BAB 2
TINJAUAN PUSTAKA

2.1 Profil Perusahaan

Hikmah Mebel merupakan sebuah usaha yang bergerak dalam bidang industri Mebel dan melakukan pengerjaan pembuatan Mebel jika terdapat pesanan dari pelanggan. Pelanggan dapat memesan *furniture* berdasarkan jenis produk yang tersedia di Hikmah Mebel seperti Lemari, Meja, *Kitchen Set*, Ranjang, *TV Cabinet*, dan Rak Majalah. Produk *furniture* yang dibuat, berasal dari bahan baku kayu yaitu Kayu Jati, Kayu Mahoni, Kayu Pinus, Kayu Plywood dan Kayu Blockboard, untuk pemilihan warna disediakan pilihan warna disertai jenis finishing untuk *furniture* yaitu HPL, Duco dan Melami. Hikmah Mebel telah memiliki kerja sama dengan sebuah perusahaan dan telah memiliki beberapa kontrak dengan sebuah proyek untuk pembuatan *furniture* sebuah toko, *apartement* dan lain-lain. Tidak hanya untuk bangunan apartment dan pertokoan saja, Hikmah Mebel pun menerima pesanan dari pelanggan perorangan.

2.1.1. Sejarah Perusahaan

Keberadaan usaha Hikmah Mebel Cimahi telah melewati jatuh bangun dalam usahanya. Semuanya berawal ketika Pak Agus Dodo Solphin yang menjadi seorang karyawan pada bidang Mebel. Beliau bekerja cukup lama dengan menjadi seorang karyawan, namun beliau memikirkan untuk membuka usaha Mebelnya sendiri dan akhirnya dengan keyakinan dan kerja kerasnya beliau dapat membangun dan merintis usaha Mebelnya sendiri dengan nama "Hikmah Mebel" pada tahun 1996 dan bertempat di dekat kediaman beliau di Jln. Cihanjuang – Cimahi. Pada tahun 1998- sekitar akhir tahun 2000 usaha Mebel ini sempat berhenti beroperasi dikarenakan krisis moneter yang tengah melanda Indonesia pada saat itu berdampak pada usaha Pak Agus ini, baru saja berumur 2 tahun namun harus tersendat karena ekonomi yang sedang kacau saat itu. Pemilik mulai kembali merintis usahanya seperti dari awal lagi pada awal tahun 2000 dengan dibantu rekan-rekannya.

Akhirnya saat ini Hikmah Mebel telah berdiri selama 17 tahun terhitung sejak tahun 2000 dan yang sekarang beralamatkan di Jalan Cihanjuang Gg. Haji Bakar No. 23 RT/RW: 02/20 - Cimahi.

2.1.2. Visi dan Misi

Visi dan Misi Hikmah Mebel adalah sebagai Berikut :

1. Visi

   Menjadi usaha *home* industri *furniture* yang profesional, berdaya saing dan inovatif serta mampu berkembang, sehat dan mandiri.

2. Misi

   a. Menyediakan pelayanan bermutu.
   b. Menghasilkan produk berkualitas.
   c. Menyelesaikan pesanan pelanggan tepat waktu
   d. Menggunakan bahan baku *furniture* berkualitas.
   e. Mengutamakan kepuasan pelanggan.

2.1.3. Logo Hikmah Mebel Cimahi

Logo perusahaan Hikmah Mebel Cimahi dapat dilihat pada gambar 2.1 dibawah ini.



Gambar 2. 1 Logo perusahaan Hikmah Mebel Cimahi

2.2. Landasan Teori

Untuk mendukung pembuatan laporan penelitian ini, dibutuhkan teori-teori yang berkaitan dengan penelitian yang dilakukan sebagai landasan pembuatan laporan.

2.2.1. Mebel

Kata mebel dalam bahasa Inggris, diterjemahkan menjadi *furniture*. Istilah "mebel" digunakan karena sifat bergeraknya atau mobilitasnya sebagai barang lepas di dalam interior arsitektural. Kata mebel berasal dari bahasa Perancis yaitu *meubel*, atau bahasa Jerman yaitu *mobel*. Pengertian mebel secara umum adalah benda pakai yang dapat dipindahkan, berguna bagi kegiatan hidup manusia, mulai dari duduk, tidur, bekerja, makan, bermain dan sebagainya, yang memberi kenyamanan dan keindahan bagi pemakainya[1].

2.2.2. Klasifikasi Posisi Produk

Pada dasarnya, perencanaan dan pengendalian produksi membedakan empat tipe posisi produk dalam lingkungan *manufacturing* yang masing-masing memberikan pengaruh yang berbeda terhadap proses perencanaan dan pengendalian. Empat tipe yang dimaksud adalah [2]:

a. Engineering to order

   Pelanggan menyediakan spesifikasi dari produk yang diinginkannya dan berdasarkan spesifikasi tersebut perusahaan membuat desain, menyediakan bahan, membuat part/komponen, merakit, menguji kinerja produk dan kemudian mengirim produk kepada pelanggan. Kegiatan produksi dilakukan apabila pelanggan telah dating mengajukan order. *Bills of materials*, gambar-gambar teknik (*engineering drawings*) dan perintah kerja (*job orders*) dipersiapkan perusahaan secara terpisah untuk setiap *order*[2].

b. Make to order

   Pelanggan menyediakan spesifikasi dan desain produk. Berdasarkan desain tersebut perusahaan menyediakan bahan, pembuatan part dan komponen, merakit dan mengirimkan produk kepada pelanggan. Sama seperti *engineering to order* kegiatan produksi dilakukan apabila pelanggan telah mengajukan permintaan. Karena *engineering design* disediakan oleh pelanggan maka perencanaan dan pengendalian produksi tidak mencakup kegiatan *engineering*.

c. Assembly to order

Perusahaan menyediakan sejumlah model dasar dari produk tetapi dilengkapi dengan berbagai alternatif dan variasi yang diperkirakan akan memperkaya pilihan bagi pelanggan. Pelanggan melakukan pemilihan terhadap model, variasi dan tipe produk yang diinginkannya dari alternatif yang tersedia. Kegiatan produksi dilakukan untuk membuat komponen-komponen standar dengan semua variasinya dan perakitan produk akhir dilakukan setelah pelanggan mengajukan permintaan.

d. Make to stock

Pelanggan tidak mempunyai kesempatan untuk memilih sesuai dengan seleranya tetapi membeli langsung produk yang sudah jadi dari persediaan. Kegiatan produksi dilakukan untuk mengisi persediaan yang jumlahnya dinyatakan dalam jadwal induk produksi. Jadwal induk produksi disusun berdasarkan peramalah terhadap potensi permintaan pelanggan untuk setiap produk akhir. Untuk mengantisipasi kekurangan persediaan khususnya akibat fluktuasi permintaan yang sering diluar batas antisipasi normal maka persediaan pengaman (*safety stock*) ditentukan. Isu utama dalam tipe *make-to-stock* ialah peramalan permintaan dan penentuan persediaan pengaman(*safety stock*)[2].

