

## **A. Gambaran umum perusahaan**

### **1. Profil Perusahaan**

#### **1.1 Sejarah Singkat**

Dr. Coffee adalah UKM atau usaha di bidang perkopian yang sedang berkembang. Beberapa usahanya antara lain produk kopi kemasan, kedai kopi, dan ekspor kopi biji asal Lampung. Dr coffee didirikan oleh M. Alghazali Qurtubi alumni Fakultas Pertanian Universitas Lampung pada tahun 2015. Nama Dr. Coffee tergabung dari kata Doctor dan Coffee, nama ini memiliki makna ahlinya kopi. Dr. Coffee juga memiliki kafe yang menjual aneka jenis kopi berkualitas seperti kopi yang dalam kemasan, minuman kopi ready to drink, es kopi susu, dan franchise coffee shop. Tidak hanya itu Dr. Coffee juga bekerja sama dengan para petani, dengan menginginkan hasil panen kopi yang berkualitas dan juga meningkatkan kesejahteraan para petani itu sendiri.

#### **1.2 Visi**

Dr. Coffee menjadi Perusahaan Unicorn pada tahun 2023.

#### **1.3 Misi**

Untuk mencapai visi diatas, Dr. Coffee mengemban misi sebagai berikut :

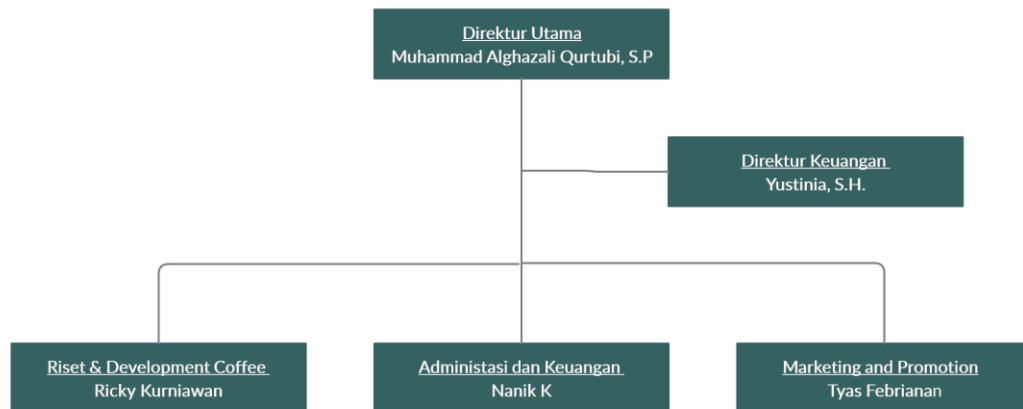
- Membuka 1000 CoffeeShop di seluruh Indonesia
- Memiliki 1000 partner Franchise.

### **2. Jenis Produk/Jasa**

Berikut ini adalah Jenis Produk yang ada di Dr. Coffee, yaitu :

- a. Produk kopi kemasan
- b. Produk kopi *Ready To Drink*
- c. Produk turunan kopi
- d. *Green Beans Fine Robusta*
- e. *Frenchise* Dr. Coffee dan kopi susu Jhon

### 3. Bagan Struktur Organisasi Perusahaan



### 4. Job Description

#### 4.1 Direktur Utama

Direktur Utama mempunyai tugas memimpin, merencanakan, mengkoordinir, melaksanakan dan mengawasi tugas para Direktur dan Kepala Bagian agar tercapai pelaksanaan operasional perusahaan secara teratur, terarah, terkendali dan terpadu.

#### 4.2 Direktur Keuangan

Direktur Keuangan bertanggung jawab atas:

- Bertanggung jawab terhadap kinerja keuangan sebuah perusahaan
- Bertanggung jawab membuat laporan keuangan perusahaan
- Mengawasi laporan keuangan perusahaan

#### 4.3 Kepala Bagian Administrasi dan Keuangan

Kepala Bagian Administrasi dan Keuangan memiliki tugas, yaitu:

- Melaksanakan, mengawasi, mengevaluasi, dan mengendalikan pencatatan transaksi bisnis perusahaan, Pelaporan Keuangan, Pelaporan Manajemen, dan segala masalah keuangan yang dibutuhkan.
- Menerima, mengevaluasi dan memverifikasi dokumen pendukung pembayaran (tagihan) dari Bagian terkait di Kantor.

