BAB II

GAMBARAN UMUM PERUSAHAAN

2.1 Sejarah Singkat Perusahaan

41Studio berawal dari usaha web developer sederhana yang di gawangi oleh tiga orang programer muda yang terdiri dari Hendra Nicolas yang sekarang berada di posisi CEO sekaligus senior rails web developer , Bainur Budyana senior web developer dan Syafikli Musyafako sebagai senior web developer di 41Studio. Asal mula nama 41Studio berasal dari nama studio musik yang bernama studio41 yang berada di daerah cimahi, pada tahun 2007 studio41 tersebut berpindah tempat dan pada saat yang bersamaan kami menggantikan studio musik tersebut menjadi kantor web developer kami dengan sedikit membalikan nama dari Studio41 menjadi 41Studio itu sejarah singkat tenatang nama perusahaan CV. 41Studio.

Pada tahun 2008 41studio mulai mengalami banyak perkembangan dan peningkatan, seiring dengan bommingnya rails di dunia dan banyaknya permintaan dari klien, kami bertiga mulai mengalami kesulitan memenuhi permintaan dari klien yang begitu banyak karena keterbatasan karyawan ksaat itu juga kami melakukan perekrutan karyawan hingga menjadi 10 karyawan, pada tahun 2010 klien 41Studio semakin meningkat dan mulai merambah ke dunia internasional, otomatis penambahan karyawan pun kami lakukan menjadi 15 orang hingga sekarang.

2.1.1. Visi 41Studio

Menjadi perusahaan web developer terbaik dan mampu bersaing dalam kanca industri internasional terutama dalam bidang ruby.

2.1.2. Misi 41Studio

1. Menjadi salah satu web developer andalan di bidang web developer khususnya dalam bahasa programming ruby.
2. Menjadikan bahasa programming ruby sebagai bahasa pemrograman yang banyak dipakai oleh para programmer Indonesia.
3. Menjadikan bahasa programming ruby lebih dikenal oleh banyak kalangan.

2.2. Logo Perusahaan

Gambar 2.1 dibawah ini merupakan logo dari CV.41Studio



Gambar 2.1 Logo Perusahaan

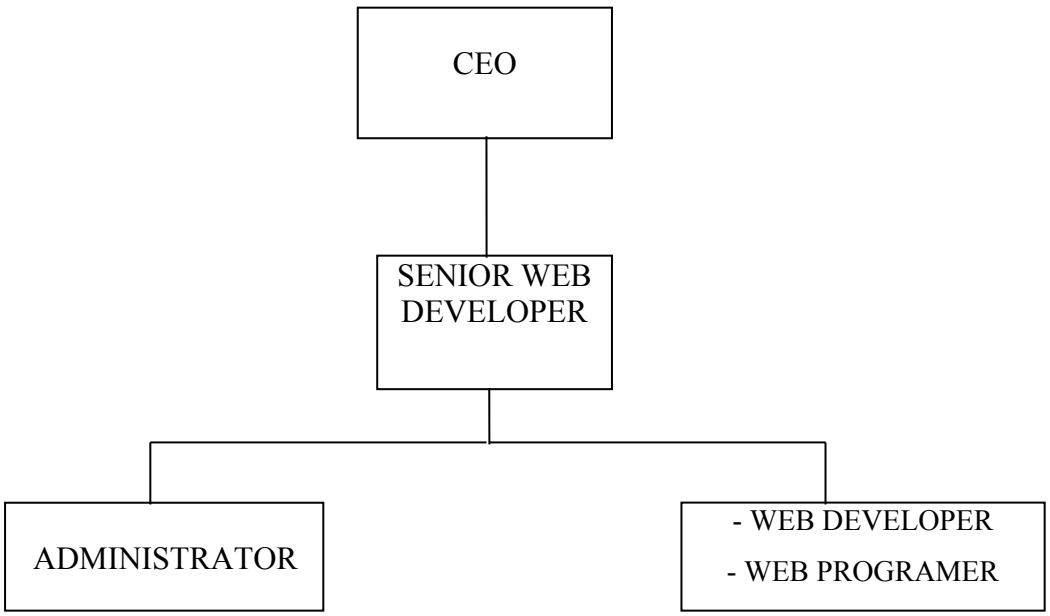
2.3. Struktur Organisasi

Struktur organisasi adalah susunan komponen-komponen (unit-unit kerja) dalam organisasi. Struktur organisasi menunjukkan adanya