2.2.3. Aplikasi

Aplikasi adalah penerapan, penggunaan atau penambahan[3]. Dari pengertian diatas, dapat disimpulkan bahwa aplikasi merupakan software yang berfungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan, penggunaan dan penambahan data.

Aplikasi adalah penggunaan dalam suatu computer, instruksi (*instruction*) atau pernyaraan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*[4].

Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas. Misalnya termasuk perangkat lunak perusahaan , *software* akuntansi , perkantoran , grafis perangkat lunak dan pemutar media. [5] Dapat disimpulkan bahwa aplikasi merupakan *software* yang berfungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan,

penggunaan dan penambahan data. Program aplikasi merupakan program siap pakai. Program yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Contoh-contoh aplikasi ialah program pemproses kata dan *Web Browser*. Aplikasi akan menggunakan sistem operasi (OS) komputer dan aplikasi yang lainnya yang mendukung.

Klasifikasi aplikasi dapat dibagi menjadi 2 (dua) yaitu:

1. Aplikasi *software* spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.
2. Aplikasi paket, dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu.

2.2.4. Android

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti bergerak. Awalnya, Google Inc. Membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan 16 peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia[6].

Fitur Android:

a. *Framework* aplikasi, memungkinkan daur ulang dan penggantian komponen.
b. *Browser* terintegrasi berbasis *engine Open Source WebKit* yang juga digunakan di *browser* Iphone dan Nokia S60v3.
c. Rancangan *handset*. Platform disesuaikan dengan kebutuhan VGA (*Video Graphics Adapter*) yang lebih besar, *library* grafik 2D dan 3D yang berdasarkan pada spesifikasi OpenGL ES 1.0 serta *layout smartphone* yang tradisional.
d. Konektivitas. Android mendukung berbagai teknologi konektivitas seperti GSM (*Global System for Mobile Communications*)/EDGE *(Enhanced Data rates for GSM Evolution*), CDMA (*Code Division Multiple Access*), EV-DO

(*Evolution-Data Optimized*), UMTS (*Universal Mobile Telecommunications System*), Bluetooth dan Wi-Fi (*Wireless Fidelity*).

e. Pesan. Android mendukung pengiriman pesan dalam bentuk SMS (*Short Message Service*) dan MMS (*Multimedia Messaging Service*). Dukungan Java. Software yang ditulis dalam bahasa Java dapat dikompilasi dan akan dieksekusi pada mesin virtual Dalvik, yang merupakan implementasi dari VM (*Virtual Machine*) yang dirancang khusus untuk penggunaan perangkat bergerak.

f. Dukungan media. Android mendukung beberapa format audio/video seperti: H.263, H.264 (dalam kontainer 3GP atau MP4), MPEG-4 SP, AMR, AMR-WB (dalam kontainer 3GP), AAC, HE-AAC (dalam kontainer MP4 atau 3GP), MP3, MIDI, OGG Vorbis, WAV, JPEG, PNG, GIF dan BMP.

g. Dukungan *hardware* tambahan. Android mendukung penggunaan kamera, layar sentuh, GPS (*Global Positioning System*), pengukur kecepatan, magnetometer, akselerasi 2D bit blits (dengan orientasi hardware, scaling, konversi format piksel) dan akselerasi grafis 3D. 17

h. Play store. Mirip dengan App Store pada iPhone OS, Android Play store adalah sebuah katalog aplikasi yang dapat di-download dan diinstal pada telepon seluler secara online, tanpa menggunakan PC (*Personal Computer*). Awalnya hanya aplikasi gratis saja yang didukung. Dan sejak tanggal 19 Februari 2009 aplikasi berbayar telah tersedia di Android *Play store* untuk Amerika Serikat.

i. *Multi-touch*. Android memiliki dukungan bawaan untuk *multi-touch* yang tersedia pada *handset* terbaru seperti HTC Hero. Pada awalnya fitur tersebut dinonaktifkan pada level kernel (mungkin untuk menghindari pelanggaran paten terhadap teknologi layar sentuh Apple). Sejak Google merilis *update* untuk Nexus One dan berencana juga untuk merilis update untuk Motorola Droid yang memungkinkan *multi-touch*.

j. Lingkungan pengembangan yang kaya, termasuk emulator, peralatan *debugging*, dan *plugin* untuk Eclipse IDE. 2.2.5.1.

2.2.5. Product Customization

Pinus Gilmore mendefinisikan kustomisasi sebagai "melakukan produksi dalam menanggapi dengan keinginan pelanggan." Acara penulis yang relevan untuk membuat perbedaan antara variasi dan kustomisasi. Kustomisasi berusaha untuk memenuhi kebutuhan pelanggan, berbagai inidividu melibatkan banyak pilihan dan pelanggan dapat melakukan pemilihan. "Pada dasarnya, pelanggan tidak ingin pilihan; mereka hanya ingin apa yang mereka inginkan. "Kustomisasi ini ditujukan untuk menambah peningkatan nilai produk, karena produk disesuaikan-dibandingkan dengan produk yang di produksi secara masal-semakin memenuhi kebutuhan pelanggan[7].

Definisi istilah kustomisasi produk, adalah sesuatu yang relevan dari perspektif antara fisik yang baik ataupun layanan yang sesuai dengan kebutuhan pelanggan tertentu. Dalam hal ini, keterlibatan pelanggan merupakan masalah penting menambahkan layanan di sekitar fisik produk dianggap sebagai kriteria tambahan untuk membedakan barang dengan kustomisasi.[7].