- Mengevaluasi kebutuhan dana dan mengusahakan sumber pendanaan eksternal.

## **5. Kepala Bagian Riset dan Pengembangan Kopi**

Kepala Bagian Riset dan Pengembangan memiliki tugas, yaitu:

- Kepala Bagian Riset dan Pengembangan di suatu Dr. Coffee bertanggung jawab untuk segala aktivitas riset dan pengembangan kopi di Dr. Coffee.
- Kepala Bagian Riset dan Pengembangan juga bertanggung jawab untuk memastikan kualitas performansi dalam perusahaan sesuai dengan standar yang telah ditetapkan.
- Kepala Bagian Riset dan Pengembangan juga bertanggung jawab mengelola sejumlah dana tertentu yang telah dianggarkan untuk riset dan pengembangan kopi.

## **6. Kepala Bagian Marketing dan Promotion**

Kepala bagian Marketing dan Promotion memiliki tugas, yaitu:

- Memimpin bidang Marketing dan Promotion beserta seluruh sumber daya yang dimiliki.
- Mengarahkan bidang Marketing dan Promotion untuk mencapai produktivitas maksimal seefisien dan seefektif mungkin.
- Membangun interaksi dan kerjasama dengan konsumen/pelanggan dengan sebaik-baiknya.

## **7. Peralatan, dan Software Pendukung**

Peralatan pendukung yang digunakan Dr. Coffee dalam melaksanakan kegiatan bisnisnya, yaitu: komputer, printer, switch, kamera, modem dan jaringan internet, untuk software yang digunakan adalah aplikasi Pawon dan Microsoft Office

## **8. Proses Produk Perusahaan**

Dari hasil laporan proses pengolahan kopi dapat disimpulkan bahwa kopi siap dijual setelah melalui beberapa proses setelah panen diantaranya proses

roasting, yang dimana pada proses ini kopi akan dipanggang hingga tingkat kematangan tertentu. Lalu setelah kopi diroasting kopi akan di resting, resting adalah proses pendiaman kopi selama kurang lebih 24 jam selama proses ini kopi sudah dapat dikonsumsi, hanya saja rasanya belum begitu nikmat, selanjutnya kopi digrinding dengan tingkat kehalusan tertentu dan yang terakhir kopi *dipacking* untuk dijual.

## **9. Produk yang Sudah di Hasilkan**

Produk yang dihasilkan oleh Dr. Coffee antara lain:

- a. Produk kopi kemasan
- b. Produk kopi *Ready To Drink*
- c. Produk turunan kopi
- d. *Green Beans Fine Robusta*

## **10. Mitra Perusahaan dan Klien**

Rekan mitra yang bekerja sama dengan Dr. Coffee, yaitu: UKM, SCOPI Sustainable Coffee Platform of Indonesia (SCOPI) dan Bank Indonesia.

## **B. Uraian Landasan Teori**

Adapun uraian landasan teori tentang Sistem Jual Beli Investasi Kopi (iCoffee) berbasis Website di CV. DR. COFFEE JAYA RAYA adalah sebagai berikut:

### **1. Pengertian Aplikasi Web**

Pada awalnya aplikasi web dibangun dengan hanya menggunakan bahasa yang disebut HTML (*HyperText Markup Language*). Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML seperti PHP dan ASP pada skrip dan Applet pada objek. Aplikasi Website dapat dibagi menjadi dua jenis yaitu aplikasi website statis dan dinamis. Website statis dibentuk dengan menggunakan HTML. Kekurangan aplikasi seperti ini terletak pada keharusan untuk memelihara program secara terus menerus untuk mengikuti setiap perkembangan yang terjadi. Kelemahan ini diatasi oleh model aplikasi website dinamis. Pada aplikasi website dinamis, perubahan informasi dalam halaman web dilakukan

