BAB II
TINJAUAN PUSTAKA

2. 1 Profil Bank Indonesia

Pengembangan aplikasi pengolahan data rumah dinas ini dilakukan di Bank Indonesia cabang Bandung bagian Logistik. Aplikasi yang dikembangkan akan digunakan oleh pegawai staf Logistik. Bank Indonesia Bandung memiliki sejarah , logo, stuktur organisasi, badan hukum serta memiliki sistem kerja yang berbeda dari setiap divisi.

2. 1. 1 Sejarah Bank Indonesia

Bank Indonesia pertama kali berdiri pada tahun 1828 dengan nama *De Javasche Bank* yang didirikan oleh Pemerintah Hindia Belanda sebagai bank sirkulasiyang bertugas mencetak dan mengedarkan uang.Undang-Undang Pokok Bank Indonesia tahun 1953 menetapkan pendirian Bank Indonesiauntuk menggantikan fungsi *De Javasche Bank* sebagai bank sentral, dengan tigatugas utama di bidang moneter, perbankan, dan sistem pembayaran. Di sampingitu, Bank Indonesia diberi tugas penting lain dalam hubungannya denganPemerintah dan melanjutkan fungsi bank komersial yang dilakukan oleh DJBsebelumnya.

Undang-Undang Bank Sentral tahun 1968 mengatur kedudukan dan tugas Bank Indonesiasebagai bank sentral, terpisah dari bank-bank lain yang melakukan fungsikomersial. Selain tiga tugas pokok bank sentral, Bank Indonesia juga bertugasmembantu Pemerintah sebagai agen pembangunan mendorong kelancaranproduksi dan pembangunan serta memperluas kesempatan kerja gunameningkatkan taraf hidup rakyat.

Tahun 1999 merupakan babak baru dalam sejarah Bank Indonesia, sesuai dengan UU No.23/1999 yangmenetapkan tujuan tunggal Bank Indonesia yaitu mencapai dan memeliharakestabilan nilai rupiah.Pemerintah mengeluarkan Peraturan Pemerintah Pengganti Undang-Undang (PerPPU) No.2 tahun 2008 tentang Perubahan Kedua atas Undang-Undang No.23 tahun 1999 tentang Bank Indonesia sebagai bagian dari upaya menjaga stabilitas sistem keuangan.

Amandemen dimaksudkan untuk meningkatkan ketahanan perbankan nasional dalam menghadapi krisis global melalui peningkatan aksesperbankan terhadap Fasilitas Pembiayaan Jangka Pendek dari Bank Indonesia[1].

2. 1. 2 Logo Bank Indonesia

Logo ataupun lambang dari Bank Indonesia sekarang adalah seperti gambar 2.1 sesuai dengan surat edaran intern No.11/40/INTERN tanggal 29 Juni 2009.



2.1. Logo Bank Indonesia

Sejarah perkembangan logo BI yang telah mengalami perubahan 7 kali sejak tahun 1953 hingga 2005. Logo BI yang sekarang ini ternyata mengadaptasi logo *De Javasche* Bank dengan mengubah huruf "J" menjadi huruf "I" [2].

2. 1. 3 Badan Hukum Bank Indonesia

Babak baru dalam sejarah Bank Indonesia sebagai Bank Sentral yang independen dimulai ketika sebuah undang-undang baru, yaitu UU No. 23/1999 tentang Bank Indonesia, dinyatakan berlaku pada tanggal 17 Mei 1999. Undang-undang ini memberikan status dan kedudukan sebagai suatu *lembaga negara independen* dan bebas dari campur tangan pemerintah ataupun pihak lainnya.

Sebagai suatu lembaga negara yang independen, Bank Indonesia mempunyai otonomi penuh dalam merumuskan dan melaksanakan setiap tugas dan wewenangnya sebagaimana ditentukan dalam undang-undang tersebut. Pihak luar tidak dibenarkan mencampuri pelaksanaan tugas Bank Indonesia, dan Bank Indonesia juga berkewajiban untuk menolak atau mengabaikan intervensi dalam bentuk apapun dari pihak manapun juga. Untuk lebih menjamin independensi tersebut, undang-undang ini telah memberikan kedudukan khusus kepada Bank Indonesia dalam struktur ketatanegaraan Republik Indonesia.