2.2.6. Virtual Reality

Virtual *Reality* dipahami sebagai simulasi komputer interaktif yang dapat mempengaruhi indra pengguna bahkan menggantikan satu atau lebih indra manusia, sehingga pengguna larut kedalam lingkungan simulasi (*virtual environment*[8]. Virtual *Reality* (VR) Secara bahasa berarti keadaan nyata/ide yang “dimasukkan” ke dalam dunia maya atau memvirtualkan objek nyata/ide yang tetap memperhitungakan sifat-sifat fisikanya. Oleh karena itu harus dibedakan dengan animasi 3D, yang terdapat pada film dan game, karena tidak memperhitungkan data dan kondisi fisik dari objek-objek yang berada di dalamnya (lingkungan virtual). Sebuah VR pasti memperhitungkan aspek ergonomis dan antropometri. Ini adalah *added value* sebuah VR. Ergonomis berarti barang yang divirtualkan harus cocok dengan anatomi tubuh manusia ketika digunakan seperti kita menggunakan barangbarang yang biasa berada di sekitar kita, sedangkan antopometri berarti di dalam virtualisasi tersebut diperhitungkan ukuran fisik dari gerakan manusia terhadap semua objek virtual di sekelilingnya. Dua hal tersebut merupakan aspek

analisis yang menjadi pembeda VR terhadap games, aspek lainnya adalah fungsionalitas. Di sisi ini komponennya adalah *reachability*, *touchability*, dan *accessability*. *Reachability* berarti objek di dalam dunia vitual dapat dijangkau, dipegang, dapat berinteraksi dengan user. *Touchability* berarti objek dapat dirasakan, objek yang kita pegang atau sentuh memiliki berat ataupun kontur permukaan, dan *accessability* berarti objek dalam dunia virtual memiliki perilaku sama dengan objek dalam dunia nyata, misalkan bila dalam tubuh manusia terdapat 25 derajat kebebasan maka dalam dunia virtual pun harus sama.

Virtual *Reality* merujuk pada pemakaian komputer untuk mensimulasikan sebuah pengalaman dengan cara yang sama dengan realita. Pada jenis-jenis VR yang paling sering dipakai, seseorang memakai sarung tangan, *earphone*, dan goggles yang disambungkan dengan komputer. Rangsangan berubah sesuai dengan gerakan orang itu, misalnya menggeleng-gelengkan kepala atau gerakan-gerakan lainnya.

Secara sederhana, Virtual *Reality* adalah pemunculan gambar-gambar tiga dimensi yang di bangkitkan komputer, yang terlihat nyata dengan bantuan sejumlah peralatan tertentu. Ciri terpentingnya adalah dengan menggunakan perangkat yang dirancang untuk tujuan tertentu, teknologi ini mampu menjadikan orang yang merasakan dunia maya tersebut terkecoh dan yakin bahwa yang dialaminya adalah nyata. Ada empat elemen dasar dari virtual *reality*[8] yaitu :

a. Virtual Enviroment

   Definisi dari VR dan VE terus berkembang dan pada saat ini istilah keduanya saling berkaitan. VR mengarah kepada teknik atau sistem berupa perangkat dan software, sedangkan VE merupakan lingkungan yang diciptakan melalui komputer. Virtual *Environment* adalah lingkungan yang disimulasikan oleh komputer, berupa lingkungan sebenarnya yang ditiru atau lingkungan yang hanya ada dalam imaginasi .

b. Virtual Presence

   yaitu sebuah perasaan keberadaan seseorang dari lingkungan virtual. Pengguna tersebut bereaksi dengan objek virtual selayaknya berinteraksi dengan objek nyata. Pengguna merepresentasikan perasaan dari berada di sebuah lingkungan

virtual. Virtual *Presence* sulit untuk dimunculkan melalui media selain VR karena media lain tersebut tidak memberikan fasilitas atau kemampuan kepada sensor aktual dan imersi fisik langsung terhadap lingkungan. Pemahaman dari kehadiran langsung telah ditingkatkan melalui konsep dari kehadiran yang dimunculkan melalui media. Imersi merupakan keadaan dimana pengguna berada di sebuah lingkungan yang berupaya untuk meningkatkan perasaan ruang atau perasaan seperti berada di keadaan nyata. Virtual *Presence* dapat dikategorikan menjadi *physical* (*sensory*) dan *mental presence*.

1. Physical (sensory) Virtual Presence

   Kehadiran virtual secara fisik mendefinisikan virtual *reality* dan sekaligus membedakannya dari media lain. Kehadiran virtual secara fisik ini didapat dari memberikan pengguna sebuah lingkungan virtual dengan satu atau lebih sensor yang dapat merubah posisi pengguna dan gerakannya. Pada umumnya sistem virtual *reality* melakukan render lingkungan virtual melalui penglihatan, pendengaran, dan sentuhan.

2. Mental Virtual Presence

   Tingkatan kemampuan kehadiran virtual secara mental tergantung pada tujuan yang ingin dicapai melalui virtual *reality*. Jika virtual *reality* digunakan dengan tujuan hiburan, maka diperlukan kehadiran virtual secara mental dengan tingkat tinggi. Bagaimanapun, kehadiran virtual secara mental kadang tidak begitu dibutuhkan. Tidak adanya kehadiran virtual secara mental tidak mendiskualifikasi media dari menjadi virtual *reality*.

c. Sensory Feedback

   Umpan balik sensoris merupakan komponen krusial dari VR. Sistem VR memberikan umpan balik sensoris secara langsung melalui informasi visual. Sistem virtual *reality* memberikan umpan balik sensoris secara langsung kepada pengguna berdasarkan lokasi fisiknya.

d. Interactivity

   Interaktivitas adalah salah satu fitur media baru yang paling banyak dibicarakan, mendapat tempat khusus di internet. Seperti halnya berbagai istilah dalam dunia cyber baru, kadang sulit memahami arti sebenarnya dari kata itu. Satu masalah

