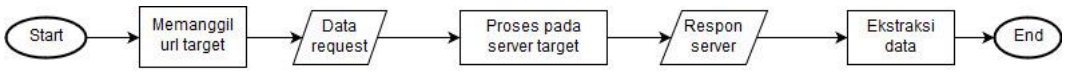
2. Proses pada server target.

   Server target akan melakukan proses untuk *request* yang telah dilakukan, kemudiah akan menyajikan data berdasarkan apa yang di *request*.

3. Ekstrasi data.

   Sistem akan melakukan ekstraksi pada data html yang yang berikan oleh server. Data yang dirasa penting akan diambil, kemudian hasil dari ekstraksi ini kemudian akan disimpan kedalam database.

Proses ini dapat digambarkan seperti pada Gambar II.1



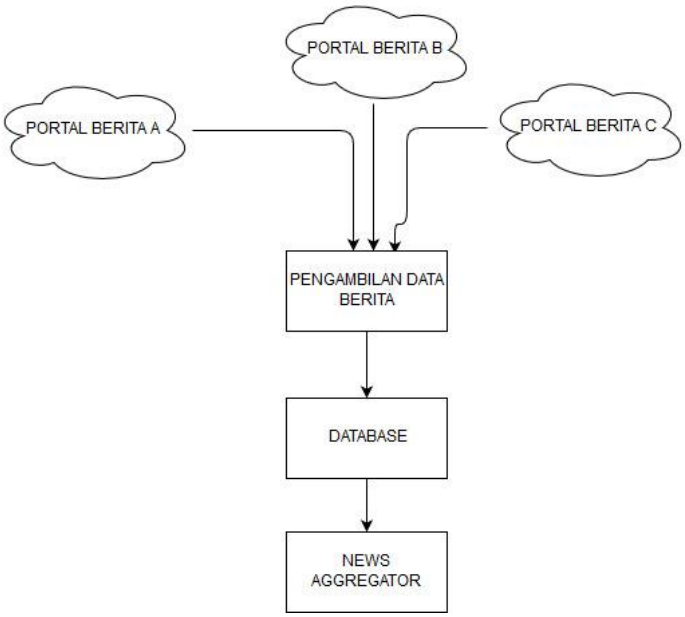
Gambar II.1 Proses Web *Scrapping*

Pada penelitian ini, teknik *web scrapping* akan digunakan untuk mengambil data dari beberapa portal berita. Data tersebut akan disimpan pada database untuk kemudian diolah oleh sistem.

II.3. *News aggregator*

News aggregator merupakan perangkat lunak atau aplikasi yang menggabungkan konten dari berbagai web [10]. contohnya seperti surat kabar online, blog, atau blog video di satu tempat agar mudah dibaca.

Teknologi ini memudahkan pengguna dengan cara menggabungkan data dari beberpa situs web ke dalam satu halaman dan dapat menunjukan informasi yang telah diperbaharui dari situs tersebut. *News Agregator* juga memberikan efisiensi waktu dan mempermudah pengguna untuk tidak selalu memeriksa situs tersebut untuk membaharuan. Cara kerja dari sistem *news aggregator* ini dapat dilihat pada Gambar II.2



Gambar II.2 Cara Kerja *News Aggregator*

II.4. *Text Mining*

Menurut O. Maimon dan L. Rokach [11], *Text mining* adalah suatu proses ekstraksi pola tertentu dari *database* dokumen teks yang besar yang bertujuan untuk menemukan pengetahuan.T*ext mining merupakan* sebuah cara atau metode untuk menemukan atau menggali informasi darikumpulan dokumen teks yang besar. *Text mining* memberikan solusi dari permasalahan seperti pemrosesan, pengorganisasian atau pengelompokkan dan menganalisa *unstructured text* dalam jumlah besar.

Text mining juga sering disebut sebagai *Knowledge Discovery in Textual Databases*. Text mining merupakan proses mengesktrak *patterns* dan *knowledge* yang bersifat menarik dan *nontrivial* (penting) dari dokumen dokumen teks. Pada intinya proses kerja *text mining* sama dengan proses kerja data

mining pada umumnya hanya saja data yang di *mining* merupakan *text databases*. Data teks akan diproses menjadi data numerik agar dapat dilakukan proses lebih lanjut.