tanpa perubahan program tetapi melalui perubahan data. Sebagai implementasi, aplikasi website dapat dikoneksikan ke basis data sehingga perubahan informasi dapat dilakukan oleh operator dan tidak menjadi tanggung jawab dari webmaster. Arsitektur aplikasi website meliputi klien, web server, middleware dan basis data. Klien berinteraksi dengan web server. Secara internal, web server berkomunikasi dengan middleware dan middleware yang berkomunikasi dengan basis data. Contoh middleware adalah PHP dan ASP. Pada mekanisme aplikasi website dinamis, terjadi tambahan proses yaitu server menerjemahkan kode PHP menjadi kode HTML. Kode PHP yang diterjemahkan oleh mesin PHP yang akan diterima oleh klien. (Abdul Kadir, 2009)

## **2. Pengertian Data**

Data adalah sumber informasi yang bentuknya masih mentah. Jogianto menjelaskan dalam bukunya yang berjudul Analisis dan Desain Sistem Informasi bahwa data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Data dapat diperoleh dalam bentuk simbol-simbol karakter huruf, angka, gambar, suara, sinyal, dan lain sebagainya. Agar dapat digunakan, data harus diolah lebih lanjut. Hasil pengolahan terhadap data ini nantinya dapat menjadi informasi (Jogiyanto, 1990).

## **3. Basis Data**

Basis Data adalah sebagai berikut: “Basis data bisa dipahami sebagai suatu kumpulan data terhubung (interrelated data) yang disimpan dengan bersama-sama pada suatu media, tanpa mengatap satu dan yang lainnya atau tidak memerlukan suatu kerangkapan data (meskipun ada maka kerangkapan data itu harus seminimal mungkin dan terkontrol (*controlled redundancy*)), data disimpan dengan cara tertentu sehingganya mudah untuk digunakan dan ditampilkan kembali, data bisa digunakan satu atau bahkan lebih program-program aplikasi secara optimal, data disimpan tanpa ketergantungan dengan program yang akan menggunakannya, data disimpan sedemikian rupa

sehingga proses pengambilan, penambahan, dan modifikasi data dapat dilakukan dengan sangat mudah dan terkontrol.” (Edhy Sutanta, 2014).

#### **4. Pengertian MariaDB**

MariaDB adalah salah satu server basis data paling populer di dunia. MariaDB dibuat oleh pengembang asli MySQL dan dijamin tetap open source. MariaDB mengubah data menjadi informasi terstruktur dalam beragam aplikasi, mulai dari perbankan hingga situs website. Database ini digunakan karena cepat, kuat, dan terukur, dengan ekosistem mesin yang kaya dengan penyimpanan, plugin, dan banyak alat lain sehingga lebih fleksibel untuk berbagai macam kasus penggunaan database (AboutMariaDB, 2019).

##### **a. Kelebihan Menggunakan MariaDB**

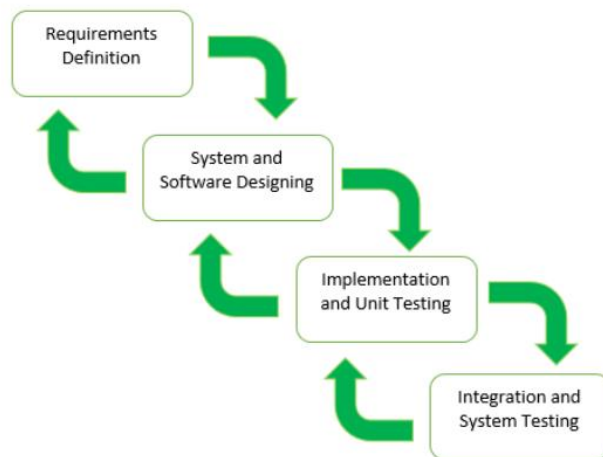
Beberapa kelebihan menggunakan MariaDB antara lain: Akses gratis atau sering disebut juga dengan Open source, memiliki storage engine yang lebih besar dibanding MySQL, semua library yang ada di MySQL dapat bekerja pada MariaDB, serta dapat bekerja dengan banyak tools klien.