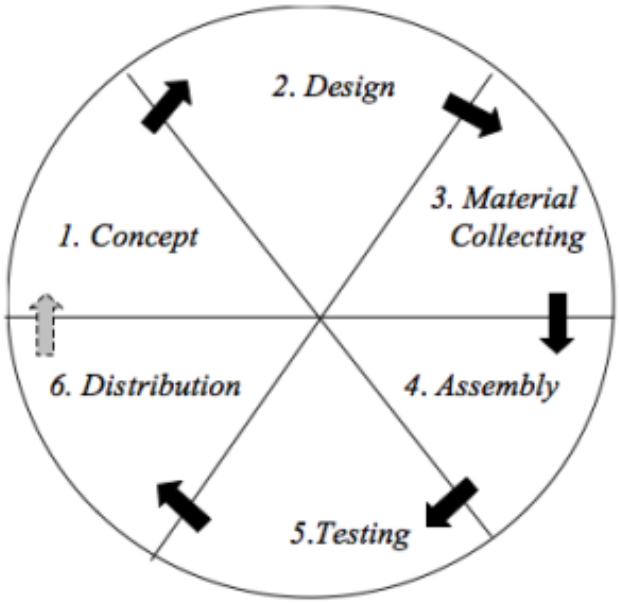
dalam mendefinisikan istilah interaktivitas adalah bahwa ia dipakai minimal dalam dua makna berbeda. Orang-orang dengan latar belakang ilmu komputer cenderung memaknainya sebagai interaksi pengguna dengan komputer, sebagaimana permainan-permainan interaktif. Definisi semacam itu menyebutkan bahwa interaktivitas berarti kemampuan pengguna untuk berkomunikasi secara langsung dengan komputer dan memiliki dampak pada pesan apapun yang sedang dibuat. Para sarjana komunikasi cenderung berpikir bahwa interaktivitas merupakan komunikasi antara dua manusia. William, Rice, dan Rogers (1988) mendefinisikan interaktivitas sebagai tingkatan dimana pada proses komunikasi para partisipan memiliki kontrol terhadap peran dan dapat bertukar peran dalam dialog mutual mereka. Interaktivitas memiliki 6 dimensi, yaitu:

1. Persuasi – menginformasikan
2. Kontrol lemah – kontrol tinggi
3. Aktivitas rendah – aktivitas tinggi
4. Satu arah – dua arah
5. Waktu tertentu – waktu fleksibel
6. Kesadaran rendah terhadap tempat – kesadaran tinggi terhadap tempat

2.2.7. Metode Luther Sutopo

Menurut Luther(1994), metodologi pengembangan multimedia terdiri dari enam tahap, yaitu *concept*(pengonsepan), *design*(pendesainan), *material collecting*(pengumpulan materi), *assembly*(pembuatan), *testing*(pengujian) dan *distribution*(pendistribusian). Keenam tahap ini tidak harus berurutan dalam praktiknya, tahap-tahap tersebut dapat saling bertukar posisi. Meskupun, tahap *concept* memang harus menjadi hal yang pertama kali dikerjakan.

Sutopo(2003) mengadopsi metodologi Luter dengan modifikasi, seperti terlihat pada Gambar 2.3[9].



Gambar 2. 2 Pengembangan multimedia Luther-Sutopo[9]

Penjelasan dari setiap tahapannya adalah sebagai berikut :

a. *Concept*

Tahap *concept*(pengonsepan) adalah tahap untuk menentukan tujuan dan siapa pengguna program(identifikasi audiens). Tujuan dan pengguna akhir program berpengaruh pada nuansa multimedia sebagai pencerminan dari identitas organisasi yang menginginkan informasi sampai pada pengguna akhir. Karakteristik pengguna termasuk kemampuan pengguna juga perlu dipertimbangkan karena dapat memengaruhi pembuatan desain.

Selain itu, tahap ini juga akan menentukan jenis aplikasi(presentasi, interaktif dan lain-lain) dan tujuan aplikasi (hiburan, pelatihan, pembelajaran dan lain-lain). Dasar aturan untuk perancangan juga ditentukan pada tahap ini, misalnya ukuran aplikasi, target dan lain-lain. Output dari tahap ini biasanya berupa dokumen yang bersifat naratif untuk mengungkapkan tujuan projek yang ingin dicapai.

b. *Design*

Design(perancangan) adalah tahap pembuatan spesifikasi mengenai arsitektur program, gaya, tampilan dan kebutuhan material atau bahan untuk program. Spesifikasi dibuat serinci mungkin sehingga pada tahap berikutnya, yaitu

material collecting dan *assembly*, pengambilan keputusan baru tidak diperlukan lagi, cukup menggunakan keputusan yang sudah ditentukan pada tahap ini.

Tahap ini biasanya menggunakan *storyboard* untuk menggambarkan deskripsi tiap *scene*, dengan mencantumkan semua objek multimedia dan tautan ke scene lain dan bagan alir(*flowchart*) untuk menggambarkan aliran dari satu *scene* ke *scene* lain.

c. *Material Collecting*

Material Collecting adalah tahap pengumpulan bahan yang sesuai dengan kebutuhan yang dikerjakan. Bahan-bahan tersebut, antara lain gambar *clip art*, foto, animasi, video, audio dan lain-lain yang dapat diperoleh secara gratis atau dengan pemesanan kepada pihak lain sesuai dengan rancangannya. Tahap ini dapat dikerjakan secara paralel dengan tahap *assembly*. Namun, pada beberapa kasus tahap *material collecting* dan tahap *assembly* akan dikerjakan secara linear dan tidak paralel.

d. *Assembly*

Tahap *assembly* adalah tahap pembuatan semua objek atau bahan multimedia. Pembuatan aplikasi didasarkan pada tahap design, seperti storyboard, bagan alir dan/atau struktur navigasi.

e. *Testing*

Tahap *testing*(pengujian) dilakukan setelah menyelesaikan tahap pembuatan(*assembly*) dengan menjalankan aplikasi atau program dan melihatnya apakah ada kesalahan atau tidak.

f. *Distribution*

Pada tahap ini, aplikasi akan disimpan dalam suatu media penyimpanan. Jika media penyimpanan tidak cukup untuk menamung aplikasinya, kompresi terhadap aplikasi tersebut akan dilakukan.[9]

2.2.8. Business Process Modelling Notations(BPMN)

BPMN merupakan teknik yang memungkinkan semua pihak yang terlibat dalam proses berkomunikasi secara jelas, benar dan efisien. Dengan cara ini, BPMN mendefinisikan notasi dan semantic Diagram Proses Bisnis(BPD). BPD

adala diagram berdasarkan Teknik ‘Flowchart’, yang dirancang untuk menyajikan urutan grafis dari semua kegiatan yang terjadi selama proses. BPMN menyediakan kemampuan memahami prosedur internal bisnis dalam notasi grafis. BPMN mengikuti tradisi flowcharting notasi untuk dibaca dan fleksibilitas. Ada 4 kategori dasar dari elemen-elemen BPD adalah sebagai berikut[10] :

a. Flow Objects

   Events : hal hal yang terjadi di antara proses bisnis. *Event* ini mempengaruhi aliran proses dan biasanya memiliki penyebab(pemicu) atau dampak (hasil). *Event* disimbolkan dengan lingkaran kecil dengan garis tipis(start event), lingkaran kecil dengan garis ganda(intermediate event), lingkaran kecil dengan garis tebal (end event).