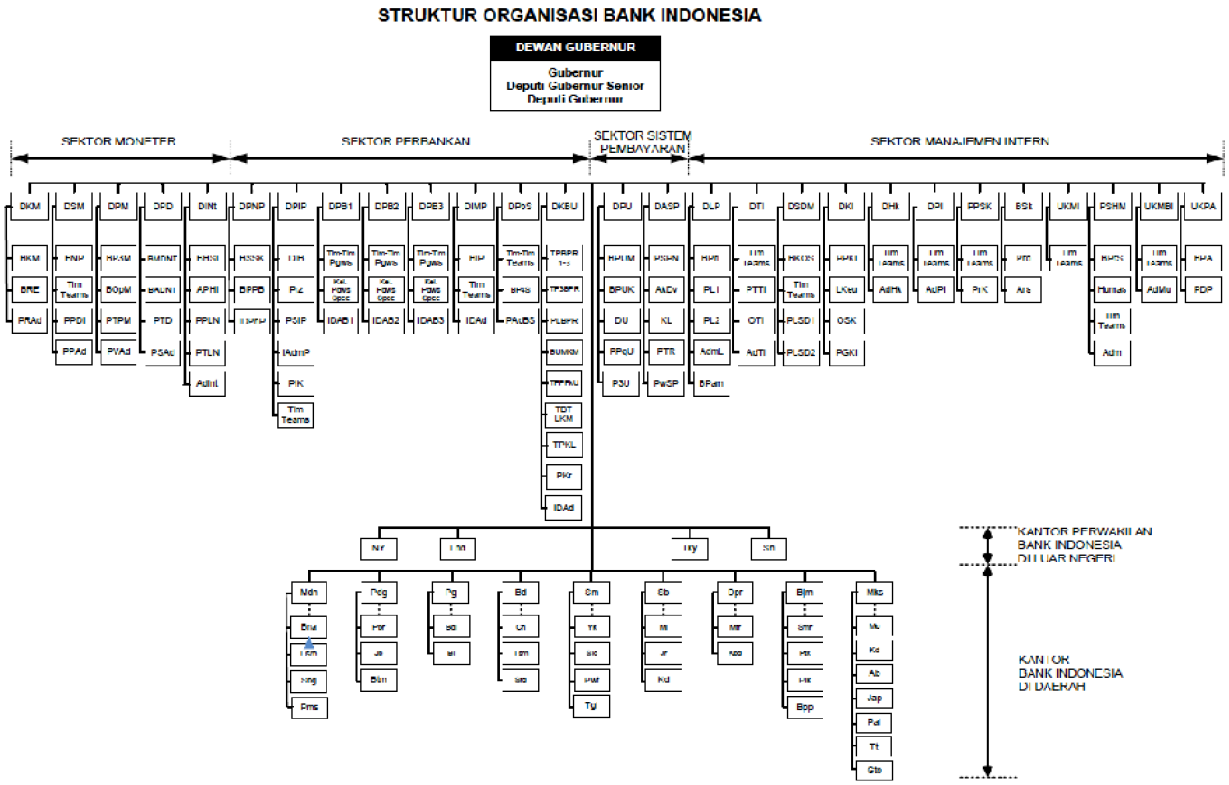
Sebagai Lembaga negara yang independen kedudukan Bank Indonesia tidak sejajar dengan Lembaga Tinggi Negara.

Kedudukan Bank Indonesia juga tidak sama dengan Departemen, karena kedudukan Bank Indonesia berada diluar Pemerintah. Status dan kedudukan yang khusus tersebut diperlukan agar Bank Indonesia dapat melaksanakan peran dan fungsinya sebagai otoritas moneter secara lebih efektif dan efisien.