##### **b. Kekurangan Menggunakan MariaDB**

Pada dasarnya, tidak ada kekurangan yang berarti pada database MariaDB, hanya saja ada beberapa sistem operasi dan hosting yang tidak mendukung database server MariaDB.

#### **5. Metode *Waterfall***

Metode Waterfall ini merupakan metode pengembangan sistem yang dalam fase pengerjaannya dilakukan secara bertahap dan berurutan. Jadi, jika langkah pertama belum dikerjakan maka tidak dapat mengerjakan langkah selanjutnya. Metode Waterfall terdiri dari beberapa langkah pengerjaan yaitu analisa kebutuhan, design system, coding and testing, pengujian dan penerapan, dan pemeliharaan (Al-Bahra, 2006). Metode Waterfall ditunjukkan pada Gambar 2 berikut.



**Gambar 2. Metode Waterfall (Ladjamudin, 2006)**

Tahapan-tahapan model proses waterfall dijelaskan berikut (Sommerville, 2011).

**a. Requirements analysis and definition**

Mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh program yang akan dibangun. Fase ini harus dikerjakan secara lengkap untuk bisa menghasilkan desain yang lengkap.

**b. System and software design**

Desain dikerjakan setelah kebutuhan selesai dikumpulkan secara lengkap.

**c. Implementation and unit testing**

Desain program diterjemahkan ke dalam kode-kode dengan menggunakan bahasa pemrograman yang sudah ditentukan. Program yang dibangun langsung diuji baik secara unit.

**d. Integration and system testing**

Penyatuan unit-unit program kemudian diuji secara keseluruhan (system testing).

**e. Operation and maintenance**

Mengoperasikan program dilingkungannya dan melakukan pemeliharaan, seperti penyesuaian atau perubahan karena adaptasi dengan situasi sebenarnya.

## 6. Framework Laravel

Pengertian framework menurut Naista adalah suatu struktur konseptual dasar yang digunakan untuk memecahkan atau menangani suatu masalah yang kompleks. Singkatnya, framework adalah wadah atau kerangka kerja dari sebuah website yang akan dibangun. Dengan menggunakan kerangka tersebut waktu yang digunakan dalam membuat website lebih singkat dan memudahkan dalam melakukan perbaikan.

Salah satu framework yang banyak digunakan oleh programmer adalah framework laravel. Laravel adalah framework berbasis PHP yang sifatnya open source, dan menggunakan konsep model – view – controller. Laravel berada di bawah lisensi MIT License dengan menggunakan Github sebagai tempat berbagi code menjalankannya (Naista, 2017).

## 7. Pengertian UML

(Rosa dan Shalahuddin, 2013) menjelaskan tentang pengertian UML adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML menyediakan serangkaian gambar dan diagram yang sangat baik. Beberapa diagram memfokuskan diri pada ketangguhan teori object oriented dan sebagian lagi memfokuskan pada detail rancangan dan konstruksi. Semua dimaksudkan sebagai sarana komunikasi antar team programmer maupun dengan pengguna.

UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh: Grady Booch OOD (*Object-Oriented Design*), Jim Rumbaugh OMT (*Object Modeling Technique*), dan Ivar Jacobson OOSE (*Object-Oriented Software Engineering*). Metode dari ketiganya digabungkan dengan cara membuang elemen yang tidak praktis ditambah dengan elemen-elemen dari metode lain yang lebih efektif dan elemen-elemen baru yang belum ada pada metode terdahulu sehingga UML lebih ekspresif dan seragam daripada metode lainnya. Pendekatan-pendekatan yang dipakai dalam analisa berorientasi objek antara lain:





1. Pendekatan top down yaitu memecahkan masalah kedalam bagian-bagian terkecil atau per level sehingga mudah untuk diselesaikan.
2. Pendekatan modul yaitu membagi sistem ke dalam modul – modul yang dapat beroperasi tanpa ketergantungan.
3. Penggunaan alat-alat bantu dalam bentuk grafik dan teks sehingga mudah untuk mengerti serta dikoreksi apabila terjadi perubahan.
4. Pendekatan dalam analisa berorientasi objek dilengkapi dengan alat–alat dan teknik-teknik yang dibutuhkan dalam pengembangan sistem, sehingga hasil akhir dari sistem yang dikembangkan akan didapatkan sistem yang dapat terdefinisi dengan baik dan jelas. Maka analisa berorientasi objek akan dilengkapi dengan alat dan tehnik di dalam mengembangkan sistem.