   Activities: aktiftas artinya perusahaan melakukan proses sebuah kegiatan dapat berupa Sub-Proses dan Task, yang disimbolkan dengan bulat persegi panjang.

   Gateways: gateway digunakan untuk mengontrol perbedaan dan konvergensi dari urutan arus dalam proses. Dengan demikian, akan menentukan percabangan, forking, penggabungan dan bergabung dengan jalur.

b. Connecting Objects

   Flow Object dihubungkan satu sama lain dalam suatu diagram untuk menciptakan struktur dasar proses bisnis. Ada tiga connecting object yang disediakan fungsi ini, yaitu :

   Sequence Flows: sebuah arus urutan digunakan untuk menunjukan urutan kegiatan yang akan dilakukan dalam proses, disimbolkan dengan garis dengan panah solid.

   Message Flow: direpresentasikan oleh garis putus-putus dengan kepala panah terbuka dan digunakan untuk menunjukan aliran pesan antara dua partisipan proses(entitas bisnis atau peran bisnis).

   Association: asosiasi direpresentasikan oleh garis titk-titik kepala panah garis dan digunakan untuk asosiasi data, teks dan artifak lainnya dengan *flow object*. Asosiasi digunakan untuk output aktivitas.

c. Swimlanes

   BPMN membagi swimlanse menjadi 2 construct utama yaitu :

Pool: merepresentaskan partisipan dalam sebuah proses. *Pool* juga bertindak sebagai grafis container untuk memisahkan sekumpulan aktivitas dari *pool-pool* yang lainnya.

Lane: sebuah lane merupakan sub-partisi di dalam *Pool* dan akan memperpanjang seluruh panjang *pool*, baik secara vertical atau horizontal.

Lane sering kali digunakan untuk memisahkan kegiatan yang beruhubungan dengan fungsi perusahaan atau peran spesifik. *Sequence Flow* memungkinkan lintas batas *lanes* di dalam *Pool*, tetapi *Message Flow* tidak mungkin digunakan di antara *Flow Object* dakan *Lane* pada *Pool* yang sama.

d. Artifacts

Sejumlah artifak dapat ditambahkan ke dalam sebuah diagram yang sesuai untuk konteks proses bisnis yang dimodelkan. Artifak tersebut adalah sebagai berikut:

Data Object: data objek merupakan mekanisme untuk menunjukkan bagaimana data dibutuhkan atau diproduksi oleh kegiatan. Mereka dikoneksikan ke aktivitas melalui *association*.

Group: sebuah grup direpresentasikan melalui empat persegi panjang yang sudutnya bulat dengan garis putus-putus. Grup dapat diunakan untuk dokumentasi atau analisis fungsi, tetapi tidak berdampak pada *sequence flow*.

Annotation: mekanisme untuk modeler untuk memberikan informasi teks tambahan untuk pembaca sebuah diagram BPMN.[10]

2.2.9. Pemrograman Berorientasi Objek

Analisis dan desain berorientasi objek adalah cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek yang merupakan kombinasi antara struktur data dan perilaku dalam suatu entitas. Model berorientasi objek bermanfaat untuk memahami masalah, komunikasi dengan ahli aplikasi, pemodelan suatu organisasi, meyiapkan dokumentasi serta perancangan program dan basis data. Pertama-tama suatu model analisis dibuat untuk menggambarkan aspek dasar dari domain aplikasi, dimana model tersebut berisiobjek yang terdapat dalam domain

aplikasi termasuk deskripsi dari keterangan objek dan perilakunya. Secara spesifik, pengertian berorientai objek berarti bahwa mengorganisasi perangkat lunak sebagai kumpulan dari objek tertentu yang memiliki struktur data dan perilakunya. Hal ini yang membedakan dengan pemograman konvensional dimana struktur data dan perilaku hanya berhubungan secara terpisah. Terdapat beberapa cara untuk menentukan karateristik dalam pendekatan berorientasi objek, tetapi secara umum mencakup empat hal, yaitu identifikasi, klasifikasi, *polymorphism* (polimorfisme) dan *inheritance* (pewarisan)[11].

2.2.10. Unified Modeling Language (UML)

Unified Modelling Language (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah system, 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 memodelkan

aplikasi prosedural dalam VB atau C. Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memilikimakna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari tiga notasi yang telah ada sebelumnya: Grady Booch OOD (*Object-Oriented Design*), Jim Rumbaugh OMT (*Object Modeling Technique*), dan Ivar Jacobson OOSE (*Object-Oriented Software Engineering*)[11].

2.2.10.1. Diagram UML

UML menyediakan 10 macam diagram untuk memodelkan aplikasi berorientasi objek, yaitu:

1. *Use Case Diagram* untuk memodelkan proses bisnis.
2. *Conceptual Diagram* untuk memodelkan konsep-konsep yang ada didalam aplikasi.
3. *Sequence Diagram* untuk memodelkan pengiriman pesan (*message*) antar objek.
4. *Collaboration Diagram* untuk memodelkan interaksi antar objek.
5. *State Diagram* untuk memodelkan perilaku objek di dalam sistem.
6. *Activity Diagram* untuk memodelkan perilaku userdan objek di dalam sistem.
7. *Class Diagram* untuk memodelkan struktur kelas.
8. *Objek Diagram* untuk memodelkan struktur objek.
9. *Component Diagram* untuk memodelkan komponen objek.
10. *Deployment Diagram* untuk memodelkan distribusi aplikasi.

Dari 10 macam diagram yang disediakan oleh UML berikut ini 4 macam diagram yang paling sering digunakan dalam pembangunan aplikasi berorientasi objek, yaitu *use case diagram, sequence diagram, collaboration diagram,* dan *class diagram*[10].

2.2.10.2. Use Case Diagram

Use Case adalah titik awal yang sangat baik untuk hampir setiap aspek pembangunan sistem berorientasi objek,

Desain, pengujian, dan dokumentasi. Mereka menggambarkan sistem persyaratan ketat dari luar memandang ke dalam; mereka menentukan nilai yang sistem memberikan kepada pengguna. Karena kasus penggunaan sistem Anda yang fungsional persyaratan, mereka harus serius pertama output dari model Anda setelah proyek dimulai[11].