Status Bank Indonesia baik sebagai badan hukum publik maupun badan hukum perdata ditetapkan dengan undang-undang.Sebagai badan hukum publik Bank Indonesia berwenang menetapkan peraturan-peraturan hukum yang merupakan pelaksanaan dari undang-undang yang mengikat seluruh masyarakat luas sesuai dengan tugas dan wewenangnya. Sebagai badan hukum perdata, Bank Indonesia dapat bertindak untuk dan atas nama sendiri di dalam maupun di luar pengadilan [6].

2. 1. 4 Struktur Organisasi Bank Indonesia

Setiap instansi ataupun setiap organisasi tentunya memiliki struktur organisasi yang merupakan suatu pembangun dan pengurus dari organisasi. Struktur organisasi Bank Indonesia secara umum sesuai dengan surat edaran No.9/12/INTERN tanggal 30 Maret 2007 dapat dilihat pada gambar 2.2. [6].



Gambar 2.2. Struktur Organisasi Bank Indonesia

2. 1. 5 *Job Descrption* Manajemen Logistik Bank Indonesia

Bank Indonesia memiliki bagian atau divisi yang memiliki tugasnya masing-masing dan kegiatan manajemen Logistik Bank Indonesia sendiri terdiri dari :

1. *Profilling*

Kegiatan ini dilakukan pada awal tahun anggaran atau pada saat akan dimulai proses kegiatan pengadaan. Tujuan dari kegiatan ini untuk:

a. Memperoleh barang dan/atau jasa yang dibutuhkan dalam waktu yang lebih cepat dengan harga dan kualitas yang baik.
b. Memperoleh standar/spesifikasi barang dan /atau jasa untuk digunakan pedoman bagi satker.
c. Memperoleh penyedia barang dan/atau jasa yang dapat memberikan tingkat layanan yang efektif dan efisien dan berdasarkan hasil survei mempunyai kinerja yang baik.
d. Memperoleh rekomendasi calon penyedia barang dan/atau jasa untuk proses pemilihan langsung/seleksi langsung atau penunjukan langsung.

Langkah-langkah Profilling

Invetarisasi Data → Pengumuman → Analisa Pasar

2. Pengadaan

Pelaksana Pengadaan sesuai dengan gambar 2.1.5 adalah :

a. Panitia Pengadaan dibentuk untuk melaksanakan Pengadaan dengan nilai di atas Rp250 juta.
b. Pengadaan dengan nilai sampai dengan Rp. 250 juta dapat dilaksanakan dengan membentuk Panitia atau dengan menunjuk seorang pegawai.
c. Pembentukan Panitia dikecualikan untuk pengadaan buku yang dilakukan melalui media elektronik dan pengadaan yang bersifat mendesak



Gambar 2.3. Pelaksanaan Pengadaan

Skala Kewenangan Pemutus Pengadaan ditunjukan dengan tabel 2.1 seperti berikut :

Tabel 2.1. Pemutusan Pengadaan

NILAI PENGADAAN	PEMUTUS PENGADAN
Lebih dari 50 Miliar Rupiah	GUBERNUR
Rp 10 Miliar – Rp 50 Miliar	ANGGOTA DEWAN GUBERNUR YANG MEMBIDANGI
Rp 5 Miliar - Rp 10 Miliar	G.VIII
Rp 1 Miliar – Rp 5 Miliar	G.VII
Rp 250 Juta – Rp 1 Miliar	G.VI
Rp 50 Juta – Rp 250 Juta	G.V
Sampai dengan Rp 50 Juta	G.IV

3. Pemeliharaan

Setiap barang milik BI dan belum dilakukan penghapusan wajib dilakukan pemeliharaan dan pemeliharaan dilaksanakan oleh Pelaksana CM, KBI atau KPwBI. Selain itu, Pemeliharaan dapat dilaksanakan sendiri atau dengan memanfaatkan jasa pihak lain yang profesional.

Pemeliharaan dengan cara memanfaatkan jasa pihak lain yang profesional merupakan penyedia jasa lain yang dipilih dengan cara pelelangan, pemilihan langsung atau penunjukan langsung serta pemeliharaan yang terkait dengan asset yang dimiliki BI dapat pula dilakukan melalui kerja sama dengan masyarakat atau aparat desa/lingkungan setempat.

