

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Pada penelitian ini penulis membuat judul penelitian dengan nama “Sistem Informasi Pengadaan Bahan Baku Pada Kedai Manis Kopi Bandung”. Penelitian terdahulu yang penulis cantumkan di bawah ini adalah sebagai referensi sehingga dapat melengkapi teori pelengkap, serta pembanding yang memadai sehingga penulisan skripsi ini lebih memadai dalam mengkaji penelitian yang dilakukan. Berikut ini adalah penelitian yang penulis kaji sebelumnya.

Penelitian yang dilakukan oleh Rizky Pratama dengan judul ‘Sistem Informasi Pengadaan Bahan Baku pada Dede Collection Bag.Produksi’ memiliki beberapa permasalahan pada prosedur yang sedang berjalan, dan butuh beberapa pengembangan. Permasalahan yang pertama mengenai sistem pengelolaan data persediaan bahan baku. Pengelolaan data masih menggunakan pencatatan manual yaitu data faktur dari supplier bahan baku akan dicatat di buku kontrol bahan baku oleh bagian gudang (bahan baku). Secara garis besar pencatatan manual seperti ini banyak menimbulkan berbagai masalah, dari mulai pencatatan itu sendiri yang kurang jelas, banyak coretan, data bahan yang masuk dan data bahan yang keluar disatukan sehingga data menjadi kurang jelas, tidak ada back up data, dan tidak ada laporan data bahan baku yang jelas yang terperinci. Permasalahan lain karna data dari buku tersebut tidak terintegrasi ke

bagian lain, maka ketika data dibutuhkan oleh bagian lain akan ada kemungkinan kesalahan komunikasi antar bagian pada saat pembagian data [3].

Selanjutnya, penelitian yang dilakukan oleh Ahmad Dumairi dengan judul ‘Perancangan Sistem Informasi Bahan Baku Pada Logohouse Food Bar’ bertujuan untuk merancang suatu sistem informasi persediaan bahan baku yang dapat mendukung segala aktivitas bahan baku yang masuk dan keluar di perusahaan. Sistem persediaan bahan baku pada Logohouse Food Bar masih menggunakan cara biasa (banyak proses manual) seperti proses pencatatan bahan baku menjadi satu menu makanan, persediaan bahan baku masih belum terkontrol dan laporan masuk atau keluar nya bahan baku masih belum di ketahui. Sehingga dengan demikian tidak menutup kemungkinan hilang atau kurang nya bahan baku yang tidak di ketahui penggunaannya. Hasil dari penelitian ini adalah memberikan suatu alternatif sistem informasi persediaan bahan baku. Diharapkan dengan adanya aplikasi yang dibuat dapat menciptakan adanya mutu peningkatan operasional pada perusahaan [2].

Persamaan penelitian penulis yang sedang dilakukan dengan kedua penelitian yang di atas yaitu merancang suatu sistem informasi bahan baku secara terkomputerisasi.

Perbedaan dengan penelitian sodara Rizky Pratama dan Ahmad Dumairi yaitu penelitian dari kedua nya hanya membahas pengadaan bahan baku dan produksi bahan baku saja, sedangkan pada penelitian ini ada pengembangan yaitu menambahkan

proses penjualan dari hasil produksi bahan baku nya sehingga bahan baku yang telah digunakan akan terlihat pengurangan nya di laporan stok bahan baku.

2.2 Teori – Teori Dasar

Berikut ini adalah teori-teori dasar dalam penelitian ini yang harus dipahami agar penelitian yang dilakukan memperoleh hasil yang baik.

2.2.1 Definisi Sistem

Beberapa pendapat para ahli yang mengemukakan pengertian dari sistem, diantaranya yaitu:

Menurut Moekijat dalam Prasajo “Sistem adalah setiap sesuatu terdiri yang dari obyek-obyek, atau unsur-unsur atau komponen-komponen yang bertata kaitan dan bertata hubungan dengan satu sama lain, sedemikian rupa sehingga unsur-unsur tersebut merupakan satu kesatuan pemerosesan atau pengolahan tertentu” [3].

Menurut Sutarman, “Sistem adalah kumpulan elemen yang saling berhubungan dan berinteraksi dalam satu kesatuan untuk menjalankan suatu proses unyuk pencapaian suatu tujuan utama” [4].

Jadi sistem itu adalah satu kesatuan elemen yang berbeda-beda, cara kerja yang berbeda-beda namu saling berhubungan untuk mencapai satu tujuan utama.

Karakteristik Sistem

Menurut Sutabri, “Sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem” [5]. Adapun karakteristik yang dimaksud adalah sebagai berikut :

1. Komponen

Komponen sistem terdiri dari bagian- bagian atau sub sistem yang saling berinteraksi untuk membentuk suatu kesatuan.

2. Batasan Sistem (*Boundary*)

Batasan merupakan area dimana area ini merupakan pemisah antara satu sistem dengan sistem lainnya atau dengan lingkungan luarnya.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan, merupakan batasan luar sistem yang dapat mempengaruhi kinerja operasi sistem. Lingkungan ini dapat memberikan keuntungan ataupun kerugian terhadap sistem, maka dari itu haruslah ada pengendalian terhadap lingkungan luar ini.