pembagian kerja dan menunjukkan bagaimana fungsi-fungsi atau kegiatan-kegiatan yang berbeda-beda tersebut diintegrasikan (koordinasi). Selain daripada itu struktur organisasi juga menunjukkan spesialisasi-spesialisasi pekerjaan, saluran perintah dan penyampaian laporan.

Adapun Struktur organisasi CV. 41Studio dapat dilihat pada gambar 2.2 di bawah ini :

Struktur Organisasi CV. 41Studio



Sumber : CV. 41Studio

Gambar 2.2. Struktur Organisasi CV. 41Studio

2.4. Deskripsi Jabatan

Secara garis besar seorang pegawai teladan memiliki tugas masing-masing dan harus bertanggung jawab dalam jabatannya agar bisa bekerja secara maksimal, berikut bagian yang ada pada CV. 41Studio adalah sebagai berikut :

1. **CEO**

   CEO adalah jabatan tertinggi di suatu perusahaan dan mempunyai tugas untuk memimpin suatu perusahaan dan bertanggung jawab untuk kestabilan perusahaan tersebut. Juga Bertugas untuk :

   a. Memonitor semua kegiatan didalam seluruh projek.
   b. Memonitor seluruh kegiatan karyawan.
   c. Kebijakan perusahaan, menentukan apakah karyawan yang melamar kerja berhak bekerja atau tidak di perusahaan.

2. **Senior Web Developer**

   Seorang Senior Web Developer memberikan arahan kepada web developer lain seperti:

   a. Kegiatan teknis, proyek manager
   b. Memberi arahan pada Web programmer
   c. Memberi arahan terhadap Web developer
   d. Memberi arahan terhadap Administator

3. **Web Developer**

   Seorang Web Developer bertugas untuk menentukan logika

pemrograman dan produksi website.

4. **Web Programmer**

   Seorang Web Programer bertugas untuk menentukan logika pemrograman dan produksi website.

5. **Administrator**

   Seorang Administator mempunyai tugas untuk mengelola keungan dalam perusahaan dan mengelola recuitmen karyawan baru.

2.5. Aspek Kegiatan Perusahaan

Pada akhir-akhir ini rails menjadi fenomena di seluruh dunia dan yang lebih penting rails menawarkan teknologi tertinggi untuk pelaksanaan berbagai aplikasi yang di sebut web.2.0 dan dengan memanfaatkan ini kami mekhusukan diri dalam proyek-proyek yang menggabungkan desain interaktif yang indah dengan teknologi cerdas.

Fokus kami terutama dalam pengembangan web, desain web, sistem management konten, aplikasi mobile , media sosial, SEO/SEM, penempatan media.mencari sebuah team yang handal dan berbakat besar dengan gairah untuk memberi hasil terbaik untuk menagani proyek anda, anda berada di tangan yang tepat.

2.6. Layanan Utama

Dalam perusahaan ini melayani permintaan seorang client dalam pembuatan applikasi aplikasi seperti :

1. **Web Development**

   41Studio.Inc memberikan layanan pengembangan website dan software berbasis internet. Layanan kami meliputi pengembangan website berbasis 2.0, aplikasi mobile untuk Blackberry, iPhone dan Android, optimalisasi website di internet (search engine optimization), integrasi dengan e-commerce system dan penerimaan pembayaran kartu kredit secara online, serta pengembangan sistem berbasis web seperti CRM dan Knowledge management.