Tujuan dari pemeliharaan yaitu :

a. Memperpanjang usia teknis barang.
b. Meningkatkan dan/atau mempertahankan kualitas barang.
c. Menjamin barang agar selalu dalam kondisi baik dan siap pakai.
d. Menjamin kelancaran, kenyamanan, keamanan, keselamatan dan kesehatan kerja bagi pengguna dalam melaksanakan aktivitas sehari-hari.
e. Menjamin akses dan penggunaan asset yang dimilki Bank Indonesia berupa tanah dan bangunan.

4. Penatausahaan

Penatausahaan memiliki kegian sebagai berikut :

a. Mencatat keberadaan, kondisi dan nilai barang milik BI dan barang sewa secara akurat untuk mewujudkan tertib administrasi MLBI.
b. Menghitung dan melakukan pembukuan penyusutan aktiva tetap.
c. Menghasilkan dokumen dan daftar aktiva tetap yang berisi harga perolehan, tahun perolehan, penyusutan, akumulasi penyusutan dan nilai buku untuk keperluan data pendukung dalam penyusunan laporan keuangan BI.
d. Setiap barang milik Bank Indonesia dan barang sewa wajib dilakukan penatausahaan

Penatausahaan di BISAIL wajib dilakukan atas setiap :

a. Barang milik BI
b. Barang sewa dengan jangka waktu 1 tahun atau lebih.
c. Penatausahaan dilakukan oleh Pelaksana CM, KBI dan KPwBI.

5. Pemanfaatan

Proses ini dilakukan untuk mendayagunakan tanah, bangunan, inventaris dan/atau inventaris rutin milik BI yang belum dimanfaatkan dan/atau mengoptimalkan pengelolaan tanah, bangunan, inventaris dan/atau inventaris rutin milik Bank Indonesia.

6. Penghapusan

Penghapusan yaitu membebaskan dari pertanggungjawaban secara fisik dan administratif barang berupa tanah, bangunan, inventaris, inventaris rutin dan/atau barang persediaan, yang kondisinya secara teknis dan ekonomis tidak dapat dimanfaatkan lagi untuk menunjang pelaksanaan tugas BI.

Penghapusan bisa lakukan dengan cara :

a. Fisik : menyerahkan Barang kepada pihak lain atau memusnahkan Barang.
b. Administratif : penghapusan penatausahaan Barang dari BISAIL, atau neraca BI dan BISAIL.

Langkah-Langkah Penghapusan

a. Dijual
b. Ditukar
c. Dihibahkan
d. Dimusnahkan [5].

2. 2 Landasan Teori

Pengembangan aplikasi data pengolahan rumah dinas memerlukan faktor-faktor yang mendukung. Faktor-faktor tersebut merupakan landasan teori yang akan digunakan dalam proses pengerjaan. Pada landasan teori pengembangan aplikasi rumah dinas dijelaskan mengenai pemakain Java sebagai bahasa pemrogramannya, Netbeans 6.9 sebagai *compiler*, *Ireport, UML*, dan menggunakan *MySql* untuk *tools* database yang digunakan.

2. 2. 1 Java

Java adalah bahasa pemrograman berorientasi obyek yang dikembangkan oleh *Sun Microsystems* sejak tahun 1991. Bahasa ini dikembangkan dengan model yang mirip dengan bahasa C++ dan *Smalltalk*, namun dirancang agar lebih mudah dipakai dan *platform independent*, yaitu dapat dijalankan di berbagai jenis sistem operasi dan arsitektur komputer. Bahasa ini juga dirancang untuk pemrograman di internet sehingga dirancang agar aman dan portabel.