4. Penghubung Sistem (*Interface*)

Adanya alat penghubung yang dapat menghubungkan sub sistem satu dengan yang lainnya, penghubung ini sekaligus sebagai media yang mengalirkan sumber daya dimana output yang dikeluarkan dari suatu sistem bisa menjadi input dari sub sistem lainnya.

5. Masukkan Sistem (*Input*)

Adanya sebuah energi yang dimasukkan ke dalam sistem komputer misalnya berupa perawatan (*maintenance*) yang dapat digunakan sistem untuk melakukan operasi dan masukkan sinyal (*signal input*) energi yang di dapat dari hasil proses untuk mendapatkan keluaran.

6. Keluaran Sistem (*Output*)

Hasil dari proses pengolahan dan menghasilkan keluaran.

7. Pengolahan Sistem

Merupakan sistem pengolahan yang dapat merubah masukan menjadi keluaran. Misalnya sistem produksi yang menghasilkan bahan jadi dari proses kelola bahan mentah, dan Sistem akuntansi yang digunakan untuk menghasilkan informasi berupa laporan-laporan yang berasal dari proses pengolahan data-data akuntansi. Berikut merupakan gambar karakteristik sistem:

8. Sasaran Sistem (*Objek*)

Suatu sistem pasti memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Jika suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.2.2 Definisi Informasi

Informasi adalah suatu data yang dapat diolah menjadi bentuk yang bermanfaat untuk menciptakan suatu keputusan. Informasi berguna untuk menciptakan keputusan karena informasi dapat menurunkan tingkat ketidakpastian (atau meningkatkan tingkat pengetahuan) Informasi sangatlah penting, Karena berdasarkan informasi manajemen dapat mengetahui kondisi obyektif pada perusahaannya. [5]

Membahas mengenai informasi, maka banyak para ahli yang bergerak dan membahas dibidangnya untuk mendefinisikan dan menjelaskan mengenai informasi tersebut, diantaranya menurut Amsyah, Zulkipli, Drs. Informasi merupakan “Data yang telah di proses menjadi bentuk yang lebih berguna bagi penggunaannya dan mempunyai nilai piker yang nyata bagi pencipta keputusan pada saat menjalankan operasional yang sedang berjalan atau untuk proyek di masa yang akan datang”. [5]

2.2.3 Konsep Dasar Informasi

Informasi adalah suatu data yang dapat diolah menjadi bentuk yang bermanfaat untuk menciptakan suatu keputusan. Informasi berguna untuk menciptakan keputusan karena informasi dapat menurunkan tingkat ketidakpastian (atau meningkatkan tingkat pengetahuan) Informasi sangatlah penting, karena berdasarkan informasi, manajemen dapat mengetahui kondisi obyektif pada perusahaannya.

Membahas mengenai informasi, maka banyak para ahli yang bergerak dan membahas dibidangnya untuk mendefinisikan dan menjelaskan mengenai informasi tersebut, diantaranya menurut Amsyah, Zulkipli, Drs. Informasi merupakan “Data yang telah di proses menjadi bentuk yang lebih berguna bagi penggunanya dan mempunyai nilai piker yang nyata bagi pencipta keputusan pada saat menjalankan operasional yang sedang berjalan atau untuk proyek di masa yang akan datang” [6].

2.2.4 Konsep Dasar Sistem Informasi

“Sistem informasi adalah sebuah sistem yang mempunyai fungsi mengumpulkan, memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk tujuan yang spesifik” Menurut Nugroho [7].

Jadi kesimpulan dari sistem informasi menurut penulis adalah berbagai macam fungsi yang berbeda dari pemrosesan data, menyimpan, menghasilkan informasi yang dibutuhkan, di era sekarang sangat banyak sekali contohnya sistem informasi yang sudah digunakan seperti contohnya marketplace tokopedia, bagaimana bisa menghubungkan antar pembeli dan penjual tidak saling bertemu dan bisa bertransaksi secara aman melalui sistem informasi tersebut.

2.2.5 Komponen Sistem Informasi

John Bruch dan Gary Grudnitski dalam Yustianti mengemukakan bahwa “Sistem informasi terdiri dari komponen – komponen yang di sebut dengan istilah blok bangunan (building block)” [8].

Blok masukan (input block)

Input yang mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang berupa dokumen-dokumen dasar.

Blok model (model blok)

Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara tertentu untuk menghasilkan keluaran yang diinginkan.

Blok keluaran (output block)

Produk dari sistem informasi ialah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta untuk semua pemakai sistem.

Blok teknologi (technology block)

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian sistem secara keseluruhan. Dalam blok ini terdiri dari 3 bagian

utama, yaitu teknisi (brainware), perangkat lunak (software) dan perangkat keras (hardware).

Blok basis data (database block)

Basis data (database) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, yang tersimpan di perangkat keras komputer dan digunakan oleh perangkat lunak untuk dimanipulasi. Basis data (data base) diakses atau dimanipulasi dengan menggunakan perangkat lunak yang disebut dengan DBMS (Database Management Systems).

2.2.6 Definisi Gudang

Gudang adalah suatu tempat yang digunakan untuk menyimpan barang baik *raw material*, barang *work in process* atau *finished good*. Dari kata gudang maka didapatkan istilah pergudangan yang berarti merupakan suatu kegiatan yang berkaitan dengan gudang. Menurut Holy Iacun Yunarto dan Martinus Getty Santika^[10] kegiatan tersebut dapat meliputi kegiatan *movement* (perpindahan), *storage* (penyimpanan), dan *information transfer* (transfer informasi)[14].