2. **Mobile Application Development**

   Selain web development 41Studio.Inc juga memberikan layanan pengembangan software dan web berbasiskan mobile SmartPhone meliputi pengembangan aplikasi mobile untuk Blackberry, iPhone, Android, iPad, Blackbook dan GalaxyTab.

3. **Search Engine Optimization (SEO)**

   Selain web development 41Studio.Inc juga memberikan layanan untuk menaikkan rating situs pencarian (seperti google, yahoo, ataupun lainnya). Dalam hal ini kami menerapkan metode yang legal dan terjamin (tidak merusak reputasi situs).

2.7. Data Umum Perusahaan

a. Nama Perusahaan :

41Studio

b. Alamat Perusahaan :

Jl. Purba Endah 1 No. 5 Sangkuriang – Cimahi

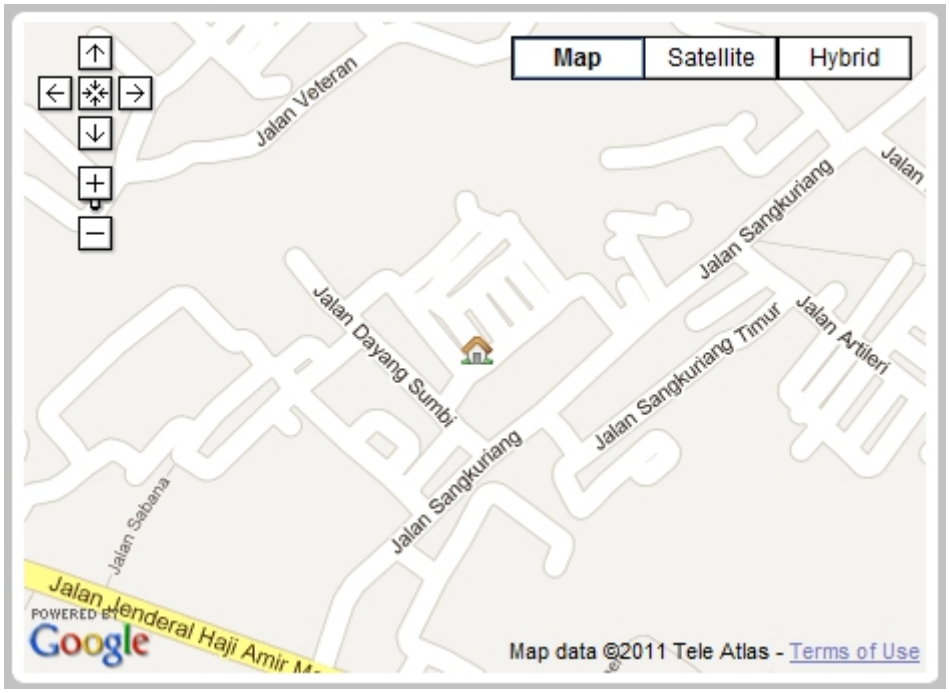
Jawa Barat, Indonesia 40511

c. Telephone : +62 81932078901

d. Fax : +62 22 6651606

2.7.1. Peta Lokasi

Gambar 2.3 merupakan tata letak lokasi 41studio



Gambar 2.3 Peta Lokasi Perusahaan 41studio

2.8. Landasan Teori

Landasan teori menjelaskan beberapa teori yang berkaitan dengan permasalahan yang dibahas sebagai dasar pemahaman dalam sebuah sistem serta metode yang dipakai untuk kegiatan pengembangan terhadap sistem sendiri.

2.8.1. Aplikasi penjualan online

Secara garis besar internet marketing adalah melakukan suatu tindakan pemasaran produk atau suatu jasa yang di pasarkan melalui media internet (online). Dalam Pengertian Internet Marketing di sebut juga e-marketing atau e-pemasaran (electronic) dan tidak jauh berbeda dengan pemasaran secara offline. Cuma bedanya kita melakukan kegiatan itu memanfaatkan media internet (online). *Internet Marketing* (emarketing) is the process of growing and promoting an organization using online media." (Wikipedia)

Jadi seperti itulah definisi dari Wikipedia, yang sering dijadikan rujukan otoritatif bagi para netter. Melanjutkan definisi itu, wikipedia menuliskan bahwa metode di dalam emarketing termasuk: optimasi dari mesin pencari, penayangan iklan, pemasaran via e-mail, affiliate

marketing, interactive advertising and viral marketing.