II.4.1. Text Preprocessing

Dalam text mining ada istilah *preprocessing* data, yaitu proses pendahulu yang diterapkan terhadap data teks yang bertujuan untuk menghasilkan data numerik. Berikut adalah tahapan *preprocessing* yang dilakukan.

a. Data *cleaning*

Tahapan ini dilakukan apabila data yang akan diolah memiliki tag html. Proses ini melakukan penghapusan tag markup dan format khusus dari data yang akan diolah. Pada tahapan ini semua tag html seperti <p>, , <a href=”#”> akan dihapus dari berita.. Berikut merupakan contoh data sebelum dan sesudah dilakukan penghapusan tag html.

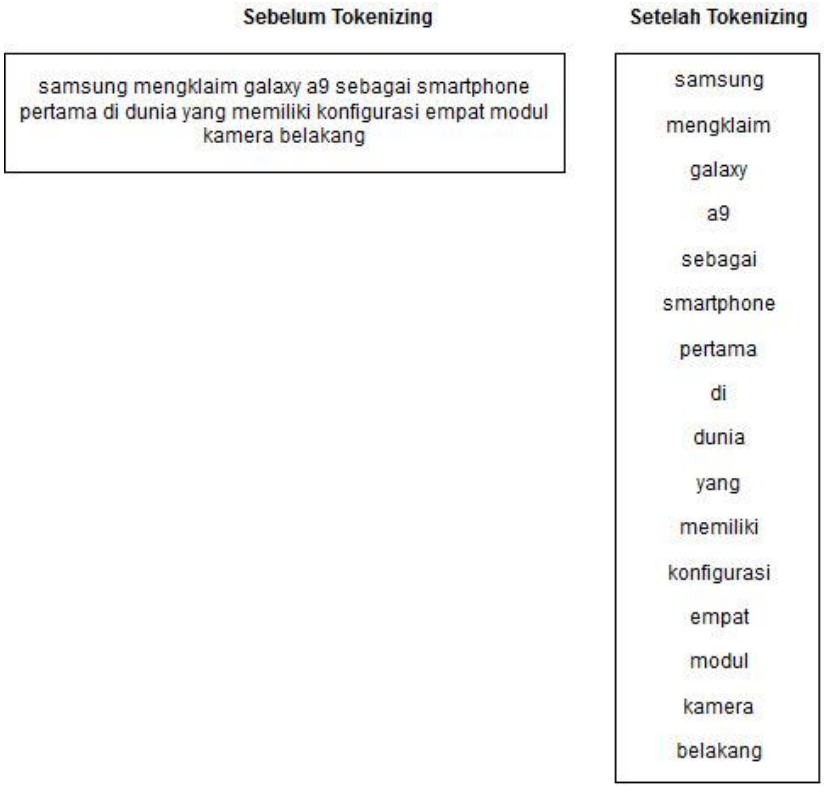
b. *Case Folding*

Tahapan ini merupkan tahapan untuk mengubah huruf besar menjadi huruf kecil atau disebut juga *case folding.* Tahap *case folding* adalah mengubah seluruh huruf dari “a” sampai dengan “z‟ dalam dokumen menjadi huruf kecil. Tidak semua kata dalam dokumen konsisten menggunakan huruf kapital, maka dari itu *case folding* mengkonversi keseluruhan teks dalam dokumen menjadi huruf kecil. Tahapan *case folding* dapat dilihat pada Gambar II.3

Gambar II.3 Contoh *Case Folding*

c. *Tokenizing*

Tahapan *Tokenizing* ini merupakan proses penguraian deskripsi yang semula berupa kalimat berisi kata-kata dan tanda pemisah antara kata seperti titik(.), koma(,), spasi dan tanda pemisah lain menjadi *token* atau potongan kata tunggal. dan menghapus karakter tertentu seperti tanda baca. Contoh tahapan *tokenizing* ini dapat dilihat pada Gambar II.4



Gambar II.4 Contoh *Tokenizing*

d. Stopword Removal

Tahap *stopword removal* merupakan proses mengambil kata -kata penting dari hasil *tokenizing*. Untuk melakukan tahap ini bisa menggunakan algoritma *stopword removal* (membuang kata yang kurang penting). Metode ini dilakukan dengan cara menghilangkan kata tidak penting (*stopword*) pada dokumen melalui pengecekan kata-kata hasil token deskripsi apakah termasuk di dalam daftar kata tidak penting atau tidak. Jika termasuk di dalam *stopword* maka kata-kata tersebut akan di-hilangkan dari dokumen sehingga kata-kata yang tersisa di dalam dokumen di anggap sebagai kata-kata penting. Contoh Proses *stopword removal* ini dapat dilihat pada Gambar II.5