2.2.7 Klasifikasi Persediaan Dalam Gudang

Gudang dimana seperti kegunaannya secara umum merupakan suatu tempat untuk menyimpan benda. Benda yang disimpan di dalam gudang ini dapat pula disebut sebagai persediaan atau *inventori*. Pada Secara umum persediaan dapat diklasifikasikan

berdasarkan dua hal yang umum, yaitu klasifikasi persediaan berdasarkan fungsi dari barang dalam gudang dan klasifikasi persediaan berdasarkan kecepatan arus aliran barang.[14]

2.2.8 Klasifikasi Persediaan Berdasarkan Fungsi Dari Barang

Secara umum, berdasarkan fungsi fisiknya, persediaan dapat dibagi menjadi tiga fungsi utama. Ketiga fungsi persediaan tersebut adalah:

1. Sebagai *Raw Material* (Bahan Mentah)

Raw Material merupakan barang yang akan diproses dan diberi nilai tambah untuk kemudian dapat dijual dan dipasarkan kepada konsumen dengan nilai yang 9 lebih tinggi. *Raw Material* dapat berbeda-beda untuk setiap perusahaan tergantung jenis usaha dan tujuan usahanya.

2. Sebagai *Work In Process* (Barang Setengah Jadi)

Barang *Work in process* dalam bahasa sehari-hari dikenal dengan nama barang setengah jadi. Barang *Work in process* ini adalah *raw material* yang dikenai proses untuk menjadi suatu produk hanya belum selesai, atau dapat dikatakan masih setengah jalan.

3. Sebagai *Finished Good* (barang siap pakai)

Finished goods merupakan barang yang siap di pakai untuk disajikan atau siap di pasarkan pada konsumen. *Finished goods* ini merupakan barang yang akan di peroleh dari bahan dasar berupa raw material yang telah di proses dan di beri nilai tambah.

2.2.9 Sistem Persediaan Bahan Baku

Menurut Sofjan Assauri, “Persediaan bahan baku merupakan persediaan dari barang – barang berwujud yang digunakan dalam proses produksi, dapat diperoleh dari sumber–sumber alam ataupun dibeli dari supplier atau dari perusahaan yang menghasilkan bahan baku bagi perusahaan pabrik yang menggunakannya” [9].

2.2.10 Metode Pengembangan *Prototype*

Metode pengembangan perangkat lunak yang digunakan adalah *prototype*. *Prototype* ini merupakan metode pengembangan perangkat lunak yang berfokus untuk menghubungkan pemahaman kebutuhan pengguna dengan pengembang sehingga persepsi antara pengguna dan pengembang itu menjadi sama. Untuk dapat berinteraksi dalam proses kegiatan pengembangan maka metode *prototype* ini akan menghasilkan sebuah *prototype* sistem yang akan dibuat sebagai tahap awal dari sistem [10].

Penulis menggunakan pengembangan sistem antara lain yaitu *prototype* “prototyping model” metode ini untuk menjelaskan terhadap perusahaan mengenai hal teknis dan memperjelas kebutuhan yang di inginkan perusahaan kepada pengembang perangkat lunak.

2.2.11 *Unified Modeling Language (UML)*

UML adalah sebuah bahasa pemodelan sistem ataupun pemodelan di bidang lain yang membutuhkan sebuah pemodelan. Pemodelan menggunakan model dalam

bentuk grafis yang bertujuan untuk menspesifikasi, memvisualisasi dan membangun atau mendokumentasikan sebuah sistem agar mudah dipahami oleh manusia atau mesin dengan mudah. UML menggunakan konsep berorientasi *object* bertujuan untuk memudahkan manusia atau pengembang memahami sistem yang akan dibuat dalam bentuk grafis [11].

UML ini memiliki beberapa *diagram* yang digunakan untuk melakukan pemodelan data ataupun sistem. Berikut ini adalah *diagramnya* :

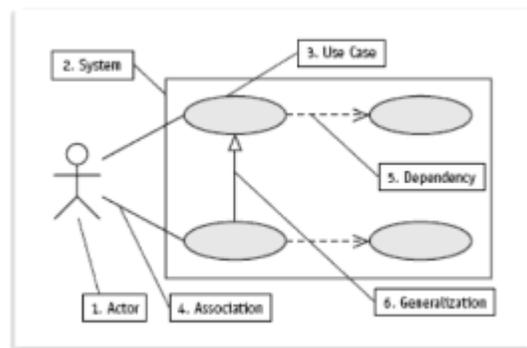
1. *Use Case Diagram*

Use Case Diagram adalah salah satu *diagram* UML yang bertujuan untuk menggambarkan hubungan antara manusia atau sistem lain dengan sistem yang mau dirancang. Manusia atau sistem lain disebut dengan aktor. Selain menggambarkan hubungan antara aktor dengan sistem, *use case* ini adalah menggambarkan sebuah fungsi dari sistem. Mudahnya adalah apa yang dilakukan oleh sistem bukan bagaimana yang dilakukan oleh sistem. Contohnya adalah aktor membuat daftar belanja dan sistem menjalankan fungsi buat daftar belanja [12].