2.2.10.3. Activity Diagram

Diagram aktivitas sangat baik pada pemodelan proses bisnis. Proses bisnis adalah satu set. Tugas-tugas terkoordinasi yang mencapai tujuan bisnis, seperti

pengiriman pesanan pelanggan. Beberapa proses bisnis alat manajemen (BPM) memungkinkan Anda untuk menentukan proses bisnis yang menggunakan diagram aktivitas, atau serupa notasi grafis, dan kemudian mereka jalankan. Hal ini memungkinkan Anda untuk menetapkan dan melaksanakan, misalnya, pembayaran proses persetujuan yang mana salah satu langkah memanggil kartu kredit persetujuan web serviceusing mudah grafis notasi seperti diagram aktivitas.

2.2.10.4. Sequence Diagram

Sequence diagram menjelaskan secara detil urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case. Interaksi yang terjadi antar class, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.[11]

2.2.10.5. Class Diagram

Class diagram merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. Class diagram menunjukkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan[11].

2.2.11. PHP *Hypertext Preprocessor*

Dalam dunia pemrograman website ada banyak sekali bahasa pemrograman yang bisa digunakan. Salah satu bahasa pemrograman yang sangat terkenal dan banyak sekali digunakan oleh para pembuat website adalah PHP (akronimnya: *Hypertext Preprocessor*), dengannya website menjadi dinamis (karena kandungan website tersebut dapat berbasis database). [12]

Salah satu keunggulan PHP dibanding bahasa pemrograman lainnya adalah PHP dapat diperoleh secara gratis, meskipun bukan berarti karena gratis kemampuannya menjadi pas-pasan. PHP sangat *powerfull.* Terbukti dengan banyaknya website yang dibangun menggunakan PHP. PHP juga terkenal lebih aman daripada bahasa pemrograman website yang lain. [12]

PHP adalah bahasa scripting yang menyatu dengan HTML (kode dasar website) dan dijalankan pada server side. Artinya, semua sintaks PHP yang diberikan akan sepenuhnya dijalankan pada server, sedangkan yang dikirimkan ke browser hanya hasilnya saja. [12]

2.2.12. Pengujian Sistem

Sub sistem yang terintegrasi untuk membuat seluruh sistem. proses pengujian berkaitan dengan menemukan kesalahan yang dihasilkan dari tak terduga interaksi antara subsistem dan komponen sistem. Hal ini juga berkaitan dengan melakukan validasi bahwa sistem memenuhi persyaratan fungsional dan non-fungsional.[13]

Pada dasarnya ada tiga jenis utama dari pengujian sistem :

a. Alpha Testing
b. Beta Testing
c. Acceptance Testing

1) Pengujian Alpha, mengacu pada sistem pengujian yang dilakukan oleh tim penguji dalam pengembangan organisasi. Pengujian alpha dilakukan pada situs pengembang di daerah kerja tim proyek. dalam tes ini, pengguna uji perangkat lunak pada platform pengembangan dan menunjukkan kesalahan untuk ko reksi. Namun, uji alpha, karena beberapa pengguna pada platform pengembangan melakukan itu, memiliki keterbatasan kemampuan untuk mengekspos kesalahan dan memperbaikinya. Pengujian Alpha dilakukan dalam lingkungan yang terkendali. itu adalah simulasi penggunaan nyata. setelah uji alpha lengkap, produk perangkat lunak siap untuk transisi ke situs pelanggan untuk pelaksanaan dan pengembangan.
2) Pengujian Beta, pengujian beta adalah sistem pengujian dilakukan oleh sebuah grup pelanggan yang dipilih.

   Jika sistem yang kompleks, perangkat lunak tidak diambil untuk pelaksanaan secara langsung. ia terinstal dan semua pengguna akan diminta untuk menggunakan perangkat lunak dalam pengujian modus. Ini disebut pengujian beta. Tes beta dilakukan di lokasi pelanggan dan dii lingkungan

dimana software terkena sejumlah pengguna. Pengembang adalah pengalaman kehidupan nyata perangkat lunak tanpa pelaksanaannya. Semua ini, pengguna akhir merekam pengamatan mereka, kesalahan, kesalahan dan seterusnya dan melaporkannya secara berkala. dalam tes beta, pengguna mungkin menyarankan modifikasi, perubahan besar atau penyimpangan. *development* telah meneliti usulan perubahan dan meletakkan ke dalam sistem manajemen perubahan untuk perubahan halus dari hanya dikembangkan software untuk direvisi, perangkat lunak yang lebih baik. ini adalah praktek standar untuk menempatkan semua perubahan dalam rilis versi berikutnya.

3) Penerimaan pengujian. penerimaan pengujian adalah sistem pengujian dilakukan oleh pelanggan untuk menentukan apakah akan menerima atau menolak sistem.

   Ketika pelanggan perangkat lunak dibangun untuk satu pelanggan, serangkaian tes penerimaan dilakukan untuk memungkinkan pelanggan untuk memvalidasi semua persyaratan. dilakukan oleh pengguna akhir daripada insinyur perangkat lunak, tes penerimaan dapat berkisar dari informal *'test drive'* untuk serangkaian direncanakan dan dilaksanakan secara sistematis tes. pada kenyataannya, penerimaan pengujian dapat dilakukan selama periode minggu atau bulan, sehingga mengungkap kumulatif kesalahan yang mungkin menurunkan sistem dari waktu ke waktu.[13]

2.2.13. Black Box Testing

Fungsional struktur program pengujian tidak dianggap. kasus uji yang ditentukan berdasarkan persyaratan atau spesifikasi dari program atau modul dan internal modul atau program yang tidak dianggap untuk seleksi uji kasus.[13]

Pengujian fungsional mengacu pada pengujian yang melibatkan hanya pengamatan output untuk nilai-nilai input tertentu dan tidak ada upaya untuk menganalisis kode, yamg menghasilkan output. Struktur internal program akan diabaikan. Untuk alasan ini, pengujian fungsional kadang-kadang disebut sebagai

pengujian *black box* (disebut juga pengujian perilaku) di mana isi dari *black box* tidak diketahui dan fungsi kotak hitam ini difahami sepenuhnya dari segi input dan output.

Pengujian *black box*, juga disebut pengujian perilaku, berfokus pada persyaratan fungsional dari perangkat lunak. Pengujian *black box* memungkinkan insinyur perangkat lunak untuk memperoleh set masukan kondisi yang sepenuhnya akan melatih semua persyaratan fungsional untuk program.