#### a. Desain Kebutuhan

Pada tahap desain proses yang dilakukan yaitu mendesain usecase diagram. Triandini dalam bukunya yang berjudul Step by Step Desain Proyek Menggunakan UML menjelaskan bahwa, diagram usecase menunjukkan tiga aspek dari system yaitu actor, usecase dan system atau sub system boundary. Actor mewakili peran orang, system yang lain atau alat ketika berkomunikasi dengan usecase (Triandini, 2012).

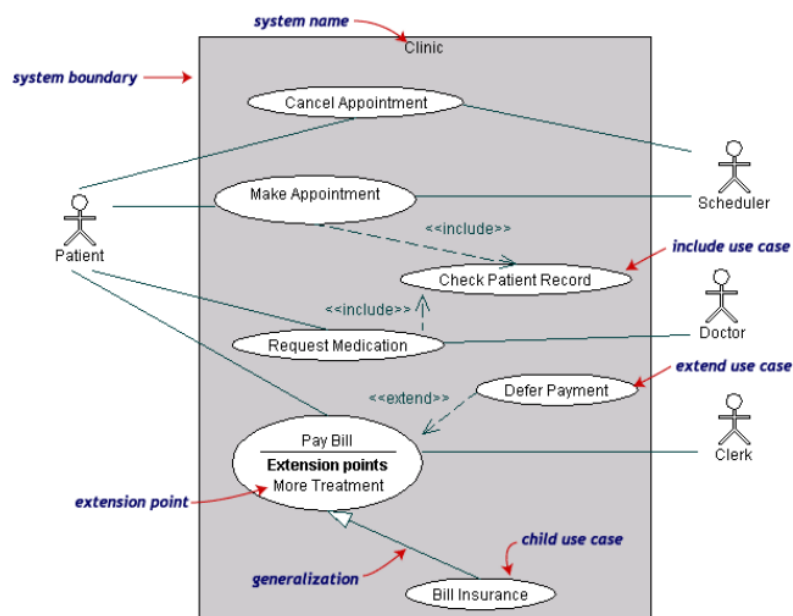
**Tabel 1. Simbol Usecase Diagram (Triandini, 2012)**

No	Gambar	Nama	Keterangan
1		<i>Use case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
2		<i>Actor</i>	Menjelaskan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .

3	————	<i>Association</i>	Menghubungkan antara objek satu dengan objek lainnya.
4	————>	<i>Directed Association</i>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, <i>directed association</i> biasanya juga disertai dengan <i>multiplicity</i> .
5	————>	<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ) (umum-khusus).
6	----->	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
7	-----> I	<i>Include</i>	Menjelaskan bahwa <i>use case</i> sumber secara eksplisit.
8	<----- E	<i>Extend</i>	Menjelaskan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i>

			sumber pada suatu titik yang diberikan.
9		<i>System</i>	Menjelaskan paket yang menampilkan sistem secara terbatas.
10		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
11		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Contoh Usecase Diagram



Gambar 3. Contoh usecase diagram ( Dharwiyanti dan Romi, 2003).

## b. Desain Proses


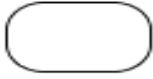

### 1. Activity Diagram

Triandini dalam bukunya yang berjudul *Step by Step Desain Proyek Menggunakan UML* menjelaskan bahwa *activity diagram* memodelkan workflow proses bisnis dan urutan aktivitas dalam sebuah proses. Diagram ini sangat mirip dengan flowchart karena memodelkan workflow dari satu aktivitas ke aktivitas lainnya atau dari aktivitas ke status. Menguntungkan untuk membuat *activity diagram* pada awal pemodelan proses untuk membantu memahami keseluruhan proses. *Activity diagram* juga bermanfaat untuk menggambarkan interaksi antara beberapa usecase (Triandini, 2012).