Kiranya definisi itu sangat cocok untuk perusahaan yang berbasis dunia mata yang menjadikan Internet hanya sebagai sebuah media. Sebagai "sarana baru" atau "pasar baru". Disini internet hanya sebagian kecil dari sebuah upaya marketing perusahaan yang jangkauannya sangat luas. Dalam pengertian ini, emarketing mungkin sangat dekat dengan fungsi promosi dan penjualan (sales) di dalam pengertian marketing tradisional.

Akan tetapi bagi perusahaan-perusahaan yang beroperasi sepenuhnya di Internet (pasar, produk, layanan, interaksi, transaksi), seperti perusahaan mesin pencari (search engine), layanan hosting, social networking, dll, pengertian emarketing tersebut terlalu sempit. Bahkan, untuk sebuah situs pribadi atau blog yang sepenuhnya hidup di dunia virtual, kebutuhan akan emarketing jauh lebih luas daripada sekedar promosi dan penjualan.

Sedangkan pengertian tentang E-Marketing menurut Armstrong dan Kottler (2004:74) adalah sebagai berikut: "*E-Marketing is the marketing side of E-Commerce, it consists of company efforts to communicate abaout, promote and sell products and services over the internet*." Yang bisa diartikan sebagai berikut: E-Marketing adalah sisi pemasaran dari E-Commerce , yang terdiri dari kerja dari perusahaan untuk mengkomunikasikan sesuatu, mempromosikan, dan menjual barang

dan jasa melalui internet.

Menurut American Marketing Association yang dikutip oleh Kleindl dan Burrow (2005) marketing adalah proses perencanaan dan pelaksanaan dari ide atau pemikiran konsep, harga, promosi dan distribusi. Marketing dapat diartikan lebih sederhana yakni pembangunan dan pemeliharaan hubungan yang saling memuaskan antara perusahaan dan konsumen.

Saat ini marketing telah berkembang seiring dengan perkembangan teknologi. Aktivitas marketing menjadi lebih luas dengan adanya internet. Penggunaan internet dan fasilitas yang ada di dalam internet untuk melakukan aktivitas marketing dikenal sebagai e-marketing.

Menurut Boone dan Kurtz (2005) e-marketing adalah salah satu komponen dalam **e-commerce** dengan kepentingan khusus oleh marketer, yakni strategi proses pembuatan, pendistribusian, promosi, dan penetapan harga barang dan jasa kepada pangsapasar internet atau melalui peralatan digital lain. Sedangkan menurut Strauss dan Frost (2001) e-marketing adalah penggunaan data dan aplikasi elektronik untuk perencanaan dan pelaksanaan konsep, distribusi, promosi, dan penetapan harga untuk menciptakan pertukaran yang memuaskan tujuan individu dan organisasi.

Keuntungan yang dapat diberikan dengan adanya penggunaan E-marketing ini bagi perusahaan yaitu:

a. Mampu menjangkau berbagai konsumen dalam suatu lingkungan yang belum dipenuhi oleh pesaing.

b. Target adalah konsumen yang telah terbagi ke dalam kelompok dan mengembangkan dialog berkelanjutan.

2.8.2. Perancangan Perangkat Lunak

Perancangan merupakan suatu hal yang sangat penting dalam pembuatan basis data. Permasalahan yang dihadapi pada waktu perancangan yaitu bagaimana basis data yang akan dibangun ini dapat memenuhi kebutuhan saat ini dan masa yang akan datang. Untuk itu diperlukan perancangan basis data baik secara fisik maupun secara konseptualnya. Perancangan konseptual akan menunjukkan entity dan relasinya berdasarkan proses yang diiginkan oleh organsisasinya. Untuk menentukan entity dan relasinya perlu dilakukan analisis data tentang informasi yang ada dalam spesifikasi di masa yang akan datang.