Di dalam *use case* ada beberapa elemen yang digunakan yaitu seperti yang sudah disinggung di atas yaitu aktor, *use case*, sistem, *association*, *dependency*, dan *generalization*.

- a. Aktor adalah manusia atau sistem lain yang berinteraksi dengan sistem.

- b. *Use case* adalah fungsionalitas yang ada dalam sebuah sistem.
- c. Sistem adalah proses yang sudah dibatasi sesuai dengan apa yang kita buat.
- d. *Association* adalah sebuah garis yang menggambarkan interaksi atau hubungan antara aktor dengan sistem.
- e. *Dependency* adalah sebuah garis yang menggambarkan interaksi atau hubungan antara dua *use case*. Interaksinya ada dua macam yaitu *include* dan *extends*. *Include* adalah sebuah garis yang dimana menghubungkan satu *use case* membutuhkan *use case* lain, sedangkan *extends* adalah sebuah garis yang dimana menghubungkan satu *use case* ketika dibutuhkan atau tergantung dengan keadaan *use case* tersebut.
- f. *Generalization* adalah konsep pewarisan yang dimana aktor atau *use case* bisa mewariskan kepada aktor dan *use case* itu sendiri atau aktor ke *use case* dan *use case* ke aktor.



Gambar 2.1 Usecase
 (Sumber: Rekayasa Perangkat Lunak [24])

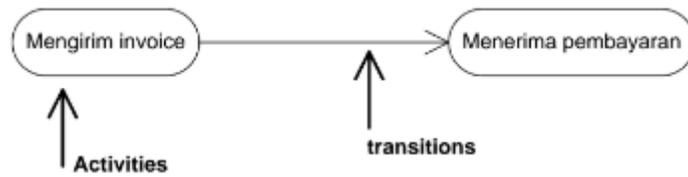
2. *Use case Scenario*

Use case scenario adalah suatu gambaran model yang berbentuk skenario sehingga nantinya ada beberapa kemungkinan-kemungkinan kejadian dalam menjalankan sebuah sistem tersebut. Kemungkinan-kemungkinan tersebut nantinya akan dipecah menjadi beberapa skenario. Di dalam proses sebuah sistem nantinya akan ada sekumpulan proses, proses inilah yang akan menjadi beberapa kemungkinan-kemungkinan yang kemudian bisa disebut dengan skenario *use case* [12].

3. *Activity Diagram*

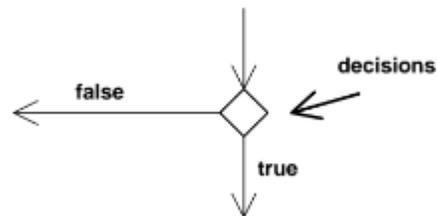
Activity diagram adalah *diagram* yang menggambarkan suatu aktivitas proses yang dilakukan dalam suatu sistem yang tentunya menghubungkan antara aktor dengan sistem. Di dalam *activity diagram* ada beberapa elemen yang digunakan yaitu berikut ini.

- a. *Activities* adalah elemen yang menggambarkan suatu aktivitas tertentu pada suatu sistem.
- b. *Transition* adalah elemen yang digunakan untuk melakukan transisi atau peralihan dari aktor ke sistem atau dari sistem ke aktor dan atau antar elemen.



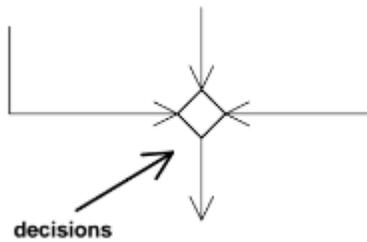
Gambar 2.2 Gambaran Transition
(Sumber: Rekayasa Perangkat Lunak [24])

- c. *Decision* adalah suatu elemen untuk melakukan percabangan logika di dalam suatu sistem. Sama seperti halnya di dalam *flowchart* ada juga yang dinamakan *decision*. Berikut ini gambarannya.



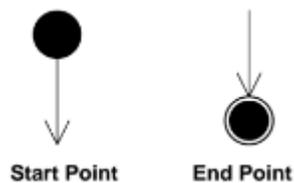
Gambar 2.3 Gambaran Decision
(Sumber: Rekayasa Perangkat Lunak [24])

- d. *Merge Point* adalah elemen yang digunakan untuk menggabungkan suatu proses di dalam sistem. Kalo diatas tadi decision itu untuk melakukan percabangan sedangkan *merge point* ini untuk melakukan penggabungan dari hasil percabangan tersebut.



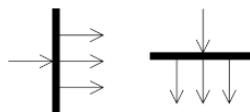
Gambar 2.4 Gambaran Merge Point
(Sumber: Rekayasa Perangkat Lunak [24])

- e. *Start Point and End Point* adalah elemen untuk menandakan dimulainya suatu aktivitas dan berakhirnya aktivitas dalam suatu sistem.



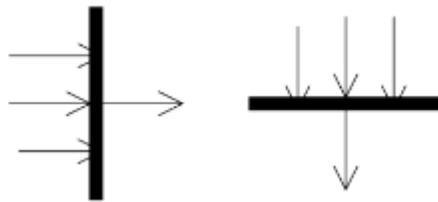
Gambar 2.5 Start Point and End Point
(Sumber: Rekayasa Perangkat Lunak [24])

- f. *Concurrency* adalah elemen untuk melakukan percabangan proses dalam suatu sistem. Berbeda halnya dengan *decision*, *decision* untuk melakukan percabangan logika tetapi kalau ini untuk melakukan percabangan proses.