Definisi *activity diagram*:

- Menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses.
- Dipakai pada business modeling untuk memperlihatkan urutan aktifitas proses bisnis.
- Struktur diagram ini mirip flowchart atau Data Flow Diagram pada perancangan terstruktur.
- Sangat bermanfaat apabila kita membuat diagram ini terlebih dahulu dalam memodelkan sebuah proses untuk membantu memahami proses secara keseluruhan.
- *Activity diagram* dibuat berdasarkan sebuah atau beberapa usecase pada usecase diagram.

**Tabel 2. Simbol *activity diagram* (Triandini, 2012)**

No	Gambar	Nama	Keterangan
1		<i>Activity</i>	Menandakan sebuah aktivitas. Bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi.
3		<i>Initial Node</i>	Titik awal, untuk memulai aktivitas.


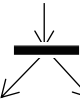
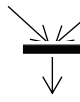
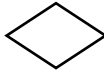
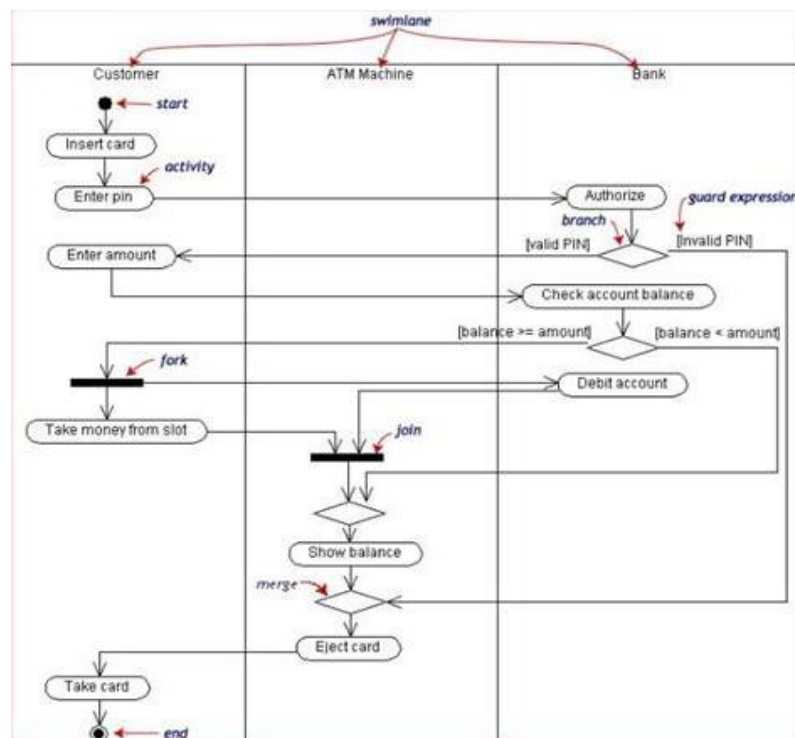
No	Gambar	Nama	Keterangan
4		<i>Activity Final Node</i>	Objek dibentuk dan diselesaikan atau di akhiri.
5		<i>Fork</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.
6		<i>Join</i>	Beberapa aliran yang pada tahap tertentu berubah menjadi satu aliran.
7		<i>Decision</i>	Pilihan untuk mengambil keputusan.

Diagram aktivitas atau *activity diagram* menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu di perhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahudin 2013).



**Gambar 4. Contoh *activity diagram* (Dharwiyanti dan Romi, 2003).**

### c. Desain Data

Pada tahap desain data yang dilakukan yaitu mendesain class diagram. Menurut Munawar dalam bukunya yang berjudul *Pemodelan Visual dengan UML*, class adalah deskripsi kelompok objek-objek dengan properti, perilaku dan relasi yang sama. Class diagram bisa memberikan pandangan global atas sebuah sistem. Hal tersebut tercermin dari class-class yang ada dan relasi satu dengan lainnya (Munawar, 2005).

Menurut Triandini (2012), class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (attribute/property) suatu sistem sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (method/function). Class Diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain.