Gambar II.5 Contoh Tahapan *Stopword Removal*

II.4.2. TF-IDF

Term Weighting TF-IDF Merupakan skema yang banyak digunakan dalam pembobotan kata. Beberapa hal yang perlu diperhatikan dalam pencarian informasi dokumen adalah pembobotan *term*. Dapat berupa kata, fase atau hasil index lainnya di dalam suatu dokumen, setiap kata di berikan indikator yang disebut dengan *term weight*.

a. Term Frequency (TF)

TF merupakan frekuensi dari munculnya sebuah kata atau *term* di dalam dokumen. Semakin besar jumlah kemunculannya semakin besar juga bobotnya atau nilai kesesuaiannya semakin besar. Berdasarkan dari buku yang ditulis oleh jiawei han [12] rumus untuk mendapatkan nilai *TF* ini adalah

$$TF(d,t) = \begin{cases} 0 & if\, freq(d,t) = 0 \\ 1 + \log(1 + \log(freq(d,t))) & otherwise. \end{cases} \quad \textbf{(2.1)}$$

Keterangan:

TF(d,t) merupakan nilai *term d* dalam dokumen *t* .

freq(d,t) merupakan jumlah *term* d dalam dokumen t.

Berdasarkan persamaan II.1 dapat dilihat bahwa jika *freq(d,t)* adalah 0, maka nilainya 0. Tetapi jika kemunculan katanya lebih dari 1 maka gunakan perhitungan untuk mencari nilai *TF(d,t)* nya.

b. Inverse Document Frequency (IDF)

Suatu perhitungan dari kata didistribusikan secara luas pada kumpulan dokumen yang bersangkutan. *IDF* menunjukkan hubungan ketersediaan sebuah *term* dalam seluruh dokumen. Semakin sedikit jumlah dokumen yang mengandung *term* yang dimaksud, maka nilai IDF semakin besar. *Inverse Document Frequency* (*IDF*) dapat dihitung dengan menggunakan perhitungan

$$IDF(t) = \log \frac{1 + |d|}{|d_t|}, \qquad \textbf{(2.2)}$$

Keterangan:

IDF(t) merupakan nilai *IDF*.

d merupakan jumlah keseluruhan dokumen

d_t merupakan jumlah dokumen yang mengandung *term t*

setelah diketahui nilai dari *TF* dan *IDF* nya selanjutnya adalah menentukan nilai *TF-IDF* nya dengan menggunakan

$$TF\text{-}IDF(d,t) = TF(d,t) \times IDF(t). \qquad \textbf{(2.3)}$$

Keterangan:

TF-IDF(d,t) merupakan nilai tf-idf nya

TF(d,t) merupakan nilai dari tf

IDF(t) merupakan nilai dari idf

II.4.3. Cosine Smilarity

Cosine similarity merupakan metode yang digunakan untuk menghitung tingkat kesamaan/*similarity* antar dua buah objek. Dalam hal ini yang dibandingkan adalah dua buah dokumen. Nilai *cosine similarity* ini adalah dari 0 menuju 1, semakin besar nilainya maka semakin besar pula kemiripan antara dua dokumen tersebut.

$$sim(v_1, v_2) = \frac{v_1 \cdot v_2}{|v_1||v_2|}, \qquad \textbf{(2.4)}$$

$$sim_{(A,B)} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} * \sqrt{\sum_{i=1}^{n} B_i^2}} \qquad \textbf{(2.5)}$$

Keterangan**:**

$\sum_{i=1}^{n} A_i B_i$ merupakan Jumlah kata yang ada pada dokumen A dan yang ada pada dokumen B.

$\sqrt{\sum_{i=1}^{n} A_i^2}$ merupakan Jumlah kata yang ada pada dokumen A.

$\sqrt{\sum_{i=1}^{n} B_i^2}$ merupakan jumlah kata yang ada pada dokumen D2.

II.4.4. *single pass clustering*

single pass clustering merupakan metode yang digunakan untuk menggabungkan beberapa dokumen kedalam satu grup yaitu dengan melakukan pengelompokan data satu persatu dan membentuk kelompok dengan evaluasi dari setiap data yang di masukan ke proses *cluster*. Evaluasi tingkat kesamaan antar data dan juga *cluster* dapat dilakukan dengan berbagai cara termasuk juga menggunakan fungsi jarak, vector similarity dan lain lain.