a. Platform Independent

Platform independent berarti program yang ditulis dalam bahasa Java dapat dengan mudah dipindahkan antar berbagai jenis sistem operasi dan berbagai jenis arsitektur komputer. Aspek ini sangat penting untuk dapat mencapai tujuan Java sebagai bahasa pemrograman Internet di mana sebuah program akan dijalankan oleh berbagai jenis komputer dengan berbagai jenis sistem operasi. Sifat ini berlaku untuk level *source code* dan *binary code* dari program Java. Berbeda dengan bahasa C dan C++, semua tipe data dalam bahasa Java mempunyai ukuran yang konsisten di semua jenis *platform. Source code* program Java sendiri tidak perlu dirubah sama sekali jika Anda ingin mengkompile ulang di platform lain. Hasil dari meng-*compile source code* Java bukanlah kode mesin atau instruksi prosesor yang spesifik terhadap mesin tertentu, melainkan berupa bytecode yang berupa file berekstensi `.class`. *Bytecode* tersebut dapat langsung dieksekusi di tiap *platform* yang dengan menggunakan *Java Virtual Machine* (JVM) sebagai interpreter terhadap *bytecode* tersebut.

b. Kompilasi

Kompilasi kode program Java dilakukan menggunakan *tool command-line* yang bernama javac, atau biasa disebut kompilator Java. Tahap kompilasi ini bertujuan untuk mengonversi kode sumber ke program biner yang berisi *bytecode*, yaitu instruksi-instruksi mesin.

c. Interpretasi

Kode program Java tidak dieksekusi di komputer secara langsung, tetapi berjalan di atas komputer hipotesis yang distandardisasikan, yang disebut Java Virtual Machine. Untuk menginterpretasi bytecode, kita menggunakan *tool* bernama java, atau biasa disebut interpreter Java.

d. *Library*

Selain kompilator dan interpreter, bahasa Java sendiri memiliki *library* yang cukup besar yang dapat mempermudah Anda dalam membuat sebuah aplikasi dengan cepat. *Library* ini sudah mencakup untuk grafik, desain *user interface*, kriptografi, jaringan, suara, database, dan lain-lain.

e. Berorientasi Obyek

Java adalah bahasa pemrograman berorientasi obyek. Pemrograman berorientasi obyek secara gamblang adalah teknik untuk mengorganisasi program dan dapat dilakukan dengan hampir semua bahasa pemrograman. Namun Java sendiri telah mengimplementasikan berbagai fasilitas agar seorang programer dapat mengoptimalkan teknik pemrograman berorientasi obyek.

2. 2. 2 NetBeans

NetBeans merupakan sebuah proyek kode terbuka yang sukses dengan pengguna yang sangat luas, komunitas yang terus tumbuh, dan memiliki hampir 100 mitra (dan terus bertambah). Sun Micro*system*s mendirikan proyek kode terbuka NetBeans pada bulan Juni 2000 dan terus menjadi sponsor utama.

Saat ini terdapat dua produk yaitu NetBeans IDE dan NetBeans *Platform*. The NetBeans IDE adalah sebuah lingkungan Pembangunan - sebuah kakas untuk pemrogram menulis, mengompilasi, mencari kesalahan dan menyebarkan program. Netbeans IDE ditulis dalam Java – namun dapat mendukung bahasa pemrograman lain. Terdapat banyak modul untuk memperluas Netbeans IDE. Netbeans IDE adalah sebuah produk bebas dengan tanpa batasan bagaimana digunakan [3].

2. 2. 3 iReport

IReport adalah *report* designer yang digunakan oleh Jasper Report. Karena kemiripannya dengan Crystal Report, IReport menjadi salah satu pilihan utama untuk melakukan proses desain *report* di Java. *Tool* untuk reporting yang paling populer di Java adalah iReport. iReport penggunaanya mirip dengan Crystal Report yang sudah banyak dikenal orang. Dengan iReport kita dapat membuat report dengan mudah. Cukup dengan *drag n drop* dan juga wizard yang sangat mudah digunakan. iReport sendiri tersedia dalam bentuk aplikasi tunggal dan yang paling baru adalah iReport sudah dapat diintegrasikan dengan Netbeans IDE. Artinya kita tidak perlu lagi menggunakan 2 tools yang berbeda untuk membuat aplikasi yang lengkap dengan reportnya. Dengan Netbeans IDE + iReport *plugin* maka *tools* Pembangunan aplikasi kita udah lengkap [10].