2.8.3. Basis Data

2.8.3.1. Pengertian Basis Data

Basis data adalah sebuah cara mendokumentasikan berbagai macam data yang kemudian dimanajemenkan dengan sebuah sistem untuk kemudian disimpan dalam sebuah media penyimpanan. Dengan demikian data-data tersebut dapat diakses dengan mudah dan cepat.

Dalam basis data, data yang ada tidak hanya disimpan begitu saja

dalam suatu media penyimpanan, akan tetapi dikelola dengan sistem pengaturan basis data yang sering disebut Database Management Sistem (DBMS). Dengan begitu pengaksesan data dapat dilakukan dengan mudah dan cepat oleh pengguna.

Berikut tujuan pemanfaatan basis data :

1. Kecepatan dan kemudahan (*Speed*)
2. Efisiensi ruang penyimpanan (*Space*)
3. Keakuratan (*Accuracy*)
4. Ketersediaan (*Availability*)
5. Kelengkapan (*Completeness*)
6. Keamanan (*Security*)
7. Kebersamaan (*Sharability*)

2.8.3.2. Bahasa Basis Data

Dua macam bahasa yang digunakan untuk mengelola dan mengorganisasikan data, yaitu :

1. Data Definition Language (DDL)

   DDL merupakan bentuk bahasa yang digunakan untuk mendefinisikan struktur tabel, sehingga dengan menggunakan bentuk bahasa ini sebuah database dan tabel dapat tercipta.

2. Data Manipulation Language (DML)

   DML digunakan untuk memanipulasi data yang ada dalam suatu tabel. Perintah yang umum dilakukan adalah:

a. SELECT, untuk menampilkan data.
b. INSERT, untuk menambahkan data baru.
c. UPDATE, untuk mengubah data yang sudah ada.
d. DELETE, untuk menghapus data.

2.8.3.3. Tahap Perancangan Basis Data

Perancangan basis data dimaksudkan untuk mengetahui dan menentukan field-field apa saja yang akan dibutuhkan untuk membangun suatu tabel sebagai dasar pembuatan basis data.

2.8.4. Unified Modeling Language (UML)

UML (Unified Modeling Language) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi.

UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C. Seperti

bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax* /semantik.

2.8.4.1. Diagram-Diagram Dalam UML

Untuk mendapatkan banyak pandangan terhadap sistem informasi yang akan dibangun, UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem. Ada beberapa diagram yang disediakan dalam UML antara lain :

a. Diagram Use Case *(use case diagram)*
b. Diagram Aktivitas *(activity diagram)*
c. Diagram Sekuensial *(sequence diagram)*
d. Diagram Kolaborasi *(collaboration diagram)*
e. Diagram Kelas *(class diagram)*
f. Diagram Statechart
g. Diagram Komponen *(componen diagram)*
h. Diagram Deployment

2.8.4.2. Use Case Diagram

Diagram use case atau *use case diagram* menyajikan interaksi antara use case dan actor. Dimana, aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun. Use case menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan pemakai.

2.8.4.3. Activity Diagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi ppada beberapa eksekusi.

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas

2.8.4.4. Sequence Diagram

Diagram sekuensial atau *sequence diagram* digunakan untuk menunjukkan aliran fungsionalitas dalam *use case*. Diagram sekuensial adalah diagram yang disusun berdasarkan urutan waktu. Kita membaca diagram sekuensial dari atas ke bawah. Setiap diagram sekuensial mempresentasikan suatu aliran dari beberapa aliran di dalam *use case*. Kita dapat membaca diagram ini dengan memperhatikan obyek-obyek dan pesan-pesan yang ada di diagram. Obyek yang terlibat dalam aliran ditunjukkan dengan bujur sangkar yang ada di atas diagram

2.8.5. Ruby On Rails (ROR)

Ruby adalah bahasa pemrograman dinamis berbasis skrip yang

berorientasi obyek. Tujuan dari ruby adalah menggabungkan kelebihan dari semua bahasa-bahasa pemrograman skrip yang ada di dunia. Ruby ditulis dengan bahasa pemrograman C dengan kemampuan dasar seperti Perl dan Python.