Dalam menggunakan algoritma ini, dua hal yang perlu menjadi perhatian adalah penentuan *objective function* dan penentuan *threshold value*. *Objective function* yang ditentukan haruslah sebisa mungkin mencerminkan keadaan data yang dimodel dan dapat memberikan nilai tingkat kesamaan atau perbedaan yang terkandung di dalam data tersebut. Penentuan threshold value juga merupakan hal

yang subjektif, makin besar nilai threshold, makin mudah suatu data untuk bergabung ke dalam suatu *cluster*, dan demikian juga sebaliknya.

Algoritma yang sering digunakan dalam *Single Pass Clustering* adalah sebagai berikut: [13]

1. untuk setiap perulangan dokumen D
    a. Cari *cluster* baru dengan membandingkan dokumen D dan satu dokumen lain untuk dicari nilai *cosine similarity*-nya.
    b. jika nilai *cosine similarity*-nya lebih dari nilai *threshold*, maka dokumen yang dibandingkan tersebut termasuk dalam *cluster* C. Nilai dari *cluster* C yang terbentuk ini adalah nilai tengah antara dokumen D dan dokumen yang dibandingkan.
    c. Jika nilai *cosine similarity*-nya kurang dari nilai *threshold*, maka buat klaster baru beranggotakan dokumen D saja
2. Perulangan berhenti

II.5. PHP

PHP merupakan singkatan dari *Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source* [14]. PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada *server (server side HTML embedded scripting).* PHP adalah *script* yang digunakan untuk membuat halaman yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru atau *up to date*. Semua *script* PHP dieksekusi pada *server* dimana *script* tersebut dijalankan. PHP ini bersifat *open source* sehingga dapat digunakan oleh semua *programmer* dan mampu digunakan di semua platform atau sistem operasi, seperti Windows, MacOS dan Linux. Logo php dapat dilihat pada Gambar II.6

Gambar II.6 Logo PHP

Adapun keunggulan yang dimiliki oleh PHP adalah:

1. *Life Cycle* yang sangat singkat, sehingga PHP selalu *up to date* mengikuti perkembangan teknologi internet.
2. *Cross Platform*, yakni PHP dapat dipakai di hampir semua *webserver* yang ada di pasaran (terutama Apache dan Microsoft IIS) dan dijalankan pada berbagai sistem operasi (Linux, Windows, FreeBSD).
3. PHP mendukung koneksi ke banyak *database* baik yang gratis maupun komersil, seperti *MySQL, mSQL, Oracle, Microsoft SQL Server, Interbase,* dan banyak lagi.
4. bersifat *open source* dan gratis. Kemudahan dalam mendapatkan dokumentasi di internet, tidak akan sulit untuk mencari baik itu referensi kode-kode PHP yang sudah jadi dan juga mengajukan pertanyaan pada grup-grup diskusi yang di dalamnya banyak sekali para ahli PHP

II.6. MYSQL

MySQL adalah perangkat lunak sistem manajemen basis data *SQL* (*database management system*) atau DBMS dari sekian banyak DBMS, seperti Oracle, MS SQL, Postagre SQL, dll [14]. *MySQL* merupakan sebuah *software* yang berguna sebagai suatu *database server* yang cukup terkenal. Kepopulerannya seiring dengan *user*an *script* PHP untuk *web programming*. *Database server* itu

sendiri merupakan suatu *software* yang bertugas untuk melayani permintaan (*request*) *query* dari *client*.

MySQL sebagai suatu *database server* mempunyai beberapa kemampuan, salah satunya harus menyediakan suatu sistem manajemen *database* yang dapat mengatur bagaimana menyimpan, menambah, mengakses data dan transaksitransaksi *database* lainnya. *MySQL* cepat sekali berkembang, karena *MySQL* merupakan suatu *software* yang *Open Source*. Logo MySQL dapat dilihat pada Gambar II.7

Gambar II.7 Logo MySQL

MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan *MySQL*, namun tidak boleh dijadikan produk turunan yang bersifat komersial. *MySQL* sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Keandalan suatu sistem *database* (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh *user* maupun program-program aplikasinya. Sebagai *database server*, *MySQL* 48 dapat dikatakan lebih unggul dibandingkan *database server* lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query*

MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase. *MySQL* memiliki beberapa keistimewaan, antara lain :