2. 2. 4 Unified Modelling Language (UML)

UML sebagai sebuah bahasa yang memberikan *vocabulary* dan tatanan penulisankata-kata dalam *MS Word* untuk kegunaan komunikasi. Sebuah bahasa model adalah sebuah bahasa yang mempunyai *vocabulary* dan konsep tatanan / aturan penulisan serta secara fisik mempresentasikan dari sebuah sistem. Seperti halnya UML adalah sebuah bahasa standard untuk pengembangan sebuah software yang dapat menyampaikan bagaimana membuat dan membentuk model-model, tetapi tidak menyampaikan apa dan kapan model yang seharusnya dibuat yang merupakan salah satu proses implementasi pengembangan *software*.

UML tidak hanya merupakan sebuah bahasa pemograman visual saja, namun juga dapat secara langsung dihubungkan ke berbagai bahasa pemograman, seperti JAVA, C++, Visual Basic, atau bahkan dihubungkan secara langsung ke dalam sebuah *object-oriented database*. Begitu juga mengenai pendokumentasian dapat dilakukan seperti *requirements*, arsitektur, *design, source code, project plan, tests, dan prototypes* [4]. UML mendefinisikan diagram-diagram sebagai berikut:

a. *Use case* Diagram

Use case diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case* diagram terdiri atas diagram untuk *use case* dan aktor. Aktor merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi. *Use case* merepresentasikan operasi-operasi yang dilakukan oleh aktor. *Use case* digambarkan berbentuk elips dengan nama operasi dituliskan di dalamnya. Aktor yang melakukan operasi dihubungkan dengan garis lurus ke *use case* [7].

a. *Class diagram*

Class diagram dapat membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class diagram* banyak memperhatikan hubungan antarkelas dan penjelasan detail tiap kelas dalam pemodelan desain (dalam *logical view*) dari suatu sistem. Selama proses analisa, *class diagram* memperhatikan aturan-aturan dan tanggung jawab

entitas yang menentukan perilaku sistem. Selama tahap desain, *class diagram* berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat [7].

Hubungan antar kelas dalam pemodelan dengan *tools* UML dijelaskan sebagai berikut :

a. Asosiasi, yaitu hubungan statis antar *class*. Umumnya menggambarkan *class* yang memiliki atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain. Panah *navigability* menunjukkan arah *query* antar *class*.
b. Agregasi, yaitu hubungan yang menyatakan bagian ("terdiri atas..").
c. Pewarisan, yaitu hubungan hirarkis antar *class*. *Class* dapat diturunkan dari *class*lain dan mewarisi semua atribut dan metoda *class* asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari *class* yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
d. Hubungan dinamis, yaitu rangkaian pesan (*message*) yang di-*passing* dari satu *class* kepada *class* lain. Hubungan dinamis dapat digambarkan dengan menggunakan *sequence diagram* yang akan dijelaskan kemudian.

b. *Statechart diagram*

Statechart diagram digunakan untuk memodelkan perilaku dinamis satu kelas atau obyek. *Statechart diagram* memperlihatkan urutan keadaan sesaat (state) yang dilalui sebuah obyek, Kejadian yang menyebabkan sebuah transisi dari suatu state atau aktivitas kepada yang lainnya. *Statechart diagram* khusus digunakan untuk memodelkan tahap-tahap diskrit dari sebuah siklus hidup obyek, sedangkan *Activity diagram* paling cocok untuk memodelkan urutan aktifitas dalam suatu proses [7].

c. *Activity diagram*

Activity diagram merupakan state diagram khusus, di mana sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (*internal processing*). *Activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak,

tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Activity diagram dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan obyek mana yang bertanggung jawab untuk aktivitas tertentu [2].

d. *Sequence diagram*

Sequence diagram menjelaskan interaksi obyekyang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan use-case. *Sequence diagram* memperlihatkan tahap demi tahap apa yang arusterjadi untuk menghasilkan suatu didalam *use case diagram*. Tipe diagram yang digunakan sebaiknya digunakan diawal tahap desain atau analisis karena kesederhanaannya dan mudah untuk di mengerti [7].

2. 2. 5 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang multi thread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU General Public License(GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL[8].