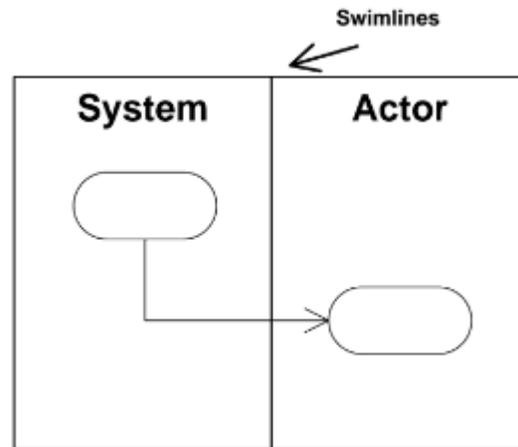
Gambar 2.6 Concurrency
(Sumber: Rekayasa Perangkat Lunak [24])

- g. *Synchronization* adalah elemen untuk melakukan menggabungkan hasil proses yang dipisahkan oleh *concurrency*.



Gambar 2.7 Synchronization
(Sumber: Rekayasa Perangkat Lunak [24])

- h. *Swimlines* adalah elemen digunakan untuk memisahkan aktor dengan sistem, aktor dengan aktor yang lain atau sistem dengan sistem yang lain.



Gambar 2.8 Swimlines
(Sumber: Rekayasa Perangkat Lunak [24])

4. *Class Diagram*

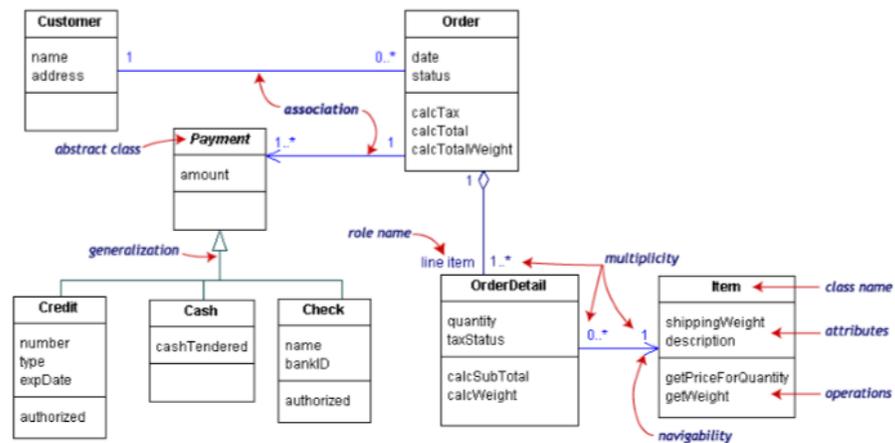
Class diagram adalah *diagram* yang menggambarkan struktur sistem secara spesifikasi yang didalamnya ada nama kelas, atribut dan metode. *Class diagram* ini jika diinstansiasi akan menghasilkan berupa objek karena berhubungan dengan konsep berorientasi objek.

Area pokok yang ada di dalam *class diagram* ada tiga yaitu nama kelas, atribut dan metoda. Atribut dan metoda ini memiliki beberapa sifat yaitu *private* atribut atau metode tidak dapat dipanggil dari luar *class* dapat dipanggil di dalam *class* yang sama. *Protected* yaitu ketika atribut dan metode hanya bisa dipanggil di dalam *class* yang sama dan bisa mewariskan ke anak-anaknya, dan ketika atribut dan metode memiliki sifat *public* maka semua atribut dan metode baik

yang ada di luar kelas atau di dalam *class* yang sama bisa dipanggil kapanpun dan oleh siapa saja.

Hubungan yang digunakan untuk menggambarkan *class diagram* adalah sebagai berikut :

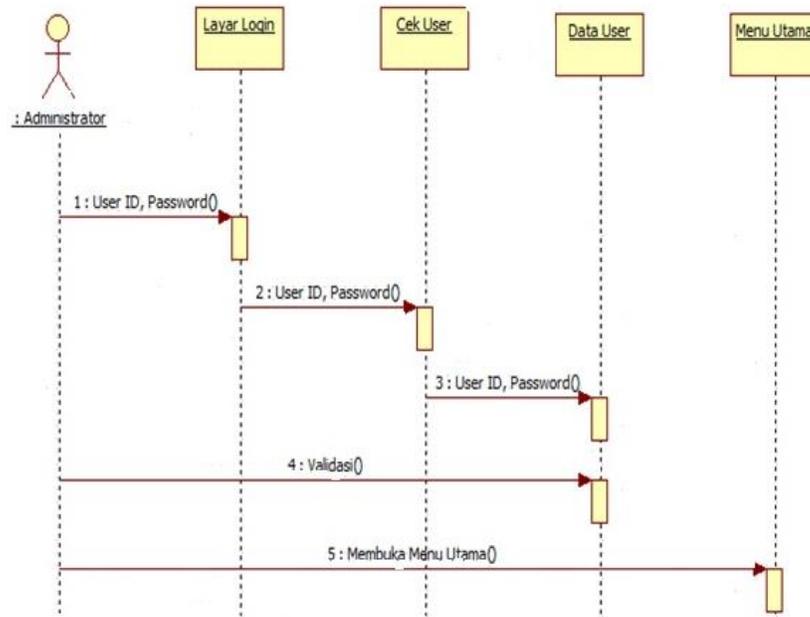
- a. Asosiasi adalah garis penghubung yang menghubungkan antar *class*. *Class* disini adalah *class* yang memiliki atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain.
- b. Agregasi adalah garis di dalam *class diagram* yang menghubungkan pernyataan bahwa *class* menjadi atribut di *class* yang lain.
- c. Pewarisan adalah *class* dapat diturunkan dari *class* lain dan bisa mewarisi semua atribut dan metoda *class* lain sehingga bisa menambahkan fungsionalitas baru menjadi disebut anak dari *class* yang diwarisinya.
- d. Hubungan dinamis adalah suatu rangkaian pesan yang dikirim dari *class* pada *class* lain.