Pencipta Ruby, Yukihiro "matz" Matsumoto, menggabungkan bagian-bagian dari bahasa-bahasa favorit beliau (Perl, Smalltalk, Eiffel, Ada dan Lisp) untuk membentuk bahasa baru yang seimbang antara pemrograman fungsional dengan pemrograman imperative.

Inti dari bahasa pemrograman *Ruby* yaitu semua objek. Setiap informasi dan kode dapat diberi *property* dan *action*. Dalam pemrograman berorientasi objek setiap pemanggilan *property* dan *action* dilakukan melalui metode. *Ruby* juga merupakan bahasa yang fleksibel dikarenakan kita dapat mengubah bagianbagian dari *Ruby*secara bebas sehingga *programmer* tidak merasa dibatasi.

Ruby on Rails merupakan *Ruby-based framework* yang menggunakan pendekatan *Model View Controller* (MVC). Jika ingin membangun suatu *website* dengan menggunakan *Ruby*, maka dibutuhkan *Rails*. *Rails* merupakan salah satu framework *Ruby* yang paling populer sekarang ini. Oleh karena itu dinamakan *Ruby on Rails*. Dilihat dari namanya sering ada salah pengertian bahwa bukan *Ruby* yang digunakan pada *Rails* tetapi *Rails* yang digunakan pada Ruby. Rails menggunakan *Model View Controller* (MVC) sebagai *design pattern*-nya. Struktur penyusun MVC ada 3 jenis yang saling bekerja sama, yaitu:

Model: mewakili data yang diproses oleh suatu aplikasi. Menyediakan suatu *link* ke penyimpanan data.

View: representasi visual dari *user interface* atau bisa dibilang tampilan suatu aplikasi.

Controller: mewakili *control flow logic*. *Keputusan* mengenai *view* mana yang harus dipanggil untuk menampilkan data tertentu, bagian mana dari *model* yang harus yang di*update* ditangani oleh *controller*. Berperan sebagai perantara dari aplikasi dan mengambil tiap *request*. Lalu memanggil *model* yang cocok untuk meng-*update* atau mengambil data. Setelah itu memilih *view* yang cocok untuk menampilkan data.

Kelebihan Ruby

a. Sintaks sederhana.

b. OOP (*Object Oriented Programming* - Pemrogram Berorientasi Objek), yang artinya semua aspek yang terdapat di Java adalah Objek. Java merupakan salah satu bahasa pemrograman berbasis objek secara murni.

c. Memiliki *exception handling* yang baik.

d. Bahasa Pemrograman Berorientasi Objek.

e. Mengusung konsep *single inheritance*.

f. *Open source*, siapa saja dapat menggunakannya dengan cuma-cuma

dan dapat ikut berpartisipasi mengembangkannya.

g. Memiliki *garbage collector* yang secara otomatis akan menghapus informasi tak terpakai dari memori.

2.8.6. MySql

MySQL (My Structure Query Language) adalah sebuah program pembuat database yang bersifat open source. MySQL sebenarnya produk yang berjalan pada platform Linux. Karena sifatnya open source, maka MySQL dapat dijalankan pada semua platform baik Windows maupun Linux. Selain itu, MySQL juga merupakan program pengakses database yang bersifat jaringan sehingga dapat digunakan untuk aplikasi Multi User (Banyak Pengguna). Saat ini database MySQL telah digunakan hampir oleh semua programer database, apalagi dalam pemograman web.