Nama lain untuk pengujian *black box* (BBT) meliputi spesifikasi pengujian, pengujian perilaku, data-driven pengujian, pengujian fungsional dan input/output berbasis pengujian

Dalam pengujian *black box*, penguji hanya tahu masukan yang dapat diberikan kepada sistem dan apa output sistem harus memberi. dalam kata lain, dasar untuk menentukan kasus uji di pengujian fungsional persyaratan atau spesifikasi sistem atau modul. bentuk pengujian juga disebut fungsional atau perilaku pengujian.[13]

2.2.14. Wawancara

Wawancara adalah bentuk komunikasi langsung antara peneliti dan responden. Komunuikasi berlangsung dalam bentuk tanya-jawab dalam hubungan tatap muka, sehingga gerak dan mimic responden merupakan pola media yang melengkapi kata-kata secara verbal. Karena itu, wawancara tidak hanya mengangkap pemahaman atau ide, tetapi juga dapat menangkap perasaan, pengalaman, emosi, motif yang dimiliki oleh responden yang bersangkutan. Di sinilah terletak keunggulan dari metode wawancara.[14]

2.2.15. Tools Yang Digunakan

Dalam pembangunan Aplikasi Kustomisasi Virtual *Furniture* 3D ini, penulis menggunakan beberapa *tools* diantaranya sebagai berikut :

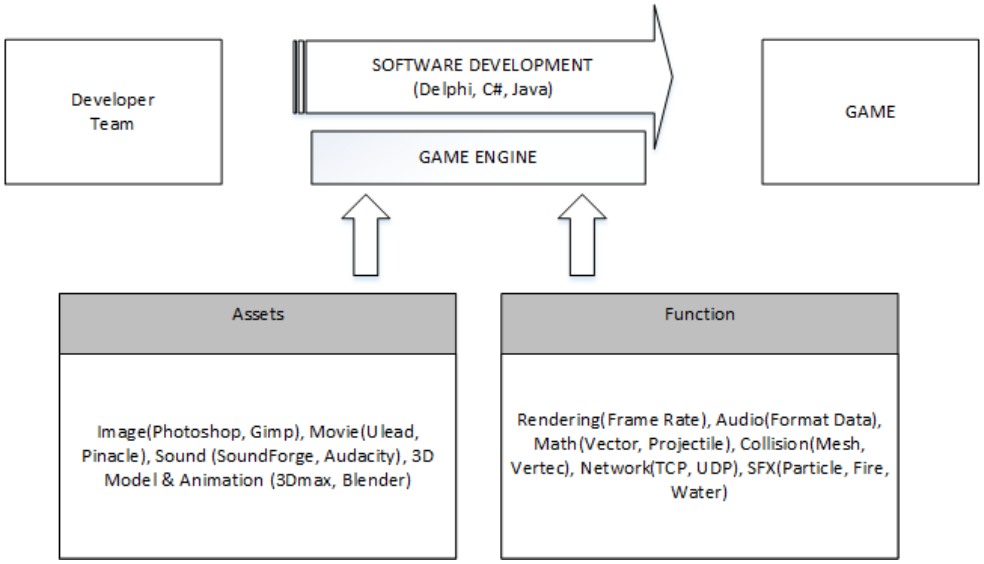
2.2.15.1. Android SDK

Android SDK (Software Development Kit) adalah tools API (Application Programming Interface) yang diperlukan untuk pengembangan atau pembangunan suatu aplikasi pada platform android menggunakan bahasa pemrograman Java. Saat ini disediakan Android SDK sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada platform android menggunakan bahasa pemrograman Java. Sebagai platform aplikasi netral, android memberi kesempatan untuk membuat aplikasi yang kita butuhkan yang bukan merupakan bawaan handphone/ smartphone. Beberapa fitur-fitur android yang paling penting adalah[15] :

a. Framework aplikasi yang mendukung penggantian komponen dan reusable.
b. Mesin Virtual Dalvik dioptimalkan untuk perangkat mobile.
c. Integrated browser berdasarkan engine opensource WebKit.
d. Grafis yang dioptimalkan dan didukung oleh libraries grafis 2D, grafis 3D berdasarkan spesifikasi opengl ES 1,0 (Opsional akselerasi hardware).
e. SQLite untuk penyimpanan data.
f. Media support yang mendukung audio, video, dan berbagai format gambar (MPEG4, H.264, MP3,AAC, AMR, JPG, PNG, GIF), GSM Telephony (tergantung hardware).
g. GSM, Bluetooth, EDGE, 3G, dan WiFi (tergantung hardware).
h. Kamera, Global Positioning System (GPS), kompas, dan accelerometer (tergantung hardware).
i. Lingkungan Development yang lengkap dan kaya termasuk perangkat emulator, tools untuk debugging, profil dan kinerja memori, dan plugin untuk IDE Eclipse..[15]

2.2.15.2. Unity

Aplikasi Unity 3D *Engine* adalah sebuah *game engine*. *Game engine* sebuah perangkat lunak yang dirancang untuk membuat sebuah *game*. Sebuah *game engine* biasanya dibangun dengan mengenkapsulasi beberapa fungsi standar yang umum digunakan dalam pembuatan sebuah *game*. Misalnya, fungsi *rendering*, pemanggilan suara, *network* atau pembuatan partikel untuk *special effect*. Sebagian besar *game engine* umumnya berupa *library* atau sekumpulan fungsi-fungsi yang penggunaannya dipadukan dengan Bahasa pemrograman[16].



Gambar 2. 3 Unity[16]

Pada beberapa kasus *game engine* kadang pula disebut sebagai *middleware*. Hal ini dikarenakan *game engine* digunakan sebagai perantara antara bahasa pemrograman dengan format data dari berbagai perangkat lunak penghasil *asset*. Hal ini tentu saja memudahkan *developer* karenakan tidak perlu lagi memikirkan cara mengolah data yang dihasilkan dari berbagai perangkat lunak yang berbeda.

Sebuah *game engine* juga biasanya dilengkapi dengan fungsi-fungsi grafis atau komputasi dasar yang jarang diketahui oleh *developer*. *Game engine* membungkus itu semua untuk menjadikan proses pembuatan menjadi lebih cepat dan mudah. Beberapa *game engine* jenis ini diantaranya adalah TrueVision, Irrlicht dan Ogre3D.