Gambar 2.9 Contoh Class Diagram
(Sumber : Rekayasa Perangkat Lunak [24])

5. Sequence Diagram

Adalah *diagram* yang menggambarkan interaksi antar objek yang ada di dalam dan diluar sistem sehingga menjadi dinamis. Kegunaan *diagram sequence* ini adalah untuk menunjukkan rangkai pesan yang dikirim antar *object* atau antar sistem. Pesan ini digambarkan terhadap waktu. *Diagram* ini terdiri dari dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait. Untuk menghasilkan output tertentu dari sebuah sistem maka harus digambarkan melalui rangkaian langkah-langkah yang diawali dengan men-*trigger* aktivitas tertentu, kemudian apa saja proses yang berubah dan dihasilkan dari sistem yang dilakukan secara internal dan *output*.



Gambar 2.10 Contoh Sequence Diagram
 (Sumber : Rekayasa Perangkat Lunak [24])

2.2.12 Database

Di dalam membuat suatu sistem aplikasi pasti ada yang dinamakan dengan basis data atau biasa disebut dengan *database*. *Database* ini sangat penting keberadaannya dalam membuat suatu aplikasi yang berbasis teknologi komputer. Karena tanpa *database* data yang diperoleh dan diproses dimana akan disimpan. Semua data membutuhkan penyimpanan yang terstruktur yang disesuaikan dengan kebutuhan dari suatu sistem yang akan dibuat.

Basis data atau *database* adalah suatu kumpulan informasi dan data yang bisa disimpan secara sistematis dan terstruktur dan memiliki hubungan satu sama lainnya. Fungsi dari *database* ini adalah untuk mengelompokkan data dan informasi sehingga lebih mudah didapatkan, mempermudah menyimpan, mengakses data atau informasi, mencegah terjadinya duplikasi data dan bisa mampu meningkatkan kinerja aplikasi [13].

Manfaat dari penggunaan *database* adalah data atau informasi tidak akan mengalami redundansi atau pengulangan data yang sama. Kemudian integritas terjaga dengan baik dan menjaga keamanan data.

2.2.13 Javascript

Javascript merupakan bahasa pemrograman berbasis web digunakan untuk membuat tampilan halaman web menjadi lebih dinamis dan interaktif. Javascript termasuk ke dalam client side programming yaitu bahasa programan web yang di proses di sisi pengguna (user).[26]

2.2.14 Definisi Algoritma

Alur pemikiran dalam menyelesaikan suatu masalah yang dituangkan secara tertulis. Alur pemikiran mempunyai makna algoritma setiap orang dapat berbeda-beda. Tertulis artinya suatu algoritma agar dimengerti oleh diri sendiri maupun oleh

orang lain dapat dituangkan kedalam kata-kata/kalimat (bahasa natural), kode tertentu, maupun dengan gambar atau tabel tertentu.[25]

2.3 Perangkat Lunak yang Digunakan

Dalam penelitian ini digunakan beberapa perangkat lunak untuk membantu membuat program agar berjalan dengan baik. Berikut ini adalah perangkat lunak yang digunakan untuk membangun dan mengembangkan sistem persediaan bahan baku ini.

2.3.1 Pemrograman PHP

Pemrograman PHP adalah salah satu bahasa pemrograman yang digunakan untuk pengembangan web yang berbentuk *script server-side*. Pada awalnya PHP merupakan singkatan dari *Personal Home Page* yang digunakan untuk kebutuhan *website* pribadi. Tetapi dengan perkembangan sekarang nama PHP berubah menjadi singkatan dari PHP : *Hypertext Preprocessor* [14].

PHP disebut juga sebagai bahasa pemrograman *script* karena kode programnya bisa disisipkan ke dalam dokumen *HTML*. *Server side scripting* adalah bahasa pemrograman yang berjalan di *server*, PHP salah satunya. Untuk bahasa pemrograman *client side scripting* adalah bahasa pemrograman yang berjalan disisi *client* (*Web browser*). Contohnya seperti *Javascript*.

Di dalam membangun *platform* yang sedang penulis kerjakan adalah menggunakan pemrograman PHP karena beberapa keunggulannya. Keunggulan dari bahasa pemrograman PHP adalah mudah dipelajari, ringkas, *open source*, pengembangannya cepat, dan *maintenance*-nya mudah.