*Class* memiliki tiga area pokok:

1. Nama (Stereotype)
2. Attribute
3. Metoda (Operation)

*Attribute* dan *operation* dapat memiliki salah satu sifat berikut:

1. Private, tidak dapat dipanggil dari luar class yang bersangkutan.
2. Protected, hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya.
3. Public, dapat dipanggil oleh siapa saja.





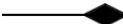

Hubungan Antar Class:


1. 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.
2. Agregasi, yaitu hubungan yang menyatakan bagian (“terdiri atas.”).
3. Pewarisan, yaitu hubungan hirarki 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.

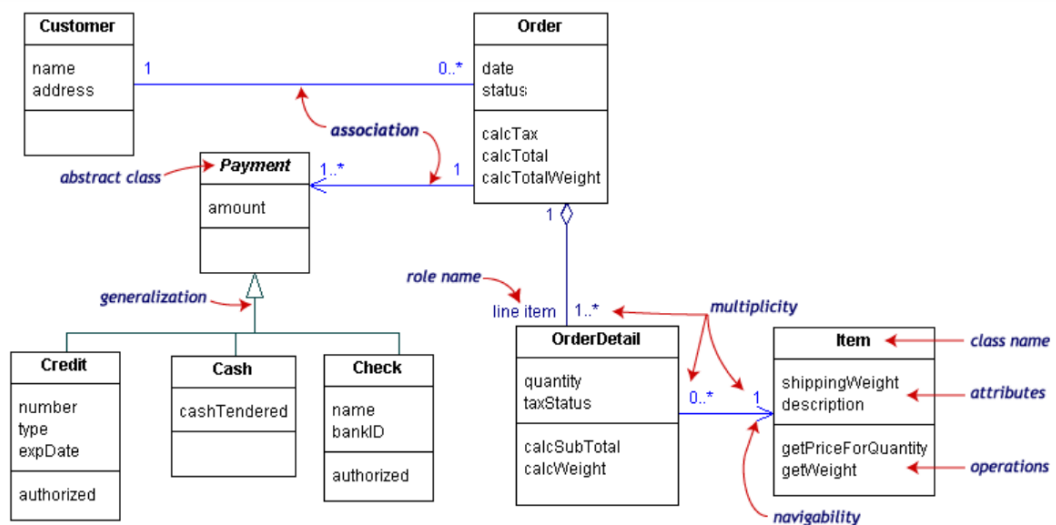
4. Hubungan dinamis, yaitu rangkaian pesan (message) yang dipassing dari satu class kepada class lain. Hubungan dinamis dapat digambarkan dengan menggunakan sequence diagram yang akan dijelaskan kemudian.

**Tabel 3. Simbol class diagram (Triandini, 2012)**

No	Gambar	Nama	Keterangan
1		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2		<i>Association</i> <i>Association</i>	Menghubungkan antara objek satu dengan objek lainnya.
3		<i>Directed</i> <i>Association</i>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, <i>directed association</i> biasanya juga disertai dengan <i>multiplicity</i> .
4		<i>Aggregation</i>	Relasi antar kelas dengan makna semua-bagian ( <i>Whole-part</i> ).
5		<i>Composition</i>	Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus merupakan bagian dari <i>class</i> yang lain, maka <i>class</i> tersebut memiliki relasi <i>composition</i> terhadap <i>class</i> tempat dia bergantung tersebut.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang

No	Gambar	Nama	Keterangan
			bergantung pada elemen yang tidak mandiri ( <i>independent</i> ).
7		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ) (umum-khusus).

Berikut ini merupakan contoh class diagram.



**Gambar 6. Contoh class diagram (Dharwiyanti dan Romi, 2003).**

### C. Analisis Proses Bisnis yang Berjalan

Dr. Coffee memiliki beberapa kelompok tani binaan yang mensuplai kebutuhan biji kopi mereka. Proses bisnis biji kopi dimulai dari pemupukan sampai dengan pasca panen membutuhkan waktu 1 tahun dengan musim panen setahun sekali. Biasanya, petani mengalami keterbatasan modal untuk mengolah biji kopi mereka dan alhasil nilai jual biji kopi panen mereka tidak maksimal. Siklus ini terjadi berulang kali lantaran tidak adanya pemodal untuk membiayai ongkos produksi mereka.