Kelebihan lain dari MySQL adalah mengguakan bahasa Query standar yang dimiliki SQL (Structure Query Language). SQL adalah suatu bahasa permintaan yang terstruktur yang telah di standarkan untuk semua program pengakses database seperti Oracle, Posgres SQL, SQL Server, dan lain-lain.

2.8.7. SQLite3

SQLite merupakan sebuah sistem manajemen basisdata relasional yang bersifat ACID-compliant dan memiliki ukuran pustaka kode yang relatif kecil, ditulis dalam bahasa C. SQLite merupakan proyek yang bersifat public domain yang dikerjakan oleh D. Richard Hipp.

Tidak seperti pada paradigma client-server umumnya, Inti SQLite bukanlah sebuah sistem yang mandiri yang berkomunikasi dengan sebuah program, melainkan sebagai bagian integral dari sebuah program secara keseluruhan. Sehingga protokol komunikasi utama yang digunakan adalah melalui pemanggilan API secara langsung melalui bahasa pemrograman. Mekanisme seperti ini tentunya membawa keuntungan karena dapat mereduksi *overhead*, *latency times*, dan secara keseluruhan lebih sederhana. Seluruh elemen basisdata (definisi data, tabel, indeks, dan data) disimpan sebagai sebuah file. Kesederhanaan dari sisi disain tersebut bisa diraih dengan cara mengunci keseluruhan file basis data pada saat sebuah transaksi dimulai.

2.8.7.1. Fitur –Fitur SQLite

Pustaka SQLite mengimplementasikan hampir seluruh elemen-elemen standar yang berlaku pada SQL-92, termasuk transaksi yang bersifat *atomic*, konsistensi basisdata, isolasi, dan durabilitas (dalam bahasa inggris lebih sering disebut ACID), trigger, dan kueri-kueri yang kompleks. Tidak ada pengecekan tipe sehingga data bisa dientrikan dalam bentuk string untuk sebuah kolom bertipe integer. Beberapa kalangan melihat hal ini sebagai sebuah inovasi yang menambah nilai guna dari sebuah basisdata, utamanya ketika digunakan dalam bahasa pemrograman berbasis script (PHP, Perl, phyton, Ruby), sementara kalangan lain melihat hal tersebut sebagai sebuah kekurangan.

Beberapa proses ataupun thread dapat berjalan secara bersamaan dan mengakses basisdata yang sama tanpa mengalami masalah. Hal ini disebabkan karena akes baca data dilakukan secara paralel. Sementara itu akses tulis data hanya bisa dilakukan jika tidak ada proses tulis lain yang sedang dilakukan; jika tidak, proses tulis tersebut akan gagal dan mengembalikan kode kesalahan (atau bisa juga secara otomatis akan mencobanya kembali sampai sejumlah nilai waktu yang ditentukan habis). Hanya saja ketika sebuah tabel temporer dibuat, mekanisme penguncian pada proses multithread akan menyebabkan masalah. Update yang terkini (versi 3.3.4) dikatakan telah memperbaiki masalah ini.

Sebuah program yang mandiri dinamakan sqlite disediakan dan bisa digunakan untuk mengeksekusi kueri dan memanajemen file-file basisdata SQLite. Program tersebut juga merupakan contoh implementasi penulisan aplikasi yang menggunakan pustaka SQLite.

2.8.7.2. SQLite Database server

Pada prinsipnya SQLite merupakan library C yang diimplementasikan secara embeddable (tertanam) sebagai SQL database engine. Hal ini akan memungkinkan program yang berhubungan dengan SQLite dapat mengakses database tanpa menjalankan proses RDBMS secaraterpisah, sebagaimana pada umumnya ketika anda menggunakan MySQL Server, PostgreSQL, Oracle, dan sebagainya.

Kemasan praktis inilah yang memberikan banyak sekali

keuntungan, di mana anda tidak perlu melakukan manajemen database server terpisah. Selain itu, ukuran kecil yang sengaja didesain pada SQLite memungkinkan kita untuk membuat aplikasi yang ringan dengan kinerja tinggi.