1. *Portabilitas*. *MySQL* dapat berjalan stabil pada berbagai sistem operasi seperti *Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga*, dan masih banyak lagi.
2. *Open Source*. *MySQL* didistribusikan secara *open source,* dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.
3. *Multiuse*. *MySQL* dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. *Performance tuning*. *MySQL* memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. Jenis Kolom. *MySQL* memiliki tipe kolom yang sangat kompleks, seperti *signed/unsigned integer*, *float*, *double*, *char*, *text*, *date*, *timestamp*, dan lainlain.
6. Perintah dan Fungsi. *MySQL* memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam perintah (*query*).
7. Keamanan. *MySQL* memiliki beberapa lapisan sekuritas seperti *level subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi
8. Skalabilitas dan Pembatasan. *MySQL* mampu menangani basis data dalam skala besar, dengan j*UML*ah rekaman (*records)* lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
9. Konektivitas. *MySQL* dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, *Unix soket* (UNIX), atau *Named Pipes* (NT).
10. Lokalisasi. *MySQL* dapat mendeteksi pesan kesalahan pada *client* dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk di dalamnya.

11. Antar Muka. *MySQL* memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).
12. Klien dan Peralatan. *MySQL* dilengkapi dengan berbagai peralatan (*tool*) yang dapat digunakan untuk administrasi basisdata, dan pada setiap peralatan yang ada disertakan petunjuk *online*.
13. Struktur tabel. *MySQL* memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan basisdata lainnya semacam PostgreSQL ataupun Oracle.

II.7. Javascript

JavaScript adalah bahasa skrip yang populer di internet dan dapat bekerja di sebagian besar penjelajah web populer seperti Internet Explorer (IE), Mozilla Firefox, Netscape dan Opera [15]. Javascript dibuat agar mudah diintegrasikan kedalam program dan aplikasi lain, misalnya *browser*. Kode JavaScript dapat disisipkan dalam halaman web menggunakan tag SCRIPT. JavaScript pertama kali dikembangkan oleh Brendan Eich dari Netscape dibawah nama Mocha, yang nantinya namanya diganti menjadi LiveScript, dan akhirnya menjadi JavaScript. JavaScript bisa digunakan untuk banyak tujuan, misalnya untuk membuat efek rollover baik di gambar maupun teks, dan yang penting juga adalah untuk membuat aplikasi web yang interaktif.

II.8. jQuery

jQuery merupakan sebuah *library* JavaScript yang dirancang untuk menyederhanakan *client-side scripting* pada HTML. jQuery adalah salah satu javascript framework terbaik saat ini [15]. jQuery bersifat gratis dan *open source* di bawah lisensi MIT. Logo jquery dapat dilihat pada Gambar II.8

Gambar II.8 Logo jQuery

Sintaks pada jQuery didesain untuk memudahkan dalam navigasi sebuah dokumen, pemilihan elemen DOM, pembuatan animasi, penanganan event, dan pengembangan aplikasi berbasis *Ajax*. jQuery juga menyediakan sebuah paradigma untuk penanganan event yang diluar pemilihan dan manipulasi elemen dasar DOM. *Event assignment* dan *event callback function* dapat dilakukan dengan hanya satu langkah atau satu baris kode. jQuery juga bertujuan menggabungkan fungsional *JavaScript* yang sering digunakan.

Keuntungan menggunakan jQuery adalah:

1. **Mendorong pemisahan antara JavaScript dan HTML**: Pustaka jQuery menyediakan sintaks yang sederhana utnuk penambahan penanganan event pada DOM dengan hanya menggunakan JavaScipt, bukan justru menambah event atribut HTML untuk memanggil fungsi JavaScript. Inilah yang mendorong para pengembang untuk memisahkan kode JavaScript dari markup HTML
2. **Keringkasan dan kejelasan**: jQuery mempromosikan keringkasan dan kejelasan kode dengan fitur seperti chainable function dan shorthand function names.
3. **Mengeliminasi ketidak kompatibilitasan antar peramban (browser)**: Engine JavaScript pada setiap peramban pastilah sedikit berbeda antara satu dengan yang lainnya, jadi kode JavaScript yang berjalan pada sebuah peramban, bisa jadi tidak berjalan pada peramban yang lainnya. Seperti toolkit JavaScript lainnya, jQuery menangani seluruh ketidak konsistenan antar peramban dan menyediakan antar-muka konsisten yang dapat bekerja pada berbagai peramban yang berbeda.
4. **Ekstensibel**: Event baru, elemen-elemen, dan method dapat dengan mudah ditambahkan dan kemudian dapat digunakan ulang sebagai sebuah plugin.