Bahasa pemrograman PHP itu sendiri memiliki perkembangan. Dari mulai PHP versi 5.0 sampai PHP versi 7.0 yang terbaru. Penulis menggunakan PHP versi 7.0 karena sudah banyak terintegrasi dengan *server* terbaru yaitu XAMPP yang mulai juga menggunakan PHP terbaru.

2.3.2 Cascading Style Sheet (CSS)

Cascading style sheet atau biasa disebut dengan CSS singkatnya adalah suatu rangkaian instruksi dalam membuat sebuah *website* yang menentukan bagaimana suatu text, *layout* dan bentuk dapat tampil di halaman *website* tersebut.

Kemudian pengertian CSS itu adalah salah satu bahasa desain *website* (*style sheet language*) yang digunakan untuk mengontrol tampilan dalam sebuah halaman *web* yang ditulis menggunakan penanda yaitu disebut *markup language*. CSS ini

umumnya digunakan untuk mendesain halaman HTML atau XHTML, seiring dengan perkembangan teknologi sekarang CSS bisa juga diaplikasikan untuk berbagai macam dokumen XML, seperti dalam sistem operasi *Android*, termasuk juga SVG.

Dalam teknisnya CSS ini memisahkan antara konten utama dengan tampilan seperti *layout*, *font* dan juga warna. Fungsi dari CSS ini adalah mendesain, merancang dan juga merubah bentuk serta merubah secara keseluruhan tampilan dari suatu *website* dengan menempelkannya pada *tag-tag html*, sehingga *tag-tag* tersebut bisa lebih fungsional dan juga lebih menarik. Cara kerja dari CSS ini juga sangat mudah, cukup menempelkan kelas yang telah kita buat dari file terpisah yang berformat *css* maka kita bisa menuliskannya pada *selector* dalam *HTML* yaitu *selector id* dan *class*.

CSS itu seiring dengan pesatnya perkembangan teknologi juga semakin diperbarui. CSS3 adalah kembangan yang lebih baru dari CSS. CSS3 ini sudah mendukung banyak efek yang bisa langsung digunakan dengan cepat. Efek ini tentu memudahkan untuk pengembang dalam hal mempercantik tampilan sebuah *website*. Efek yang dihasilkan dari CSS3 ini adalah seperti efek tulisan, gambar maupun objek. Kelebihan dari CSS3 ini juga bisa menghemat *bandwidth* website karena aksesnya lebih ringan dan dapat membuat *website* lebih interaktif.

2.3.3 Template Bootstrap

Program yang dibuat ini menggunakan *template* bootstrap sebagai tampilan halaman *website* yang dibuat agar lebih cepat dan lebih mudah karena *template*-nya dapat didownload secara gratis.

Bootstrap adalah *front-end framework* atau *framework* (kerangka kerja) yang digunakan untuk membuat tampilan *website* menjadi lebih mudah, lebih bagus, dan lebih menarik yang mengedepankan tampilan yang bisa dibuka dalam semua perangkat atau *device* baik itu *handphone*, *smartphone*, *laptop* atau komputer. Untuk mempermudah dan mempercepat pengembangan sebuah *website* maka bootstrap menyediakan HTML, CSS, dan *Javascript* yang siap pakai dan mudah digunakan untuk para pengembang *website*.

Secara teknisnya adalah kita sebagai pengembang tinggal menggunakan nama kelas yang sudah dibuat oleh *bootstrap* lalu kita tinggal menggunakan nya dalam *tag-tag* HTML. Penulis mengembangkan *platform* ini menggunakan *template* bawaannya bootstrap agar mudah digunakan dan juga agar *website* terlihat lebih menarik. Penulis menggunakan bootstrap versi 4.0 yang terbaru dan bisa di download dari laman resminya yaitu <https://getbootstrap.com/>.

2.3.4 XAMPP

Dalam membangun *platform* kursus *online* ini penulis menggunakan XAMPP sebagai *local server* untuk membuat *platform* berjalan dengan baik ketika dikembangkan dalam komputer penulis.

Menurut situs resmi XAMPP, XAMPP adalah *software open source* yang digunakan untuk membuat *server* sendiri pada komputer kita (istilahnya adalah *Localhost*) dapat di *download* secara gratis dari situs resminya yaitu <https://www.apachefriends.org/> dan *software* ini dapat berjalan di semua sistem operasi.

Penulis menggunakan XAMPP ini karena terdapat banyak kemudahan dalam membuat aplikasi sebelum nanti di *online* kan. Jadi penulis membuat dulu aplikasinya menggunakan *server localhost* pada komputer lalu kemudian di buat *online* dengan memiliki *domain* dan *hosting*.

Di dalam XAMPP ada beberapa hal penting yang harus diketahui fungsinya yaitu adalah *Htdocs*, *Phpmyadmin*, dan *Control Panel*.

1. *Htdocs* adalah salah satu *folder* yang ada dalam XAMPP yang berfungsi untuk menyimpan seluruh *file-file* dari aplikasi yang akan kita buat nantinya. Kemudian *folder Htdocs* ini akan ditampilkan di *browser* menggunakan bantuan *server localhost* sehingga aplikasi yang kita buat bisa tampil dalam *browser*.
2. *Phpmyadmin* adalah salah satu fitur yang ada dalam XAMPP yang berfungsi untuk mengolah data dalam *database website* yang akan kita kembangkan.
3. *Control Panel* adalah sama seperti Cpanel pada *website* yang sudah *online*, kalau di XAMPP *Control Panel* ini juga fungsinya kurang lebih sama yaitu mengatur *website*, *database*, *FTP*, *modul*, dan lain-lainnya.