2.8.7.3. Kelebihan SQLite

Sqlite merupakan salah satu embbeded database yang wajib dipertimbangkan untuk digunakan pada aplikasi bukan enterprise karena memiliki beberapa kelebihan yaitu :

a. Secara umum cukup stabil.
b. Perfomanya lebih efesien dan cepat dibandingkan database yang lain.
c. Sedikit menggunakan memory, hanya membutuhkan sigle library untuk mengakses database.
d. Berjalan dibanyak platform yang berbeda, dan dapat dipindahkan dengan mudah tanpa setting administrasi yang rumit.
e. Memenuhi standart ACID (Atomicity, Consistensy, Isolation, dan Durability).
f. Mendukung ANSI 92 SQl Standar.
g. Mempunyai API untuk bahasa C++, PHP, Perl, Phyton,Tcl dan Ruby.
h. License public domain, free untuk redistributed.

2.8.7.4. Kekurangan SQLite

Meskipun merupakan database server yang cukup sederhana dengan beberapa fasilitas, bukan berarti SQLite tidak memiliki kekurangan. Beberapa fasilitas yang tidak didukung oleh SQLite sengaja ditiadakan untuk mempertahankan SQLite sebagai embeddable database server. Berikut beberapa fasilitas yang umum ada pada database server dan tidak dimiliki atau tidak didukung oleh SQLite.

a. CHECK constraint

   Fasilitas ini umumnya digunakan sebagai pemeriksa, namun ditiadakan karena masih dapat digantikan oleh NOT NULL serta UNIQUE.

b. Variabel subquery

   Meskipun memiliki arti yang penting, penggunaan subquery atau perintah query di dalam query ini tidak didukung.

c. Foreign Key Constraint

   Sama halnya dengan CHECK constraint, fasilitas ini sengaja dihilangkan oleh SQLite.

d. Alter Table

   Oleh karena tidak memiliki fasilitas ini, maka untuk mengubah struktur tabel harus dilakukan dengan menghapus tabel kemudian membuat ulang.

e. Grant dan Revoke

   Fasilitas ini cukup penting untuk keamanan, namun karena SQLite melakukan pembacaan serta penulisan pada file disk, maka hak izin

akses bergantung pada sistem operasi yang digunakan.

f. Right dan Full Outer Join

Meskipun penggunaan right serta full outer join tidak diperkenankan, akan tetapi SQLite masih menyediakan left outer join.

g. Penulisan pada View

SQLite hanya memperbolehkan penggunaan view untuk dapat dibaca saja, dengan demikian anda tidak dapat melakukan modifikasi pada view. Sebagai gantinya, SQLite menyediakan trigger yang dapat digunakan untuk memodifikasi view.

2.8.8. Webrick 1.3.1

Webrick adalah library Ruby menyediakan layanan web server sederhana HTTP. Webrick terutama ditulis oleh Masayoshi Takahashi dan Yuuzou Gotou, dengan kontribusi dari pengembang lain melalui model open source pengembangan perangkat lunak. Server juga menyediakan kode untuk layanan server sederhana lain selain HTTP.

Hal ini digunakan oleh Ruby on Rails kerangka untuk menguji aplikasi dalam lingkungan pengembangan. Meskipun popularitasnya, webrick telah memperoleh ketenaran beberapa karena kode ini agak tidak terdokumentasikan.

2.8.9. Ruby Gems

Rubygems adalah manajer paket untuk bahasa pemrograman Ruby

yang menyediakan format standar untuk mendistribusikan program Ruby dan perpustakaan (dalam format mandiri disebut "Ruby"), alat yang dirancang untuk mudah mengelola instalasi permata, dan server untuk mendistribusikan mereka. Hal ini analog dengan EasyInstall untuk bahasa pemrograman Python. Rubygems sekarang merupakan bagian dari perpustakaan standar dari Ruby versi 1.9.