Di awal tahun 2000-an, *game engine* mengalami perkembangan yang cukup signifikan. Beberapa *game engine* mulai dilengkapi dengan *World Editor*. Sehingga alih-alih menggunakan perangkat lunak 3D seperti 3Dmax atau Blender, pembuatan level atau 'dunia' game dapat dibuat melalui sebuah perangkat lunak tersendiri yang telah di rancang khusus untuk *game engine* tersebut.

Perkembangan ini bukannya tanpa masalah. Di satu sisi, software ini memudahkan para *developer* pemula yang membutuhkan kecepatan unuk melihat produk dari rancangan mereka. Tapi di sisi lain, perangkat lunak ini bias 'membatasi' kemampuan *developer* untuk berkreasi lebih jauh. Sehingga bagi sebagian *developer*, membangun 'dunia' game lewat script tetap puluhan kali lebih *powerfull*.

Unity Technologies dibangun di tahun 2004 oleh David Helgason, Nicholas Francis dan Joachim Ante. *Game engine* ini dibangun atas dasar kepedulian mereka terhadap *indie developer* yang tidak bias membeli *game engine* karena terlalu mahal. Focus perusahaan ini adalah membuat sebuah perangkat lunak yang bias digunakan oleh semua orang, khususnya untuk membangun sebuah game. Di tahun 2009, Unity diluncurkan secara gratis dan di april 2012, Unity mencapai popularitas tertinggi dengan lebih dari 1 juta *developer* terdaftar di seluruh dunia[16].

2.2.15.3. Blender

Blender merupakan aplikasi grafis 3D yang dirilis sebagai perangkat lunak bebas (*open source*) dibawah naungan GNU *General Public License*. Blender digunakan untuk *modeling, texturing, particle,* dan *simulations, non-linear editing, compositing,* dan membuat *interactive* 3D *applications*, termasuk *games*.

Blender tercipta dari komunitas pecinta animasi dari berbagai latar belakan. Mereka beraliansi membuat sebuah perangkat lunak yang bias dikembangkan (open source) dan bebas gratis. Blender digagas oleh seorang berkebangsaan Belanda yakni *Ton Roosendaal*. Pada awalnya dia mendirikan studio animasi di tahun 1988 kemudian blender menjadi *software share ware animation*. Di tahun yang sama dia juga mendirikan Blender Institute.

Sejak priode 2002 sampai sekarang, Blender tak berhenti update fitur, interface dan selalu meningkatkan performanya, sehingga kemampuan blender untuk digunakan produksi animasi tidak bias dibantahkan lagi, terbukti dengan hasil open movie beberapa film keluaran sintel (Blender Institute).

Blender, merupakan salah satu software animasi 3D yang mumpuni. Selain untuk animasi 3D, Blender juga dapat digunakan untuk mengerjakan konten-konten kreatif di bidang multimedia seperti spesial efek, koreks warna, pembuatan game iteraktif dan penyuntingan (*editing*) video. Salah satu kelebihan lainnya, Blender dapat berjalan di berbagai sistem operasi[17].

2.2.15.4. MySQL

Menurut Bimo Sunarfrihantono (2002), "mengatakan bahwa MySQL merupakan *multi user database* yang menggunakan bahasa *Structured Query Language* (SQL)"[18].

Menurut Lukmanul Hakim dan Uus Musalini (2004), "mengatakan bahwa MySQL merupakan salah satu jenis program *database server*, yaitu sebuah program yang berfungsi untuk mengolah, menyimpan dan memanipulasi data dari server"[19].

MySQL dikembangkan oleh MySQL AB, awal mula perkembangan MySQL adalah penggunaan mSQL untuk koneksi ke tabel menggunakan rutin level rendah (ISAM) setelah beberapa pengujian ternyata mSQL tidak cukup cepat dan fleksibel untuk memenuhi kebutuhan, sehingga dihasilkan SQL baru pada database tetapi dengan API yang mirip dengan mSQL dengan masa MySQL. Menambahkan, mengakses, dan memproses data yang tersimpan di dalam operasi, hal ini yang membuat PHP dan MySQL menjadi alternatif oleh para programmer web dalam membuat program database webnya.

2.2.15.5. StarUML

StarUML adalah sebuah proyek open source untuk pengembangan yang cepat, fleksibel, *extensible,featurful* dan tersedia secara gratis di platform UML/MDA pada platform win32. Tujuan dari proyek StarUML adalah untuk

membangun sebuah alat pemodelan perangkat lunak dan juga pengganti platform komersial yang menarik seperti Rational Rose, Together dan lainnya[20].

e. UML 2.0: UML terus memperluas standar yang dikelola oleh OMG(*Object Management Group*). Baru-baru ini, UML 2.0 dikeluarkan dan StarUML mendukung UML 2.0 dan akan mendukung UML standar.
f. MDA(*Model Driven Architecture*): MDA adalah sebuah teknologi baru yang diperkenalkan oleh OMG. Untuk mendapatkan keuntungan dari MDA, alat pemodelan perangkat lunak harus mendukung banyak kustomisasi variable. StarUML dirancang untuk mendukung MDA dan menyediakan banyak kustomisasi variable seperti UML profil, pendekatan, kerangka model, NX(*Notation Extension*), Kode MDA, template dokumen, dan lain-lain. Akan membantu untuk sebuah organisasi, proses dan proyek.
g. *Plug-in Architecture*: Banyak pengguna membutuhkan lebih banyak fungsi untuk pemodelan perangkat lunak. Untuk memenuhi hal tersebut, alat arus di definisikan dengan baik oleh platform *plug-in*. StarUML menyediakan arsitektur *plug-in* yang sederhana dan kuat sehingga siapapun dapat mengembangkan modul *plug-in* dalam bahasa yang kompatibel (C++,Delphi,C#,VB,…).
h. *Usability*: adalah sebuah isu yang paling penting dalam pengembangan perangkat lunak. StarUML diimplementasikan untuk memberikan banyak *user-friend* fitur seperti dialog cepat, manipulasi keyboard, tampilan diagram, dll.

StarUML sebagian besar ditulis dalam Delphi. Namun, StarUML adalah proyek multi bahasa dan tidak terikat untuk bahasa pemrograman tertentu. Sehingga setiap bahasa perograman dapat digunakan untuk mengembangkan StarUML. (Sebagai contoh C/C++, Java, Visual Basic, Delphi, JScript, VBScript, C#,VB.NET,…)[20].