2.3.5 Database MySQL

MySQL adalah suatu sistem manajemen pengelolaan *database* yang relasional dengan menggunakan *client-server* model. *Software* yang digunakan untuk membuat dan mengelola *database* adalah RDBMS menggunakan model relasional. Berikut ini adalah *software* yang terkait dengan RDBMS :

1. *Database* adalah sekumpulan data yang terstruktur yang mana bisa dengan mudah untuk menyimpan data dan mengambil data.
2. *Open Source, Software* RDBMS ini adalah *software* yang bebas digunakan oleh siapa saja, bebas menginstal dan bebas memodifikasinya.
3. *Client server model* adalah *software* RDBMS ini bersifat sebagai *client*, yang artinya berjalan dari sisi *client* agar bisa mengakses data dan terhubung dengan *server*.
4. SQL adalah salah satu *brand* yang terkenal dari *software* DBMS yang sudah menerapkan *client-server model*. Jadi singkatnya ketika *client* bisa berkomunikasi dengan *server* yaitu harus melewati bahasa yang spesifik dengan domain *Structured Query Language* (SQL).

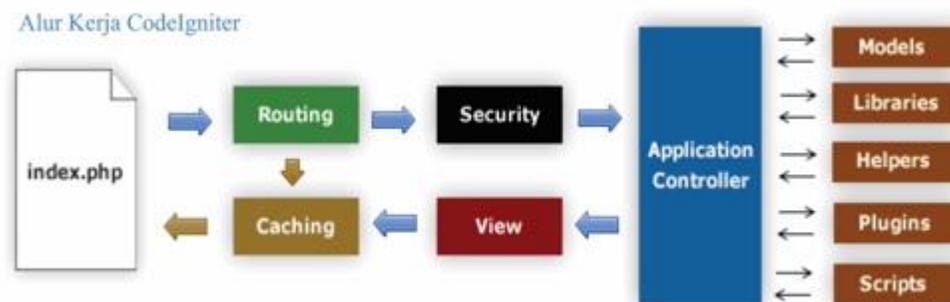
Dalam menginstruksikan *server* untuk menjalankan operasi tertentu maka data *query* meminta informasi yang spesifik dari *database* yang sudah ada, kemudian melakukan manipulasi data seperti menambah, menghapus, melakukan operasi bilangan, dan juga bisa mengubah data [15].

2.3.6 Framework Codeigniter

Framework Codeigniter adalah sebuah *framework* untuk mengembangkan *website* agar lebih cepat dan mudah karena sudah disediakan dokumentasinya dan *source code* nya untuk memudahkan para pengembang *website*.

Penulis menggunakan *framework* dalam membangun sistem ini karena ingin membuat suatu sistem yang cepat dan mudah dalam pembangunan dan pengembangan. Ketika menggunakan *framework* dalam hal ini *framework* PHP dapat dengan mudah melakukan *maintenance* karena sudah ada pola terstruktur guna membantu pengembang dalam membuat *website* [20].

Codeigniter ini adalah sebuah *framework* PHP yang dibangun untuk membuat suatu *website* agar lebih mudah digunakan oleh pengembang. Codeigniter ini sebuah *web application network* yang sifatnya *open source* atau terbuka untuk digunakan oleh siapapun guna membangun aplikasi PHP yang dinamis [20].



Gambar 2.11 Alur Kerja CodeIgniter

(Sumber : Menjadi Master PHP dengan Framework Codeigniter [20])

2.4 Perangkat Pendukung

Dibawah ini adalah beberapa perangkat pendukung yang digunakan dalam penelitian ini agar sistem yang dibangun bisa berjalan dengan baik.

2.4.1 Hardware

Hardware adalah perangkat komputer yang terdiri atas susunan komponen-komponen elektronik berbentuk fisik (berupa benda). Hardware atau perangkat keras adalah sebuah alat atau benda yang bisa dilihat, sentuh, pegang dan memiliki fungsi tertentu. Peralatan yang secara fisik terlihat dan bisa diraba atau dipegang [17].

2.4.2 Brainware

Brainware merupakan orang yang memiliki, sesuatu yang dapat membangun dan menjalankan sistem informasi. Istilah orang disini bukanlah orang sembarang tetapi dalam arti wujudnya yaitu seseorang yang memiliki kompetensi (ilmu pengetahuan dan keterampilan atau keahlian) Seperti Analis sistem, Programmer, Administrator database, Operator dan lain lain [18].

2.4.3 Jaringan Komputer

Merupakan sebuah jaringan yang biasanya terdiri dari dua atau lebih komputer yang saling berhubungan diantara satu dengan yang lainnya, dan saling berbagi sumber daya misalnya *CDROM*, *Printer*, *Pertukaran File*, atau memungkinkan untuk saling

berkomunikasi secara elektronik. Komputer yang terhubung tersebut dimungkinkan berhubungan dengan media kabel, saluran telepon, gelombang *radio*, *satelit* atau *infrared*. dan terdapat beberapa jenis jaringan komputer seperti LAN (*Local Area Network*), WAN (*Wide Area Network*) dan MAN (*Metropolitan Network*) [19].